

IBM WebSphere Virtual Enterprise V7.0.0 documentation

View the latest documentation for WebSphere Virtual Enterprise. This information applies to V7.0.0 and to all subsequent releases and modifications until otherwise indicated in new editions.

Getting started

Product overview

➔ System requirements

Installing WebSphere Virtual Enterprise

Migrating to V7.0.0

Preparing the hosting environment for dynamic operations

Common tasks

Deploying application editions without loss of service

Managing the deployment environment

Monitoring the deployment environment

Troubleshooting and support

Troubleshooting WebSphere Virtual Enterprise

➔ Latest fixes

➔ WebSphere Virtual Enterprise support portal

➔ IBM Software Support home

More information

➔ WebSphere Virtual Enterprise V7.0.0 animations

➔ WebSphere Virtual Enterprise V7.0.0 library

➔ Education Assistant

➔ WebSphere Virtual Enterprise Wiki

➔ WebSphere Virtual Enterprise user community

➔ Redbooks home

Product overview

WebSphere® Virtual Enterprise extends WebSphere Application Server Network Deployment, using service level management to provide an enhanced quality of service in dynamic operations and extended manageability.

What is WebSphere Virtual Enterprise?

WebSphere Virtual Enterprise provides application server virtualization, resource management, and a host of advanced operational facilities, such as performance visualization, health management, and application edition management. This combination of capabilities is sometimes collectively referred to as *dynamic operations*, which is the core functionality that provides application virtualization, application server virtualization, and application infrastructure virtualization. Use Virtual Enterprise to enhance operational efficiency by deploying dynamic operations, service high-volume transactional workloads with linear scalability and high availability, and manage large scale, continuously available application server environment.

Application edition management provides the ability to roll out new versions of applications without experiencing downtime for a maintenance window. Health management provides the ability for you to specify conditions to watch for and corrective actions to take when the conditions are observed. Both of these capabilities improve the resiliency and availability of applications that are managed by Virtual Enterprise. Dynamic workload management allows an application's footprint to change automatically during runtime based on incoming demand for that application. As a result, multiple applications that are sharing a common resource pool can provision the average usage and share a set of resources to handle peak usage scenarios.

WebSphere Virtual Enterprise does not change any Programming Interfaces (APIs) or binary files of WebSphere Application Server Network Deployment. Therefore, if an application works correctly with WebSphere Application Server Network Deployment, it will also work correctly with WebSphere Virtual Enterprise. You might be limited in the use of some WebSphere Virtual Enterprise features with particular applications, however, but any limitations depend on the design, performance, and quality of service requirements for the application. Most limitations can be overcome by tuning your topology or reconfiguring your applications.

For example, assume that you cannot stop an application safely without disrupting service, and you cannot tolerate a service disruption during peak hours. Although WebSphere Virtual Enterprise quiesces your application servers before stopping them, sessions can still be lost if the application does not have sessions that can be serialized and if sessions remain after the maximum quiesce period has timed out. In this case, you can work around this issue by not using dynamic clusters in automatic mode or by tuning the dynamic clusters appropriately.

z/OS Restriction: The product does not support Session Initiation Protocol (SIP) features on the z/OS® operating system.

[What is new in this release](#)

IBM WebSphere Virtual Enterprise Version 7.0 provides features that help lower your total cost of ownership while increasing your quality of service by providing value in the areas of high availability, performance management, application infrastructure virtualization, and cloud computing management.

z/OS [z/OS considerations](#)

Before you install and customize the product for z/OS product, consider certain updates and considerations.

[Virtualization and WebSphere Virtual Enterprise](#)

By configuring application infrastructure virtualization in WebSphere Virtual Enterprise, you can pool together resources that are normally kept separate to accommodate the fluctuations of workload in your environment and increase the quality of service. You can also use application infrastructure virtualization with hardware virtualization capabilities that are provided by the physical hardware on which WebSphere Virtual Enterprise is hosted.

[Dynamic operations](#)

WebSphere Virtual Enterprise contains functionality to monitor the virtualized application server environment and make workload management optimizations or recommendations that are based on the data observed, increasing the quality of service. This capability is referred to as *dynamic operations*.

[Elasticity mode](#)

Use elasticity mode to add logic that causes the application placement controller to minimize the number of nodes that are used, as well as remove nodes that are not needed, while still meeting service policy goals. Additionally, you can use elasticity mode to add logic so that when the controller recognizes a particular dynamic cluster is not meeting service policies and has started all possible servers, the controller adds a node.

[Extended manageability](#)

Use extended manageability to configure the dynamic operation environment and the visualization capability to understand the operational state of the environment.

[Supported middleware server types](#)

You can create representations of servers that run other middleware software in the administrative console. After you create these representations, you can then manage the servers from the administrative console.

Middleware nodes and servers

The term *middleware server* refers to a server on any middleware platform. Middleware servers include the following types: WebSphere Application Server or WebSphere Virtual Enterprise, Apache Tomcat servers, JBoss servers, BEA WebLogic servers, PHP servers, and so on.

Middleware agent

The middleware agent is a server process that is used to manage servers that are not running WebSphere Virtual Enterprise.

Related tasks:

[Installing and customizing the product](#)

[Migrating WebSphere Virtual Enterprise](#)

[Preparing the hosting environment for dynamic operations](#)

[Deploying and managing application editions without loss of service](#)

[Managing the WebSphere Virtual Enterprise environment](#)

[Monitoring operations](#)

[Configuring WebSphere Virtual Enterprise to work with other IBM products](#)

Related reference:

[Reference](#)

Related information:

[IBM WebSphere Virtual Enterprise Version 7.0](#)

[Site map](#)

What is new in this release

IBM® WebSphere® Virtual Enterprise Version 7.0 provides features that help lower your total cost of ownership while increasing your quality of service by providing value in the areas of high availability, performance management, application infrastructure virtualization, and cloud computing management.

Version 7.0

Usability enhancements

- Centralized logging: You can use centralized logging to easily enable mustGather traces, perform a per-request trace analysis, and follow the flow of a request through both the ODR and application server tiers. You can enable tracing based on the type of problem that you experience, for example, 503 HTTP response codes.
- Maintenance mode for on demand routers (ODRs): Virtual Enterprise Version 7.0 supports both node and server maintenance modes for ODRs. You can use maintenance mode to perform node-level maintenance mode for the ODR nodes just as you could previously perform node-level maintenance mode for application server nodes. Similarly, you can use server maintenance mode for your ODRs just as you can for application servers. See [Setting maintenance mode](#) for more information.
- Simple Network Management Protocol (SNMP) trap generation: Virtual Enterprise can generate SNMP Version 1 traps for health events and dynamic cluster start and stop application server events.
- Support for fine-grained security: The panels in the administrative console are filtered to appropriately govern access to the product resources based on the administrator's roles.

High availability enhancements

- Custom Performance Monitoring Infrastructure (PMI) metrics: You can use a custom health policy to configure any PMI metric. In releases prior to Version 7.0, the number of PMI metrics available through health management was limited to the metrics that are provided in the administrative console.
- Excessive memory health policy enhancements: In releases prior to Version 7.0, the excessive memory health policy calculated the amount of heap usage used by a Java™ virtual machine (JVM) by periodically sampling the amount of free heap. You can set the **usedHeapModule** custom property so that only the amount of free heap after a full garbage collection is sampled. See [Health controller custom properties](#) for more information.
- ODR dynamic clusters: You can create dynamic clusters of ODRs, which means that the application placement controller selects the best node on which to start the minimum number of ODRs. If an ODR stops for any reason, the application placement controller will then start a new instance. See [Creating a dynamic cluster of ODRs](#) for more information.
- ODR health monitoring: Health monitoring is now extended to the ODR tier, which means you can set an ODR as the target of a health policies, thus detecting and automatically correcting ODR health issues. As a result, your environment is more highly available.
- ODR OutOfMemory prevention: You can use the **ODR.heapUsage.max**, **ODR.heapUsage.algorithm**, **ODR.heapUsage.errorcode** custom properties to prevent the ODR from running out of memory when the number of concurrent requests sent to the ODR increases. See [On demand router system and custom properties](#) for more information.
- Segregating HTTP traffic by ODR cluster: Virtual Enterprise provides a service to automatically regenerate and propagate the web server plug-in configuration file each time a change is made, which should affect how the plug-in routes work requests. When you use this service, the web server plug-in routes traffic by application, deployment target, or node group to a specific ODR cluster. See [Segregating HTTP traffic by ODR clusters](#) for more information.

Performance management enhancements

- Application placement parallel start: You can use the **apcConcurrentStartSize** custom property to define the maximum number of server instances the application placement controller should attempt to start concurrently on a particular node. See [Application placement custom properties](#) for more information.
- Elasticity mode: Elasticity mode adds logic that causes the application placement controller to minimize the number of nodes that are used while still meeting service policy goals.
- Multi-cell performance management: Configure and manage multi-cell performance in your environment to avoid overprovisioning resources, such as CPU and memory utilization. See [Configuring multi-cell performance management](#) for more information.
- Rate-based autonomic request flow manager (ARFM): ARFM uses a rate-based algorithm that results in more consistent loading and protecting of application server resources by ARFM.
- Service policies without response time goals: You can set the **disableResponseTimeGoals** custom property to configure service policies without response time goals when multiple service policies are configured, and the response time goals cannot be accurately determined for any of the service policies. See [Service policy custom properties](#) for more information.
- Support for AIX® Workload Partitioning (WPARs): AIX supports multiple levels of virtualization. Each physical machine can contain multiple Logical Partitions (LPARs), and each LPAR can contain multiple WPARs. Version 7.0 now supports sharing CPU between WPARs.
- Support for Solaris 10 Zones with shared CPU: You can install and deploy the product on Solaris 10 Zones in non-dedicated mode in which CPU and memory are shared between zones.
- Quiesce dynamic cluster instances: You can use the **quiesceTimeOutMS** custom property to set the quiesce timeout value for dynamic cluster instances. See [Dynamic cluster custom properties](#) for more information.

Parent topic: [Product overview](#)

z/OS considerations

Before you install and customize the product for z/OS® product, consider certain updates and considerations.

Considerations for WebSphere Virtual Enterprise for z/OS

Considerations for using WebSphere® Virtual Enterprise for z/OS are as follows:

Table 1. z/OS installation and customization

Installing and customizing the product for z/OS	See Customizing WebSphere Virtual Enterprise for z/OS for an overview of the installation and customization process.
Customizing your environment	See Installing the WebSphere Customization Toolbox .

Table 2. Middleware node support on z/OS

You cannot federate middleware nodes to a z/OS deployment manager	Federating middleware agents from distributed platforms into a z/OS deployment manager is not supported.
--	--

Table 3. Session Initiation Protocol (SIP) support on z/OS

SIP is not supported on the z/OS operating system	SIP is not supported on the z/OS operating system for WebSphere Virtual Enterprise .
--	--

Parent topic: [Product overview](#)

Related information:
[Product overview](#)

Virtualization and WebSphere Virtual Enterprise

By configuring application infrastructure virtualization in WebSphere® Virtual Enterprise, you can pool together resources that are normally kept separate to accommodate the fluctuations of workload in your environment and increase the quality of service. You can also use application infrastructure virtualization with hardware virtualization capabilities that are provided by the physical hardware on which WebSphere Virtual Enterprise is hosted.

Application infrastructure virtualization

With *application infrastructure virtualization*, you can separate applications from the physical infrastructure on which they are hosted. Workloads can then be dynamically placed and migrated across a pool of application server resources, which allows the infrastructure to dynamically adapt and respond to business needs. Requests are prioritized and intelligently routed to respond to the most critical applications and users.

Typically, applications and Java™ Platform, Enterprise Edition (Java EE) resources are statically bound to a specific server. Some of these applications might experience periodic increases in load that last a short time. The most costly time for an application to become unavailable is during a period of high demand. You must build your IT infrastructures to be able to accommodate these peaks. During the majority of time when your systems experience normal load, a large percentage of your computing capacity might go unused, making inefficient use of IT investments.

In a static environment, applications often span multiple enterprise archives (.ear files), and are not comprehensively defined so that the application can be portable between environments. Statically deployed applications rely on information that is found in the server to which they are deployed.

In the virtualized dynamic operations environment of WebSphere Virtual Enterprise, the static relationship is replaced with a dynamic relationship with looser coupling of applications or resources and server instances. Instead of statically binding applications to servers or clusters, you deploy applications to dynamic clusters, which are application deployment targets that can expand and contract depending on the workload in your environment.

After you deploy your applications to be mobile by using dynamic clusters, the placement of the applications is determined by the operational policies that you define. Autonomic managers control the placement of the server instances and how workload is routed for each application. If workload increases for a specific application, the number of server instances for the dynamic cluster that is hosting the application can increase, using available resources from other applications that are not experiencing increased workload.

Application infrastructure virtualization benefits:

- **Improved management of software and applications:** Management processes become more repeatable and less error-prone by using automated services and operational policies.
- **Allocation of software resources:** Dynamic reallocation of resources can occur based on shifting distributions of load among applications.
- **Increased number of applications:** More applications can run in a virtualized application environment than in a static configuration.
- **Reduced configuration complexity:** Loosened coupling between applications and the application server instances reduces the overall complexity and provides for a better, more usable environment.

Application infrastructure virtualization example

Figure 1. Application infrastructure virtualization in a WebSphere Virtual Enterprise environment. You deploy an application to a dynamic cluster that has a specified membership policy or node group. You do not deploy your applications to specific application servers. Instead, the application placement controller starts application server instances for your dynamic cluster based on the settings that you chose for the dynamic cluster.

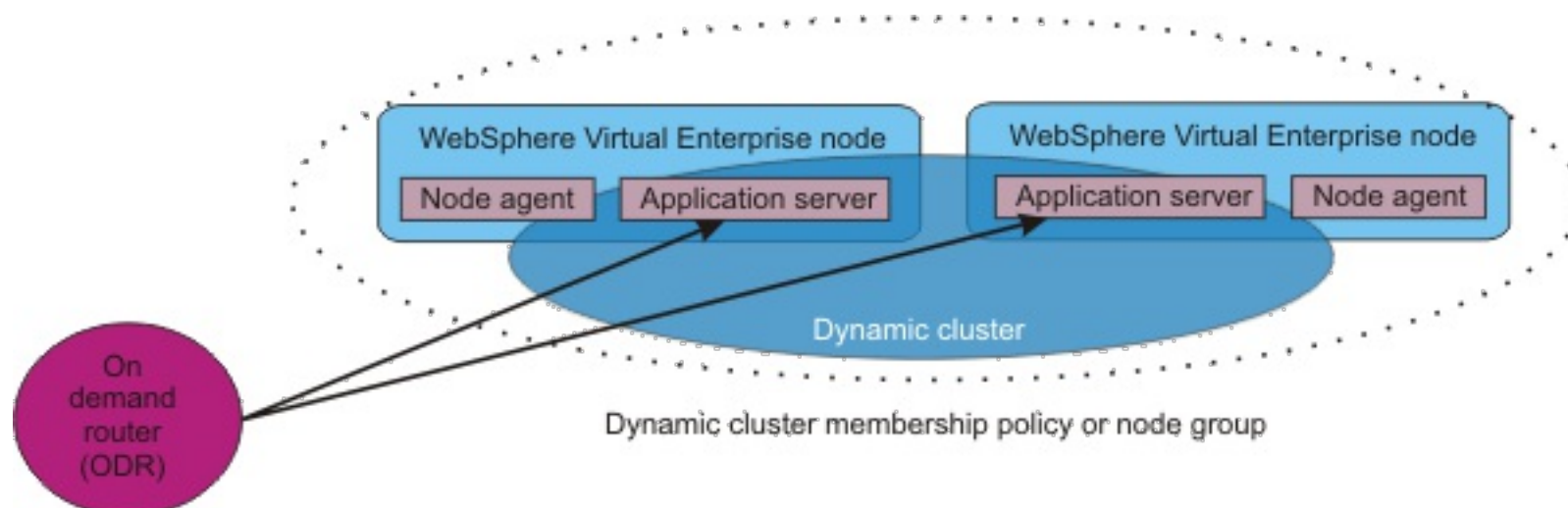
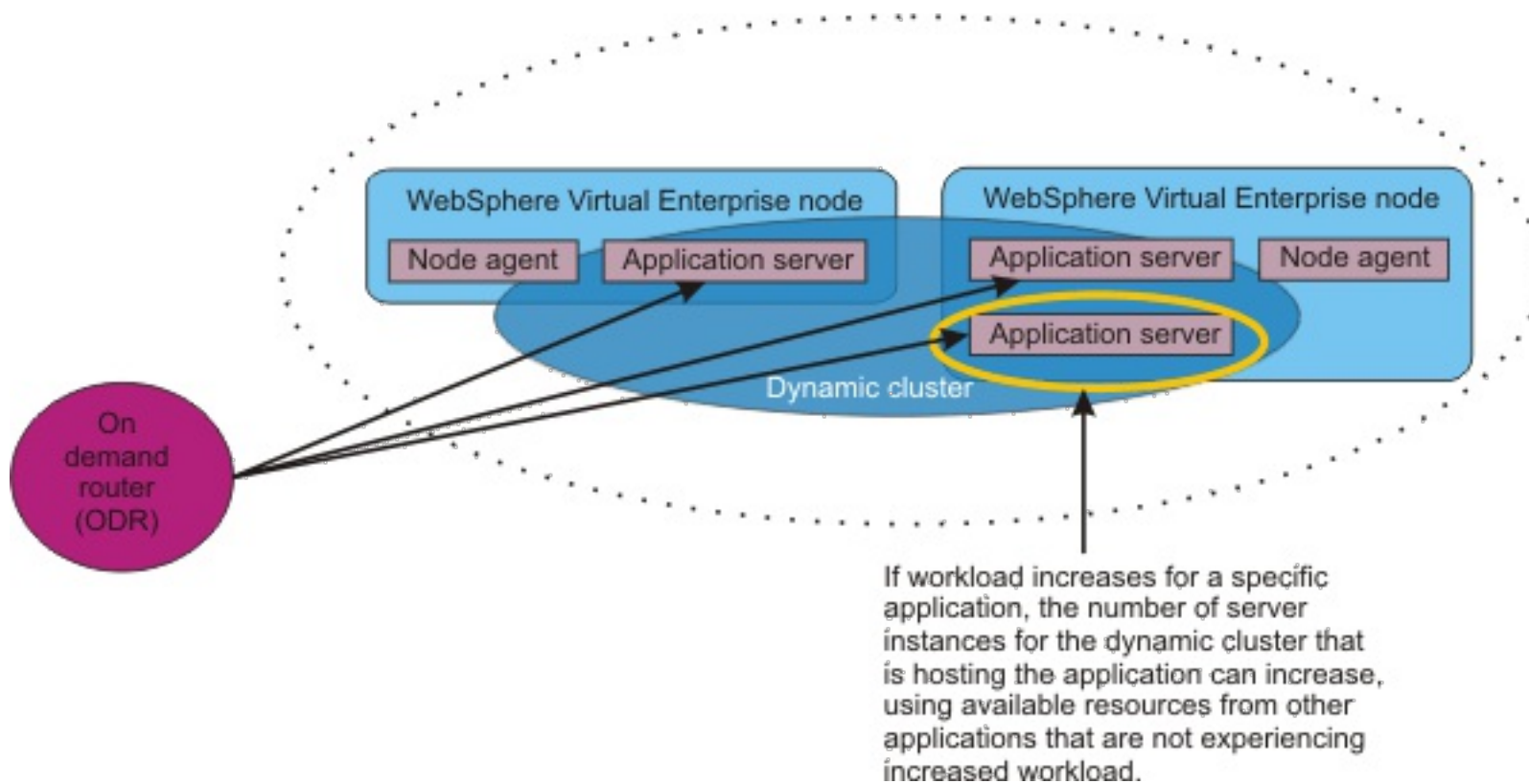


Figure 2. The starting of an additional application server to react to changes in application load. Additional application servers can start on the nodes that are selected by your dynamic cluster membership policy to handle additional requests that are coming in for the application.



Hardware virtualization

While WebSphere Virtual Enterprise provides virtualization of applications in your environment, you can also deploy WebSphere Virtual Enterprise on virtualized hardware, such as ESX, to take advantage of the hardware virtualization capabilities provided by the hosting environment.

Hardware virtualization benefits:

- **Reduced amount of hardware in your environment:** You can run multiple WebSphere Virtual Enterprise nodes on the same physical hardware.
- **Improved hardware management:** You can more easily manage your environment because you have fewer physical computers and can use the hardware virtualization software to manage your images.
- **High availability of hardware:** By configuring server failover, your physical hardware can be highly available. When one server fails, it can be replaced by another server.
- **Dynamic allocation of hardware:** The physical resources, such as processors and memory, on your hosting computers can be shared among the virtual servers in your environment and dynamically allocated as needed. Because the resources are dynamically allocated, restarting the servers is not necessary.
- **Shared storage:** Multiple virtual servers or logical partitions can share the same physical storage. You do not need a physical hard drive for each virtual machine or LPAR.

WebSphere Virtual Enterprise in an environment with server virtualization

WebSphere Virtual Enterprise can operate in supported virtualized hardware environments. Different hardware vendors provide different virtualization capabilities, so the behavior of WebSphere Virtual Enterprise in different hardware virtualization environments can vary. However, common themes exist in hardware virtualization environments, such as the ability to share hardware resources across the virtual servers or logical partitions. Server virtualization environments can run in *shared processor mode* or *dedicated processor mode*. When you use shared processor mode, the physical processors are pooled and shared between the servers or logical partitions that are running on the physical computer. When you use dedicated processor mode, the physical processors are statically assigned to each virtual server or logical partition.

Figure 3. Shared processor mode. In shared processor mode, the physical processors are pooled and shared among the virtual servers or logical partitions.

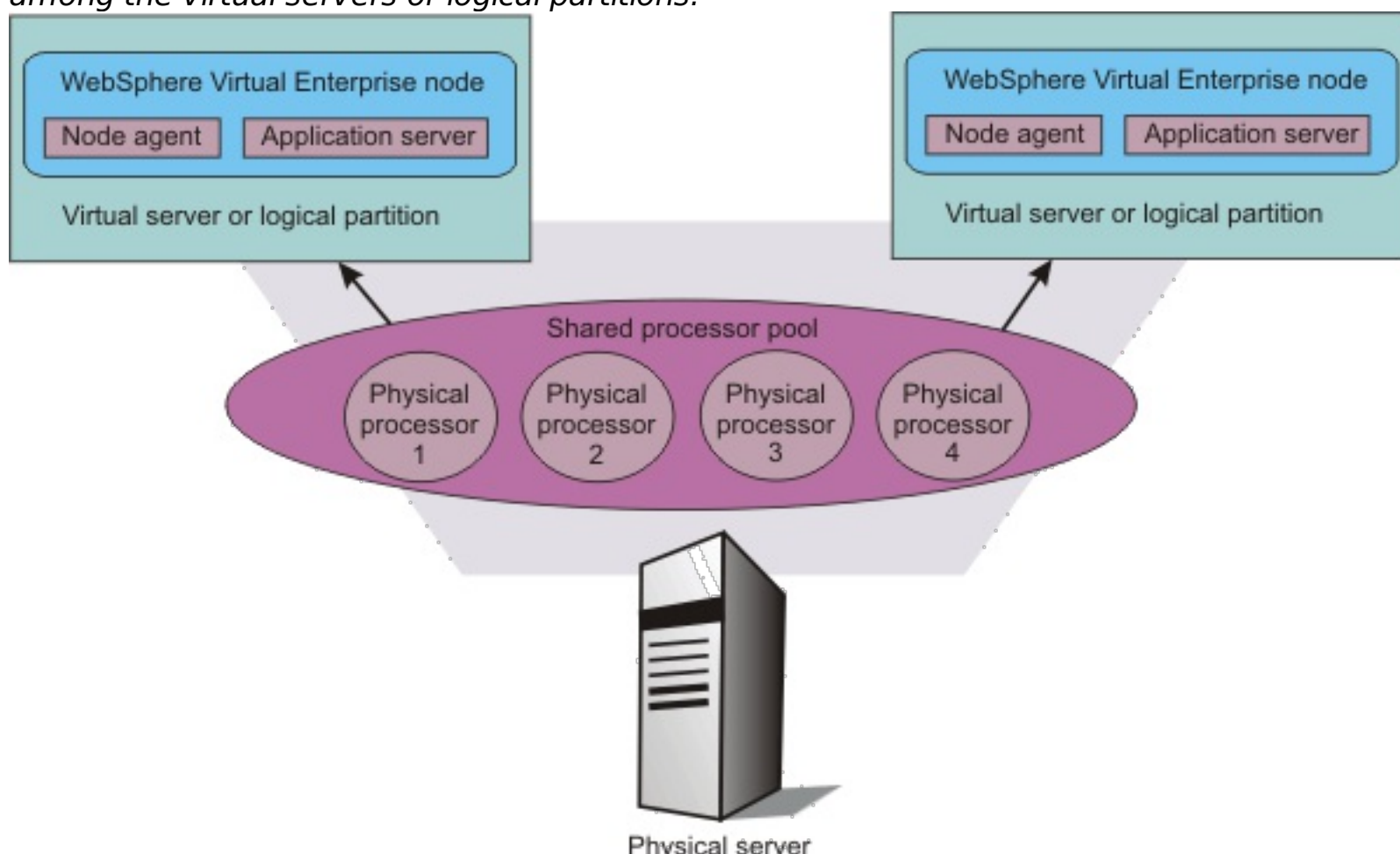
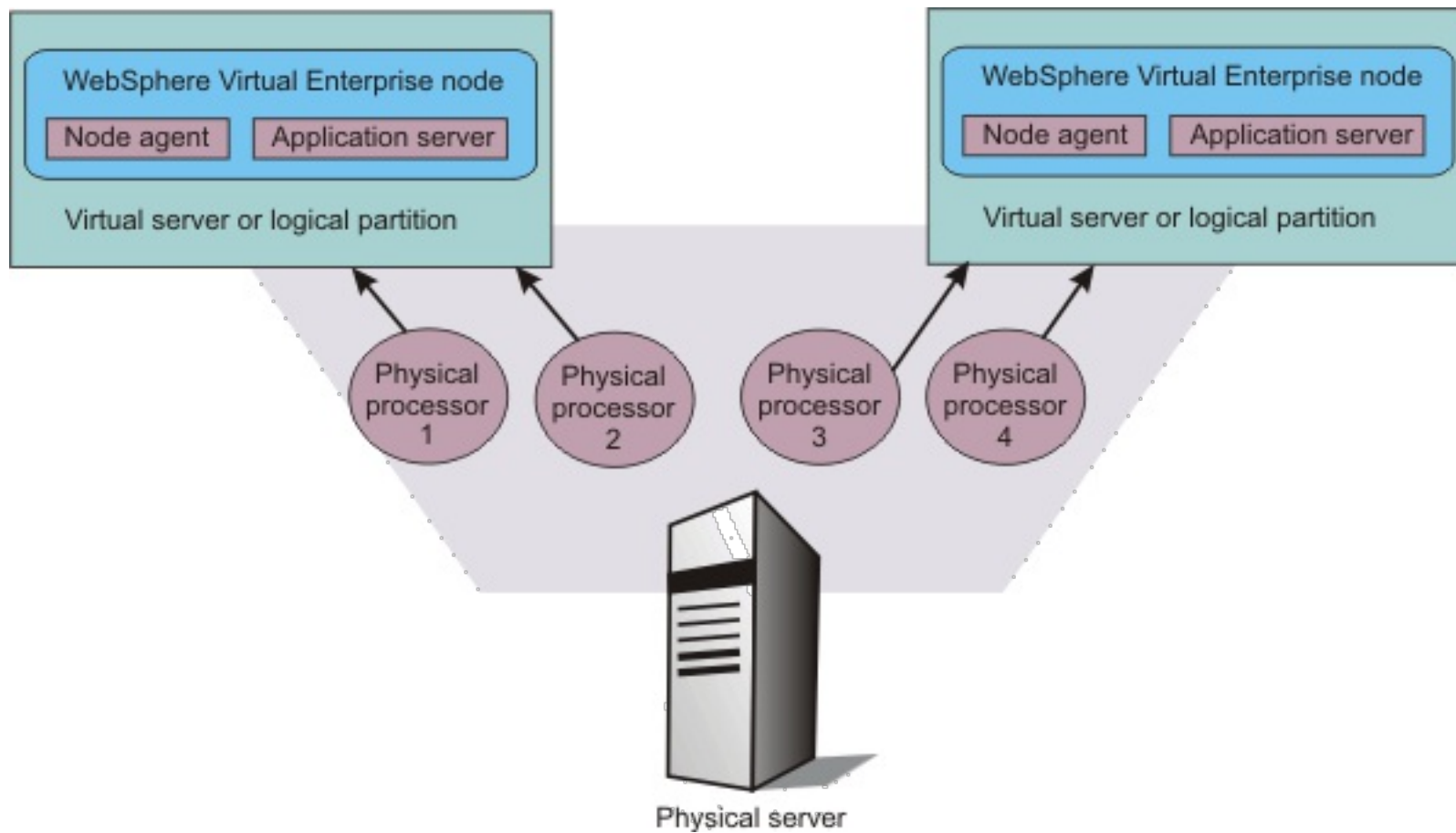
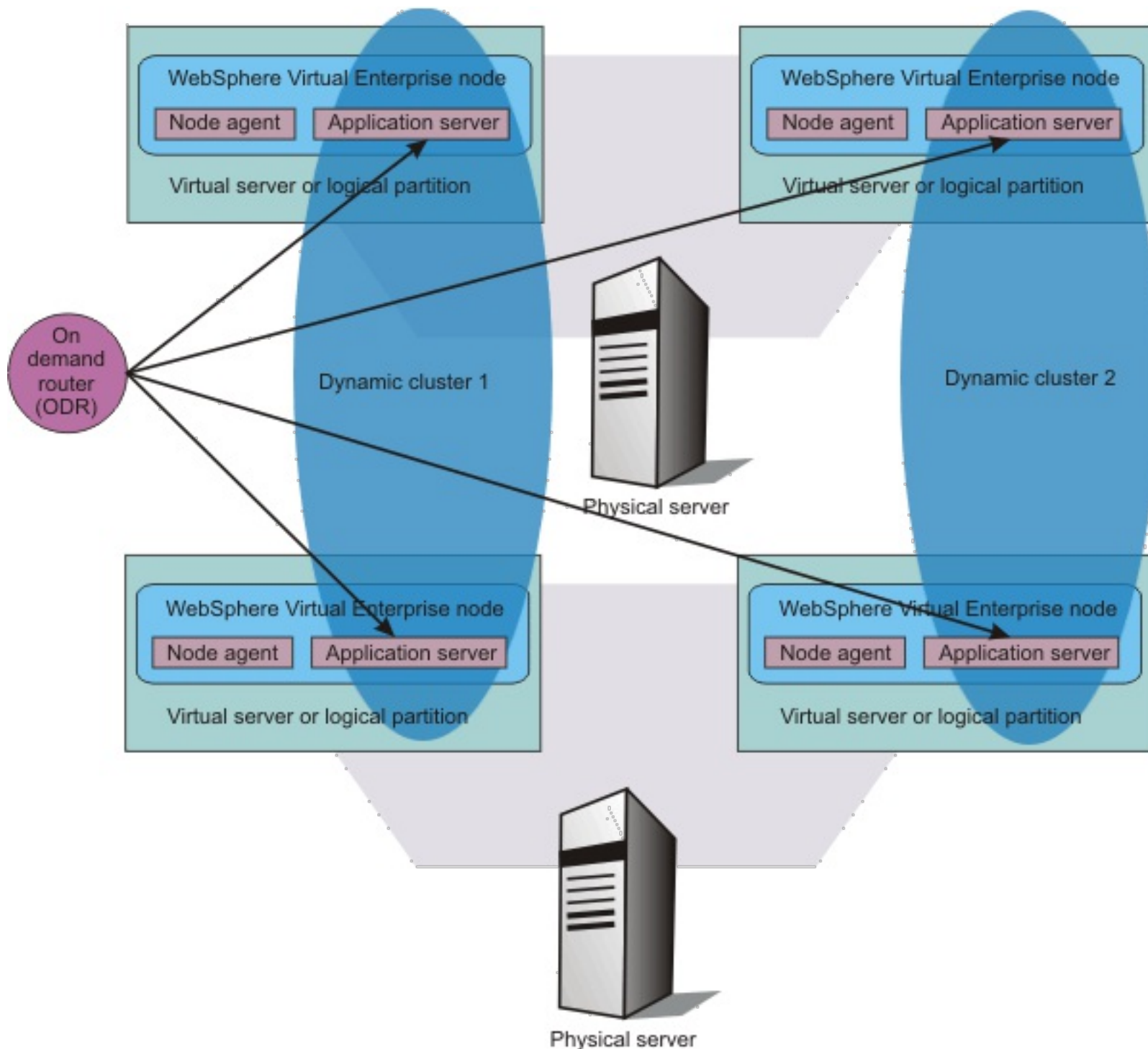


Figure 4. Dedicated processor mode. In dedicated processor mode, the physical processors are statically assigned to each virtual server or logical partition.



WebSphere Virtual Enterprise can also run in hardware virtualization environments with dedicated processor mode. The processor capacity is statically fixed to each virtual server or logical partition. The capacity and assignment do not change dynamically. Because the processor resource does not change for each virtual server or logical partition, using dedicated processor mode does not affect the traffic management and virtualization features of WebSphere Virtual Enterprise.

Figure 5. Coexistence of application infrastructure and hardware virtualization.



On demand router in an environment with server virtualization

The on demand router (ODR) should never be constrained by either CPU or memory usage. Therefore, when you install the ODR in an environment with server virtualization, configure the virtual machine or LPAR in which the ODR runs in dedicated processor mode, or configure it in a mode that guarantees the ODR receives a sufficient amount of CPU resources and dedicated memory when the ODR runs.

[Configuring AIX 5.3 and AIX 6.1 on POWER5 and POWER6 \(Micro-Partitioning\)](#)

To configure AIX® Micro-Partitioning® in shared and uncapped mode to work with WebSphere Virtual Enterprise, you must enable performance collection in the Hardware management console (HMC).

[Supported server virtualization environments](#)

Before you deploy WebSphere Virtual Enterprise on virtualized servers, you must understand the limitations for the server virtualization platform that you are using.

[Virtualization options](#)

Running WebSphere Virtual Enterprise in an environment with server virtualization has advantages, because of the juxtaposition to hardware and the operating system. With application infrastructure virtualization, you can separate applications from the physical infrastructure on which they are hosted. Workloads can then be dynamically placed and migrated across a pool of application server resources, which allows the infrastructure to dynamically adapt and respond to business needs.

[Deploying the product on Solaris 10 Zones with shared CPU](#)

You can install and deploy WebSphere Virtual Enterprise Version 7.0 on Solaris 10 Zones in non-dedicated mode in which CPU and memory are shared between zones.

Parent topic: [Product overview](#)

Configuring AIX 5.3 and AIX 6.1 on POWER5 and POWER6® (Micro-Partitioning)

To configure AIX® Micro-Partitioning® in shared and uncapped mode to work with WebSphere® Virtual Enterprise, you must enable performance collection in the Hardware management console (HMC).

Before you begin

- You must have access to the HMC for your AIX servers.

About this task

If you do not configure this setting in your HMC, but you are using AIX micro-partitioning in shared and uncapped mode, the following message displays in your system out log of the node agent process:

```
APMI_XDSYSTEM_HW_INFO_WARN=ASPS0023W: The logical partition on which this node resides has a mode of shared, a type of uncapped, but performance collection is not enabled.
```

You must enable this setting so that WebSphere Virtual Enterprise can recognize more than the entitled capacity for a logical partition (LPAR).

Procedure

1. Enable the performance hardware collection setting for each LPAR. In the HMC, click **Systems Management > Servers > *physical_server_name* > *LPAR_name***. Select **Allow Performance Hardware collection**.
2. Click **OK**.
3. Restart the LPAR.

Results

WebSphere Virtual Enterprise can recognize more than the entitled capacity for a logical partition (LPAR).

[WebSphere Virtual Enterprise in a micro-partitioned environment](#)

You can use WebSphere Virtual Enterprise in a micro-partitioned environment. WebSphere Virtual Enterprise can understand the utilization of shared processor partitions.

Parent topic: [Virtualization and WebSphere Virtual Enterprise](#)

WebSphere Virtual Enterprise in a micro-partitioned environment

You can use WebSphere Virtual Enterprise in a micro-partitioned environment. WebSphere Virtual Enterprise can understand the utilization of shared processor partitions.

Terminology for server virtualization on POWER AIX systems

You must understand the following terms when you are using WebSphere® Virtual Enterprise on a POWER® AIX® system:

Logical partitioning

The ability to divide the resources of a system to create multiple separate servers. Each server runs its own operating system.

Micro-partitioning

The ability to share a pool of physical processors across multiple logical partitions. Physical processors can be allocated to a partition in increments of 0.1 a processor.

Shared processor partition

A partition that is configured to use a shared processor pool. A shared processor partition is a type of micro-partition.

Entitled capacity

The percentage of processor usage that is granted to a partition; specified in terms of .01 of a processor.

Capped partition

A partition that cannot be granted more processing units beyond the configured entitlement for the partition.

Uncapped partition

A partition that can exceed its configured entitlement when needed, if resources permit.

Entitled capacity

A shared processor partition has a metric called **Entitled Capacity Percentage**. This metric represents the percentage the partition is using of its entitlement at a given point in time. The metric is visible in popular AIX system monitoring tools like the **lparstat** command, the **nmon** command, and the **topas** command.

An uncapped, shared processor partition can be assigned more processing capacity beyond its entitlement. The amount of processing capacity above its entitlement depends on the availability of the availability of processors in the shared pool and the maximum amount that the virtual processor configuration allows.

WebSphere Virtual Enterprise and shared processor partitions

WebSphere Virtual Enterprise Version 6.1.1 and later can understand the utilization of shared processor partitions as well as the dynamic capacity of the shared processor pool on the physical hardware in order to operate in a shared processor partition environment.

New statistics have been added that are specific to shared processor partitions. These statistics can help you understand the utilization of a shared processor partition and the dynamic capacity of the shared processor pool. These statistics are helpful in monitoring WebSphere Virtual Enterprise when operating in a micro-partitioned environment.

Parent topic: [Configuring AIX 5.3 and AIX 6.1 on POWER5 and POWER6 \(Micro-Partitioning\)](#)

Related reference:

[AIX command reference](#)
[NodeStatsHistoricCache](#)

Supported server virtualization environments

Before you deploy WebSphere® Virtual Enterprise on virtualized servers, you must understand the limitations for the server virtualization platform that you are using.

Remember: This page lists the most up to date information on the support of various server virtualization platforms.

Table 1. Server virtualization environments

Virtualization platform	Restrictions	Supported processor sharing mode
AIX® 5.3 and AIX 6.1 on POWER5 and POWER6® (Micro-Partitioning®)	<p>See Configuring AIX 5.3 and AIX 6.1 on POWER5 and POWER6 (Micro-Partitioning) for required configuration steps.</p> <p>For uncapped shared processor partitions, equal partition weights are recommended. WebSphere Virtual Enterprise does not have a partition to physical machine mapping and thus does not use the partition weight when making workload distribution and server placement decisions.</p>	<p>All WebSphere Virtual Enterprise features are supported on AIX Micro-partitioning in capped or uncapped mode.</p> <p>Shared capped, shared uncapped, and dedicated modes are supported for both POWER5 and POWER6. Support for dedicated donating and multiple shared pools modes are available on POWER6 only.</p>
AIX 7.1 on POWER7®	None.	<p>POWER7 processor in POWER6 compatibility mode</p> <p>POWER7 processor mode</p>
Application Workload Partition (WPAR) AIX 6.1 and 7.1	None.	Shared.
KVM in SUSE Linux Enterprise Server (SLES) 11	None.	Dedicated and shared.
Red Hat KVM as delivered with Red Hat Enterprise Linux (RHEL)	None.	Dedicated and shared.

5.4, 5.5, 6.0		
Linux on POWER® (Micro-Partitioning)	None.	Dedicated.
Linux on z/VM®	<ul style="list-style-type: none"> The guest operating system must be Red Hat Enterprise Linux (RHEL) 5.0, 5.1, 5.2 or SUSE Linux Enterprise Server (SLES) 10. Mapping a Linux on zSeries® operating system image to a physical logical partition (LPAR): WebSphere Virtual Enterprise can balance workload across multiple z/VM virtual machines that are running Linux as the guest operating system. However, it does not have knowledge of the LPAR that is hosting the virtual machines on the z/VM image and is therefore unable to make workload balancing decisions based on the workload at the LPAR level. Service policy goals: Workload that is not under the control of WebSphere Virtual Enterprise might be running on other Virtual Machines on the same z/VM image. This workload might affect service policy goals. 	Dedicated and shared.
Solaris Operating Environment 10 on Sun (Zones)	None for dedicated.	Dedicated and shared.
VMware ESX 3.5, 4.0, 4.1 VMware vSphere 4	<ul style="list-style-type: none"> See VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise for a list of limitations. 	Dedicated and shared modes are supported for ESX or VMware vSphere 4, but you must configure WebSphere Virtual Enterprise to communicate with the ESX or VMware vSphere 4 hypervisor or the vCenter that is in control of the hypervisors where WebSphere Virtual Enterprise is running.
Linux Xen	<ul style="list-style-type: none"> You must be using SLES 11 or later. You must use the following guidelines for setting up the clock correctly: SUSE Linux Virtual Machines: Section 7.4.3 Virtual Machine Clock Settings. Use para-virtualized mode over full-virtualized mode. Full-virtualized mode can lead to network bottlenecks that can adversely affect performance of the system. This performance problem is particularly possible for large topologies. 	Dedicated.
HP-UX 11i v3 on HP using	Currently not supported.	None.

Virtual Partitions (vPar s) or Integrity VMs		
Microsoft Hyper-V	Currently not supported.	None.

Parent topic: [Virtualization and WebSphere Virtual Enterprise](#)

Virtualization options

Running WebSphere® Virtual Enterprise in an environment with server virtualization has advantages, because of the juxtaposition to hardware and the operating system. With application infrastructure virtualization, you can separate applications from the physical infrastructure on which they are hosted. Workloads can then be dynamically placed and migrated across a pool of application server resources, which allows the infrastructure to dynamically adapt and respond to business needs.

While there are benefits to server virtualization, this virtualization option can sometimes be too far removed from the application layer, and can therefore lack a quality of service based on application information, such as response times and response codes. Application infrastructure virtualization, on the other hand, can provide a quality of service at the application layer. Additionally, this virtualization option provides optimization and autonomic health corrections for the middleware layer through the use of Java™ virtual machines (JVMs). In other words, WebSphere Virtual Enterprise minimizes the footprint that is required by the middleware layer, and keeps the middleware highly available through health management. A third benefit of application infrastructure virtualization is the application-demand-driven optimization of the operating system and virtual machine layers. In this sense, the application infrastructure virtualization layer optimizes the server virtualization layer by creating and deleting virtual machines as needed to meet application demand. This is a fully autonomic and optimized private cloud, minimizing the overhead, especially memory, that is incurred by idle virtual machines.

Table 1. Differences between virtualization options

Server virtualization	Application infrastructure virtualization
Hardware is separated from guest operating systems	Application server container is separated from the application
Increases server hardware utilization with server virtualization being used to create a pool of server resources	Increases application server container utilization with application infrastructure virtualization being used to create pools of application server resources
Leverages a layer of insulation that is called the hypervisor	Leverages a layer of insulation that is called the application fabric or application server virtual layer
Resource groups can be assigned resource profiles that are connected to quality of service needs	Application server resources are assigned to dynamic clusters to which applications are deployed
Equates to pinning more cores to the guest operating system (vertical scaling) as opposed to assigning more guests to the server (horizontal scaling)	New JVMs can be restarted in the dynamic cluster to scale out or can be shut down to scale down
Resource sharing is controlled by policies that are expressed in terms of hardware units (CPU cores, memory size)	Resource sharing is controlled by policies that are expressed in terms of what end users see (response time)
Has visibility to the hardware and guest operating system layers	Has visibility to the application, middleware, guest operating system, and hardware layers

Parent topic: [Virtualization and WebSphere Virtual Enterprise](#)

Deploying the product on Solaris 10 Zones with shared CPU

You can install and deploy WebSphere® Virtual Enterprise Version 7.0 on Solaris 10 Zones in non-dedicated mode in which CPU and memory are shared between zones.

Before you begin

If Solaris 10 Zones are configured to share CPU and memory between zones, WebSphere Virtual Enterprise must be installed on the global zone of each Solaris machine. This can be a middleware agent installation or a standard node installation.

About this task

In releases prior to Version 7.0, WebSphere Virtual Enterprise supports Solaris Zones only in dedicated mode in which there is no sharing of CPU or memory between zones. Version 7.0 supports Solaris Zones in non-dedicated mode in which CPU and memory are shared between zones.

The WebSphere Virtual Enterprise middleware agent or node agent that is running in the global zone of each Solaris machine provides the necessary information for all zones on the machine. As long as the middleware agent or node agent process is running, it will automatically gather the appropriate information required for WebSphere Virtual Enterprise to function appropriately in this environment.

Procedure

- Install the node agent for WebSphere Virtual Enterprise.
 1. Stop all the processes that are running on the workstation that you are preparing for installation. Run the **stopManager** and **stopNode** commands to stop the deployment manager and nodes.
 2. Run the **install** command to start the installation wizard, and complete all of the required fields on the wizard panels.
 3. Repeat the process to install the product on each node in the cell.
- Install the middleware agent for WebSphere Virtual Enterprise.
 1. Run the **install** command to start the installation wizard, and complete all of the required fields on the wizard panels.

Parent topic: [Virtualization and WebSphere Virtual Enterprise](#)

Related concepts:

[Virtualization and WebSphere Virtual Enterprise](#)

Related tasks:

[Installing and customizing the product](#)

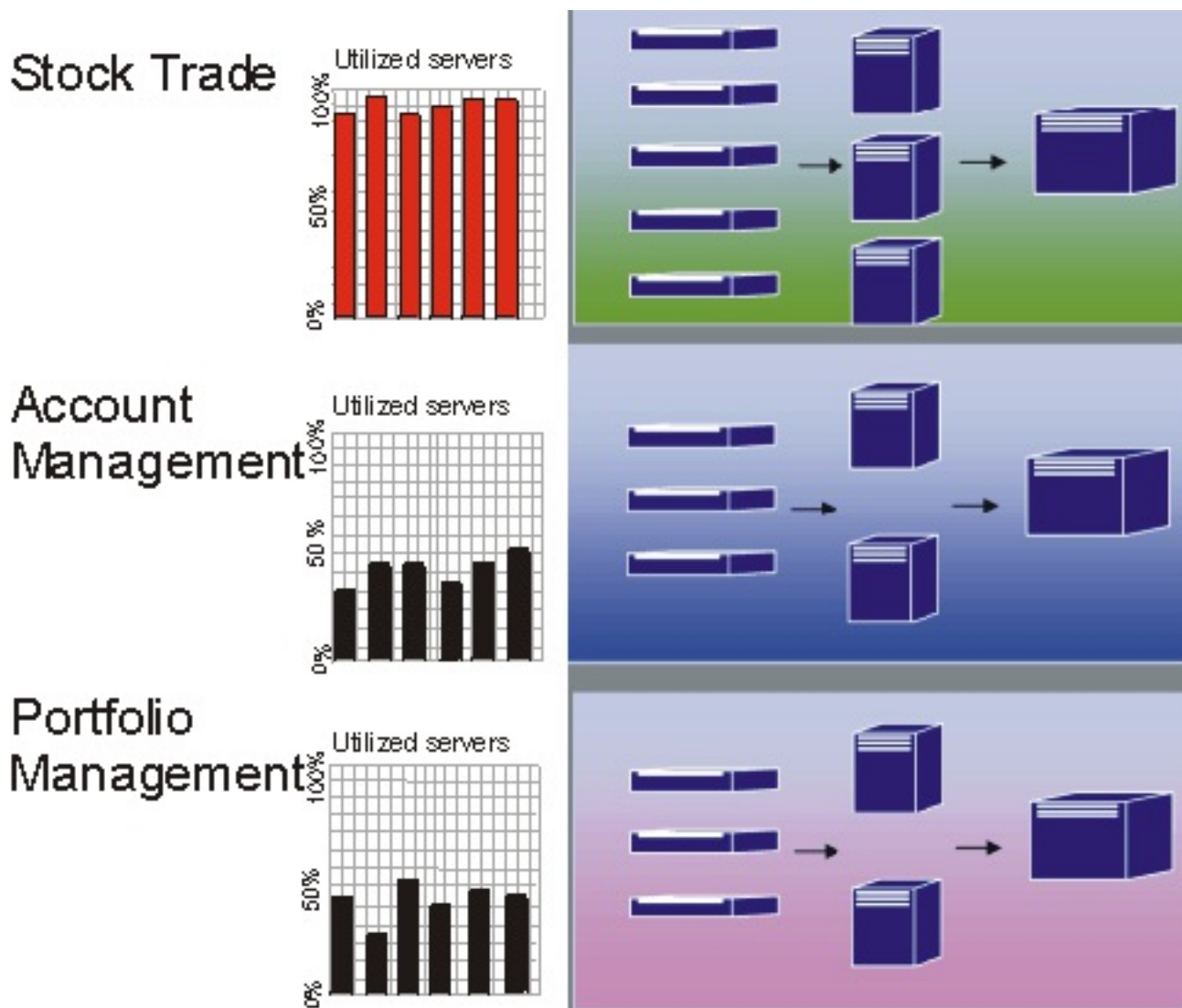
Dynamic operations

WebSphere® Virtual Enterprise contains functionality to monitor the virtualized application server environment and make workload management optimizations or recommendations that are based on the data observed, increasing the quality of service. This capability is referred to as *dynamic operations*.

Dynamic operations capabilities are available to customers using the WebSphere Virtual Enterprise product, and to those who are using it in combination with the WebSphere Extended Deployment Compute Grid product. Increasingly, businesses are rigidly tied to the availability and speed of applications that deliver essential services to customers. Loss of availability translates into lost business, which means lost opportunity and lost revenue. Dynamic operations is a fluid and dynamic environment, supporting the continuous availability of applications through application server virtualization and application virtualization, the dynamic placement of applications, prioritization and flow control of work to the applications, and integrating with overall dynamic operations infrastructure management.

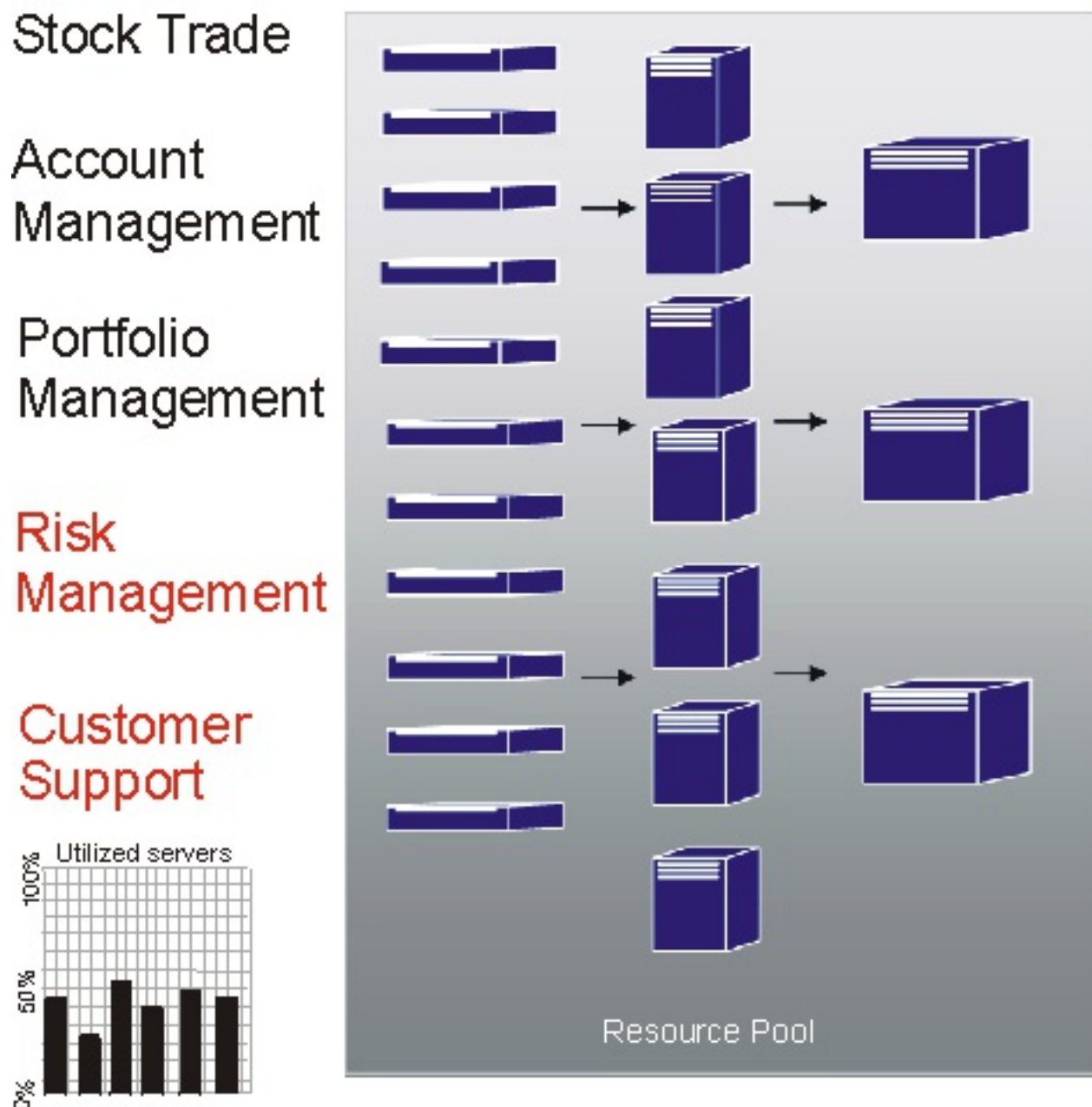
In a typical WebSphere Application Server environment, there are sometimes static islands of dedicated resources to particular applications. This static structure leads to an inefficient use of resources. Some servers are not used to their full capability, and other servers are overloaded.

Figure 1. Typical WebSphere Application Server environment



WebSphere Virtual Enterprise supports a far more flexible environment using its dynamic operations features. Dynamic operations consists of autonomic managers that maximize utilization using defined business goals. These autonomic managers monitor performance metrics, analyze the monitored data, offer a plan for running actions, and can start these actions in response to the flow of work.

Figure 2. WebSphere Virtual Enterprise environment



WebSphere Virtual Enterprise offers the following autonomic managers as part of the dynamic operation functionality:

Autonomic request flow manager

Controls the order of requests into the application server tier and the rate of request flows. Using classification and the defined service goals, the autonomic request flow manager (ARFM) decides how and when to dispatch HTTP requests to the next tier. The ARFM also decides when Internet Inter-ORB Protocol (IIOP) and Java™ Message Service (JMS) requests are run at the application server tier, even though these requests are not routed through the ODR. For IIOP requests, only standalone Enterprise JavaBeans (EJB) clients are supported. Note that JMS support is only for message-driven beans.

Dynamic workload manager (DWLM)

Performs load balancing across available application servers. In particular, for a given request flow, DWLM balances requests across the available nodes to regulate response times. DWLM dynamically updates the application status as the application placement controller modifies a running application infrastructure.

Application placement controller

Creates and removes application instances to manage HTTP, SIP, and IIOP traffic. The application placement controller can dynamically address periods of intense workflow that otherwise require the manual intervention of a system administrator. For IIOP requests, only standalone EJB clients are supported.

Health management

Maintains a robust application server environment using a health policy to identify the criteria that require action. When the criteria is met, action is taken to ensure that the environment remains healthy.

Autonomic managers with the on demand router (ODR) are the primary functional parts of dynamic operations. An ODR can be defined and started before any service policies are defined, but operational policies can be defined before the appearance of the work to which they apply. However, if policies are not defined, the early work is handled by the default policies. When work enters the ODR, an optimization effort achieves a balance of performance results. As the work flows, the dynamic workload manager balances the load. As work variations change and the balance of work in the nodes is upset, the application placement controller, autonomic request flow manager, and the dynamic workload manager rebalance running applications to ensure efficient work flows.

The combination of these autonomic managers provides a seamless, end-to-end dynamic runtime ability.

[Overview of dynamic operations](#)

WebSphere Virtual Enterprise is built upon a virtualized infrastructure that redefines the traditional concepts of Java Platform, Enterprise Edition (Java EE) resources and applications and their relationships with one another. This application infrastructure virtualization facilitates the product's ability to automate operations in an optimal manner, increasing the quality of service. By introducing an automated operating environment with workload management, you can reduce your total cost of ownership by performing more work while using less hardware.

[Overview of request flow prioritization](#)

With WebSphere Virtual Enterprise, you can define performance goals and bind them to specific subsets

of the incoming traffic. The on demand router (ODR) and associated autonomic managers support business goals in times of high load by making smart workload management decisions about the work that is being sent through the ODR. Not all the work in your configuration is equally important. The ODR can support this concept by forwarding different flows of requests more or less quickly to achieve the best balanced result and maintain the quality of service.

Dynamic operations environment

The dynamic operations environment, consisting of the autonomic, or on demand, features of WebSphere Virtual Enterprise supports three modes of operation: manual, supervised, and automatic. This environment is supported by autonomic managers that perform tasks such as application placement, workload management, health management, and request flow management.

Components of dynamic operations

WebSphere Virtual Enterprise is built on features of autonomic computing in the dynamic operations environment. With these features, the virtualized application server environment can expand and contract as the business demands. Using the autonomic managers in the product environment, dynamic operations can make logical decisions based on business goals.

Parent topic: [Product overview](#)

Related concepts:

[Overview of dynamic operations](#)
[Components of dynamic operations](#)
[Dynamic operations environment](#)
[Dynamic application placement](#)

Related information:

[Product overview](#)

Overview of dynamic operations

WebSphere® Virtual Enterprise is built upon a virtualized infrastructure that redefines the traditional concepts of Java™ Platform, Enterprise Edition (Java EE) resources and applications and their relationships with one another. This application infrastructure virtualization facilitates the product's ability to automate operations in an optimal manner, increasing the quality of service. By introducing an automated operating environment with workload management, you can reduce your total cost of ownership by performing more work while using less hardware.

The features of WebSphere Virtual Enterprise apply to the management of the WebSphere Extended Deployment Compute Grid environment and its associated background workload only when the two features are deployed together. Therefore, if you are a customer of WebSphere Extended Deployment Compute Grid, you must have both products to take advantage of the features in WebSphere Virtual Enterprise.

Increasingly, businesses are tied rigidly to the availability and speed of applications that deliver essential services to customers. Loss of availability translates into lost business, which means lost opportunity and lost revenue. To meet this need, the dynamic operations environment is a fluid and dynamic environment, enabling applications to be available continuously through application virtualization, the virtualization of WebSphere resources, provisioning of WebSphere applications, prioritization and scheduling of applications, and integrating with overall dynamic operations environment infrastructure management.

The dynamic operations environment consists of autonomic managers whose purpose is to maximize utilization using business goals that you have defined. You can monitor performance metrics, analyze the monitored data, offer a plan for executing actions, and have the capability to execute these actions in response to the flow of work.

Parent topic: [Dynamic operations](#)

Related concepts:

[Dynamic operations](#)

[Virtualization and WebSphere Virtual Enterprise](#)

Overview of request flow prioritization

With WebSphere® Virtual Enterprise, you can define performance goals and bind them to specific subsets of the incoming traffic. The on demand router (ODR) and associated autonomic managers support business goals in times of high load by making smart workload management decisions about the work that is being sent through the ODR. Not all the work in your configuration is equally important. The ODR can support this concept by forwarding different flows of requests more or less quickly to achieve the best balanced result and maintain the quality of service.

Role of the ODR

The ODR is a server that acts as an HTTP proxy or a SIP proxy. An ODR contains the autonomic request flow manager (ARFM). ARFM prioritizes inbound traffic according to service policy configuration and protects downstream servers from being overloaded. Traffic is managed to achieve the best balanced performance results, considering the configured service policies and the offered load. Note that for an inbound User Datagram Protocol (UDP) or Session Initiation Protocol (SIP) message, the ODR can route the message to another ODR to properly check for and handle UDP retransmissions.

The on demand configuration (ODC) component allows the ODR to sense its environment. ODC dynamically configures the routing rules at runtime to allow the ODR to accurately route traffic to those application servers. An ODR is able to route HTTP requests to WebSphere Virtual Enterprise servers, WebSphere Application Server Network Deployment servers, and servers that are not running WebSphere software. The ODR, like the Web server plug-in for WebSphere Application Server, uses session affinity for routing work requests. After a session is established on a server, later work requests for the same session go to the original server, which maximize cache usage and reducing queries to back-end resources.

Service policies

A service policy is a user-defined categorization that is assigned to potential work as an attribute that is read by the ARFM. You can use a service policy to classify requests based on request attributes, including the URI, the client name and address, and the user ID or group. By configuring service policies, you apply varying levels of importance to the actual work. You can use multiple service policies to deliver differentiated services to different categories of requests. Service policy goals can differ in performance targets as well as importances.

The autonomic request flow manager (ARFM)

The ARFM exists in the ODR and controls request prioritization. When the following components that the ARFM contains are working together, they are able to properly prioritize incoming requests:

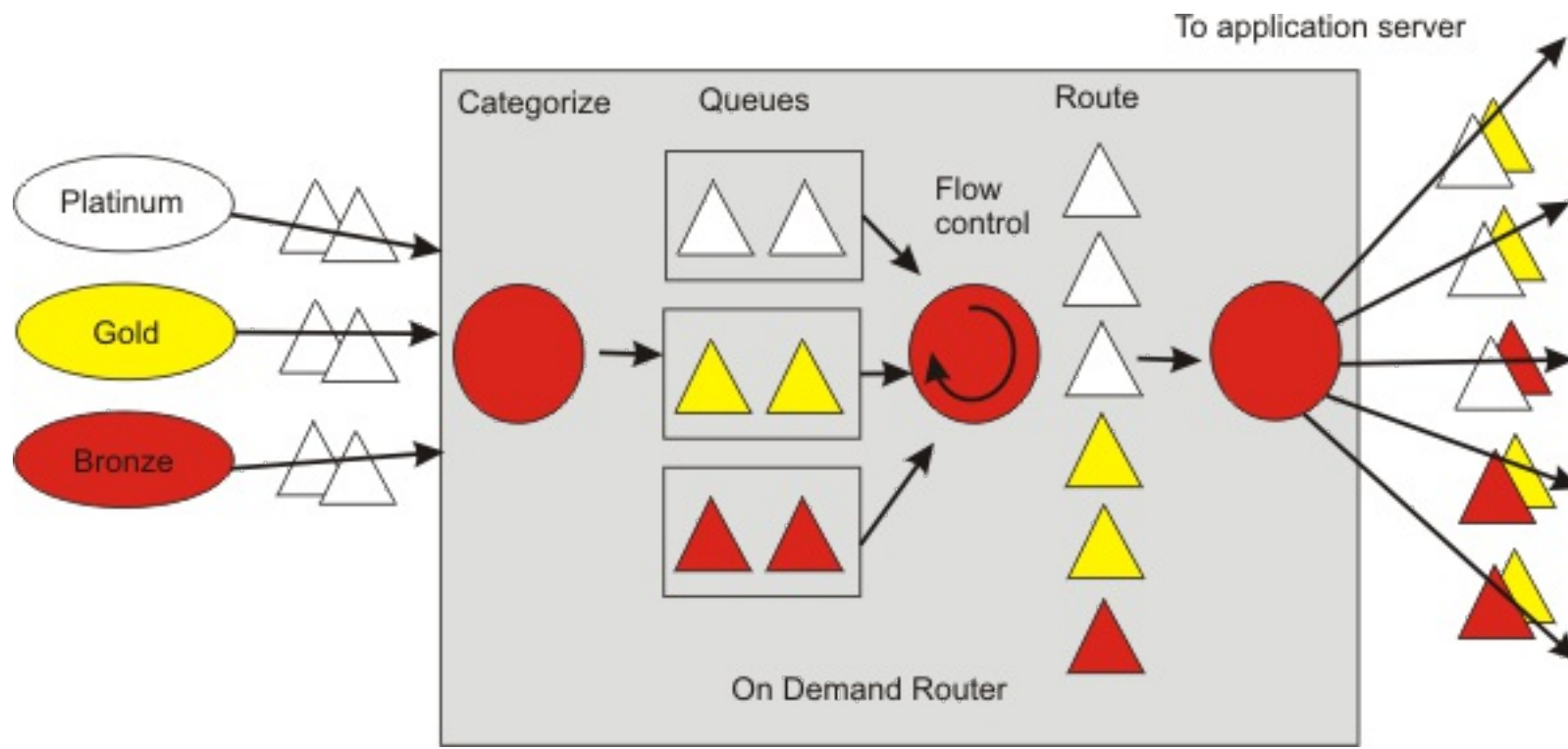
- A compute power controller per target cell. That is, a cell to which some ARFM gateway directly sends work. This is an `HAMangedItem` that can run in any node agent, ODR, or deployment manager.
- A gateway per a used combination of protocol family, proxy process, and deployment target. A gateway runs in its proxy process. For HTTP and SIP, the proxy processes are the on demand routers; for Java™ Message Service (JMS) and IIOP, the proxy processes are the WebSphere application servers.
- A work factor estimator per target cell. This is an `HAMangedItem` that can run in any node agent, ODR, or deployment manager.

Dynamic workload management (DWLM)

Dynamic workload management (DWLM) is a feature of the ODR that applies the same principles as workload management (WLM), such as routing based on a weight system, which establishes a prioritized routing system. DWLM autonomously sets the routing weights to WLM. With WLM, you manually set static weights in the administrative console. With DWLM, the system can dynamically modify the weights to stay current with the business goals. DWLM can be shut off. If you intend to use the automatic operating modes for the components of dynamic operations, then setting a static WLM weight on any of your dynamic clusters could get in the way of allowing the on demand aspect of the product to function properly. The WebSphere Application Server Network Deployment WLM and WebSphere Virtual Enterprise DWLM are not limited to the on demand routers, but also apply to IIOP traffic when the client is using the WebSphere Application Server Java Development Kit (JDK) and object request broker (ORB) and *prefer local* routing is not employed.

The following diagram shows an equal amount of requests flow into the ODR. Platinum, gold, and bronze are used to depict a descending order of importance, respectively. After the work is categorized, prioritized, and queued, a higher volume of more important work (platinum) is processed, while a lower volume of less important (bronze) work waits to get queued. Because bronze is delayed, the long-term average rate of Bronze coming out of the ODR is not less than the long-term average rate of bronze going in. The features of the dynamic operations attempt to keep the work within the target time allotted for completion.

Figure 1. Flow of requests into and through the on demand router



Parent topic: [Dynamic operations](#)

Related concepts:

[Dynamic operations](#)

[Dynamic operations environment](#)

Related tasks:

[Creating and configuring ODRs](#)

[Defining a service policy](#)

Dynamic operations environment

The dynamic operations environment, consisting of the autonomic, or on demand, features of WebSphere® Virtual Enterprise supports three modes of operation: manual, supervised, and automatic. This environment is supported by autonomic managers that perform tasks such as application placement, workload management, health management, and request flow management.

Defining the mode of the operating environment involves configuring the WebSphere Virtual Enterprise components. You might find that you want a mixed environment, with some components running automatically, or autonomously, while others are not. Flexibility is built into the product to support logical systems, rather than a switch to turn things on or off. In addition, the product supports a supervised operating mode, where the system administrator accepts or denies the recommendations of the autonomic managers.

The following are the features of the dynamic operations, as well as the operating modes they support:

Autonomic manager	Description
Application placement controller	<p>Characteristics:</p> <ul style="list-style-type: none"> • Supports two modes: enabled or disabled. <p>If enabled, the application placement controller supports automatic, supervised, and manual mode for each dynamic cluster with which it is affiliated. The mode is set on the dynamic cluster.</p> <ul style="list-style-type: none"> • Applicable to a dynamic cluster. • Not autonomous functionality, as this is dependent upon the dynamic cluster setting. • Is a distributed resource managed by the high availability manager. <p>The application placement controller exists within all node agent and deployment manager processes. The controller is active in one of these processes. If the active process fails, the controller becomes active on another node agent or deployment manager process.</p>
Autonomic request flow manager (ARFM)	<p>Characteristics:</p> <ul style="list-style-type: none"> • Supports two modes: manual or automatic. • Autonomous functionality, as this can be enabled individually. • If you are not in ARFM manual mode, you can set the <code>arfmManualAllocation</code> custom property to disable the autonomic setting of the ARFM control knobs.
Dynamic WLM (DWLM)	<p>Characteristics:</p> <ul style="list-style-type: none"> • Supports two modes: enabled or disabled. • Applicable to static clusters and dynamic clusters. DWLM is enabled on dynamic clusters by default, and disabled on static clusters by default. To enable DWLM on a static cluster, click Servers > Clusters > <i>cluster_name</i> > Dynamic workload management (DWLM). To disable DWLM on a dynamic cluster, click Servers > Dynamic clusters > <i>dynamic_cluster_name</i> > Dynamic workload management (DWLM). • Autonomous functionality, as this can be enabled individually.
Health controller	<p>Characteristics:</p> <ul style="list-style-type: none"> • Supports two modes: enabled or disabled. • Applicable to a cell, or subset of a cell. • Autonomous functionality, as this can be enabled individually. • Is a distributed resource managed by the high availability manager. <p>The health controller exists within all node agent and deployment manager processes. The controller is active in one of these processes. If the active process fails, the controller becomes active on another node agent or deployment manager process.</p>

Parent topic: [Dynamic operations](#)

Related concepts:

[Overview of dynamic operations](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

[Configuring dynamic application placement](#)

[Configuring the health controller](#)

[Creating dynamic clusters](#)

Related reference:

[Static clusters versus dynamic clusters](#)

Related information:

[Autonomic request flow manager custom properties](#)

Components of dynamic operations

WebSphere® Virtual Enterprise is built on features of autonomic computing in the dynamic operations environment. With these features, the virtualized application server environment can expand and contract as the business demands. Using the autonomic managers in the product environment, dynamic operations can make logical decisions based on business goals.

Restriction: WebSphere Virtual Enterprise does not support Session Initiation Protocol (SIP) features on the z/OS® operating system.

The dynamic operations environment has the following components:

Operational policy

An operational policy is a business, or performance objective that supports specific goals for specific requests. Operational policies include service and health policies. A service policy defines a business goal and an importance and contains one or more transaction classes. For a given work class, a rule condition maps to a transaction class which belongs to a service policy. The service policy contains the business goal requirements and the work class contains the work description upon which the service policy is applicable. The combination of these policies is read by the dynamic operations environment to make decisions on HTTP, SOAP, JMS, SIP, and IOP work requests.

Dynamic clusters

A dynamic cluster is an application deployment target that is expandable and contractible, as needed by the dynamic operations environment. Dynamic clusters provide the core functionality of application infrastructure virtualization. Dynamic cluster instances are created on nodes that meet the criteria of a *membership policy* that you specify when you create the dynamic cluster.

Autonomic request flow manager

The autonomic request flow manager (ARFM) has numerous functions:

- Uses queuing of incoming messages in edge-based gateways to provide computing power overload protection and differentiated service. The computing resource protected from overload is typically CPU power. The differentiated service aims to provide the best balanced performance results among various flows of traffic, relative to the given operational policy and current offered load.
- Can optionally exert dialog/session-oriented admission control for the sake of computing power overload protection.
- Sends information to the placement controller about each cluster to enable the placement controller to optimize the placement for the operational policy and currently offered load. The information about a given cluster is the relationship between computing power and service utility for that cluster.

On demand router

On demand router (ODRs) are intelligent HTTP proxies and/or a SIP proxy. ODRs are the point of entry into a product environment and are gateways through which HTTP requests and SIP messages flow to back-end application servers. It can momentarily queue requests for less important applications in order to allow request from more important applications to be handled more quickly or to protect back-end application servers from being overloaded. The ODR is aware of the current location of a dynamic cluster's server's instances, so that requests can be routed to the correct endpoint. The ODR can also dynamically adjust the amount of traffic sent to each individual server instance based on process utilization and response times. Note that for an inbound SIP/UDP message, the ODR might route the message to another ODR in order to properly check for and handle UDP retransmission.

Dynamic workload manager

The autonomic request flow manager classifies and prioritizes requests to application servers based on the demand and policies. The dynamic workload manager then distributes the requests among the nodes in a dynamic cluster to balance the work.

Application placement controller

The application placement controller is an autonomic manager in the dynamic operations infrastructure that supports application virtualization, or the fluid mobility of applications within a dynamic cluster. The application placement controller adds application server instances when the work is more than can be handled by the current application, and stops application servers when there are too few requests for the number of started applications.

Health controller

The health controller constantly monitors the defined health policies. When a condition specified by a health policy is not met in the environment, the health controller assures that the configured actions are taken to correct the problem.

EWLM

The enterprise workload manager (EWLM) manages sub-goals and resource allocations for the larger environment which contains WebSphere Virtual Enterprise.

Operational policies

WebSphere Virtual Enterprise uses service level management and policy-driven goals to achieve a healthy and robust environment with a high quality of service. Operational policies consist of service policies and health policies. With service policies, you can differentiate between applications according to their perceived level of importance and target values. With health policies, you can identify conditions to

watch for and the product acts on these conditions to ensure a healthy environment.

Parent topic: [Dynamic operations](#)

Related concepts:

[Overview of dynamic operations](#)

Related tasks:

[Configuring health management](#)

[Enabling the on demand router to work with IBM Enterprise Workload Manager](#)

[Configuring dynamic application placement](#)

[Creating ODRs](#)

[Configuring the autonomic request flow manager](#)

[Creating dynamic clusters](#)

[Deploying applications with defined service levels](#)

Operational policies

WebSphere® Virtual Enterprise uses service level management and policy-driven goals to achieve a healthy and robust environment with a high quality of service. Operational policies consist of service policies and health policies. With service policies, you can differentiate between applications according to their perceived level of importance and target values. With health policies, you can identify conditions to watch for and the product acts on these conditions to ensure a healthy environment.

Service policy

Service policies and, for most kinds of work requests, work classes are used to categorize and prioritize work requests. A service policy consists of a user-defined performance goal and (in some cases) an importance level. Service policies are related to work requests through transaction classes. Each work request belongs to exactly one transaction class, and each transaction class belongs to exactly one service policy. For most kinds of work requests, work classes are used to map incoming requests to transaction classes. Each work class is attached to a Java™ Platform, Enterprise Edition (Java EE) application and a basic request feature; Uniform Resource Identifier (URI) prefix for HTTP, method name for Internet Inter-ORB Protocol (IIOP), and bus and destination for Java Message Service (JMS). Each work class specifies how the relevant requests are classified into transaction classes. For generic server clusters, and for Session Initiation Protocol (SIP), work classes are not used; instead, the rules for classifying requests to transaction classes are configured on the on demand routers.

A health policy, in contrast to a service policy, is the definition of specific health criteria that you want your product to protect itself against. The health management function uses the defined policy to search the environment for software malfunctions.

Different service policies can have different kinds of goals. The discretionary goal has no associated value or importance. An average response time goal has an associated response time threshold and importance, whereas a response time percentile goal has two associated values; percentage and time, and an importance.

The performance management done by the autonomic request flow manager, the dynamic workflow manager and the application placement controller achieves a defined balance of the performance results. The defined balance among nondiscretionary flows is achieved by either having the flows all under threshold by the same relative amount, expressed as a fraction of the threshold, or by exceeding the threshold by a relative amount that is inversely proportional, 100 - importance. The discretionary flows are given a minimal allocation.

A performance goal strategy requires a monitoring capability by the autonomic manager to determine whether specified performance goals are met, and a reporting capability to notify a provisioning module when changes are required. Furthermore, to account for the case when it is impossible to satisfy all performance goals, it is possible to assign a business value to each performance goal. Administrators must have an in-depth understanding of deployed applications so that they can create realistic performance goals.

Health policy

A health policy works much the same as the service policy, except that the health policy provides a health goal for the environment. A health policy consists of a *health condition* and a *health action*. A health condition specifies a problematic scenario in your environment. If this scenario occurs, the specified health action runs to make the condition better. You can specify a health policy to monitor different targets, such as a dynamic cluster or a server. Servers can be simultaneously monitored by multiple health policies. Armed with a set of conditions to look for, WebSphere Virtual Enterprise monitors the environment until a problem is detected, and action is taken.

Parent topic: [Components of dynamic operations](#)

Related concepts:

[Health management](#)
[Overview of work classes](#)

Related tasks:

[Configuring the health controller](#)
[Configuring health management](#)
[Deploying applications with defined service levels](#)
[Defining a service policy](#)

Related reference:

[Routing and service policies](#)
[Health policy administrative tasks](#)

Elasticity mode

Use elasticity mode to add logic that causes the application placement controller to minimize the number of nodes that are used, as well as remove nodes that are not needed, while still meeting service policy goals. Additionally, you can use elasticity mode to add logic so that when the controller recognizes a particular dynamic cluster is not meeting service policies and has started all possible servers, the controller adds a node.

Dynamic clusters - cluster level (physical machine/PM). Typical or default APC behavior.

1. Starts JVMs when needed.
 - a. to meet minimum instances
 - b. to meet CPU, mem usage per PM usages
2. Stops JVMs when needed.
 - a. to meet maximum instances. Ex: max instance set to 2. if 3 PMs running, APC stops the 3rd
 - b. lazy start/proactive idle stop (custom property)
 - c. resource balancing: to consolidate (shared) resources (CPU/mem) from one PM to another

mins <_ |JVMs| <_ max

Elasticity:

1. Enables APC to have the ability to start, stop nodes. Can add or remove nodes by running scripts. (VE stand-alone)
2. Enables APC to have the ability to start, stop, add, remove nodes at the VM level. (IMP/IWD)

Overview

Elasticity mode enables a WebSphere® cell to grow or shrink dynamically by adding or removing nodes. An elasticity operation defines the runtime behaviors to monitor, and the corrective actions to take when the behaviors are present. As part of the steps for configuring elasticity mode, you create custom actions to define the actions that are associated with the elasticity operations: the Add operation and the Remove operation. The Add operation is issued when all of the resources of the application placement controller are being used, but more resources are still needed to meet the current demand. The Remove operation is issued when the application placement controller has an excessive amount of resources.

Add operation

When elasticity mode is enabled, the application placement controller issues an Add operation when all of the members of a dynamic cluster cannot meet the current demand. Furthermore, the controller attempts to consolidate and start all of the servers on the minimum number of nodes as possible.

When the action that is associated with the Add operation is complete, the controller starts a server on the new node. The new node must be added as a member of the dynamic cluster that requested the addition. If a new node is not added, the controller continues to issue the Add operation until all required resources are received or the demand decreases.

Remove operation

The Remove operation first stops any started dynamic cluster instances before starting the associated actions. It is important to know that if the dynamic cluster is set to manual mode, the Remove operation is issued on any nodes that do not have any started application servers. When elasticity mode is enabled and a node is no longer required to meet the current demand, the application placement controller issues a Remove operation. Any nodes that are not part of any dynamic clusters with no running application servers are first removed. Next, an attempt is made to remove a node that does contain a dynamic cluster instance as long as the instance is not running and no other application servers are running. Finally, an attempt is made to remove nodes that only have one or more started dynamic clusters. The Remove operation occurs only if that node is not required to meet the minimum number of instances for a dynamic cluster, or is not required to meet the current demand.

When the application placement controller is running without elasticity mode enabled, the controller issues start and stop operations for application servers. Servers are started due to an increased demand for CPU or memory, but the servers are not stopped after they are started. When elasticity mode is enabled and the servers are not needed, however, the stop operation is issued and the servers are stopped even after they are started. After all of the servers on the physical machine or virtual machine are stopped, the application placement controller issues the Remove operation.

Considerations when using elasticity mode

Consider the following information when using elasticity mode.

- The application placement controller will not issue a Remove operation on any node that contains a deployment manager or a stand-alone on demand router (ODR).
- Do not enable application lazy start with elasticity mode. The application placement controller issues the Remove operation for all of the nodes on that dynamic cluster. In certain environments, this might cause problems, because all of the custom nodes can then be lost.
- You must configure the application placement controller to always start on the deployment manager or node that will not be removed. Doing so prevents the controller from issuing a Remove operation for the node that the controller is active on. If you do not configure the controller to start on the deployment manager, an attempt to remove the node in which the controller is running might occur. As a result, data can be lost, any actions defined by the Remove operation do not occur, and the runtime tasks in the

administrative console are not properly updated.

- When you use elasticity mode in an environment in which multi-cell performance management is configured, you must configure certain controllers to start on the deployment managers of the center cell and the point cells. Configure the application placement controller to start on the deployment manager of the center cell. Configure the cell agent to start on the deployment managers of the point cells.
- Change the `minTimeBetweenPlacementChange` custom property from 15 minutes to 3 minutes to ensure that the application placement controller does not wait too long to issue an Add operation. If the default value of 15 minutes is used, the controller might issue two Add operations over a period of 30 minutes.

Parent topic: [Product overview](#)

Parent topic: [Configuring elasticity mode](#)

Related tasks:

[Configuring multi-cell performance management](#)

[Configuring elasticity mode](#)

Related reference:

[Scripts](#)

[Custom properties](#)

Extended manageability

Use extended manageability to configure the dynamic operation environment and the visualization capability to understand the operational state of the environment.

The features of WebSphere® Virtual Enterprise apply to the management of the WebSphere Extended Deployment Compute Grid environment and its associated background workload only when the two features are deployed together. Therefore, if you are a customer of WebSphere Extended Deployment Compute Grid, you must have both products to take advantage of the WebSphere Virtual Enterprise features.

Use runtime operations in the administrative console to visualize the inner workings of the WebSphere Extended Deployment environment. You can see what types of decisions are made regarding deployment of applications and allocation of hardware. You can also discover where in a virtual resource pool the applications are running to facilitate debugging, manageability, and resiliency. Use runtime operations to view the current state of the environment and take administrative actions. For instance, you can override decisions made by the provisioner, or modify application priority and availability through the operations console.

Runtime operations are divided into several functions. With the runtime operations navigational group, you can view the status of the WebSphere Virtual Enterprise environment and create reports to view the performance of your environment over time. With the task manager, you can approve and deny system suggestions through a one-click option.

Important: JavaScript must be enabled in your browser for the runtime operations panels to work properly.

Parent topic: [Product overview](#)

Related tasks:

[Monitoring operations](#)

Related information:

[Product overview](#)

Supported middleware server types

You can create representations of servers that run other middleware software in the administrative console. After you create these representations, you can then manage the servers from the administrative console.

Middleware servers encompass all servers in the middleware tier that provide the infrastructure for applications or their data. The steps vary depending on if you are configuring assisted life-cycle management middleware servers or complete life-cycle management middleware servers.

With assisted life-cycle servers, you use templates to create representations of the servers in the administrative console. However, these servers still exist within the administrative domain of their respective middleware platform. You can control the servers operationally, monitor and view server health and performance, and configure the administrative console to display log files and configuration files for these servers. The following server types are supported with assisted life-cycle middleware server support:

- Apache HTTP Server Version 1.3, Version 2.0, Version 2.2.17
- Apache Tomcat Version 4.1.x, Version 5.0.x, Version 5.5.x, Version 6.0, Version 7.0.14
- Apache Geronimo Version 1.0 and Version 1.1
- JBoss Version 4.0.x, Version 4.3.x, Version 5.1.x, Version 6.0.x
- Oracle WebLogic Server Version 8.x, Version 9.x, Version 10.3.5
- External WebSphere® application servers, Version 5.1, Version 6.1
- WebSphere Application Server Community Edition Version 2.0, Version 2.1.1.5, Version 3.0

Complete life-cycle servers include any servers that the environment can instantiate or create. You can control the servers operationally, deploy applications, and monitor and view server health and performance. The following server types are supported with complete life-cycle middleware server support:

- WebSphere Application Server Community Edition Version 2.0 (all releases), Version 2.1.1.5, Version 3.0
- PHP Version 4.x, Version 5.3.6

To support any other server that supports HTTP traffic, you can create a custom HTTP server representation of the server in the administrative console.

Parent topic: [Product overview](#)

Related concepts:

[Middleware agent](#)
[Middleware nodes and servers](#)

Related tasks:

[Adding middleware servers to configurations](#)

Middleware nodes and servers

The term *middleware server* refers to a server on any middleware platform. Middleware servers include the following types: WebSphere® Application Server or WebSphere Virtual Enterprise, Apache Tomcat servers, JBoss servers, BEA WebLogic servers, PHP servers, and so on.

Support in Version 6.1 and later

WebSphere Virtual Enterprise Version 6.1 and later offers enhanced support for environments outside of the product domain. Application servers that run on other middleware platforms are represented more thoroughly in the product administrative domain because the middleware agent is installed on these machines. The application placement controller is capable of managing dynamic clusters that are made up of these server types. Some health management is also supported for the other middleware platform servers.

Middleware agent

The middleware agent is a light weight agent that you can install on the nodes that you want WebSphere Virtual Enterprise to manage. The middleware agent can run on any node. Nodes that run the middleware agent do not need WebSphere Application Server or WebSphere Virtual Enterprise installations on the node. The middleware agent replaces the remote agent.

Important: Middleware agents are not supported on z/OS® systems. You cannot federate middleware agents from distributed platforms into a z/OS deployment manager.

Middleware server types

- **Complete lifecycle management servers**

Complete lifecycle management servers include servers that the product can both create and manage to support application server virtualization. A managed node contains an application server process that runs within the deployment manager cell. The following managed middleware server types are supported:

- WebSphere Application Server related servers, including application servers, on demand routers (ODR), proxy servers, and so on
- PHP servers
- WebSphere Application Server Community Edition 2.0 (all releases)

All servers that are administered by complete lifecycle support must be administered from the WebSphere Virtual Enterprise administrative console. For example, do not administer complete lifecycle WebSphere Application Server Community Edition servers from the WebSphere Application Server Community Edition console.

Create complete lifecycle WebSphere Application Server Community Edition servers from previously unused installations of WebSphere Application Server Community Edition Version 2.0 or Version 2.0.0.1.

- **Assisted lifecycle management servers**

Assisted lifecycle management servers include servers that WebSphere Virtual Enterprise can manage, but are created outside of the WebSphere Virtual Enterprise administrative domain, usually specific to the system with which the server is associated. Install the middleware agent on these hosts and associate them with the WebSphere Virtual Enterprise administrative domain. On configured hosts, WebSphere Virtual Enterprise can then start and stop the servers, route traffic to the servers, prioritize requests to the servers, and so on. WebSphere Virtual Enterprise supplies templates for the following assisted lifecycle middleware server types:

- Apache HTTP Server Version 1.3, Version 2.0, and Version 2.2
- Apache Tomcat Version 4.1.x, Version 5.0.x, Version 5.5.x, and Version 6.0
- Apache Geronimo Version 1.0 and Version 1.1
- JBoss Version 4.0.x
- BEA WebLogic Server Version 8.x, Version 9.x, Version 10.x
- WebSphere Application Server Community Edition
- External WebSphere application servers, Version 5.1 and later. External WebSphere application servers are application servers that run outside of your WebSphere Virtual Enterprise cell.
- Custom HTTP servers

For any other server types, you can develop your own templates.

Administer assisted lifecycle servers from their respective administrative consoles. For example, assisted lifecycle WebSphere Application Server Community Edition servers are administered from the WebSphere Application Server Community Edition console. Any representational revisions must be entered into the WebSphere Virtual Enterprise administrative console.

- **Discovered servers**

Discovered servers are servers that are found by middleware discovery. Middleware discovery can find existing installations of WebSphere Application Server Community Edition and create representations of these servers in the WebSphere Virtual Enterprise administrative console. These servers are represented as assisted lifecycle servers. You can group these servers together into a dynamic cluster, but the dynamic clusters must have manual membership. You cannot create expression-based dynamic clusters of discovered servers. Middleware discovery can also find the applications that are deployed to these servers and represent them in the WebSphere Virtual Enterprise administrative console as unmanaged applications.

Like assisted lifecycle servers, administer discovered servers from their respective administrative consoles. For example, administer WebSphere Application Server Community Edition servers from the WebSphere Application Server Community Edition console. Make any representational revisions in the WebSphere Virtual Enterprise administrative console.

Definition

The support that WebSphere Virtual Enterprise offers differs between complete, discovered, and assisted lifecycle servers.

- For complete lifecycle management middleware servers, define the servers in the administrative console. Management of these servers is the same as in previous releases.
- For assisted lifecycle middleware servers, use the following approach to register the servers with the WebSphere Virtual Enterprise configuration:

Install the middleware agent on the server and manually define the server in the administrative console. Use *server templates* to define the middleware servers. Server templates include the following information:

- Default ports for routing traffic to the server
 - Commands for starting and stopping the server
 - Default list of core configuration files that make up the external server for the external configuration editing service
 - Default list of directories where log files exist for the external log viewing service
- WebSphere Virtual Enterprise is delivered with a set of default server templates for the different middleware server types.
- For discovered servers, after you install the middleware agent, middleware discovery locates existing installations of WebSphere Application Server Community Edition and their installed applications, and creates a representation of these servers and applications in the WebSphere Virtual Enterprise cell. These servers and applications are represented as assisted lifecycle servers and unmanaged applications.

Management

You can manage certain aspects of assisted lifecycle and discovered middleware servers with the administrative console.

- With *server operations*, you can run a Java™ or non-Java executable file on your middleware server from the administrative console.
- With the *log viewer*, you can view the log files for your middleware servers from the administrative console.
- With *external configuration*, you can configure the administrative console so that you can view and edit configuration documents for your middleware servers.

Other WebSphere Virtual Enterprise management features, such as dynamic clusters, health policies, service policies, runtime tasks, reporting, and so on, offer varied support for assisted lifecycle and discovered middleware servers.

Distributed platforms | **z/OS** **Note:** When attempting to remove a middleware agent from a cell using the administrative console, the configuration files are removed from that cell, but the agent process is still running. To remove a middleware agent from a cell, run the **removeAgent.sh|bat** script available in the agent_install_root/bin directory.

Parent topic: [Product overview](#)

Related concepts:

[Middleware agent](#)
[Health management](#)

Related tasks:

[Preparing the hosting environment for dynamic operations](#)
[Federating middleware nodes](#)
[Creating PHP servers and PHP dynamic clusters](#)
[Adding middleware servers to configurations](#)
[Creating middleware server templates](#)
[Configuring the external log viewing service](#)
[Configuring middleware server operations](#)

Middleware agent

The middleware agent is a server process that is used to manage servers that are not running WebSphere® Virtual Enterprise.

The remote agent versus the middleware agent

In WebSphere Extended Deployment Version 6.0.x, limited support for servers that are not running the product was provided. With the remote agent that was provided with the mixed server environment offering, generic server cluster could be configured to route work to these other server types. The remote agent monitors the external node and sends CPU utilization, node speed, and number of CPUs information to the autonomic controllers. With this configuration, application placement is not supported.

WebSphere Virtual Enterprise Version 6.1 extends this support with the *middleware agent*. The middleware agent replaces the remote agent.

Overview

The middleware agent can run on any node. You can install the middleware agent on a node that is running other middleware servers that were created outside of WebSphere Virtual Enterprise. You then can create representations of these nodes and servers and manage them in the administrative console.

Middleware agents have a framework with which they can communicate. The deployment manager can communicate with the middleware agents. The middleware agents must be at the same fix pack level as the deployment manager.

z/OS Important: Middleware agents are not supported on the z/OS® systems. You cannot federate a middleware agents from distributed platforms into a z/OS deployment manager.

Supported server types

When you install the middleware agent on a node, you can then manage the servers on that node with WebSphere Virtual Enterprise. The following server types are supported:

- Apache HTTP Server Version 1.3, Version 2.0, Version 2.2.17
- Apache Tomcat Version 4.1.x, Version 5.0.x, Version 5.5.x, Version 6.0, Version 7.0.14
- Apache Geronimo Version 1.0, Version 1.1
- JBoss Version 4.0.x, Version 4.3.x, Version 5.1.x, Version 6.0.x
- Oracle WebLogic Server Version 8.x, Version 9.x, Version 10.3.5
- WebSphere Application Server Community Edition Version 2.1.1.5, Version 3.0.x
- External WebSphere application servers, Version 5.1, Version 6.1
- PHP Version 4.x, Version 5.3.6

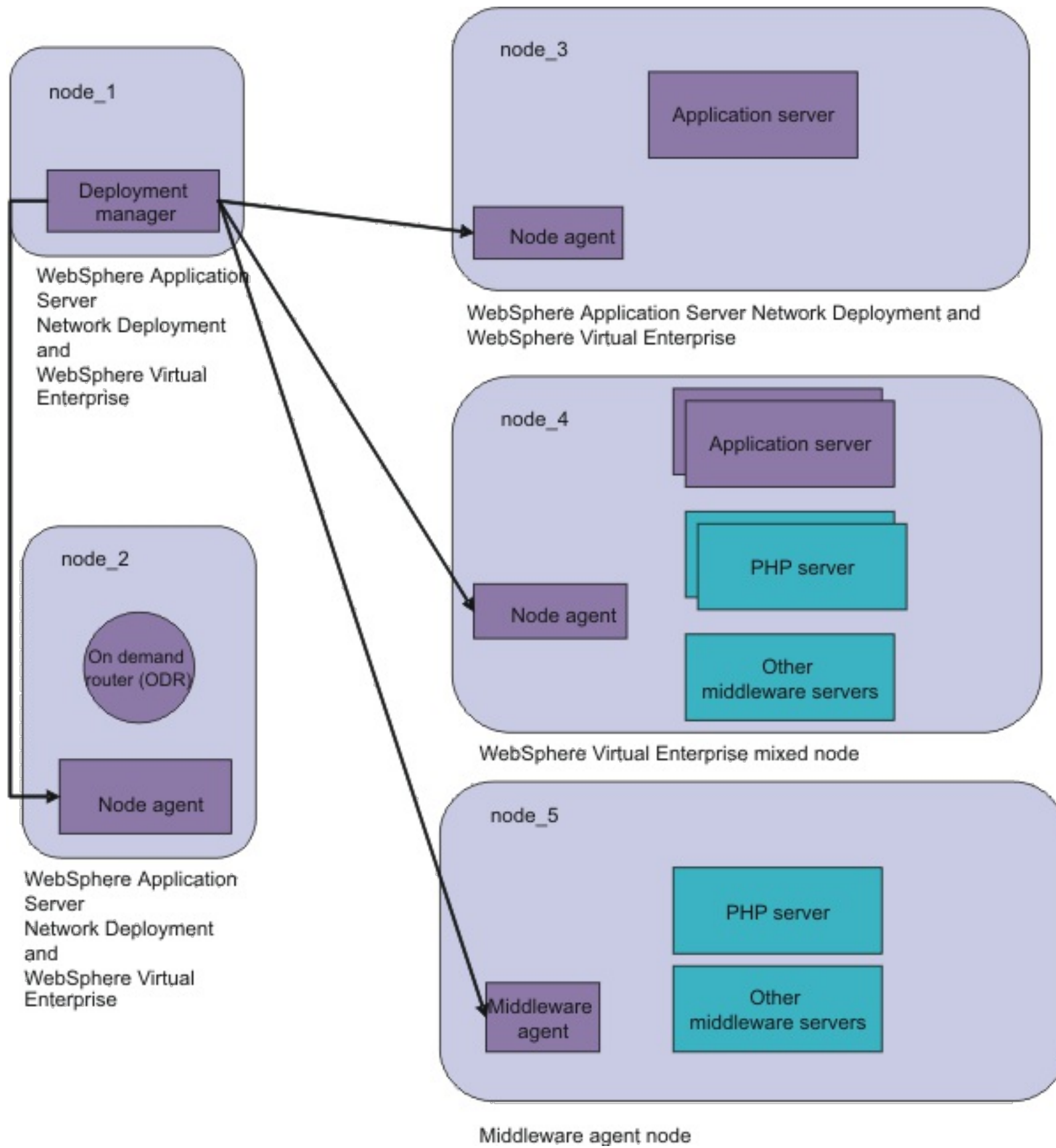
To support any other server that supports HTTP traffic, you can create a custom HTTP server representation of the server in the administrative console.

Sample topologies for WebSphere Virtual Enterprise

The middleware agent is necessary for all of these servers to be managed by WebSphere Virtual Enterprise. The middleware agent is installed automatically on any WebSphere Virtual Enterprise machines. However, in a separate step, you must install the middleware agent on servers that run only other middleware servers. Configure the WebSphere Virtual Enterprise machines, then federate any other middleware servers to the WebSphere Virtual Enterprise configuration.

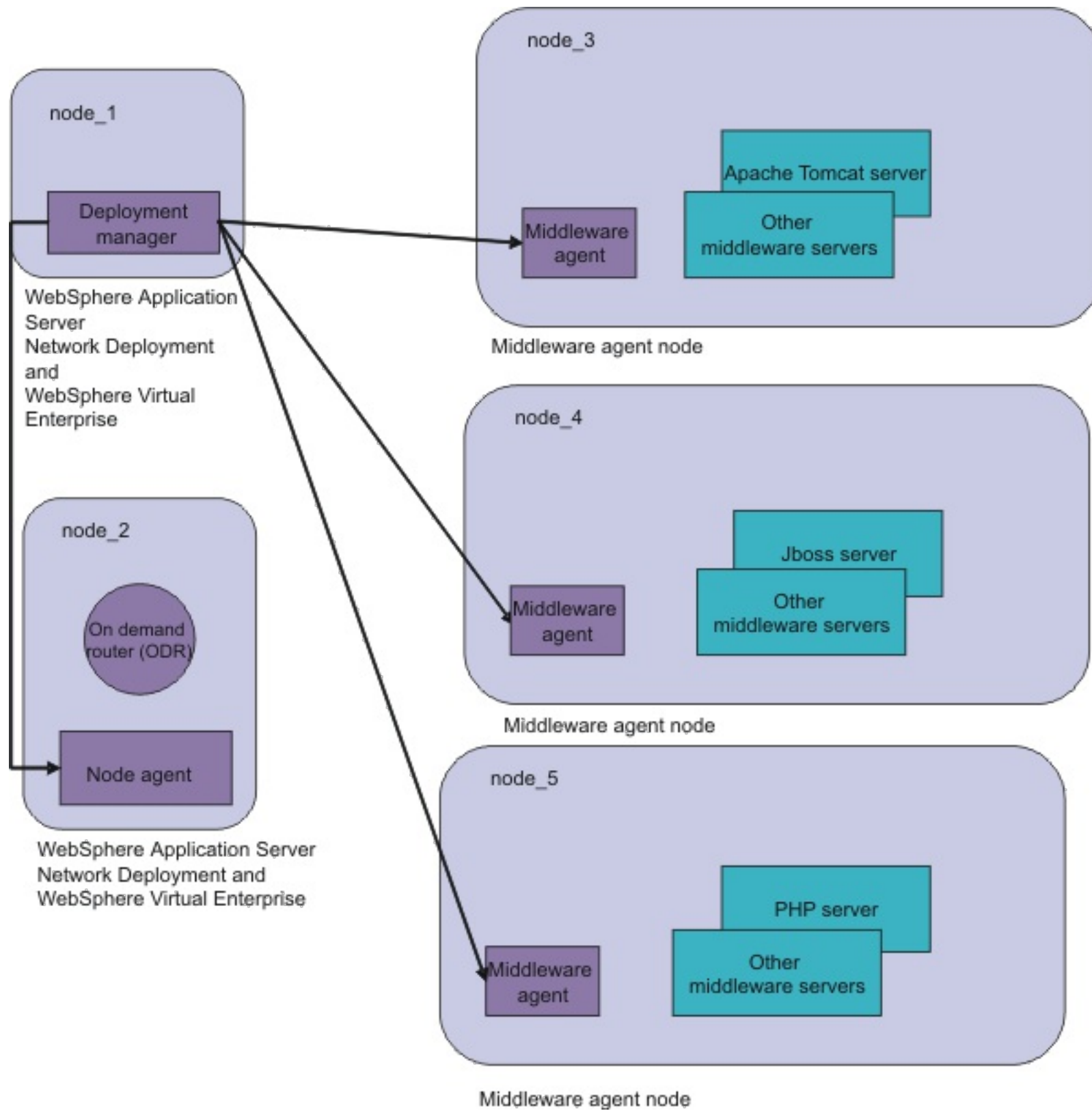
In the following topology example, a mixed environment is displayed. You are running WebSphere Virtual Enterprise machines, other middleware machines that run servers such as Tomcat or PHP, and mixed machines that include both WebSphere Virtual Enterprise and other middleware servers. On any nodes that are running WebSphere Application Server or WebSphere Virtual Enterprise, you can use the existing node agent to manage the node. On nodes that only run other middleware software, install the middleware agent.

Figure 1. Middleware agent topology: The middleware agent introduced into an environment that includes WebSphere products and other middleware software



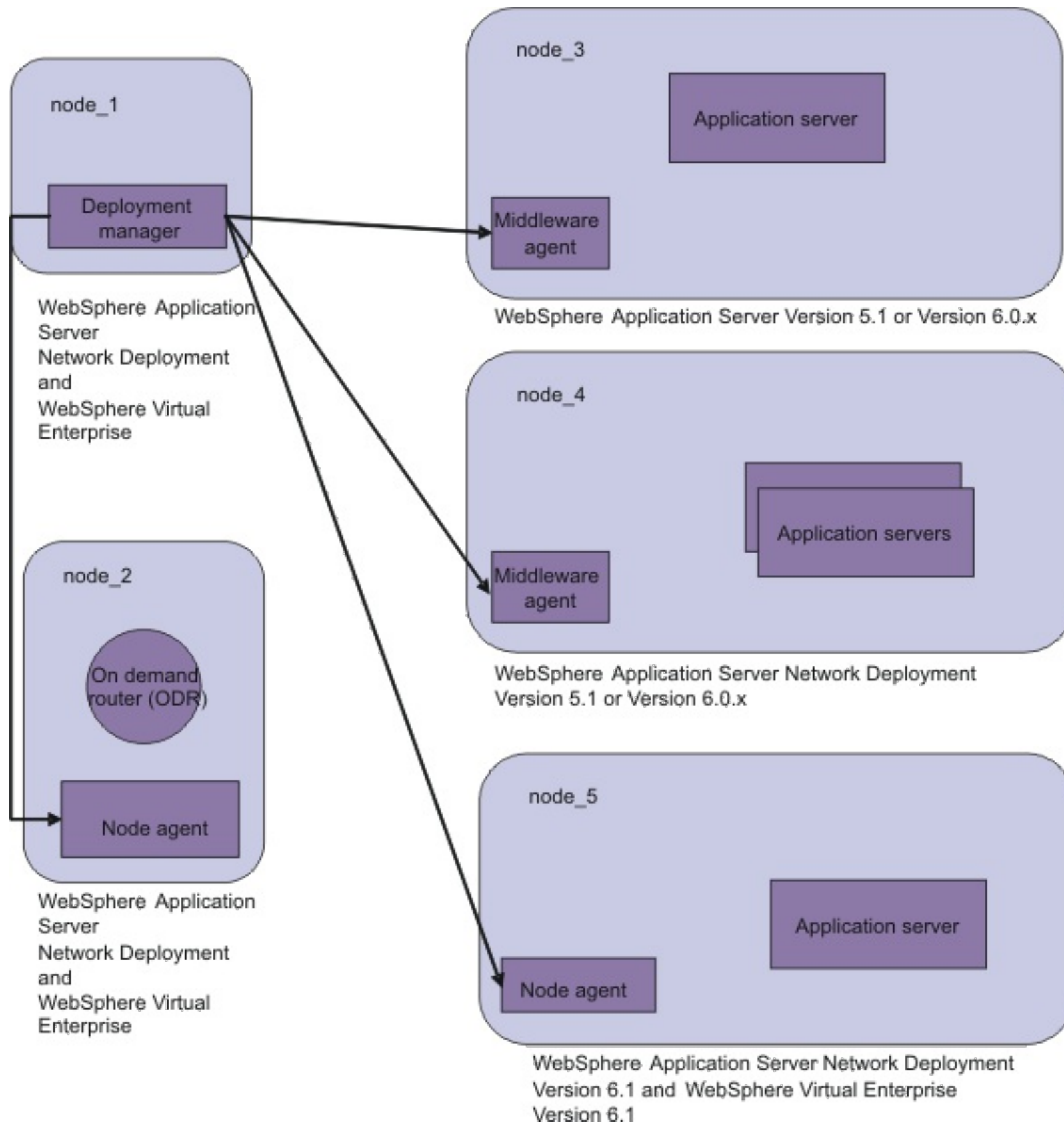
In the following topology example, WebSphere Virtual Enterprise is introduced into an environment that was originally running only other middleware server types. You install WebSphere Application Server Network Deployment and WebSphere Virtual Enterprise on two nodes. These nodes become the deployment manager and on demand router nodes. On your existing nodes that run other middleware servers, you install the middleware agent.

Figure 2. Middleware agent topology: Extended Deployment is introduced into an environment that did not previously use any WebSphere products



In the following topology example, some nodes are running previous versions of WebSphere Application Server. The middleware agent is required on any node that is not federated to the deployment manager. In the following example, node_3 and node_4 are not federated to the node_1 deployment manager, and have the middleware agent installed.

Figure 3. Middleware agent topology: Multiple versions of WebSphere Virtual Enterprise and WebSphere Application Server



Parent topic: [Product overview](#)

Related tasks:

[Preparing the hosting environment for dynamic operations](#)

[Federating middleware nodes](#)

[Creating PHP servers and PHP dynamic clusters](#)

[Adding middleware servers to configurations](#)

Installing and customizing the product

This section describes how to install the product in new and existing environments. The information includes planning, preparing for, completing, and maintaining product installations, including configuring the product after installation.

About this task

WebSphere® Virtual Enterprise represents on demand capabilities in an extended environment. Use the product to enhance operational efficiency by deploying dynamic operations, service high-volume transactional workloads with linear scalability and near 100% availability, or manage large scale, continuously available WebSphere Application Server environments. Use the links provided in this topic to learn about the types of installation available.

[Hardware and software requirements](#)

Review the hardware and software requirements before you install the product to ensure that an enhanced quality of service is provided in your environments.

[Planning the product installation](#)

Use this information to install and configure product packages on AIX®, HP-UX, Linux, Solaris, and Windows operating systems.

z/OS

[Customizing WebSphere Virtual Enterprise for z/OS](#)

Using the WebSphere Customization Toolbox, you can generate and run customized jobs to customize WebSphere Virtual Enterprise for z/OS®.

Related tasks:

[Migrating WebSphere Virtual Enterprise](#)

[Preparing the hosting environment for dynamic operations](#)

[Deploying and managing application editions without loss of service](#)

[Managing the WebSphere Virtual Enterprise environment](#)

[Monitoring operations](#)

[Configuring WebSphere Virtual Enterprise to work with other IBM products](#)

Related reference:

[Reference](#)

Related information:

[IBM WebSphere Virtual Enterprise Version 7.0](#)

[Product overview](#)

[Site map](#)

Hardware and software requirements

Review the hardware and software requirements before you install the product to ensure that an enhanced quality of service is provided in your environments.

For a detailed list of supported hardware and software options by operating system, see the [System requirements](#). If there is a conflict between the information provided in the information center and the information on the System Requirements page, the information at the web site takes precedence.

Parent topic: [Installing and customizing the product](#)

Related tasks:

[Planning the product installation](#)

[Customizing WebSphere Virtual Enterprise for z/OS](#)

Planning the product installation

Use this information to install and configure product packages on AIX®, HP-UX, Linux, Solaris, and Windows operating systems.

Before you begin

Familiarize yourself with the minimum hardware and software requirements for the product.

About this task

As part of the installation process, review common installation scenarios, determine your product topology, and prepare your existing environments to begin establishing application serving capability and extended manageability.

[Using Installation Manager to install the product](#)

Starting with Version 7.0.0.1, you can use Installation Manager to install WebSphere® Virtual Enterprise with WebSphere Application Server Network Deployment Version 8.0.0.x.

[Using ISMP programs to install the product](#)

You can use programs based on InstallShield MultiPlatform (ISMP) to install and update WebSphere Virtual Enterprise Version 7.0.0.

Parent topic: [Installing and customizing the product](#)

Related concepts:

[Hardware and software requirements](#)

Using Installation Manager to install the product

Starting with Version 7.0.0.1, you can use Installation Manager to install WebSphere® Virtual Enterprise with WebSphere Application Server Network Deployment Version 8.0.0.x.

Before you begin

- Install the latest WebSphere Application Server Network Deployment fix pack if you are installing the product in an existing WebSphere Application Server Network Deployment environment. See [Recommended fixes for WebSphere Application Server](#) for more information.
- Install Installation Manager using the product media, using a file obtained from the Passport Advantage® site, or using a file containing the most current version of Installation Manager from the [IBM® Installation Manager web site](#).

Note: If you do not have a Passport Advantage ID and password, you must install the product from the product repositories on the media or local repositories.

About this task

WebSphere Virtual Enterprise Version 7.0.0.1 is the first version to be installed by IBM Installation Manager rather than by the programs based on InstallShield MultiPlatform (ISMP) that are used to install, update, and uninstall previous versions. Installation Manager is a single installation program that can use remote or local software flat-file repositories to install, modify, or update the product. It determines and shows available packages, including products, fix packs, interim fixes, and so on, checks prerequisites and interdependencies, and installs the selected packages. You also use Installation Manager to easily uninstall the packages that it installed.

You can use one of the three following modes to install Installation Manager:

- In admin mode, Installation Manager is installed from an administrator or a root ID and can be invoked by any administrator or root user.
- In nonAdmin mode, which is also called user mode, Installation Manager can be invoked only by the user who installed it.
- In group mode, Installation Manager can be invoked by any user ID that is connected to the default group of the user who installed it.

Each software product that you can install by using Installation Manager is referred to as a package. An installed package has a product level and an installation location. A package group consists of all the products that are installed at a single location.

Procedure

1. Click **Install** on the main Installation Manager page to start the wizard.
2. Configure the Installation Manager repository. Click **File Preferences**.
 - a. Select **Repositories** from the list, select the WebSphere Application Server Network Deployment Version 8.0.0.x repository, and click **Add Repository**.
3. On the Install Packages panel, select the WebSphere Application Server Network Deployment Version 8.0.0.x package, and click **Next**.
4. Accept the license agreement.
5. Define the package group directory.
6. Define the installation directory.
7. Define the features, such as the language and the product features, to install.
8. Review the installation summary, and click **Install**.
9. Select the Profile Management Tool from the list of programs to start, and click **Finish**.
10. On the WebSphere Customization Toolbox Welcome panel, select the Profile Management Tool, and click **Launch Selected Tool**.
11. Add WebSphere Virtual Enterprise to the Installation Manager repository. Select the FTP URL from the list, and click **Install**. A list of the packages for WebSphere Virtual Enterprise is displayed.
12. Select the WebSphere Virtual Enterprise package that you want to install, and click **Install**.
13. Repeat steps 4 through 8 to complete the installation of WebSphere Virtual Enterprise.

When defining the package group directory, select the **Use the existing package group** option since you previously installed WebSphere Application Server Network Deployment. A package group is a location that contains one or more packages.

If the installation fails, select the **Delete fetched files** option, and click **OK**.

What to do next

Configure each node to meet your objectives, which you can do either from the administrative console or by using the wsadmin tool.

- Click **Servers > All servers** to configure each node from the administrative console.
- See [Scripts](#) to use the wsadmin tool to configure each node.

Using a response file to install the product

You can install the product using Installation Manager response files.

Parent topic: [Planning the product installation](#)

Related information:

 [Installation Manager Version 1.4 Information Center](#)

 [Installation Manager Version 1.5 Information Center](#)

Using a response file to install the product

You can install the product using Installation Manager response files.

Before you begin

- Install the latest WebSphere® Application Server Network Deployment fix pack if you are installing the product in an existing WebSphere Application Server Network Deployment environment. See [Recommended fixes for WebSphere Application Server](#) for more information.
- Install Installation Manager using the product media, using a file obtained from the Passport Advantage® site, or using a file containing the most current version of Installation Manager from the [IBM® Installation Manager web site](#).

Note: If you do not have a Passport Advantage ID and password, you must install the product from the product repositories on the media or local repositories.

About this task

You can work with Installation Manager response files in the following ways to silently install the product:

- Record a response file to install the product.
- Manually create a new response file.
- Modify an existing response file.

Procedure

1. Optional: Record a response file to install the product.
 - a. From the eclipse subdirectory of the directory in which you installed Installation Manager, start the tool using the -record option as shown in the following examples:

- **Windows** Administrator or non-administrator:

```
IBMIM.exe -skipInstall "C:\temp\imRegistry" -record  
C:\temp\install_response_file.xml
```

- **AIX** **HP-UX** **Solaris** **Linux** Administrator:

```
./IBMIM -skipInstall /var/temp/imRegistry -record  
/var/temp/install_response_file.xml
```

- **AIX** **HP-UX** **Solaris** **Linux** Non-administrator:

```
./IBMIM -skipInstall user_home/var/temp/imRegistry -record  
user_home/var/temp/install_response_file.xml
```

Specify the -skipInstall parameter to speed up the recording since no files are actually installed. If you use a temporary data location with the -skipInstall parameter, the installation registry is written to the specified data location while recording. When you start Installation Manager again without specifying the -skipInstall parameter, you can use your response file to install against the actual installation registry.

Do not use the -skipInstall parameter on the actual agent data location that is used by the Installation Manager. Use a clean, writeable location and reuse that location for future recording sessions.

- b. Add the repositories of your choice to your Installation Manager preferences.
 - i. In the top menu, click **File > Preferences**, and select **Repositories**.
 - ii. For each repository, click **Add Repository**, and enter the path to the repository.config file in the remote web-based repository or the local directory in which you extracted the repository files. See the following examples.

- Examples of remote repositories:

```
https://downloads.mycorp.com:8080/WVE_70_repository
```

or

```
http://www.ibm.com/software/repositorymanager/com.ibm.websphere.WVE.v70
```

- Examples of local repositories:

```
C:\repositories\wve70\local-repositories
```

or

```
/var/repositories/wve70/local-repositories
```

- iii. Complete the remaining prompts.
- c. Click **Install**. If you are prompted to authenticate, use your registered IBM ID and password. Installation Manager searches the defined repositories for available packages.
- d. Complete the following actions.
 - i. Select **IBM WebSphere Virtual Enterprise** and the appropriate product version, and click **Next**.
 - ii. Accept the license agreement.
 - iii. Select WebSphere Application Server Network Deployment Version 8.0.0.1 or later as the package group, and click **Next**.
 - iv. On the feature panel, click **Next**.
 - v. Review the installation summary, and click **Install**.
 - vi. Click **Finish**.
 - vii. Click **File > Exit** to close the tool.
 - viii. **Optional:** If you are using an authenticated remote repository, create a keyring file for the silent installation. You can store credentials for URLs that require authentication, such as remote repositories.

1. From the eclipse subdirectory of the directory in which you installed Installation Manager, start the tool using the -record option as shown in the following examples:

- **Windows** Administrator or non-administrator:

```
IBMIM.exe -skipInstall "C:\temp\imRegistry" -keyring  
C:\IM\im.keyring -record  
C:\temp\keyring_response_file.xml
```

- **AIX** **HP-UX** **Solaris** **Linux** Administrator:

```
./IBMIM -skipInstall /var/temp/imRegistry -keyring  
/var/IM/im.keyring -record  
/var/temp/keyring_response_file.xml
```

- **AIX** **HP-UX** **Solaris** **Linux** Non-administrator:

```
./IBMIM -skipInstall user_home/var/temp/imRegistry -  
keyring user_home/var/IM/im.keyring -record  
user_home/var/temp/keyring_response_file.xml
```

2. When prompted, enter your credentials for the authenticated remote repository and save them.
3. Click **File > Exit** to close the tool.

2. Use the response files to install the product.

- a. Optional: Use the response file to silently install the keyring file. From the eclipse/tools subdirectory of the directory in which you installed Installation Manager, run the **imcl** command as shown in the following examples:

What to do next

Configure each node to meet your objectives, which you can do either from the administrative console or by using the wsadmin tool.

- Click **Servers > All servers** to configure each node from the administrative console.
- See [Scripts](#) to use the wsadmin tool to configure each node.

Parent topic: [Using Installation Manager to install the product](#)


Related information:

- [Installation Manager Version 1.4 Information Center](#)
- [Installation Manager Version 1.5 Information Center](#)

Using ISMP programs to install the product

You can use programs based on InstallShield MultiPlatform (ISMP) to install and update WebSphere® Virtual Enterprise Version 7.0.0.

Before you begin

- Read the readme file, which is in the root directory of the product media.
- Install the latest WebSphere Application Server Network Deployment fix pack if you are installing the product in an existing WebSphere Application Server Network Deployment environment. See [Recommended fixes for WebSphere Application Server](#) for more information.
- Exit the WebSphere Application Server Network Deployment directory. Installation errors can occur if you inadvertently leave a command line or browser instance open in the `profile_root/installedApps` directory.
-  See the Program Directory on the [Library page](#) for z/OS® information.

Procedure

1. Stop all the processes that are running on the workstation that you are preparing for installation.

Run the following command to stop the deployment manager:

- o     `./stopManager.sh`
- o  `stopManager.bat`

Run the following command to stop the nodes:

- o     `./stopNode.sh`
- o  `stopNode.bat`

2. Run the following command to start the installation wizard:

- o     `install`
- o  `install.bat`

3. Complete all of the required fields on the wizard panels.

Optionally, select the check box to add the product to the centralized installation manager repository. Click **Browse** to specify the location of the repository. The option to add the product to the centralized installation manager repository is unavailable if you previously chose not to install the repository when you installed Virtual Enterprise.

4. Repeat the process to install the product on each node in the cell.

Tip: Use the same CD or DVD to install the product on the nodes.

Results

The installation wizard installs the product files in `install_root` directory.

What to do next

Configure each node to meet your objectives, which you can do either from the administrative console or by using the `wsadmin` tool.

- Click **Servers** > **All servers** to configure each node from the administrative console.
- See [Scripts](#) to use the `wsadmin` tool to configure each node.

[Installation scenarios](#)

You can install WebSphere Virtual Enterprise in various ways to enhance your application server environment. Depending on your existing environment, the installation process includes certain features that might not be available in other installation scenarios.

[Installing in a WebSphere Application Server Network Deployment environment](#)

WebSphere Application Server Network Deployment is an essential prerequisite for a successful product installation. Before you begin the installation process, you must prepare your WebSphere Application Server Network Deployment environment.

[Installing the middleware agent](#)

Install the middleware agent on a node that is running other middleware servers created outside of the product environment. You can create representations of these nodes and servers and manage them in the administrative console.

[Installing the product silently](#)

Use either a fully-qualified response file, which you configure specifically to your needs, or parameters that pass to the command line to silently install the product and start the nodes.

[Installation parameters](#)

Specify parameters at the command line to customize and configure your product installation.

[Using the centralized installation manager repository tool to add products](#)

Use the centralized installation manager repository tool to add one or more WebSphere Extended

Deployment products to the repository. From the administrative console, you can use the centralized installation manager to install your added components from the repository to the nodes.

[Using the Installation Factory plug-in to create and install customized packages](#)

Use the Installation Factory plug-in to create a customized installation package (CIP) or an integrated installation package (IIP). A CIP contains a single product installation package and various optional assets. An IIP combines one or more installation packages into a single installation workflow that you design.

[Creating and augmenting profiles](#)

After you install the product, create unique profile types and augment existing profiles for WebSphere Virtual Enterprise.

[Uninstalling the product](#)

You can uninstall the product and any maintenance files for the product that are installed in your environment. Complete the uninstallation process either silently or by following the graphical user interface (GUI).

Parent topic: [Planning the product installation](#)

Installation scenarios

You can install WebSphere® Virtual Enterprise in various ways to enhance your application server environment. Depending on your existing environment, the installation process includes certain features that might not be available in other installation scenarios.

This topic provides common installation scenarios for the product. The installation scenarios include descriptions to help you determine what components to install for a given topology. For optimal quality of service and extended manageability, review the scenarios to determine which topology best fits your needs.

WebSphere Virtual Enterprise does not change any Programming Interfaces (APIs) or binary files of WebSphere Application Server Network Deployment. Therefore, if an application works correctly with WebSphere Application Server Network Deployment, it will also work correctly with WebSphere Virtual Enterprise. You might be limited in the use of some WebSphere Virtual Enterprise features with particular applications, however, but any limitations depend on the design, performance, and quality of service requirements for the application. Most limitations can be overcome by tuning your topology or reconfiguring your applications.

For example, assume that you cannot stop an application safely without disrupting service, and you cannot tolerate a service disruption during peak hours. Although WebSphere Virtual Enterprise quiesces your application servers before stopping them, sessions can still be lost if the application does not have sessions that can be serialized and if sessions remain after the maximum quiesce period has timed out. In this case, you can work around this issue by not using dynamic clusters in automatic mode or by tuning the dynamic clusters appropriately.

You can install WebSphere Virtual Enterprise in an environment that contains WebSphere Application Server Network Deployment and either Compute Grid, eXtreme Scale, or both. During installation, the product automatically identifies the WebSphere Application Server Network Deployment installation and extends the functionality to embrace the themes and added function of the product. The installation of the Virtual Enterprise controller applies to any WebSphere Application Server node that contains either an on demand router or a deployment manager that is augmented with Virtual Enterprise. The installation of the Virtual Enterprise node applies to Virtual Enterprise installed on any other WebSphere Application Server node or other middleware server node with the Virtual Enterprise middleware agent. The middleware agent is a server process that is used to manage other middleware servers, such as Tomcat or PHP.

When you install Virtual Enterprise in environments in which other middleware servers are running, the installation wizard does not present the option to add the product to the centralized installation manager repository. However, you can add the product later by using the centralized installation manager tool. See [Using the centralized installation manager repository tool to add products](#) for more information.

When you integrate eXtreme Scale with an existing installation of WebSphere Application Server Network Deployment and Virtual Enterprise, you can install both the eXtreme Scale client and server, or you can install the client only. See the WebSphere eXtreme Scale Information Center for more information.

[Virtual Enterprise topology](#)

Install the product first as a simple topology configuration to familiarize yourself with its features in a test environment. When you are familiar with the advanced features, you can design advanced topologies that apply specifically to your application server environment.

Parent topic: [Using ISMP programs to install the product](#)

Virtual Enterprise topology

Install the product first as a simple topology configuration to familiarize yourself with its features in a test environment. When you are familiar with the advanced features, you can design advanced topologies that apply specifically to your application server environment.

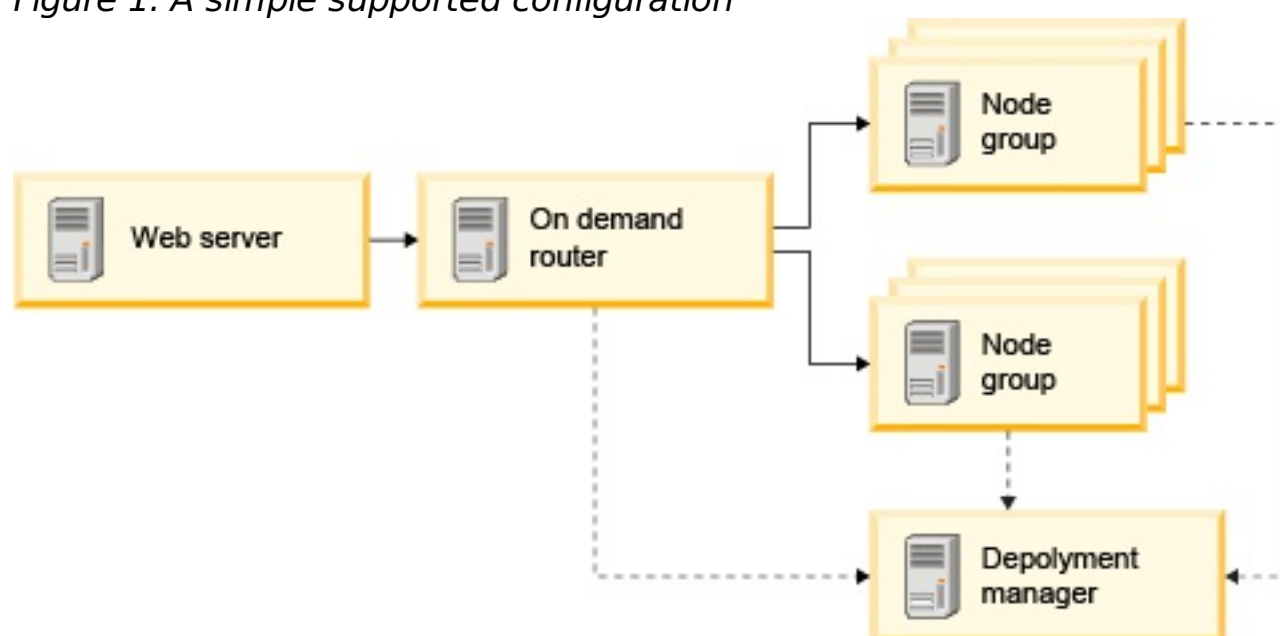
The following graphic represents a supported topology of the product. In this topology, the on demand router is between the Web server and the node groups. The ODR, with the assistance of the autonomic managers, categorizes and prioritizes the work before routing work to the appropriate nodes. The deployment manager is the centralized management communication vehicle, as indicated by the dotted lines.

Install the deployment manager on a separate workstation from the application server nodes so that an outage of a single node, either planned or unplanned, does not impact both the application serving capacity and management capability.

When you are using a single physical computer that is hosting multiple nodes, the deployment manager can run on one of the nodes, and application servers can run on the other nodes. The application placement controller has been updated to consider the placement decisions made by the deployment manager.

When you are familiar with the product in a simple topology, you can add layers of complexity to your test environment. Include a firewall to secure your environment, and stress test to see the product features in action. Typically, when the workload increases, you must ensure that you have enough resources to manage the workload and that the resources configure and function together. You can also include an Internet Protocol (IP) sprayer in the configuration to eliminate any single point of failure, and make the topology highly available. An IP sprayer distributes requests from one IP address to multiple IP addresses.

Figure 1. A simple supported configuration



Parent topic: [Installation scenarios](#)

Related concepts:

[Hardware and software requirements](#)

Related tasks:

[Installing in a WebSphere Application Server Network Deployment environment](#)

[Using ISMP programs to install the product](#)

Installing in a WebSphere Application Server Network Deployment environment

WebSphere® Application Server Network Deployment is an essential prerequisite for a successful product installation. Before you begin the installation process, you must prepare your WebSphere Application Server Network Deployment environment.





Before you begin

- See the [WebSphere Application Server detailed system requirements](#) for a detailed list of supported hardware and software options by operating system.
- See the [WebSphere Application Server Library page](#) to access the latest online documentation.

About this task

A basic topology consists of a single deployment manager, a single on demand router (ODR), and two application servers. Use three workstations (A, B, and C) for this basic topology. Set up the deployment manager and the nodes with one workstation by combining the requirements of workstations A and B if it is a high-power workstation. Otherwise, set up workstations A and B separately for optimal performance. Set up the deployment manager first.

Procedure

1. Set up workstation A, the deployment manager:
 - a. Install WebSphere Application Server Network Deployment.
 - b. Install the latest WebSphere Application Server Network Deployment fix pack to update your product level. See [Recommended fixes for WebSphere Application Server](#) for more information.
 - c. Create a deployment manager profile.
 - d. Run the **stopManager** command to stop the deployment manager in preparation for a product installation.
2. Set up workstation B, the managed nodes:
 - a. Install WebSphere Application Server Network Deployment and the latest fix pack to update your product level.
 - b. Create two custom profiles, and federate the node within each custom profile into the deployment manager cell.
 - c. Run the **stopNode** command to stop each node in the node agent process in preparation for a product installation.
3. Set up workstation C, the on demand router (ODR):
 - a. Install WebSphere Application Server Network Deployment and the latest fix pack to update your product level.
 - b. Create a custom profile, and federate the node within the custom profile into the deployment manager cell.
 - c. Run the **stopNode** command, and verify that the node stopped in preparation for a product installation.
4.     Increase the size of the file limit of your operating system:
 - a. Run the `ulimit -n 8192` command.
 - b. Run the `iscdeploy.sh -restore` command from the `was_home/bin/` directory

Results

Your WebSphere Application Server Network Deployment environment is now prepared to support the installation of WebSphere Virtual Enterprise.

What to do next

Install the product on the deployment manager and on each managed node in the cell if the nodes do not share the same installation location. Before you begin the installation process, however, you must restart the deployment manager and each of the nodes to which you plan to install the product. To start the deployment manager, run the following command from the `profile_root/bin` directory of the WebSphere Application Server Network Deployment installation:

-  `startManager.bat`
-     `./startManager.sh`

To start each of the nodes, run the following command from the `profile_root/bin` directory of the WebSphere Application Server Network Deployment installation:

-  `startNode.bat`
-     `./startNode.sh`

Parent topic: [Using ISMP programs to install the product](#)

Related tasks:

[Using ISMP programs to install the product](#)

Related information:

 [WebSphere Application Server Library page](#)

- [WebSphere Application Server system requirements](#)
- [Recommended fixes for WebSphere Application Server](#)

Installing the middleware agent

Install the middleware agent on a node that is running other middleware servers created outside of the product environment. You can create representations of these nodes and servers and manage them in the administrative console.

Before you begin

Augment the deployment manager profile with the product. For more information, see [Creating and augmenting profiles](#).


About this task

The middleware agent is a server process that is used to manage servers that are not running the product. When you install a WebSphere Virtual Enterprise fix pack to your deployment manager, you must also apply the fix pack to your middleware agent installations. The deployment manager and the middleware agents must be at the same fix pack level. Complete the following steps to install the middleware agent on the nodes.

Procedure

1. Run the following command to start the installer:

- o     `./install.sh`
- o  `install.bat`

 **Attention:** By default, the installation wizard selects the appropriate version of the middleware agent to install based on the version of the operating system. For example, if the wizard detects a 64-bit operating system, the wizard automatically installs a 64-bit version of the middleware agent. To override the default selection and install a 32-bit version of the middleware agent on a 64-bit operating system, you can either use the graphical user interface (GUI) or complete the silent installation process. Run the following command to start the installer:

- o `./install.aix.ppc32`
- o `./install.aix.ppc32 -silent -options`

2. Complete all of the required fields on the wizard panels.

Results

The installation wizard installs the middleware agent files to an empty directory that does not contain any existing files.

What to do next

You can further configure the middleware agent with custom properties. See [Middleware agent custom properties](#) for more information.

Next topic: [Adding middleware servers to configurations](#)

Next topic: [Middleware agent custom properties](#)

Parent topic: [Using ISMP programs to install the product](#)

Related concepts:

[Middleware agent](#)

Related tasks:

[Updating the middleware agent](#)

[Adding middleware servers to configurations](#)

Installing the product silently

Use either a fully-qualified response file, which you configure specifically to your needs, or parameters that pass to the command line to silently install the product and start the nodes.

Before you begin

- Stop all processes that are running in your WebSphere® Application Server Network Deployment environment. Stop all the nodes and the node agent, but do not stop the entire cell. See [Command-line utilities](#) for more information.

About this task

A silent installation uses the same installation program that the graphical user interface (GUI) version uses. However, instead of displaying a wizard interface, the silent installation reads all of your responses from a file that you customize, or from parameters that you pass to the command line.

Procedure

1. Optional: If you choose to install the product using a response file, first customize the file.
 - a. Copy the response file from the product CD or DVD to your disk drive.
 - b. Open and edit the response file in the text editor of your choice. You must specify the following parameters:
 - The license agreement
 - The location of the product installation
 - The list of profiles to augment
 - The location of the centralized installation manager repository

Tip: The installer uses the location that you select for your installation to determine where the existing instance of WebSphere Application Server Network Deployment is installed. If you install on a node with multiple WebSphere Application Server Network Deployment instances, clearly define your location.

- c. Run the following script to start the installation.

```
./install.sh|bat -options C:/drive_path/response_file.txt -silent
```

The log files are located in the `install_root/logs/xd_operations/install` directory. If the installation fails before the installation location is located or created, the log files are located in the `user_home/wxd_install_logs` directory. For example, if you log in as root, `user_home` is the `/root` directory.

If you install WebSphere Virtual Enterprise in a Network Deployment environment, you can install the centralized installation manager repository, or include the product in the existing repository by further customizing the response file.

2. Optional: If you choose to install the product by passing certain parameters to the command line, run the following script to start the installation:

```
./install.sh|bat -silent -OPT silentInstallLicenseAcceptance=true -OPT installLocation=install_location
```

Results

You successfully installed the product, and started the nodes without any errors occurring.

What to do next

Configure each node to meet your objectives, which you can do either from the administrative console or by using the wsadmin tool.

- Click **Servers > All servers** to configure each node from the administrative console.
- See [Scripts](#) to use the wsadmin tool to configure each node.

Parent topic: [Using ISMP programs to install the product](#)

Related concepts:

[Hardware and software requirements](#)

Related tasks:

[Creating and augmenting profiles](#)

[Preparing the hosting environment for dynamic operations](#)

Related reference:

[Installation parameters](#)

Installation parameters

Specify parameters at the command line to customize and configure your product installation.

Parameters

You can pass the following parameters during a command-line or options file installation of the product:

-silent

Suppresses the graphical user interface (GUI). Specify the **-options** parameter to indicate that the installer completes the installation according to a customized options file. If you do not specify the **-options** parameter, the default values are used instead.

Example usage

```
./install.sh|bat -silent -options options_file.txt
```

-options path_name/file_name

Specifies an options file that the installer uses to complete a silent installation. Properties on the command line take precedence.

Example usage

```
./install.sh|bat -options c:/path_name/options_file.txt
```

-log # !file_name @event_type

Generates an installation log file that logs the following event types:

- err
- wrn
- msg1
- msg2
- dbg
- ALL

Example usage

```
./install.sh|bat -log # !c:/temp/logfiles.txt @ALL
```

-is:log path_name/file_name

Creates a log file that contains the Java™ Virtual Machine (JVM) searches of the installer while attempting to start the GUI. The log file is not created unless specified.

Example usage

```
./install.sh|bat -is:log c:/logs/javalog.txt
```

-is:javaconsole

Displays a console window during the installation process.

Example usage

```
./install.sh|bat -is:javaconsole
```

-is:silent

Suppresses the Java initialization window that is typically displayed as the installer starts.

Example usage

```
./install.sh|bat -is:silent
```

-is:tempdir path_name

Specifies the temporary directory that the installer uses during the installation.

Example usage

```
./install.sh|bat -is:tempdir c:/temp
```

-OPT installCimgrRepository=true -OPT cimgrInstallLocation=installation_location/cimgr

Installs the centralized installation manager repository to the specified installation location.

Example usage

```
./install.sh|bat -OPT installCimgrRepository=true -OPT  
cimgrInstallLocation=c:/mydirectory/cimgr
```

-OPT uninstallCimgr=true -silent

Uninstalls the centralized installation manager repository in silent mode.

Example usage

```
./uninstall.sh|bat -OPT uninstallCimgr=true -silent
```

-OPT profileAugmentList=*profiles_separated_by_commas*

Specifies certain profiles to augment.

Example usage

```
./install.sh|bat -OPT profileAugmentList=profile01, profile02
```

Parent topic: [Installing the product silently](#)

Parent topic: [Using ISMP programs to install the product](#)






Using the centralized installation manager repository tool to add products

Use the centralized installation manager repository tool to add one or more WebSphere® Extended Deployment products to the repository. From the administrative console, you can use the centralized installation manager to install your added components from the repository to the nodes.

Before you begin

- To populate the repository for the first time, ensure that the destination directory is empty. Additionally, you can add a product to the repository only if the repository does not already contain that specific product.
- Use the centralized installation manager repository tool to add the WebSphere Virtual Enterprise installation image to the repository if you chose not to add it when prompted by the installation wizard. See [Using ISMP programs to install the product](#) for more information.
- The repository tool cannot be used to add the required binary files for WebSphere Application Server Community Edition to the repository. To use the centralized installation manager to install WebSphere Application Server Community Edition on remote servers, you must follow the steps that are outlined in [Downloading and installing WebSphere Application Server Community Edition](#).

Procedure

1. Start the repository tool from the following location:
 -     `install_root/bin/xd_cimgrrepository.sh`
 -  `install_root/bin/xd_cimgrrepository.bat`
2. Select one or more products to include in the repository, and click **Next**.
3. Click **Browse** to provide the location of the product installation image.

If you selected to add multiple products in step 2, the panel is displayed again to prompt you for the installation image location for each product.

If you plan to set up a high availability deployment manager environment, you must specify the location of the repository from the current installation location. For example, if you install the product in the C:\ProgramFiles\IBM\WebSphere\AppServer directory, specify the repository location C:\ProgramFiles\IBM\WebSphere\AppServer\repository_folder.

4. Optional: You can also install a customized installation package (CIP) to the centralized installation manager repository. Manually copy the installation descriptors from the CIP to the `install_root/properties/xdcimgr` directory on the deployment manager workstation:
 - For a Virtual Enterprise CIP:
`wxdop_package/custom/InstallPackageXD61*.xml`
 - For a Compute Grid CIP:
`wxdcg_package/custom/InstallPackageXD61*.xml`
 - For an eXtreme Scale Version 6.1.0.x only CIP:
`wxddg_package/custom/InstallPackageXD61*.xml`

Attention: While you can add a CIP to the centralized installation manager repository, the repository tool cannot be used to add an integrated installation package (IIP) to the repository.

Results

The centralized installation manager repository now contains one or more WebSphere Extended Deployment products. For information on how to install the repository from the command line, see [Installing the product silently](#).

What to do next

Use the centralized installation manager to install the products to the nodes and begin managing your environments. In the administrative console, click **System administration > Centralized Installation Manager**.

Parent topic: [Using ISMP programs to install the product](#)

Parent topic: [Using ISMP programs to install the product](#)

Related tasks:

[Using the centralized installation manager to manage your environment](#)
[Configuring a high availability deployment manager environment](#)

Using the Installation Factory plug-in to create and install customized packages

Use the Installation Factory plug-in to create a customized installation package (CIP) or an integrated installation package (IIP). A CIP contains a single product installation package and various optional assets. An IIP combines one or more installation packages into a single installation workflow that you design.

Before you begin

Before you can create and install customized packages, you must download the following products:

- [IBM® Installation Factory for WebSphere® Application Server](#)
- [IBM Installation Factory plug-in for WebSphere Virtual Enterprise](#)

About this task

Using the Installation Factory, you create a CIP by combining a single product component with maintenance packages, customization scripts, and other files. When you create an IIP, you aggregate individual components, or installation packages, into a single installation package.

Build definition file

A build definition file is an XML document that specifies how to build and install a customized installation package (CIP) or an integrated installation package (IIP). The Installation Factory plug-in reads the package details of the build definition file to generate a CIP or an IIP.

Creating a build definition file and generating a CIP

The IBM Installation Factory plug-in for WebSphere Extended Deployment generates a CIP according to the details that the build definition file provides. The build definition specifies the product package to install, the location of the CIP, the maintenance packages to include in the installation, the install scripts, and any additional files to include in the CIP.

Creating a build definition file and generating an IIP

The IBM Installation Factory plug-in for WebSphere Extended Deployment generates an IIP based on the properties that the build definition file provides, such as which installation packages to include in the IIP, the order in which the Installation Factory installs each package, and the location of the IIP.

Silently installing a CIP or an IIP

You can silently install a CIP or an IIP for the product by using either a fully-qualified response file, which you configure specifically to your needs, or parameters that you pass to the command line.

Parent topic: [Using ISMP programs to install the product](#)

Build definition file

A build definition file is an XML document that specifies how to build and install a customized installation package (CIP) or an integrated installation package (IIP). The Installation Factory plug-in reads the package details of the build definition file to generate a CIP or an IIP.

Before you can create a CIP or an IIP, you must create a build definition file for each customized package. The build definition file describes which product components, or installation packages, to install, the location of the CIP or IIP, the maintenance packages to include, the installation scripts, and other files that you choose. You can also specify in the build definition file for the IIP the order in which the Installation Factory plug-in installs each installation package.

The Build definition wizard steps you through the process of creating a build definition file. You can also use the wizard to modify an existing build definition file. Each panel in the Build definition wizard prompts you for information about a customized package, such as the package identification, the installation location for the build definition, and the installation location for the customized package. All of this information is saved in the new build definition file, or modified and saved in an existing build definition file. For more information, see the [CIP Build definition wizard panels](#) and the [IIP Build definition wizard panels](#).

To create only the build definition file, you can use the command-line interface tool to generate the customized package outside of the GUI. See [Silently installing a CIP or an IIP](#) for more information.

Parent topic: [Using the Installation Factory plug-in to create and install customized packages](#)

Related tasks:

[Creating a build definition file and generating a CIP](#)

[Creating a build definition file and generating an IIP](#)

Creating a build definition file and generating a CIP

The IBM® Installation Factory plug-in for WebSphere® Extended Deployment generates a CIP according to the details that the build definition file provides. The build definition specifies the product package to install, the location of the CIP, the maintenance packages to include in the installation, the install scripts, and any additional files to include in the CIP.



Before you begin

Gather the components to include in the CIP. Optional assets can include maintenance packages and additional files to install on the system. Ensure that all of the assets are available on the workstation on which you intend to create your CIP.

About this task

You can use the Build definition wizard to create a build definition file and generate a CIP.

Procedure

1. Run the following script from the *IF_HOME/bin* directory to start the Installation Factory:
 - o  ifgui.sh
 - o  ifgui.batClick the **New Build Definition** icon.
2. Select the product to include in the build definition file, and click **Finish** to start the Build definition wizard.
3. Follow the prompts in the wizard.

On the Install and Uninstall Scripts panel, click **Add Scripts...** to populate the table with any customized installation scripts. Type the location of the script files, and clear the check box to continue if an error message is displayed. The operation is stopped by default. Click **OK** to return to the panel.

Results

You created and customized the build definition file, and you generated the CIP if you chose to work in the connected mode.

If the Build definition wizard does not provide you with the option to generate the CIP from the build definition file, you can still generate it by running the *ifcli.sh|bat* script from the *IF_HOME/bin* directory.

What to do next

Install the CIP.

[Installing a CIP](#)

Simplify the product installation process by installing a customized installation package (CIP), which is a single product installation image that can include one or more maintenance packages, configuration scripts, and other files.

[Installing a CIP to apply maintenance to an existing product installation](#)

You can apply maintenance packages to an existing product installation by installing a customized installation package (CIP).

[Uninstalling CIP updates from an existing product installation](#)

You can remove CIP updates from an existing product installation without removing the entire product. Use the IBM Update Installer for WebSphere Software to uninstall any CIP updates.

Parent topic: [Using the Installation Factory plug-in to create and install customized packages](#)

Related concepts:

[Build definition file](#)

Related tasks:

[Installing a CIP](#)

[Silently installing a CIP or an IIP](#)

Installing a CIP

Simplify the product installation process by installing a customized installation package (CIP), which is a single product installation image that can include one or more maintenance packages, configuration scripts, and other files.

Before you begin

Create a build definition file to specify what options to include in the CIP.

About this task

A CIP combines and installs a single product component with maintenance packages, customization scripts, and other files.

Procedure

1. Stop all the processes that are running on the workstation that you are preparing for installation.

Run the following command to stop the deployment manager:

-     `profile_root/bin/stopManager.sh`
-  `profile_root\bin\stopManager.bat`

Run the following command to stop the nodes:

-     `profile_root/bin/stopNode.sh`
-  `profile_root\bin\stopNode.bat`

2. Run the following command to start the wizard:

-     `CIP_home/bin/install`
-  `CIP_home\bin\install.bat`

3. Follow the prompts to complete the installation.

Results

You successfully installed the CIP.

What to do next

You can use the Profile Management Tool plug-in or the **manageprofiles** command to create and augment profiles for the product.

Parent topic: [Creating a build definition file and generating a CIP](#)

Related tasks:

[Using the graphical user interface to create profiles](#)

[Using the centralized installation manager to manage your environment](#)

Installing a CIP to apply maintenance to an existing product installation

You can apply maintenance packages to an existing product installation by installing a customized installation package (CIP).

Before you begin

Create a build definition file to specify what options to include in the CIP.

About this task

When applying maintenance with a CIP that contains a refresh pack, a fix pack, or both, all previously installed authorized program analysis reports (APAR) are uninstalled by the wizard. If the CIP is at the same level as the product, previously installed APARs remain only if they are packaged in the CIP. To successfully apply maintenance or updates to an installation, the CIP must include the installed features. To successfully apply maintenance to an existing installation, which is referred to as a *slip installation*, you must include the installed features in the CIP.

Procedure

1. Stop all the processes that are running on the workstation that you are preparing for installation.

Run the following command to stop the deployment manager:

- `Linux` `AIX` `HP-UX` `Solaris` `profile_root/bin/stopManager.sh`
- `Windows` `profile_root\bin\stopManager.bat`

Run the following command to stop the nodes:

- `Linux` `AIX` `HP-UX` `Solaris` `profile_root/bin/stopNode.sh`
- `Windows` `profile_root\bin\stopNode.bat`

2. Run the following command to start the wizard:

- `Linux` `AIX` `HP-UX` `Solaris` `CIP_home/bin/install`
- `Windows` `CIP_home\bin\install.bat`

3. Follow the prompts to complete the installation.

Results

The installation preview summary lists the resulting product version and any applicable features and interim fixes. Next, the wizard successfully applies the maintenance and updates the features of the product.

The product binary files are copied to the `was_home/properties/version/nif/backup` directory. You can use the Update Installer to uninstall any updates and restore your workstation.

Parent topic: [Creating a build definition file and generating a CIP](#)

Related concepts:

[Build definition file](#)

Related tasks:

[Uninstalling CIP updates from an existing product installation](#)

[Creating a build definition file and generating a CIP](#)

Uninstalling CIP updates from an existing product installation

You can remove CIP updates from an existing product installation without removing the entire product. Use the IBM® Update Installer for WebSphere® Software to uninstall any CIP updates.

Before you begin

You must have at least one existing copy of the product installed on the system.

About this task

Complete the following procedure to remove CIP updates from an existing product installation.

Procedure

1. Download the latest version of the Update Installer. See [Recommended fixes](#) for more information.
2. Install the Update Installer. See [Installing the Update Installer for WebSphere Software](#) in the WebSphere Application Server Network Deployment Information Center for more information.
3. Uninstall any fix pack, refresh pack, or interim fix that you added to your environment after you installed the CIP.
4. Uninstall any interim fixes that you included in the slip installation. This process is the same as uninstalling a single fix pack or refresh pack. However, the maintenance that was included in the CIP is now included in a single operation.
5. Uninstall the CIP by using the Update Installer. The maintenance levels return to the pre-update state, and the CIP is denoted by the CIP identifier prepended to its filename.

Results

You successfully removed the CIP updates from an existing product installation.

Parent topic: [Creating a build definition file and generating a CIP](#)

Related tasks:

[Installing a CIP to apply maintenance to an existing product installation](#)

Related information:

[IBM Support and downloads](#)


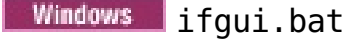
Creating a build definition file and generating an IIP

The IBM® Installation Factory plug-in for WebSphere® Extended Deployment generates an IIP based on the properties that the build definition file provides, such as which installation packages to include in the IIP, the order in which the Installation Factory installs each package, and the location of the IIP.

About this task

Using the Build definition wizard, specify the details of an IIP in the build definition file, which is then used to generate the customized package.

Procedure

1. Run the following script from the *IF_home/bin* directory to start the Installation Factory:
 - o  ifgui.sh
 - o  ifgui.bat
2. Click the **Create New Integrated Installation Package** icon to start the Build definition wizard.
3. Follow the prompts to complete the process.
 - a. On the Construct the IIP panel, select a supported installation package from the list, and click **Add Installer** to add the installation package to the IIP. A panel that displays the package name, the package identifier, and the package properties is displayed. To view specific information about the selected package, click **View Installation Package Information**. Click **Modify** to enter the directory path to the installation package for each operating system. If you are currently adding an installation package for WebSphere Extended Deployment, select the checkbox, which provides you with the option to use the same package for all supported operating systems. Click **OK** and return to the Construct the IIP panel. An invocation is created by default.
 - To modify the directory path to an installation package, select the package from the Installation packages used in the IIP list, and click **Modify**.
 - To modify an invocation, select the invocation, and click **Modify**. Specify the default installation location for the invocation on each operating system. Specify the location to the response file if you select a silent installation as the default installation mode.
 - Click **Add Invocation** to add an invocation contribution to the installation package. A panel from which you can specify properties for the invocation is displayed.
 - Click **Remove** to remove installation packages or invocations.
4. Review the summary of your selections, select the **Save build definition file and generate integrated installation package** option, and click **Finish**.

Alternatively, you can choose to save the build definition file without generating the IIP. With this option, you actually generate the IIP outside of the wizard by running the `ifcli.sh|bat` script from the *IF_home/bin* directory.

Results

You created and customized the build definition file for an IIP.

What to do next

Install the IIP.

[Installing an IIP](#)

Use the IBM Installation Factory plug-in for WebSphere Extended Deployment to install an IIP. An IIP combines one or more installation packages into a single workflow that you design.

[Modifying an existing build definition file for an IIP](#)

You can edit or add to the properties of an IIP to further customize the installation.

Parent topic: [Using the Installation Factory plug-in to create and install customized packages](#)

Related concepts:

[Build definition file](#)

Related tasks:

[Silently installing a CIP or an IIP](#)

Installing an IIP

Use the IBM® Installation Factory plug-in for WebSphere® Extended Deployment to install an IIP. An IIP combines one or more installation packages into a single workflow that you design.






Before you begin

Create a build definition file to specify what options to include in the IIP.

About this task

An IIP can include one or more generally available installation packages, one or more customized installation packages (CIPs), and other optional files and directories. Create an IIP to aggregate multiple installation packages, or *contributions*, into a single package, and install the contributions one after the other in a specific order to complete an end-to-end installation.

Procedure

1. Run the following script to start the wizard:
 - o     `IIP_home/bin/install`
 - o  `IIP_home\bin\install.bat`
2. Click **About** on the Welcome panel to view the details of the IIP, such as the package identifier, the supported operating systems, and the included installation packages.

Optional: To modify the installation options for each package, click **Modify**.

Optional: Two **View Log** buttons are displayed on the wizard panel. To view the log of each package, click the **View Log** button that is displayed next to the table that lists the installation packages. To view the overall log details of the IIP, click the **View Log** button that is displayed next to the status information.
3. Select the installation packages to run, and click **Install**. A list of all the contributions in the order of invocation that the IIP contains is displayed. To designate which contribution invocations should not be run during the installation, clear the checkbox located next to the **Installation name** field.

Results

You successfully installed an IIP.

Parent topic: [Creating a build definition file and generating an IIP](#)

Related tasks:

[Creating a build definition file and generating an IIP](#)


Modifying an existing build definition file for an IIP

You can edit or add to the properties of an IIP to further customize the installation.

About this task

To change the properties of an IIP, modify the existing build definition file.

Procedure

1. Run the following script from the *IF_HOME/bin* directory to start the Installation Factory:
 -  ifgui.sh
 - ifgui.bat
2. Click the **Open Build Definition** icon, and select the build definition file that you want to modify.
3. Select the specific properties of the IIP that you want to modify. The following list contains the possible modifications that you can make:
 - Change your current mode selection. In connected mode, you create the build definition for use, and optionally generate the IIP, from your current workstation. In disconnected mode, you create the build definition file for use on another workstation.
 - Add or remove the existing operating systems that the IIP supports.
 - Edit the existing identifier and version for the IIP.
 - Edit the target location for the build definition file.
 - Edit the target location for the IIP.
 - Change whether to display an installation wizard for the IIP. The wizard provides information about the IIP and the installation options when the IIP runs.
 - Add, remove, and edit the installation packages that are contained in the IIP.

Important: If you added a supported operating system and you have not updated the properties of the installation package in the IIP, you receive a warning message stating that the selected contributions do not contain installation packages that are identified for all of the operating systems that the IIP supports. Click **Yes** to continue, or click **No** to edit the installation package.
4. Review the summary of your selections, select **Save build definition file and generate integrated installation package**, and click **Finish**.

Parent topic: [Creating a build definition file and generating an IIP](#)

Related concepts:

[Build definition file](#)

Related tasks:

[Creating a build definition file and generating an IIP](#)

Silently installing a CIP or an IIP

You can silently install a CIP or an IIP for the product by using either a fully-qualified response file, which you configure specifically to your needs, or parameters that you pass to the command line.

Before you begin

Create the build definition file for the CIP or IIP.

About this task

A silent installation uses the same installation program that the graphical user interface (GUI) version uses. However, instead of displaying a wizard interface, the silent installation reads all of your responses from a file that you customize, or from parameters that you pass to the command line. If you are silently installing an IIP, you can invoke a contribution with a combination of options that you specify directly on the command line, as well as options that you specify in a response file. However, any contribution options that you pass to the command line causes the IIP installer to ignore all of the options that are specified in a specific contribution's response file. See the detailed [IIP installation options](#) for more information.

Procedure

1. Optional: If you choose to install the CIP or IIP using a response file, first customize the file.
 - a. Copy the response file from the product DVD to your disk drive.
 - b. Open and edit the response file in the text editor of your choice.

Tip: The installer uses the location that you select for your installation to determine where your WebSphere® Application Server Network Deployment instance is installed. If you install on a node with multiple instances, clearly define your location.

- c. Run the following script to start your customized response file.

- `Linux` ▶ `AIX` ▶ `HP-UX` ▶ `Solaris` `install -options /absolute_path/response_file.txt -silent`
- `Windows` `install.bat -options C:\drive_path\response_file.txt -silent`

2. Optional: If you choose to install the CIP or IIP by passing certain parameters to the command line, run the following script to start the installation:
 - o `Linux` ▶ `AIX` ▶ `HP-UX` ▶ `Solaris` `install -silent -OPT silentInstallLicenseAcceptance=true -OPT installLocation=install_location`
 - o `Windows` `install.bat -silent -OPT silentInstallLicenseAcceptance=true -OPT installLocation=install_location`where *install_location* is the location of your existing WebSphere Application Server Network Deployment installation.
3. Review the resulting logs for errors or an installation failure.

Results

You silently installed the CIP or IIP.

What to do next

Use the Profile Management Tool plug-in or the **manageprofiles** command to create and augment profiles if you chose not to augment profiles for the product during the installation process.

If you already augmented profiles for the product, you can begin creating and configuring servers and deploying applications in your environment.

Parent topic: [Using the Installation Factory plug-in to create and install customized packages](#)

Related concepts:

[Build definition file](#)

Related tasks:

[Creating a build definition file and generating a CIP](#)

[Creating a build definition file and generating an IIP](#)

[Creating and augmenting profiles](#)

[Preparing the hosting environment for dynamic operations](#)

Creating and augmenting profiles

After you install the product, create unique profile types and augment existing profiles for WebSphere® Virtual Enterprise.

Before you begin

Ensure that your environment meets all of the hardware and software requirements.

About this task

To utilize the product features, use Profile Management Tool plug-in or the **manageprofiles** command to create and augment profiles for the product.

[Using the graphical user interface to create profiles](#)

Use the graphical user interface (GUI), which is provided by the Profile Management Tool plug-in, to create profiles for the product. A profile is a set of files that define the runtime environment.

[Using the graphical user interface to augment profiles](#)

After you install the product, you can augment an existing profile to make it compatible with WebSphere Virtual Enterprise.

[manageprofiles command](#)

Use the **manageprofiles** command utility to create and augment profiles with the WebSphere Virtual Enterprise template. To use the features of the product, your environment must contain at least one profile augmented for the product.

[Non-root profile creation and augmentation](#)

Provide a non-root user permissions for files and directories so that the non-root user can create a profile for the product, and augment a profile that was created by a root user, a different non-root user, or the same non-root user.

What to do next

Depending on which task you choose to complete, start the First steps console for assistance with configuring and testing your product environment. Alternatively, repeat any of the preceding tasks to create or augment additional profiles.

Parent topic: [Using ISMP programs to install the product](#)

Using the graphical user interface to create profiles

Use the graphical user interface (GUI), which is provided by the Profile Management Tool plug-in, to create profiles for the product. A profile is a set of files that define the runtime environment.

About this task

The Profile Management Tool plug-in enables the graphical user interface (GUI) to assist you in creating profiles, such as a stand-alone application server profile, a deployment manager profile, a cell profile, and a custom profile.

Procedure

1. Choose one of the following options to start the profile creation wizard:
 - Select **Profile Management Tool** from the First steps console.
 - Access the Profile Management Tool from the **Start** menu.
 - Run the `./pmt.sh|bat` script from the `install_root/bin/ProfileManagement` directory.The Action Selection page is displayed only if at least one profile and the augment templates exist.
2. Follow the prompts to complete the process.

What to do next

You can create additional profiles or augment existing profiles. To restart the Profile Management tool, run the `./pmt.sh|bat` command from the `install_root/bin/ProfileManagement` directory, or select **Profile Management Tool** in the First steps console.

[Creating profiles for multiple products](#)

You can create profiles for multiple products to further enhance your environment and utilize all of the product features that are available.

[Creating profiles for use with other IBM products](#)

You can create profiles that combine WebSphere Extended Deployment features with the features of other IBM® products, such as WebSphere Commerce Server and WebSphere Portal for Multiplatforms.

Parent topic: [Creating and augmenting profiles](#)

Related concepts:

[Non-root profile creation and augmentation](#)

Related tasks:

[Using the graphical user interface to augment profiles](#)

Creating profiles for multiple products

You can create profiles for multiple products to further enhance your environment and utilize all of the product features that are available.

Procedure

1. Create a new profile for WebSphere® Virtual Enterprise.
 - a. Choose one of the following options to start the profile creation wizard:
 - Select **Profile Management Tool** from the First steps console.
 - Access the Profile Management Tool from the **Start** menu.
 - Run the `./pmt.sh|bat` script from the `install_root/bin/ProfileManagement` directory.The Action Selection page is displayed only if at least one profile and the augment templates exist.
 - b. Follow the prompts on the panels.
 - c. Review the summary of your selections, and click **Create**.
 - d. Review the completion panel, and click **Finish**.
2. Augment the profile with the WebSphere Extended Deployment Compute Grid augment template.
 - a. Start the Profile Management Tool plug-in. Run the `./pmt.sh|bat` command from the `install_root/bin/ProfileManagement` directory, or select **Profile Management Tool** from the First steps console. Click **Augment**.
 - b. Select the existing profile that you want to augment, and click **Next**.
 - c. Select the **WebSphere Extended Deployment Compute Grid** augment, and click **Next**. The Profile Verification panel is displayed, which verifies if the profile can be augmented by the selected augment.
 - d. Review the Profile Augmentation Summary panel, and click **Augment**. The Profile Augmentation Complete panel is displayed. Because you are creating the profile specifically for multiple products, click **Augment another profile** to return to the Profile Selection panel.
3. Augment the profile with the WebSphere eXtreme Scale augment template.
 - a. Select the existing profile that you want to augment, and click **Next**.
 - b. Select the **WebSphere eXtreme Scale** augment, and click **Next**. The Profile Verification panel is displayed.
 - c. Review the Profile Augmentation Summary panel, and click **Augment**. The Profile Augmentation Complete panel is displayed. Click **Finish** to exit the wizard.

Parent topic: [Using the graphical user interface to create profiles](#)

Related concepts:

[Non-root profile creation and augmentation](#)

Related tasks:

[Augmenting profiles with multiple product augment templates](#)

Related reference:

[manageprofiles command](#)

Creating profiles for use with other IBM products

You can create profiles that combine WebSphere® Extended Deployment features with the features of other IBM® products, such as WebSphere Commerce Server and WebSphere Portal for Multiplatforms.

Procedure

1. Create a new profile for WebSphere Virtual Enterprise. If you installed multiple WebSphere Extended Deployment products, you are not required to create multiple profiles. Instead, create a profile for only one of the products, depending on your topology and any support requirements.
 - a. Choose one of the following options to start the Profile Management Tool plug-in:
 - Select **Profile Management Tool** from the First steps console.
 - Access the Profile Management Tool from the **Start** menu.
 - Run the `./pmt.sh|bat` script from the `install_root/bin/ProfileManagement` directory.
The Action Selection page is displayed only if at least one profile and the augment templates exist.
 - b. Follow the prompts on the panels.
 - c. Review the summary of your selections, and click **Create**.
 - d. Review the completion panel, and click **Finish**.
2. Augment the profile with the augment templates for the other products that exist in your environment.

Restriction: Some augment templates cannot be used more than once with each profile. If unsuccessfully attempted, a message appears that states the selected template does not support multiple augmentations. Click **OK**, and select a different template.

Results

The profile is augmented to use the features of WebSphere Virtual Enterprise and other IBM products.

Parent topic: [Using the graphical user interface to create profiles](#)

Related concepts:

[Non-root profile creation and augmentation](#)

Related reference:

[manageprofiles command](#)

Using the graphical user interface to augment profiles

After you install the product, you can augment an existing profile to make it compatible with WebSphere® Virtual Enterprise.

About this task

When you augment an existing profile, you change the profile by applying an augmentation template.

Procedure

1. Stop all the processes in the cell.

Run the following command to stop the deployment manager:

- **Windows** `profile_root\bin\stopManager.bat`
- **Linux** **AIX** **HP-UX** **Solaris** `profile_root/bin/stopManager.sh`

Run the following command to stop the nodes:

- **Windows** `profile_root\bin\stopNode.bat`
- **Linux** **AIX** **HP-UX** **Solaris** `profile_root/bin/stopNode.sh`

2. Augment the deployment manager profile using one of the following options. Complete the steps in the wizard.
 - Select **Profile Management Tool** from the First steps console.
 - Access the **Profile Management Tool** from the **Start** menu.
 - Run the `./pmt.sh|bat` script from the `install_root/bin/ProfileManagement` directory.
3. Start the deployment manager. To start the deployment manager, run the following command from the `profile_root/bin` directory:
 - **Windows** `startManager.bat`
 - **Linux** **AIX** **HP-UX** **Solaris** `./startManager.sh`
4. Augment your existing profiles on each node using one of the following options. Complete the steps in the wizard.
 - Select **Profile Management Tool** from the First steps console.
 - Access the **Profile Management Tool** from the **Start** menu.
 - Run the `./pmt.sh|bat` script from the `install_root/bin/ProfileManagement` directory.
5. Run the `syncNode` command to synchronize the deployment manager with the nodes.

```
WAS_install_root/AppServer/bin ./syncNode.sh dmgrSystem dmgrPort -  
conntype soap -username user_name -password user_password
```

```
WAS_install_root/AppServer2/bin ./syncNode.sh dmgrSystem dmgrPort -  
conntype soap -username user_name -password user_password
```

6. Start the node agents. To start each of the nodes, run the following command from the `profile_root/bin` directory:
 - **Windows** `startNode.bat`
 - **Linux** **AIX** **HP-UX** **Solaris** `./startNode.sh`

Results

You successfully augmented your profiles.

What to do next

Set up your product environment. To federate one or more nodes into the cell, run the `addNode` command in the administrative console.

Parent topic: [Creating and augmenting profiles](#)

Related tasks:

[Using the graphical user interface to create profiles](#)

Related reference:

[manageprofiles command](#)

manageprofiles command

Use the **manageprofiles** command utility to create and augment profiles with the WebSphere® Virtual Enterprise template. To use the features of the product, your environment must contain at least one profile augmented for the product.

Location

The command file is in the *install_root/bin* directory.

Purpose

The **manageprofiles** command creates the runtime environment for a product process in a set of files called a profile. When you augment an existing profile, you change the profile by applying a product-specific augmentation template. You can use the **manageprofiles** command to complete the following tasks:

- [Create a deployment manager profile](#)
- [Create a custom profile](#)
- [Create a stand-alone application server profile](#)
- [Create a cell profile](#)
- [Augment a deployment manager profile](#)
- [Augment a custom profile](#)
- [Augment a stand-alone application server profile](#)
- [Augment a cell profile](#)
- [Unaugment any type of profile](#)

In the following sections, the tasks you can perform using the **manageprofiles** command, along with a list of required parameters, is described. For details on the optional parameters to specify for each task, see the [manageprofiles command](#) in the WebSphere Application Server Network Deployment Information Center.

Create a deployment manager profile

You can use the **manageprofiles** command to create a deployment manager profile. The deployment manager administers the application servers that are federated into the cell.

Required parameters

-create

Creates the profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/dmgr
```

Example

```
./manageprofiles.sh|bat -create -profileName profile01 -templatePath  
install_root/profileTemplates/xd_augment/dmgr
```

Create a custom profile

You can use the **manageprofiles** command to create a custom profile.

Required parameters

-create

Creates the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/managed
```

Example

```
./manageprofiles.sh|bat -create -profileName profile01 -templatePath
install_root/profileTemplates/xd_augment/managed
```

Create a stand-alone application server profile

You can use the **manageprofiles** command to create a stand-alone application server profile.

Required parameters

-create

Creates the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/default
```

Example

```
./manageprofiles.sh|bat -create -profileName profile01 -templatePath
install_root/profileTemplates/xd_augment/default
```

Create a cell profile

You can use the **manageprofiles** command to create a cell profile, which consists of a deployment manager and an application server.

Required parameters

Specify the following parameters for the deployment manager profile:

-create

Creates the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/cell/dmgr
```

Specify the following parameters for the application server profile:

-create

Creates the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/cell/default
```

Example

```
./manageprofiles.sh|bat -create -profileName profile01 -templatePath
install_root/profileTemplates/xd_augment/cell/dmgr
```

```
./manageprofiles.sh|bat -create -profileName profile01 -templatePath
install_root/profileTemplates/xd_augment/cell/default
```

Augment a deployment manager profile

You can use the **manageprofiles** command to augment a deployment manager profile.

Required parameters

-augment

Augments the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/dmgr
```

Example

```
./manageprofiles.sh|bat -augment -profileName profile01 -templatePath  
install_root/profileTemplates/xd_augment/dmgr
```

Augment a custom profile

You can use the **manageprofiles** command to augment a custom profile.

Required parameters

-augment

Augments the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/managed
```

Example

```
./manageprofiles.sh|bat -augment -profileName profile01 -templatePath  
install_root/profileTemplates/xd_augment/managed
```

Augment a stand-alone application server profile

You can use the **manageprofiles** command to augment a stand-alone application server profile.

Required parameters

-augment

Augments the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/default
```

Example

```
./manageprofiles.sh|bat -augment -profileName profile01 -templatePath  
install_root/profileTemplates/xd_augment/default
```

Augment a cell profile

You can use the **manageprofiles** command to augment a cell profile.

Required parameters

Specify the following parameters for the deployment manager profile:

-augment

Augments the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/cell/dmgr
```

Specify the following parameters for the application server profile:

-augment

Augments the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/cell/default
```

Example

```
./manageprofiles.sh|bat -augment -profileName profile01 -templatePath  
install_root/profileTemplates/xd_augment/cell/dmgr
```

```
./manageprofiles.sh|bat -augment -profileName profile01 -templatePath  
install_root/profileTemplates/xd_augment/cell/default
```

Unaugment a profile

To unaugment a profile, specify the **-ignoreStack** parameter with the **-templatePath** parameter in addition to specifying the required **-unaugment** and **-profileName** parameters.

Required parameters

-augment

Augments the existing profile. (Required)

-profileName

Specifies the name of the profile. (Required)

-ignoreStack

Used with the **-templatePath** parameter to unaugment a particular profile that has been augmented. (Required)

-templatePath *template_path*

Specifies the path to the template files in the installation root directory. (Required)
Use the following format:

```
-templatePath install_root/profileTemplates/xd_augment/profile_type
```

where *profile_type* is one of four profile types:

- dmgr: deployment manager profile
- managed: custom profile
- default: stand-alone application server profile
- cell: cell profile

Example

```
./manageprofiles.sh|bat -unaugment -profileName profile01 -ignoreStack -  
templatePath install_root/profileTemplates/xd_augment/profile_type
```


Parent topic: [Creating and augmenting profiles](#)

Related concepts:

[Non-root profile creation and augmentation](#)

Related reference:

 [manageprofiles command](#)

Non-root profile creation and augmentation

Provide a non-root user permissions for files and directories so that the non-root user can create a profile for the product, and augment a profile that was created by a root user, a different non-root user, or the same non-root user.

Overall, non-root users are limited in being able to create and use profiles in their environment. Within the Profile Management Tool plug-in, unique names and port values are disabled for non-root users. The non-root user must change the default field values in the Profile Management tool for the profile name, node name, cell name, and port assignments. Consider assigning non-root users a range of values for each of the fields. You can assign responsibility to the non-root users for adhering to their proper value ranges and for maintaining the integrity of their own definitions.

The term *installer* refers to either a root or non-root user. As an installer, you can grant non-root users permissions to create profiles and establish their own product environments. For example, a non-root user might create a product environment to test application deployment with a profile that he owns. Specific tasks that you can complete to allow non-root profile creation include the following items:

- Creating a profile and assigning ownership of the profile directory to a non-root user so that the non-root user can start WebSphere® Application Server for a specific profile.
- Granting write permission of the appropriate files and directories to a non-root user, which allows the non-root user to then create the profile. With this task, you can create a group for users who are authorized to create profiles, or give individual users the ability to create profiles.
- Installing maintenance packages for the product, which includes required services for existing profiles that are owned by a non- user. As the installer, you are the owner of any new files that the maintenance package creates.

For further details, read the detailed information on creating profiles for non-root users, which includes the steps to complete the preceding task examples, in the [WebSphere Application Server Network Deployment Information Center](#).

As an installer, you can also grant permissions for a non-root user to augment profiles. For example, a non-root user can augment a profile that is created by an installer, or augment a profile that he creates. Follow the WebSphere Application Server Network Deployment non-root user augmentation process to complete these tasks.

However, there are exceptions to note. When a non-root user augments a profile that is created by the installer, the following files do not need to be created by the non-root user before augmentation, because they were established during the profile creation process:

- `app_server_root/logs/manageprofiles.xml`
- `app_server_root/properties/fsdb.xml`
- `app_server_root/properties/profileRegistry.xml`

Also, do not modify the Linux or UNIX operating systems **Start menu** permissions.

When a non-root user augments a profile that he creates, the non-root user must modify the permissions for the documents that are located within the WebSphere Virtual Enterprise profile templates.

Parent topic: [Creating and augmenting profiles](#)

Parent topic: [Creating and augmenting profiles](#)

Uninstalling the product

You can uninstall the product and any maintenance files for the product that are installed in your environment. Complete the uninstallation process either silently or by following the graphical user interface (GUI).

Before you begin

The `install_root/uninstall_xd_operations` subdirectory must exist. If the process fails before the uninstallation files are copied to the workstation, you cannot complete this task. Additionally, you must uninstall any existing maintenance files before attempting to uninstall the product. The wizard will not proceed with the uninstallation if existing maintenance files are detected.

Procedure

1. Stop all processes before starting the uninstallation process.

Run the following command to stop the deployment manager:

- o `Linux` `AIX` `HP-UX` `Solaris` `./stopManager.sh`
- o `Windows` `stopManager.bat`

Run the following command to stop the nodes:

- o `Linux` `AIX` `HP-UX` `Solaris` `./stopNode.sh`
- o `Windows` `stopNode.bat`

`Linux` `AIX` `HP-UX` `Solaris` Complete a search to validate that the Java™ processes are stopped.

`Windows` Use the task manager to stop the Java processes.

2. Uninstall the product using the wizard or silently.

Run the following script from the `install_root/uninstall_xd_operations` directory to start the wizard:

- o `Linux` `AIX` `HP-UX` `Solaris` `uninstall`
- o `Windows` `uninstall.exe`

Run the following script from the `install_root/uninstall_xd_operations` directory to silently uninstall the product:

- o `Linux` `AIX` `HP-UX` `Solaris` `./uninstall.sh -silent`
- o `Windows` `uninstall.bat -silent`

3. Uninstall the deployment manager. Run the following script from the `install_root/uninstall_xd_operations` directory:

- o `Linux` `AIX` `HP-UX` `Solaris` `uninstall`
- o `Windows` `uninstall.exe`

4. Uninstall the centralized installation manager repository. Run the uninstaller in silent mode, and specify the following option:

`Windows`

```
uninstall.bat -OPT uninstallCimgr=true -silent
```

`Linux` `AIX` `HP-UX` `Solaris`

```
./uninstall -OPT uninstallCimgr=true -silent
```

Restriction: This option is only supported with the uninstallation of WebSphere® Virtual Enterprise.

Results

The product is uninstalled. Review the log files in the `install_root/logs/package_name/uninstall/log` directory for any errors.

Parent topic: [Planning the product installation](#)

Parent topic: [Using ISMP programs to install the product](#)

Related tasks:

[Using ISMP programs to install the product](#)

Customizing WebSphere Virtual Enterprise for z/OS

Using the WebSphere® Customization Toolbox, you can generate and run customized jobs to customize WebSphere Virtual Enterprise for z/OS®.

Before you begin

- Install WebSphere Application Server Network Deployment for z/OS. Depending on which version of the product you are running, your system must meet one of the following minimum requirements:
 - WebSphere Application Server Network Deployment Version 7.0.0.17 or later
 - WebSphere Application Server Network Deployment Version 8.0.0.1 or laterSee [Latest fix packs for WebSphere Application Server](#) for more information.
- Install WebSphere Virtual Enterprise Version 7.0.0.1 or later for z/OS. See the Program Directory on the [Library page](#) for more information.

z/OS

[Installing the WebSphere Customization Toolbox](#)

Install the WebSphere Customization Toolbox to customize WebSphere Virtual Enterprise .

z/OS

[Generating customization definitions](#)

Use the Profile Management Tool provided in the WebSphere Customization Toolbox to generate customization definitions and create customized jobs for WebSphere Virtual Enterprise .

z/OS

[Uploading and running customized jobs](#)

After you generate the appropriate customization definitions, you can upload and run customized jobs associated with the definitions to your target z/OS systems.

Parent topic: [Installing and customizing the product](#)

Related concepts:

[Hardware and software requirements](#)

Related tasks:

[Migrating WebSphere Virtual Enterprise for z/OS](#)

Installing the WebSphere Customization Toolbox

Install the WebSphere® Customization Toolbox to customize WebSphere Virtual Enterprise .

Before you begin

- You must use the latest version of the WebSphere Customization Toolbox to successfully install the product extension files.

Note: You can use WebSphere Customization Tools Version 7.0.0.x to customize WebSphere Virtual Enterprise Version 7.0.0 for z/OS®, but any existing Version 6.1.1 profiles are rendered unusable and must be deleted and reaugmented. Therefore, it is highly recommended that you install the we.wct extension file in Version 8.0.0.x of the WebSphere Customization Toolbox, and augment WebSphere Virtual Enterprise Version 7.0.0 for z/OS profiles in this environment.

- When you augment WebSphere Virtual Enterprise Version 7.0.0 for use with WebSphere Application Server Network Deployment Version 8.0.0, avoid trouble by ensuring that the product levels are at WebSphere Virtual Enterprise Version 7.0.0.1 or later and WebSphere Application Server Network Deployment Version 8.0.0.1 or later.
- Verify that the WebSphere Customization Toolbox build ID string, which is displayed on the About the WebSphere Customization Toolbox panel, is at a level of cf01xxxx or higher.

About this task

The WebSphere Customization Toolbox is a workstation based graphical tool you use to create customized jobs that build the Virtual Enterprise runtime environment for z/OS.

Procedure

1. Use FTP to copy the Virtual Enterprise we.wct extension file from your z/OS system to the workstation on which you are installing the WebSphere Customization Toolbox. The extension file is in the /usr/lpp/zWebSphereVE/V7R0/util/WCT directory on your z/OS system.
2. Download and install the latest version of the WebSphere Customization Toolbox.
3. Install the we.wct extension file into the WebSphere Customization Toolbox application.
 - a. Start the WebSphere Customization Toolbox on your workstation.
 - b. Click **Help > Software Updates > Manage Extension**.
 - c. From the WebSphere Customization Toolbox Extension panel, click **Install**.
 - d. From the Source Archive File panel, click **Browse**, navigate to the directory in which you copied the extension file in Step 1, and click **Open**.
 - e. Click **Next** on the Source Archive File panel.
 - f. Click **Next** on the Extension Summary panel, and click **Finish**.
 - g. Click **Finish** on the WebSphere Customization Toolbox Extension panel.

What to do next

You can use the Profile Management Tool to generate customization definitions for Virtual Enterprise.

Parent topic: [z/OS Customizing WebSphere Virtual Enterprise for z/OS](#)

Related tasks:

[Generating customization definitions](#)

[Uploading and running customized jobs](#)

Generating customization definitions

Use the Profile Management Tool provided in the WebSphere® Customization Toolbox to generate customization definitions and create customized jobs for WebSphere Virtual Enterprise .

Before you begin

- Download and install the latest version of the WebSphere Customization Toolbox, and install the we.wct extension file.
- Install and configure WebSphere Application Server Network Deployment for z/OS®.
- Install WebSphere Virtual Enterprise for z/OS.

About this task

You can generate customization definitions using the Profile Management Tool. A *customization definition* is a set of files that is used to create customized jobs for the purpose of configuring WebSphere Virtual Enterprise for z/OS .

Procedure

1. Start the Profile Management Tool. Click **Start > Programs > IBM WebSphere > WebSphere Customization Toolbox**.
2. Add an existing location or create a new location of the customization definition that you want to create. On the **Customization Locations** tab, click **Add**. If you create a new location, the Version box refers to the existing version of WebSphere Application Server Network Deployment installed on your z/OS system.

Note: Do not use the same location you are using for other Virtual Enterprise customization definitions.

3. Generate the customization definition. On the **Customization Definitions** tab, click **Augment**.
4. Select the type of definition environment to create:
 - **Deployment manager**
 - **Application server**
 - **Managed (custom) node**

Note: When you customize the product for a federated application server node that is not migrated, select **Application server** as the type of definition environment to create. When you customize the product for a federated application server node that has been migrated, select **Managed (custom) node** as the type of definition environment to create. When a federated application server node itself is migrated, the migration type is changed to a managed (custom) node.

5. Complete the fields on the panels. Specify the values for the parameters that are used to create your z/OS system.
6. Click **Augment** to generate the customization definition.

What to do next

Upload the customized jobs to your target z/OS system.

Parent topic: [z/OS Customizing WebSphere Virtual Enterprise for z/OS](#)

Related tasks:

[Uploading and running customized jobs](#)

[Installing the WebSphere Customization Toolbox](#)

Uploading and running customized jobs

After you generate the appropriate customization definitions, you can upload and run customized jobs associated with the definitions to your target z/OS® systems.

Before you begin

Generate the customization definitions for the jobs that you want to upload to your z/OS systems.

About this task

Upload and run the customized jobs that you created using the WebSphere® Customization Toolbox to administer and monitor your WebSphere Virtual Enterprise for z/OS environments.

Procedure

1. Upload the customized jobs. On the **Customization Definitions** tab, select the jobs that you want to upload and click **Process**.
2. Upload the jobs to the FTP server on your z/OS system. Specify the required information on the **Upload Customization Definition** panel.
3. Click **Finish**.
4. Run the customized jobs. Click the **Customization Instructions** tab, and follow the customization instructions for each job.

Parent topic: [z/OS Customizing WebSphere Virtual Enterprise for z/OS](#)

Related tasks:

[Generating customization definitions](#)

Uninstalling the product on z/OS

You can uninstall WebSphere® Virtual Enterprise Version 7.0.0.4 that is installed in your z/OS® environment.

Before you begin

Ensure that WebSphere Virtual Enterprise Version 7.0.0.4 for z/OS is running in your current cell. For more information about the program directory see the Library page.

To uninstall the product, stop all the servers on all nodes.

Some of the commands are specific to the application serving environment configured with the **zpmt** command. To determine the environment of a node, examine the `config_root/properties/profileRegistry.xml` file, or examine your **zpmt** profile. Some of the paths that are used in the commands are intermediate links that were configured when the environment was created.

About this task

To uninstall WebSphere Virtual Enterprise Version 7.0.0.4, follow this procedure:

1. Uninstall the deployment manager node.
2. Uninstall the nodes that contain an on demand router (ODR).
3. Uninstall the nodes that do not contain an ODR.

Run the commands for the deployment manager from the Virtual Enterprise Version 7.0.0 `install_root/bin` directory, as shown in the following examples. If the current cell is customized with a WebSphere Virtual Enterprise Version 7.0 intermediate link, the value of the `install_root` variable is the intermediate link directory, and not the actual installation root directory string.

Procedure

1. Shut down the cell.
2. Unaugment, and unlink WebSphere Virtual Enterprise Version 7.0.0 from the deployment manager node:
 - a. Run the **manageprofiles** command for the deployment manager.

If you created the deployment manager using **zpmt** and the management environment, run this command:

```
cd config_root/DeploymentManger/bin
./manageprofiles.sh -unaugment -profileName default -
templatePath
config_root/DeploymentManager/profileTemplates/xd_augment/dmgr -
ignoreStack
```

If you created the deployment manager and the application server together using **zpmt** with the cell (deployment manager and application server) environment, run this command:

```
cd config_root/DeploymentManger/bin
./manageprofiles.sh -unaugment -profileName default -
templatePath
config_root/DeploymentManager/profileTemplates/xd_augment/cell/dmgr -
ignoreStack
```

- b. Run the **remove** command to unlink the WebSphere Virtual Enterprise files. If the cell is customized with a WebSphere Virtual Enterprise Version 7.0.0 intermediate link, the value of the `install_root` variable is the intermediate link directory string, and not the actual installation root directory string.

```
cd install_root/bin
./removeVE700.sh install_root config_root/DeploymentManager
userID userPW > /tmp/rmVE700Dmgr.output
```

3. Unaugment, and unlink WebSphere Virtual Enterprise Version 7.0.0 from the application node that contains an ODR. In the following example, `AppServer3` is an empty managed node.
 - a. Run the **manageprofiles** command for the empty managed node.

```
cd config_root/AppServer3/bin
./manageprofiles.sh -unaugment -profileName default -
templatePath
config_root/AppServer3/profileTemplates/xd_augment/managed -
ignoreStack
```

- b. Run the **remove** command to unlink the WebSphere Virtual Enterprise files. If the cell is customized with a WebSphere Virtual Enterprise Version 7.0.0 intermediate link, the value of the *install_root* variable is the intermediate link directory string, and not the actual installation root directory string.

```
cd VE700_install_root/bin
./removeVE700.sh install_root config_root/AppServer3 userID
userPW > /tmp/rmVE700APP3.output
```

4. Unaugment, and unlink WebSphere Virtual Enterprise Version 7.0.0 from the application nodes that do not contain an ODR. In the following example, *AppServer2* is an empty managed node.
 - a. Run the **mangeprofiles** command for the empty managed node.

```
cd config_root/AppServer2/bin
./manageprofiles.sh -unaugment -profileName default -
templatePath
config_root/AppServer2/profileTemplates/xd_augment/managed -
ignoreStack
```

- b. Run the **remove** command to unlink the WebSphere Virtual Enterprise files. If the cell is customized with a WebSphere Virtual Enterprise Version 7.0.0 intermediate link, the value of the *install_root* variable is the intermediate link directory string, and not the actual installation root directory string.

```
cd install_root/bin
./removeVE700.sh install_root config_root/AppServer2 userID
userPW > /tmp/rmVE700APP2.output
```

5. Unaugment, and unlink WebSphere Virtual Enterprise Version 7.0.0 from the application nodes that do not contain an ODR. In the following example, *AppServer* is a federated application server node.
 - a. Run the **mangeprofiles** command for the federated application server node.

If you created the application server using **zpmf** with the application server environment, run this command:

```
cd config_root/AppServer/bin
./manageprofiles.sh -unaugment -profileName default -
templatePath
/config_root/AppServer/profileTemplates/xd_augment/default -
ignoreStack
```

If you created the deployment manager and the application server together using **zpmf** with the cell (deployment manager and application server) environment, run this command:

```
cd config_root/AppServer/bin
./manageprofiles.sh -unaugment -profileName default -
templatePath
config_root/AppServer/profileTemplates/xd_augment/cell/default -
ignoreStack
```

- b. Run the **remove** command to unlink the WebSphere Virtual Enterprise files. If the cell is customized with a WebSphere Virtual Enterprise Version 7.0.0 intermediate link, the value of the *install_root* variable is the intermediate link directory string, and not the actual installation root directory string.

```
cd VE700_install_root/bin
./removeVE700.sh install_root config_root/AppServer userID
userPW > /tmp/rmVE700APP.output
```

6. Unmount the *install_root* hierarchical file system (HFS).
7. Start the cell.

Related information:

 [Program Directory for WebSphere Virtual Enterprise V7.0.0 for z/OS](#)

Migrating WebSphere Virtual Enterprise

Migrating involves copying the configuration from a previous release of WebSphere® Virtual Enterprise into a new release.

About this task

You can perform an incremental migration of your Version 6.1.0.x or Version 6.1.1.x configuration to Version 7.0. As a result, your system temporarily operates in a mixed cell environment in which you migrate the deployment manager to Version 7.0 before you migrate any nodes.

1. You migrate the deployment manager to Version 7.0. The deployment manager must be at the highest release and fix level within the cell.
2. You migrate the on demand routers (ODR) one at a time to Version 7.0. Each ODR must be at the same release and fix level as the deployment manager.
3. You migrate the application server nodes one at a time.

Note: Even though the method of incremental migration creates a mixed cell environment, avoid running the mixed cells for an extended period of time. You should have an existing plan in place to migrate your entire cell to the highest level to ensure consistent administration of the nodes.

[Migrating to a WebSphere Virtual Enterprise Version 7.0 cell](#)

Use **XDUUpgrade** command to migrate WebSphere Virtual Enterprise from a previous release to Version 7.0. Migrate the deployment manager first, and then migrate each application server node one at a time.

z/OS

[Migrating WebSphere Virtual Enterprise for z/OS](#)

Use this information to migrate to WebSphere Virtual Enterprise Version 7.0 for z/OS®.

Related tasks:

[Installing and customizing the product](#)

[Preparing the hosting environment for dynamic operations](#)

[Deploying and managing application editions without loss of service](#)

[Managing the WebSphere Virtual Enterprise environment](#)

[Monitoring operations](#)

[Configuring WebSphere Virtual Enterprise to work with other IBM products](#)

Related reference:

[Reference](#)

Related information:

[IBM WebSphere Virtual Enterprise Version 7.0](#)

[Product overview](#)

[Site map](#)

Migrating to a WebSphere Virtual Enterprise Version 7.0 cell

Use **XDUUpgrade** command to migrate WebSphere® Virtual Enterprise from a previous release to Version 7.0. Migrate the deployment manager first, and then migrate each application server node one at a time.

Before you begin

Update WebSphere Application Server Network Deployment Version 7.0.0.x to the latest maintenance level that is available. See the [Latest fix packs for WebSphere Application Server](#) for more information.

Procedure

1. Run the **backupConfig** command on the deployment manager and all nodes. The **backupConfig** command is a utility that is used to back up the configuration of the deployment manager and the nodes.
2. Install WebSphere Application Server Network Deployment Version 7.0.0.x on each deployment manager and node. Specify a different installation location than the location in which the previous version exists. For more information, see [Installing the product and additional software](#) in the WebSphere Application Server Network Deployment Version 7.0 Information Center.
3. Use the Profile Management Tool or the **manageprofiles** command to create your profiles for Version 7.0.0.x, but do not federate the nodes. See [Managing profiles](#) in the WebSphere Application Server Network Deployment Version 7.0 Information Center.
4. Migrate the deployment manager to your Version 7.0.0.x cell. You can use the **WASPreUpgrade** and **WASPostUpgrade** commands. The **WASPreUpgrade** command saves the configuration of Version 6.1.x to a migration-specific backup directory, and the **WASPostUpgrade** command retrieves the saved configuration. For more information, see the [WASPreUpgrade command](#) and [WASPostUpgrade command](#).
5. Install WebSphere Virtual Enterprise Version 7.0 in your environment. See [Planning the product installation](#) for more information.
6. Augment your profiles for WebSphere Virtual Enterprise Version 7.0 with the xd_augment profile template. The following example augments a deployment manager profile.

```
./manageprofile.sh|.bat -augment -profileName profile01 -templatePath  
install_root/profileTemplates/xd_augment/dmgr
```

7. Use the **VEUpgrade** command to migrate the deployment manager to Version 7.0.

To migrate the product using the **VEUpgrade** command, run the following script:

- o **Linux** **AIX** **HP-UX** **Solaris** target_profile/bin/VEUpgrade.sh
- o **Windows** target_profile\bin\VEUpgrade.bat

8. Migrate the remaining nodes one at a time. The procedure to migrate each node is similar to the preceding procedure, except WebSphere Application Server Network Deployment already exists in your environment, and you create and augment custom profiles instead of a deployment manager profile.
 - a. Use the Profile Management Tool or the **manageprofiles** command to create a WebSphere Application Server Network Deployment custom profile.
 - b. Augment the custom profile with the Virtual Enterprise profile template. The following example augments a custom profile with the xd_augment profile template.

```
./manageprofile.sh|.bat -augment -templatePath  
install_root/profileTemplates/xd_augment/managed
```

- c. Migrate the node to your WebSphere Application Server Network Deployment Version 7.0.0.x cell. Run the **WASPreUpgrade** and **WASPostUpgrade** commands.
- d. Migrate the node to WebSphere Virtual Enterprise Version 7.0. Run the **VEUpgrade** command.

Results

Your cell now contains WebSphere Virtual Enterprise Version 7.0.

[VEUpgrade command](#)

Use the **VEUpgrade** command to migrate product configuration data from an existing WebSphere Virtual Enterprise cell to a new WebSphere Virtual Enterprise Version 7.0 cell.

[Migrating large topologies of external WebSphere application servers](#)

You can manage previous versions of WebSphere product application servers in the WebSphere Virtual Enterprise administrative console. If you have a large topology where manually migrating the topology would be cumbersome, you can use the migration toolkit to automatically detect your configurations and import the configuration into the Version 6.1 or later cell.

[Using the interim fix tool to check for maintenance packages](#)

Use the interim fix tool to verify that your product environment contains the recommended interim fixes that are necessary for WebSphere Virtual Enterprise to function correctly.

Using the Update Installer to install maintenance packages

Use the IBM® Update Installer for WebSphere Software to update WebSphere Virtual Enterprise with various types of maintenance, such as interim fixes, fix packs, and refresh packs.

Parent topic: [Migrating WebSphere Virtual Enterprise](#)

VEUpgrade command

Use the **VEUpgrade** command to migrate product configuration data from an existing WebSphere® Virtual Enterprise cell to a new WebSphere Virtual Enterprise Version 7.0 cell.

Location

Run the command from the `target_profile_root/bin` directory.

Parameters

-userID

Specifies the administrative user name for authentication if security is enabled on the workstation. Because the migration utility does not check for the **-userID** parameter, you must specify it or the migration process fails. (Required)

-password

Specifies the administrative password for authentication if security is enabled on the workstation. Because the migration utility does not check for the **-password** parameter, you must specify it or the migration process fails. (Required)

-sourceWasHome

Specifies the `install_root` directory of the existing product installations from which the configuration is migrated. (Required)

-sourceProfilePath

Specifies the fully qualified path of the existing profile from which the configuration is migrated. (Required)

-targetProfilePath

Specifies the fully qualified path to the target profile. (Required)

-traceLevel

Specifies the trace level for generating diagnostic messages. (Optional)

-traceDir

Specifies the directory of the `XDUUpgrade.log` and `XDUUpgrade.trace` files. The default directory is `target_profile_root/logs`. (Optional)

Usage

The following example migrates the deployment manager to the new cell:

```
VEUpgrade.sh|.bat -userid user_name -password user_password  
-sourcewashome install_root/DeploymentManager -sourceprofilepath  
install_root/DeploymentManager/profiles/default  
-targetprofilepath target_profile_path/DeploymentManager/profiles/default
```

The following example migrates an application server to the new cell:

```
VEUpgrade.sh|.bat -userid user_name -password user_password  
-sourcewashome install_root/AppServer -sourceprofilepath  
install_root/AppServer/profiles/default  
-targetprofilepath target_profile_path/AppServer/profiles/default
```

Parent topic: [Migrating to a WebSphere Virtual Enterprise Version 7.0 cell](#)

Parent topic: [z/OS Migrating to a Version 7.0 for z/OS cell](#)

Related tasks:

[Distributed platforms Creating and augmenting profiles](#)

[z/OS Migrating to a Version 7.0 for z/OS cell](#)

Migrating large topologies of external WebSphere application servers

You can manage previous versions of WebSphere® product application servers in the WebSphere Virtual Enterprise administrative console. If you have a large topology where manually migrating the topology would be cumbersome, you can use the migration toolkit to automatically detect your configurations and import the configuration into the Version 6.1 or later cell.

Before you begin

- Install and federate your Version 7.0 middleware agents on your Version 6.1.x computers, including the deployment manager. For the purposes of this task, the Version 6.1.x deployment manager is referred to as the *previous deployment manager*. If you do not have the middleware agent running on your previous deployment manager, you can copy the AGENT_HOME/lib/legacycell.jar file and AGENT_HOME/bin/Extractor.sh|.bat file from another workstation that has the middleware agent deployed to the previous deployment manager.
- Install or augment your deployment manager for Version 7.0.

About this task

The migration toolkit can automatically detect older WebSphere configurations and import them into a Version 7.0 cell.

The migration toolkit uses series of scripts to connect to one cell, read the data, and then import the data into the other cell. Because of the version differences, data must be manipulated. The scripts also function as an adaptor or integration interface. The migration script logic handles all of the routing information and environment detection, sets data to makes things work after the import, and performs integrity checks to ensure a stable configuration.

For more information about the migration toolkit capabilities, see [Migration toolkit](#).

Procedure

1. Generate the mappings file. The mappings file defines additional data that cannot be gathered automatically. You must generate the mappings file before you import the middleware deployment.
 - a. Log in to the previous deployment manager computer.
 - b. On the deployment manager computer, go to the AGENT_HOME/bin directory, or the directory to which you copied the middleware agent files from another workstation.
 - c. Run the Extractor.sh or Extractor.bat script. For example, run the following command:

```
./Extractor.sh install_root/config
```

The single install_root directory is the default for the deployment manager profile directory. If install_root does not specify the deployment manager profile directory as the default, pass in install_root/profiles/deployment_manager_profile/config as an argument.

When this script completes, a Mapping.dat file is in the directory from which you ran the script.

- d. Copy the Mapping.dat file to the install_root/bin directory on your WebSphere Virtual Enterprise Version 7.0 deployment manager.
2. Run the automation script.
 - a. Log in to your WebSphere Virtual Enterprise Version 7.0 deployment manager computer.
 - b. On the deployment manager computer, navigate to the install_root/bin directory.
 - c. Run the MirrorCell.bat or MirrorCell.sh script. The MirrorCell script uses the following format:

```
MirrorCell.* [-props PROPS] [-mode MODE] [-nosec, -nosave]
```

The only required parameter is the -props parameter, which specifies the full path to the properties file. For more information about the MirrorCell script parameters, see [MirrorCell.bat|.sh script](#).

Note: If you include paths that contain spaces in your script call, surround these paths with quotation marks, for example:

```
"C:\Program Files"
```

By default, the script connects to the previous deployment manager and generates any necessary delta information between the newly detected configuration and a copy of the last known configuration. The script then uses this delta information to recreate the configuration within the Version 7.0 cell. Status updates are reported at each milestone.

- d. Log in to the WebSphere Virtual Enterprise cell and verify that the configuration was created properly. To check that your servers are represented in the console, click **Servers > Other middleware servers > External WebSphere application servers**. To check that representations of your applications were created, click **Applications > All applications**.

What to do next

You can manage your previous configurations from the WebSphere Virtual Enterprise cell.

[Migration toolkit](#)

You can use the migration toolkit to automatically detect your previous configurations of WebSphere Application Server and import the configuration into the WebSphere Virtual Enterprise cell.

Parent topic: [Migrating to a WebSphere Virtual Enterprise Version 7.0 cell](#)

Related tasks:

[Configuring external WebSphere application servers](#)

Related reference:

[MirrorCell.bat|.sh script](#)

Migration toolkit

You can use the migration toolkit to automatically detect your previous configurations of WebSphere® Application Server and import the configuration into the WebSphere Virtual Enterprise cell.

Toolkit capabilities

- **Bidirectional and atomic automation:** With the migration toolkit, you can read from an older cell configuration and create the Version 7.0 configuration in an atomic process. You do not need to manually make any updates after the migration process begins.
- **Multiple migration modes:** You can also phase your migration. For example, you can read the data from the previous cell and manipulate the data before continuing with the migration.
- **Delta generation:** The migration toolkit scripts persist a copy of the previously generated environment data. When you rerun the scripts, the scripts create a delta between the configuration versions and can update and delete servers or applications as needed. Only major configurations such as servers, applications, and modules are factored into the delta. If port or endpoint information changes, for example, you must modify the data directly from the administrative console.
- **Automatic data entry:** The migration toolkit scripts automatically preload data into your cell.
- **Limited dynamic clustering:** After the scripts complete, your static clusters are represented as dynamic clusters. Dynamic clusters can start and stop servers as your service policies or demand requires. The dynamic clustering capabilities for your migrated servers are limited. The limitations that apply to dynamic clusters of assisted life cycle middleware servers also apply to the dynamic clusters that the migration toolkit creates.
- **HTTP traffic shaping:** Because the applications are registered within the WebSphere Virtual Enterprise configuration, the on demand router (ODR) can route to your clustered applications based on the demand and statistical analysis of your HTTP traffic. Traffic can be routed to your Version 6.1.x applications in the same way that traffic can be routed to Version 7.0 and later applications. Service and routing policies can be applied.

Functional limitations

- The representations of your servers receive server status from the middleware agent. The middleware agent is installed on your previous server and provides the status information. However, this status might not be accurately represented at all times.
- You cannot fully monitor middleware applications from your Version 7.0 cell. The status of the applications is tied to the status of the target servers or clusters with which they are associated. When the corresponding servers or clusters start, so do the middleware applications, independent of the actual application status.

Parent topic: [Migrating large topologies of external WebSphere application servers](#)

Related concepts:

[Dynamic clusters](#)

Related reference:

[MirrorCell.bat|.sh script](#)

Using the interim fix tool to check for maintenance packages

Use the interim fix tool to verify that your product environment contains the recommended interim fixes that are necessary for WebSphere® Virtual Enterprise to function correctly

Before you begin

- You can use the interim fix tool with WebSphere Virtual Enterprise Version 6.1.0 and later. The tool is not supported for Version 6.0.2.
- You must run the tool on the deployment manager.

About this task

Run the interim fix tool on the deployment manager to check if any recommended interim fixes for Virtual Enterprise are not installed in your cell. Because Virtual Enterprise runs in an existing WebSphere Application Server Network Deployment environment, the tool checks for recommended interim fixes for both products. The tool creates two ZIP files: one file contains the interim fixes for WebSphere Application Server Network Deployment and one file contains the interim fixes for Virtual Enterprise.

Note: The recommended interim fixes for WebSphere Application Server Network Deployment include only the interim fixes that are needed for the features of Virtual Enterprise to operate and function correctly. However, you might need to install other WebSphere Application Server Network Deployment fixes that are not identified as recommended by the interim fix tool. See [Recommended fixes for WebSphere Application Server](#) for more information.

Procedure

To start the interim fix tool, run the `install_root/bin/ifixtool.sh|.bat` script, and specify the following commands.

downloadAndCheckCell

Downloads the interim fixes that are not yet applied to the cell.

checkCell ND path_to_ND_zip_file

Creates a ZIP file containing only the WebSphere Application Server Network Deployment interim fixes that are needed for the cell.

checkCell VE path_to_VE_zip_file

Creates a ZIP file containing only the Virtual Enterprise interim fixes that are needed for the cell.

download ND product_version

Downloads a ZIP file containing WebSphere Application Server Network Deployment interim fixes to the current directory.

download VE product_version

Downloads a ZIP file containing Virtual Enterprise interim fixes to the current directory.

If your deployment manager node has Internet access, you can use the **downloadAndCheckCell** command. Assuming there are both WebSphere Application Server Network Deployment and WebSphere Virtual Enterprise interim fixes for the product releases in your cell, the **downloadAndCheckCell** command creates two ZIP files of interim fixes in the `install_root/cimrepos` directory.

If your deployment manager node does not have Internet access, manually download the appropriate ZIP files for each product release from <ftp://ftp.software.ibm.com/software/websphere/extended/support/ifixInfo/latest/> and transfer the files to the deployment manager node. You can then run the **checkCell ND path_to_ND_zip_file** and **checkCell VE path_to_VE_zip_file** commands separately to create the corresponding ZIP file for each product in the `install_root/cimrepos` directory.

Each ZIP file in the `install_root/cimrepos` directory contains a `README.txt` file and a `HIERARCHY.txt` file. The `README.txt` files provide a list and descriptions of the interim fixes. The `HIERARCHY.txt` files list any superseded interim fixes. You can review the lists of interim fixes to determine which ones you want to install.

Parent topic: [Migrating to a WebSphere Virtual Enterprise Version 7.0 cell](#)

Related tasks:

[Using the Update Installer to install maintenance packages](#)

[Using the centralized installation manager to manage your environment](#)

[Using ISMP programs to install the product](#)

Related reference:

[ifixtool.sh|.bat script](#)

Related information:

[Recommended fixes for WebSphere Extended Deployment](#)

Using the Update Installer to install maintenance packages

Use the IBM® Update Installer for WebSphere® Software to update WebSphere Virtual Enterprise with various types of maintenance, such as interim fixes, fix packs, and refresh packs.

Before you begin

Install the product.

About this task

Use the Update Installer to install and apply various types of maintenance packages for WebSphere Virtual Enterprise. Because the Update Installer undergoes regular maintenance, you must use the most current version of the tool.

Procedure

1. Stop all processes that are running in your environment.

To stop the deployment manager, run the following command from the `profile_root/bin` directory of the deployment manager profile:

- o `Linux` ▶ `AIX` ▶ `HP-UX` ▶ `Solaris` `./stopManager.sh`
- o `Windows` `stopManager.bat`

To stop each node agent, run the following command from the `profile_root/bin` directory of each node:

- o `Linux` ▶ `AIX` ▶ `HP-UX` ▶ `Solaris` `./stopNode.sh`
- o `Windows` `stopNode.bat`

2. Download the latest version of the Update Installer. See [Recommended fixes](#) for more information.
3. Install the Update Installer. See [Installing the Update Installer for WebSphere Software](#) in the WebSphere Application Server Network Deployment Information Center for more information.

Attention: To enable the centralized installation manager to call the copy of the Update Installer that you install, you must install the Update Installer into the `WAS_HOME/UpdateInstaller` directory. Otherwise, the centralized installation manager installs an additional copy of the Update Installer into the `WAS_HOME/UpdateInstaller` directory.

4. Download into the `updi_root/maintenance` directory the maintenance packages that you intend to install. See the [Support site](#) for more information.
5. Use the Update Installer to install the maintenance package. You can install the maintenance package by running the graphical user interface (GUI), or by running the Update Installer in silent mode.

Run the following command from the `updi_root` directory to start the GUI:

- o `Linux` ▶ `AIX` ▶ `HP-UX` ▶ `Solaris` `./update.sh`
- o `Windows` `update.bat`

Run the following command from the `updi_root` directory to run the Update Installer in silent mode:

- o `Linux` ▶ `AIX` ▶ `HP-UX` ▶ `Solaris` `./update.sh -silent -options responsefile/file_name`
- o `Windows` `update.bat -silent -options responsefile\file_name`

If the installation process fails, refer to the temporary log file, which is located in the `updi_root/logs/update/tmp` directory.

Results

Your product environment is now updated with the latest maintenance packages. The Update Installer creates the `install_root/logs/update/maintenance_package.install` directory in which the installation log files are located.

[Updating the middleware agent](#)

Install WebSphere Extended Deployment Version 6.1 Fix Pack 3 (6.1.0.3) and the IBM software development kit (SDK), Java™ Technology Edition 1.5 SR7 Cumulative Fix for WebSphere Application Server Version 6.1.0.17 to update the middleware agent to the latest maintenance levels.

Parent topic: [Migrating to a WebSphere Virtual Enterprise Version 7.0 cell](#)

Related tasks:

[Using the interim fix tool to check for maintenance packages](#)

[Using ISMP programs to install the product](#)

[Using the centralized installation manager to manage your environment](#)

Related information:

[update command](#)

Updating the middleware agent

Install WebSphere® Extended Deployment Version 6.1 Fix Pack 3 (6.1.0.3) and the IBM® software development kit (SDK), Java™ Technology Edition 1.5 SR7 Cumulative Fix for WebSphere Application Server Version 6.1.0.17 to update the middleware agent to the latest maintenance levels.

Before you begin

Download and install the IBM Update Installer for WebSphere Software. See [Using the Update Installer to install maintenance packages](#) for more information.

About this task

Use the Update Installer to apply maintenance packages to your existing middleware agent installations. To successfully update the middleware agent with WebSphere Extended Deployment Version 6.1.0.3, you must also install the IBM SDK, Java Technology Edition 1.5 SR7 Cumulative Fix for WebSphere Application Server Version 6.1.0.17.






Procedure

1. Download the IBM SDK 1.5 SR7 Cumulative Fix for WebSphere Application Server Version 6.1.0.17 into the updi_root/maintenance directory. See the [IBM SDK 1.5 SR7 Cumulative Fix for WebSphere Application Server download Web site](#).
2. Download WebSphere Extended Deployment Version 6.1.0.3 into the updi_root/maintenance directory. See the [Support site for WebSphere Extended Deployment](#) for more information.
3. Use the Update Installer to individually install each maintenance package. You can install the maintenance packages by running the graphical user interface (GUI), or by running the Update Installer in silent mode.

Issue the following command from the updi_root directory to start the GUI:

- o  update.bat
- o     ./update.sh

Issue the following command from the updi_root directory to run the Update Installer in silent mode:

- o  update.bat -silent -options *responsefile\file_name*
- o     ./update.sh -silent -options *responsefile/file_name*

If the installation process fails, then refer to the temporary log file, which is located in the updi_root/logs/update/tmp directory.

Click **Relaunch** on the last panel of the Update Installer to install the second maintenance package after the tool installs the first maintenance package.

Parent topic: [Using the Update Installer to install maintenance packages](#)

Related concepts:

[Middleware agent](#)

Related tasks:

[Installing the middleware agent](#)

Migrating WebSphere Virtual Enterprise for z/OS

Use this information to migrate to WebSphere® Virtual Enterprise Version 7.0 for z/OS®.

z/OS

[Migrating the product from Version 6.1.0.5 or Version 6.1.1 to Version 7.0](#)

You can migrate WebSphere Virtual Enterprise for z/OS from Version 6.1.0.5 or Version 6.1.1 to Version 7.0.

z/OS

[Migrating to a Version 7.0 for z/OS cell](#)

You can migrate your existing product configurations to a WebSphere Virtual Enterprise Version 7.0 for z/OS and WebSphere Application Server Network Deployment Version 7.0 for z/OS cell.

Parent topic: [Migrating WebSphere Virtual Enterprise](#)

Related tasks:

[Customizing WebSphere Virtual Enterprise for z/OS](#)

Related information:

[WebSphere Virtual Enterprise Library page](#)

Migrating the product from Version 6.1.0.5 or Version 6.1.1 to Version 7.0

You can migrate WebSphere® Virtual Enterprise for z/OS® from Version 6.1.0.5 or Version 6.1.1 to Version 7.0.

Before you begin

- Ensure that WebSphere Virtual Enterprise Version 6.1.0.5 for z/OS or WebSphere Virtual Enterprise Version 6.1.1 for z/OS is installed and running in your current cell. See the Program Directory on the [Library page](#) for more information.
- To meet minimum requirements, update WebSphere Application Server Network Deployment for z/OS to Version 7.0.0.17. See the [Latest fix packs for WebSphere Application Server](#) for more information.
- Start all servers on all nodes to incorporate the changes for the WebSphere Application Server Network Deployment upgrade.
- Stop all servers.

About this task

The process of migrating WebSphere Virtual Enterprise from Version 6.1.0.5 or Version 6.1.1 to Version 7.0 while maintaining the same version of WebSphere Application Server Network Deployment is known as an *upgrade*. To complete the following procedure, upgrade your deployment manager first. Next, upgrade each on demand router (ODR) before you migrate your application servers. To complete the migration process successfully, you must follow the steps as outlined, and you must run the commands within the steps as outlined.

Note: If either WebSphere eXtreme Scale Version 6.1.x or WebSphere Extended Deployment Compute Grid Version 6.1.x, or both, are configured in your environment, you must complete the steps to remove the products from the deployment manager and all other nodes in the cell. The commands to remove the products are outlined in the steps of the migration procedure.

Procedure

1. Create a copy of the current config HFS or config HFSES for the cell. This copy is used as input to the **VEUpgrade** command.
2. Mount the new product HFS for WebSphere Virtual Enterprise Version 7.0 in the `/usr/lpp/zWebSphereVE/V7R0` directory.
3. Enable access to the new WebSphere Virtual Enterprise Version 7.0 file system. Edit the `server.policy` file. You can use the policy tool in the `WAS_HOME/java/jre/bin` directory to edit the policy file, or you can manually edit the policy file. If you choose to manually edit the file, set permissions for the deployment manager and all of the application server nodes.

To set permissions for the deployment manager, add the following lines to the `server.policy` file in the `/WAS_config_root/DeploymentManager/profiles/default/properties` directory:

```
grant codeBase "file:/usr/lpp/zWebSphereVE/V7R0/lib/-" { permission
java.security.AllPermission; };
grant codeBase "file:/usr/lpp/zWebSphereVE/V7R0/-" { permission
java.security.AllPermission; };
```

To set permissions for each application server node, add the following two lines to the `server.policy` file in the `/WAS_config_root/AppServer/profiles/default/properties` directory:

```
grant codeBase "file:/usr/lpp/zWebSphereVE/V7R0/lib/-" { permission
java.security.AllPermission; };
grant codeBase "file:/usr/lpp/zWebSphereVE/V7R0/-" { permission
java.security.AllPermission; };
```

4. Optional: Customize your current cell with a WebSphere Virtual Enterprise Version 7.0 intermediate link. Run the following command to point an intermediate link to the WebSphere Virtual Enterprise Version 7.0 installation root. Specify the installation root as the value of the `VE700_install_root` variable, and specify the name of a new and unique directory as the value of the `symlinkName` variable.

```
ln -s VE700_install_root symlinkName
```

See the following usage example:

```
ln -s /usr/lpp/zWebSphereVE/V7R0/WebSphere/ND/myVE700link
```

5. Migrate the deployment manager from Version 6.1.0.5 or Version 6.1.1 to WebSphere Virtual Enterprise Version 7.0. If WebSphere Extended Deployment Compute Grid or WebSphere eXtreme Scale are installed, run the **removeCG610** and **removeOG610** commands, which removes Compute Grid and eXtreme Scale for both Version 6.1.0.5 and Version 6.1.1 even though the command names contain the 610 string.

- a. Unaugment the existing Version 6.1.0.5 or Version 6.1.1 product.
 - i. Run the following command if Compute Grid is installed.

```
cd WAS_config_root/DeploymentManager/bin
./manageprofiles.sh -unaugment -profileName default
-templatePath
WAS_config_root/DeploymentManager/profileTemplates/wxdcg_augment
/dmgr -ignoreStack
```

- ii. Run the following command if eXtreme Scale is installed.

```
cd WAS_config_root/DeploymentManager/bin
./manageprofiles.sh -unaugment -profileName default
-templatePath
WAS_config_root/DeploymentManager/profileTemplates/wxddg_augment
/dmgr -ignoreStack
```

- iii. Run the following command to remove Virtual Enterprise.

```
cd WAS_config_root/DeploymentManager/bin
./manageprofiles.sh -unaugment -profileName default
-templatePath
WAS_config_root/DeploymentManager/profileTemplates/xd_augment/dm
gr -ignoreStack
```

- b. Run the commands for the deployment manager from the WebSphere Virtual Enterprise Version 7.0 VE700_install_root/bin directory as shown in the following examples. If the current cell is customized with a WebSphere Virtual Enterprise Version 7.0 intermediate link, the value of the VE700_install_root variable is the intermediate link directory and not the actual installation root directory string.

- i. Run the following command if Compute Grid is installed.

```
removeCG610.sh VE700_install_root
WAS_config_root/DeploymentManager user_name user_password
```

- ii. Run the following command if eXtreme Scale is installed.

```
removeOG610.sh VE700_install_root
WAS_config_root/DeploymentManager user_name user_password
```

- iii. Run the **removeOps610** command as shown in the following example.

```
removeOps610.sh VE700_install_root
WAS_config_root/DeploymentManager user_name user_password
```

- iv. Run the **removeBase610** command as shown in the following example.

```
removeBase610.sh VE700_install_root
WAS_config_root/DeploymentManager user_name user_password
```

6. Augment the deployment manager node profile for WebSphere Virtual Enterprise Version 7.0 for z/OS. See [Customizing WebSphere Virtual Enterprise for z/OS](#) for more information.
7. Run the **VEUpgrade** command as shown in the following example.

Important: The **-sourceshome** and **-sourceprofilepath** parameters point to the copy of the HFS that you created in step 1.

```
cd WAS_config_root/DeploymentManager/bin
VEUpgrade.sh -userid user_name -password user_password
-sourceshome copy_WAS_config_root/DeploymentManager
-soureprofilepath copy_WAS_config_root/DeploymentManager/profiles/default
-targetprofilepath WAS_config_root/DeploymentManager/profiles/default
```

8. Start the deployment manager.
9. Migrate the ODR node from Version 6.1.0.5 or Version 6.1.1 to WebSphere Virtual Enterprise Version 7.0. If WebSphere Extended Deployment Compute Grid or WebSphere eXtreme Scale are installed, run the **removeCG610** and **removeOG610** commands, which removes Compute Grid and eXtreme Scale for both Version 6.1.0.5 and Version 6.1.1 even though the command names contain the 610 string.

- a. Unaugment the existing Version 6.1.0.5 or Version 6.1.1 product.
 - i. Run the following command if Compute Grid is installed. In the following example, it is assumed that the node type is a federated application server node. If it is a custom managed node, change the wxdcg_augment/default string to wxdcg_augment/managed.

```
cd WAS_config_root/AppServer/bin
./manageprofiles.sh -unaugment -profileName default
-templatePath
WAS_config_root/AppServer/profileTemplates/wxdcg_augment/default
-ignoreStack
```

- ii. Run the following command if eXtreme Scale is installed. In the following example, it is assumed that the node type is a federated application server node. If it is a custom managed node, change the wxddg_augment/default string to wxddg_augment/managed.

```
cd WAS_config_root/AppServer/bin
./manageprofiles.sh -unaugment -profileName default
-templatePath
WAS_config_root/AppServer/profileTemplates/wxddg_augment/default
-ignoreStack
```

- iii. Run the following command to remove Virtual Enterprise. In the following example, it is assumed that the node type is a federated application server node. If it is a custom managed node, change the wxd_augment/default string to wxd_augment/managed.

```
cd WAS_config_root/AppServer/bin
./manageprofiles.sh -unaugment -profileName default
-templatePath
WAS_config_root/AppServer/profileTemplates/xd_augment/default -
ignoreStack
```

- b. Run the commands from the WebSphere Virtual Enterprise Version 7.0 VE700_install_root/bin directory as shown in the following examples. If the current cell is customized with a WebSphere Virtual Enterprise Version 7.0 intermediate link, the value of the VE700_install_root variable is the intermediate link directory and not the actual installation root directory string.

- i. Run the following command if Compute Grid is installed.

```
removeCG610.sh VE700_install_root WAS_config_root/AppServer
user_name user_password
```

- ii. Run the following command if eXtreme Scale is installed.

```
removeOG610.sh VE700_install_root WAS_config_root/AppServer
user_name user_password
```

- iii. Run the **removeOps610** command as shown in the following example.

```
removeOps610.sh VE700_install_root WAS_config_root/AppServer
user_name user_password
```

- iv. Run the **removeBase610** command as shown in the following example.

```
removeBase610.sh VE700_install_root WAS_config_root/AppServer
user_name user_password
```

10. Augment the application server node for WebSphere Virtual Enterprise Version 7.0 for z/OS. See [Customizing WebSphere Virtual Enterprise for z/OS](#) for more information.

11. Run the **VEUpgrade** command as shown in the following example.

Important: The **-sourceshome** and **-sourceprofilepath** parameters point to the copy of the HFS that you created in step 1.

```
cd WAS_config_root/AppServer/bin
VEUpgrade.sh -userid user_name -password user_password
-sourceshome copy_WAS_config_root/AppServer
-sourceprofilepath copy_WAS_config_root/AppServer/profiles/default
-targetprofilepath WAS_config_root/AppServer/profiles/default
```

12. Complete steps 9, 10, and 11 to migrate, augment, and upgrade each additional application server node

to WebSphere Virtual Enterprise Version 7.0 for z/OS. Remember to specify the correct path name for each node.

13. Stop the deployment manager.
14. Start your new Version 7.0 cell.
15. Optional: You can delete the copy of the HFS that you created in step 1 after you verify that the migration is complete.

Parent topic: [z/OS Migrating WebSphere Virtual Enterprise for z/OS](#)

Related tasks:

[Migrating to a Version 7.0 for z/OS cell](#)

Related reference:

[manageprofiles command](#)

Migrating to a Version 7.0 for z/OS cell

You can migrate your existing product configurations to a WebSphere® Virtual Enterprise Version 7.0 for z/OS® and WebSphere Application Server Network Deployment Version 7.0 for z/OS cell.

Before you begin

- To meet minimum requirements, update WebSphere Application Server Network Deployment for z/OS to Version 7.0.0.17. See the [Latest fix packs for WebSphere Application Server](#) for more information.
- Install WebSphere Application Server Network Deployment Version 7.0 for z/OS. Use the same cell name and node names that you used for the cell name and node names of your existing WebSphere Application Server Network Deployment Version 6.1.0.x configuration. See [Installing your application serving environment](#) for more information.
- Install WebSphere Virtual Enterprise Version 7.0 for z/OS. Use the same cell name and node names that you used for the cell name and node names of your existing Version 6.1.1 configuration. See the Program Directory on the [Library page](#) for more information.
- Augment the deployment manager and application server node profiles for WebSphere Virtual Enterprise Version 7.0 for z/OS. See [Customizing WebSphere Virtual Enterprise for z/OS](#) for more information.

z/OS [Migrating the deployment manager for z/OS](#)

You can migrate the deployment manager from your current Version 6.1.0.x for z/OS and Version 6.1.1 for z/OS cell to a new WebSphere Application Server Network Deployment Version 7.0 for z/OS and WebSphere Extended Deployment Version 7.0 cell.

z/OS [Migrating an application server node for z/OS](#)

You can migrate an application server node from your current WebSphere Application Server Network Deployment Version 6.1.0.x for z/OS and WebSphere Virtual Enterprise Version 6.1.1 for z/OS cell to a new WebSphere Application Server Network Deployment Version 7.0 for z/OS and WebSphere Virtual Enterprise Version 7.0 for z/OS cell.

z/OS [VEUpgrade command](#)

Use the **VEUpgrade** command to migrate product configuration data from an existing WebSphere Virtual Enterprise cell to a new WebSphere Virtual Enterprise Version 7.0 cell.

Parent topic: **z/OS** [Migrating WebSphere Virtual Enterprise for z/OS](#)

Related tasks:

[Migrating the product from Version 6.1.0.5 or Version 6.1.1 to Version 7.0](#)

Related reference:

[VEUpgrade command](#)

Migrating the deployment manager for z/OS

You can migrate the deployment manager from your current Version 6.1.0.x for z/OS® and Version 6.1.1 for z/OS cell to a new WebSphere® Application Server Network Deployment Version 7.0 for z/OS and WebSphere Extended Deployment Version 7.0 cell.

Before you begin

Install and configure your WebSphere Virtual Enterprise Version 7.0 for z/OS and WebSphere Application Server Network Deployment Version 7.0 for z/OS cell.

Procedure

1. Create and mount a new configuration file system as the target of the migration.
2. Mount the product HFS for WebSphere Application Server Network Deployment Version 7.0 for z/OS at `/usr/lpp/zWebSphere/V7R0`.
3. Use the z/OS Migration Management Tool or the `zmmt` command to generate and run the Job Control Language (JCL) jobs for WebSphere Application Server Network Deployment Version 7.0 for z/OS. See [Migrating product configurations with migration tools](#) for more information.
4. Use the WebSphere Customization Toolbox to generate the Version 7.0 augmentation job for the deployment manager. Specify the directory name that you specified in step 1 as the mount point of the configuration file system. See [Generating customization definitions](#) for more information.
5. Use the WebSphere Customization Toolbox to upload and run the augmentation job. See [Uploading and running customized jobs](#) for more information.
6. Run the **VEUpgrade** command to copy the existing configuration to the new Version 7.0 cell as shown in the following example:

```
VEUpgrade.sh -userID user_name -password user_password -sourceWasHome  
install_root/DeploymentManager  
-sourceProfilePath install_root/DeploymentManager/profiles/default -  
targetProfilePath  
configuration_root/DeploymentManager/profiles/default
```

Note: You must run the **VEUpgrade** command from the `configuration_root/DeploymentManager/bin` directory.

After you run the **VEUpgrade** command, review the `VEUpgrade.log` and `VEUpgrade.trace` files in the `configuration_root/DeploymentManager/logs` directory.

7. Start the new deployment manager.

What to do next

Migrate each application server node one at a time.

Parent topic: [z/OS Migrating to a Version 7.0 for z/OS cell](#)

Related tasks:

[Migrating an application server node for z/OS](#)
[Customizing WebSphere Virtual Enterprise for z/OS](#)

Related reference:

[VEUpgrade command](#)

Migrating an application server node for z/OS

You can migrate an application server node from your current WebSphere® Application Server Network Deployment Version 6.1.0.x for z/OS® and WebSphere Virtual Enterprise Version 6.1.1 for z/OS cell to a new WebSphere Application Server Network Deployment Version 7.0 for z/OS and WebSphere Virtual Enterprise Version 7.0 for z/OS cell.

Before you begin

Migrate the deployment manager.

Procedure

1. Create and mount a new configuration file system as the target of the migration.
2. Mount the product HFS for WebSphere Application Server Network Deployment Version 7.0.x for z/OS at `/usr/lpp/zWebSphere/V7R0`.
3. Use the z/OS Migration Management Tool or `zmmt` command to generate and run the Job Control Language (JCL) jobs for Version 7.0.x. See [Migrating product configurations with migration tools](#) for more information.
4. Use the WebSphere Customization Toolbox to generate the WebSphere Virtual Enterprise Version 7.0 for z/OS augmentation job for each node. Specify the directory name that you specified in step 1 as the mount point of the configuration file system. See [Generating customization definitions](#) for more information.
5. Use the WebSphere Customization Toolbox to upload and run the augmentation job for each node. See [Uploading and running customized jobs](#) for more information.
6. Start the deployment manager if it is not already started.
7. Run the **VEUpgrade** command to copy the existing configuration to the new Version 7.0 cell as shown in the following example:

```
VEUpgrade.sh -userID user_name -password user_password -sourceWasHome  
install_root/AppServer  
-sourceProfilePath install_root/AppServer/profiles/default -  
targetProfilePath configuration_root/AppServer/profiles/default
```

Note: You must run the **VEUpgrade** command from the `configuration_root/AppServer/bin` directory.

After you run the **VEUpgrade** command, review the `VEUpgrade.log` and `VEUpgrade.trace` files in the `configuration_root/AppServer/logs` directory.

8. Start each node.

Parent topic: [z/OS Migrating to a Version 7.0 for z/OS cell](#)

Related tasks:

[Migrating the deployment manager for z/OS](#)
[Customizing WebSphere Virtual Enterprise for z/OS](#)

Related reference:

[VEUpgrade command](#)

Preparing the hosting environment for dynamic operations

Dynamic operations monitor the middleware server environment and can make recommendations or changes based on the data that is observed. To prepare the hosting environment for dynamic operations, you must federate middleware nodes, create the on demand router (ODR), create middleware servers and dynamic clusters, and deploy applications to your servers and dynamic clusters with an associated service policy.

Before you begin

- [Install the product.](#)
- [Install the middleware agent.](#)

About this task

To set up the dynamic operations environment you must create ODRs, servers and dynamic clusters. These actions set up the application server virtualization portion. You can then deploy your applications and define service policies, which configures application virtualization. The autonomic managers work with these elements to maximize the performance of your environment.

[Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

To configure WebSphere Virtual Enterprise to work with VMware Infrastructure 3 platforms, you must configure security so that the servers can communicate with each other and configure custom properties on your deployment manager to define the vCenter or ESX servers.

[Federating middleware nodes](#)

You can use the command line to federate middleware nodes to a WebSphere Virtual Enterprise cell.

[Creating and configuring ODRs](#)

The on demand router (ODR) is an intelligent HTTP and Session Initiation Protocol (SIP) proxy server in WebSphere Virtual Enterprise. The ODR is the point of entry into a WebSphere Virtual Enterprise environment and is a gateway through which HTTP requests and Session Initiation Protocol (SIP) messages flow to back-end application servers. You can configure the ODR to determine how it handles failure scenarios and how it tunes certain work requests.

[Creating dynamic clusters](#)

When you create a dynamic cluster, weights are used to dynamically balance the workloads of the cluster members, based on performance information that is collected from the cluster members. Creating dynamic clusters enables application server virtualization.

[Adding middleware servers to configurations](#)

Using middleware servers, you can manage all of the servers in your environment, including servers that are not created with WebSphere Virtual Enterprise, by using the product administrative domain.

[Deploying applications with defined service levels](#)

By deploying applications to your middleware servers and dynamic clusters and defining service policies on those applications, you can meet the performance goals for your environment.

[Routing and service policies](#)

Two types of policies are applied to a request: routing and service. You can create routing policies for HTTP and SOAP requests, and you can create service policies for HTTP, IIOP, SOAP, JMS, and SIP requests. Additionally, work classes can contain classification rules for both policy types with the exception of JMS. Classification rules are not supported for JMS work classes.

[Service policy rules](#)

Service policy rules define which service policy goal should be used to service a single piece of incoming work (request through the on demand router (ODR)). This is done by associating a transaction class with a Boolean expression. Each transaction class is associated with exactly one service policy goal. The Boolean expression can be customized to match any specific piece of work (request through the ODR).

Related tasks:

- [Installing and customizing the product](#)
- [Migrating WebSphere Virtual Enterprise](#)
- [Deploying and managing application editions without loss of service](#)
- [Managing the WebSphere Virtual Enterprise environment](#)
- [Monitoring operations](#)
- [Configuring WebSphere Virtual Enterprise to work with other IBM products](#)

Related reference:

- [Reference](#)

Related information:

- [IBM WebSphere Virtual Enterprise Version 7.0](#)
- [Product overview](#)
- [Site map](#)

Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise

To configure WebSphere® Virtual Enterprise to work with VMware Infrastructure 3 platforms, you must configure security so that the servers can communicate with each other and configure custom properties on your deployment manager to define the vCenter or ESX servers.

Before you begin

- Configure the VMware Infrastructure 3 platforms environment on your physical servers. Your VMware Infrastructure 3 platforms environment must meet the following requirements:
 - Your VMware Infrastructure 3 platforms environment must be on servers that are running Solaris Operating Environment on Intel hardware, Windows, or Linux x86 operating systems.
 - You must use VMware products that support VMware Infrastructure 3 platforms. The supported versions are:
 - VMware VirtualCenter Version 2.5
 - VMware ESX Version 3.5
 - VMware vSphere Version 4.0, which includes VMware ESXi and VMware vCenter ServerThe documentation generically refers to these servers with the following terminology:
 - **ESX server:** Refers to VMware ESX Version 3.5 or a VMware ESXi server in VMware vSphere Version 4.0.
 - **vCenter server:** Refers to VMware VirtualCenter Version 2.5 or a VMware vCenter server in VMware vSphere Version 4.0.
- Install and configure WebSphere Virtual Enterprise on each virtual machine.

About this task

When you have multiple nodes running on a physical computer with VMware Infrastructure 3 platforms, WebSphere Virtual Enterprise can contact VMware through Web services. You can configure this communication in the administrative console by creating cell-wide custom properties. These custom properties define the URL, user ID, and password for the vCenter or ESX servers. You also must configure your key stores to retrieve signers from the vCenter or ESX servers.

How you configure your VMware environment to work with WebSphere Virtual Enterprise is dependent on your VMware configuration. You must create the custom properties for enough of the servers in your environment to make WebSphere Virtual Enterprise aware of all of the virtual machines and physical computers. Set the custom properties on the cell level by clicking **System administration > Cell > Custom properties**.

- If you are using only ESX servers, you must configure the enough of the individual servers to make WebSphere Virtual Enterprise aware of the physical servers and virtual machines in the environment.
- If you are using a vCenter server to manage your environment, you can connect to the vCenter server, which establishes communication with all of the virtual machines and servers that the vCenter server manages. You do not need to connect to each ESX server. If a vCenter is available, the best practice is to connect to the vCenter server instead of each ESX server.
- If you are running multiple vCenter servers with a Microsoft Cluster Server (MSCS) to provide high availability, you can configure the key stores and custom properties for each vCenter server.

If you do not configure WebSphere Virtual Enterprise to work with VMware Infrastructure 3 platforms, the WebSphere Virtual Enterprise environment does not understand that the nodes are on virtual machines, and as a result, the machine processor or memory might be overloaded.

Procedure

- **If you are configuring WebSphere Virtual Enterprise to communicate with a vCenter server:**

1. Retrieve and store a signer certificate from the vCenter server and configure WebSphere Virtual Enterprise to communicate with the vCenter server:

```
./wsadmin.sh -lang jython -f retrieveVMwareCertificate.py
  -host:<vmware_virtual_center_host_name> -port:
<vmware_virtual_center_ssl_port_number>
  -user:<vmware_user_id>
  -password:<vmware_password>
```

Where *<vmware_virtual_center_host_name>* is the host name of the vCenter server, *<vmware_virtual_center_ssl_port_number>* is the secure SSL port of the vCenter server, *<vmware_user_id>* is the VMware userid that is used to access the vCenter server, and *<vmware_password>* is the password associated with the *<vmware_user_id>*.

- **If you are configuring WebSphere Virtual Enterprise to communicate with ESX servers:**

1. Retrieve and store a signer certificate from the ESX server and configure WebSphere Virtual Enterprise to communicate with the ESX server.

```
./wsadmin.sh -lang jython -f retrieveVMwareCertificate.py
  -host:<vmware_esx_server_host_name> -port:
```



```
<vmware_esx_server_ssl_port_number>  
-user:<vmware_user_id>  
-password:<vmware_password>
```

Where `<vmware_esx_server_host_name>` the host name of the ESX server, `<vmware_esx_server_ssl_port_number>` is the secure SSL port of the ESX server, `<vmware_user_id>` is the VMware userid that is used to access the ESX server, and `<vmware_password>` is the password associated with the `<vmware_user_id>` value. You should use host names instead of an IP address for the `-host` parameter.

2. Repeat the previous step for all of your ESX servers by using the script to retrieve and store a signer certificate for each ESX server.

Results

By configuring WebSphere Virtual Enterprise to work with vCenter or ESX, better service differentiation management results than using vCenter or ESX alone. With WebSphere Virtual Enterprise, you can add application-level goals and characteristics so that the autonomic managers can perform the necessary flow control in your virtualized environment.

What to do next

If timeout errors are occurring, you can increase the `com.ibm.websphere.webservices.http.connectionTimeout` and `com.ibm.websphere.webservices.http.SocketTimeout` custom property values from the default of 300 seconds to 600 seconds. Consider making this change when you have a virtualized environment with a large number of physical and virtual machines. For example, if your environment has 400 physical machines, when requests are sent from WebSphere Virtual Enterprise to the hypervisor for configuration information, the hypervisor contacts each of the 400 physical machines. If each request takes 1 second to complete, the default timeout of 300 seconds is not long enough to process all of the requests, and a read timeout results. See [HTTP transport custom properties for Web services applications](#) for more information about the custom properties.

Configure middleware servers on your WebSphere nodes.

[VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

WebSphere Virtual Enterprise uses the VMware Infrastructure SDK (VI SDK) to communicate through Web services with VMware Infrastructure 3 platforms. Any VMware Infrastructure 3 platforms that expose the VMware Infrastructure SDK (VI SDK) as a Web service and can work with WebSphere Virtual Enterprise, such as ESX or vCenter. VMware software provides operating system level virtualization, while WebSphere Virtual Enterprise provides virtualization at an application level.

[Advanced configuration for VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

The `retrieveVMwareCertificate.py` script can complete all of the steps that are needed to configure VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise. However, you can also complete these steps manually by creating the signer certificate and required custom properties in the administrative console.

Next topic: [Adding middleware servers to configurations](#)

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Related tasks:

↳ [Retrieving signers from a remote SSL port](#)

Related reference:

↳ [VMware Infrastructure SDK](#)

↳ [HTTP transport custom properties for Web services applications](#)

Related information:

[VMware custom properties](#)

VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise

WebSphere® Virtual Enterprise uses the VMware Infrastructure SDK (VI SDK) to communicate through Web services with VMware Infrastructure 3 platforms. Any VMware Infrastructure 3 platforms that expose the VMware Infrastructure SDK (VI SDK) as a Web service and can work with WebSphere Virtual Enterprise, such as ESX or vCenter. VMware software provides operating system level virtualization, while WebSphere Virtual Enterprise provides virtualization at an application level.

Supported VMware releases

The supported versions include:

- VMware VirtualCenter Version 2.5
- VMware ESX Version 3.5
- VMware vSphere Version 4.0 and Version 4.1, both of which include VMware ESXi and VMware vCenter Server

The documentation generically refers to these servers with the following terminology:

- **ESX server:** Refers to VMware ESX Version 3.5 or VMware ESXi servers in VMware vSphere Version 4.0 and Version 4.1.
- **vCenter server:** Refers to VMware VirtualCenter Version 2.5 or VMware vCenter servers in VMware vSphere Version 4.0 and Version 4.1.

WebSphere Virtual Enterprise and virtualization technologies

When you use virtualization technologies with WebSphere Virtual Enterprise, a node is no longer a representation of a physical computer. Because several virtual machines can be created on a single physical computer, you can create multiple WebSphere Virtual Enterprise nodes on the same physical computer.

WebSphere Virtual Enterprise communicates with VMware Infrastructure 3 platforms that expose the VMware Infrastructure SDK (VI SDK) as a Web service. With this integration, the runtime environments can communicate with each other through Web services. By using the SDK, WebSphere Virtual Enterprise can make decisions about the placement of servers with the knowledge about where nodes are running in the cell. This integration is only supported if you are using VMware Infrastructure 3 platforms.

VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise function

WebSphere Virtual Enterprise is aware that nodes are not necessarily distinct, standalone resources. In some cases, a node might be sharing memory and processor space with other nodes, with parameters specified that control and manage the systems.

When you have multiple nodes running on a physical computer with VMware Infrastructure 3 platforms, WebSphere Virtual Enterprise can communicate with the VMware product through Web services. You can configure this communication in the administrative console by creating cell-wide custom properties. These custom properties define the URL, user ID, and password for a vCenter or ESX server. You configure the custom properties with information about all of the virtual machines and physical servers in your environment. For example, if you are running vCenter, you can create a custom property for the vCenter server.

Virtual machines and nodes are correlated by Media Access Control (MAC) addresses.

By configuring WebSphere Virtual Enterprise to know about the vCenter or ESX servers, the application placement controller can use the VMware settings to properly set the maximum processor utilization limits for each node. WebSphere Virtual Enterprise sets the processor utilization limit for a given virtual machine to be the minimum of either the processor utilization limit that is configured in vCenter or ESX, or the number of processors that are assigned to the virtual machine divided by the actual number of processors on the physical workstation. For example, if the physical computer has 3 processors, and the virtual machine is assigned 1 processor, then the processor utilization limit for the virtual machine is 33% of the total processor on the physical workstation.

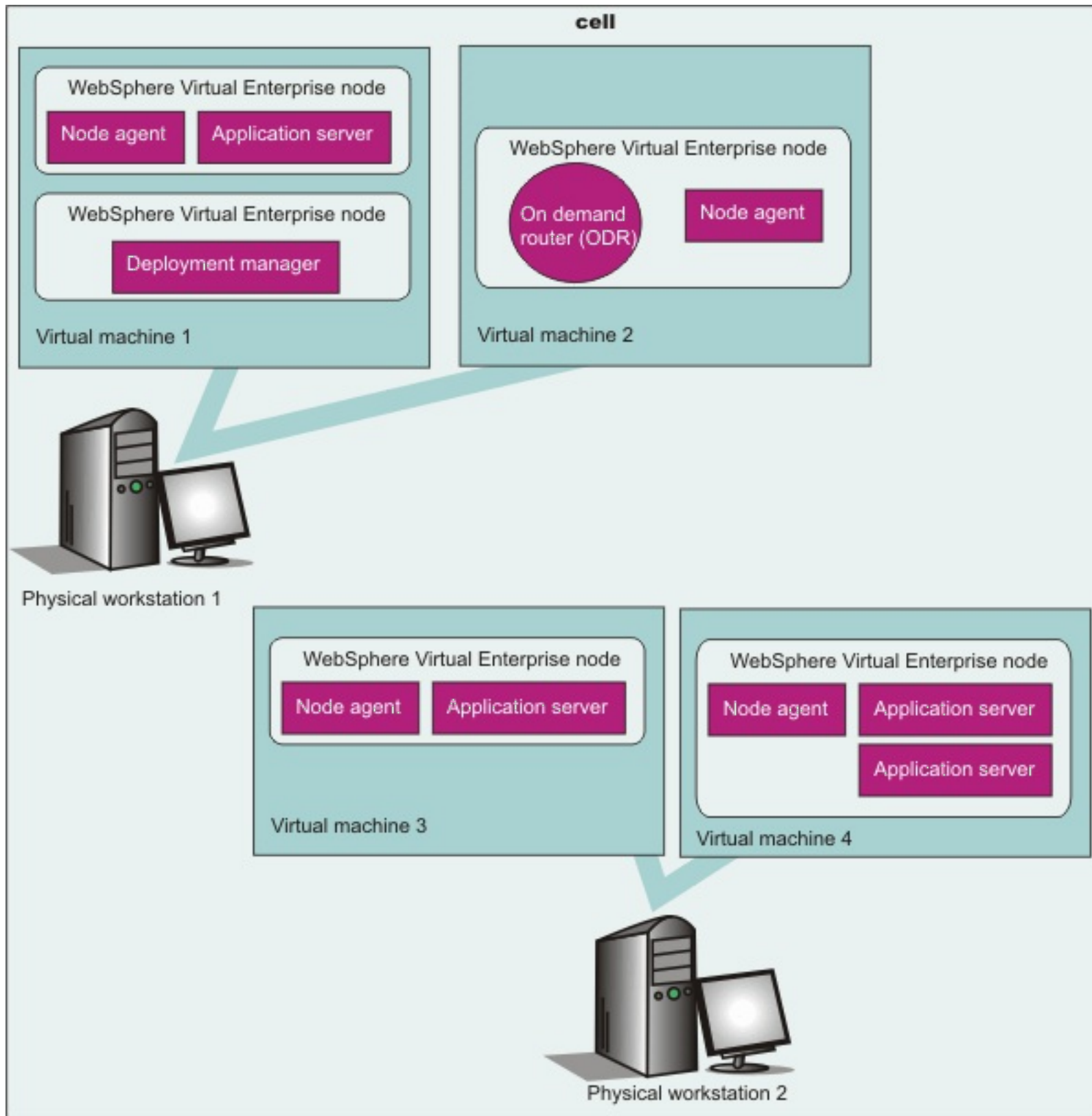
Benefit of using WebSphere Virtual Enterprise in addition to vCenter or ESX

Although VMware Infrastructure 3 platforms provide service differentiation and management, you cannot control performance at an application level. WebSphere Virtual Enterprise adds the capability to define application level goals. The autonomic managers work to maintain the defined goals.

Sample topology

The following sample topology shows how multiple virtual machines can be created on a single physical workstation. In this topology, the deployment manager and on demand router (ODR) are on the same physical workstation, but two different virtual machines.

Figure 1. Virtual machines integrated into a WebSphere Virtual Enterprise environment



Parent topic: [Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

Related tasks:

[Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

[Advanced configuration for VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

Related reference:

[VMware Infrastructure SDK](#)

Related information:

[VMware custom properties](#)

Advanced configuration for VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise

The retrieveVMwareCertificate.py script can complete all of the steps that are needed to configure VMware Infrastructure 3 platforms and WebSphere® Virtual Enterprise. However, you can also complete these steps manually by creating the signer certificate and required custom properties in the administrative console.

Before you begin

- Configure the VMware Infrastructure 3 platforms environment on your physical servers. Your VMware Infrastructure 3 platforms environment must meet the following requirements:
 - Your VMware Infrastructure 3 platforms environment must be on servers that are running Solaris Operating Environment on Intel hardware, Windows, or Linux x86 operating systems.
 - You must use VMware products that support VMware Infrastructure 3 platforms. The supported versions are:

- VMware VirtualCenter Version 2.5
 - VMware ESX Version 3.5
 - VMware vSphere Version 4.0, which includes VMware ESXi and VMware vCenter Server
- The documentation generically refers to these servers with the following terminology:

- **ESX server:** Refers to VMware ESX Version 3.5 or a VMware ESXi server in VMware vSphere Version 4.0.
 - **vCenter server:** Refers to VMware VirtualCenter Version 2.5 or a VMware vCenter server in VMware vSphere Version 4.0.
- Install and configure WebSphere Virtual Enterprise on each virtual machine.

About this task

You can retrieve a signer certificate with a script or in the administrative console, and then define the required custom properties in the administrative console. You can also complete these steps with the script only. See [Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#) for more information.

Procedure

- **If you are configuring WebSphere Virtual Enterprise to communicate with a vCenter server:**

1. Retrieve a signer from the vCenter server and store the signers in the CellDefaultTrustStore key store. To retrieve the signer, you can either use the administrative console or run the retrieveVMwareCertificate.py script.

To retrieve the signer certificate by running the script:

```
./wsadmin.sh -lang jython -f retrieveVMwareCertificate.py  
-host:<vmware_virtual_center_host_name> -port:  
<vmware_virtual_center_ssl_port_number>
```

Where *<vmware_virtual_center_host_name>* is the host name of the vCenter and *<vmware_virtual_center_ssl_port_number>* is the secure SSL port of the vCenter.

To retrieve the signer certificate using the administrative console:

- a. Navigate to the signer certificates administrative console panel. In the administrative console, click **Security > SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore > Signer certificates > Retrieve from port.**
 - b. Enter the host and port information for the vCenter server and an alias or name for the certificate. The alias should follow the syntax: *<vmware_virtual_center_short_host>-vmware*. For example, if the hostname of the vCenter server is *myvmwarevc.foo.net*, the alias name would be *myvmwarevc-vmware*. For Hypertext Transfer Protocol Secure (HTTPS), the default port value is 443.
 - c. Click **Retrieve signer information.**
 - d. Click **Apply.** This action indicates that you accept the credentials of the signer.
- The signer certificate that is retrieved from the vCenter server is stored in the CellDefaultTrustStore keystore.
2. Configure custom properties for the vCenter server so that WebSphere Virtual Enterprise can use Web services to communicate with the VMware Infrastructure SDK (VI SDK). In the administrative console, click **Cells > Custom Properties > New.** Create the following cell-wide custom properties:

- *vmware.service.unique_id.url*
- *vmware.service.unique_id.userid*
- *vmware.service.unique_id.password*

Note: For the *vmware.service.unique_id.userid* custom property, the following privileges are required by Virtual Enterprise to read certain properties and to perform various operations:

- System.Anonymous
- System.Read
- System.View
- Sessions.TerminateSession

The *unique_id* value is a unique identifier that represents the vCenter. For example, if the host name of the vCenter server is myvmwarevc.foo.net and the port is 443, the *unique_id* value would be myvmwarevc_foo_net_443. Following the same example, the names of the custom properties would be:

```
vmware.service.myvmwarevc_foo_net_443.url
vmware.service.myvmwarevc_foo_net_443.userid
vmware.service.myvmwarevc_foo_net_443.password
```

• **If you are configuring WebSphere Virtual Enterprise to communicate with ESX servers:**

1. Retrieve a signer from the ESX server and store the signers in the CellDefaultTrustStore key store. To retrieve the signer, you can either use the administrative console or run the retrieveVMwareCertificate.py script.

To retrieve the signer certificate by running the script:

```
./wsadmin.sh -lang jython -f retrieveVMwareCertificate.py
  -host:<vmware_esx_server_host_name> -port:
<vmware_esx_server_ssl_port_number>
```

Where *<vmware_esx_server_host_name>* is the host name of the ESX server and *<vmware_esx_server_ssl_port_number>* is the secure SSL port of the ESX server.

To retrieve the signer certificate using the administrative console:

- a. Navigate to the signer certificates administrative console panel. In the administrative console, click **Security > SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore > Signer certificates > Retrieve from port.**
 - b. Enter the host and port information for the ESX server and an alias name for the certificate. The alias should follow the syntax: *<vmware_esx_server_short_host>-vmware*. For example, if the hostname of the ESX server is myvmwareesx.foo.net, the alias name would be myvmwareesx-vmware. For Hypertext Transfer Protocol Secure (HTTPS), the default port value is 443.
 - c. Click **Retrieve signer information.**
 - d. Click **Apply.** This action indicates that you accept the credentials of the signer.
- The signer certificate that is retrieved from the ESX server is stored in the CellDefaultTrustStore keystore.
2. Configure custom properties for the ESX servers so that WebSphere Virtual Enterprise can use Web services to communicate with the VMware Infrastructure SDK (VI SDK). In the administrative console, click **Cells > Custom Properties > New.** Create the following cell-wide custom properties:
 - vmware.service.unique_id.url
 - vmware.service.unique_id.userid
 - vmware.service.unique_id.password

The *unique_id* value is a unique identifier that represents the ESX server. For example, if the host name of the ESX server is myvmwareesx.foo.net and the port is 443, the *unique_id* value would be myvmwareesx_foo_net_443. Following the same example, the names of the custom properties would be:

```
vmware.service.myvmwareesx_foo_net_443.url
vmware.service.myvmwareesx_foo_net_443.userid
vmware.service.myvmwareesx_foo_net_443.password
```

Repeat these steps for each ESX server in your configuration.

[Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

To configure WebSphere Virtual Enterprise to work with VMware Infrastructure 3 platforms, you must configure security so that the servers can communicate with each other and configure custom properties on your deployment manager to define the vCenter or ESX servers.

[VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

WebSphere Virtual Enterprise uses the VMware Infrastructure SDK (VI SDK) to communicate through Web services with VMware Infrastructure 3 platforms. Any VMware Infrastructure 3 platforms that expose the VMware Infrastructure SDK (VI SDK) as a Web service and can work with WebSphere Virtual Enterprise, such as ESX or vCenter. VMware software provides operating system level virtualization, while WebSphere Virtual Enterprise provides virtualization at an application level.

Next topic: [Adding middleware servers to configurations](#)

Parent topic: [Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

Related tasks:

☞ [Retrieving signers from a remote SSL port](#)

Related reference:

[VMware Infrastructure SDK](#)

[HTTP transport custom properties for Web services applications](#)

Related information:

[VMware custom properties](#)

Federating middleware nodes

You can use the command line to federate middleware nodes to a WebSphere® Virtual Enterprise cell.

Before you begin

- Augment a WebSphere Application Server deployment manager profile to use WebSphere Virtual Enterprise. Install the middleware agent on the node.
- Each of your middleware nodes must have the same network view for each endpoint and resource that is related to WebSphere Application Server. All host entries or host alias entries that are related to WebSphere Application Server within a hosts file must be included in the hosts file on each of the other middleware nodes in the configuration, including hosts files on nodes that run the middleware agent. For example, if you define a product related host alias called agentA in the hosts file for the nodeA node, then the hosts file on each middleware node must contain an entry for the agentA host alias.
- **z/OS Restriction:** You cannot federate a middleware node into a deployment manager that is running on the z/OS® operating system.

About this task

Federate a middleware node if it was not federated into the cell during the middleware agent installation. You can perform this task by using a command line or the administrative console.

Procedure

- To federate the node from the command line:
 1. Federate the node by running the addAgent.sh or addAgent.bat command from the middleware node computer. The command has the following format:

```
agent_install_root/bin/addAgent.sh(bat) -host deployment_manager_host -port
deployment_manager_console_port -user administrative_id
-password administrative_password
```

For example: The following example federates the node from the xdblade03b computer:

```
addAgent.sh -host xdblade03b -port 9060 -user admin -password admin
```

2. Start the agent. Run the following command from the middleware node:

```
agent_install_root/bin/StartAgent.sh(bat)
```

- To start the agent from the administrative console:
 1. In the administrative console, select **System administration > Middleware nodes**.
 2. From the **Select operational action** menu, select **Start agent**.
 3. Click **Run**.

Results

The middleware node is displayed as an unmanaged node in the product cell. If you encounter problems in federating and adding nodes, check the following log files:

- `install_root/logs/addAgent.log` file for addNode command errors
- `install_root/logs/middlewareagent/SystemOut.log` file
- `install_root/logs/middlewareagent/SystemErr.log` file

What to do next

Configure middleware servers and dynamic clusters.

Parent topic: [Adding middleware servers to configurations](#)

Previous topic: [Using the graphical user interface to augment profiles](#)

Next topic: [Creating dynamic clusters](#)

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Related concepts:

[Middleware agent](#)

[Middleware nodes and servers](#)

Related tasks:

[Installing the middleware agent](#)

Creating and configuring ODRs

The on demand router (ODR) is an intelligent HTTP and Session Initiation Protocol (SIP) proxy server in WebSphere® Virtual Enterprise. The ODR is the point of entry into a WebSphere Virtual Enterprise environment and is a gateway through which HTTP requests and Session Initiation Protocol (SIP) messages flow to back-end application servers. You can configure the ODR to determine how it handles failure scenarios and how it tunes certain work requests.

Before you begin

z/OS SIP is not supported on the z/OS® operating system.

About this task

The ODR can momentarily queue requests for less important applications in order to allow requests from more important applications to be handled more quickly or to protect back-end application servers from being overloaded. The ODR is aware of the current location of a dynamic cluster instance, so that requests can be routed to the correct endpoint. The ODR can also dynamically adjust the amount of traffic sent to each individual server instance based on process utilization and response times. By default, the ODR binds to ports 80 and 443 for listening on HTTP and HTTPS, which requires running the ODR as a root user. If you want to run the ODR as a non-root user, you must change the PROXY listening ports to values greater than 1024.

The ODR is fully aware of the dynamic state of the cell, so that if one server in the cell fails, the requests are routed to another server. When the ODR is notified that the application has initialized on the restarted server, the ODR will route requests to that server again.

The ODR will not route any requests to the application on the application server until the application is through starting/initializing. If the application is started on other application server(s), then the requests will be routed to them. If the application is not started on any other servers, then the ODR will still not route to the starting-in-progress application server. Instead, a 503 is returned.

Procedure

- [Creating ODRs](#)

An ODR is a proxy server with advanced capabilities that are used to route work to server nodes. Note that the configuration of the ODR in the DMZ is not supported. To configure ODRs to perform a SSL offload, see [Configuring SSL offload for all HTTPS traffic](#). For information on other custom properties, see [On demand router system and custom properties](#).

- Follow the WebSphere Application Server Network Deployment instructions in the [Proxy server settings](#) topic for most of the instructions to configure ODRs. See [Configuring ODRs](#) for WebSphere Virtual Enterprise specific fields.

Note: In the WebSphere Virtual Enterprise administrative console, use the following path to define the configuration of the ODR: **Servers > On demand routers > odr_name > On Demand Router Settings > On Demand Router Properties**.

- [Configuring SSL offload for all HTTPS traffic](#)

By default, the ODR matches the incoming protocol to the outgoing protocol. For inbound HTTP requests, the request is forwarded over outbound HTTP. For inbound HTTPS, the request is forwarded over outbound HTTPS. This default behavior can either be changed for all HTTP and HTTPS traffic that is handled by the ODR, or on a per-Web module basis.

- [On demand router system and custom properties](#)

You can use ODR custom properties to change the behavior of your ODR. For example, you can change the error code that the ODR returns when messages are rejected because of processor or memory overload.

- [Configuring a Web server as a trusted proxy server](#)

A Web server should be configured as a trusted secure proxy because a trusted security proxy is allowed to pass information such as the virtual host name, or user identity to the ODR in private HTTP headers.

- Optional: [Defining routing policies for generic server clusters](#)

- [Defining a service policy](#)

Routing and service policies for SIP are defined at the ODRs.

What to do next

Configure the middleware servers and dynamic clusters for your environment.

[Creating ODRs](#)

You can create on demand routers (ODRs) to route requests to WebSphere Virtual Enterprise nodes. The ODR is fully aware of the dynamic state of the cell, so that if one server in the cell fails, the requests are routed to another server. The configuration of the ODR in the DMZ is not supported.

Configuring ODRs

You can configure an on demand router (ODR) to determine how it handles failure scenarios and tunes certain work requests. You can configure the connections and requests to the application server, configure the requests that must be rejected, define how error responses are handled, and specify the location of the proxy logs. The configuration of the ODR in the DMZ is not supported.

Configuring a Web server as a trusted proxy server

If your environment consists of a topology that includes a Web server, you must configure it as a trusted secure proxy server. By configuring the secure proxy server, you can inform the on demand router (ODR) that the Web server is a trusted secure proxy so that the ODR can receive requests.

Configuring SSL offload for all HTTPS traffic

You can change the default outgoing protocol for requests that go through the on demand router. By default, the on demand router matches the incoming protocol to the outgoing protocol.

Defining routing policies for generic server clusters

You can define routing policies for generic server clusters. A routing policy is a set of rules that determine how the server routes incoming requests.

Defining service policies for generic server clusters

Optionally, you can define service policies for generic server clusters.

Creating ODRs

You can create on demand routers (ODRs) to route requests to WebSphere Virtual Enterprise nodes. The ODR is fully aware of the dynamic state of the cell, so that if one server in the cell fails, the requests are routed to another server. The configuration of the ODR in the DMZ is not supported.

Configuring ODRs

You can configure an on demand router (ODR) to determine how it handles failure scenarios and tunes certain work requests. You can configure the connections and requests to the application server, configure the requests that must be rejected, define how error responses are handled, and specify the location of the proxy logs. The configuration of the ODR in the DMZ is not supported.

Configuring a Web server as a trusted proxy server

If your environment consists of a topology that includes a Web server, you must configure it as a trusted secure proxy server. By configuring the secure proxy server, you can inform the on demand router (ODR) that the Web server is a trusted secure proxy so that the ODR can receive requests.

Enabling cell affinity

The cell affinity function preserves sessions even in the event of an on demand router (ODR) outage when you configure unbridged, on demand router (ODR) topologies.

Using the binary trace facility

WebSphere Virtual Enterprise provides a binary trace facility (BTF) that, by default, gathers trace information in a production environment that is useful for customers and development teams, because the level of trace gathered facilitates troubleshooting problems without having to reproduce the problem. Further, optimal performance in your environment is not compromised.

Tuning the on demand router (ODR)

ODRs are servers that act as an intermediary for HTTP requests that are serviced by application servers or Web servers. There are many factors that affect ODR performance. If you discover that your ODRs are not working the way you want, then you might need to tune your ODRs.

Configuring custom logs

You can configure custom logs through the wsadmin tool with the ruleset administrative tasks. You can configure custom logs through the rule set configuration objects. Validation is limited to type checking. More thorough validation is performed during an on demand router (ODR) startup by issuing runtime exceptions to SystemOut.

Configuring SSL offload for all HTTPS traffic

You can change the default outgoing protocol for requests that go through the on demand router. By default, the on demand router matches the incoming protocol to the outgoing protocol.

Configuring SSL offload for partial HTTPS traffic

Configure the ODR to perform SSL offload for partial HTTPS traffic, or on a per-web module basis.

Integrating the SIP on demand router with Load Balancer

You can integrate the Session Initiation Protocol (SIP) on demand router (ODR) with Load Balancer. Load Balancer for IBM® WebSphere Application Server can help maximize the potential of your Web site by providing a powerful, flexible, and scalable solution to peak-demand problems.

Defining routing policies for generic server clusters

You can define routing policies for generic server clusters. A routing policy is a set of rules that determine how the server routes incoming requests.

Defining service policies for generic server clusters

Optionally, you can define service policies for generic server clusters.

Configuring communication between core groups that are in the same cell

A cell is comprised of one or more processes that serve as hosts for the runtime components. Each cell

has one or more named core groups. A core group is a group of processes that are directly accessible to each other and is connected using a local area network (LAN). If core groups that are in the same cell need to share workload management (WLM) information, then you must configure these core groups to communicate.

BBSON bulletin board

WebSphere Virtual Enterprise offers a bulletin board service over a network (BBSON) that is independent of the WebSphere Application Server Network Deployment high availability manager. The high availability manager provides a mechanism that allows servers to easily exchange state data. This mechanism is commonly referred to as the bulletin board.

Configuring WebSphere Virtual Enterprise for cross-cell communication

You can enable cross-cell communication when a WebSphere Virtual Enterprise administrative cell contains servers that are enabled with an on demand router (ODR) that routes to other WebSphere Application Server administrative cells.

Configuring an ODR to dynamically update the web server plug-in configuration

In a topology with a web server which forwards requests to an on demand router (ODR), the ODR can be set to dynamically generate and update the web server plug-in configuration file `plugin-cfg.xml`. By configuring an ODR to dynamically update `plugin-cfg.xml`, you avoid manually updating the file during administrative actions, which can affect the content.

Generating the plug-in configuration in a high availability environment

In a high availability environment, you can configure the web server to generate the plug-in configuration. By doing so, you ensure that the configuration is generated by one process in the cell, regardless of other processes that might not be available.

Segregating HTTP traffic by ODR clusters

WebSphere Virtual Enterprise provides a service to automatically regenerate and propagate the Web server plug-in configuration file each time a change is made, which should affect how the plug-in routes. When you use this service, the Web server plug-in routes traffic by application, deployment target, or node group to a specific ODR cluster.

Controlling the generation of the plugin-cfg.xml file

You can override the default attribute values of the `plugin-cfg.xml` file that is automatically generated by the on demand router (ODR).

Routing requests directly from a web server to a back-end application server

Define a new custom property to configure the web server plug-in configuration to route requests directly from the web server to a back-end application server. You can configure certain requests to route through an on demand router (ODR) and configure other requests to route directly to a back-end application server.

Configuring passive HTTP session affinity in the on demand router

When the on demand router (ODR) processes a request, it obtains the session affinity descriptor policy of the cluster to which the server belongs. If you changed the default settings for any middleware servers, you might need to update the middleware descriptor properties so that the ODR can obtain the descriptor policy. In most cases, the on demand router (ODR) does not require configuration to support HTTP session affinity. However, some special cases exist where you must configure the ODR to learn about backup servers that the back end servers might be setting on the session affinity cookie.

Configuring the on demand router for multi-cluster failover and load balancing routing

To configure the on demand router (ODR) to route requests to a different cluster, you can configure custom properties for configuring multi-cluster failover and load balancing routing policies. You might route requests to a cluster in another cell if your primary cluster fails, to balance the load in your environment between multiple clusters, or to direct the ODR to route requests to a specific cluster.

Configuring ODR OutOfMemory prevention

You can create cell-level custom properties to prevent the on demand router (ODR) from running out of memory when the number of concurrent requests sent to the ODR increases.

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Next topic: [Adding middleware servers to configurations](#)

Next topic: [Adding middleware servers to configurations](#)

Next topic: [Creating dynamic clusters](#)

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Related concepts:

[Overview of request flow prioritization](#)

Related reference:

[createodr.jacl script](#)

Creating ODRs

You can create on demand routers (ODRs) to route requests to WebSphere® Virtual Enterprise nodes. The ODR is fully aware of the dynamic state of the cell, so that if one server in the cell fails, the requests are routed to another server. The configuration of the ODR in the DMZ is not supported.

Before you begin

- The ODR is a server that acts as an intermediary for HTTP requests that are serviced by application servers or Web servers. Make sure you have an application or Web server installed. By default, the ODR binds to ports 80 and 443 for listening on HTTP and HTTPS, which requires running the ODR as a root user. If you want to run the ODR as a non-root user, you must change the PROXY listening ports to values greater than 1024.
- **z/OS** SIP is not supported on the z/OS® operating system.

About this task

A deployment manager profile cannot be used as the target profile for an ODR. Only an application server profile can be used as the target node when creating an ODR. To collocate an ODR with a deployment manager, a separate WebSphere Virtual Enterprise-enabled application server profile must be created and federated on the same machine as the WebSphere Virtual Enterprise-enabled deployment manager profile. You can then create an ODR within the application server profile.

Procedure

1. Navigate to the ODR creation wizard in the administrative console. In the administrative console, select **Servers > On demand routers**. Click **New**.
2. Select the node on which you want the ODR to be created. The selected node is pre-populated with available nodes in the cell. If your environment is a heterogeneous mix of WebSphere Virtual Enterprise and nodes that are running other middleware software, select a product node. If you select a node that is not running the product, your ODR does not start.

z/OS Tip: Select a node to run the ODR that does not run any dynamic clusters, unless you are using the z/OS operating system. In a z/OS environment, you can co-locate the ODR on a node that hosts application servers.

3. Determine whether to generate HTTP ports, SIP ports, or both. The Generate unique ports option is selected by default and is the recommended option. If you deselect this option, port conflicts might occur.
4. Select a server template to base your new ODR. You can use an application server template on which to model the servers for the new ODR. You can use a default template or map an existing application server. Mapping preexisting ODRs can save time. You can build one ODR, apply all of the configurations that your environment needs, and use that ODR as a template.

If you are running WebSphere Virtual Enterprise for z/OS, ensure that the SIP option is not selected.

Results

The ODR that you created automatically routes HTTP requests to product cells.

What to do next

To enable routing to another product cell, configure your cell to communicate with other product cells.

You might want to configure the ODR to route work to nodes that are not running WebSphere Virtual Enterprise nodes. After you create the ODR and apply any optional configuration parameters, you can define the ability to route work to nodes that are not running WebSphere Virtual Enterprise. Note that the configuration of the ODR in the DMZ is not supported.

Parent topic: [Creating and configuring ODRs](#)

Next topic: [Configuring WebSphere Virtual Enterprise for cross-cell communication](#)

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring ODRs](#)

[Configuring the on demand router for multi-cluster failover and load balancing routing](#)

Related reference:

[createodr.jacl script](#)

[Administrative roles and privileges](#)

Related information:

[Proxy server settings](#)

Configuring ODRs

You can configure an on demand router (ODR) to determine how it handles failure scenarios and tunes certain work requests. You can configure the connections and requests to the application server, configure the requests that must be rejected, define how error responses are handled, and specify the location of the proxy logs. The configuration of the ODR in the DMZ is not supported.

Before you begin

You must first create an ODR, a proxy with advanced capabilities that WebSphere® Virtual Enterprise uses to route work to application server nodes. See [Creating ODRs](#) for more details.

About this task

You can define the configuration of the ODR by editing its proxy configuration. Define the configuration further by clicking **Servers > On Demand Routers > odr_name > On demand router properties > On demand router settings**. Note that the configuration of the ODR in the DMZ is not supported. To configure ODRs to perform a SSL offload, see [Configuring SSL offload for all HTTPS traffic](#). For information on other custom properties, see [On demand router system and custom properties](#).

Procedure

1. Follow the WebSphere Network Deployment instructions in the [Proxy server settings](#) topic for instructions to configure ODRs, and the [Overview of custom error page policy](#) topic for additional information. The WebSphere Virtual Enterprise-specific field for Compression Policy information follows.
2. Compression Policy. Check **Enable** to enables the compression of the HTTP response message body before sending it to the client. Choices include:

Option	Description
gzip	Compress the response using the gzip compression mechanism. The response is compressed only if it is acceptable to the client, based on the Accept-Encoding request header.
deflate	Compress the response using the deflate compression mechanism. The response is compressed only if it is acceptable to the client, based on the Accept-Encoding request header.
auto	Use either gzip, deflate, or no compression as determined by the client preference.

What to do next

After you create and configure the ODR and apply any configuration parameters, you can define the ability to route work to nodes that are not a part of your WebSphere Virtual Enterprise environment.

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring SSL offload for all HTTPS traffic](#)

[Configuring the on demand router for multi-cluster failover and load balancing routing](#)

Related reference:

[createodr.jacl script](#)

[Administrative roles and privileges](#)

Related information:

[On demand router system and custom properties](#)

[Proxy server settings](#)

Configuring a Web server as a trusted proxy server

If your environment consists of a topology that includes a Web server, you must configure it as a trusted secure proxy server. By configuring the secure proxy server, you can inform the on demand router (ODR) that the Web server is a trusted secure proxy so that the ODR can receive requests.

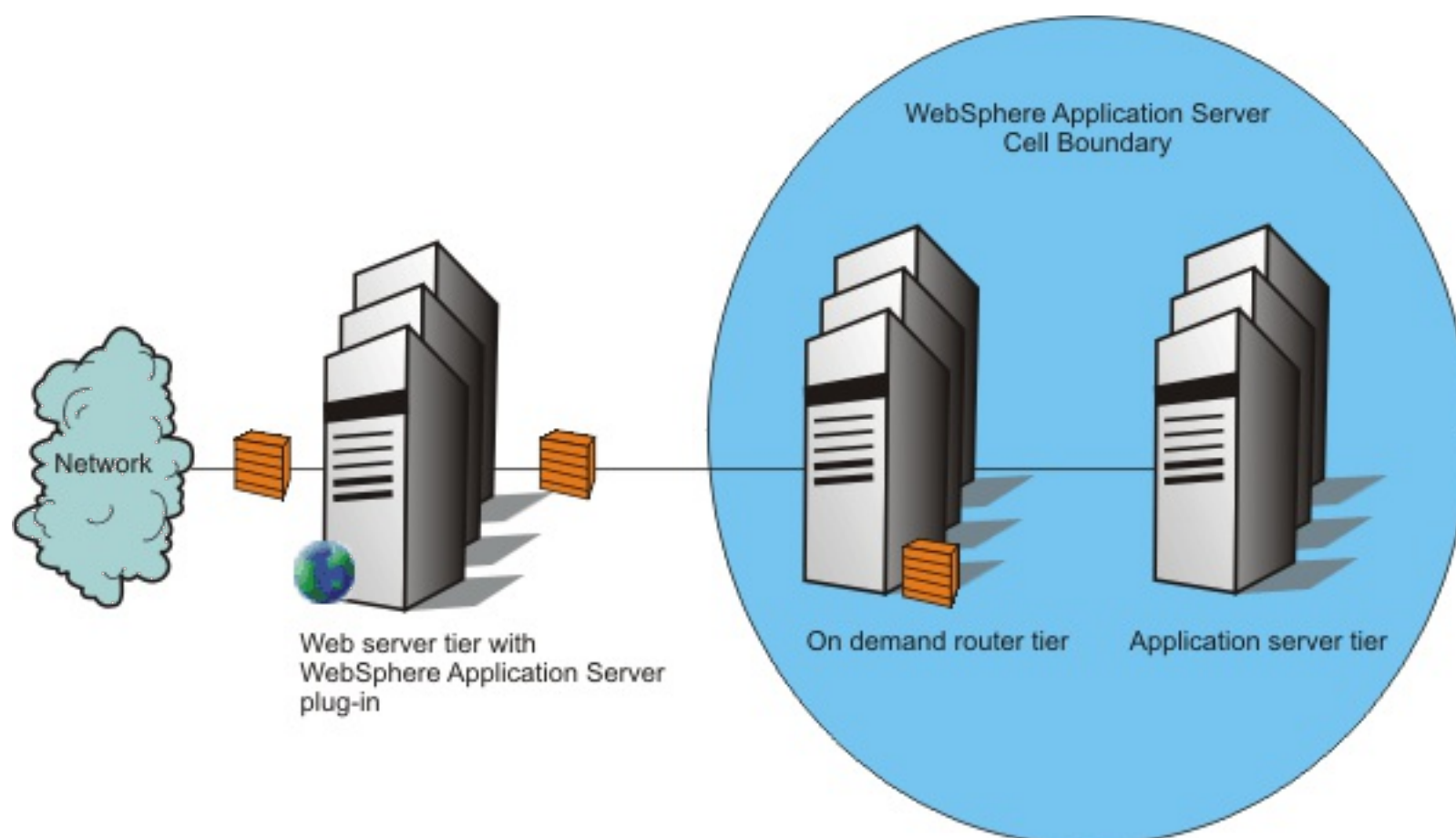
Before you begin

You must first create an ODR, a proxy with advanced capabilities that WebSphere® Virtual Enterprise uses to route work to application server nodes. See [Creating ODRs](#) for more details.

About this task

A Web server should be configured as a trusted secure proxy because a trusted security proxy is allowed to pass information such as the virtual host name, or user identity to the ODR in private HTTP headers. Web servers read incoming requests to verify which ODR they are routed to. Private headers received from an untrusted proxy are discarded by the ODR. This configuration field enables intermediaries other than the ODR server to handle the request by explicitly telling the ODR that it is to trust them. A trusted security proxy receives requests prior to the ODR and then forwards requests to the ODR. For example, when the Web server with the WebSphere Application Web server plugin forwards requests to the ODR, the Web server must be configured as a trusted security proxy.

Figure 1. Example topology of a simple WebSphere Virtual Enterprise environment supported configuration including a Web server



Procedure

1. To configure a Web server as a trusted proxy server, in the administrative console, click **Servers > On Demand Routers > `on_demand_router_name` > On Demand Router Properties > On Demand Router settings**.
2. Specify the name of the Web server in the **Trusted Security Proxies**. This configuration field enables intermediaries other than the ODR server to handle the request by explicitly telling the ODR that it can trust the Web server you specify. Use an internet protocol or fully-qualified host name in this field. For example, `myhost.com` or an IP address such as `10.1.1.1`.
3. Click **Apply**.
4. Click **Save**.

Results

Your Web server is now configured as a trusted proxy server.

Parent topic: [Creating and configuring ODRs](#)

Previous topic: [Creating ODRs](#)

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring ODRs](#)

[Preparing the hosting environment for dynamic operations](#)

Related reference:

[createodr.jacl script](#)

Enabling cell affinity

The cell affinity function preserves sessions even in the event of an on demand router (ODR) outage when you configure unbridged, on demand router (ODR) topologies.

Before you begin

You can use this feature if you have installed Version 6.0.2.6, or Version 6.1.0.5 or later.

About this task

The cell affinity function prevents the loss of sessions when there are multiple ODRs within multiple, unbridged cells, and the IBM® HTTP Server is configured to forward session traffic, either through load-balancing or failover, to more than one ODR.

Procedure

1. In the administrative console, select **System Administration > Cell > Custom Properties**.
2. Set the custom cell property odrSessionAffinityEnabled to true. See [Custom properties](#) for more information.
3. Configure generic server clusters for other ODR cells. See [Defining generic server clusters for remote ODR cells](#) for more information.
4. If you are configuring a multi-tiered configuration, the ODRs in one tier must be configured and operate independently from the ODRs in the other tiers. See [Configuring cell affinity in a multi-tiered environment](#) for more information. Note that you may also optionally perform this step in a single-tiered environment if you simply want to use a cookie name other than ODRSESSIONID.
5. Generate a plugin-cfg.xml. See [Generating a plugin-cfg.xml](#) for more information.
6. Merge the plugin-cfg.xml files into one plugin-cfg.xml file. See [pluginmerge.bat|.sh script](#) for more information.
7. Install the new plugin-cfg.xml file on the IBM HTTP server. Refer to [Configuring an ODR to dynamically update the web server plug-in configuration](#) for more information on how to install a new plugin-cfg.xml file.

[Cell affinity function](#)

Using the cell affinity function, you can configure unbridged, on demand router (ODR) topologies to preserve sessions even in the event of on demand router (ODR) outages. With this function, you can configure your topology in such a way that, when an ODR receives misrouted, in-session traffic, the ODR reroutes the traffic back to a working ODR in the original cell. Thus, you can configure an IBM HTTP Server to route to ODRs in multiple cells and still preserve session affinity.

[Defining generic server clusters for remote ODR cells](#)

In order for the on demand router (ODR) to route to other cells, generic server clusters must be defined to represent those cells. The purpose of defining generic server clusters is to allow each ODR to recognize the other ODRs that are in remote cells.

[Configuring cell affinity in a multi-tiered environment](#)

You can configure your network to contain multiple tiers of on demand routers (ODRs). In multi-tiered configurations, you must configure the ODRs in one tier independently from the ODRs in the other tiers, because the ODRSESSIONIDs of two different tiers must not be the same.

[Configuring the multi-cluster routing policy](#)

In order to configure the on demand router (ODR) to route requests to a generic server cluster, you must configure a multi-cluster routing policy (MCRP) on each ODR. This allows the ODRs in one cell to send requests to a remote cell in the case that no application servers are available in the local cell.

[Generating a plugin-cfg.xml](#)

In order for traffic to be distributed between two or more cells, the HTTP server which fronts the ODRs must be configured to route to the ODRs in all cells.

[Cell affinity when an ODR fails](#)

The cell affinity function enables you to configure unbridged, multi-cell, on demand router (ODR) topologies that preserve sessions even in the event of ODR outages. This topology diagram shows the cell affinity request/response flow when an IBM HTTP server request must be sent to an ODR which is not on the original session path, because the original ODR has become non-functional

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Cell affinity function](#)

Related tasks:

[Configuring cell affinity in a multi-tiered environment](#)

[Defining generic server clusters for remote ODR cells](#)

[Creating and configuring ODRs](#)

[Configuring ODRs](#)

Related reference:
[pluginmerge.bat|.sh script](#)

Cell affinity function

Using the cell affinity function, you can configure unbridged, on demand router (ODR) topologies to preserve sessions even in the event of on demand router (ODR) outages. With this function, you can configure your topology in such a way that, when an ODR receives misrouted, in-session traffic, the ODR reroutes the traffic back to a working ODR in the original cell. Thus, you can configure an IBM® HTTP Server to route to ODRs in multiple cells and still preserve session affinity.

The cell affinity function prevents the loss of sessions when there are multiple ODRs within multiple, unbridged cells, and the IBM HTTP Server is configured to forward session traffic, either through load-balancing or failover, to more than one ODR. For example, in a network configuration where ODRs are situated between the IBM HTTP Server and the back-end application servers, the IBM HTTP Server is not able to recognize the servers identified on the JSESSIONID cookies contained within in-session traffic because it is configured to recognize and route to the ODRs. Thus, the IBM HTTP Server generally selects different ODRs and sprays the session requests. If the IBM HTTP Server selects a router within the same cell as the hosting application server, or if the application servers all share session data through a common database, the risk of losing sessions is not a concern. However, without cell affinity, if the IBM HTTP Server selects an ODR within another cell, the ODR does not recognize the server ID, does not know how to route the request, and consequently the session is lost.

Aspects in cell affinity function

There are two aspects in the cell affinity function. The first aspect enables the IBM HTTP Server to affine, or always route to, a specific ODR after a session is established. Configuring the IBM HTTP Server to preserve session affinity through a particular ODR is accomplished by enabling cell affinity, producing a plugin-cfg.xml,, moving the plugin-cfg.xml to the IBM HTTP Server, and restarting that server. The produced plugin-cfg.xml instructs the IBM HTTP Server plug-in to use the ODRSESSIONID cookie for its session ID, thereby enabling session affinity to the ODRs.

The second aspect of cell affinity is the ability to route session traffic across cell boundaries in order to direct misrouted traffic into the correct cell. To enable this, in addition to enabling cell affinity, generic server clusters (GSCs) must be configured for each cell for which an ODR can receive traffic. The members of those GSCs must be the ODRs in those remote cells. When an ODR receives misrouted session traffic and cell affinity is enabled, it will look at its GSC lists to see whether it can find the ODR identified by the ODR session ID. If it finds a match, it will reroute the traffic to the containing GSC. If the reroute is successful, the final ODR will adopt the session and route the traffic to the appropriate backend server for the user's session.

Parent topic: [Enabling cell affinity](#)

Related tasks:

[Enabling cell affinity](#)

[Configuring cell affinity in a multi-tiered environment](#)

[Defining generic server clusters for remote ODR cells](#)

[Creating and configuring ODRs](#)

[Configuring ODRs](#)

Related reference:

[pluginmerge.bat|.sh script](#)

Defining generic server clusters for remote ODR cells

In order for the on demand router (ODR) to route to other cells, generic server clusters must be defined to represent those cells. The purpose of defining generic server clusters is to allow each ODR to recognize the other ODRs that are in remote cells.

About this task

Defining the generic server clusters provides a way for the ODRs in one cell to send traffic to the ODRs in the other cell. There are several reasons why this is important. First, if all the application servers in a cell are unavailable, the ODRs in that cell need to send requests somewhere. They will send the requests to the GSC (representing the ODRs in the other cell). The ODRs in the other cell will then route the requests to application servers in their own cell, ensuring that the requests are handled successfully. Second, if a request which has session data associated with an application server in Cell1 is mistakenly sent to an ODR in Cell2, the ODR in Cell2 needs to be able to forward the request to an ODR in Cell1, as only the ODRs in Cell1 can send the request to the appropriate application server (The ODRs in Cell2 cannot send requests directly to an application server in Cell1). The GSC will again be used to allow the ODR in Cell2 to forward the request to an ODR in Cell1 which will then handle the request.

Procedure

1. Configure trusted proxies. All ODRs need to include as trusted proxies all other ODRs (including those in remote cells), and all WebServer/IHS Servers that will send traffic to this ODR. Follow the WebSphere® Network Deployment instructions in the [Proxy server settings](#) topic for most of the instructions to configure ODRs. See [Configuring ODRs](#) for WebSphere Virtual Enterprise specific fields.
2. From the administrative console, create a new generic server cluster. Select **Servers > Generic Server Clusters > New**.
3. Provide a name, select a protocol, and click **OK**.
4. Click the newly-created generic server cluster, and click **Ports**.
5. Click **New**, and specify the host name and port number of the ODR as the member of the generic server cluster.

When defining GSC members in which more than one GSC member (including members in different GSCs) resides on the same node (i.e. has the same host name), you must define the `server=<uniqueServerName>` custom property on the GSC member definition in order to ensure uniqueness of the GSC member names in the ODC. Without this custom property, the GSC member name will be based on the host name configured, which will not be unique for two members on the same node and will result in improper routing. With this property, the GSC member name will be the value provided in the custom property.

The ODR normally calculates the cloneID for a server from the host name and the non-SSL ODR port. This same mechanism is used to calculate GSC member cloneIDs. Thus, when an ODR in one cell is represented by a GSC member in another cell, the cloneIDs automatically match. However, an override mechanism is provided which allows the cloneID to be specified manually. To do so, define a custom GSC member property named `ODRClonelD` with the value set to the desired cloneID. This value must match the calculated cloneID of the ODR in the remote cell.

6. Repeat step 5 for each ODR in the cell that the generic server cluster represents.
7. Repeat steps 1 through 5 for each cell in your topology.
8. Click **Save** to save your configuration changes.

Results

Configuring the GSC representations of the ODRs in the remote cell allows traffic which was misrouted to one cell to be forwarded to the correct cell, maintaining cell affinity. It does not enable failover, meaning that if an ODR has no available servers in its cell, it will be unable to service the request. To enable the ODR to send the request to an application server in a remote cell, in the case that no local servers are available, see [Configuring the on demand router for multi-cluster failover and load balancing routing](#).

Parent topic: [Enabling cell affinity](#)

Related concepts:

[Cell affinity function](#)

Related tasks:

[Creating and configuring ODRs](#)

[Configuring ODRs](#)

[Enabling cell affinity](#)

[Configuring cell affinity in a multi-tiered environment](#)

Related reference:

[pluginmerge.bat/sh script](#)

Configuring cell affinity in a multi-tiered environment

You can configure your network to contain multiple tiers of on demand routers (ODRs). In multi-tiered configurations, you must configure the ODRs in one tier independently from the ODRs in the other tiers, because the ODRSESSIONIDs of two different tiers must not be the same.

Before you begin

You must enable cell affinity and configure the ODRs. See [Creating and configuring ODRs](#) and [Enabling cell affinity](#) for details.

About this task

An example environment for configuring cell affinity might include a cluster of ODRs that distribute load to multiple IBM® Portal application servers at the front-end of the network, while at the back-end there are clusters of ODRs that distribute load to multiple Web content managers.

There are three important configuration requirements for this task:

1. Within a given tier, any generic server cluster configurations you create for cell affinity must refer only to ODRs within that tier.
2. Within a given tier, the ODR session identifier cookie name, which is named ODRSESSIONID by default, must be unique for all the other tiers. You must configure this cookie name as a custom property on each ODR within that tier, as described in the procedure in this topic.
3. If you are using the highly available (HA) plugin configuration generator, each plugin-cfg.xml configuration must define the ODR session identifier cookie name to use the IBM HTTP Server to configure ODR affinity. To define the ODR session identifier cookie name, define an additional cell-wide custom property associated with the plugin-cfg.xml generation configuration:
ODCPluginCfgOdrSessionIdCookie_<configname>=<CookieName>. For more information on configuring the HA plug-in configuration generator, see the [Configuring an ODR to dynamically update the web server plug-in configuration](#) topic.

Procedure

1. From the administrative console, select **Servers > On Demand Routers > <odr_name>**.
2. Expand **On Demand Router Properties**.
3. Click **On Demand Router Settings > Custom Properties > New**.
4. Type **odrSessionIdCookieName** for the name value.
5. Type the new cookie name. For example, ODRSID_TIER1.
6. Repeat steps 3 through 5 for every ODR within the tier.

Parent topic: [Enabling cell affinity](#)

Related concepts:

[Cell affinity function](#)

Related tasks:

[Creating and configuring ODRs](#)

[Configuring ODRs](#)

[Enabling cell affinity](#)

[Defining generic server clusters for remote ODR cells](#)

Related reference:

[pluginmerge.bat/sh script](#)

Configuring the multi-cluster routing policy

In order to configure the on demand router (ODR) to route requests to a generic server cluster, you must configure a multi-cluster routing policy (MCRP) on each ODR. This allows the ODRs in one cell to send requests to a remote cell in the case that no application servers are available in the local cell.

About this task

Configure a multi-cluster routing policy (MCRP) on each ODR, in all cells, in order to configure the on demand router (ODR) to route requests to a generic server cluster. To configure all ODRs in each cell, complete the following steps:

Procedure

1. **Servers > On demand routers > ODR_name > On demand router properties > On demand router settings > Custom properties > New.** \$<applicationName> is optional. If not specified, this rule applies to all traffic sent to this ODR. If specified, it only applies to traffic destined for the application with the name <applicationName>.
2. For the property name, type MCRP@<thisCellName>\$<applicationName>.
3. For the property value, type failover@<thisCellName>\$<PrimaryDynamicCluster>, <thisCellName>\$<GSCName>. This tells the ODR to first attempt to send traffic which matches this MCRP rule (based on the application name specified in Step 2, if any) to the local dynamic cluster of application servers that are in this cell. If none of the local application servers are available, it will failover to the generic server cluster specified (which is the ODRs in the other cell, which will then route the traffic to the application servers local to that cell).

Previous topic: [Defining generic server clusters for remote ODR cells](#)

Previous topic: [Enabling cell affinity](#)

Previous topic: [Configuring the on demand router for multi-cluster failover and load balancing routing](#)

Parent topic: [Enabling cell affinity](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Creating and configuring ODRs](#)

Related information:

 [Proxy server settings](#)

Generating a plugin-cfg.xml

In order for traffic to be distributed between two or more cells, the HTTP server which fronts the ODRs must be configured to route to the ODRs in all cells.

About this task

The HTTP server which fronts the ODRs must be configured to route to the ODRs in all cells.

Procedure

1. Generate a new plugin-cfg.xml file for each of the involved cells. Following are the two options for doing this:
 - a. Generate the plug-in configuration in a high availability environment. This ensures that, despite the other processes available, the configuration is always generated by one process in the cell. See [Generating the plug-in configuration in a high availability environment](#) for details.
 - b. Automatically generate the plugin using an ODR in the cell. You must override the default attributes and ensure that they are all on cell-wide scope. See [Configuring an ODR to dynamically update the web server plug-in configuration](#) for details.
2. Merge the plugin-cfg.xml files into one plugin-cfg.xml file by using the plug-in tool merge tool. See the [pluginmerge.bat|.sh script](#) topic to learn how to use the tool.
3. Install the new plugin-cfg.xml on the IBM HTTP Server. See [Configuring custom HTTP servers](#) for details.

Parent topic: [Enabling cell affinity](#)

Related concepts:

[Cell affinity function](#)

Related tasks:

[Creating and configuring ODRs](#)

[Configuring ODRs](#)

[Enabling cell affinity](#)

[Configuring cell affinity in a multi-tiered environment](#)

Related reference:

[pluginmerge.bat|.sh script](#)

Cell affinity when an ODR fails

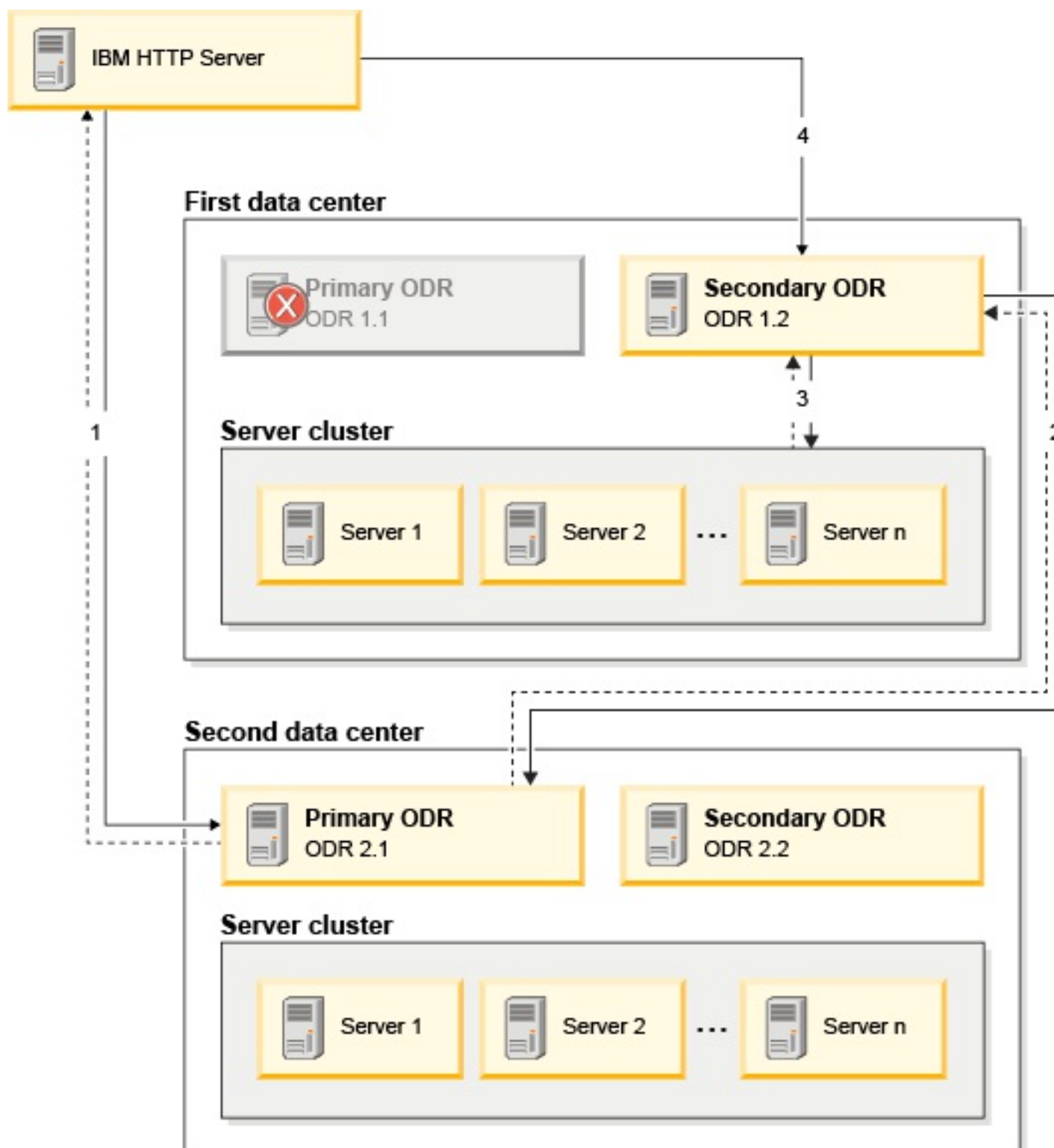
The cell affinity function enables you to configure unbridged, multi-cell, on demand router (ODR) topologies that preserve sessions even in the event of ODR outages. This topology diagram shows the cell affinity request/response flow when an IBM® HTTP server request must be sent to an ODR which is not on the original session path, because the original ODR has become non-functional

A scenario featuring a request/response flow is shown in the following diagram. In this scenario, the browser has sent an in-session request to the IBM HTTP server. The IBM HTTP server has determined it cannot forward the request to the original ODR 1.1, so it has chosen to forward the request to ODR 2.1 instead (normally, this would break the session). The solid arrows in the diagram represent requests, whereas the broken arrows represent responses. The flows are explained below in the following sequence:

1. The browser sends a request to IBM HTTP server. ODR1.1 is not functioning. In an attempt to failover, IBM HTTP Server routes to ODR2.1.
2. ODR2.1 notes that the request was originally destined for ODR1.1, so ODR2.1 finds a generic server cluster generic server cluster containing ODR 1.1, and routes back to an active ODR within the generic server cluster, namely. ODR 1.2.
3. ODR1.2 marks this session for adoption during response processing and forwards the request to the original back-end target cluster.

The IBM HTTP server could route directly to ODR 1.2 after detecting that ODR 1.1 failed. In this case, ODR 1.2 would forward the request to the correct back-end target cluster and adopts the session during response processing as described in #3 and #4 previously.

The following diagram illustrates a request/response flow scenario in which a browser sends an in-session request to the IBM HTTP Server.



Parent topic: [Enabling cell affinity](#)

Related tasks:

[Enabling cell affinity](#)
[Configuring cell affinity in a multi-tiered environment](#)
[Defining generic server clusters for remote ODR cells](#)
[Creating and configuring ODRs](#)
[Configuring ODRs](#)

Related reference:

[pluginmerge.bat|.sh script](#)

Using the binary trace facility

WebSphere® Virtual Enterprise provides a binary trace facility (BTF) that, by default, gathers trace information in a production environment that is useful for customers and development teams, because the level of trace gathered facilitates troubleshooting problems without having to reproduce the problem. Further, optimal performance in your environment is not compromised.

About this task

The primary advantage of using the BTF is to provide an efficient means, with respect to CPU and disk space, of gathering trace information in a production environment that is usable and understandable to customers, support, or development teams. With a more efficient trace facility, the level of trace gathered by default facilitates troubleshooting problems without having to reproduce the problem or affecting good performance in your environment. Three ways to use the BTF are described as follows:

1. Use the trace data that is generated by default.
2. Modify trace dynamically by setting a cell custom property. See [Binary trace facility custom properties](#).
3. Modify the WAS_install/etc/trace-cfg.xml file. All BTF tracing is performed on the log directory of each server. By default, the files are currently named btrace.#, where # is a monotonically increasing integer starting at 1. You can edit and copy this file to the profile_root/config/cells/cell_name directory or to the same directory that contains the server.xml file for the process.

Note: Modifying the trace-cfg.xml file is not recommended. It is not a dynamic file, is a more technical method, and could break during future migrations.

Procedure

1. To get a complete usage description of the trace command, run the following command from the install_root/bin/trace directory. The following is a sample output:

```
myServer:/opt/WAS611/bin #./trace
Usage error: too few arguments
Options:
  To read binary trace:
    read <logDirectory> [<traceSpec>] [-filePrefix <prefix>]
  To get information about trace groups, records, and fields:
    info [<groupOrRecordOrFieldName>] [-depth <maxDepth>]
where:
  <logDirectory> is the path to a server log directory
  <traceSpec> is trace specification, which is a comma-separated list of
elements of the form:
  <groupOrRecord>[:<traceLevel>]
  <prefix> is the trace file prefix name (default value is 'btrace.')
  <groupOrRecordOrFieldName> is the name of a group, record, or field
for which information is needed
  <maxDepth> is the maximum depth to recurse when printing group,
record, or field information.
xdblade01b02:/opt/WAS611/bin #
```

2. To get an XML description of everything that can be traced, run the following command:

```
install_root/bin/trace info
```

3. To control the amount of information printed, use the -depth <maxDepth> option. To control the type of information that is printed, add an additional argument specifying the name of a group, record, or field. For example, to see everything immediately below the HTTP group, type the following:

```
install_root/bin/trace info http.request -depth 1
```

[Options for filtering information in the trace-cfg.xml](#)

Various options are available for controlling the output of the trace-cfg.xml, and for reading the output when using the binary trace facility.

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Options for filtering information in the trace-cfg.xml](#)

Related information:

[Binary trace facility custom properties](#)

Options for filtering information in the trace-cfg.xml

Various options are available for controlling the output of the trace-cfg.xml, and for reading the output when using the binary trace facility.

Overview

There are three types of information defined in the trace-cfg.xml file: groups, records, and fields. These three types of information are arranged hierarchically as follows:

1. Groups: can contain groups or records
2. Records: can contain records or fields
3. Fields are the leaf of the tree and cannot contain sub-elements.

Using the -depth and read options in the BTF, you can control what information is printed, and determine what you want filtered out of trace by specifying a traceSpec.

-depth options

You can use the -depth <maxDepth> option to control the types of information you want printed. To do this, add an additional argument specifying the name of a group, record, or field that you want to use as a filter. For example, to see all information immediately following the http group, you would specify `<InstallRoot>/bin/trace info http -depth 1`. To see information immediately below the http.request group, you would run `<InstallRoot>/bin/trace info http.request -depth 1`.

-depth with arguments

Add arguments to the -depth command to further filter your information. The depth that you specify will include all groups, records, and fields that are within the specified depth from the initial argument provided. For example, to see all information immediately following the http group in the trace-cfg.xml, run the following command:

```
<InstallRoot>/bin/trace info http -depth 1
```

To see all information immediately following the http.request group, run the following command:

```
<InstallRoot>/bin/trace info http.request -depth 1
```

Reading trace

To read the btrace files in the logs directory, run the following command, where <logsDirectory> is the log directory, and <traceSpec> is the variable:

```
<Root>/bin/trace read <logsDirectory> [<traceSpec>]
```

For example, to see all trace in the http group in the logs from the current directory, run the following:

```
<InstallRoot>/bin/trace read. http
```

Parent topic: [Using the binary trace facility](#)

Related tasks:

[Using the binary trace facility](#)

Related information:

[Binary trace facility custom properties](#)

Tuning the on demand router (ODR)

ODRs are servers that act as an intermediary for HTTP requests that are serviced by application servers or Web servers. There are many factors that affect ODR performance. If you discover that your ODRs are not working the way you want, then you might need to tune your ODRs.

Before you begin

- Before modifying the JVM heap size, read [Tuning Java™ virtual machines](#).
- Remove all tracing information except for `*=info`, because additional tracing strings impact ODR performance when tuning. To change the tracing information, perform the following steps:
 1. Select **Servers > On Demand Routers > <on demand router> > Troubleshooting > Logging and Tracing > Diagnostic Trace Service > Change Log Detail levels**.
 2. Ensure that only `*=info` is specified.
- The ODR should never be constrained by either CPU or memory usage. Therefore, when you install the ODR in an environment with server virtualization, configure the virtual machine or LPAR in which the ODR runs in dedicated processor mode, or configure it in a mode that guarantees the ODR receives a sufficient amount of CPU resources and dedicated memory when the ODR runs.

About this task

Use the topics in this section to troubleshoot your ODRs when they are not operating in a typical fashion, or you are experiencing unusual problems. The following list contains tuning suggestions for improving ODR performance suggestions in order of importance for troubleshooting.

Procedure

1. Check the JVM settings. See [Modifying the JVM heap size for the on demand router](#).
2. **HP-UX** HP and Sun provide additional tuning parameters to optimize garbage collection. For generational garbage collection JVMs such as Sun and HP, or IBM's J9 JVM when using gencon garbage collection, set the permanent memory region to approximately 100MB to encompass the base 90MB footprint that exists in the ODR. Further, a SurvivorRatio of 16 further optimizes processing in the young generation. On HP JVM, you can turn NIO to yield an increase in performance by using the `-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.DevPollSelectorProvider` selector provider and disabling the extra poll before a read: `-XX:-ExtraPollBeforeRead`.
3. Tune the connection keep alive settings. See [Tuning ODR persistent connections](#).
4. Increase the ODR maximum connections per server. See [Tuning ODR maximum connections](#)
5. Disable ODR caching when not in use. When the ODR caching is enabled, the ODR must go through the process of determining whether a request should be cached, then examine the cache repository to check whether the request was previously cached. This causes additional overhead.
6. Disable access logging if not needed. If you do need access logging, then the proxy logging is preferred over the HTTP Channel/NCSA logging as the proxy access logging happens outside of the request/reply path. Thus, it does not affect the response time of the request. Access logging on a fairly fast disk is typically 5% overhead, but the percentage is highly dependant on disk performance.
7. Use the same thread group for both inbound and outbound work, which will avoid moving requests across threads and eliminate the resulting overhead. The ODR has a set of threads that tune itself under most circumstances. Queuing and throttling requests are dispatched to the default thread pool, which you can tune so that it will only handle the overflow requests. The primary thread group will continue to handle most requests. All requests on the thread pool are asynchronous with no blocking calls, so the number should not be more than one or two threads per CPU. Complete the following steps to use the same thread group for both inbound and outbound work:
 - a. Select **Servers > On Demand Routers > on demand router > Thread Pools default thread pool > Custom Properties > New**
 - b. Specify **combineSelectors** for the name.
 - c. Specify **1** for the value.
 - d. Click **OK**.
 - e. Click **Save**.

[Modifying the JVM heap size for the on demand router](#)

The Java virtual machine (JVM) heap size setting is the single most critical factor for efficient on demand router (ODR) performance. You might need to modify the JVM heap size setting based on your environment configuration.

[Tuning ODR persistent connections](#)

Tune the persistent connection settings for ODR performance to minimize the time you spend setting up new connections and taking down older connections during a barrage of requests. Primarily, this tuning is relevant to the connections between the ODR and the application servers.

[Tuning ODR maximum connections](#)

Increasing ODR maximum connections per server allows for more connections to be created and pooled, and enables keeping connections for a significant amount of time.

Parent topic: [Creating and configuring ODRs](#)

Related tasks:

[Creating and configuring ODRs](#)

[Tuning ODR persistent connections](#)

Modifying the JVM heap size for the on demand router

The Java™ virtual machine (JVM) heap size setting is the single most critical factor for efficient on demand router (ODR) performance. You might need to modify the JVM heap size setting based on your environment configuration.

About this task

The base setting for the ODR under 100% load is approximately 90MB. However, the ODR also needs memory for temporary working space. Since the ODR acts the same for every request, the major factor to determine how much working space is required is throughput. The ODR generates approximately 4 to 5KB of temporary objects per request. The desired time between garbage collection cycles should be at least 10 seconds to minimize heap contention, and generally under 60 seconds to avoid excess delays in completing garbage collection cycles. Test this value, as may factors such as high numbers of queued requests or large in-memory caches can increase the required memory. Testing is important, because values cannot be determined arbitrarily for all possible applications. Therefore, the recommended initial heap settings is calculated as

$$90MB + 0.05 \text{ MB/request} \times \text{peak requests per second} \times \text{desired time between garbage collection (sec)}$$

Procedure

1. In the administrative console click **Servers > On Demand Router > on demand router name > Server Infrastructure > Java and process management > Process definition > Java virtual machine**.
2. Specify the initial heap size and maximum heap size.

Parent topic: [Tuning the on demand router \(ODR\)](#)

Related tasks:

[Tuning the on demand router \(ODR\)](#)

[Tuning ODR persistent connections](#)

[Creating and configuring ODRs](#)

Tuning ODR persistent connections

Tune the persistent connection settings for ODR performance to minimize the time you spend setting up new connections and taking down older connections during a barrage of requests. Primarily, this tuning is relevant to the connections between the ODR and the application servers.

Before you begin

Modify your JVM heap settings. See [Modifying the JVM heap size for the on demand router](#).

About this task

Adjust the application server's number of persistent requests so that the ODR reuses the connection to the application server for as long as it can. To do this, configure the ODR transport chain to tune inbound connections to the ODR. To tune outbound connections from the ODR, configure the transport chains of the application servers that are members of a dynamic cluster. To do so, configure the transport chain of the dynamic cluster from within the dynamic cluster server template. Changes to the dynamic cluster's server template are automatically applied to all cluster members.

You should set number of persistent requests high enough to avoid having to bring down and recreate a connection during a barrage of requests, so set the value to at least the number of requests in a single barrage of request. The barrage can be a single page with associated data with a long think time, or a series of pages and associated data in short succession. Setting a value higher than the number of requests in a single barrage of request will not negatively affect performance provided the persistent timeout is set appropriately to close idle connections.

Procedure

- Tune the ODR's HTTP_PROXY_ADDRESS and HTTPS_PROXY_ADDRESS transport chains. In the administrative console, click **Servers > On Demand Routers > odr_name > Ports**.
 1. Click **View associated transports** for the port whose transport chains you want to view.
 2. Click *transport_chain_name*.
 3. Click **HTTP inbound channel**.
 4. Ensure that **Use persistent (keep-alive) connections** is checked.
 5. Specify an unlimited number of persistent requests per connection. Select **Unlimited persistent requests per connection**.
 6. Specify a maximum number of persistent requests per connection. Select **Maximum persistent requests per connection**. Setting this value to **-1** yields the best performance.
 7. Click **Apply** and **OK** to save your changes.
- Tune the application servers' HTTP_PROXY_ADDRESS and HTTPS_PROXY_ADDRESS transport chains. In the administrative console, click **Servers > Dynamic clusters > dynamic_cluster_name > Server template > Web Container Settings > Web container transport chains**.
 1. Configure the WCInboundDefault transport chain.
 - a. Click **WCInboundDefault**.
 - b. Click **HTTP inbound channel**.
 - c. Ensure that **Use persistent (keep-alive) connections** is checked.
 - d. Specify an unlimited number of persistent requests per connection. Select **Unlimited persistent requests per connection**.
 - e. Specify a maximum number of persistent requests per connection. Select **Maximum persistent requests per connection**, and set the value to **-1**. This allows an infinite number of requests to flow over a single connection. Connections that are not used for an extended period of time are still closed due to the inactivity timeout for the connection.
 - f. Click **Apply** and **OK** to save your changes.
 2. Configure the WCInboundDefaultSecure transport chain.
 - a. Click **WCInboundDefaultSecure**.
 - b. Click **HTTP inbound channel**.
 - c. Ensure that **Use persistent (keep-alive) connections** is checked.
 - Select **Unlimited persistent requests per connection** to specify an unlimited number of persistent requests per connection.
 - Select **Maximum persistent requests per connection** to specify a maximum number of persistent requests per connection. Setting this value to **-1** yields the best performance.
 - d. Click **Apply** and **OK** to save your changes.

Parent topic: [Tuning the on demand router \(ODR\)](#)

Related tasks:

[Creating and configuring ODRs](#)

[Tuning ODR maximum connections](#)

[Modifying the JVM heap size for the on demand router](#)

[Tuning the on demand router \(ODR\)](#)

Tuning ODR maximum connections

Increasing ODR maximum connections per server allows for more connections to be created and pooled, and enables keeping connections for a significant amount of time.

Before you begin

Modify your JVM heap settings. See [Modifying the JVM heap size for the on demand router](#).

About this task

To enable keeping connections for a significant time, the ODR maximum connections per server should be increased to allow for more connections to be created and pooled. The ODR allocates connections beyond the setting specified, but will not pool those connections when they are returned to the pool if the pool already contains this amount of idle connections. A general setting for the maximum connections per server can be calculated as the peak number of concurrent clients divided by the number of servers allotted for an application. For dynamic server clusters, the number of servers is the minimum number of servers the cluster is defined to.

The following steps show you how to adjust the application server's maximum connections per server. From the administrative console:

Procedure

1. Click **Servers > On Demand Routers *odr_name* > On Demand Router Properties > On Demand Router Settings**.
2. Use the calculation described previously and enter your result in the **maximum connections per server** field.

Parent topic: [Tuning the on demand router \(ODR\)](#)

Related tasks:

[Creating and configuring ODRs](#)

[Modifying the JVM heap size for the on demand router](#)

[Tuning ODR persistent connections](#)

[Tuning the on demand router \(ODR\)](#)

Configuring custom logs

You can configure custom logs through the wsadmin tool with the ruleset administrative tasks. You can configure custom logs through the rule set configuration objects. Validation is limited to type checking. More thorough validation is performed during an on demand router (ODR) startup by issuing runtime exceptions to SystemOut.

About this task

You can create custom logs with the following commands. You can also create custom logs with the manageODR.py script.

Procedure

1. Create a new rule set:

```
wsadmin>$AdminTask createRuleset {-odrname odr1 -nodename myNode01 -
rulesetName myRuleset -rulesetType HTTP -defaultContinue true}
```

2. Add a rule to the rule set that you just created, with the expression set to match when the virtualhost is NOT *badvhost*.

```
wsadmin>$AdminTask addRuleToRuleset {-odrname odr1 -nodename myNode01 -
rulesetName myRuleset -ruleName myRule -rulePriority 0 -expression
"virtualhost <> 'badvhost'"}}
```

3. Add a custom log action. Note the @ symbol that is used in the actionValue argument. The following example shows the wsadmin command for listing the configurable attributes for the action and the results:

```
wsadmin>$AdminTask addActionToRule {-odrname odr1 -nodename myNode01 -
rulesetName myRuleset -ruleName myRule -actionName myCustomLogAction -
actionType log -actionValue "Custom1.log %r %B %h %z %Z@Custom2.log %t %r
%s" -actionContinue true}
```

4. Save your changes.

```
wsadmin>$AdminConfig save
```

Example

Two examples of custom log outputs follow. These examples are based on the formats specified in the previous actions, after running several HTTP requests.

```
"GET /BADAPP HTTP/1.1" 70 myremotehost.mycompany.com - -
"GET /A/ServerInfo HTTP/1.1" 422 myremotehost.mycompany.com 9.44.111.191
myCell01/myNode01/MyClusterA_myNode01
"GET /A/ServerInfo HTTP/1.1" 90 myremotehost.mycompany.com - -
"GET /favicon.ico HTTP/1.1" 70 myremotehost.mycompany.com - -
"GET /favicon.ico HTTP/1.1" 70 myremotehost.mycompany.com - -
"GET /A/ServerInfo HTTP/1.1" 422 myremotehost.mycompany.com 9.44.111.191
myCell01/myNode01/MyClusterA_myNode01
"GET /A/ServerInfo HTTP/1.1" 422 myremotehost.mycompany.com 9.44.111.191
myCell01/myNode01/MyClusterA_myNode01
```

```
01/Jul/2009:20:35:35 EDT "GET /BADAPP HTTP/1.1" 404
01/Jul/2009:20:35:43 EDT "GET /A/ServerInfo HTTP/1.1" 200
01/Jul/2009:20:37:21 EDT "GET /A/ServerInfo HTTP/1.1" 503
01/Jul/2009:20:37:21 EDT "GET /favicon.ico HTTP/1.1" 404
01/Jul/2009:20:37:24 EDT "GET /favicon.ico HTTP/1.1" 404
01/Jul/2009:20:42:55 EDT "GET /A/ServerInfo HTTP/1.1" 200
01/Jul/2009:20:42:59 EDT "GET /A/ServerInfo HTTP/1.1" 200
```

[Custom logs overview](#)

Custom logs give you the flexibility as to what data is captured, the manner in which it is displayed, and the ability to have multiple conditional logs. You can create rulesets to specify the data to include, the order of how it is displayed, the conditional logic to filter which requests are included, and the number and names of log files.

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Custom logs overview](#)

Related reference:

[Ruleset administrative tasks](#)

[manageODR.py script](#)

[HTTP operands](#)

Custom logs overview

Custom logs give you the flexibility as to what data is captured, the manner in which it is displayed, and the ability to have multiple conditional logs. You can create rulesets to specify the data to include, the order of how it is displayed, the conditional logic to filter which requests are included, and the number and names of log files.

Custom property objects

Custom property objects are allowed at the ruleset, rule, and action level. Rules are used to determine the conditions upon which log actions are executed. An expression field is used to determine the requests and responses to log. This expression uses the standard HTTP language operands, plus operand *response.code* to filter by HTTP response codes, such as 404, 503, and so on. For example, *response.code = 200* would get all responses of response code 200. The HTTP operand *targetserver* is used to show the server where the request was sent in a WebSphere® Application Server format. The HTTP custom log operand *service.time* is the time frame between when the request is sent to the application server, and when a response is received from the application server. The HTTP custom log operand *response.time* is the timeframe between when the request in the ODR is received, and when the request until the response from the ODR is sent. See the [HTTP operands](#) topic for HTTP custom log operands information.

Rules have priorities to determine the order in which they should be evaluated. There are log actions which are done when rules are matched. The log actions have a *continue* field. When *continue=true*, subsequent log actions can still be executed. However, if *continue=false*, once this log action is finished, no subsequent log actions will be performed. You can use the *manageODR.py* script to create a cluster of ODRs and to manage custom logging. For more information, see the [manageODR.py script](#) topic.

ruleset

name

Specifies the name of the ruleset. (String, required)

type

Specifies the type of ruleset. (String, protocol identifier such as HTTP)

continueDefault

Determines the default value for *continue*. (boolean, required)

properties

Optional

rule

Required

rule

name

Specifies the name of the rule. (String, unique within ruleset)

priority

Required, integer (unique within ruleset)

expression

String, required.

action

Required.

properties

Optional

Action

name

Specifies the name of the action. (String, unique within rule)

priority

Optional, integer (unique within rule).

type

String, required, log

value

Required.(String, contains a series of logFormats delimited by @. For example: value="custom01.log %a%z@custom02.log %A%Z"

continue

Optional, boolean. If not set, defaults to continueDefault value; true, false.

properties

Optional

customProperties

name

Specifies the name custom property. (String)

value

Required.(String)

continue

Optional, boolean. If not set, defaults to continueDefault value; true, false.

Log action format

The action's value attribute contains a set of arguments to indicate the filename and format of the log. To create multiple logs in the value attribute, delimit the arguments with a @. The first argument for the log action is the name of the log file, relative to the ODR 's logs directory. The following arguments are parameters to specify what information to place into the log. The data is shown in the order that the parameters are specified.

Table 1. Custom log parameters

Parameter	Description
%a	Remote IP -address.
%A	Local IP - address.
%B	Bytes sent, excluding HTTP headers.
%b	Bytes sent, excluding HTTP headers. In CLF format, that is, a '-' rather than a zero when no bytes are sent.
% {FOOBAR }C	The contents of the cookie in the request sent to the server.
% {FOOBAR }i	The contents of Foobar: header line(s) in the request sent to the server.
%l	Bytes received, including request and headers, cannot be zero.
% {FOOBAR }e	The contents of the debug argument ("-D") FOOBAR specified for the JVM of the ODR process.
%h	Remote host.
%H	The request protocol HTTP or HTTPS.
%m	The request method.
% {FOOBAR }o	The contents of Foobar, header line(s) in the reply.
%O	Bytes sent, including headers; the number cannot be zero.
%p	The canonical port of the server serving the request.
%q	The query string, prepended with a ? if a query string exists. Otherwise, it is an empty string.
%r	First line of request.
%R	Response time in milliseconds. Combined time spent in ODR and application server.
%s	Status, HTTP response code, that is, 503, 404, 200.
%t	Time, in common log format time format, standard English format.
% {format}t	The time, in the form given by format, which should be in strftime(3) format. Potentially localized.
%T	The time taken to serve the request, in milliseconds.
%U	The URL path requested, not including any query string.
%v	The canonical ServerName of the server serving the request.
%z	Destination IP-address.
%Z	Destination server name (cell/node/server).

Parent topic: [Configuring custom logs](#)

Related tasks:

[Configuring custom logs](#)

Related reference:

[manageODR.py script](#)

[HTTP operands](#)

Configuring SSL offload for all HTTPS traffic

You can change the default outgoing protocol for requests that go through the on demand router. By default, the on demand router matches the incoming protocol to the outgoing protocol.

Before you begin

Create the on demand router. See [Creating ODRs](#) for more information.

About this task

The on demand router has the following default behaviors:

- For inbound Hypertext Transfer Protocol (HTTP) requests, the request is forwarded over outbound HTTP.
- For inbound Hypertext Transfer Protocol Secure (HTTPS) requests, the request is forwarded over outbound HTTPS.

You can change this default behavior for all HTTP and HTTPS traffic that the on demand router handles, or on a per-web module basis.

Procedure

1. Configure the on demand router (ODR) to perform Secure Sockets Layer (SSL) offload for all HTTPS traffic. In the administrative console, click **Servers > On Demand Routers > *on_demand_router* > On Demand Router Properties > On Demand Router settings > Custom properties > New**.
2. Specify a custom property named **http.protocolMap** and a value of SSL-offload.
3. Click **Apply**.

Parent topic: [Creating and configuring ODRs](#)

Configuring SSL offload for partial HTTPS traffic

About this task

Configure the ODR to perform SSL offload for partial HTTPS traffic, or on a per-web module basis.

Procedure

1. In the administrative console, click **Applications > Enterprise Applications > *application_name* > Manage Modules > *web_module_name* > Web Module Proxy Configuration**.
2. For **Web Module Transport Protocol**, select **HTTP**.

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Preparing the hosting environment for dynamic operations](#)

[Creating ODRs](#)

[Configuring ODRs](#)

Related reference:

[createodr.jacl script](#)

Related information:

[On demand router system and custom properties](#)

Configuring SSL offload for partial HTTPS traffic

Configure the ODR to perform SSL offload for partial HTTPS traffic, or on a per-web module basis.

Procedure

1. In the administrative console, click **Applications > Enterprise Applications > *application_name* > Manage Modules > *web_module_name* > Web Module Proxy Configuration**.
2. For **Web Module Transport Protocol**, select **HTTP**.

Parent topic: [Creating and configuring ODRs](#)

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Preparing the hosting environment for dynamic operations](#)

[Creating ODRs](#)

[Configuring ODRs](#)

Related reference:

[createodr.jacl script](#)

Related information:

[On demand router system and custom properties](#)

Integrating the SIP on demand router with Load Balancer

You can integrate the Session Initiation Protocol (SIP) on demand router (ODR) with Load Balancer. Load Balancer for IBM® WebSphere® Application Server can help maximize the potential of your Web site by providing a powerful, flexible, and scalable solution to peak-demand problems.

Before you begin

z/OS Restriction: WebSphere Virtual Enterprise does not support SIP features on the z/OS® operating system.

- Install Load Balancer for IBM WebSphere Application Server Version 6.1. See the [Edge Components Information Center](#) for installation information.
- Start the Session Initiation Protocol (SIP) proxy server.

Procedure

1. Start the Load Balancer.
 - a. From the command prompt, type `dsserver start`.
 - b. Then type `lbadmin` to start the administrative console for the Load Balancer.
 - c. From the administrative console, right click on **Dispatcher**, and then select **Connect to Host**.
 - d. Right click on the hostname and select **Start Executor**.
2. Start the configuration wizard for the load balancer. Right-click **Dispatch > Start Configuration Wizard**.
 - a. Select default host.
 - b. Type a cluster address. The cluster address should not be pingable before the Executor starts. You must specify this same value for host when you create a user-defined port.
 - c. Type a port number, such as 5060.
 - d. Add servers to the port. Add each server to which the load balancer will proxy traffic. In your configuration, the load-balanced server is the ODR server for your WebSphere Application Server configuration.
 - e. Start an advisor by clicking **Yes**. For example, for HTTP traffic, start the HTTP advisor. For SIP traffic, start the SIP advisor. The advisor tells the manager whether or not a specific port is accepting traffic.
3. Alias the cluster address on the SIP proxy server loopback adapter. For example, type `lb-alias.sh <cluster_ip_address>`:

```
#!/bin/sh

CLUSTER=$1

if [ -f /proc/sys/net/ipv4/conf/all/arp_ignore -a -f
/proc/sys/net/ipv4/conf/all/arp_announce ]
then
echo Using arp_ignore
echo "3" >/proc/sys/net/ipv4/conf/all/arp_ignore
echo "2" >/proc/sys/net/ipv4/conf/all/arp_announce
ip addr add $CLUSTER/32 scope host dev lo
elif [ -f /proc/sys/net/ipv4/conf/all/hidden -a -f
/proc/sys/net/ipv4/conf/lo/hidden ]
then
echo Using hidden
echo "1" > /proc/sys/net/ipv4/conf/all/hidden
echo "1" > /proc/sys/net/ipv4/conf/lo/hidden
ip addr add $CLUSTER/32 dev lo
elif [ ! -z "$( which arptables )" ]
then
echo Using arptables
arptables -A IN -s $CLUSTER -j DROP
arptables -A OUT -s $CLUSTER -j mangle --mangle-ip-s $(hostname)
ip addr add $CLUSTER/32 dev lo
else
echo Using iptables
iptables -t nat -A PREROUTING -d $CLUSTER -j REDIRECT
fi
```

[Load Balancer Administration Guide](#) for more information.

4. Configure an IP sprayer from the WebSphere Virtual Enterprise administrative console.
 - a. From the administrative console, click **Servers > On Demand Routers > odr_name > SIP On Demand Router Settings > SIP On Demand Router Settings**.
 - b. Under General Properties, in the IP sprayer configuration section, select the checkbox for the IP sprayer from which you want the SIP proxy server to receive traffic: **Enable TCP sprayer, Enable SSL sprayer, or Enable UDP sprayer**.
 - c. Enter a value for the **Host**. This is the value you specified in step 2b.
 - d. Enter a value for the **Port**. This is the value you specified in step 2c.
 - e. Click **Apply**, and then click **Save**.
5. Define SIP ODR custom properties from the WebSphere Virtual Enterprise administrative console.
 - a. From the administrative console, click **Servers > On Demand Routers > odr_name > SIP On Demand Router Settings > SIP On Demand Router Settings**.
 - b. Under General Properties, in the Additional Properties section, click **Custom properties**.
 - c. Create the following SIP ODR custom properties:
 - LBIPAddr** : The IP address of the load balancer machine.
 - SIPAdvisorMethodName**: The type of messages sent by the Load Balancer advisory. This should be INFO.
 - serverUDPInterface**: The IP address of the server machine.
 - serverUDPPort** : Specify an unused port number. For example, 5080.
6. Create a user-defined port from the WebSphere Virtual Enterprise administrative console.
 - a. From the administrative console, click **Servers > On demand routers > on_demand_router > Communications > Ports**.
 - b. Click **New**.
 - c. Select User-defined port.
 - d. Enter SIP_LB_Address for the **Port Name**.
 - e. Enter a value for the **Host**. This is the value you specified in step 2b.
 - f. Enter a value for the **Port**. This is the value you specified in 2c. Click **OK. Save** changes.
 - g. Under Ports, modify the **PROXY_SIP_ADDRESS** from * to the actual host name of the proxy server machine.
 - h. Click **Apply**, and then click **Save**.
7. Modify the SIP proxy transports.
 - a. From the administrative console, click **Servers > On Demand Routers > odr_name > SIP On Demand Router Settings > SIP On Demand Router Settings > SIP On Demand Router transports > > UDP_SIP_PROXY_CHAIN > UDPInbound Channel (UDP_1)**.
 - b. From the **Port** drop-down list, select **SIP_LB_Address**.
 - c. Click **Apply**, and then click **Save**.
8. Restart the proxy server to save your changes. Use the following additional steps to integrate the ODR with the load balancer:
9. Verify that the host name of the ODR machine is set for the PROXY_SIP_ADDRESS. To verify from the administrative console, click **Servers > On Demand Routers > odr_name > Communications > Port > PROXY_SIP_ADDRESS**.
10. Create a new UDP transport chain. Click **Servers > On Demand Routers > odr_name > SIP On Demand Router Settings > SIP On Demand Router transports > New**.
 - a. Type a name for the new chain, such as ODR_SIP_PROXY_CHAIN.
 - b. In the pull-down, select **Proxy-UDP**.
 - c. Select **Next** and then **Finish** on the summary page. Note: The PROXY_SIP_ADDRESS might not show in the list of existing ports. In this case, select any of the ports to complete the transport chain creation, then modify the UDP inbound channel of the just created new chain afterwards. Select the new transport chain, then select UDP inbound channel and change the port to PROXY_SIP_ADDRESS.
11. Create a system property named clientUDPInterface on each ODR and set it to the IP address of the cluster address. To create the system property, select **Servers > On Demand Routers > odr_name > Java™ and Process Management > Process Definition > Java Virtual Machine > Custom Properties > New**.
12. **Save** and synchronize the configuration.
13. Restart the ODR.

Next topic: [Overview of request flow prioritization](#)

Parent topic: [Creating and configuring ODRs](#)

Related tasks:

[Installing a Session Initiation Protocol proxy server](#)

[Creating and configuring ODRs](#)
[Configuring a Web server as a trusted proxy server](#)

Related information:

[On demand router system and custom properties](#)

Defining routing policies for generic server clusters

You can define routing policies for generic server clusters. A routing policy is a set of rules that determine how the server routes incoming requests.

Procedure

1. In the administrative console, click **Servers > On demand routers > ODR_name > On demand router properties > Generic server cluster routing properties**.
2. Click **Work classes for HTTP requests** and click **New**.
3. Type a name for the work class in the **Name** field.
4. Click **Finish** to confirm your new work class.
5. To set a URI match condition for an existing rule, type the name of the URI in the **If URI matches** field, and from the **For virtual host** list, select a host name.
6. Assign a work class with no rules and provide choices for the other fields in the panel.
7. Click **Apply** or **OK** to commit your new rule settings. The new ODR has a default routing policy, a URI pattern of asterisk (*), a virtual host selection of default_host and a default routing action of Reject with return code with the reject return code set to 404.

What to do next

After you create the ODR and apply any optional configuration parameters, you can define the ability to route work to nodes that are not running WebSphere® Virtual Enterprise.

[Configuring rewriting rules](#)

You can configure rewriting rules to redirect requests to the proxy server, which then routes the requests to the target server. The proxy server acts as an intermediary for HTTP requests that are serviced by application servers or Web servers.

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring rewriting rules](#)

[Defining service policies for generic server clusters](#)

[Configuring the on demand router for multi-cluster failover and load balancing routing](#)

Related reference:

[createodr.jacl script](#)

[Administrative roles and privileges](#)

Related information:

[Proxy server settings](#)

Configuring rewriting rules

You can configure rewriting rules to redirect requests to the proxy server, which then routes the requests to the target server. The proxy server acts as an intermediary for HTTP requests that are serviced by application servers or Web servers.

About this task

Typically, the proxy server recognizes which target server is active and routes all administrative communication to that target. In some cases, however, a target server might receive a request without the proxy server intercepting that request. Rewriting rules modify URLs to point requests to the proxy server instead of pointing requests directly to a target server.

Procedure

1. Click **Servers > On Demand Routers > odr_name > On Demand Router Properties > Rewriting rules > New** in the administrative console.
2. Type the original URL pattern in the **From URL pattern** field. The pattern can include the following wild card symbol: *
3. Type the modification for the URL in the **To URL pattern** field. The pattern can include the following wild card symbol: *

For example, you receive a redirected response with the original location header of `http://internalserver/secure/page.html`. You configure a rule with the following patterns:

- `http://internalserver/*` as the **From URL pattern**
- `http://publicserver/*` as the **To URL pattern**

As a result, the rule rewrites the location header as `http://publicserver/secure/page.html`.

Parent topic: [Defining routing policies for generic server clusters](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Defining routing policies for generic server clusters](#)

Related reference:

[createodr.jacl script](#)

[Administrative roles and privileges](#)

Related information:

[Rewriting rules collection](#)

Defining service policies for generic server clusters

Optionally, you can define service policies for generic server clusters.

About this task

Use the administrative console to define service policies for generic server clusters by clicking **Servers > On demand routers > ODR_name > On demand router properties > Generic server cluster service properties**.

Procedure

1. Select a work class for HTTP requests and click **New**.
2. Type a name for the work class in the **Name** field.
3. Click **Finish** to confirm your new work class.
4. To set a URI match condition for an existing rule, type the name of the URI in the **If URI matches** field, and from the **For virtual host** list, select a host name. These fields are required.
5. If a work class has no rules, from the **If no classification rules apply, then classify to this transaction class** list assign the work class to a transaction class. This field is required whether or not rules exist and is the default action in the absence of rules.
6. Click **Apply** or **OK** to commit your new rule settings.

What to do next

After you create the ODR and apply any optional configuration parameters, you can define the ability to route work to nodes that are not running WebSphere® Virtual Enterprise.

Parent topic: [Creating and configuring ODRs](#)

Related tasks:

[Defining routing policies for generic server clusters](#)

[Configuring the on demand router for multi-cluster failover and load balancing routing](#)

Related reference:

[createodr.jacl script](#)

[Administrative roles and privileges](#)

Related information:

[Proxy server settings](#)

Configuring communication between core groups that are in the same cell

A cell is comprised of one or more processes that serve as hosts for the runtime components. Each cell has one or more named core groups. A core group is a group of processes that are directly accessible to each other and is connected using a local area network (LAN). If core groups that are in the same cell need to share workload management (WLM) information, then you must configure these core groups to communicate.

Before you begin

Create the core groups for your cell. Configure all of the processes within the core groups, including your servers and clusters. The recommendation is a maximum of 40 WebSphere® Application Server Network Deployment related processes such as application servers, node agents, and on demand routers defined in your core group.

About this task

You might configure multiple core groups for a cell in the following scenarios:

- One or more firewalls exist within a cell. A core group cannot contain members from multiple firewall protection domains.
- A large number of processes in the cell and the core group protocols, such as the View Synchrony Protocol, consume correspondingly large amounts of resources, such as processor resources.
- Core group protocols, such as the Failure Detection Protocol, need tuning or configuring to use values that work best with smaller numbers of core group members.

If you are using WebSphere Application Server Network Deployment, you must configure bridge interfaces and core group access points. In WebSphere Virtual Enterprise, this configuration is simplified, because these required elements are automatically configured.

Procedure

1. Add node agents and servers to each core group. Configure at least two node agents for each node group to ensure high availability. When a process starts if core group bridges are not configured, then every product process automatically configures a core group access point for each configured core group. The product assumes that each of the node agents in each core group is a core group bridge. The configuration is modified to make each node agent a core group bridge. You might need to restart the processes again for this configuration to complete.
2. Start the node agents. If two or more node agents per core group are running, the core groups can communicate.

Results

Core groups that are in the same cell can communicate and share WLM information.

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)
[Core groups \(high availability domains\)](#)

Related tasks:

[Configuring WebSphere Virtual Enterprise for cross-cell communication](#)
[Creating ODRs](#)

☞ [Creating a new core group](#)

☞ [Managing node agents](#)

Related reference:

[coregroupsplit.py script](#)

BBSON bulletin board

WebSphere® Virtual Enterprise offers a bulletin board service over network (BBSON) that is independent of the WebSphere Application Server Network Deployment high availability manager. The high availability manager provides a mechanism that allows servers to easily exchange state data. This mechanism is commonly referred to as the bulletin board.

Overview

BBSON, which is enabled by default in WebSphere Virtual Enterprise Version 7.0 and later, alleviates the dependency on the WebSphere Application Server Network Deployment high availability manager. By using BBSON, you avoid constructing and managing core groups and bridges. You can use BBSON for new cells that are bridged to existing cells running WebSphere Virtual Enterprise Version 7.0 and later. For configuration information on the WebSphere Application Server Network Deployment high availability manager, see [Setting up a high availability environment](#) in the WebSphere Application Server Network Deployment information center.

Considerations for using BBSON

Typically, a cell with 50 or more Java™ virtual machines (JVMs) necessitates core group configuration and bridging. If you anticipate your cell to increase to 50 or more JVMs, use BBSON.

To improve efficiency, run WebSphere Virtual Enterprise on all the cells in your node.

Troubleshooting

For problem diagnosis and debugging purposes, use the `manageWWEBB.py` script to collect BBSON state dumps from every process in the product cell. See [manageWWEBB.py script](#) for more information.

Restrictions

The following restrictions apply when using BBSON:

- Do not use BBSON if your cell contains mixed versions, unless the oldest version is WebSphere Virtual Enterprise Version 7.
- Do not use BBSON if you have previous versions of WebSphere Virtual Enterprise in the same cell. BBSON is not compatible with prior versions.

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Core groups \(high availability domains\)](#)

Related tasks:

☞ [Creating a new core group](#)

Related information:

☞ [Setting up a high availability environment.](#)

Configuring WebSphere Virtual Enterprise for cross-cell communication

You can enable cross-cell communication when a WebSphere® Virtual Enterprise administrative cell contains servers that are enabled with on demand router (ODR) that routes to other WebSphere Application Server administrative cells.

Before you begin

Configure the cells that need to communicate with each other and create the on demand router.

About this task

When the ODR needs to route work to servers that are in different cells, you can configure cross-cell communication by creating a core group bridge.

You must use the **crossCellCGBCfg** command to configure WebSphere Virtual Enterprise cells to communicate. Do not create a core group bridge in the administrative console.

The **crossCellCGBCfg** command performs the following configuration actions:

- Enables the core group bridge service between the cell that is running the ODR and another WebSphere Application Server administrative cell.
- Configures core group bridges on all of the node agents in both cells.
- Enables the WebSphere Virtual Enterprise overlay communication function between the cells.

Running the **crossCellCGBCfg** does not remove any existing core group bridge configurations. After you run the script, you can update the core group bridge configuration for your environment.

Procedure

1. Start the node agents and deployment managers that you want to participate in cross-cell communication.
2. Verify that the XD-CGB-EXPORT file is created for each cell. The XD-CGB-EXPORT file is in the *install_root/profiles/deployment_manager_profile_name/config/cells/cell_name* directory. If you are using a firewall, verify that the firewall is open for the ports that are listed in the XD-CGB-EXPORT file to support communication.
3. Verify that the overlaynodes.config file is created for each cell. The overlaynodes.config file is in the *install_root/profiles/deployment_manager_profile_name/config/cells/cell_name* directory. If you are using a firewall, verify that the firewall is open for the ports that are listed in the overlaynodes.config file to support communication.

4. [Enabling communication between cells that have security enabled](#)

If the cells that you are using for cross-cell communication have some form of security enabled, such as Lightweight Directory Access Protocol (LDAP) administration security, set up communication between the cells.

5. Run the **crossCellCGBCfg** command in each cell.

- a. From a command prompt, type the following command against the ODR deployment manager process for each back-end cell to which the ODR cell routes:

```
crossCellCGBCfg create deployment_manager_host deployment_manager_SOAP_port  
path_to_external_cell_XD-CGB-EXPORT path_to_external_cell_overlaynodes.config_file
```

When security is enabled for the **crossCellCGBCfg** command, include the user ID and password as additional parameters: `crossCellCGBCfg create deployment_manager_host deployment_manager_SOAP_port path_to_external_cell_XD-CGB-EXPORT path_to_external_cell_overlaynodes.config_file user_name password`

This command identifies to the ODR the external cell that is represented by the XD-CGB-EXPORT and overlaynodes.config files. The XD-CGB-EXPORT file is a subset of the serverindex.xml file that isolates the required endpoints for cross-cell high availability manager communication in a runtime environment. If you examine the contents of the serverindex.xml file directly or by way of the endpoint manager for a node agent in the administrative console, the endpoint that is used by the cross cell function is P2P_CGBS_UNICAST_ADDRESS. Any firewall that is employed between the ODR cell and the back-end cell needs to allow the traffic to flow through the ports that are represented by this endpoint on each node in the ODR and back-end cells.

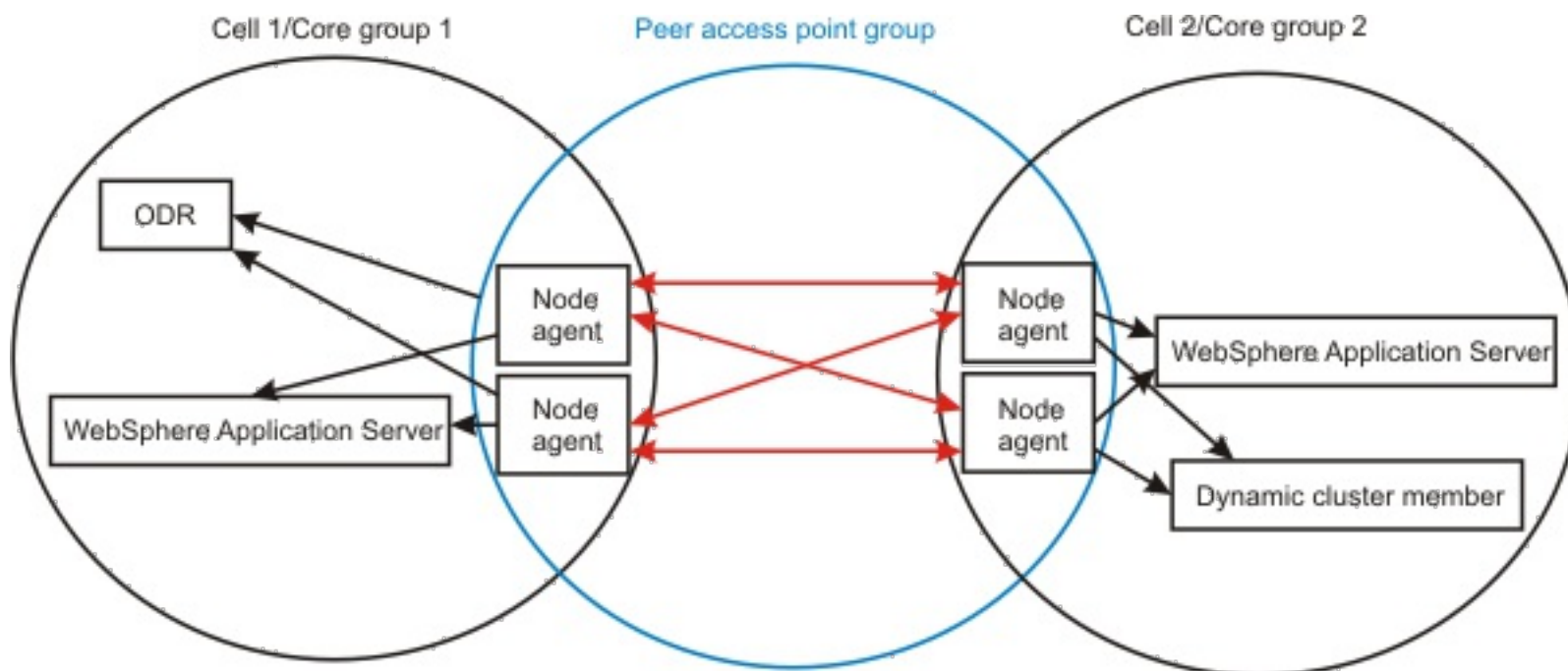
- b. Run the **crossCellCGBCfg** command on each back-end cell, against the deployment manager process of that cell, to import the XD-CGB-EXPORT and overlaynodes.config files of the ODR cell. Importing the files between two back-end cells is not supported.
6. Stop all of the processes in each cell.
 7. Restart each cell.

Results

The on demand router routes work to servers that are in different cells.

Example

Figure 1. Cross-cell communication topology diagram. This topology shows the logical division between two cells where each cell is running several servers. Each node agent runs a peer access point bridge between the cells, making the bridges part of a peer access point group.



What to do next

If you want to clear previous imports and start over, run the following command:

```
crossCellCGBCfg clear deployment_manager_host
deployment_manager_SOAP_port
```

Or, if you want the cross cell communication to use Secure Sockets Layer (SSL), set the CGBS_USE_SSL system property to a non-null value on each server in the ODR and backend cells.

Enabling communication between cells that have security enabled

When two cells have security enabled, such as Lightweight Directory Access Protocol (LDAP), and you must perform additional steps so that these cells can communicate with each other.

Enabling communication between cells that have security enabled

When two cells have security enabled, such as Lightweight Directory Access Protocol (LDAP), and you must perform additional steps so that these cells can communicate with each other.

Parent topic: [Creating and configuring ODRs](#)

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

[Core groups \(high availability domains\)](#)

Related tasks:

[Creating ODRs](#)

[Preparing the hosting environment for dynamic operations](#)

☞ [Creating a new core group](#)

Related reference:

[coregroupsplit.py script](#)

Enabling communication between cells that have security enabled

When two cells have security enabled, such as Lightweight Directory Access Protocol (LDAP), and you must perform additional steps so that these cells can communicate with each other.

Before you begin

You must be able to access the deployment manager for each cell you want to communicate.

About this task

You can add a signer certificate the trust.p12 file, allowing that cell to securely communicate with another cell. You edit the trust.p12 file at the cell level for each cell, and then run the bin/retrieveSigners.sh script in each cell. After you run the script, the cells can communicate through Secure Socket Layer (SSL) connections.

Procedure

1. Start the deployment manager for each cell.
2. On each deployment manager, edit the *deployment_manager_profile/properties/ssl.client.props* file to change the com.ibm.ssl.trustStore value to the cell-level default trust store.

For example, change the line `com.ibm.ssl.trustStore=${user.root}/etc/trust.p12` to `com.ibm.ssl.trustStore=deployment_manager_profile/config/cells/cell_name/trust.p12`.

Remember the original value. You change this value back to the original value after you run the script.

3. Run the bin/retrieveSigners.sh script from the first cell, including information for the second cell in the script.

For example:

```
retrieveSigners.sh CellDefaultTrustStore ClientDefaultTrustStore -autoAcceptBootstrapSigner -  
conntype SOAP -port 8879 -host seconddmgr.host.ibm.com
```

4. On the first cell, edit the *deployment_manager_profile/properties/ssl.client.props* file and change the value back to the original com.ibm.ssl.trustStore value.
5. On the second deployment manager, check the *deployment_manager_profile/properties/ssl.client.props* file that the com.ibm.ssl.trustStore value is the cell-level default trust store. Run the bin/retrieveSigners.sh script from the second cell, including information for the first cell.
6. On the second cell, edit the *deployment_manager_profile/properties/ssl.client.props* file to change back to the original com.ibm.ssl.trustStore value.
7. Restart all of the cells that you are configuring or ensure that all of the cells have been fully synchronized.

Results

The two cells can establish SSL connections with each other.

Parent topic: [Configuring WebSphere Virtual Enterprise for cross-cell communication](#)

Related tasks:

[Configuring WebSphere Virtual Enterprise for cross-cell communication](#)

[Configuring application placement for cells that share the same nodes](#)

[Using the retrieveSigners command to enable server to server trust](#)

[Managing nodes](#)

Configuring an ODR to dynamically update the web server plug-in configuration

In a topology with a web server which forwards requests to an on demand router (ODR), the ODR can be set to dynamically generate and update the web server plug-in configuration file `plugin-cfg.xml`. By configuring an ODR to dynamically update `plugin-cfg.xml`, you avoid manually updating the file during administrative actions, which can affect the content.

Before you begin

Consider the environment topology before configuring the `plugin-cfg.xml` generation; each ODR can generate a `plugin-cfg.xml` file which includes all URIs that it can handle. The generated `plugin-cfg.xml` file can be configured to: `all`, `cell`, `node`, and `server`. The configuration scope determines which ODRs are included in `plugin-cfg.xml`, and therefore, what ODRs a web server forwards requests to. By default, `plugin-cfg.xml` includes only running ODRs.

The `plugin-cfg.xml` configuration scope can be:

- `All`: Includes all ODRs in the environment.
- `Cell`: Includes all ODRs within the same cell as the ODR that generates `plugin-cfg.xml`.
- `Node`: Includes all ODRs on the same node as the ODR that generates `plugin-cfg.xml`.
- `Server`: Includes only the ODR that generates `plugin-cfg.xml`.

About this task

An ODR configured to generate `plugin-cfg.xml` must be running in order for dynamic updates to occur. If a single ODR is configured to generate the `plugin-cfg.xml` file, and that ODR is stopped, `plugin-cfg.xml` is not regenerated until the ODR is restarted. Therefore, run the `plugin-cfg.xml` file generator as a highly available service. For more information, read about generating the plug-in configuration in a high availability environment.

The `plugin-cfg.xml` file is generated by an ODR process. Therefore the file is generated on the file system that hosts the generating ODR. For `all`, `cell`, and `node`, `WAS_HOME/profiles/profile_name/etc/` is the default directory where the ODR generates the `plugin-cfg.xml` file. The default directory for `server` is `WAS_HOME/profiles/profile_name/etc/odr_name/`. The default generation directory can be overridden by setting a Java™ virtual machine (JVM) system property on the generating ODR. To set the JVM system property, go to **Servers > Server types > On demand routers > ODR_name > Java and process management > Process definition > Java Virtual Machine > Custom properties > New**. Name the JVM system property `ODCPluginXmlDir` and enter a value that contains the absolute path to the generation directory.

Procedure

1. In the administrative console, go to the **On demand router settings** page:
 - For stand-alone ODR processes, click **Servers > Server types > On demand routers > ODR_name > On demand router properties > On demand router settings**.
 - For a static cluster of ODR processes, click **Servers > Clusters > On demand router clusters > ODR_static_cluster_name > On demand router properties > On demand router settings**.
 - For a dynamic cluster of ODR processes, go to **Servers > Clusters > Dynamic clusters > ODR_dynamic_cluster_name > Server template > > On demand router properties > On demand router settings**.
2. In the **Proxy plugin configuration policy** section, select the scope from the **Generate plugin configuration** drop-down list.
3. Optional: In the **Proxy plugin configuration policy** section, define an absolute path to the script that you enter in the **Plugin config change script** text box. The defined script is invoked every time the `plugin-cfg.xml` is dynamically updated. This approach promotes the newly generated `plugin-cfg.xml` to a remote web server that does not reside on the same system as the ODR.
4. Click **Apply**, and save and synchronize the changes.
5. Restart the ODR for the changes to take effect. `plugin-cfg.xml` is generated every time the ODR starts, when there are administrative actions performed that impact what URIs the ODR handles, and when changes in the environment affect the content of the `plugin-cfg.xml` file.

Results

The ODR automatically updates the `plugin-cfg.xml` file when configuration changes occur.

What to do next

You can override the `plugin-cfg.xml` default attribute values automatically generated by the ODR. For more information, read about controlling the generation of the `plugin-cfg.xml` file.

[Controlling the generation of the plugin-cfg.xml file](#)

You can override the default attribute values of the `plugin-cfg.xml` file that is automatically generated by the on demand router (ODR).

[Generating the plug-in configuration in a high availability environment](#)

In a high availability environment, you can configure the web server to generate the plug-in configuration. By doing so, you ensure that the configuration is generated by one process in the cell, regardless of other processes that might not be available.

Parent topic: [Creating and configuring ODRs](#)

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Creating ODRs](#)

[Routing requests directly from a web server to a back-end application server](#)

Generating the plug-in configuration in a high availability environment

In a high availability environment, you can configure the web server to generate the plug-in configuration. By doing so, you ensure that the configuration is generated by one process in the cell, regardless of other processes that might not be available.

Before you begin

- Establish your high availability environment. For more information, read about setting up a high availability environment.
- Using the high availability plugin-cfg generation service does not automatically disable the on demand router (ODR) plugin-cfg generation, if it is enabled. The two services are independent. It is redundant to enable both services; when you enable the high availability plugin-cfg generation service, disable any previously configured ODR plugin-cfg generation. For more information, read about configuring an ODR to dynamically update the web server plug-in configuration.

About this task

Define a set of custom properties at the cell level for each plugin-cfg.xml file that you want to generate. The custom properties are formatted as `ODCPluginCfg<property>_<config>=value`, where `<property>` is one of the following elements:

- OdrList
- OutputPath
- UpdateScript
- OdrClusterList
- Disabled

The `<config>` variable is the name of the configuration that you choose. As a result, properties ending in the same `<config>` variable define a single generation definition within the high availability plug-in configuration generation service.

Procedure

1. In the administrative console, click **System administration > Cell > Custom properties > New**.
2. Define the following custom properties:

ODCPluginCfgOdrList_<configName>=cell1:node1:odr1,cell2:node2:[cell1:node3:odr3],[cell1:node4:odr4]

This property specifies the ODRs that you want to include in the plugin-cfg.xml file. Use the asterisk (*) symbol as a valid wildcard for each path segment. **odr1** and **odr2** are marked as primary servers. **odr3** and **odr4** are marked as backup servers.

ODCPluginCfgOutputPath_<configName>=/path/file_name.txt

This property specifies the location in which the plugin-cfg.xml file is placed after the file is generated. Because you can generate the plug-in configuration on any node in the cell, you must ensure the output directory exists on each node.

ODCPluginCfgUpdateScript_<configName>=/path/script <parameter1> <parameter2>

This property defines the absolute path to your script and the arguments to be passed to your defined script. The defined script will be invoked each time a plugin-cfg.xml is generated.

ODCPluginCfgOdrClusterList_<configName>=cell1:cluster1,cell1:cluster2,cell1:[cell1:cluster3],[cell1:cluster4]

This property specifies a cluster of ODRs that you want to include in the plugin-cfg.xml file. Use the asterisk (*) symbol as a valid wildcard for each path segment. **cluster1** and **cluster2** are marked as primary servers. **cluster3** and **cluster4** are marked as backup servers.

ODCPluginCfgDisabled_<configName>

This property disables the generation of a particular configuration without disabling all the properties for that configuration. The default value is `false`.

ODCPluginCfgOdrSessionIdCookie_<configName>

Defines the name of the cookie used to maintain IBM® HTTP server/ODR affinity when using ODR cell affinity.

ODCPluginCfgIHSCfgProperties_<configName>

This property is used instead of configuring JVM properties (as is done during non-HA plugin-cfg.xml generation) to set IBM HTTP server specific configuration properties. The value of this property is a comma separated list of **ATTRIBUTE_NAME=value** pairs where **ATTRIBUTE_NAME** is the name of an attribute represented in the plugin-cfg.xml.

For example, if a configuration name of 1 is being used a cell property named

ODCPluginCfgIHSCfgProperties_1 with a value of

TrustedProxyEnable=true,LogLevel=INFO,CloneSeparatorChange=true,ServerIOTimeout=60 would be created to set the **TrustedProxyEnable**, **LogLevel**, **CloneSeparatorChange**, and **ServerIOTimeout** attributes contained in the generated plugin-cfg.xml.

ODCPluginCfgTrustedProxyList_<configName>=trustedproxy1,trustedproxy2

This property is used to specify the trusted proxies that you want to include in the plugin-cfg.xml file.

See Controlling the generation of the plugin-cfg.xml file for the list of valid property names and values.

Set the following cell custom properties to generate the plug-in configuration for a collection of ODRs that are not in an ODR cluster. Note that all properties end with _1, which ties them together into a single configuration.

Table 1. Custom properties used to generate a plug-in configuration for ODRs that are not in a cluster

Property name	Property value	Description
ODCPluginCfgOdrList_1	myCell:*:*	Generates a plug-in configuration that will route to all ODRs in the myCell cell.
ODCPluginCfgOutputPath_1	/tmp/plugin-cfg1.xml	Writes the generated plug-in configuration to the /tmp/plugin-cfg1.xml file.
ODCPluginCfgUpdateScript_1	/root/bin/pluginCfgUpdate1	The path to the script that will be invoked each time the /tmp/plugin-cfg1.xml file is updated.

Set the following custom properties to generate the plug-in configuration for a cluster of ODRs named myCell/myOdrCluster. Note that all properties end with _2, which ties them together into a single configuration.

Table 2. Custom properties used to generate the plug-in configuration for myCell/myOdrCluster

Property name	Property value	Description
ODCPluginCfgOdrClusterList_2	myCell:myOdrCluster	Generates a plug-in configuration that will route to all ODRs of the myOdrCluster cluster in the myCell cell.
ODCPluginCfgOutputPath_2	/tmp/plugin-cfg2.xml	Writes the generated plug-in configuration to the /tmp/plugin-cfg2.xml file.
ODCPluginCfgUpdateScript_2	/root/bin/pluginCfgUpdate2	The path to the script that is ran each time the /tmp/plugin-cfg2.xml file is updated.
ODCPluginCfgOdrIncludeStopped_2	true or false	Includes or excludes stopped ODRs.

What to do next

Because the generation of the plugin-cfg.xml file can occur on any node in the cell, you can determine the specific location in which the generation service is running:

In the administrative console, click **Runtime operations** > **Component stability** > **Core components**. Verify that HAPPluginCfgGenerator is displayed in the table.

Parent topic: [Configuring communication between core groups that are in the same cell](#)

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Segregating HTTP traffic by ODR clusters](#)

[Configuring an ODR to dynamically update the web server plug-in configuration](#)

Segregating HTTP traffic by ODR clusters

WebSphere® Virtual Enterprise provides a service to automatically regenerate and propagate the Web server plug-in configuration file each time a change is made, which should affect how the plug-in routes. When you use this service, the Web server plug-in routes traffic by application, deployment target, or node group to a specific ODR cluster.

Before you begin

The ability to segregate HTTP traffic by ODR cluster is supported only through the use of the high availability plugin-cfg generation service. It is not supported when a specific ODR is configured to generate a plug-in configuration file.

Procedure

Set the following custom property to configure the high availability plugin-cfg generation service and direct traffic to different ODR clusters.

- a. In the administrative console, select **System administration > Cell > Custom Properties > New**.
- b. Specify the name of the custom property as `ODCPluginCfgRoutingPolicy_<config>`. The `<config>` variable is the name of the configuration that you choose. As a result, the properties end in the same variable to define a single configuration for the generation of the high availability plugin-cfg.xml file.
- c. Specify the value of the custom property as a comma-separated list of elements formatted as `<webModuleSpec>=ODR=<odrClusterName>`. You can specify the `<webModuleSpec>` portion of the value as one of the following formats:

- `WebModule:<cell>[/<application>/[<edition>/[<webModule>]]]`

Specifies one or more Web modules. Note that the brackets indicate optional elements. If any part is omitted or if the value of any part equals "*", it matches any value. For example, a custom property named `ODCPluginCfgRoutingPolicy_1` with a value of `WebModule:myCell/myApp1=ODR=odrCluster1,WebModule:myCell/myApp2=ODR=odrCluster2` routes traffic for the `myApp1` application through `odrCluster1` and traffic for the `myApp2` application through `odrCluster2`.

- `Cluster:<cell>/<cluster>`

Specifies Web modules that are deployed to the cluster. For example, a custom property named `ODCPluginCfgRoutingPolicy_1` with a value of `Cluster:myCell/myCluster1=ODR=odrCluster1,Cluster:myCell/myCluster2=ODR=odrCluster2` routes traffic targeted for any application, which is deployed to `myCluster1`, through `odrCluster1`. Similarly, traffic targeted for any application that is deployed to `myCluster2` is routed through `odrCluster2`.

- `NodeGroup:<cell>/<nodeGroup>`

Specifies Web modules that are deployed to one or more nodes in the node group. For example, a custom property named `ODCPluginCfgRoutingPolicy_1` with a value of `NodeGroup:myCell/myNodeGroup1=ODR=odrCluster1,Cluster:myCell/myNodeGroup2=ODR=odrCluster2` routes traffic targeted for any application, which is deployed to a node in `nodeGroup1`, through `odrCluster1`. Traffic targeted for any application that is deployed to a node in `nodeGroup2` is routed through `odrCluster2`.

- `VirtualHost:<cell>[/<VirtualHost>]`

Specifies all of the Web modules that are deployed to the virtual host. For example, a custom property named `ODCPluginCfgRoutingPolicy_1` with a value of `VirtualHost:myCell/default_host=ODR=odrCluster1,VirtualHost:myCell/special_host=ODR=odrCluster2` routes traffic for applications that are deployed to `default_host` through `odrCluster1`. Traffic targeted for applications that are deployed to `special_host` is routed through `odrCluster2`.

You can specify the `<odrClusterName>` variable as a simple cluster name, or you can format it as `<cell>/<cluster>` to reference an ODR cluster in a different cell from which the service is running. In general, the value is read as meaning all traffic that matches the `<webModuleSpec>` variable is routed through the ODR cluster named `<odrClusterName>`.

Parent topic: [Creating and configuring ODRs](#)

Related tasks:

[Generating the plug-in configuration in a high availability environment](#)

Controlling the generation of the plugin-cfg.xml file

You can override the default attribute values of the plugin-cfg.xml file that is automatically generated by the on demand router (ODR).

The following elements are formatted as `com.ibm.ws.odr.plugincfg.<element>.<attributeName>`, where `<element>` is the name of the configuration file keyword that denotes the element that contains an attribute, and `<attributeName>` is the name of the attribute that is located within that element. Configure the elements as JVM system properties. To set a JVM system property on the ODR in the administrative console, click **Servers > On Demand Routers > on_demand_router > Java™ and Process Management > Process Definition > Java Virtual Machine > Custom Properties > New**.

Config

This element starts the plug-in configuration file. It can include one or more of the following elements and attributes.

com.ibm.ws.odr.plugincfg.config.ASDisableNagle

Specifies whether the user wants to disable Nagle's algorithm for the connection between the plug-in and the server.

The default value is false.

com.ibm.ws.odr.plugincfg.config.AcceptAllContent

Specifies whether or not you can include content in POST, PUT, GET, and HEAD requests when a Content-Length or Transfer-encoding header is contained in the request header. You can specify one of the following values for this attribute:

- True if content is to be expected and read for all requests
- False if content only is only to be expected and read for POST and PUT requests.

The default value is false.

com.ibm.ws.odr.plugincfg.config.AppServerPortPreference

Specifies which port number is used to build URI's for a sendRedirect.

The default value is HostHeader.

com.ibm.ws.odr.plugincfg.config.ChunkedResponse

Specifies whether the plug-in groups the response to the client when a Transfer-Encoding : Chunked response header is present in the response.

You can specify one of the following values for this attribute:

- True if the plug-in is to chunk the response to the client when a Transfer-Encoding : Chunked response header is present in the response.
- False if the response is not to be chunked.

The default value is false.

com.ibm.ws.odr.plugincfg.config.IISDisableNagle

Specifies if you want to disable nagle algorithm.

The default value is false.

com.ibm.ws.odr.plugincfg.config.IISPluginPriority

Specifies the priority in which the Web server loads the plug-in. You can specify one of the following values for this attribute:

- High
- Medium
- Low

The default value is High.

com.ibm.ws.odr.plugincfg.config.IgnoreDNSFailures

Specifies if the plug-in ignores DNS failures within a configuration when started. When set to true, the plug-in ignores DNS failures within a configuration and starts successfully if at least one server in each ServerCluster resolves the host name. Any server for which the host name is not resolved is marked *unavailable* for the life of the configuration. The host name is not resolved later during the routing of requests. If a DNS failure occurs, a log message is written to the plug-in log file and the plug-in continues initializing instead of the Web server not starting.

The default value is false.

com.ibm.ws.odr.plugincfg.config.RefreshInterval

The time interval, in seconds, at which the plug-in should check the configuration file for updates or changes. The plug-in checks the file for any modifications that occur since the plug-in configuration was loaded.

The default value is 60. In a development environment in which changes are frequent, set the time interval to less than 60 seconds. In production, set a higher value than the default, because updates to the configuration do not occur as often. If the plug-in reload is not successful, the plug-in log file contains a message, and the previous configuration is used until the plug-in configuration file successfully reloads. Refer to the plug-in log file for more information if an error occurs.

com.ibm.ws.odr.plugincfg.config.ResponseChunkSize

The plug-in file reads the response body in 64k chunks until all of the response data is read, which causes a performance problem for requests whose response body contains large amounts of data.

Use this attribute to specify the maximum chunk size to use when reading the response body. For example, `Config ResponseChunkSize="N">`, where N equals the chunk size in kilobytes.

If the content length of the response body is unknown, a buffer size of N kilobytes is allocated and the body is read in N kilobyte size chunks, until the entire body is read. If the content length is known, then a buffer size of either content length or N is used to read the response body.

The default chunk size is 64k.

com.ibm.ws.odr.plugincfg.config.VHostMatchingCompat

Specifies to use the port number for virtual host matching. The following values can be specified:

- True for physically matching by using the port number for which the request is received.
- False for logically matching by using the port number contained in the host header.

The default is false.

com.ibm.ws.odr.plugincfg.odrIncludeStopped

Specifies to allow the inclusion of stopped ODRs. The following values can be specified:

- True for including stopped ODRs.
- False for not including stopped ODRs.

The default is false.

com.ibm.ws.odr.plugincfg.config.TrustedProxyEnable

Specifies the trusted proxies. The following values can be specified:

- True for including trusted proxies.
- False for not including trusted proxies.

The default is false. The trusted proxies are collected from the defined trusted security proxies on the **On Demand Router Properties > On Demand Router settings** administrative console page.

Log

The log describes the location and level of log messages that are written by the plug-in. If a log is not specified within the configuration file, the log messages might be written to the error log.

com.ibm.ws.odr.plugincfg.log.Name

The fully qualified path to the log file to which the plug-in writes error messages.

The default value is `profileRoot/logs/http_plugin.log`.

com.ibm.ws.odr.plugincfg.log.LogLevel

The level of detail of the log messages that the plug-in writes to the log. You can specify one of the following values for this attribute:

- Trace. All of the steps in the request process are logged in detail.
- Stats. The server selected for each request and other load balancing information relating to request handling is logged.
- Warn. All warning and error messages resulting from abnormal request processing are logged.
- Error. Only error messages resulting from abnormal request processing are logged.
- Debug. All of the critical steps performed in processing requests are logged.
- Detail. All of the information about requests and responses are logged.

The default value is Error.

CAUTION:

A lot of messages are logged at the trace level, which can cause the file system to fill up very quickly. Never use a trace setting in a normally functioning environment as it adversely affects performance.

ServerCluster

This element is a group of servers that are generally configured to service the same type of requests.

In the simplest case, the cluster contains only one server definition. When you define more than one server, the plug-in completes a load balance across the defined servers by using either a Round Robin or a Random algorithm. The default is Round Robin.

com.ibm.ws.odr.pluginfg.cluster.CloneSeparatorChange

Some pervasive devices cannot handle the colon character (:) that is used to separate clone IDs in conjunction with session affinity. This attribute tells the plug-in to expect the plus character (+) as the clone separator. You must change the server configurations so that the server separates clone IDs with the plus character as well.

The default value is false.

com.ibm.ws.odr.pluginfg.cluster.LoadBalance

The Round Robin implementation has a random starting point. The first server is selected randomly, and the Round Robin value is used to select servers from that point forward. This implementation ensures that in multiple process-based Web servers, all of the processes do not start by sending the first request to the same server.

The default load balancing type is Round Robin.

com.ibm.ws.odr.pluginfg.cluster.PostSizeLimit

The maximum number of bytes of request content allowed for the plug-in to attempt to send the request to a server. If a request is received that is greater than this size, the plug-in ends the request.

The default value is -1 bytes, which indicates that there is no limit for the post size.

com.ibm.ws.odr.pluginfg.cluster.RemoveSpecialHeaders

The plug-in adds special headers to the request before it is forwarded to the server. These headers store information about the request that the application then uses. By default, the plug-in removes these headers from incoming requests before adding the required headers.

The default value is true. If you set the attribute to false, you introduce a potential security exposure by not removing headers from incoming requests.

com.ibm.ws.odr.pluginfg.cluster.RetryInterval

An integer that specifies the length of time that elapses from the time a server is marked down to the time that the plug-in reattempts to make a connection.

The default value is 60 seconds.

Server

This element is a server instance that is configured to handle requests routed to it based on the routing rules of the plug-in configuration. The server element corresponds to an application server running on either the local workstation or a remote workstation.

com.ibm.ws.odr.pluginfg.server.ServerIOTimeout

Enables the plug-in to set a timeout value, in seconds, for sending requests to and reading responses from the server.

If a value is not set, the plug-in, by default, uses blocked I/O to write request to and read response from the server until the TCP connection times out. When selecting a value for this attribute, remember that it might take several minutes for a server to process a request. Setting the value of the ServerIOTimeout attribute too low might cause the plug-in to send a false server error response to the client.

The default value is 60.

For more information on how to configure the ServerIOTimeout attribute, see the [web server plug-in configuration technote](#).

com.ibm.ws.odr.pluginfg.server.ConnectTimeout

Enables the plug-in to perform non-blocking connections with the application server, which are beneficial when the plug-in is unable to contact the destination to determine if the port is available or unavailable.

If no value is specified, the plug-in performs a blocking connect in which the plug-in waits until an operating system times out and allows the plug-in to mark the server *unavailable*. A value greater than 0 specifies the number of seconds the plug-in waits for a successful connection. If a connection does not occur after that time interval, the plug-in marks the server *unavailable*, and proceeds with one of the other servers defined in the cluster.

The default value is 0.

com.ibm.ws.odr.pluginfg.server.ExtendedHandShake

Used when a proxy firewall is between the plug-in and the application server. In such a case, the plug-in is not failing over, as expected.

The plug-in marks a server as stopped when the connect() ends. However, when a proxy firewall is in between the plug-in and the application server, the connect() succeeds, even though the back-end application server is stopped. This causes the plug-in to not failover correctly to other application servers.

The plug-in contains this attribute to ensure that it is started before sending the request.

The default value is false.

com.ibm.ws.odr.pluginfg.server.MaxConnections

Specifies the maximum number of pending connections to a server that flows through a Web server process at any point in time.

The default value is set to -1.

com.ibm.ws.odr.pluginfg.cluster.WaitForContinue

Specifies whether to use the HTTP 1.1 100 Continue support before sending the request content to the application server. The default value is false. The plug-in does not wait for the 100 Continue response from the application server before sending the request content.

This property is ignored for POST requests to prevent a failure from occurring if the application server closes a connection because of a time-out.

Enable this function when configuring the plug-in to work with certain types of proxy firewalls.

Property

com.ibm.ws.odr.pluginfg.property.ESIEnable

Used to enable or disable the Edge Side Include (ESI) processor. If the ESI processor is disabled, the other ESI elements in the file are ignored.

The default value is true.

com.ibm.ws.odr.pluginfg.property.ESIMaxCacheSize

An integer specifying, in 1K byte units, the maximum size of the cache. The default maximum size of the cache is 1024K bytes (1 megabyte). If the cache is full, the first entry to be evicted from the cache is the entry that is closest to its expiration time.

com.ibm.ws.odr.pluginfg.property.ESIInvalidationMonitor

Indicates whether or not the ESI processor receives invalidations from the application server.

The default value is false.

com.ibm.ws.odr.pluginfg.property.https.keyring

Use this element to specify the initialization parameters when the protocol of the transport is set to HTTPS.

This attribute is a supported name that is recognized by the transport. The default value for the keyring element is *profileRoot/etc/plugin-key.kdb*.

com.ibm.ws.odr.pluginfg.property.https.stashfile

The default value for the stash file is *profileRoot/node/etc/plugin-key.sth*

com.ibm.ws.odr.pluginfg.property.PluginInstallRoot

Specifies the installation path for the plug-in. The default value is "". Set the value, however, to the fully qualified path of the plug-in installation root, or the property is not displayed in the plugin-cfg.xml file.

Parent topic: [Configuring an ODR to dynamically update the web server plug-in configuration](#)

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Creating and configuring ODRs](#)

[Routing requests directly from a web server to a back-end application server](#)

Propagating the plugin-cfg.xml file to web servers

For web servers that are defined in the WebSphere® Application Server Network Deployment (ND) configuration, you can automatically propagate the generated plugin-cfg.xml file to web servers using the **autoPropagate** keyword option, or by using the **propagatePluginCfg.py** script.

About this task

- **autoPropagate** keyword option:
 - The format of this option is `autoPropagate,<cellName:><nodeName:><webServerName>`.
 - More than one `<cellName:><nodeName:><webServerName>` combination can be defined using a comma as the delimiter.
 - Each field `<cellName:><nodeName:><webServerName>` can use an asterisk for a wildcard. For example:

```
autoPropagate,*:*:*
```

```
autoPropagate,cellA:nodeA:webServerA,cellA:nodeB:webServerB
```

- **propagatePluginCfg.py** script:

The script is available in the `<WAS_HOME>/bin/propagatePluginCfg.py` directory. To learn more about this script, read about **propagatePluginCfg.py**.

To propagate the generated plugin-cfg.xml file using the **autoPropagate** keyword option, or the **propagatePluginCfg.py** script, follow this procedure:

Procedure

1. In the administrative console, go to the **On demand router settings** page:
 - For stand-alone on demand router (ODR) processes, click **Servers > Server types > On demand routers > ODR_name > On demand router properties > On demand router settings**.
 - For a static cluster of ODR processes, click **Servers > Clusters > On demand router clusters > ODR_static_cluster_name > On demand router properties > On demand router settings**.
 - For a dynamic cluster of ODR processes, go to **Servers > Clusters > Dynamic clusters > ODR_dynamic_cluster_name > Server template > On demand router properties > On demand router settings**.
2. In the **Proxy plugin configuration policy** section, select the scope from the **Generate plugin configuration** drop-down list.
3. In the **Proxy plugin configuration policy** section, define an absolute path to the script that you enter in the **Plugin config change script** text box. The defined script is invoked every time the plugin-cfg.xml is dynamically updated.
 - Enter the **autoPropagate** keyword in the **Plugin config change script** text box to automatically propagate the generated plugin-cfg.xml file.

Example:

```
▪ autoPropagate,*:*:*
```

```
▪ /tmp/executeFirst.sh;/bin/executeSecond.sh;autoPropagate,*:*:*
```

- Enter the **propagatePluginCfg.py** script in the **Plugin config change script** text box to automatically propagate the generated plugin-cfg.xml file.

Example:

```
▪ /opt/WAS/bin/wsadmin.sh -f /opt/WAS/bin/propagatePluginCfg.py -  
lang jython https://myIHSHost:8192  
ihsusr ihsusrPassword /tmp/plugin-cfg.xml  
/opt/HTTPServer/plugins/config/webserver1/plugin-cfg.xml
```

```
▪ /tmp/executeFirst.sh;/bin/executeSecond.sh;/opt/WAS/bin/wsadmin.  
sh -f /opt/WAS/bin/propagatePluginCfg.py  
-lang jython https://myIHSHost:8192 ihsusr ihsusrPassword  
/tmp/plugin-cfg.xml  
/opt/HTTPServer/plugins/config/webserver1/plugin-cfg.xml
```

4. Click **Apply**, and save and synchronize the changes.
5. Restart the ODR for the changes to take effect. plugin-cfg.xml is generated every time the ODR starts, when there are administrative actions performed that impact what URIs the ODR handles, and when changes in the environment affect the content of the plugin-cfg.xml file.

Related tasks:

[Configuring an ODR to dynamically update the web server plug-in configuration](#)

[Generating the plug-in configuration in a high availability environment](#)

Related information:

[propagatePluginCfg.py](#)

Routing requests directly from a web server to a back-end application server

Define a new custom property to configure the web server plug-in configuration to route requests directly from the web server to a back-end application server. You can configure certain requests to route through an on demand router (ODR) and configure other requests to route directly to a back-end application server.

About this task

Typically, an ODR acts as a gateway through which requests sent by a web server flow to a back-end application server. You can define the `ODR_Module_Routing_Policy` custom property for the `plugin-cfg.xml` file to redirect specific requests that you do not want to route through the ODR. Instead, the requests are routed directly from the web server to the back-end server. Alternatively, you can reset the custom property so that the ODR resumes intercepting requests.

The format of the custom property value is a comma-separated list of module paths, such as `cell_name/app_name/edition/module_name=value`.

Procedure

1. In the administrative console, click **System administration > Cell > Custom Properties > New**.
2. Type `ODR_Module_Routing_Policy` as the name of the custom property.
3. Type the value of the custom property.
 - Set the value to `cell_name/app_name/edition/module_name=direct` to route requests directly to the back-end server.
 - Set the value to `cell_name/app_name/edition/module_name=ODR` to route requests through the ODR before the back-end server receives the requests.

For example, if you set the value to `cell/app/edition/module=direct,cell/app2/edition/module=ODR`, each module is configured independently as to whether requests for that module are sent through the ODR or directly to the back-end server. You can use a wildcard (*) in place of the `app_name`, `edition`, and `module_name` variables.
4. Click **Apply** and save your changes.

Example

In the following example, the custom property is set to route requests to a back-end application server. A wildcard is used in place of the `app_name`, `edition`, and `module_name` variables.

```
myCell/*/*/*=direct
```

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring an ODR to dynamically update the web server plug-in configuration](#)

Related information:

[Controlling the generation of the plugin-cfg.xml file](#)

Configuring passive HTTP session affinity in the on demand router

When the on demand router (ODR) processes a request, it obtains the session affinity descriptor policy of the cluster to which the server belongs. If you changed the default settings for any middleware servers, you might need to update the middleware descriptor properties so that the ODR can obtain the descriptor policy. In most cases, the on demand router (ODR) does not require configuration to support HTTP session affinity. However, some special cases exist where you must configure the ODR to learn about backup servers that the back end servers might be setting on the session affinity cookie.

Before you begin

The servers in your configuration must be in a generic server cluster or a dynamic cluster. You can use passive HTTP session affinity with a static cluster.

About this task

In environments where the ODR forwards requests to generic server cluster members and to non-federated WebSphere® Application Server servers, a set of properties must be set for the ODR to correctly uphold session affinity. Passive HTTP session affinity means that the ODR passes the session cookie set by the backend server through to the client, as opposed to the ODR setting the WSJSESSIONID cookie. Passive HTTP session affinity is used in the following situations:

- When the ODR routes to servers that are not running WebSphere Application Server middleware products.
- When the ODR routes to WebSphere Application Server application servers that are in different core groups that are not connected by the core group bridge.
- When the application uses Java™ Platform, Enterprise Edition (Java EE) HTTP session affinity that is not standard. For example, the application's session ID cookie name is something other than JSESSIONID

Procedure

1. If any of the default values for the server have changed, modify the session affinity descriptor. In the administrative console, click **System administration > Middleware descriptors > middleware_server > default**.
2. Define the session affinity descriptor properties. Modify the values for any of the following fields that apply:
 - **Learn clone IDs**
 - **Cookie names**
 - **URL rewrite**
 - **Clone ID separator**
 - **Alternate clone ID separator**
 - **Affinity mode**

Set the value of the **Learn clone IDs** field to true so that the ODR parses the clone IDs from the response cookie that are sent back to the client. Because the ODR recognizes the server that sends the response back at this point, the parsed clone ID is associated with the server. Therefore, future requests are matched against the known set of clone IDs in order to uphold session affinity in other middleware server environments. Set the **Learn clone IDs** field to true when the ODR does not have an on demand configuration of the server. Note that the ODR can only parse the response cookie if the session ID is in a JSESSIONID format that the ODR understands.

The **Cookie names** field indicates which response header contains session ID information, and should be parsed to determine the clone ID. The **Clone ID separator** field indicates on which part of the session cookie the **Clone ID** field begins. The **Cookie names** and **Clone ID separator** fields are also used by the ODR to parse the clone IDs from the request cookie to enforce session affinity.

In cases where there is no on demand configuration information for servers, such as servers that are members of generic server clusters, set the **Learn clone IDs** field to true so that the ODR parses the session ID for the clone ID. If the session ID in the response is not in the JSESSIONID format, you must set the affinity mode to Active[-conditional] affinity. In this case, the ODR internally assigns each backend application server a clone ID, which is set in the WSJSESSIONID header. As a result, the ODR maintains session affinity when operating with backend environments that cannot generate session IDs in the JSESSIONID format. Active affinity means that the ODR always sets a WSJSESSIONID cookie with the clone ID of the backend server that is sending the response. Active-conditional affinity means that the ODR sets only the WSJSESSIONID cookie if it recognizes the Set-Cookie header in the response.

In WebSphere Application Server environments where the clone IDs are available to the ODR by way of the on demand configuration, the clone ID information is never learned by setting the **Learn clone IDs** field to true. The clone IDs are available to the ODR by way of the on demand configuration, if the ODR is in the same core group as the application servers, if the ODR is in a different core group but the core groups are bridged, or if the bulletin board service overly network (BBSON) is enabled. Note that BBSON is enabled by default in WebSphere Virtual Enterprise Version 7.0 and later. Setting the **Learn clone IDs** to true or false does not have an impact in these environments.

Note: In releases of WebSphere Application Server Version 7.0x prior to Version 7.0.0.17, applying APAR PM28768 can prevent the on demand configuration component from gathering the clone IDs correctly. In WebSphere Virtual Enterprise environments, you can bypass the defect by augmenting the middleware

server with WebSphere Virtual Enterprise.

Results

When the ODR processes a request, it obtains the session affinity descriptor policy that is configured for the cluster to which the server belongs. The method in which the server clone identification is obtained depends on the property values of the policy attributes.

Parent topic: [Creating and configuring ODRs](#)

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

[BBSON bulletin board](#)

Related tasks:

[Routing requests to external nodes with generic server clusters](#)

Configuring the on demand router for multi-cluster failover and load balancing routing

To configure the on demand router (ODR) to route requests to a different cluster, you can configure custom properties for configuring multi-cluster failover and load balancing routing policies. You might route requests to a cluster in another cell if your primary cluster fails, to balance the load in your environment between multiple clusters, or to direct the ODR to route requests to a specific cluster.

Before you begin

- Create the ODR for your cell. See [Creating ODRs](#).
- Configure and deploy the clusters and applications for your multi-cluster policy.
- If the clusters are in different cells, configure the core group bridge so that they can communicate, or use a generic server cluster definition.

About this task

Use multi-cluster routing policies for failover and load balancing. With *multi-cluster failover*, you can specify a cluster to take over the workload when the primary cluster fails. With *load balancing routing*, you can balance the request loads between multiple clusters.

Procedure

1. Create a custom property for the multi-cluster routing policy. In the administrative console, click **Servers > On demand routers > ODR_name > On demand router properties > On demand router settings > Custom properties > New**.
2. Type a name for the multi-cluster routing policy in the **Name** field. The name must start with the token MCRP@ string. The full syntax of the name field follows:

```
MCRP@cell_name[$application_name[$web_module_name[$cluster_name]]]
```

Note: The recommendation for configuring multi-cluster failover and load balancing routing policies is through the wsadmin tasks, unless you already have it configured through a custom property. See [Rules for ODR routing policy administrative tasks](#).

Table 1. Components of the name field syntax

Option	Description
MCRP	Specifies that the custom property is a multi-cluster routing policy (MCRP). This prefix must be specified in uppercase letters.
@	Required symbol. This symbol is the separator between the policy name and the cell. In this configuration, it is generally used to separate a policy name attribute from a cell name.
cell_name	Specifies the name of the cell. This cell must be a valid cell that runs WebSphere Application Server. The case and spelling must match the WebSphere Application Server configuration.
\$	Separates the WebSphere Application Server objects.
application_name	Specifies the application name without the file extension. For example, if the enterprise application name is StockTrade.ear, then specify StockTrade as the application_name value.
web_module_name	Specifies the name of the Web module without the .war file extension.
cluster_name	Specifies the name of the cluster in which the application is deployed.
[]	Indicates variables that are optional.

3. Type a value in the **Value** field. The full syntax of the value field follows:

```
policy_type@cell_name1$cluster_name1[,cell_name2$cluster_name2,...]
```

Table 2. Components of the value field syntax

Option	Description
policy	The policy_type value is not case sensitive. The failover, wlor, or wrr values can be specified in uppercase or lowercase.
i	Valid values:

<p>c y - t y p e</p>	<p>failover: When a request for the application Web module in the cell that is specified in the Name field fails, the request fails over to the cell and cluster that are specified in the Value field after the @ symbol. Requests route only to the configured cell and cluster when the primary cell is down. The cell status is indicated by an HTTP status code of 503, <i>service unavailable</i>.</p> <p>wlor: Specifies a weighted least outstanding request load balancing policy. This policy comes into effect when the ODR is active and reads its custom property configuration. This load balancing policy not only considers weights, but also how many outstanding HTTP requests exist in a cluster. This policy will more efficiently distribute requests to clusters that can handle them. Wlor is recommended over wrr.</p> <p>New weight values are obtained every 15 seconds from the dynamic workload manager (DWLM), which takes into account the application level response time. Use the mcrp.ui system property to set the new update time in seconds.</p> <p>wrr: Specifies a weighted round-robin load balancing policy. This policy comes into effect when the ODR is active and reads its custom property configuration. .</p> <p>New weight values are obtained every 15 seconds from the dynamic workload manager (DWLM), which takes into account the application level response time. Use the mcrp.ui system property to set the new update time in seconds</p>
<p>c e l l - n a m e</p>	<p>Specifies the name of the cell. This cell must be a valid cell that runs WebSphere Application Server. The case and spelling must match the cell name in WebSphere Application Server.</p>
<p>c l u s t e r - n a m e</p>	<p>The cluster names can be names of clusters or dynamic clusters in the local cell, a cluster that is in a cell that is bridged with the core group bridge service, or a generic server cluster.</p> <p>The cluster name value must be capitalized and spelled the same way that you specified the name when you created the cluster in the administrative console.</p>
<p>,</p>	<p>The comma (,) is used to separate a set of values in the list.</p>

All Java™ Platform, Enterprise Edition (Java EE) artifact names such as *cell_name*, *application_name*, and *cluster_name* must be spelled the way that they were spelled in the WebSphere Application Server configuration.

The *cell_name* and *cluster_name* values in the **Name** or **Value** field can be a wildcard (*). If you use the wildcard in place of a cell name, all the cells in the cell group are indicated. A cell group is defined by any cells that are bridged together with the core group bridge. If you use the wildcard in place of the *cluster_name* value, all of the clusters in a given cell are indicated. Using a wildcard value is only relevant when you are using multi-cluster load balancing routing.

Examples for the value field follow:

The following policy configures a failover policy. When a failure occurs, the requests can fail over to the myGSC1 generic server cluster in the thesaharaCell01 cell:

```
failover@myCell01$myCluster1,myCell01$myGSC1
```

The following policy configures a weighted least outstanding request load balancing policy.

```
wlor@thesaharaCell01$myCluster1,myCell2$myCluster2
```

The following policy configures a weighted round robin policy.

```
wrr@thesaharaCell01$myNYCGSC,cell_2$cluster_2
```

The following value balances load across all of the cell and cluster combinations where the configured application is deployed.

```
wrr@*$*
```

4. Click **Apply** or **OK** to commit your new custom settings.

Results

The ODR routes to multiple clusters, as you configured in the multi-cluster routing policy.

Previous topic: [Creating ODRs](#)

Parent topic: [Creating and configuring ODRs](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring WebSphere Virtual Enterprise for cross-cell communication](#)

Related reference:

[createodr.jacl script](#)

Related information:

 [Proxy server settings](#)

Configuring ODR OutOfMemory prevention

You can create cell-level custom properties to prevent the on demand router (ODR) from running out of memory when the number of concurrent requests sent to the ODR increases.

About this task

Because the ODR is completely asynchronous, it can scale to an extremely high number of concurrent connections. If application server threads begin to slow down or hang due to some condition, such as a slow database, the number of concurrent requests through a single ODR can increase dramatically. To prevent the ODR from running out of memory due to the large number of connections that can accrue in the ODR in this scenario, the ODR automatically starts rejecting requests, and thus closing connections, when the heap utilization exceeds a maximum threshold.

Procedure

1. Set the following custom property to configure a percentage value that determines at what amount of heap usage the ODR rejects requests. If heap usage exceeds 90%, the ODR rejects any incoming request and a 503 error code is returned.
 - a. In the administrative console, select **System administration > Cell > Custom Properties > New**.
 - b. Specify the name of the custom property as `ODR.heapUsage.max`.
 - c. Specify a value for the custom property. The default value is 90. To disable this feature, set the value to 100.
2. Set the following custom property to configure the type of error code that is returned when a request is received and the heap usage exceeds the maximum threshold.
 - a. In the administrative console, select **System administration > Cell > Custom Properties > New**.
 - b. Specify the name of the custom property as `ODR.heapUsage.errorCode`.
 - c. Specify a value for the custom property. The default value is 503.

Parent topic: [Creating and configuring ODRs](#)

Creating dynamic clusters

When you create a dynamic cluster, weights are used to dynamically balance the workloads of the cluster members, based on performance information that is collected from the cluster members. Creating dynamic clusters enables application server virtualization.

Before you begin

- You must have configurator administrative privileges to create a dynamic cluster.
- Verify that the application placement controller is enabled. The application placement controller is enabled by default. The application placement controller enables the autonomic capabilities of dynamic clusters. To enable the application placement controller, click **Operational policies > Autonomic managers > Application placement controller**. For more information about the properties that you can change on the application placement controller, see [Monitoring and tuning the application placement controller](#)
- Decide if you are going to use *vertical stacking*. Vertical stacking can improve bottleneck conditions in deployed applications by enabling the placement controller to start more than one instance of the dynamic cluster on a node. With vertical stacking enabled, the autonomic managers limit the processor percentage that is used by each stacked instance. The general formula is $100\% / \text{max-number-of-stacked-instances}$. For example, if you configure three stacked instances, the workload is throttled to prevent any single instance from using more than 33% of the processor capacity. See [Configuring vertical stacking](#) for more information.
- If you are creating a dynamic cluster of externally created middleware servers with assisted life-cycle management, create representations of these servers in the product environment before you create a dynamic cluster. All of these servers must have the same applications installed and have the same version of middleware software installed. See [Adding assisted life-cycle middleware servers](#) for more information.

About this task

A *dynamic cluster* is an application deployment target that can expand and contract depending on the workload in your environment. Dynamic clusters work with autonomic managers, including the application placement controller and the dynamic workload manager to maximize the use of your computing resources. Dynamic clusters are required for many of the product autonomic functions, including high availability and service policies.

In Version 6.1.1 only, when you create a dynamic cluster of application servers, the recommendation is to have a maximum number of 50 application servers, node agents, and on demand routers per core group. A dynamic cluster cannot span multiple core groups. Consider this limit when you set the number of instances and stacking number for your dynamic cluster. This limitation does not apply to dynamic clusters of assisted life-cycle servers or complete life-cycle dynamic clusters of PHP or WebSphere® Application Server Community Edition servers.

If you already have over 40 servers in your core group, you can use the `coregroupsplit.py` script to split your existing cell into multiple core groups. See [coregroupsplit.py script](#) for more information.

Procedure

1. Create the dynamic cluster. In the administrative console, click **Servers > Dynamic clusters > New**.
2. Select the dynamic cluster server type. The dynamic cluster server type determines the type of servers that are members of this dynamic cluster. Depending on the type, you enter the name of the dynamic cluster on this panel or when you select the membership method.
3. For some dynamic cluster types, you can select the membership method. The membership method defines how servers join the dynamic cluster as cluster instances.

Option	Description
Automatically define cluster members with rules	You can Automatically define cluster members with rules if you are using servers with complete life-cycle management. With this option, you create a membership policy that defines the nodes on which cluster instances can be placed.
Manually define cluster members	If you are using servers with assisted life-cycle management, you can Manually define cluster members . With this option, you select existing servers to add to the dynamic cluster. The servers that you select must be homogenous: that is, they must be of the same server type, middleware server version, and have the same applications installed. Note: To add a new middleware server to an existing dynamic cluster when one or more applications are targeted to the dynamic cluster, you must install the applications on the middleware server and target the applications to the server before you add the server as member of the dynamic cluster.

4. Define the dynamic cluster members.

- If you selected **Automatically define cluster members with rules** in the previous step, use the subexpression builder to build a membership policy expression for your dynamic cluster. This expression is compared to all the nodes in the cell, selecting any nodes for which the subexpression is true.
- If you selected **Manually define cluster members**, the action you take depends on what kind of servers are in your dynamic cluster.
 - If your server type is application server, choose an existing static cluster to convert to a dynamic cluster.
 - If your server type is an externally created middleware server with assisted life-cycle management, select servers from the list, and add them to your dynamic cluster. Before you add assisted life-cycle servers, be sure that they are the same type, middleware server version, and have the same set of applications installed.

Note: Before you can add a new middleware server to an existing dynamic cluster when one or more applications are targeted to the dynamic cluster, you must install the applications on the middleware server and target the applications to the server.

- a. Deploy your unmanaged application to the middleware server. See [Deploying unmanaged Web applications](#) for more information.
- b. Define the deployment target for the application. In the administrative console, click **Applications > All applications > unmanaged_app_name**. Select the target and click **Add**.
- c. Click Apply and save your changes.

5. Select a dynamic cluster server template. You can select a dynamic cluster server template only for dynamic clusters that consist of servers with complete life-cycle management. You can choose an existing predefined template, or create your own server templates to use when you create your dynamic cluster. See [Creating middleware server templates](#) for more information about creating server templates.

Note: If your configuration consists of mixed versions of WebSphere Application Server Network Deployment, specifically the deployment manager is at a higher version than the version of the node, you cannot use a predefined server template to create a dynamic cluster. If you are running a Version 6.1 node and a Version 7.0 deployment manager, for example, you can create a static cluster of servers on the node, and then convert the static cluster to a dynamic cluster.

6. Specify dynamic cluster-specific properties.
 - a. Define the minimum number of cluster instances. The default minimum number of instances is one instance and the maximum default is no limit on instances. If a minimum value is excessive, performance degradation might occur.
 - b. Define the maximum number of cluster instances. The default value has no limit on the number of cluster instances.
 - c. Determine whether to enable vertical stacking. When you configure vertical stacking, more than one dynamic cluster instance can start on the same node.
 - d. Specify an isolation preference for the dynamic cluster.
7. Confirm the dynamic cluster creation. Click **Finish > Save** to save the changes to the master configuration.
8. Select the mode of operation. In the administrative console, click **Servers > Dynamic clusters**. Select the dynamic clusters that you want to modify. Choose the operational mode and click **Set mode**.

Important: To use dynamic application placement, click **Automatic** or **Supervised** as the mode of operation.

z/OS If you have nodes that run on z/OS® systems, use dynamic clusters in supervised or automatic mode. If you want to prevent WebSphere Virtual Enterprise from automatically starting a cluster member on a logical partition (LPAR) that is hosting a dynamic cluster when the LPAR processor is too busy, then you must define the `cpuUtilizationThreshold` custom property. In this scenario, the product starts the cluster member only if transaction demand requires additional cluster members and the processor utilization on the LPAR is less than the `cpuUtilizationThreshold` value. See [cpuUtilizationThreshold](#) for more information.

Results

When the dynamic clusters start, at least one instance of each dynamic cluster in your environment becomes available as soon as possible. Multiple instances on the same node can start concurrently if you have multiple processors on the same node. For example, if you have two processors on a node, two instances can start concurrently. The application placement controller continues to start instances evenly across the nodes for all the dynamic clusters until the minimum number of instances for each dynamic cluster is reached.

Example

Use the following placement scenarios as a guideline for your dynamic cluster settings:

- To specify that exactly two servers are started when the dynamic cluster is running: Click **Keep multiple instances started at all times**. Set the **Number of instances** value to 2. Click **Limit the number of instances that can start**, and set the **Number of instances** value to 2.
- To limit the number of started servers to five, and to stop the servers when no activity occurs, click **Stop all instances during periods of inactivity**. Set the maximum number of instances by clicking **Limit**

the number of instances that can start, and set the **Number of instances** value to 5.

- To keep at least one instance active at all times, and to support an unlimited number of instances to start, click **Keep one instance started at all times** and **Do not limit the number of instances that can start**.

What to do next

- To edit your dynamic cluster settings, click **Servers > Dynamic clusters > *dynamic_cluster_name***. If you want to make changes to all of the members of the dynamic cluster, you can edit the dynamic cluster server template. Click **Server template**.
- Deploy an application to your dynamic cluster.
- Monitor performance with reports and operations tabs. Click **Servers > Dynamic clusters > *dynamic_cluster_name***. Click the **Reports** tab or the **Operations** tab.
- If you are using supervised mode, the autonomic managers generate recommended actions and runtime tasks upon which you can act. To view all the runtime tasks that the supervised operating mode created, click **System administration > Task management > Runtime tasks**. If you want to avoid monitoring the runtime tasks queue, you can define e-mail notification. You can create an e-mail notification profile by clicking **System administration > Task management > Notifications**.
- If you are using automatic mode, you can prevent servers from starting or restarting during the cell shutdown by adding commands to disable the application placement controller and health controller to the script that you use to stop the cell. For example, you might add the following lines to your script:

```
wsadmin -profile PlacementControllerProcs.jacl -c "disable"  
wsadmin -profile HmmControllerProcs.jacl -c "disable"
```

Remember: To change your server ports after the server has been created, you must change the ports on each server instance. You cannot change the ports through the dynamic cluster template. Because multiple servers can be on the same node by configuring vertical stacking, the ports must be unique for each server instance.

[Creating a static cluster of ODRs](#)

You can create a static cluster of on demand routers (ODRs). A static cluster is a group of application servers in a WebSphere Application Server Network Deployment environment that participates in workload management.

[Creating a dynamic cluster of ODRs](#)

You can create a dynamic cluster of on demand routers (ODRs), which means that the application placement controller selects the best node on which to start the minimum number of ODRs. If an ODR stops for any reason, the application placement controller will then start a new instance.

[Dynamic clusters](#)

A dynamic cluster is a server cluster that uses weights and workload management to balance the workloads of its cluster members dynamically, based on performance information that is collected from the cluster members. Dynamic clusters enable application server virtualization.

[Static clusters versus dynamic clusters](#)

Static clusters in a WebSphere Application Server Network Deployment configuration are different from the dynamic clusters that you can define in WebSphere Virtual Enterprise. Both types of clusters support workload balancing, however, dynamic clusters are controlled by autonomic managers that can optimize the performance of the cluster.

[Dynamic cluster isolation](#)

You can use *dynamic cluster isolation* to isolate applications from other applications that are deployed in the cell. For example, you might create a dynamic cluster isolation configuration to isolate the critical applications that an external customer uses from your internal applications, which can tolerate some instability.

[Configuring application lazy start](#)

By configuring application lazy start, you can release resources that are being consumed by inactive dynamic clusters so that other cluster instances in the cell can use these resources.

[Configuring vertical stacking](#)

You can configure vertical stacking to enable the application placement controller to start more than one instance of the dynamic cluster to improve bottleneck conditions within an application.

[HTTP session rebalancing](#)

You can dynamically and actively balance the distribution of HTTP sessions among application servers by using HTTP session rebalancing.

Next topic: [PlacementControllerProcs.jacl script](#)

Next topic: [HmmControllerProcs.jacl](#)

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Related concepts:

[Components of dynamic operations](#)

[BBSON bulletin board](#)

Related tasks:

[Deploying applications with defined service levels](#)
[Creating and managing reports](#)
[Creating PHP servers and PHP dynamic clusters](#)

Related reference:

[Administrative roles and privileges](#)

Related information:

[Application placement custom properties](#)

Creating a static cluster of ODRs

You can create a static cluster of on demand routers (ODRs). A static cluster is a group of application servers in a WebSphere® Application Server Network Deployment environment that participates in workload management.

About this task

You might consider creating a static cluster of ODRs when you are integrating the Session Initiation Protocol (SIP) ODR with Load Balancer for IBM® WebSphere Application Server.

You might consider creating a static cluster of HTTP ODRs to administer all of the ODRs at one time. If you do not create a static cluster, you must administer each ODR individually.

Procedure

1. Create an ODR to use as a template when you create your cluster. You can create the ODR in the administrative console or with the createodr.jacl script. See [Creating ODRs](#) and [createodr.jacl script](#) for more information.
2. Run the following wsadmin command to create a static cluster.

```
$AdminTask createCluster {-clusterConfig {-clusterName odr_cluster_name -
clusterType ONDEMAND_ROUTER} -convertServer {-serverNode node_name -
serverName my_odr}}
```

For the my_odr value, specify the name of the ODR that you created as a template.

3. Add ODRs to the static cluster. Run the following command for each ODR cluster member:

```
$AdminTask createClusterMember {-clusterName odr_cluster_name -
memberConfig {-memberNode node_name -memberName odr_cluster_member_name}}
```

4. Save your changes. Run the following command:

```
$AdminConfig save
```

What to do next

To modify the settings of an ODR static cluster, such as the action type and rule expression, use the wsadmin tool to complete administrative tasks at the cluster level. For more information, see [Rules for ODR routing policy administrative tasks](#) and [Rules for ODR service policy administrative tasks](#).

Parent topic: [Creating dynamic clusters](#)

Related tasks:

[Creating a dynamic cluster of ODRs](#)

[Integrating the SIP on demand router with Load Balancer](#)

[Creating ODRs](#)

[ClusterConfigCommands command group for the AdminTask object](#)

Creating a dynamic cluster of ODRs

You can create a dynamic cluster of on demand routers (ODRs), which means that the application placement controller selects the best node on which to start the minimum number of ODRs. If an ODR stops for any reason, the application placement controller will then start a new instance.

Before you begin

- The size of an ODR dynamic cluster will not currently increase beyond the minimum number of instances.
- Before running the manageODR.py script, ensure that you have the WAS_HOME environment variable configured to point to the directory of your WebSphere® installation.
- When an ODR is a member of a cluster, routing rules and other configuration settings that are set at the ODR level are ignored. Configuration settings at the cluster level take precedence over any configuration settings at the server level.
- For Session Initiation Protocol (SIP), an ODR cluster defines the other ODRs to which an ODR can route. An ODR might route a UDP message to another ODR to guarantee that server affinity is maintained in the case that the UDP message is a retransmission.

About this task

You can configure a dynamic cluster isolation policy to prevent ODRs from running on the same node with application servers. You can use the administrative console, or you can run the manageODR.py script to create ODR dynamic clusters. You can also set a cell-level custom property to enable elasticity mode for ODR dynamic clusters.

Procedure

- Create ODR dynamic clusters in the administrative console to create ODR dynamic clusters.
 1. Click **Servers > Clusters > Dynamic clusters > New**.
 2. Select **On demand router** as the dynamic cluster server type.
 3. Select the membership method, which defines how servers join the dynamic cluster as cluster instances. Select **Automatically define cluster members with rules** to create a membership policy that defines the nodes on which cluster instances can be placed. Select **Manually define cluster members** to add existing servers to the dynamic cluster. The servers that you select must be homogenous: that is, they must be of the same server type, middleware server version, and have the same applications installed.
 4. Define the dynamic cluster members.
 5. Select a dynamic cluster template.
 6. Specify dynamic cluster-specific properties.
 7. Confirm the creation of the ODR dynamic cluster. Click **Finish > Save** to save the changes to the master configuration.
 8. Select the mode of operation. Select **Servers > Clusters > Dynamic clusters > ODR_dynamic_cluster**. Choose the operational mode, and click **Set mode**.
- Run the manageODR.py script to create ODR dynamic clusters. The manageODR.py script is in the install_root/bin directory.

In the following example, an ODR dynamic cluster named myOdrCluster is created on all nodes in the node group named myOdrNodeGroup.

```
./wsadmin.sh -f manageODR.py createDynamicCluster myOdrCluster myOdrNodeGroup
```

- Set the APC.predictor custom property to enable elasticity mode for ODR dynamic clusters. By setting the custom property, the application placement controller starts and stops servers based on CPU usage alone. Furthermore, the controller no longer retrieves data from the autonomic request flow manager (ARFM) regarding what servers to start and stop.
 1. Click **System administration > Cell > Custom Properties > New**.
 2. Specify the name of the custom property as APC.predictor.
 3. Specify the value of the custom property as CPU.
 4. Click **OK**, and save and synchronize your changes.

What to do next

- After you create your ODR dynamic clusters, you can manage them from the Dynamic clusters panel in the administrative console. Click **Servers > Clusters > Dynamic clusters**.
- Monitor performance with reports and operations tabs. Click **Servers > Clusters > Dynamic clusters > ODR_dynamic_cluster**. Select the **Reports** tab or the **Operations** tab.

Parent topic: [Creating dynamic clusters](#)

Related concepts:

[Dynamic clusters](#)
[Elasticity mode](#)

Related tasks:

[Creating a static cluster of ODRs](#)

Related reference:

[Scripts](#)

[Custom properties](#)

Dynamic clusters

A dynamic cluster is a server cluster that uses weights and workload management to balance the workloads of its cluster members dynamically, based on performance information that is collected from the cluster members. Dynamic clusters enable application server virtualization.

A *dynamic cluster* is an application deployment target that can expand and contract depending on the workload in your environment. Dynamic clusters work with autonomic managers, including the application placement controller and the dynamic workload manager to maximize the use of your computing resources. Dynamic clusters are required for many of the WebSphere® Virtual Enterprise autonomic functions, including high availability and service policies.

Dynamic clusters are similar to the server clusters that you can create with WebSphere Application Server Network Deployment, but key differences exist that make dynamic clusters much more robust. For complete life-cycle management servers, the product controls the creation and deletion of server instances and can start and stop servers. For assisted life-cycle management servers, the product can control the state of servers by stopping and starting servers from a pool of predefined server instances.

Dynamic cluster membership

In WebSphere Virtual Enterprise Version 6.1 and later, two options exist for adding members to a dynamic cluster: automatically define cluster members with rules or manually define cluster members.

Note: Dynamic cluster membership definitions have changed from Version 6.0. In Version 6.0, you define a node group that contains the nodes on which you want the dynamic cluster members to run. In Version 6.1, you define a membership policy that specifies the nodes on which the dynamic cluster members run.

Automatically define cluster members with rules

By automatically defining cluster members with rules, you can create a subexpression that automatically selects nodes to host dynamic cluster members based on different node properties. This subexpression is called a *membership policy*. After you create the membership policy, you can preview the node membership before you finish creating the dynamic cluster.

Automatically defining cluster members with rules is available only for servers that have complete life-cycle management. After you create the dynamic cluster with a membership policy, dynamic cluster instances can start on any of the selected nodes. If nodes become available that meet the criteria of your membership policy, dynamic cluster instances can also start on these nodes.

Manually define cluster members

When you manually define cluster members, you statically define which servers are cluster members by selecting servers to add to the cluster. You use this option instead of the membership policy for the following reasons:

- You have an existing static cluster that you want to convert to a dynamic cluster.
- You are using *assisted life-cycle management servers*. Assisted life-cycle management servers cannot be created from the administrative console. With this option, you create representations of the servers as cluster members. These members must be homogeneous, that is, be all of the same server type, for example, a group of BEA WebLogic servers. The same version of the middleware software must be installed on all the nodes in the dynamic cluster, and you must deploy the same applications to these servers before you create the dynamic cluster.

Server templates

A *server template* is a copy of a server configuration that can be used as a starting point when a server is added to the dynamic cluster. Predefined templates exist for different versions of WebSphere Virtual Enterprise and the different middleware server types. You can also define your own server templates.

Dynamic cluster server templates

After you create a dynamic cluster, the *dynamic cluster server template* defines the properties for all of the members in the dynamic cluster.

Note: When you make a change to a dynamic cluster server template, this will overwrite any changes you made that are unique to an individual member of that cluster.

Cluster instances

You can control the creation and management of cluster instances for your dynamic cluster. These options include:

- Creating a minimum and maximum number of cluster instances.
- Stopping cluster instances when other dynamic clusters need resources.
- Allowing more than one cluster instance to start on the same node, also known as *vertical stacking*. With vertical stacking, you can improve bottleneck conditions within an application. The *stacking number* defines how many cluster instances can start on a single node.
- Specifying if cluster instances from other dynamic clusters can start on the same node, also known as *dynamic cluster isolation*.

Operating modes

Dynamic clusters act differently depending on the operating mode. Choose one of the following options for mode of operation:

- **Manual.** In manual mode, the dynamic cluster is no different from the standard application server environments with static clusters. Manual mode does not support application placement, or runtime task suggestions. The autonomic request flow manager and dynamic workload management (DWLM) can work with the cluster.
- **Supervised.** In supervised mode, the environment provides information about required corrective actions by generating runtime tasks. You can accept or deny the recommendations of the autonomic managers in the task management panel in the administrative console. To manage runtime tasks, click **System administration > Task management > Runtime tasks**.
- **Automatic.** In automatic mode, the environment takes corrective actions automatically.

Important: To use dynamic application placement, click **automatic** or **supervised** as the mode of operation.

If you are using **manual** mode for the autonomic request flow manager (ARFM) on either the cell level or deployment target level, then you also must put your dynamic clusters in manual mode. You can also use static clusters when ARFM is in manual mode for the cell or deployment target. If ARFM is in automatic mode, you can use any of the operating modes for the cluster.

Parent topic: [Creating dynamic clusters](#)

Parent topic: [Creating dynamic clusters](#)

Related concepts:

[Dynamic cluster isolation](#)

[Dynamic application placement](#)

Related tasks:

[Configuring vertical stacking](#)

Related reference:

[Static clusters versus dynamic clusters](#)

Static clusters versus dynamic clusters

Static clusters in a WebSphere® Application Server Network Deployment configuration are different from the dynamic clusters that you can define in WebSphere Virtual Enterprise. Both types of clusters support workload balancing, however, dynamic clusters are controlled by autonomic managers that can optimize the performance of the cluster.

Static cluster

A static cluster is a group of application servers in a WebSphere Application Server Network Deployment environment that participates in workload management.

Dynamic cluster

A dynamic cluster is a fundamental building block of the dynamic operations environment. You must configure dynamic clusters to get WebSphere Virtual Enterprise functionality, such as high availability and service policies.

Table 1. Static clusters versus dynamic clusters

Characteristic	Static clusters	Dynamic clusters
Membership	You must manually add application servers to static clusters.	You can automatically define cluster members with rules or you can manually define the cluster members. If you are using servers that have complete life-cycle management, you can create a membership policy that automatically selects nodes on which to host dynamic cluster members based on different node properties. If you are using servers that have assisted life-cycle management, you can statically define the servers that are cluster members.
Cluster management	You define the application servers that are in the static cluster and then start or stop all the application servers in the cluster.	A dynamic cluster can start and stop instances of servers as required. If the dynamic cluster is in automatic mode, then the server instances stop and start automatically. If the dynamic cluster is in supervised mode, runtime tasks generate to advise the administrator to start and stop servers at certain times. In manual mode, the administrator decides when instances of servers are stopped and started.
Cluster templates	When you define a static cluster, you can select an application server template on which to base all the application server instances that you create. However, any changes that you make to the template after creating the instances do not change the instances.	When you define a dynamic cluster, you can define an application server template for the application server instances. After you define the dynamic cluster, you can use the dynamic cluster server template to edit the cluster member properties. The changes are propagated to all of the application server instances.
Application server weights	You explicitly assign a weight value to each application server instance. You can also enable the dynamic workload manager to assign weight values. To enable the dynamic workload manager, click Servers > Clusters > cluster_name > Dynamic WLM .	The dynamic workload manager is enabled by default and assigns weights to the application server instances. To disable the dynamic workload manager, click Servers > Clusters > cluster_name > Dynamic WLM .
Availability	You can use static clusters in a WebSphere Application Server Network Deployment environment, or a WebSphere Virtual Enterprise environment. If you use static clusters in a WebSphere Virtual Enterprise environment, then the behavior is identical to static clusters in a WebSphere Application Server Network Deployment environment.	You can use dynamic clusters in a WebSphere Virtual Enterprise environment only.

Parent topic: [Creating dynamic clusters](#)

Related concepts:
[Dynamic clusters](#)

Related tasks:
[Creating dynamic clusters](#)

Related reference:
[Dynamic cluster administrative tasks](#)

Dynamic cluster isolation

You can use *dynamic cluster isolation* to isolate applications from other applications that are deployed in the cell. For example, you might create a dynamic cluster isolation configuration to isolate the critical applications that an external customer uses from your internal applications, which can tolerate some instability.

Scenario

Your company hosts Web applications for external customers and for internal departments in the company. To provide the most stable and secure service for your external customers, you want to be sure that their applications run on separate computers. Your hosting environment must adhere to the following requirements:

- All customer applications must run on different servers than applications from other companies for optimal security. For example, customer_1 applications must run on a different set of servers than customer_2 applications.
- Customer_2 also has a critical application that must be completely separate from both their other applications and any other applications in the cell.
- You also must host several internal applications that do not have performance and security requirements, including the company employee directory.

Solution

Use dynamic cluster isolation to meet the requirements of your customers. *Dynamic cluster isolation* specifies whether the dynamic cluster runs on the same nodes as other instances of dynamic clusters, or whether the dynamic cluster is the only dynamic cluster that is running on a node. The following configurations meet your customer requirements:

- Create dynamic clusters for customer_1 and customer_2. For the isolation requirement for each of the dynamic clusters, click *group isolation* when you create the dynamic cluster. With *group isolation*, a dynamic cluster instance can run on the same node only with instances of dynamic clusters that are a part of the same isolation group. Create an isolation group for customer_1 dynamic clusters, and another isolation group for customer_2 dynamic clusters.
- To separate the critical customer_2 application, define *strict isolation* for the dynamic cluster that is hosting the critical application. With *strict isolation*, a dynamic cluster instance can run only with other instances of the same dynamic cluster on a node.
- Because the internal employee directory application does not have any isolation requirements, click **No isolation requirements** when you create the dynamic cluster for that application.

By configuring the dynamic cluster with specific isolation requirements, you are providing the most stable and secure service for customers while also hosting internal applications for your company.

Priority of isolated dynamic clusters

Dynamic cluster isolation ensures that dynamic cluster instances from different dynamic clusters do not run on the same node, but it does not make guarantees about how the system avoids a violation of the isolation mode. Configuring strict isolation does not give a dynamic cluster priority over any other dynamic clusters.

For example, you might have an environment with one available node, and two dynamic clusters. Each dynamic cluster has the minimum number of cluster instances set to 1. Consider the following isolation configuration scenarios:

- Both of the dynamic clusters are configured with strict isolation.
- One of the dynamic clusters is configured with strict isolation, and the other dynamic cluster does not have strict isolation defined.

In both of these scenarios, the application placement controller can place a single cluster instance for one of the dynamic clusters. Dynamic cluster instances cannot be placed for both dynamic clusters because only one node is available. In both scenarios, no guarantee is made about which dynamic cluster starts an instance. Even in the second scenario, where one of the dynamic clusters has strict isolation defined, the isolated dynamic cluster does not have priority over the other dynamic cluster. This restriction can be problematic for small systems, such as a single node environment, or environments with a large number of constraints.

Parent topic: [Creating dynamic clusters](#)

Parent topic: [Creating dynamic clusters](#)

Related concepts:

[Dynamic clusters](#)

Related reference:

[Dynamic cluster administrative tasks](#)

Configuring application lazy start

By configuring application lazy start, you can release resources that are being consumed by inactive dynamic clusters so that other cluster instances in the cell can use these resources.

Before you begin

- Application lazy start requires that requests are routed through the on demand router. To create an on demand router, see [Creating ODRs](#) for more information.
- Internet Inter-ORB Protocol (IIOP) and Java™ Message Service (JMS) requests cannot be used because they are not routed through the ODR. Do not use application lazy start on dynamic clusters that run Session Initiation Protocol (SIP) applications.
- Create a dynamic cluster. See [Creating dynamic clusters](#) for more information. You can also configure application lazy start when you are creating a dynamic cluster.

About this task

An *application lazy start* is the activation of the first application server instance of a deactivated dynamic cluster when an application request arrives. You decide which applications to deactivate and subsequently lazily start. Use application lazy start if you have an environment in which the ratio of the number of dynamic clusters to the number of nodes is high, and if many dynamic clusters are not accessed for a long period. By using application lazy start, you can increase the performance and efficiency of your environment.

Procedure

1. Edit the dynamic cluster properties. In the administrative console, click **Servers > Dynamic clusters > *dynamic_cluster_name***.
2. Select the **If other dynamic clusters need resources, stop all instances of this cluster during periods of inactivity** option.
3. Set the **Time to wait before stopping instances** in minutes if the application placement controller determines that the resource is required by some other dynamic application cluster.

Results

The application placement controller tracks the amount of time that a dynamic cluster has been inactive. If another dynamic cluster needs resources, the application placement controller can stop the inactive dynamic cluster after the specified time period. The resources that were consumed by the inactive instance are released and made available for use by the dynamic cluster that requires additional resources. If a request arrives for the stopped dynamic cluster instance, the lazy start controller is activated and at least one server instance is started. In the meantime, HTTP error code 503 (server unavailable) is generated. The error page informs you that the requested application is starting and the request is resubmitted shortly.

What to do next

- Application lazy start only stops dynamic cluster instances if other dynamic clusters need resources. If you want to stop dynamic cluster instances after the specified time period even if the resources are not needed, configure the [proactiveldleStop custom property](#).
- You can configure a custom error page for the ODR that can be used for the 503 error. The error page can include an HTTP meta refresh tag that causes the browser to automatically send the request again after a certain waiting period. See [Configuring ODRs](#) for more information about configuring custom error pages.

[Application lazy start](#)

An *application lazy start* is the activation of the first application server instance of a deactivated dynamic cluster when an application request arrives. You decide which applications to deactivate and subsequently lazily start. Use application lazy start if you have an environment in which the ratio of the number of dynamic clusters to the number of nodes is high, and if many dynamic clusters are not accessed for a long time period.

[The lazy start controller](#)

The lazy start controller monitors the workload for dynamic clusters that you explicitly mark for deactivation. This controller triggers the placement controller to run off cycle and activate an inactive dynamic cluster when workload is detected. The lazy start controller also informs the placement controller when it is appropriate to deactivate these clusters.

Parent topic: [Creating dynamic clusters](#)

Related concepts:

[Dynamic clusters](#)

Related reference:

[Dynamic cluster administrative tasks](#)

Related information:

[Dynamic cluster custom properties](#)

Application lazy start

An *application lazy start* is the activation of the first application server instance of a deactivated dynamic cluster when an application request arrives. You decide which applications to deactivate and subsequently lazily start. Use application lazy start if you have an environment in which the ratio of the number of dynamic clusters to the number of nodes is high, and if many dynamic clusters are not accessed for a long time period.

Application lazy start is available for requests that are routed through the on demand router (ODR). Internet Inter-ORB Protocol (IIOP) and Java™ Message Service (JMS) requests cannot be used because they are not routed through the ODR. Do not use application lazy start on dynamic clusters that run Session Initiation Protocol (SIP) applications.

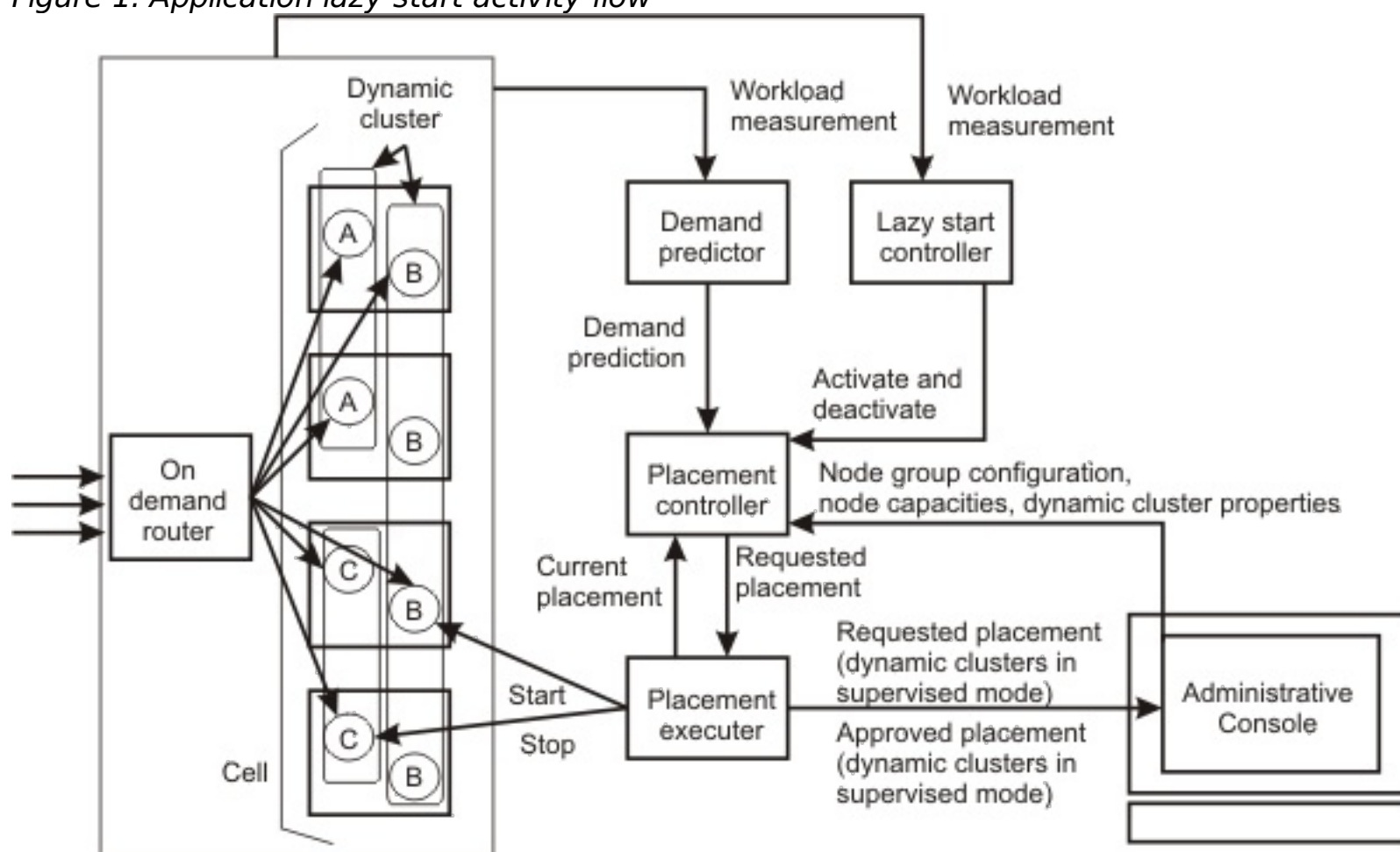
Application lazy start process

To make valuable resources available for other dynamic clusters in an environment that routes requests through the ODR, you can temporarily deactivate idle dynamic clusters, stop all server instances, and release valuable resources for other active clusters. Later, when a request arrives for one of the deactivated clusters, the cluster is activated and at least one server instance is started. In the meantime, a HTTP error code 503 (server unavailable) page is displayed when a user tries to access the server. The error page informs you that the requested application is starting and to resubmit the request shortly. You can configure the ODR to display a special error page that includes an HTTP meta refresh tag so that the browser can automatically re-send the request after a certain time period.

A lazy start controller monitors request activity for dynamic clusters that can be deactivated when idle, and lazily started when a request arrives. When a request arrives at the ODR for an inactive dynamic cluster, lazy start controller triggers placement controller to run off cycle and start an instance for that cluster. The lazy start controller also advises the placement controller when to deactivate the inactive clusters.

The following diagram demonstrates the activity flows of the lazy start and placement controller:

Figure 1. Application lazy start activity flow



Note: You can set the inactivity timeout on a dynamic cluster in automatic mode, but the application placement controller does not necessarily stop an instance after that period of inactivity if no memory contention exists on the computer that hosts this server instance. The application placement controller leverages the inactivity timeout to stop a dynamic cluster instance only if a host does not have enough memory to keep the current number of server instances running. The lazy start controller does not stop instances unless absolutely necessary or `proactivelyIdleStop` is in use. See the `proactivelyIdleStop` custom property in the [Dynamic cluster custom properties](#) topic.

Parent topic: [Configuring application lazy start](#)

Related concepts:

[The lazy start controller](#)

Related tasks:

[Creating dynamic clusters](#)

Related reference:

[Dynamic cluster administrative tasks](#)

The lazy start controller

The lazy start controller monitors the workload for dynamic clusters that you explicitly mark for deactivation. This controller triggers the placement controller to run off cycle and activate an inactive dynamic cluster when workload is detected. The lazy start controller also informs the placement controller when it is appropriate to deactivate these clusters.

The lazy start controller maintains the following information for each dynamic cluster that can be deactivated:

Deactivation timeout

Specifies the minimum idle time before the deactivation of a dynamic cluster, which can be deactivated. Deactivation stops all instances of a dynamic cluster. The lazy start controller deactivates a cluster only if it perceives a condition of memory contention on the node that is running an idle application server.

Time of last activity

Specifies the time at which the cluster was last active. The lazy start controller compares the time of last activity to the current time to determine how long the cluster has been inactive. This value is then compared to the deactivation timeout value to determine if the dynamic cluster should be deactivated if another dynamic cluster needs resources.

State

Specifies if the dynamic cluster is in Active, Inactive, and Unknown state. A dynamic cluster state is initially Unknown. If queue activity or active instances are detected, then the state is changed to Active. If the cluster is idle for a sufficient time, according to the deactivation timeout property, the state is switched to Inactive.

Parent topic: [Configuring application lazy start](#)

Parent topic: [Configuring application lazy start](#)

Related concepts:

[Application lazy start](#)

[Dynamic clusters](#)

Related tasks:

[Creating dynamic clusters](#)

Related reference:

[Dynamic cluster administrative tasks](#)

Configuring vertical stacking

You can configure vertical stacking to enable the application placement controller to start more than one instance of the dynamic cluster to improve bottleneck conditions within an application.

Before you begin

- **z/OS** Vertical stacking is supported on the z/OS® operating system, but the WebSphere® Application Server for z/OS multiple servant feature is the preferred alternative. Consider using vertical stacking for application servers on the z/OS operating systems only for application servers that host applications that are constrained to run in a single servant application server.
- To use vertical stacking, the nodes in your dynamic cluster must be at the same product version as the deployment manager, or at a later version. Nodes that are at an earlier release than the deployment manager cannot start multiple server instances on the same node.
- Vertical stacking can occur only on nodes that have complete life cycle management. You cannot configure vertical stacking for dynamic clusters that have manually defined cluster members.

About this task

Use vertical stacking to improve bottleneck conditions in your application. With vertical stacking, multiple server instances can start on a node. By configuring multiple server instances, you can use all the power that is available on the nodes when a large workload exists for the application.

Procedure

1. Determine whether your application has an internal bottleneck by profiling the application. The application placement controller assumes that the only resource bottleneck that an application might have is either in the processor or in memory. Otherwise, the application might have an internal bottleneck.
 - a. Run an instance of the application on each node.
 - b. Increase the load for the application.
 - c. Analyze the runtime reports, the throughput curves and processor utilization charts for the nodes that are running the application. Click **Runtime operations > Reports**.

If the throughput curve saturates while increasing the load, but the processor utilization remains well below 100%, the application might have an internal bottleneck. Configure vertical stacking to improve these bottleneck conditions. If the server instances are able to reach processor utilization near 100%, do not enable vertical stacking.

2. Determine the appropriate stacking number. The stacking number is the number of application servers for a cluster that are needed to use all the power of a node. For more information about determining a stacking number, see [Determining the stacking number for a dynamic cluster](#).
3. Enable vertical stacking and the stacking number. If your dynamic cluster is deployed to a node group that has homogeneous nodes, you can configure the stacking number one time in the administrative console. If your dynamic cluster is deployed to a node group that has heterogeneous nodes, you must configure the stacking number by specifying custom properties.
 - To enable vertical stacking for a dynamic cluster that is on homogenous nodes, see [Configuring a dynamic cluster with homogeneous nodes to support vertical stacking](#).
 - To enable vertical stacking for a dynamic cluster that is on heterogeneous nodes, see [Configuring a dynamic cluster with heterogeneous nodes to support vertical stacking](#).

Results

When a large workload exists, the application placement controller can fully use the power of the nodes by starting multiple cluster instances on each node.

The general formula to determine the percentage of the processor that is used by each instance on a node is $100\% / \text{maximum_number_of_stacked_instances}$. For example, if three instances are stacked, then workload is throttled to prevent any one of these instances from using more than 33% of the processor.

What to do next

Tune and monitor your application placement configuration.

[Determining the stacking number for a dynamic cluster](#)

You can use profiling to determine the stacking number for your dynamic cluster. The stacking number is the number of cluster instances that are needed to use all the power of a node.

[Configuring a dynamic cluster with homogeneous nodes to support vertical stacking](#)

You can configure vertical stacking for a dynamic cluster with nodes that have similar computing power by updating settings in the administrative console.

[Configuring a dynamic cluster with heterogeneous nodes to support vertical stacking](#)

You can configure vertical stacking for a dynamic cluster with nodes that each have different computing power by configuring custom properties for each node.

[Determining the stacking number for a dynamic cluster](#)

You can use profiling to determine the stacking number for your dynamic cluster. The stacking number is the number of cluster instances that are needed to use all the power of a node.

[Configuring a dynamic cluster with homogeneous nodes to support vertical stacking](#)

You can configure vertical stacking for a dynamic cluster with nodes that have similar computing power by updating settings in the administrative console.

[Configuring a dynamic cluster with heterogeneous nodes to support vertical stacking](#)

You can configure vertical stacking for a dynamic cluster with nodes that each have different computing power by configuring custom properties for each node.

Parent topic: [Creating dynamic clusters](#)

Next topic: [Monitoring and tuning the application placement controller](#)

Parent topic: [Creating dynamic clusters](#)

Related concepts:

[Dynamic clusters](#)

Related tasks:

[Deploying enterprise applications](#)

Related reference:

[Static clusters versus dynamic clusters](#)

Related information:

[Application placement custom properties](#)

Determining the stacking number for a dynamic cluster

You can use profiling to determine the stacking number for your dynamic cluster. The stacking number is the number of cluster instances that are needed to use all the power of a node.

Before you begin

- Determine if your applications have an internal bottleneck. See [Configuring vertical stacking](#) for more information.

About this task

The **Allow more than one instance to start on the same node** setting in the administrative console enables vertical stacking on the node. When you select this option, you must provide the number of instances to start on the same node. This value is the stacking number.

You can determine the stacking number by profiling of your application. If your dynamic cluster is homogeneous, and all of the nodes for the dynamic cluster have similar power, you need to profile and determine the stacking number on one node only. If the member nodes for the dynamic cluster are heterogeneous and vary in power, you must profile each node and determine the stacking number for each node.

Procedure

1. Start one server instance on the test node.
2. View **CPU utilization** and average throughput curves for the node. To view the charts, click **Runtime operations > Reports**.
3. Increase the load on the node, with the goal of reaching 90-100% node utilization.
4. If you cannot reach 90-100% node utilization, add and start another server instance on the test node. Continue adding server instances until the node utilization reaches 90-100%.
5. The current number of server instances on the test node is the stacking number. Remember this number so that you can configure the stacking number in the administrative console.
6. If your dynamic cluster has member nodes that are heterogeneous, repeat these steps for each node and record a stacking number for each node.

What to do next

Enable vertical stacking for your cluster. With vertical stacking enabled, the autonomic managers can limit the processor percentage that is used by each stacked server instance. The general formula to determine the amount of processor that can be used by a single instance is: $100\% / \text{maximum_number_of_stacked_instances}$. For example, if you have three stacked server instances on a node, the workload is throttled to prevent any single server instance from using more than 33 percent of the processor.

Parent topic: [Configuring vertical stacking](#)

Next topic: [Configuring a dynamic cluster with homogeneous nodes to support vertical stacking](#)

Next topic: [Configuring a dynamic cluster with heterogeneous nodes to support vertical stacking](#)

Parent topic: [Configuring vertical stacking](#)

Related concepts:

[Dynamic clusters](#)

Configuring a dynamic cluster with homogeneous nodes to support vertical stacking

You can configure vertical stacking for a dynamic cluster with nodes that have similar computing power by updating settings in the administrative console.

Before you begin

- Determine the stacking number for your dynamic cluster. See [Determining the stacking number for a dynamic cluster](#) for more information. If you determine that your application does not have an internal bottleneck when you perform application profiling, do not enable vertical stacking.

About this task

You can enable vertical stacking when you create the dynamic cluster, or you can modify the settings on an existing dynamic cluster to enable vertical stacking. Perform the following steps to modify the settings on an existing dynamic cluster.

Procedure

1. In the administrative console, click **Servers > Dynamic clusters > *dynamic_cluster_name***.
2. To enable vertical stacking, click **Allow more than one instance to start on the same node**.
3. Enter the number of instances to start on the same node. This value is the stacking number that you calculated.

What to do next

Tune and monitor your application placement configuration.

Parent topic: [Configuring vertical stacking](#)

Previous topic: [Determining the stacking number for a dynamic cluster](#)

Next topic: [Monitoring and tuning the application placement controller](#)

Parent topic: [Configuring vertical stacking](#)

Related concepts:

[Dynamic clusters](#)

Configuring a dynamic cluster with heterogeneous nodes to support vertical stacking

You can configure vertical stacking for a dynamic cluster with nodes that each have different computing power by configuring custom properties for each node.

Before you begin

- Determine the stacking number for each of your dynamic cluster nodes. See [Determining the stacking number for a dynamic cluster](#) for more information. If you determine that your application does not have an internal bottleneck when you perform application profiling, do not enable vertical stacking.

About this task

If your dynamic cluster is deployed to a group of nodes that are heterogeneous, you must configure the stacking number for each individual node. You can configure the stacking number for each node by creating custom properties on the dynamic cluster.

Procedure

1. Enable vertical stacking on the dynamic cluster.
 - a. In the administrative console, click **Servers > Dynamic clusters > cluster_name**.
 - b. To enable vertical stacking, click **Allow more than one instance to start on the same node**.
 - c. Enter the number of instances to start on the same node. Set this value to the largest stacking number that you have determined for your nodes. When you define the individual stacking numbers as custom properties for each node, this value is overridden. However, if no custom property value is specified, the node uses the number of instances value that is defined on the administrative console page.
2. Create a custom property for the stacking number on one of your nodes. In the administrative console, click **Servers > Dynamic clusters > cluster_name > Custom properties > New**.
3. Enter the name of the custom property. The name is `numVerticalInstances.node_name`, where `node_name` is the name of your node.

For example: If your node is named `node1`, the name of your custom property is `numVerticalInstances.node1`.

4. Enter the value of the custom property. The value of this custom property is the stacking number that you calculated for the specified node.
5. Add a custom property that defines the stacking number for each one of your nodes in the dynamic cluster.

Example

You have determined that among the three nodes on which your dynamic cluster runs, that each node has a different stacking number.

Table 1. Heterogeneous node stacking number scenario

Node name	Stacking number
node_1	3
node_2	5
node_3	8

To configure these stacking numbers in the administrative console, use the following configuration:

1. Set the **number of instances** value for the dynamic cluster to 8. The number of instances value must be greater than or equal to your largest stacking number.
2. Specify the following custom properties:
 - `numVerticalInstances.node_1 = 3`
 - `numVerticalInstances.node_2 = 5`
 - `numVerticalInstances.node_3 = 8`

What to do next

Tune and monitor your application placement configuration.

Parent topic: [Configuring vertical stacking](#)

Previous topic: [Determining the stacking number for a dynamic cluster](#)

Next topic: [Monitoring and tuning the application placement controller](#)

Parent topic: [Configuring vertical stacking](#)

Related concepts:

[Dynamic clusters](#)

Related information:

[Dynamic cluster custom properties](#)

HTTP session rebalancing

You can dynamically and actively balance the distribution of HTTP sessions among application servers by using HTTP session rebalancing.

WebSphere® Application Server assigns HTTP sessions based on application server *affinity*. After a session is established in a particular application server, subsequent requests that belong to the same session are assigned to the application server that has the established session affinity.

Session affinities cause the session load to balance out for a cluster in the long term, and generally any imbalances that occur are short-lived and tolerable. However, you can use session rebalancing to help some of the shorter term imbalances that can occur, without modifying the performance benefits of session affinities.

WebSphere Virtual Enterprise uses session rebalancing to expedite the balancing of sessions across a cluster. You can use session rebalancing if you use distributed sessions and track your sessions with cookies. By default, WebSphere Virtual Enterprise application servers are configured with Session tracking enabled using cookies, but without distributed sessions.

The steps that you use for configuring servers in static clusters can also be used for servers in dynamic clusters. If you are configuring dynamic clusters, leave the dynamic cluster in manual mode initially so that the servers in the cluster are not started automatically. After the distributed environment settings of all the servers in the cluster are changed, switch the dynamic cluster to automatic mode. If you do not switch modes, you must stop and restart the servers that are already started in the dynamic cluster to pick up the distributed environment setting changes.

When a new server becomes available in a dynamic cluster, that server does not have any assigned HTTP sessions. However, the existing servers in the dynamic cluster have sessions with assigned affinity.

Session rebalancer configuration

You can enable or disable session rebalancing by adjusting the `HttpSessionRebalanceOff` custom property on dynamic clusters in the administrative console. To enable session rebalancing on the other middleware server types and WebSphere eXtreme Scale servers, you must set the `HttpSessionRebalanceOff` custom property to false for the dynamic cluster. Session rebalancing is automatically enabled for dynamic clusters that have application servers. Set the custom property on the specific dynamic cluster.

Additionally, set the `NoAffinitySwitchBack` custom property on each application server that the on demand router (ODR) routes to. Set the property value to true to maintain affinity to the new member even after the original member comes back up. When a cluster member fails, requests are routed to a different cluster member, and sessions are activated in the other member. Session affinity is maintained to the new member, and the session requests that were created in the original cluster member are routed back to it after it comes back up. Even though session rebalancing is turned on by default, sessions are not rebalanced if the session replication is turned off for a dynamic cluster.

Version 6.0.1 and later session rebalancer

Note: If you are using a mixed environment that includes both Version 6.0 (all releases) and Version 6.1 nodes, the session rebalancer continues to run the way that it did before you installed Version 6.1.

With Version 6.0.1 and later, the goal of session rebalancing is to reassign sessions so that the number of sessions that are running in each server in the dynamic cluster is proportional to the assigned weight of the servers. The dynamic workload manager (DWLM) performs the session rebalancing function. DWLM decides how many sessions to move and where to move the sessions based on the knowledge about the entire dynamic cluster. The DWLM component factors in the session location and can initiate session moves in addition to changing the weight on specific servers. Any events that DWLM monitors can initiate the moves, such as new servers starting, overloaded servers, and so on.

DWLM often picks a very uneven distribution of routing weights. This approach is allowed because the only objective DWLM has is equalizing service times. A secondary goal is added to DWLM. DWLM attempts to even out the routing weights when possible without significant degradation of the primary objective of equalizing service times.

If you are using WebSphere eXtreme Scale-based HTTP session support instead of the normal WebSphere Application Server session management, HTTP session rebalancing is not supported.

If you are using sessions that are maintained by Uniform Resource Locator (URL) rewriting or Secure Sockets Layer (SSL) IDs, no rebalancing is performed. Rebalancing is also not performed on HTTP POST requests. Sessions are not rebalanced if the session is accessed by issuing the HTTP request directly to the application server HTTP port, bypassing the ODR. The PMI counter for memory sessions reflects the moving sessions. Specifically, the PMI counter decreases on existing servers as sessions move and increases on the new servers that are recently started. However, new sessions are load balanced across any of the servers in the cluster. See the [WebSphere Application Server Network Deployment Information Center](#) for more information about analyzing PMI data and best practices for using HTTP sessions.

Version 6.1 session rebalancer

In WebSphere Virtual Enterprise Version 6.1 and later, the session rebalancer request and response filters are in the ODR. These filters keep track of which sessions are being sent to which servers, to get a live session count.

Using information from the dynamic workload manager (DWLM) for each dynamic cluster, the ODR also tracks the number of sessions that are being sent to each server, ranking the servers in the dynamic cluster. This ranking is used to determine how many sessions to move between servers. The default configurations vary depending on the type of dynamic cluster that you are using:

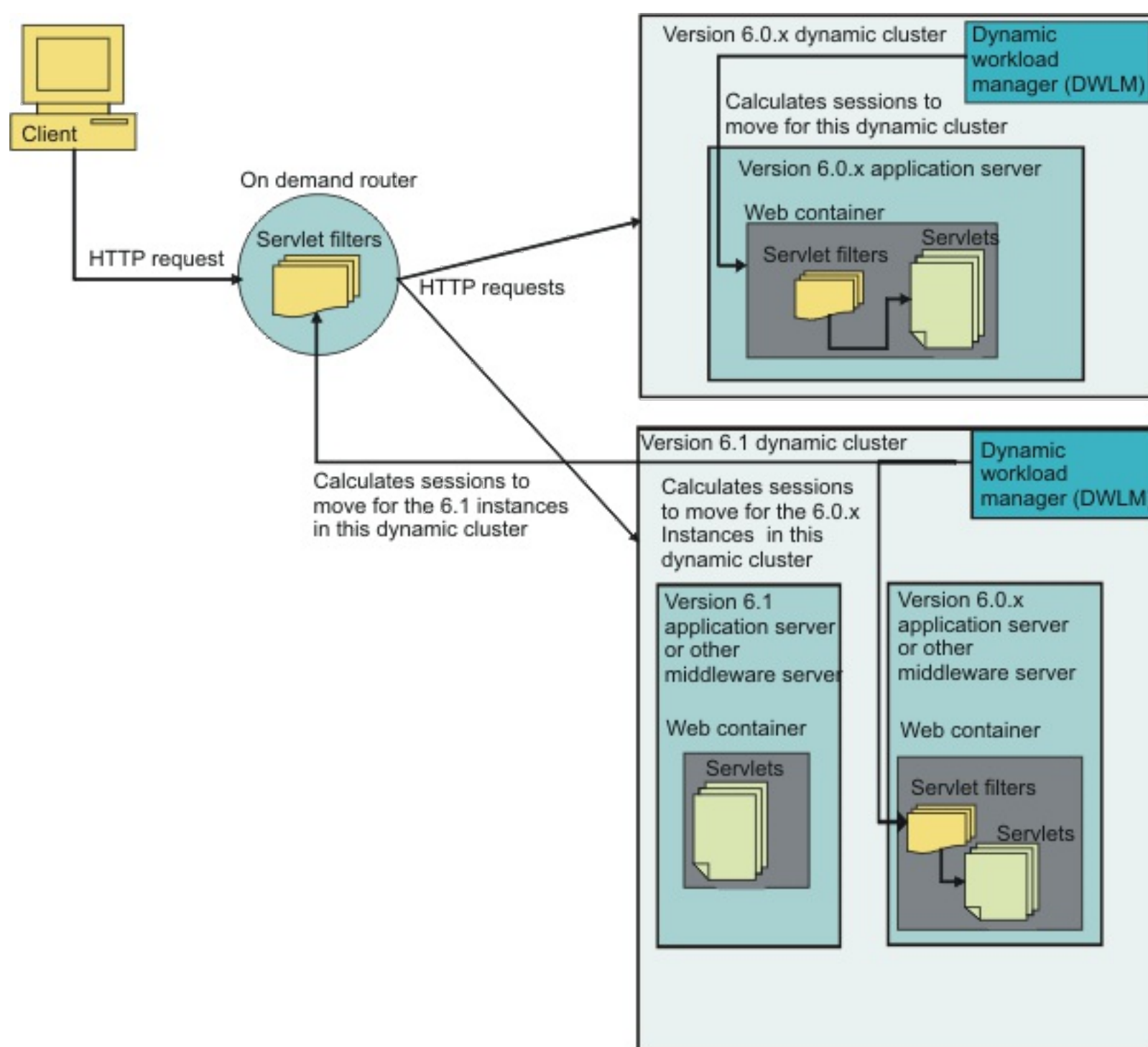
- For dynamic clusters that are made of WebSphere Application Server application servers, the distributed session configuration is detected. Sessions are automatically rebalanced unless you turn off the session rebalancing by setting the `HttpSessionRebalanceOff` custom property to `true`.
- For dynamic clusters that have other types of servers, the runtime cannot detect if the server is using distributed sessions. Session rebalancing is disabled by default. You can set the `HttpSessionRebalanceOff` custom property to `false` on the dynamic cluster to enable session rebalancing.

HTTP session balancing is supported when you are using WebSphere eXtreme Scale-based HTTP session support instead of the normal WebSphere Application Server session management.

WebSphere Virtual Enterprise rebalancing scenario

The following example illustrates session rebalancing in a Version 6.1 environment. In this example, both Version 6.0.1 and Version 6.1 dynamic clusters exist in the configuration. The Version 6.0.1 servers continue to perform session rebalancing within the server. The Version 6.1 server performs session rebalancing by sending information from DWLM to the ODR. The ODR then routes the HTTP sessions to the appropriate server.

Figure 1. HTTP session rebalancing in WebSphere Virtual Enterprise Version 6.1



Parent topic: [Creating dynamic clusters](#)

Related concepts:
[Dynamic clusters](#)

Related tasks:
[Creating dynamic clusters](#)

Related information:
[Dynamic cluster custom properties](#)

☞ [WebSphere Application Server Version 7.0 Information Center](#)

☞ [WebSphere Application Server Version 8.0 Information Center](#)

Adding middleware servers to configurations

Using middleware servers, you can manage all of the servers in your environment, including servers that are not created with WebSphere® Virtual Enterprise, by using the product administrative domain.

Before you begin

If you are configuring representations in the domain of externally created middleware servers, perform the following tasks on the nodes:

1. Install and configure the middleware software on the nodes.
2. [Installing the middleware agent](#)

The middleware agent on each node must be at the same fix pack level as the deployment manager.

3. [Federating middleware nodes](#)
4. [Creating and configuring ODRs](#)
5. Each of your nodes must have the same network view for each endpoint and resource that is related to WebSphere Application Server. All host entries or host alias entries that are related to WebSphere Application Server within a hosts file must be included in the hosts file on each of the other nodes in the configuration, including any hosts file on nodes that run the node agent or middleware agent. For example, if you define a product related host alias called xdaagentA in the hosts file for the nodeA node, then the hosts file on each middleware node must contain an entry for the xdaagentA host alias.

About this task

Middleware servers encompass all servers in the middleware tier that provide the infrastructure for applications or their data. The steps vary depending on if you are configuring complete life-cycle management middleware servers or assisted life-cycle management middleware servers.

Procedure

- [Adding complete life-cycle middleware servers](#)

Complete life-cycle servers include any servers that the environment can instantiate, or create. These server types include WebSphere Application Server types such as application servers, generic servers, Web servers, and proxy servers. The product adds complete life-cycle management for PHP servers and WebSphere Application Server Community Edition servers.

- [Adding assisted life-cycle middleware servers](#)

With assisted life-cycle servers, you use templates to create representations of servers in the administrative console. However, these servers still exist within the administrative domain of their respective middleware platform. These servers can be controlled operationally. Server health and performance are monitored, and you can configure the administrative console to display log files and configuration files for these servers.

What to do next

To easily manage groups of servers to host an application, configure dynamic clusters. You can create clusters of either type of server. However, dynamic application placement is supported only on complete life cycle servers. When you create a dynamic cluster of assisted life cycle middleware servers, the same version of middleware software and the same applications must be installed on the servers. If the servers in a dynamic cluster have varying sets of applications installed, routing through the on demand router might result in failed requests.

The logs and trace views in the administrative console that you access by clicking **Troubleshooting > Logs and trace** are not supported for externally created middleware servers and PHP servers. You can use the external log viewing service to view the log files for these server types in the administrative console. To configure the external log viewing service, navigate to the configuration panel for the server and click **External log service**.

[Adding assisted life-cycle middleware servers](#)

By configuring assisted life-cycle middleware servers, you can manage representations of externally created middleware servers that were created outside of the administrative domain.

[Adding complete life-cycle middleware servers](#)

You can complete life-cycle middleware servers so that the dynamic operations environment can govern all aspects of server operations, create and remove server instances, and deploy applications to the servers.

[Adding complete life-cycle middleware servers](#)

You can complete life-cycle middleware servers so that the dynamic operations environment can govern all aspects of server operations, create and remove server instances, and deploy applications to the servers.

[Adding assisted life-cycle middleware servers](#)

By configuring assisted life-cycle middleware servers, you can manage representations of externally created middleware servers that were created outside of the administrative domain.

Managing middleware servers

To define middleware servers, you can use middleware server templates and middleware descriptors. To manage middleware servers that are defined, you can view your middleware server log files and configuration files in the administrative console and configure server operations to run on your middleware servers.

Previous topic: [Federating middleware nodes](#)

Previous topic: [Installing the middleware agent](#)

Previous topic: [Creating and configuring ODRs](#)

Next topic: [Creating dynamic clusters](#)

Next topic: [Configuring the external log viewing service](#)

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Related concepts:

[Middleware nodes and servers](#)

[Middleware agent](#)

Related reference:

[Middleware server creation administrative tasks](#)

Adding complete life-cycle middleware servers

You can complete life-cycle middleware servers so that the dynamic operations environment can govern all aspects of server operations, create and remove server instances, and deploy applications to the servers.

About this task

By configuring complete life-cycle servers, the environment can perform the following tasks on your middleware servers:

- Create and remove server instances based on the needs of the dynamic operations environment
- Govern all aspects of server configuration
- Provide operational control over the server
- Deploy applications to the server
- Visualize and monitor server health and performance
- Use expression-based dynamic clustering (for PHP servers, WebSphere® Application Server Community Edition servers, and application servers)

Procedure

Follow the steps to create your complete life-cycle middleware server. The steps vary depending on the server type:

- **On demand router (ODR):** The on demand router is a specialized proxy server that can route work to application server nodes. The ODR is required for a WebSphere Virtual Enterprise environment. See [Creating and configuring ODRs](#) for more information.
- **PHP server:** WebSphere Virtual Enterprise can create PHP servers. See [Creating PHP servers and PHP dynamic clusters](#) for more information.
- **WebSphere Application Server Community Edition server:** WebSphere Virtual Enterprise supplies assisted and complete life-cycle management for WebSphere Application Server Community Edition servers. Complete life-cycle management is supported for WebSphere Application Server Community Edition Version 2 and later servers. WebSphere Application Server Community Edition Version 1 and Version 2 servers can be represented in the administrative console as assisted life-cycle servers. See [Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#) for more information.
- **Generic server:** Generic servers are managed in, but not supplied by WebSphere Application Server or WebSphere Virtual Enterprise. These servers might be a Java™ server, a C or C++ server or process, a CORBA server, or a Remote Method Invocation (RMI) server. See [Creating generic servers](#) for more information.
- **Proxy server:** Proxy servers, like the ODR, route requests to application server nodes. However, the ODR is the required proxy for dynamic operations. See [Creating proxy servers](#) for more information.
- **Web server:** WebSphere Application Server and WebSphere Virtual Enterprise can work with a Web server to route requests for dynamic content, such as servlets, from Web applications. See [Communicating with Web servers](#) for more information.
- **Application server:** Application servers are supported for both WebSphere Application Server and WebSphere Virtual Enterprise. By creating dynamic clusters of application servers, you enable application server virtualization. See [Creating application servers](#) for more information.

What to do next

You can create dynamic clusters of application servers, WebSphere Application Server Community Edition servers, and PHP servers.

[Creating PHP servers and PHP dynamic clusters](#)

By creating PHP Hypertext Preprocessor (PHP) servers or dynamic clusters, you can deploy PHP applications. You can use the product to manage these applications and servers.

[Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#)

You can create WebSphere Application Server Community Edition complete life-cycle management servers and expression-based dynamic clusters. You can deploy managed applications to those servers and dynamic clusters.

[Creating PHP servers and PHP dynamic clusters](#)

By creating PHP Hypertext Preprocessor (PHP) servers or dynamic clusters, you can deploy PHP applications. You can use the product to manage these applications and servers.

Distributed platforms

[Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#)

You can create WebSphere Application Server Community Edition complete life-cycle management servers and expression-based dynamic clusters. You can deploy managed applications to those servers and dynamic clusters.

Previous topic: [Creating and configuring ODRs](#)

Next topic: [Creating dynamic clusters](#)

Parent topic: [Adding middleware servers to configurations](#)

Related concepts:

[Middleware nodes and servers](#)

[Supported middleware server types](#)

Related tasks:

- [Creating application servers](#)
- [Creating generic servers](#)
- [Creating proxy servers](#)
- [Communicating with Web servers](#)

Creating PHP servers and PHP dynamic clusters

By creating PHP Hypertext Preprocessor (PHP) servers or dynamic clusters, you can deploy PHP applications. You can use the product to manage these applications and servers.

Before you begin

- Install Apache HTTP Server and PHP on the nodes that you want to host PHP servers. The supported Apache HTTP Server versions are 1.3 (all releases), 2.0 (all releases), and 2.2 (all releases). The supported PHP versions are 4 and 5 (all releases). Default server templates are provided for these combinations of Apache HTTP Server and PHP.
- Federate these nodes into the cell.
- **Attention:** If you are using Apache HTTP Server Version 2.2 and PHP Version 5.2 (all releases), then you must perform some manual edits to the `httpd.conf` file. Instead of performing these edits each time you create a PHP server or dynamic cluster, create a PHP server with the `APACHE22_PHP5` template, edit the `httpd.conf` file, and create a server template from that server, which you can use to create the rest of your PHP servers.

You can also choose an existing server to use as the template for a new PHP server by using the PHP server administrative tasks, or by selecting a predefined server instead of a template. You cannot select a user-defined template from the PHP server wizard.

About this task

A PHP server is defined in WebSphere® Virtual Enterprise as an Apache HTTP Server with the `mod_php` module. You can define a PHP server on nodes that are running the product or the middleware agent. PHP server configuration consists of three documents: the `server.xml`, `httpd.conf` and `php.ini` files. The `server.xml` file contains properties that describe the Apache and PHP runtime locations and server operations. The `httpd.conf` file is the Apache HTTP Server configuration file that includes PHP modules. You can define variables in the administrative console to be used in the `httpd.conf` file. The `php.ini` file contains configuration data that is used by the PHP interpreter.

Procedure

1. Configure the middleware descriptors so that you can run discovery to find the Apache and PHP runtimes. In the administrative console, click **System administration > Middleware descriptors > *middleware_platform_name***. The default locations are listed in the `apacheWebServerRuntime` and `phpRuntime` middleware descriptors. The discovery service uses middleware descriptors to define where to look for the Apache and PHP runtimes. Verify that the middleware descriptors contain the correct installation locations for your environment. Use a semicolon delimiter for Windows paths, and a colon delimiter for UNIX paths.
2. Run discovery to find the PHP and Apache HTTP Server runtimes on the nodes that you want to host PHP servers. Discovery runs automatically when the middleware agent starts. Discovery also runs at a predefined time interval that you can specify in the middleware descriptor. However, you can also invoke discovery:
 - a. In the administrative console, click **System administration > Middleware nodes**.
 - b. Select the nodes that you want to run through discovery, and select the **Run discovery** operational action.
 - c. Click **Run**.
 - d. Verify that the discovery service found the run time. In the administrative console, click **System administration > Middleware nodes > *node_name* > Node Installation Properties**. If any runtime environments exist, then properties that begin with `APACHE` or `PHP` strings are displayed.Only the default installation locations are listed in the `apacheWebServerRuntime` and `phpRuntime` middleware descriptors. You can modify the middleware descriptor so that the discovery looks for the correct installation location.
3. Create PHP deployment targets. The deployment target is where you deploy the PHP application. You can create individual PHP servers or PHP dynamic clusters.
 - Create a PHP server.
 - a. In the administrative console, click **Servers > Add a server**. Choose **Create a new server instance**, and select the **PHP server** type. You can also click **Servers > Other middleware servers > PHP servers > New**.
 - b. Select the node on which you want the PHP server to run. This node must have the PHP and Apache HTTP Server runtime environments.
 - c. Choose the appropriate Apache HTTP Server and PHP runtime environments for the node.

For both runtimes, a list of variables displays at different levels of granularity such as `APACHE`, `APACHE_2`, `APACHE_2_0`, and `APACHE_2_0_59`. With these variables, you can select the runtime that corresponds to either a specific or a general version that you need. For example, the most generic variables are `APACHE` and `PHP`, which always point to the newest versions that are installed on your node. Select these variables if you always want to use the newest runtimes and you do not need a specific version. You can also choose a runtime that is based on major version such as `APACHE_2`. With this variable, you can use any Apache 2

runtime. The newest Apache 2 runtime that is installed on your node is used. By always using the newest Apache Version 2 runtime, you can upgrade between minor versions without any manual server configuration changes. If you update from Apache Version 2.0.58 to Apache Version 2.0.59 and add the new runtime location to the middleware descriptor, the runtime service automatically detects the newer runtime and updates the server configuration to point to the new runtime.

- d. Choose the PHP server template.
- e. Confirm and save the PHP server.
- o. Create a PHP dynamic cluster.
 - a. In the administrative console, click **Servers > Dynamic clusters > New**.
 - b. Choose the PHP server dynamic cluster type, and name the dynamic cluster.
 - c. With PHP servers, you must use automatic membership. Define a membership policy to identify which nodes host the PHP servers in the dynamic cluster. These nodes must have the PHP and Apache HTTP Server runtime environments installed.

For example, you might use the following membership policy:

```
node_property$APACHE IS NOT NULL and node_property$PHP IS NOT NULL
```

Restriction: PHP dynamic cluster members must be at the same two digit Apache version and one digit PHP version. This limitation is caused by the differences in configuration between releases. If multiple versions are installed on your system, update the membership policy to include the version number, like in the following example:

```
node_property$APACHE_2_0 IS NOT NULL and node_property$PHP_4 IS NOT NULL
```

- d. Select the server template for the PHP runtime and Apache HTTP Server runtime that you are using.
 - e. Specify other dynamic cluster properties.
 - f. Save your PHP dynamic cluster.
4. If you are using PHP Version 5.2 (any release), then you must manually edit the `httpd.conf` file to use the `php5apache2_2.dll` module, instead of the default `php5apache2.dll` module.

- a. Verify that you created the PHP server with the `APACHE22_PHP5` template.
- b. Open the `httpd.conf` file. In the administrative console, click **Servers > Other middleware servers > PHP servers > *php_server_name* > External configuration**. Choose the `httpd.conf` file, and click **Retrieve**.
- c. Modify the file to use the `php5apache2_2.dll` module. Search for the following string in the `httpd.conf` file, where `php_server_root` is the location of your PHP server:

```
LoadModule php5_module "php_server_root/php5apache2.dll"
```

Edit this line to reference the `php5apache2_2.dll` module. See the following example:

```
LoadModule php5_module "php_server_root/php5apache2_2.dll"
```

- d. Apply and save your changes.
 - e. Optional: To avoid repeating these steps for each PHP Version 5.2 server that you create, make a template of your current server that you can use when you create subsequent PHP servers. In the administrative console, click **Servers > Other middleware servers > PHP servers > Templates > New**. Select the server from which you want to create a template, specify properties for the template, and save your changes. Choose this template when you create other PHP servers or dynamic clusters. You can also choose an existing server to create your new PHP server from the PHP server wizard.
5. Optional: Update the HTTP and HTTPS ports for your servers. The default port is uniquely generated. modifying the port changes the port for the Apache server, which updates the `httpd.conf` file.
- a. In the administrative console, click **Servers > Other middleware servers > PHP servers > *php_server***.
 - b. Edit **HTTP connector** or **HTTPS connector**. Click **OK** and save and synchronize your changes.

What to do next

Deploy PHP applications to your PHP servers and dynamic clusters.

Parent topic: [Adding complete life-cycle middleware servers](#)

Previous topic: [Federating middleware nodes](#)

Next topic: [Deploying applications with defined service levels](#)

Next topic: [Deploying PHP applications](#)

Parent topic: [Adding complete life-cycle middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Creating dynamic clusters](#)
[Creating middleware server templates](#)
[Configuring middleware descriptors](#)

Related reference:

[PHP server administrative tasks](#)
[Dynamic cluster administrative tasks](#)

Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters

You can create WebSphere® Application Server Community Edition complete life-cycle management servers and expression-based dynamic clusters. You can deploy managed applications to those servers and dynamic clusters.

Before you begin

- Determine the version of WebSphere Application Server Community Edition that you want to run:
 - For complete life-cycle management, you must use WebSphere Application Server Community Edition Version 2 or later.
 - For WebSphere Application Server Community Edition Version 2.1, you must be running WebSphere Virtual Enterprise Version 6.1.0.5 or later. Additional steps are required to enable Java™ Management Extensions (JMX) security in WebSphere Application Server Community Edition Version 2.1. See [Configuring JMX security for WebSphere Application Server Community Edition Version 2.1 servers](#) for more information.
 - If you are using WebSphere Application Server Community Edition Version 1 (all releases), then you can only use assisted life-cycle management. To configure assisted life-cycle servers, see [Configuring assisted life-cycle WebSphere Application Server Community Edition servers](#).
- Install the WebSphere Application Server Community Edition runtime environment to the nodes on which you want to run servers. You can either use the installation wizard that is included in WebSphere Application Server Community Edition, or you can configure the centralized installation manager to install the runtime environment. See [Downloading and installing WebSphere Application Server Community Edition](#) for more information.
- Federate your nodes into the cell. See [Federating middleware nodes](#) for more information.

Procedure

1. Configure the middleware descriptors so that you can run discovery to find the WebSphere Application Server Community Edition runtime environment. If you use the centralized installation manager to install WebSphere Application Server Community Edition, you can skip this step. In the administrative console, click **System administration > Middleware descriptors > wasceRuntime**. The discovery service uses middleware descriptors to define where to look for the runtime environment. Verify that the middleware descriptors contain the correct installation locations for your environment. Use a semicolon delimiter for Windows paths, and a colon delimiter for UNIX paths. The following default values exist:
 - C:\Program Files\IBM\WebSphere\AppServerCommunityEdition
 - /opt/IBM/WebSphere/AppServerCommunityEdition
2. Run discovery to find the WebSphere Application Server Community Edition runtimes on the nodes that you want to host servers. If you use the centralized installation manager to install WebSphere Application Server Community Edition, you can skip this step. Discovery runs automatically when the middleware agent starts. Discovery also runs at a predefined time interval that you can specify in the middleware descriptor. However, you can also invoke discovery with the following steps:
 - a. In the administrative console, click **System administration > Middleware nodes**.
 - b. Select the nodes that you want to run through discovery, and select the **Run discovery** operational action.
 - c. Click **Run**.
 - d. Verify that the discovery service found the runtime environment. In the administrative console, click **System administration > Middleware nodes > node_name > Node Installation Properties**. If any runtime environments exist, then properties that begin with WASCE strings are displayed.

Only the default installation location is listed in the wasceRuntime middleware descriptor. You can modify the middleware descriptor so that the discovery looks for the correct installation location.

3. Create WebSphere Application Server Community Edition deployment targets. The deployment target is where you deploy the WebSphere Application Server Community Edition application. You can create individual WebSphere Application Server Community Edition servers or dynamic clusters.
 - Create a WebSphere Application Server Community Edition server.
 - a. In the administrative console, click **Servers > Add a server**. Choose **Create a new server instance**, and select the **WebSphere Application Server Community Edition server** type. You can also click **Servers > Other middleware servers > WebSphere Application Server Community Edition servers > New**.
 - b. Select the node on which you want the WebSphere Application Server Community Edition server to run. This node must have the WebSphere Application Server Community Edition runtime.
 - c. Choose the WebSphere Application Server Community Edition server template.
 - d. Confirm and save the WebSphere Application Server Community Edition server.
 - Create a WebSphere Application Server Community Edition dynamic cluster.
 - a. In the administrative console, click **Servers > Dynamic clusters > New**.
 - b. Choose the WebSphere Application Server Community Edition server dynamic cluster type, and name the dynamic cluster.
 - c. To create a dynamic cluster with complete life-cycle management, choose **Automatically**

define cluster members with rules. Define a membership policy to identify which nodes host the WebSphere Application Server Community Edition servers in the dynamic cluster. These nodes must have the WebSphere Application Server Community Edition runtime environment installed. The default membership policy follows:

```
node_property$IS_WASCE_DISCOVERED = 'TRUE'
```

This membership policy looks for WebSphere Application Server Community Edition Version 2.0 and later runtimes.

If you want to specify a specific version, you might use the following membership policy:

```
node_property$WASCE_2.0.0 IS NOT NULL
```

This policy defines membership as all nodes that run WebSphere Application Server Community Edition Version 2.0.0. The group of nodes that you define must have the exact same version of WebSphere Application Server Community Edition installed. For example, you cannot combine Version 1 and 2 servers in the same dynamic cluster.

- d. Select the server template for the WebSphere Application Server Community Edition runtime environment that you are using.
- e. Specify other dynamic cluster properties.
- f. Save your WebSphere Application Server Community Edition dynamic cluster.

Results

The new server displays in the list on the **Servers > Other middleware servers > WebSphere Application Server Community Edition servers** administrative console page. If you see an error status icon (🚫), the creation process did not complete. This error can occur because of problems in the WebSphere Application Server Community Edition server runtime environment. Delete the server and try creating the server again.

The list of servers that displays on this administrative console page includes all of your complete life-cycle, assisted life-cycle, and discovered WebSphere Application Server Community Edition servers. If you have WebSphere Application Server Community Edition dynamic clusters defined, the cluster members are also listed on this administrative console page.

What to do next

- Deploy applications to your WebSphere Application Server Community Edition servers and dynamic clusters. See [Deploying WebSphere Application Server Community Edition applications](#) for more information.
- Stop discovery from examining the installation location that you used to create your managed servers.
 - If you used discovery to find the WebSphere Application Server Community Edition runtime environment, you can disable discovery globally or you can remove the specific installation location from the middleware descriptor.
 - If you installed the WebSphere Application Server Community Edition runtime environment with the centralized installation manager, discovery was not needed to register the configuration. Do not include centralized installation manager paths in the discovery paths in the middleware descriptor. If you used the default centralized installation manager installation location, you must globally disable discovery or change the default discovery location to a different path.

See [Configuring middleware descriptors](#) for more information.

Do not use the WebSphere Application Server Community Edition console to edit your complete life-cycle servers, except for functions that WebSphere Virtual Enterprise does not support. Using the WebSphere Application Server Community Edition console can cause unexpected results. For example, if you edit the server port numbers in the WebSphere Application Server Community Edition console, the port value that is configured for WebSphere Virtual Enterprise is overridden with the new value.

[WebSphere Application Server Community Edition servers](#)

You can centrally manage WebSphere Application Server Community Edition servers and applications along with your other servers and applications from the WebSphere Virtual Enterprise environment.

Distributed platforms

[WebSphere Application Server Community Edition servers](#)

You can centrally manage WebSphere Application Server Community Edition servers and applications along with your other servers and applications from the WebSphere Virtual Enterprise environment.

Distributed platforms

[Configuring JMX security for WebSphere Application Server Community Edition Version 2.1 servers](#)

If you use the Java Management Extensions (JMX) security feature in WebSphere Application Server Community Edition Version 2.1, additional steps are required to enable the security feature in WebSphere Virtual Enterprise.

Parent topic: [Adding middleware servers to configurations](#)

Previous topic: [Federating middleware nodes](#)

Next topic: [Deploying WebSphere Application Server Community Edition applications](#)

Next topic: [Creating middleware server templates](#)

Next topic: [Configuring middleware descriptors](#)

Parent topic: [Adding complete life-cycle middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Creating dynamic clusters](#)

[Configuring assisted life-cycle WebSphere Application Server Community Edition servers](#)

[Configuring JMX security for WebSphere Application Server Community Edition Version 2.1 servers](#)

Related reference:

[Middleware server creation administrative tasks](#)

[Middleware server management administrative tasks](#)

WebSphere Application Server Community Edition servers

You can centrally manage WebSphere® Application Server Community Edition servers and applications along with your other servers and applications from the WebSphere Virtual Enterprise environment.

Complete life-cycle servers

You can create WebSphere Application Server Community Edition servers in the WebSphere Virtual Enterprise environment that have complete life-cycle management. To take advantage of complete life-cycle management, you must install WebSphere Application Server Community Edition Version 2 or later. Complete life-cycle servers provide the following benefits:

- You can create WebSphere Application Server Community Edition servers from the administrative console, which creates a server on the WebSphere Application Server Community Edition runtime environment. These servers can also be vertically stacked on a server from a single run time.
- You can create expression-based dynamic clusters of WebSphere Application Server Community Edition servers.
- You can view the performance of WebSphere Application Server Community Edition servers in the runtime operations panels.
- You can install managed applications to WebSphere Application Server Community Edition servers and dynamic clusters. When you install managed applications, you deploy the application within the WebSphere Virtual Enterprise administrative console.
- You can associate complete life-cycle servers with health policies to enable health management and monitoring.

Important: Do not use the WebSphere Application Server Community Edition console to edit your complete life-cycle servers, except for functions that WebSphere Virtual Enterprise does not support.

Discovered servers

Middleware discovery locates existing installations of WebSphere Application Server Community Edition servers and their installed applications, and creates the corresponding configuration to include servers and applications in the WebSphere Virtual Enterprise cell. Middleware discovery can preserve the time investment that went into building your original WebSphere Application Server Community Edition environment.

Middleware discovery can find WebSphere Application Server Community Edition Version 1 and later and Version 2 and later servers. These servers are represented as assisted life-cycle servers. You cannot create expression-based dynamic clusters of these servers, but you can group the servers together in a dynamic cluster. The members of these dynamic clusters must have the same version of WebSphere Application Server Community Edition and the same applications installed.

Any discovered applications are represented as unmanaged applications. You can install managed applications with the WebSphere Virtual Enterprise administrative console on discovered WebSphere Application Server Community Edition Version 2 and later servers. For WebSphere Application Server Community Edition Version 1 discovered servers, you can create representations of the applications that have been installed in the WebSphere Application Server Community Edition console as unmanaged applications only.

Assisted life-cycle servers

If you configured WebSphere Application Server Community Edition servers with WebSphere Virtual Enterprise, you manually created representations of the servers in your environment that were assisted life-cycle servers.

You cannot create expression-based dynamic clusters of these servers, but you can group the servers together in a dynamic cluster. The members of these dynamic clusters must have the same version of WebSphere Application Server Community Edition and the same applications installed.

You can install managed applications to assisted life-cycle WebSphere Application Server Community Edition Version 2 and later servers with the administrative console or administrative tasks. For WebSphere Application Server Community Edition Version 1 assisted life-cycle servers, you can install unmanaged applications only. You install unmanaged applications within the WebSphere Application Server Community Edition administrative console and create representations of the applications in the WebSphere Virtual Enterprise administrative console.

Topology

Figure 1. WebSphere Virtual Enterprise and WebSphere Application Server Community Edition topology

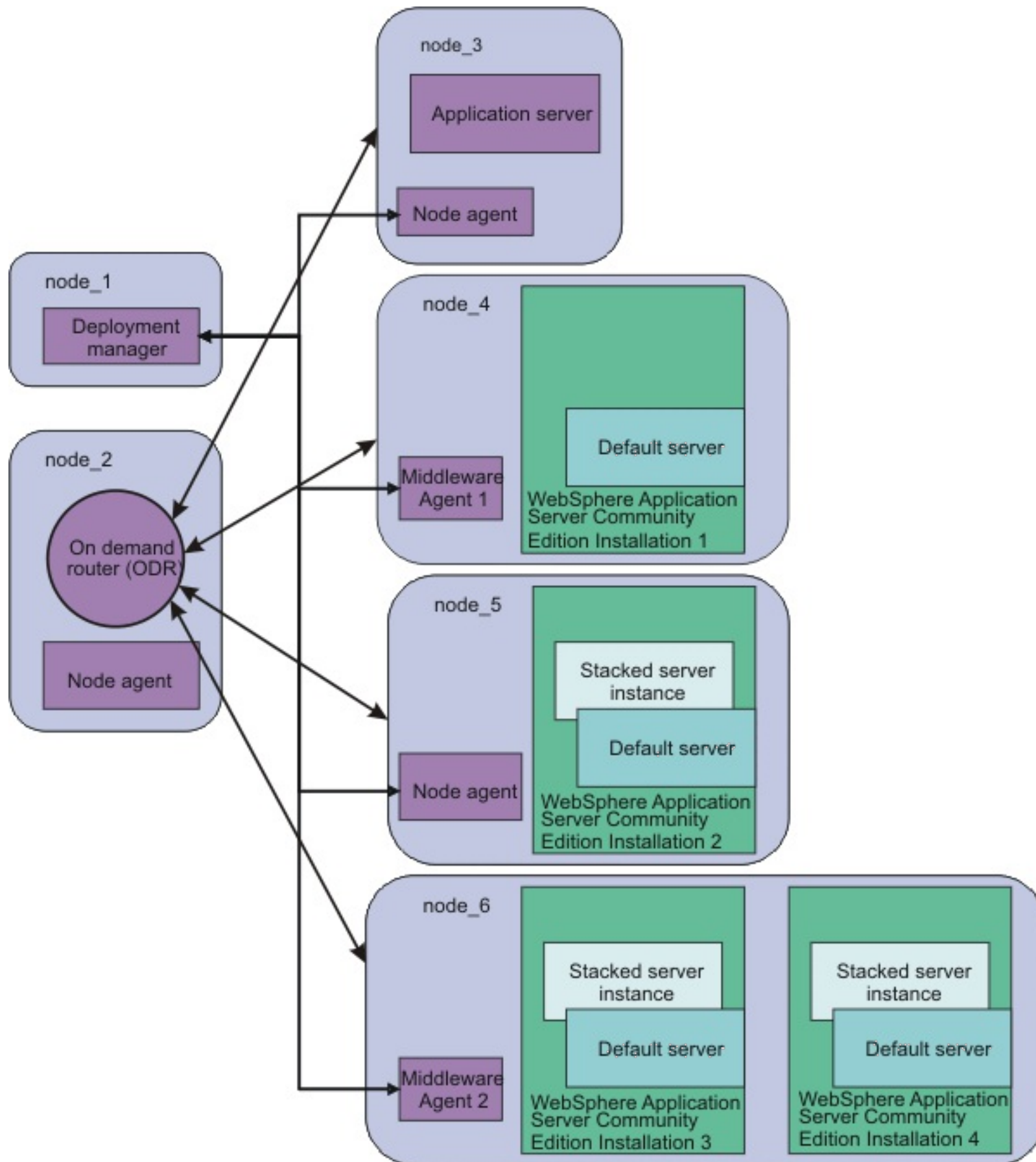


Table 1. Topology diagram installation locations

Installation in topology diagram	Installation path
WebSphere Application Server Community Edition installation 1	/opt/IBM/WebSphere/AppServerCommunityEdition
WebSphere Application Server Community Edition installation 2	/opt/WASCE2.0
WebSphere Application Server Community Edition installation 3	/opt/WASCE2.0
WebSphere Application Server Community Edition installation 4	/IBM/WebSphere/CommunityEdition/AppServer/

Parent topic: [Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#)

Parent topic: [Distributed platforms](#) [Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#)

Related concepts:
[Middleware nodes and servers](#)

Related tasks:
[Deploying WebSphere Application Server Community Edition applications](#)
[Configuring assisted life-cycle WebSphere Application Server Community Edition servers](#)

Configuring JMX security for WebSphere Application Server Community Edition Version 2.1 servers

If you use the Java™ Management Extensions (JMX) security feature in WebSphere® Application Server Community Edition Version 2.1, additional steps are required to enable the security feature in WebSphere Virtual Enterprise.

Before you begin

- You must have WebSphere Application Server Community Edition Version 2.1 installed with JMX security enabled.

About this task

JMX security is a feature that is added in WebSphere Application Server Community Edition Version 2.1. When you create new servers or dynamic clusters in the administrative console in WebSphere Virtual Enterprise Version 6.1.0.5 or later, you can use the wasce21s server template to create servers that have JMX security enabled. If you have discovered servers, you must perform the additional steps in this topic to configure JMX security.

Procedure

- If you have an existing server, start the WebSphere Application Server Community Edition Version 2.1 server for which JMX security needs to be enabled.
- Import the WebSphere Application Server Community Edition certificates into the WebSphere Application Server trust store.
 - In the administrative console, click **Security > SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore > Signer certificates**.
 - Enter the host name of the node where WebSphere Application Server Community Edition Version 2.1 is running, the HTTPS port number, and an alias for the certificate.
 - Click **Retrieve signer information**.
 - Save and synchronize your settings.
- Configure your WebSphere Application Server Community Edition Version 2.1 server. If you have not yet created your servers in the environment, you can create the server using the wasce21s server template. If the server has already been discovered by WebSphere Virtual Enterprise, you must perform the additional steps that are required to enable security that are listed in this step.

To create a new WebSphere Application Server Community Edition Version 2.1 server or dynamic cluster with JMX security enabled, perform the following steps:

- To create a WebSphere Application Server Community Edition server in the administrative console, click **Servers > Other middleware servers > WebSphere Application Server Community Edition servers > New**. To create a WebSphere Application Server Community Edition dynamic cluster in the administrative console, click **Servers > Dynamic cluster > New**. Choose WebSphere Application Server Community Edition as the server type.
- Select the wasce21s template in the wizard.
- Save and synchronize your changes.

For a server that has already been discovered by WebSphere Virtual Enterprise, perform the following steps:

- Stop the WebSphere Application Server Community Edition Version 2.1 server.
- Edit the config.xml file to disable the JMX service and enable the JMX secure connector.
 - In the administrative console, open the WebSphere Application Server Community Edition Version 2.1 server configuration. Click **Servers > Other middleware servers > WebSphere Application Server Community Edition servers > wasce_server_name**.
 - Click the **External configuration** tab. Edit the config.xml file.
 - Make the following changes, highlighted in bold text:

```
<module name="org.apache.geronimo.framework/j2ee-security/2.1.1/car">
  <gbean name="JMXService" load="false">
    <attribute name="protocol">rmi</attribute>
    <attribute name="host">${ServerHostname}</attribute>
    <attribute name="port">${JMXPort + PortOffset}</attribute>
    <attribute
name="urlPath">/jndi/rmi://${ServerHostname}:${NamingPort +
PortOffset}/JMXConnector</attribute>
  </gbean>
</module>
...
...
...
<module name="org.apache.geronimo.configs/clustering/2.1.1/car" load="false">
  <gbean name="Node">
```



```

        <attribute name="nodeName">${clusterNodeName}</attribute>
    </gbean>
</module>
...
...
...
<module name="org.apache.geronimo.framework/jmx-security/2.1.1/car" load="true">
    <gbean name="JMXSecureConnector" load="true">
        <attribute name="protocol">rmi</attribute>
        <attribute name="host">${ServerHostname}</attribute>
        <attribute name="port">${JMXSecurePort + PortOffset}</attribute>
        <attribute
name="urlPath">/jndi/rmi://${ServerHostname}:${NamingPort +
PortOffset}/JMXSecureConnector</attribute>
        <attribute name="clientAuth">>false</attribute>
    </gbean>
</module>

```

- c. Add the `JAVA_OPTS` variable.
 - i. In the administrative console, click **Servers > Other middleware servers > WebSphere Application Server Community Edition servers > wasce_server_name > Variables > New**.
 - ii. Create a new variable called `JAVA_OPTS`.
 - iii. Enter the following text in the **Value** field, specifying the appropriate values for the system properties:

```

-Djavax.net.ssl.keyStore=${GERONIMO_HOME}/var/security/keystores/geronimo-default
-Djavax.net.ssl.keyStorePassword=secret
-Djavax.net.ssl.trustStore=${GERONIMO_HOME}/var/security/keystores/geronimo-default
-Djavax.net.ssl.trustStorePassword=secret

```
 - iv. Click **OK**.
- d. Add the UseJMXSecureConnector custom property.
 - i. In the administrative console, click **Servers > Other middleware servers > WebSphere Application Server Community Edition servers > wasce_server_name > Custom properties > New**
 - ii. Enter UseJMXSecureConnector as the value in the **Name** field.
 - iii. Enter true in the **Value** field.
 - iv. Click **OK**.
- e. Save and synchronize your changes.
- f. Restart your server.

Results

You can use WebSphere Application Server Community Edition Version 2.1 with JMX security enabled.

What to do next

Configure your WebSphere Application Server Community Edition servers. See [Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#) for more information.

Parent topic: [Distributed platforms](#) [Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#)

Related concepts:

[WebSphere Application Server Community Edition servers](#)

Related tasks:

[Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#)
[Configuring middleware descriptors](#)

Related information:

[Middleware server custom properties and variables](#)

Adding assisted life-cycle middleware servers

By configuring assisted life-cycle middleware servers, you can manage representations of externally created middleware servers that were created outside of the administrative domain.

Before you begin

See [Adding middleware servers to configurations](#) for information about installing the middleware agent on nodes and federating those nodes into the configuration.

About this task

With assisted life-cycle middleware servers, you can create a representation of the server in the administrative console. The middleware agent provides the information that WebSphere Virtual Enterprise needs to manage these servers. You can configure the following assisted life-cycle middleware server types:

- Apache Tomcat
- JBoss Application Server
- Custom HTTP servers
- BEA WebLogic Server
- WebSphere Application Server Community Edition
- Apache HTTP Server
- External WebSphere application server

After you install the middleware agent on WebSphere Application Server Community Edition nodes and federate the nodes, middleware discovery can automatically create representations of these servers in the WebSphere Virtual Enterprise administrative console. You do not need to perform the manual steps to create the representation of the server in the administrative console. You can continue to manage these servers in the WebSphere Application Server Community Edition console. Make any representative changes in the WebSphere Virtual Enterprise administrative console.

Procedure

1. Create a representation of the middleware server in the administrative console.
 - a. Add an existing server. In the administrative console, click **Servers > Add a server** and select **Add an existing server**.
 - b. Select the node on which the middleware server that you are configuring runs. The node must be running the middleware agent or a node agent. Create a name for the server that is unique among all servers in the cell.
 - c. Specify the server template that you want to use for the middleware server.
 - d. Click **Finish**.
 - e. Click **Save** to save your changes to the master configuration.
2. Perform additional configuration steps for the middleware server, including adjusting the values for the WebSphere variables for each server type and configuring server operations to stop and start your servers:
 - **Apache Tomcat servers:** See [Configuring Apache Tomcat servers](#) for more information.
 - **BEA WebLogic servers:** See [Configuring BEA WebLogic servers](#) for more information.
 - **JBoss servers:** See [Configuring JBoss servers](#) for more information.
 - **WebSphere Application Server Community Edition servers:** See [Configuring assisted life-cycle WebSphere Application Server Community Edition servers](#) for more information.
 - **Custom HTTP servers:** See [Configuring custom HTTP servers](#) for more information.
 - **External WebSphere application server:** You can manage previous versions of WebSphere application servers, from Version 5.1 and later. See [Configuring external WebSphere application servers](#) for more information.

The WebSphere variables define settings for the middleware server such as the installation location and vary depending on the middleware server type. By editing the server start and stop operations, you can specify the username and password that is required to start and stop the servers in the middleware server.

3. Start the middleware server. In the administrative console, click **Servers > All servers**. Select the server that you want to start and click **Start**. The start server operation that is defined for the server runs.

Note: If the middleware agent and the server are stopped at the same time, the last known status of the server is reported. Because the last known status is reported as started, the on demand router (ODR) tries to route to the server.

What to do next

You can create a middleware server template that is based on your configured server. After you create a middleware server template, you can create additional servers using the template, which creates servers with the same settings as your original server.

To easily manage groups of existing servers to host an application, configure dynamic clusters. By configuring a dynamic cluster, the product can adjust the number of running servers to meet the application service policy.

For assisted life-cycle middleware servers, you group together the representations that you created. These servers must have the same applications installed.

The logs and trace views in the administrative console are not supported for assisted life-cycle middleware servers. Configure the external log viewing service to view the log files for these server types in the administrative console.

[Configuring external WebSphere application servers](#)

You can create middleware server representations of WebSphere application servers that are in other cells.

[Configuring Apache Tomcat servers](#)

After you create a middleware server representation of an Apache Tomcat server in the administrative console, you must update a few configuration settings so that they are defined specifically for your Apache Tomcat installation.

[Configuring BEA WebLogic servers](#)

After you create a middleware server representation of a BEA WebLogic server in the administrative console, you must update configuration settings so that they are defined specifically for your BEA WebLogic installation.

[Configuring JBoss servers](#)

After you create a middleware server representation of a JBoss server in the administrative console, you must update a few configuration settings so that they are defined specifically for your JBoss installation.

[Configuring assisted life-cycle WebSphere Application Server Community Edition servers](#)

After you manually create a middleware server representation of a WebSphere Application Server Community Edition server in the administrative console, you must update a few configuration settings so that they are defined specifically for your WebSphere Application Server Community Edition installation. If the middleware discovery function is enabled, you do not need to perform these steps.

[Configuring assisted life-cycle WebSphere Application Server Community Edition servers](#)

After you manually create a middleware server representation of a WebSphere Application Server Community Edition server in the administrative console, you must update a few configuration settings so that they are defined specifically for your WebSphere Application Server Community Edition installation. If the middleware discovery function is enabled, you do not need to perform these steps.

[Configuring Apache Tomcat servers](#)

After you create a middleware server representation of an Apache Tomcat server in the administrative console, you must update a few configuration settings so that they are defined specifically for your Apache Tomcat installation.

[Configuring BEA WebLogic servers](#)

After you create a middleware server representation of a BEA WebLogic server in the administrative console, you must update configuration settings so that they are defined specifically for your BEA WebLogic installation.

[Configuring JBoss servers](#)

After you create a middleware server representation of a JBoss server in the administrative console, you must update a few configuration settings so that they are defined specifically for your JBoss installation.

[Configuring custom HTTP servers](#)

After you create a middleware server representation of a custom HTTP server in the administrative console, you must update a few configuration settings so that they are defined specifically for your custom HTTP server installation.

[Configuring external WebSphere application servers](#)

You can create middleware server representations of WebSphere application servers that are in other cells.

Parent topic: [Adding middleware servers to configurations](#)

Next topic: [Configuring the external log viewing service](#)

Next topic: [Configuring middleware server operations](#)

Parent topic: [Adding middleware servers to configurations](#)

Related concepts:

[Middleware nodes and servers](#)

[Middleware agent](#)

Related tasks:

[Creating dynamic clusters](#)

[Installing the middleware agent](#)

[Creating middleware server templates](#)

Configuring assisted life-cycle WebSphere Application Server Community Edition servers

After you manually create a middleware server representation of a WebSphere® Application Server Community Edition server in the administrative console, you must update a few configuration settings so that they are defined specifically for your WebSphere Application Server Community Edition installation. If the middleware discovery function is enabled, you do not need to perform these steps.

Before you begin

Before you perform this task, consider the other options for configuring your WebSphere Application Server Community Edition servers:

- If you do not have an existing WebSphere Application Server Community Edition configuration, consider using complete life-cycle servers. By configuring complete life-cycle servers, you can deploy managed applications and create expression-based dynamic clusters that are made of WebSphere Application Server Community Edition servers. See [Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#) for more information.
- If you enable middleware discovery, representations of your existing WebSphere Application Server Community Edition servers and applications are automatically created. These representations are assisted life-cycle servers and unmanaged applications. You do not need to perform any of the additional steps in this task if middleware discovery is enabled. To enable middleware discovery for WebSphere Application Server Community Edition servers, click **System administration > Middleware descriptors > wasceRuntime**. If the **Discovery interval** value is set to -1, discovery is disabled. Change this value to indicate the interval at which you want middleware discovery to run.

If middleware discovery is disabled, you must perform the manual steps in this task to create a representation of your WebSphere Application Server Community Edition in the WebSphere Virtual Enterprise cell. As a prerequisite, federate the node that is running the WebSphere Application Server Community Edition server and create a representation of the server in the administrative console.

About this task

Perform this task only if you have an existing WebSphere Application Server Community Edition environment and you do not want to enable middleware discovery to automatically create a representation of your environment.

Procedure

1. Update the WebSphere variables so that they match the settings on the node that is running the WebSphere Application Server Community Edition server. In the administrative console, click **Environment > WebSphere Variables**.

Edit the WASCE_HOME variable, which specifies the home directory of your WebSphere Application Server Community Edition installation. Default values are set at the cell scope level:

- **Windows** C:\Program Files\IBM\WebSphere\AppServerCommunityEdition
- **AIX** **HP-UX** **Solaris** /opt/IBM/WebSphere/AppServerCommunityEdition

If the home directory is different on different nodes in your configuration, select the node scope and create the WASCE_HOME variable for the specific node.

2. Update server operations for the WebSphere Application Server Community Edition server to include the user name and password.
 - a. In the administrative console, click **Servers > Other middleware servers > WebSphere Application Server Community Edition servers > wasce_server > Server operations**.
 - b. Edit the start and stop server operations to include the user name and password that is required to run these commands.

Results

You created a representation of your WebSphere Application Server Community Edition server in the administrative console. To view all of your WebSphere Application Server Community Edition servers, click **Servers > Other middleware servers > WebSphere Application Server Community Edition servers**.

The list of servers that displays on this administrative console page includes all of your complete life-cycle, assisted life-cycle, and discovered WebSphere Application Server Community Edition servers. If you have WebSphere Application Server Community Edition dynamic clusters defined, the cluster members are also listed on this administrative console page.

What to do next

- To use this configuration when you are configuring your other WebSphere Application Server Community Edition servers, create a server template of your current WebSphere Application Server Community Edition server.
- To view the log files for your WebSphere Application Server Community Edition server in the administrative console, use the external log viewing service. This service is enabled by default. The following directories are configured by default:
 - \${WASCE_HOME}/var/logs

- `${WASCE_HOME}/catalina/logs`
- `${AGENT_HOME}/logs/${WAS_SERVER_NAME}`
- To edit configuration files for your WebSphere Application Server Community Edition server, use the external configuration service. The `${WASCE_HOME}/var/config/config.xml` file is included by default.
- After you create representations of all the WebSphere Application Server Community Edition servers, you can add the representations to a dynamic cluster. To cluster these servers, each server must be running the same applications and be running the same version of WebSphere Application Server Community Edition.

Parent topic: [Adding assisted life-cycle middleware servers](#)

Parent topic: [Adding middleware servers to configurations](#)

Previous topic: [Federating middleware nodes](#)

Next topic: [Configuring middleware server operations](#)

Next topic: [Creating middleware server templates](#)

Next topic: [Configuring the external log viewing service](#)

Next topic: [Creating dynamic clusters](#)

Parent topic: [Adding assisted life-cycle middleware servers](#)

Related concepts:

[WebSphere Application Server Community Edition servers](#)

Related tasks:

[Configuring WebSphere variables](#)

Related reference:

[Middleware server creation administrative tasks](#)

[Middleware server management administrative tasks](#)

Configuring Apache Tomcat servers

After you create a middleware server representation of an Apache Tomcat server in the administrative console, you must update a few configuration settings so that they are defined specifically for your Apache Tomcat installation.

Before you begin

Federate the node that is running the Apache Tomcat server and create a representation of the server in the administrative console.

About this task

If you installed Apache Tomcat in a location other than the default directory, update the WebSphere® variables so that the installation directory is correct. To perform server operations on your server from the administrative console, you must configure the server operations for the Apache Tomcat middleware server representation.

Procedure

1. Update the WebSphere variables so that they match the settings on the node that is running the Apache Tomcat server. In the administrative console, click **Environment > WebSphere Variables**. Edit the following variables:

CATALINA_HOME

Specifies the home directory of your Apache Tomcat installation.
Default values are set at the cell scope level:

Windows C:\Program Files\Apache Software Foundation\Tomcat
AIX **HP-UX** **Solaris** /usr/local/apache-tomcat

If the home directory is different on different nodes in your configuration, select the node scope and create the CATALINA_HOME variable for the specific node.

JAVA_HOME

Specifies the directory of your Java™ Development Kit (JDK) installation.

If the JAVA_HOME variable is not set in the environment of the node, define the JAVA_HOME variable in the administrative console at the cell scope or at the node scope.

2. Update server operations for the Tomcat server to include the user name and password.
 - a. In the administrative console, click **Servers > Other middleware servers > Apache Tomcat servers > tomcat_server > Server operations**.
 - b. Edit the start and stop server operations to include the user name and password that is required to run these commands.
3. Optional: Update the HTTP and HTTPS ports for your servers. The default port is 8080. If you use a value other than the default, you must change the port in the administrative console.
 - a. In the administrative console, click **Servers > Other middleware servers > Apache Tomcat servers > tomcat_server**.
 - b. Edit **HTTP connector** or **HTTPS connector**. Click **OK** and save and synchronize your changes.
4. **AIX** **HP-UX** **Solaris** Optional: Disable security for the Tomcat server. By default, the -security argument is enabled when you create the server. If you want to start the server without security, you must remove the -security argument in the administrative console.
 - a. In the administrative console, click **Servers > Apache Tomcat servers > tomcat_server > Start**.
 - b. Remove the -security argument from the list of **Executable arguments**. Click **OK**, and save and synchronize your changes.

If you want to create a new Tomcat server that does not have security enabled, you must create a custom Tomcat server template. See [Creating middleware server templates](#) for more information.

What to do next

- To use this configuration when you are configuring your other Apache Tomcat servers, create a server template of your current Apache Tomcat server.
- To view the log files for your Apache Tomcat server in the administrative console, use the external log viewing service. This service is enabled by default. The following directories are configured by default:
 - {\$CATALINA_HOME}/logs
 - {\$WASCE_HOME}/catalina/logs
 - {\$AGENT_HOME}/logs/{\$WAS_SERVER_NAME}
- To edit configuration files for your Apache Tomcat server, use the external configuration service. The following files are included by default:
 - {\$CATALINA_HOME}/conf/server.xml
 - {\$CATALINA_HOME}/conf/catalina.policy
 - {\$CATALINA_HOME}/conf/web.xml
- After you create representations of all the Apache Tomcat servers, you can add the representations to a

dynamic cluster.

Parent topic: [Adding middleware servers to configurations](#)

Previous topic: [Federating middleware nodes](#)

Next topic: [Creating dynamic clusters](#)

Parent topic: [Adding assisted life-cycle middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

[Supported middleware server types](#)

Related tasks:

[Configuring middleware server operations](#)

[☞ \[Configuring WebSphere variables\]\(#\)](#)

[Creating middleware server templates](#)

[Configuring the external log viewing service](#)

Related reference:

[Middleware server creation administrative tasks](#)

Configuring BEA WebLogic servers

After you create a middleware server representation of a BEA WebLogic server in the administrative console, you must update configuration settings so that they are defined specifically for your BEA WebLogic installation.

Before you begin

Federate the node that is running the BEA WebLogic server and create a representation of the server in the administrative console.

About this task

If you installed BEA WebLogic in a location other than the default directory, update the WebSphere® variables so that the installation directory is correct. To perform operations on your server from the administrative console, you must configure the server operations for the BEA WebLogic middleware server representation.

Procedure

1. Update the WebSphere variables so that they match the settings on the nodes that are running the BEA WebLogic server. In the administrative console, click **Environment > WebSphere Variables**. Edit the following variables:

WEBLOGIC_ADMINHOST

Specifies the host name of the WebLogic administration server.

Default value: *localhost*

Scope: cell

WEBLOGIC_ADMINPORT

Specifies the port of the WebLogic administration server.

Default value: 7001

Scope: cell

WEBLOGIC_ADMINPROTOCOL

Specifies the protocol to use when connecting to the WebLogic administration server.

Default value: *t3*

Scope: cell

WEBLOGIC_ADMINURL

Specifies the URL for the WebLogic administration server. The value is created from other variables.

Default value:

`${WEBLOGIC_ADMINPROTOCOL}://${WEBLOGIC_ADMINHOST}:${WEBLOGIC_ADMINPORT}`

Scope: cell

WEBLOGIC_SERVERROOT

Specifies an alias to the WEBLOGIC_DOMAINDIR variable.

Default value: `${WEBLOGIC_DOMAIN_DIR}`

Scope: cell

WEBLOGIC_SERVERNAME

Specifies the name of the server in the WebLogic administration environment.

Default value: `${WAS_SERVER_NAME}`

Scope: server

BEA_HOME

Specifies the root installation location.

Windows **Default value:** *c:\bea*

AIX **HP-UX** **Solaris** **AIX** **HP-UX** **Default value:** */opt/bea*

Scope: cell

WEBLOGIC_HOME

Specifies the WebLogic root directory. The root is version-specific, for example `${BEA_HOME}\weblogic91`.

Default value: `${BEA_HOME}\weblogic`

Scope: cell

WEBLOGIC_DOMAINNAME

Specifies the name of the WebLogic domain.

Default value: *myDomain*

Scope: cell

WEBLOGIC_DOMAINDIR

Builds the path to the WebLogic domain.

Default value: `${BEA_HOME}\user_projects\domains\${WEBLOGIC_DOMAINNAME}`

Scope: cell

If the home directory is different on different nodes in your configuration, select the node scope and create another variable with the same name for the specific node.

2. Update server operations for the BEA WebLogic server to include the user name and password.

- a. In the administrative console, click **Servers > Other middleware servers > BEA WebLogic servers > weblogic_server > Server operations**.
 - b. Edit the stop server operation to include the user name and password that you use to perform this operation.
3. Optional: Update the HTTP and HTTPS ports for your servers. The default port is 7001. If you use a value other than the default, you must change the port in the administrative console.
- a. In the administrative console, click **Servers > Other middleware servers > BEA WebLogic servers > weblogic_server**.
 - b. Edit **HTTP connector** or **HTTPS connector**. Click **OK** and save and synchronize your changes.

What to do next

- To use this configuration when you are configuring your other BEA WebLogic servers, create a server template of your current BEA WebLogic server.
- To view the log files for your BEA WebLogic server in the administrative console, use the external log viewing service. This service is enabled by default. The following directories are configured by default:
 - `${WEBLOGIC_SERVER_ROOT}/${WEBLOGIC_SERVER_NAME}`
 - `${AGENT.HOME}/logs/${WAS_SERVER_NAME}`
- To edit configuration files for your BEA WebLogic server, use the external configuration service. The following file is included by default:
 - `${WEBLOGIC_SERVER_ROOT}/config.xml` on host `${WEBLOGIC_ADMINHOST}`
- After you create representations of all the BEA WebLogic servers, you can add the representations to a dynamic cluster.

Parent topic: [Adding assisted life-cycle middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

[Supported middleware server types](#)

Related tasks:

[Configuring middleware server operations](#)

 [Configuring WebSphere variables](#)

[Federating middleware nodes](#)

[Adding middleware servers to configurations](#)

[Creating middleware server templates](#)

[Configuring the external log viewing service](#)

[Creating dynamic clusters](#)

Related reference:

[Middleware server creation administrative tasks](#)

Configuring JBoss servers

After you create a middleware server representation of a JBoss server in the administrative console, you must update a few configuration settings so that they are defined specifically for your JBoss installation.

Before you begin

Federate the node that is running the JBoss server and create a representation of the server in the administrative console.

About this task

If you installed JBoss in a location other than the default directory, update the WebSphere® variables so that the installation directory is correct. To perform server operations on your server from the administrative console, you must configure the server operations for the JBoss middleware server representation.

Procedure

1. Update the WebSphere variables so that they match the settings on the node that is running the JBoss server. In the administrative console, click **Environment > WebSphere Variables**.

Edit the JBOSS_DIST variable, which specifies the home directory of your JBoss installation. Usually, the default installation location includes the JBoss version, so it is likely that you need to update the default values. Default values are set at the cell scope level:

- o **Windows** C:\Program Files\jboss
- o **AIX** **HP-UX** **Solaris** /usr/local/jboss

Edit the JBOSS_PROFILE variable, which specifies the configuration set in use for the server representation, and defaults to the name of the server itself. The default value is set at the server scope level: \${WAS_SERVER_NAME}. If these settings are different on different nodes in your configuration, select the node scope and create the variables for the specific node.

2. Update server operations for the JBoss server to include the user name and password.
 - a. In the administrative console, click **Servers > Other middleware servers > JBoss servers > jboss_server > Server operations**.
 - b. Edit the start and stop server operations to include the user name and password that is required to run these commands.
3. Optional: Update the HTTP and HTTPS ports for your servers. The default port is 8080. If you use a value other than the default, you must change the port in the administrative console.
 - a. In the administrative console, click **Servers > Other middleware servers > JBoss servers > jboss_server**.
 - b. Edit **HTTP connector** or **HTTPS connector**. Click **OK** and save and synchronize your changes.

What to do next

- To use this configuration when you are configuring your other JBoss servers, create a server template of your current JBoss server.
- To view the log files for your JBoss server in the administrative console, use the external log viewing service. This service is enabled by default. The following directories are configured by default:
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/logs
 - o \${AGENT_HOME}/logs/\${WAS_SERVER_NAME}
- To edit configuration files for your JBoss server, use the external configuration service. The following files are included by default:
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/conf/jboss-minimal.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/conf/jboss-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/conf/log4j.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/conf/login-config.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/conf/standardjaws.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/conf/standardjboss.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/conf/standardjbosscomp-jdbc.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/conf/jndi.properties
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/conf/server.policy
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/bsh-deployer.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/cache-invalidation-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/client-deployer-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/ear-deployer-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/ejb-deployer-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/hsqldb-ds.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/jbossjca-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/jmx-invoker-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/jsr88-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/mail-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/monitoring-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/properties-service.xml
 - o \${JBOSS_DIST}/server/\${JBOSS_PROFILE}/deploy/schedule-manager-service.xml

- `${JBOSS_DIST}/server/${JBOSS_PROFILE}/deploy/schedule-service.xml`
- `${JBOSS_DIST}/server/${JBOSS_PROFILE}/deploy/sqlexception-service.xml`
- After you create representations of all the JBoss servers, you can add the representations to a dynamic cluster.

Parent topic: [Adding middleware servers to configurations](#)

Previous topic: [Federating middleware nodes](#)

Next topic: [Creating middleware server templates](#)

Next topic: [Configuring the external log viewing service](#)

Next topic: [Creating dynamic clusters](#)

Parent topic: [Adding assisted life-cycle middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

[Supported middleware server types](#)

Related tasks:

[Configuring middleware server operations](#)

 [Configuring WebSphere variables](#)

Related reference:

[Middleware server creation administrative tasks](#)

Configuring custom HTTP servers

After you create a middleware server representation of a custom HTTP server in the administrative console, you must update a few configuration settings so that they are defined specifically for your custom HTTP server installation.

Before you begin

Federate the node that is running the custom HTTP server and create a representation of the server in the administrative console.

About this task

To perform server operations on your server from the administrative console, you must configure the server operations for the custom HTTP server middleware server representation.

Procedure

Configure server operations for the custom HTTP server to include the user name and password.

- a. In the administrative console, click **Servers > Other middleware servers > Custom HTTP servers > custom_http_server > Server operations**.
- b. Edit the start and stop server operations.
- c. Define the location of the process ID (PID) file on the server. Create the PID file name environment variable. The PID file name specifies the location of the PID file where the server operation runs. To define the PID file name variable, click **Servers > All servers > custom_http_server > Variables > New or Environment > WebSphere® Variables > New**. Enter the same name for the variable that you entered in the server operations panel, and for the value, enter the fully qualified or variable qualified location of the PID file.

What to do next

- To use this configuration when you are configuring your other custom HTTP servers, create a server template of your current custom HTTP server.
- Use the external log viewing service to view the log files for your custom HTTP server from the administrative console.
- Use the external configuration service to edit the configuration files for your custom HTTP server from the administrative console.
- After you create representations of all the custom HTTP servers, you can add the representations to a dynamic cluster.

Parent topic: [Adding assisted life-cycle middleware servers](#)

Previous topic: [Federating middleware nodes](#)

Next topic: [Configuring middleware server operations](#)

Next topic: [Creating middleware server templates](#)

Next topic: [Configuring the external log viewing service](#)

Next topic: [Creating dynamic clusters](#)

Parent topic: [Adding assisted life-cycle middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

[Supported middleware server types](#)

Related tasks:

[Adding middleware servers to configurations](#)

Related reference:

[Middleware server creation administrative tasks](#)

Configuring external WebSphere application servers

You can create middleware server representations of WebSphere® application servers that are in other cells.

Before you begin

- If you have a large topology of WebSphere application servers, you can use the migration toolkit to create the representations of these servers in the environment. See [Migrating large topologies of external WebSphere application servers](#) for more information.
- Federate the node that is running the WebSphere application server and create a representation of the server in the administrative console.

About this task

External WebSphere application servers are application servers that reside outside of your WebSphere Virtual Enterprise cell. For example, you might have two cells: one cell that is running WebSphere Application Server Version 5.1 application servers, and another cell that is running WebSphere Virtual Enterprise. In this scenario, you can install the middleware agent on the Version 5.1 nodes in the other cell, federate these nodes into your WebSphere Virtual Enterprise cell, and create representations of your external WebSphere application servers in your WebSphere Virtual Enterprise administrative console. Using external WebSphere application servers can be useful in a migration scenario because you can continue to route traffic to servers that are running an older version with the ODR while you work on migrating your servers and applications to the latest version. However, external WebSphere application servers are assisted life-cycle servers and therefore do not receive the full life-cycle management that is provided when you create application servers directly in your WebSphere Virtual Enterprise cell.

When you configure the external WebSphere application servers, if you installed WebSphere Application Server in a location other than the default directory, update the WebSphere Variables so that the installation directory is correct. To perform server operations on your server from the administrative console, you must configure the server operations for the WebSphere Application Server middleware server representation.

Procedure

1. Update the WebSphere variables so that they match the settings on the node that is running the WebSphere Application Server. In the administrative console, click **Environment > WebSphere Variables**. Edit the following variables:

WAS51_HOME

Specifies the location of the WebSphere Application Server Version 5.1 installation.
Default values are set at the cell scope level:

Windows	c:\Program Files\IBM\WebSphere\AppServer		
AIX	HP-UX	Solaris	/opt/IBM/WebSphere/AppServer

If the home directory is different on different nodes in your configuration, select the node scope and create the WAS51_HOME variable for the specific node.

WAS6_HOME

Specifies the location of the WebSphere Application Server Version 6 installation location.
Default values are set at the cell scope level:

Windows	c:\Program Files\IBM\WebSphere\profiles\AppServer		
AIX	HP-UX	Solaris	opt/IBM/WebSphere/profiles/AppServer

If the home directory is different on different nodes in your configuration, select the node scope and create the WAS6_HOME variable for the specific node.

WAS6_PROFILE_NAME

Specifies the name of the profile in which the external WebSphere Application Server server resides on its physical computer.

2. Update server operations for the WebSphere Application Server to include the user name and password.
 - a. In the administrative console, click **Servers > Other middleware servers > External WebSphere Application Server > WebSphere_Application_Server > Server operations**.
 - b. Edit the start and stop server operations to include the user name and password that is required to run these commands.

What to do next

- To easily configure your remaining external WebSphere application servers, create a server template of your current external WebSphere application server.
- To view the log files for your external WebSphere application server in the administrative console, use the external log viewing service. This service is enabled by default. The following directories are configured by default:
 - `${INSTALL_ROOT}/logs/${WAS_SERVER_NAME}`
- To edit configuration files for your external WebSphere Application Server, use the external configuration service. The following files are included by default:

- `{WAS_SERVER_HOME}/var/conf/server.xml`
- After you create representations of all the external WebSphere application servers, you can add the representations to a dynamic cluster.

Parent topic: [Adding middleware servers to configurations](#)

Parent topic: [Adding assisted life-cycle middleware servers](#)

Previous topic: [Federating middleware nodes](#)

Next topic: [Creating middleware server templates](#)

Next topic: [Viewing middleware server configuration files](#)

Next topic: [Configuring the external log viewing service](#)

Next topic: [Creating dynamic clusters](#)

Parent topic: [Adding assisted life-cycle middleware servers](#)

Related concepts:

[Migration toolkit](#)

Related tasks:

[Migrating large topologies of external WebSphere application servers](#)

[Configuring middleware server operations](#)

[Configuring WebSphere variables](#)

Related reference:

[Middleware server creation administrative tasks](#)

Managing middleware servers

To define middleware servers, you can use middleware server templates and middleware descriptors. To manage middleware servers that are defined, you can view your middleware server log files and configuration files in the administrative console and configure server operations to run on your middleware servers.

Before you begin

To configure middleware server templates, middleware server operations, the external configuration service, or the external log viewing service, you must create the middleware server. See [Adding middleware servers to configurations](#) for more information.

You can configure middleware descriptors before or after you add your middleware servers.

Procedure

- Configure middleware descriptors. Middleware descriptors provide information about your middleware platform, such as the installation location and the interval at which an installation can be discovered, if discovery applies to your specific platform. See [Configuring middleware descriptors](#) for more information.
- Define middleware server templates. By creating a middleware server template, you copy all of the configuration data for that server so that you can create other servers from the template. If you have many middleware servers with a similar configuration, middleware server templates can make this configuration easier. See [Creating middleware server templates](#)
- Configure middleware server operations. You can use middleware server operations to run executable files on your middleware servers from the administrative console. Middleware operations can perform tasks such as enabling or disabling tracing, starting or stopping servers, or querying the running state of the server. See [Configuring middleware server operations](#)
- View and edit configuration files for your middleware servers. You can specify the path for the configuration files that you want to view or edit in the administrative console. See [Viewing middleware server configuration files](#) for more information.
- Configure the external log viewing service. With the external log viewing service, you can view the log files from your middleware servers in the administrative console. See [Configuring the external log viewing service](#) for more information.

[Creating middleware server templates](#)

By creating a middleware server template, you save a copy of server configuration data that you can use as a starting point when you create a new server.

[Configuring middleware descriptors](#)

Middleware descriptors provide information about different middleware platform types including middleware discovery intervals, enablement, and installation information.

[Configuring middleware server operations](#)

You can use server operations to run executable files on your middleware servers from the administrative console.

[Viewing middleware server configuration files](#)

You can view and edit configuration files for your middleware servers in the administrative console with the external configuration service.

[Configuring the external log viewing service](#)

To view the log files for externally created middleware servers, configure the external log viewing service. By using this service, you can view the log files for servers with the administrative console instead of viewing the log files on each node.

[Configuring middleware descriptors](#)

Middleware descriptors provide information about different middleware platform types including middleware discovery intervals, enablement, and installation information.

[Creating middleware server templates](#)

By creating a middleware server template, you save a copy of server configuration data that you can use as a starting point when you create a new server.

[Configuring middleware server operations](#)

You can use server operations to run executable files on your middleware servers from the administrative console.

[Viewing middleware server configuration files](#)

You can view and edit configuration files for your middleware servers in the administrative console with the external configuration service.

[Configuring the external log viewing service](#)

To view the log files for externally created middleware servers, configure the external log viewing service. By using this service, you can view the log files for servers with the administrative console instead of viewing the log files on each node.

Parent topic: [Adding middleware servers to configurations](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Adding complete life-cycle middleware servers](#)

[Adding assisted life-cycle middleware servers](#)

Configuring middleware descriptors

Middleware descriptors provide information about different middleware platform types including middleware discovery intervals, enablement, and installation information.

Before you begin

If you want to add or change installation information to the middleware descriptor, you must know the installation paths of the middleware software that you want to configure.

About this task

You can edit the following middleware descriptors that are provided with the product:

- Application server
- Apache HTTP server
- JBoss server
- PHP server
- Apache Tomcat server
- BEA WebLogic server
- Custom HTTP server
- WebSphere® Application Server Community Edition server

Procedure

1. Edit middleware platform descriptor properties. In the administrative console, click **System administration > Middleware descriptors > *middleware_platform_name***.

If you are using an Apache, WebSphere Application Server Community Edition, or PHP runtime: Configure the interval at which the discovery function of the middleware platform runs. Supply a **Discovery interval** and units for this discovery interval in seconds, minutes, hours, or days. To disable middleware discovery, enter -1 in the **Discovery interval** field. Click **OK** to save these changes. This automated discovery is supported for Apache, WebSphere Application Server Community Edition, and PHP runtimes only.

2. Edit the middleware platform version details. Click the version that you want to edit. The version properties vary depending on the middleware platform type, for example, for some types, you can edit the session affinity descriptor.

For Apache Web servers, WebSphere Application Server Community Edition servers, and PHP servers:

If you installed the phpRuntime, wasceRuntime, or apacheWebServerRuntime to a location other than the default path, you can edit the installation location. You can also define multiple installation locations. Do not include installation locations for server installations for which discovery is not preferred, such as centralized installation manager installations of WebSphere Application Server Community Edition. If you change or remove an installation location from which servers have previously been discovered, changes to those servers are no longer discovered.

Results

When enabled, automated discovery can run for Apache, WebSphere Application Server Community Edition, and PHP runtimes that you configured.

Parent topic: [Managing middleware servers](#)

Parent topic: [Managing middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Adding middleware servers to configurations](#)

[Configuring passive HTTP session affinity in the on demand router](#)

[Federating middleware nodes](#)

Creating middleware server templates

By creating a middleware server template, you save a copy of server configuration data that you can use as a starting point when you create a new server.

Before you begin

Create a middleware server from which you want to create a template. See [Adding middleware servers to configurations](#) for more information.

About this task

When you create a server template, you copy all of the configuration data for that server so that you can create other servers from the template. If your middleware servers each have a similar configuration, then a middleware server template can make creating additional servers easier.

Procedure

1. Navigate to the server templates collection page. In the administrative console, click **All servers > Templates**. The list of existing server templates is displayed.
2. To create a new middleware server template, click **New**. Choose the type of server template that you want to create.
3. Select an existing server from which to create the server template. Select a server and click **OK**.
4. Specify a name and description for the server template.
5. Click **OK**. To save the template to the master configuration, click **Save**.

Results

A server template is created based on the configuration of the existing server.

What to do next

When you create a new middleware server that is the same type of server that you selected to create your template, you can choose the template as the configuration starting point of the new server.

Parent topic: [Managing middleware servers](#)

Parent topic: [Managing middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Adding middleware servers to configurations](#)

Related reference:

[Middleware server template administrative tasks](#)

Configuring middleware server operations

You can use server operations to run executable files on your middleware servers from the administrative console.

Before you begin

- Create and configure the middleware server. See [Adding middleware servers to configurations](#) for more information.
- Create a Java™ or non-Java executable file that performs the operation that you want to run on your middleware server. You specify this file when you create the server operation.

About this task

Server operations consist of Java or non-Java process definitions that you can define to run on middleware servers. You can create these operations to perform tasks such as enabling or disabling tracing, starting or stopping applications, and querying the running state of the server.

Procedure

1. Navigate to the server operations wizard. In the administrative console, click **Servers > All servers > *middleware_server* > Server operations > New**.
2. Specify the properties for your server operation. The properties are different if you choose a Java or non-Java server operation.

For operations that run non-Java and Java executable files:

- Specify a name for the action, the name of the executable file, and any arguments to pass to the executable file.
- Specify variables for the user name and password, and specify the actual values to pass for the user name and password at the time the executable file runs.
- Specify the operating systems on which the executable files can run.
- Specify a working directory, which is where to run the executable file.

For operations that run Java executable files, you must also specify the following properties:

- Specify the location of the Java executable directory.
 - Specify the type of target: a Java class, or an executable Java archive (JAR) file.
 - Specify the environment variable that stores the process ID (PID) file name. The PID file name is stored in a WebSphere variable. Remember the name of the variable so that you can create the variable in a later step.
3. Save the server operation.
 4. Create the PID file name environment variable. The PID file name specifies the location of the PID file where the server operation runs. To define the PID file name variable, click **Servers > All servers > *middleware_server* > Variables > New** or **Environment > WebSphere Variables > New**. Enter the same name for the variable that you entered in the server operations panel, and for the value, enter the fully qualified or variable qualified location of the PID file.
 5. To run the server operation, select the operation from the list and click **Execute**.

Parent topic: [Managing middleware servers](#)

Parent topic: [Managing middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Adding middleware servers to configurations](#)

Related reference:

[Middleware server management administrative tasks](#)

Viewing middleware server configuration files

You can view and edit configuration files for your middleware servers in the administrative console with the external configuration service.

Before you begin

- Create and configure the middleware server in the WebSphere® Virtual Enterprise environment. See [Adding middleware servers to configurations](#) for more information.
- You must know the fully qualified path name of the configuration file that you want to view on the middleware server. This configuration file must be less than 10 megabytes.

About this task

By configuring the external configuration service, you can view and edit the configuration files for middleware servers from the administrative console.

Procedure

1. Enable the external configuration service for the middleware server. In the administrative console, click **Servers > Other middleware servers > *server_name* > External configuration service**. Select **Enable external configuration service**. Click **OK**.
2. Specify the configuration files that you want to edit or view in the administrative console. Depending on the server type, one or more configuration files might be listed by default. To add a file to this list, enter the fully qualified path name of the file, and specify if this file is writable. Click **Add** to add the file to the list. To change the file permission for a file, select the file and click **Toggle privileges**.
3. View and edit the configuration files. In the administrative console, click **Servers > Other middleware servers > *server_name***. Click the **External configuration** tab. Select the configuration file and click **Retrieve**. When you complete editing the file, click **Apply** to save your changes.

What to do next

View the log files for your middleware servers.

Parent topic: [Managing middleware servers](#)

Previous topic: [Adding middleware servers to configurations](#)

Next topic: [Configuring the external log viewing service](#)

Parent topic: [Managing middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

Configuring the external log viewing service

To view the log files for externally created middleware servers, configure the external log viewing service. By using this service, you can view the log files for servers with the administrative console instead of viewing the log files on each node.

Before you begin

- Create and configure your middleware servers.
- You must know the path to the log files for the server.
- To add new log files to the service, you need administrator administrative privileges. If you want to view existing log file output, you need configurator administrative privileges.

About this task

By configuring the external log viewing service, you can view logs for your other middleware servers in the administrative console. By default, some log files are configured to be viewed in the external log viewing service with the server template for your middleware server type. You can add to this default list of log files.

Restriction: You cannot use the graphical logs and trace settings that are included in WebSphere® Application Server for other middleware server types, including middleware agents. Instead, use the external log viewing service to view the logs for these server types.

Procedure

1. Navigate to the external log service panel for your middleware server. In the administrative console, click **Servers > Other middleware servers > server_type > server_name > External log service**.
2. To enable the remote log viewing services, select **Enable remote log viewing services**.
3. List all the directories that contain log files that you want to view using the remote log viewer. Type a directory name, and click **Add**. The list contains all the paths to the log files. You can use variables such as `${USER_INSTALL_ROOT}` in your file path.
4. Click **OK**. To save your changes to the master configuration, click **Save**.
5. View the log files. In the administrative console, click **Servers > Other middleware servers > server_type > server_name**. Click the **Log viewer** tab. Select a log file to view, and click **Retrieve**.

Results

You can view the log file to troubleshoot any issues with your middleware server.

Parent topic: [Managing middleware servers](#)

Previous topic: [Adding middleware servers to configurations](#)

Parent topic: [Managing middleware servers](#)

Related concepts:

[Middleware nodes and servers](#)

[Middleware agent](#)

Related tasks:

[Creating middleware server templates](#)

Related reference:

[Middleware server creation administrative tasks](#)

Deploying applications with defined service levels

By deploying applications to your middleware servers and dynamic clusters and defining service policies on those applications, you can meet the performance goals for your environment.

Before you begin

- Configure the on demand router. See [Creating and configuring ODRs](#) for more information.
- Create the servers and dynamic clusters for your environment, specifically the servers to which you want to deploy applications. See [Creating dynamic clusters](#) and [Adding middleware servers to configurations](#) for more information.
- Plan the classification and handling of traffic for your application.

About this task

Service policies provide the ability to differentiate applications according to levels of importance and target values that can be prioritized to meet specific goals.

Procedure

1. Deploy the application. Different processes are involved for the different application types:
 - **PHP Hypertext Preprocessor (PHP) applications:** You can deploy PHP applications to PHP deployment targets, including PHP servers and PHP dynamic clusters. See [Deploying PHP applications](#) for more information.
 - **WebSphere Application Server Community Edition applications:** You can deploy Java 2 Platform, Enterprise Edition (J2EE) applications and Geronimo modules with Geronimo artifacts to WebSphere Application Server Community Edition deployment targets, including servers and dynamic clusters. See [Deploying WebSphere Application Server Community Edition applications](#) for more information.
 - **Unmanaged Web applications:** Unmanaged Web applications are installed and configured outside of the product environment, typically on assisted life cycle middleware servers. Configure unmanaged Web applications so that you can route HTTP requests to assisted life cycle servers. By providing some basic information about the application, including context roots, virtual hosts, and servers, the ODR can route HTTP requests to these applications. See [Deploying unmanaged Web applications](#) for more information.
 - **Enterprise applications:** You can deploy enterprise applications to dynamic clusters that are running WebSphere Application Server application servers. See [Deploying enterprise applications](#) for more information.
2. Define service levels with service policies. A service policy is a user-defined business goal, and correlates to transaction and work class components. For more information, read about defining a service policy.
3. Classify application requests with service policy work classes. In the administrative console, click **Applications > All applications > application_name**. Click the **Service Policies** tab. Work classes contain Universal Resource Identifiers (URI), Web services, Enterprise JavaBeans (EJB), or Java Message Service (JMS) patterns and conditional rules for associating requests to transaction classes. Transaction classes are then associated with service policies. For more information, see [Routing and service policies](#).

Alternatively, you can use the `workclassoperations.py` script to define and modify the settings of work classes. For more information, see [workclassoperations.py script](#).

4. Start the application.
 - **Enterprise applications:** In the administrative console, click **Applications > Enterprise applications**. Select the application, and click **Start**.
 - **PHP Hypertext Preprocessor (PHP) applications:** In the administrative console, click **Applications > All applications**. Select the PHP application that you want to start. Choose the **Start** action and click **Submit Action**. This action starts all of the PHP servers that are associated with this application. To start the servers individually, click **Servers > Other middleware servers > PHP servers**.
 - **WebSphere Application Server Community Edition applications:** The application displays as started in the administrative console when the middleware server on which the application is deployed starts.
 - **Unmanaged Web applications:** The application displays as started when the server on which the application is deployed starts.

Attention: If the server or middleware agent stops, the unmanaged Web application might still show as started or partially deployed in the administrative console.

What to do next

- Use reporting to view statistics and performance of your applications, dynamic clusters, servers, and service policies.
- To achieve a healthy product environment, configure health policies. Health policies are similar to service policies, except that health policies are based on a health goal for the environment.
- To ensure positive application performance, such as applying interruption-free application updates, you can use the application edition manager to install new editions of your application.

[Deploying enterprise applications](#)

By deploying applications to dynamic clusters, requests to the application can be managed autonomically. The dynamic cluster can expand and contract with application server virtualization. By deploying applications to dynamic clusters, you enable application virtualization.

[Deploying PHP applications](#)

You can use the various qualities of service of WebSphere Virtual Enterprise product to manage PHP-based applications in addition to Java Platform, Enterprise Edition 5 (Java EE 5) applications.

[Deploying unmanaged Web applications](#)

You can create representations of unmanaged Web applications so that the on demand router (ODR) can route requests to applications that are installed outside of the product environment to assisted life cycle middleware servers.

[Deploying WebSphere Application Server Community Edition applications](#)

From within the WebSphere Virtual Enterprise administrative console, you can install managed WebSphere Application Server Community Edition Version 2.0 applications to servers and dynamic clusters.

[Deploying enterprise applications](#)

By deploying applications to dynamic clusters, requests to the application can be managed autonomically. The dynamic cluster can expand and contract with application server virtualization. By deploying applications to dynamic clusters, you enable application virtualization.

[Deploying unmanaged Web applications](#)

You can create representations of unmanaged Web applications so that the on demand router (ODR) can route requests to applications that are installed outside of the product environment to assisted life cycle middleware servers.

[Deploying PHP applications](#)

You can use the various qualities of service of WebSphere Virtual Enterprise product to manage PHP-based applications in addition to Java Platform, Enterprise Edition 5 (Java EE 5) applications.

[Deploying WebSphere Application Server Community Edition applications](#)

From within the WebSphere Virtual Enterprise administrative console, you can install managed WebSphere Application Server Community Edition Version 2.0 applications to servers and dynamic clusters.

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Next topic: [Defining a service policy](#)

Next topic: [Configuring health management](#)

Next topic: [Deploying and managing application editions without loss of service](#)

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Related concepts:

[Operational policies](#)

Related tasks:

[Monitoring operations](#)

[Creating and managing reports](#)

Related reference:

[Middleware application administrative tasks](#)

Deploying enterprise applications

By deploying applications to dynamic clusters, requests to the application can be managed autonomically. The dynamic cluster can expand and contract with application server virtualization. By deploying applications to dynamic clusters, you enable application virtualization.

Before you begin

Verify that you have WebSphere® Virtual Enterprise running and have access to an application for installation. Configure a dynamic cluster. You must know the specific cell and dynamic cluster on which you want to deploy.

About this task

When you deploy your application to a dynamic cluster that is in automatic mode, the product can automatically control the size and placement of dynamic clusters on the nodes, according to the business goals of your application.

Procedure

1. Deploy the application to your dynamic cluster. In the administrative console, click **Applications > Install new application** or **Applications > Install new middleware application > Java™ Platform, Enterprise Edition (Java EE) type**.

Restriction: The application name must be unique among all the applications that are deployed in the cell.

For example, you cannot deploy a PHP application and a Java EE application that have the same name.

2. On the **Map modules to application servers** panel in the wizard, map your application to the cell and dynamic cluster name, or the deployment target, that is specific to your dynamic cluster.
3. Complete the wizard panels and click **Finish**.

What to do next

Define a service policy for the newly deployed application.

Parent topic: [Deploying applications with defined service levels](#)

Related concepts:

[Operational policies](#)

Related tasks:

[Defining a service policy](#)

[Creating dynamic clusters](#)

[Deploying applications with defined service levels](#)

[Deploying and administering J2EE applications](#)

Related reference:

[servicepolicy.py script](#)

Deploying unmanaged Web applications

You can create representations of unmanaged Web applications so that the on demand router (ODR) can route requests to applications that are installed outside of the product environment to assisted life cycle middleware servers.

Before you begin

- Create and configure your middleware servers and dynamic clusters. See [Adding middleware servers to configurations](#) and [Creating dynamic clusters](#) for more information.
- Deploy the application to your assisted life cycle middleware servers, using the steps that are specified for the software that your middleware server runs. Be sure that you know or have access to the deployment information for the application, including the module name, context root, virtual host name, and the server or cluster names on which you want to deploy the application.

About this task

To route HTTP requests to applications that are deployed on other middleware servers and classify these requests to service policies, the runtime environment needs information about the application. By creating an *unmanaged Web application*, you create a representation of the application that is installed on the middleware server that includes this information. WebSphere Virtual Enterprise does not manage the life cycle of the application in any way, but can route requests to the application. The status of unmanaged Web applications is directly linked to the status of the server to which the unmanaged Web applications are deployed. The ODR can only route HTTP requests to these applications.

An unmanaged application for WebSphere Application Server Community Edition is a representation of an application that has been installed in the WebSphere Application Server Community Edition Version 1 console, specifically. With *middleware discovery*, an unmanaged WebSphere Application Server Community Edition application can also be a representation of a discovered application. Middleware discovery locates existing installations of WebSphere Application Server Community Edition servers and their installed applications, and creates the corresponding configuration to include servers and applications in the WebSphere Virtual Enterprise cell.

Procedure

1. Navigate to the unmanaged Web application installation wizard. In the administrative console, click **Applications > Install New Middleware Application > Unmanaged Web Applications > New**.
2. Enter the general properties for the application. Specify a name for the application that is unique among all of the applications in the cell, and edition information about the application.

About application editions for unmanaged Web applications: Because the application is unmanaged, you cannot perform edition control commands such as rolling out or validating new editions. However, if you install another edition on the externally created servers, then you can create a new edition of the application in the administrative console by creating a new representation of the unmanaged Web application with a different edition name. You can then specify routing policies to control how to route the requests to the multiple editions.

3. Define the deployment properties of the application. Specify module, context root, and virtual host information for the application. Choose the servers and clusters on which the application is deployed by selecting deployment targets. You can filter the list of deployment targets by only clusters, only servers, or by entering a name. Click **Add** or **Remove** to specify which servers and clusters run the application. Click **Apply** to add the module and deployment targets to the module list.

The dynamic cluster to which you deploy an unmanaged Web application must be homogenous, which means the same application must be deployed to all the members of the dynamic cluster, and all members must be running the same middleware software, including the same version. When you add a cluster member that does not contain the same application as the cluster, the status of the application becomes partially deployed. The application is not automatically deployed to the newly added cluster member.

4. Confirm the new unmanaged Web application representation, and save the changes.
5. Start the application. With middleware applications, applications display as started in the administrative console when the middleware server on which the application runs is started. To start middleware servers from the administrative console, see [Configuring middleware server operations](#).

Attention: If the server or middleware agent stops, the application might still show as started or partially deployed in the administrative console.

What to do next

Define service policies and routing policies for the application.

Parent topic: [Deploying applications with defined service levels](#)

Next topic: [Defining a service policy](#)

Parent topic: [Deploying applications with defined service levels](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Deploying PHP applications](#)

[Deploying WebSphere Application Server Community Edition applications](#)

[Configuring middleware server operations](#)

[Adding assisted life-cycle middleware servers](#)

Related reference:

[Middleware application administrative tasks](#)

[Middleware server creation administrative tasks](#)

Deploying PHP applications

You can use the various qualities of service of WebSphere® Virtual Enterprise product to manage PHP-based applications in addition to Java™ Platform, Enterprise Edition 5 (Java EE 5) applications.

Before you begin

- Create a PHP deployment target by creating either a PHP dynamic cluster or a PHP server. Verify that you can start the deployment targets. See [Creating PHP servers and PHP dynamic clusters](#) for more information.
- Prepare your PHP application to be deployed. Your PHP application consists of an archive file and optional setup or clean-up scripts. The supported archive file types are zip, tar, tar.gz, tar.tgz and jar. The optional scripts run for each server on which the application is deployed. The setup script runs during the application installation, and the clean-up script runs when you uninstall the application to remove any residual artifacts. The setup scripts can include commands such as setting file permissions and creating or deleting directories.

About this task

You can experience the same qualities of service that the product provides for Java EE 5 applications for PHP applications.

Procedure

1. Navigate to the PHP application wizard. In the administrative console, click **Applications > Install new middleware application**. Choose the PHP application type, and click **Next**.
2. Specify the location of the application archive, setup, and clean-up scripts.
3. Specify installation options, including the application name, edition, and edition description. The application name must be unique among all of the applications that are deployed in the cell, including applications that are other types, such as Java EE 5 or unmanaged Web applications.
4. Define the deployment properties of the application.
 - a. Specify the context root and virtual host information for the application.
 - b. Choose the servers and clusters on which the application runs by selecting deployment targets. Choose PHP servers or PHP dynamic clusters. You can filter the list of deployment targets by only clusters, only servers, or by entering a name. Click **Add** or **Remove** to specify which servers and clusters run the application.
 - c. For an unmanaged Web application with multiple modules, click **Apply** to add the module and deployment targets to the module list.
5. Confirm the new PHP application, and save the changes. The application archive expands and the setup script runs.
6. Start the application. With middleware applications, applications start when you start the PHP server in the administrative console. Click **Servers > Other middleware servers > PHP servers**. Select the servers that you chose as deployment targets, and click **Start**. You can also start the application from **Applications > All applications** in the administrative console. However, doing so from this panel will start all the servers associated with this application. The same rule applies if you stop PHP applications from this panel.

Results

Your PHP application runs on the defined PHP deployment targets. If you deployed the application to a PHP dynamic cluster, then the dynamic cluster can add or remove PHP servers as needed.

What to do next

Define service policies and routing policies for the application. All application edition manager functions are available. When you roll out a new edition, the clean-up script from the first edition runs, the application archive of the new edition is expanded, and the setup script of the new edition runs.

Restriction: Only one active edition of a PHP application is supported on a node. If you have multiple active editions of the same PHP application, ensure that the editions are not deployed to servers that are on the same node.

Parent topic: [Deploying applications with defined service levels](#)

Related tasks:

[Deploying applications with defined service levels](#)

[Defining a service policy](#)

[Deploying and managing application editions without loss of service](#)

[Creating PHP servers and PHP dynamic clusters](#)

Related reference:

[Middleware application administrative tasks](#)

[Middleware server creation administrative tasks](#)

Deploying WebSphere Application Server Community Edition applications

From within the WebSphere® Virtual Enterprise administrative console, you can install managed WebSphere Application Server Community Edition Version 2.0 applications to servers and dynamic clusters.

Before you begin

- Create a WebSphere Application Server Community Edition deployment target by creating a complete life cycle WebSphere Application Server Community Edition server or dynamic cluster. Verify that you can start the deployment targets. See [Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#) for more information.
- Prepare your WebSphere Application Server Community Edition application to be deployed. The application consists of an archive file and an external deployment plan, which are both deployed to the WebSphere Application Server Community Edition server.

Important: With managed WebSphere Application Server Community Edition applications, you must complete all deployment actions from within the WebSphere Virtual Enterprise administrative console only. Use the WebSphere Application Server Community Edition console only for functions that are not supported by the product.

About this task

Deploy a WebSphere Application Server Community Edition application to experience the same qualities of service that WebSphere Virtual Enterprise provides for Java™ Platform, Enterprise Edition 5 (Java EE 5) applications and PHP applications.

Procedure

1. Open the WebSphere Application Server Community Edition application wizard. In the administrative console, click **Applications > Install new middleware application**. Select **WebSphere Application Server Community Edition** as the application type, and click **Next**.
2. Specify either the local or remote location of the application archive file and external deployment plan, and click **Next**.
3. Specify the name, edition, and edition description of the application. The application name must be unique among all of the applications that are deployed in the cell, including applications that are other types, such as Java EE 5 or unmanaged Web applications.
4. Select the servers and clusters on which the application runs by selecting the deployment targets. You can filter the list of deployment targets by only clusters, only servers, or by entering a name. Click **Add** or **Remove** to specify which servers and clusters run the application.
5. Specify the location of the virtual host for the Web modules.
6. Review the confirmation details of the WebSphere Application Server Community Edition application, and save the changes.
7. Start the application. The application displays as started in the administrative console when the middleware server on which the application is deployed starts. To start middleware servers from the administrative console, see [Configuring middleware server operations](#).

Attention: If the server or middleware agent stops, the application might still show as started or partially deployed in the administrative console.

Results

Your WebSphere Application Server Community Edition application runs on the defined deployment targets.

What to do next

Define service policies and routing policies for the application. All application edition manager functions are available.

Parent topic: [Deploying applications with defined service levels](#)

Related tasks:

[Deploying applications with defined service levels](#)

[Defining a service policy](#)

[Deploying and managing application editions without loss of service](#)

[Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#)

Related reference:

[Middleware application administrative tasks](#)

[Middleware server creation administrative tasks](#)

Routing and service policies

Two types of policies are applied to a request: routing and service. You can create routing policies for HTTP and SOAP requests, and you can create service policies for HTTP, IIOP, SOAP, JMS, and SIP requests. Additionally, work classes can contain classification rules for both policy types with the exception of JMS. Classification rules are not supported for JMS work classes.

Valid routing policies

Table 1. Routing policies

Routing policy	Description
permit: <i>application_name</i>	<i>application_name</i> is the application name to route to with an optional edition specifier.
permitMM: <i>application_name</i>	<i>application_name</i> is the application name to route to with an optional edition specifier. Routing in this way allows the request to continue as normal. Note that the server must be in maintenance mode.
permitsticky: <i>application_name</i>	<p>The permitsticky routing policy is the same as the permit routing policy, except that the on demand router (ODR) also maintains client-to-server affinity for any future requests that come from the same client. In this case, the ODR adds a SET-COOKIE header to the response before it sends the response to the client.</p> <p>The permitsticky action means that the ODR actively establishes affinity between the client and server if affinity was not already established by the application. The ODR accomplishes this by adding a SET-COOKIE: WSJSESSIONID=xx:serverID; path=webModuleContextRoot to the response if:</p> <ul style="list-style-type: none"> • The response does not already have a SET-COOKIE that will establish server affinity, and • The corresponding request does not indicate that server affinity had already been established. <p>The serverID is the server identifier, and is also referred to as the clone ID. The webModuleContextRoot is the context root of the Web module to which the request was mapped.</p>
permitstickyMM: <i>application_name</i>	This routing policy is the same as the permit routing policy, except that the ODR also maintains client to server affinity for any future requests that come from the same client. In this case, the ODR adds a SET-COOKIE header to the response before it sends the response to the client. Note that the server must be in maintenance mode.
reject: <i>HTTP_error_code</i>	This routing policy causes the ODR to reject the request and return the specified HTTP error code. For example, reject:503 returns a 503 Service is unavailable error.
reject: <i>URL</i>	With this routing policy, the ODR redirects the request to the specified URL. The URL has the pattern of protocol://URI. An example of a valid URL is http://w3.ibm.com.

Valid service policies

The valid service policies are the list of transaction class names. The transaction class refers to a single service class.

[Defining a service policy](#)

You can define service policies and, for most kinds of work requests, work classes to categorize and prioritize work requests. A service policy consists of a user-defined performance goal and an importance level, in some cases.

[Configuring service policies without response time goals](#)

Configure service policies without response time goals when multiple service policies are configured and the response time goals cannot be accurately determined for any of the service policies, or when there might be occasional long-running or stuck requests.

Subexpression builder operands for routing and service policies

Operands and their associated protocols are supported from the initial installation, and can be used in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping. An expression can be built from several sub-expressions using logical operators.

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Related tasks:

[Defining a service policy](#)

[Setting maintenance mode](#)

[Activating concurrent application editions](#)

[Validating an edition](#)

Related reference:

[Subexpression builder operands for routing and service policies](#)

Related information:

[Rules for ODR routing policy administrative tasks](#)

[Rules for ODR service policy administrative tasks](#)

Defining a service policy

You can define service policies and, for most kinds of work requests, work classes to categorize and prioritize work requests. A service policy consists of a user-defined performance goal and an importance level, in some cases.

Before you begin

- To create, modify, or remove service policies and transaction classes, you must have administrator or configurator administrative privileges. To modify rules through the rule builder, you must have administrator privileges. See [Administrative roles and privileges](#) for details.

About this task

Service policies are related to work requests through transaction classes. Each work request belongs to exactly one transaction class, and each transaction class belongs to exactly one service policy. For most kinds of work requests, work classes are used to map incoming requests to transaction classes. Each work class is attached to a Java™ Platform, Enterprise Edition (Java EE) application and a basic request feature; URI prefix for HTTP, method name for IIOP, and bus+destination for Java Message Service (JMS). Each work class specifies how the relevant requests are classified into transaction classes. For generic server clusters and for SIP, work classes are not used; instead, the rules for classifying requests to transaction classes are configured on the ODRs.

You can use the service policy custom properties to provide service policy alerting for persistent service policy violations on a transaction class basis. See [Service policy custom properties](#).

For SIP over UDP traffic, you must enable the admission control for CPU overload protection to prevent retransmissions from occurring because of CPU overload. When using admission control for CPU overload protection for SIP, the discretionary type of goal must NOT be used. Only the average response time or percentile response time goals should be used. The response time threshold specified in the goal should be well under the value of the client's T1 timer (which defaults to 500 milliseconds). The rejection average response time threshold (the value derived from the goal's response time threshold and the rejection policy configured on the ARFM control panel, should be less than the client's T1 timer. See [Configuring the autonomic request flow manager](#) for instructions on enabling the admission control for CPU overload protection.

Restriction: When dialog/session orientation is enabled for HTTP or SIP, a service policy cannot apply to messages that are part of pre-existing dialogs or sessions, *and* messages that are NOT part of pre-existing dialogs or sessions.

Using the *service time*, or response time of a single request or small number of requests on a non-loaded or lightly loaded system, that is, how long a single request takes on a non-loaded system, then constructing service policies where the response times are less than that will never result in additional instances being started (either average or percentile response time goals). The system will determine that starting additional instances will not improve the capability of meeting the goal. For percentile goals, ARFM and APC are very sensitive to the relationship between the parameters in question: the Response Time Target (RTT), or the Goal Value, and Percentile Threshold (PCT), or the Goal Percentage.

Following are some sample ranges starting with the response time, increasing to the RTT of 2, 3, 4 times (and so on) of the single request service time. Low and high end PCTs are then provided. These can vary to some degree from application to application, but these ranges are given as a starting point for fine-tuning your service policies for you specific needs For percentile goals, the ARFM and the APC are aware of the relationship between the parameters, RTT, or the goal value, and PCT, or the goal percentage.

- RTT of 2 times the service time:
 - A tight percentile goal or PCT of 75% completing within the response time specified (RTT)
 - A loose percentile goal or PCT of 50% completing within the response time specified (RTT)
- RTT of 3 times the service time: PCTs from 88% to 65%
- RTT of 4 times the service time: PCTs from 94% to 76%
- RTT of 5 times the service time: PCTs from 97% to 83%
- RTT of 6 times service time: PCTs from 99% to 88%
- RTT of 7 times the service time: PCTs from 99% to 92%
- RTT of 8 times the service time: PCTs from 99% to 94%

Procedure

1. From the administrative console click **Operational policies > Service Policy**. You can select an existing service policy to edit, or click **New** to create a service policy. To edit an existing service policy, click the service policy name.
2. Create a name, description, and a goal type for your new service policy. The goal type can be either discretionary, average response time, percentile response time, or completion time:
 - A discretionary goal is the default, and indicates work that does not have significant value. As a result, work of this type can see a degradation in performance when resources are constrained.
 - Average response time goals are indicative of work with a higher priority than discretionary. The average response time goal is assigned a specific time goal.
 - Percentile response time goals are another measure for work with a higher priority than discretionary. The percentile response goals are defined with specific criteria on the following panel. The percentile response time target is the percentage of requests whose response time is T or less that should be P or more; a target has particular values for T and P.

- Completion time goals specify the maximum amount of time (minutes) that is acceptable for a job to complete and still maintain the level of service that the service policy implies. Completion time is queue time plus execution time of a grid job. Completion time combined with the *importance* associated with service policies ensure that important jobs are dispatched first. All jobs are dispatched immediately if there is capacity. This completion time goal type is used only when there are more jobs than what can be processed immediately. The attempt is to get the job completed by completion time, and not just get dispatched. The application placement controller (APC) assesses the historical data for a job and dispatches the job based on that data. For example, if completion time is set to 30 minutes, and from historical data the APC knows that the job takes 30 minutes to complete, then this job will be dispatched right away. The class of a job is important when predicting the performance characteristics of batch jobs. The APC design is such that the system assumes that a job in class A will have, generally, the same performance characteristics as other jobs in class A. Queue time is deprecated in WebSphere Virtual Enterprise Version 6.1
3. Optional: If you select a goal type of average response time, percentile response time or completion time, you are prompted to define the specifics and select an importance.

For average response time goals, type a goal value, associate an importance with the service policy, and select **Monitor for persistent policy violations** to set up the creation of a runtime task when a policy violation occurs.

When you associate an importance with the service policy, note that the options for importance vary from lowest to highest. Some planning is essential to select the right importance value, because negative results can occur if all work is rated as highest. This rating can create a bottleneck within the environment. To define a policy violation, specify the **Goal delta value** and the **Time period value**:

- In the **Goal delta value** field, type an integer to indicate the maximum allowable amount of time that exceeds the configured goal value. Acceptable values are 0 to 3000 milliseconds, 0 to 300 seconds, and 0 to 2147483647 minutes.
- In the **Time period value** field, type an integer to indicate the milliseconds, seconds or minutes after which the goal value is in violation. This can be 0 to 1 day, inclusive.

For percentile response time, set the goal percentile to the percentage of requests that must meet the goal value that is defined in the next field. Next, type a goal value, associate an importance with the service policy, and select **Monitor for persistent policy violations** to set up the creation of a runtime task when a policy violation occurs.

For the goal value, type the maximum allowable time for the service policy. The environment tries to stay below the defined goals, and continually adjusts to achieve the most balanced result. When you associate an importance with the service policy, note that the options for importance vary from lowest to highest. Some planning is essential to select the right importance value, because negative results can occur if all work is rated as highest. To define a policy violation, specify the **Goal delta percentage** and the **Time period value**:

- In the **Goal delta value** field, type an integer type an integer that indicates the percentage of request below the goal value for which to monitor. Acceptable values are 0 to 100, inclusive.
- In the **Time period value** field, type an integer to indicate the milliseconds, seconds or minutes after which the goal value is in violation.

A runtime task is generated when certain criteria are violated. For example, in the following percentile response time example, with a percentile goal of 90% and a goal delta of 5%, the service policy is breached when less than 85% of requests meet the service time goal of 1 second (for 5 consecutive seconds), that is, when more than 15% of requests exceed the service time goal of 1 second (for 5 consecutive seconds). The system will still prioritize traffic in such a way as to attempt to meet the 90% goal, however no notification of a breach will be issued unless the 85% (90% minus 5%) threshold is not met.

Table 1. Percentile response time example

Description	Value
Goal percentile	90%
Goal value	1
Importance	1
Monitor for persistent service policy violations	true
Goal Delta Percentage:	5%
Time Period Value	5 seconds

For completion time, type a goal value and associate an importance with the service policy. For the goal value, type the maximum allowable time for the service policy. The environment continually adjusts all automatically adjustable controls, aiming to reach and maintain the best possible balance of relative performance results. When you associate an importance with the service policy, note that the options for importance vary from lowest to highest. Some planning is essential to select the right importance value, because negative results can occur if all work is rated as highest. This rating can create a bottleneck within the environment.

4. Associate transaction class members to the service policy, or create a new transaction class. If the transaction class that you are seeking does not exist, create a new transaction class.
5. To create a work class for your service policy, from the administrative console click **Applications > Enterprise Applications > application_name > Service Policies**. Select an existing service policy

and for the request type, click **New**.

To create a new service policy for HTTP, specify a name for the work class, select a module, and select the members to add. Optionally, to use a custom URI, type its name, and click **Add Pattern** in the **Custom URI Pattern** field. For example, a custom URI is necessary to do JavaServer Pages (JSP) work.

To create a new service policy for SOAP, specify a name for the work class, select a module, and select the Web service operations to add.

To create a new service policy for IIOP, specify a name for the work class, select a module, and select the EJB methods to add. Optionally, to use a custom EJB, type the information in the **Custom EJB name** and **Custom EJB method** fields, and click **Add Pattern**.

To create a new service policy for JMS, type a name for the work class, select a module, select a defined bus, and select the EJB methods. Optionally, to use a custom bus, type the information in the **Custom bus name** and **Custom bus destination** fields, and click **Add Pattern**.

To create a service policy for SIP, you must create the following two policies:

- a. Create a default SIP policy with the following values:
 - Goal Type = Average Response Time
 - Goal Value = 75 milliseconds
 - Importance = High
 - b. Create an INVITE policy with the following values:
 - Goal Type = Average Response Time
 - Goal Value = 75 milliseconds
 - Importance = Low
 - c. Set the service policy SIP rules:
 - If request.method = INVITE, then classify to transaction class Default_TC_INVITE (INVITE).
 - If no rules apply, then classify to transaction class Default_TC_def_sip (def_sip).
6. The system automatically picks up any changes you make to your service policy configuration. You do not need to restart any servers when you update your service policies and work classes.

Results

You have defined a business goal and applied that goal to application URIs using the service policy and routing rules. Your system can now categorize and prioritize work.

Parent topic: [Routing and service policies](#)

Related concepts:

[Overview of work classes](#)

[Work class types](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

Related reference:

[Routing and service policies](#)

[Administrative roles and privileges](#)

Related information:

[Service policy custom properties](#)

[Rules for ODR routing policy administrative tasks](#)

[Rules for ODR service policy administrative tasks](#)

Configuring service policies without response time goals

Configure service policies without response time goals when multiple service policies are configured and the response time goals cannot be accurately determined for any of the service policies, or when there might be occasional long-running or stuck requests.

Before you begin

- Enabling this feature enables it for all service policies. It is not possible to have some service policies with response time goals and others without response time goals.
- ARFM queueing cannot be disabled for this feature to work correctly. In other words, the `disableARFM.py` script cannot be used to set the `arfmManageCpu` cell custom property to `false`.

About this task

To aid in the ease of use of the product, performance management without response time goals is supported in WebSphere® Virtual Enterprise Version 7.0. Providing correct response time goals requires load testing to be performed to determine a reasonable value. This feature eliminates the need for performance testing. Instead, the product simply uses the importance of a service policy to manage performance.

To determine when a new dynamic cluster instance should be started when there are no response time goals, a utility value is calculated based upon the fraction of time that requests wait. The utility value is a value between -1 and +1 based on the importance of the service class. Requests with zero waiting time get a +1 utility value, and requests with almost all time waiting get a -1 utility value.

A utility value of zero is obtained if the fraction of time waiting equals the relative importance of the request service class, where relative importance is zero for very high importance and one for very low importance. Hence, the utility value is a function of the performance level achieved, which is measured as fraction of time waiting, and the importance of the request, which is measured as relative importance. For a given importance level, a better performance level means higher utility values. When the utility value is less than zero, the application placement controller attempts to start a new instance if capacity to do so can be found.

Procedure

1. In the administrative console, select **System administration > Cell > Custom Properties > New**.
2. Specify the name of the custom property as `disableResponseTimeGoals`.
3. Specify the value of the property as `true`.

Parent topic: [Routing and service policies](#)

Subexpression builder operands for routing and service policies

Operands and their associated protocols are supported from the initial installation, and can be used in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping. An expression can be built from several sub-expressions using logical operators.

WebSphere® Virtual Enterprise includes routing and service policy rules that use the subexpression builder. The rule builder validates the rule when you apply the changes, and alerts you to mismatched parentheses, and unsupported logic operators. Choices of operands depend on the protocol you use.

SOAP operands

Use SOAP operands in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping.

SIP operands

You can use the SIP operands and their associated protocols in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping.

IIOp operands

Use IIOp operands and their associated protocols in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping.

HTTP operands

Use the HTTP operands in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping.

Parent topic: [Routing and service policies](#)

Related tasks:

[Defining a service policy](#)

Related reference:

[SOAP operands](#)

[SIP operands](#)

[IIOp operands](#)

[HTTP operands](#)

SOAP operands

Use SOAP operands in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping.

Use the following subtopic links to find specific information:

- [Rules](#)
- [SOAP operands](#)
- [Xpath expression](#)
- [Operators](#)

Rules

Each work class contains an optional ordered list of rules that are evaluated for a particular request to determine the policy for that request. Each rule consists of a Boolean expression and a policy value. If the expression evaluates to true for a particular request, the policy associated with that rule is used.

The syntax and semantics of a Boolean expression for a rule are similar to the WHERE clause of a Structured Query Language (SQL) expression. More precisely, the syntax of an expression is defined by the Java™ Message Service (JMS) 1.1 specification. See [Rule-based request classification](#) for more information.

In the JMS specification, identifiers refer to various attributes that can be associated with a request, for example, a specific query parameter, cookie, or HTTP header. A JMS identifier can be thought of as a request variable, or *operand*. These operands can be specific to a protocol. For example, the SOAP service name is an operand that is valid only in a SOAP work class.

Because the SOAP is over HTTP, the HTTP operands are also valid in a SOAP request. JMS specification uses literals to specify a specific value to use in a comparison to a request variable. For example, in the expression:

```
clienthost LIKE '%.ibm.com'
```

'%.ibm.com' is a literal that is used to compare to the client host name for a request. This expression is true for all requests that originate from a computer in the *ibm.com*® domain. Enclose string literals by single quotes. Do *not* enclose numeric literals in single quotes. Parentheses together with the AND, OR, and NOT operators can also be used to form compound Boolean expressions. See the JMS 1.1 specification for a detailed description.

SOAP operands

Table 1. Operands supported in SOAP

Operand	Syntax	Description
Client host	clienthost	The fully-qualified client host name. This is the value of the internet protocol (IP) command host name. This operand does <i>not</i> support numeric operators such as >, >=, <, <=.
Client IPV4	clientipv4	The IP address of the client using the Internet Protocol version 4 (IPv4) dotted quad address type <i>n.n.n.n</i> .
Client IPV6	clientipv6	The Internet Protocol version 6 (IPv6) 128-bit address type of <i>x:x:x:x:x:x:x:x</i> following Request for Comments 1924 (RFC 1924) of the client computer.
Cookie name	cookie\$<name>	A cookie name. For example, the expression <code>cookie\$My_Cookie_Name='My_Cookie_Value'</code> tests a request to see if it contains a cookie named <i>My_Cookie_Name</i> with a value of <i>My_Cookie_Value</i> . To test for the presence or absence of a particular cookie, use one of the following expressions: <pre>cookie\$MyCookieName IS NOT NULL cookie\$MyCookieName IS NULL</pre>
Header name	header \$<name>	A header name and value. For example, the expression <code>header\$Host='localhost'</code> tests a request to see if it contains an HTTP host header with a value of <i>localhost</i> . To test for presence or absence of the host header, use one of the following expressions: <pre>header\$Host IS NOT NULL header\$Host IS NULL</pre>

HTTP method	HTTPMethod	The HTTP method for the request. Possible values are POST, GET, PUT, and DELETE.
MIME type	MIMEType	The MIME type of the request.
Operation	operation	The name of a Web service operation.
Percentage	percentage\$<val>	The percentage operand evaluates to true, a fixed percentage of the time. For example, percentage\$50 evaluates to true on average 50% of the time.
Port	port	The listening port on which the request was received.
Protocol	protocol	The communications protocol that transmits the request. Currently supported protocols are HTTP, HTTPS, SOAP, and SOAPS.
Query parameter	queryparm\$<name>	A header name and value. For example, the expression queryparm\$timezone='EST' tests a request to see if the request contains an HTTP query parameter named <i>timezone</i> with a value of <i>EST</i> . To test for presence or absence of a query parameter, use one of the following forms: <pre>queryparm\$timezone IS NOT NULL queryparm\$timezone IS NULL</pre>
Rampup	rampup\$<startTime>\$<completionTime>	The rampup operand evaluates to true a variable percentage of the time. It always evaluates to false before <startTime> and to true after <completionTime>. As time progresses from <startTime> to <completionTime>, it evaluates to true, a linearly increasing percentage. The format of <startTime> and <completionTime> is day/month/year::hour:min:sec. where <i>day</i> is the day of the month, <i>month</i> is one of the twelve months: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec, <i>year</i> is the 4 digit year, <i>hour</i> is the 2 digit hour of the 24 hour clock, and <i>min</i> and <i>sec</i> are 2 digit values for minute and second, respectively. For example, rampup\$01/Jan/2007::08:00:00\$01/Jan/2007::17:00:00 begins to occasionally evaluate to true at 8 AM on Jan 1, 2007 and always evaluates to true by ramp up completion time at 5 PM of the same day.
Server host	serverhost	The fully-qualified host name of the server. This operand does <i>not</i> support numeric operators such as >, >=, <, <=.
Server IPV4	serveripv4	The IP address of the server computer using the IPv4 dotted quad address type <i>n.n.n.n</i> .
Server IPV6	serveripv6	The IPv6 128-bit address type of <i>x:x:x:x:x:x:x:x</i> following RFC 1924 of the server computer.
Service	service	The name of a Web service.
Time	time	Used to define the date and time of day that a given request must be honored. Two optional fields are StartTime and EndTime. If a request is received outside of that defined window, the request will not be processed.

		<p>The Start Time and End Time fields each have the following format: dayOfWeek/dayOfMonth/month/year::hour:minute:second.</p> <p>For example, Thursday, the 11th of April, year 2007 at 1:03:45 PM is specified as: Thu/11/Apr/2007::13:03:45</p> <p>Any field can use a wild card with the value *.</p> <p>For example, the 1st of each month is specified as */1.</p> <p>The <i>dayOfWeek</i> values are Sun, Mon, Tue, Wed, Thu, Fri, Sat, and the <i>dayOfMonth</i> values range from 1-31.</p> <p>The <i>month</i> value is a non-numeric value that represents the twelve months: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec.</p> <p>The <i>year</i> value is comprised of the year's four digits. For example, 2007.</p> <p>The <i>hour</i> value is the hour of day in the 24-hour clock. For example, 8am is represented as ::8. The <i>minute</i> and <i>second</i> are integers ranging from 0-59.</p> <p>The forward slash (/) is used to separate date parameters, the double colon (::) is used to separate the date parameters, and the colon (:) is used to separate the time of day parameters. Note that it is the Boolean result of the entire rule in which the <i>time</i> operand is used that determines the routing action taken.</p>
Virtual Portal	virtualportal	Virtual portals are created within WebSphere® Portal Server, and WebSphere Virtual Enterprise supports this operand for better integration with WebSphere Portal Server. The virtual portal is the request URL minus the context root for the WebSphere Portal application's Web module. If a given request matches the virtual portal defined, then the routing action defined for that rule is taken.

Xpath expression

The syntax for the Xpath expression is `xpathexpr` and is supported by SOAP.

The XPath string expression has a required field that contains the XPath expression and an optional field for defining local namespace(s). If all the name spaces contained in the XPath expression are standard, then the second field can be optional. If there are multiple local namespaces, then separate each with a comma (,).

The following example shows an `xpathexpr` with one local namespace definition:

```
xpathexpr$/soap:Envelope/soap:Body/m:getTimeZone/n:clientId$m\\=http://test.classify.ws.ibm.com,n\\=http://test2.classify.ws.ibm.com = \\'1000\\' or operation IS NOT NULL
```

In the previous expression, the XPath expression is `/soap:Envelope/soap:Body/n:getTimeZone/n:clientId`. It contains the local namespace of `n`. So, the second field of the `xpathexpr` is defined as `n \\=http://test.classify.ws.ibm.com`. The dollar sign (\$) is used to denote the start of a field definition. The double backslashes are the escape sequence in this example. The first equals sign (=) is escaped because it is part of the local namespace definition, whereas the second equals sign (=) is the operator in the classification expression and must not be escaped.

The following example shows an `xpathexpr` with two local namespaces:

```
xpathexpr$/soap:Envelope/soap:Body/m:getTimeZone/n:clientId$m\\=http://test.classify.ws.ibm.com,n\\=http://test2.classify.ws.ibm.com = \\'1000\\' or operation IS NOT NULL
```

The following table shows the standard namespaces for the `xpathexpr` expression:

Table 2. Namespaces

Namespace	URL
soap	http://schemas.xmlsoap.org/soap/envelope/
soap-env	http://schemas.xmlsoap.org/soap/envelope/

soapenc	http://schemas.xmlsoap.org/soap/encoding/
soapbind	http://schemas.xmlsoap.org/wsdl/soap/
xsd	http://www.w3.org/2001/XMLSchema
xsi	http://www.w3.org/2001/XMLSchema-instance
xsi	http://ws-i.org/schemas/conformanceClaim/
wsdl	http://schemas.xmlsoap.org/wsdl/

Operators

WebSphere Virtual Enterprise supports the operators in the following table in the rules expressions. These operators are also referred to as *predicates* in SQL terminology because they appear inside of a WHERE or HAVING clause. Operators are case insensitive.

Table 3. Request classification operators

Operator	Description
OR	The logical OR operator.
AND	The logical AND operator.
NOT	The negation operator.
IN	<p>Expresses an operand with multiple values in a single expression. Its meaning is consistent with the SQL standard meaning of the operator.</p> <p>For example, if you want to express that the port value could be any or all of the values such as 9080, 9090, 9091, use the expression fragment:</p> <pre>port IN (9080,9090,9091)</pre> <p>In SQL, how the values inside the parenthesis are expressed depends on the data type of port. If the port is an integer, the values without the single quotation marks are syntactically correct. If the port is a string, the correct expression is:</p> <pre>port IN ('9080','9090','9091')</pre>
LIKE	<p>Expresses pattern matching for string operand values. The value must contain the wildcard character (%) in the position where the pattern matching is expected to start.</p> <p>For example, the expression:</p> <pre>host LIKE %blanca</pre> <p>matches the word blanca and any other word that ends in blanca, while the expression:</p> <pre>host LIKE blanca%</pre> <p>matches the word blanca and any other word that starts with blanca. The expression:</p> <pre>host LIKE %blanca%</pre> <p>matches the word blanca and any word that has the token blanca imbedded in it.</p> <p>The <code>java.util.regex.Pattern</code> class is used.</p>
=	The equality operator expresses a match in case-sensitive match.
>	Greater-than operator for use with numeric operands.
>=	Greater-than or equal operator for use with numeric operands.
<	Less-than operator for use with numeric operands.
<=	Less-than or equal operator for use with numeric operands.
< >	Not-equal operator.
BETWEEN	Used with AND to select a range of values inclusive of the first (low) value and the last (high) value. Together, they operate on numbers and dates values.
IS NULL	Tests for an operand having a NULL value.
IS NOT NULL	Tests for an operand having a value other than NULL.

Parent topic: [Subexpression builder operands for routing and service policies](#)

Related concepts:

[Rule-based request classification](#)

Related tasks:

[Defining a service policy](#)

[Setting maintenance mode](#)

[Activating concurrent application editions](#)

[Validating an edition](#)

Related information:

[Rules for ODR routing policy administrative tasks](#)

[Rules for ODR service policy administrative tasks](#)

SIP operands

You can use the SIP operands and their associated protocols in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping.

Use the following subtopic links to find specific information:

- [Rules](#)
- [SIP operands](#)
- [Operators](#)

Rules

Each work class contains an optional ordered list of rules that are evaluated for a particular request to determine the policy for that request. Each rule consists of a Boolean expression and a policy value. If the expression evaluates to true for a particular request, the policy associated with that rule is used.

The syntax and semantics of a Boolean expression for a rule are similar to the WHERE clause of a Structured Query Language (SQL) expression. More precisely, the syntax of an expression is defined by the Java™ Message Service (JMS) 1.1 specification. See [Rule-based request classification](#) for more information.

In the JMS specification, identifiers refer to various attributes that can be associated with a request, for example, a specific query parameter, cookie, or HTTP header. A JMS identifier can be thought of as a request variable, or *operand*. These operands can be specific to a protocol. For example, the SOAP service name is an operand that is valid only in a SOAP work class.

Because the SOAP is over HTTP, the HTTP operands are also valid in a SOAP request. JMS specification uses literals to specify a specific value to use in a comparison to a request variable. For example, in the expression:

```
clienthost LIKE '%.ibm.com'
```

'%.ibm.com' is a literal that is used to compare to the client host name for a request. This expression is true for all requests that originate from a computer in the *ibm.com*® domain. Enclose string literals by single quotes. Do *not* enclose numeric literals in single quotes. Parentheses together with the AND, OR, and NOT operators can also be used to form compound Boolean expressions. See the JMS 1.1 specification for a detailed description.

SIP operands

Table 1. Operands in the subexpression builder

Operand	Syntax	Description
Client host	clienthost	The fully-qualified client host name. This is the value of the internet protocol (IP) command host name. This operand does <i>not</i> support numeric operators such as >, >=, <, <=.
Client IPV4	clientipv4	The IP address of the client using the Internet Protocol version 4 (IPv4) dotted quad address type <i>n.n.n.n</i> .
Client IPV6	clientipv6	The Internet Protocol version 6 (IPv6) 128-bit address type of <i>x:x:x:x:x:x:x:x</i> following Request for Comments 1924 (RFC 1924) of the client computer.
Contact header	request.contact	The Contact header field.
Contact URI	request.contact.uri	The URI of the Contact header field.
From header	request.from	The From header field.
From header display name	request.from.display-name	The display name in the From header field.
From URI	request.from.uri	The URI of the From header field.
From URI host	request.from.uri.host	The host in the From header field.
From URI port	request.from.uri.port	The port in the URI of the From header field.

From URI user	request.from.uri.user	The user in the From header field.
Header name	header \$<name>	A header name and value. For example, the expression header\$Host='localhost' tests a request to see if it contains an HTTP host header with a value of <i>localhost</i> . To test for presence or absence of the host header, use one of the following expressions: <pre>header\$Host IS NOT NULL header\$Host IS NULL</pre>
Percentage	percentage\$<val>	The percentage operand evaluates to true, a fixed percentage of the time. For example, percentage\$50 evaluates to true on average 50% of the time.
Port	port	The listening port on which the request was received.
Rampup	rampup\$<startTime>\$<completionTime>	The rampup operand evaluates to true a variable percentage of the time. It always evaluates to false before <startTime> and to true after <completionTime>. As time progresses from <startTime> to <completionTime>, it evaluates to true, a linearly increasing percentage. The format of <startTime> and <completionTime> is day/month/year::hour:min:sec. where <i>day</i> is the day of the month, <i>month</i> is one of the twelve months: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec, <i>year</i> is the 4 digit year, <i>hour</i> is the 2 digit hour of the 24 hour clock, and <i>min</i> and <i>sec</i> are 2 digit values for minute and second, respectively. For example, rampup\$01/Jan/2007::08:00:00\$01/Jan/2007::17:00:00 begins to occasionally evaluate to true at 8 AM on Jan 1, 2007 and always evaluates to true by ramp up completion time at 5 PM of the same day.
Request transport	request.transport	The transport of the request.
Request URI	request.uri	The request URI.
Request URI host	request.uri.host	The host in the request URI.
Request URI port	request.uri.port	The port in the request URI.
Request URI user	request.uri.user	The user in the request URI.
Scheme of From header	request.from.uri.scheme	The scheme of the From header field.
Scheme of To header	to.uri.scheme	The scheme of the To header field.
Scheme of	request.uri.scheme	The scheme of the URI.

URI		
Server host	serverhost	The fully-qualified host name of the server. This operand does <i>not</i> support numeric operators such as <code>></code> , <code>>=</code> , <code><</code> , <code><=</code> .
Server IPV4	serveripv4	The IP address of the server computer using the IPv4 dotted quad address type <i>n.n.n.n</i> .
Server IPV6	serveripv6	The IPv6 128-bit address type of <i>x:x:x:x:x:x:x:x</i> following RFC 1924 of the server computer.
SIP method	request.method	The SIP method for the request. Possible values are INVITE, TRYING, RINGING, ACK, OK, and BYE. If the message is not a request, this operand returns null.
SIP response code	response.code	The response code of the response. If the message is not a response, this operand returns -1.
To client host	clienthost	The client host in the To header field.
To header	request.to	The To header field.
To header display name	request.to.display-name	The display name in the To header field.
To URI	request.to.uri	The URI of the To header field.
To URI host	request.to.uri.host	The port in the URI of the To header field.
To URI user	request.to.uri.user	The user in the To header field.

Operators

WebSphere® Virtual Enterprise supports the operators in the following table in the rules expressions. These operators are also referred to as *predicates* in SQL terminology because they appear inside of a WHERE or HAVING clause. Operators are case insensitive.

Table 2. Request classification operators

Operator	Description
OR	The logical OR operator.
AND	The logical AND operator.
NOT	The negation operator.
IN	<p>Expresses an operand with multiple values in a single expression. Its meaning is consistent with the SQL standard meaning of the operator.</p> <p>For example, if you want to express that the port value could be any or all of the values such as 9080, 9090, 9091, use the expression fragment:</p> <pre>port IN (9080,9090,9091)</pre> <p>In SQL, how the values inside the parenthesis are expressed depends on the data type of port. If the port is an integer, the values without the single quotation marks are syntactically correct. If the port is a string, the correct expression is:</p> <pre>port IN ('9080','9090','9091')</pre>
LIKE	Expresses pattern matching for string operand values. The value must contain the wildcard character (%) in the position where the pattern matching is expected to start.

	<p>For example, the expression:</p> <pre>host LIKE %blanca</pre> <p>matches the word blanca and any other word that ends in blanca, while the expression:</p> <pre>host LIKE blanca%</pre> <p>matches the word blanca and any other word that starts with blanca. The expression:</p> <pre>host LIKE %blanca%</pre> <p>matches the word blanca and any word that has the token blanca imbedded in it.</p> <p>The <code>java.util.regex.Pattern</code> class is used.</p>
=	The equality operator expresses a match in case-sensitive match.
>	Greater-than operator for use with numeric operands.
>=	Greater-than or equal operator for use with numeric operands.
<	Less-than operator for use with numeric operands.
<=	Less-than or equal operator for use with numeric operands.
< >	Not-equal operator.
BETWEEN	Used with AND to select a range of values inclusive of the first (low) value and the last (high) value. Together, they operate on numbers and dates values.
IS NULL	Tests for an operand having a NULL value.
IS NOT NULL	Tests for an operand having a value other than NULL.

Parent topic: [Subexpression builder operands for routing and service policies](#)

Related concepts:

[Rule-based request classification](#)

Related tasks:

[Defining a service policy](#)

[Setting maintenance mode](#)

[Activating concurrent application editions](#)

[Validating an edition](#)

Related information:

[Rules for ODR routing policy administrative tasks](#)

[Rules for ODR service policy administrative tasks](#)

IIOp operands

Use IIOp operands and their associated protocols in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping.

Use the following subtopic links to find specific information:

- [Rules](#)
- [IIOp operands](#)
- [Operators](#)

Rules

Each work class contains an optional ordered list of rules that are evaluated for a particular request to determine the policy for that request. Each rule consists of a Boolean expression and a policy value. If the expression evaluates to true for a particular request, the policy associated with that rule is used.

The syntax and semantics of a Boolean expression for a rule are similar to the WHERE clause of a Structured Query Language (SQL) expression. More precisely, the syntax of an expression is defined by the Java™ Message Service (JMS) 1.1 specification. See [Rule-based request classification](#) for more information.

In the JMS specification, identifiers refer to various attributes that can be associated with a request, for example, a specific query parameter, cookie, or HTTP header. A JMS identifier can be thought of as a request variable, or *operand*. These operands can be specific to a protocol. For example, the SOAP service name is an operand that is valid only in a SOAP work class.

Because the SOAP is over HTTP, the HTTP operands are also valid in a SOAP request. JMS specification uses literals to specify a specific value to use in a comparison to a request variable. For example, in the expression:

```
clienthost LIKE '%.ibm.com'
```

'%.ibm.com' is a literal that is used to compare to the client host name for a request. This expression is true for all requests that originate from a computer in the *ibm.com*® domain. Enclose string literals by single quotes. Do *not* enclose numeric literals in single quotes. Parentheses together with the AND, OR, and NOT operators can also be used to form compound Boolean expressions. See the JMS 1.1 specification for a detailed description.

IIOp operands

Table 1. Operands supported by IIOp

Operand	Syntax	Description
Application	application	The name of the enterprise application where the EJB is contained.
Client host	clienthost	The fully-qualified client host name. This is the value of the internet protocol (IP) command host name. This operand does <i>not</i> support numeric operators such as >, >=, <, <=.
Client port	clientport	The client port name.
EJB module	ejbmodule	The module name of an EJB.
EJB name	ejb	The name of an EJB.
EJB method	ejbmethod	The name of a method within the EJB.
Percentage	percentage\$<val>	The percentage operand evaluates to true, a fixed percentage of the time. For example, percentage\$50 evaluates to true on average 50% of the time.
Port	port	The listening port on which the request was received.
Rampup	rampup\$<startTime>\$<completionTime>	The rampup operand evaluates to true a variable percentage of the time. It always evaluates to false before <startTime> and to true after <completionTime>. As time progresses from <startTime> to <completionTime>, it evaluates to true, a linearly increasing percentage.

		<p>The format of <code><startTime></code> and <code><completionTime></code> is <code>day/month/year::hour:min:sec</code>.</p> <p>where <i>day</i> is the day of the month, <i>month</i> is one of the twelve months: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec, <i>year</i> is the 4 digit year, <i>hour</i> is the 2 digit hour of the 24 hour clock, and <i>min</i> and <i>sec</i> are 2 digit values for minute and second, respectively.</p> <p>For example,</p> <p><code>rampup\$01/Jan/2007::08:00:00\$01/Jan/2007::17:00:00</code> begins to occasionally evaluate to true at 8 AM on Jan 1, 2007 and always evaluates to true by ramp up completion time at 5 PM of the same day.</p>
Server host	serverhost	The fully-qualified host name of the server. This operand does <i>not</i> support numeric operators such as <code>></code> , <code>>=</code> , <code><</code> , <code><=</code> .
Time	time	<p>Used to define the date and time of day that a given request must be honored. Two optional fields are <code>StartTime</code> and <code>EndTime</code>. If a request is received outside of that defined window, the request will not be processed.</p> <p>The <code>Start Time</code> and <code>End Time</code> fields each have the following format: <code>dayOfWeek/dayOfMonth/month/year::hour:minute:second</code>.</p> <p>For example, Thursday, the 11th of April, year 2007 at 1:03:45 PM is specified as:</p> <p><code>Thu/11/Apr/2007::13:03:45</code></p> <p>Any field can use a wild card with the value <code>*</code>.</p> <p>For example, the 1st of each month is specified as <code>*/1</code>.</p> <p>The <i>dayOfWeek</i> values are Sun, Mon, Tue, Wed, Thu, Fri, Sat, and the <i>dayOfMonth</i> values range from 1-31.</p> <p>The <i>month</i> value is a non-numeric value that represents the twelve months: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec.</p> <p>The <i>year</i> value is comprised of the year's four digits. For example, 2007.</p> <p>The <i>hour</i> value is the hour of day in the 24-hour clock. For example, 8am is represented as <code>::8</code>. The <i>minute</i> and <i>second</i> are integers ranging from 0-59.</p> <p>The forward slash (/) is used to separate date parameters, the double colon (::) is used to separate the date parameters, and the colon (:) is used to separate the time of day parameters. Note that it is the Boolean result of the entire rule in which the <i>time</i> operand is used that determines the routing action taken.</p>

Operators

WebSphere® Virtual Enterprise supports the operators in the following table in the rules expressions. These operators are also referred to as *predicates* in SQL terminology because they appear inside of a `WHERE` or `HAVING` clause. Operators are case insensitive.

Table 2. Request classification operators

Operator	Description
OR	The logical OR operator.
AND	The logical AND operator.
NOT	The negation operator.

IN	<p>Expresses an operand with multiple values in a single expression. Its meaning is consistent with the SQL standard meaning of the operator.</p> <p>For example, if you want to express that the port value could be any or all of the values such as 9080, 9090, 9091, use the expression fragment:</p> <pre>port IN (9080,9090,9091)</pre> <p>In SQL, how the values inside the parenthesis are expressed depends on the data type of port. If the port is an integer, the values without the single quotation marks are syntactically correct. If the port is a string, the correct expression is:</p> <pre>port IN ('9080','9090','9091')</pre>
LIKE	<p>Expresses pattern matching for string operand values. The value must contain the wildcard character (%) in the position where the pattern matching is expected to start.</p> <p>For example, the expression:</p> <pre>host LIKE %blanca</pre> <p>matches the word blanca and any other word that ends in blanca, while the expression:</p> <pre>host LIKE blanca%</pre> <p>matches the word blanca and any other word that starts with blanca. The expression:</p> <pre>host LIKE %blanca%</pre> <p>matches the word blanca and any word that has the token blanca imbedded in it.</p> <p>The <code>java.util.regex.Pattern</code> class is used.</p>
=	The equality operator expresses a match in case-sensitive match.
>	Greater-than operator for use with numeric operands.
>=	Greater-than or equal operator for use with numeric operands.
<	Less-than operator for use with numeric operands.
<=	Less-than or equal operator for use with numeric operands.
< >	Not-equal operator.
BETWEEN	Used with AND to select a range of values inclusive of the first (low) value and the last (high) value. Together, they operate on numbers and dates values.
IS NULL	Tests for an operand having a NULL value.
IS NOT NULL	Tests for an operand having a value other than NULL.

Parent topic: [Subexpression builder operands for routing and service policies](#)

Related concepts:

[Rule-based request classification](#)

Related tasks:

[Defining a service policy](#)

[Setting maintenance mode](#)

[Activating concurrent application editions](#)

[Validating an edition](#)

Related information:

[Rules for ODR routing policy administrative tasks](#)

[Rules for ODR service policy administrative tasks](#)

HTTP operands

Use the HTTP operands in the subexpression utility builder, an optional tool that helps you build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping.

- [Rules](#)
- [HTTP request operands](#)
- [HTTP response operands](#)
- [Xpath expression](#)
- [Operators](#)

Rules

Each work class contains an optional ordered list of rules that are evaluated for a particular request to determine the policy for that request. Each rule consists of a Boolean expression and a policy value. If the expression evaluates to true for a particular request, the policy associated with that rule is used.

The syntax and semantics of a Boolean expression for a rule are similar to the WHERE clause of a Structured Query Language (SQL) expression. More precisely, the syntax of an expression is defined by the Java™ Message Service (JMS) 1.1 specification. See [Rule-based request classification](#) for more information.

In the JMS specification, identifiers refer to various attributes that can be associated with a request, for example, a specific query parameter, cookie, or HTTP header. A JMS identifier can be thought of as a request variable, or *operand*. These operands can be specific to a protocol. For example, the SOAP service name is an operand that is valid only in a SOAP work class.

Because the SOAP is over HTTP, the HTTP operands are also valid in a SOAP request. JMS specification uses literals to specify a specific value to use in a comparison to a request variable. For example, in the expression:

```
clienthost LIKE '%.ibm.com'
```

'%.ibm.com' is a literal that is used to compare to the client host name for a request. This expression is true for all requests that originate from a computer in the *ibm.com*® domain. Enclose string literals by single quotes. Do *not* enclose numeric literals in single quotes. Parentheses together with the AND, OR, and NOT operators can also be used to form compound Boolean expressions. See the JMS 1.1 specification for a detailed description.

HTTP request operands

Table 1. HTTP request operands

Operand	Syntax	Description
Client host	clienthost	The fully-qualified client host name. This is the value of the internet protocol (IP) command host name. This operand does <i>not</i> support numeric operators such as >, >=, <, <=.
Client IPV4	clientipv4	The IP address of the client using the Internet Protocol version 4 (IPv4) dotted quad address type <i>n.n.n.n</i> .
Client IPV6	clientipv6	The Internet Protocol version 6 (IPv6) 128-bit address type of <i>x:x:x:x:x:x:x:x</i> following Request for Comments 1924 (RFC 1924) of the client computer.
Cookie name	cookie\$<name>	A cookie name. For example, the expression <code>cookie\$My_Cookie_Name='My_Cookie_Value'</code> tests a request to see if it contains a cookie named <i>My_Cookie_Name</i> with a value of <i>My_Cookie_Value</i> . To test for the presence or absence of a particular cookie, use one of the following expressions: <pre>cookie\$MyCookieName IS NOT NULL cookie\$MyCookieName IS NULL</pre>
Header name	header\$<name>	A header name and value. For example, the expression <code>header\$Host='localhost'</code> tests a request to see if it contains an HTTP host header with a value of <i>localhost</i> . To test for presence or absence of the host header, use one of the following expressions: <pre>header\$Host IS NOT NULL header\$Host IS NULL</pre>

HTTP method	HTTPMethod	The HTTP method for the request. Possible values are POST, GET, PUT, and DELETE.
MIME type	MIMETYPE	The MIME type of the request.
Percentage	percentage\$<val>	The percentage operand evaluates to true, a fixed percentage of the time. For example, percentage\$50 evaluates to true on average 50% of the time.
Port	port	The listening port on which the request was received.
Protocol	protocol	The communications protocol that transmits the request. Currently supported protocols are HTTP, HTTPS, SOAP, and SOAPS.
Query parameter	queryparm\$<name>	A header name and value. For example, the expression queryparm\$timezone='EST' tests a request to see if the request contains an HTTP query parameter named <i>timezone</i> with a value of <i>EST</i> . To test for presence or absence of a query parameter, use one of the following forms: <pre>queryparm\$timezone IS NOT NULL queryparm\$timezone IS NULL</pre>
Rampup	rampup\$<startTime> \$<completionTime>	The rampup operand evaluates to true a variable percentage of the time. It always evaluates to false before <startTime> and to true after <completionTime>. As time progresses from <startTime> to <completionTime>, it evaluates to true, a linearly increasing percentage. The format of <startTime> and <completionTime> is day/month/year::hour:min:sec. where <i>day</i> is the day of the month, <i>month</i> is one of the twelve months: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec, <i>year</i> is the 4 digit year, <i>hour</i> is the 2 digit hour of the 24 hour clock, and <i>min</i> and <i>sec</i> are 2 digit values for minute and second, respectively. For example, rampup\$01/Jan/2007::08:00:00\$01/Jan/2007::17:00:00 begins to occasionally evaluate to true at 8 AM on Jan 1, 2007 and always evaluates to true by ramp up completion time at 5 PM of the same day.
Scheme of URI	request.uri.scheme	The scheme of the URI.
Server host	serverhost	The fully-qualified host name of the server. This operand does <i>not</i> support numeric operators such as >, >=, <, <=.
Server IPV4	serveripv4	The IP address of the server computer using the IPv4 dotted quad address type <i>n.n.n.n</i> .
Server IPV6	serveripv6	The IPv6 128-bit address type of <i>x:x:x:x:x:x:x:x</i> following RFC 1924 of the server computer.
Service	service	The name of a Web service.
Time	time	Used to define the date and time of day that a given request must be honored. Two optional fields are StartTime

		<p>and EndTime. If a request is received outside of that defined window, the request will not be processed.</p> <p>The Start Time and End Time fields each have the following format: <code>dayOfWeek/dayOfMonth/month/year::hour:minute:second</code>.</p> <p>For example, Thursday, the 11th of April, year 2007 at 1:03:45 PM is specified as: <code>Thu/11/Apr/2007::13:03:45</code></p> <p>Any field can use a wild card with the value <code>*</code>.</p> <p>For example, the 1st of each month is specified as <code>*/1</code>.</p> <p>The <i>dayOfWeek</i> values are Sun, Mon, Tue, Wed, Thu, Fri, Sat, and the <i>dayOfMonth</i> values range from 1-31.</p> <p>The <i>month</i> value is a non-numeric value that represents the twelve months: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec.</p> <p>The <i>year</i> value is comprised of the year's four digits. For example, 2007.</p> <p>The <i>hour</i> value is the hour of day in the 24-hour clock. For example, 8am is represented as <code>::8</code>. The <i>minute</i> and <i>second</i> are integers ranging from 0-59.</p> <p>The forward slash (/) is used to separate date parameters, the double colon (::) is used to separate the date parameters, and the colon (:) is used to separate the time of day parameters. Note that it is the Boolean result of the entire rule in which the <i>time</i> operand is used that determines the routing action taken.</p>
URI	uri	Uniform Resource Identifier
Virtual Host	virtualhost	Virtual host target of the request, used for configuring Web applications to a particular host name.
Virtual Port	numeric	Virtual port target of the request, used for configuring Web applications to a particular port.
Virtual Portal	virtualportal	Virtual portals are created within WebSphere® Portal Server, and WebSphere Virtual Enterprise supports this operand for better integration with WebSphere Portal Server. The virtual portal is the request URL minus the context root for the WebSphere Portal application's Web module. If a given request matches the virtual portal defined, then the routing action defined for that rule is taken.

HTTP response operands

When you are using a custom log file, you can use the following operands in addition to the operands in Table 1:

Table 2. HTTP response operands

Operand	Syntax	Description
Response code	<code>response.code</code>	Filters by HTTP response codes, such as 404, 503, and so on.
Response time	<code>response.time</code>	The number of milliseconds between when the request in the ODR is received, and until the response from the ODR is sent.
Response write error	<code>response.write.error</code>	Logs errors that can occur when writing a response to a client.
Service time	<code>service.time</code>	The number of milliseconds between when the

		request is sent to the application server, and when a response is received from the application server.
Target server	targetserver	Shows the server where the request was sent, in a WebSphere Application Server format. For example, mycell/mynode/myserver.

Xpath expression

The syntax for the Xpath expression is `xpathexpr` and is supported by HTTP.

The XPath string expression has a required field that contains the XPath expression and an optional field for defining local namespaces. If all the namespaces contained in the XPath expression are standard, the second field can be optional. If there are multiple local namespaces, separate each with a comma (,).

The following example shows an `xpathexpr` with one local namespace definition:

```
xpathexpr$/http:Envelope/soap:Body/m:getTimeZone/n:clientId$m\\=http://test.classify.ws.ibm.com,n\\=http://test2.classify.ws.ibm.com = \\'1000\\' or operation IS NOT NULL
```

In the previous expression, the XPath expression is `/http:Envelope/soap:Body/n:getTimeZone/n:clientId`. It contains the local namespace of `n`. So, the second field of the `xpathexpr` is defined as `n\\=http://test.classify.ws.ibm.com`. The dollar sign (\$) is used to denote the start of a field definition. The double backslashes are the escape sequence in this example. The first equals sign (=) is escaped because it is part of the local namespace definition, whereas the second equals sign (=) is the operator in the classification expression and must not be escaped.

The following example shows an `xpathexpr` with two local namespaces:

```
xpathexpr$/http:Envelope/http:Body/m:getTimeZone/n:clientId$m\\=http://test.classify.ws.ibm.com,n\\=http://test2.classify.ws.ibm.com = \\'1000\\' or operation IS NOT NULL
```

The following table shows the standard namespaces for the `xpathexpr` expression:

Table 3. Namespaces

Namespace	URL
soap	http://schemas.xmlsoap.org/soap/envelope/
soap-env	http://schemas.xmlsoap.org/soap/envelope/
soapenc	http://schemas.xmlsoap.org/soap/encoding/
soapbind	http://schemas.xmlsoap.org/wsdl/soap/
xsd	http://www.w3.org/2001/XMLSchema
xsi	http://www.w3.org/2001/XMLSchema-instance
xsi	http://ws-i.org/schemas/conformanceClaim/
wsdl	http://schemas.xmlsoap.org/wsdl/

Operators

WebSphere Virtual Enterprise supports the operators in the following table in the rules expressions. These operators are also referred to as *predicates* in SQL terminology because they appear inside of a WHERE or HAVING clause. Operators are case insensitive.

Table 4. Request classification operators

Operator	Description
OR	The logical OR operator.
AND	The logical AND operator.
NOT	The negation operator.
IN	Expresses an operand with multiple values in a single expression. Its meaning is consistent with the SQL standard meaning of the operator. For example, if you want to express that the port value could be any or all of the values such as 9080, 9090, 9091, use the expression fragment: <code>port IN (9080 9090 9091)</code>

	<pre>port IN (9080,9090,9091)</pre> <p>In SQL, how the values inside the parenthesis are expressed depends on the data type of port. If the port is an integer, the values without the single quotation marks are syntactically correct. If the port is a string, the correct expression is:</p> <pre>port IN ('9080', '9090', '9091')</pre>
LIKE	<p>Expresses pattern matching for string operand values. The value must contain the wildcard character (%) in the position where the pattern matching is expected to start.</p> <p>For example, the expression:</p> <pre>host LIKE %blanca</pre> <p>matches the word blanca and any other word that ends in blanca, while the expression:</p> <pre>host LIKE blanca%</pre> <p>matches the word blanca and any other word that starts with blanca. The expression:</p> <pre>host LIKE %blanca%</pre> <p>matches the word blanca and any word that has the token blanca imbedded in it.</p> <p>The <code>java.util.regex.Pattern</code> class is used.</p>
=	The equality operator expresses a match in case-sensitive match.
>	Greater-than operator for use with numeric operands.
>=	Greater-than or equal operator for use with numeric operands.
<	Less-than operator for use with numeric operands.
<=	Less-than or equal operator for use with numeric operands.
< >	Not-equal operator.
BETWEEN	Used with AND to select a range of values inclusive of the first (low) value and the last (high) value. Together, they operate on numbers and dates values.
IS NULL	Tests for an operand having a NULL value.
IS NOT NULL	Tests for an operand having a value other than NULL.

Parent topic: [Custom logs overview](#)

Parent topic: [Subexpression builder operands for routing and service policies](#)

Related concepts:

[Rule-based request classification](#)

Related tasks:

[Defining a service policy](#)

[Setting maintenance mode](#)

[Activating concurrent application editions](#)

[Validating an edition](#)

Related reference:

[manageODR.py script](#)

Related information:

[Rules for ODR routing policy administrative tasks](#)

[Rules for ODR service policy administrative tasks](#)

Service policy rules

Service policy rules define which service policy goal should be used to service a single piece of incoming work (request through the on demand router (ODR). This is done by associating a transaction class with a Boolean expression. Each transaction class is associated with exactly one service policy goal. The Boolean expression can be customized to match any specific piece of work (request through the ODR).

Before you begin

Define a service policy to categorize and prioritize work requests. See [Defining a service policy](#) for more information.

About this task

Optionally, you can add rules to your service policy. If the expression evaluates to true for a particular request, the policy associated with that rule is used. A work class contains a set of rules that is used by WebSphere® Virtual Enterprise and is used to classify requests, such as an HTTP request or IOP call. Complete the following steps to define a service policy rule. This is just one example of how to create service policy rules.

Procedure

1. Select **Applications > Enterprise Applications > *application_name* > Service Policies**. Click **OK**.
2. From the **Service policies** tab, expand the work request type and the work class for which you want to create a rule, and click **Add Rule> Rule Builder > Add**
3. Select the type of rule, such as **Group ID** This displays the rule builder panel. Continue to build a rule, specify the transaction class, or click **OK**
4. Click the new rule to set its operators A predefined set of operators displays for the type of rule condition that is selected.
5. Select the operator that you want to use and enter the appropriate information in the provided field. For example, you can classify incoming requests for the StockTradeWork work class by a group ID to use a different transaction class.
6. Select the **(=)** operator and type HTTP in the provided field, to provide a different transaction class for HTTP requests.

[Rules support](#)

Support for HTTP rules applies for both routing and service policy rules. They can be variously located and have a specific order of priority.

[Routing policy action types](#)

You can create routing policies for HTTP and SOAP and SIP requests. The routing policy actions types vary for different locations of rules. The table in this topic shows the actions types for routing policies based on location and protocol.

[Overview of work classes](#)

A work class is a grouping of work that is to be done by an application server. Work is an HTTP request, a SIP message, an IOP call, or a JMS message. Each work class contains a set of rules that is used by WebSphere® Virtual Enterprise to determine how to handle that work.

[Work class types](#)

You can use either default work classes that are created during a system application installation or define your own. Default work classes and directories for system applications are created during profile augmentation to support the high availability deployment manager. Default and new application work classes are defined on a per application edition basis.

[Request classification operators](#)

WebSphere Virtual Enterprise supports operators in the rules expressions. These operators are also referred to as *predicates* in Structured Query Language (SQL) terminology because they appear inside of a WHERE or HAVING clause. Operators are case insensitive.

[Rule-based request classification](#)

With any rule-based technology, rule-based processing involves three basic areas: the vocabulary that forms the language, the grammar for expressing the vocabulary in statements, and the rule-processing engine.

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Related concepts:

[Work class types](#)

Related tasks:

[Defining a service policy](#)

Related reference:

[Subexpression builder operands for routing and service policies](#)

[Routing and service policies](#)

Rules support

Support for HTTP rules applies for both routing and service policy rules. They can be variously located and have a specific order of priority.

Overview

HTTP rules support applies for both routing and service policy rules. They can be located:

1. Inside work classes at the application level
2. Inside work classes at the ODR level
3. At the ODR level. There is no default rule supported for HTTP rules located here. The rules created take precedence over any rules at the application level.

The order of priority is

1. Custom HTTP rules at the ODR level is used to route the request.
2. If none of those rules match, custom HTTP rules located inside work classes at the application level is used to route the request.
3. If none of those rules match, the default HTTP rule located inside work classes at the application level is guaranteed to be used to route the request.

The default rule for HTTP service and routing policy rules, at the on demand router (ODR) level, is not supported.

Parent topic: [Service policy rules](#)

Related concepts:

[Work class types](#)

Related tasks:

[Service policy rules](#)

[Defining a service policy](#)

Related reference:

[Subexpression builder operands for routing and service policies](#)

[Routing and service policies](#)

Routing policy action types

You can create routing policies for HTTP and SOAP and SIP requests. The routing policy actions types vary for different locations of rules. The table in this topic shows the actions types for routing policies based on location and protocol.

Action types

Table 1. Action types for rules based on protocol

R u l e s	Location	Action type
H T T P	Inside work classes at the application level	<ul style="list-style-type: none"> • Permit routing to (permit) • Permit routing with affinity (permitsticky) • Redirect routing to (redirect) • Reject routing with return code (reject) • Permit routing only to servers in maintenance mode (permitMM) • Permit routing with affinity only to servers in maintenance mode (permitstickyMM)
S O A P	Inside work classes at the application level	<ul style="list-style-type: none"> • Permit routing to (permit) • Permit routing with affinity (permitsticky) • Redirect routing to (redirect) • Reject routing with return code (reject) • Permit routing only to servers in maintenance mode (permitMM) • Permit routing with affinity only to servers in maintenance mode (permitstickyMM)
H T T P	Inside work classes at the ODR level	<ul style="list-style-type: none"> • Permit routing to (permit) • Permit routing with affinity (permitsticky) • Redirect routing to (redirect) • Reject routing with return code (reject) • Permit routing only to servers in maintenance mode (permitMM) • Permit routing with affinity only to servers in maintenance mode (permitstickyMM)
H T T P	At the ODR level. There is no default rule supported for HTTP rules in this location. The rules created take precedence over any rules at the application level.	<ul style="list-style-type: none"> • Permit routing to (permit) • Permit routing with affinity (permitsticky) • Redirect routing to (redirect) • Permit routing only to servers in maintenance mode (permitMM) • Permit routing with

		affinity only to servers in maintenance mode (permitstickyMM)
S IP	At the ODR level.	<ul style="list-style-type: none"> • Permit routing to (permit) • Reject routing with return code (reject)

Selecting an action type

1. Selecting one of the following options:

- **Permit routing to:** From the **Select edition name here** list select the edition name.
- **Reject routing with return code:** From the **Select edition name here** list select the edition name and in the **Enter in return code** field, type the return code.
- **Redirect routing to:** From the **Select edition name here** list select the edition name and in the **Enter URI to redirect to** field, type the URI.
- **Permit routing with affinity to:** From the **Select edition name here** list select the edition name.

2. Click **Apply**.

Alternatively, you can apply new classification rules by clicking **Add Rule** and taking the following actions:

- If you know your rule name, type the new routing rule name, and select one of the following options from the **Then** list:
 - **Permit routing to:** From the **Select edition name here** list select the edition name.
 - **Reject routing with return code:** From the **Select edition name here** list select the edition name and in the **Enter in return code** field, type the return code.
 - **Redirect routing to:** From the **Select edition name here** list select the edition name and in the **Enter URI to redirect to** field, type the URI.
 - **Permit routing with affinity to:** From the **Select edition name here** list select the edition name.
- To build a new rule, click **Rule builder**, select the type of rule that you want to create from the rule **Condition** list, and click **Add**:
 1. Click the new rule to set its operators. A predefined set of operators displays for the type of rule condition selected.
 2. Select the operator that you want to use and enter the appropriate information in the field provided.

Parent topic: [Service policy rules](#)

Related concepts:

[Work class types](#)

Related tasks:

[Defining a service policy](#)

Related reference:

[Subexpression builder operands for routing and service policies](#)
[Routing and service policies](#)

Related information:

[Rules for ODR routing policy administrative tasks](#)

Overview of work classes

A work class is a grouping of work that is to be done by an application server. Work is an HTTP request, a SIP message, an IIOP call, or a JMS message. Each work class contains a set of rules that is used by WebSphere® Virtual Enterprise to determine how to handle that work.

Work class overview

A work class is used to classify requests to apply a policy to the request. There are two main types of work classes:

1. Service policy work classes - Work class rules associate incoming work with a service policy, thus indicating to WebSphere Virtual Enterprise when to forward the work to the application server.
2. Routing policy work classes - Work class rules associate incoming work with routing policy, thus indicating to WebSphere Virtual Enterprise where to send the work. Each unit of work (request, message, or call) is associated with a single service policy work class. HTTP requests and SIP messages are also associated with a single routing work class. Routing work classes do not exist for IIOP and JMS because these protocols do not flow through the on demand router (ODR); thus no routing policy is needed.

Creating a work class

A work class is applied to the patterns that are associated with it. Every pattern has an object (application module or on demand router) with which it is associated. These pattern and object pairs make up the definition of which requests the work class is mapped to. Custom work class defined patterns take precedent over the default work class patterns. Every default work class contains a "/" pattern, which means every request matches for that object.

Custom work class definitions are evaluated before default work class definitions when trying to find the matching definition. If an incoming request cannot be matched by any custom work class definitions, the default work class definition is used.

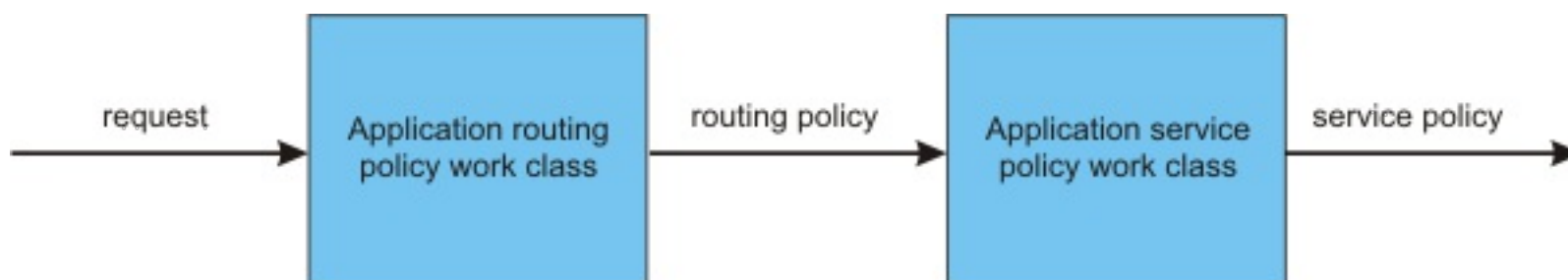
Work class types

There are different types of work classes for service and routing policies:

Table 1. Work class types

Work class	Description
Application routing policy	Specifies how to determine the routing policy for a request to an application that is installed on WebSphere Virtual Enterprise.
Application service policy	Specifies how to determine the service policy for a request to an application that is installed on WebSphere Virtual Enterprise.
Generic server cluster routing policy	Specifies how to determine the routing policy for a request to a generic server cluster.
Generic server cluster service policy	Specifies how to determine the service policy for a request to a generic server cluster.

The following diagram shows the flow of a request that is targeted for an application that is installed on WebSphere Virtual Enterprise. The request is applied to an application routing policy work class to determine the routing policy. If the resulting routing policy is permit or permitsticky, the request continues to an application service policy work class to determine the service policy and the transaction class name.



Each application contains default application routing and application service policy work classes. You can also create additional non-default work classes. Each work class has a default match action. The default routing policy for an application is `permit:application_name`. The default service policy is `Default_TC`, the default transaction class.

Parent topic: [Service policy rules](#)

Related concepts:

[Work class types](#)

[Routing policy action types](#)

Related tasks:

[Defining a service policy](#)

Related reference:

[Subexpression builder operands for routing and service policies](#)

Work class types

You can use either default work classes that are created during a system application installation or define your own. Default work classes and directories for system applications are created during profile augmentation to support the high availability deployment manager. Default and new application work classes are defined on a per application edition basis.

Default application work classes

Each default work class has membership that is equivalent to a wildcard expression for all the work of that protocol type for that application. This work class is matched to last, with any new user defined work classes taking precedence. Default work classes cannot have their membership altered manually nor can they be deleted. They are meant to define how any work directed toward the application that does not get classified into any user-defined work classes into a service policy definition. While the membership cannot be deleted, classification rules can be defined on the default work class. This is especially useful if the environment does not need to classify based on work class membership, but does need to classify on some advanced criteria such as group identification or host name.

The default matchAction on the default work classes for the application is to classify to the default transaction class of the default service policy. This can be changed to select an alternate transaction class/service policy pair.

Routing and service policies for work classes are not supported for IIOP or JMS for applications deployed to z/OS® platforms. WebSphere® Application Server z/OS provides IIOP and JMS service classification.

New application work classes

Each edition of the application has its own definitions on how it should be classified into service policies. After the on demand router (ODR) determines which application edition should be routed to, its service policy work class definitions are evaluated to determine how its work should be classified. When a new edition of an application is installed, you can choose an edition of the application to clone or choose none at all. If an edition is chosen, all of its work classes are cloned with the defaults appropriately renamed with the new application edition name. If no edition is chosen, only the defaults are created.

Configuration placement location

The configuration placement location for work classes for applications is:

```
<context>
  <context-name>applications</context-name>
  <child-context-names>
    <child-context-name>deployments</child-context-name>
    <child-context-name>workclasses</child-context-name>
  </child-context-names>
</context>
<context>
  <context-name>deployments</context-name>
  <child-context-names>
    <child-context-name>workclasses</child-context-name>
  </child-context-names>
</context>
<context>
  <context-name>workclasses</context-name>
  <root-document-type>WorkClass</root-document-type>
  <child-document-names>
    <child-document-name>WorkClass</child-document-name>
  </child-document-names>
</context>
```

Default system application work classes

Default work classes and directories for system applications, such as the adminconsole.ear, are created during profile augmentation to support the high availability deployment manager. An xd directory under the cell context mimics the systemApps structure and contains the default work classes. The default work classes are created under the following contexts:

```
cells/<cellName>/xd/systemApps/<earName>/workclasses/<workclass>/
```

```
cells/<cellName>/xd/systemApps/<earName>/xddeployments/<appName>/workclasses/<workclass>/
```

WebSphere Virtual Enterprise listens for changes to the systemapps.xml file under the node context for any updates:

```
cells/<cellName>/nodes/<nodeName>/systemapps.xml
```

Middleware application work class location

The middleware application work classes location is:

```
<context>
  <context-name>middlewareapps</context-name>
  <child-context-names>
    <child-context-name>middlewareappedititions</child-context-name>
    <child-context-name>workclasses</child-context-name>
    <child-context-name>preferences</child-context-name>
  </child-context-names>
</context>
<context>
  <context-name>middlewareappedititions</context-name>
  <child-context-names>
    <child-context-name>workclasses</child-context-name>
  </child-context-names>
</context>
```

WebSphere Virtual Enterprise listens for changes to the systemapps.xml file under the node context for any updates:

```
cells/<cellName>/nodes/<nodeName>/systemapps.xml
```

Parent topic: [Service policy rules](#)

Related concepts:
[Routing policy action types](#)

Related tasks:
[Defining a service policy](#)

Related reference:
[Routing and service policies](#)

Request classification operators

WebSphere® Virtual Enterprise supports operators in the rules expressions. These operators are also referred to as *predicates* in Structured Query Language (SQL) terminology because they appear inside of a WHERE or HAVING clause. Operators are case insensitive.

Operators

Table 1. Request classification operators

Operator	Description
OR	The logical OR operator.
AND	The logical AND operator.
NOT	The negation operator.
IN	<p>Expresses an operand with multiple values in a single expression. Its meaning is consistent with the SQL standard meaning of the operator.</p> <p>For example, if you want to express that the port value could be any or all of the values such as 9080, 9090, 9091, use the expression fragment:</p> <pre>port IN (9080,9090,9091)</pre> <p>In SQL, how the values inside the parenthesis are expressed depends on the data type of port. If the port is an integer, the values without the single quotation marks are syntactically correct. If the port is a string, the correct expression is:</p> <pre>port IN ('9080', '9090', '9091')</pre>
LIKE	<p>Expresses pattern matching for string operand values. The value must contain the wildcard character (%) in the position where the pattern matching is expected to start.</p> <p>For example, the expression:</p> <pre>host LIKE %blanca</pre> <p>matches the word blanca and any other word that ends in blanca, while the expression:</p> <pre>host LIKE blanca%</pre> <p>matches the word blanca and any other word that starts with blanca. The expression:</p> <pre>host LIKE %blanca%</pre> <p>matches the word blanca and any word that has the token blanca imbedded in it.</p> <p>From a code implementation viewpoint, the <code>java.util.regex.Pattern</code> class is used.</p>
=	The equality operator expresses a match in a case-sensitive fashion.
>	Greater-than operator for use with numeric operands.
>=	Greater-than or equal operator for use with numeric operands.
<	Less-than operator for use with numeric operands.
<=	Less-than or equal operator for use with numeric operands.
< >	Not equal operator.
BETWEEN	Used with AND to select a range of values inclusive of the first (low) value and the last (high) value. Together, they operate on numbers and dates values.
IS NULL	Tests for an operand having a NULL value.
IS NOT NULL	Tests for an operand having a value other than NULL.

Parent topic: [Service policy rules](#)

Related concepts:

[Rule-based request classification](#)
[Work class types](#)

Related tasks:

[Defining a service policy](#)
[Setting maintenance mode](#)
[Activating concurrent application editions](#)
[Validating an edition](#)

Related reference:

[Subexpression builder operands for routing and service policies](#)

Related information:

[Rules for ODR routing policy administrative tasks](#)
[Rules for ODR service policy administrative tasks](#)

Rule-based request classification

With any rule-based technology, rule-based processing involves three basic areas: the vocabulary that forms the language, the grammar for expressing the vocabulary in statements, and the rule-processing engine.

Vocabulary

Vocabulary consists of the operators, variable keywords known as operands, and control flow statements. The language is the Java™ Message Service (JMS 1.1), and the Message Selector Syntax. The message selector is a string whose syntax is based on a subset of the SQL92 conditional expression. The general statement syntax is:

```
operand operator literalExpression [| compoundOperator expression] .
```

In the following example,

```
serverhost like '%ibm.com' or clienthost = 'myhost.raleigh.ibm.com'
```

`serverhost` and `clienthost` are operands, `like` and `=` are operators, `or` is the compound operator, and `'%ibm.com'` and `'myhost.raleigh.ibm.com'` are the `literalExpression`. The `[]` indicates that the expression enclosed within is optional, and the `|` indicates that after the operator, a `literalExpression` or a compound operator can be followed by more expressions.

The result of an expression is an action that is taken. From a grammatical viewpoint, these actions are literals that are provided by a policy provider. Two types of policies are supported:

1. Routing
2. Service

The actions that are taken are dictated by the policy provider. For routing, the actions are:

- permit
- reject
- permitsticky
- redirect

Each of the actions has the appropriate target; the recipient of an action. If the result of evaluating an expression is to take the action of permit, the target of that action is the application for which routing is permitted. For service policies, the target is encapsulated in the action, and the action is a transaction class.

A complete statement consists of the rule expression, and the action to be taken is represented differently depending on the input source. In the administrative console, the actions are separated into forms and fields that are easily selectable. If you are using scripting, the complete statement might look as follows:

```
expression<delimiter>action
For example, clienthost='localhost' and serverhost like
'%.ibm.com'?permit?DefaultApplication.ear
```

From an implementation viewpoint, work classes, which are XML documents, are used to capture the rules expression, matched actions, and other implementation artifacts. Therefore, a work class is an XML document that contains zero or more `matchRules` elements, and one or more `workClassModules` elements.

Operators

WebSphere® Virtual Enterprise supports the operators in the rules expressions. In general, you might not know the true data type of a given operand. If you use the Hypertext Transfer Protocol (HTTP), every operand is treated as a data type string and use the operator as an indicator for the real data type of the operand for data validation purposes. An example of an operator which tests for an operand that has a null value is: IS NULL.

Operands

When new protocols are added and new sets of operands defined, operands are valid within protocol scopes. If an operand is specified in a scope for which it is not valid, an error condition is indicated. In this release, supported protocols are:

- HTTP
- **Distributed platforms** JMS
- **z/OS** SOAP over HTTP represented simply as SOAP
- **Distributed platforms** Internet Inter-ORB Protocol (IIOP)

An example of an operand for HTTP requests is: MIMEType.

Parent topic: [Service policy rules](#)

Related reference:

[Routing and service policies](#)
[workclassoperations.py script](#)

Deploying and managing application editions without loss of service

The application edition manager increases application performance and manages interruption-free production application deployments. Interruption-free deployment prevents loss of service when you install an application update in your environment.

Before you begin

- Ensure that your product environment contains Version 7.0.
- If you are a user with either a monitor or an operator role, you can only view the edition manager information. If you have the role of configurator or administrator, you have all the configuration privileges for the application edition manager.

About this task

Deploy and use application editions when you install or update applications. You can perform an upgrade while maintaining continuous application availability.

[Application edition manager concepts](#)

By knowing the difference between application edition manager versions and editions, the methods of deploying and upgrading your application, and edition validation and compatibility, you can fully use the application edition manager to manage your application deployments.

[Installing an application edition](#)

Installing an application edition is similar to installing an application. The only difference is that during the process, you specify edition and edition description information.

[Activating an edition](#)

Three methods exist to activate an application edition: activating, performing a rollout to a new application edition, or validating the application edition. Edition activation distinguishes between two states in which an application edition might exist. When an application edition is first installed, the application edition is in the inactive state. You can start the application edition only when it is in the active state. The transition from inactive to active is known as activation.

[Creating routing policies for application editions](#)

Update your routing rules after you activate an application edition to ensure that the on demand router (ODR) can route requests to the appropriate application edition.

[Validating an edition](#)

Validating an edition is the process of determining if a new edition is available and ready to move into production and replace the current edition. You can install and validate a new edition under realistic conditions while your production application edition continues to serve requests.

[Performing a rollout on an edition](#)

When you perform a rollout on an edition, you replace an active edition with a new edition. The new edition might be a simple modification to the application, or contain a more substantial change. As long as the new edition is backward compatible, then you can perform a rollout to replace the active edition without impacting existing clients. To perform a rollout on a new edition, you must first install the application edition with the new edition information.

[Performing a rollback on an edition](#)

If you perform a rollout on a new edition and an error exists in your application, you can undo your change and easily revert back to the previous edition. To rollback to the edition, perform the rollout steps, but select a former edition to replace the current edition.

[Deleting an installed edition](#)

After multiple successive editions of an application are deployed and rolled out, which is not unusual in the life span of many production applications, you might have a number of editions stored in your repository that are no longer needed. You can delete an active edition without performing a rollout to a new edition.

[Troubleshooting application edition manager](#)

When you encounter a problem with the application edition manager, verify that the servers are running or refer to the log files to further investigate the errors..

Related tasks:

[Installing and customizing the product](#)

[Migrating WebSphere Virtual Enterprise](#)

[Preparing the hosting environment for dynamic operations](#)

[Managing the WebSphere Virtual Enterprise environment](#)

[Monitoring operations](#)

[Configuring WebSphere Virtual Enterprise to work with other IBM products](#)

Related reference:

[Reference](#)

[Administrative roles and privileges](#)

[Application edition management administrative tasks](#)

Related information:

[IBM WebSphere Virtual Enterprise Version 7.0](#)

[Product overview](#)

[Site map](#)

Application edition manager concepts

By knowing the difference between application edition manager versions and editions, the methods of deploying and upgrading your application, and edition validation and compatibility, you can fully use the application edition manager to manage your application deployments.

Unmanaged web applications can be defined with an edition. However, a rollout cannot be performed on them nor can they be placed into validation mode. Unmanaged web applications are supported with the exception that not all capability is available due to their nature as applications of assisted lifecycle management.

Versions and editions

The terms *version* and *edition* distinguish between what occurs in your development and build environment from what occurs in your deployment and operational environment. A version is a successive generation of an interface, function, implementation, or entire application, and so forth. Version is a development and build concept. An edition is a successive deployment generation, for example, the deployment of a particular set of versioned artifacts. Edition is a deployment and operational concept.

Application edition

An *application edition* is a unique deployment of a particular application. In the WebSphere® Application Server administrative environment, an application edition is an application that is uniquely identified by the combination of an application name and an edition name. Multiple editions of the same application have the same application name, but different edition names. The edition name can be numeric, such as 1.0 or 2.0, or the name can be descriptive, such as *first edition* or *blue edition*.

Base edition

Base edition refers to a deployed application that has no associated edition information. For example, applications that are installed before you add the application edition manager support to your product cell are displayed in the application edition manager as base editions.

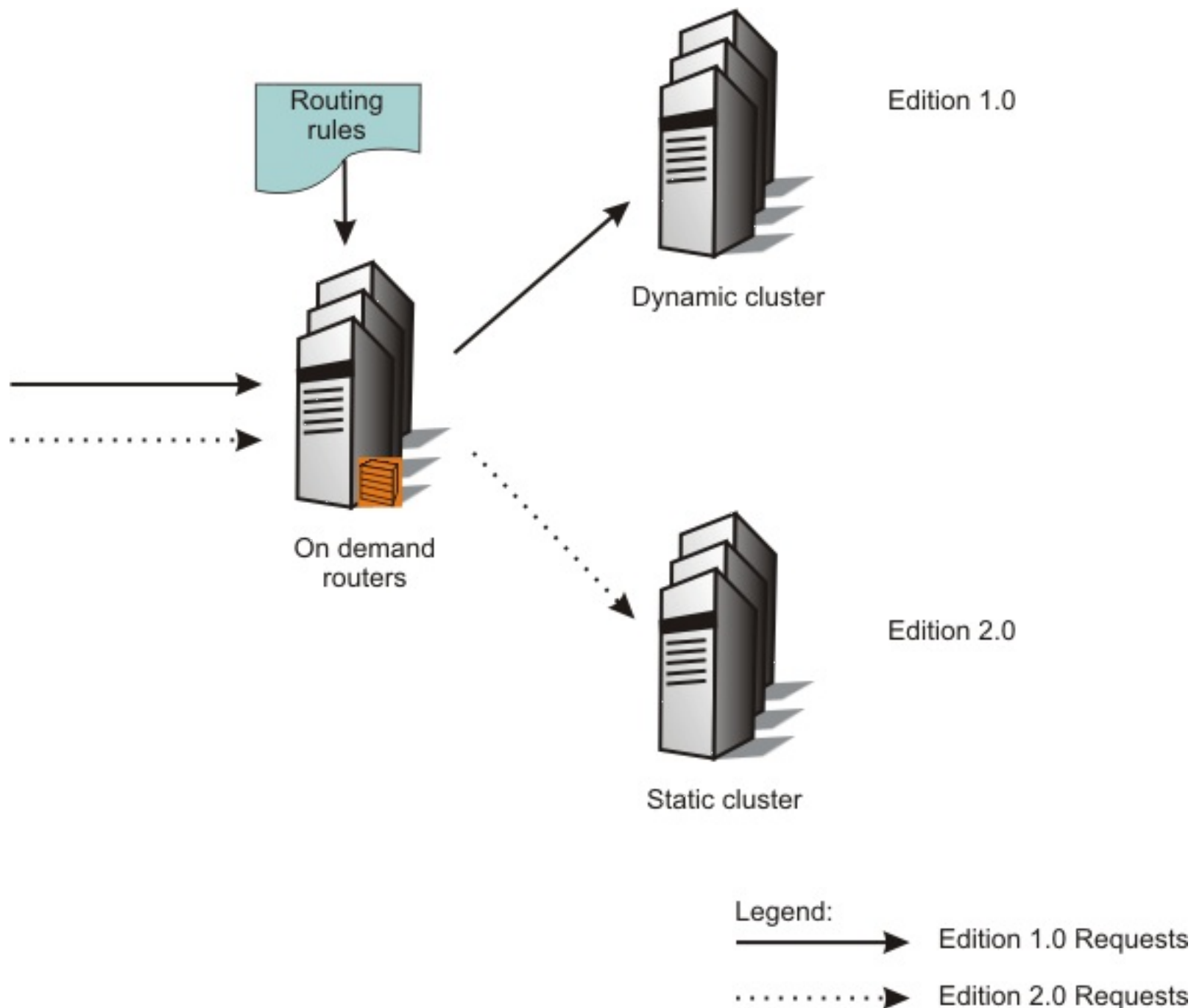
Edition activation

Edition activation distinguishes between two states in which an application edition might exist. When an edition is first installed, the edition is in the *inactive* state. You can start the edition only when it is in the *active* state. The transition from inactive to active is known as *activation*.

Concurrent activation

Concurrent activation exists when multiple editions of the same application are active and started simultaneously. Concurrently active editions can provide one set of users access to one edition and other users with access to another. For example, if you introduce a new edition of an application, you might want a select group of users to test the edition, and not want all users to have access to the edition. With concurrent activation, you must establish a *routing policy* to distinguish which users have access to an edition. A routing policy is stored as part of the configuration metadata for an application. Also, a routing policy prevents ambiguity and determines which edition receives control. The following example is a diagram of concurrently active editions.

Figure 1. Concurrently active editions



Application upgrade and deployment

Many business applications require constant availability. The standard for application availability asserts that applications are deployed on application server clusters. The redundancy of a cluster is essential to provide continuous availability. *Interruption-free application upgrade* refers to the ability to upgrade while maintaining application continuous availability. In other words, users of the application experience no loss of service during the application upgrade.

Edition rollout

When you perform a *rollout* to an edition, you replace an active edition with a new edition. To provide interruption-free application upgrades, performing a rollout to an edition includes the following items:

- Fencing a server from receiving new requests.
- Quiescing requests for the application in a particular server.
- Stopping the currently active edition.
- Starting the new edition.
- Resuming the flow of requests to the edition.

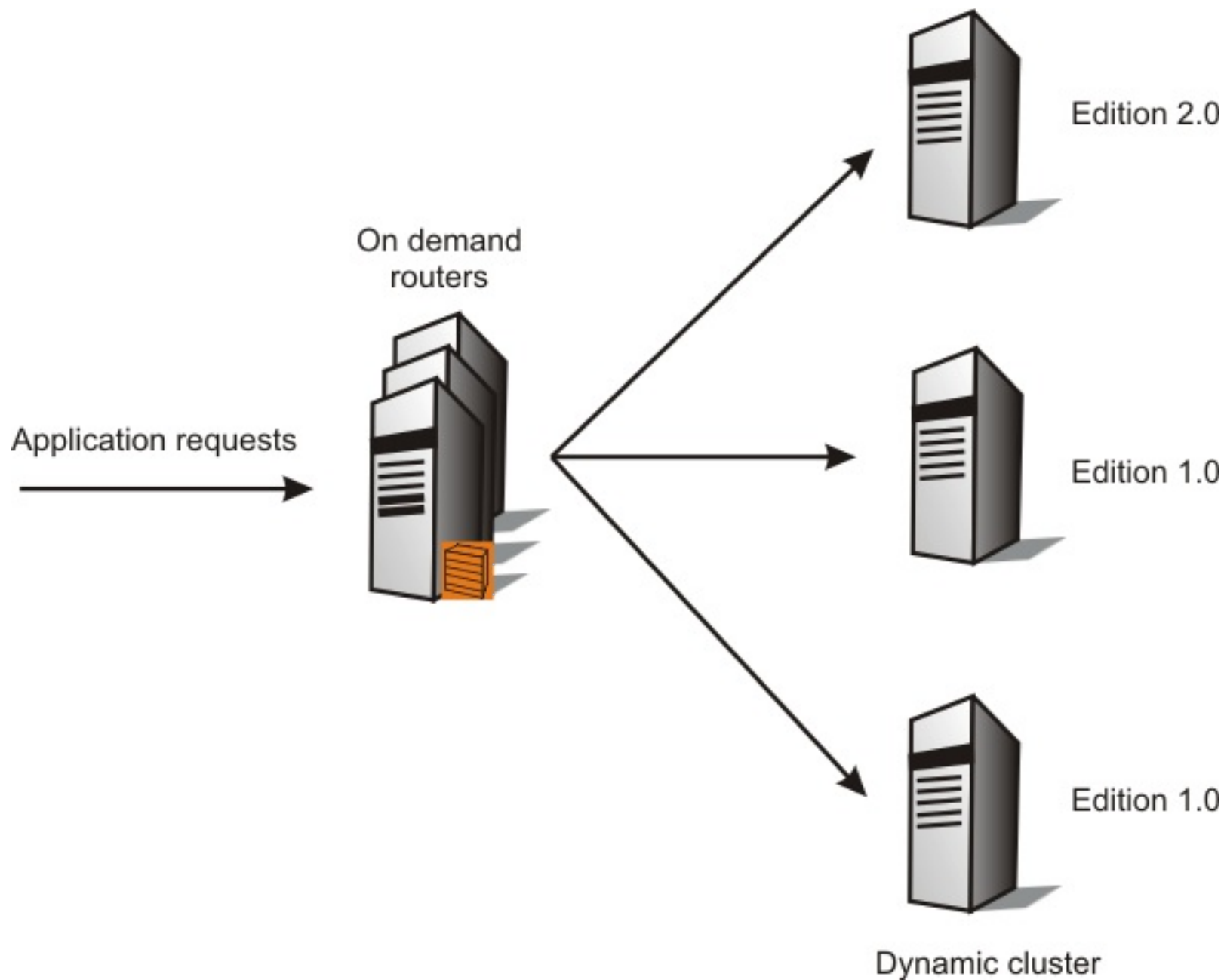
When you perform a rollout to an edition across an application server cluster, you complete the following activities across the set of the servers that are in that cluster:

Group rollout

To perform a rollout to a target cluster, you can divide the cluster into groups, and define a group size, which specifies the number of nodes to process at a time. Performing a rollout to a group results in the servers in each group being upgraded to the new edition at the same time. Each server in the group is quiesced, stopped, and reset. A rollout can be performed on only one group at a time in the administrative console. When any member in the new edition becomes available, all requests are routed to the new edition.

As you perform a rollout to the edition, some servers in the cluster move from the previous edition to the new edition, some servers are in the process of making the transition, and other servers have not started the transition. All application requests are sent to any server that has an active, running instance of the latest edition of the requested application. For example, when you perform a rollout from edition 1.0 to 2.0, all application requests are served by edition 2.0 when edition 2.0 becomes available on a server. Any servers that are still running edition 1.0 do not serve requests until this server is updated to edition 2.0. The following example is a diagram of a group rollout.

Figure 2. Group rollout

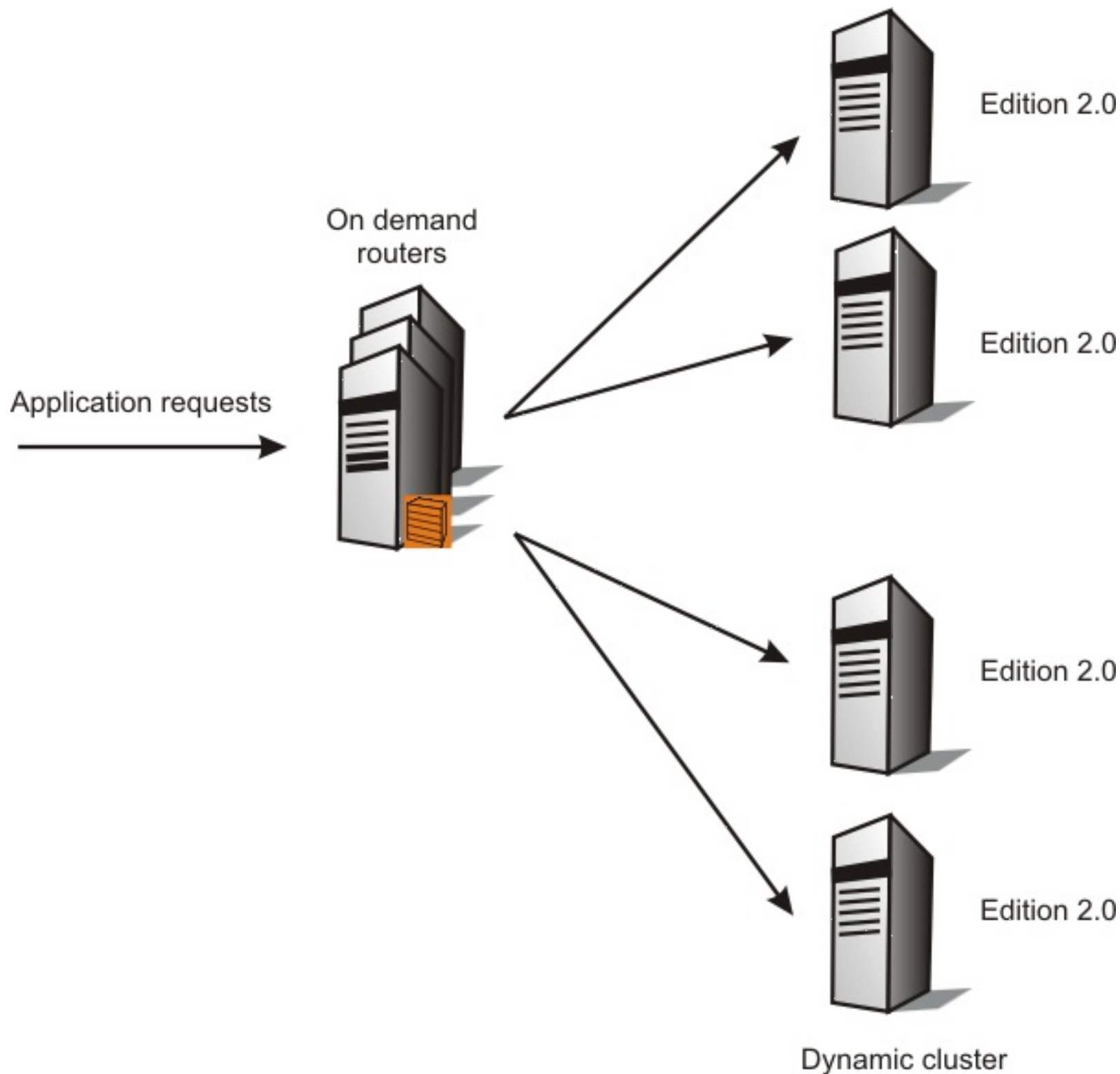


Atomic rollout

Performing an atomic rollout to an edition replaces an edition on half of the cluster at a time to serve all user requests with a consistent edition of the application. All user requests are served by either the previous or the new edition; user requests are never served by both editions.

An atomic rollout ensures that all application requests are served by a consistent edition, for example, either edition 1.0 or 2.0, but not by both. The currently available edition is taken offline from half of the servers that comprise the cluster. In those servers, the new edition is activated and started, but those servers are held offline until the next step completes. The next step is to take the currently active edition offline in the remaining servers. At this point, no server has an instance of either edition available to serve application requests. The ODR temporarily queues any request that arrives for this application. After the application is fully offline, the first half of the cluster is brought back online. The second half of the cluster transitions from the previous edition to the next edition and is brought back online. The following example is a diagram of an atomic rollout:

Figure 3. Atomic rollout



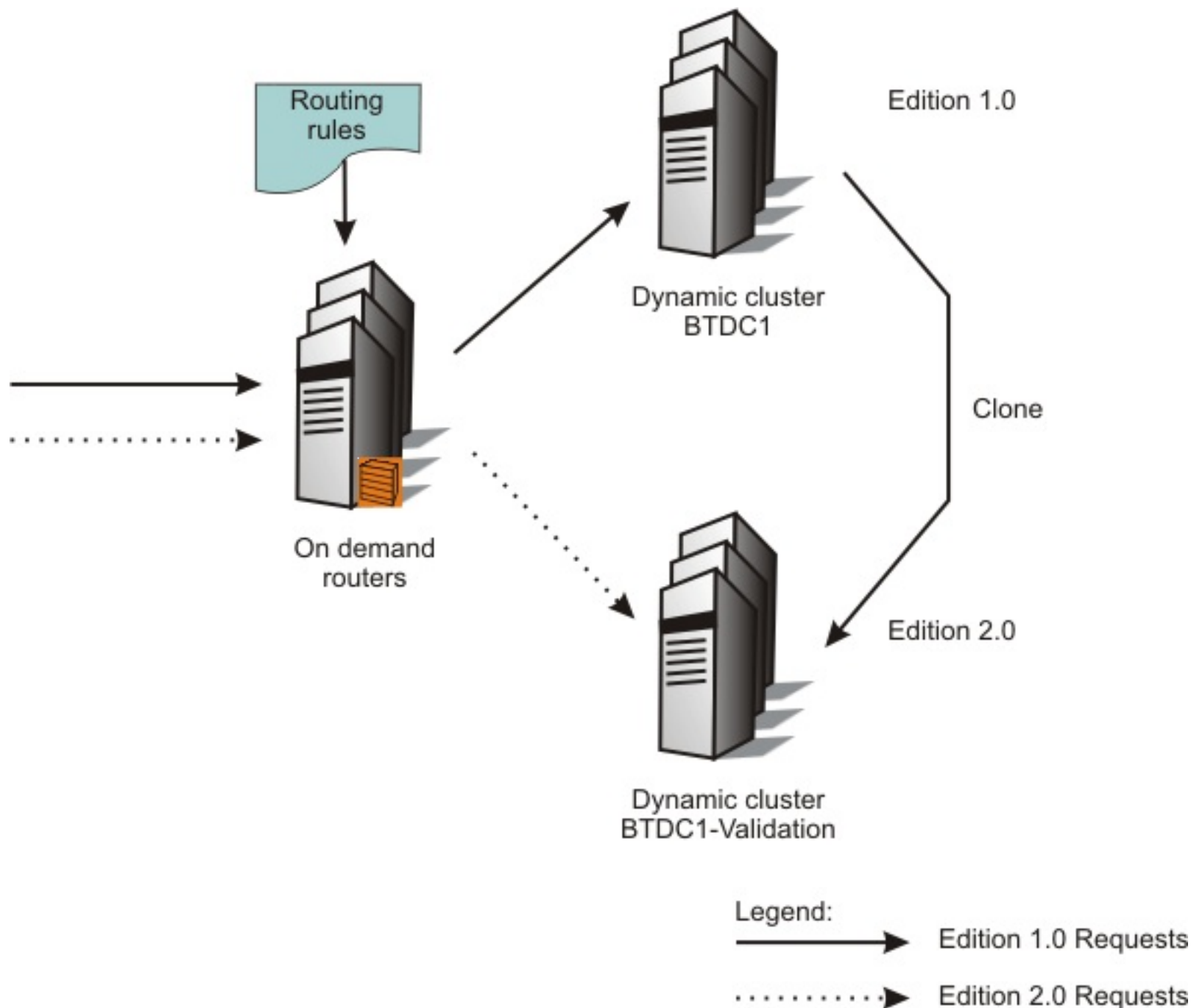
Edition validation and compatibility

Edition validation

Edition validation refers to a special case of concurrent activation, where an assigned deployment target of the edition, for example, a dynamic cluster, is cloned and the edition is made ready to be started on the cloned deployment target. The cloned deployment target is known as a *validation target*. Routing rules must be used to designate which application requests are sent to the edition undergoing validation. When an edition is in validation, it is in *validation mode*.

Validation mode ensures that a new edition of an application functions in its production environment without taking the currently available edition offline. Typically, a test load is sent to an edition in validation mode to confirm that aspects of the application environment and setup, such as connectivity and database access, work as expected. When you perform a rollout to an edition validation mode, the operation occurs on the deployment target on which the edition was originally installed. Performing a rollout causes the edition to exit validation mode. When validation completes, the validation target is deleted. The following example is a diagram of edition validation.

Figure 4. Edition validation



Edition compatibility

Some application changes are transparent, while other application changes are seen by the end user. When an application upgrade delivers at least the same application programming interfaces as the last change did and no semantic change to essential behavior, that application edition is an upgrade that is *compatible* with previous versions. As an existing user, you can use the upgraded application without changing how you use it and without recognizing a difference between the current and former editions.

An application upgrade that requires existing users to change how they use the application is an *incompatible* upgrade. You might need to drop former function, or change interfaces, for example, to improve maintainability or other factors, and might introduce incompatible changes to your deployment environment. Incompatible changes require careful planning to manage the impact to existing users.

[Application edition manager](#)

The application edition manager for WebSphere Virtual Enterprise manages interruption-free production application deployments. Interruption-free deployment prevents loss of service when you install an application update in your environment.

[Application edition manager states](#)

All applications can be in the running or stopped runtime states, and in the nonexistent or existent configuration states. WebSphere Virtual Enterprise application editions have additional configured states beyond those of WebSphere Application Server applications. Consider these additional configuration states as sub-states of the existing state.

[Operational environment](#)

The operational environment in which the application edition manager runs is a WebSphere Virtual Enterprise cell.

[Edition compatibility](#)

The application edition manager supports only compatible application upgrades for edition roll out, which means that interruption-free upgrade is only for editions that are compatible with previous editions. When you deploy editions with incompatible changes, the editions require the concurrent activation pattern and use of routing rules to separate request traffic between users of the former edition and users of the current edition. With concurrent activation, you can host multiple editions of the same application simultaneously with each edition supporting a distinct, non-intersecting set of users. However, concurrent activation might not provide interruption-free upgrade.

Parent topic: [Application edition manager](#)

Parent topic: [Deploying and managing application editions without loss of service](#)

Related reference:

[Application edition manager states](#)

Application edition manager

The application edition manager for WebSphere® Virtual Enterprise manages interruption-free production application deployments. Interruption-free deployment prevents loss of service when you install an application update in your environment.

The application edition manager provides an application versioning model that supports multiple deployments of the same application in a WebSphere Virtual Enterprise cell. Each deployment has a unique edition name. The application edition manager offers a choice of which edition to activate on a WebSphere Virtual Enterprise cluster, so that you can perform a rollout of an application update or revert to a previous level.

The application edition manager is fully integrated with WebSphere Virtual Enterprise, interacting with the on demand router (ODR), dynamic workload balancing, and the application placement manager. This integration ensures predictable application behavior when you apply application updates, and a smooth transition from one application edition to another while the system continues to manage your application performance goals. You can access application update processes with the administrative console, including edition activation across the application servers. Scripting application programming interfaces enable the integration of edition management functions with automated application deployment.

Application editions are supported for Java™ Platform, Enterprise Edition (Java EE) applications, including enterprise archive (EAR) files, and Compute Grid applications that conform to one of the grid programming models. The following table summarizes the supported features, which vary depending on the type of application that you are using.

Table 1. Application edition manager support for applications

Application type	Multiple editions of applications	Application rollout	Interruption-free application update, driven by the ODR and HTTP communication	Validation mode	Concurrent activation
Managed Web applications, including PHP Hypertext Preprocessor (PHP) applications	Supported	Supported	Supported	Supported	Supported
IIOP, EJB, or JMS applications	Supported	Supported	Not supported ¹	Not supported	Not supported
Unmanaged Web applications	Supported	Not supported	Not supported	Not supported	Supported
WebSphere Application Server Community Edition applications	Supported	Supported	Supported	Supported	Supported
Session Initiation Protocol (SIP) applications	Supported	Supported	Supported ²	Supported	Supported
Converged HTTP-SIP applications	Supported	Supported	Supported ²	Supported	Supported

¹ The ODR does not support communication with IIOP. Any EJB, JMS, or IIOP components called directly by an external client cannot utilize this feature.

² This feature is driven by the ODR and HTTP/SIP communication.

WebSphere Application Server includes an administrative function called *rollout update*. Rollout update provides a basic application upgrade, but is not interruption-free. With WebSphere Virtual Enterprise installed, the rollout update function is available, but the application edition manager is the preferred way to upgrade applications.

The application edition manager supports your overall application life cycle, and enables application updates and seamless, interruption-free application deployments to your production environment.

Parent topic: [Application edition manager concepts](#)

Related concepts:

[Application edition manager concepts](#)

Application edition manager states

All applications can be in the running or stopped runtime states, and in the nonexistent or existent configuration states. WebSphere® Virtual Enterprise application editions have additional configured states beyond those of WebSphere Application Server applications. Consider these additional configuration states as sub-states of the existing state.

Application edition manager editions transition into multiple states. The transition of an application between the running and stopped runtime states occurs through start and stop operations. The transition of an application between the non-existent or existent configuration states occurs when you install or uninstall the application. As the application edition transitions from one state to another, various actions occur, such as installing, validating, activating, performing a rollout, deactivating, or uninstalling. The following table features a list of the transitions and the resulting actions:

Table 1. State transitions

From state	To state	Transition action	Notes
Nonexistent	Inactive	Install application edition	None
Inactive	Validation	Validate ¹	Validation deployment target is created.
Inactive	Active	Activate or perform a rollout ¹	Performing a rollout deactivates the previous edition.
Validation	Active	Perform a rollout ¹	Validation deployment target is deleted.
Active	Inactive	Deactivate ¹	None
Active	Nonexistent	Uninstall application edition	None
Validation	Inactive	Cancel validation	Validation deployment target is deleted.
Inactive	Nonexistent	Uninstall application edition	None

¹Validate, activate, perform a rollout, and deactivate are new application edition operations introduced by the application edition manager.

Parent topic: [Application edition manager](#)

Parent topic: [Application edition manager concepts](#)

Related concepts:

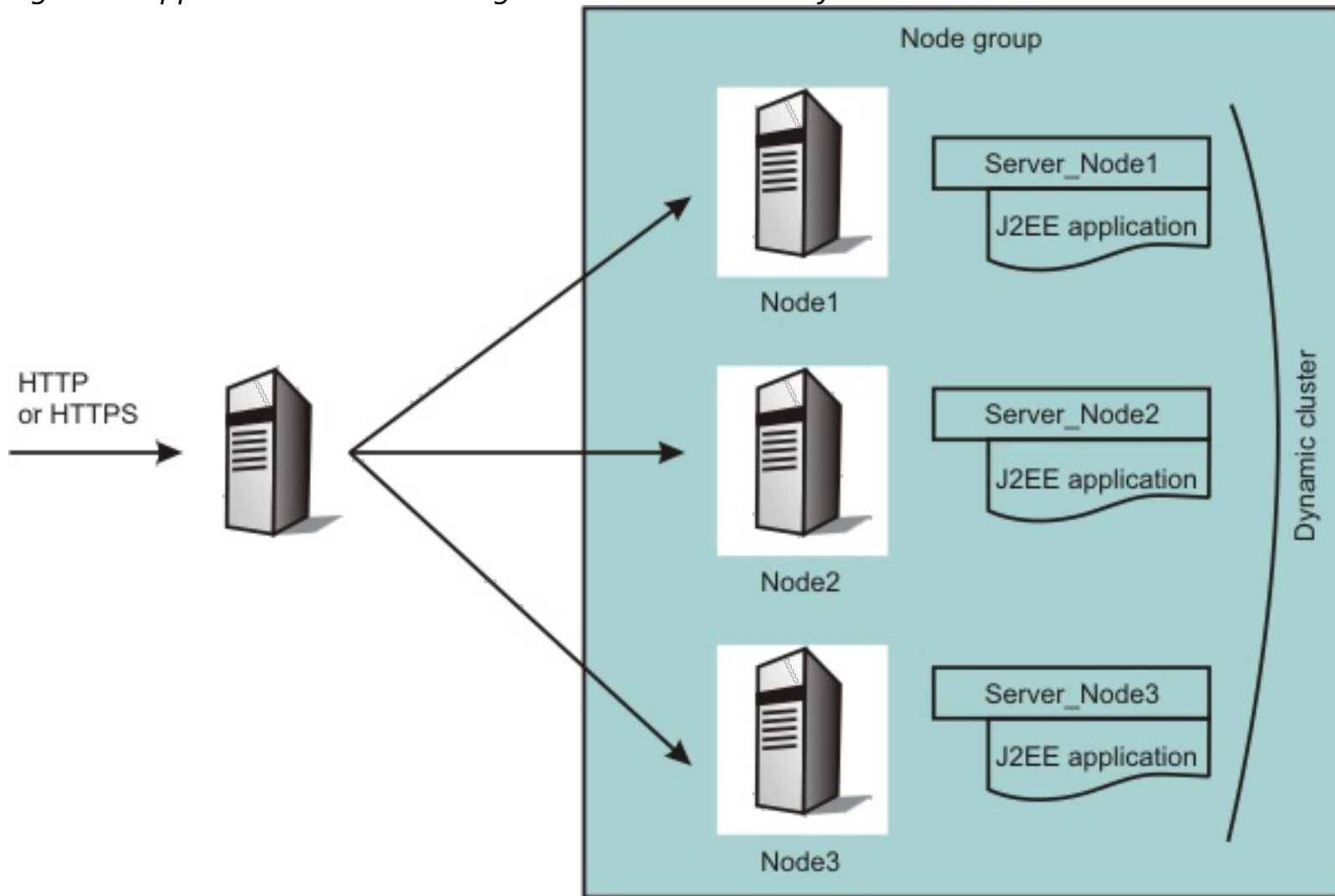
[Operational environment](#)

Operational environment

The operational environment in which the application edition manager runs is a WebSphere® Virtual Enterprise cell.

In the following illustration, the application edition manager manages applications that are deployed to dynamic clusters that receive work requests through an on demand router (ODR).

Figure 1. Application edition manager with an ODR and dynamic cluster



The SIP protocol is supported in WebSphere Virtual Enterprise Version 6.1.0.1 and later.

The application edition manager provides support for interruption-free application upgrades only for applications accessed through the ODR by way of the hypertext transfer protocol (HTTP) or hypertext transfer protocol running under the secure sockets layer (HTTPS). Service continuity during application upgrade is not assured for inter application access, for example, one Java™ Platform, Enterprise Edition (Java EE) application calling another, unless the inter application access is accomplished by way of HTTP or HTTPS through another ODR layer.

Parent topic: [Application edition manager](#)

Parent topic: [Application edition manager concepts](#)

Related concepts:

[Edition compatibility](#)

Edition compatibility

The application edition manager supports only compatible application upgrades for edition roll out, which means that interruption-free upgrade is only for editions that are compatible with previous editions. When you deploy editions with incompatible changes, the editions require the concurrent activation pattern and use of routing rules to separate request traffic between users of the former edition and users of the current edition. With concurrent activation, you can host multiple editions of the same application simultaneously with each edition supporting a distinct, non-intersecting set of users. However, concurrent activation might not provide interruption-free upgrade.

Consider the following compatibility issues when you deploy editions:

- **Application interfaces or semantics:** Active users who are currently using the application might be affected if application interfaces or semantics are changed between editions when a roll out is attempted. Examples of changes include those to existing interfaces, including modification or removal of an existing interface. Also, a change to the semantic behavior of an interface might also affect active users. For example, if an interface formerly allowed a parameter to be null, then the change requires that the same parameter is non-null. Changes that impact existing clients are not considered to be backward compatible and cannot be a part of an interruption-free upgrade. If impact to existing clients is not an issue, then use the WebSphere® Application Server roll out update.
- **Hypertext transfer protocol (HTTP) session state:** If the HTTP session state is persisted or replicated, then application changes that add or change the data types that are stored in session also represent incompatible change. The current edition might be unable to use the session state created by a former edition.
- **Web content caching:** If a new application edition includes changed static web content and you are using the on demand router (ODR) to cache content, then you might need to flush the cache as part of the edition roll out. For more information, see the [Cache monitor utility](#) topic in the WebSphere Application Server information center to perform this task manually.

Parent topic: [Application edition manager concepts](#)

Related concepts:

[Application edition manager](#)

↳ [Cache monitor](#)

[Operational environment](#)

Related tasks:

↳ [Updating applications with the console](#)

Installing an application edition

Installing an application edition is similar to installing an application. The only difference is that during the process, you specify edition and edition description information.

Before you begin

- Before you install your application edition, you must create a deployment target for your application.
- If you are a user with either a monitor or an operator role, then you can view the application edition manager information only. If you have the role of configurator or administrator, then you have all the configuration privileges for the application edition manager.

About this task

Multiple versions of the same shared library are installed if you give them different names during creation. Use a naming scheme that adds a version number to the shared library name, for example, Production Library 1.0 and Production Library 2.0. Each edition of an application is updated to use a distinct version of the shared library by binding the edition to the appropriate library. You can also clone an existing edition.

Procedure

1. Begin the steps to install the application. In the administrative console, select **Applications > Install new application** or **Applications > Install new middleware application**. Enter the following information in the wizard panels.
 - a. Specify the application type.
 - b. Specify the EAR, WAR, JAR, or SAR module to upload and install.
 - c. Optional: **For enterprise applications:** If you want to clone work classes from another edition of the application, select **Show me all installation options and parameters**.
2. In the **Application edition** field, specify your edition information. For example, type 1.0. The edition identifier does not have to be numeric and can be any combination of letters, numbers, and certain special characters, such as ~!@#\$. -.

Session Initiation Protocol (SIP), Java™ EE applications: The edition identifier becomes part of a directory name under which the application is installed. Any character that you use in a file name on your system can also be used in the edition identifier.

PHP Hypertext Preprocessor (PHP), WebSphere® Application Server Community Edition, unmanaged web applications: The edition identifier becomes a configuration attribute in the repository and accepts any characters.

3. Specify the installation options. In the **Application description** field, specify the type of edition you are installing. For example, type First edition.
4. Specify the deployment targets for the application.
 - **Enterprise applications, Session Initiation Protocol (SIP) applications:** If you are installing an enterprise application, you can use split deployment. With split deployment, you deploy a module in a single Java EE application archive and divide the module across multiple deployment targets. For example, you might deploy an enterprise archive (EAR) file that contains a web application module and the enterprise beans module is installed in the WebSphere Virtual Enterprise environment. As a result, the web application module is installed on a server and the enterprise beans module is deployed on a cluster.
 - **Unmanaged Web applications:** Define the deployment properties of the application. Specify module, context root, and virtual host information for the application. Choose the servers and clusters on which the application is deployed by selecting deployment targets.
 - **PHP, WebSphere Application Server Community Edition applications:** Choose the servers and clusters on which the application runs by selecting the deployment targets. Choose PHP servers or PHP dynamic clusters for PHP applications. Choose WebSphere Application Server Community Edition servers or dynamic clusters for WebSphere Application Server Community Edition applications.
5. Optional: **If you are installing an enterprise application:** To save time and reuse work classes, you can choose an existing edition of the application to clone. On the first administrative console panel of the application installation, you must select the **Show me all installation options and parameters** option to see the clone work classes option. From the **Clone existing work classes from this application edition** list, select the work class, and click **Next**. The work class establishes default routing rules for the application edition. The work classes of an application constitute the routing policy of that application. If you select an existing application edition, all of its work classes are cloned with the defaults appropriately renamed with the new application edition name. If you do not select an application edition, the defaults are created.
6. Specify the location of the virtual host for the web modules, and edit the context root that is defined in the deployment descriptor file.
7. Save and synchronize your nodes.
8. Start the application.
 - **Enterprise applications:** In the administrative console, select **Applications > Enterprise applications > application_name > Start**.

- **PHP applications:** In the administrative console, select **All applications**. Select the PHP application that you want to start. Choose the **Start** action, and click **Submit**. This action starts all of the PHP servers that are associated with this application. To start the servers individually, select **Servers > Other middleware servers > PHP servers**.
- **Unmanaged web applications:** The application is displayed as started when the server on which the application is deployed starts.
- **WebSphere Application Server Community Edition applications:** Start application editions individually. In the administrative console, select **Applications > Edition control center > application_name > Activate**.

What to do next

Now that your application edition is installed, you can perform a rollout, backout, concurrent activation, or validation.

Parent topic: [Deploying and managing application editions without loss of service](#)

Related tasks:

[Performing a rollout on an edition](#)

[Deploying unmanaged Web applications](#)

[Deploying PHP applications](#)

[Deploying WebSphere Application Server Community Edition applications](#)

[Deploying enterprise applications](#)

Related reference:

[Administrative roles and privileges](#)

[Application edition management administrative tasks](#)

Activating an edition

Three methods exist to activate an application edition: activating, performing a rollout to a new application edition, or validating the application edition. Edition activation distinguishes between two states in which an application edition might exist. When an application edition is first installed, the application edition is in the inactive state. You can start the application edition only when it is in the active state. The transition from inactive to active is known as activation.

Before you begin

- You must have an application installed and in the inactive state.
- To activate an application edition, you must have a configurator or administrator administrative role.

Note: The first edition of an application installed before any other editions is automatically activated, increasing its availability and making it eligible to be started.

Procedure

1. In the administrative console, select **Applications > Edition control center**.
2. Select the application edition that you want to activate.
3. Complete one of the following actions.
 - a. Click **Activate**. Activating marks an application edition as available to be started. After you activate an application edition, you must update the routing rules, and click **Apply** and **Save** regardless of what item is displayed in the list. See [Creating routing policies for application editions](#) for more information.

Note: After you update the routing rules, you must click **Apply** and **Save** even if only one edition of the application is displayed in the menu list.
 - b. Click **Roll out**. Performing a rollout to an application edition activates one edition in place of another. The new application edition automatically starts because it replaces a running application edition. See [Performing a rollout on an edition](#) for more information.
 - c. Click **Validate**. Validation activates an application edition on a clone of its original deployment target. See [Validating an edition](#) for more information.

Results

The application edition is activated.

What to do next

If you used the activate or validate options, you can start your edition application.

[Activating concurrent application editions](#)

Activate multiple editions of the same application concurrently when you are validating before production, piloting an application to a select group of users, or rolling out a branch when an application upgrade requires a corresponding change on identifiable branches of client machines.

Parent topic: [Deploying and managing application editions without loss of service](#)

Related tasks:

[Performing a rollout on an edition](#)
[Validating an edition](#)

Related reference:

[Administrative roles and privileges](#)
[Application edition management administrative tasks](#)

Activating concurrent application editions

Activate multiple editions of the same application concurrently when you are validating before production, piloting an application to a select group of users, or rolling out a branch when an application upgrade requires a corresponding change on identifiable branches of client machines.

Before you begin

- You must have at least two editions of the same application installed. For example, the `my_application` application edition 1.0 is installed on the `dynamic_cluster_1` dynamic cluster, and application edition 2.0 is installed on the `dynamic_cluster_2` dynamic cluster.
- Privileges for the application edition manager differ, depending on the various roles. Roles include monitor, operator, configurator, and administrator. If you are a user with either a monitor or an operator role, you can only view the application edition manager information. If you have the role of configurator or administrator, you have all the configuration privileges for the application edition manager.

About this task

Each application edition must be active on a separate deployment target. When multiple editions of the same application are concurrently available to users in the same environment, the on demand router (ODR) cannot differentiate between the active editions without some information available to process the request and route it to the intended edition. You can use routing rules or unique interfaces for each application edition to prevent ambiguity.

Restriction: Only one active edition of a PHP Hypertext Preprocessor (PHP) application is supported on a node. If you have active editions of the same PHP application, do not deploy the application to servers that are on the same node.

Procedure

1. Activate the application editions. Click **Applications > Edition control center > *application_name***. Select the inactive edition, and click **Activate**. For example, select the `my_application` application and activate application edition 2.0.
2. Create routing policies for each application edition. See [Creating routing policies for application editions](#) for more information.
3. Verify that the ODR is running. Click **Servers > On demand routers**. To route requests, the status must be **Started**.
4. Test the concurrent access to application editions. Select the two application editions by selecting the servers associated with the two dynamic clusters, and click **Start**.

Results

Edition 1.0 is serviced by the routing rule that you create for that particular edition, and edition 2.0 is serviced by its specific routing rule.

Example

To run a preproduction test of an application edition in the production environment with a selected set of users, you can clone the deployment target, including its resource and security definitions, and activate the target edition on the cloned environment. Use routing rules to direct the ODR to divert a selected subset of users to the application edition.

Additionally, to pilot your application, you can use routing rules to separate the pilot users on edition 2.0 from the general users on edition 1.0.

In the case of branch rollout, use routing rules to direct each branch to the appropriate edition. As the client code at each successive branch updates, the server-side routing rules can be updated to qualify the clients from the newly updated branch to be sent to the appropriate edition.

For cases where the routing rules are insufficient to differentiate user requests or where the user prefers an alternative to routing rules, each edition can be given its own unique URI and Enterprise JavaBeans (EJB) Java™ Naming and Directory Interface (JNDI) name. Unlike routing rules, unique interfaces for each edition are exposed to the application users. Therefore, you must choose the appropriate name to drive the appropriate edition.

What to do next

Perform validation to test the availability and resiliency of your new edition under realistic conditions. See [Validating an edition](#) for more information.

Next topic: [Validating an edition](#)

Parent topic: [Activating an edition](#)

Related concepts:

[Application edition manager concepts](#)

Related tasks:

[Installing an application edition](#)

[Performing a rollout on an edition](#)
[Performing a rollback on an edition](#)
[Creating routing policies for application editions](#)

Related reference:

[Routing and service policies](#)
[Administrative roles and privileges](#)

Creating routing policies for application editions

Update your routing rules after you activate an application edition to ensure that the on demand router (ODR) can route requests to the appropriate application edition.

Before you begin

Activate your application edition.

About this task

A routing policy is stored as part of the configuration metadata for an application. With a routing policy, you can express rules that instruct an on demand router (ODR) to send particular application requests to one edition or another based on a set of criteria. You can use various criteria that specify which requests are sent to a particular application edition. Use this process to send requests from certain users to one edition and requests from other users to another edition.

Note: If your application has an existing application edition with a defined multi-cluster routing policy and you install a new edition, you must create a new multi-cluster routing policy for that specific edition. Use the administrative tasks to add multi-cluster routing policies for your new editions. See [Rules for ODR routing policy administrative tasks](#) for more information.

Procedure

1. Navigate to the routing policies for the application. Click **Applications > Enterprise applications > *application_name*** or **Applications > All applications > *application_name***. Click the **Routing policies** tab. For example, select the `my_application` application.
2. Expand **Work classes for HTTP requests**. Because no routing rules are specified, all requests are routed to the edition that is displayed on this page. For example, all requests are routed to application edition, `my_application-edition2.0`.
3. Click **Rule builder**.
4. From the rule list, select a rule. For example, select **Client host (clienthost)**, and click **Add**.
5. Select criteria for the rule. For example, select an operator of **Equals (=)** and type a value of your client host name. Click **OK**.
6. Click **OK** again.
7. Expand **Work classes for HTTP requests**.
8. Set the action associated with the new rule. For example, requests from the host route to edition, `my_application-edition1.0`. Select the corresponding action from the **Then** list, and click **Apply** to save the rule.
9. From the top of the **Routing policies** tab, click **Apply**.
10. Save the changes to the configuration repository and synchronize the nodes.

What to do next

If you are routing to multiple editions of the same application, you can validate your new application edition before moving to the new edition. See [Validating an edition](#).

Next topic: [Validating an edition](#)

Parent topic: [Deploying and managing application editions without loss of service](#)

Related concepts:

[Application edition manager concepts](#)

Related tasks:

[Installing an application edition](#)

[Performing a rollout on an edition](#)

[Performing a rollback on an edition](#)

[Activating concurrent application editions](#)

Related reference:

[Routing and service policies](#)

[Administrative roles and privileges](#)

[Application edition management administrative tasks](#)

Validating an edition

Validating an edition is the process of determining if a new edition is available and ready to move into production and replace the current edition. You can install and validate a new edition under realistic conditions while your production application edition continues to serve requests.

Before you begin

- **Note:** Edition validation does not work properly for applications deployed on a dynamic cluster that has been converted from a static cluster in Version 7.0. If you are using Version 7.0.0.1 or later, you do not encounter this issue.
- You must ensure that all of the modules for your application are deployed to the same deployment targets.
- You must define unique routing rules for edition 2.0. Routing rules enable editions to run concurrently and hypertext transfer protocol (HTTP) requests for the validation edition to correctly route to the validation target without interfering with edition 1.0. For this scenario, use the `my_application` application. Install both application editions, 1.0 and 2.0, on the `dynamic_cluster_1` dynamic cluster. See [Creating routing policies for application editions](#) for more information about creating routing rules for application editions.
- To set the operational mode of the cloned validation cluster to a different mode from the production cluster, create the `VALIDATION_OPERATIONALMODE` custom property in the administrative console. Otherwise, the validation cluster is set to the same operational mode as the production cluster after it is created. Set the value to `automatic`, `manual`, or `supervised`. If you specify any other value or you do not specify a value, the validation dynamic cluster is set to manual mode.
- **Restriction:** Only two cluster members can be used or created in validation mode. You can map routing and service policies to the validation mode application, but no more than two cluster members are started to maintain the work. You can overwrite this setting after the validation cluster is created by changing the minimum and maximum number of dynamic cluster instances.
- If you are a user with either a monitor or an operator role, then you can only view the application edition manager information. If you have the role of configurator or administrator, then you have all the configuration privileges for the application edition manager.
- To validate a new edition, add the validation cluster to the service integration bus (SIB).
- If you are running a release that is previous to Version 6.1.0.5, only one application per deployment target can be in validation mode at a single point in time. With Version 6.1.0.5 and later, you can have multiple applications in validation mode on the same deployment target. However, these applications must be different and cannot be different editions of the same application.

About this task

Consider the following scenario as an example of how validation is performed on an edition: Edition 1.0 of an application is installed, active, and running on a dynamic cluster. Edition 2.0 is the candidate validation edition and is installed on the same deployment target in the inactive state. Validating edition 2.0 clones the edition 2.0 deployment target. For example, the validation might create a new dynamic cluster, such as the `DC-Validation` dynamic cluster, and map edition 2.0 to this new cluster. The cloned cluster uses the existing cluster members as the server template for the creation of the cloned servers.

After the validation clone target is created, edition 2.0 is activated, and the routing rules are defined, you can start, stop, and reconfigure the edition.

Procedure

1. Click **Applications > Edition control center** to verify that the application has two installed editions, with only one active edition.
2. Optional: If you want to create a validation cluster that has a different operational mode than your production cluster, you can define the `VALIDATION_OPERATIONALMODE` custom property on the production cluster. Add the validation cluster to the service integration bus (SIB). If you do not define this custom property, your validation cluster has the same operational mode as the production cluster.
3. Update the Enterprise JavaBeans (EJB) reference bindings to specify the new cluster name. Before rolling out the application from the validation cluster, the bindings must be changed back to the original value.
4. Click the `my_application` application.
5. Select edition 2.0 and click **Validate**. The validation status page shows each step of validating the `dynamic_cluster_1` dynamic cluster and deploying edition 2.0 to the cloned cluster. The application edition control center shows that one of the editions is in validation mode, and the manage editions page shows that the edition 2.0 target is now the `dynamic_cluster_1-Validation` dynamic cluster. The dynamic cluster page shows that the `dynamic_cluster_1-Validation` dynamic cluster is created, and the servers page shows the cloned servers.

Tip: If you want to save the validation cluster after you perform the rollout, you can create the `saveClonedCluster` custom property on the validation cluster. Otherwise, the validation target is deleted after the edition rollout or after the validation is canceled for all of the applications on the validation target. For example, if you have two applications deployed to the validation target, and one of the applications is validated and rolled out, the validation target is not deleted until the second application is validated. The `saveClonedCluster` custom property applies only to dynamic clusters. For more

information, see [Application edition manager custom properties](#).

6. Verify that the validation occurred correctly. Click **Applications > Enterprise applications** or **Applications > All applications**. Edit the my_application-edition2.0 application.
 - **For PHP and WebSphere® Application Server Community Edition applications:**

Verify that the context root, deployment targets, and so on are pointing to the cloned cluster.
 - **For enterprise (Java™ Platform, Enterprise Edition (Java EE)) applications:**

Select **Manage modules**. Verify that edition 2.0 is mapped to the validation cluster. From the Map Enterprise JavaBeans (EJB) references to beans detail view, verify that the Java Naming and Directory Interface name is adjusted for the new cloned target name.

For an application edition with fully qualified bindings based on the original deployment target name to operate correctly on a validation deployment target, you must change its binding names to reflect the fully qualified binding names based on the validation deployment target name. For example, an application with a resource reference bound to /clusters/clusterb1/jdbc/CustomerData must have the binding changed to /clusters/cluster1-validation/jdbc/CustomerData as the application is prepared to run on the deployment target clone.
7. Test the new edition. Start the validation cluster, and with your routing rules in place, try sending a request load to the edition 2.0 edition to test the edition. The edition 1.0 edition remains in production.

What to do next

If you successfully complete the edition 2.0 edition testing, you can replace the edition 1.0 edition with the edition 2.0 edition. If you encounter errors in your testing, you can cancel the validation mode.

- To replace edition 1.0 with edition 2.0:
 1. Stop the validation target, for example, dynamic_cluster_1-Validation.
 2. Delete the routing rules that are specific to edition 2.0 to route all requests for the application to a single edition.
 3. Save your changes and synchronize the nodes.
 4. Perform rollout to the new edition. Click **Applications > Edition control center > application_name**. Select edition 2.0 and click **Rollout**. During the rollout, edition 2.0 is retargeted to its original deployment target, for example, dynamic_cluster_1. The edition state transitions from validation to active.
- If edition 2.0 has errors, you can cancel validation mode and move edition 2.0 back to its original inactive state. As a result, the duplicate dynamic cluster that was created for validation is removed. See [Canceling an application validation](#) for more information.

[Canceling an application validation](#)

You can cancel validation of an edition and return the application edition to inactive state.

Previous topic: [Activating concurrent application editions](#)

Parent topic: [Deploying and managing application editions without loss of service](#)

Related concepts:

[Application edition manager concepts](#)

Related tasks:

[Installing an application edition](#)
[Performing a rollout on an edition](#)
[Performing a rollback on an edition](#)
[Creating dynamic clusters](#)
[Troubleshooting application edition manager](#)

Related reference:

[Routing and service policies](#)
[Administrative roles and privileges](#)
[Application edition management administrative tasks](#)

Related information:

[Application edition manager custom properties](#)

Canceling an application validation

You can cancel validation of an edition and return the application edition to inactive state.

Before you begin

You must have an application installed and in the validate state.

About this task

By canceling an application in validation mode, the application returns to the inactive state. The deployment target for the edition is reset to its original deployment target. When you cancel an application in validation mode, the validation target is deleted.

Procedure

- Complete the following steps to cancel an application in validation mode:
 1. In the administrative console, click **Applications > Edition control center > *application_name***.
 2. Select the application that is in the validate state and click **Cancel Validation**.
 3. View the cancel validation status page to view each step of canceling the validation. During the validation cancellation, the following actions occur:
 - The application is set to the inactive state.
 - The validation cleanup process is performed
 - The validation target is deleted. If the validation target is a dynamic cluster, you can prevent it from being deleted by setting the saveClonedCluster custom property on dynamic cluster. See [Application edition manager custom properties](#) for more information.

The edition control center shows that edition is now in the inactive state, and the manage editions page shows that the deployment target for the edition is restored to its original deployment target. The dynamic cluster page shows that the validation dynamic cluster no longer exists, and the servers page shows the cloned servers no longer exist.

Remember: The validation deployment target is deleted unless other applications are still deployed on the validation deployment target.

4. Verify that the cancelation of validation occurred correctly. Click **Applications > Enterprise applications** or **Applications > All applications**. Edit your edition 2.0 application.
 - **For PHP and WebSphere® Application Server Community Edition applications:**

Verify that the context root, deployment targets, and so on are restored to the original cluster.
 - **For Java™ Platform, Enterprise Edition (Java EE) applications:**

Select Manage modules. Verify that edition 2.0 is mapped to the original cluster. From the **Map Enterprise JavaBeans (EJB) references to beans** detail view, verify that the Java Naming and Directory Interface name is adjusted for the original target name.
 5. Delete the routing rules that are set up to direct requests to the edition in the validation state. Click **Applications > All applications > *application_name* > Routing Policies**. Save and synchronize the nodes.
- Complete the following steps to remove an application in validation mode when using a version of WebSphere Virtual Enterprise that is earlier than Version 6.1.0.4:
 1. Stop the application edition that is in validation mode. In the administrative console, click **Applications > All applications > *application_name* > Stop > Submit Action**.
 2. Delete the routing rules that are specific to the application edition. Click **Applications > All applications > *application_name* > Routing Policies**. Save your changes and synchronize the nodes.
 3. Uninstall the application from the validation target. Select the application edition that is in validation mode, and click **Remove**.
 - Stop the validation target if the target is an application server, or stop the members of the validation target if the target is a cluster.
 - Delete the target.

Results

The application edition is inactive and is restored back to its original deployment target.

Parent topic: [Validating an edition](#)

Related tasks:

[Validating an edition](#)

Related reference:

[Application edition management administrative tasks](#)

Related information:

[Application edition manager custom properties](#)

Performing a rollout on an edition

When you perform a rollout on an edition, you replace an active edition with a new edition. The new edition might be a simple modification to the application, or contain a more substantial change. As long as the new edition is backward compatible, then you can perform a rollout to replace the active edition without impacting existing clients. To perform a rollout on a new edition, you must first install the application edition with the new edition information.

Before you begin

You must have an application edition that is installed and started, and have *configurator* or *administrator* privileges to perform a rollout.

Note: Performing a rollout fails when two user IDs on two administrative consoles attempt to complete the process in parallel.

Note: Tune the SOAP connector properties to set the request timeout value for the deployment manager to be greater than the total time required to perform a rollout on your system, and restart the deployment manager. Not setting the property can cause the rollout process to fail when the requestTimeout value expires. The formula to estimate the value to set is $\text{number-of-groups-to-rollout} * (\text{drainage-interval} + \text{internal-quiesce-timeouts-approximately-5minutes} + \text{application-or-server-restart-times-approximately-10minutes})$. Alternatively, you can set the value to zero to disable the timeout.

- If you are performing the rollout using the wsadmin tool, adjust the request timeout value by setting the `com.ibm.SOAP.requestTimeout` property in the `soap.client.props` in the deployment manager profile. The default value is 180 seconds and should be increased adequately.
- If you are performing the rollout using the administrative console, adjust the request timeout value by clicking **System administration > Deployment manager > Administration Services > JMX connectors > SOAPConnector > Custom Properties > requestTimeout**. The default value is 600 seconds and should be increased adequately.

For more information, see [Java™ Management Extensions connector properties](#).

If you are performing a rollout within the administrative console, set the session expiration for the administrative console to a value greater than the amount of time required for the entire rollout process to end. Multiply the request timeout value by the number of groups processed during the rollout. For more information, see [Changing the console session expiration](#).

Note: You must define a multi-cluster routing policy for each new edition you install before you can perform a rollout. Use the administrative tasks to add multi-cluster routing policies for your new editions. See [Rules for ODR routing policy administrative tasks](#) for more information.

About this task

You can also use the application edition manager if you are using the Compute Grid component and want to perform a rollout to Compute Grid applications. These are Java Enterprise Edition 5 (Java EE 5) applications that conform to one of the grid programming models.

Procedure

1. Install the new edition. Use the same steps that are described in [Installing an application edition](#) but specify your new edition information. For example, type 2.0 in the **Application edition** field and Second edition in the **Application description** field. Select the same deployment targets that are used for the current edition.
2. Save and synchronize your nodes.
3. Specify the rollout settings. Click **Applications > Edition control center > application_name**. Select your new edition, for example, 2.0, and click **Roll out**.

Specify the following settings for enterprise or other middleware applications:

- a. Select **Atomic** or **Grouped** rollout type.

Use **group rollout** to replace editions on members of the target cluster in a group of one. Group rollout is the most typical choice, and is useful when the cluster contains four or more members. Alternatively, you can perform group rollout with a specified group size through scripting. For more information, read about [Application edition management administrative tasks](#). When the new edition becomes available during group rollout, all requests are directed to the new edition.

Use **atomic rollout** to replace one edition with another on half of the cluster at a time. This rollout type serves all user requests with a consistent edition of the application. Because all user requests are served a consistent edition, your cluster runs at half capacity. If your cluster has four or more members, consider dividing up the cluster into smaller groups by performing a group rollout. Atomic mode is also used with a single server deployment target. In a single server deployment target, the actions that are carried out against the second half of the cluster are omitted. If you stop your deployment targets before you start atomic rollout, the deployment targets are started when the new edition replaces the active edition regardless of the reset strategy you choose. This procedure provides better availability to the requests that are serviced during the rollout period.

Note: Before you perform an atomic rollout, determine the load capability of the target server cluster. Performing an atomic rollout activates the new edition on half of the cluster first, and then activates the edition on the remaining half of the cluster. While the first half of the cluster is taken offline and updated, application requests are routed to the second half of the cluster. Verify that half the cluster can handle the entire load during the rollout period.

- b. Select the reset strategy. The reset strategy instructs the application edition manager how each deployment target loads the new edition into the server runtime.

Use a *Soft* reset strategy to reset the application by stopping or restarting the application in each server of the cluster as the next edition replaces the old edition in that server. Soft reset is the most typical choice and the most optimal performing application reset because it results in loading the new edition by recycling the application in the running application server. The server stays up during this process. With soft reset, native libraries are not unloaded from memory. Soft reset is generally safe for applications that use no native libraries. When soft reset is used in a production environment, monitor the application server process to ensure that sufficient virtual memory exists.

A *Hard* reset strategy recycles the each entire application server of the cluster as the next edition replaces the former edition in the server, refreshing both process memory and any native libraries used by the application. This strategy prevents virtual storage exhaustion and allows new versions of native libraries to be loaded. Select hard reset as your reset strategy when you perform a rollout on an edition that depends on new versions of native libraries or other dependencies that are refreshed only by recycling the entire application server, or if you have large applications that consume a lot of memory for just-in-time compilation (JIT).

- c. Set the drainage interval in seconds. The drainage interval gives the HTTP sessions time to complete before the application or server is reset. The drainage interval specifies the amount of time that the application edition manager waits before the reset strategy starts.

Affinities, such as transaction, activity, and compensation-scope, and activities unknown to WebSphere® Virtual Enterprise, lengthen the effective drainage interval because the server does not stop until these units of work complete. Applications with activities unknown to WebSphere Virtual Enterprise can use the AppEditionManager MBean quiesce initiated notification as a trigger to begin shutdown processing and exploit the drainage interval as a time period during which to complete the shutdown. This process is unnecessary for persistent sessions, for example, those backed in database or replicated through VMware Distributed Resource Scheduler (DRS), but is important for transient (in memory) sessions.

The goal of the drainage interval is to allow requests with affinities and inflight requests to complete. To prevent the loss of transient sessions, set the drainage interval to exceed the application session timeout interval. After the rollout starts, as each server updates, the server is marked as ineligible to begin any new sessions. Set this value to 0 to not wait for sessions to complete.

The application edition quiesce manager might not wait the full length of the drainage interval. Performance Monitoring Infrastructure (PMI) statistics are available for the quiesce manager to determine if all active requests on a server have been quiesced. If all requests are quiesced before the drainage interval, the application edition quiesce manager does not need to wait for the full drainage interval. To force a soft reset to wait the entire drainage interval, you can set the `appedition.rollout.softreset.fulldrainageinterval` system property to true on the deployment manager.

The drainage interval allows existing sessions to complete. However, at the end of the drainage interval, a period of time exists during which inflight requests can still arrive. In such cases, the on demand router (ODR) provides a timeout value of 60 seconds within which to complete the quiesce operation. If the requests end within 60 seconds or the 60 seconds expire, the application, or the server in the case of a hard reset strategy, is stopped. Next, if inflight requests have still not ended, WebSphere Application Server Network Deployment provides a quiesce time of 60 seconds before stopping the application or the server instance. For hard reset strategies, WebSphere Application Server Network Deployment provides a quiesce time of 180 seconds before stopping the server instance. You can use the `com.ibm.ws.webcontainer.ServletDestroyWaitTime` custom property to define the amount of time that the Web container waits for the requests to complete. For more information, see [Web container custom properties](#).

You can use the `com.ibm.ejs.sm.server.quiesceTimeout` custom property to define the amount of the time that the server instance waits for the requests to complete before initiating shutdown. For more information, see [Java virtual machine custom properties](#). You must set both the `com.ibm.ws.webcontainer.ServletDestroyWaitTime` custom property and the `com.ibm.ejs.sm.server.quiesceTimeout` custom property on each of the server instances on which the application editions are deployed.

Specify the following settings for Session Initiation Protocol (SIP) applications:

- a. Choose a quiesce strategy. The quiesce strategy specifies how old servers or cluster members that host the current edition are removed. This setting does not affect the new edition that is being rolled out.
 - **Quiesce server or cluster members after all active sessions or dialogs are completed:** Removes the server or cluster member when all of the active sessions and dialogs for the server or cluster member complete.

- **Quiesce servers or cluster member after the specified interval:** Removes the server or cluster member after a specified time period. Specify an amount of time, in seconds, minutes, or hours.

Attention: Performing a rollout is not supported for SIP applications that are deployed on a dynamic cluster that has been converted from a static cluster.

4. Start the rollout. Click **OK**. This action launches an interruption-free replacement of the previous edition with your new edition.

Results

For an edition that is not in validation mode, the new edition replaces the current edition after the rollout completes. An edition that is in validation rolls out on the original deployment target and the cloned environment is deleted. The routing rules are updated to begin routing to your new edition.

During the application rollout process, errors that occur in the first group of targets are handled differently than errors that occur in later groups. In atomic rollouts, the first group is the first half of the targets where the new edition is activated. In group rollouts, the first group refers to the first group of targets where the new edition is activated. If an error occurs during the rollout on the first group of targets (for example an application or a server fails to start), the rollout process fails. The current application edition remains in active state. If an error occurs after the rollout on the first group of targets, the rollout process completes successfully. The new application edition is now in active state. The old application edition moves into inactive state.

What to do next

To validate the results, click **Applications > Edition control center > *application_name***. Your new edition is the active edition on the deployment target. The new edition automatically starts, because it replaces a running edition.

When you perform a rollout on an edition in validation mode, the binding names are changed back to the original values. For example, `/clusters/cluster1-validation/jdbc/CustomerData` is changed back to `/clusters/cluster1/jdbc/CustomerData`.

Algorithm for performing a rollout

The algorithm for performing a rollout to a new edition has operational implications on your environment. The installation and distribution of an application edition is separate from its activation.

Parent topic: [Deploying and managing application editions without loss of service](#)

Related concepts:

[Application edition manager concepts](#)

Related tasks:

[Installing an application edition](#)

[Performing a rollback on an edition](#)

[Activating concurrent application editions](#)

[Validating an edition](#)

Related reference:

[Routing and service policies](#)

[Administrative roles and privileges](#)

[Application edition management administrative tasks](#)

Related information:

[Application edition manager custom properties](#)

Algorithm for performing a rollout

The algorithm for performing a rollout to a new edition has operational implications on your environment. The installation and distribution of an application edition is separate from its activation.

Two basic patterns exist for interruption-free replacement: group rollout or atomic rollout. The steps that occur to perform a rollout to a new edition vary depending on which of these options you choose.

The dynamic clusters are put in manual mode during the rollout. If your load becomes heavy during the rollout, application placement does not occur. Plan your rollouts so that you avoid peak periods or heavy loads. When the rollout completes, the dynamic cluster is put back into its original mode.

Note: Do not perform a rollout during periods of heavy traffic.

Group rollout

When you choose to perform a group rollout, the rollout occurs across the clusters in groups of servers. The following steps occur for each server:

1. Quiesce work to the server.
2. Stop the application or stop the server.
3. Update the server configuration.
4. Restart the application or the server.
5. The server is ready with the new edition.

Atomic rollout

Before you perform an atomic rollout, determine the load capability of the target server cluster. Performing an atomic rollout activates the new edition on half of the cluster first, and then activates the edition on the remaining half of the cluster. While the first half of the cluster is taken offline and updated, application requests are routed to the second half of the cluster. Verify that half the cluster can handle the entire load during the rollout period.

When you choose to perform an atomic rollout, the following steps occur:

1. Quiesce work to half of the servers.
2. Stop the applications or servers in the first half of the servers.
3. Update the configurations.
4. Start the applications or servers in the first half of the servers.
5. Quiesce work to the second half of the servers.
6. Start routing requests to the new edition, which is running on the first half of servers.
7. On the second half of the servers, stop the applications or servers, update the configurations, and start the applications or servers.
8. Rollout is complete.

Default rollout settings

The following options are preset for the rollout actions in the administrative console:

- Group rollout:
 - rollout strategy = group, group size = 1
 - reset strategy = application
 - drainage interval = 30 seconds
- Atomic rollout:
 - rollout strategy = atomic
 - reset strategy = application
 - drainage interval = 30 seconds

Scripting interface rollout options

The group and atomic rollout options in the administrative console offer a preset selection of rollout options. Greater flexibility over these options is possible through the scripting interface. Read about application edition management administrative tasks for more information. The following scripting options exist:

- **Rollout strategy:** Specifies the rollout method, either groups of nodes updated serially or the divide-and-swap atomic strategy.
 - **Group:** Specifies that the target cluster is divided into groups for rollout. Group rollout is most effective when your cluster is large. You can specify the group size with a sub-option. The group size gives the number of nodes to process at a time. The default is 1.
 - **Atomic:** Specifies that only one edition of the application can serve requests during the rollout period. This results in half of the application server cluster being taken offline and updated, and then the other. Application requests that arrive while both halves of the cluster are offline are queued by the on demand router (ODR).
- **Reset strategy:** Specifies whether to recycle, for example, stop and restart, the application or the entire application server.
 - **Application:** Activates the new edition in each application server by recycling the application. The application server continues to run.
 - **Server:** Activates the new edition in each application server by recycling the server itself. This

option is necessary if you need to refresh connectors, native libraries, or reset the Java™ virtual machine.

- **Drainage interval:** Specifies the amount of time to wait for processing requests to complete before the application or application server is stopped. The default is 30 seconds.

Parent topic: [Performing a rollout on an edition](#)

Related concepts:
[Application edition manager](#)

Related reference:
[Application edition management administrative tasks](#)

Performing a rollback on an edition

If you perform a rollout on a new edition and an error exists in your application, you can undo your change and easily revert back to the previous edition. To rollback to the edition, perform the rollout steps, but select a former edition to replace the current edition.

Before you begin

If you are a user with either a monitor or an operator role, you can only view the application edition manager information. If you have the role of configurator or administrator, then you have all the configuration privileges for the application edition manager.

The edition to which you want to roll back must be in the system. If your edition is not in the system, install the edition and specify edition information that indicates that the edition is earlier than your current edition. For example, if the current edition is 2.0, specify the edition that you want to rollback to as 1.0. Select the same deployment targets for both editions. Clone the work classes from the current edition to the previous edition.

About this task

For application resiliency, rollback an edition when there are errors with your currently active application edition. You can revert to the previous edition of the application. Rollback the current edition by rolling out a previous edition.

Procedure

1. Expand **Applications > Edition control center** in the administrative console. Select the former edition, for example 1.0.
2. Click **Rollout**. This action launches an interruption-free replacement of the current edition with the selected previous edition.

Results

The former application edition is active.

What to do next

Troubleshoot errors with the newer application edition, and perform an edition rollout to activate the newer edition.

Parent topic: [Deploying and managing application editions without loss of service](#)

Related concepts:

[Application edition manager concepts](#)

Related tasks:

[Installing an application edition](#)

[Performing a rollout on an edition](#)

[Activating concurrent application editions](#)

[Validating an edition](#)

Related reference:

[Administrative roles and privileges](#)

[Application edition management administrative tasks](#)

Deleting an installed edition

After multiple successive editions of an application are deployed and rolled out, which is not unusual in the life span of many production applications, you might have a number of editions stored in your repository that are no longer needed. You can delete an active edition without performing a rollout to a new edition.

Before you begin

- You must have an installed, active edition.
- You must have a *configurator* or *administrator* administrative role.

About this task

An edition can exist in one of three states. Therefore, the method for deleting or uninstalling an edition varies slightly based on its current state.

- **Inactive:** The edition is neither assigned for use in any deployment target, nor the default. You can remove an inactive edition at any time.
- **Active:** The edition is currently assigned for use in one or more deployment targets. Use caution when you delete an active edition, especially if the edition is started. Stop the edition first, remove any routing rules that reference the edition, deactivate it, and you can then remove or uninstall the edition. After you delete an active edition, you must update the routing rules, and click **Apply** and **Save** regardless of what item is displayed in the list.
- **Validate:** The edition is currently assigned for use in the cloned deployment targets. See [Canceling an application validation](#) to remove an application in validation mode.

Procedure

1. Remove any routing rules that reference the edition you are uninstalling. Click **Applications** > **Enterprise applications** > *application_name* or **Applications** > **All applications** > *application_name*. Click the **Routing policies** tab and select the appropriate routing rules.
2. Click **Applications** > **Edition control center**. Select the target application.
3. Stop the application edition.
4. Deactivate the active edition. Select the edition, and click **Deactivate**.

Instead of deactivating an active edition, you can replace it with a new edition. You can perform a rollout to a new edition, which makes the old edition inactive. Use rollout to ensure an interruption-free deactivation of the edition you want to delete.

5. Uninstall the application. Click **Applications** > **Enterprise applications** or **Applications** > **All Applications**. For an enterprise application, select the edition that you want to uninstall, click **Uninstall**. For a middleware application, select the edition, and click **Remove**. You are not required to restart the servers.
6. Save and synchronize your changes.

Results

You removed the edition.

Parent topic: [Deploying and managing application editions without loss of service](#)

Related concepts:

[Edition compatibility](#)

Related tasks:

[Performing a rollout on an edition](#)

[Validating an edition](#)

Related reference:

[Administrative roles and privileges](#)

[Application edition management administrative tasks](#)

Troubleshooting application edition manager

When you encounter a problem with the application edition manager, verify that the servers are running or refer to the log files to further investigate the errors..

Procedure

- While performing a rollout, verify that your servers are running. Node agents, the on demand router (ODR), and servers must be running to perform the rollout action.
- Check the log files of the deployment manager, the node agents, and the servers that are involved in the current operation.
 - **Distributed platforms** On distributed operating systems, the log file is the SystemOut.log file.
 - **z/OS** On z/OS® systems, the log file is the servant job log.

You can use the log files when you encounter errors. If you report the problem, it is helpful to collect and save these files. The preferred mechanism is to use the collector utility on each affected node, including the deployment manager, and be prepared to send the output. The collector utility is in the install_root/bin directory on each product installation directory.

- If you are experiencing frequent deployment manager timeouts on a z/OS system, configure your timeout values to either increase or disable timeouts.

[Configuring timeout values for application edition manager on z/OS systems](#)

If you are using application edition manager on a z/OS system, and are experiencing deployment manager timeouts, you should either increase the deployment manager timeout values or set the timeout values to zero.

[Application edition manager frequently asked questions](#)

When you start to use application edition manager, you might come across some questions about the application edition manager functionality.

Parent topic: [Deploying and managing application editions without loss of service](#)

Related concepts:

[Application edition manager](#)

Configuring timeout values for application edition manager on z/OS systems

If you are using application edition manager on a z/OS® system, and are experiencing deployment manager timeouts, you should either increase the deployment manager timeout values or set the timeout values to zero.

Before you begin

Configure your application editions.

About this task

To avoid deployment manager timeouts, you might want to either increase the system timeouts based on the system responses and the site environment, or you might disable timeouts. By disabling the deployment manager timeouts, you can prevent timeouts from occurring and keep the deployment manager servant region from stopping and restarting.

Procedure

1. Change the workload manager (WLM) timeout value on the object request broker (ORB) to zero.
 - a. In the administrative console, click **System administration > Deployment manager > ORB Service > z/OS additional settings**.
 - b. Change the **Workload manager timeout** value to 0. The default value is 300 seconds.
This setting affects Internet Inter-ORB Protocol (IIOP) work that is queued to the servants.
2. Modify the `ConnectionResponseTimeout` custom properties. These properties specify the amount of time that the Java™ Platform, Enterprise Edition (Java EE) server waits for an application component to respond to a Hypertext Transfer Protocol (HTTP) request.
3. Update the timeout values in the Web container transport chains.
 - a. In the administrative console, click **System administration > Deployment manager > Web container transport chains**.
 - b. Modify the `WCInboundAdmin` variable. Modify the following transport channels:
 - Click **TCP inbound channel (TCP_1)** and set the **Inactivity timeout** to 0 seconds.
 - Click **HTTP inbound channel (HTTP_1)** and set the **Read timeout**, **Write timeout**, and **Persistent timeout** values to 0 seconds.The **Web container inbound channel (WCC_1)** channel does not require any changes.
 - c. Modify the `WCInboundAdminSecure` variable. Modify the following transport channels:
 - Click **TCP inbound channel (TCP_3)** and set the **Inactivity timeout** to 0 seconds.
 - Click **HTTP inbound channel (HTTP_2)** and set the **Read timeout**, **Write timeout**, and **Persistent timeout** values to 0 seconds.The **SSL inbound channel (SSL_1)** and **Web container inbound channel (WCC_2)** channels do not require changes.

What to do next

Configure other application edition manager features.

Parent topic: [Troubleshooting application edition manager](#)

Related concepts:

[Application edition manager concepts](#)

Application edition manager frequently asked questions

When you start to use application edition manager, you might come across some questions about the application edition manager functionality.

Can the application edition manager ensure interruption-free application upgrade for workloads that are driven by Internet InterORB Protocol (IIOP) or Java™ Message Service (JMS) protocols?

No. The application edition manager can ensure interruption-free application upgrades only for workloads managed by the on demand router (ODR). The ODR supports HTTP and HTTPS only.

Can the application edition manager ensure interruption-free application upgrades for workloads that include calls between Java Platform, Enterprise Edition (Java EE) Version 5 applications?

Yes, the application edition manager can ensure an interruption-free update if each interapplication request uses HTTP or HTTPS protocol through an on demand router.

Can the application edition manager ensure interruption-free application upgrades for workloads that are hosted on static clusters?

Yes, if an on demand router is managing the work sent to that static cluster.

Can the application edition manager support the deployment of one edition of an application to one cluster and the deployment of another edition to a different cluster?

Yes. The application edition manager supports different editions simultaneously deployed on different targets. You must establish routing policies for routing traffic to different editions.

Does the application edition manager support the activation of different editions on different members of the same cluster?

No.

Does the application edition manager restart the application or the entire server to activate a new edition?

No, the application edition manager does not restart the application or server. You must perform the following steps:

1. Stop the active edition.
2. Deactivate the active edition.
3. Activate the new edition.
4. Start the new edition.

Can the application edition manager handle the rollout of editions with incompatible changes?

Yes, the application edition manager can handle this situation if the on demand router is configured to perform edition-aware routing. See [Edition compatibility](#).

Can the application edition manager handle the rollout of editions that include database schema changes?

Yes, you can perform rollout of editions that include database schema changes if the database changes are compatible with earlier versions.

Parent topic: [Troubleshooting application edition manager](#)

Parent topic: [Troubleshooting application edition manager](#)

Related concepts:

[Application edition manager](#)

Related tasks:

[Deploying and managing application editions without loss of service](#)

Managing the WebSphere Virtual Enterprise environment

You can configure additional functions to enhance the autonomic functions of the application infrastructure virtualization environment. These functions include maintenance mode, health management, centralized installation manager, high availability deployment manager, repository checkpoints, and autonomic managers.

Before you begin

Configure the on demand router (ODR), servers, dynamic clusters, applications, and service policies.

About this task

After you set up your basic hosting environment with application server virtualization, you can tune the behavior of the managers that make decisions in your environment. You can also configure several features to help with administration.

[Setting maintenance mode](#)

Set maintenance mode before you perform diagnostic tests, maintenance, or tuning on a node or server. Maintenance mode can prevent the disruption of client requests by routing client traffic that is targeted for a server or node that is in maintenance mode to another server or node.

[Routing to servers that are in maintenance mode](#)

You can build complex rule conditions from subexpressions to modify routing policy Session Initiation Protocol (SIP) rules, and route certain messages to servers that are in maintenance mode while routing other traffic to servers that are not in maintenance mode from within the same dynamic cluster.

[Manually replacing SSL certificates for the middleware agent](#)

If your existing middleware agent Secure Sockets Layer (SSL) certificates expire, manually replace the certificates from within the administrative console.

[Fine-grained security](#)

The panels in the WebSphere Virtual Enterprise administrative console are filtered to appropriately govern access to the product resources based upon the administrator's roles.

[Generating Simple Network Management Protocol \(SNMP\) traps](#)

You can generate Simple Network Management Protocol (SNMP) Version 1 traps for health events and dynamic cluster start and stop application server events.

[Configuring the autonomic managers](#)

Use these instructions to configure the behavior of the autonomic managers in WebSphere Virtual Enterprise.

[Using the centralized installation manager to manage your environment](#)

Use the centralized installation manager to simplify the tasks of deploying product components to your WebSphere Application Server Network Deployment cell or other server environments.

[Routing requests to nodes that are not running WebSphere Virtual Enterprise](#)

Use this task for configuring multiple tiers.

[Configuring a high availability deployment manager environment](#)

The high availability (HA) deployment manager function can be configured to eliminate single point of failure for administrative functions in a WebSphere Application Server Network Deployment cell.

[Configuring a checkpoint](#)

Repository checkpoints represent saved images of the repository before configuration changes are made and can be used to restore the configuration repository back to a prior state. Checkpoints are either full or delta images. A full checkpoint is created manually by the administrator and is a copy of the entire configuration repository. This includes applications and connectors. Delta checkpoints are optional and are not enabled by default. A delta checkpoint is created automatically when configuration changes are made. The delta checkpoint is formed by making a copy of the configuration documents affected by the configuration change before changes are actually applied.

[Using centralized logging to diagnose problems](#)

You can use centralized logging to easily enable mustGather traces, perform a per-request trace analysis, and follow the flow of a request through both the ODR and application server tiers.

Related tasks:

[Installing and customizing the product](#)

[Migrating WebSphere Virtual Enterprise](#)

[Preparing the hosting environment for dynamic operations](#)

[Deploying and managing application editions without loss of service](#)

[Monitoring operations](#)

[Configuring WebSphere Virtual Enterprise to work with other IBM products](#)

Related reference:

[Reference](#)

Related information:

[IBM WebSphere Virtual Enterprise Version 7.0](#)

[Product overview](#)

[Site map](#)

Setting maintenance mode

Set maintenance mode before you perform diagnostic tests, maintenance, or tuning on a node or server. Maintenance mode can prevent the disruption of client requests by routing client traffic that is targeted for a server or node that is in maintenance mode to another server or node.

Before you begin

- You can set maintenance mode on a server or a node that is represented in the administrative console.
- To view the nodes that are configured, click **System administration > Middleware nodes**. To view all of the servers that are configured, click **Servers > All servers**.
- You can use maintenance mode when you route requests with on demand routers (ODR) or proxy servers. If you are using a web server to route requests, maintenance mode is not recognized.
- ODR maintenance mode is supported when the ODR is fronted by a web server plug-in. The plugin-cfg.xml file generator generates the plug-in appropriately to enforce the various modes of node and server maintenance mode. You can use node and server maintenance mode to apply service to the ODR tier in the same way that you use maintenance mode to apply service to the application server tier.

About this task

When a node or server is in *running* state, the dynamic operations environment considers that node or server to be available for servicing application requests. When problems occur, you must perform diagnostics, maintenance, or tuning on the node or server. Use *maintenance mode* to stop routing traffic to the node or server while you determine the problem.

When a server is in maintenance mode, the application placement controller cannot control that server. The server does not count toward the minimum or maximum running instances setting for the dynamic cluster. For vertical stacking, the configured maximum minus the number of servers in maintenance mode on the node is used to determine if any servers need to be created on the node. By using this method, the dynamic operations environment does not create or destroy server instances to support the configured maximum number of instances as the dynamic cluster instances move in and out of maintenance mode.

Version 7.0 supports setting node and server maintenance mode for ODRs. You can use maintenance mode to perform node-level maintenance mode for the ODR nodes just as you can perform node-level maintenance mode for application server nodes. Similarly, you can use server maintenance mode for your ODRs just as you can for application servers.

Two routing policies, the permitMM and permitstickyMM routing policies, are work class match actions that you can use to only route HTTP traffic to servers in maintenance mode. See [Routing and service policies](#) for more information.

Procedure

- Place nodes into or out of maintenance mode.
 1. Select the nodes. In the administrative console, click **System administration > Middleware nodes**. Select the nodes for which you want to change the maintenance mode.
 2. Select the maintenance mode.
 - **Maintenance mode**: The on demand router (ODR) will continue to send requests with affinity to an application server in this mode, but will typically not send requests without affinity (unless the request matches a rule which explicitly permits it to be routed to an application server in maintenance mode).
 - **Maintenance immediate stop**: This mode is the same as the break affinity mode, except that the application server is also immediately stopped. When the application server is restarted, it is still in break affinity mode.
 - **Normal**: The ODR sends requests with and without affinity to an application server in this mode.
 3. Click **Set mode**.
- Place servers into or out of maintenance mode. In the administrative console, click **Servers > All servers**. Select the servers for which you want to change the maintenance mode.
 1. Select the servers.
 2. Select the maintenance mode.
 - **Maintenance mode**: The ODR will continue to send requests with affinity to an application server in this mode, but will typically not send requests without affinity (unless the request matches a rule which explicitly permits it to be routed to an application server in maintenance mode).
 - **Maintenance mode - break affinity**: The ODR will typically not send any requests to an application server in this mode (unless the request matches a rule which explicitly permits it to be routed to an application server in maintenance mode).
 - **Maintenance mode - immediate stop**: This mode is the same as the break affinity mode, except that the application server is also immediately stopped. When the application server is restarted, it is still in break affinity mode.
 - **Normal**: The ODR sends requests with and without affinity to an application server in this mode.
 3. Click **Set mode**.

Results

HTTP and Session Initiation Protocol (SIP) traffic is no longer routed to the nodes or servers that you have in maintenance mode. Enterprise JavaBeans (EJB) affinity is not broken. Requests with EJB affinity and Java™ Message Service (JMS) traffic are still routed to the server that is in maintenance mode.

What to do next

The health controller also uses server maintenance mode as an action that is taken when the health policy is breached. For example, if you are using an excessive response time condition, and the response time exceeds the specified maximum for a server, then the server can move to maintenance mode.

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Related tasks:

[Creating health policies](#)

[Creating ODRs](#)

[Routing to servers that are in maintenance mode](#)

Related reference:

[Maintenance mode administrative tasks](#)

[Routing and service policies](#)

Routing to servers that are in maintenance mode

You can build complex rule conditions from subexpressions to modify routing policy Session Initiation Protocol (SIP) rules, and route certain messages to servers that are in maintenance mode while routing other traffic to servers that are not in maintenance mode from within the same dynamic cluster.

Before you begin

To modify rules through the rule builder, you must have administrator privileges.

z/OS **Restriction:** WebSphere® Virtual Enterprise does not support SIP features on the z/OS® operating system.

About this task

Typically, only servers that are not in maintenance mode receive and service application requests. You can also route to servers that are in maintenance mode, so you can route to a subset of servers to verify configuration changes or troubleshoot issues without interfering with availability. You can direct SIP traffic to specific clusters with routing rules. The expressions for SIP routing rules are enhanced to support an expression that provides more functionality and flexibility in defining the target cluster. For example, the expression `cluster='TestCell/TestClusterA'` and `serverMaintenanceMode='affinity'` direct the on demand router (ODR) to route calls to members of TestClusterA that are in maintenance mode. You can also use this function in conjunction with the prioritized routing rules.

Procedure

1. From the administrative console, click **Servers > On demand routers > odr_name > SIP on demand router settings > Routing policy SIP rules > Specify by > Expression > Subexpression builder**.
2. Select **and** as the logical operator.
3. Select **Server Maintenance Mode** as the operand.
4. Select **Equals (=)** as the operator.
5. Select **Affinity** as the value. Click **Generate subexpression**.
6. Click **Append** to add the subexpression to your rule.

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Related concepts:

[Rule-based request classification](#)

Related tasks:

[Setting maintenance mode](#)

Related reference:

[Routing and service policies](#)

Related information:

[Rules for ODR routing policy administrative tasks](#)

[Rules for ODR service policy administrative tasks](#)

Manually replacing SSL certificates for the middleware agent

If your existing middleware agent Secure Sockets Layer (SSL) certificates expire, manually replace the certificates from within the administrative console.

Before you begin

You must be able to access the deployment manager for each cell with which you want to communicate.

About this task

Renew expired SSL certificates so the middleware agent can continue to securely communicate with the deployment manager and nodes.

Procedure

1. Run the **backupConfig** command on the deployment manager.

- o **Linux** **AIX** **HP-UX** **Solaris**

```
./backupConfig.sh backup_file [options]
```

- o **Windows**

```
backupConfig.bat backup_file [options]
```

2. Stop the middleware agent.
 - o In the administrative console, click **System administration** > **Middleware nodes**. Select the specific middleware agent, and select **Stop agent** from the **Select operational action** menu. Click **Run**.
 - o From the command line, run the **stopAgent.sh | .bat** command from the *agent_install_root/bin* directory.
3. Create a new middleware agent certificate.
 - a. Click **Security** > **SSL certificate and key management** > **Manage endpoint security configurations** > *node_name* > **Manage certificates**.
 - b. Click **Create a self-signed certificate**.
 - c. Enter the following attributes for the new certificate, and click **OK**:
 - **Alias:** *node_name_default*
 - **Common name:** *host_name*
 - **Validity of period:** *number_of_days*
 - **Organization:** *company_name*Click **Save** to save your changes.
4. Replace the existing certificate with the new certificate.
 - a. Click **Security** > **SSL certificate and key management** > **Manage endpoint security configurations** > *node_name* > **Manage certificates**. Select the existing certificate, and click **Replace**.
 - b. Select and accept the new certificate.

Note: Do not select **Delete old certificate after replacement** or **Delete old signers**.
 - c. Select the existing certificate, and click **Delete** > **OK**. Click **Save** to save your changes.
5. Add the signer certificate for the node to the CellDefaultTrustStore key store.
 - a. Click **Security** > **SSL certificate and key management** > **Manage endpoint security configurations** > *node_name*. Select **Key stores and certificates**.
 - b. Select **NodeDefaultKeyStore** and **CellDefaultTrustStore**, and click **Exchange signers**.
 - c. Select the certificate you created in step 3, and click **Add**. Click **OK**, and click **Save** to save your changes.
6. Delete the existing certificates, and extract the new certificates.
 - a. Click **Security** > **SSL certificate and key management** > **Key stores and certificates** > **CellDefaultTrustStore** > **Signer certificates**.
 - b. Select the existing certificates, and click **Delete**.

Tip: To verify which certificates to delete, compare the fingerprint or expiration dates with the personal certificate in the key stores.
7. Stop the deployment manager.
 - o In the administrative console, click **System administration** > **Deployment manager** > **Stop**.
 - o From the command line, run the **stopManager.sh | .bat** command from the *profile_root/bin* directory of the deployment manager profile.
8. Copy the trust.p12 and key.p12 files from the deployment manager node to the middleware agent node.

The trust.p12 and key.p12 files are in the *profile_root/dmgr/config/cells/cell_name/nodes/middleware_agent_node* directory.

9. From the *install_root/config/cells/cell_name/nodes/node_name* directory on the middleware agent node, save the trust.p12 and key.p12 files to a backup directory. Copy the trust.p12 and key.p12 files that you copied in step 8 to the directory.
10. Start the deployment manager. Run the **startManager.sh|.bat** command.
11. Start the middleware agent.
 - In the administrative console, click **System administration > Middleware nodes**. Select the specific middleware agent, and select **Start agent** from the **Select operational action** menu. Click **Run**.
 - From the command line, run the **startAgent.sh|.bat** command from the *agent_install_root/bin* directory.

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Related tasks:

[Configuring a high availability deployment manager environment](#)
[Enabling communication between cells that have security enabled](#)

Fine-grained security

The panels in the WebSphere® Virtual Enterprise administrative console are filtered to appropriately govern access to the product resources based upon the administrator's roles.

In releases prior to Version 7.0, users granted administrative roles could administer all of the resource instances under the cell. WebSphere Virtual Enterprise is now more fine-grained, meaning that access can be granted to each user per resource instance. Some of the resources include on demand router (ODR) servers and clusters, dynamic clusters, middleware nodes, and middleware servers.

To achieve fine-grained security, resources that require the same privileges are placed in a group called the *administrative authorization group* or *authorization group*. Users can be granted access to the authorization group by assigning to them the required administrative role. You can assign users and groups to the `adminsecuritymanager` role on the cell level through `wsadmin` scripts and the administrative console. Using the `adminsecuritymanager` role, you can assign users and groups to the administrative user roles and administrative group roles.

Administrative security commands

There are several administrative security commands that you can use to create authorization groups, map resources to authorization groups, and to assign users to administrative roles within the authorization groups. Following are some examples using `wsadmin`:

- **Create a new authorization group**

```
$AdminTask createAuthorizationGroup {-authorizationGroupName group_name}
```

- **Delete an authorization group**

```
$AdminTask deleteAuthorizationGroup {-authorizationGroupName group_name}
```

- **Add resources to an authorization group**

```
$AdminTask addResourceToAuthorizationGroup {-authorizationGroupName group_name -resourceName Application=appl}
```

- **Remove resources from an authorization group**

```
$AdminTask removeResourceFromAuthorizationGroup {-authorizationGroupName group_name -resourceName Application=appl}
```

- **Add user IDs to roles in an authorization group**

```
$AdminTask mapUsersToAdminRole {-authorizationGroupName group_name -roleName administrator -userids user1}
```

- **Add group IDs to roles in an authorization group**

```
$AdminTask mapGroupsToAdminRule {-authorizationGroupName group_name -roleName administrator -groupids group1}
```

- **Remove user IDs from roles in an authorization group**

```
$AdminTask removeUsersFromAdminRole {-authorizationGroupName group_name -roleName administrator -userids user1}
```

- **Remove group IDs from roles in an authorization group**

```
$AdminTask removeGroupsFromAdminRole {-authorizationGroupName group_name -roleName administrator -groupids group1}
```

For more information, see [Fine-grained administrative security](#) in the WebSphere Application Server Network Deployment Information Center.

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Generating Simple Network Management Protocol (SNMP) traps

You can generate Simple Network Management Protocol (SNMP) Version 1 traps for health events and dynamic cluster start and stop application server events.

Before you begin

Any standard SNMP management endpoint that is capable of receiving Version 1 traps can receive traps that are sent by WebSphere® Virtual Enterprise. See the documentation associated with your SNMP management endpoint for instructions on how to import the Management Information Base (MIB) file that is provided with WebSphere Virtual Enterprise.

About this task

You can generate SNMP Version 1 traps for the following two types of events:

- Health events: In this case, sending an SNMP trap is one of the actions that can be performed when a health policy is triggered. The SNMP trap contains the name of the health policy that was triggered, when the health policy was triggered, and where the health policy was triggered.
- Dynamic cluster start and stop application server events: When the application placement controller starts or stops an application server, the controller can send an SNMP trap notifying an SNMP management endpoint.

The precise events are defined by a MIB file in the WAS_HOME/etc/wvemib.txt file. You can use this MIB file to import the events into your SNMP management endpoint.

Complete the following procedure to generate SNMP traps.

Procedure

1. Configure the host name and an optional port of the SNMP management endpoint. Set the following custom property:
 - a. In the administrative console, select **System administration > Cell > Custom properties > New**.
 - b. Specify the name of the custom property as `snmp.manager`.
 - c. Specify the value of the custom property as `hostname[:port]`. This is the host name and UDP port to which the SNMP traps are sent. The default port value is 162.

For example, setting the value of the custom property to `mySnmpManagerHost:162` indicates that SNMP Version 1 traps should be sent to UDP port 162 of the host named `mySnmpManagerHost`.

2. Configure the product to send an SNMP trap for health policy violations. Create a new health policy and select the **Generate an SNMP trap** predefined health policy action as one of the actions to perform when the health policy is triggered. In the administrative console, select **Operational policies > Health Policies > New**.
3. Set the following custom property to disable SNMP traps from being sent when the application controller starts or stops a member of a dynamic cluster.
 - a. In the administrative console, select **System administration > Cell > Custom properties > New**.
 - b. Specify the name of the custom property as `snmp.disabledTraps`.
 - c. Specify the value of the custom property as `dynamicClusterStartStopServer`.
4. Configure the SNMP community string. By default, the product sends an SNMP trap with a community string value equal to `public`. Set the following custom property to override the default value.
 - a. In the administrative console, select **System administration > Cell > Custom properties > New**.
 - b. Specify the name of the custom property as `snmp.community`.
 - c. Specify a value for the custom property. The value of the custom property is then used as the community string value.

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Related tasks:

[Configuring health management](#)

[Configuring the autonomic managers](#)

Configuring the autonomic managers

Use these instructions to configure the behavior of the autonomic managers in WebSphere® Virtual Enterprise.

Before you begin

Set up your system by [Preparing the hosting environment for dynamic operations](#).

About this task

Configure autonomic managers after you have prepared your environment to support autonomic management, and if you want to maximize utilization using your defined business goals. These autonomic managers monitor performance metrics, analyze the monitored data, offer a plan for running actions, and can start these actions in response to the flow of work. The following autonomic managers all work in or with the WebSphere Virtual Enterprise environment.

[Configuring the autonomic request flow manager](#)

You can fine tune the autonomic request flow manager (ARFM) by changing the default settings in the administrative console.

[Configuring dynamic application placement](#)

Dynamic application placement is the process by which the features of the dynamic operations of WebSphere Virtual Enterprise work together to start and stop application instances to meet the fluctuating demand of work requests of varying service policy definitions. This set of actions is controlled by the application placement controller.

[Overview of IOP and JMS request flow prioritization](#)

Internet Inter-ORB Protocol (IOP) and Java™ Message Service (JMS) request flow prioritization is achieved in WebSphere Virtual Enterprise with the autonomic managers that control the flow of requests, prioritization of requests, and dynamic workload management. Only IOP requests from a stand-alone Enterprise JavaBeans (EJB) client are handled by WebSphere Virtual Enterprise IOP request flow prioritization. EJB calls from servlets, for example, are not prioritized. This limitation exists because the system cannot prioritize at multiple tiers, such as the Web tier and EJB tier, requests associated with the same overall user request. However, given the asynchronous nature of JMS, no restrictions exist on where the requests originate.

[Optimizing the database tier for performance monitoring](#)

You can configure WebSphere Virtual Enterprise to provide real time information about the database tier, including central processing unit utilization, average response time, and throughput. You can access this information through the runtime topology in order to make adjustments when bottlenecks occur.

[Configuring health management](#)

Health management is the process by which the dynamic operations of WebSphere Virtual Enterprise monitor and manage servers that might fail, to preserve an optimal server environment.

[Tuning the WebSphere Virtual Enterprise cell](#)

Start here for a summary of key performance tuning parameters for your WebSphere Virtual Enterprise configuration.

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Related concepts:

[Dynamic operations](#)

[Overview of request flow prioritization](#)

Configuring the autonomic request flow manager

You can fine tune the autonomic request flow manager (ARFM) by changing the default settings in the administrative console.

Before you begin

To change the settings on the autonomic request flow manager, you must have operator, configurator, or administrator administrative privileges. Operators can only view the information on the configuration tab, but can change the settings on the runtime tab. Configurators can change settings on the configuration tab, but cannot change settings on the runtime tab. Administrators have all privileges.

When security is enabled, some fields are not editable without proper security authorization.

About this task

The autonomic request flow manager contains the following components:

- A compute power controller per target cell, such as a cell to which an ARFM gateway directly sends work. This is an HAManagedItem that runs in any node agent, on demand router (ODR), or deployment manager.
- A gateway per used combination of protocol family, proxy process, and deployment target. A gateway runs in its proxy process. For HTTP and Session Initiation Protocol (SIP), the proxy processes are the on demand routers; for Java™ Message Service (JMS) and Internet Inter-ORB Protocol (IIOP), the proxy processes are the WebSphere® Application Server application servers.
- A work factor estimator per target cell, which is an HAManagedItem process that can run in any node agent, ODR, or deployment manager.

The gateways intercept and queue the incoming HTTP, SIP, JMS, and IIOP requests, while the controller provides control signals, or directions, to the gateways and the placement controller. The work profiler continually estimates the computational requirements of the various kinds of requests, based on observations of the system in operation. Working together, these components properly prioritize incoming requests.

If you are using the Compute Grid product with the WebSphere Virtual Enterprise product, the dynamic application placement function with the job scheduler is supported. The application placement controller, along with the scheduler and autonomic request flow manager, provides overload protection of servers as long as both online and batch workloads are on dynamic clusters. This overload protection is not supported for static clusters. Because batch jobs can consume a lot of processor and run for a long period of time, the server utilization limit might be exceeded.

When the WebSphere Virtual Enterprise product is installed with the Compute Grid product, the application placement controller is consulted by the job scheduler during its endpoint selection process. You can configure the UseAPCEndpointSelection custom property set to false on the job scheduler to disable the application placement controller/job scheduler integration. Use this custom property to disable the application placement controller during the job scheduler's endpoint selection process.

z/OS Dynamic placement function with job scheduler is not supported on z/OS® servers.

Procedure

1. Modify the appropriate ARFM settings. In the administrative console, click **Operational policies > Autonomic Managers > Autonomic Request Flow Manager**.
2. Click **OK** or **Apply** when you have completed your changes.
3. Click **Save** to save the changes to the master repository.
4. Test the settings you have just defined and iterate as often as necessary to get the request flow performance that you want.

Example

The following table provides specific guidance for configuring each setting.

Table 1. ARFM configuration properties

Field	Purpose	Tips for setting
Aggregation period	Each ARFM gateway broadcasts aggregated statistics periodically, and this parameter specifies the period. The statistics reported by the gateways support: the runtime charting in the administrative console, the operation of ARFM controllers, the operation of the application placement controller, and the operation of work profilers.	<p>When setting the aggregation period, ensure the value is high enough to allow for the collection of a sufficient number of performance samples. Samples are collected by the gateways for each request. A few hundred samples are necessary to produce a good statistical measure.</p> <p>Using an example - requests associated with a service class run in 250 milliseconds, and on average 10 requests run concurrently. The concurrency value is calculated automatically, based on the cluster size and the resources in the environment. The concurrency value can</p>

		<p>be seen on the visualization panels, under the Runtime Operations category in the console. As a result, the service class handles about 40 requests per second. Therefore, setting the aggregation period value to 15 seconds results in the collection of 600 samples for each aggregation period. The metrics provided by a 600 sample survey are useful and reliable.</p> <p>Setting an aggregation period value too low results in unreliable performance metrics. Performance metrics derived from fewer samples are more noisy and less reliable, than a higher sample size. Because the ARFM controller is activated when new statistics are produced, setting an aggregation period value that is too long results in less frequent recomputation of the control settings. Therefore, WebSphere Virtual Enterprise becomes less responsive to sudden changes in traffic intensities and patterns.</p>
Control cycle length minimum	<p>This parameter defines how often the ARFM controller is activated. Controller activation is the process of evaluating inputs and producing new control settings as a result of the input received. The activation process for an ARFM controller is initiated when new statistics are received from one of its gateways AND the elapsed time since the previous activation is greater than or equal to the control cycle minimum length, or the controller has never activated before.</p>	<p>This setting determines the control cycle length giving it a lower bound. For example, if you have just one ODR and set the aggregation period to 30 seconds and the control cycle minimum length to 60 seconds, you might find that one activation occurs at 12:00:00.0 and the next occurs 90.1 seconds later at 12:01:30.1 because the previous statistics arrival time was 12:00:59.9. To ensure a reliable control cycle of around 60 seconds, set the control cycle minimum length to 58 or 59 seconds.</p>
Smoothing window	<p>This setting defines how sensitive the ARFM controller reaction is to the incoming gateway statistics, by allowing a concatenation of gateway statistics. For any gateway, its ARFM controller uses a running average of the last few statistics reports from that gateway. The smoothing window controls the number of reports that are combined.</p>	<p>A low smoothing window setting makes the controller more sensitive and react more quickly. However, a low parameter also creates a sensitive reaction to noise, or anomalies, in the data.</p> <p>The product of the smoothing window and the aggregation period should be roughly the same as the actual control cycle length, which is sometimes slightly greater than the configured control cycle minimum length.</p>
Maximum queue length	<p>This parameter is used to bound the length of each ARFM queue to a maximum number of requests that can be held in queue. ARFM divides all incoming traffic into flows, and has a separate queue for each flow. Flow particulars include requests that have a particular service class, are served on a particular deployment target, or go through a particular ODR.</p> <p>When a request arrives and its queue is full, the request is rejected.</p>	<p>A lower parameter in this field increases the possibility that a request will be rejected due to short-term traffic bursts, while a higher parameter in this field can allow requests to linger longer in the queues. Queued requests consume memory. The default setting is 1000, but you can experiment with this setting to find the one that is a best match for your environment.</p>
Maximum CPU usage	<p>The ARFM provides overload protection, in addition to its</p>	<p>Higher values give better resource utilization; lower values give more robust operation. Real</p>

	<p>prioritization capabilities. An ARFM will queue requests in its gateways to avoid overloading the application servers.</p> <p>For this release, load is determined in terms of processor utilization on the first tier of application servers. The maximum CPU utilization parameter tells ARFM how heavily to load the servers. During severe peak conditions this utilization limit might be briefly exceeded.</p>	<p>load is noisy and variable. The performance management techniques in WebSphere Virtual Enterprise react to changes in the load, but with some time delay. During that reaction time, the system might operate outside its configured region; this includes having higher processor utilization than configured. Operation with one application server at 100 percent processor utilization for multiple minutes has been observed to break some internal communication mechanisms, to the detriment of many features.</p> <p>The performance management in this release of WebSphere Virtual Enterprise does not work well if the first tier of application server machines are loaded with other work besides WebSphere requests that arrive through HTTP through the ODRs.</p> <p>This setting affects application placement. If the total predicted demand is above the Maximum CPU utilization limit, the placement controller uniformly reduces the demand of all the dynamic clusters before calculating best placement.</p> <p>Set the arfmManageCpu custom property to false to disable processor overload protection and request prioritization. The arfmManageCpu is a cell custom property that you need to create.</p>
<p>Admission control for CPU overload protection</p>	<p>The purpose of admission control for processor overload protection is to deliberately <i>not</i> accept dialogs based on judgments concerning how much can be accepted without overloading the compute power in the nodes being managed and compromising the response time of the accepted messages.</p> <p>The Admission control for CPU overload protection value applies only to HTTP and Session Initiation Protocol (SIP); it does not apply to IIOP and JMS.</p> <p>Enable it when queuing for processor overload protection is not enough; when it is important to make deliberate refusals of some offered load.</p>	<p>Disabled by default. To configure:</p> <ol style="list-style-type: none"> 1. Define service policies with achievable performance goals, and set the goal type of the policies either to response time or percentile, not discretionary. 2. In the ARFM panel, set the CPU utilization limit to no higher than 90%. Select the third button for Rejection Policy. The rejection policy determines whether the admission control for processor overload protection is enabled and, if so, how the response time threshold used for admission control is related to the response time threshold that appears in the performance goal. 3. At the cell level, set a cell custom property named arfmInitialMsgDlgRatio. The value is a decimal-formatted float that is the initial estimate for the ratio of each of the dialog-continuing message flows to the dialog-initiating message flow within the same (protocol family, deployment target). That is, it is the number of incoming follow-up messages per dialog. Set arfmInitialMsgDlgRatio to a value that is comparable among the collection of all dialog-continuing message flows. <p>This custom property is also relevant when dialog orientation for processor overload protection and differentiated service is enabled.</p> <ol style="list-style-type: none"> 4. Save your changes. <p>The admission control for processor overload protection is working if, in a heavily loaded system, the processor utilization is about the same as the setting for processor overload protection.</p>
<p>See Memory</p>	<p>Specifies the maximum</p>	<p>Maximum percentage of the WebSphere</p>

overload protection for more information.	percentage of the heap size to be used for each application server.	Application Server heap size to use. Set the value to less than 100.
Request rejection policy	Specifies the behavior for HTTP, SIP and SOAP requests that are associated with a performance goal when an overload condition is detected.	Choose among the options to determine when to reject messages to prevent the CPU from being overloaded. You can reject no messages, or specify a rejection threshold value that determines when to reject messages. The default is to reject no messages. Discretionary work is assumed to have a response time threshold of 60 seconds.

What to do next

Use the mustGather documents to troubleshoot the autonomic request flow manager and application placement. The Support team provides and maintains the [mustGather documents](#) for each version of WebSphere Extended Deployment.

[Rate-based autonomic request flow manager \(ARFM\)](#)

In WebSphere Virtual Enterprise Version 7.0, the autonomic request flow manager (ARFM) uses a rate-based algorithm that results in a more consistent loading and protecting of application server resources by ARFM.

[Configuring emergency throttle](#)

The on demand router (ODR) and associated autonomic managers are able to support business goals in times of intense request flows by making smart decisions about the work coming into the server. The autonomic request flow manager (ARFM) controls HTTP request prioritization in the ODR. At times, emergency conditions result when certain sensors detect such overloaded situations. These overload situations include extremely high node utilization, intermittent communication failures between ARFM controller and request scheduling gateways, and intermittent communication failures between AsyncPMI monitoring data producers and the gateways. To prevent prolonging of these conditions, should they occur, and the accompanying degradation in performance, the gateways are equipped with emergency throttle controllers that control, and safeguard request dispatch rates to backend nodes. ARFM is handled in the back end for IIOP/JMS requests.

[Memory overload protection](#)

Memory overload protection limits the rate at which the on demand router (ODR) forwards traffic in order to prevent an OutOfMemoryException from occurring in an application server. If traffic without server affinity arrives at the ODR and the rate for all potential servers has been exceeded, the traffic is rejected. Memory overload protection does not reject traffic that has server affinity. For example, HTTP requests with session affinity or SIP in-dialog messages.

[Configuring memory overload protection](#)

Follow these instructions to configure memory overload protection from the administrative console.

[Trace settings for autonomic request flow manager and application placement](#)

Use the WebSphere Virtual Enterprise mustGather documents to troubleshoot the autonomic request flow manager and application placement.

[Request prioritization problems](#)

Occasionally, you might encounter flow prioritization behavior that is unexpected. You can look for some common things when request flow prioritization is not working as expected.

Parent topic: [Configuring the autonomic managers](#)

Related concepts:

[Memory overload protection](#)

Related tasks:

[Managing the WebSphere Virtual Enterprise environment](#)

[Configuring work factors in multiple tier configurations](#)

[Defining a service policy](#)

Related reference:

[Administrative roles and privileges](#)

[arfmController.py script](#)

Related information:

[Autonomic request flow manager custom properties](#)

[Autonomic request flow manager advanced custom properties](#)

Rate-based autonomic request flow manager (ARFM)

In WebSphere® Virtual Enterprise Version 7.0, the autonomic request flow manager (ARFM) uses a rate-based algorithm that results in a more consistent loading and protecting of application server resources by ARFM.

ARFM controls the flow of requests for HTTP and SIP traffic through the on demand router (ODR) and for IIOP and MDB traffic from within an application server. In releases prior to Version 7.0, ARFM enforces a concurrency limit, such as the maximum number of requests that can be outstanding, on a per resource basis. The resources include an ODR process, protocol, deployment target, and service policy. This concurrency limit poses problems in some scenarios in which periodic long-running or hung requests occur.

In Version 7.0, the rate-based ARFM feature is enabled by default. Complete the following steps to disable the feature.

1. In the administrative console, select **System administration > Cell > Custom Properties > New**.
2. Specify the name of the custom property as `enableRateBasedARFM`.
3. Specify the value of the custom property as `false`.
4. Restart the cell.

Parent topic: [Configuring the autonomic request flow manager](#)

Configuring emergency throttle

The on demand router (ODR) and associated autonomic managers are able to support business goals in times of intense request flows by making smart decisions about the work coming into the server. The autonomic request flow manager (ARFM) controls HTTP request prioritization in the ODR. At times, emergency conditions result when certain sensors detect such overloaded situations. These overload situations include extremely high node utilization, intermittent communication failures between ARFM controller and request scheduling gateways, and intermittent communication failures between AsyncPMI monitoring data producers and the gateways. To prevent prolonging of these conditions, should they occur, and the accompanying degradation in performance, the gateways are equipped with emergency throttle controllers that control, and safeguard request dispatch rates to backend nodes. ARFM is handled in the back end for IIOP/JMS requests.

The ARFM contains two parts: a controller and a gateway. The ARFM function is implemented, for each node group, by a controller plus a collection of gateways in the ODRs. The ARFM controller (triggered by the eWLM controller if available on the system) might initiate typical throttling directives to the gateways. In a typical mode, throttling directives come from the ARFM controller by way of the RatesMessages, and are immediately enforced at the gateway by the throttle controller.

A throttle is attached to each queue in the gateway, and is not in the throttle state by default. When an emergency occurs or when rate messages arrive from the ARFM controller, it receives directives from the throttle controller and changes to the throttled state.

In the event that one or more overload sensors detect overload condition, in spite of typical throttling, the gateway throttle controller enters emergency mode. An emergency blackout sensor senses communication failures between an ARFM controller and request scheduling gateways, or communication failures between AsyncPMI monitoring data producers and the gateways. The term *blackout* means that the sensor does not receive expected messages. In emergency mode, the throttle controller gradually reduces the dispatch rates of the gateway queues until the overloaded sensor(s) stops firing. Then it gradually restores the rates to their original, pre-emergency mode settings. While doing this, the throttle controller ensures that rate directives from ARFM controller are never exceeded, thereby preserving the integrity of throttling decisions made by different controllers. Working together, these components can properly limit incoming requests.

Multiple sensors detect emergency conditions, resulting in the throttle controller going into emergency mode. Each sensor can be in one of two states: fired or unfired. During an emergency, there are two phases for the throttle controller: `emergency_throttle` and `emergency_unthrottle`. During the `emergency_throttle` phase, the throttle reduces all queue rates as long as one of the sensors still fires. In the `emergency_unthrottle` phase, all the sensors return to the unfired state and gradually restore all queue rates to their original values they had before entering emergency mode.

In Version 6.1.0.5 and later, emergency throttling is disabled by default. ARFM4998W messages might still be displayed in the log if an emergency condition is detected, however this condition does not throttle traffic. You can enable emergency throttling by adding the following entry to the `WAS_HOME/profiles/node/properties/arfm.cfg` file on the ODR host.

```
EnableEmergencyThrottling=true
```

Enforcing rate directives from ARFM controller (initiated by eWLM) is enabled by default. You can disable it by adding the following entry to the `arfm.cfg` file.

```
EnableExternalThrottling=false
```

See the following list for other configuration parameters that you can add to the `arfm.cfg` file.

- `EmergencyRateChangeStep=x` where `x` is an integer in the range 0 to 100, specifying the percentage change in rate in each step of the gradual reduction/increase of throttle rate. The default value is 20.
- `EmergencyRateChangeInterval=x` where `x` is the time between two successive rate change steps in emergency mode in milliseconds. The default value is 15000.
- `EmergencyBlackoutMultiplier=x` where `x` is a multiplier multiplied by different normal message cycles used as input to emergency blackout sensor. The `EmergencyBlackoutMultiplier` parameter is a configuration parameter which tells the sensor indirectly how long it should wait before firing. This interval is determined as the product (multiplication) of the parameter and the normal anticipated interval between successive messages. The default value is 2.
- `EmergencyCPUUtilLimit=x` where `x` is an integer in the range of 0 to 100, specifying the processor utilization watermark on backend nodes, that triggers emergency throttling. The default value is 100.
- `TokenBucketSizeMillis=x` where `x` is the number of tokens that can be accumulated in queue's token bucket. The default value is 1000.

Parent topic: [Configuring the autonomic request flow manager](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

Memory overload protection

Memory overload protection limits the rate at which the on demand router (ODR) forwards traffic in order to prevent an `OutOfMemoryException` from occurring in an application server. If traffic without server affinity arrives at the ODR and the rate for all potential servers has been exceeded, the traffic is rejected. Memory overload protection does not reject traffic that has server affinity. For example, HTTP requests with session affinity or SIP in-dialog messages.

To protect against memory overload, memory overload protection must initially discover the maximum rate, that is, calls per second, that can be sustained without exceeding the maximum percentage of the maximum heap size. As it is discovering the maximum rate, memory overload protection slowly allows more traffic through without affinity, but will reject the remainder. Initially, a potentially large number of HTTP requests or SIP messages without affinity are rejected with a 503 (unless the error code is changed). WebSphere Virtual Enterprise persists the maximum rate across server restarts, so it needs to learn the maximum rate once. The maximum rate can change over time due to changes in the session or dialog lifetimes, but these lifetimes generally change relatively slowly and memory overload protection is able to react to such changes. To discover the maximum rate, the product must keep the rate relatively steady for at least an averaging window. The averaging window must be at least as long as the lifetime of most of the HTTP sessions, SIP dialogs, or application sessions. Therefore, the longer the averaging window, the longer it will take to initialize.

For SIP and HTTP, memory overload protection and CPU overload protection are not guaranteed to work if a dynamic cluster is in automatic mode, because of the CPU and heap overhead incurred by replication

WebSphere eXtreme Scale considerations

WebSphere® eXtreme Scale might allocate additional memory in a running application server when another application server starts or stops. Memory overload protection does not currently control this memory allocation. So, if the memory utilization is already high, the additional uncontrolled allocation of memory can cause an `OutOfMemoryException` to occur. For example, if the maximum percentage memory setting is 90% and the current heap utilization is near 90% in application server AS1, and application server AS2 starts or stops, an `OutOfMemoryException` might occur in AS1 due to replication to AS2. The maximum heap percentage should be set low enough so that there is always enough memory in reserve for the potential replication needs when an application server starts or stops. Memory overload protection will prevent an `OutOfMemoryException` in a dynamic cluster if the maximum percentage memory setting is set to 56%.

Parent topic: [Configuring the autonomic request flow manager](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring memory overload protection](#)

Related information:

[Autonomic request flow manager custom properties](#)

Configuring memory overload protection

Follow these instructions to configure memory overload protection from the administrative console.

Before you begin

Configure the autonomic request flow manager (ARFM). See [Configuring the autonomic request flow manager](#).

About this task

To protect against memory overload, the memory overload protection must initially discover the maximum rate, that is, calls per second, that can be sustained without exceeding the maximum percentage of the maximum heap size. As it is discovering the maximum rate, memory overload protection slowly allows more traffic through without affinity, but will reject the remainder. Initially, a potentially large number of HTTP requests or SIP messages without affinity will be rejected with a 503 (unless the error code is changed). WebSphere Virtual Enterprise persists the maximum rate across server restarts, so it must discover the maximum rate once. The maximum rate can change over time due to changes in the session or dialog lifetimes, but these lifetimes generally change relatively slowly and memory overload protection is able to react to such changes. When discovering the maximum rate, WebSphere Virtual Enterprise must keep the rate relatively steady for at least an averaging window. The averaging window must be at least as long as the lifetime of most of the HTTP sessions, SIP dialogs, or application sessions. Therefore, the longer the averaging window, the longer it will take to start.

Memory overload protection is disabled by default. To enable, follow these steps from the administrative console:

Procedure

1. Expand **Operational Policies > Autonomic Controllers > Autonomic Request Flow Manager**
2. Type a value less than 100 in the **Memory overload protection: Maximum percentage of the WebSphere® Application Server maximum heap size to use** field.
3. Expand **Servers > Dynamic clusters > <dynamic cluster> Server template > Class loader > Process definition > Java™ Virtual Machine**
4. Set **Generic JVM arguments** to `-agentlib:HeapDetect -Xgcpolicy:gencon -Xdump:heap:events=user`.

Parent topic: [Configuring the autonomic request flow manager](#)

Related concepts:

[Memory overload protection](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

Related information:

[Autonomic request flow manager custom properties](#)

Trace settings for autonomic request flow manager and application placement

Use the WebSphere® Virtual Enterprise mustGather documents to troubleshoot the autonomic request flow manager and application placement.

The Support team provides and maintains the [mustGather documents](#) for each version of WebSphere Virtual Enterprise.

Parent topic: [Configuring the autonomic request flow manager](#)

Related concepts:

[Configuring emergency throttle](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

Related reference:

[Request prioritization problems](#)

Request prioritization problems

Occasionally, you might encounter flow prioritization behavior that is unexpected. You can look for some common things when request flow prioritization is not working as expected.

HTTP requests are all discretionary

If your environment treats all of the incoming requests equally, you might not have a service policy defined and applied to the proper application module Uniform Resource Identifiers (URI). A best effort approach, also called *discretionary*, is the default policy. Take the following actions to ensure that the service policy is configured and applied:

Action	Performed by
Verify that your service policies are created.	From the administrative console, click Operational policies > Service policies . All of the currently defined service policies display. If you do not see your service policy listed, configure a new service policy by clicking New .
Verify that your service policies are applied to the proper application URIs.	From the administrative console, click Operational policies > Service policies > Select an existing service policy . Verify the assigned transaction classes are in the Transaction Classes field. You can create a new transaction class by clicking New . If you do not see the transaction class members you are looking for, check that they are not already assigned to another service policy. Also, ensure that the application to which you are applying a service policy to is deployed in your environment.

Slow response time for request prioritization

If the autonomic request flow manager (ARFM) is not reacting fast enough, resulting in slowed response time for request prioritization, then you can tune the ARFM settings. See [Configuring the autonomic request flow manager](#) for more information. In particular, review the **Set Control Cycle Length Minimum** setting to ensure that a suitable value is selected.

CPU use remains at 100% on one or more back-end nodes

ARFM is continuously computing the amount of work each transaction class requires from the system and refining as requests come into the system. To ensure that the ARFM is optimized, run your system for significant periods of time with largely varying workloads. This fluctuation of work, combined with significant up time, allows ARFM to fine tune itself and provide more accurate estimates to avoid problem situations.

If one or more nodes remain at high use rates, while the remainder of the nodes are working comfortably, you might want to fine tune the ARFM. See [Configuring the autonomic request flow manager](#) for more information. In particular, review **Maximum CPU Utilization** setting to ensure that you select a lesser value than your node is currently utilizing.

Verify that you are consistently grouping transaction classes. For example, do not group URIs with a discretionary, or low service value in the same transaction class as URIs with a high service policy value. Mixing requests with widely varying demands in the same transaction class causes the ARFM to produce inaccurate estimates. To modify your transaction classes, click **Operational policies > Service policies > Select an existing service policy** and verify the transaction class field to ensure consistent groupings.

In a heterogeneous cell, not all the nodes in the cell are used equally

The system is working as designed. The load balancer tries to equalize response time on all the back-end nodes in a cluster. If one node is less powerful than another, then the load balancer might distribute less work to the less powerful node so that response time can be similar to the response time from a faster node.

MustGather documents

Use the WebSphere® Virtual Enterprise mustGather documents to troubleshoot the autonomic request flow manager and application placement. The Support team provides and maintains the [mustGather documents](#) for each version of WebSphere Virtual Enterprise.

Parent topic: [Configuring the autonomic request flow manager](#)

Related concepts:
[Operational policies](#)

Related tasks:
[Configuring the autonomic request flow manager](#)

Configuring dynamic application placement

Dynamic application placement is the process by which the features of the dynamic operations of WebSphere Virtual Enterprise work together to start and stop application instances to meet the fluctuating demand of work requests of varying service policy definitions. This set of actions is controlled by the application placement controller.

Before you begin

- Before configuring application placement, create an on demand router (ODR) and configure the autonomic request flow manager. See [Configuring the autonomic request flow manager](#).
- Create a service policy. See [Defining a service policy](#).
- Verify that you have the correct administrative role. If you want to change the configuration of the application placement controller, you must have a Configurator or Administrator administrative role. If you want to change runtime settings, you must have an Operator or Administrator administrative role.

Procedure

- [Creating dynamic clusters](#)

Dynamic clusters are the deployment target for your applications. When you create a dynamic cluster, you define a minimum and maximum number of application instances, or cluster members. The application placement controller works to keep these numbers of instances available along with working to meet the demand of your defined service policies.

- [Monitoring and tuning the application placement controller](#)

You can enable or disable the application placement controller and adjust other settings such as the minimum amount of time between placement changes, the server operation timeout, and so on.

- [Configuring multi-cell performance management](#)

Configure and manage multi-cell performance in your environment to avoid overprovisioning resources, such as CPU and memory utilization.

- [Configuring elasticity mode](#)

You can enable elasticity mode to add logic that causes the application placement controller to minimize the number of nodes that are used, as well as remove nodes that are not needed, while still meeting service policy goals.

[Overview of application placement](#)

Dynamic application placement for Internet InterORB Protocol (IIOP), Java Message Service (JMS), and HTTP request types is achieved through the cooperation of various product components. These components include dynamic clusters, the on demand router (ODR) and associated autonomic managers, and the application placement controller. The fluctuating volume of work requests for applications is dynamically met on the application server side by the application placement feature, which ensures the integrity of the defined business goals.

[Dynamic application placement](#)

In WebSphere Virtual Enterprise, dynamic application placement is based on load distribution, service policy, and available resources. You can run applications at various levels within a WebSphere Virtual Enterprise environment, ranging from changing relative weights of applications to expanding a WebSphere Virtual Enterprise cell. As changes become more drastic and their impacts become more extreme, applications take more time to run.

[Configuring multi-cell performance management](#)

Configure and manage multi-cell performance in your environment to avoid overprovisioning resources, such as CPU and memory utilization. You can configure multi-cell performance management in a star topology either automatically or manually.

[Manually disabling communication between multiple cells](#)

You can complete a set of steps to manually disable the overlay communication between multiple cells as an alternative to running the `unlinkCells.sh|.bat` script.

[Configuring elasticity mode](#)

Configure elasticity mode to add logic that causes the application placement controller to minimize the number of nodes that are used, as well as remove nodes that are not needed, while still meeting service policy goals. Additionally, you can configure elasticity mode to add logic so that when the controller recognizes a particular dynamic cluster is not meeting service policies and has started all possible servers, the controller adds a node.

[Monitoring and tuning the application placement controller](#)

The application placement controller is designed to work with the default settings. However, there can be times when fine tuning the application placement controller becomes necessary to ensure the best results.

[Application placement frequently asked questions](#)

Occasionally, you might encounter application placement behavior that is not expected. This topic describes some commonly asked questions and things to look for when application placement is not working as you expect.

[Application placement controller logs](#)

You can use the log files to troubleshoot issues with the application placement controller, such as the placement of new instances.

Parent topic: [Configuring the autonomic managers](#)

Related concepts:

[Overview of application placement](#)

Related information:

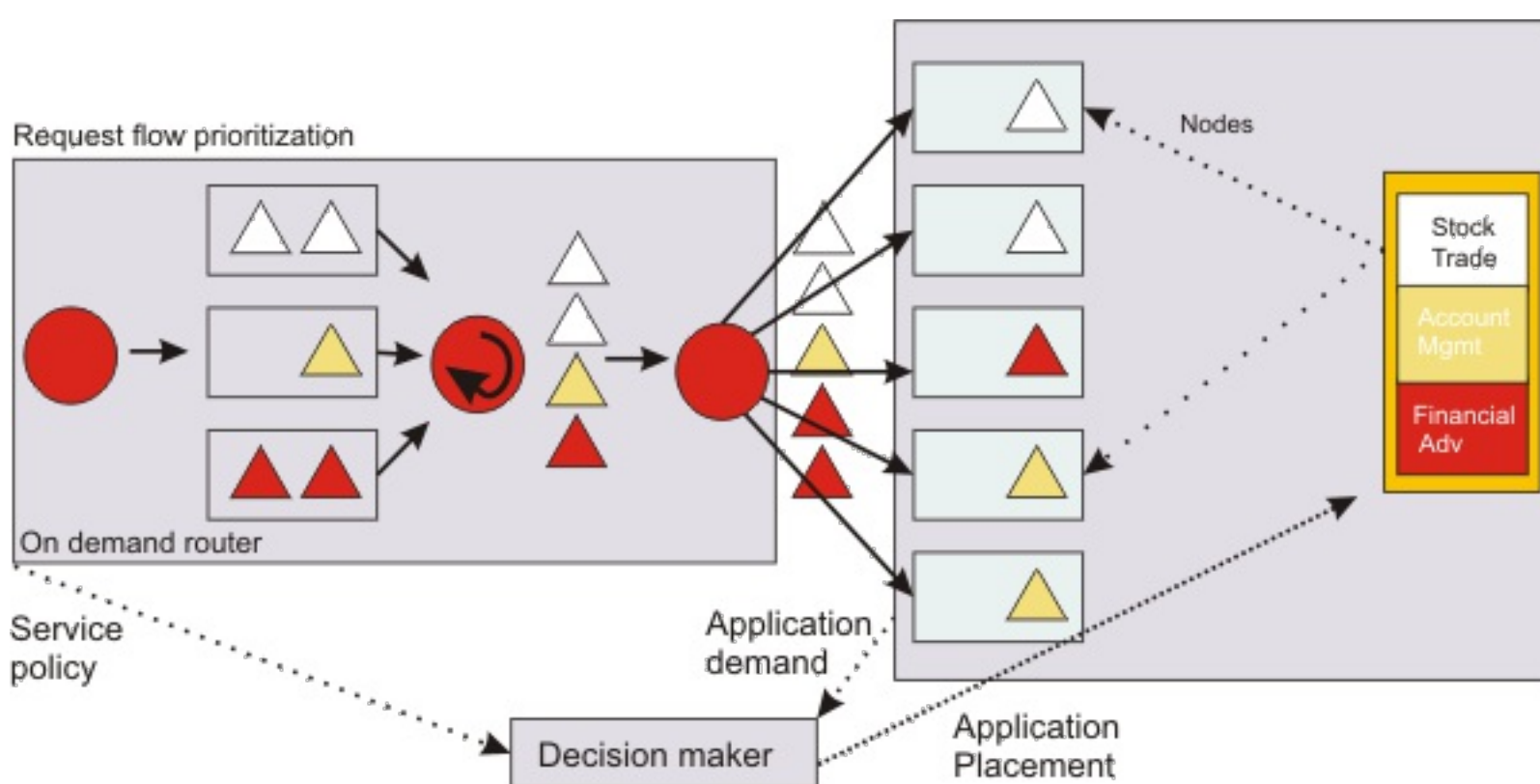
[Application placement custom properties](#)

Overview of application placement

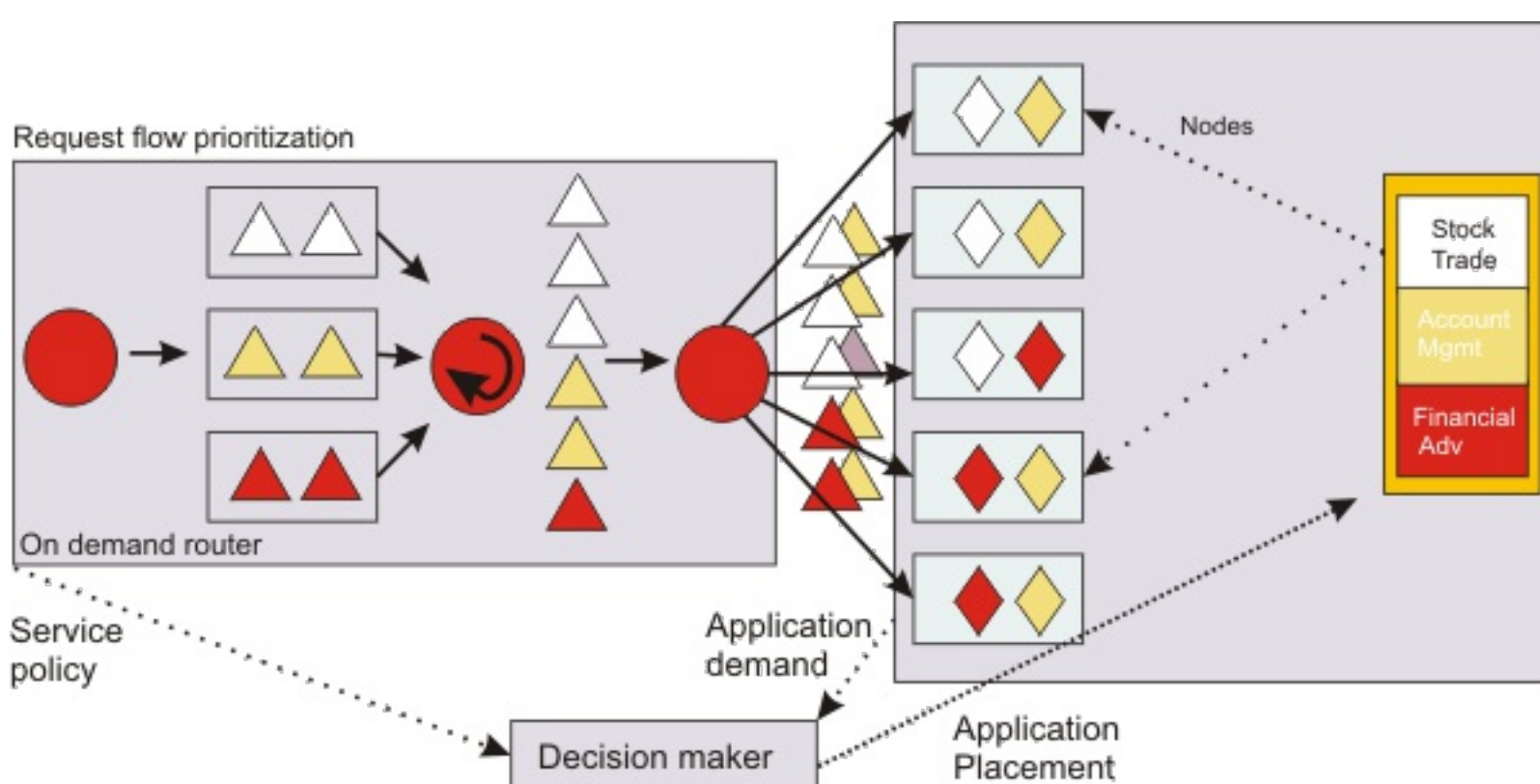
Dynamic application placement for Internet InterORB Protocol (IIOP), Java™ Message Service (JMS), and HTTP request types is achieved through the cooperation of various product components. These components include dynamic clusters, the on demand router (ODR) and associated autonomic managers, and the application placement controller. The fluctuating volume of work requests for applications is dynamically met on the application server side by the application placement feature, which ensures the integrity of the defined business goals.

Application placement is a natural extension to the request flow prioritization feature. While the ODR and its associated autonomic managers ensure that the work flows appropriately according to the defined policy, the application placement feature ensures that the applications and the nodes that they run on are kept at appropriate levels to support the influx of work. In times of less work flow, the application instances that run within the resource pool are kept at a minimum. In times of significant work flow, the application instances that run are increased to keep pace with the requests.

The following image shows the application placement process for a stock trading firm in times of less demanding workflow; that is, the request flow is during an off-peak hour. The ODR and its managers are controlling the incoming request flow prioritization. The decision making unit of the application placement function is made aware of the service policy for the incoming Universal Resource Identifier (URI) requests, as well as the demand level for the given application server resources. As a result, the application instances that run are comfortable for the given demand.



The following image shows a sudden increase in incoming URI requests occurs. For example, it is now lunch time and all online trade customers are now free to access the Web for personal use. The sudden work request fluctuation, which in other environments requires manual intervention to address, is handled by the autonomic features of dynamic operations.



During times of more robust request flow, the application placement function must ensure that the business goals created for the many application URIs are met. As a result, the autonomic managers of dynamic operations must work together to bring a balance to the environment. You can see this balance achieved in the previous graphic. The work flow fluctuation is met by increasing the application instances that run on the available nodes, while balancing the service policy definitions. When the peak declines, the application placement function reduces the running number of application instances accordingly.

The functionality of dynamic application placement requires fundamental WebSphere® Virtual Enterprise

components. Application placement requires dynamic clusters and service policies. . Dynamic clusters are application deployment targets. Service policies are performance goals that are assigned to a specific application URI.

These components, when integrated with the autonomic managers, support dynamic application placement.

If you are using the WebSphere Extended Deployment Compute Grid component with the WebSphere Virtual Enterprise component, the dynamic application placement function with the job scheduler is supported. The application placement controller, along with the scheduler and autonomic request flow manager, provides overload protection of servers as long as both online and batch workloads are on dynamic clusters. This overload protection is not supported for static cluster members. Because batch jobs can consume a lot of processor capacity and run for a long period of time, the utilization limit might be exceeded.

When the WebSphere Virtual Enterprise component is installed with the WebSphere Extended Deployment Compute Grid component, the application placement controller is consulted by the job scheduler during its endpoint selection process. You can configure the UseAPCEndpointSelection custom property on the job scheduler to `false` to disable the application placement controller and job scheduler integration. Using this custom property to prevents the job scheduler from asking the application placement controller to choose an endpoint. The particular endpoint is chosen by the job scheduler when the custom property is set to `false`.

Parent topic: [Configuring dynamic application placement](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring dynamic application placement](#)

Related information:

[Job scheduler custom properties](#)

Dynamic application placement

In WebSphere® Virtual Enterprise, dynamic application placement is based on load distribution, service policy, and available resources. You can run applications at various levels within a WebSphere Virtual Enterprise environment, ranging from changing relative weights of applications to expanding a WebSphere Virtual Enterprise cell. As changes become more drastic and their impacts become more extreme, applications take more time to run.

By dynamically placing resources, WebSphere Virtual Enterprise can utilize hardware more efficiently. It is unlikely that all applications are in high demand at the same time, assuming a varied assortment of deployed applications. WebSphere Virtual Enterprise exploits this situation by supporting resource allocation where needed, thereby increasing the utilization of hardware. The result is that an enterprise no longer requires enough hardware to satisfy each application maximum load simultaneously, which can translate to a significant reduction of requisite IT investments.

The dynamic application placement capability is contained by three autonomic managers: the application placement controller, dynamic workload manager (DWLM), and autonomic request flow manager (ARFM).

Each autonomic manager provides capabilities that work to achieve a common goal, which is to allocate the available capacity among the deployed applications and configured services classes. When it is necessary to modify the capacity that is allocated to a given application, the autonomic manager takes actions starting with the first or fastest level. As changing demand renders these lightweight adjustments ineffective, more drastic measures are required. For example, the WebSphere Virtual Enterprise autonomic managers might start by changing the dispatching priorities of requests that are associated with a service policy. WebSphere Virtual Enterprise makes such adjustments every few seconds. However, such adjustments might be ineffective if the size of a given cluster is too small. WebSphere Virtual Enterprise changes the sizes of dynamic clusters over a time scale of tens of minutes.

Dynamic application placement challenges

The challenge of dynamically placing a given application includes increasing and decreasing its capacity, which can be accomplished at several levels. The decisions on how to change the resource allocations are made using the autonomic managers that monitor various performance metrics, analyze the monitored data, and then guide the planned actions. WebSphere Virtual Enterprise supports various configurable levels of autonomicity. In the most autonomic mode, autonomic managers are granted the freedom to run the actions they plan. In the most manual mode, autonomic managers suggest actions, which an administrator must run manually.

Parent topic: [Configuring dynamic application placement](#)

Related concepts:

[Overview of dynamic operations](#)

Related tasks:

[Configuring dynamic application placement](#)

Configuring multi-cell performance management

Configure and manage multi-cell performance in your environment to avoid overprovisioning resources, such as CPU and memory utilization. You can configure multi-cell performance management in a star topology either automatically or manually.

Before you begin

- If you are using WebSphere® Virtual Enterprise Version 7.0, you must install PM37795 in your WebSphere Application Server Network Deployment environment before you can run the linkCells.sh|.bat script. See [Fix Central](#) to download the interim fix.
- Create and configure the cells that need to communicate with each other.

About this task

When you configure multi-cell performance management, multiple cells are managed as a single unit, because the cells share common resources. From a functional perspective, the performance management of WebSphere Virtual Enterprise, such as dynamic clusters and overload protection, operates on multiple cells of the same hardware in the same way that performance management operates in a single cell. This feature applies directly to the application placement controller to allow management, through dynamic placement and elasticity, throughout the topology.

You can configure multi-cell performance management either automatically or manually in a star topology environment. A star topology applies to server virtualized environments, such as AIX® LPARs/WPARs, zLinux, VMware, and Solaris Zones, as well as non-virtualized environments in which multiple cells share the same physical hardware. In a star topology, a single cell is designated as the center cell while the other cells are designated as the point cells. The center cell can perform work, such as contain ODRs and application servers, and only the center cell makes autonomic decisions to start or stop application servers.

The following procedure describes a sample scenario in which multi-cell performance management is configured in a star topology environment so that work requests can be routed from an ODR to dynamic cluster members across cells. The ODR is installed and running on CellA, which is the center cell. The two-point cells, CellB and CellC, contain the dynamic clusters and applications.

Note: Complete either the automatic step or the manual step to configure multi-cell performance management. Do not complete both steps.

Procedure

- Automatically configure multi-cell performance management.
 1. Set the following custom property to designate CellA as the center cell.
 - a. In the administrative console, select **System administration > Cell > Custom properties > New**.
 - b. Specify the name of the custom property as CenterCell.
 - c. Specify the value of the custom property as true.
 - d. Click **OK**, and save and synchronize your changes.
 - Note:** Set only one custom property to true.
 2. Set the following custom property individually for each cell that you want to designate as a point cell. In this sample scenario, complete steps 2a through 2c to designate CellB as one-point cell. Next, repeat steps 2a through 2c to designate CellC as another point cell.
 - a. In the administrative console, select **System administration > Cell > Custom properties > New**.
 - b. Specify the name of the custom property as CenterCell.
 - c. Specify the value of the custom property as false.
 - d. Click **OK**, and save and synchronize your changes.
 3. If global security is enabled, in the administrative console, go to **Security > SSL certificate and key management**. Select the check box to dynamically update the run time when SSL configuration changes occur. Sync the nodes on all cells.
 4. **Distributed platforms** Run the **linkCells.sh** script from CellA to configure the overlay communication between the cells. The **linkCells.sh** script is available in the install_root/bin directory. Run the script to link the center cell with each point cell individually. Do not run the script to link the center cell to multiple point cells.

Note: Run the **linkCells.sh** script from the cell with the oldest product version.

```
./linkCells.sh
CellA_deployment_manager_host:CellA_soap_port:user_id:password
CellB_deployment_manager_host:
  CellB_soap_port:user_id:password
```

```
./linkCells.sh
CellA_deployment_manager_host:CellA_soap_port:user_id:password
```

```
CellC_deployment_manager_host:  
CellC_soap_port:user_id:password
```

z/OS On z/OS® systems running WebSphere Virtual Enterprise version 7.0.0.3 and later, run the **linkCellsZOS.sh** script from CellA to configure the overlay communication between the cells. The **linkCellsZOS.sh** script is available in the `install_root/bin` directory. Run the script to link the center cell with each point cell individually. Do not run the script to link the center cell to multiple point cells.

Note: Run the **linkCellsZOS.sh** script from the cell with the oldest product version.

```
./linkCellsZOS.sh  
CellA_deployment_manager_host:CellA_soap_port:user_id:password  
CellB_deployment_manager_host:  
CellB_soap_port:user_id:password
```

```
./linkCellsZOS.sh  
CellA_deployment_manager_host:CellA_soap_port:user_id:password  
CellC_deployment_manager_host:  
CellC_soap_port:user_id:password
```

5. If you installed middleware agents in the cells, add the middleware agents to the root certificates. Complete steps 4a through 4e for every cell.
 - a. In the administrative console, select **Security > SSL certificate and key management > Key stores and certificates > NodeDefaultKeyStore > Personal certificates > Create > Chained Certificate...**
 - b. Enter the name of the middleware agent in the **Alias** and **Common name** fields, and click **OK**.
 - c. Save and synchronize your changes.
 - d. Replace the default self-signed certificate with the new certificate. Select the default self-signed certificate from the table, and click **Replace...**
 - e. Select the **Delete old certificate after replacement** check box, verify that the name of the new certificate is displayed in the **Replace with** field, and click **OK**.
 - f. Save and synchronize your changes.
- Manually configure multi-cell performance management.
 1. Set the following custom property to designate CellA as the center cell.
 - a. In the administrative console, select **System administration > Cell > Custom properties > New**.
 - b. Specify the name of the custom property as `CenterCell`.
 - c. Specify the value of the custom property as `true`.
 - d. Click **OK**, and save and synchronize your changes.

Note: Set only one custom property to true.

2. Set the following custom property individually for each cell that you want to designate as a point cell. In this sample scenario, complete steps 2a through 2c to designate CellB as one-point cell. Next, repeat steps 2a through 2c to designate CellC as another point cell.
 - a. In the administrative console, select **System administration > Cell > Custom properties > New**.
 - b. Specify the name of the custom property as `CenterCell`.
 - c. Specify the value of the custom property as `false`.
 - d. Click **OK**, and save and synchronize your changes.
3. If global security is enabled, in the administrative console, go to **Security > SSL certificate and key management**. Select the check box to dynamically update the run time when SSL configuration changes occur. Sync the nodes on all cells.
4. Configure the overlay communication between the cells. Run the **importOverlayConfig.py** script, and specify the `overlaynodes.config` file of each cell. The `overlaynodes.config` file is in the `WAS_HOME/profiles/deployment_manager_profile_name/config/cells/cell_name` directory of each cell.
 - Run the **importOverlayConfig.py** script from the deployment manager on CellA to link the overlay anchor transports of CellA to CellB. Specify the path to the `overlaynodes.config` file of CellB.

```
WAS_HOME/bin/wsadmin.sh -profileName profile_name -f  
importOverlayConfig.py link  
path_to_CellB_overlaynodes.config_file
```

- Run the **importOverlayConfig.py** script from the deployment manager on CellA to link the overlay anchor transports of CellA to CellC. Specify the path to the `overlaynodes.config` file of CellC.

```
WAS_HOME/bin/wsadmin.sh -profileName profile_name -f
```

```
importOverlayConfig.py link
path_to_CellC_overlaynodes.config_file
```

- Run the **importOverlayConfig.py** script from the deployment manager on CellB to link the overlay anchor transports of CellB to CellA. Specify the path to the overlaynodes.config file of CellA.

```
WAS_HOME/bin/wsadmin.sh -profileName profile_name -f
importOverlayConfig.py link
path_to_CellA_overlaynodes.config_file
```

- Run the **importOverlayConfig.py** script from the deployment manager on CellC to link the overlay anchor transports of CellC to CellA. Specify the path to the overlaynodes.config file of CellA.

```
WAS_HOME/bin/wsadmin.sh -profileName profile_name -f
importOverlayConfig.py link
path_to_CellA_overlaynodes.config_file
```

5. If you installed middleware agents in the cells, add the middleware agents to the root certificates. Complete steps 4a through 4e for each and every cell.
 - a. In the administrative console, select **Security > SSL certificate and key management > Key stores and certificates > NodeDefaultKeyStore > Personal certificates > Create > Chained certificate**
 - b. Enter the name of the middleware agent in the **Alias** and **Common name** fields, and click **OK**.
 - c. Save and synchronize your changes.
 - d. Replace the default self-signed certificate with the new certificate. Select the default self-signed certificate from the table, and click **Replace**
 - e. Select the **Delete old certificate after replacement** check box, verify that the name of the new certificate is displayed in the **Replace with** field, and click **OK**.
 - f. Save and synchronize your changes.
6. Exchange the root certificates between the cells. Complete the following steps multiple times to extract and copy the root certificate of CellA to each point cell, and to extract and copy the root certificate of each point cell to CellA.
 - a. In the administrative console, select **Security > SSL certificate and key management > Key stores and certificates**.
 - b. Select **Root certificates keystore** from the **Keystore usages** menu. A new panel is displayed. Click **DmgrDefaultRootStore**.
 - c. Select **Personal certificates** from the list of **Additional Properties**, select the root certificate, and click **Extract**.
 - d. Specify the fully qualified path to where the file will be stored on the deployment manager in the **Certificate file name** field, for example /tmp/myrootcertificate. Click **OK**.
 - e. Save and synchronize your changes.
 - f. Copy the root certificate to the deployment manager of the target cell.
7. Specify the location of each root certificate between the cells. Complete the following steps multiple times to specify the location of the root certificate of CellA to each point cell, and to specify the location of the root certificate of each point cell to CellA. In this example, the location of the root certificate of CellA is specified in the administrative console of CellB.
 - a. In the administrative console of CellB, select **Security > SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore**.
 - b. Select **Signer Certificates** from the list of **Additional properties**. A new panel is displayed. Click **Add**.
 - c. Specify the location of the root certificate on the deployment manager of CellA, and click **OK**.
 - d. Save and synchronize your changes.

Parent topic: [Preparing the hosting environment for dynamic operations](#)

Parent topic: [Configuring dynamic application placement](#)

Related tasks:

[Configuring elasticity mode](#)

[Manually disabling communication between multiple cells](#)

[Managing the WebSphere Virtual Enterprise environment](#)

Related reference:

[Scripts](#)

[Custom properties](#)

Manually disabling communication between multiple cells

You can complete a set of steps to manually disable the overlay communication between multiple cells as an alternative to running the unlinkCells.sh|.bat script.

About this task

Typically, you enable the overlay communication between multiple cells when you configure multi-cell performance management in a star topology. Each cell contains a overlaynodes.config file, and the files must be copied between the cells to enable multi-cell performance management. Running the unlinkCells.sh|.bat script is one way to disable the overlay communication between the cells. You can also complete a set of steps to manually disable communication as an alternative to running the unlinkCells.sh|.bat script.

For example, assume you set up a star topology in which three cells are running, and you want to disable communication between certain cells without disrupting service to the other cells in the environment. CellA is the center cell, and CellB and CellC are the point cells. Complete the following procedure to manually disable communication between CellA and CellB while maintaining communication between CellA and CellC.

Procedure

1. Copy the overlaynodes.config file from the deployment manager of CellB to a temporary directory on CellA's deployment manager. For example, copy the file to the /tmp/CellB_overlaynodes.config directory. The overlaynodes.config file is in the WAS_HOME/profiles/deployment_manager_profile_name/config/cells/cell_name directory of each cell.
2. Run the importOverlayConfig.py script from the deployment manager on CellA. Specify the path to the temporary directory in which you saved the overlaynodes.config file of CellB.

```
WAS_HOME/bin/wsadmin.sh -lang jython -f ./importOverlayConfig.py unlink /tmp/CellB_overlaynodes.config
```

3. Copy the overlaynodes.config file from the deployment manager of CellA to a temporary directory on CellB's deployment manager. For example, copy the file to the /tmp/CellA_overlaynodes.config directory.
4. Run the importOverlayConfig.py script from the deployment manager on CellB. Specify the path to the temporary directory in which you saved the overlaynodes.config file of CellA.

```
WAS_HOME/bin/wsadmin.sh -lang jython -f ./importOverlayConfig.py unlink /tmp/CellA_overlaynodes.config
```

5. Optional: If security is enabled, you might need to delete the security certificates that were exchanged when the cells were linked together.
 - a. Log on to the administrative console of CellA, and delete the security certificate of CellB. Click **Security > SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore**.
 - b. Select **Signer Certificates** from the list of **Additional Properties**. A new panel is displayed.
 - c. Select the signer certificate for the remote cell, and click **Delete**.
 - d. Save and synchronize your changes.
 - e. Log on to the administrative console of CellB, and repeat the steps to delete the security certificate of CellA.
6. Delete the CenterCell custom property from each cell.
 - a. In the administrative console of CellA, click **System administration > Cell > Custom Properties**.
 - b. Select the CenterCell custom property, and click **Delete**.
 - c. Save and synchronize your changes.
 - d. Log on to the administrative console of CellB, and repeat the steps to delete the CenterCell custom property from CellB.

Parent topic: [Configuring dynamic application placement](#)

Related tasks:

[Configuring elasticity mode](#)

[Configuring multi-cell performance management](#)

Related reference:

[Scripts](#)

[Custom properties](#)

Configuring elasticity mode

Configure elasticity mode to add logic that causes the application placement controller to minimize the number of nodes that are used, as well as remove nodes that are not needed, while still meeting service policy goals. Additionally, you can configure elasticity mode to add logic so that when the controller recognizes a particular dynamic cluster is not meeting service policies and has started all possible servers, the controller adds a node.

Before you begin

- For optimal performance, ensure that your dynamic clusters are running in supervised mode or automatic mode. It is not recommended to have elasticity mode enabled when your dynamic clusters are running in manual mode. However, if the dynamic clusters are running in manual mode with elasticity enabled, consider the following items:
 - The application placement controller does not add nodes to dynamic clusters in manual mode.
 - The application placement controller does not remove nodes from dynamic clusters in manual mode when a server is started on the specific nodes.
 - The application placement controller does remove nodes from dynamic clusters in manual mode when a server is not started on the specific nodes.
- It is not recommended to enable elasticity mode when the following option is set in the administrative console for one or more dynamic clusters:

If other dynamic clusters need resources, stop all instances of this cluster during periods of inactivity.

If you have elasticity mode enabled and the option set, the application placement controller can remove all of the custom nodes in the cell.

Procedure

1. Set the `HAManagedItemPreferred_apc` custom property to configure the application placement controller to start on a deployment manager or a node that will not be removed.

Complete the following steps to configure the application placement controller to start on the deployment manager:

- a. In the administrative console, select **System administration > Deployment manager > Java and process management > Process definition > Java virtual machine > Custom properties**.
- b. Enter the name of the custom property as `HAManagedItemPreferred_apc`.
- c. Set the value of the custom property to `true`.
- d. Click **Apply**, and save your changes.
- e. Restart the current process in which the application placement controller is running.

Complete the following steps to configure the application placement controller to start on the node:

- a. In the administrative console, select **System administration > Nodes > node_name > node_agent_name > Java and process management > Process definition > Java virtual machine > Custom properties**.
 - b. Enter the name of the custom property as `HAManagedItemPreferred_apc`.
 - c. Set the value of the custom property to `true`.
 - d. Click **Apply**, and save your changes.
 - e. Restart the current process in which the application placement controller is running.
2. Set the `HAManagedItemPreferred_apc` and `HAManagedItemPreferred_cellagent` custom properties to configure the controllers to start on the deployment managers of the center cell and the point cells when you enable elasticity mode in an environment in which multi-cell performance management is configured.

Complete the following steps to configure the application placement controller to start on the deployment manager of the center cell:

- a. In the administrative console, select **System administration > Deployment manager > Java and process management > Process definition > Java virtual machine > Custom properties**.
- b. Enter the name of the custom property as `HAManagedItemPreferred_apc`.
- c. Set the value of the custom property to `true`.
- d. Click **Apply**, and save your changes.
- e. Restart the current process in which the application placement controller is running.

Complete the following steps to configure the cell agent to start on the deployment managers of the point cells:

- a. In the administrative console, select **System administration > Deployment manager > Java and process management > Process definition > Java virtual machine > Custom properties**.
 - b. Enter the name of the custom property as `HAManagedItemPreferred_cellagent`.
 - c. Set the value of the custom property to `true`.
 - d. Click **Apply**, and save your changes.
 - e. Restart the current process in which the application placement controller is running.
3. Enable elasticity mode. Select **Operational policies > Autonomic controllers > Application placement controller**.
 - a. Select the **Enable elasticity operation** check box.
 - b. Set elasticity to either supervised or automatic mode.
 - c. Set the timeout value. This value defines the allotted period of time that the application placement controller allows the task to complete. If the task does not complete within the specified amount of time, the application placement controller marks the task as failed. Specify the value in minutes. The default value is 60.
 - d. Click **Apply**, and save your changes.
 4. Define the actions of the elasticity operations. An elasticity operation defines the runtime behaviors to monitor, and the corrective actions to take when the behaviors are present. Select **Operational policies > Autonomic controllers > Application placement controller > Elasticity operation > operation..**
 - a. To add additional actions to the Add operation, click **Add Action...**. Select the custom action from the list of Custom elasticity operation actions. If no custom actions are defined, select **Operational policies > Autonomic controllers > Application placement controller > Elasticity Custom Actions > New**.
 - b. To add additional actions to the Remove operation, click **Add Action...**. Select the custom action from the list of Custom elasticity operation actions. If no custom actions are defined, select **Operational policies > Autonomic controllers > Application placement controller > Elasticity Custom Actions > New**.

Elasticity mode

Use elasticity mode to add logic that causes the application placement controller to minimize the number of nodes that are used, as well as remove nodes that are not needed, while still meeting service policy goals. Additionally, you can use elasticity mode to add logic so that when the controller recognizes a particular dynamic cluster is not meeting service policies and has started all possible servers, the controller adds a node.

Parent topic: [Configuring dynamic application placement](#)

Related tasks:

[Configuring multi-cell performance management](#)
[Creating dynamic clusters](#)

Related reference:

[Custom properties](#)

Monitoring and tuning the application placement controller

The application placement controller is designed to work with the default settings. However, there can be times when fine tuning the application placement controller becomes necessary to ensure the best results.

Before you begin

Before you can tune application placement, you need to have dynamic application placement running. See [Configuring dynamic application placement](#) to enable application placement.

Depending on your administrative role, you are allowed the following specific privileges when you are configuring the application placement controller:

- **Monitor:** Can view the information.
- **Operator:** Can view the information on the configuration tab. Can change the settings on the runtime tab.
- **Configurator:** Can change the configuration but not the runtime settings.
- **Administrator:** Has all privileges.

About this task

You might tune the application placement controller for the following reasons:

- To adjust the amount of time before runtime tasks expire
- To adjust the amount of time that passes before starting or stopping a server is considered a failure
- To adjust the amount of time between placement changes for applications

Procedure

1. In the administrative console, click **Operational policies > Autonomic controllers > Application placement controller**.
2. Determine whether you want to modify the persistent settings in the configuration tab, which are permanent between stops and starts of the application placement controller, or test runtime settings in the runtime tab, which last for the life of the application placement controller. You can choose to modify the runtime settings and then find those settings are actually what you want to use as permanent settings. You can make runtime settings persistent settings by clicking **Save to repository**.
3. Modify the application placement controller settings, as needed.

The table following these steps contains the general configuration settings.

4. Define custom properties to change placement behavior.

You can specify the [reservedMemoryFixed and reservedMemoryPercent custom properties](#) to define how much node memory to reserve for processes that are not related to WebSphere Application Server or WebSphere Virtual Enterprise.

For information about custom properties that you can set on the application placement controller, see [Application placement custom properties](#).

5. Click **Apply** or **OK**. Save your changes.
6. Iterate and modify the settings as needed. Make changes until you discover the best possible configuration for your environment.

Results

Table 1. General configuration settings

Setting	Description
Enable	Enables or disables the application placement controller. If you disable the application placement controller, you are disabling all autonomic operations for the dynamic clusters. After the placement controller is disabled, no dynamic changes occur with regard to the size and placement of applications on the dynamic cluster. This action is equivalent to turning all dynamic clusters to static clusters.
Approval timeout	Controls how long the runtime task sits in the queue awaiting action before it expires automatically. Acceptable time values are 1 to 60 minutes. This setting is particular to the supervised operating mode. When the operating environment is set to run in supervised mode, the application placement controller creates tasks, but must wait for approval from the system administrator before making changes. The application placement controller treats runtime tasks that are timed out as user-rejected tasks.

Server operation timeout

Represents the amount of time, in minutes, that the application placement controller waits for a start or stop operation to complete before the operation is considered a failure.

Set this value to the predicted worst case time to start or stop a server.

Note: When the completion time for an operation and the server operation timeout exceed the amount of time configured for the request timeout, increase the request timeout to a value greater than the value for the server operation timeout.

1. In the administrative console, click **System administration > Node agents > *node_agent_name* > Administration Services > JMX Connectors > SOAPConnector > Custom properties > requestTimeout**.
2. Provide a value greater than the value for the server operation timeout. For example, if the server operation timeout is 30 minutes, set the value to 1920 (32 minutes).
3. Click **OK**.

Acceptable values are between 1 to 60 minutes. If a server fails to start before the timeout, the server is put into maintenance mode. The application placement controller avoids trying to start that server again until you manually fix the start issue and manually start the server. When the server starts, the application placement controller detects that the server has started.

Servers are not automatically placed in maintenance mode when the servers fail to start before the server operation timeout. Instead, a runtime task is generated when the server fails to start. The application placement controller does not attempt to start this server again until the server is successfully started manually. When you successfully start the server manually, the application placement controller receives a notification that the server has started.

If you want the servers to be put into maintenance mode, you can set the [maintenanceModeOnOperationFail](#) custom property on the application placement controller.

If you want to take servers out of maintenance mode after starting successfully, you can specify the [unsetMaintenanceModeAfterStart](#) custom property on the application placement controller.

Minimum time between placement change

Specifies the amount of time that the application placement controller waits before initiating a new batch of changes. The application placement controller might wait for a batch of changes after completing previous changes, or encountering a timeout. Acceptable values can range from 1 minute to 24 hours.

When setting this value, consider the overhead that is associated with starting and stopping servers. Starting or stopping servers can take several minutes and might introduce an additional load to the nodes.

If you allow the placement controllers to readjust application placement too often, the added overhead negates the increased performance gains that can be earned by readjusting the size of dynamic clusters. For example, if a server takes one minute to start, and the setting for the minimum time between placement changes is 20 minutes, then placement changes have a performance impact of approximately 5%.

Set this value at least 20 to 30 times larger than the time necessary to start a server. A value greater than several hours prevents application placement changes from happening more than once a day. If you believe that traffic load and application demands adjust several times during a day, you might want to allow placement changes to occur more frequently.

|

The application placement controller is tuned for its best performance.

Parent topic: [Configuring dynamic application placement](#)

Related concepts:

[Overview of application placement](#)

Related tasks:

[Configuring dynamic application placement](#)

Related reference:

[Administrative roles and privileges](#)

[PlacementControllerProcs.jacl script](#)

Related information:

[Application placement custom properties](#)

Application placement frequently asked questions

Occasionally, you might encounter application placement behavior that is not expected. This topic describes some commonly asked questions and things to look for when application placement is not working as you expect.

Where is the application placement controller running?

To find where the application placement controller is running, you can use the administrative console or scripting. To check the location in the administrative console, click **Runtime Operations > Extended Deployment > Core components**. You also run the checkPlacementLocation.jacl script to display the server on which the application placement controller is running.

When does the application placement controller start a server?

The application placement controller starts servers for the following reasons:

- To meet the minimum number of application instances that are defined for the dynamic cluster.
- When a request routed through the on demand router for a deactivated dynamic cluster.
- When a dynamic cluster could benefit from additional capacity. The autonomic request flow manager sends a signal indicating how beneficial it would be for a dynamic cluster to have more additional capacity, and additional instances start for the dynamic cluster.

For a view of what is running in view of the application placement controller, see the SystemOut.log messages.

When does the application placement controller stop a server?

The application placement controller stops a server for the following reasons:

- A memory constraint on a node exists. The application placement controller understands the minima for the dynamic cluster, or the amount of capacity needed for that dynamic cluster and the processor constraints and memory constraints of the system. If available memory becomes low on a node, the application placement controller attempts to stop instances to try and prevent the node from swapping.
- The dynamic cluster is configured for application lazy start and proactive idle stop, and no demand exists for the dynamic cluster. If the dynamic cluster has no demand, the application placement controller attempts to stop instances of that cluster to eliminate the resource consumption of an inactive dynamic cluster.

Why didn't the application placement controller start a server?

The application placement controller might not display that the server is started for one of the following reasons:

- **The configuration has not enabled dynamic application placement**
 1. Verify that the placement controller is enabled. In the administrative console, click **Operational policies > Autonomic managers > Application placement controller**.
 2. Verify that the subject cluster or clusters are dynamic clusters. The application placement controller acts upon dynamic clusters only. In the administrative console, click **Servers > Dynamic clusters**. Check that the **Operational Mode** field for each of the subject clusters is **Automatic**. If not, select the dynamic clusters and click **Automatic**. After you select automatic for your dynamic clusters, click **Set Mode**.
 3. Verify that the configured minimum time between placement change parameters is not set too high. In the administrative console, click **Operational policies > Autonomic managers > Application placement controller**. Set the value in the **Minimum time between placement changes** field to a suitable value. Acceptable values range from 1 minute to 24 hours.
- **Server operation timeout value set too low**

Sometimes the application placement controller does not start a server because the server operation times out. You can configure the amount of time before a timeout occurs in the administrative console. Click **Operational policies > Autonomic managers > Application placement controller**. Edit the **Server operation timeout** field. If your cell is large, your system is slow, or your system is under high workload, set this field to a higher value. This value represents the amount of time for each server to start, but the timeout occurs based on the number of servers in your cell. For example, if you have 5 servers and set the value to 10 minutes, then a timeout occurs after 50 minutes.

- **Not enough memory available**

You can diagnose when not enough memory is available by looking at the failed starts in the SystemOut.log file.

The application placement controller uses the following formulas to calculate the memory consumption of a dynamic cluster member:

If no other dynamic cluster instances are running (cold start):

```
Server memory consumption = 1.2*maxHeapSize + 64 MB
```

If other dynamic cluster instances are running, the application placement controller memory profiler uses the following formula:

```
Server memory consumption = .667*resident memory size + .333*virtual
```

memory size

Memory profiles are not persisted when the application placement controller restarts. If you want to debug, you can disable the application placement controller memory profiler by setting the `memoryProfile.isDisabled` custom property to `true`.

Viewing failed start information

Remember: The failed start list is not persisted when the application placement controller restarts or moves between nodes.

You can view failed start information with one of the following options:

- **Use the `PlacementControllerProcs.jacl` script to query failed server operations.**

Run the following command:

```
./wsadmin.sh -profile PlacementControllerProcs.jacl -c  
"anyFailedServerOperations"
```

- **Use commands in the `wsadmin` tool to display failed starts.**

For example, you might run the following commands:

```
wsadmin>apc =  
AdminControl.queryNames('WebSphere:type=PlacementControllerMBean,process=  
dmgr,*')  
wsadmin>print AdminControl.invoke(apc,'anyFailedServerOperations')
```

When the server becomes available, the failed to start flag is removed. You can use the following `wsadmin` tool command to list the servers that have the failed to start flag enabled:

```
wsadmin>print AdminControl.invoke(apc,'anyFailedServerOperations')  
OpsManTestCell/xdblade09b09/DC1_xdblade09b09
```

- **View the failed starts in the `SystemOut.log` file.**

Why did the application placement controller start more servers than I expected?

More servers can start than expected when network or communication issues prevent the application placement controller from receiving confirmation that a server has started. When the application placement controller does not receive confirmation, it might start an additional server.

How do I know when the application placement controller has completed an action or is going to complete an action?

You can check the actions of the application placement controller with runtime tasks. To view the runtime tasks, click **System administration > Task management > Runtime tasks**. The list of runtime tasks includes tasks that the application placement controller is completing, and confirmation that changes were made. Each runtime task has a status of succeeded, failed, or unknown. An unknown status means that there was no confirmation either way whether the task was successful.

How does the application placement controller work with VMware, which hardware virtualization environments are supported?

For more information about how the application placement controller works with VMware and other hardware virtualization environments, see [Virtualization and WebSphere Virtual Enterprise](#) and [Supported server virtualization environments](#).

How can I start or stop a server without interfering with the application placement controller?

If you start or stop a server while the dynamic cluster is in automatic mode, the application placement controller might decide to make changes to your actions. To avoid interfering with the application placement controller when you start or stop a server, put the dynamic cluster into manual mode before you start or stop a server.

In a heterogeneous system (mixed hardware or operating systems), how does the application placement controller pick where it is going to start a server?

The membership policy for a dynamic cluster defines the eligible nodes on which servers can start. From this set of nodes, the application placement controller selects a node on which to start a server by considering system constraints such as available processor and memory capacity. The application placement controller does not make any decisions about server placement based on operating systems.

When my dynamic cluster is under load, when does the application placement controller start another server?

The application placement controller works with the autonomic request flow manager (ARFM) and defined

service policies to determine when to start servers. Service policies set the performance and priority maximums for applications and guide the autonomic controllers in traffic shaping and capacity provisioning decisions. Service policy goals indirectly influence the actions that are taken by the application placement controller. The application placement controller provisions more servers based on information from the ARFM about how much capacity is required for the number of concurrent requests that are being serviced by the ARFM queues. This number is determined based on how much capacity each request uses when it is serviced and how many concurrent requests ARFM determines is appropriate. The number of concurrent requests is based on application priority, goal, and so on.

The performance goals that are defined by service policies are not guarantees. WebSphere Virtual Enterprise cannot make your application respond faster than its limit. In addition, more capacity is not provisioned if enough capacity is already provisioned to meet the demand, even if the service policy goal is being breached. WebSphere Virtual Enterprise can prevent unrealistic service policy goals from introducing instability into the environment.

How does the application placement controller determine the maximum heap size of my server?

You can change the heap size of the server in the dynamic cluster template. See [Modifying the JVM heap size](#) for more information.

How does the application placement controller work with Compute Grid in particular, dispatching jobs and endpoint selection?

When the application placement controller and Compute Grid are configured to work together, Compute Grid delegates the task of endpoint selection to the placement controller. When jobs are submitted, Compute Grid asks the placement controller to select an endpoint on which to run the job. The information that Compute Grid includes in this request to the application placement controller includes the class of the job, the completion time goal, and the nodes, clusters, or servers where the job is permitted to run. When the application placement controller has selected an endpoint, that information is passed back to Compute Grid, which is responsible for starting the job. When the application placement controller selects the endpoint, it considers the completion time goal for the job, available resources in the system, the execution profile of previous jobs in the same class, and other work, including both transactional and batch work, that the system needs to handle. The application placement controller attempts to select an endpoint in so that the job has enough resources to complete on or before its completion time goal. Because of the selection process that must occur, the application placement controller does not always immediately select an endpoint for the job. In this situation, the job does not start immediately. This delay is likely if the system is small, has other transactional or batch work, and the completion time goal for the job is long.

How does the application placement controller work with WebSphere eXtreme Scale?

The application placement controller integrates with WebSphere eXtreme Scale. In particular, before stopping a dynamic cluster member that contains a container server, the application placement controller contacts the container server to quiesce any work that is being done, and moves any vital data to backup locations as needed.

The catalog service automatically starts in the configured deployment manager. If you want to configure failover for the catalog service, you can create a catalog service grid. See [Starting the catalog service process in a WebSphere Application Server environment](#) for more information.

Application placement also recognizes the replication zone concept of WebSphere eXtreme Scale. The application placement controller ensures that when multiple instances of a dynamic cluster are needed, the instances are placed across multiple replication zones. For this functionality, the dynamic cluster must be mapped to multiple replication zone node groups. See [Using zones for replica placement](#) for more information.

Why are the dynamic cluster members not inheriting properties from the template?

You must save dynamic clusters to the master repository before making changes to the server template. If you have dynamic cluster members that do not inherit the properties from the template, the server template probably incurred changes in an unsaved workspace. To fix this issue, delete the dynamic cluster, then recreate it.

Save your changes to the master repository. You can ensure that your changes are saved to the master repository after clicking **Finish**, by clicking **Save** in the message window in the top frame. Click **Save** again in the **Save to Master Configuration** window. Click **Synchronize changes with nodes**.

Why does my dynamic cluster have too few active servers?

If you encounter problems where not enough servers are running in the dynamic cluster, try the following actions:

- When the nodes in the node group are not highly utilized, verify the service policy is met. At times the policy might not be defined clearly and although the system is able to meet them, although not to your expectations. To check or change a service policy in the administrative console click **Operational policies > Service policies > Select an existing policy**. Check the goal type, goal value, and importance of the policy, and make any necessary changes.
- When the nodes in the node group are highly utilized, compare the service policy goals of this cluster to service policy goals of other active clusters. If the traffic that belongs to this cluster has lower importance or looser target service goals relative to the other clusters, it is more likely that the system instantiates fewer servers for this cluster. To check or change a service policy in the administrative console, click

Operational policies > Service policies > *Select an existing policy.*

- When the node group seems to have some extra capacity, but your service policies are not met, check the configuration settings on the dynamic cluster. There might be too few instances of the dynamic cluster created as a result of the maxInstances policy setting.

Parent topic: [Configuring dynamic application placement](#)

Related tasks:

[Configuring dynamic application placement](#)

Related reference:

[Administrative roles and privileges](#)

[checkPlacementLocation.jacl script](#)

Related information:

[Application placement custom properties](#)

Application placement controller logs

Use the log files to review the status of the application placement controller.

SystemOut.log file

Look for DCPC messages in the SystemOut.log file of the node that runs the application placement controller. These messages contain basic information about runtime tasks, dynamic cluster configuration changes, and changes in the state of the servers in the topology.

apc.log file

The application placement controller logs information about application placement decisions to the apc.log file. This log file is created on the node that runs the application placement controller. The file contains information about the input and output of a placement cycle. This information helps you understanding the decisions that are made by the placement controller.

Use the apc.log.enablePlacementLog custom property to enable the application placement controller logs. To change the size and the number of log files, edit the apc.log.LogFileSize and apc.log.numHistoricalFiles custom properties.

apc.log example:

```
DCPC0331I: The application placement controller Input is
(Map Time=1360347071465)

Cluster Name: xdhv01v04Cell02/pc1dc1 (Input) (changesAreNecessary) Fri Feb 08
13:11:26 EST 2013
  Cluster Properties: Type=DYNAMIC, Operational Mode=AUTOMATIC, Min/Max
Instances=1/2147483647, Min/Max Nodes=1/2147483647, Isolation Preference=None,
Vertical Stacking Number=Disabled, Lazy Start Time=Disabled, Idle Stop
Time=Disabled, Current Memory Size=475.0 MB, Maximum Heap Size=256.0 MB
  Cluster Resource Demand:
    CPU(mhz)= 0.0
    Memory(mb)= 0.0(maximal)
  Cluster Utility: {UtilityFunctionAlwaysOne}
  Cluster Node Membership:
    xdhv01v04Cell02/xdhv01v04Node03  Running: false  Maintenance
Mode: false
    xdhv01v04Cell02/xdhv01v05Node02  Running: false  Maintenance
Mode: false
  Cluster Members:
    xdhv01v04Cell02/xdhv01v04Node03/pc1dc1_xdhv01v04Node03
Running: false  Maintenance Mode: false
    xdhv01v04Cell02/xdhv01v05Node02/pc1dc1_xdhv01v05Node02
Running: false  Maintenance Mode: false
  Cluster Placements:
    None
Node Capacity and Demands:
  Background CPU(mhz)=0.0  COP=0.9  Node Memory(mb)=7740.0  Fixed
Memory(mb)=65.536  MOP=0.05
  P:xdhv01v04.rtp.raleigh.ibm.com:  Capacity(Total/Available):
CPU(mhz)=5866/5866  Memory(mb)=7287/6465
    xdhv01v04Cell02/xdhv01v04CellManager02/dmgr  Demand:
CPU(mhz)= 0  Memory(mb)= 371
    xdhv01v04Cell02/xdhv01v04Node03/nodeagent  Demand:
CPU(mhz)= 0  Memory(mb)= 451

Cluster Resource Allocation:
  /cell4/pc4dc1 CPU(mhz)= 0  Desired
Failed Server Starts:
None
```

```
[2/8/2013 13:11:26:653 EDT] DCPC0331I: The application placement controller
Output is
(Map Time=1360347071465)
DCPC0335I: The Application Placement Controller is suggesting to start a
server for xdhv01v04Cell02/pc1dc1 to meet the required number of running
instances.
```

This data is printed each time APC completes a placement cycle. Use the following cluster information to review and diagnose issues in the topology:

- **Cluster's properties:** Displays cluster configuration information.
- **Cluster's Demands:** Displays the cluster demands on processor and memory.
- **Cluster's Utility:** Displays the cluster capacity calculation.
- **Cluster's Node Membership:** Displays the nodes on which a cluster member can reside, and the related state. For example: Is the node running or is it in maintenance mode.
- **Cluster's Members:** Displays the members of the cluster and the related state.
- **Cluster's Placements:** Displays the running members of the cluster.

In addition to cluster data, `apc.log` also contains useful node information:

- **Background CPU:** The amount of CPU consumed on a node that is not related to the WebSphere® application server.
- **COP:** A value 0 - 1 that represents the percentage of node CPU that can be used. This value corresponds to the CPU overload protection value from ARFM.
- **Node Memory:** The total amount of memory on a node.
- **Fixed Memory:** The fixed amount of node memory not available for use by a WebSphere application server. By default, the APC custom property that controls this value is set to `APC.XD.reservedMemoryFixed=65.536`.
- **MOP:** The percentage of node memory not available for use by a WebSphere application server. By default, the APC custom property that controls this value is set to `APC.XD.reservedMemoryPercent=5`.

The application placement controller prints input and output information each time it runs a processing cycle. The input displays the current cluster and node information, and the output displays any actions suggested by APC. In the previous example, APC suggests starting a server for the cluster `xdhv01v04Cell02/pc1dc1` to meet the minimum instances requirement. To confirm this suggestion, review the input data: minimum instances is 1 (Min/Max Instances=1/2147483647), and there are currently no running servers (Cluster Placements: None).

Related information:

[Application placement custom properties](#)

Overview of IIOP and JMS request flow prioritization

Internet Inter-ORB Protocol (IIOP) and Java™ Message Service (JMS) request flow prioritization is achieved in WebSphere® Virtual Enterprise with the autonomic managers that control the flow of requests, prioritization of requests, and dynamic workload management. Only IIOP requests from a stand-alone Enterprise JavaBeans (EJB) client are handled by WebSphere Virtual Enterprise IIOP request flow prioritization. EJB calls from servlets, for example, are not prioritized. This limitation exists because the system cannot prioritize at multiple tiers, such as the Web tier and EJB tier, requests associated with the same overall user request. However, given the asynchronous nature of JMS, no restrictions exist on where the requests originate.

For IIOP and JMS, the back-end application server processes that are hosting the applications run the autonomic request flow manager (ARFM) gateways. These ARFM gateways prioritize request flow. The request flows are managed to achieve the best balanced performance results, considering the configured service policies and the offered load.

With WebSphere Virtual Enterprise, you can define performance goals and bind them to specific subsets of the incoming traffic. The ARFM and associated autonomic managers are able to support business goals in times of high load by making smart decisions about the work coming into the application servers. Not all work in your configuration is created equal. The ARFM is able to support this concept by forwarding different flows of requests on for execution more or less quickly to achieve the best balanced result.

The ARFM is aware of its environment because of a component called an on demand configuration (ODC). The ODC automatically gets information about all the WebSphere Virtual Enterprise application servers and applications that are deployed in the cell, and any service policy and work classes associated with those configuration artifacts.

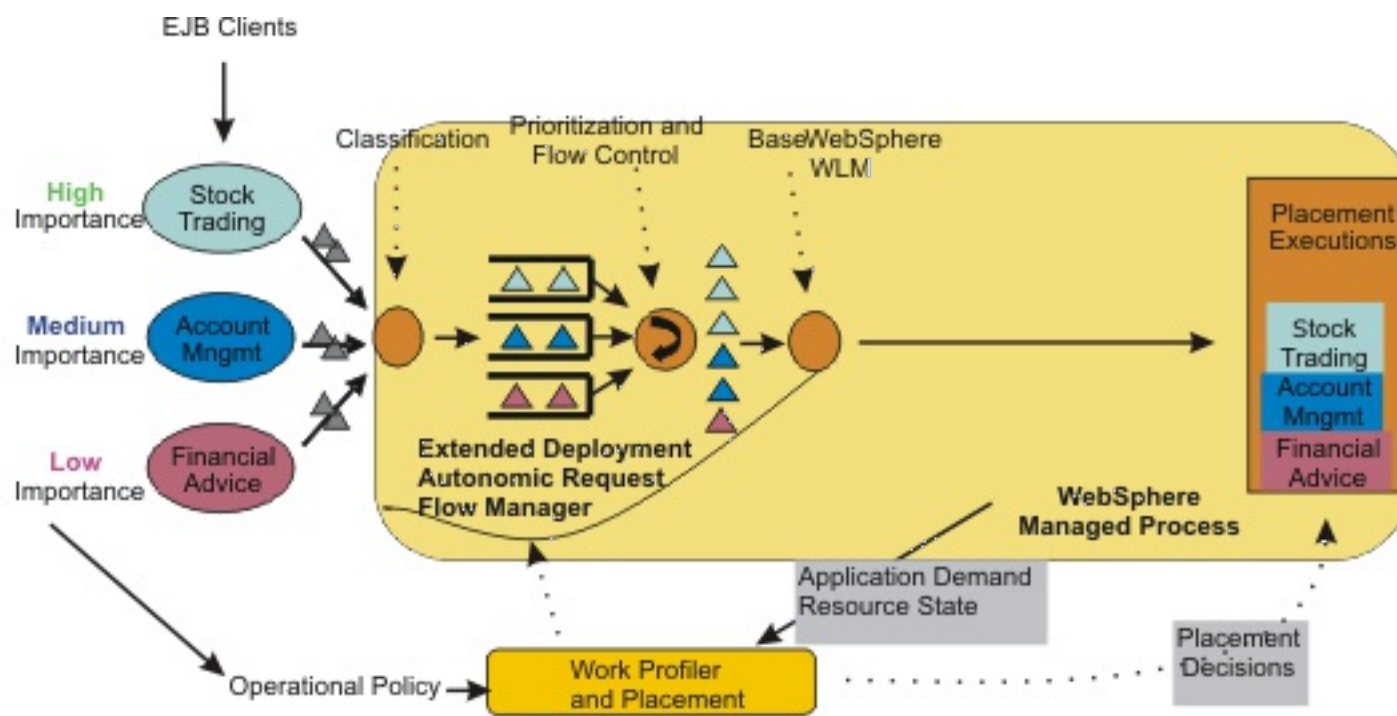
Note: The ODC cannot read environments other than a homogeneous WebSphere Virtual Enterprise environment.

A service policy is a user-defined categorization that is assigned to potential work as an attribute that is read by the ARFM. For IIOP, you can use a service policy to classify requests based on request attributes, including the application name, EJB method name, EJB module name, such as the EJB jar file, and the EJB name. For JMS, you can classify based on destination name, either topics or queues. By configuring service policies, you apply varying levels of importance to the actual work. You can use multiple service policies to deliver differentiated services to different categories of requests. Service policy goals can differ in performance targets and importance.

The ARFM exists in the application server process and controls request prioritization. The autonomic request flow manager contains two parts: a controller and a gateway. The ARFM function is implemented, for each cell, by a controller plus a collection of gateways in the application servers. The gateways intercept and queue the incoming IIOP requests, while the controller provides control signals, or directions, to the gateways, and the placement controller. The ARFM also includes the work profiler, which estimates the computational load characteristics of the different flows of requests. Working together, these components can properly prioritize incoming requests.

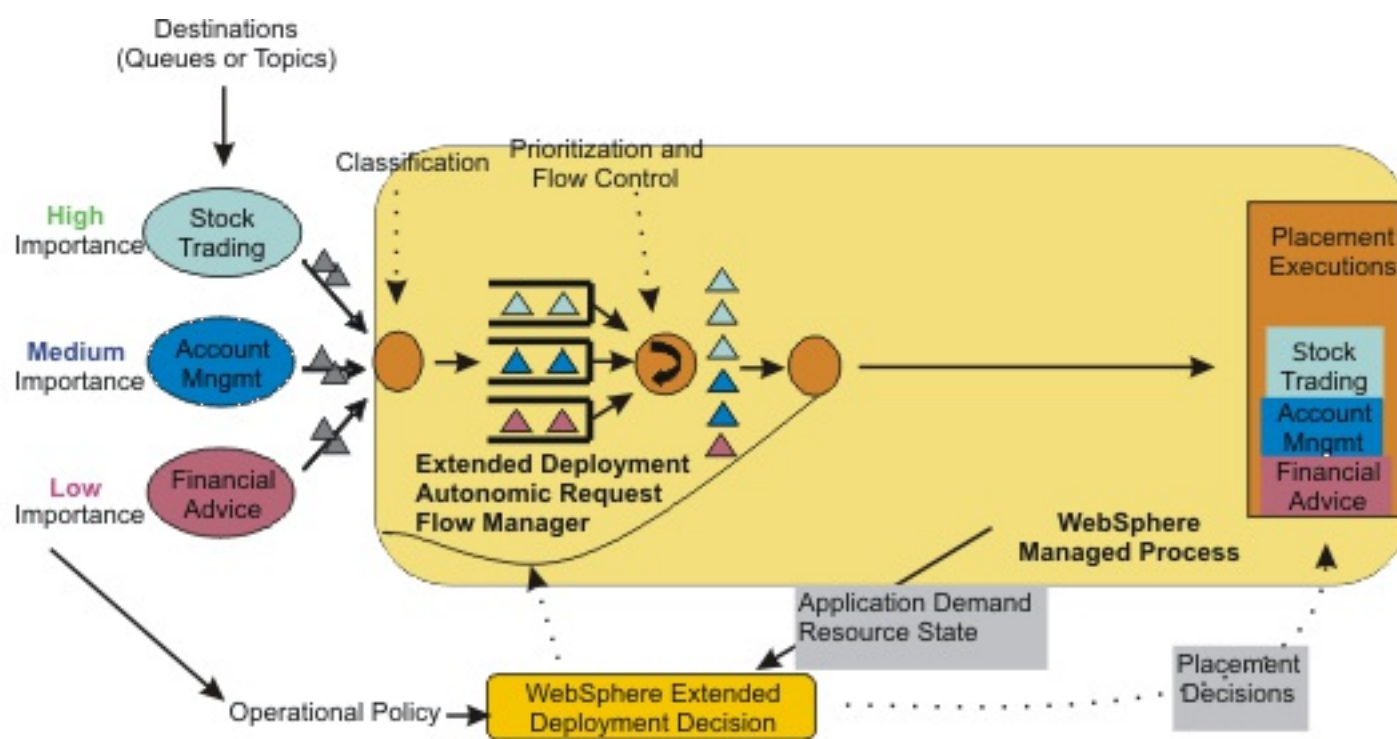
Dynamic workload management (DWLM) is a feature that applies the same principles as workload management (WLM), such as routing based on a weight system, which establishes a prioritized routing system. DWLM is an optional add-on that adds autonomic setting of the routing weights to WLM. With WLM, manually set static weights in the administrative console. With DWLM, the system can dynamically modify the weights to stay current with the business goals. DWLM can be shut off. If you intend to use the automatic operating modes for the components of dynamic operations, then setting a static WLM weight on any of your dynamic clusters might get in the way of allowing the on demand aspect of the product to function properly. For IIOP, these weights are consumed by base WebSphere EJB WLM and factor to where new EJB client requests are directed, as shown in the following diagram:

Figure 1. IIOP flow



DWLM does not influence JMS traffic. The destinations shown in the following figure can be running in the same WebSphere Managed process or a different WebSphere Managed process.

Figure 2. JMS flow



As the previous diagrams depicts, an equal amount of requests flow into the application server, but after the work is categorized, prioritized, and queued, a higher volume of more important Platinum work is sent to be processed while a lower volume of less important Bronze work waits to get queued. However, just because the lower priority work is delayed the most, that does not make the long-term average rate of Bronze work that runs in the application server less than the long-term average rate of Bronze requests that come in. In the end, the features of the dynamic operations attempt to keep the work within the target time allotted for completion.

Overview of JMS support in WebSphere Virtual Enterprise

WebSphere Virtual Enterprise extends the dynamic operations environment to JMS by providing the capability to categorize and prioritize JMS messages. JMS messages can be classified by Destination, that is, topic or queue. Service policies can be defined to prioritize messages differently based on the destination name.

Parent topic: [Configuring the autonomic managers](#)

Related tasks:

[Defining a service policy](#)

Overview of JMS support in WebSphere Virtual Enterprise

WebSphere® Virtual Enterprise extends the dynamic operations environment to JMS by providing the capability to categorize and prioritize JMS messages. JMS messages can be classified by Destination, that is, topic or queue. Service policies can be defined to prioritize messages differently based on the destination name.

While JMS messages can be consumed by standalone clients or enterprise applications in a client container or enterprise applications running in an application server, WebSphere Virtual Enterprise *only* supports classification and flow control of messages for enterprise applications inside application server. For JMS, autonomic request flow manager gateways runs in the application server hosting the application, which means that JMS traffic is not handled by the ODR server.

For Message Driven Beans, service policy goals are based on the *service times* of the onMessage method. For synchronous JMS message consumers, service time is the interval between time of message delivery to the application and the application returning for the next message on the same consumer session.

Restriction: Routing policies including application edition based routing are not supported for JMS. Classification and flow control of JMS messages are only supported only when using the default message provider of WebSphere Application Server. A given destination is allowed to be part of only one JMS work class.

Parent topic: [Overview of IOP and JMS request flow prioritization](#)

Related concepts:

[Overview of IOP and JMS request flow prioritization](#)

Related tasks:

[Deploying applications with defined service levels](#)

Optimizing the database tier for performance monitoring

You can configure WebSphere® Virtual Enterprise to provide real time information about the database tier, including central processing unit utilization, average response time, and throughput. You can access this information through the runtime topology in order to make adjustments when bottlenecks occur.

About this task

Bottlenecks can occur in the database tier and cause a slower response time. If there is a problem in the database tier, WebSphere Virtual Enterprise can provide you with the information you need to identify and address the problem.

Procedure

1. [Configuring the database tier](#)

Follow these steps to set up WebSphere Virtual Enterprise for monitoring the database tier.

2. [Monitoring the database tier](#)

After configuring the database tier, you can view real time information in order to detect and prevent bottlenecks.

[Configuring the database tier](#)

To access useful information about the database tier, you must first configure WebSphere Virtual Enterprise for performance monitoring. With database tier performance monitoring, you can view central processing unit utilization, average response time, and throughput. The information provided in performance monitoring is valuable for detecting and preventing bottlenecks.

[Monitoring the database tier](#)

To prevent bottlenecks, WebSphere Virtual Enterprise has several tools that provide the information necessary to monitor the database tier. Use the following list of monitoring tools and links to configure WebSphere Virtual Enterprise to fit your needs.

Parent topic: [Configuring the autonomic managers](#)

Parent topic: [Configuring the database tier](#)

Parent topic: [Monitoring the database tier](#)

Parent topic: [Configuring the autonomic managers](#)

Configuring the database tier

To access useful information about the database tier, you must first configure WebSphere® Virtual Enterprise for performance monitoring. With database tier performance monitoring, you can view central processing unit utilization, average response time, and throughput. The information provided in performance monitoring is valuable for detecting and preventing bottlenecks.

Before you begin

Ensure that a DB2® or Oracle database is installed on the node, and WebSphere Virtual Enterprise is installed with an augmented profile. See ./install/tprofilechange.html for more information.

About this task

After you have installed either a DB2 or Oracle database on the node and installed WebSphere Virtual Enterprise with an augmented profile, you can then configure the database tier for performance monitoring. Monitoring the database tier provides the information needed to prevent the system from experiencing slower response time due to bottlenecks.

Procedure

1. Create a database server in the WebSphere Application Server configuration.
 - a. Create a new generic server and give it a name, for example DB2Instance1. Select the database node from the list.
 - b. Click **Generic Servers > DB2Instance1**.
 - c. Under **Additional Properties**, click **Process Definition > Java™ Virtual Machine > Custom Properties**.
 - d. Add the following entries:

```
GENERIC_SERVER_TYPE = DBSERVER
GENERIC_SERVER_SUB_TYPE = DB2 (for DB2) or ORACLE (for Oracle)
INSTANCE_NAME = name of database instance (DB2) or SID (Oracle)
```

2. Configure data sources for Java Database Connectivity (JDBC) monitoring:
 - a. Click **Resources > JMS Providers > JDBC Providers > Database_server_name > Data sources > Data_source_name > Customer properties**.

Add the property: dbServerName = *generic server name*, for example, DB2Instance1.
 - b. Follow these steps, depending on the database type and Java database connectivity (JDBC) driver type you use:
 - For type 2 DB2 data sources. If the database name on the server is different from the database alias name used on the local DB2 client, take these actions. For the **Database name** field on the Data source Configuration panel, specify the alias name. Under **Additional Properties**, click **Custom Property**. Add a remote Database Name custom property, using database name as the value on the database server.
 - For Oracle data sources: If the database name does not display in the connection URL, under **Additional Properties**, click **Custom Property**. Set the value of the database name property to the name of the Oracle database.
3. Enable performance monitoring infrastructure (PMI) for JDBC statistics collection:
 - a. Click **Monitoring and Tuning > Performance Monitoring Infrastructure (PMI)**

This action displays a list of servers in the cell.
 - b. For every server in the list that has a data source:
 - i. Click the server name.
 - ii. In the **Configuration** panel, select the **Enable Performance Monitoring Infrastructure (PMI)** check box. Set the **Currently monitored statistic set** to **Extended** or **All**. However, if the currently monitored statistic set is set to **Custom**, and for improved performance, enable the JDBC counters in the Custom Level.

Results

Your system is now ready for extended monitoring of WebSphere Virtual Enterprise components, including JDBC. You can view your database from the runtime topology.

What to do next

See [Monitoring the database tier](#) for more information.

Parent topic: [Optimizing the database tier for performance monitoring](#)

Parent topic: [Optimizing the database tier for performance monitoring](#)

Related tasks:

[Monitoring the database tier](#)

Monitoring the database tier

To prevent bottlenecks, WebSphere® Virtual Enterprise has several tools that provide the information necessary to monitor the database tier. Use the following list of monitoring tools and links to configure WebSphere Virtual Enterprise to fit your needs.

Before you begin

Configure the database tier. See [Configuring the database tier](#) for more information.

About this task

Before a bottleneck occurs, WebSphere Virtual Enterprise can help you anticipate potential problems and adjust the system accordingly. Customize your system using the following information.

Procedure

- Enable historic logging of statistical information in the environment.
See [Configuring the visualization data service](#) for more information.
- Evaluate historical information by using the data exporter to filter out historical logs or parse logs directly.
See [Configuring the operations reports to log data for reuse in external programs](#) for more information.
- View CPU utilization at the database node level, and view average response time and throughput at the database server level.
See [Creating and managing reports](#) for more information.
- Enable automatic e-mail notifications for new tasks.
See [Defining email notification](#) for more information.
- Validate incoming tasks by viewing the details and the corresponding monitors to determine if the task is valid.
See [Managing runtime tasks](#) for more information.
- Use the topology and charting feature when a problem is detected or to validate a task.
See [Monitoring operations](#) for more information.

Parent topic: [Optimizing the database tier for performance monitoring](#)

Parent topic: [Optimizing the database tier for performance monitoring](#)

Related reference:

[Logs](#)

Configuring health management

Health management is the process by which the dynamic operations of WebSphere® Virtual Enterprise monitor and manage servers that might fail, to preserve an optimal server environment.

Before you begin

Create the environment that you want to monitor. When you create a health policy, you select members for the policy, so these members must exist before you create the policy.

About this task

Health management is the ability of the system to take a policy-driven approach to monitoring the application server environment and to take action when certain criteria are discovered.

[Enabling and disabling health management](#)

Health management monitoring is enabled by default. You can enable or disable health management monitoring.

[Health management](#)

With the health monitoring and management subsystem, you can take a policy-driven approach to monitoring the application server environment and take action when certain criteria are discovered.

[Creating health policies](#)

A health policy is the definition of specific health criteria that you want your WebSphere Virtual Enterprise to protect itself against. The health management function uses the defined policy to search the environment for software malfunctions.

Next topic: [Managing runtime tasks](#)

Parent topic: [Configuring the autonomic managers](#)

Related tasks:

[Preparing the hosting environment for dynamic operations](#)

Related reference:

[Administrative roles and privileges](#)

[Health policy administrative tasks](#)

Enabling and disabling health management

Health management monitoring is enabled by default. You can enable or disable health management monitoring.

Before you begin

The health monitor uses health policies. A health policy is a combination of health conditions and actions. Health conditions define triggers from which the system can protect itself, for example, a memory leak. Health actions are the specific steps that the system takes when a health condition is triggered. For example, with a memory leak condition, you might choose to restart the associated servers as the health action. A number of default health policies are installed with WebSphere® Virtual Enterprise. In addition to these broad default health policies, you can define more specific policies that apply for your environment. For more information about the default health policies, see [Health management](#). For more information about creating a health policy, see [Creating health policies](#).

Depending on your administrative role, you are allowed specific privileges when configuring the autonomic managers. The following list shows the administrative roles and privileges for configuring the autonomic managers:

Monitor: Can view the information.

Operator: Can view the information on the configuration tab. Can change the settings on the runtime tab.

Configurator: Can change the configuration but not the runtime settings.

Administrator: Has all privileges.

About this task

Health management is enabled by default. Use health management to protect your system from many user application malfunctions, including memory leaks and application hangs. Health management uses health policies to define a set of conditions that WebSphere Virtual Enterprise can use to monitor the health of the system. See [Health management](#) and [Creating health policies](#) for more information.

Procedure

1. In the administrative console, click **Operational policies > Autonomic managers > Health controller**.
2. Enable or disable health monitoring. When the check box is selected, the environment is monitored for health conditions. When the check box is not selected, health monitoring is turned off.

Results

When enabled, the system is monitored for conditions that are defined by your health policies.

What to do next

Manage other health monitoring settings. See [Monitoring and tuning health management](#) and [HmControllerProcs.jacl](#) for more information.

Parent topic: [Configuring health management](#)

Related reference:

[Administrative roles and privileges](#)

Health management

With the health monitoring and management subsystem, you can take a policy-driven approach to monitoring the application server environment and take action when certain criteria are discovered.

Health monitoring and management subsystem

The health management subsystem continuously monitors the state of servers and the work that is performed by the servers in your environment. The health management subsystem consists of two main elements: the *health controller* and *health policies*.

The health controller is the autonomic manager that controls the health monitoring and management subsystem, and acts on your health policies to ensure certain conditions exist. The health controller is a distributed resource that is managed by the high availability manager and exists within all of the node agent and deployment manager processes. The health controller is active in one of these processes. If the active process fails, the health controller can become active on another node agent or deployment manager process.

The health controller runs on a *control cycle*. The control cycle length defines the amount of time that passes between the checks that the health controller runs on the environment. After the control cycle ends, the health controller checks the environment and generates runtime tasks to resolve any breaches in the health conditions.

You define the health policies, which include the *health conditions* that you want to monitor in your environment and the *health actions* to take if these conditions are not met.

You can disable or enable health management using the health controller, while still having multiple health policies defined on the system. You can also apply limits to the frequency that the server restarts or prohibit restarts during certain periods.

The health management subsystem functions when WebSphere® Virtual Enterprise is in automatic or supervised operating mode. When the reaction mode on the policy is set to automatic, the health management system takes action when a health policy violation is detected. In supervised mode, the health management system creates a runtime task that proposes one or more reactions. The system administrator can approve or deny the proposed actions.

Health conditions

Health conditions define the variables that you want to monitor in your environment. Several categories of health policy conditions exist. You can choose from the following predefined health conditions:

Age-based condition

Tracks the amount of time that the server is running. If the time exceeds the defined threshold, the health actions run.

Excessive response time condition

Tracks the amount of time that requests take to complete. If the time exceeds the defined response time threshold, the health actions run.

Attention: Requests that exceed the timeout value that is configured for the excessive request timeout condition are not counted towards this health condition. For example, if the default timeout value of 60 seconds is being used for the excessive request timeout condition, any request that exceeds 60 seconds does not activate the health actions that you have defined for the excessive response time condition. This restriction applies even if you do not have the excessive request timeout condition defined in your environment.

Excessive request timeout condition

Specifies a percentage of HTTP requests that can time out. When the percentage exceeds the defined value, the health actions run. The timeout value that is used depends on how your environment is configured. For more information about the excessive request timeout health condition, see [Excessive request timeout health policy target timeout value](#).

Memory condition: excessive memory usage

Tracks the memory usage for a member. When the memory usage exceeds a percentage of the heap size for a specified time, health actions run to correct this situation.

Memory condition: memory leak

Tracks consistent downward trends in free memory that is available to a server in the Java™ heap. When the Java heap approaches its maximum configured size, you can perform either heap dumps or server restarts.

Storm drain detection

Tracks requests that have a significantly decreased response time. This policy relies on change point detection on given time series data.

Workload condition

Specifies a number of requests that are serviced before policy members restart to clean out memory and cache data.

For more information about these conditions, click the help icon on the Health policy settings panel in the

administrative console.

With these predefined health policy conditions, actions have been taken to optimize the distribution of the needed data and minimize the impact of monitoring and enforcing the health policy on your overall environment.

You can also define custom conditions within your health policy. Use a custom condition when the predefined health conditions do not fit your needs. You define custom conditions as a subexpression that is tested against metrics in your environment. When you define a custom condition, consider the cost of collecting the data, analyzing the data, and if needed, enforcing the health policy. This cost can increase depending on the amount of traffic and number of servers in your network, so you must analyze the performance of your custom health conditions before you move them into production.

Health actions

Health actions define the process to use when a health condition is not met. Depending on the conditions that you define, the actions can vary. The following table lists the health actions that are supported in various server environments:

Table 1. Predefined health action support for different server types

Health Action	WebSphere Application Server or WebSphere Virtual Enterprise servers that run in the same cell as the health controller	Other middleware servers, including external WebSphere application servers, that run the middleware agent
Restart server	Supported	Supported
Take thread dumps	Supported for servers that are running on the IBM® Software Development Kit	Not supported
Take Java virtual machine (JVM) heap dumps	Supported for servers that are running on the IBM Software Development Kit	Not supported
Put server into maintenance mode	Supported	Supported
Put server into maintenance mode and break HTTP and SIP request affinity to the server	Supported	Supported
Take server out of maintenance mode	Supported	Supported

You can also define a custom action. With a custom action, you define an executable file to run when the health condition breaches. You must define custom actions before you create the health policy that contains the custom actions.

Health policy targets

Health policy targets can be a single server, each of the servers in a cluster or dynamic cluster, the on demand router (ODR), or each of the servers in a cell. You can define multiple health policies to monitor the same set of servers.

If you are using a predefined policy, then the support of the policy varies depending on the server type. Other middleware servers do not support all of the policy types. The following table summarizes the health policy support, by server type:

Table 2. Health policy support for different server types

Predefined Health Policy	WebSphere Application Server or WebSphere Virtual Enterprise servers that run in the same cell as the health controller	Other middleware servers, including external WebSphere application servers, that run the middleware agent
Age-based policy	Supported	Supported
Workload policy	Supported	Supported
Memory leak detection	Supported	Not supported
Excessive	Supported	Supported for WebSphere Application Server Community Edition servers.

memory usage		Not supported for other middleware server types.
Excessive request timeout	Supported	Supported for other middleware servers to which the ODR routes requests.
Excessive response time	Supported	Supported
Storm drain detection	Supported	Supported

Default health policies

Default health policies are a set of predefined supervised mode, cell-level policies that are installed with the product. You can modify the default policies for your environment, or delete the default health policies. Because the default health policies monitor each server in supervised mode, you can use these policies to prevent health problems. You can define policies with more detailed settings or automated mode operation for particular servers or collections of servers in addition to the default policies. The following list shows the five default cell-wide health policies that are created during installation:

- **Default memory leak:** Default standard detection level. The default memory leak health policy uses the performance advisor functionality, so the performance advisor is enabled when this policy is enabled. To disable the performance advisor, you must remove this health policy or narrow the membership of the health policy. To preserve the health policy for future use, consider keeping the default memory leak policy, but removing all of the members. To change the members, click **Operational policies > Health policies > Default_Memory_Leak**. You can edit the health policy memberships by adding and removing specific members from the policy.
- **Default excessive memory usage:** Set to 95 percent of the JVM heap size for 15 minutes
- **Default excessive request timeout:** Set for 5 percent of the requests timing out
- **Default excessive response time:** Set to 120 seconds
- **Default storm drain:** Default standard detection level

To view the recommendations that are made by the default health policies and to take actions on these recommendations, click **System administration > Task management > Runtime tasks**.

[Excessive request timeout health policy target timeout value](#)

The excessive request timeout health policy specifies an acceptable percentage of requests that can time out after being routed from the on demand router (ODR). The target timeout value can vary depending on settings in your environment.

Parent topic: [Configuring health management](#)

Related concepts:

[Excessive request timeout health policy target timeout value](#)

Related tasks:

[Configuring health management](#)

[Creating health policies](#)

[Setting maintenance mode](#)

[Creating health policy custom actions](#)

[Managing runtime tasks](#)

Excessive request timeout health policy target timeout value

The excessive request timeout health policy specifies an acceptable percentage of requests that can time out after being routed from the on demand router (ODR). The target timeout value can vary depending on settings in your environment.

About the excessive request timeout health policy

The excessive request timeout health policy defines an acceptable percentage of timed out requests that are routed through the ODR. For each control cycle of the health controller, the number of requests that time out after being routed from the ODR is examined. If the percentage of timed out requests exceeds the configured percentage, runtime tasks are generated to correct the situation.

Default values

By default, the target timeout value for requests is 60 seconds. The default value for the percentage of requests that can time out in the excessive request timeout health policy is 5%. However, setting other properties in your environment can affect the target timeout value that is used.

Settings that affect the timeout value

The following settings in the administrative console affect the target timeout value that is used:

- **ODR outbound request timeout** setting

To change the setting, click **Servers > On demand routers > on_demand_router_name > On demand router properties > On demand router settings** and set the **ODR outbound request timeout** value to a number of seconds.

- [goodServiceTimeLimitSpec](#) custom property
- [timeoutFactor](#) custom property
- Average response time value that is defined in the service policy

To change the setting, click **Operational policy > Service policy > service_policy_name**. Specify **Average response time** as the **Goal type** and enter a **Goal value**.

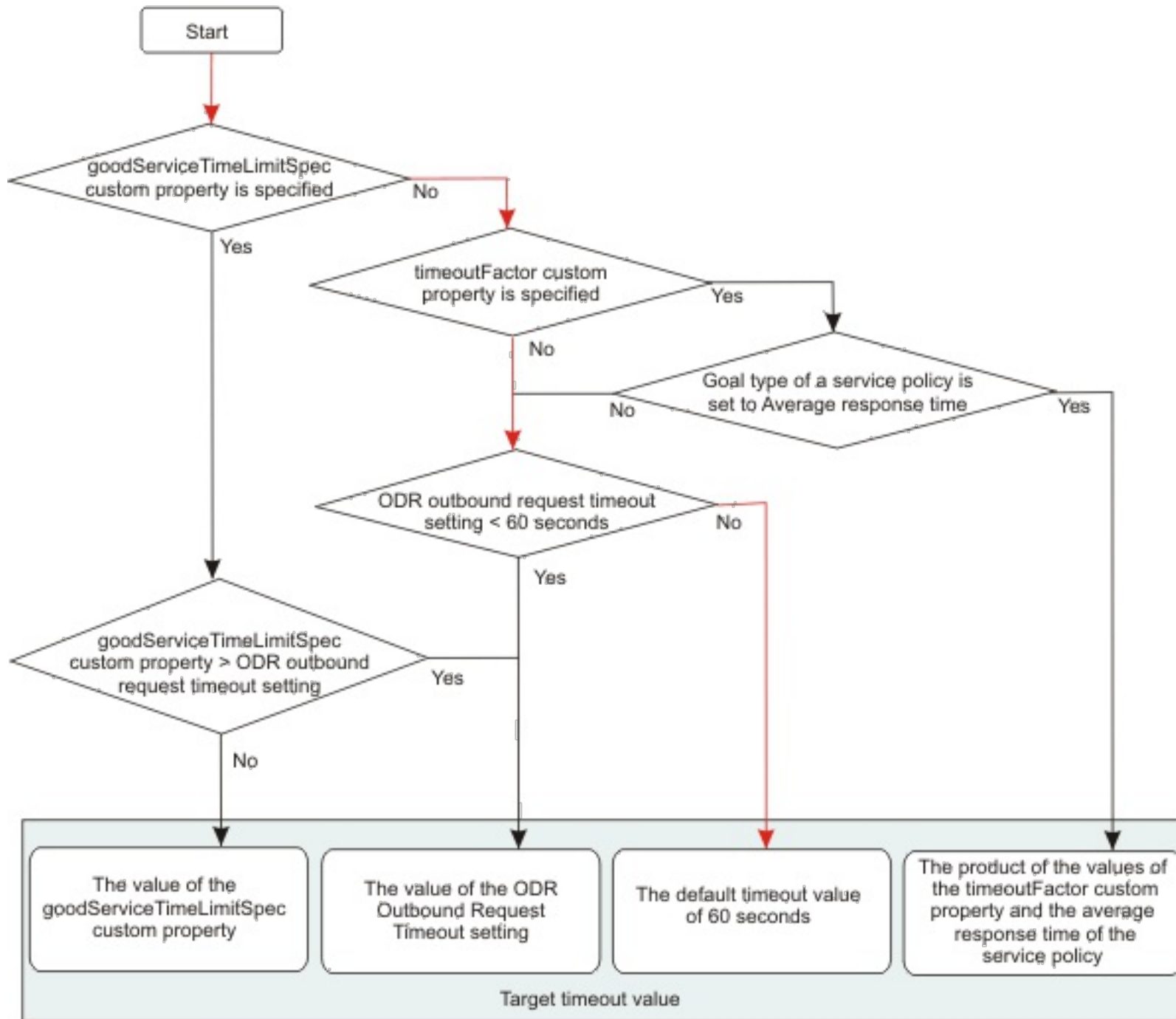
Determining the target timeout value

Depending your configuration, the timeout value that is used as a target value for the excessive request timeout health policy can vary. The following decision tree can help you determine the target timeout value for your environment. The default setting is the path that is indicated with red arrows, or a target timeout value of 60 seconds, which is the default setting for the ODR outbound request timeout.

Figure 1. Target timeout value decision tree.

The settings that affect the target timeout value combine together to indicate the target timeout value.

- The value of the `goodServiceTimeLimitSpec` custom property is used as the target timeout value if the `goodServiceTimeLimitSpec` custom property is specified and is larger than the **ODR outbound request timeout** value.
- The value of the **ODR outbound request timeout** setting is used as the target timeout value if the `goodServiceTimeLimitSpec` custom property is not specified, the `timeoutFactor` custom property is not specified, and the **ODR outbound request timeout** setting is set to less than 60 seconds.
- The default value of 60 seconds is used as the target timeout value if the `goodServiceTimeLimitSpec` custom property is not specified, the `timeoutFactor` custom property is not specified, and the **ODR outbound request timeout** setting is set to greater than 60 seconds.
- The timeout value is set to the product of the values of the `timeoutFactor` custom property and the average response time of the service policy if the `goodServiceTimeLimitSpec` custom property is not specified, the `timeoutFactor` custom property is specified, and the goal type of the service policy is set to average response time.



Parent topic: [Health management](#)

Related concepts:
[Health management](#)

Related tasks:
[Creating health policies](#)
[Managing runtime tasks](#)

Creating health policies

A health policy is the definition of specific health criteria that you want your WebSphere® Virtual Enterprise to protect itself against. The health management function uses the defined policy to search the environment for software malfunctions.

Before you begin

- To create a health policy, you must have configurator or administrator administrative privileges. The health controller must be enabled.
- If you want a *custom action* to run on the target server when the health condition that you define in the health policy breaches, you must define the custom action before you create the health policy. See [Creating health policy custom actions](#) for more information.

About this task

Health policies work with the health controller to monitor the operation of the servers in your environment. When the health controller detects that your servers are not meeting a defined health policy, you can take action to fix the problem. You can notify the administrator of problems, or WebSphere Virtual Enterprise can fix the problems automatically.

Procedure

1. In the administrative console, click **Operational policies > Health policies > New**.
2. Define health policy health condition properties.

Remember: The excessive request timeout and storm drain conditions do not apply to Java™ Message Service (JMS) and Internet Inter-ORB Protocol (IIOP) traffic.

Health policy conditions include the following properties:

- Set properties that pertain to the health condition that you selected. If you chose to create a custom health condition, you specify a subexpression that represents the metrics that you are evaluating in your custom condition. For more information about the conditions that you can set, click **Syntax help**.

As a best practice, consider the cost of collecting the data, analyzing the data, and if needed, enforcing the health policy when you define a custom condition. Consider the amount of traffic going over your network, especially when you scale out the number of servers that produce data. Before introducing new health policies into the production environment, perform analysis of these aspects of your custom health conditions.

You can further configure your custom health conditions that leverage PMI modules, most notably the webAppModules, at finer granularities than the server granularity. For example, you can use the subexpression builder to create a webAppModule policy as a starting point, then edit the expression to define a finer granularity:

```
PMIMetric_FromServerStart$webAppModule$SlamSess\#SlamSess.war\webAppModule.servlets\SlamSess\responseTime > 100L
```

In this example, the application name is displayed as SlamSess when you list the applications in the administrative console. If you are using an EAR file, specify the Web archive (WAR) file name after the EAR file name. If the WAR is not embedded in an EAR file, specify only the WAR file name. The SlamSess value is the servlet name that is listed in the web.xml file. The responseTime value is the statistic that is listed in the Performance Monitoring Infrastructure (PMI) module definition.

- Choose a reaction mode. **Supervise** mode gives more control to the administrator, so that they can approve or reject actions before they are taken.
 - Select the actions to take when the health policy conditions are not met. The available actions depend on the health condition type. These actions can be the existing default actions, or you can define custom actions to run an executable file. A list of actions are displayed in the order that they are run when the health condition breaches. You can add and remove steps from this list.
3. Select the memberships to monitor for your health policy. Layers of logic can apply to monitored memberships. For example, you might want to apply a specific health policy to each member of a cluster and to an application server outside of the cluster.
 4. Review and save your health policy.

Results

You created a health policy and applied that policy to a target environment. The health controller monitors the conditions that you defined for the health policy members, and takes the defined actions on the members when the conditions in the health policy breach.

What to do next

If you chose the **Supervise** reaction mode, then you receive recommendations to improve your health conditions. These recommendations display as runtime tasks that you can accept, deny, or close. To manage runtime tasks, click **System administration > Task Management > Runtime Tasks** in the administrative console. If you chose the **Automatic** reaction mode, actions to improve the health of your environment occur

automatically.

For supervised reaction mode runtime tasks, you can set the Java virtual machine (JVM) `com.ibm.ws.xd.hmm.controller.ControlConfig.approvalTimeOutMinutes` custom property, which specifies the number of minutes that can pass before a runtime task for the health controller expires. If you set the value to 5 minutes or less, the default value of 30 minutes is automatically used instead. If you do not take any actions on the runtime task, the task expires in the number of minutes that is specified in this property. If the runtime task expires when the health condition still exists, a new runtime task is generated.

If you configure your health policies often, consider using AdminTask commands to automate the process.

Custom health condition subexpression builder

Use the custom health condition subexpression builder to define a custom health condition for your health policy. Use the build subexpression utility to build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping. The subexpression builder validates the rule when you apply the changes, and alerts you to mismatched parentheses and unsupported logic operators.

Creating health policy custom actions

Use custom actions to define custom corrective tasks that you can use when a health condition breaches.

Modifying the JVM heap size

The Java virtual machine (JVM) heap size setting directly relates to how many server instances can be started within a dynamic cluster on a specific node. You might need to modify the JVM heap size setting based on your environment configuration. The default value is 256 MB. Modifying the heap size can affect application placement and the health management function.

Monitoring and tuning health management

Health management comes equipped with smart defaults that accommodate most environments. However, if you discover that your health controller is not working the way you want, then tune the default parameters.

Troubleshooting health management

You can look for the following problems when health management is not working, or not working the way you expect.

Parent topic: [Configuring health management](#)

Parent topic: [Configuring health management](#)

Related concepts:

[Health management](#)

Related tasks:

[Enabling and disabling health management](#)

[Creating health policy custom actions](#)

Related reference:

[Administrative roles and privileges](#)

[Troubleshooting health management](#)

Related information:

[Health controller custom properties](#)

Custom health condition subexpression builder

Use the custom health condition subexpression builder to define a custom health condition for your health policy. Use the build subexpression utility to build complex rule conditions from subexpressions by using AND, OR, NOT and parenthetical grouping. The subexpression builder validates the rule when you apply the changes, and alerts you to mismatched parentheses and unsupported logic operators.

To view this administrative console page, click **Operational policies > Health policies > New**. If you choose a custom health condition, the **Run reaction plan when** field is displayed. Click **Subexpression builder** to build the custom health condition.

Select the properties that you want to be in your custom health condition, and click **Generate subexpression**. The subexpression value displays. To append the subexpression to your custom health condition, click **Append**.

Parent topic: [Creating health policies](#)

Logical operator

Specifies the operator that is used to append this subexpression to the previous subexpression in the custom health condition.

and

Both subexpressions that are around the and operator must be true for actions to be taken on the health policy.

or

To select a node, one of the two subexpressions that are around the or operator must be true for actions to be taken on the health policy.

Operand

- **PMI metric: From server start** and **PMI metric: From last interval**

The **PMI metric: From server start operand** uses an average number of the reported values from the time that the server started.

The **PMI metric: From last interval** operand uses an average of the reported values in the last interval. The interval is the length of the health controller cycle.

Both PMI operands have the following PMI modules:

- **Connection pool module (JDBC):** These metrics can only be used on servers that are running WebSphere Application Server. The connection pool module has the following metrics:
 - **Faults:** Specifies the number of connection timeouts in the pool.
 - **Number of creates:** Specifies the total number of connections that are created.
 - **Percent used:** Specifies the average percent of the pool that is in use. The value is based on the total number of configured connections in the connection pool, not the current number of connections.
 - **Prepared statement cache discards:** Specifies the number of statements that are discarded because the cache is full.
 - **Number of destroys:** Specifies the total number of connections that are closed.
 - **Pool size:** Specifies the size of the connection pool.
 - **Number of connection handles:** Specifies the number of connection objects that are discarded because the cache is full.
 - **Concurrent waiters:** Specifies the average number of threads that are waiting for a connection at the same time.
 - **Number of managed connections:** Specifies the number of ManagedConnection objects that are in use for a particular connection pool. This metric applies to Version 5.0 data sources only.
 - **Percent maxed:** Specifies the average percent of the time that all connections are in use.
 - **JDBC time (milliseconds):** Specifies the average time, in milliseconds, that is spent running Java™ Database Connectivity (JDBC) calls. This time includes time that is spent in the JDBC driver, network, and database. This metric applies to Version 5.0 data sources only.
 - **Average use time (milliseconds):** Specifies the average time in milliseconds that a connection is used. The value is the difference between the time at which the connection is allocated and returned. This value includes the JDBC operation time.
 - **Number of returns:** Specifies the total number of connections that are returned to the pool.
 - **Free pool size:** Specifies the number of free connections in the pool.
 - **Number of allocates:** Specifies the number of connections that are allocated.
 - **Average wait time (milliseconds):** Specifies the average waiting time in milliseconds
- **System module:** These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent. The system module has the following metrics:
 - **CPU utilization: Since server start:** Specifies the average CPU utilization since the server started.

- **CPU utilization: Last interval:** Specifies the average CPU utilization since the last query.
 - **Free memory (KB):** Specifies a snapshot of free memory, in kilobytes.
- **Process module (for other servers):** These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent. The process module has the following metrics:
 - **Process resident memory (KB):** Specifies the process resident memory, in kilobytes.
 - **Process CPU utilization: Since server start:** Specifies the process CPU utilization since the server start.
 - **Process CPU utilization: Last interval:** Specifies the process CPU utilization in the last interval.
 - **Process total memory (KB):** Specifies the process total memory, in kilobytes.
- **EJB module:** These metrics can only be used on servers that are running WebSphere Application Server. The EJB module has the following metrics:
 - **Average concurrent active methods:** Specifies the average number methods that are active at the same time.
 - **Total method calls:** Specifies the number of calls to the remote methods of the bean.
 - **Methods submodule: Method loads:** Specifies the method load in the methods submodule.
 - **Stores:** Specifies the number of time that the bean data was store in persistent storage.
 - **Message count:** Specifies the number of messages that were delivered to the onMessage method of the bean. This message count applies to message-driven beans.
 - **Average concurrent live beans:** Specifies the average beans that are alive at the same time.
 - **Removes:** Specifies the number of times that beans were removed.
 - **Returns to pool:** Specifies the number of calls that are returning an object to the pool.
 - **Passivates:** Specifies the number of times that beans were moved to passivated state.
 - **Gets from pool:** Specifies the number of calls that are retrieving an object from the pool.
 - **Drains from pool:** Specifies the number of times that the daemon found the pool idle and attempted to clean the pool.
 - **Ready count:** Specifies the number of bean instances that are in ready state.
 - **Average create time (milliseconds):** Specifies the average time, in milliseconds, to run a bean create call. This time includes the time to load the bean.
 - **Returns discarded** Specifies the number of times that the returning object was discarded because the pool was full.
 - **Activates:** Specifies the number of times that beans were activated.
 - **Server session usage (percentage):** Specifies the percentage of the ServerSession pool that is being used. This metric applies to: message-driven beans.
 - **Loads:** Specifies the number of times that bean data was loaded from persistent storage.
 - **Message backout count:** Specifies the number of backed out messages that failed to be delivered to the onMessage method of the bean. This metric applies to: message-driven beans.
 - **Methods submodule: Method response time (milliseconds):** Specifies the method response time, in milliseconds.
 - **Passivation count:** Specifies the number of beans that are in a passivated state.
 - **Pool size:** Specifies the average number of objects that are in the pool.
 - **Load time (milliseconds):** Specifies the average time in milliseconds for loading the bean data from persistent storage.
 - **Average remove time (milliseconds):** Specifies the average time, in milliseconds, to run a beanRemove call. This time includes the time at the database.
 - **Gets found:** Specifies the number of times that a retrieve call found an available object in the pool.
 - **Activation time:** Specifies the average time, in milliseconds, for activating a bean object
 - **Average drain size:** Specifies the average number of objects that are discarded in each drain.
 - **Methods submodule: Method calls:** Specifies the number of method calls.
 - **Destroys:** Specifies the number of times that beans were destroyed.
 - **Average server session wait time (milliseconds):** Specifies the average time, in milliseconds, that is required to obtain a server session from the pool. This metric applies to message-driven beans.
 - **Creates:** Specifies the number of times that beans were created.
 - **Average method response time (milliseconds):** Specifies the average response time, in milliseconds, that elapses for remote method calls on the bean.
 - **Instantiates:** Specifies the number of times that beans were instantiated.
 - **Store time (milliseconds):** Specifies the average time, in milliseconds, for storing the bean data to persistent storage.
 - **Passivation time (milliseconds):** Specifies the average time, in milliseconds, for bean object passivation to occur.
- **Web application module:** These metrics can only be used on servers that are running WebSphere Application Server. Web application modules have the following metrics:
 - **Number of errors:** Specifies the total number of times that an error has been received from the servlet or JavaServer Pages (JSP) files.
 - **Total requests:** Specifies the total number of requests that a servlet processed.
 - **Response time (milliseconds):** Specifies the average response time, in milliseconds, in which servlet requests finish.
 - **Concurrent requests:** Specifies the number of requests that are currently processing.
 - **Number of reloads:** Specifies the number of servlets that are reloaded.
 - **Number of loaded servlets:** Specifies the number of servlets that are loaded.

- **JVM runtime module:** These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent. Java Virtual Machine (JVM) runtime modules have the following metrics:
 - **Free memory (KB):** Specifies the free memory, in kilobytes, in the JVM runtime.
 - **Up time (seconds):** Specifies the amount of time, in seconds, that the JVM has been running.
 - **Total memory (KB):** Specifies the total memory, in kilobytes, in the JVM runtime.
 - **Used memory (KB):** Specifies the amount of used memory, in kilobytes, in the JVM runtime.
- **Thread pool module:** These metrics can only be used on servers that are running WebSphere Application Server. Thread pool modules have the following metrics:
 - **Number of thread stops:** Specifies the number of threads that are declared as stopped.
 - **Percent maxed:** Specifies the average percent of the time that all threads are in use.
 - **Average active time (milliseconds):** Specifies the average time in milliseconds that the threads are in active state.
 - **Thread destroys:** Specifies the total number of threads that are destroyed.
 - **Pool size:** Specifies the average number of threads in a pool.
 - **Thread creates:** Specifies the total number of threads that are created.
 - **Concurrently hung threads:** Specifies the number of threads that are concurrently stopped.
 - **Number of cleared thread stops:** Specifies the number of thread stops that are cleared.
 - **Active threads:** Specifies the number of threads that are concurrently active.

Subexpression format for PMI Metric: From server start:

```
PMIMetric_FromServerStart$moduleName$metricName operator LongValueL (with "L" suffix)
```

Example:

```
PMIMetric_FromServerStart$systemModule$cpuUtilization > 90L
```

Subexpression format for PMI Metric: Last reported interval:

```
PMIMetric_FromLastInterval$moduleName$metricName operator LongValueL (with "L" suffix)
```

Example:

```
PMIMetric_FromLastInterval$webAppModule$responseTime > 200L
```

• ODR server level metric: From server start

You can use a subset of server level metrics that the on demand router (ODR) publishes. These metrics are cumulative and reported since the server start.

Metric name

You can use the following server level metrics:

- **Departs:** Specifies the number of requests that are dispatched from the queue to the server during the reported interval. A request is considered to be dispatched just once, even if it fails at the first server and is retried at another time. The next event after dispatch is return. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Response time (milliseconds):** Specifies an average response time for requests. To calculate this average, add the sum of the response times for requests that returned from the server to the client during the reported interval. The sum is in units of milliseconds. Divide by the serviced metric value to get the average response time. The response time of a request is the sum of the request's waiting time and service time. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Currently executing requests:** Specifies the number of requests that are running at the end of the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Service time (milliseconds)** Specifies the average service time for requests. To calculate this average, add the sum of each request service time for requests that returned to the server during the reported interval. The sum is in units of milliseconds. Divide by the serviced metric value in the same interval to get the average. The service time of a request is the time from dispatch to return. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Wait time (milliseconds):** Specifies the average wait time for requests. To calculate this average, add the sum of the time that each request spent waiting in the queue over the reported interval. The sum is in units of milliseconds. Divide by the number of departs to get the average wait time. Dropped requests do not contribute to this sum. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Errors:** Specifies the number of requests that returned from the server with an error indicator during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.

- **Serviced:** Specifies the number of requests that returned from server to client during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Timeouts:** Specifies the number of requests that returned due to service timeout during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.

Subexpression format:

```
ODRServerMetric_FromServerStart$metricName operator LongValueL (with "L" suffix)
```

Example:

```
ODRServerMetric_FromServerStart$errors > 100L
```

• **ODR server level metric: Last reported interval**

You can use the same set of metrics as the ODR server level metric: From server start operand. This operand uses an average of the reported values in the last interval. The interval is the length of the health controller cycle.

- **Departs:** Specifies the number of requests that are dispatched from the queue to the server during the reported interval. A request is considered to be dispatched just once, even if it fails at the first server and is retried at another time. The next event after dispatch is return. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Response time (milliseconds):** Specifies an average response time for requests. To calculate this average, add the sum of the response times for requests that returned from the server to the client during the reported interval. The sum is in units of milliseconds. Divide by the serviced metric value to get the average response time. The response time of a request is the sum of the request's waiting time and service time. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Currently executing requests:** Specifies the number of requests that are running at the end of the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Service time (milliseconds)** Specifies the average service time for requests. To calculate this average, add the sum of each request service time for requests that returned to the server during the reported interval. The sum is in units of milliseconds. Divide by the serviced metric value in the same interval to get the average. The service time of a request is the time from dispatch to return. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Wait time (milliseconds):** Specifies the average wait time for requests. To calculate this average, add the sum of the time that each request spent waiting in the queue over the reported interval. The sum is in units of milliseconds. Divide by the number of departs to get the average wait time. Dropped requests do not contribute to this sum. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Errors:** Specifies the number of requests that returned from the server with an error indicator during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Serviced:** Specifies the number of requests that returned from server to client during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Timeouts:** Specifies the number of requests that returned due to service timeout during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.

Subexpression format:

```
ODRServerMetric_FromLastInterval$metricName operator LongValue (with "L" suffix)
```

Example:

```
ODRServerMetric_FromLastInterval$serviced > 10000L
```

• **ODR cell level metric: From ODR start**

You can use a subset of cell level metrics that the ODR publishes. These metrics are cumulative and reported since the server start. You can use the following set of metrics:

- **Departs:** Specifies the number of requests that are dispatched from the queue to the server during the reported interval. A request is considered to be dispatched just once, even if it fails at the first server and is retried at another time. The next event after dispatch is return. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Response time (milliseconds):** Specifies an average response time for requests. To calculate this average, add the sum of the response times for requests that returned from the server to the

client during the reported interval. The sum is in units of milliseconds. Divide by the serviced metric value to get the average response time. The response time of a request is the sum of the request's waiting time and service time. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.

- **Current® queue length:** Specifies the length of the queue at the end of the reported interval.
- **Service time (milliseconds)**
- **Errors:** Specifies the number of requests that returned from the server with an error indicator during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Average queue length:** Specifies the average length of the queue. To calculate this average, add the sum of the queue lengths that are reported at each request arrival before insertion, and divide the sum by the number of arrivals. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Serviced:** Specifies the number of requests that returned from server to client during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Timeouts:** Specifies the number of requests that returned due to service timeout during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Currently executing requests:** Specifies the number of requests that are running at the end of the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Arrivals:** Specifies the number of requests that arrived during the reported interval. The next event, if any, after arrival is either dispatch or drop. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Queue overflow drops:** Specifies the number of requests that were initially accepted into the queue at some time and then, at some time during the reported interval, were ejected from the queue. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Queue drops:** Specifies the number of requests that were initially accepted into the queue at some time and then were ejected from the queue at some time during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Delayed:** Specifies the number of requests that arrived during the reported interval and were not immediately dispatched or dropped. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Wait time (milliseconds):** Specifies the average wait time for requests. To calculate this average, add the sum of the time that each request spent waiting in the queue over the reported interval. The sum is in units of milliseconds. Divide by the number of departs to get the average wait time. Dropped requests do not contribute to this sum. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.

Subexpression format:

```
ODRCellMetric_FromServerStart$metricName operator LongValue (with "L" suffix)
```

Example:

```
ODRCellMetric_FromServerStart$arrivals > 10000L
```

• ODR cell level metric: Last reported interval

You can use the same set of metrics as the ODR cell level metric: From server start operand. This operand uses an average of the reported values in the last interval. The interval is the length of the health controller cycle.

- **Departs:** Specifies the number of requests that are dispatched from the queue to the server during the reported interval. A request is considered to be dispatched just once, even if it fails at the first server and is retried at another time. The next event after dispatch is return. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Response time (milliseconds):** Specifies an average response time for requests. To calculate this average, add the sum of the response times for requests that returned from the server to the client during the reported interval. The sum is in units of milliseconds. Divide by the serviced metric value to get the average response time. The response time of a request is the sum of the request's waiting time and service time. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Current queue length:** Specifies the length of the queue at the end of the reported interval.
- **Service time (milliseconds)** Specifies the average service time for requests. To calculate this average, add the sum of each request service time for requests that returned to the server during the reported interval. The sum is in units of milliseconds. Divide by the serviced metric value in the same interval to get the average. The service time of a request is the time from dispatch to return.

These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.

- **Errors:** Specifies the number of requests that returned from the server with an error indicator during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Average queue length:** Specifies the average length of the queue. To calculate this average, add the sum of the queue lengths that are reported at each request arrival before insertion, and divide the sum by the number of arrivals. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Serviced:** Specifies the number of requests that returned from server to client during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Timeouts:** Specifies the number of requests that returned due to service timeout during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Currently executing requests:** Specifies the number of requests that are running at the end of the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Arrivals:** Specifies the number of requests that arrived during the reported interval. The next event, if any, after arrival is either dispatch or drop. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Queue overflow drops:** Specifies the number of requests that were initially accepted into the queue at some time and then, at some time during the reported interval, were ejected from the queue. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Queue drops:** Specifies the number of requests that were initially accepted into the queue at some time and then were ejected from the queue at some time during the reported interval. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Delayed:** Specifies the number of requests that arrived during the reported interval and were not immediately dispatched or dropped. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.
- **Wait time (milliseconds):** Specifies the average wait time for requests. To calculate this average, add the sum of the time that each request spent waiting in the queue over the reported interval. The sum is in units of milliseconds. Divide by the number of departs to get the average wait time. Dropped requests do not contribute to this sum. These metrics can be used on servers that are running WebSphere Application Server or that are running other middleware servers with the middleware agent.

Subexpression format:

```
ODRCellMetric_FromLastInterval$metricName operator LongValue (with "L" suffix)
```

Example:

```
ODRCellMetric_FromLastInterval$timeouts > 100L
```

• MBean operation metric: Long return type and MBean operation metric: String return type

For Managed Bean (Mbean) operation metric operands, you must specify the Object name query string and the MBean method name. These metrics can only be used on servers that are running WebSphere Application Server.

Object name query string

When you are creating the Object name query string, all special characters must be escaped with a backslash character. The backend expects the object name query string in a form where all special characters have been escaped appropriately.

The value that you enter for the object name query string must have both the process=<process_name>, and node=<node_name> strings specified, or none specified. If you specify both process=<process_name>, and node=<node_name>, the backend creates a singleton MBeanSensor sensor that senses the particular MBean on a server and node. If you specify none, the backend appends the name of the current server as the process name and the name of the current node as the node name, creating a MBeanSensor sensor for each server to which the health policy applies. If you specify only one of the two process=<process_name> or node=<node_name>, an error results.

MBean method name

Specifies the name of the MBean method to invoke.

Subexpression format long metrics:

```
MBeanOperationMetric_TypeLong$objectNameQueryString$methodName operator LongValueL (with "L" suffix)
```

Example for long metrics:

```
MBeanOperationMetric_TypeLong$WebSphere\:\*\,type\=HealthConditionLanguageInitializer\,node\=hipods3\,process\=nodeagent$getNumOfOperands > 10L
```

Subexpression format for string metrics:

```
MBeanOperationMetric_TypeString$ObjectNameQueryString$methodName operator StringValue
```

Example for string metrics:

```
MBeanOperationMetric_TypeString$WebSphere\:\*\,type\=HealthConditionLanguageInitializer\,node\=hipods3\,process\=nodeagent$getNumOfOperands = 't'
```

- **MBean attribute metric: Long return type** and **MBean attribute metric: String return type**

The MBean attribute metrics are used for querying an attribute of a MBean rather than invoking a method on the MBean. The operand takes the Object name query string and the attribute name as inputs. These metrics can only be used on servers that are running WebSphere Application Server.

Object name query string

When you are creating the Object name query string, all special characters must be escaped with a backslash character. The backend expects the object name query string in a form where all special characters have been escaped appropriately.

The value that you enter for the object name query string must have both the process=<process_name>, and node=<node_name> strings specified, or none specified. If you specify both process=<process_name>, and node=<node_name>, the backend creates a singleton MBeanSensor sensor that senses the particular MBean on a server and node. If you specify none, the backend appends the name of the current server as the process name and the name of the current node as the node name, creating a MBeanSensor sensor for each server to which the health policy applies. If you specify only one of the two process=<process_name> or node=<node_name>, an error results.

Attribute name

Specifies the attribute that is queried on the MBean.

Subexpression format for long metrics:

```
MBeanAttributeMetric_TypeLong$ObjectNameQueryString$attributeName operator LongValue
```

Example for long metrics:

```
MBeanAttributeMetric_TypeLong$WebSphere\:\*\,type\=HealthConditionLanguageInitializer\,node\=hipods3\,process\=nodeagent$Number0fOperands > 10L
```

Subexpression format for string metrics:

```
MBeanAttributeMetric_TypeString$ObjectNameQueryString$attributeName operator StringValue
```

Example for string metrics:

```
MBeanAttributeMetric_TypeString$WebSphere\:\*\,type\=HealthConditionLanguageInitializer\,node\=hipods3\,process\=nodeagent$OperatorList = 'test'
```

- **URL return code metric**

With this operand, you can ping any relative path (URI) on the server that is the target of this policy. The return value is used in the condition expression for the custom health policy.

URL port number

Specifies the port number to ping.

URL relative path

Specifies the URL to ping. Any special characters in the string must be escaped with a backslash (\) character.

Value

Specifies an integer that is the expected return code of the ping.

You can use this operand to ping any general purpose URL by selecting the on demand router (ODR) as the target of the health policy and by setting the appropriate routing rules in the ODR.

You can use this operand to select members that are running WebSphere Application Server, or that are running other middleware software with the middleware agent.

Subexpression format:

```
URLReturnCodeMetric$portNumber$relativePath operator IntValue
```

Example:

```
URLReturnCodeMetric$9060$ibm\console\login.do = 200
```

The URL sensor returns 0 if the Web site cannot be reached:

```
URLReturnCodeMetric$9060$ibm\console\login.do = 0
```

- **External URL return code metric**

With this operand, you can enter an absolute URL instead of a relative URL. By doing so, you can periodically send a ping request to other targets than application servers or on demand routers.

Always create a custom health action without specifying a target server.

Example:

```
ExternalURLReturnCodeMetric$http://foo.bar.com <> 200
```

If http://foo.bar.com returns a different response code than 200, the health policy is triggered. Take this custom action to run a custom script, for example a script to recycle a web server.

Operator

- **Equals (=)**: The equality operator expresses a case-sensitive match.
- **Not Equals (<>)**: The not equal operator expresses that the operand value is not equal to the value you enter.
- **Greater Than (>)**: The greater-than operator is for use with numbers.
- **Greater Than or Equals (>=)**: The greater-than or equal to operator is for use with numbers.
- **Less Than (<)** : The less-than operator is for use with numbers.
- **Less Than or Equals (<=)**: The less-than or equal to operator is for use with numbers.
- **Between (BETWEEN)**: The value must be between a **Lower bound** and **Upper bound** that you specify.
- **In (IN)**: The value must be in a list of values. You can type in values and add them to a list.

Value

Depending on the operator that you choose, type in a value for the subexpression that you want to create.

Subexpression

After you click **Generate subexpression**, this field displays the generated subexpression fragment based on the options that you selected. To add this subexpression to your custom health condition, click **Append**.

Creating health policy custom actions

Use custom actions to define custom corrective tasks that you can use when a health condition breaches.

Before you begin

Create a Java™ or non-Java executable file to run when the health condition breaches. You can use environment variables in your executable file to display the cell, node, and server that is affected by the health action, for example:

```
#!/bin/sh
echo "The sick server is "
echo $server
echo " which is on node "
echo $node
echo " which is a part of cell "
echo $cell
```

About this task

By default, you can define the following actions to occur when a health condition breaches: restart the server, take thread dumps, or take heap dumps. If you have more specific needs for actions that occur when a health condition breaches, then create a *custom action*. Custom actions consist of Java or non-Java process definitions, and can run on the deployment manager, a node that is hosting servers that breach health conditions, the node where the health management controller is running, or a node that you specify.

Create a custom action before you create a health policy. When you define the health policy, select the custom action as part of the action plan that runs when the health condition breaches. You can also define a custom action when you are creating an action plan in the health policy creation wizard.

When a health policy contains multiple custom actions, updating the custom actions can cause the health controller runtime to be out-of-sync with the administrative console configuration. If you add, delete or modify a custom action for a policy that contains multiple custom actions, start and stop the health controller after saving your changes.

Procedure

1. Select a Java or non-Java action type. In the administrative console, click **Operational policies > Custom action > New**. Choose the type of action that you want to create.
2. Define the custom action properties.

For health actions that run non-Java and Java executable files:

- Specify a name for the action, the fully qualified path and name of the executable file, and any arguments to pass to the executable file.
- If required, specify variables for the user name and password, and specify the actual values to pass for the user name and password at the time the executable file runs.
- Specify the operating systems on which the executable files can run.
- Specify a fully qualified path for a working directory, which is where to run the executable file.

For health actions that run Java executable files, you must also specify the following properties:

- Specify the location of the Java executable directory.
- Specify the type of target: a Java class, or an executable Java Archive (JAR) file.
- Specify the environment variable that stores the process ID (PID) file name. The PID file name is stored in a WebSphere® variable. Remember the name of the variable so that you can create the variable in a later step.

For example, if you had an executable file to collect logs called logCollector.sh, you might fill in the following values for the custom action:

- **Name:** LogCollect
- **Executable:** /opt/mws/bin/logCollector.sh
- **Executable arguments:** None
- **Operating system:** UNIX
- **Working directory:** /opt/mws/bin/

3. Save the custom action.
4. If your health action is running a Java executable file, create the PID file name environment variable. The PID file name specifies the location of the PID file where the server operation runs. To define the PID file name variable, click **Servers > All servers > middleware_server > Variables > New or Environment > WebSphere Variables > New**. Enter the same name for the variable that you entered in the health action panel, and for the value, enter the fully qualified or variable qualified location of the PID file.

Results

You created a custom action that you can add to the action plans of your health policies.

What to do next

Create a health policy. In the administrative console, click **Operational policies > Health policies > New**.

Parent topic: [Creating health policies](#)

Related concepts:
[Health management](#)

Related tasks:
[Creating health policies](#)
[Configuring health management](#)

Related reference:
[Health policy administrative tasks](#)

Modifying the JVM heap size

The Java™ virtual machine (JVM) heap size setting directly relates to how many server instances can be started within a dynamic cluster on a specific node. You might need to modify the JVM heap size setting based on your environment configuration. The default value is 256 MB. Modifying the heap size can affect application placement and the health management function.

Before you begin

Before modifying the JVM heap size, see [Tuning Java virtual machines](#).

About this task

The total value of all server JVM heap sizes within the dynamic cluster for a specific node must be less than half of the total RAM of that computer. To determine the maximum heap size setting for a single server instance, use the following equation:

$$\text{total_RAM} / 2 / \text{number_of_servers} = \text{maximum_heap_size}$$

For example, to support three servers on a machine with 1.5 GB of RAM:

$$\begin{aligned} 1.5 \text{ GB} / 2 &= 750 \text{ MB} \\ 750 \text{ MB} / 3 &= 250 \text{ MB} \end{aligned}$$

The maximum heap size is 250 MB for each server instance.

The following table contains some maximum heap size samples.

Table 1. Maximum heap size samples

WebSphere® Virtual Enterprise server total RAM in GB	Number of supported servers	Dynamic cluster size in MB	Approximate maximum heap size in MB
1.5	1	Less than 750	750
1.5	2	Less than 750	375
1.5	3	Less than 750	250
2	1	Less than 1000	1000
2	2	Less than 1000	500
2	3	Less than 1000	333

When a dynamic cluster creates server instances, a server template is used to set initial values for the newly created server instance. The following steps outline the procedure for modifying JVM heap size within the server template. Newly created server instances can then use the maximum heap size you specified.

Procedure

1. In the administrative console click **Servers > dynamic clusters > *dynamic_cluster_name* > Server template > Java and process management > Process definition > Java virtual machine**.
2. Modify the heap size accordingly. Typically, the total value of all server instance JVM heap sizes on a specific node must be less than half of the total RAM of that computer. The default maximum heap size value is 256 MB.
3. Click **OK**.

Results

The heap size that you define in the dynamic cluster template is configured on each server in the dynamic cluster. If the server and application placement controller are on the same node, the application placement controller can retrieve the heap size from this configured value.

If the server is not on the same node as the application placement controller, the application placement controller obtains the heap size from the configuration repository in the deployment manager. If the deployment manager is not available or the heap size is not defined on the deployment manager, the application placement controller tries to obtain the heap size by contacting the server. If the server is not available, the application placement controller uses the heap size value in the <os>.systemlaunch.properties file.

Parent topic: [Creating health policies](#)

Monitoring and tuning health management

Health management comes equipped with smart defaults that accommodate most environments. However, if you discover that your health controller is not working the way you want, then tune the default parameters.

Before you begin

Verify that you have proper security authorization in the console to modify these settings. Privileges for health policies differ, depending on the administrative role of the user. Roles include monitor, operator, configurator, and administrator. If you are a user with either a monitor or an operator role, you can only view health policy information. If you are a user with either a configurator or an administrator role, you have all configuration privileges for health policies.

About this task

Use the following steps to modify the health controller parameters. Tune these parameters when the health management infrastructure is not working the way that you want.

Procedure

1. Access the health controller settings in the administrative console by clicking **Operational policies > Autonomic managers > Health controller**.
2. Determine whether you want your changes to be persistent or applied to the current runtime for testing purposes. This page is divided into two tabs: Configuration and Runtime. With the Configuration tab, you can look at fields that are previously configured, and in some cases, make edits to those fields. With the Runtime tab, you can view the fields that are currently used by the health controller, and in some cases, make changes to these values. In contrast to the Configuration tab, these changed values are sent directly to the health controller to change the parameters that the controller uses. These changes are not stored into the repository by default, allowing the user to make a temporary change to the parameters that are not persisted.

Tip: Apply your changes to the Runtime tab and test the changes before committing them. You can select **Save to configuration** in the Runtime tab, so you can make configuration changes and test them in the runtime. After you are sure that you want to commit those changes permanently, click **Save to configuration**.

3. Modify and test your settings.

Control Cycle Length	Specifies the time between consecutive health checks. The value is specified in minutes and ranges from 1 to 60 minutes. Longer control cycles reduce the health monitoring load. The disadvantage is that health conditions that occur during that period are not detected until the next control cycle. For example, if you have a health policy with a workload condition of 10,000 requests associated to an application server and the value is specified as 60 minutes, the health controller checks every 60 minutes to determine if the application server has served 10,000 requests. If it has served 9,999 requests when it checks, it checks again for another 60 minutes (the control cycle length), so the server can actually serve more than 10,000 requests prior to a restart.
Maximum Consecutive Restarts	Specifies the number of attempts to revive an application server after a restart decision is made. If this number is exceeded, the assumption is that the operation failed and restarts are disabled for the server. The value must be a whole number between 1 and 5, inclusive.
Minimum Restart Interval	Controls the minimum amount of time that must elapse between consecutive restarts of an application server instance. If a health condition for an application server is breached during that time, the restart is set to a pending state. When the minimum restart interval passes, the restart occurs. The value can range from 15 minutes to 365 days, inclusive. A value of 0 disables the minimum restart value.
Restart Timeout	Consists of the sequence of stop and start server actions. The restart timeout specifies how long to wait for a server to stop before explicitly checking its state and attempting startup. If the length of time to start and stop an application server is unusually high, you might want to set this value accordingly, so that the restart action does not time out. Always specify the value in minutes. The value can range from 1 minute to 60 minutes, specified as a whole number.

Enable Health Monitoring	Enables or disables the operation of the health controller. When enabled, the health controller continuously monitors the health policies in the system. You can disable the health controller without removing the health policies from the system.
Prohibited Restart Times	<p>Specifies the times and days of the week during which a restart of an application server instance is prohibited. Specify the start and end times by selecting the hour and minute using a 24 hour clock and by selecting the days of the week.</p> <p>You can specify multiple time blocks, if needed. If you specify a start time and end time, you must also specify at least one day of the week that these times are prohibited. The block between the start time and end time cannot cross the midnight boundary. If you need to specify a block of, for example, 10:00 PM to 1:00 AM, you need to specify two blocks, one from 22:00 to 23:59 and one from 00:00 to 01:00. Click Add to add additional time constraints.</p> <p>You can remove an existing constraint by selecting the check box next to the constraint and by clicking the Remove. As with the minimum restart interval, a restart for a health condition that is violated is delayed until after a prohibited time interval passes.</p>

Results

You have modified the health management configuration settings to tune your system.

What to do next

See [Troubleshooting health management](#) for more information about modifying the health management settings when they are not working the way that you want.

Parent topic: [Creating health policies](#)

Related reference:

[checkHmmLocation.jacl](#)

[Administrative roles and privileges](#)

Troubleshooting health management

You can look for the following problems when health management is not working, or not working the way you expect.

Finding the right logs

The health controller is a distributed resource that is managed by the high availability (HA) manager. It exists within all node agent and deployment manager processes and is active within one of these processes. If a process fails, the controller becomes active on another node agent or deployment manager process.

To determine where the health controller is running, click **Runtime Operations > Extended Deployment > Core components** in the administrative console. The location and stability status of the health controller displays.

Performance advisor is enabled with the default memory leak health policy

The default memory leak health policy uses the performance advisor functionality, so the performance advisor is enabled when this policy has members assigned. To disable the performance advisor, you must remove this health policy or narrow the membership of the health policy. To preserve the health policy for future use, consider keeping the default memory leak policy, but removing all of the members. To change the members, click **Operational policies > Health policies > Default_Memory_Leak**. You can edit the health policy memberships by adding and removing specific members from the policy.

Health controller settings

The following list contains issues that are encountered as a result of the health controller settings:

Health controller is disabled

Verify the setting in the administrative console by clicking **Operational policies > Autonomic controllers > Health controller** and select both the Configuration and Runtime tabs. The health controller is enabled by default.

Restarts are prohibited at this time

Verify the prohibited restart times in the administrative console by clicking **Operational policies > Autonomic controllers > Health controller** and by selecting the **Prohibited restart** field. By default, no times are prohibited.

Restarting too soon after the previous restart

To check the minimum restart interval in the administrative console, click **Operational policies > Autonomic controllers > Health controller** modify the **Minimum Restart Interval** field. No minimum interval is defined by default.

Control cycle is too long

To check the control cycle length in the administrative console, click **Operational policies > Autonomic controllers > Health controller** and adjust the value if necessary. The health controller checks for policy violations periodically. If its control cycle length is too long, it might not restart servers quickly enough.

The server has been restarted *X* times consecutively, and the health condition continues to be violated

In this case, *X* indicates the maximum consecutive restart parameter of the health controller. The health controller concludes that restarts are not fixing the problem, and disables the restarts for the server. The following message displays in the log:

```
WXDH0011W: Server servername exceeded maximum verification failures: disabling restarts.
```

The health controller continues to monitor the server and displays messages in the log if the health policy is violated:

```
WXDH0012W: Server servername with restarts disabled failed health check.
```

You can enable restarts for the server by performing any of the following actions:

- Disable and then enable the health controller.
- Adjust the **Maximum Consecutive Restarts** controller setting.
- Run the following command from the prompt:

```
wsadmin -profile HmmControllerProcs.jacl enableServer servername
```

This script is available in the <install_root>\bin directory on the node agent or deployment manager nodes. This script requires a running deployment manager.

Health policy settings

The following issues are encountered as a result of the health policy settings:

The server is not part of a health policy

Verify that the health policy memberships apply to your server in the administrative console by clicking **Operational policies > Health policies**.

The reaction mode of a policy containing the server is supervised

Check the administrative console by clicking **Runtime Operations > Task Management > Runtime tasks** to find approval requests for a restart action for a policy in **Supervised** mode. Servers are restarted automatically when you set **Automatic** as the reaction mode. The following message is written to the log for the supervised condition:

```
WXDH0024I: Server server name has violated the health policy health condition, reaction mode is supervised.
```

The server is a member of a static cluster and is the only cluster member running

The health policy does not bring down all members of a cluster at the same time. If a cluster has one cluster member, or one cluster member is running, then the cluster is not restarted.

The server is a member of a dynamic cluster, the number of running instances does not exceed the minimum value, and the placement controller is disabled

Check the **minimum number of instances required** for the dynamic cluster by clicking **Servers > Dynamic clusters** in the administrative console. In this case, health management treats the dynamic cluster like a static cluster, using the minimum number of instances parameter.

The health controller has not received the policy

The health controller does not run on the deployment manager where the health policies are created. If the deployment manager is restarted after the health controller started, the health controller might not have the new policy.

You can alleviate this problem by performing the following steps:

1. Disable the health controller. In the administrative console click **Operational policies > Autonomic managers > Health controller**.
2. Synchronize the configuration repositories with the back-end nodes. In the administrative console, click **System Administration > Nodes**. Select the nodes to synchronize, and click **Synchronize**.
3. Restart the health controller. In the administrative console click **Operational policies > Autonomic managers > Health controller**.
4. Synchronize the configuration repositories with the back-end nodes. In the administrative console, click **System Administration > Nodes**. Select the nodes to synchronize, and click **Synchronize**.

Application placement controller interactions

The following list contains issues that are encountered as a result of the health management and application placement controller interactions:

The server is a member of a dynamic cluster, but the placement controller cannot be contacted

For dynamic cluster members, health monitoring checks with the application placement controller to determine whether a server can be restarted. If the application placement controller is enabled, but cannot be contacted, the following message displays in the log:

```
WXDH1018E: Could not contact the placement controller
```

Verify that the placement controller is running. To determine where the health controller is running, click **Runtime Operations > Extended Deployment > Core components** in the administrative console. The location and stability status of the health controller displays. The health controller logs messages to the particular node agent or deployment manager indicated by the current location.

The server is a member of a dynamic cluster, the placement controller is running, and the placement controller instructs health management not to restart the server

The placement controller might require the server instance to remain running.

The server is stopped, but not started.

In a dynamic cluster, a restart can take one of several forms:

- Restart in place (stop server, start server).
- Start a server instance on another node, and stop the failing one.
- Stop the failing server only, assuming that the remaining application instances can satisfy demand.

The placement controller determines which form a restart takes, and if necessary, where to start the new instance. After a restart is performed in a dynamic cluster, health management issues a request to the placement controller to recompute its placement.

Sensor problems

The following list contains issues that are encountered as a result of the health management and node group membership settings:

No sensor data is received for the server.

Health management cannot detect a policy violation if it receives no data from the sensors that are required by the policy. If sensor data is not received during the control cycle, health management prints the following log message:

```
WXDH3001E: No sensor data received during control cycle from server server_name for health class healthpolicy.
```

For response time conditions, health management receives data from the on demand router (ODR). No data is generated for these conditions until requests are sent through the ODR.

Parent topic: [Creating health policies](#)

Tuning the WebSphere Virtual Enterprise cell

Start here for a summary of key performance tuning parameters for your WebSphere® Virtual Enterprise configuration.

About this task

You can complete any of these steps to improve the performance of your WebSphere Virtual Enterprise configuration.

Procedure

- Assign all URL patterns from an application to transaction classes. Any URL that does not map to a transaction class is classified as "best effort", also called discretionary, the default policy. Transaction classes are part of service policies. To change transaction classes from the administrative console, select **Operational policies > Service policies > *service_policy_name* > Transaction classes**.
- Check service policy goals against the actual performance of the service class and adjust the goals if necessary. For example, your service policy has an average response goal of 200 milliseconds and the measured average service time is 300 milliseconds. You might consider adjusting your goal to 500 milliseconds.
- To change how quickly the application placement controller reacts to load changes, adjust the following properties:
 - Configure the minimum time between placement changes in the administrative console. Select **Operational policies > Autonomic controllers > Application placement controller**. Set the minimum time between placement changes to a suitable value.
 - Configure the application placement controller cycle from the administrative console. Select **Operational policies > Autonomic controllers > Application placement controller > Custom properties**. Add a custom property with the name **minControlCycleLength** and a suitable value in minutes.
- Adjust the control cycle length. The control cycle length minimum property determines how quickly the autonomic request flow manager (ARFM) reacts to load changes. You can change the ARFM control cycle length. From the administrative console select **Operational policies > Autonomic controllers > Autonomic request flow manager**. Set the control cycle length minimum property to a suitable value.
- Adjust ARFM to limit the maximum use of the nodes.
 1. In the administrative console, select **Operational policies > Autonomic controllers > Autonomic request flow manager**.
 2. Set the Maximum CPU Utilization property. Higher values give better resource use, lower values give more robust operation.

Parent topic: [Configuring the autonomic managers](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

[Configuring the health controller](#)

[Managing the WebSphere Virtual Enterprise environment](#)

Using the centralized installation manager to manage your environment

Use the centralized installation manager to simplify the tasks of deploying product components to your WebSphere® Application Server Network Deployment cell or other server environments.

Before you begin

You must first install the centralized installation manager repository on the deployment manager, and add one or more product components to the repository. Complete this task during the installation process of WebSphere Virtual Enterprise. For more information, see [Using ISMP programs to install the product](#).

Alternatively, you can use the centralized installation manager repository tool to add one or more product components to the repository. You can optionally install this tool when you install WebSphere Virtual Enterprise only. For more information, see [Using the centralized installation manager repository tool to add products](#).

About this task

The centralized installation manager installs selected product components from its repository, which is located on the deployment manager, to the nodes. With this feature, you can shorten the number of steps that are required to create and manage your environments. The various tasks you can complete by using the centralized installation manager include installing product components to target workstations, applying maintenance to server environments, monitoring submitted requests, and creating additional installation targets.

Procedure

1. Install product components to one or more installation targets. See [Installing packages](#) for more information.
2. Apply various types of maintenance to your product environment, your WebSphere Application Server Network Deployment environment, or other server environments. See [Downloading the Update Installer for WebSphere Software](#) for more information.
3. Monitor your submitted requests by viewing the progress, completion status, and log files of each. See [Monitoring requests](#) for more information.
4. Create additional installation targets to enhance your environment, and manage any existing installation targets. See [Managing installation targets](#) for more information.

What to do next

Refer to the detailed instructions for completing each task to begin managing and enhancing your environments.

[Centralized installation manager overview](#)

Use the centralized installation manager to install and maintain your workstations that comprise a WebSphere Virtual Enterprise cell. As an administrator, you can remotely install or uninstall product packages and maintenance to specific nodes directly from the administrative console without having to log in and repetitively perform these tasks.

[Installing packages](#)

Use the centralized installation manager to install one or more packages to specified installation targets.

[Downloading package descriptors and the associated binary files](#)

To enhance your product environment, download additional installation packages and maintenance files to the centralized installation manager repository to install later on installation targets. Use this topic to manage the installation packages and maintenance files that are located in your centralized installation manager repository.

[Managing installation targets](#)

Use this topic to add or remove an installation target, which is the workstation on which selected software packages might be installed. You can also edit the configuration of an existing installation target, and store the administrative ID and password of each target for later use when installing or uninstalling packages.

[Centralized installation manager overview](#)

Use the centralized installation manager to install and maintain your workstations that comprise a WebSphere Virtual Enterprise cell. As an administrator, you can remotely install or uninstall product packages and maintenance to specific nodes directly from the administrative console without having to log in and repetitively perform these tasks.

[Installing packages](#)

Use the centralized installation manager to install one or more packages to specified installation targets.

[Uninstalling packages](#)

Use the centralized installation manager to uninstall a previously installed package from remote workstations.

Monitoring requests

After you submit one or more requests to the centralized installation manager, you can monitor the progress of and view specific details about each installation and uninstallation request.

Managing installation targets

Use this topic to add or remove an installation target, which is the workstation on which selected software packages might be installed. You can also edit the configuration of an existing installation target, and store the administrative ID and password of each target for later use when installing or uninstalling packages.

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Related tasks:

[Using Remote Execution and Access to access remote installation targets](#)

Centralized installation manager overview

Use the centralized installation manager to install and maintain your workstations that comprise a WebSphere® Virtual Enterprise cell. As an administrator, you can remotely install or uninstall product packages and maintenance to specific nodes directly from the administrative console without having to log in and repetitively perform these tasks.

The centralized installation manager does not replace the product installation wizard or the IBM® Update Installer for WebSphere Software. Instead, the centralized installation manager starts the product installation wizard or the Update Installer to install or uninstall the packages or maintenance.

The various product packages and maintenance files that you can install or uninstall are included in the following list:

- WebSphere Virtual Enterprise
- WebSphere Extended Deployment Compute Grid
- WebSphere Virtual Enterprise refresh packs, fix packs, and interim fixes
- WebSphere Application Server refresh packs, fix packs, and interim fixes
- Update Installer for WebSphere Software
- WebSphere Application Server Community Edition Version 2.1.0.x
- WebSphere Application Server Community Edition Version 2.1.1.x

Remember: The centralized installation manager does not update or maintain the deployment manager workstation, because the centralized installation manager operates from the binaries of the deployment manager.

See [Installing and configuring the product](#) to learn more about installing or uninstalling product packages or maintenance from the deployment manager, and uninstalling product packages when no augmented profiles exist.

See [Installing the product and additional software](#) for more information about installing and uninstalling WebSphere Application Server.

The following sections provide information to consider when you use the centralized installation manager.

Starting the node agent or middleware agent

The centralized installation manager relies on current information regarding the versions of WebSphere Application Server and WebSphere Virtual Enterprise that are installed on each node. This information is kept current on the deployment manager configuration by the node agent or middleware agent that is running on each node. The deployment manager contains the correct versions of WebSphere Application Server and WebSphere Virtual Enterprise installed on each node if the node agent or middleware agent of each node is started at least once after each update is applied. To ensure that the deployment manager receives this information, the centralized installation manager automatically starts the node agent or middleware agent after each installation or uninstallation process.

Important: To locally apply updates on the nodes without using the centralized installation manager, issue the **startNode** command or the **startAgent** command after you complete the operation to manually start the node agent or middleware agent.

Augmented profiles

When you install the product, augment the node profiles for that particular package. After you augment the profile and restart the middleware agent, the node metadata is updated with the current product version level. Without that information, the centralized installation manager cannot detect what version is installed, or if the product is installed at all.

Additionally, select one of the WebSphere Extended Deployment products from the Profile Management Tool plug-in when you create a new profile. The middleware agent automatically starts, and the metadata for the new node is updated when you federate the node to the cell using the **addNode** command. For more information on profile creation and augmentation, see [Using the Profile Management tool to create and augment profiles](#).

Update Installer for WebSphere Software

The centralized installation manager installs an appropriate level of the Update Installer on installation targets that it uses to install fix packs and other maintenance. If you manually install the Update Installer on installation targets in a directory location other than <WAS_INSTALL_ROOT>/UpdateInstaller, uninstall this feature by using its uninstallation process. Then, you can use the centralized installation manager to install the Update Installer on installation targets. The centralized installation manager can install the feature either explicitly or automatically when you install fix packs or other maintenance on installation targets.

For more information, see [Downloading the Update Installer for WebSphere Software](#).

Temporary installation locations

After the centralized installation manager successfully completes the installation process on a remote node, it then deletes the installation image files that are located in the temporary location that you specify during the installation process. If the installation is unsuccessful, the files remain in the temporary location for you to use

to determine what caused the installation error. However, you can delete the files after you troubleshoot.

Parent topic: [Using the centralized installation manager to manage your environment](#)

Parent topic: [Using the centralized installation manager to manage your environment](#)

Related tasks:

[Installing packages](#)

[Downloading package descriptors and the associated binary files](#)

[Downloading the Update Installer for WebSphere Software](#)

[Downloading and installing WebSphere Application Server Community Edition](#)

[Managing installation targets](#)

[Using Remote Execution and Access to access remote installation targets](#)

Installing packages

Use the centralized installation manager to install one or more packages to specified installation targets.

Before you begin

To successfully install a package, you must first define an *installation target*, which is the remote workstation on which selected software packages might be installed. By default, all of the workstations that contain nodes that are defined in the cell appear as installation targets.

Important: The centralized installation manager does not install maintenance on the deployment manager. Instead, use the IBM® Update Installer for WebSphere® Software to apply maintenance to the deployment manager. For more information, see [Downloading the Update Installer for WebSphere Software](#).

During the installation process, the wizard prompts you to select an authentication method: user name and password or Secure Shell (SSH) public/private key. If you choose to use the SSH public/private key method, you must first create a pair of keys and install the public key on all the installation targets to successfully complete this task. For more information, see [Installing the Secure Shell public key to access installation targets](#).

Ensure that the latest version of WebSphere Virtual Enterprise is installed and operational on the deployment manager workstation, and the centralized installation manager repository is populated with the installation image for the product packages that you want to install on the remote workstations. For more information on the steps to install the product and the centralized installation manager repository, see [Using ISMP programs to install the product](#).

You must first install the repository to use the features of the centralized installation manager. For example, if you plan to install WebSphere Extended Deployment Compute Grid or WebSphere eXtreme Scale on the remote workstations by using the centralized installation manager, you must also install each product component on the deployment manager workstation. If you did not install the repository during the product installation, you can still set up the repository and install the binary installation images. For more information, see [Using the centralized installation manager repository tool to add products](#).

About this task

The number of steps to complete this task can vary depending on the type of installation package that you choose to install.

Procedure

1. Access the wizard from the administrative console:
 - a. Click **System administration > Centralized Installation Manager > Available installations**.
 - b. Select a package type, which is the type of installation you want to perform. For example, you can choose to complete a product installation, or an installation that applies various types of maintenance files.

Next, select an installation package. If you choose a package that includes available features, select each feature from the **Select one or more features** list. This list does not appear if you choose an installation package that does not include available features.
 - c. Click **Show installation targets** to populate the table with a list of applicable target workstations on which to install the selected software package.
 - d. Select one or more installation targets from the list, and click **Install** to start the Installation wizard.
2. Accept the license agreement. Click **View License Agreement** to read the agreement, accept the terms, and click **Next** to continue.
3. Select an authentication method to access the installation target, and click **Next**. You can choose to use either the Secure Shell (SSH) public/private key method, or the user name and password method to authenticate.
4. Provide the authentication settings, and click **Next**. Depending on the authentication method that you choose in step 3, provide the appropriate user name and password for one or more installation targets, or provide the location of the SSH private key file and password on the deployment manager.

If you choose to authenticate by using the user name and password method, you can provide a common user name and password to access all of the installation targets, or you can configure unique user names and passwords for each target.

5. Specify the installation location and the working location of each installation target, and click **Next**. The installation location is the remote location of the installation target in which the packages are installed. You might also specify additional installation locations if you choose to install certain packages that require more than one installation location, such as WebSphere Application Server Community Edition. When you install packages that require an empty location, the following reminder is displayed in the column:
 - **Specify an empty directory**

The working location specifies the directory on the remote target that contains the installation files.

- Optional: Select a node from the list, augment the profile of the selected node, and click **Next**. By default, all nodes on the target installation location are selected.

This step is presented only when the wizard detects a package that requires profile augmentation. If you choose not to augment the profile, you must manually perform the profile augmentation by issuing the **manageprofiles** command on the target host.

- Specify additional parameters that are required to successfully install the specified package. This step appears only with certain installation package types. Also, some installation packages might have more than one additional parameter that you specify.

Attention: By default, the centralized installation manager selects the appropriate version of the installation package to install based on the version of the operating system. For example, if both 32-bit and 64-bit binaries are included for a certain operating system and processor architecture, the centralized installation manager installs the 64-bit installation binaries on 64-bit operating systems. To override the default behavior and use the 32-bit installation binaries on 64-bit operating systems instead, select the appropriate check box.

- Read the installation summary, and click **Finish** to submit the installation request to the centralized installation manager for processing.

Results

You completed the steps to install one or more packages to the specified target workstations. The centralized installation manager receives your installation request, processes the information that you provided, and then installs the package to the workstations.

What to do next

In the administrative console, check the status of your pending requests on the Installations in Progress panel, and review the log files of your submitted installation requests from the Installation History panel. Read the details about the options that you can use to further monitor the progress of each request.

[Downloading the Update Installer for WebSphere Software](#)

Use the IBM Update Installer for WebSphere Software Version 7.0.0.x to install interim fixes, refresh packs, or fix packs on remote installation targets. Before you can use the centralized installation manager to apply maintenance to your installation targets, download the latest version of the Update Installer.

[Downloading the Update Installer for WebSphere Software](#)

Use the IBM Update Installer for WebSphere Software Version 7.0.0.x to install interim fixes, refresh packs, or fix packs on remote installation targets. Before you can use the centralized installation manager to apply maintenance to your installation targets, download the latest version of the Update Installer.

[Downloading package descriptors and the associated binary files](#)

To enhance your product environment, download additional installation packages and maintenance files to the centralized installation manager repository to install later on installation targets. Use this topic to manage the installation packages and maintenance files that are located in your centralized installation manager repository.

[Downloading and installing WebSphere Application Server Community Edition](#)

Download the binary files for WebSphere Application Server Community Edition, and use the centralized installation manager to install the installation package on your target workstations.

[Installing sudo on target AIX operating systems to apply maintenance as a non-root user](#)

Install and configure sudo, an open source product, on remote AIX® operating systems to allow a non-root user to use sudo and run the **slibclean** command without providing a password. The non-root user can then apply maintenance to the remote AIX operating systems.

Parent topic: [Using the centralized installation manager to manage your environment](#)

Related tasks:

[Downloading package descriptors and the associated binary files](#)

[Managing installation targets](#)

[Installing the Secure Shell public key to access installation targets](#)

[Monitoring requests](#)

[Creating and augmenting profiles](#)

Downloading the Update Installer for WebSphere Software

Use the IBM® Update Installer for WebSphere® Software Version 7.0.0.x to install interim fixes, refresh packs, or fix packs on remote installation targets. Before you can use the centralized installation manager to apply maintenance to your installation targets, download the latest version of the Update Installer.

Before you begin

Before you can install any interim fixes and fix packs on your installation targets, the centralized installation manager repository must already contain the binaries for the Update Installer.

About this task

The Update Installer for WebSphere Software Version 7.0.x is the tool that you use to apply maintenance to your WebSphere Extended Deployment and WebSphere Application Server Network Deployment environments. The centralized installation manager does not install maintenance on the deployment manager. Instead, use the following steps to download the Update Installer, which you can then use to apply maintenance to the deployment manager. For more information about the Update Installer, see the [IBM Update Installer Support site](#).

Procedure

1. In the administrative console, click **System administration > Centralized Installation Manager > Add or Remove Packages**.
2. Click **Update Installer for WebSphere Software** in the table that displays the list of installation packages.
3. Select one or more operating systems, and click **Download**.
4. Review the summary, and click **Download** to start downloading the Update Installer binary files for the selected operating systems.

You can monitor the download status of the files from the Add or Remove Packages panel after the download process begins. Click the icon to refresh the contents of the table if necessary.

What to do next

From the Available installations panel, use the centralized installation manager to install the Update Installer in one of two ways:

- Select **Refresh pack, fix pack, or maintenance tool** from the **Package type** list. From the **Installation package** list, select the specific maintenance that you want to apply, and click **Show installation targets**. A list of applicable installation targets is displayed. Select one or more targets, and click **Install**.

The centralized installation manager then checks if the Update Installer is already installed on the installation target. If no existing installation exists, the centralized installation manager automatically selects the appropriate version of the Update Installer to install based on the version of the operating system on which WebSphere Application Server Network Deployment is installed. For example, if WebSphere Application Server Network Deployment is installed on a 32-bit operating system, the centralized installation manager selects and installs the 32-bit version of the Update Installer.

- Select **Refresh pack, fix pack, or maintenance tool** from the **Package type** list. Select **Update Installer for WebSphere Software** as the type of installation package, and click **Show installation targets**. A list of applicable installation targets on which to install the Update Installer is displayed. Select one or more targets, and click **Install**.

[Installing sudo on target AIX operating systems to apply maintenance as a non-root user](#)

Install and configure sudo, an open source product, on remote AIX® operating systems to allow a non-root user to use sudo and run the **slibclean** command without providing a password. The non-root user can then apply maintenance to the remote AIX operating systems.

[Installing interim fixes](#)

Install selected interim fixes to specific installation targets to update your product environment. The IBM Update Installer for WebSphere Software is the tool that you use to apply maintenance to your WebSphere Extended Deployment and WebSphere Application Server Network Deployment environments.

[Installing refresh packs or fix packs](#)

Install recommended fix packs or refresh packs to specific installation targets to update your product environment. Use the IBM Update Installer for WebSphere Software to apply maintenance to your WebSphere Virtual Enterprise and WebSphere Application Server Network Deployment environments.

Parent topic: [Installing packages](#)

Related tasks:

[Downloading package descriptors and the associated binary files](#)
[Manually adding files to the repository](#)

Downloading package descriptors and the associated binary files

To enhance your product environment, download additional installation packages and maintenance files to the centralized installation manager repository to install later on installation targets. Use this topic to manage the installation packages and maintenance files that are located in your centralized installation manager repository.

Before you begin

You must first install the centralized installation manager repository, and add one or more product packages to the repository, on the host workstation. Complete this task during the installation process of WebSphere® Virtual Enterprise. For more information, see [Using ISMP programs to install the product](#).

Alternatively, you can use the centralized installation manager repository tool to add one or more product packages to the repository. This tool is automatically installed when you install WebSphere Virtual Enterprise only. For further details about the repository tool, see [Using the centralized installation manager repository tool to add products](#).

About this task

In the administrative console, download the descriptor files, and any associated binary files, of new or additional installation packages to the centralized installation manager repository. You can selectively download only the binary files of the platforms that you might need, or download specific maintenance files from the IBM® support Web site. The following list provides the four types of installation packages:

Product installation

This type includes any of the three WebSphere Extended Deployment products: WebSphere Virtual Enterprise, WebSphere Extended Deployment Compute Grid, or WebSphere eXtreme Scale. The descriptor and binary files for this installation type are not available to download, because the files are included during the product installation.

Refresh packs or fix packs

You can download the binary files for this type based on specific platforms.

Maintenance tool

This type includes the IBM Update Installer for WebSphere Software, which is the tool that you use to apply maintenance to your WebSphere Virtual Enterprise and WebSphere Application Server Network Deployment environments. Before you can use the centralized installation manager to apply maintenance to your remote workstations, you must download the latest level of the Update Installer.

Note: You must use the Update Installer to install maintenance on the deployment manager.

First apply maintenance to the deployment manager before you apply maintenance to the cell. Begin by using the centralized installation manager to download the required files, which are then stored in the repository that you specify during the initial installation process. Alternatively, you can find the repository location in the `CENTRALIZED_INSTALL_REPOSITORY_ROOT` property in the `WAS_HOME/properties/cimgr.props` file.

Interim fix

Search for the identifying Authorized Program Analysis Report (APAR) number, and click **Search** to download a specific interim fix.

To access this page, click **System administration > Centralized Installation Manager > Add or remove packages**.

Procedure

1. Click **Add Packages** to download a new installation package descriptor to the centralized installation manager repository if the descriptor is not included in the table. The Download Descriptors panel is displayed.

Tip: Ensure that the descriptor file for the type of package that you choose is not included as part of the product installation. The installation package descriptors that are included during the product installation are provided in the following list:

- Maintenance for WebSphere Application Server Network Deployment
- Maintenance for WebSphere Extended Deployment
- Update Installer for WebSphere Software
- WebSphere Virtual Enterprise, if the package is installed
- WebSphere Extended Deployment Compute Grid, if the package is installed

2. Select one or more descriptor files from the list, and click **Download**.

After the descriptor file is downloaded, you can view it from the Installation Packages panel. If the downloaded file does not appear immediately, click the refresh icon to refresh the contents of the table.

To download the binary files for the installation packages in the preceding list, click the name of the descriptor, and proceed to step 3. To download additional package descriptors from the IBM support Web site, click **Add packages**.

3. Download the binary files from the Installation Packages panel. You can download the associated binary files of the specific descriptor file that you just downloaded, or you can also download the binary files of another package type, which might already contain the descriptor file.

Determine the type of installation package to download by the viewing the descriptions of each type in the table. The steps to download the binary files differ, depending on the package type.

- To download the binary files for a **Refresh pack, fix pack, or maintenance tool** package type, which includes the Update Installer, complete the following steps:
 - a. Click the name of the package in the table. A new panel is then displayed.
 - b. Select one or more platforms in the table, and click **Download**.
 - c. Click **Download** on the confirmation page to start downloading the binaries. After the download process begins, the previous panel is displayed, from which you can check the download status of the files. Click the refresh icon to refresh the contents of the table if necessary.
- To download the binary files for an **Interim fix** package type, complete the following steps:
 - a. Click the name of the package in the table. A new panel is then displayed.
 - b. Click **Add Files** to navigate to the Download Files panel.
 - c. Either type the specific APAR, and click **Search** to navigate directly to the corresponding FTP location. You can also specify the ftp URL directly, and click **Go** from the Download Options section.
 - d. Click the APAR number, select the individual maintenance files that are contained in the directory, and click **Download**. The binary files are then downloaded to the centralized installation manager repository.
 - e. Click **Download** on the confirmation page to start downloading the binaries. After the download process begins, the previous panel is displayed, from which you can check the download status of the files. Click the refresh icon to refresh the contents of the table if necessary.

Results

The Update Installer repository now contains maintenance files to install later on the remote workstations.

What to do next

Install interim fixes, refresh packs, or other maintenance files to specific installation targets to update your product environment.

[Manually adding files to the repository](#)

To use the centralized installation manager to download the IBM Update Installer for WebSphere Software and maintenance files to the repository, along with additional descriptors to the deployment manager workstation, the deployment manager must have access to the public IBM Web sites. When the deployment manager workstation does not have internet access, you must first download the descriptors and files to a separate workstation that has internet access, and then manually transfer those files to the centralized installation manager repository.

[Installing interim fixes](#)

Install selected interim fixes to specific installation targets to update your product environment. The IBM Update Installer for WebSphere Software is the tool that you use to apply maintenance to your WebSphere Extended Deployment and WebSphere Application Server Network Deployment environments.

[Installing refresh packs or fix packs](#)

Install recommended fix packs or refresh packs to specific installation targets to update your product environment. Use the IBM Update Installer for WebSphere Software to apply maintenance to your WebSphere Virtual Enterprise and WebSphere Application Server Network Deployment environments.

[Manually adding files to the repository](#)

To use the centralized installation manager to download the IBM Update Installer for WebSphere Software and maintenance files to the repository, along with additional descriptors to the deployment manager workstation, the deployment manager must have access to the public IBM Web sites. When the deployment manager workstation does not have internet access, you must first download the descriptors and files to a separate workstation that has internet access, and then manually transfer those files to the centralized installation manager repository.

Parent topic: [Installing packages](#)

Related tasks:

[Installing interim fixes](#)

[Installing refresh packs or fix packs](#)

Installing interim fixes

Install selected interim fixes to specific installation targets to update your product environment. The IBM® Update Installer for WebSphere® Software is the tool that you use to apply maintenance to your WebSphere Extended Deployment and WebSphere Application Server Network Deployment environments.

Before you begin

- Define an *installation target*, which is the remote workstation on which selected software packages might be installed. By default, all of the workstations that contain nodes that are defined in the cell appear as installation targets. During the installation process, the wizard prompts you to select an authentication method: user name and password or Secure Shell (SSH) public/private key. If you choose to use the SSH public/private key method, you must first create a pair of keys and install the public key on all the installation targets to successfully complete this task.
- Download the latest version of the Update Installer to the centralized installation manager repository. The following message is displayed if you attempt to install an interim fix before you download the Update Installer:

```
The installation binary files required for the install_package_name or its dependent package Update Installer for WebSphere Application Server for workstation_operating_system do not exist.
```

You are not required to install the Update Installer after you download it. If the Update Installer is not installed on the installation target, the centralized installation manager installs the Update Installer automatically before installing any interim fixes, fix packs, or refresh packs. For more information, see [Downloading the Update Installer for WebSphere Software](#)

- Download the binary files for an interim fix package type. The descriptors for an interim fix package type, however, are installed when you install WebSphere Virtual Enterprise. These specific descriptors are included to apply the following types of updates:
 - Maintenance for WebSphere Application Server Network Deployment
 - Maintenance for WebSphere Virtual Enterprise

For details on how to locate the descriptor and associated files, see [Downloading package descriptors and the associated binary files](#).

About this task

Complete the following procedure to install interim fixes for both WebSphere Virtual Enterprise and WebSphere Application Server Network Deployment.

Procedure

1. Start the wizard from the administrative console:
 - a. Click **System administration > Centralized Installation Manager > Available installations**.
 - b. Select **Interim fix** as the package type. Next, select the maintenance installation package.

If you previously downloaded any interim fixes by using the **Add or remove packages** function, each interim fix is displayed in the list that is displayed in the **Select one or more maintenance packs** field. Select one or more interim fixes from this list.
 - c. Click **Show installation targets** to populate the table with a list of applicable target workstations on which to install the selected interim fixes. After you select one or more installation targets, click **Install** to start the installation wizard.
2. Read and accept the license agreement.
3. Select an authentication method to access the installation target, and click **Next**. You can select to either use the Secure Shell (SSH) public/private key method or the user name and password method to authenticate.

Depending on the authentication method that you choose, provide the appropriate user name and password for one or more installation targets, or provide the location of the SSH private key file and password on the deployment manager. If you choose to authenticate by using the user name and password method, you can provide a common user name and password to access all of the installation targets, or you can configure unique user names and passwords for each target.

4. Verify the installation and the working locations of each installation target, and click **Next**.

The installation location is the remote location of each installation target in which the packages are installed. The working location specifies the directory on the remote target where the files are sent before the package is installed in the specified location.

5. Read the installation summary, and click **Finish** to submit the installation request to the centralized installation manager for processing.

Results

Your installation request is sent to the centralized installation manager for processing. Check the status of your request from the Installations in progress panel in the administrative console. You can review the log files for each of the installation requests that you submit from the Installation history panel.

What to do next

Update the deployment manager workstation with same interim fixes that you install on the product. For more information, see [Using the Update Installer to install maintenance packages](#).

Parent topic: [Downloading the Update Installer for WebSphere Software](#)

Parent topic: [Downloading package descriptors and the associated binary files](#)

Related tasks:

[Using the interim fix tool to check for maintenance packages](#)

[Downloading package descriptors and the associated binary files](#)

[Managing installation targets](#)

Related reference:

[Centralized installation manager administrative tasks](#)

Installing refresh packs or fix packs

Install recommended fix packs or refresh packs to specific installation targets to update your product environment. Use the IBM® Update Installer for WebSphere® Software to apply maintenance to your WebSphere Virtual Enterprise and WebSphere Application Server Network Deployment environments.

Before you begin

- Define an *installation target*, which is the remote workstation on which selected software packages might be installed. By default, all of the workstations that contain nodes that are defined in the cell appear as installation targets. During the installation process, the wizard prompts you to select an authentication method: user name and password or Secure Shell (SSH) public/private key. If you choose to use the SSH public/private key method, you must first create a pair of keys and install the public key on all the installation targets to successfully complete this task.
- Download the latest version of the Update Installer. You are not required to install the Update Installer after you download it. If the Update Installer is not installed on the installation target, the centralized installation manager installs the Update Installer automatically before installing any interim fixes, fix packs, or refresh packs. For more information, see [Downloading the Update Installer for WebSphere Software](#)
- Download the installation package descriptor and binary files for a refresh pack or fix pack. For more information, see [Downloading package descriptors and the associated binary files](#).

About this task

Any interim fixes that you previously installed on the remote targets will be uninstalled by the Update Installer prior to installing the refresh pack or fix pack. If the refresh pack or fix pack does not include the official fixes that were included in the removed interim fixes, you must reinstall the interim fixes after you install the refresh pack or fix pack.

Complete the following steps to install recommended fix packs or refresh packs for both WebSphere Virtual Enterprise and WebSphere Application Server Network Deployment.

Procedure

1. Start the wizard from the administrative console:
 - a. Click **System administration > Centralized Installation Manager > Available installations**.
 - b. Select **Refresh pack, fix pack, or maintenance tool** as the package type. Next, select the specific installation package that contains the refresh pack or fix pack that you want to install on the remote workstations.
 - c. Click **Show installation targets** to populate the table with a list of applicable target workstations on which to install the selected package. After you select one or more installation targets, click **Install** to start the installation wizard.
2. Read and accept the license agreement.
3. Select an authentication method to access the installation target, and click **Next**. You can select to either use the Secure Shell (SSH) public/private key method or the user name and password method to authenticate. Provide your authentication information, and click **Next**.

Depending on the authentication method that you choose, provide the appropriate user name and password for one or more installation targets, or provide the location of the SSH private key file and password on the deployment manager. If you choose to authenticate by using the user name and password method, you can provide a common user name and password to access all of the installation targets, or you can configure unique user names and passwords for each target.

4. Verify the location of the product installation, and click **Next**.
5. Read the installation summary, and click **Finish** to submit the installation request to the centralized installation manager for processing.

Results

Your installation request is sent to the centralized installation manager for processing. Check the status of your request from the Installations in progress panel. You can review the log files for each of the installation requests that you submit from the Installation history panel.

What to do next

Update the deployment manager workstation with the same refresh pack or fix pack that you install. For more information, see [Using the Update Installer to install maintenance packages](#).

Parent topic: [Downloading the Update Installer for WebSphere Software](#)

Parent topic: [Downloading package descriptors and the associated binary files](#)

Related tasks:

[Downloading package descriptors and the associated binary files](#)

[Manually adding files to the repository](#)

[Managing installation targets](#)

Manually adding files to the repository

To use the centralized installation manager to download the IBM® Update Installer for WebSphere® Software and maintenance files to the repository, along with additional descriptors to the deployment manager workstation, the deployment manager must have access to the public IBM Web sites. When the deployment manager workstation does not have internet access, you must first download the descriptors and files to a separate workstation that has internet access, and then manually transfer those files to the centralized installation manager repository.

Before you begin

You must first install the centralized installation manager repository, and add one or more product components to the repository, on the host workstation. Complete this task during the installation process of WebSphere Virtual Enterprise. For more information, see [Using ISMP programs to install the product](#).

Alternatively, you can use the repository tool to add one or more product components to the repository. This tool is automatically installed when you install the WebSphere Virtual Enterprise product only. See [Using the centralized installation manager repository tool to add products](#) for more information.

About this task

The Update Installer for WebSphere Software and the maintenance files that are required by the centralized installation manager to remotely install maintenance are the same tool and files that are used to apply maintenance to the deployment manager workstation. Complete the steps to download the Update Installer and maintenance files as normal but without using the centralized installation manager.

The repository consists of various directories in which the installation image for the Update Installer and maintenance files is located. The following list contains the directories, along with the URL to use to download additional descriptors:

UPDI70

Includes the 7.0.0.*-WS-UPDI-*.zip file that contains the installation image for the Update Installer. Download the latest version of the Update Installer from the following URL:

<ftp://ftp.software.ibm.com/software/websphere/appserv/support/tools/UpdateInstaller/7.0.x/>.

1. Navigate to the URL from a workstation that has internet access.
2. Select the folder for the operating system and architecture you need, and download the latest 7.0.0.*-WS-UPDI-*.zip file to your workstation.
3. Repeat the steps for all impacted operating systems and architectures.
4. Transfer the compressed files to the UPDI70 directory on the deployment manager workstation.

ND61Updates

Includes all the interim fixes for WebSphere Application Server Network Deployment Version 6.1. Copy the .pak files for all your WebSphere Application Server Network Deployment interim fixes to this directory. You can also remove any .pak files from this directory that you no longer need.

ND61FP n

Includes various .pak files that make up a specific fix pack for WebSphere Application Server Network Deployment. For WebSphere Application Server Network Deployment Version 6.1 Fix Pack 11, for example, copy the 6.1.0-WAS-WAS-WinX32-FP0000011.pak file, the 6.1.0-WAS-WASSDK-WinX32-FP0000011.pak file, and the 6.1.0-WAS-WASWebSvc-WinX32-FP0000011.pak file to the ND61FP11 directory.

XD61Updates

Includes all the interim fixes for WebSphere Virtual Enterprise. Copy the .pak files for all your WebSphere Virtual Enterprise interim fixes to this directory. You can also remove any .pak files from this directory that you no longer need.

XD61FP n

Includes the .zip file that contains the contents of a specific fix pack. For WebSphere Virtual Enterprise Version 6.1 Fix Pack 1, for example, download and copy the 6.1.0-WAS-WXD-FP0000001.zip file to the XD61FP1 directory. The URL to use to download the .zip file for Fix Pack 1 is

<ftp://ftp.software.ibm.com/software/websphere/extended/support/fixpacks/6.1.0.1/MULTIPLATFORM>.

XD61RP n

Includes the .zip file that contains the contents of a specific refresh pack. For WebSphere Virtual Enterprise Version 6.1 Refresh Pack 1, for example, download and copy the 6.1.0-WAS-WXD-RP0000001.zip file to the XD61RP1 directory. The URL to use to download the .zip file for Refresh Pack 1 is

<ftp://ftp.software.ibm.com/software/websphere/extended/support/fixpacks/6.1.1.0/MULTIPLATFORM>.

Add the required files to the centralized installation manager repository to later install recommended fix packs or refresh packs to specific installation targets. Download the descriptors and files to a separate workstation that has internet access, and then manually transfer those files to the centralized installation manager repository.

Procedure

1. In the administrative console, click **System administration > Centralized Installation Manager > Add or remove packages**. Click **Add Packages** and the Download descriptors panel is displayed.

2. Determine the location of the FTP site from which you download the descriptors. Expand **Download Options** to view the URL that is used by the centralized installation manager. The URL format is ftp://ftp.software.ibm.com/software/websphere/extended/support/cim/xd61_yyyymmdd.

If the deployment manager workstation does not have internet access, an error message is displayed indicating that the host name, ftp.software.ibm.com, is not known.

3. Use the URL in step 2 to download the available descriptors from a separate workstation that has internet access.
4. Transfer the downloaded descriptors to the *install_root/properties/xdcimgr* directory on the deployment manager workstation.

Results

The centralized installation manager repository now contains maintenance files to install later on the remote workstations.

Parent topic: [Downloading package descriptors and the associated binary files](#)

Related tasks:

[Downloading the Update Installer for WebSphere Software](#)

[Downloading package descriptors and the associated binary files](#)

[Installing refresh packs or fix packs](#)

Downloading and installing WebSphere Application Server Community Edition

Download the binary files for WebSphere® Application Server Community Edition, and use the centralized installation manager to install the installation package on your target workstations.

Before you begin

Ensure that the latest version of WebSphere Virtual Enterprise is installed and the centralized installation manager repository is set up on the deployment manager workstation. For more information, see [Using ISMP programs to install the product](#).

The remote installation target must contain a middleware node that is augmented for the product. The middleware node can be either a WebSphere Application Server Network Deployment node or a node that has the middleware agent installed. Ensure that you meet this requirement before you begin downloading and installing WebSphere Application Server Community Edition. To install the middleware agent on a node that does not contain WebSphere Application Server, see [Installing packages](#).

About this task

Use the centralized installation manager to remotely install WebSphere Application Server Community Edition. This particular installation package contains one of two features that you select during the installation process: **Server only** or **Server and IBM software development kit (SDK), Java 2 Technology Edition**.

Because the product does not contain the required binary files for this particular installation package, you must first download and copy the files to the centralized installation manager repository. Complete the following steps to download the binary files, and then use the centralized installation manager to install the package to your installation targets.

Procedure

1. Download the binary files for WebSphere Application Server Community Edition from [developerWorks®](#). Click **Multiplatform** to begin the download process.
 - a. Log in as a registered user with your IBM® ID and password. Complete the required fields by providing your contact information, selecting the appropriate privacy settings, and viewing the license agreement. Click **I confirm** to continue.
 - b. Select one or more of the available installation bundles, and click **Download now**.

Select the **Server only** bundles if you already installed the latest version of the software developer kit on the target workstation. These bundles are available with the following operating systems:

- UNIX 32-bit
- Windows Intel 32-bit

Select any one of the **Server and IBM SDK** options to download WebSphere Application Server Community Edition with the IBM SDK, Java™ 2 Technology Edition and the application server. This option is available with the following operating systems:

- AIX® PowerPC® 32-bit
- Linux Intel 32-bit
- Linux Power PC 32-bit
- Linux Intel or AMD 64-bit
- Solaris Operating Environment SPARC 32-bit
- Windows Intel 32-bit

Attention: The centralized installation manager cannot uninstall the IBM SDK for the Windows operating system that is installed with the **Server and SDK for Windows** bundle. The centralized installation manager also cannot uninstall the IBM SDK for the Windows operating system by using the Windows Installer CleanUp Utility and manually deleting the SDK. Instead, you must uninstall the SDK manually from the target host by using the Windows **Add or Remove Programs** control panel function.

2. Copy the downloaded files to the following directory:
 - For WebSphere Application Server Community Edition Version 2.0, copy the files to the *CIM_repository_root/WASCE20* directory.
 - For WebSphere Application Server Community Edition Version 2.1, copy the files to the *CIM_repository_root/WASCE21* directory.
3. Install WebSphere Application Server Community Edition on the target host.
 - a. From the Available Installations panel in the administrative console, select the package type, the installation package, and one of the two accompanying features that are available with the installation package:
 - Package type: **Product install**
 - Installation package: **WebSphere Application Server Community Edition**
 - Feature: **Server only** or **Server and IBM software developer kit (SDK), Java 2 Technology Edition**

You can install the **Server only** feature multiple times, and distribute the same SDK between it and the **Server and IBM software developer kit (SDK), Java 2 Technology Edition** feature on the same target host.

After you make your selections and click **Show Installation Targets**, a table that contains a list of applicable target workstations on which to install the installation package is displayed.

- b. Select one or more installation targets from the list, and click **Install** to start the Installation wizard.
- c. Accept the license agreement as the first step of the wizard. Click **View License Agreement** to read the agreement, and accept the terms. Click **Next** to continue.
- d. Select an authentication method to access the installation target, and click **Next**. You can choose to use either the Secure Shell (SSH) public/private key method or the user name and password method to authenticate.
- e. Provide the authentication settings, and click **Next**. Depending on the authentication method that you choose in step 3, provide the appropriate user name and password for one or more installation targets, or provide the location of the SSH private key file and password on the deployment manager.

If you choose to authenticate by using the user name and password method, you can provide a common user name and password to access all of the installation targets, or you can configure unique user names and passwords for each target.

- f. Specify the installation location and the working location of each installation target, and click **Next**.

The installation location is the remote location of the installation target in which the WebSphere Application Server Community Edition server is installed.

The working location specifies the directory on the remote target that contains the installation files.

- g. Specify the installation location of a supported Java 2 environment on the target host if you choose **Server only** in step a. If you do not specify the path to the location, the installation program attempts to locate a supported Java 2 environment. However, the installation fails if the program cannot locate the existing environment.

If you choose **Server and IBM software developer kit (SDK), Java 2 Technology Edition** in step a, review the installation location of the SDK for the bundle, and click **Next**.

The following list contains the default installation location for each operating system:

- AIX PowerPC 32-bit: /usr/java5
- Linux Intel 32-bit: /opt/ibm/java2-i386-50
- Linux PowerPC 32-bit: /opt/ibm/java2-ppc-50
- Linux Intel or AMD 64-bit: /opt/ibm/java2-x86_64-50
- Solaris Operating Environment SPARC 32-bit: \${INSTALL_LOCATION}/_jvm
- Windows Intel 32-bit: C:/Program Files/IBM/Java50

A remote host can contain only one copy of the SDK for each of the preceding operating systems, except for the Solaris operating system. The SDK for the Solaris operating system is installed as a private Java Virtual Machine (JVM) in the installation directory for WebSphere Application Server Community Edition. Therefore, you can install the **Server and IBM SDK for Solaris/SPARC** bundle multiple times on the same host in different directories.

- h. Specify additional parameters that are required to successfully install the installation package. Select a node on each of the target hosts to designate as the manager of the WebSphere Application Server Community Edition server. Next, select the check box to disable a prerequisite check on each operating system. The prerequisite check is enabled by default.
- i. Read the installation summary, and click **Finish** to submit the installation request to the centralized installation manager for processing.

Alternatively, issue the **installSoftware** command in the wsadmin tool to install WebSphere Application Server Community Edition. Specify certain parameters depending on which feature you choose to install with the installation package. For more information, see [Centralized installation manager administrative tasks](#).

Attention: The following error message might be displayed on the Installation history detail panel when you install WebSphere Application Server Community Edition on either Solaris operating systems or AIX operating systems:

```
XCIM0199W: Installation completed but errors were detected. Check logs
for error messages.
..... log1 .....
..... log2 .....
```

After you check one of the log files, ignore the error if it is similar to one of the following examples:

Solaris

```
XCIM01661: Standard error from command:
wasce_setup-2.*-unix.bin -is:silent....
Must be attached to terminal for 'am I' option
Must be attached to terminal for 'am I' option
```



```
XCIM01661: Standard error from command:  
wasce_setup-2.*-unix.bin -is:silent...  
who: 0551-012 The process is not attached to a terminal.  
Do not run who am i as a background process.
```

Results

The centralized installation manager now contains a copy of the required binary files for installing WebSphere Application Server Community Edition. The centralized installation manager receives your installation request, processes the information that you provided, and then installs the package to the target workstation.

What to do next

In the administrative console, check the status of your pending requests on the Installations in Progress panel, and review the log files of your submitted installation requests from the Installation History panel.

After the installation completes, you can create managed WebSphere Application Server Community Edition servers. See [Creating complete life-cycle WebSphere Application Server Community Edition servers and dynamic clusters](#)

Parent topic: [Installing packages](#)

Related tasks:

[Downloading package descriptors and the associated binary files](#)

[Managing installation targets](#)

[Creating and augmenting profiles](#)

Installing sudo on target AIX operating systems to apply maintenance as a non-root user

Install and configure sudo, an open source product, on remote AIX® operating systems to allow a non-root user to use sudo and run the **slibclean** command without providing a password. The non-root user can then apply maintenance to the remote AIX operating systems.

Before you begin

Download the AIX installp image for the rpm package manager for POWER®. See the [IBM® AIX Toolbox Download Web site](#) for more information.

About this task

As the root user, install and configure sudo, which is an open source product, on the target AIX operating systems before a non-root user can use the centralized installation manager to install or uninstall maintenance on AIX operating systems. Complete the installation and configuration operations locally as the root user of the AIX operating systems without using centralized installation manager. You are required to complete this task only once.

Procedure

1. Download sudo from the IBM AIX Toolbox Download Web site. See the [IBM AIX Toolbox Download Web site](#) for more information.
2. Issue the following command to install sudo:

```
rpm -i sudo-1.6.7p5-3.aix5.1.ppc.rpm
```

3. Authorize a non-root user ID, which you specify, to run the **slibclean** command as a root user without providing a password. Issue the **visudo** command to add the following entry to the `/etc/sudoers` configuration file:

```
user_id ALL = NOPASSWD: /usr/sbin/slibclean
```

4. Log in with the specified user ID, and issue the **sudo -l** command.

If successful, a message that is similar to the following example is displayed:

```
User user_id may run the following commands on this host:
```

```
(root) NOPASSWD: /usr/sbin/slibclean
```

If you do not have sudo installed, or sudo is installed but not configured correctly for the specified user ID, error messages are displayed.

What to do next

Download the latest version of the Update Installer for WebSphere® Software and the binary files for the fix packs to your centralized installation manager repository.

Parent topic: [Downloading the Update Installer for WebSphere Software](#)

Parent topic: [Installing packages](#)

Related tasks:

[Using ISMP programs to install the product](#)

[Installing refresh packs or fix packs](#)

[Installing interim fixes](#)

Uninstalling packages

Use the centralized installation manager to uninstall a previously installed package from remote workstations.

Before you begin

The wizard prompts you to select an authentication method: user name and password or Secure Shell (SSH) public/private key. If you choose to use the SSH public/private key method, you must first create a pair of keys and install the public keys on all the installation targets.

About this task

The number of steps for this task can vary depending on the type of installation package that you choose to uninstall.

Procedure

1. Access the wizard from the administrative console:
 - a. Click **System administration > Centralized Installation Manager > Available installations**.
 - b. Select a package type and an installation package, and click **Next**. Depending on the package type that you choose, you can also select to uninstall features or maintenance packages that accompany the installation package.
 - c. Click **Show uninstallation targets** to populate the table with a list of applicable target workstations from which to remove the selected software package. After you select one or more uninstallation targets, click **Uninstall** to start the wizard.
2. Select an authentication method to access the installation target, and click **Next**. You can choose to use the Secure Shell (SSH) public/private key method or the user name and password method to authenticate.
3. Provide the authentication settings, and click **Next**. Depending on the authentication method that you choose in step 2, provide the appropriate user name and password for one or more installation targets, or provide the location of the SSH private key file and password on the deployment manager.

If you choose to authenticate by using the user name and password method, you can provide a common user name and password to access all of the installation targets, or you can configure unique user names and passwords for each target.

4. Specify the installation location of each installation target, and click **Next**.

The installation location is the remote location of the installation target in which the packages are installed.

5. Read the summary, and click **Finish** to submit the request to the centralized installation manager for processing.

Results

Your uninstallation request is sent to the centralized installation manager for processing. Check the status of your request from the Installations in progress panel in the administrative console.

What to do next

Review the log files for each of the uninstallation requests that you submit from the Installation history panel in the administrative console.

Parent topic: [Using the centralized installation manager to manage your environment](#)

Related tasks:

[Downloading package descriptors and the associated binary files](#)

[Managing installation targets](#)

[Installing the Secure Shell public key to access installation targets](#)

Monitoring requests

After you submit one or more requests to the centralized installation manager, you can monitor the progress of and view specific details about each installation and uninstallation request.

About this task

In the administrative console, the Installations in Progress and Installation History panels provide you with information on the status of the installation and uninstallation requests that you submit to the centralized installation manager for processing. However, each panel provides you with different options for using that information to monitor and manage your requests. Installations in Progress provides you with options to view and monitor the progress of each request, and you can also cancel any pending requests from this panel. From the Installation History panel, you can monitor the completion status, delete the history records, and access the error messages and log files of each incomplete request.

Procedure

- Complete the following steps to monitor the progress of requests:
 1. Click **System administration > Centralized Installation Manager > Installations in Progress** in the administrative console.
 2. Review the table for specific details about each request, which are described in the following list:
 - **Host name** specifies the name of the workstation on which the request is performed.
 - **Operation** specifies the type of request, such as install, uninstall, or install SSH public key.
 - **Package and Features** specifies the name of the software package and any accompanying features that make up the installation request.
 - **Creation time** specifies the date and time you submit the request.
 - **Status** specifies the progress of the request.
 3. Select one or more rows from the table, and click **Cancel Pending Request** to cancel only the requests that are not yet started.
 4. Review the confirmation panel, and click **OK** to return to the Installations in Progress panel.
- Complete the following steps to view the completion status and details of requests:
 1. Click **System administration > Centralized Installation Manager > Installation History** in the administrative console.
 2. Review the table for specific details about each request.

The table that is displayed lists the same descriptive information as the table on the Installations in Progress panel, except that the status is displayed as one of the following completion types:

- Succeeded
- Failed
- Installation succeeded, Configuration failed

3. Click **Remove** to delete the history records from the deployment manager.

Review the confirmation panel, and click **Remove** again.

4. Click **View details** to view the log files and any error messages.

A new page now displays any errors that might occur, and the links to the actual log content. The error messages always display before the links.

- Click the specific link to read the content of a log file.

If you previously deleted the log files from the remote workstation, an error message is displayed. If you replace existing log files with new ones, the updated content is displayed.

- Click **OK** to return to the Installation History panel.

What to do next

Return to the Available Installations panel to resubmit a canceled or failed request, or submit a new request to the centralized installation manager.

In the case of certain failed requests, you might need to correct the error on the remote workstations before resubmitting the requests. For installations that are partially successful, examine the logs to correct the problem. You can manually complete the remaining installation steps. With this option, you do not need to resubmit the requests. Alternatively, if the failure state of the request is closer to the starting state, you can return the workstation to the starting state before you resubmit the requests.

Parent topic: [Using the centralized installation manager to manage your environment](#)

Related tasks:

[Installing packages](#)

[Managing installation targets](#)

Managing installation targets

Use this topic to add or remove an installation target, which is the workstation on which selected software packages might be installed. You can also edit the configuration of an existing installation target, and store the administrative ID and password of each target for later use when installing or uninstalling packages.

Before you begin

You must first create an installation target to install one or more software packages on your workstations. By default, all of the workstations containing nodes that are defined in the cell display as installation targets.

About this task

From the Installation Targets panel in the administrative console, you can add additional installation targets that are located outside of the cell. For example, you can install the middleware agent on a node that is running other middleware servers that were created outside of the product cell by adding the remote workstation as a new installation target. Other tasks that you can complete to further manage your installation targets include removing installation targets, editing the configuration of installation targets, and installing a Secure Shell (SSH) public key on installation targets. To access this page, click **System administration > Centralized Installation Manager > Installation targets**.

Procedure

- To add additional installation targets that are located outside of the cell, click **Add Installation Target**. The configuration page is displayed next.
 1. Provide the host name and platform of the installation target, and optionally specify the administrative ID and password, which the centralized installation manager later uses to install one or more packages on the installation target.
 2. Optional: Click **Test Connection** to test the connection using the administrative ID and password that you provide.

Click **OK** after you specify the configuration settings to return to the Installation Targets panel. The new installation target is displayed in the table.

- To remove existing installation targets, select one or more from the table, and click **Remove Installation Target**. The confirmation page then lists each selected installation target.
 1. Click **Remove** to complete the action, and to return to the Installation Targets panel.
- To edit the configuration settings of an existing installation target, click the host name. The configuration page is displayed.
 1. Edit any of the configuration settings that display on the page, which are the same fields that you complete to configure a newly created installation target. Click **OK** when you complete your changes to return to the Installation Targets panel. Any changes that you make are displayed in the table.
- To install a Secure Shell (SSH) public key on specific installation targets, select one or more targets from the table, and click **Install SSH Public Key**.

As a result, the wizard is then launched to complete the SSH public key installation process. The actual wizard steps are further explained in [Installing the Secure Shell public key to access installation targets](#). Refer to this task for the detailed wizard instructions, and for more information on accessing your remote workstations by using the SSH public/private key pair authentication method.

What to do next

You can now begin installing packages to specific installation targets. For more information, see [Installing packages](#).

[Using Remote Execution and Access to access remote installation targets](#)

To use Remote Execution and Access to remotely access installation targets, you must first follow certain target-specific requirements. WebSphere Virtual Enterprise utilizes Remote Execution and Access Version 2.3.0.1 to provide new management features, such as starting and stopping node agent and middleware processes, and initiating installations of product packages and maintenance for distributed platforms from the administrative console.

[Installing the Secure Shell public key to access installation targets](#)

To use Secure Shell (SSH) public/private key as an authentication method to access remote installation targets, you must first install the public key of a public/private key pair on installation targets. You can then securely connect to installation targets by using the corresponding private key.

[Running Cygwin as root to use SSH on target Windows operating systems](#)

To use a Secure Shell (SSH) public/private key-pair to access target Windows operating systems, change the ownership of Cygwin files and directories to root.

[Updating Secure Shell public key installation records](#)

If you install the Secure Shell (SSH) public key on installation targets using methods other than the centralized installation manager, use the **updateKeyInstallationRecords** administrative task

command to update the SSH public key records that the centralized installation manager maintains.

Parent topic: [Using the centralized installation manager to manage your environment](#)

Related tasks:

[Installing the Secure Shell public key to access installation targets](#)

[Installing packages](#)

[Uninstalling packages](#)

Related reference:

[Centralized installation manager administrative tasks](#)

Using Remote Execution and Access to access remote installation targets

To use Remote Execution and Access to remotely access installation targets, you must first follow certain target-specific requirements. WebSphere® Virtual Enterprise utilizes Remote Execution and Access Version 2.3.0.1 to provide new management features, such as starting and stopping node agent and middleware processes, and initiating installations of product packages and maintenance for distributed platforms from the administrative console.

Before you begin

- On target Windows Vista operating systems, verify the Remote Registry in Windows Services is started and port 445 is opened in the firewall.

About this task

There are certain conditions, which are based on the operating systems installed on the target workstations, that you are required to meet to use Remote Execution and Access. The following sections briefly describe the target-specific requirements:

Using Remote Execution and Access on Windows operating systems

- Disable Simple File Sharing on Windows XP operating systems. Simple Networking requires that you log in as guest, however, a guest login does not have the authorizations necessary for Remote Execution and Access to work correctly.
- Enable the Internet Connection Firewall (ICF). On Windows XP operating systems, the ICF is disabled by default. On Windows XP Service Pack 2 operating systems, however, the Windows firewall is enabled by default. If either firewall is enabled on a Windows target operating system, Remote Execution and Access cannot access the workstation.
- Enable the remote registry administration and administrative sharing on Windows XP operating systems.
- Modify configuration settings of Windows Vista operating systems.

Using Remote Execution and Access on Linux, AIX®, HP-UX, and Solaris operating systems:

- Enable Secure Shell (SSH) protocol. The centralized installation manager uses SSH protocol to access target Linux, AIX, HP-UX, and Solaris operating systems. Remote Execution and Access, however, does not supply SSH code for AIX, HP-UX, and Solaris operating systems. OpenSSH Version 3.7.1 or later contains security enhancements that are not available in earlier releases. Remote Execution and Access cannot establish connections with any target AIX, HP-UX, and Solaris operating systems that have all remote access protocols disabled.

Procedure

- Complete the following steps to use Remote Execution and Access on the Windows XP operating system:
 1. Disable Simple File Sharing.
 - a. From Windows Explorer, click **Tools > Folder Options > View > Use Simple File Sharing**.
 - b. Clear the check box located next to **Use Simple File Sharing**.
 - c. Click **Apply** and **OK**.
 2. Disable the Internet Connection Firewall (ICF).
 - a. Click **Start > Settings**.
 - b. Select your network connection, and click **Change Windows Firewall Setting** from the Network Task navigator.
 - c. Select **File and Printing Sharing** from the **Exceptions** tab on the Windows firewall configuration, and click **OK**.
 3. Verify the remote registry administration is enabled and started on the target workstation for Remote Execution and Access to run commands and scripts.
 - a. Click **Start > Programs > Administrative Tools > Services > Remote Registry**.
 4. Enable administrative sharing.
 - a. Click **Start > My Computer**.
 - b. Right click the disk drive you are enabling for administrative sharing.
 - c. Click **Share and Security > Share this folder**.
 - d. Type the name of the disk share, and click **OK**.

Examples of the default administrative disk share are C\$ and D\$. If you disable sharing, directories located within the drives are considered as hidden by Remote Execution and Access, and the following message is displayed:

```
XCIM0009E: Error connecting to remote target <host_name>. Exception:  
java.io.FileNotFoundException: CTGRI0003E The remote path name specified cannot be found:  
<file_or_directory_path>. Cause: com.starla.smb.SMBException: The network name is  
incorrect.
```

- Complete one of the following options to use Remote Execution and Access on the Windows Vista operating system:
 1. Optional: Configure the deployment manager and the remote workstation as members of a Windows domain. Use a user account in the selected domain, or in a trusted domain, when you

- connect to the target.
2. Optional: Enable and use the provided administrator account to connect to the target workstation.
 - a. Click **Control Panel > Administrative Tools > Local Security Policy > Security Settings > Local Policies > Security Options**.
 - b. Double-click **Accounts: Administrator account status**.
 - c. Click **Enable** and **OK**.
 3. Optional: Disable the User Account Control that is enabled by default if you are using a different user account to connect to the target workstation.
 - a. Click **Control Panel > Administrative Tools > Local Security Policy > Security Settings > Local Policies > Security Options**.
 - b. Double-click **User Account Control: Run all administrators in Admin Approval Mode**.
 - c. Click **Disable** and **OK**.

Note: To save any configuration changes, you must restart the workstation.

- Complete the following steps to use Remote Execution and Access on Linux, AIX, HP-UX, and Solaris operating systems:
 1. Install and enable Secure Shell (SSH) on any target workstation that you want to access. Use the Bourne shell, sh, as the target shell on all target AIX and HP-UX operating systems. For target Solaris operating systems, use the Korn shell, ksh, instead.
 2. Edit the /etc/ssh/sshd_config file on Linux operating systems, and any other system, with SSH protocol enabled for Remote Execution and Access to use password authentication to access the target workstations.
 - a. Set the password authentication to yes.
 - b. Run /etc/init.d/sshd stop to stop the SSH daemon.
 - c. Run /etc/init.d/sshd start to restart the SSH daemon.

Note: The centralized installation manager uses Remote Execution and Access Version 2.3 to transfer files to target systems. OpenSSH Version 4.7.0.5302 for AIX Version 5.3 is not compatible with Remote Execution and Access Version 2.3. If your target systems are running AIX Version 5.3 with OpenSSH Version 4.7.0.5302 installed, the file transfer might stop in the middle of the transfer. For example, from the Installations in progress panel in the administrative console, the status of the installation remains in the transferring files state for longer than 90 minutes.

To determine the level of OpenSSH that is installed on your AIX system, run the following command:

```
lslpp -l | grep openssh
```

To resolve the issue, revert the OpenSSH version from Version 4.7.0.5302 to Version 4.7.0.5301.

Parent topic: [Managing installation targets](#)

Related tasks:

[Installing packages](#)

[Downloading package descriptors and the associated binary files](#)

[Managing installation targets](#)

Installing the Secure Shell public key to access installation targets

To use Secure Shell (SSH) public/private key as an authentication method to access remote installation targets, you must first install the public key of a public/private key pair on installation targets. You can then securely connect to installation targets by using the corresponding private key.

Before you begin

Linux **AIX** **HP-UX** **Solaris** To successfully complete this task, you must have SSH installed and enabled on the installation target. First create a pair of keys, and install the public key on all the installation targets. Run the following command to ensure that SSH is enabled on the workstation:

```
ps -e | grep sshd
```

Additionally, you must know the location of the SSH public key file on the deployment manager, and the administrative ID and password for the installation target. You use the same administrative ID and password to later install or uninstall software packages on the same installation target.

About this task

With the centralized installation manager, you can install product packages and maintenance for distributed platforms directly from the administrative console. AIX®, HP-UX, Solaris, and Linux operating systems generally support the use of SSH protocol. For Windows operating systems, however, you might have to install software from an independent software vendor to use SSH protocol. See [Running Cygwin as root to use SSH on target Windows operating systems](#) for more information.

To use the SSH public/private key authentication method to communicate with installation targets, first install the SSH public key.

Procedure

1. To access the wizard from the administrative console, click **System administration > Centralized Installation Manager > Installation targets**.
2. Select one or more existing installation targets from the table, and click **Install SSH Public Key**.
3. Enter the appropriate user ID and password settings, and click **Next**. You can either select to specify the same user ID and password to access all of the installation targets, or you can configure individual user IDs and passwords for each installation target.
4. Specify the location of the SSH public key file on the deployment manager, and click **Next**.
5. Review the summary of your selections, and click **Finish** to complete the installation process.

Results

You successfully installed the SSH public key on specific installation targets.

What to do next

You can install the same SSH public key on other installation targets. You can also use the user ID that you specified in the preceding steps to install fix packs and interim fixes. In the administrative console, click **System administration > Centralized Installation Manager > Available installations**.

Parent topic: [Managing installation targets](#)

Related tasks:

[Using Remote Execution and Access to access remote installation targets](#)

[Updating Secure Shell public key installation records](#)

[Running Cygwin as root to use SSH on target Windows operating systems](#)

[Installing packages](#)

Related reference:

[Centralized installation manager administrative tasks](#)

Running Cygwin as root to use SSH on target Windows operating systems

To use a Secure Shell (SSH) public/private key-pair to access target Windows operating systems, change the ownership of Cygwin files and directories to root.

Before you begin

Use the information provided in this topic only if you want to use the SSH public/private key authentication method to access remote installation targets that are running any of the Windows operating systems. You can skip this topic if you plan to use the user name and password authentication method to access the installation targets.

Ensure that software from an independent software vendor, such as a Cygwin SSH server, is installed on the Windows installation target.

About this task

The normal setup of a Cygwin sshd server to run as a Windows service designates the server to run under the Local SYSTEM account or, for a Windows 2003 Server, to run under a local account `ssd_server` specifically created with special privileges to run the service. With an SSH server configured and started on the Windows target, the server authenticates user logins using a public/private key-pair. However, with this setup, installation programs that are located on the Windows target and invoked by the centralized installation manager, which is using public SSH public/private key authentication to gain access to the installation target, are run using the identity of the account under which the SSH server is running. As a result, this causes problems with certain centralized installation manager operations when the files or directories on the target system, that the operation is to operate on, are created using different identities. To work around this, change the service that the Cygwin sshd server runs under to log on with the same account, root, that is used to install software on that specific Windows installation target.

Assuming that a local ID root that has Administrator authority to install software on the Windows installation target has been created, the following steps outline how to change the Cygwin sshd server to run under the ID root:

Procedure

1. Change the login ID of the Cygwin sshd service.
 - a. From the Windows Start menu, click **Settings > Control Panel > Administrative Tools > Services**.
 - b. From the Services window, right-click CYGWIN sshd, and select **Properties**.
 - c. From the Properties window, select the **General** tab, and click **Stop** to stop the sshd service.

Next, select the **Log on** tab, Under the **Log on as** section or prompt, clear the **Local System account** radio button, and select **This account**. Type `.\root` as the ID and type the password for the account. Click **Apply**.
2. Grant additional rights to the root account. Ensure that the account has the required privileges in addition to membership to the Administrators group.
 - a. From the Windows Start menu, click **Settings > Control Panel > Administrative Tools > Local Security Policy**.
 - b. From the Local Security Settings window, expand **Local Policies**, and select **User Rights Assignment**.
 - c. From the resulting page that appears on the right, verify that the root account has the following four rights:
 - Adjust memory quotas for a process
 - Create a token object
 - Log on as a service
 - Replace a process level tokenIf not, add root as a user with the four rights.

For Windows 2000, the first item in the preceding list is displayed as Increase quotas instead of Adjust memory quotas for a process.
 - d. Close the **Local Security Settings** window.
3. From a Cygwin console panel, change ownership of the following directories and files to root:
 - o `$ chown root /var/log/sshd.log`
 - o `$ chown -R root /var/empty`
 - o `$ chown root /etc/ssh*`
4. Restart the Cygwin sshd server.
 - a. From the **Properties** page of the Cygwin sshd server, select the **General** tab, and click **Start**. Verify that Cygwin is now running under the root user account.

What to do next

You can now install product packages and maintenance to your Windows installation targets. From the administrative console, click **System administration > Centralized Installation Manager > Installation targets**.

Parent topic: [Managing installation targets](#)

Related tasks:

[Installing the Secure Shell public key to access installation targets](#)

[Updating Secure Shell public key installation records](#)

[Using Remote Execution and Access to access remote installation targets](#)

Updating Secure Shell public key installation records

If you install the Secure Shell (SSH) public key on installation targets using methods other than the centralized installation manager, use the **updateKeyInstallationRecords** administrative task command to update the SSH public key records that the centralized installation manager maintains.

Before you begin

To update the SSH public key records, you must have administrator privileges.

About this task

You install the public key of a SSH public/private key pair to securely access installation targets. When you install the public key using methods other than the centralized installation manager, update the SSH public key installation records and save the user ID that is used to access installation targets.

Procedure

1. Run the **updateKeyInstallationRecords** command. The following example updates the SSH public key records for two sample hosts: abc.com and river.com.

- Using Jacl:

```
$AdminTask updateKeyInstallationRecords {-add "abc.com,river.com"}
$AdminTask listKeyInstallationRecords
```

- Using Jython:

```
AdminTask.updateKeyInstallationRecords (['-add "abc.com,river.com"'])
print AdminTask.listKeyInstallationRecords()
```

The **listKeyInstallationRecords** command lists the updated records.

2. Specify the configuration settings for the installation target. In the administrative console, click **System administration > Centralized Installation Manager > Installation targets**.
 - Click **Add Installation Target** to define the installation target if you did not do so already. The Installation target settings panel is displayed. Enter the host name, user ID, password that you use to access the installation target. Select the platform type, and click **Save**.
 - Click *host_name* if you already defined the installation target. The Installation target settings panel is displayed. Enter the user ID and password you use to access the installation target, and click **Save**. The Installation Targets panel is displayed. Repeat the process for all installation targets containing SSH public keys that are installed using methods other than the centralized installation manager.

Results

You can access your installation targets using the SSH authentication method.

What to do next

Use the user ID that you specified in the preceding steps to install fix packs and interim fixes. In the administrative console, click **System administration > Centralized Installation Manager > Available installations**.

Parent topic: [Managing installation targets](#)

Related tasks:

[Running Cygwin as root to use SSH on target Windows operating systems](#)
[Using Remote Execution and Access to access remote installation targets](#)
[Installing packages](#)

Related reference:

[Centralized installation manager administrative tasks](#)
[Administrative roles and privileges](#)

Routing requests to nodes that are not running WebSphere Virtual Enterprise

Use this task for configuring multiple tiers.

Before you begin

Plan your topology. See [WebSphere Virtual Enterprise topology terminology](#) for more information.

About this task

To make the autonomic request flow manager aware of nodes that are not running WebSphere Virtual Enterprise, you must use one of the following options to configure the node in the WebSphere Virtual Enterprise configuration.

Procedure

- The most common scenario is to install the middleware agent on each node that is running other middleware software. See [Adding middleware servers to configurations](#) for more information.
- Set the cell to run in Magic N mode. Magic N is the total number of requests that can run concurrently on the node group. To configure Magic N mode, specify the magicNMode and magicN custom properties. For more information about these custom properties, see [Autonomic request flow manager custom properties](#). Do not use this approach if your different service classes have very different computational demands.
- Specify work factors or speed factors and the computing power of each node. Speed factors are computed automatically by the work profiler for target tiers. For external nodes, you can manually configure speed factors for transaction class modules, which are a transaction class and web module pair. See [Configuring work factors in multiple tier configurations](#) for more information.
- Set the autonomic request flow manager in manual mode and specify the various settings. Let the autonomic request flow manager estimate speed factors based only on observations from an on demand router (ODR).

In manual mode, you must supply seat counts, or concurrency allocations, for each combination of ODR, deployment target, and service class. The dequeuing weights and gateway concurrency limits are derived from this information.

To allocate a certain number of seats to each combination of ODR, deployment target, and service class, use the *ProfileRoot/properties/arfm/manual/cell_name.xml* file. In WebSphere Extended Deployment Version 5.1, the file was *was_root/properties/arfm/manual/cell_name/node_group_name.xml*. Use this file on the machine where the autonomic request flow manager (ARFM) controller is running. With WebSphere Extended Deployment Version 6.0.1 and later, the ARFM controller can run on the node agent or an ODR.

The manual/ directory contains an example file that describes the format.

See [Configuring the autonomic request flow manager](#) for more information.

[Routing requests to external nodes with generic server clusters](#)

Use this task to route requests from the on demand router (ODR) to external nodes.

[Multiple tiers of processing](#)

The autonomic request flow manager (ARFM) can process requests in multiple tiers. The tiers might include both WebSphere Application Server software and other software.

[WebSphere Virtual Enterprise topology terminology](#)

This topic introduces the necessary terminology for creating a topology that involves servers that are outside of the WebSphere Virtual Enterprise cell.

[Configuring work factors in multiple tier configurations](#)

Use this task to compute and configure work factors for your multiple tier configuration.

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Related tasks:

[Configuring work factors in multiple tier configurations](#)

[Routing requests to external nodes with generic server clusters](#)

Routing requests to external nodes with generic server clusters

Use this task to route requests from the on demand router (ODR) to external nodes.

Before you begin

Note: To configure dynamic operations support for other middleware server types, you can define your other middleware servers as members of a dynamic cluster. See [Adding middleware servers to configurations](#) for more information.

You must have one or more nodes that are not running WebSphere Virtual Enterprise. When you perform this task, you add the nodes that are not running WebSphere Virtual Enterprise as generic server cluster ports.

To create and edit generic server clusters, you need administrator or configurator administrative privileges.

The following tasks must be completed before you configure the ODR to send requests to external nodes:

1. Create an ODR.
2. Configure service classes and transaction classes for the ODR.

About this task

The ODR supports automatic routing to nodes that are in remote cells if the ODR's local cell is connected to the remote cell by a core group bridge. To route to nodes that do not have WebSphere Virtual Enterprise installed, configure a generic server cluster. A generic server cluster is a collection of transport end points that can be used as a target for an ODR routing rule. Servers and nodes are inferred from the generic server cluster end points and properties. The inferred servers can be foreign or generic servers. Generic servers can be stopped and started by a WebSphere Application Server node agent. Foreign servers are usually not stopped and started by the WebSphere Application Server node agent. Some examples of foreign servers include Apache Tomcat or JBoss Application Server.

Procedure

1. Optional: Create one or more unmanaged nodes. You need to create unmanaged nodes in any of the following configuration situations:
 - If you want the remote agent to listen for requests from a port other than the default port, 9980.
 - If you are using a custom statistics collector plug-in to obtain node speed and average CPU utilization for the work profiler.
 - If your nodes are multi-homed, that is, they have more than one network interface card or multiple host names.

In the administrative console, click **System administration > Nodes > Add node** to create an unmanaged node.

2. Optional: If the remote agent listens for incoming requests from a port other than the default port, 9980, specify the `stats.collector.remote.agent.port` custom property on the unmanaged node.
 - a. Click **System administration > Nodes > node_name > Custom properties > New**.
 - b. For the **Name**, enter `stats.collector.remote.agent.port`.
 - c. For the **Value**, enter the port number where the remote agent listens for requests.

Remember: Write down the value of this port. You use the port value when you start the remote agent on the unmanaged node.

3. Optional: If you are using a custom statistics collector plug-in to obtain node speed and average CPU utilization for the work profiler, define the name of the class as a custom property on the unmanaged node. In this case, the plug-in might or might not use a remote agent. Before you configure this custom property, you must have a custom statistics collector plug-in coded and installed in the `lib` directory.
 - a. Click **System administration > Nodes > node_name > Custom properties > New**.
 - b. For the **Name**, enter `stats.collector.implementation.class`.
 - c. For the **Value**, enter the name of the custom statistics collector plug-in class. The default is the `com.ibm.ws.externalnode.plugins.DefaultStatsCollector` class.
 - d. You can specify additional configuration parameters by adding the `stats.collector.config` custom property on the unmanaged nodes. The provider of the custom plug-in must provide you with the information. The `stats.collector.config` is not required if you are using the default `com.ibm.ws.externalnode.plugins.DefaultStatsCollector` class.
4. Create a generic server cluster.
 - a. In the administrative console, click **Servers > Generic server clusters > New**.
 - b. Type a **Name** and select a **Protocol** for your generic server cluster.
 - c. Click **Apply**.
 - d. Specify the generic cluster server end points, or ports. On the generic cluster server settings page, click **Ports**.
 - e. Click **New** to add end points to the generic server cluster.
5. Optional: If you configured the remote agent to listen on a non-default port in a previous step, if you are using a custom statistics collector plug-in, or if your nodes are multi-homed, that is, they have more than

one network interface card or multiple host names, you must define the node custom property on the generic server cluster end points.

- a. In the administrative console, click **Servers > Generic server clusters > *cluster_name* > Additional properties > Ports > *host_name* > Additional properties > Custom properties > New**.
 - b. Define the custom property. Type the name of the custom property as **node**. The **value** of the custom property should be the name of the unmanaged node that is associated with the generic server cluster port.
 - c. Click Apply.
6. [Install your application on the external nodes using your defined port numbers.](#)
 7. [Enable external monitoring by enabling the remote agent on all of your external nodes.](#)
 8. [Define routing policies for generic server clusters.](#)

Before creating routing rules, you must create the appropriate virtual hosts. The virtual hosts can be defined in the console by clicking **Environment > Virtual hosts**.

9. [Define service policies for generic server clusters.](#)

Results

The work profiler obtains average CPU utilization and node speed from external nodes and can route traffic to the external nodes.

What to do next

Continue configuring the on demand router and the autonomic request flow manager.

[Enabling external monitoring with the remote agent](#)

Enable the remote agent on external nodes.

Parent topic: [Routing requests to nodes that are not running WebSphere Virtual Enterprise](#)

Related tasks:

- [Adding middleware servers to configurations](#)
- [Enabling external monitoring with the remote agent](#)
- [Creating ODRs](#)
- [Defining a service policy](#)
- [Configuring the autonomic request flow manager](#)

Enabling external monitoring with the remote agent

Enable the remote agent on external nodes.

Before you begin

The remote agent that is described in this task is not shipped with WebSphere® Virtual Enterprise Version 6.1. The middleware agent that is provided with Version 6.1 replaces the remote agent. See [Installing the middleware agent](#) for more information.

Set up generic server clusters with ports for all of your external nodes.

About this task

By installing the mixed server environment, a lightweight Java™ process called the remote agent runs on the node that is being monitored. There is also a plug-in that polls the remote agent for CPU information. The plug-in runs in the on demand router that has the work profiler running.

Procedure

1. Start the remote agent on nodes that you want to monitor. On a command line for each node, start the remote agent.
 - a. Navigate to the directory where the remote agent is installed. For example, type `cd <installed_dir>/bin`.
 - b. Start the remote agent. Run `./startAgent.sh` or `startagent.bat`.

Note: If you start the remote agent using the `startAgent.sh` or `startAgent.bat` script from the `bin` directory, the Java Runtime Environment (JRE) that is bundled with the installer is used. The JRE is installed in the `_jvm` subdirectory of the `_uninstXD` directory.

You can also supply the port number as a parameter to the script if the default port, 9980, is used by another process that is running on the node. By default, the remote agent can be run by any user. You can limit access to the directory that contains the remote agent for increased security.

2. Optional: Stop the remote agent. To stop the remote agent, use the command window where you started the remote agent. Type the command **quit**. You can also stop the remote agent by pressing `Control-C` or by using the UNIX `kill` command if you are running AIX®, HP-UX, Linux, or Solaris.

Results

The remote agent sends average CPU utilization, node speed, number of CPUs, and several other attributes and values to the plug-in that polls the remote agent for CPU information.

Example

You can start the remote agent in any way that a standalone Java program can be started. You can run the remote agent in the background.

On the Windows platform, you can start the remote agent as a scheduled task during system startup.

1. From Windows, click **Start > Settings > Control Panel > Scheduled Tasks > Add Scheduled Task**.
2. In the Scheduled Task Wizard, navigate to the `bin` directory where you installed the WebSphere Extended Deployment mixed server environment and select **startAgent.bat**.
3. Specify when you want to perform the task. For example, choose **When my computer starts**.
4. Specify the user ID and password for the user that you want to use to run **startAgent.bat**.

On the Linux or UNIX operating system, add an entry to the `/etc/inittab` file of the operating system. If you installed the feature in the `/opt/WAS60/MixedServer` directory, the new entry for the file is:

```
was:2:once:/opt/WAS60/MixedServer/bin/startAgent.sh >/dev/console 2>&1
```

For more information, see the UNIX manual page on `inittab` by typing `man inittab`.

What to do next

Enabling the remote agent is a requirement when routing requests to nodes that are not running WebSphere Virtual Enterprise. See [Routing requests to external nodes with generic server clusters](#) for more information.

Parent topic: [Routing requests to external nodes with generic server clusters](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

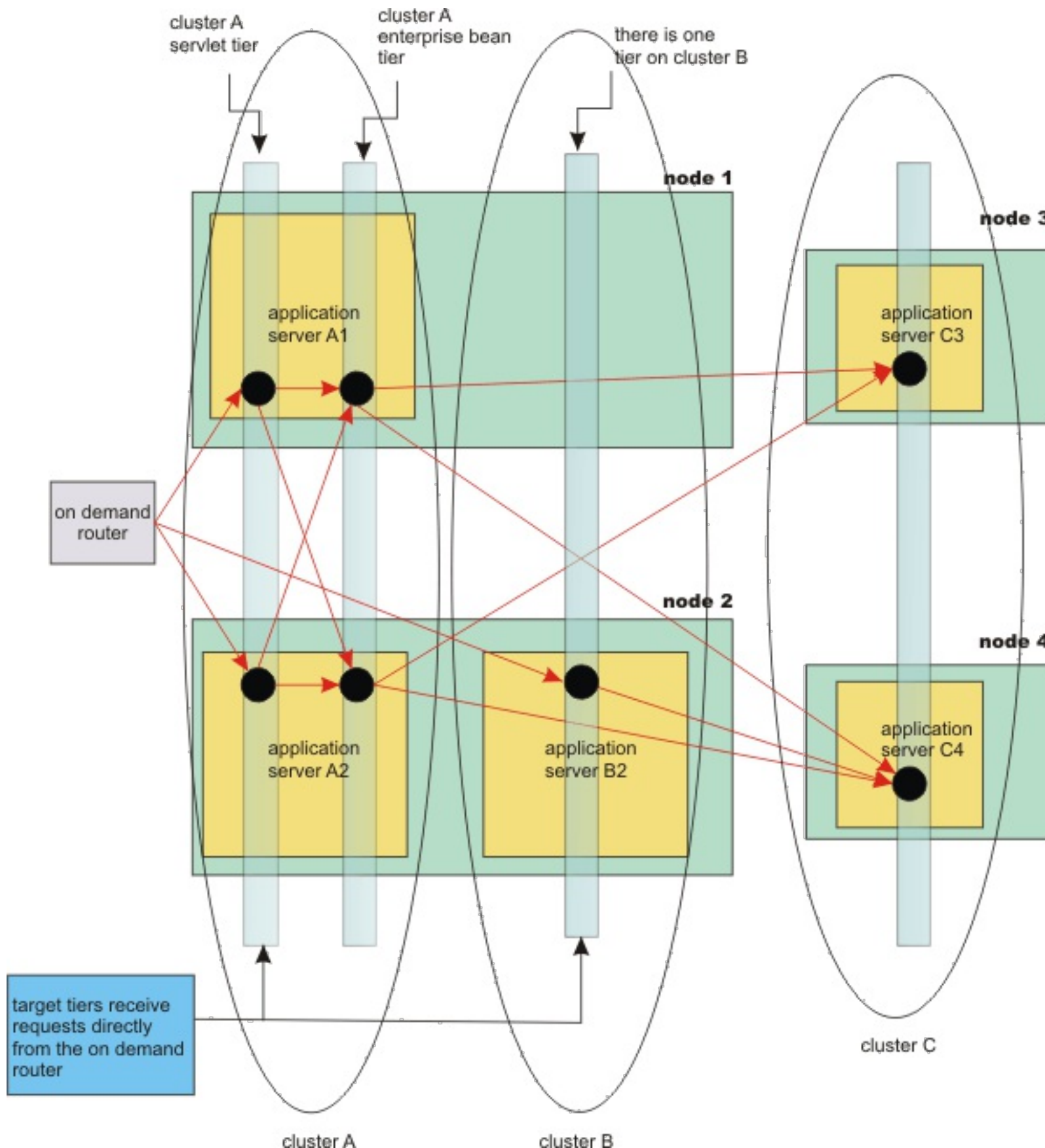
[Routing requests to external nodes with generic server clusters](#)
[Adding middleware servers to configurations](#)

Multiple tiers of processing

The autonomic request flow manager (ARFM) can process requests in multiple tiers. The tiers might include both WebSphere® Application Server software and other software.

In a configuration that has multiple tiers, it is assumed that all of the software is organized into nodes, server processes or application servers, server process clusters, deployment targets, which can include both clusters and server processes that are not clustered, and processing tiers. See [WebSphere Virtual Enterprise topology terminology](#) for more information about servers and clusters that are not running WebSphere software.

Figure 1. Multiple tiers of processing



In the preceding diagram, there are four nodes that have application servers that are members of a cluster. Each application server has one or more areas of processing, represented by black circles in the diagram.

Cluster A is made of WebSphere Application Server application servers. Server A1 and server A2 are both members of the cluster. Servlets and enterprise beans are deployed on the cluster. The servlets send work to the enterprise beans.

Each tier is specific to one deployment target. There are two tiers in cluster A. Cluster B and cluster C have one tier each. The servlet tier in cluster A and the only tier in cluster B are *target tiers*. A target server, target node, target cluster, target cell, or target tier receives requests directly from the on demand router (ODR).

The ARFM applies control at the entry to the system. In this case, the entry is the ODR. The ARFM does not measure traffic to or from tiers that are any deeper than the first tier. You must explicitly configure non-target tiers.

Restriction: The work profiler cannot determine good work factors when a node hosts both target and non-target tiers. The ARFM sends the application placement controller speed requests for the target clusters only. Each target cluster should have only one processing tier and should run on nodes that host target clusters only. Other nodes in your configuration can host non-target clusters that each contain multiple tiers.

For more information about configuring multiple tier configurations, see [Routing requests to nodes that are not running WebSphere Virtual Enterprise](#).

Parent topic: [Routing requests to nodes that are not running WebSphere Virtual Enterprise](#)

Related tasks:

[Configuring work factors in multiple tier configurations](#)

[Routing requests to external nodes with generic server clusters](#)

[Routing requests to nodes that are not running WebSphere Virtual Enterprise](#)

WebSphere Virtual Enterprise topology terminology

This topic introduces the necessary terminology for creating a topology that involves servers that are outside of the WebSphere® Virtual Enterprise cell.

- [Types of servers: three kinds of server processes](#)
- [Types of nodes: four kinds of machines](#)
- [Deployment targets](#)

Parent topic: [Routing requests to nodes that are not running WebSphere Virtual Enterprise](#)

Types of servers: three kinds of server processes

WebSphere Application Server application server

Any application server that is created and managed by WebSphere Application Server.

Generic server

A generic server can be stopped and started by a WebSphere Application Server node agent.

Foreign server

A server that does not fit into one of the preceding definitions. For example, a foreign server might be an application server that is not stopped and started by a WebSphere Application Server node agent. A foreign server also might be a database or other type of server.

Types of nodes: four kinds of machines

WebSphere Virtual Enterprise machine

Runs the WebSphere Virtual Enterprise node agent, plus any other kinds of servers.

WebSphere Application Server machine

Runs a WebSphere Application Server Version 5.1 or Version 6.0 node agent, plus any kinds of servers. These machines do not run WebSphere Virtual Enterprise. These machines might be monitored with the remote agent.

Pure external machine

A machine that runs foreign servers. It does not run a WebSphere Application Server node agent, but supports external monitoring with the remote agent or middleware agent. You can enable external monitoring on external machines by installing WebSphere Virtual Enterprise for a mixed server environment. See [Enabling external monitoring with the remote agent](#) for more information.

Foreign machine

A machine that runs foreign servers and does not support monitoring.

Unmanaged machines. WebSphere Application Server stores information about unmanaged machines in the configuration files. Unmanaged machines can have custom properties. Pure external machines and foreign machines might be unmanaged, but also might not have information stored in the configuration files. Machines that do not have any information stored in the configuration files are neither managed or unmanaged.

Deployment targets

A deployment target is a place that can contain a processing tier. There are five kinds of deployment targets:

- WebSphere Application Server application servers that are not clustered
- A static cluster of WebSphere Application Server application servers
- A dynamic cluster of WebSphere Application Server application servers
- Generic server cluster

A generic server cluster is a collection of transport end points that can be used as a target for an ODR routing rule. Generic server clusters are not collections of generic servers. Servers and nodes are inferred from the generic server cluster end points and properties. The inferred servers can be foreign or generic servers. Generic server cluster end points can have a custom property that identifies its node, or the node is identified by the host name from the end point. The node for a generic server cluster can be any type of node. See [Routing requests to external nodes with generic server clusters](#) for more information about configuring generic server clusters.

- External cluster

An external cluster is used for foreign and generic servers that are not referenced by generic server clusters. For example, an external cluster might reference database servers. External clusters cannot be a target, and cannot contain target servers. Servers that belong to an external cluster can run on any type of machine. You must explicitly configure the server placement and speeds of the unmonitored nodes in the ODR cell using the external.placement custom property.

Related tasks:

[Routing requests to external nodes with generic server clusters](#)

Related information:

[Autonomic request flow manager advanced custom properties](#)

Configuring work factors in multiple tier configurations

Use this task to compute and configure work factors for your multiple tier configuration.

Before you begin

- Install WebSphere® Virtual Enterprise.
- Install applications that are operational under a workload.

About this task

A work factor exists for every combination of transaction class, target web module, and processing tier. The work factor describes how heavily a request of the given transaction class loads the processing tier. You can define work factors at varying levels of granularity. The autonomic request flow manager (ARFM) uses work factors at the level of service class, target deployment target, and processing tier. You can define work factors at a variety of levels for any processing tier that is not a target tier or is not the one and only processing tier in the target module.

In a configuration that has multiple tiers, the work profiler automatically computes work factors for the target tier, which communicates directly with the on demand router (ODR). For any tiers that are deeper than the target tier, you must define work factors. If your deployment target contains both a target tier and a non-target tier, you must configure the work factors for both tiers because the work profiler cannot automatically compute work factors in that situation. You can compute the work factor by dividing the average processor utilization by the average number of executing requests per second. This task describes how to find these values and configure the work factor for your multiple tier configuration.

Procedure

1. Generate traffic for a transaction class and module pair. You can generate traffic by using an application client or a stress tool.
2. Monitor processor utilization in your configuration. Determine an average processor utilization. You need the processor utilization of all the machines that are involved in serving your traffic, and all machines that have performance interactions with them, to be at the configured limit that is defined with the **Maximum CPU utilization** property on the **Operational policies > Autonomic managers > Autonomic request flow manager** panel. Disable all the autonomic managers so that you can ensure the system does not make changes while you take the processor utilization measurement.
 - The application placement controller: Disable the application placement controller by putting it in manual mode. Click **Operational policies > Autonomic managers > Application placement controller**. Select the **Enable** check box to disable the application placement controller.
 - The autonomic request flow manager: You can set the `arfmManageCpu` custom property to false at the cell level to disable the ARFM.
 - Dynamic workload management: Disable dynamic workload management for each dynamic cluster. Click **Servers > Dynamic clusters > *dynamic_cluster_name* > Dynamic WLM**. Select the **Dynamic WLM** check box to disable dynamic workload management.If you disable the autonomic managers, you can add processor load through background tasks. Use an external monitoring tool for your hardware.
3. Using the runtime charting in the administrative console, monitor the number of requests per second (throughput). Click **Runtime Operations > Runtime Topology**. You can view the number of requests per second.
4. Compute the work factor for the deployment target. Use the following equation to calculate the work factor:

```
work factor = (normalized CPU speed) * (CPU utilization) /  
(number of requests per second, measured at entry and exit of the target  
tier)
```

5. Configure the work factor in the administrative console. You set the custom property on the deployment target, for example, a cluster of servers, or a stand-alone application server. For more information about the overrides that you can create with the `workFactorOverrideSpec` custom property, see [Autonomic request flow manager advanced custom properties](#).
 - a. Define a case for each tier in the deployment target. Each case is separated by a comma, and contains a pattern that is set to a value that is equal to the work factor that you calculated. The pattern defines the set of service classes, transaction classes, applications, or modules that you can override for the particular tier, as shown in the following example:

```
service-class:transaction-class:application:module:[tier,  
optional]=value
```

You can specify a wild card for any of the service class, transaction class, application, or module by entering a * symbol. Each pattern can include at most one application, at most one module, at most one service class, and at most one transaction class. The tier is optional, and represents the deployment target name and relative tier name. Set the value to a work factor override number or to none to define no override.

In the following examples, work factor override values are set for two-tier configurations:

- Set an override value to 100 for the one and only processing tier in the target cluster:

```
*:*:*:*100
```

- Set an override value to none for the first tier of the MyDynamicCluster cluster. Set an override value to 100 for the second tier of the MyDynamicCluster cluster in the default cell:

```
*:*:*:*=none,*:*:*:*:MyDynamicCluster+2=100
```

- Set an override value to none for the first tier. Set an override value to 0.7 for the CICS+1 tier in the DbCel cell:

```
*:*:*:*=none,*:*:*:*:../DbCel/CICS=0.7
```

- b. Create the custom property in the administrative console. Click **Servers > Dynamic clusters > dynamic_cluster_name > Custom properties > New**. Set the name of the property to workFactorOverrideSpec. Set the value of the property to the string you created in the previous step.
- c. Save your configuration.

Results

Work factors are configured to override the work factor values that are created by the work profiler and support performance management of more than one tier.

What to do next

Repeat these steps for each transaction class module and non-target tier node pair. You also must configure the node speed for each external node. See [Configuring node computing power](#) for more information.

[Overriding work factor estimates](#)

Use this task to override the values that are computed by the work profiler.

[Configuring node computing power](#)

You must configure the node computing power if you are using speed factor or work factor overrides to configure your multiple tier configuration.

Parent topic: [Routing requests to nodes that are not running WebSphere Virtual Enterprise](#)

Related concepts:

[Multiple tiers of processing](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

Overriding work factor estimates

Use this task to override the values that are computed by the work profiler.

Before you begin

If you have already specified a speed factor override, you do not need to configure a work factor override. See [Configuring work factors in multiple tier configurations](#) for more information about speed factors.

About this task

The work profiler computes a work factor for each transaction class module (TCM). Specifying the override depends on the version of the product that you are using. In WebSphere® Extended Deployment Version 5.1, explicit transaction classes were specified. In WebSphere Extended Deployment Version 6.0 and later, you can use work classes to classify work into transaction classes.

When using explicit transaction classes like in WebSphere Extended Deployment Version 5.1, each TCM is stored in the file that defines the service class to which it belongs. A service class is defined by a `profile/config/cells/cell name/serviceclasses/service class name/serviceclass.xml` file. In this file, each `<TransactionClassModules>` element defines a single TCM, and you can attach a `workFactor` attribute to override the work profiler calculations for that TCM.

When you are using work classes, an XML element does not exist for each TCM. In this case, the work factor overrides related to a given deployment target (cluster, or unclustered server) can be specified in a custom property that is attached to that deployment target.

Use the `workFactorOverrideSpec` custom property. The value for this custom property is a string that contains a set of matching rules, or cases, that define the work factor override, either a value or none, for every TCM of that deployment target. The syntax of that string is the same as the syntax of the speed factor overrides, except that you cannot specify a tier for a work factor override case. See [Configuring work factors in multiple tier configurations](#) for more information about speed factors. A work factor is in units of millions of CPU clock cycles on a standard kind of machine, and describes the average amount of work that must be done on the managed tier for a single request of that TCM.

Procedure

1. Reduce the number of back-end nodes in the node group to one back-end node in the node group panel in the administrative console.
2. Using a workload generator, send traffic to one TCM only. Generate enough traffic to load the back-end node to above 70% utilization.
3. Record the average values for the steady state throughput and corresponding CPU utilization. You can get those values from the runtime topology view of the admin console. To view the runtime topology, click **Runtime operations > Runtime topology**.
4. Compute the work factor value for the transaction class module. Use the value from the following equation:

```
work_factor = (CPU_Utilization * node_speed) / Throughput
```

In this case, `Node_speed` equals the value of the `node.speed` node custom property. See [Configuring node computing power](#) for more information about setting the `node.speed` custom property. It is assumed that all the nodes in the node group are homogeneous, with equal node speeds.

5. Compose a string to override the work factor for the deployment target. The pattern defines the set of service classes, transaction classes, applications, or modules that you can override. The pattern is :

```
service-class:transaction-class:application:module = value
```

You can separate each work factor with a comma, or specify a wild card for any of the variables by using an asterisk. Set the value variable to the computed work factor value. See [Autonomic request flow manager advanced custom properties](#) for more information about the grammar for work factor override specifications.

6. Create the custom property.
 - a. Click **Custom Properties > New** in the deployment target.
 - b. Enter the **Name** of the custom property as `workFactorOverrideSpec`.
 - c. The **Value** of the custom property is equal to the string composed in the previous step to override the work factor.

What to do next

Configure the speed of the external nodes. See [Configuring node computing power](#) for more information.

Parent topic: [Configuring work factors in multiple tier configurations](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

[Configuring work factors in multiple tier configurations](#)
[Routing requests to nodes that are not running WebSphere Virtual Enterprise](#)

Related information:
[Product overview](#)

Configuring node computing power

You must configure the node computing power if you are using speed factor or work factor overrides to configure your multiple tier configuration.

Before you begin

Configure work factor or speed factor overrides. See [Configuring work factors in multiple tier configurations](#) and [Overriding work factor estimates](#) for more information. You must know which kind of node you are configuring:

- A *managed node* is a node with an application server and a node agent that belongs to a cell. Managed nodes can have custom properties defined.
- An *unmanaged node* is a node that is defined in the cell topology that does not have a node agent that manages the process. An unmanaged node is typically used to manage Web servers. Unmanaged nodes are defined in the administrative console and can have custom properties defined.
- A node that is *not managed and not unmanaged* does not have any information stored in the configuration files. You cannot define custom properties for nodes that are not managed and not unmanaged.

About this task

Define node computing power when you are using speed factor or work factor overrides to define your multiple tier configuration.

Procedure

- If your node is managed or unmanaged, specify the node speed using the `node.speed` custom property.
 1. In the administrative console, click **System administration > Nodes > *node_name* > Custom Properties > New**.
 2. Enter the **Name** as `node.speed`.
 3. Enter the **Value**. The value equals the node speed in MHz. In a multi-processor node, multiply the processor speed by the number of processors on a multi-processor node.
- If your node is not managed and not unmanaged, configure the `external.placement` custom property. For more information about the `external.placement` custom property, see [Autonomic request flow manager advanced custom properties](#).
 1. Compose a string that defines a case for each external node that is not managed and not unmanaged. Each case is separated by a semicolon. Each case consists of information about the node that is defined in the following pattern:

```
nodeName:nodeSpeed:plmplist
```

The `nodeName` variable is the name of the node. The `nodeSpeed` variable is the speed of the node. The `plmplist` is a placement list of deployment targets. Separate the deployment targets with commas. For example, you have the DBA and DBB unmonitored nodes in the SysX cell. The DBA node has a power of 4.7. The DBB node has a power of 2.7. There are two external clusters in the SysX cell that are named DB1 and DB2®. The DB1 cluster is placed on the DBA node only. The DB2 cluster is placed on both the DBA node and the DBB node. Use the following string to define this configuration:

```
../SysX/DBA:4.7:DB1,DB2; ../SysX/DBB:2.7:DB2
```

2. Using the string that describes the external nodes, create the `external.placement` custom property. In the administrative console, click **System administration > Cell > Custom Properties > New**. Enter the name of the custom property as `external.placement`. The **Value** equals the string that you created to define the external nodes.

Parent topic: [Configuring work factors in multiple tier configurations](#)

Related information:

[Autonomic request flow manager advanced custom properties](#)

Configuring a high availability deployment manager environment

The high availability (HA) deployment manager function can be configured to eliminate single point of failure for administrative functions in a WebSphere® Application Server Network Deployment cell.

Before you begin

This feature is supported only on distributed platforms and is not available on z/OS®.

Privileges for the high availability (HA) deployment manager differ, depending on the various roles. Roles include monitor, operator, configurator, and administrator. If you are a user with either a monitor or a configurator role, you can only view the HA deployment manager information. If you have the role of operator or administrator, you have all the privileges for the HA deployment manager.

Before you follow these steps ensure that the system environment has a suitable shared file system available, such as IBM® SAN FS or NFS Version 4. To verify that your file system is suitable for use with the high availability deployment manager, use the [IBM File System Locking Protocol Test for WebSphere Application Server](#). The shared file system requirements are the same for both the failover of transaction logs and for the high availability deployment manager.

Note: You must stop all deployment managers that are running in your environment before you can perform maintenance on the NFS drive. Use the extended repository service in conjunction with the HA deployment manager feature. In the event of a NFS failure, you can recover the latest configuration changes by using the extended repository service.

Each deployment manager must share the same master configuration repository and workspace area, which must be on the shared file system. Each deployment manager must also have read and write permissions to the shared master configuration repository and workspace area. The master configuration repository and workspace area default locations are *install_root/profiles/deployment_manager/config* and *install_root/profiles/deployment_manager/wstemp/* directories.

File sharing is typically accomplished by installing the deployment manager on the shared file system. Sharing the entire installation is not mandatory, and an alternative is to locate only the deployment manager profiles on the shared file system. In the scenarios that are described in this topic, WebSphere Virtual Enterprise is installed on the shared file system. For scenarios in which only the deployment manager configuration repository and workspace reside on the Network Attached Storage (NAS), see the [xd_hadmgrAdd command](#).

About this task

Set up a high availability deployment manager environment to eliminate the deployment manager as a single point of failure. When one deployment manager fails, a configured deployment manager can resume the work. See [The high availability deployment manager](#) for more information.

Procedure

1. Choose a topology. The HA deployment manager function can be configured in a variety of topologies. In a production environment, it is recommended that each deployment manager runs on its own computer to increase availability. If a single on demand router (ODR) is in the topology, it must run on its own computer. To eliminate the ODR as a single point of failure, you can use multiple on demand routers in conjunction with an IP sprayer. A typical HA deployment manager environment requires at least three computers:

Computer name	Processes
A	Deployment manager
B	Standby deployment manager
C	Node agent and on demand router

2. You can either create a new cell with an HA deployment manager configuration, or convert an existing cell to an HA deployment manager configuration.

To create a new cell with an HA deployment manager configuration:

- a. Install WebSphere Application Server on the shared file system.
- b. Install WebSphere Virtual Enterprise on the shared file system.
- c. Create a deployment manager profile using the IP address for computer A.
- d. Start the deployment manager on computer A.
- e. Create a custom profile using the IP address for computer C. Federate computer C with the deployment manager.
- f. Create an ODR on the federated custom profile.
- g. Create a deployment manager profile using the IP address for computer B.
- h. Run the `xd_hadmgrAdd` command within the deployment manager profile for computer B to convert it into a standby deployment manager. For more information, see [xd_hadmgrAdd](#)

[command](#).

To convert an existing cell to an HA deployment manager configuration:

- a. Install WebSphere Application Server on the shared file system.
 - b. Install WebSphere Virtual Enterprise on the shared file system.
 - c. Create a deployment manager profile using the IP address for computer A and with the same cell and node name as the existing deployment manager.
 - d. Use the backupConfig and restoreConfig command line utilities to relocate the existing cell configuration to the shared file system.
 - e. Copy the tmsStorage folder and its contents from the existing deployment manager profile to the new deployment manager profile on the shared file system.
 - f. Ensure that the existing cell configuration contains an ODR. Create an ODR if an ODR does not exist.
 - g. Create a deployment manager profile using the IP address for computer B.
 - h. Run the xd_hadmgrAdd command within the deployment manager profile for computer B to convert it into a standby deployment manager. For more information, see [xd_hadmgrAdd command](#).
3. Restart the deployment managers and on demand routers.

Results

Multiple deployment managers are configured. The deployment manager is eliminated as a single point of failure because a primary deployment manager hosts the administrative function for the cell, and has backup deployment managers that are in standby node. If the primary deployment manager fails, the standby deployment managers can resume without losing work.

What to do next

After your high availability deployment manager environment is set up, you can manage the deployment managers in the administrative console. Click **System administration > Deployment manager > All deployment managers** to view and manage the configured deployment managers in your environment.

If you need to remove a high availability deployment manager from the cell, use the command line utility to remove the function. See [xd_hadmgrRemove command](#) for more information.

[Adding the signer certificate from the secondary deployment manager to the local trust store](#)

To enable Secure Sockets Layer (SSL) in your high availability deployment manager environment, the local trust store must contain the signer certificate from the secondary deployment manager. If the trust store does not contain the signer certificate, add the certificate to the trust store to prevent errors and enable secure communication among the core group members.

[The high availability deployment manager](#)

The high availability (HA) deployment manager function is configured using a shared file system. When this configuration option is chosen, multiple deployment managers are configured. The benefit of the HA deployment manager function is that it eliminates the deployment manager as a single point of failure for cell administration. This is important in environments that have significant reliance on automated operations, including application deployment and server monitoring.

[xd_hadmgrAdd command](#)

The **xd_hadmgrAdd** command incorporates a new deployment manager profile into a cell. The new deployment manager profile is converted to a highly available (HA) deployment manager peer in the same cell as the existing deployment manager profile.

[xd_hadmgrRemove command](#)

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Related concepts:

[The high availability deployment manager](#)

Related tasks:

[Adding the signer certificate from the secondary deployment manager to the local trust store](#)

[Configuring rewriting rules](#)

[Creating and configuring ODRs](#)

Related reference:

[Administrative roles and privileges](#)

Related information:

[xd_hadmgrRemove command](#)

[xd_hadmgrAdd command](#)

Adding the signer certificate from the secondary deployment manager to the local trust store

To enable Secure Sockets Layer (SSL) in your high availability deployment manager environment, the local trust store must contain the signer certificate from the secondary deployment manager. If the trust store does not contain the signer certificate, add the certificate to the trust store to prevent errors and enable secure communication among the core group members.

About this task

To elect the secondary deployment manager to take over as the primary deployment manager when SSL is enabled in your environment, the signer certificate of the secondary deployment manager must exist in the local trust store. Specifically, the `com.ibm.ssl.trustStore` value must be set to the cell-level default trust store in the `deployment_manager_profile/properties/ssl.client.props` file. If the certificate cannot be located in the local trust store, the SSL handshake fails and you might receive the following error message:

```
CWPKI0022E: SSL HANDSHAKE FAILURE: A signer with SubjectDN "CN=xdblade36b07.rtp.raleigh.ibm.com, O=IBM, C=US" was sent from target host:port "*:9043".  
The extended error message from the SSL handshake exception is: "No trusted certificate found".
```

Add the signer certificate from the secondary deployment manager to the local trust store to enable secure communication in your high availability deployment manager environment.

Procedure

1. In the administrative console, click **Security > SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore > Signer certificates > Retrieve from port**.
2. Define the following general properties to retrieve the signer certificate from the remote SSL port, and click **Retrieve signer information**:

Host

Specifies the host name that you connect to when you retrieve the signer certificate from the SSL port

Port

Specifies the SSL port that you connect to when you retrieve the signer certificate

SSL configuration for outbound connection

Specifies the configuration that is used to connect to the SSL port

This configuration is the SSL configuration that contains the signer certificate after you add the certificate to the trust store.

Alias

Specifies the certificate alias that is used in the SSL configuration

Results

The configuration can connect to and accurately check the status of the secondary deployment manager.

Parent topic: [Configuring a high availability deployment manager environment](#)

Related tasks:

[Configuring a high availability deployment manager environment](#)

[Configuring WebSphere Virtual Enterprise for cross-cell communication](#)

[Configuring communication between core groups that are in the same cell](#)

Related information:

[Errors configuring Secure Sockets Layer encrypted access](#)

The high availability deployment manager

The high availability (HA) deployment manager function is configured using a shared file system. When this configuration option is chosen, multiple deployment managers are configured. The benefit of the HA deployment manager function is that it eliminates the deployment manager as a single point of failure for cell administration. This is important in environments that have significant reliance on automated operations, including application deployment and server monitoring.

Deployment manager overview

The deployment managers exist as peers. One is considered active, also known as primary, and hosts the administrative function of the cell, while the others are backups in standby mode. If the active manager fails a standby takes over and is designated the new active deployment manager. A new command line utility is provided with WebSphere® Extended Deployment to clone the original cell deployment manager into additional deployment managers. Each deployment manager is installed and configured to run on a different physical or logical computer. The deployment managers need not be hosted on homogenous operating platforms, although like platforms are recommended. Each deployment manager shares the same instance of the master configuration repository and workspace area. These must be located on a shared file system.

The file system must support fast lock recovery. The IBM® General Parallel File System™ (GPFS™) is recommended, and the Network File System Version 4 (NFS) is also an option. If you use the high availability deployment manager on AIX® Version 5.3 and are using NFS Version 4, you must have bos.net.nfs.client at Version 5.3.0.60 or later.

Note: You must stop all deployment managers that are running in your environment before you can perform maintenance on the NFS drive. Use the extended repository service in conjunction with the HA deployment manager feature. In the event of a NFS failure, you can recover the latest configuration changes by using the extended repository service.

Normal operation includes starting at least two deployment managers. A new highly available deployment manager component runs in each deployment manager to control which deployment manager is elected as the active one. Any other deployment manager in the configuration is in standby mode. The WebSphere Extended Deployment on demand router (ODR) is configured with the communication endpoints for the administrative console, the wsadmin tool, and scripting. The ODR recognizes which deployment manager instance is active and routes all administrative communication to that instance. The HA deployment manager function supports only use of the JMX SOAP connector. The JMX RMI connector is not supported in this configuration.

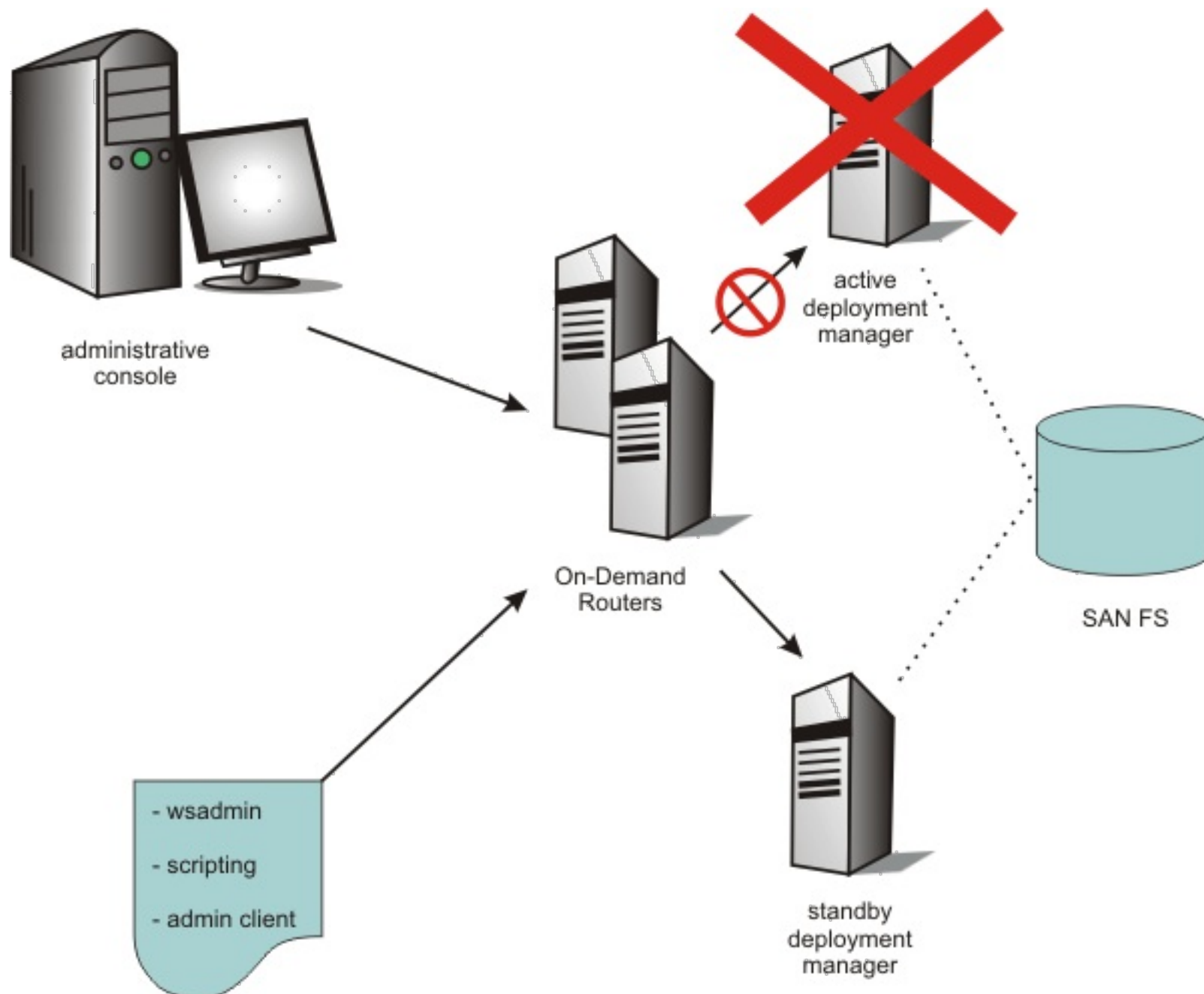
Configuration

The deployment managers are initially configured into the same core group. Configuring the deployment managers in the same core group is important so that the routing information that is exposed to the ODR is consistent across all the deployment managers. If the deployment managers are placed into separate core groups, the core groups must be connected with a core group bridge.

A typical HA deployment manager configuration consists of two deployment managers that are located on separate workstations. The deployment managers share a master repository that is located on a SAN FS. All administrative operations are performed through the elected active deployment manager. The standby deployment manager is fully initialized and ready to do work but cannot be used for administration. This is because the administrative function does not currently support multiple concurrent server processes writing to the same configuration. Therefore, the standby rejects any login and JMX requests.

However, if the active deployment manager is stopped or fails, the highly available deployment manager component recognizes the loss of the active deployment manager and dynamically switches the standby into active mode so it can take over for the lost deployment manager. The active and standbys share work spaces. When a deployment manager takeover occurs, work is not lost, because the ODR automatically recognizes the election of the new active deployment manager and reroutes administrative requests to the new active deployment manager. Note that there is a sub 1 minute period of time where the deployment manager will not be available until failover to the secondary is complete.

Failover to the new active deployment manager is depicted in the following diagram:



While the HA deployment manager component is able to detect deployment manager failure and initiate takeover, there are edge conditions where each deployment manager could temporarily believe it is the active deployment manager. To prevent this situation from occurring, the active deployment manager holds a file lock in the shared file system. Because of this, the takeover of the active deployment manager by the standby will take a brief period of time approximately equal to the time it takes for the shared file system to detect the loss of the active deployment manager and release the lock. SAN FS and NFS both use a lock lease model and have configurable times for lock release for failed lock holders. This can be configured as low as 10 seconds for SAN FS.

z/OS Note: The alternative for deployment manager HA on z/OS® is based on starting the deployment on a different logical partition (LPAR). This is documented in IBM® Techdoc *WP100415: Starting Deployment Manager on another MVS™ image*.

Parent topic: [Configuring a high availability deployment manager environment](#)

Related tasks:

- [Configuring a high availability deployment manager environment](#)
- [Configuring communication between core groups that are in the same cell](#)
- [Configuring WebSphere Virtual Enterprise for cross-cell communication](#)

Related information:

- [IBM General Parallel File System \(GPFS\) Information Center](#)
- [IBM Techdoc WP100415: Starting Deployment Manager on another MVS image](#)

xd_hadmgrAdd command

The **xd_hadmgrAdd** command incorporates a new deployment manager profile into a cell. The new deployment manager profile is converted to a highly available (HA) deployment manager peer in the same cell as the existing deployment manager profile.

You must run the **xd_hadmgrAdd** command within the new deployment manager profile that is being added. The HA deployment manager function supports use of the JMX SOAP connector only. The JMX RMI connector is not supported in this configuration.

Usage

Distributed platforms

```
xd_hadmgrAdd -hostname primary_dmgr_host [-port primary_dmgr_port]  
-configRoot fully_qualified_path_to_shared_configuration  
-workspaceRoot fully_qualified_path_to_workspace  
-proxyServerJmxSoapAddress JMX_SOAP_host:JMX_SOAP_port  
-proxyServerHttpPort HTTP_portHTTP_secure_port  
[-uniquePort] [-user user_name] [-password password]  
[-quiet] [-logfile file_name] [-replacelog] [-trace] [-help]
```

The **proxyServerJmxSoapAddress** and **proxyServerHttpPort** parameters are required only when you are creating the first standby deployment manager.

Parameters

The following options are available for the **xd_hadmgrAdd** command:

-hostname <host name>

Specifies the host name that is used to connect to the existing deployment manager.

-port <port>

Specifies the default SOAP port that is used to connect to the existing deployment manager.

-configRoot <fully qualified path>

Specifies the fully-qualified configuration path.

-workspaceRoot <fully qualified path>

Specifies the fully-qualified workspace path.

-proxyServerJmxSoapAddress <host:port>

Specifies the host name and IP address of the on demand router (ODR) and the SOAP_CONNECTOR_ADDRESS port for the ODR. Use this port to run wsadmin scripts.

-proxyServerHttpPort <port> <secure_port>

Specifies the HTTP ports for the proxy server. Select two free ports on the ODR and IP sprayer host. These ports should be used to access the administrative console by way of the ODR.

-user <user name>

Specifies the user name that is used to connect to the existing deployment manager.

-password <password>

Specifies the password that is used to connect to the existing deployment manager.

-quiet

Suppresses the progress information that the **xd_hadmgrAdd** command prints in normal mode.

-logfile <filename>

Specifies the location of the log file to which information gets written. By default, the hadmgrAdd.log file is created in the logs directory of the profile for the node that is being added.

-replacelog

Replaces the log file instead of appending to the current log. By default, the **xd_hadmgrAdd** command appends to the existing trace file. This option causes the **xd_hadmgrAdd** command to overwrite the trace file.

-trace

Generates additional trace information in the log file for debugging purposes.

-uniquePort

Checks for port conflict. If the new port is conflict with the existing ports, the new port increments by one until a free port is found. This process is not necessary if unique ports were assigned to the deployment manager profile when it was created.

-help

Displays syntax help.

Example

1. Start with an existing WebSphere® Application Server Network Deployment cell that is installed and configured with the following directory structure:
 - WebSphere Application Server home directory: /WebSphere/AppServer/
 - Existing deployment manager profile: /shared/profiles/PrimaryManagerNodewhere the /shared *mountpoint* is on a SAN FS device.
2. Run the Profile Management Tool plug-in or the **wasprofile** command line utility to create another deployment manager profile associated with this same Network Deployment installation. Create the profile on the shared disk, /shared *mountpoint* for this example.

You must specify the correct host name for this deployment manager to listen on. You can specify the explicit port numbers for your standby deployment manager when you create the profile or you can specify the **-uniquePort** parameter when you run the **xd_hadmgrAdd** command, as shown in this example. You can specify any value for the cell name, because it is changed to match the cell name to which this HA deployment manager peer is added during the conversion process. You must specify a node name that is unique in the cell to which this HA deployment manager peer is added.

You have the following directory structure after you create the profile:

- WebSphere Application Server home directory: /WebSphere/AppServer/
 - Existing deployment manager profile: /shared/profiles/PrimaryManagerNode
 - New deployment manager profile: /shared/profiles/StandbyManagerNode
3. Run the **xd_hadmgrAdd** command to convert the new deployment manager profile into an HA deployment manager peer in the same cell in which the existing deployment manager profile exists.

```
xd_hadmgrAdd.sh|bat -hostname a.a.a.a -port 8879 -configRoot
/shared/profiles/PrimaryManagerNode/config
-workspaceRoot /shared/profiles/PrimaryManagerNode -
proxyServerJmxSoapAddress x.x.x.x:8880

-proxyServerHttpPort 9060 9043
-user wsadmin
-password *****
-uniquePort
```

If you install the centralized installation manager repository, and after you create the second deployment manager profile, change the value of the `CENTRALIZED_INSTALL_REPOSITORY_ROOT` property in the `install_root\properties\cimgr.props` file from `c:\ProgramFiles\IBM\WebSphere\AppServer\repository_folder` to `${WAS_INSTALL_ROOT}/repository_folder`.

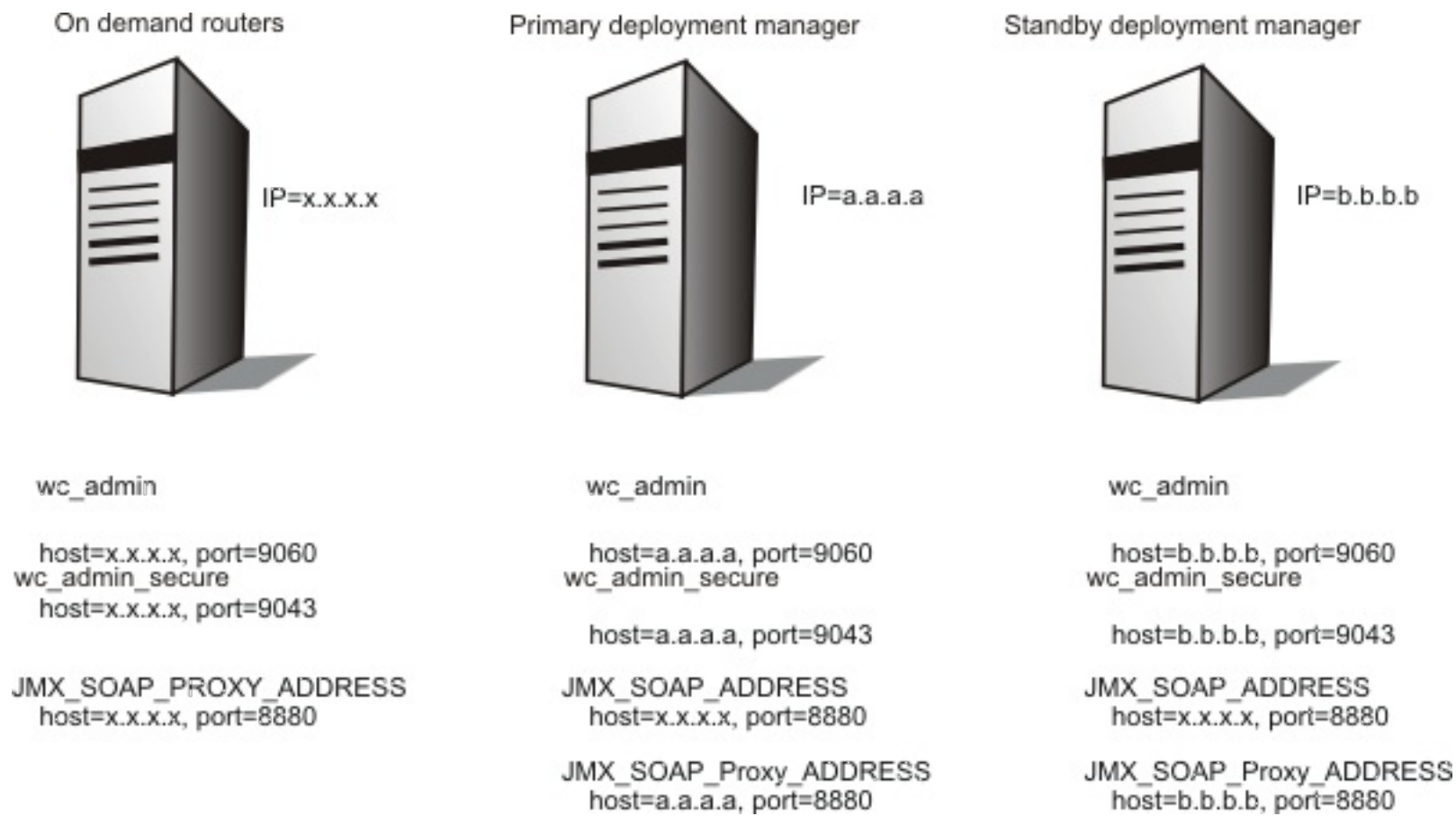
When you use the **xd_hadmgrAdd** command, the following changes are made to your cell configuration:

- The new deployment manager is configured to use the same configuration repository instance as the existing deployment manager, which must be on a shared device with the existing deployment manager.
- The new deployment manager is configured to use the same specified workspace shared directory, which must be on a device shared with the existing deployment manager.
- The ODR is configured with the `wc_admin` and `ws_admin_secure` HTTP transport channels and the `JMX_SOAP_PROXY_ADDRESS` endpoint.
- The `JMX_SOAP_CONNECTOR` addresses of the deployment managers point to the `JMX_SOAP_PROXY_ADDRESS` of the ODR.
- The deployment managers have an added `JMX_SOAP_PROXY_ADDRESS` that holds the same configured host or port as their original `JMX_SOAP_ADDRESS`.

As a result of the configuration change, the ODR hosts the logical endpoints for the deployment manager and proxies communication requests for the deployment manager to the active deployment manager.

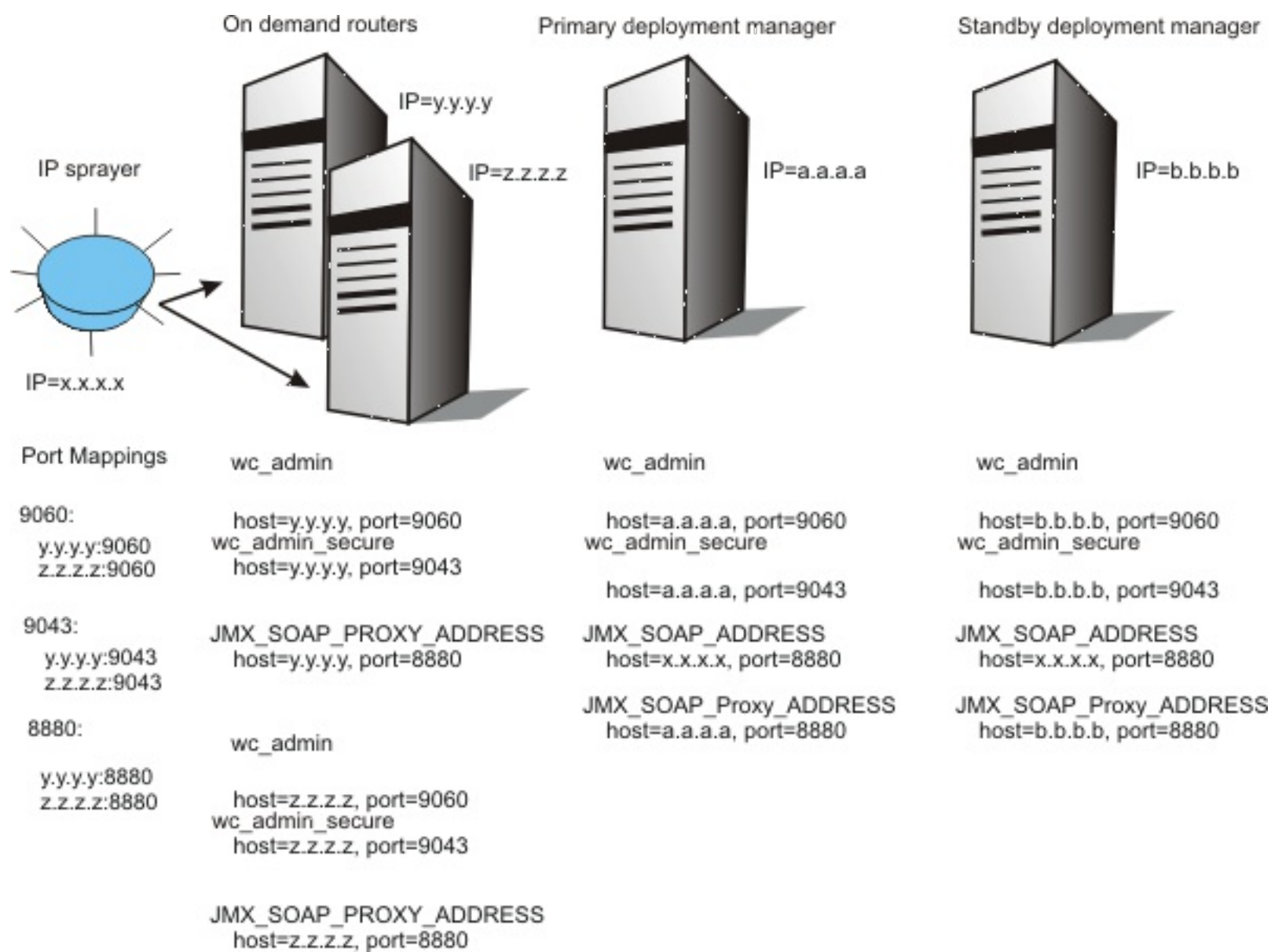
For example in a non-HA ODR configuration, the deployment manager configured host name points to the ODR.

Figure 1. ODR configuration without high availability



To eliminate the ODR as a single point of failure, set up a HA ODR configuration that has at least two on demand routers receiving traffic from an internet protocol spraying device. In this case, the deployment managers JMX_SOAP_ADDRESS configured host name points to the device.

Figure 2. High availability ODR configuration with two ODRs that includes an IP sprayer



Parent topic: [Configuring a high availability deployment manager environment](#)

Related concepts:
[The high availability deployment manager](#)

Related tasks:
[Creating and augmenting profiles](#)
[Using the centralized installation manager repository tool to add products](#)

Related information:
[xd_hadmgrRemove command](#)

xd_hadmgrRemove command

The **xd_hadmgrRemove** command removes a highly available deployment manager from a cell. The command must be run within the deployment manager profile that is being removed.

Syntax

Distributed platforms **z/OS** The command syntax is as follows:

```
xd_hadmgrRemove -hostname (primary_dmgr_host) [-port (primary_dmgr_port)]  
[-user (uid)] [-password (pwd)] [-quiet] [-logfile (filename)]  
[-replacelog] [-trace] [-help]
```

Parameters

The following options are available for the **xd_hadmgrRemove** command:

-hostname <host name>

Specifies the host name that is used to connect to the existing deployment manager.

-port <port>

Specifies the port that is used to connect to the existing deployment manager.

-user <user>

Specifies the user name that is used to connect to the existing deployment manager.

-password <password>

Specifies the password that is used to connect to the existing deployment manager.

-quiet

Suppresses the progress information that the **xd_hadmgrRemove** command prints in normal mode.

-logfile <filename>

Specifies the location of the log file to which information gets written. By default, the log file is called `hadmgrRemove.log` and is created in the `logs` directory of the profile for the node being added.

-replacelog

Replaces the log file instead of appending to the current log. By default, the **xd_hadmgrRemove** command appends to the existing trace file. This option causes the **xd_hadmgrRemove** command to overwrite the trace file.

-trace

Generates additional trace information in the log file for debugging purposes.

-help

Displays syntax help.

Parent topic: [Configuring a high availability deployment manager environment](#)

Related concepts:

[The high availability deployment manager](#)

Related tasks:

[Configuring a high availability deployment manager environment](#)

Related information:

[xd_hadmgrAdd command](#)

Configuring a checkpoint

Repository checkpoints represent saved images of the repository before configuration changes are made and can be used to restore the configuration repository back to a prior state. Checkpoints are either full or delta images. A full checkpoint is created manually by the administrator and is a copy of the entire configuration repository. This includes applications and connectors. Delta checkpoints are optional and are not enabled by default. A delta checkpoint is created automatically when configuration changes are made. The delta checkpoint is formed by making a copy of the configuration documents affected by the configuration change before changes are actually applied.

Before you begin

If you are a user with either a monitor or an operator role, you can only view the repository checkpoint information. If you are a user with either a configurator or an administrator role, you have all configuration privileges for repository checkpoints.

[AIX](#) [HP-UX](#) [Solaris](#) Ensure that you have an adequate number of open file descriptors available. The default number of open files setting is 2000, which is typically sufficient for most applications. If the value set for this parameter is too low, errors might occur when opening files or establishing connections. Because this value limits the number of file descriptors that a server process might open, a value that is too low prevents optimum performance. See [Tuning operating systems](#) for more information.

About this task

To create a full checkpoint, from the administrative console, click **System Administration > Extended Repository Service > under Additional Properties > Repository Checkpoints**. On this page you can create, delete, and restore checkpoints.

Procedure

1. Select **New**. You will be prompted for confirmation before proceeding. While the checkpoint is being created, the repository is locked. You will have read access only to configuration data while the checkpoint is being created. Any attempt to make a configuration change during this period will fail.
2. Name the checkpoint.
3. Type a checkpoint description.
4. Click **OK** to create a full checkpoint.
5. Click **Extended repository service** to return to the Configuration page.
6. To create a delta checkpoint, select the **Enable automatic repository checkpoints** box and define the number of delta checkpoints that you want to keep in the **Automatic Checkpoint Depth** field .
7. Click **Apply** or **OK**.

Results

You configured a checkpoint to back up copies of files from the master configuration repository. If you created a full checkpoint, you made a complete copy of the entire configuration repository. If you enabled delta checkpoints, subset snapshots of the configuration repository are created when you make a change to the configuration.

What to do next

To undo recent changes, restore delta checkpoints in the reverse order in which they were created. If you created a full checkpoint, you can restore the entire configuration repository back to the state it was in at the time the full checkpoint was made.

[The repository checkpoint and restore function](#)

With the repository checkpoint and restore function, you can back up copies of files from the master configuration repository. You can use the backups to restore the configuration to a previous state if future configuration changes cause operational problems. By using this function, you can reduce recovery time for problems that are caused by configuration changes. Studies of unplanned outages have shown that as much as 36 percent of unplanned outages are due to operator errors. A common source of operator error is a bad configuration change. The ability to quickly undo a bad configuration change is critical to minimizing the outage window.

[Restoring a checkpoint](#)

Use a full checkpoint to restore the entire configuration repository back to the state it was in at the time the full checkpoint was made.

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Related concepts:

[The repository checkpoint and restore function](#)

Related tasks:

[Restoring a checkpoint](#)

Related reference:

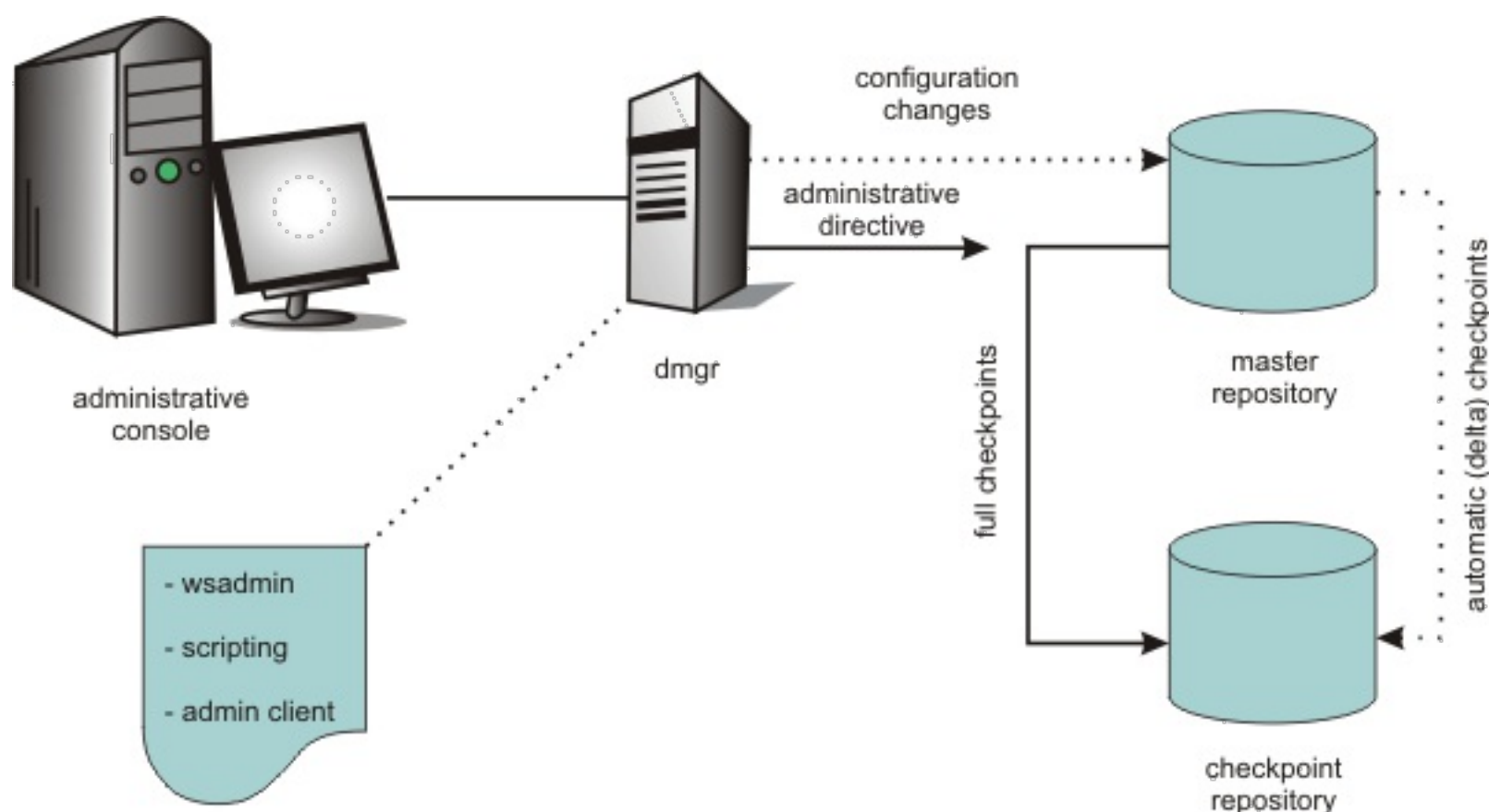
[Administrative roles and privileges](#)

The repository checkpoint and restore function

With the repository checkpoint and restore function, you can back up copies of files from the master configuration repository. You can use the backups to restore the configuration to a previous state if future configuration changes cause operational problems. By using this function, you can reduce recovery time for problems that are caused by configuration changes. Studies of unplanned outages have shown that as much as 36 percent of unplanned outages are due to operator errors. A common source of operator error is a bad configuration change. The ability to quickly undo a bad configuration change is critical to minimizing the outage window.

Two checkpoint types are supported: *full checkpoint* and *delta checkpoint*. A *full checkpoint* is a complete copy of the entire configuration repository. A full checkpoint is useful to take a snapshot of a known working configuration to establish a baseline. Full checkpoints are created manually at administrative discretion. *Delta checkpoints* are created automatically by the system each time a configuration change is made. As the name implies, a delta checkpoint is not a full copy of the configuration, but rather, it is a subset. The subset is comprised of a before-image snapshot of the individual configuration files modified by a discrete configuration change. A configuration **save** marks the end of a discrete configuration change. Delta checkpoints must be restored in the reverse order of their creation to achieve a multilevel undo capability, much like using the undo function in a word processor. Restoring delta checkpoints out of order renders all the other delta checkpoints useless until the automatic checkpoint configuration is reset.

Figure 1. Delta and full checkpoint types



Use the administrative console or scripting to enable or disable automatic delta checkpoints and configure the number of checkpoints to save. When the specified limit is reached, the next automatic checkpoint is saved and the oldest is discarded. Checkpoints are stored in the file system. You can configure the location where checkpoints are stored. To facilitate disaster recovery, you can store checkpoints on a different file system from the system that contains the master configuration repository.

You can use an MBean interface to script and programmatically control checkpoint create and restore operations. See the ExtendedConfigRepository MBean information in the Reference section for more information.

Parent topic: [Configuring a checkpoint](#)

Related tasks:

[Configuring a checkpoint](#)

[Restoring a checkpoint](#)

Restoring a checkpoint

Use a full checkpoint to restore the entire configuration repository back to the state it was in at the time the full checkpoint was made.

Before you begin

Privileges for managing repository checkpoints differ, depending on the administrative role of the user. Roles include monitor, operator, configurator, and administrator. If you are a user with either a monitor or an operator role, you can only view the repository checkpoint information. If you are a user with either a configurator or an administrator role, you have all configuration privileges for repository checkpoints.

AIX **HP-UX** **Solaris** Ensure that you have an adequate number of open file descriptors available. The default number of open files setting is 2000, which is typically sufficient for most applications. If the value set for this parameter is too low, errors might occur when opening files or establishing connections. Because this value limits the number of file descriptors that a server process might open, a value that is too low prevents optimum performance. See [Tuning operating systems](#) for more information.

About this task

Use delta checkpoints to undo recent changes. Delta checkpoints can only be restored in the reverse order in which they were created. Each delta checkpoint has a sequence number. The highest sequence number represents the most recent delta checkpoint. Therefore, delta checkpoints can be restored in descending sequence number only. After the configuration repository is restored from a delta checkpoint, that checkpoint is destroyed. To restore a checkpoint, from the administrative console select **System Administration > Extended Repository Service > Additional Properties > Repository checkpoints**

When you restore a checkpoint, save conflicts will occur if you have uncommitted changes in your workspace. The checkpoint gets restored, but the uncommitted changes are flagged as a *save conflict* should you attempt to save them. Also, if more than one user is working on configuration changes to the repository through the administrative console or otherwise, then other users with uncommitted changes will get save conflicts as well if one user performs a checkpoint restoration.

Procedure

1. Select a repository checkpoint.
2. Click **Restore**. Delta checkpoints must be restored in descending sequence number order only. Note that selecting multiple checkpoints for restoration is not supported. Restore checkpoints one at a time. Select the latest delta checkpoint, the one with the largest sequence number, then restore it. Do this for each checkpoint you want to restore.

What to do next

Before you attempt to verify the success of a checkpoint restoration, you must log out of the administrative console and log in again. This prevents problems or abnormal behavior resulting from workspace issues.

Parent topic: [Configuring a checkpoint](#)

Related concepts:

[The repository checkpoint and restore function](#)

Related reference:

[Administrative roles and privileges](#)

Using centralized logging to diagnose problems

You can use centralized logging to easily enable mustGather traces, perform a per-request trace analysis, and follow the flow of a request through both the ODR and application server tiers.

Before you begin

- Centralized logging is a tool to help you collect the data necessary for troubleshooting and diagnosing a problem. However, for performance reasons, the features of centralized logging are not enabled all the time. Therefore, you must enable the centralized logging features when you anticipate capturing a particular problem.
- When you run the `mustgather.py` script, trace specifications are overwritten. If you want to restore the trace specifications, save the trace strings before you run the script.
- If you use the remote log collection feature of the `mustgather.py` script, ensure that your target file system has the appropriate amount of space necessary to contain the gathered log files.

About this task

Using centralized logging, you can enable tracing based on the type of problem that you experience, for example, 503 HTTP response codes. In previous releases of WebSphere® Virtual Enterprise, particular sets of mustGathers (such as DWLM mustGathers) were commonly requested by IBM® Support teams. However, obtaining those traces required knowledge of the trace settings associated with the requested mustGathers. Now, with centralized logging, specific types of mustGathers can be identified by pre-defined strings. Some examples of pre-defined mustGathers include application editioning, 404 error and 503 response codes, health monitoring, and more.

Procedure

- Run the `mustgather.py` script to enable tracing, collect mustGather documents, and disable tracing. The supported commands are included in the following list:

enable [mustgatherType]

Enables tracing

collect [mustgatherType] [destination]

Collects a specific type of mustGather document

disable [mustgatherType]

Disables tracing

1. Determine the type of mustGather document that you need. The supported types of mustGather documents are included in the following list:

404

404 HTTP response code

503

503 HTTP response code

504

504 HTTP response code

agent

Middleware agent

appedition

Application edition manager

arfm

Autonomic request flow manager

dc

Dynamic cluster

hadmgr

High availability deployment manager

hmm

Health monitoring

odr

On demand router

operations

Visualization issues with the Extended Deployment and Operations tabs

reports

Visualization issues with the Reports tab

reportsPerf

Visualization issues with performance data that is displayed on the Reports tab

repository

Extended Repository Service

sip

SIP request routing

2. Run the **enable** command with the appropriate mustGather type specified to set the appropriate tracing on all relevant servers in the cell.

```
wsadmin -lang jython -f c:\WebSphere\AppServer\bin\mustgather.py
enable 404
```

3. Recreate the desired scenario.
4. Run the **collect** command with the appropriate mustGather type and a destination file specified to collect the local and remote mustGather documents.

```
wsadmin -lang jython -f c:\WebSphere\AppServer\bin\mustgather.py
collect 404
"c:\mydocs\collection.zip"
```

5. Run the **disable** command with the appropriate mustGather type specified. Running this command sets the trace to *=info on all relevant servers in the cell.

```
wsadmin -lang jython -f c:\WebSphere\AppServer\bin\mustgather.py
disable 404
```

- Run the setupReqBasedTracing.py script to enable or disable request-based tracing rules. The supported commands are included in the following list:

enableReqBasedTracing

Sets up a request-based tracing rule. A rule consists of an expression and identifier called a rule ID. Optionally, a rule can also contain ODR trace and application server trace strings. Start and end markers are placed in the log files for requests that match one or more rules.

The ODR logs a start marker when a request that matches one or more rules arrives and logs an end marker before the request is dispatched to the backend application server. The ODR also logs a start marker when a response is received from the application server and an end marker after response is sent back to the user. The application server logs a start marker when a matched request arrives from the ODR and an end marker before a response is sent to the ODR. You can use these markers to find a particular request, or set of requests, and correlate them with processing on the application server.

ListRuleIDs

Lists all the rules. This command will output all the rules that are set on all the ODRs. After an ODR is restarted, the rules must be recreated.

disableReqBasedTracing

Disables a request-based tracing rule.

1. Enable request-based tracing.

```
./wsadmin.sh -lang jython -f setupReqBasedTracing.py
enableReqBasedTracing
-ruleExpression:<expression> -odrTraceSpec:<trace strings>
-appServerTraceSpec:<trace string> -ruleID:<rule ID>
```

where

-ruleExpression:<expression>

Specifies an expression that is used to match the requests. (Required)

-odrTraceSpec:<trace string>

Specifies an ODR trace string that is set at run time for the requests that match the specified expression. If the parameter is not specified, the trace specification is not dynamically set. (Optional)

-appServerTraceSpec:<trace string>

Specifies an application server trace string that is set at run time for the requests that match the specified expression. If the parameter is not specified, the trace specification is not dynamically set. (Optional)

-ruleID:<rule ID>

Specifies the ID for the request-based tracing rule. If the parameter is not specified, a rule ID is generated by the script in the form of ruleID-<time stamp>. (Optional)

2. List all the rules.

```
./wsadmin.sh -lang jython -f setupReqBasedTracing.py listRuleIDs
```

3. Disable request-based tracing.

```
./wsadmin.sh -lang jython -f setupReqBasedTracing.py  
disableReqBasedTracing  
-ruleIDs:<rule ID1>,<rule ID2>...,<ruleIDn>
```

where

-ruleIDs:<rule ID1>,<rule ID2>...,<ruleIDn>

Specifies a list of rule IDs to disable. (Required)

For more information on rule expressions, see [HTTP operands](#).

Parent topic: [Managing the WebSphere Virtual Enterprise environment](#)

Monitoring operations

You can easily monitor the status of your environment. With the operational summaries, you can identify where applications, servers, or autonomic managers are running in your environment, the health of your environment, and if your environment is performing to your service-level agreements.

Before you begin

Configure your cell to autonomically manage your environment. See [Preparing the hosting environment for dynamic operations](#) for more information.

[Runtime operations overview](#)

With WebSphere Virtual Enterprise, you can manage complex system operations with real time, meaningful visualization tools. Gradually controlled implementation of autonomic capabilities helps you to reduce the cost of managing your resources.

[Creating and managing reports](#)

With reports, you can view the performance of your virtualized environment. You can view statistics such as availability, response time, traffic, or throughput.

[Configuring the visualization data service](#)

The visualization data service logs historic data in text files for reuse with other charting programs. Historic data is logged in comma-separated values with time stamps in standard long value from the java.util.Date class. By using the visualization data service, you can log historical data, calculate charge back values, or perform capacity planning.

[Task management](#)

You can be notified about decisions that are made by autonomic managers. Notifications can represent either planned or unplanned events.

[Troubleshooting extended administration](#)

Occasionally, when you are using the extended manageability features, you might encounter behavior that is not expected. For example, you can correct Web browser configurations so that your Web browser can support the visualization function.

Related tasks:

[Installing and customizing the product](#)

[Migrating WebSphere Virtual Enterprise](#)

[Preparing the hosting environment for dynamic operations](#)

[Deploying and managing application editions without loss of service](#)

[Managing the WebSphere Virtual Enterprise environment](#)

[Configuring WebSphere Virtual Enterprise to work with other IBM products](#)

Related reference:

[Reference](#)

Related information:

[IBM WebSphere Virtual Enterprise Version 7.0](#)

[Product overview](#)

[Site map](#)

[Runtime operations custom properties](#)

Runtime operations overview

With WebSphere® Virtual Enterprise, you can manage complex system operations with real time, meaningful visualization tools. Gradually controlled implementation of autonomic capabilities helps you to reduce the cost of managing your resources.

Operational alerts

Operational alerts inform you of the current state of your environment, including any issues, so that you can take action if necessary. For example, you might see an operational alert when a service policy is breached. For notification about issues that are continuing to occur over multiple intervals, runtime tasks are generated.

Reports

You can use customized charting to show if goals are being met. To benefit from this virtualized environment, you must know how your applications are performing. Dynamic charting provides a visual perspective of application performance. Specifically, statistics such as availability, response time, traffic, and throughput are supported. A wide range of options from which you can create various charts is also provided.

You can organize your charts into *chart groups* to easily access charts from any reports view.

You can move charts to a new window to continue viewing the chart while performing other tasks in the administrative console. You can also move the chart back to the original chart group.

You can also view reports in the Reports tab of the following settings panels:

- Applications
- Dynamic clusters
- Clusters
- Servers
- Service policies

Runtime operations reports use Scalable Vector Graphics (SVG) to display the data. Microsoft Internet Explorer does not have an SVG viewer installed by default. You can install a SVG plug-in or configure runtime operations to display JPEG images. See [Creating and managing reports](#) for more information.

The runtime operations reports are blank until you begin running an active load against your cell.

Dashboard

The dashboard displays a high level summary of your overall environment. You can use the dashboard preferences to configure the information that displays in the dashboard, such as the default reports or the data from a specified chart group.

Runtime summaries

You can view the runtime information for the on demand routers, nodes, core groups, applications, deployment targets, service policies, and core components in your environment. Core components include a variety of autonomic controllers and managers. The runtime summaries include a list of instances for the particular resource type, status, stability and other information.

The charts that are included in the applications, deployment targets, and service policies summary views do not automatically refresh. To update the data, you can click the refresh icon on the chart. These charts display the average response time at a certain point in time. The default number of data sets that are displayed is 5. You can, however, configure custom properties to specify that a unique number of data sets be displayed instead. See [Runtime operations custom properties](#) for more information.

If you are not running any work load requests through the on demand router (ODR) to your applications, no data is displayed in the chart. Also, if you are charting service policy related metrics, no goal lines for the data set are displayed in the key. After the ODR sends the work load requests, the chart is automatically updated and the statistics are displayed. However, the key table is not automatically updated. Refresh the panel to view the goal lines that are added to the data set.

To access the runtime summaries, click **Runtime Operations** in the administrative console.

Operations tabs

The operations tabs show the overall status of the selected item in the following settings panels:

- Applications
- Dynamic clusters
- Clusters
- Servers
- Core groups

Parent topic: [Monitoring operations](#)

Related tasks:

[Monitoring operations](#)

[Creating and managing reports](#)

[Configuring the visualization data service](#)

Related reference:

[Administrative roles and privileges](#)

Creating and managing reports

With reports, you can view the performance of your virtualized environment. You can view statistics such as availability, response time, traffic, or throughput.

Before you begin

- Your servers, clusters, applications, and so on must be running to display data in the charts.
- **Windows** By default, runtime operations reports use Scalable Vector Graphics (SVG) charts. Microsoft Internet Explorer does not have an SVG viewer installed by default. You can install the SVG viewer plug-in or update the reports preferences to display JPEG images. To update the reports preferences, click **Runtime operations > Reports > Reports preferences**. Edit the **Default chart format** field.

About this task

Reports are charts that show runtime data. You can use this data to monitor your environment, and take correctional actions when necessary.

You can view reports which display live data, or you can use charts displaying historic data logged over a period of days, weeks, months, years. In order to use historic charting, configure and enable the visualization data service. For more information, read about configuring the visualization data service. For historic charting, use the sliding bar to adjust the percentage of time to view. You can focus on a specific time interval and include more data in the chart.

Note: To observe historic data over a long period of time, increase the maximum heap size of the deployment manager. The default maximum heap size is 256 MB. To increase the performance, and to prevent heap dumps, go to **System administration > Deployment manager > Java and process management > Process definition > Java Virtual Machine** and set the maximum heap size to a higher value, for example 2048 MB.

Procedure

1. Open the **Reports** tab. In the administrative console, you can access the **Reports** tab from:
 - **Runtime operations > Reports**
 - **Servers > All servers > server_name > Reports**

Restriction: The **Reports** tab is not available for on demand router (ODR) servers.

 - **Servers > Clusters > Dynamic clusters > dynamic_cluster_name > Reports**
 - **Servers > WebSphere Application server clusters > cluster_name > Reports**
 - **Applications > All application > application_name > Reports**
 - **Operational policies > Service policies > service_policy_name > Reports**
2. Act on operational alerts. Operational alerts display the status of resources in your environment. The name of the specific resource is highlighted. Go to the configuration panel or view the chart for that resource to take corrective actions.
3. Configure report and chart preferences. Expand **Reports preferences**. The preferences are global and apply to any new charts you create. However, the preference change does not apply to existing charts. To change the preferences for an existing chart, click **Preferences** on the chart.
4. Add a chart. Click **Open a new chart tab**. A new tab opens with a blank chart. Click **Add data...** to specify the data set and metrics that you want to monitor. If you accessed the **Reports** tab through the **Runtime operations > Reports** panel, you can specify the scope of the chart. Click **Change scope...**
5. Create, access, and remove chart groups. Chart groups are global and can be accessed from any **Reports** tab.
 - **Create:** If you have a specific set of charts in the tabs that you want to save, then type a name in **Save current group of chart tabs configuration as a chart group**, and click **Save**.
 - **Access:** To access the chart group from any reports panel later, go to **Saved chart groups**, and click the name of the chart group.

You can also open a chart in a new window, so that you can continue doing other tasks in the administrative console while still monitoring your chart. Click **View chart in new window**. To return the chart from the new window to the chart group, click **View chart in chart group**.

 - **Remove:** To remove a chart group, select the chart group from **Saved chart groups**, and click **Remove chart group**.

What to do next

If your operational alerts are taking a long time to load, you can configure custom properties to change how the alerts are loaded while the cache initializes.

You can configure custom properties to specify the maximum number of data sets that are displayed or to limit what cell information is displayed.

Next topic: [Operational alerts custom properties](#)

Next topic: [Runtime operations custom properties](#)

Parent topic: [Monitoring operations](#)

Related concepts:

[Runtime operations overview](#)

[Task management](#)

Related tasks:

[Preparing the hosting environment for dynamic operations](#)

[Monitoring operations](#)

[Configuring the visualization data service](#)

Related reference:

[Adobe SVG Viewer download area](#)

Configuring the visualization data service

The visualization data service logs historic data in text files for reuse with other charting programs. Historic data is logged in comma-separated values with time stamps in standard long value from the `java.util.Date` class. By using the visualization data service, you can log historical data, calculate charge back values, or perform capacity planning.

Before you begin

You must use the deployment manager to implement this feature. Ensure that if you are using multiple core groups, they are correctly bridged.

If you are a user with either a *monitor* or an *operator* administrative role, then you can only view the visualization data service information. If you have a *configurator* administrative role, then you can change the configuration. If you have an *administrator* role, then you have all the privileges for the visualization data service.

About this task

Attention: You must configure the visualization data service before you enable logging. If you need to make changes to the configuration after you have enabled logging, you must restart the deployment manager after you make the configuration changes.

Procedure

1. In the administrative console, click **System administration > Visualization data service**.
2. Enter a value in the **Time stamp format** field. The time stamp format specifies a time and date pattern that is used when logging the visualization data. Use the `SimpleDateFormat` Java™ class to format your time stamp. For example, to output the 12.06.2006 5:26:30:978 PM PDT timestamp, use the following time stamp format value:

```
MM.dd.yyyy hh:mm:ss:SSS aaa z
```

If you are using IBM® Tivoli® Usage and Accounting Manager, then use a format that separates the date and time into different fields:

```
'MM.dd.yyy, hh:mm:ss:SSS'  
'yyyy.MMMMM.dd, hh:mm:ss'
```

You can also specify the time stamp format with the `wsadmin` tool:

```
wsadmin.sh -lang jython  
wsadmin>> vds = AdminConfig.getid("/Cell:OpsManTestCell/VisualizationDataService/")  
wsadmin>> vdl = AdminConfig.showAttribute(vds,"visualizationDataLog")  
wsadmin>> AdminConfig.modify(vdl,[["timestampFormat","MM.dd.yyyy hh:mm:ss:SSS aaa z"]])  
wsadmin>> print AdminConfig.show(vdl)  
wsadmin>> AdminConfig.save()
```

3. In the **Maximum file size** field, type a whole integer for the maximum file size for logs.
4. In the **Maximum number of historical files** field, type a whole integer for the maximum number of logs to be generated per historic cache type.
5. In the **File name** field, type the path where the log files are generated. You can use a variable in the file name value, for example: `${LOG_ROOT}/visualization`.
6. In the **Data log write interval** field, type a whole number between 1 and 365 for the interval in which the logs are generated in seconds, minutes, hours, or days.
7. From the **Data transformer** action list, select **Average** or **Skip** to specify how to transform data when the interval reaches its maximum value. More data points are provided than you might want to use. The **AVERAGE** option averages the existing data points between the specified interval, and the **SKIP** option skips the data points to only use the points specifically at the intervals.
8. Select **Enable log** to start logging historic data.
9. If logging was enabled before you configured the visualization data service, restart your deployment manager.

Results

Operational data is exported to the file name that you specified.

What to do next

Now that you have configured the visualization data service, you can import data into an external charting program.

Parent topic: [Monitoring operations](#)

Related tasks:

[Monitoring operations](#)

[Configuring communication between core groups that are in the same cell](#)

Related reference:

[Logs](#)

[Administrative roles and privileges](#)

[SimpleDateFormat class API documentation](#)

Task management

You can be notified about decisions that are made by autonomic managers. Notifications can represent either planned or unplanned events.

Planned events

Planned events are events for which the runtime environment has an action plan. For example, you might have a health policy defined that has an average response time that is breaching its configured limit, which might trigger an increased footprint of a dynamic cluster. If the product is operating in automatic mode, then the action plan runs, and you can view a notification of the action that was taken. In supervised mode, you can view and approve the action plan. The interaction modes can vary in your configuration. A mix of automatic and supervised mode activities by dynamic clusters and health policies can exist, for example.

Unplanned events

If an event occurs that is not assigned to an action plan, then you can be notified that something unexpected has happened. After you receive the notification, develop a plan to correct the situation, if it is indeed problematic.

Runtime tasks

A runtime task is generated when an event occurs. Runtime tasks provide information from which you can accept, deny or close the recommended action plan. Tasks that have taken action or expired tasks stay in the runtime task list for 24 hours by default. You can change this default by setting a cell level custom property. See the [Runtime task custom properties](#) topic to learn how to set the custom property.

Runtime tasks reside in the deployment manager's memory, and are logged in `<was_root>/profiles/<Dmgr_profile_name>/tmsStorage` as well. When the deployment manager is restarted, the runtime task entries are read into the deployment manager's memory from `tmsStorage`.

Event logging

You can enable logging for events. See the [Task management service event logger](#) topic for more information.

Notifications

You can have e-mail notifications sent to specified users when runtime tasks occur. See [Defining email notification](#) for more information.

[Managing runtime tasks](#)

A runtime task is generated by a runtime component within the product. Runtime tasks provide information from which you can accept, deny or close the recommended action plan.

[Task management service event logger](#)

You can enable an event logger that logs all Java Management Extensions (JMX) notifications from the TaskManagement MBean into its own log file.

[Defining email notification](#)

In addition to actively monitoring tasks with the Runtime Tasks panel, you can also have email notifications sent to specified users when runtime tasks occur.

Parent topic: [Monitoring operations](#)

Related tasks:

[Monitoring operations](#)

[Managing runtime tasks](#)

Managing runtime tasks

A runtime task is generated by a runtime component within the product. Runtime tasks provide information from which you can accept, deny or close the recommended action plan.

Before you begin

Configure your hosting environment. For more information, see [Preparing the hosting environment for dynamic operations](#).

About this task

Use runtime tasks to view and manage the recommendations that the autonomic controllers are providing to improve the health and performance of your environment. If you are running in supervised mode, you must accept, deny, or close each runtime task. In automatic mode, the autonomic controllers automatically take these actions.

A task is lost when it is sent from a node when the deployment manager is not running. The same holds true in a high availability deployment manager environment. Even though the down time is minimal between the time when an active highly available deployment manager shuts down and when the standby deployment manager becomes active, tasks are still lost. After a failover of a highly available deployment manager occurs, the executing tasks at the original active deployment manager are changed to the unknown state.

Procedure

1. From the administrative console, click **System administration > Task management > Runtime tasks**.
2. Click the task ID link to view the task target objects and corresponding monitors of a specific task. A task target object might be a server, cluster, service policy, node, health policy, application, and so on. One of these objects might be linked to configuration, performance or log monitor views in the administrative console.

With a configuration monitor, the placement controller might recommend that an instance of the TestCluster dynamic cluster start on the test4 node. The recommendation states that this action is required for the dynamic cluster to meet its configured minimum number of running instances. You can click the link to the configuration panel for the TestCluster dynamic cluster to view its settings and verify or change the minimum number of running cluster instances.

For a performance monitor, you can click a link to a chart with data that is specified in the monitor and specific to the target object. For a log monitor, you can click the **Runtime** tab of the target object Java™ virtual machine logs to investigate the displayed logs.

3. To view the action plan for the task, click the Task ID.
4. To act on a specific task, select the corresponding task, and from the action list select **Accept**, **Deny** or **Close**. Accepting the task commits the previewed action plan. Closing the task means that the task is successfully handled by the task management service or is manually closed. Denying the task places the task in inactive status if the task is not submitted again in the next batch of task submissions by the originating component. You can also select multiple tasks with the same actions. After you act on a task, the action list is unavailable for that task.
5. Click **Submit**.

Results

The **Runtime Tasks** panel refreshes, and performs the actions that you selected to perform on each task.

Parent topic: [Task management](#)

Related concepts:
[Task management](#)

Related tasks:
[Preparing the hosting environment for dynamic operations](#)
[Defining email notification](#)

Task management service event logger

You can enable an event logger that logs all Java™ Management Extensions (JMX) notifications from the TaskManagement MBean into its own log file.

The event logger is a value add to the WebSphere® Application Server Network Deployment's Task Management Service. It logs all JMX notifications: tasknew, taskstatechange, taskseveritychange, taskstatuschange. The log for every task event contains the following fields:

Table 1. Event log fields

Field	Description
timestamp	The time when the event was received and logged.
taskID	The global ID of the task that the event concerns.
originatedTime:	The time the task was originated.
cell	The cell that the task event was received in.
notification	The type of JMX event / notification that was received.
submitter	The originator of the event.
reasonMsg	The task's reason message.
taskType	The type of task.
severity	The global severity of the task.
actions	Any actions associated with the task (if applicable).
state	The task's lifecycle state.
status:	The status of a terminated task.
statusMsg	The status message related to the status of a terminated task.

Enabling and configuring the event logger

The event logger must be enabled and can be configured through cell level custom properties. To access the event logger, go to **System Administration > cell > Custom Properties**. The following table describes the name and default values for the event logger.

Table 2. Names and default values for the event logger

Property Name	Default value	Description
tmslog.delim		The delimiter to use between fields in the log file.
tmslog.enable	false	Whether or not to enable logging of TMS events.
tmslog.fileLocation	\${LOG_ROOT}	The location of where the log file will be written out to.
tmslog.maxFiles	1	The maximum number of historic files to keep.
tmslog.maxSize	20	The maximum log file size to allow before rolling log files (in MB)
tmslog.timeFormat	empty string: ""	The format to use when logging timestamp and originatedTime. Reference the javadoc for SimpleDateFormat for information on how to construct a timestamp pattern. If you want the time to span fields, such as to separate date and time into different columns, then use the delimiter in the pattern. If you do not specify a value for the custom property, the epoch time format is used.

The custom properties are not recognized during runtime Restart the deployment manager to pick up any changes to the custom properties.

Parent topic: [Task management](#)

Related tasks:

[Monitoring operations](#)
[Managing runtime tasks](#)

Defining email notification

In addition to actively monitoring tasks with the Runtime Tasks panel, you can also have email notifications sent to specified users when runtime tasks occur.

Before you begin

When security is enabled, some fields are not available without proper authorization. You must have *configurator* administrative privileges to define email notification.

About this task

By defining email notification, you can specify a set of email addresses to be notified when runtime tasks are generated.

The sender user ID is preset to wasxd. You might need to register the sender user ID to in your Simple Mail Transfer Protocol (SMTP) registry before the cell can successfully send email notifications.

Procedure

1. Optional: Set the user ID from which the emails are sent.

Requirement: To change the user ID, you must install the appropriate fix pack for your WebSphere® Virtual Enterprise release and your WebSphere Application Server Network Deployment release:

Table 1. Required interim fixes to use custom user IDs for email notification

WebSphere Virtual Enterprise Version	WebSphere Application Server Network Deployment Version	Required interim fixes
6.0.2.1	6.0.2.x	PK55085
6.0.2.1	6.1.0.11 or 6.1.0.13	PK54881 and PK55085
6.0.2.1	6.1.0.15 or later	PK55085
6.1.0.0	6.1.0.11 or 6.1.0.13	PK54881 and PK55085
6.1.0.0	6.1.0.15 or later	PK55085
6.1.0.1	6.1.0.11 or 6.1.0.13	PK54881
6.1.0.1	6.1.0.15 or later	No interim fix is required

By default, the value is preset to wasxd. To change this default, perform the following steps:

- a. In the administrative console, click **System administration > Cell > Custom properties > New**.
 - b. Enter the name of the custom property as `task.email.global.sender.id`, and set the value to the specific email user ID that you want to use.
 - c. Click **Apply**. The change becomes effective when you save the configuration.
2. Configure email notification in the administrative console. Select **System administration > Task management > Notifications**. Select **Enable notifications**. When runtime tasks are generated, an email notification is sent to each of the specified email addresses.
 3. Verify that email is sent for a given task. When all of the changes are made, click **Test Email** to verify that the email is sent for a given task. If the email server uses spam-blocking software, this test can fail and prevent the email from being displayed in the test inbox. To save your changes, click **Apply** or **OK**.

Results

When notification is enabled, an email is sent to each of the specified addresses when a task is generated. When notification is enabled and no email addresses are specified, no emails are sent.

What to do next

- You can configure custom properties to get more detailed task information in the emails or to customize the subject line in the emails.
- Manage the runtime tasks in the administrative console.

Next topic: [Managing runtime tasks](#)

Parent topic: [Task management](#)

Related concepts:

[Task management](#)

Related reference:

[Administrative roles and privileges](#)

Related information:

[Runtime task custom properties](#)

Troubleshooting extended administration

Occasionally, when you are using the extended manageability features, you might encounter behavior that is not expected. For example, you can correct Web browser configurations so that your Web browser can support the visualization function.

Linux

Visualization is not working using Mozilla software on Linux systems

For visualization to function correctly, some Linux libraries are required to view the runtime chart. Without these libraries, your environment might not have the visual enhancements that are available in the administrative console.

If you are not able to view the runtime charts, verify that you have the following shared library:

```
/usr/lib/libstdc++-libc6.2-2.so.3
```

This library is not installed by default for Red Hat Enterprise Linux 3.0. The following package contains this file:

```
compat-libstdc++-7.3-2.96.122.i386.rpm
```

Charting not reflecting expectations

Check whether the service policy goals are being met, and that the relative importance of different service policies is respected. Most likely, the system is working as designed. However, to look at or change a service policy, click **Operational policies > Service policies > *service_policy_name*** in the administrative console.

Runtime operations reports are empty

To see data in the runtime operations reports, you must run an active load against your cell.

Parent topic: [Monitoring operations](#)

Related tasks:

[Monitoring operations](#)

[Creating and managing reports](#)

Configuring WebSphere Virtual Enterprise to work with other IBM products

WebSphere® Virtual Enterprise can work with other IBM® products.

Procedure

Configure the on demand router (ODR) to make Application Response Measurement (ARM) calls to report response times to IBM Enterprise Workload Manager™. IBM Enterprise Workload Manager can use the response time information to monitor the environment both inside and outside of the WebSphere Virtual Enterprise domain. See [Enabling the on demand router to work with IBM Enterprise Workload Manager](#) for more information.

[Enabling the on demand router to work with IBM Enterprise Workload Manager](#)

Use this task to enable IBM Enterprise Workload Manager to monitor WebSphere Virtual Enterprise.

Related tasks:

[Installing and customizing the product](#)

[Migrating WebSphere Virtual Enterprise](#)

[Preparing the hosting environment for dynamic operations](#)

[Deploying and managing application editions without loss of service](#)

[Managing the WebSphere Virtual Enterprise environment](#)

[Monitoring operations](#)

Related reference:

[Reference](#)

Related information:

[IBM WebSphere Virtual Enterprise Version 7.0](#)

[Product overview](#)

[Site map](#)

Enabling the on demand router to work with IBM Enterprise Workload Manager

Use this task to enable IBM® Enterprise Workload Manager™ to monitor WebSphere® Virtual Enterprise.

Before you begin

To perform this task, you must own IBM Enterprise Workload Manager software and have it installed. See the IBM Enterprise Workload Manager Information Center for installation instructions.

About this task

By performing this task, the on demand router (ODR) makes Application Response Measurement (ARM) calls to report response times to IBM Enterprise Workload Manager. IBM Enterprise Workload Manager can use the response time information to monitor the environment both inside and outside of the WebSphere Virtual Enterprise domain. See [WebSphere Virtual Enterprise and IBM Enterprise Workload Manager](#) for more information.

Procedure

1. Install WebSphere Virtual Enterprise. See [Planning the product installation](#) for more information.
2. Configure service and transaction classes in IBM Enterprise Workload Manager. See the Enterprise Workload Manager Information Center for more information. The service and transaction classes that you configure in IBM Enterprise Workload Manager override any service classes that are configured in WebSphere Virtual Enterprise.
3. Configure IBM Enterprise Workload Manager to manage WebSphere Virtual Enterprise. After you complete this step, any service policies that are configured in WebSphere Virtual Enterprise are not valid. IBM Enterprise Workload Manager has its own configured service and transaction classes.
4. Copy the ARM libraries from IBM Enterprise Workload Manager to the WebSphere Virtual Enterprise configuration. Include the ARM libraries from IBM Enterprise Workload Manager in the `install_root/bin` directory. Remember the name of this class so that you can supply the information in the administrative console. Alternatively, you can add two custom properties to the application server or ODR Java™ virtual machine (JVM). First, define the `java.library.path` custom property and set its value to the IBM Enterprise Workload Manager library path, for example `c:\Program Files\IBM\VE2\EWLMMS\classes\ms`. Next, create the `ws.ext.dirs` custom property and set its value to the folder that contains the ARM libraries that are provided by IBM Enterprise Workload Manager, for example, `c:\Program Files\IBM\VE2\EWLMMS\classes\ARM`.
5. Enable ARM reporting. Read the *Getting performance data from request metrics* topic in the WebSphere Application Server Information Center for more information. On the request metrics administrative console page, use the following settings:
 - Select the appropriate application component, such as servlets or Web services, that needs to report ARM statistics.
 - Defining the trace level is optional.
 - Enter the name of the IBM Enterprise Workload Manager ARM factory in the ARM transaction factory implementation class name. The transaction class factory name for IBM Enterprise Workload Manager is `com.ibm.wlm.arm40SDK.transaction.Arm40TransactionFactory`.
6. Enable ARM reporting on your other tiers and applications, such as IBM DB2®, IBM HTTP Server, and also for the various components running on WebSphere Application Server. You can enable ARM instrumentation in specific WebSphere Application Server components by selecting the appropriate components. Read the *Getting performance data from request metrics* topic in the WebSphere Application Server Information Center for more information. See the product documentation for your specific tier applications regarding more information on enabling ARM reporting. Any tiers that send requests to the WebSphere Virtual Enterprise tier must be ARM enabled to facilitate the interaction between IBM Enterprise Workload Manager and WebSphere Virtual Enterprise. Any tiers that receive requests from the WebSphere Virtual Enterprise tier do not need to have ARM enabled, however, enabling ARM on these tiers is strongly encouraged.
7. To enable WebSphere Application Server and WebSphere Virtual Enterprise with IBM Enterprise Workload Manager, you must manually update the WebSphere Application Server `server.policy` files. You must update the `server.policy` file for all application servers and on demand routers. See the *server.policy file permissions* topic in the WebSphere Application Server Information Center for more information about `server.policy` files.
8. Restart all applications and servers that IBM Enterprise Workload Manager is now monitoring.

Results

After you complete this task, the on demand routers in the WebSphere Virtual Enterprise infrastructure begin to make ARM calls to report response times on requests, so IBM Enterprise Workload Manager can monitor the WebSphere Virtual Enterprise environment.

[WebSphere Virtual Enterprise and IBM Enterprise Workload Manager](#)

You can enable WebSphere Virtual Enterprise to work with other autonomic managers, such as IBM Enterprise Workload Manager.

Parent topic: [Configuring WebSphere Virtual Enterprise to work with other IBM products](#)

Related concepts:

[WebSphere Virtual Enterprise and IBM Enterprise Workload Manager](#)

Related tasks:

[Configuring the health controller](#)

[Configuring the autonomic request flow manager](#)

☞ [Getting performance data from request metrics](#)

Related reference:

☞ [server.policy file permissions](#)

Related information:

☞ [IBM Enterprise Workload Manager Information Center](#)

WebSphere Virtual Enterprise and IBM Enterprise Workload Manager

You can enable WebSphere® Virtual Enterprise to work with other autonomic managers, such as IBM® Enterprise Workload Manager™.

If you enable the on demand router (ODR) to report request metrics, Application Response Measurement (ARM)-enabled autonomic managers like Enterprise Workload Manager can influence how requests are processed in WebSphere Virtual Enterprise. Enterprise Workload Manager has knowledge and control outside of the WebSphere Virtual Enterprise domain. Enterprise Workload Manager can use the request metrics to determine if service class policies and transaction class policies that are defined in the Enterprise Workload Manager configuration are met. If you need to make changes to meet the requirements, Enterprise Workload Manager uses its knowledge and control outside of WebSphere Virtual Enterprise to assist WebSphere Virtual Enterprise in meeting the broader response time goals for Enterprise Workload Manager.

Support of ARM instrumentation in the ODR enables the reporting of response time metrics on a per transaction basis. ARM is an industry standard for accurate measurements of request and response transactions as they move through various tiers of applications. For more information about ARM and request metrics, see the WebSphere Application Server Network Deployment Information Center.

Any tiers that send requests to WebSphere Virtual Enterprise must be ARM-enabled to facilitate the interaction between Enterprise Workload Manager and WebSphere Virtual Enterprise. Any tiers that receive requests from WebSphere Virtual Enterprise do not need to have ARM enabled, however, enabling ARM on these tiers is encouraged. For example, enable ARM on all application servers in software that works in cooperation with WebSphere Virtual Enterprise, such as IBM DB2® and so on. Enterprise Workload Manager can make resource level measurements for tiers that do not have ARM enabled. You can also enable ARM reporting on any Web servers that forward requests to the ODR.

For more information about the IBM Enterprise Workload Manager, see the IBM Enterprise Workload Manager Information Center.

Parent topic: [Enabling the on demand router to work with IBM Enterprise Workload Manager](#)

Related tasks:

[Enabling the on demand router to work with IBM Enterprise Workload Manager](#)

[Getting performance data from request metrics](#)

Related information:

[IBM Enterprise Workload Manager Information Center](#)

Reference

Reference information is organized to help you locate particular facts quickly.

[Release notes](#)

Links are provided to the product support web site, to product documentation, and to last minute updates, limitations, and known problems for the product.

[Port number settings](#)

Identify the default port numbers used in the various configuration processes of WebSphere Virtual Enterprise to avoid port conflicts.

[Scripts](#)

You can use the WebSphere Virtual Enterprise scripts after you install the product.

[Administrative roles and privileges](#)

Roles and privileges vary depending on your administrative role and the component.

[Administrator scripting interfaces](#)

This reference information provides examples of how to complete various tasks using the administrative scripting interfaces.

[Custom properties](#)

Custom properties are unique settings in the administrative console. They are name-value pairs that you can enter on specific administrative console pages. Use the documentation for the specific custom property to determine where the property must be set.

[Logs](#)

You can use log files to help with performance monitoring, accounting and problem determination.

[Messages](#)

When you encounter a message in a log or other parts of the product interface, look up the message by its component prefix to find out more information.

Related tasks:

[Installing and customizing the product](#)

[Migrating WebSphere Virtual Enterprise](#)

[Preparing the hosting environment for dynamic operations](#)

[Deploying and managing application editions without loss of service](#)

[Managing the WebSphere Virtual Enterprise environment](#)

[Monitoring operations](#)

[Configuring WebSphere Virtual Enterprise to work with other IBM products](#)

Related information:

[IBM WebSphere Virtual Enterprise Version 7.0](#)

[Product overview](#)

[Site map](#)

Release notes

Links are provided to the product support web site, to product documentation, and to last minute updates, limitations, and known problems for the product.

- [Accessing last-minute updates, limitations, and known problems](#)
- [Accessing system and software requirements](#)
- [Accessing product documentation](#)
- [Accessing the product support Web site](#)
- [Contacting IBM Software Support](#)

Accessing last-minute updates, limitations, and known problems

The release notes are available on the product support site as technotes. To see a list of all the technotes for WebSphere® Virtual Enterprise, go to the [Support web page](#).

- To see a list of the release notes for Version 7.0, including all fix packs, go to the [Support web page](#).
- To see a list of the release notes for Version 6.1.1, including all fix packs, go to the [Support web page](#).
- To see a list of the release notes for Version 6.1, including all fix packs, go to the [Support web page](#).

Accessing system and software requirements

For details on the hardware and software requirements, see the [Detailed system requirements](#).

Accessing product documentation

For the entire information set, see the [Library page](#).

Accessing the product support Web site

To search for the latest technotes, downloads, fixes, and other support-related information, see the [Support page](#).

Contacting IBM Software Support

If you encounter a problem with the product, first try the following actions:

- Follow the steps described in the product documentation
- Look for related documentation in the online help
- Look up error messages in the message reference

If you cannot resolve your problem by any of the preceding methods, contact IBM® Technical Support.

Parent topic: [Reference](#)

Port number settings

Identify the default port numbers used in the various configuration processes of WebSphere® Virtual Enterprise to avoid port conflicts.

Open certain ports in any firewalls that are running between the deployment manager and node agent server processes to avoid conflicts with other assigned ports when configuring Version 6.1 resources or assigning port numbers to other applications. Additionally, when you configure a firewall, enable access to specific port numbers.

If you modify the ports, or if you want to confirm the assigned port, check the port assignments in the `Install_Root\config\cells\cellname\nodes\nodename\serverindex.xml` file. If more than one node exists, you must check the port assignment for each node.

The following table is a list of port assignments that the node agent server uses by default. When you federate an application server node into a deployment manager cell, the deployment manager instantiates the node agent server process on the application server node. The node agent server uses these port assignments by default. When multiple processes share a port on the same node, the configuration uses the next number in the sequence for the additional processes. For example, if three processes are running, the BOOTSTRAP_ADDRESS port for each process is 2809, 2810, and 2811.

For a complete list of default port definitions, see [Port settings in WebSphere Application Server versions](#). For information on security ports, see [Configuring inbound transports](#).

Table 1. Default port definitions for the Version 6.1 node agent server process

Port Name	Description	Default Value (increments for multiple processes)
BOOTSTRAP_ADDRESS	The TCP/IP port on which the name service listens. This port is also the Remote Method Invocation (RMI) connector port. Specify this port with the administrative console or with the <code>chgwassvr</code> script. For more information, see The chgwassvr script .	2809
ORB_LISTENER_ADDRESS	The TCP/IP port on which the application server Object Request Broker (ORB) listens for requests. This also the port on which the location service daemon for the node listens. Specify this port with the administrative console or with the <code>chgwassvr</code> script. For more information, see The chgwassvr script .	9100
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS	The TCP/IP port on which the Secure Association Services (SAS) listen for inbound authentication requests. Specify this port with the administrative console or with the <code>chgwassvr</code> script. For more information, see The chgwassvr script .	9901
CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS	The TCP/IP port on which the Common Secure Interoperability Version 2 (CSIV2) Service listens for inbound client authentication requests. Specify this port with the administrative console or with the <code>chgwassvr</code> script. For more information, see The chgwassvr script .	9202

CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS	The TCP/IP port on which the Common Secure Interoperability Verison 2 (CSIV2) Service listens for inbound server authentication requests. Specify this port with the administrative console or with the chgwassvr script. For more information, see The chgwassvr script .	9201
NODE_DISCOVERY_ADDRESS	The TCP/IP port on which the node discovery service for the node agent listens. Specify this port with the administrative console or with the chgwassvr script. For more information, see The chgwassvr script .	7272
NODE_MULTICAST_DISCOVERY_ADDRESS	The TCP/IP port for the multicast discovery service on which the node agent listens. Specify this port with the administrative console or with the chgwassvr script. For more information, see The chgwassvr script .	5000
SOAP_CONNECTOR_ADDRESSES	This port is required by every WebSphere process to enable SOAP connectivity for JMX calls when using wsadmin.	8879
OVERLAY_UDP_LISTENER_ADDRESS	Used for peer-to-peer (P2P) communication. The ODC (On Demand Configuration) and asynchronous PMI components use P2P as their transport. This port is required by every WebSphere Extended Deployment process.	11001
OVERLAY_TCP_LISTENER_ADDRESS	Used for P2P communication. The ODC (On Demand Configuration) and asynchronous PMI components use P2P as their transport. This port is required by every WebSphere Extended Deployment process.	11002
XD_AGENT_PORT	The deployment manager, the node agents, and the middleware agents each have one XD_AGENT_PORT. Note that, unlike the OVERLAY ports, the application servers are not configured with XD_AGENT_PORTS. Used to enable communication between the deployment manager, the node agents, and the middleware agents. The ODR uses this port to	7061

	collect information from other servers, including node agents. This port should be available to all servers that the WebSphere Virtual Enterprise ODR is managing.	
DRS_CLIENT_ADDRESS	Deprecation: This port is deprecated and is no longer used in the current version of the product.	7873

During the **addNode** command operation, the **filetransfer** application uses port 9090 by default. The **filetransfer** application uses the same HTTP transport port that is used by the administrative console. If security is enabled, the default secured port 9043 must be opened in the firewall. If you modify the ports, or if you want to confirm the assigned port, check the port assignments in the `Install_Root\config\cells\cellname\nodes\nodename\servers\dmgr\server.xml` file.

Table 2. Default port definitions for the **fileTransfer** application

Port Name	Default Value
Default fileTransfer application Port	9090
Secured - Default fileTransfer application port	9043

When you federate an application server node with the embedded messaging server feature into a deployment manager cell, the deployment manager instantiates a Java™ Message Service (JMS) server process, `jmsserver`, on the application server node. The following table lists the port assignments that the JMS server uses by default:

Table 3. Default port definitions for the JMS server

Port Name	Default Value
JMSERVER_DIRECT_ADDRESS	5559
JMSERVER_QUEUED_ADDRESS	5558
SOAP_CONNECTOR_ADDRESS	8879
JMSERVER SECURITY PORT	5557

Parent topic: [Reference](#)

Related concepts:

[Middleware agent](#)
[Middleware nodes and servers](#)

Related tasks:

[Using the graphical user interface to augment profiles](#)

Scripts

You can use the WebSphere® Virtual Enterprise scripts after you install the product.

Scripts

The scripts are in the `install_root/bin` directory. In addition, you can use a new series of scripting objects. To access the new scripting objects, invoke the `$AdminConfig types` command in the `wsadmin` tool.

[arfmController.py script](#)

You can use a Python script to cause the autonomic request flow manager (ARFM) to forget all of its historical data.

[checkHmmLocation.jacl](#)

You can use the `checkHmmLocation.jacl` script to locate the health controller. Locating the health controller is useful when you are troubleshooting your configuration.

[checkPlacementLocation.jacl script](#)

You can use the `checkPlacementLocation.jacl` script to locate the application placement controller. Locating the application placement controller is useful when you are troubleshooting your configuration.

[createDynamicCluster.jacl script](#)

You can use the `createDynamicCluster.jacl` script to create a dynamic cluster.

[createodr.jacl script](#)

You can use the `createodr.jacl` script to create an on demand router (ODR) from the command line.

[coregroupsplit.py script](#)

You can use the `coregroupsplit.py` script to split your existing cell into multiple core groups. Consider running this script if you have more than 40 WebSphere Application Server related processes such as application servers, node agents, and on demand routers (ODRs) defined in your core group.

[DataReaderWrapper class](#)

You can use a command-line Java™ class to configure the runtime operations reports to log data for reuse in external programs.

[deleteDynamicCluster.jacl script](#)

You can use the `deleteDynamicCluster.jacl` script to delete dynamic clusters.

[deleteodr.jacl script](#)

You can use the `deleteodr.jacl` script to delete an on demand router (ODR) from the command line.

[dumpOdrState.jacl script](#)

You can use the `dumpOdrState.jacl` script to dump on demand router (ODR) states, and for problem diagnosis and debugging purposes. The detailed state of the queues and throttles of each gateway running in the ODR, as well as other global counters, are returned.

[dumpXdState.py script](#)

You can use the `dumpXdState.py` script to dump the current state of various WebSphere Virtual Enterprise processes. You can use this script for problem diagnosis and debugging purposes when requested by IBM® support. The detailed state of the on demand router (ODR), the deployment manager, and application server processes are displayed when using this script.

[Health controller commands with the AdminConfig object](#)

You can use the `AdminConfig` object to modify the health controller settings. You can change the controller cycle, prohibit server restarts at certain times of the day, and so on.

[HmmControllerProcs.jacl](#)

Using the `HmmControllerProcs.jacl` script, you can modify your health management runtime configuration to enable or disable the health controller and automated restarts for the server, and set or retrieve values for other health management settings.

[ifixtool.sh|.bat script](#)

You can use the `ifixtool.sh|.bat` script to verify that your product environment contains the recommended interim fixes that are necessary for WebSphere Virtual Enterprise to function correctly.

[importOverlayConfig.py script](#)

When setting up a star topology, you can use the `importOverlayConfig.py` script to configure the overlay communication between multiple cells.

[linkCells|linkCellsZOS script](#)

When you are setting up a star topology, you can use the `linkCells` script to configure the overlay communication between multiple cells. On z/OS® systems, use the `linkCellsZOS` script instead.

[ve_manageODC.py script](#)

The `ve_manageODC.py` script manages the ODC tree. The ODC tree is an in-memory representation of the state of a WebSphere Application Server cell.

[manageODR.py script](#)

You can use the **manageODR.py** script to manage custom logging and create a cluster of on demand routers (ODR).

[**manageWVEBB.py script**](#)

You can use the **manageWVEBB.py** script to gather the statistics usage of the BBSON bulletin board and periodically dump BBSON states. The statistics are available in the standard output and trace log files.

[**MirrorCell.bat|.sh script**](#)

You can use the **MirrorCell** script to create representations of servers that are running previous versions of WebSphere Application Server in the WebSphere Virtual Enterprise environment.

[**odrDebug.py script**](#)

You can use the **odrDebug.py** script to diagnose errors that an on demand router (ODR) returns.

[**PlacementControllerProcs.jacl script**](#)

You can use the **PlacementControllerProcs.jacl** script to modify the application placement runtime configuration.

[**pluginmerge.bat|.sh script**](#)

You can use the **pluginmerge .bat | .sh** script to combine the plugin-cfg.xml files from two, unbridged cells. Thus, the IBM HTTP server plug-in will either load-balance across the cells, or failover to a second (backup) cell.

[**serverQuiesce.py script**](#)

You can use the **serverQuiesce.py** script to decrease workload to a Session Initiation Protocol (SIP) container over an interval, and optionally stop the server. You can also cancel the quiesce during a set time interval if certain conditions are met.

[**servicepolicy.py script**](#)

You can use the **servicepolicy.py** script to perform operations on service policies from the command line, such as creating service policies, removing service policies, and editing transaction classes.

[**unlinkCells|unlinkCellsZOS script**](#)

You can use the script to disable the overlay communication between multiple cells.

[**useWVEBB.py**](#)

Run the **useWVEBB.py** script to enable the BBSON bulletin board and alleviate the dependency on the WebSphere Application Server Network Deployment high availability manager.

[**encodePassword.sh|bat script**](#)

You can use the **encodePassword.sh | bat** script to encode a password value that you enter in the administrative console. By encoding your password, you can prevent the password from being shared.

[**workclassoperations.py script**](#)

You can use the **workclassoperations.py** script to create, update, and manage work classes.

[**xd_APConfig.jacl script**](#)

Use the **xd_APConfig.jacl** script to make persistent changes to the application placement controller configuration.

Parent topic: [Reference](#)

arfmController.py script

You can use a Python script to cause the autonomic request flow manager (ARFM) to forget all of its historical data.

Purpose

Use the arfmController.py script to cause the ARFM to forget all of its historical data. This is useful if the application characteristics suddenly change.

Location

The arfmController.py script is in the install_root/bin directory.

Usage

To run the arfmController.py script from a command line, use the following command:

```
./wsadmin.sh -f arfmController.py -lang jython reset
```

Parent topic: [Scripts](#)

Related tasks:

[Creating ODRs](#)

[Configuring the autonomic request flow manager](#)

[Getting started with scripting](#)

Related reference:

[Administrative roles and privileges](#)

checkHmmLocation.jacl

You can use the checkHmmLocation.jacl script to locate the health controller. Locating the health controller is useful when you are troubleshooting your configuration.

Purpose

With the checkHmmLocation.jacl script, you can get the location of the health controller. The health controller can run in the node agent on any node in the cell that is not also running the deployment manager.

Location

The checkHmmLocation.jacl script is in the *install_root/bin* directory.

Usage

Before you run this script, verify that the deployment manager for the cell is running.

To run the checkHmmLocation.jacl script with the wsadmin tool, use the following command:

```
wsadmin -f checkHmmLocation.jacl
```

or

```
wsadmin.sh -f checkHmmLocation.jacl
```

Example

Sample output from running this script is as follows:

```
# WASX7209I: Connected to process "dmgr" on node eutil33Manager
using SOAP connector; The type of process is: DeploymentManager
looking for group name:
HAMItemGroup:com.ibm.ws.xd.hmm.controller.HmmControllerImpl_com.ibm.ws.xd.hmm.controller.HmmControll
er_Default

Results:
{SERVER_NAME nodeagent}
{MEMBER_NAME
HAMItemMember:com.ibm.ws.xd.hmm.controller.HmmControllerImpl_com.ibm.ws.xd.hmm.controller.HmmControll
er_Default_eutil33Network\df245\nodeagent_1}
{NODE_NAME df245}
{GROUP_ID
HAMItemGroup:com.ibm.ws.xd.hmm.controller.HmmControllerImpl_com.ibm.ws.xd.hmm.controller.HmmControll
er_Default}
```

The `NODE_NAME` df245 indicates the specific location information for the health controller. In this case, the controller is running on the df245 node.

Parent topic: [Scripts](#)

Related tasks:

[Configuring the health controller](#)
[Monitoring and tuning health management](#)

Related reference:

[Administrative roles and privileges](#)

checkPlacementLocation.jacl script

You can use the checkPlacementLocation.jacl script to locate the application placement controller. Locating the application placement controller is useful when you are troubleshooting your configuration.

Purpose

The checkPlacementLocation.jacl script returns the location information for the application placement controller. You can use this script for debugging the application placement feature. Run the script and then enable tracing to verify any problems that might occur.

Location

The checkPlacementLocation.jacl script is in the *install_root/bin* directory.

Usage

Before you run this script, verify that the deployment manager for the cell is running.

To run the checkPlacementLocation.jacl script with the wsadmin tool, use the following command:

```
wsadmin -f checkPlacementLocation.jacl
```

or

```
wsadmin.sh -f checkPlacementLocation.jacl
```

Example

Sample output from running this script is as follows:

```
# WASX7209I: Connected to process "dmgr" on node eutil33Manager
using SOAP connector; The type of process is: DeploymentManager
looking for group name: HAMItemGroup:com.ibm.ws.xd.placement.controller.impl.
PlacementControllerImpl_com.ibm.ws.xd.placement.controller.PlacementController_Default

Results:
{SERVER_NAME nodeagent}
{NODE_NAME xddesktop}
{MEMBER_NAME
HAMItemMember:com.ibm.ws.xd.placement.controller.impl.PlacementControllerImpl_com.ibm.ws.xd.
placement.controller.PlacementController_Default
{NODE_NAME df245}
{GROUP_ID HAMItemGroup:com.ibm.ws.xd.placement.controller.
PlacementControllerImpl_com.ibm.ws.xd.placement.controller.PlacementController_Default}
```

NODE_NAME df245 indicates the specific location information for the application placement controller. In this case, the controller is running on the df245 node.

Parent topic: [Scripts](#)

Related tasks:

[Configuring the health controller](#)

[Monitoring and tuning health management](#)

[Configuring health management](#)

CreateDynamicCluster.py script

You can use the `CreateDynamicCluster.py` script to create dynamic clusters.

The `CreateDynamicCluster.py` script creates a dynamic cluster, and associates it with a node group.

Location

The `CreateDynamicCluster.py` script is in the `install_root/bin` directory.

Syntax

To see a list of all available script options, use the following command:

```
wsadmin -lang jython -f CreateDynamicCluster.py
```

Table 1. . Script arguments

Argument	Description
<code>-dcPrefix</code>	Specifies the dynamic cluster prefix. Mandatory argument.
<code>-ngName</code> or <code>-createNodeGroup</code>	Specifies the node group name, or creates a node group with the specified name. Mandatory argument.

Example usage

```
wsadmin -lang jython -f CreateDynamicCluster.py -dcPrefix:dyncluster -  
ngName:DefaultNodeGroup
```

Related concepts:

[Dynamic clusters](#)

Related tasks:

[Creating dynamic clusters](#)

Related reference:

[Dynamic cluster administrative tasks](#)

[deleteDynamicCluster.jacl script](#)

[Wsadmin tool](#)

createDynamicCluster.jacl script

You can use the createDynamicCluster.jacl script to create a dynamic cluster.

Purpose

The createDynamicCluster.jacl script creates a dynamic cluster and associates it with a node group.

depfat: Although you can still use the Jacl scripts to create and delete dynamic clusters, the administrative tasks provide all of the options that are available for configuring dynamic clusters in Version 6.1 and later. See [Dynamic cluster administrative tasks](#) for more information.

Location

The createDynamicCluster.jacl script is in the *install_root/bin* directory.

Usage

To run the createDynamicCluster.jacl script with the wsadmin utility, use the following command:

```
wsadmin -profile createDynamicCluster.jacl node_group_name dynamic_cluster_name
```

In place of *node_group_name* and *dynamic_cluster_name*, use the name of the node group that you want to associate with your new dynamic cluster. You might need to modify the wsadmin command to wsadmin.sh or wsadmin.bat, depending on your operating environment.

The code in the runtime creates a membership policy when you create your dynamic cluster. For example, if you indicate the node_group_1 node group, the membership policy of node_nodegroup='node_group_1' is created.

Parent topic: [Scripts](#)

Related concepts:

[Dynamic clusters](#)

Related tasks:

[Creating dynamic clusters](#)

Related reference:

[Dynamic cluster administrative tasks](#)

[deleteDynamicCluster.jacl script](#)

[Wsadmin tool](#)

createodr.jacl script

You can use the createodr.jacl script to create an on demand router (ODR) from the command line.

Purpose

You can use the createodr.jacl script to create an ODR server.

To create an ODR, you must have administrator or configurator privileges.

Location

The createodr.jacl script is in the *install_root/bin* directory.

Usage

The default script usage follows:

```
./wsadmin.sh|bat -f createodr.jacl
```

Procedures

Specify the following parameters with the createodr.jacl script:

- *nodename*: The name of the node on which you want to create your ODR.
- *odr_template*: In Version 6.1, a template parameter is required. The following list contains the options for *odr_template*:
 - odr
 - sip_odr_server
 - http_sip_odr_server

z/OS On z/OS®, the only option for *odr_template* is odr_zos.

- *odr_name*: The name of the ODR you want to create.

Alternatively, you can first launch the scripting client. In wsadmin:

```
wsadmin> source createodr.jacl  
wsadmin> createodr nodename
```

In place of *nodename*, specify the name of the node on which you want to create your ODR.

Example

Create an ODR.

```
./wsadmin.sh|.bat -f createodr.jacl node01 odr
```

Create an ODR that is named great_ODR.

```
./wsadmin.sh|.bat -f createodr.jacl node01 http_sip_odr_server great_ODR
```

z/OS

```
./wsadmin.sh|.bat -f createodr.jacl node01 odr_zos
```

Parent topic: [Scripts](#)

Related tasks:

[Creating ODRs](#)

[Getting started with scripting](#)

Related reference:

[Administrative roles and privileges](#)

coregroupsplit.py script

You can use the `coregroupsplit.py` script to split your existing cell into multiple core groups. Consider running this script if you have more than 40 WebSphere® Application Server related processes such as application servers, node agents, and on demand routers (ODRs) defined in your core group.

Purpose

The `coregroupsplit.py` script divides your existing cell into multiple core groups. If a server changes its core group membership, then you must restart the entire cell to prevent partitions from forming. For this reason, this default script options for script do not change the core group membership of servers that are members of any core groups other than the DefaultCoreGroup core group.

Running the script attempts to satisfy the following best practices for core groups:

- Each core group must have fewer than 40 servers. This best practice is possible only when you have an adequate ratio of nodes to servers in your cell.
- Each core group must be equipped with at least three core group bridges. This best practice is possible only when you have an adequate ratio of nodes to servers in your cell.
- Each cluster must remain fully mapped to a single core group.
- If you use the `-createbridges` parameter, the core group bridges and coordinators are preferably hosted on otherwise idle nodes.

This script also tunes the high availability manager for optimum performance. By default, the script configures a core group bridge on each node agent in the cell. The script increases the number of core groups until it reaches the optimal level, depending on the number of nodes and servers in the cell. The core group bridge node agents are configured as a part of the DefaultAccessPointGroup access point group in a mesh topology. In the preferred mesh topology, all of the access points are collected into a single access point group. As a result, all bridge interfaces can directly intercommunicate.

Remember that you must give core group bridges at least 512 MB of Java™ virtual machine (JVM) space.

Location

The `coregroupsplit.py` script is in the `install_root/bin` directory.

Usage

The default script usage follows:

```
./wsadmin.sh -lang jython -f coregroupsplit.py
```

Running this script might result in unbalanced core groups, in which some core groups are loaded more or less than other core groups. You can rerun this script to rebalance core group membership, but in this case you must restart the entire cell for the changes to take effect. To rerun the script, use the following command:

```
./wsadmin.sh -lang jython -f coregroupsplit.py -reconfig
```

Parameters

-reconfig

Performs a full reconfiguration to rebalance the distribution of servers among the core groups.

-linked

Creates a ring topology of core group bridges.

-createbridges

Creates separate core group bridge processes instead of creating the bridge in the node agent.

-numcoregroups

Specifies the number of core groups to create.

-datastacksize

Specifies a number of megabytes that overrides the default data stack size.

-proxycoregroup

Places the on demand routers (ODR) and proxy servers in a separate core group.

-odrcoregroup

Places the on demand routers (ODR) and proxy servers in a separate core group.

-nosave

Does not save any changes made to the core group. You can use this option to test setting parameters and running the script.

-debug

Prints troubleshooting information.

-nodesPerCG: number

Specifies the number of node agents that are required for each core group.

-numberOfServersPerCG: *number*

Specifies the maximum number of servers for each core group.

-bridgeHeapSize: *number*

Specifies the server heap size of the core group bridge in megabytes.

Examples

The following example results in a linked topology where the core group bridges are connected in a ring:

```
./wsadmin.sh -lang jython coregroupsplit.py -linked
```

You can also use this script to create static cluster that are dedicated as core group bridges for communication within the core group. Use the following example:

```
./wsadmin.sh -lang jython coregroupsplit.py -createbridges
```

Parent topic: [Scripts](#)

Related concepts:

[Dynamic clusters](#)

Related tasks:

[Creating dynamic clusters](#)

[Configuring communication between core groups that are in the same cell](#)

[Configuring WebSphere Virtual Enterprise for cross-cell communication](#)

DataReaderWrapper class

You can use a command-line Java™ class to configure the runtime operations reports to log data for reuse in external programs.

Purpose

You can read and filter the data logs into your own files by invoking this Java class from a command line.

Location

The DataReaderWrapper class is in the `WAS_HOME/appserver/lib/visualization.cacheservice.jar` file.

Usage

To run the DataReaderWrapper class from a command line, use the following command:

```
java DataReaderWrapper
```

Parameters

The following parameters are available:

- *log_type*

Type of log. You can use the *log_type* parameter alone to retrieve a help list of properties and their descriptions for the specified log. These properties can be used for the *property_filter* parameter.

Example: `java DataReaderWrapper ServerStatsCache`

- *log_directory*

Directory of the log file.

Example: `java DataReaderWrapper ServerStatsCache log`

- *output_file_name*

Specifies the name of the output file.

Example: `java DataReaderWrapper ServerStatsCache log data.log`

- *interval_start_time*

The start time of the log interval in long or string form.

Examples:

- Long form: `java DataReaderWrapper ServerStatsCache log data.log 1113189469300`
- String form (yy.MM.dd_HH.mm.ss): `java DataReaderWrapper ServerStatsCache log data.log 05.05.18_12.02.45`

- *interval_end_time*

The end time of the log interval in long or string form.

Examples:

- Long form: `java DataReaderWrapper ServerStatsCache log data.log 1113189469300 1113189472500`
- String form (yy.MM.dd_HH.mm.ss): `java DataReaderWrapper ServerStatsCache log data.log 05.05.18_12.02.45 05.05.18_01.02.00`

- *property_filter*

Property value pairs.

Example: `java DataReaderWrapper ServerStatsCache log data.log 1113189469300 1113189472500 name server1 node dabtcNode01`

- ?

Displays syntax help.

Example: `java DataReaderWrapper ?`

In the following example, the DataReaderWrapper reads the ServerStatsCache log files in the log directory from the time 1113189469300 to 1113189472500 on the server1 machine and the dabtcNode01 node:

```
java DataReaderWrapper ServerStatsCache log data.out 1113189469300 1113189472500 name server1 node dabtcNode01
```

Parent topic: [Scripts](#)

Related concepts:

[Runtime operations overview](#)

Related tasks:

[Creating and managing reports](#)

deleteDynamicCluster.jacl script

You can use the deleteDynamicCluster.jacl script to delete dynamic clusters.

Purpose

The deleteDynamicCluster.jacl script deletes a dynamic cluster.

depfeat: Although you can still use the Jacl scripts to create and delete dynamic clusters, the administrative tasks provide all of the options that are available for configuring dynamic clusters in Version 6.1 and later. See [Dynamic cluster administrative tasks](#) for more information.

Location

The deleteDynamicCluster.jacl script is in the *install_root/bin* directory.

Usage

To run the deleteDynamicCluster.jacl script with the wsadmin utility, use the following command:

```
wsadmin -f deleteDynamicCluster.jacl node_group_name dynamic_cluster_name
```

In place of *node_group_name* and *dynamic_cluster_name*, use the name of the node group that you want to disassociate with your deleted dynamic cluster. You might need to modify the wsadmin command to **wsadmin.sh** or **wsadmin.bat**, depending upon your operating environment.

Parent topic: [Scripts](#)

Related concepts:

[Dynamic clusters](#)

Related tasks:

[Creating dynamic clusters](#)

Related reference:

[Dynamic cluster administrative tasks](#)
[createDynamicCluster.jacl script](#)

[Wsadmin tool](#)

deleteodr.jacl script

You can use the deleteodr.jacl script to delete an on demand router (ODR) from the command line.

Purpose

You can use the deleteodr.jacl script to delete an ODR.

To delete an ODR, you must have administrator or configurator privileges.

Location

The deleteodr.jacl script is located in the *install_root/bin* directory.

Usage

The default script usage follows:

```
./wsadmin.sh|.bat -f deleteodr.jacl
```

Procedure

Specify the following parameter with the deleteodr.jacl script:

nodename

The name of the node on which your ODR exists.

Alternatively, you can first launch the scripting client. In wsadmin:

```
wsadmin> source deleteodr.jacl  
wsadmin> deleteodr nodename
```

In place of *nodename*, specify the name of the node on which your ODR exists.

Example

Delete an ODR.

```
./wsadmin.sh|.bat -f deleteodr.jacl node01
```

Parent topic: [Scripts](#)

Related tasks:

[Creating ODRs](#)

[Getting started with scripting](#)

Related reference:

[Administrative roles and privileges](#)

dumpOdrState.jacl script

You can use the dumpOdrState.jacl script to dump on demand router (ODR) states, and for problem diagnosis and debugging purposes. The detailed state of the queues and throttles of each gateway running in the ODR, as well as other global counters, are returned.

Purpose

You can use the dumpOdrState.jacl script to dump ODR states.

Location

The dumpOdrState.jacl script is in the *install_root/bin* directory.

Usage

The default script usage follows:

```
./wsadmin.sh|bat -profile dumpOdrState.jacl
```

Procedures

Specify the following parameter with the dumpOdrState.jacl script:

ODR_name

Specifies the name of the ODR.

Example

```
./wsadmin.sh|.bat -profile dumpOdrState.jacl myODR
```

Results

The dumpOdrState.jacl script dumps the ODR state of the ODR that you specified by invoking the WsmmProxyMBean managed bean (MBean).

This script returns one string that contains a description of the state of every Autonomic Request Flow Manager (ARFM) gateway in the ODR. The description lists the queued and running requests, and the throttling state of each queue. The description also contains debug-level information.

Parent topic: [Scripts](#)

Related tasks:

[Creating ODRs](#)

Related reference:

[Administrative roles and privileges](#)

dumpXdState.py script

You can use the dumpXdState.py script to dump the current state of various WebSphere® Virtual Enterprise processes. You can use this script for problem diagnosis and debugging purposes when requested by IBM® support. The detailed state of the on demand router (ODR), the deployment manager, and application server processes are displayed when using this script.

Purpose

You can dump the state of active WebSphere Virtual Enterprise processes with this script.

Location

The dumpXdState.py script is in the install_root\bin directory.

Usage

The dumpXdState.py script must be run from the install_root\bin directory using wsadmin.sh. To automatically detect active WebSphere Virtual Enterprise processes and write their state information to a file, use the following command:

```
wsadmin.sh -lang jython -f dumpXdState.py --auto > output_file_name
```

Where *output_file_name* is the name of a file to save the output.

Windows Use wsadmin.bat for the Windows operating systems.

Options

--auto

Exports information for all of the WebSphere Virtual Enterprise processes into the specified text file. This option exports a large amount of data, for example, a small topology might result in a text file that is over 2500 kilobytes.

--dom

Specifies a domain. The default is *, or all domains.

--dmgr=**deployment_manager_name**

Exports information for the deployment manager that you specify in the option.

--help

Displays the help for this script.

--odr=**odr_name**

Exports information for the ODR that you specify in the option. The output for this option is similar to output that you get from the dumpOdrState.jacl script, but includes additional information from internal processes.

--svr=**server_name**

Exports information for the server that you specify in the option.

The following example automatically detects WebSphere Virtual Enterprise processes and writes their state information to a file1:

```
wsadmin.sh -lang jython -f dumpXdState.py --auto > file1
```

Results

The dumpXdState.py script dumps the state of various WebSphere Virtual Enterprise processes by invoking the managed beans (MBean) of those processes. The dumped information is written to the output file name specified on the command line. Provide this file to IBM support for diagnosis and debugging of problems.

Parent topic: [Scripts](#)

Related tasks:

[Creating ODRs](#)

[Configuring the autonomic request flow manager](#)

[Getting started with scripting](#)

Related reference:

[Administrative roles and privileges](#)

[dumpOdrState.jacl script](#)

Health controller commands with the AdminConfig object

You can use the AdminConfig object to modify the health controller settings. You can change the controller cycle, prohibit server restarts at certain times of the day, and so on.

Purpose

Health management comes equipped with smart defaults that accommodate most environments. However, if you discover that your health controller is not working the way you want, then tune the default parameters. You can change these settings in the administrative console or with the wsadmin tool.

Usage

To change configuration settings for the health controller, you must have configurator or administrator privileges. If you have operator privileges, you can only change the runtime configuration.

To edit the health controller properties, run the following commands:

```
.\wsadmin.sh -lang jython
hcid = AdminConfig.getid("/HealthController:/")
AdminConfig.modify(hcid, [{"attribute_name", value}])
AdminConfig.save()
```

Attributes

You can edit the following attributes:

controlCycleLength

Specifies the time between consecutive health checks to determine if a health policy condition is breached.

Default: 5

enable

Specifies if health monitoring is enabled.

Default: true

maxConsecutiveRestarts

Specifies the number of attempts to revive a server after a restart decision is made. If this number is exceeded, a failed operation is assumed and restarts are disabled for the server.

Valid values: whole numbers between 1 and 5

Default: 3

minRestartInterval

Controls the minimum amount of time that must pass between consecutive restarts of an server instance.

Valid values: The value can range from 15 minutes to 365 days, inclusive. Indicate the units with the **minRestartIntervalUnits** attribute. A value of 0 disables the minimum restart value.

Default: 0 (disabled)

minRestartIntervalUnits

Indicates the units to use with the **minRestartInterval** attribute

Valid values: 2 (minutes), 3 (hours), or 4 (days)

Default: 2

prohibitedRestartTimes

Specifies the times and days of the week during which a restart of an application server instance is prohibited. You cannot change this attribute with the AdminConfig object. To update the prohibited restart times in the administrative console, click **Operational policies > Autonomic managers > Health controller**. Edit the **Prohibited restart times** field.

properties

Specifies a custom property on the health controller.

restartTimeout

Specifies a number of minutes to wait for a server to stop before explicitly checking its state and attempting another start.

Valid values 1 to 60 minutes, specified as a whole number

Default: 5

Example

The following command example changes the restart timeout setting:

```
hcid = AdminConfig.getid("/HealthController:/")
AdminConfig.modify(hcid, [{"restartTimeout", 6}])
AdminConfig.save()
```

The following command sets the `com.ibm.ws.xd.hmm.controller.approvalTimeOutMinutes` custom property to 40 minutes:

```
.\wsadmin.sh -lang jython
hcid = AdminConfig.getid("/HealthController:/")
AdminConfig.create('Property', hcid, [['name',
'com.ibm.ws.xd.hmm.controller.approvalTimeOutMinutes'], ['value', '40']])
AdminConfig.save()
```

Parent topic: [Scripts](#)

Related concepts:
[Health management](#)

Related tasks:
[Monitoring and tuning health management](#)
[Configuring health management](#)

Related reference:
[HmmControllerProcs.jacl](#)

Related information:
[Health controller custom properties](#)

HmmControllerProcs.jacl

Using the HmmControllerProcs.jacl script, you can modify your health management runtime configuration to enable or disable the health controller and automated restarts for the server, and set or retrieve values for other health management settings.

Purpose

To change runtime settings, you must have operator or administrator administrative privileges.

You can use the HmmControllerProcs.jacl file to complete the following actions.

- Enable or disable the health controller.
- Enable or disable automated restarts for the server.
- Set or retrieve values for other health management settings.

The HmmControllerProcs.jacl script only modifies the runtime configuration.

Location

The HmmControllerProcs.jacl script is in the *install_root/bin* directory.

Usage

Run the following command.

```
wsadmin -profile HmmControllerProcs.jacl -c "insert_procedure_parameters"
```

Replace the *insert_procedure_parameters* variable with the name of the procedure and the proper variable values to complete your changes. You might have to modify the wsadmin command to *wsadmin.sh* or *wsadmin.bat*, depending on your operating environment.

Procedures

To see a list of all available procedures, use the following command.

```
wsadmin -profile HmmControllerProcs.jacl -c "help"
```

enable

Enables the health controller.

disable

Disables the health controller.

isEnabled

Verifies that the health controller is enabled.

enableServer *node_name server_name*

Enables automated restarts for the server on the specified node.

disableServer *node_name server_name*

Disables automated restarts for the server on the specified node.

isServerEnabled *node_name server_name*

Verifies if automated restarts are enabled for the server on the specified node.

isNodeMaintenance *node_name*

Verifies if the health controller acknowledges that the specified node is in maintenance mode.

getMaxConsecutiveRestarts

Returns the number of attempts to revive an application server after a restart is performed. If the maximum value is exceeded, the server is declared failed and server restarts are disabled.

setMaxConsecutiveRestarts *number_of_consecutive_restarts*

Sets the number of attempts to revive an application server after a restart is performed.

getControlCycleLength

Returns the length of time, in minutes, between health policy checks on the application server instances to determine if breaches occurred.

setControlCycleLength *time_in_minutes*

Sets the length of time, in minutes, between health policy checks on the application server instances to determine if breaches occurred.

getMinRestartInterval

Returns the length of time that must pass between application server instance restarts.

setMinRestartInterval *time_in_minutes*

Sets the length of time that must pass between application server instance restarts.

getRestartTimeout

Returns the length of time that the controller uses to wait for start and stop events during a restart before polling the server status.

setRestartTimeout *time_in_minutes*

Sets the length of time that the controller uses to wait for start and stop events during a restart before polling the server status.

Example

The following command enables automatic restarts on the server_1 server, which runs on the node_1 node:

```
wsadmin.sh -profile HmmControllerProcs.jacl -c "enableServer node_1 server_1"
```

Parent topic: [Scripts](#)

Related concepts:

[Health management](#)

Related tasks:

[Configuring health management](#)

[Enabling and disabling health management](#)

[Monitoring and tuning health management](#)

Related reference:

[Administrative roles and privileges](#)

ifixtool.sh|.bat script

You can use the ifixtool.sh|.bat script to verify that your product environment contains the recommended interim fixes that are necessary for WebSphere® Virtual Enterprise to function correctly.

Purpose

The ifixtool.sh|.bat script checks if any recommended interim fixes for Virtual Enterprise are not installed in your cell. Because Virtual Enterprise runs in an existing WebSphere Application Server Network Deployment environment, the ifixtool.sh|.bat script checks for recommended interim fixes for both products.

Location

The ifixtool.sh|.bat script is in the install_root/bin directory.

Usage

You must run the ifixtool.sh|.bat script on the deployment manager.

```
ifixtool.sh|.bat [-downloadAndCheckCell] [-checkCell ND path_to_ND_zip_file]
[-checkCell VE path_to_VE_zip_file]
[-download ND product_version] [-download VE product_version]
```

Parameters

downloadAndCheckCell

Downloads the interim fixes that are not yet applied to the cell. Internet access on the deployment manager is required.

checkCell ND path_to_ND_zip_file

Creates a ZIP file containing only the WebSphere Application Server Network Deployment interim fixes that are needed for the cell.

checkCell VE path_to_VE_zip_file

Creates a ZIP file containing only the Virtual Enterprise interim fixes that are needed for the cell.

download ND product_version

Downloads a ZIP file containing WebSphere Application Server Network Deployment interim fixes to the current directory.

download VE product_version

Downloads a ZIP file containing Virtual Enterprise interim fixes to the current directory.

The following example creates the new configurations only and disables the persistence of changes:

```
./ifixtool.sh -downloadAndCheckCell
```

Parent topic: [Scripts](#)

Related tasks:

[Using the interim fix tool to check for maintenance packages](#)

[Using the Update Installer to install maintenance packages](#)

importOverlayConfig.py script

When setting up a star topology, you can use the importOverlayConfig.py script to configure the overlay communication between multiple cells.

Purpose

Use the importOverlayConfig.py script to enable communication between WebSphere® Virtual Enterprise cells. Each cell contains a overlaynodes.config file, and the overlaynodes.config files must be copied between the cells to enable multi-cell performance management.

Location

The importOverlayConfig.py script is in the install_root/bin directory.

The overlaynodes.config file is in the WAS_HOME/profiles/deployment_manager_profile_name/config/cells/cell_name directory of each cell.

Usage

See the following examples to configure the overlay communication between multiple cells. Consider a scenario in which there are three cells: CellA is the center cell, and CellB and CellC are the point cells.

Run the importOverlayConfig.py script from the deployment manager on CellA to link the overlay anchor transports of CellA to CellB.

```
WAS_HOME/bin/wsadmin.sh -profileName profile_name -f importOverlayConfig.py  
link path_to_CellB_overlaynodes.config_file
```

Run the importOverlayConfig.py script from the deployment manager on CellA to link the overlay anchor transports of CellA to CellC.

```
WAS_HOME/bin/wsadmin.sh -profileName profile_name -f importOverlayConfig.py  
link path_to_CellC_overlaynodes.config_file
```

Run the importOverlayConfig.py script from the deployment manager on CellB to link the overlay anchor transports of CellB to CellA.

```
WAS_HOME/bin/wsadmin.sh -profileName profile_name -f importOverlayConfig.py  
link path_to_CellA_overlaynodes.config_file
```

Run the importOverlayConfig.py script from the deployment manager on CellC to link the overlay anchor transports of CellC to CellA.

```
WAS_HOME/bin/wsadmin.sh -profileName profile_name -f importOverlayConfig.py  
link path_to_CellA_overlaynodes.config_file
```

See the following example to unlink two cells.

```
WAS_HOME/bin/wsadmin.sh -profileName profile_name -f importOverlayConfig.py  
unlink path_to_target_cell_overlaynodes.config_file
```

Parent topic: [Scripts](#)

Related tasks:

[Configuring multi-cell performance management](#)

linkCells|linkCellsZOS script

When you are setting up a star topology, you can use the **linkCells** script to configure the overlay communication between multiple cells. On z/OS® systems, use the **linkCellsZOS** script instead.

Purpose

Use the **linkCells** script to enable communication between a WebSphere® Virtual Enterprise cell containing servers that are enabled with an on demand router (ODR) that routes work requests to other administrative cells.

In version 7.0.0.3 and later, on z/OS systems, use the **linkCellsZOS** script.

Location

Distributed platforms The **linkCells** script is available in the app_server_root/bin directory.

z/OS The **linkCellsZOS** script is available in the app_server_root/bin directory.

Usage

Distributed platforms Run the **linkCells** script from the center cell to link the center cell with a point cell:

```
./linkCells.sh centerHost:center_cell_soap_port:user_id:password  
pointHost:point_cell_soap_port:user_id:password
```

z/OS Run the **linkCellsZOS** script from the center cell to link the center cell with a point cell:

```
./linkCellsZOS.sh centerHost:center_cell_soap_port:user_id:password  
pointHost:point_cell_soap_port:user_id:password
```

Example

Consider a scenario in which there are two cells, center and point1, with security enabled in both. For the center cell, the host name of the deployment manager is centerHost, the SOAP port is 8879, the user name is centerUID, and the password is centerPWD. For the point cell, the host name of the deployment manager is point1Host, the SOAP port is 8880, the user name is point1UID, and the password is point1PWD. The following example illustrates how to link the center and point1 cells together as is needed to support a star topology.

```
./linkCells.sh centerHost:8879:centerUID:centerPWD  
point1Host:8880:point1UID:point1PWD
```

Parent topic: [Scripts](#)

Related tasks:

[Configuring multi-cell performance management](#)

ve_manageODC.py script

The `ve_manageODC.py` script manages the ODC tree. The ODC tree is an in-memory representation of the state of a WebSphere® Application Server cell.

Purpose

The `ve_manageODC.py` script can add and remove nodes and edges, or modify the value of properties on a node. You can also use the script when troubleshooting routing policy errors for the on demand router (ODR).

Location

The `ve_manageODC.py` script is located in the `install_root/bin` directory.

Usage

The script usage for general help follows:

```
./wsadmin.sh|bat -lang jython -f ve_manageODC.py
```

The script usage for operation-specific help follows:

```
./wsadmin.sh|bat -lang jython -f ve_manageODC.py operation --help
```

Generate a `target.xml` file to determine the ODC names to plug into the script.

Operations

You can perform the following operations with the `ve_manageODC.py` script:

- **removeODCNode:** Removes a node.
 - `odcNodePath:` Specifies the full ODC tree path of the node to remove.
 - `nodeName:` Specifies the name of the WebSphere node containing the server that initiates the removal.
 - `serverName:` Specifies the name of the server to initiate the removal.
- **addODCNode:** Creates a new node.
 - `odcParentNodePath:` Specifies the full ODC tree path for the parent of the new node to be created.
 - `odcNodeType:` Specifies the ODC node type of the new node to be created.
 - `newNodeName:` Specifies the name of the new node to be created.
 - `nodeName:` Specifies the name of the WebSphere node containing the server that initiates the addition.
 - `serverName:` Specifies the name of the server to initiate the addition.
 - `[-p odcPropertyDescriptor priority::value]:` Specifies the name of the ODC property to be modified on the new node priority. Value is the priority and value to set the ODC property to on the new node. Priority can be omitted if the default value is used.
 - `[-l linkOdcNodePath]:` Specifies the full ODC tree path of the node for which an edge is to be created.
- **removeODCEdge:** Removes the link between two nodes.
 - `odcNodePathA odcNodePathB:` Specifies the full ODC tree paths of the nodes to be unlinked.
 - `nodeName:` Specifies the name of the WebSphere node containing the server that initiates the removal.
 - `serverName:` Specifies the name of the server to initiate the removal.
- **addODCEdge:** Links one node to another node.
 - `odcNodePathA odcNodePathB:` Specifies the full ODC tree paths of the nodes to be linked.
 - `nodeName:` Specifies the name of the WebSphere node containing the server that initiates the addition.
 - `serverName:` Specifies the name of the server to initiate the addition.
- **modifyODCProperty:** Modifies a specified ODC property of a node.
 - `odcNodePath:` Specifies the full ODC tree path of the node whose property is to be modified.
 - `odcPropertyDescriptor:` Specifies the name of the ODC property to be modified.
 - `priority::value:` Specifies the priority and value to set the ODC property to. Priority can be omitted if the default value is used.
 - `nodeName:` Specifies the name of the WebSphere node containing the server that initiates the modification.
 - `serverName:` Specifies the name of the server to initiate the modification.

Example

Generate a `target.xml` file to determine the ODC names to include in the script. The following code example shows a shortened version of a `target.xml` file, where parameters for the cell, node, and server that you want to use in the script are located.

```
<cellGroup name="target">
  <!-- cell section -->
  <cell name="Cell1">
    <!-- node section -->
```

```
<node name="metis07">
  <!-- server section -->
  <server name="odr">
    <property name="state" priority="1" value="STOPPED" />
  </server>
</node>
```

To delete the server named odr from the ODC tree, type the following command:

```
./wsadmin.sh -lang jython -f manageODC.py removeODCNode /cell/Cell1/node/memis07/server/odr <myNode>
<myServer>
```

Note that /cellGroup/target is never specified as part of the path.

In this example, the property ODC object is named state, with a value of STOPPED, and a priority of 1. To change the property to STARTED, use the following command:

```
./wsadmin.sh -lang jython -f manageODC.py modifyODCProperty /cell/Cell1/node/memis07/server/odr
state 1::STARTED <myNode> <myserver>
```

Parent topic: [Scripts](#)

Parent topic: [Scripts](#)

Related tasks:

[Creating and configuring ODRs](#)

[Configuring ODRs](#)

manageODR.py script

You can use the **manageODR.py** script to manage custom logging and create a cluster of on demand routers (ODR).

Location

The **manageODR.py** script is located in the `install_root/bin` directory. Before running this script, ensure that you have the environment variable `WAS_HOME` configured to point to the directory of your WebSphere® installation.

Operations

You can perform the following operations with the **manageODR.py** script:

- **convertToCluster** `<node:odr> <cluster>`: Convert an existing ODR on node `node` with name `odr` to an ODR cluster named `<cluster>`
- **addToCluster** `<node:odr> <cluster>`: Create a new ODR on node `node` with name `odr` and add it to ODR cluster `cluster`.
- **insertCustomLogRule** `<node:odr> <cluster><odrServerOrCluster> <ruleNumber> <condition> <logFileFormat>`: Insert a custom log rule into the list at position `<ruleNumber>`.
- **RemoveCustomLogRule** `<odrServerOrCluster> <ruleNumber>`: Remove the custom log rule number `<ruleNumber>`.
- **ListCustomLogRules** `<odrServerOrCluster>`: List the custom log rules associated with an ODR or ODR cluster.
- **insertCustomLogRule** `<node:odr> <cluster><odrServerOrCluster> <ruleNumber> <condition> <logFileFormat>`: Insert a custom log rule into the list at position `<ruleNumber>`. If `<ruleNumber>` is larger than the number of rules in the list, the new rule is appended to the list with the next available rule number.

where:

Table 1. Arguments

Argument	Description
<code><node:odr></code>	Name of the node and ODR. For example, <code>mynode:myodr</code> .
<code><cluster></code>	Name of an ODR cluster.
<code><odrServerOrCluster></code>	Is either <code><node:odr></code> or <code><cluster></code>
<code><ruleNumber></code>	Number of the rule, from 1 to the total number of rules (or larger for <code>insertCustomLogRule</code>).
<code><condition></code>	Condition (that is, boolean expression) which must evaluate to true in order to trigger the associated custom logging.
<code><logFileFormat></code>	Specification denoting the file name and format of the log entry. See the <i>log action format</i> section in the Custom logs overview topic.

The following example shows how to create an HTTP ODR named `odr`, on `node 1`:

```
wsadmin.sh -f createodr.jacl node1 odr odr
```

The following example shows how to convert an ODR named `odr` on node `node1` to an ODR cluster named `ODRCluster`:

```
wsadmin.sh -f manageODR.py -lang jython convertToCluster node1:odr ODRCluster
```

The following example shows how to add a custom log rule to put all requests whose service time is longer than 2 seconds in the `slow.log` custom log, and include the application server to which the request was sent and the service time:

```
wsadmin.sh -f manageODR.py -lang jython insertCustomLogRule myNode02:odr1 1 "service.time > 2000" "slow.log %t %r %Z %T"
```

The following example shows how to add a custom log rule to put all 503 responses in `503.log`. The rule is added at position 2. If there is an existing rule at position 2, the new rule is inserted before the existing rule.

```
wsadmin.sh -f manageODR.py -lang jython insertCustomLogRule myNode02:odr1 2 "response.code = 503" "503.log %t %r %s"
```

The following example shows how to add a custom log rule at position 1 to log errors that can occur when writing a response to a client in the response.write.error.log:

```
wsadmin.sh -f manageODR.py -lang jython insertCustomLogRule node1:odr 1  
"response.write.error" "response.write.error.log %t %r %s"
```

Parent topic: [Scripts](#)

Related concepts:

[Custom logs overview](#)

[Cell affinity function](#)

Related tasks:

[Configuring custom logs](#)

[Creating and configuring ODRs](#)

[Configuring ODRs](#)

[Enabling cell affinity](#)

[Configuring cell affinity in a multi-tiered environment](#)

[Defining generic server clusters for remote ODR cells](#)

manageWVEBB.py script

You can use the manageWVEBB.py script to gather the statistics usage of the BBSO bulletin board and periodically dump BBSO states. The statistics are available in the standard output and trace log files.

Purpose

For problem diagnosis and debugging purposes, use the manageWVEBB.py script to collect BBSO state dumps from every process in the product cell.

Location

The manageWVEBB.py script is in the *install_root/bin* directory.

Usage

Run the following command on each WebSphere® Virtual Enterprise cell:

```
wsadmin -f manageWVEBB.py operation [nodeName serverName]
```

You must specify the *operation* variable, which can be one of the following choices:

- *dump*: Dumps the state of the bulletin board
- *getsnapshot*: Obtains statistics about the current state of the bulletin board
- *getStatisticsPrintInterval*: Shows the current interval between printing bulletin board statistics in seconds
- *setStatisticsPrintInterval*: Sets the interval between printing bulletin board statistics in seconds

If you do not specify the *[nodeName serverName]* argument, the MBean is invoked against every process in the cell. Provide a node name and server name to narrow the invocation to the specific node and server that are listed.

Counter definitions

Name	Description
subjCreate	The number of created Subjects
subjForget	The number of closed Subjects
subscrCreate	The number of created SubjectSubscriptions
subscrClose	The number of closed SubjectSubscriptions
postCreate	The number of created SubjectPosts
postClose	The number of closed SubjectPosts
localUpdate	The number of updates that are performed by local posters
remoteUpdate	The number of updates that are performed by remote posters
localPostSizeSum	The data length in bytes that is sent by the local update calls
remotePostSizeSum	The data length in bytes that is sent by the remote update calls
localCallbackCall	The number of the update callback calls
localCallbackNumPostersSum	The number of posters that data is passed to in the local update callbacks
localCallbackPostSizeSum	The size in bytes of the data payloads that are passed in the update callbacks
localCallbackTimeSum	The time that is spent during the local update callbacks calls
localSubjPostCallback	The number of callbacks that are registered through SubjectPost#registerCallback
outOpenedTCP	The number of outgoing TCP connections that are open
outClosedTCP	The number of outgoing TCP connections that are closed
inOpenedTCP	The number of incoming TCP connections that are open
inClosedTCP	The number of incoming TCP connections that are closed
nothingToSend	The number of times an outgoing connection is opened with nothing to send at the moment the connection is completed

Parent topic: [Scripts](#)

Related concepts:

[BBSON bulletin board](#)

[Core groups \(high availability domains\)](#)

Related tasks:

[Creating a new core group](#)

Related reference:

[useWEBB.py](#)

MirrorCell.bat|.sh script

You can use the MirrorCell script to create representations of servers that are running previous versions of WebSphere® Application Server in the WebSphere Virtual Enterprise environment.

Purpose

The MirrorCell script creates a representation in the WebSphere Virtual Enterprise administrative console of WebSphere Application Server application servers from previous versions, such as Version 5.1 or Version 6. Before you run this script you must generate the mappings file. See [Migrating large topologies of external WebSphere application servers](#) for more information.

Location

The MirrorCell script is in the install_root/bin directory.

Usage

```
MirrorCell.bat|.sh [-props PROPS] [-mode MODE] [-nosec, -nosave]
```

Parameters

PROPS

Specifies the full path to the properties file.

MODE

The default mode is **ALL** if you do not specify the value or specify a value that is not valid. You can specify the following modes:

- **ALL**: Runs through every phase of the process.
- **READ**: Reads your cell configuration into an XML definition file only.
- **DIFF**: Differentiates your previously known configuration from the new configuration to detect changes or unnecessary configurations.
- **WRITE**: Creates the new configurations from the XML definitions file.
- **PROPSGEN**: Generates the properties file only. One useful application of the PROPSGEN mode is to run this mode as a cell administrator, set the user name and password while in the default encryption mode, and then give the properties file to a non-administrator to use when running the scripts. This action ensures that only the initial administrator that generates the file knows the values for the user name and password.

Options

-nosec

Specifies that your properties file does not encrypt the user name and password; however you must then use this option every time you run the scripts. Otherwise, the logic assumes that you are using encrypted data.

-nosave

Specifies that the script preforms trial run and does not persist any of the changes that were made. To ensure that the script runs correctly, use this option when you first run the script.

The following example creates the new configurations only and disables the persistence of changes:

```
./MirrorCell.sh -mode WRITE -nosave
```

Parent topic: [Scripts](#)

Related concepts:

[Migration toolkit](#)

Related tasks:

[Migrating large topologies of external WebSphere application servers](#)
[Configuring external WebSphere application servers](#)

odrDebug.py script

You can use the `odrDebug.py` script to diagnose errors that an on demand router (ODR) returns.

Purpose

You can use the `odrDebug.py` script to diagnose errors that an ODR returns, such as 404 errors and 503 errors.

Location

The `odrDebug.py` script is located in the `install_root/bin` directory.

Usage

The default script usage follows:

```
./wsadmin.sh|.bat -lang jython -f odrDebug.py
```

Operations

Specify the following parameters with the `odrDebug.py` script:

setHttpDebug:

- `nodeName`: Specifies the name of the node that you want to debug.
- `odrName`: Specifies the name of the ODR that you want to debug.
- `errorCode`: Specifies the HTTP error code. You can specify 404, 503, and so on.
- `expression`: Specifies if the error code is being debugged. Specify `false` to disable debugging when the error occurs. Specify `true` to enable debugging when the error occurs.
- `debugLevel`: Specifies how much information is provided for debugging. You can use one of the following debug levels:
 - 0: prints a concise description on a single line
 - 1: prints a subset of target.xml file information
 - 2: prints the entire target.xml file

Example

```
./wsadmin.sh|.bat -lang jython -f odrDebug.py setHttpDebug my_node my_ODR 503 true 1
```

Result

If a 503 error occurs because no servers are started, the `SystemOut.log` file contains the following message:

```
[11/3/07 19:20:00:427 EDT] 00000034 HttpDebugResp I URL:
http://draco02.rtp.raleigh.ibm.com:80/A/CpuAndSleepBound,
statusCode: 503, reason: no servers in cluster xd61/DC1 are running web module
xd61/A/microwebapp.war
webModule=xd61/A/microwebapp.war
cluster=xd61/DC1
server=xd61/draco03/DC1_draco03
state=STOPPED
weight=0
reachable=true
server=xd61/draco04/DC1_draco04
state=STOPPED
weight=0
reachable=true
server=xd61/draco05/DC1_draco05
state=STOPPED
weight=0
reachable=true
server=xd61/draco06/DC1_draco06
state=STOPPED
weight=0
reachable=true
server=xd61/draco07/DC1_draco07
state=STOPPED
weight=0
reachable=true
```

Parent topic: [Scripts](#)

Related tasks:

[Creating ODRs](#)

Related reference:

[Administrative roles and privileges](#)

PlacementControllerProcs.jacl script

You can use the PlacementControllerProcs.jacl script to modify the application placement runtime configuration.

Purpose

Depending on your administrative role, you are allowed specific privileges when configuring the autonomic managers. To use this script to configure the autonomic managers, you must have administrator privileges.

Use the PlacementControllerProcs.jacl script to modify an entire series of application placement controller configuration options. You can complete the following operations on the runtime configuration.

- Enable or disable the application placement controller
- Make the specified node managed or unmanaged by the application placement controller
- Enable or disable a dynamic cluster
- Retrieve or set the values of additional settings

This script modifies only the runtime configuration of the application placement controller. To make persistent changes to the configuration, use the xd_APCconfig.jacl script.

Location

The PlacementControllerProcs.jacl script is in the install_root/bin directory.

Usage

To run the PlacementControllerProcs.jacl script with the wsadmin utility, use the following command.

```
wsadmin -profile PlacementControllerProcs.jacl -c "insert_proc_parameters"
```

In place of the insert_proc_parameters variable, use the name of the procedure and the proper variable values to complete your changes. You must enclose the procedures in quotation marks ("). You might have to modify the wsadmin command to wsadmin.sh or wsadmin.bat, depending on your operating environment.

Procedures

You can use the following command to see a list of all the available procedures.

```
wsadmin -profile PlacementControllerProcs.jacl -c "help"
```

The following procedures are available.

anyFailedServerOperations

Displays a list of failed starts by servers that the application placement controller tried to start.

closeLockMultiAPCGate removeAPCID

Closes and locks a gate for a multiple application placement controller setup. The APCGate mechanism can optionally be used to ensure that multiple placement controllers in different WebSphere® Virtual Enterprise systems do not run their decision-making cycles at the same time. Enabling the APCGate mechanism is important when multiple separate WebSphere Virtual Enterprise cells are sharing the same physical hardware.

deferFailedServerOperations

Disables the enablement of maintenance mode on any servers that failed to start by the application placement controller.

disable

Disables the application placement controller.

disableDynamicCluster dynamic_cluster_name

Sets the specified dynamic cluster to run manually.

disableNode node_name

Makes the specified node unmanaged by the application placement controller.

enable

Enables the application placement controller.

enableDynamicCluster dynamic_cluster_name

Sets the specified dynamic cluster to run automatically.

enableNode node_name

Enables the application placement controller for the specified node.

findBestLocation cell_name node_name server_name

Returns one of three values that indicates an action to take when you attempt to stop a server that is a member of a dynamic cluster. This procedure returns one of the following values:

- **null**: You can stop the server that was passed into the procedure without problems.
- **server name**: You should not stop the server name that you passed into the procedure.

- **alternate server name:** If the name of another server is returned, you should stop that server first.

getApprovalTimeout

When running in supervised mode, returns the amount of time to wait for an administrator to approve a task before the task times out.

getMinTimeBetweenPlacementChange

Returns the minimum time between two consecutive placement changes.

getNodeName

Displays the node on which the application placement controller is running.

getServerOperationTimeout

Returns the amount of time after which the start and stop operation that is performed on the server is considered a failure, if not completed.

handleFailedServerOperations

Places any servers into maintenance mode that did not start when the application placement controller tried to start the server.

isNodeInUse node_name

Checks if a node is still in use after it is in maintenance mode.

isPrimary

Returns if this node is the primary node where the application placement controller is running.

isEnabled

Returns if the application placement controller is enabled.

openMultiAPCGate remoteAPCID

Opens the gate for a multiple application placement controller setup.

recomputePlacement

Triggers the application placement controller to compute its optimization and perform any necessary placement changes.

setMinTimeBetweenPlacementChange time

Sets the minimum time, in minutes, between two consecutive placement changes.

setServerOperationTimeout timeout

Sets the timeout, in minutes, after which the start or stop operation that is performed on the server is considered a failure, if not completed.

setApprovalTimeout approval_timeout

Sets the amount of time, in minutes, to wait for the administrator approval when operating in supervised mode before considering the task denied.

unlockMultiAPCGate remoteAPCID

Unlocks the gate for a multiple application placement controller setup.

enableElasticity

Enables elasticity mode in the application placement controller. Valid arguments are true or false.

isElasticityModeEnabled

Specifies whether elasticity mode is enabled. The return value is either true or false.

elasticityMode

Sets the runtime tasks for application placement elasticity as supervised or automatic. Valid arguments are 0 for automatic or 1 for supervised.

getElasticityMode

Specifies if application placement elasticity is set to supervised or automatic. The return value is either 0 or 1.

elasticityModeTimeout

Specifies the amount of time, in minutes, that an elasticity action is allowed to complete before it is considered a failure. For example, if you want to set the operation timeout value to 30 minutes, specify the procedure as **elasticityModeTimeout 30 0**.

getElasticityTimeoutMins

Returns the amount of time, in minutes, that an elasticity action is allowed to complete before it is considered a failure.

Parent topic: [Scripts](#)

Related concepts:

[Overview of application placement](#)

Related tasks:

[Configuring dynamic application placement](#)

[Monitoring and tuning the application placement controller](#)

Related reference:

[Administrative roles and privileges](#)

pluginmerge.bat|.sh script

You can use the **pluginmerge .bat|.sh** script to combine the plugin-cfg.xml files from two, unbridged cells. Thus, the IBM® HTTP server plug-in will either load-balance across the cells, or failover to a second (backup) cell.

Purpose

The **pluginmerge .bat|.sh** script combines the plugin-cfg.xml files from two, unbridged cells so that the IBM HTTP server plug-in will either load-balance across the cells, or failover to a second (backup) cell.

Location

The **pluginmerge .bat|.sh** script is located in the *install_root/bin* directory. Before running this script, ensure that you have the environment variable **WAS_HOME** configured to point to the directory of your WebSphere® installation.

Usage

The default script usage follows (Linux):

```
pluginMerge.sh [-L] file1.xml file2.xml outfile.xml
```

To display help, invoke the tool with no arguments.

Parameters

-L

The generated merged-plugin-cfg.xml instructs the IBM HTTP server plug-in to load-balance traffic to applications which are common to both cells. The IBM HTTP server continues to affine to a particular server for in-session requests, but for out-of-session requests, a different ODR for each new request is selected. If -L is not specified, the IBM HTTP server will configure the servers in plugin-cfg1.xml as the primary servers and the servers in plugin-cfg2.xml as the backup servers. With this configuration, the behavior of the server is to send requests to the backup servers only if the primary are unavailable

plugin-cfg1.xml

The plugin-cfg.xml file generated by an ODR in the first cluster

plugin-cfg2.xml

The plugin-cfg.xml file generated by an ODR in the second cluster.

merged-plugin-cfg.xml

This is the output, merged plugin-cfg.xml

Note: the plug-in merge script only supports the merging of exactly two plugin-cfg.xml files.

Parent topic: [Scripts](#)

Parent topic: [Scripts](#)

Related concepts:

[Cell affinity function](#)

Related tasks:

[Creating and configuring ODRs](#)

[Configuring ODRs](#)

[Enabling cell affinity](#)

[Configuring cell affinity in a multi-tiered environment](#)

[Defining generic server clusters for remote ODR cells](#)

propagatePluginCfg.py

Use the `propagatePluginCfg.py` script to automatically propagate the generated `plugin-cfg.xml` file.

The `propagatePluginCfg.py` script file is located in the `<WAS_HOME>/bin/propagatePluginCfg.py` directory.

To use the `propagatePluginCfg.py` script to propagate the generated `plugin-cfg.xml` file, configure the **Plugin config change script** using the administrative console, or use the `ODCPluginCfgUpdateScript_<definitionID>` custom property. For more information about the administrative console procedure, read about propagating the `plugin-cfg.xml` file. Read about generating the plug-in configuration in a high availability environment for more information about the `ODCPluginCfgUpdateScript_<definitionID>` custom property.

The `propagatePluginCfg.py` script file supports two operations: **copyToRepository** and **uploadToIHSAdminServer**:

- `copyToRepository <localPath> <remoteRepositoryPath> [<remoteRepositoryPath> ...]` copies a locally generated `plugin-cfg.xml` file into one or more locations in the deployment manager configuration repository, and performs a node synchronization to push the `plugin-cfg.xml` file to all remote nodes.

Some usage examples are:

- `<WAS_HOME>/bin/wsadmin.sh -f <WAS_HOME>/bin/propagatePluginCfg.py -lang jython copyToRepository /tmp/plugin-cfg.xml cells/myCell/nodes/myIHSNode/servers/myIHSServer/plugin-cfg.xml`
- `<WAS_HOME>/bin/wsadmin.sh -f <WAS_HOME>/bin/propagatePluginCfg.py -lang jython copyToRepository /tmp/plugin-cfg.xml cells/myCell/nodes/myIHSNode1/servers/myIHSServer1/plugin-cfg.xml cells/myCell1/nodes/myIHSNode2/servers/myIHSServer2/plugin-cfg.xml`
- `uploadToIHSAdminServer <url> <uid> <pwd> <localPath> <remotePath>` uploads a local `plugin-cfg.xml` file to a remote IBM® HTTP Server (IHS) administration server.

where:

- `localPath` is the absolute path to the local file that is to be copied or uploaded.
- `remoteRepositoryPath` is the relative path in the repository where the local file can be copied to.
- `remotePath` is the absolute path of the file on the IHS administration server.
- `url` is a IHS administration server URL of the form `<proto>://<host>:<port>`, where `proto` is HTTP or HTTPS.
- `uid` is the user name identifier to use in authenticating to the IHS administration server.
- `pwd` is the password to use in authenticating to the IHS administration server.

A usage example is:

```
<WAS_HOME>/bin/wsadmin.sh -f <WAS_HOME>/bin/propagatePluginCfg.py -lang
jython
https://myIHSHost:8192 ihsusr ihsusrPassword /tmp/plugin-cfg.xml
/opt/HTTPServer/plugins/config/webserver1/plugin-cfg.xml
```

You can also automatically propagate the generated `plugin-cfg.xml` file to web servers using the **autoPropagate** keyword option in the administrative console. Read about propagating the `plugin-cfg.xml` file to web servers using the **autoPropagate** keyword option for more information.

Related tasks:

- [Configuring an ODR to dynamically update the web server plug-in configuration](#)
- [Generating the plug-in configuration in a high availability environment](#)
- [Propagating the plugin-cfg.xml file to web servers](#)

serverQuiesce.py script

You can use the serverQuiesce.py script to decrease workload to a Session Initiation Protocol (SIP) container over an interval, and optionally stop the server. You can also cancel the quiesce during a set time interval if certain conditions are met.

Purpose

With this script, you can gradually decrease workload to a SIP container over an interval. You can also optionally stop the server.

z/OS Restriction: WebSphere® Virtual Enterprise does not support SIP features on the z/OS® operating system.

Location

The serverQuiesce.py script is in the *install_root/bin* directory.

Usage

Run the following command.

```
./wsadmin.sh -f serverQuiesce.py <operation> [options]
```

Replace the *<operation>* and *[options]* variables with the appropriate operation and options for your task.

Operations

Use the following operations with the serverQuiesce.py script:

quiesceServer

Gradually decreases the workload to a SIP container.

Table 1. quiesceServer options

Option	Description
nodeName	Specifies the name of the node.
serverName	Specifies the name of the server.
phaseOneInterval	Specifies the time in milliseconds for the interval.
performPhaseTwo	Specifies a boolean that indicates whether to stop the server.

cancelQuiesce

Cancels the quiesce during the time interval that you set. To successfully cancel the quiesce, ensure that the time interval has not expired and the optional server stop occurs at the end of the interval.

Table 2. cancelQuiesce options

Option	Description
nodeName	Specifies the name of the node.
serverName	Specifies the name of the server.

Example

```
./wsadmin.sh -f serverQuiesce.py quiesceServer node1 server1 300000 false
```

Parent topic: [Scripts](#)

Related concepts:

[Overview of request flow prioritization](#)

Related reference:

[Administrative roles and privileges](#)

Related information:

[Rules for ODR service policy administrative tasks](#)

servicepolicy.py script

You can use the servicepolicy.py script to perform operations on service policies from the command line, such as creating service policies, removing service policies, and editing transaction classes.

Purpose

You can complete the following actions with the servicepolicy.py script.

- Create a service policy
- Remove a service policy
- Create a transaction class
- Remove a transaction class

To create, modify, and remove service policies and transaction classes, you must have configurator or administrator administrative privileges.

Location

The servicepolicy.py script is in the `install_root/bin` directory.

Usage

The script usage for general help follows:

```
./wsadmin.sh|bat -lang jython -f servicepolicy.py
```

The script usage for operation-specific help follows:

```
./wsadmin.sh|bat -lang jython -f servicepolicy.py operation --help
```

Operations

createServicePolicy

Creates a service policy with the specified options. You must create and associate transaction classes separately.

- **--spname**: Specifies a name for the service policy that is unique in the cell.
- **--spgt**: Specifies an integer that represents one of the following service policy goal types:
 - 0: discretionary
 - 1: average response time
 - 2: percentile response time
 - 4: completion time
- **--spgv**: Specifies a service policy goal value for non-discretionary goals. This value is assumed to be in milliseconds if you do not specify the units.
- **--spgvu**: Specifies an integer that represents a service policy goal value for non-discretionary goals. This value is assumed to be in milliseconds if you do not specify the units.
 - 0: milliseconds
 - 1: seconds
 - 2 : minutes
- **--sppgv**: Specifies an integer that represents a percentile value for a service policy with percentile response time goal between 1 and 100.
- **--spi**: Specifies an integer that represents one of the following service policy goal types:
 - 1: highest
 - 2: higher
 - 3: high
 - 4: medium
 - 5: low
 - 6: lower
 - 7: lowest
- **--spd**: Specifies a service policy description.

removeServicePolicy

Deletes an existing service policy with the specified option.

- **--spname**: Specifies the unique name for the service policy that you want to remove.

createTransactionClass

Creates a transaction class with the specified options.

- **--spname**: Specifies a name for the service policy that is unique in the cell.
- **--tcname**: Specifies a name for the transaction class that you want to create that is unique in the cell.
- **--tcd**: Specifies a transaction class description.

removeTransactionClass

Removes a transaction class with the specified option. All Uniform Resource Identifiers (URIs) in the transaction class are no longer associated with the parent service policy. If a request comes in for these URIs

and they are not associated with a new service policy and transaction class, they are classified to the default service policy with a discretionary goal.

- **--tcname:** Specifies the cell-unique name for the transaction class that you want to remove.
-

Example

Create a service policy:

```
./wsadmin.sh|bat -lang jython -f servicepolicy.py createServicePolicy --spname Platinum --spgt 2 --spgv 3000 --spgvu 0 --sppgv 80 --spi 5
```

Remove an existing service policy:

```
./wsadmin.sh|bat -lang jython -f servicepolicy.py removeServicePolicy --spname Bronze
```

Create a new transaction class:

```
./wsadmin.sh|bat -lang jython -f servicepolicy.py createTransactionClass --spname Platinum --tcname PlatinumWorkload --tcd 'my platinum workload'
```

Remove an existing transaction class:

```
./wsadmin.sh|bat -lang jython -f servicepolicy.py removeTransactionClass --tcname PlatinumWorkload
```

Parent topic: [Scripts](#)

Related tasks:

[Defining a service policy](#)

Related reference:

[Administrative roles and privileges](#)

unlinkCells|unlinkCellsZOS script

You can use the script to disable the overlay communication between multiple cells.

Purpose

Distributed platforms Use the **unlinkCells** script to disable communication between WebSphere® Virtual Enterprise cells that were previously linked by using the **linkCells** script. After you run the **unlinkCells** script, the on demand router (ODR) no longer routes work requests to the unlinked cell.

z/OS On z/OS® systems use **unlinkCellsZOS**.

Location

Distributed platforms The **unlinkCells** script is located in the *app_server_root/bin* directory.

z/OS The **unlinkCellsZOS** script is located in the *app_server_root/bin* directory.

Usage

Distributed platforms Run the **unlinkCells** script from the center cell to unlink the center cell from a point cell.

```
./unlinkCells.sh  
center_cell_deployment_manager:center_cell_soap_port:user_id:password  
point_cell_deployment_manager:point_cell_soap_port:user_id:password
```

z/OS Run the **unlinkCellsZOS** script from the center cell to unlink the center cell from a point cell.

```
./unlinkCellsZOS.sh  
center_cell_deployment_manager:center_cell_soap_port:user_id:password  
point_cell_deployment_manager:point_cell_soap_port:user_id:password
```

Example

Consider a scenario in which there are two security-enabled cells, center and point1, that were previously linked together by using the **linkCells** script. For the center cell, the host name of the deployment manager is centerHost, the SOAP port is 8879, the user name is centerUID, and the password is centerPWD. For the point cell, the host name of the deployment manager is point1Host, the SOAP port is 8880, the user name is point1UID, and the password is point1PWD. The following example illustrates how to unlink the center and point1 cells.

```
./unlinkCells.sh centerHost:8879:centerUID:centerPWD  
point1Host:8880:point1UID:point1PWD
```

Parent topic: [Scripts](#)

Related tasks:

[Configuring multi-cell performance management](#)
[Manually disabling communication between multiple cells](#)

Related reference:

[linkCells|linkCellsZOS script](#)
[importOverlayConfig.py script](#)

useWVEBB.py

Run the useWVEBB.py script to enable the BBSON bulletin board and alleviate the dependency on the WebSphere® Application Server Network Deployment high availability manager.

Purpose

You can use the useWVEBB.py script to enable the BBSON bulletin board, which alleviates the dependency on the WebSphere Application Server Network Deployment high availability manager. See [BBSON bulletin board](#) for restrictions and considerations.

Location

The useWVEBB.py script is in the *install_root/bin* directory.

Usage

To run the useWVEBB.py script, run the following command on each WebSphere Virtual Enterprise cell.

```
./wsadmin.sh -lang jython -f useWVEBB.py
```

Run the following command to disable the BBSON and revert to the product's default bulletin board.

```
./wsadmin.sh -lang jython -f useWVEBB.py -useHAMBB
```

Parent topic: [Scripts](#)

Related concepts:

[BBSON bulletin board](#)

[Core groups \(high availability domains\)](#)

Related tasks:

☞ [Creating a new core group](#)

Related information:

☞ [Setting up a high availability environment.](#)

encodePassword.sh|bat script

You can use the **encodePassword.sh|bat** script to encode a password value that you enter in the administrative console. By encoding your password, you can prevent the password from being shared.

Purpose

You can use the **encodePassword.sh|bat** script to encode the passwords that you enter as custom property values into the administrative console.

Location

The **encodePassword.sh|bat** script is located in the *install_root/bin* directory.

Usage

The default script usage follows:

```
./encodePassword.sh|bat
```

Parameters

Use the following parameter with the **encodePassword.sh|bat** script:

password

Specifies the password that you want to encode.

Example

```
./encodePassword.sh|bat password01
```

Result

The script returns an encoded password that you can copy and use for your custom property value.

Parent topic: [Scripts](#)

Related tasks:

[Configuring application placement for cells that share the same nodes](#)

Related information:

[VMware custom properties](#)

workclassoperations.py script

You can use the workclassoperations.py script to create, update, and manage work classes.

Purpose

You can complete the following actions with the workclassoperations.py script:

- Create and update work classes for enterprise applications and generic server clusters
- List the rules in work classes
- List the members of work classes
- Delete work classes for the various policies and communication protocols
- Set default actions
- Modify default actions
- Fetch default actions
- Add rules
- Delete rules
- Delete members

Location

The workclassoperations.py script is in the *install_root/bin* directory.

Usage

The script usage for general help follows:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py
```

The script usage for operation-specific help follows:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py operation --help
```

Depending on where you are issuing the command, you might have to specify the path to the workclassoperation.py script. For example, you create a profile named Dmgr01 located in the *c:\WAS_INSTALL_ROOT\profiles\Dmgr01\bin* directory.

```
./wsadmin.sh -lang jython -f ../../../../bin/workclassoperations.py
```

Note that if the deployment manager is not started, you must include the additional parameter **-conntype NONE**:

```
./wsadmin.sh -lang jython -conntype NONE workclassoperations.py
```

Operations

ListWorkClasses

Lists work classes by type. The format of listing is *workclassname type appname/odrname deploymentname*

- **--type**: Specifies the type of work class to list:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
- **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
- **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.

createWorkClass

Creates a work class. The new service policy does not contain any application modules or classification rules. You must create and associate application modules and classification rules separately.

- **:**
 - **--type**: Specifies the type of work class to list:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--nodename**: Specifies the name of the node. Required if you specify the **--type** parameter as GSP or GRP.

- **--wcname**: Specifies the unique name for the work class within the specified type.
- **--protocol**: Specifies the communications protocol that transmits the request. Currently supported protocols are HTTP, IIOP, SOAP, and JMS.
- **--wcaction**: Specifies the default action to take when a request is matched to a member for the specified work class.
- **--module**: Specifies the application module to associate members. Required if you specify the **--appname** and **--members** parameters.
- **--members**: Specifies the protocol-specific pattern:
 - HTTP: /test1?/test2/*
 - IIOP: ejbName:ejbMethod?ejbName:ejbMethod
 - SOAP: webService:operationName?webService:operationName
 - JMS: bus:destination?bus:destination
- **--rule**: Specifies a classification rule in the format of "priority?rule?action"
- **--virtualhost**: Specifies the virtual host. Required if you specify the **--type** parameter as GSP or GRP.
- **removeWorkClass**: Removes a work class.
 - **--type**: Specifies the type of work class to remove:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.
- **addMembers**: Adds members to a work class. The ODR uses the new member to match incoming requests.
 - **--type**: Specifies the type of work class to remove:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.
 - **--protocol**: Specifies the communications protocol that transmits the request. Currently supported protocols are HTTP, IIOP, SOAP, and JMS.
 - **--module**: Specifies the application module to associate members. Required if you specify the **--appname** and **--members** parameters.
 - **--members**: Specifies the protocol-specific pattern:
 - HTTP: /test1?/test2/*
 - IIOP: ejbName:ejbMethod?ejbName:ejbMethod
 - SOAP: webService:operationName?webService:operationName
 - JMS: bus:destination?bus:destination
 - **--ejbname**: Specifies the name of the EJB. Required only when you specify the **--protocol** parameter as JMS and the **--module** parameter as IIOP.
- **removeMembers**: Removes members from a work class. The member is no longer used by the ODR.
 - **--type**: Specifies the type of work class to modify:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.
 - **--protocol**: Specifies the communications protocol that transmits the request. Currently supported protocols are HTTP, IIOP, SOAP, and JMS.
 - **--module**: Specifies the application module from which to disassociate members. Required if you specify the **--appname** and **--members** parameters.
 - **--members**: Specifies the protocol-specific pattern:
 - HTTP: /test1?/test2/*
 - IIOP: ejbName:ejbMethod?ejbName:ejbMethod
 - SOAP: webService:operationName?webService:operationName
 - JMS: bus:destination?bus:destination
- **listMembers**: Lists members of a work class.
 - **--type**: Specifies the type of work class to list:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter

- as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--nodename**: Specifies the name of the node. Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.
 - **--protocol**: Specifies the communications protocol that transmits the request. Currently supported protocols are HTTP, IIOP, SOAP, and JMS.
- **addRules**: Adds a classification rule that is used by the ODR to a work class. If you issue the command with the like wildcard '%' on UNIX operating systems, replace the escape sequence (\%) with (%).
- **--type**: Specifies the type of work class to modify:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--nodename**: Specifies the name of the node. Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.
 - **--rule**: Specifies a classification rule in the format of "priority?rule?action"
- **removeRule**: Removes a classification rule from a work class. The rule is no longer used by the ODR.
 - **--type**: Specifies the type of work class to modify:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.
 - **--expression**: Specifies the rule expression.
 - **--priority**: Specifies the priority of the rule to match. The rule with the lowest priority is matched first.
- **listRules**: Lists classification rules.
 - **--type**: Specifies the type of work class to modify:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.
- **modifyDefaultAction**: Edits the default action for a work class.
 - **--type**: Specifies the type of work class to modify:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.
 - **--waction**: Specifies the default action to take when a request is matched to a member for the specified work class.
- **getDefaultAction**: Specifies the default action for a work class.
 - **--type**: Specifies the type of work class to list:
 - ASP: application service policy
 - ARP: application routing policy
 - GSP: generic server service policy
 - GRP: generic server routing policy
 - **--appname**: Specifies the name of the application. Required if you specify the **--type** parameter as ASP or ARP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.
- **modifyVirtualHost**: Edits the virtual host for a work class.
 - **--type**: Specifies the type of work class to modify. The valid types are GSP and GRP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.

- **--wcname**: Specifies the unique name for the work class within the specified type.
- **--virtualhost**: Specifies the virtual host. Required if you specify the **--type** parameter as GSP or GRP.
-
- **getVirtualHost**: Lists the virtual host for a work class.
 - **--type**: Specifies the type of work class to list. The valid types are GSP and GRP.
 - **--odrname**: Specifies the name of the on demand router (ODR). Required if you specify the **--type** parameter as GSP or GRP.
 - **--wcname**: Specifies the unique name for the work class within the specified type.

Example

List all work classes:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py listWorkClasses
```

List all work classes for application Trade:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py listWorkClasses --appname Trade
```

Create a HTTP application routing policy work class for application Trade with a default action of reject and reject code of 404:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py createWorkClass --type ARP --wcname CustomWorkClass --protocol HTTP --waction "reject?404" --appname Trade --module myModule.war --members "/test1?/test2"
```

List all routing policy work classes for application Trade:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py listWorkClasses --type ARP --appname Trade
```

View the default action of a HTTP application routing policy work class for application Trade:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py getDefaultAction --type ARP --wcname CustomWorkClass --appname Trade
```

Modify the default action to Default_TC in a HTTP application service policy work class for application Trade-edition1.0:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py modifyDefaultAction --type ASP --wcname Default_HTTP_WC --waction Default_TC --appname Trade-edition1.0
```

Map the default HTTP work class of edition 1.2 of the DayTrader application to the default transaction class for the service policy GOLD:

```
import java.lang.System as System

wasinstallroot=System.getProperty("was.install.root")
execfile(wasinstallroot+"/bin/XDPYModules.py")
import java.lang.System as System

wasinstallroot=System.getProperty("was.install.root")
execfile(wasinstallroot+"/bin/XDPYModules.py")

#application with name DayTrader installed with edition 1.2
appName="DayTrader-edition1.2"
#Default_TC_GOLD is the default transaction class mapped to service policy GOLD
modifyAppSLADefaultAction("Default_HTTP_WC", appName, tcName, 0)

AdminConfig.save()
```

Add a classification rule to the application service policy work class for application Trade:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py addRules --wcname CustomWorkClass --appname Trade --type ARP --rule "1?clienthost='localhost' and serverhost like '\\%.ibm.com'?permit?Trade"
```

Remove a classification rule by priority from the application service policy work class for application Trade:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py removeRule --wcname CustomWorkClass --appname Trade --type ARP --priority 1
```

Remove a HTTP application routing policy work class for application Trade:

```
./wsadmin.sh|bat -lang jython -f workclassoperations.py removeWorkClass --type ARP --wcname CustomWorkClass --appname Trade
```


Parent topic: [Scripts](#)

Related concepts:
[Overview of work classes](#)

Related reference:
[Routing and service policies](#)

xd_APCconfig.jacl script

Use the xd_APCconfig.jacl script to make persistent changes to the application placement controller configuration.

Purpose

You can use the xd_APCconfig.jacl script to modify attributes and custom properties of the application placement controller and dynamic clusters.

Location

The xd_APCconfig.jacl script is in the *install_root/bin* directory.

Usage

To change the configuration, you must have configurator or administrator administrative privileges.

To run the xd_APCconfig.jacl script with the wsadmin tool, use the following command:

```
wsadmin.sh -profile xd_APCconfig.jacl -c "insert_proc"
```

In place of the *insert_proc* variable, use the name of the procedure and the proper variable values to complete your changes. You must enclose the procedures in quotation marks. You might have to modify the **wsadmin** command to **wsadmin.sh** or **wsadmin.bat**, depending upon your operating environment.

Procedures

To receive a list of the valid attributes that you can set on the application placement controller or dynamic cluster, run the following **wsadmin** command:

```
$AdminConfig show [$AdminConfig getid "/DynamicCluster:clustername/"]
```

The following procedures are available:

setAPCAtribute *attribute_value*

Sets the defined attribute and value on the application placement controller. For a list of the valid attributes for the application placement controller, run the following **wsadmin** command:

```
$AdminConfig show [$AdminConfig getid "/AppPlacementController:/"]
```

getAPCAtribute *attribute_name*

Displays the specified application placement controller attribute and value.

setAPCCustomProperty *property value description*

Sets the specified application placement controller custom property name, value, and description. See [Application placement custom properties](#) for custom properties that you can set on the application placement controller. For example, the following command sets the numVerticalInstances custom property:

```
wsadmin.sh -profile xd_APCconfig.jacl -c "setAPCCustomProperty numVerticalInstances.myNode 3 verticalstacking"
```

getAPCCustomProperty *custom_property_name*

Gets the value and description for the specified application placement controller custom property. For example, the following command gets the value of the numVerticalInstances custom property:

```
wsadmin.sh -profile xd_APCconfig.jacl -c "getAPCCustomProperty numVerticalInstances.myNode"
```

setDynamicClusterAttribute *dynamic_cluster_name attribute_value*

Sets the defined attribute and value for the specified dynamic cluster. For a list of the valid attributes for dynamic clusters, run the following **wsadmin** command:

```
$AdminConfig show [$AdminConfig getid "/DynamicCluster:clustername/"]
```

getDynamicClusterAttribute *dynamic_cluster_name attribute*

Displays the value for the specified dynamic cluster attribute.

setDynamicClusterCustomProperty *dynamic_cluster_name custom_property_name value description*

Sets a custom property on the specified dynamic cluster. See [Dynamic cluster custom properties](#) for custom properties that you can set on dynamic clusters. For example, the following command sets the HTTPSessionRebalanceOff custom property:

```
wsadmin.sh -profile xd_APCconfig.jacl -c "setDynamicClusterCustomProperty my_dyncluster HTTPSessionRebalanceOff true disablerebalancing"
```

getDynamicClusterCustomProperty *dynamic_cluster_name property*

Gets the value of the specified custom property for a dynamic cluster. For example, the following command gets the value of the HTTPSessionRebalanceOff custom property:

```
wsadmin.sh -profile xd_APCconfig.jacl -c "getDynamicClusterCustomProperty my_dyncluster HTTPSessionRebalanceOff"
```

Parent topic: [Scripts](#)

Related tasks:

[Configuring dynamic application placement](#)

Related reference:

[Administrative roles and privileges](#)

Administrative roles and privileges

Roles and privileges vary depending on your administrative role and the component.

Administrative roles and privileges

For complete definitions of administrative roles in WebSphere® Application Server and how to assign them, see [Authorizing access to administrative roles](#).

Table 1. Administrative roles and privileges

GUI	Monitor privileges	Operator privileges	Configurator privileges	Administrator privileges
Dynamic clusters	View the configuration.	View the configuration. Can change the operating mode of the dynamic cluster. Can start/stop cluster members.	Can alter the configuration of the dynamic cluster, including server template for the dynamic cluster. Can create new and delete dynamic clusters. Cannot change the operating mode of the dynamic cluster. Cannot start/stop cluster members.	Has all privileges.
Task management notification configuration	View the information.	View the information.	Can change the configuration.	Has all privileges.
Task management runtime tasks	View the information	Can run tasks.	View information.	Has all privileges.
Runtime operations / runtime topology	View the information	View the information. Can start/stop servers. Can set maintenance mode on nodes.	View the information.	Has all privileges.
Runtime operations / runtime charting	Has all privileges.	Has all privileges.	Has all privileges.	Has all privileges.
Runtime operations / runtime map	Has all privileges.	Has all privileges.	Has all privileges.	Has all privileges.
Runtime operations / job management	View the jobs.	Has all privileges.	View the jobs.	Has all privileges.
Service policies	View the information.	View the information.	Can create and delete service policies and transaction classes.	Has all privileges.

On demand router (ODR) configuration	View the information.	View information. Can start and stop ODRs.	Can edit create, and delete ODRs.	Has all privileges.
ODR configuration / generic server cluster / service and routing policies	View the information.	View the information.	Has all privileges except modifying rules through the rule builder	Has all privileges.
Application routing policies	View the information.	View the information.	Has all privileges.	Has all privileges.
Application edition control center	View the information.	View the information.	Has all privileges.	Has all privileges.
Job scheduler (Compute Grid only)	View the information.	View the information.	Has all privileges.	Has all privileges.
Visualization data service	View the information.	View the information.	Can change the configuration.	Has all privileges.
High Availability (HA) deployment manager	View the information.	Has all privileges.	View the information.	Has all privileges.
Autonomic manager configuration	View the information.	View the information on the configuration tab. Can change the settings on the runtime tab.	Can change the configuration but not the runtime settings.	Has all privileges.
Health policies	View the information.	View the information.	Has all privileges.	Has all privileges.
Node maintenance mode	View the information.	Can: <ul style="list-style-type: none"> • set and unset maintenance nodes • synchronize nodes • stop nodes 	Can: <ul style="list-style-type: none"> • view the configuration. • add and remove nodes. 	Has all privileges.
Extended repository service	View the information.	View the information.	Has all privileges.	Has all privileges.
Middleware applications	View the information.	View the information.	Has all privileges.	Has all privileges.

Parent topic: [Reference](#)

Related tasks:

☞ [Authorizing access to administrative roles](#)

Administrator scripting interfaces

This reference information provides examples of how to complete various tasks using the administrative scripting interfaces.

Scripting interfaces

Use the following scripting interfaces to automate tasks that you might normally accomplish with the administrative console.

[Application edition management administrative tasks](#)

You can use the Jacl or Jython scripting languages to manage application editions. You can also use the AdminApp object to manage applications with specific editions.

[Centralized installation manager administrative tasks](#)

You can use the Jacl or Jython scripting languages to use the various features of the centralized installation manager with the wsadmin tool. Use the commands and parameters to install, uninstall, and manage various software packages and maintenance files.

[Dynamic cluster administrative tasks](#)

Use the dynamic cluster commands to view or edit the dynamic clusters, without using the administrative console.

[Default work class administrative tasks](#)

You can use the default work class administrative tasks to create work classes.

[Health policy administrative tasks](#)

You can use the following administrative tasks to create, delete, and manage health policies.

[Maintenance mode administrative tasks](#)

You can use the server maintenance mode commands to view or edit the maintenance mode of your servers without using the administrative console. To set maintenance mode on a node, you can use the NodeGroupManager MBean.

[Middleware application administrative tasks](#)

You can use the Jacl or Jython scripting languages to create and manage middleware applications with the wsadmin tool.

[Middleware descriptor administrative tasks](#)

You can use the middleware descriptor administrative tasks to list the middleware descriptors and versions, display or change middleware descriptor information, and modify the middleware descriptor discovery interval.

[Middleware server creation administrative tasks](#)

You can use administrative tasks to write a script that can recreate your middleware server configuration.

[Middleware server management administrative tasks](#)

You can use the Jython and Jacl scripting languages to manage middleware servers with the wsadmin tool. Use the commands and parameters in the MiddlewareServerManagement group.

[Middleware server template administrative tasks](#)

Use middleware server template administrative tasks to create server templates for middleware servers that you have created.

[PHP server administrative tasks](#)

Use the following commands to create and configure PHP servers and server templates.

[Rules for ODR routing policy administrative tasks](#)

You can use administrative tasks to configure HTTP or Session Initiation Protocol (SIP) rules for the on demand router (ODR) routing policy.

[Rules for ODR service policy administrative tasks](#)

You can use administrative tasks to configure Session Initiation Protocol (SIP) or HTTP rules for an on demand router (ODR) service policy.

[Rules for generic server cluster routing policy administrative tasks](#)

You can use administrative tasks to configure routing rules for the generic server cluster routing policy.

[Rules for generic server cluster service policy administrative tasks](#)

You can use administrative tasks to configure service rules for the generic server cluster routing policy.

[Ruleset administrative tasks](#)

You can use the ruleset commands to add, remove, or modify rules and actions of the ruleset.xml file without using the administrative console.

[Runtime operations user preferences administrative tasks](#)

You can use the Jacl and Jython scripting languages to clone, set, and get preferences with the wsadmin tool. Use the commands and parameters in the UserPreferences group.

Parent topic: [Reference](#)

Related reference:

 [Wsadmin tool](#)

Application edition management administrative tasks

You can use the Jacl or Jython scripting languages to manage application editions. You can also use the AdminApp object to manage applications with specific editions.

Use the following commands to activate, deactivate, perform a rollout on, validate, list, and verify an application edition. You can also clone an existing dynamic cluster to use a different node group, and you can cancel the validation mode on an application edition:

- [activateEdition](#)
- [deactivateEdition](#)
- [rolloutEdition](#)
- [validateEdition](#)
- [listEditions](#)
- [isEditionExists](#)
- [cloneDynamicCluster](#)
- [cancelValidation](#)
- [getEditionState](#)

You can perform the following operations using the AdminApp object in wsadmin to manage applications with editions:

- [Install an application with an edition](#)
- [Update an application with an edition](#)
- [Edit an application with an edition](#)
- [Uninstall an application with an edition](#)

For a complete list of operations that you can perform using the AdminApp object, see [Commands for the AdminApp object](#).

activateEdition

The **activateEdition** command activates an edition.

Target object

None.

Parameters

-appName

Specifies the name of the application. (String)

-edition

Specifies the name of the application edition. (String)

Return value

The command returns:

- true: success
- false: failure

Batch mode example usage

- Using Jacl:

```
$AdminTask activateEdition {-appName BeenThere -edition 1.0}
```

- Using Jython string:

```
AdminTask.activateEdition ('[-appName BeenThere -edition 1.0]')
```

- Using Jython list:

```
AdminTask.activateEdition (['-appName', 'BeenThere', '-edition', '1.0'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask activateEdition {-interactive}
```

- Using Jython string:

```
AdminTask.activateEdition ('[-interactive]')
```

- Using Jython list:

```
AdminTask.activateEdition (['-interactive'])
```

deactivateEdition

The **deactivateEdition** command deactivates an edition.

Target object

None.

Parameters

-appName

Specifies the name of the application. (String)

-edition

Specifies the name of the application edition. (String)

Return value

The command returns:

- true: success
- false: failure

Batch mode example usage

- Using Jacl:

```
$AdminTask deactivateEdition {-appName BeenThere -edition 1.0}
```

Using Jython string:

```
AdminTask.deactivateEdition (['-appName BeenThere -edition 1.0'])
```

- Using Jython list:

```
AdminTask.deactivateEdition (['-appName', 'BeenThere', '-edition', '1.0'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask deactivateEdition {-interactive}
```

Using Jython string:

```
AdminTask.deactivateEdition (['-interactive'])
```

- Using Jython list:

```
AdminTask.deactivateEdition (['-interactive'])
```

rolloutEdition

The **rolloutEdition** command rolls out an edition and specifies the group size.

Target object

None.

Parameters

-appName

Specifies the name of the application. (String)

-edition

Specifies the value of the custom property. (String)

-params

Specifies configuration values for the rollout. (String)

rollout strategy

- **grouped**: Activates the specified application in place of the current edition of the same application across the cluster on which the application cluster deploys, N servers at a time, as specified by the **groupSize** keyword. The default group size is 1. During rollout, both editions of the application can serve requests. Always set the group number to be at least one less than the cluster size so that at least one cluster member serves the old edition while the rollout of the new edition starts.
- **atomic**: Activates the specified application in place of the current edition of the same application across the cluster on which the application cluster deploys, one half of the cluster at a time, such that only one edition of the application serves requests at any given time.

reset strategy

Specifies how to start an application edition instance during the rollout operation.

- **hard**: Stops or restarts the application server.
- **soft**: Stops or restarts the application instance, while leaving the application server running.

group size for group rollout

Specifies the number of servers to update when **rolloutStrategy** is grouped.

- integer

drainage interval

Specifies the number of seconds to wait before stopping an application edition instance during the rollout such that sessions can complete. The default is 30 seconds.

- integer

Specify the following properties for Session Initiation Protocol (SIP) application rollout:

quiesce strategy

Specifies if cluster members and servers quiesce after all active dialogs and sessions are complete, or if the cluster members and servers quiesce after a specified interval.

DEFAULT: Cluster members and servers quiesce when active dialogs and sessions complete.

INTERVAL: Cluster members and servers quiesce after a specified interval. You must also specify the **quiesceInterval** as an integer value in number of seconds.

Return value

The command returns:

- true: success
- false: failure

Batch mode example usage

- Using Jacl:

```
$AdminTask rolloutEdition {-appName BeenThere -edition 1.0 -params "{rolloutStrategy grouped}{resetStrategy soft}{groupSize 1}{drainageInterval 30}"}
```

```
$AdminTask rolloutEdition {-appName BeenThere -edition 1.0 -params "{quiesceStrategy INTERVAL}{quiesceInterval 30}"}
```

Using Jython string:

```
AdminTask.rolloutEdition ('[-appName BeenThere -edition 1.0 -params "{rolloutStrategy grouped}{resetStrategy soft}{groupSize 1}{drainageInterval 30}"']')
```

```
AdminTask.rolloutEdition ('[-appName BeenThere -edition 1.0 -params "{quiesceStrategy INTERVAL}{quiesceInterval 30}"']')
```

Note: The following example applies only to Version 6.1.0.5 running with WebSphere® Application Server Network Deployment Version 7.0.0.x.

```
AdminTask.rolloutEdition ('[-appName BeenThere -edition 1.0 -params [[rolloutStrategy grouped]{resetStrategy soft}[groupSize 1][drainageInterval 30]]']')
```

- Using Jython list:

```
AdminTask.rolloutEdition (['-appName', 'BeenThere', '-edition', '1.0', '-params', '{rolloutStrategy grouped}{resetStrategy soft}{groupSize 1}{drainageInterval 30}']')
```

```
AdminTask.rolloutEdition (['-appName', 'BeenThere', '-edition', '1.0', '-params',
'{quiesceStrategy INTERVAL}{quiesceInterval 30}'])
```

Note: The following example applies only to Version 6.1.0.5 running with WebSphere Application Server Network Deployment Version 7.0.0.x.

```
AdminTask.rolloutEdition (['-appName', 'BeenThere', '-edition', '1.0', '-params',
'[[rolloutStrategy grouped][resetStrategy soft][groupSize 1][drainageInterval 30]]'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask rolloutEdition {-interactive}
```

Using Jython string:

```
AdminTask.rolloutEdition (['-interactive'])
```

- Using Jython list:

```
AdminTask.rolloutEdition (['-interactive'])
```

validateEdition

The **validateEdition** command validates an edition.

Target object

None.

Parameters

-appName

Specifies the name of the application. (String)

-edition

Specifies the name of the application edition. (String)

-params

Specifies the parameters for setting the size of the cluster that is being cloned during validation. By default, the size of the original cluster is used if the parameters are not specified.

- Dynamic cluster:

dynClusterMaxSize

Specifies the maximum size of the dynamic cluster.

dynClusterMinSize

Specifies the minimum size of the dynamic cluster.

- Static cluster:

staticClusterSize

Specifies the size of the static cluster. The value of **staticClusterSize** cannot exceed the size of the static cluster that is being cloned.

Return value

The command returns:

- true: success
- false: failure

Batch mode example usage

- Using Jacl:

```
$AdminTask validateEdition {-appName BeenThere -edition 1.0 -params "{dynClusterMaxSize 2}
{dynClusterMinSize 1}"}
```

Using Jython string:

```
AdminTask.validateEdition (['-appName BeenThere -edition 1.0 -params "{dynClusterMaxSize 2}
{dynClusterMinSize 1}"]')
```

Note: The following example applies only to Version 6.1.0.5 running with WebSphere Application Server Network Deployment Version 7.0.0.x.

```
AdminTask.validateEdition ('[-appName BeenThere -edition 1.0 -params [[dynClusterMaxSize 2]
[dynClusterMinSize 1]]')
```

- Using Jython list:

```
AdminTask.validateEdition (['-appName', 'BeenThere', '-edition', '1.0', '-params',
'{dynClusterMaxSize 2}{dynClusterMinSize 1}'])
```

Note: The following example applies only to Version 6.1.0.5 running with WebSphere Application Server Network Deployment Version 7.0.0.x.

```
AdminTask.validateEdition (['-appName', 'BeenThere', '-edition', '1.0', '-params',
'[[dynClusterMaxSize 2][dynClusterMinSize 1]]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask validateEdition {-interactive}
```

Using Jython string:

```
AdminTask.validateEdition ('[-interactive]')
```

- Using Jython list:

```
AdminTask.validateEdition (['-interactive'])
```

listEditions

The **listEditions** command provides a list of all the installed editions of the application.

Target object

None.

Parameters

-appName

Specifies the name of the application. (String)

Return value

The command returns a string array of all the installed editions of the application.

Note: The return value for base editions is displayed as "".

Batch mode example usage

- Using Jacl:

```
$AdminTask listEditions {-appName HelloWorld}
```

Using Jython string:

```
AdminTask.listEditions ('[-appName HelloWorld]')
```

- Using Jython list:

```
AdminTask.listEditions (['-appName', 'HelloWorld'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listEditions {-interactive}
```

Using Jython string:


```
AdminTask.listEditions (['-interactive'])
```

- Using Jython list:

```
AdminTask.listEditions (['-interactive'])
```

isEditionExists

The **isEditionExists** command verifies that the specified edition exists for the particular application.

Target object

None.

Parameters

-appName

Specifies the name of the application. (String)

-edition

Specifies the name of the application edition. (String)

Return value

The command returns:

- true: success
- false: failure

Batch mode example usage

- Using Jacl:

```
$AdminTask isEditionExists {-appName HelloWorld -edition 2.0}
```

Using Jython string:

```
AdminTask.isEditionExists (['-appName HelloWorld -edition 2.0'])
```

- Using Jython list:

```
AdminTask.isEditionExists (['-appName', 'HelloWorld', '-edition', '2.0'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask isEditionExists {-interactive}
```

Using Jython string:

```
AdminTask.isEditionExists (['-interactive'])
```

- Using Jython list:

```
AdminTask.isEditionExists (['-interactive'])
```

cloneDynamicCluster

The **cloneDynamicCluster** command clones a dynamic cluster to use a different node group.

Target object

None.

Parameters

-clusterName

Specifies the name of the original dynamic cluster. (String)

-newClusterName

Specifies the name of the cloned dynamic cluster. (String)

-nodeGroupName

Specifies the name of the node group in which the cloned dynamic cluster is located. By default, the name is set to the current node group if this parameter is not specified. (String)

-opMode

Specifies the operational mode of the cloned dynamic cluster. By default, the mode of the original dynamic cluster is used if this parameter is not specified. Set the value to automatic, manual, or supervised. (String)

For more information about operational modes, see [Dynamic clusters](#).

-templateName

Specifies the name of an existing template in the format of *cell_name/node_name/server_name*. The template is used to create the server instances of the cloned cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask cloneDynamicCluster {-clusterName SourceCluster -newClusterName newCluster -nodeGroupName CloneNodeGroup -opMode supervised}
```

```
$AdminTask cloneDynamicCluster {-clusterName SourceCluster -newClusterName newCluster -nodeGroupName CloneNodeGroup -opMode supervised -templateName xdCell/xdNode/SourceCluster_member1}
```

Using Jython string:

```
AdminTask.cloneDynamicCluster ('[-clusterName SourceCluster -newClusterName newCluster -nodeGroupName CloneNodeGroup -opMode supervised]')
```

```
AdminTask.cloneDynamicCluster ('[-clusterName SourceCluster -newClusterName newCluster -nodeGroupName CloneNodeGroup -opMode supervised -templateName xdCell/xdNode/SourceCluster_member1]')
```

- Using Jython list:

```
AdminTask.cloneDynamicCluster (['-clusterName', 'SourceCluster', '-newClusterName', 'newCluster', '-nodeGroupName', 'CloneNodeGroup', '-opMode', 'supervised'])
```

```
AdminTask.cloneDynamicCluster (['-clusterName', 'SourceCluster', '-newClusterName', 'newCluster', '-nodeGroupName', 'CloneNodeGroup', '-opMode', 'supervised', '-templateName', 'xdCell/xdNode/SourceCluster_member1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask cloneDynamicCluster {-interactive}
```

Using Jython string:

```
AdminTask.cloneDynamicCluster ('[-interactive]')
```

- Using Jython list:

```
AdminTask.cloneDynamicCluster (['-interactive'])
```

cancelValidation

The **cancelValidation** command cancels the validation mode on an application edition.

The `getEditionState` command returns the state of the application edition. The state of an existing edition can be either ACTIVE, INACTIVE or VALIDATE. For a non-existent application or edition the method returns null.

Target object

None.

Parameters

-appName

Specifies the name of the application. (String)

-edition

Specifies the name of the application edition. (String)

Return value

The command returns:

- **true**: The cancellation of the validation succeeded.
- **false**: The cancelation of the validation failed.

Batch mode example usage

- Using Jacl:

```
$AdminTask cancelValidation {-appName BeenThere -edition 1.0}
```

Using Jython string:

```
AdminTask.cancelValidation (['-appName BeenThere -edition 1.0'])
```

- Using Jython list:

```
AdminTask.cancelValidation (['-appName', 'BeenThere', '-edition', '1.0'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask cancelValidation {-interactive}
```

Using Jython string:

```
AdminTask.cancelValidation (['-interactive'])
```

- Using Jython list:

```
AdminTask.cancelValidation (['-interactive'])
```

getEditionState

The **getEditionState** command returns the state of the application edition. The state of an existing edition can be either ACTIVE, INACTIVE or VALIDATE. For a non-existent application or edition, the method returns null.

Target object

None.

Parameters**-appName**

Specifies the name of the application. (String)

-edition

Specifies the name of the application edition. (String)

Return value

The command returns:

- **ACTIVE**
- **INACTIVE**
- **VALIDATE**

Batch mode example usage

- Using Jacl:

```
$AdminTask getEditionState {-appName BeenThere -edition 1.0}
```

Using Jython string:

```
AdminTask.getEditionState ('[-appName BeenThere -edition 1.0]')
```

- Using Jython list:

```
AdminTask.getEditionState (['-appName', 'BeenThere', '-edition', '1.0'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getEditionState {-interactive}
```

Using Jython string:

```
AdminTask.getEditionState ('[-interactive]')
```

- Using Jython list:

```
AdminTask.getEditionState (['-interactive'])
```

Install an application with an edition

Use the AdminApp **install** command to install an application with an edition. You must pass the **-edition** parameter to specify the edition identifier.

You can use the following example in wsadmin to install the application BeenThere with edition 1.0.

- Using Jacl:

```
$AdminApp install /tmp/BeenThere1.0.ear {-appname BeenThere -edition 1.0 -nopreCompileJSPs -  
-distributeApp -noseMetaDataFromBinary -nodeployejb -createMBeansForResources -noreloadEnabled  
-nodeployws -validateinstall warn -noprocessEmbeddedConfig -filepermission  
.*\.dll=755#.*\so=755#.*\a=755#.*\sl=755 -noallowDispatchRemoteInclude -  
noallowServiceRemoteInclude -MapModulesToServers {"BeenThere WAR" BeenThere.war,WEB-  
INF/web.xml WebSphere:cell=TestCell,cluster=TestClusterB} {"BeenThere EJB" BeenThere.jar,META-  
INF/ejb-jar.xml WebSphere:cell=TestCell,cluster=TestClusterB}}
```

- Using Jython:

```
AdminApp.install("/tmp/BeenThere1.0.ear", ['-appname BeenThere -edition 1.0 -nopreCompileJSPs  
-distributeApp -noseMetaDataFromBinary -nodeployejb -createMBeansForResources -  
noreloadEnabled -nodeployws -validateinstall off -processEmbeddedConfig -filepermission  
.*\.dll=755#.*\so=755#.*\a=755#.*\sl=755 -buildVersion Unknown -  
noallowDispatchRemoteInclude -noallowServiceRemoteInclude -MapModulesToServers [{"BeenThere  
WAR" BeenThere.war,WEB-INF/web.xml WebSphere:cell=TestCell,cluster=TestClusterB} [{"BeenThere  
EJB" BeenThere.jar,META-INF/ejb-jar.xml WebSphere:cell=TestCell,cluster=TestClusterB}]])
```

Update an application with an edition

Use the AdminApp **update** command to update an application with an edition. Pass the full application name, which is the name of the application and the edition, to specify the edition.

You can use the following example in wsadmin to update edition 1.0 of the application BeenThere with the BeenThereE1-update.ear file.

- Using Jacl:

```
$AdminApp update BeenThere-edition1.0 app {-operation update -contents /tmp/BeenThereE1-  
update.ear -nopreCompileJSPs -distributeApp -noseMetaDataFromBinary -nodeployejb -  
createMBeansForResources -noreloadEnabled -nodeployws -validateinstall warn -  
noprocessEmbeddedConfig -filepermission .*\.dll=755#.*\so=755#.*\a=755#.*\sl=755 -  
noallowDispatchRemoteInclude -noallowServiceRemoteInclude}
```

- Using Jython:

```
AdminApp.update('BeenThere-edition1.0', 'app', ['-operation update -contents /tmp/BeenThereE1-  
update.ear -nopreCompileJSPs -distributeApp -noseMetaDataFromBinary -nodeployejb -  
createMBeansForResources -noreloadEnabled -nodeployws -validateinstall warn -  
noprocessEmbeddedConfig -filepermission .*\.dll=755#.*\so=755#.*\a=755#.*\sl=755 -  
noallowDispatchRemoteInclude -noallowServiceRemoteInclude]')
```

Edit an application with an edition

Use the AdminApp **edit** command to edit an application with an edition. Pass the full application name, which is the name of the application and the edition, to specify the edition.

You can use the following example in wsadmin to edit edition 1.0 of the application BeenThere by changing the context root of the Web module to /beenthere.

- Using Jacl:

```
$AdminApp edit BeenThere-edition1.0 {-CtxRootForWebMod {"BeenThere WAR" BeenThere.war,WEB-INF/web.xml /beenthere}}
```

- Using Jython:

```
AdminApp.edit('BeenThere-edition1.0', '[-CtxRootForWebMod [{"BeenThere WAR" BeenThere.war,WEB-INF/web.xml /beenthere}]]')
```

Uninstall an application with an edition

Use the AdminApp **uninstall** command to uninstall an application with an edition. Pass the full application name, which is the name of the application and the edition, to specify the edition.

You can use the following example in wsadmin to uninstall edition 1.0 of the application BeenThere.

- Using Jacl:

```
$AdminApp uninstall BeenThere-edition1.0
```

- Using Jython:

```
AdminApp.uninstall('BeenThere-edition1.0')
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Application edition manager](#)

Related tasks:

[Deploying and managing application editions without loss of service](#)

[Performing a rollout on an edition](#)

[Canceling an application validation](#)

[Installing an application edition](#)

Centralized installation manager administrative tasks

You can use the Jacl or Jython scripting languages to use the various features of the centralized installation manager with the wsadmin tool. Use the commands and parameters to install, uninstall, and manage various software packages and maintenance files.

The administrative tasks for the centralized installation manager include the following commands:

- [installWASExtension](#)
- [installSoftware](#)
- [installMaintenance](#)
- [listPackagesForInstall](#)
- [listFeaturesForInstall](#)
- [showPackageInfo](#)
- [showLicenseAgreement](#)
- [getManagedNodesOnHostByInstallLoc](#)
- [listManagedNodesOnHost](#)
- [testConnectionToHost](#)
- [testConnectionToHostUsingSSHKey](#)
- [installSSHPublicKeyOnHost](#)
- [listKeyInstallationRecords](#)
- [updateKeyInstallationRecords](#)
- [listPendingRequests](#)
- [listInProgressRequests](#)
- [listRequestsForTarget](#)
- [showLatestInstallStatus](#)
- [uninstallSoftware](#)
- [uninstallMaintenance](#)

installWASExtension

The installWASExtension command installs the specified WebSphere® Application Server extension package on a specified host that contains one or more WebSphere Application Server Network Deployment nodes. The nodes must be defined and part of the WebSphere Application Server Network Deployment cell.

Target object

None.

Required parameters

- **-packageName**: Specifies the name of the software package. (String, required)
- **-hostName**: Specifies the name of the remote host. (String, required)
- **-augment**: Specifies a list of nodes to augment. Valid nodes are those defined on the host under the same installation location for WebSphere Application Server. Specify ALL_NODES to augment all of the nodes defined for the same installation location. (String, required)
- **-adminName**: Specifies the administrative ID for the remote host. (String, required)
- **-acceptLicense**: Specifies if the license agreement is accepted. Specify true to indicate that you reviewed and agreed to the terms of the IBM® International Program License Agreement accompanying this program. Otherwise, you cannot proceed with the installation of the program or component. (String, required)

Optional parameters

- **-installLocation**: Specifies the path of the installation directory on the remote host. Specify this parameter only if there are multiple installation locations that exist within the current cell on the same host. (String, optional)
- **-featureList**: Specifies a list of features to install on the remote target. (String, optional)
- **-adminPassword**: Specifies the administrative password for the remote host. Specify either the **-adminPassword** parameter or the **-privateKeyStore** parameter to authenticate. (String, optional)
- **-privateKeyStore**: Specifies the path to the private key file, which is located on the deployment manager. Specify either the **-adminPassword** parameter or the **-privateKeyStore** parameter to authenticate. (String, optional)
- **-keyStorePassword**: Specifies an optional password, which is also known as the pass phrase, that is used to protect the private key file. (String. Required if a non-blank password is used to protect the key file.)
- **-specialParms**: Specifies optional name-value pairs for other parameters that might be required. Obtain information about any name-value pairs from the provider of the software package. You can also use the **showPackageInfo** command to gather this information. (String, optional)
- **-tempDir**: Specifies the location of the temporary directory on the target host. If this parameter is omitted, the centralized installation manager uses the default temporary directory of the target host. (String, optional)

Batch mode example usage

- Using Jacl:


```
$AdminTask installWASExtension {-packageName XD0ps -hostName river.com -augment ALL_NODES -adminName admin1 -adminPassword passwd1 -acceptLicense true}
```

- Using Jython string:

```
AdminTask.installWASExtension ('[-packageName XD0ps -hostName river.com -augment ALL_NODES -adminName admin1 -adminPassword passwd1 -acceptLicense true]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask installWASExtension {-interactive}
```

- Using Jython string:

```
AdminTask.installWASExtension ('[-interactive]')
```

installSoftware

The installSoftware command installs the specified software package on the target host.

Target object

None.

Required parameters

- **-packageName**: Specifies the name of the software package. (String, required)
- **-hostName**: Specifies the name of the remote host. (String, required)
- **-platformType**: Specifies the operating system of the remote workstation. The valid types are: Windows, AIX®, HP-UX, Linux, UNIX, or Solaris. This parameter is not case-sensitive. (String, required)
- **-installLocation**: Specifies the path to the installation directory on the remote host. Specify this parameter only if there are multiple installation locations that exist within the current cell on the same host. (String, required)
- **-featureList**: Specifies features of WebSphere Application Server Community Edition Version 2.0 to install on the remote target. (String, required)

Specify **-ServerAndSDK** to install WebSphere Application Server Community Edition and the IBM SDK together.

Specify **-ServerOnly** to install WebSphere Application Server Community Edition without installing the IBM SDK.

Optional: You can specify the directory that contains a supported Java™ software development kit (SDK) by issuing **-JAVA_INSTALL_DIR** with the **-specialParms** parameter. If you do not specify **-JAVA_INSTALL_DIR**, an attempt is made to locate an IBM SDK on the workstation by default. The request to install WebSphere Application Server Community Edition fails if the centralized installation manager cannot locate a SDK.

- **-adminName**: Specifies the administrative ID for the remote host. (String, required)
- **-acceptLicense**: Specifies if the license agreement is accepted. Specify true to indicate that you reviewed and agreed to the terms of the IBM International Program License Agreement accompanying this program. Otherwise, you cannot proceed with the installation of the program or component. (String, required)

Optional parameters

- **-adminPassword**: Specifies the administrative password for the remote host. Specify either the **-adminPassword** parameter or the **-privateKeyStore** parameter to authenticate. (String, optional)
- **-privateKeyStore**: Specifies the path to the private key file, which is located on the deployment manager. Specify either the **-adminPassword** parameter or the **-privateKeyStore** parameter to authenticate. (String, optional)
- **-keyStorePassword**: Specifies an optional password, which is also known as the pass phrase, that is used to protect the private key file. (String. Required if a non-blank password is used to protect the key file.)
- **-specialParms**: Specifies optional name-value pairs for other parameters that might be required. Obtain information about any name-value pairs from the provider of the software package. You can also use the **showPackageInfo** command to gather this information. (String, optional)

Optionally, you can specify the following with the **-specialParms** parameter when you install WebSphere Application Server Community Edition Version 2.0:

- **DISABLE_OS_PREREQ_CHECKING**: Set the parameter to true to disable prerequisite checking on the operating system. Set the parameter to false to enable prerequisite checking on the operating system.
- **NODE_TO_MANAGE_WASCE**: Issue this parameter to designate a node on the target host as the manager of WebSphere Application Server Community Edition. This parameter is required if the

target host contains multiple nodes that are augmented for WebSphere Virtual Enterprise.

By default, the centralized installation manager selects the appropriate version of the installation package to install based on the version of the operating system. For example, if both 32-bit and 64-bit binaries are included for a certain operating system and processor architecture, the centralized installation manager installs the 64-bit installation binaries on 64-bit operating systems. To override the default behavior and use the 32-bit installation binaries on 64-bit operating systems, set the **USE_32BIT_IMAGE_ON_64BIT_OS** parameter to true.

- **-tempDir**: Specifies the location of the temporary directory on the target host. If this parameter is omitted, the centralized installation manager uses the default temporary directory of the target host. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask installSoftware {-packageName WASCE20 -featureList ServerAndSDK -hostName abc.com -platformType windows -installLocation "D:/WASCE20/CE_22409" -adminName admin1 -adminPassword passwd1 -acceptLicense true -specialParms "{NODE_TO_MANAGE_WASCE abcNode01}{DISABLE_OS_PREREQ_CHECKING true}"}
```

```
$AdminTask installSoftware {-packageName XDAGent -hostName abc.com -platformType aix -installLocation C:/WXD61/Agent -adminName admin1 -adminPassword passwd1 -acceptLicense true -specialParms "{DISABLE_OS_PREREQ_CHECKING true}{USE_32BIT_IMAGE_ON_64BIT_OS true}"}
```

- Using Jython string:

```
AdminTask.installSoftware ('[-packageName WASCE20 -featureList ServerAndSDK -hostName abc.com -platformType windows -installLocation "D:/WASCE20/CE_22409" -adminName admin1 -adminPassword passwd1 -acceptLicense true -specialParms "[NODE_TO_MANAGE_WASCE abcNode1][DISABLE_OS_PREREQ_CHECKING true]"'])
```

```
AdminTask.installSoftware ('[-packageName XDAGent -hostName abc.com -platformType aix -installLocation C:/WXD61/Agent -adminName admin1 -adminPassword passwd1 -acceptLicense true -specialParms "[DISABLE_OS_PREREQ_CHECKING true][USE_32BIT_IMAGE_ON_64BIT_OS true]"'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask installSoftware {-interactive}
```

- Using Jython string:

```
AdminTask.installSoftware ('[-interactive]')
```

installMaintenance

The installMaintenance command installs maintenance on the target host.

Target object

None.

Required parameters

- **-packageName**: Specifies the name of the software package. (String, required)
- **-hostName**: Specifies the name of the remote host. (String, required)
- **-adminName**: Specifies the administrative ID for the remote host. (String, required)
- **-acceptLicense**: Specifies if the license agreement is accepted. Specify true to indicate that you reviewed and agreed to the terms of the IBM International Program License Agreement accompanying this program. Otherwise, you cannot proceed with the installation of the program or component. (String, required)

Optional parameters

- **-fileList**: Specifies a list of files to install on the remote target. This parameter is ignored if you install a predefined maintenance package. (String, optional)
- **-installLocation**: Specifies the path of the installation directory in which to install the package on the remote host. Specify this parameter only if there are multiple installation locations that exist within the current cell on the same host. (String, optional)
- **-adminPassword**: Specifies the administrative password for the remote host. Specify either the **-adminPassword** parameter or the **-privateKeyStore** parameter to authenticate. (String, optional)
- **-privateKeyStore**: Specifies the path to the private key file, which is located on the deployment manager. Specify either the **-adminPassword** parameter or the **-privateKeyStore** parameter to

authenticate. (String, optional)

- **-keyStorePassword**: Specifies an optional password, which is also known as the pass phrase, that is used to protect the private key file. (String. Required if a non-blank password is used to protect the key file.)
- **-tempDir**: Specifies the location of the temporary directory on the target host. If this parameter is omitted, the centralized installation manager uses the default temporary directory of the target host. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask installMaintenance {-packageName ND61Maintenance -fileList "sample_file01, sample_file02" -hostName river.com -installLocation D:/WAS61 -adminName admin1 -adminPassword passw0rd1 -acceptLicense true}
```

- Using Jython string:

```
AdminTask.installMaintenance ('[-packageName ND61Maintenance -fileList "sample_file01.pak, sample_file02.pak" -hostName river.com -installLocation D:/WAS61 -adminName admin1 -adminPassword passw0rd1 -acceptLicense true]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask installMaintenance {-interactive}
```

- Using Jython string:

```
AdminTask.installMaintenance ('[-interactive]')
```

listPackagesForInstall

The listPackagesForInstall command lists all of the software packages that you can use the centralized installation manager to install.

Target object

None.

Required parameters

None.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask listPackagesForInstall
```

- Using Jython string:

```
AdminTask.listPackagesForInstall ()
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listPackagesForInstall {-interactive}
```

- Using Jython string:

```
AdminTask.listPackagesForInstall ('[-interactive]')
```

listFeaturesForInstall

The listFeaturesForInstall command lists the available features of a software package that you can use the

centralized installation manager to install.

Target object

None.

Required parameters

- **-packageName**: Specifies the name of the software package. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask listFeaturesForInstall {-packageName sample_package}
```

- Using Jython string:

```
AdminTask.listFeaturesForInstall ('[-packageName sample_package]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listFeaturesForInstall {-interactive}
```

- Using Jython string:

```
AdminTask.listFeaturesForInstall ('[-interactive]')
```

showPackageInfo

The showPackageInfo command displays general information about a specific software package.

Target object

None.

Required parameters

- **-packageName**: Specifies the name of the software package. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask showPackageInfo {-packageName sample_package}
```

- Using Jython string:

```
AdminTask.showPackageInfo ('[-packageName sample_package]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask showPackageInfo {-interactive}
```

- Using Jython string:

```
AdminTask.showPackageInfo ('[-interactive]')
```

showLicenseAgreement

The showLicenseAgreement command displays the license agreement associated with the specified installation

package.

Target object

None.

Required parameters

- **-packageName**: Specifies the name of the software package. (String, required)

Optional parameters

- **-showLicenseInfoOnly**: Specifies that only the content of the license file is shown. The default is false. (String, required)

Batch mode example usage

- Using Jacl:

```
$AdminTask showLicenseAgreement {-packageName sample_package}
```

- Using Jython string:

```
AdminTask.showLicenseAgreement ('[-packageName sample_package]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask showLicenseAgreement {-interactive}
```

- Using Jython string:

```
AdminTask.showLicenseAgreement ('[-interactive]')
```

getManagedNodesOnHostByInstallLoc

The `getManagedNodesOnHostByInstallLoc` command returns the names of the managed nodes that are defined in the current deployment manager cell. Issue this command when a host contains multiple installations of WebSphere Application Server Network Deployment with nodes that are federated into the same cell.

Target object

The required target object is the host name of the workstation containing the managed nodes that are federated into the current deployment manager cell.

Required parameters

None.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask getManagedNodesOnHostByInstallLoc host_name
```

- Using Jython string:

```
AdminTask.getManagedNodesOnHostByInstallLoc ('host_name')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getManagedNodesOnHostByInstallLoc {-interactive}
```

- Using Jython string:

```
AdminTask.getManagedNodesOnHostByInstallLoc ('[-interactive]')
```

listManagedNodesOnHost

The listManagedNodesOnHost command list the managed nodes that are located on the federated host in the current deployment manager cell.

Target object

The required target object specifies the host name of the workstation containing the managed nodes that are federated into the deployment manager cell.

Required parameters

None.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask listManagedNodesOnHost host_name
```

- Using Jython string:

```
AdminTask.listManagedNodesOnHost ('host_name')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listManagedNodesOnHost {-interactive}
```

- Using Jython string:

```
AdminTask.listManagedNodesOnHost ('[-interactive]')
```

testConnectionToHost

The testConnectionToHost command verifies that a connection can be established from the deployment manager to the remote host by using an administrator ID and password for the remote host.

Target object

None.

Required parameters

- **-hostName: Specifies the name of the remote host. (String, required)**
- **-platformType: Specifies the operating system of the remote workstation. The valid types are: Windows, AIX, HP-UX, Linux, UNIX, or Solaris. This parameter is not case-sensitive. (String, required)**
- **-adminName: Specifies the administrative ID for the remote host. (String, required)**
- **-adminPassword: Specifies the administrative password for the remote host. (String, required)**

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask testConnectionToHost {-hostName big.mountain.com -platformType linux -adminName root -adminPassword passw0rd3}
```

- Using Jython string:

```
AdminTask.testConnectionToHost ('[-hostName big.mountain.com -platformType linux -adminName
```



```
root -adminPassword passw0rd3]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask testConnectionToHost {-interactive}
```

- Using Jython string:

```
AdminTask.testConnectionToHost ('[-interactive]')
```

testConnectionToHostUsingSSHKey

The testConnectionToHostUsingSSHKey command verifies that a connection can be established from the deployment manager to the remote host by using the Secure Shell (SSH) private key for the remote host.

Target object

None.

Required parameters

- **-hostName:** Specifies the name of the remote host. (String, required)
- **-adminName:** Specifies the administrative ID for the remote host. (String, required)
- **-privateKeyStore:** Specifies the path to the private key file, which is located on the deployment manager. (String, required)
- **-keyStorePassword:** Specifies the password, which is also known as the pass phrase, that is used to protect the private key file. (String. Required if a non-blank password is used to protect the key file.)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask testConnectionToHostUsingSSHKey {-hostName abc.com -adminName root -privateKeyStore /root/.ssh/id_rsa}
```

- Using Jython string:

```
AdminTask.testConnectionToHostUsingSSHKey ('[-hostName abc.com -adminName root -privateKeyStore /root/.ssh/id_rsa]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask testConnectionToHostUsingSSHKey {-interactive}
```

- Using Jython string:

```
AdminTask.testConnectionToHostUsingSSHKey ('[-interactive]')
```

installSSHPublicKeyOnHost

The installSSHPublicKeyOnHost command installs the administrative Secure Shell (SSH) public key on the remote host.

Target object

None.

Required parameters

- **-hostName:** Specifies the name of the remote host. (String, required)
- **-adminName:** Specifies the administrative ID for the remote host. (String, required)
- **-adminPassword:** Specifies the administrative password for the remote host. (String,

required)

- **-publicKeyStore:** Specifies the path to the public key file, which is located on the deployment manager, in either Internet Engineering Task Force (IETF) standard format or OpenSSH format.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask installSSHPublicKeyOnHost {-hostName abc.com -adminName root -adminPassword  
passw0rd3 -publicKeyStore /root/.ssh/id_rsa.pub}
```

- Using Jython string:

```
AdminTask.installSSHPublicKeyOnHost ('[-hostName abc.com -adminName root -adminPassword  
passw0rd3 -publicKeyStore /root/.ssh/id_rsa.pub]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask installSSHPublicKeyOnHost {-interactive}
```

- Using Jython string:

```
AdminTask.installSSHPublicKeyOnHost ('[-interactive]')
```

listKeyInstallationRecords

The listKeyInstallationRecords command lists the SSH public key installation records that the centralized installation manager maintains.

Target object

None.

Required parameters

None.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask listKeyInstallationRecords
```

- Using Jython string:

```
AdminTask.listKeyInstallationRecords ()
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listKeyInstallationRecords {-interactive}
```

- Using Jython string:

```
AdminTask.listKeyInstallationRecords ('[-interactive]')
```

updateKeyInstallationRecords

The `updateKeyInstallationRecords` command updates the SSH public key installation records that the centralized installation manager maintains.

Target object

None.

Required parameters

None.

Optional parameters

- **-add: Adds a list of host names to the installation records. (String, optional)**
- **-remove: Removes a list of host names from the installation records. (String, optional)**

Batch mode example usage

- Using Jacl:

```
$AdminTask updateKeyInstallationRecords {-add "abc.com,river.com"}
```

- Using Jython string:

```
AdminTask.updateKeyInstallationRecords ('[-add "abc.com,river.com"]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask updateKeyInstallationRecords {-interactive}
```

- Using Jython string:

```
AdminTask.updateKeyInstallationRecords ('[-interactive]')
```

listPendingRequests

The `listPendingRequests` command lists the submitted installation or uninstallation requests that are not started

Target object

None.

Required parameters

None.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask listPendingRequests
```

- Using Jython string:

```
AdminTask.listPendingRequests ()
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listPendingRequests {-interactive}
```

- Using Jython string:

```
AdminTask.listPendingRequests ('[-interactive]')
```

listInProgressRequests

The listInProgressRequests command lists the installation or uninstallation requests that are in progress for completion.

Target object

None.

Required parameters

None.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask listInProgressRequests
```

- Using Jython string:

```
AdminTask.listInProgressRequests ()
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listInProgressRequests {-interactive}
```

- Using Jython string:

```
AdminTask.listInProgressRequests ('[-interactive]')
```

listRequestsForTarget

The listRequestForTarget command lists all of the submitted installation and uninstallation requests for a specific host.

Target object

The required target object is the host name of the target workstation. You must specify the same host name that you specify with the **installSoftware** and **uninstallSoftware** commands.

Required parameters

None.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask listRequestsForTarget host_name
```

- Using Jython string:

```
AdminTask.listRequestsForTarget ('host_name')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listRequestsForTarget {-interactive}
```

- Using Jython string:

```
AdminTask.listRequestsForTarget ( '[-interactive]'
```

showLatestInstallStatus

The showLatestInstallStatus command lists all of the submitted installation requests for a specific host.

Target object

The required target object is the host name of the target workstation. You must specify the same host name that you specify with the **installSoftware** command.

Required parameters

None.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask showLatestInstallStatus host_name
```

- Using Jython string:

```
AdminTask.showLatestInstallStatus ('host_name')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask showLatestInstallStatus {-interactive}
```

- Using Jython string:

```
AdminTask.showLatestInstallStatus ( '[-interactive]'
```

uninstallSoftware

The uninstallSoftware command uninstalls the software package from the remote host.

Target object

None.

Required parameters

- **-packageName:** Specifies the name of the software package. (String, required)
- **-hostName:** Specifies the name of the remote host. (String, required)
- **-platformType:** Specifies the operating system of the remote workstation. The valid types are: Windows, AIX, HP-UX, Linux, UNIX, or Solaris. This parameter is not case-sensitive. (String, required)
- **-installLocation:** Specifies the path to the installation directory on the remote host. Specify this parameter only if there are multiple installation locations that exist within the current cell on the same host. (String, required)
- **-adminName:** Specifies the administrative ID for the remote host. (String, required)

Optional parameters

- **-adminPassword:** Specifies the administrative password for the remote host. Specify either the **-adminPassword** parameter or the **-privateKeyStore** parameter to authenticate. (String, optional)
- **-privateKeyStore:** Specifies the path to the private key file, which is located on the deployment manager. Specify either the **-adminPassword** parameter or the **-privateKeyStore** parameter to authenticate. (String, optional)
- **-keyStorePassword:** Specifies an optional password, which is also known as the pass phrase, that is used to protect the private key file. (String. Required if a non-blank

password is used to protect the key file.)

Batch mode example usage

- Using Jacl:

```
$AdminTask uninstallSoftware {-packageName XDAGent -hostName abc.com -platformType windows -installLocation C:/WXD61/Agent -adminName admin1 -adminPassword passw0rd1}
```

- Using Jython string:

```
AdminTask.uninstallSoftware ('[-packageName XDAGent -hostName abc.com -platformType windows -installLocation C:/WXD61/Agent -adminName admin1 -adminPassword passw0rd1]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask uninstallSoftware {-interactive}
```

- Using Jython string:

```
AdminTask.uninstallSoftware ('[-interactive]')
```

uninstallMaintenance

The `uninstallMaintenance` command uninstalls maintenance, such as fix packs and interim fixes, from the remote host.

Target object

None.

Required parameters

- **-packageName:** Specifies the name of the software package. (String, required)
- **-hostName:** Specifies the name of the remote host. (String, required)
- **-adminName:** Specifies the administrative ID for the remote host. (String, required)

Optional parameters

- **-fileList:** Specifies a list of maintenance files to uninstall on the remote target. (String, optional)
- **-installLocation:** Specifies the path to the installation directory on the remote host. Specify this parameter only if there are multiple installation locations that exist within the current cell on the same host. (String, optional)
- **-adminPassword:** Specifies the administrative password for the remote host. Specify either the `-adminPassword` parameter or the `-privateKeyStore` parameter to authenticate. (String, optional)
- **-privateKeyStore:** Specifies the path to the private key file, which is located on the deployment manager. Specify either the `-adminPassword` parameter or the `-privateKeyStore` parameter to authenticate. (String, optional)
- **-keyStorePassword:** Specifies an optional password, which is also known as the pass phrase, that is used to protect the private key file. (String. Required if a non-blank password is used to protect the key file.)

Batch mode example usage

- Using Jacl:

```
$AdminTask uninstallMaintenance {-packageName XD61Maintenance -hostName river.com -adminName admin1 -adminPassword passw0rd1 -fileList "6.1.0.0-WS-WXDOP-IFPKxxxxx.pak,6.1.0.0-WS-WXDOP-IFPKyyyyy.pak"}
```

- Using Jython string:

```
AdminTask.uninstallMaintenance ('[-packageName XD61Maintenance -hostName river.com -adminName admin1 -adminPassword passw0rd1 -fileList "6.1.0.1-WS-WXDOP-IFPKxxxxx.pak,6.1.0.1-WS-WXDOP-IFPKyyyyy.pak"]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask uninstallMaintenance {-interactive}
```

- Using Jython string:

```
AdminTask.uninstallMaintenance ('[-interactive]')
```

showLatestUninstallStatus

The showLatestUninstallStatus command displays the status of the most recently submitted uninstallation request.

Target object

The required target object is the host name of the target workstation. You must specify the same host name that you specify with the **uninstallSoftware** command.

Required parameters

None.

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask showLatestUninstallStatus host_name
```

- Using Jython string:

```
AdminTask.showLatestUninstallStatus ('host_name')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask showLatestUninstallStatus {-interactive}
```

- Using Jython string:

```
AdminTask.showLatestUninstallStatus ('[-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related tasks:

[Using the centralized installation manager to manage your environment](#)
[Using Remote Execution and Access to access remote installation targets](#)

Dynamic cluster administrative tasks

Use the dynamic cluster commands to view or edit the dynamic clusters, without using the administrative console.

Use the following commands to create dynamic clusters, add or remove servers from clusters, list dynamic clusters, or modify dynamic clusters:

- [createDynamicCluster](#)
- [createPHPDynamicCluster](#)
- [createDynamicClusterFromStaticCluster](#)
- [createDynamicClusterFromForeignServers](#)
- [createODRDynamicCluster](#)
- [addForeignServersToDynamicCluster](#)
- [removeForeignServersFromDynamicCluster](#)
- [deleteDynamicCluster](#)
- [getDynamicClusterMembers](#)
- [getDynamicClusterOperationalMode](#)
- [getDynamicClusterMembershipPolicy](#)
- [getDynamicClusterServerType](#)
- [getDynamicClusterMinInstances](#)
- [getDynamicClusterMaxInstances](#)
- [getDynamicClusterVerticalInstances](#)
- [getDynamicClusterIsolationProperties](#)
- [listDynamicClusters](#)
- [listDynamicClusterIsolationGroups](#)
- [listDynamicClusterIsolationGroupMembers](#)
- [modifyDynamicClusterIsolationProperties](#)
- [setDynamicClusterOperationalMode](#)
- [setDynamicClusterMembershipPolicy](#)
- [setDynamicClusterMinInstances](#)
- [setDynamicClusterMaxInstances](#)
- [setDynamicClusterVerticalInstances](#)
- [testDynamicClusterMembershipPolicy](#)
- [createNonWASDynamicCluster](#)
- [Edit the dynamic cluster server template](#)

createDynamicCluster

The `createDynamicCluster` command creates a new dynamic cluster.

Required parameters

- `-membershipPolicy`: Specifies the membership policy. (String, required)

Optional parameters

- `-dynamicClusterProperties`: Specifies the dynamic cluster properties. (String, optional)
You can specify the following options:
 - `operationalMode`: Specifies the operational mode. The valid values are: `manual`, `supervised`, and `automatic`.
 - `minInstances`: Specifies the minimum number of cluster instances as an integer.
 - `serverInactivityTime`: Specifies the time to wait before stopping instances, in minutes. This value is valid only when the `minInstances` parameter value is set to 0.
 - `maxInstances`: Specifies the maximum number of cluster instances as an integer. If you do not want a limit to the number of cluster instances that can start, specify `-1`.
 - `numVerticalInstances`: Specifies the number of instances that can start on the same node by creating vertical instances. Specify an integer value.
- `-clusterProperties`: Specifies the cluster properties. (String, optional) You can specify the following options:
 - `preferLocal`: Specifies whether enterprise bean requests are routed to the node on which the client resides when possible. The valid values are `true` or `false`.
 - `createDomain`: Specifies that a replication domain is created for this cluster. Replication domains can be created only for WebSphere® Application Server application servers. The valid values are `true` or `false`.
 - `coreGroup`: Specifies the name of the core group for the dynamic cluster. You only need to specify the core group for WebSphere Application Server application servers. The value can be `DefaultCoreGroup` or the core group name.
 - `templateName`: Specifies the name of the server template. The valid input includes the following values:
 - For WebSphere Application Server dynamic clusters: `defaultXD`, `defaultXDZ0S`, `cell_name/node_name/was_server_name`
 - `z/OS` `clusterShortName`: For z/OS® platforms, specifies the short name of the cluster. After the dynamic cluster has been created, you can use the `AdminTask.changeClusterShortName` command to update this field.

- **z/OS** `serverSpecificShortNames`: For z/OS platforms, specifies the specific short name of cluster member in a comma-separated list format, for example: SSN1,SSN2. Use commas to separate multiple short names. If you do not provide enough short names to be used for all of the cluster members, then the remaining cluster members are assigned generated generic short names, such as BBOS001, BBOS002, and so on. After the dynamic cluster has been created, you can update this list with the `serverSpecificShortNames` custom property. This property is set on the dynamic cluster. If you want to change this property with scripting after the dynamic cluster has been created, you can use the `AdminTask changeServerSpecificShortName` command.

Return value

The command returns `javax.management.ObjectName` value of the dynamic cluster that you created.

Batch mode example usage

- Using Jacl:

```
$AdminTask createDynamicCluster dcName {-membershipPolicy node_nodegroup='ngName'}
```

or

```
$AdminTask createDynamicCluster dcName {-membershipPolicy "node_nodegroup = 'ngName'" -
dynamicClusterProperties "{operationalMode automatic}{minInstances 0}{maxInstances -1}
{numVerticalInstances 1}{serverInactivityTime 1440}" -clusterProperties "{preferLocal false}
{createDomain false}{templateName defaultXD}{coreGroup DefaultCoreGroup}{clusterShortName
BBOC001}{serverSpecificShortNames BBOS001, BBOS002}"}
```

- Using Jython string:

```
AdminTask.createDynamicCluster(dcName, ['-membershipPolicy "node_nodegroup = \'ngName\'" -
dynamicClusterProperties "{operationalMode automatic}{minInstances 0}{maxInstances -1}
{numVerticalInstances 1}{serverInactivityTime 1440}" -clusterProperties "{preferLocal false}
{createDomain false}{templateName defaultXD}{coreGroup DefaultCoreGroup}{clusterShortName
BBOC001}{serverSpecificShortNames BBOS001, BBOS002}"'])
```

or, if you are running on WebSphere Application Server Version 7.0 or later:

```
AdminTask.createDynamicCluster('dcName', ['-membershipPolicy "node_nodegroup = \'ngName\'" -
dynamicClusterProperties "[operationalMode automatic][minInstances 0][maxInstances -1]
[numVerticalInstances 1][serverInactivityTime 1440]" -clusterProperties "[preferLocal false]
[createDomain false][templateName defaultXD][coreGroup DefaultCoreGroup][clusterShortName
BBOC001][serverSpecificShortNames BBOS001, BBOS002]"'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createDynamicCluster {-interactive}
```

- Using Jython string:

```
AdminTask.createDynamicCluster ('[-interactive]')
```

createPHPDynamicCluster

The `createPHPDynamicCluster` command creates a new PHP dynamic cluster.

Required parameters

- `-membershipPolicy`: Specifies the membership policy subexpression. (String, required)

Optional parameters

- `-dynamicClusterProperties`: Specifies the dynamic cluster properties. (String, optional)
You can specify the following options:
 - `operationalMode`: Specifies the operational mode. The valid values are: `manual`, `supervised`, and `automatic`.
 - `minInstances`: Specifies the minimum number of cluster instances as an integer.
 - `serverInactivityTime`: Specifies the time to wait before stopping instances, in minutes. This value is valid only when the `minInstances` parameter value is set to 0.
 - `maxInstances`: Specifies the maximum number of cluster instances as an integer. If

- **numVerticalInstances:** Specifies the number of instances that can start on the same node by creating vertical instances. Specify an integer value.
- **-clusterProperties:** Specifies the name of the server template. The valid input includes the `templateName` value:
 - **For PHP dynamic clusters:** APACHE13_PHP4, APACHE13_PHP5, APACHE20_PHP4, APACHE20_PHP5, APACHE22_PHP4, APACHE22_PHP5, `cellName/nodeName/php_serverName`.

Return value

The command returns `javax.management.ObjectName` value of the dynamic cluster that you created.

Batch mode example usage

- Using Jacl:

```
$AdminTask createPHPDynamicCluster dcName {-membershipPolicy "node_property$APACHE_2_0 IS NOT NULL AND node_property$PHP_5 IS NOT NULL"}
```

or

```
$AdminTask createPHPDynamicCluster dcName {-membershipPolicy "node_property$APACHE_2_0 IS NOT NULL AND node_property$PHP_5 IS NOT NULL" -dynamicClusterProperties "{operationalMode automatic}{minInstances 1}{maxInstances -1}{numVerticalInstances 1}" -clusterProperties "{templateName APACHE20_PHP5}"}
```

- Using Jython string:

```
AdminTask.createPHPDynamicCluster(dcName, '[-membershipPolicy "node_property$APACHE_2_0 IS NOT NULL AND node_property$PHP_5 IS NOT NULL" -dynamicClusterProperties "{operationalMode automatic}{minInstances 1}{maxInstances -1}{numVerticalInstances 1}" -clusterProperties "{templateName APACHE20_PHP5}"']')
```

or, if you are running on WebSphere Application Server Version 7.0 or later:

```
AdminTask.createPHPDynamicCluster('dcName', '[-membershipPolicy "node_property$APACHE_2_0 IS NOT NULL AND node_property$PHP_5 IS NOT NULL" -dynamicClusterProperties "[operationalMode automatic][minInstances 1][maxInstances -1][numVerticalInstances 1]" -clusterProperties "[templateName APACHE20_PHP5]"']')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createPHPDynamicCluster {-interactive}
```

- Using Jython string:

```
AdminTask.createPHPDynamicCluster ('[-interactive]')
```

createDynamicClusterFromStaticCluster

The `createDynamicClusterFromStaticCluster` command creates a new dynamic cluster from an existing static cluster.

Optional parameters

- **-templateName:** Specifies the name of the existing static cluster member template. The format of the name must be in the `cell_name/node_name/server_name` format. If the template is not specified, an existing static cluster member template is randomly chosen. If no members exist, the defaultXD server template is used. (String, optional)
- **-dynamicClusterProperties:** Specifies the dynamic cluster properties. (String, optional) You can specify the following options:
 - **operationalMode:** Specifies the operational mode. The valid values are: `manual`, `supervised`, and `automatic`.
 - **minInstances:** Specifies the minimum number of cluster instances as an integer.
 - **serverInactivityTime:** Specifies the time to wait before stopping instances, in minutes. This value is valid only when the `minInstances` parameter value is set to 0.
 - **maxInstances:** Specifies the maximum number of cluster instances as an integer. If you do not want a limit to the number of cluster instances that can start, specify -1.

Return value

The command returns the `javax.management.ObjectName` value of the dynamic cluster that you created.

Batch mode example usage

- Using Jacl:

```
$AdminTask createDynamicClusterFromStaticCluster clusterName {-templateName  
cellName/nodeName/serverName}
```

- Using Jython string:

```
AdminTask.createDynamicClusterFromStaticCluster('clusterName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createDynamicClusterFromStaticCluster {-interactive}
```

- Using Jython string:

```
AdminTask.createDynamicClusterFromStaticCluster ('[-interactive]')
```

createDynamicClusterFromForeignServers

The `createDynamicClusterFromForeignServers` command creates a new dynamic cluster of assisted life-cycle servers. Create representations of these servers with an administrative task or in the administrative console before you create a dynamic cluster.

Required parameters

- **-foreignServers [[node_name1 server_name1][node_name2 server_name2] ...]: Specifies the node and server names of the existing other middleware servers. (String, required)**

Optional parameters

- **-dynamicClusterProperties: Specifies the dynamic cluster properties. (String, optional)**
You can specify the following options:
 - **operationalMode: Specifies the operational mode. The valid values are: manual, supervised, and automatic.**
 - **minInstances: Specifies the minimum number of cluster instances as an integer.**
 - **serverInactivityTime: Specifies the time to wait before stopping instances, in minutes. This value is valid only when the minInstances parameter value is set to 0.**
 - **maxInstances: Specifies the maximum number of cluster instances as an integer. If you do not want a limit to the number of cluster instances that can start, specify -1.**

Return value

The command returns the `javax.management.ObjectName` identification of the server type that you updated.

Batch mode example usage

- Using Jacl:

```
$AdminTask createDynamicClusterFromForeignServers dcName {-foreignServers {{nodeName1  
serverName1}{nodeName2 serverName2}}}
```

- Using Jython string:

```
AdminTask.createDynamicClusterFromForeignServers('dcName', '[-foreignServers [[nodeName1  
serverName1] [nodeName2 serverName2]]]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createDynamicClusterFromForeignServers {-interactive}
```

- Using Jython string:

```
AdminTask.createDynamicClusterFromForeignServers ('[-interactive]')
```

createODRDynamicCluster

Use the `createODRDynamicCluster` command to create on demand router dynamic clusters.

Target object

Specifies the name of the dynamic cluster.

Required parameters

membershipPolicy: Specifies the membership policy.

Optional parameters

- **-dynamicClusterProperties**: Specifies the dynamic cluster properties. (String, optional) You can specify the following options:
 - **operationalMode**: Specifies the operational mode. The valid values are: manual, supervised, and automatic.
 - **minInstances**: Specifies the minimum number of cluster instances as an integer.
 - **serverInactivityTime**: Specifies the time to wait before stopping instances, in minutes. This value is valid only when the **minInstances** parameter value is set to 0.
 - **maxInstances**: Specifies the maximum number of cluster instances as an integer. If you do not want a limit to the number of cluster instances that can start, specify -1.
 - **numVerticalInstances**: Specifies the number of instances that can start on the same node by creating vertical instances. Specify an integer value.
- **-clusterProperties**: Specifies the cluster properties. (String, optional) You can specify the following options:
 - **preferLocal**: Specifies whether enterprise bean requests are routed to the node on which the client resides when possible. The valid values are true or false.
 - **createDomain**: Specifies that a replication domain is created for this cluster. Replication domains can be created only for WebSphere Application Server application servers. The valid values are true or false.
 - **coreGroup**: Specifies the name of the core group for the dynamic cluster. You only need to specify the core group for WebSphere Application Server application servers. The value can be DefaultCoreGroup or the core group name.
 - **templateName**: Specifies the name of the server template. The valid input includes the following values:
 - **For WebSphere Application Server dynamic clusters**: defaultXD, defaultXDZOS, *cell_name/node_name/was_server_name*
 - **z/OS clusterShortName**: For z/OS platforms, specifies the short name of the cluster. After the dynamic cluster has been created, you can use the **AdminTask changeClusterShortName** command to update this field.
 - **z/OS serverSpecificShortNames**: For z/OS platforms, specifies the specific short name of cluster member in a comma-separated list format, for example: SSN1,SSN2. Use commas to separate multiple short names. If you do not provide enough short names to be used for all of the cluster members, then the remaining cluster members are assigned generated generic short names, such as BBOS001, BBOS002, and so on. After the dynamic cluster has been created, you can update this list with the **serverSpecificShortNames** custom property. This property is set on the dynamic cluster. If you want to change this property with scripting after the dynamic cluster has been created, you can use the **AdminTask changeServerSpecificShortName** command.

Batch mode example usage

- Using Jacl:

```
$AdminTask createODRDynamicCluster dcName {parameters} {-membershipPolicy "node_nodgroup = 'DefaultNodeGroup' AND node_property$com.ibm.websphere.wxdopProductShortName = 'WXDOP'" -dynamicClusterProperties "{operationalMode automatic}{minInstances 2}" -clusterProperties "{templateName defaultXD}{preferLocal false}"}
```

- Using Jython string:

```
AdminTask.createODRDynamicCluster('dcName', '[-membershipPolicy "node_nodgroup = 'DefaultNodeGroup' AND node_property$com.ibm.websphere.wxdopProductShortName = 'WXDOP'" -dynamicClusterProperties [[operationalMode automatic][minInstances 2]] -clusterProperties [[templateName defaultXD][preferLocal false]]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createODRDynamicCluster {-interactive}
```

- Using Jython string:

```
AdminTask.createODRDynamicCluster ('[-interactive]')
```

addForeignServersToDynamicCluster

The `addForeignServersToDynamicCluster` command adds foreign servers to an existing dynamic cluster.

Required parameters

- `-foreignServers [[node_name1 server_name1][node_name2 server_name1] ...]`: Specifies the node and server names of the existing other middleware servers. (String, required)

Return value

The command returns the `javax.management.ObjectName` identification of the server type that you updated.

Batch mode example usage

- Using Jacl:

```
$AdminTask addForeignServersToDynamicCluster dcName {-foreignServers {{nodeName1 serverName1}
{nodeName2 serverName2}}}
```

- Using Jython string:

```
AdminTask.addForeignServersToDynamicCluster('dcName', '[-foreignServers [[nodeName1
serverName1] [nodeName2 serverName2]]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addForeignServersToDynamicCluster {-interactive}
```

- Using Jython string:

```
AdminTask.addForeignServersToDynamicCluster ('[-interactive]')
```

removeForeignServersFromDynamicCluster

The `removeForeignServersFromDynamicCluster` command removes foreign servers from the dynamic cluster

Required parameters

- `-foreignServers [[node_name1 server_name1][node_name2 server_name1] ...]`: Specifies the node and server names of the existing other middleware servers. (String, required)

Return value

The command returns the `javax.management.ObjectName` identification of the server type that you updated.

Batch mode example usage

- Using Jacl:

```
$AdminTask removeForeignServersFromDynamicCluster dcName {-foreignServers {{nodeName1
serverName1}{nodeName2 serverName2}}}
```

- Using Jython string:

```
AdminTask.removeForeignServersFromDynamicCluster('dcName', '[-foreignServers [[nodeName1
serverName1] [nodeName2 serverName2]]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeForeignServersFromDynamicCluster {-interactive}
```

- Using Jython string:

```
AdminTask.removeForeignServersFromDynamicCluster ('[-interactive]')
```

deleteDynamicCluster

The **deleteDynamicCluster** deletes a dynamic cluster from the configuration.

Return value

The command returns the `javax.management.ObjectName` identification of the dynamic cluster that you deleted.

Batch mode example usage

- Using Jacl:

```
$AdminTask deleteDynamicCluster dcName
```

- Using Jython string:

```
AdminTask.deleteDynamicCluster('dcName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask deleteDynamicCluster {-interactive}
```

- Using Jython string:

```
AdminTask.deleteDynamicCluster ('[-interactive]')
```

getDynamicClusterMembers

The **getDynamicClusterMembers** command displays the members of the specified dynamic cluster and node name. If the node name is not specified, then all of the members of the dynamic cluster are displayed.

Optional parameters

- **-nodeName: Specifies the name of a node. (String, optional)**

Return value

The command returns a list of `ClusterMember` objects.

Batch mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterMembers dcName
```

- Using Jython string:

```
AdminTask.getDynamicClusterMembers('dcName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterMembers {-interactive}
```

- Using Jython string:

```
AdminTask.getDynamicClusterMembers ('[-interactive]')
```

getDynamicClusterOperationalMode

The **getDynamicClusterOperationalMode** command displays the operational mode of the dynamic cluster.

Return value

The command returns the value of `operationalMode`. String.

Batch mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterOperationalMode dcName
```

- Using Jython string:

```
AdminTask.getDynamicClusterOperationalMode('dcName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterOperationalMode {-interactive}
```

- Using Jython string:

```
AdminTask.getDynamicClusterOperationalMode (['-interactive'])
```

getDynamicClusterMembershipPolicy

The `getDynamicClusterMembershipPolicy` command displays the dynamic cluster membership policy.

Return value

The command returns the value of `membershipPolicy`. String.

Batch mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterMembershipPolicy dcName
```

- Using Jython string:

```
AdminTask.getDynamicClusterMembershipPolicy('dcName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterMembershipPolicy {-interactive}
```

- Using Jython string:

```
AdminTask.getDynamicClusterMembershipPolicy (['-interactive'])
```

getDynamicClusterServerType

The `getDynamicClusterServerType` command displays the dynamic cluster server type.

Return value

The command returns the value of `serverType`. String.

Batch mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterServerType dcName
```

- Using Jython string:

```
AdminTask.getDynamicClusterServerType('dcName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterServerType {-interactive}
```

- Using Jython string:

```
AdminTask.getDynamicClusterServerType ('[-interactive]')
```

getDynamicClusterMinInstances

The **getDynamicClusterMinInstances** command displays the minimum number of cluster instances for the specified dynamic cluster.

Return value

The command returns the value of minInstances. Integer.

Batch mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterMinInstances dcName
```

- Using Jython string:

```
AdminTask.getDynamicClusterMinInstances ('dcName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterMinInstances {-interactive}
```

- Using Jython string:

```
AdminTask.getDynamicClusterMinInstances ('[-interactive]')
```

getDynamicClusterMaxInstances

The **getDynamicClusterMaxInstances** command displays the maximum number of cluster instances for the specified dynamic cluster.

Return value

The command returns the value of maxInstances. Integer.

Batch mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterMaxInstances dcName
```

- Using Jython string:

```
AdminTask.getDynamicClusterMaxInstances ('dcName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterMaxInstances {-interactive}
```

- Using Jython string:

```
AdminTask.getDynamicClusterMaxInstances ('[-interactive]')
```

getDynamicClusterVerticalInstances

The **getDynamicClusterVerticalInstances** command displays the number of dynamic cluster vertical stacking of instances on the node.

Return value

The command returns the value of numVerticalInstances. Integer.

Batch mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterVerticalInstances dcName
```

- Using Jython string:

```
AdminTask.getDynamicClusterVerticalInstances('dcName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterVerticalInstances {-interactive}
```

- Using Jython string:

```
AdminTask.getDynamicClusterVerticalInstances ('[-interactive]')
```

getDynamicClusterIsolationProperties

The **getDynamicClusterVerticalInstances** command displays the dynamic cluster isolation properties.

Return value

The command returns the value of isolationGroup and strictIsolationEnabled. String.

Batch mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterIsolationProperties dcName
```

- Using Jython string:

```
AdminTask.getDynamicClusterIsolationProperties('dcName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask getDynamicClusterIsolationProperties {-interactive}
```

- Using Jython string:

```
AdminTask.getDynamicClusterIsolationProperties ('[-interactive]')
```

listDynamicClusters

The **listDynamicClusters** command displays all the dynamic clusters in the cell.

Return value

The command returns each of the dynamic cluster names. String.

Batch mode example usage

- Using Jacl:

```
$AdminTask listDynamicClusters
```

- Using Jython string:

```
AdminTask.listDynamicClusters()
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listDynamicClusters {-interactive}
```

- Using Jython string:

```
AdminTask.listDynamicClusters ('[-interactive]')
```

listDynamicClusterIsolationGroups

The **listDynamicClusterIsolationGroups** command displays all the dynamic cluster isolation groups in the cell.

Return value

The command returns a list of the dynamic cluster isolation groups. String.

Batch mode example usage

- Using Jacl:

```
$AdminTask listDynamicClusterIsolationGroups
```

- Using Jython string:

```
AdminTask.listDynamicClusterIsolationGroups()
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listDynamicClusterIsolationGroups {-interactive}
```

- Using Jython string:

```
AdminTask.listDynamicClusterIsolationGroups ('[-interactive]')
```

listDynamicClusterIsolationGroupMembers

The **listDynamicClusterIsolationGroupMembers** command displays the dynamic cluster isolation group members for the specified isolation group.

Return value

The command returns a list of the dynamic cluster isolation group members. String.

Batch mode example usage

- Using Jacl:

```
$AdminTask listDynamicClusterIsolationGroupMembers isolationGroupName
```

- Using Jython string:

```
AdminTask.listDynamicClusterIsolationGroupMembers('isolationGroupName')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listDynamicClusterIsolationGroupMembers {-interactive}
```


- Using Jython string:

```
AdminTask.listDynamicClusterIsolationGroupMembers ('[-interactive]')
```

modifyDynamicClusterIsolationProperties

The `modifyDynamicClusterIsolationProperties` command modifies dynamic cluster isolation properties.

Optional parameters

- **-isolationGroup:** Specifies the name of the isolation group. (String, optional)
- **-strictIsolationEnabled:** Indicates if strict isolation is enabled. (String, optional)

Return value

The return value is void.

Batch mode example usage

- Using Jacl:

```
$AdminTask modifyDynamicClusterIsolationProperties dcName {-isolationGroup isolationGroupName  
-strictIsolationEnabled true}
```

- Using Jython string:

```
AdminTask.modifyDynamicClusterIsolationProperties('dcName', '[-isolationGroup  
isolationGroupName -strictIsolationEnabled true]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask modifyDynamicClusterIsolationProperties {-interactive}
```

- Using Jython string:

```
AdminTask.modifyDynamicClusterIsolationProperties ('[-interactive]')
```

setDynamicClusterOperationalMode

The `setDynamicClusterOperationalMode` command sets the operational mode for the dynamic cluster.

Required parameters

- **-operationalMode:** Specifies the operational mode of the dynamic cluster. Set the value to automatic, manual, or supervised. (String, required)

Return value

The return value is void.

Batch mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterOperationalMode dcName {-operationalMode manual}
```

- Using Jython string:

```
AdminTask.setDynamicClusterOperationalMode('dcName', '[-operationalMode manual]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterOperationalMode {-interactive}
```

- Using Jython string:

```
AdminTask.setDynamicClusterOperationalMode ('[-interactive]')
```

setDynamicClusterMembershipPolicy

The **setDynamicClusterMembershipPolicy** command sets the membership policy for the dynamic cluster.

Required parameters

-membershipPolicy

Specifies the membership policy subexpression. (String, required)

Return value

The return value is void.

Batch mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterMembershipPolicy dcName {-membershipPolicy  
node_nodegroup='ngName'}
```

- Using Jython string:

```
AdminTask.setDynamicClusterMembershipPolicy('dcName', '[-membershipPolicy  
node_nodegroup=\'ngName\']')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterMembershipPolicy {-interactive}
```

- Using Jython string:

```
AdminTask.setDynamicClusterMembershipPolicy ('[-interactive]')
```

setDynamicClusterMinInstances

The **setDynamicClusterMinInstances** command sets the minimum number of cluster instances for the dynamic cluster.

Required parameters

-minInstances

Specifies the minimum number of cluster instances. (Integer, required)

Return value

The return value is void.

Batch mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterMinInstances dcName {-minInstances 1}
```

- Using Jython string:

```
AdminTask.setDynamicClusterMinInstances('dcName', '[-minInstances 1]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterMinInstances {-interactive}
```

- Using Jython string:

```
AdminTask.setDynamicClusterMinInstances ( '[-interactive]'
```

setDynamicClusterMaxInstances

The **setDynamicClusterMaxInstances** command sets the maximum number of cluster instances for the dynamic cluster.

Required parameters

-maxInstances

Specifies the minimum number of cluster instances. (Integer, required)

Return value

The return value is void.

Batch mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterMaxInstances dcName {-maxInstances -1}
```

- Using Jython string:

```
AdminTask.setDynamicClusterMaxInstances('dcName', '[-maxInstances 1]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterMaxInstances {-interactive}
```

- Using Jython string:

```
AdminTask.setDynamicClusterMaxInstances ( '[-interactive]'
```

setDynamicClusterVerticalInstances

The **setDynamicClusterVerticalInstances** command sets the number of dynamic cluster vertical stacking instances on the node.

Required parameters

-numVerticalInstances

Specifies the number of vertical stacking of instances on a node. (Integer, required)

Return value

The return value is void.

Batch mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterVerticalInstances dcName {-numVerticalInstances 2}
```

- Using Jython string:

```
AdminTask.setDynamicClusterVerticalInstances('dcName', '[-numVerticalInstances 2]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask setDynamicClusterVerticalInstances {-interactive}
```

- Using Jython string:

```
AdminTask.setDynamicClusterVerticalInstances ( '[-interactive]'
```

testDynamicClusterMembershipPolicy

The `testDynamicClusterMembershipPolicy` command tests the dynamic cluster membership policy to see which nodes are returned.

Required parameters

-membershipPolicy

Specifies the membership policy subexpression. (String, required)

Return value

The command returns a String [] list of node names.

Batch mode example usage

- Using Jacl:

```
$AdminTask testDynamicClusterMembershipPolicy {-membershipPolicy node_nodegroup='ngName'}
```

- Using Jython string:

```
AdminTask.testDynamicClusterMembershipPolicy(['-membershipPolicy node_nodegroup=\'ngName\'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask testDynamicClusterMembershipPolicy {-interactive}
```

- Using Jython string:

```
AdminTask.testDynamicClusterMembershipPolicy (['-interactive'])
```

createNonWASDynamicCluster

The `createNonWASDynamicCluster` command creates a new dynamic cluster of complete life-cycle management servers.

Required parameters

- **-serverType: Specifies the server type. (String, required)**

To see the server types, view the `install_root\profiles\dmgr_profile\config\templates\servertypes` directory.

- **-membershipPolicy: Specifies the membership policy subexpression. (String, required)**

Optional parameters

- **-dynamicClusterProperties: Specifies the dynamic cluster properties. (String, optional)**

You can specify the following options:

- **operationalMode: Specifies the operational mode. The valid values are: manual, supervised, and automatic.**
- **minInstances: Specifies the minimum number of cluster instances as an integer.**
- **serverInactivityTime: Specifies the time to wait before stopping instances, in minutes. This value is valid only when the minInstances parameter value is set to 0.**
- **maxInstances: Specifies the maximum number of cluster instances as an integer. If you do not want a limit to the number of cluster instances that can start, specify -1.**
- **numVerticalInstances: Specifies the number of instances that can start on the same node by creating vertical instances. Specify an integer value.**
- **-clusterProperties: Specifies the cluster properties (String, optional)**
 - **templateName: Specifies the name of the server template. The valid input includes the following values:**
 - **For PHP dynamic clusters: APACHE13_PHP4, APACHE13_PHP5, APACHE20_PHP4, APACHE20_PHP5, APACHE22_PHP4, APACHE22_PHP5, cellName/nodeName/php_serverName.**
 - **For WebSphere Application Server Community Edition dynamic clusters: wasce20, cell_name/node_name/wasce_server_name**

Return value

The command returns the `javax.management.ObjectName` value of the dynamic cluster that you created.

Batch mode example usage

- Using Jacl:

```
$AdminTask createNonWASDynamicCluster dcName {-serverType WASCE_SERVER -membershipPolicy "node_property$com.ibm.websphere.wasceProductShortName = 'WASCE' OR node_property$WASCE_2.0.0.0 IS NOT NULL" -dynamicClusterProperties "{operationalMode automatic}{minInstances 1}{maxInstances -1}{numVerticalInstances 1}" -clusterProperties "{templateName wasce20}"}
```

- Using Jython string:

```
AdminTask.createNonWASDynamicCluster(dcName, '[-serverType WASCE_SERVER -membershipPolicy "node_property$com.ibm.websphere.wasceProductShortName = \'WASCE\' OR node_property$WASCE_2.0.0.0 IS NOT NULL" -dynamicClusterProperties "{operationalMode automatic}{minInstances 1}{maxInstances -1}{numVerticalInstances 1}" -clusterProperties "{templateName wasce20}"]')
```

or, if you are running on WebSphere Application Server Version 7.0 or later:

```
AdminTask.createNonWASDynamicCluster('dcName', '[-serverType WASCE_SERVER -membershipPolicy "node_property$com.ibm.websphere.wasceProductShortName = \'WASCE\' OR node_property$WASCE_2.0.0.0 IS NOT NULL" -dynamicClusterProperties "[operationalMode automatic][minInstances 1][maxInstances -1][numVerticalInstances 1]" -clusterProperties "[templateName wasce20]"]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createNonWASDynamicCluster {-interactive}
```

- Using Jython string:

```
AdminTask.createNonWASDynamicCluster('[-interactive]')
```

Edit the dynamic cluster server template

Identify the dynamic cluster server template and assign the template to the **serverid** variable.

Batch mode example usage

- Using Jacl:

```
set serverid [$AdminConfig getid /Cell:mycell/DynamicCluster:mydynamiccluster/Server:mydynamiccluster/]
```

- Using Jython string:

```
serverid = AdminConfig.getid('/Cell:mycell/DynamicCluster:mydynamiccluster/Server:mydynamiccluster/')
```

You can use the **serverid** variable to modify, create, or delete configuration objects in the dynamic cluster server template.

Parent topic: [Administrator scripting interfaces](#)

Related tasks:

[Using the AdminTask object for scripted administration](#)
[Creating dynamic clusters](#)
[Configuring vertical stacking](#)

Related reference:

[Commands for the AdminConfig object](#)

Related information:

[Dynamic cluster custom properties](#)

Default work class administrative tasks

You can use the default work class administrative tasks to create work classes.

The administrative tasks for default work classes include the following commands:

- [createDefaultASPWorkClass](#)
- [createDefaultARPWorkClass](#)
- [createDefaultGSPWorkClass](#)
- [createDefaultGRPWorkClass](#)
- [createDefaultSystemSPWorkClass](#)
- [createDefaultSystemRPWorkClass](#)

createDefaultASPWorkClass

The createDefaultASPWorkClass command creates default application service policy work classes.

Required parameters

-appname

Specifies the name of the application deployment. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask createDefaultASPWorkClass {-appname A-edition1.0}
```

- Using Jython:

```
AdminTask.createDefaultASPWorkClass ('[-appname A-edition1.0]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createDefaultASPWorkClass {-interactive}
```

- Using Jython:

```
AdminTask.createDefaultASPWorkClass ('[-interactive]')
```

createDefaultARPWorkClass

The createDefaultARPWorkClass command creates default application routing policy work classes.

Required parameters

-appname

Specifies the name of the application deployment. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask createDefaultARPWorkClass {-appname A-edition1.0}
```

- Using Jython:

```
AdminTask.createDefaultARPWorkClass ('[-appname A-edition1.0]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createDefaultARPWorkClass {-interactive}
```

- Using Jython:

```
AdminTask.createDefaultARPWorkClass ( ['-interactive'] )
```

createDefaultGSPWorkClass

The createDefaultGSPWorkClass command creates default generic server service policy work classes.

Required parameters

-odrname

Specifies the name of the on demand router (ODR). (String, required)

-odrcode

Specifies the name of the node where the ODR runs. (String, required)

-vhost

Specifies the name of the virtual host. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask createDefaultGSPWorkClass {-odrname myodr -odrcode mynode -vhost default_host}
```

- Using Jython:

```
AdminTask.createDefaultGSPWorkClass ( ['-odrname myodr -odrcode mynode -vhost default_host'] )
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createDefaultGSPWorkClass {-interactive}
```

- Using Jython:

```
AdminTask.createDefaultGSPWorkClass ( ['-interactive'] )
```

createDefaultGRPWorkClass

The createDefaultGRPWorkClass command creates default generic server routing policy work classes.

Required parameters

-odrname

Specifies the name of the on demand router (ODR). (String, required)

-odrcode

Specifies the name of the node where the ODR runs. (String, required)

-vhost

Specifies the name of the virtual host. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask createDefaultGRPWorkClass {-odrname myodr -odrnodel mynode -vhost default_host}
```

- Using Jython:

```
AdminTask.createDefaultGRPWorkClass ('[-odrname myodr -odrnodel mynode -vhost default_host]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createDefaultGRPWorkClass {-interactive}
```

- Using Jython:

```
AdminTask.createDefaultGRPWorkClass ('[-interactive]')
```

createDefaultSystemSPWorkClass

The createDefaultSystemSPWorkClass command creates default system application service policy work classes.

Required parameters

-appname

Specifies the name of the application deployment. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask createDefaultSystemSPWorkClass {-appname A-edition1.0}
```

- Using Jython:

```
AdminTask.createDefaultSystemSPWorkClass ('[-appname A-edition1.0]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createDefaultSystemSPWorkClass {-interactive}
```

- Using Jython:

```
AdminTask.createDefaultSystemSPWorkClass ('[-interactive]')
```

createDefaultSystemRPWorkClass

The createDefaultSystemRPWorkClass command creates default system application routing policy work classes.

Required parameters

-appname

Specifies the name of the application deployment. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask createDefaultSystemRPWorkClass {-appname A-edition1.0}
```

- Using Jython:

```
AdminTask.createDefaultSystemRPWorkClass ('[-appname A-edition1.0]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createDefaultSystemRPWorkClass {-interactive}
```

- Using Jython:

```
AdminTask.createDefaultSystemRPWorkClass ('[-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Overview of work classes](#)

[Work class types](#)

Health policy administrative tasks

You can use the following administrative tasks to create, delete, and manage health policies.

- [createHealthPolicy command](#)
- [modifyHealthPolicy command](#)
- [deleteHealthPolicy command](#)
- [listHealthPolicies command](#)
- [createHealthAction command](#)
- [modifyHealthAction command](#)
- [deleteHealthAction command](#)
- [listHealthActions command](#)
- [executeHealthAction command](#)

createHealthPolicy command

The createHealthPolicy command creates a new health policy.

Table 1. createHealthPolicy command arguments

Argument	Description
name (required)	Specifies a name for the health policy that is unique in the cell.
description	Specifies a description for the health policy.
reactionMode	Specifies if runtime tasks are created before taking actions on a health policy, or if actions are automatic. The value can be SUPERVISE or AUTOMATIC.

Table 2. addCondition step arguments

Argument	Description
type (required)	Specifies the condition type. The value can be CUSTOM, AGE, WORKLOAD, MEMORY_LEAK, EXCESSIVE_MEMORY, EXCESSIVE_RESPONSE_TIME, EXCESSIVE_REQUEST_TIMEOUT, or STORM_DRAIN. Each value has different parameters that you must specify.
expression	If you are using a custom condition, then specify a subexpression to evaluate.
params	<ul style="list-style-type: none"> • AGE <ul style="list-style-type: none"> ◦ maxAge: Specifies maximum age. ◦ ageUnits: Specifies the type of units for the age parameter. The value can be HOURS or MINUTES. The default is HOURS. • WORKLOAD <ul style="list-style-type: none"> ◦ totalRequests: Specifies total requests to service before restarting the members. The number must be greater than or equal to 1000. • MEMORY_LEAK <ul style="list-style-type: none"> ◦ level: Specifies the detection level for the memory leak. The value can be FAST, NORMAL, or CONSERVATIVE. The default is FAST. • EXCESSIVE_MEMORY <ul style="list-style-type: none"> ◦ timeOverThreshold: Specifies the offending time period for the excessive memory condition. The default is 1 MINUTE. ◦ timeUnits: Specifies the units for the timeOverThreshold value. The value can be MINUTES or SECONDS. The default is MINUTES. ◦ memoryUsed: Specifies a percentage that represents the maximum Java™ virtual machine (JVM) heap size to use for the JVM process. The value ranges from 1 to 99. • EXCESSIVE_RESPONSE_TIME <ul style="list-style-type: none"> ◦ responseTime: Specifies the average response time that is considered to be excessive. When the average reaches this value, members restart. ◦ responseTimeUnits: Specifies the time units for the response time. The value can be MINUTES, SECONDS, or MILLISECONDS. • EXCESSIVE_REQUEST_TIMEOUT <ul style="list-style-type: none"> ◦ timeoutPercent: Specifies the threshold value for the percentage of the timed out requests to cause a breach of condition. This value ranges from 1 to 99. • STORM_DRAIN <ul style="list-style-type: none"> ◦ level: Specifies the detection level. The value can be NORMAL or CONSERVATIVE. NORMAL is less accurate, but can detect a storm drain condition quickly. CONSERVATIVE is the most accurate, but takes a longer time to detect a storm drain condition.

Table 3. addAction step arguments.

Use the addAction step arguments in table format. With table format, you specify the parameters in order without the parameter name.

Argument	Description
type (required)	Specifies the action type. The value can be CUSTOM, HEAPDUMP, THREADDUMP, RESTART_SERVER, SET_MAINT_MODE, MAINT_MODE_BREAK, or MAINT_MODE_STOP.
stepNum (required)	Specifies the order in which to take the actions. Step 1 is completed first, and so on.
action	Specifies the name of a custom action. Use the createHealthAction command to create a custom action.
node	Specifies the node on which to take the action.
server	Specifies the server on which to take the action.

Table 4. addMember step arguments.

Use the addMember step arguments in table format. With table format, you specify the parameters in order without the parameter name.

Argument	Description
type (required)	Specifies the type of member. The value can be CELL, CLUSTER, DYNAMIC_CLUSTER, SERVER, or ODR.
name (required)	Specifies the name of the member.
node	Specifies the name of the node. The node name is required if you select the SERVER type.

createHealthPolicy command examples

Interactive mode example usage:

- Using Jacl:

```
$AdminTask createHealthPolicy {-interactive}
```

- Using Jython:

```
AdminTask.createHealthPolicy ('[-interactive]')
```

Batch mode example usage:

The following command creates a health policy with one of the predefined health conditions:

```
$AdminTask createHealthPolicy { -name myHealthPolicy -description "My Health Policy" -reactionMode SUPERVISE -addCondition { -type AGE -params {{maxAge 12} {ageUnits HOURS}} } -addAction {{HEAPDUMP 1}{CUSTOM 2 myAction myNode myServer} } -addMember {SERVER myServer myNode} }
```

The following command creates a health policy with a custom condition:

```
$AdminTask createHealthPolicy { -name myHealthPolicy -description "My Health Policy" -reactionMode SUPERVISE -addCondition { -type CUSTOM -expression "ODRServerMetric_FromServerStart$errors > 100L" } -addAction {{HEAPDUMP 1}{CUSTOM 2 myAction myNode myServer} } -addMember {SERVER myServer myNode} }
```

Put the expression for your custom condition in quotes.

modifyHealthPolicy command

The modifyHealthPolicy command changes an existing health policy.

Table 5. Initial arguments

Argument	Description
name (required)	Specifies the name of the health policy that you want to change.
reactionMode	Specifies if runtime tasks are created before taking actions on a health policy, or if actions are automatic. The value can be SUPERVISE or AUTOMATIC.

Table 6. modifyCondition step arguments

Argument	Description
type	Specifies the condition type. The value can be CUSTOM, AGE, WORKLOAD, MEMORY_LEAK, EXCESSIVE_MEMORY, EXCESSIVE_RESPONSE_TIME, EXCESSIVE_REQUEST_TIMEOUT, or STORM_DRAIN. Each value has different parameters that you must specify.

expression	If you are using a custom condition, then specify a subexpression to evaluate.
params	<ul style="list-style-type: none"> • AGE <ul style="list-style-type: none"> ◦ maxAge: Specifies maximum age. ◦ ageUnits: Specifies the type of units for the age parameter. The value can be HOURS or MINUTES. The default is HOURS. • WORKLOAD <ul style="list-style-type: none"> ◦ totalRequests: Specifies total requests to service before restarting the members. The number must be greater than or equal to 1000. • MEMORY_LEAK <ul style="list-style-type: none"> ◦ level: Specifies the detection level for the memory leak. The value can be FAST, NORMAL, or CONSERVATIVE. The default is FAST. • EXCESSIVE_MEMORY <ul style="list-style-type: none"> ◦ timeOverThreshold: Specifies the offending time period for the excessive memory condition. The default is 1 MINUTE. ◦ timeUnits: Specifies the units for the timeOverThreshold value. The value can be MINUTES or SECONDS. The default is MINUTES. ◦ memoryUsed: Specifies a percentage that represents the maximum Java virtual machine (JVM) heap size to use for the JVM process. The value ranges from 1 to 99. • EXCESSIVE_RESPONSE_TIME <ul style="list-style-type: none"> ◦ responseTime: Specifies the average response time that is considered to be excessive. When the average reaches this value, members restart. ◦ responseTimeUnits: Specifies the time units for the response time. The value can be MINUTES, SECONDS, or MILLISECONDS. • EXCESSIVE_REQUEST_TIMEOUT <ul style="list-style-type: none"> ◦ timeoutPercent: Specifies the threshold value for the percentage of the timed out requests to cause a breach of condition. This value ranges from 1 to 99. • STORM_DRAIN <ul style="list-style-type: none"> ◦ level: Specifies the detection level. The value can be NORMAL or CONSERVATIVE. NORMAL is less accurate, but can detect a storm drain condition quickly. CONSERVATIVE is the most accurate, but takes a longer time to detect a storm drain condition.

Table 7. removeAction step arguments

Argument	Description
name	Specifies the name of the health action to remove. The name parameter is only used for custom actions.
type (required)	Specifies the action type. The value can be CUSTOM, HEAPDUMP, THREADDUMP, RESTART_SERVER, SET_MAINT_MODE, MAINT_MODE_BREAK, or MAINT_MODE_STOP.

Table 8. addAction step arguments

Argument	Description
type (required)	Specifies the action type. The value can be CUSTOM, HEAPDUMP, THREADDUMP, RESTART_SERVER, SET_MAINT_MODE, MAINT_MODE_BREAK, or MAINT_MODE_STOP.
name	Specifies the name of the health action. The name parameter is only used for custom actions.
node	Specifies the node on which to take the action.
server	Specifies the server on which to take the action.
stepNum (required)	Specifies the order in which to take the actions. Step 1 is completed first, and so on.

Table 9. orderAction step arguments

Argument	Description
name	Specifies the name of the health action. The name parameter is only used for custom actions.
type	Specifies the action type. The value can be CUSTOM, HEAPDUMP, THREADDUMP, RESTART_SERVER, SET_MAINT_MODE, MAINT_MODE_BREAK, or MAINT_MODE_STOP.
stepNum (required)	Specifies the order in which to take the actions. Step 1 is completed first, and so on.

Table 10. addMember step arguments

Argument	Description
type (required)	Specifies the type of member. The value can be CELL, CLUSTER, DYNAMIC_CLUSTER, SERVER, or ODR.
name (required)	Specifies the name of the member.
node	Specifies the name of the node. The node name is required if you select the SERVER type.

modifyHealthPolicy command examples

Interactive mode example usage:

- Using Jacl:

```
$AdminTask modifyHealthPolicy {-interactive}
```

- Using Jython:

```
AdminTask.modifyHealthPolicy ('[-interactive]')
```

Batch mode example usage:

```
$AdminTask modifyHealthPolicy { -name myHealthPolicy -removeAction { -type HEAPDUMP } }
```

deleteHealthPolicy command

The deleteHealthPolicy command removes the named health policy.

Table 11. deleteHealthPolicy command arguments

Argument	Description
name (required)	Specifies the health policy name to delete.

deleteHealthPolicy command examples

Batch mode example usage:

- Using Jacl:

```
$AdminTask deleteHealthPolicy {-name MyHealthPolicy}
```

- Using Jython:

```
AdminTask.deleteHealthPolicy('MyHealthPolicy')
```

Interactive mode example usage:

- Using Jacl:

```
$AdminTask deleteHealthPolicy {-interactive}
```

- Using Jython:

```
AdminTask.deleteHealthPolicy ('[-interactive]')
```

listHealthPolicies command

The listHealthPolicies command lists each configured health policy in the cell.

listHealthPolicies command examples

- Using Jacl:

```
$AdminTask listHealthPolicies
```

- Using Jython:

```
AdminTask.listHealthPolicies
```

createHealthAction command

The createHealthAction command creates a new health action that you can use when you define your health policies.

Table 12. createHealthAction command arguments

Argument	Description
name (required)	Specifies a name for the health action.

You can create a Java action or an action that is any other type of executable file.

Table 13. non-Java step arguments

Argument	Description
executable	Specifies the executable file path. This argument is required if you specify a nonJava step argument.
executableArgs	Specifies a string array of arguments to pass to the executable file.
osNames	Specifies a comma-delimited list of operating systems. Specify null to have the action apply for each operating system. The valid operating system names include: windows, unix, and zos.
workingDir	Specifies the location in the file system from where the command is invoked. This argument is required if you specify a nonJava step argument.
usernameVar	Specifies a variable for the user name.
userNameVal	Specifies the value of the user name variable.
passwordVar	Specifies a variable for the password.
passwordVal	Specifies the value of the password variable.

Table 14. Java step arguments

Argument	Description
javaDir	Specifies the location of your Java executable file, for example, \${JAVA_HOME}. You can specify a variable or a path.
executableTargetType	Specifies the file type that you are using for your Java executable file. The value can be EXECUTABLE_JAR or JAVA_CLASS.
executable	Specifies the executable file path. This argument is required if you specify a Java step argument.
executableArgs	Specifies a string array of arguments to pass to the executable file.
osNames	Specifies a comma-delimited list of operating systems. Specify null to have the action apply for each operating system. The valid operating system names include: windows, unix, and zos.
pidVarName	Specifies the Java process id variable name.
workingDir	Specifies the location in the file system from where the command is invoked. This argument is required if you specify a Java step argument.
usernameVar	Specifies a variable for the user name.
usernameVal	Specifies the value of the user name variable.
passwordVar	Specifies a variable for the password.
passwordVal	Specifies the value of the password variable.

createHealthAction command examples

Batch mode example usage:

- Using Jacl:

```
$AdminTask createHealthAction { -name myAction -java { -javaDir c:\java\bin -
executableTargetType EXECUTABLE_JAR
-executable test.jar -executableArgs "{arg1}{arg2a arg2b}" -osNames "windows, unix" -
pidVarName pvn
-workingDir c:\workingDir } }
```

Interactive mode example usage:

- Using Jacl:

```
$AdminTask createHealthAction {-interactive}
```

- Using Jython

```
AdminTask.createHealthAction('[interactive]')
```

modifyHealthAction command

Use the modifyHealthAction command to modify an existing health action.

Table 15. modifyHealthAction command arguments

Argument	Description
name (required)	Specifies a name for the health action that you want to modify.

Table 16. non-Java step arguments

Argument	Description
executable	Specifies the executable file path.
executableArgs	Specifies a string array of arguments to pass to the executable file.

workingDir	Specifies the location in the file system from where the command is invoked.
usernameVar	Specifies a variable for the user name.
usernameVal	Specifies the value of the user name variable.
passwordVar	Specifies a variable for the password.
passwordVal	Specifies the value of the password variable.

Table 17. Java step arguments

Argument	Description
javaDir	Specifies the location of your Java executable file, for example, <code>\${JAVA_HOME}</code> . You can specify a variable or a path.
executableTargetType	Specifies the file type that you are using for your Java executable file. The value can be <code>EXECUTABLE_JAR</code> or <code>JAVA_CLASS</code> .
executable	Specifies the executable file path.
executableArgs	Specifies a string array of arguments to pass to the executable file.
pidVarName	Specifies the Java process id variable name.
usernameVar	Specifies a variable for the user name.
usernameVal	Specifies the value of the user name variable.
passwordVar	Specifies a variable for the password.
passwordVal	Specifies the value of the password variable.
workingDir	Specifies the location in the file system from where the command is invoked.

modifyHealthAction command examples

Batch mode example usage:

```
$AdminTask modifyHealthAction { -name myAction -java { -javaDir c:\java\bin -executableTargetType EXECUTABLE_JAR -executable test.jar -executableArgs "{arg1}{arg2a arg2b}" -pidVarName pvn -workingDir c:\workingDir } }
```

Interactive mode example usage:

- Using Jacl:

```
$AdminTask modifyHealthAction {-interactive}
```

- Using Jython

```
AdminTask.modifyHealthAction('[interactive]')
```

deleteHealthAction command

The deleteHealthAction command removes a health action.

Table 18. deleteHealthAction command arguments

Argument	Description
name (required)	Specifies a name for the health action.

deleteHealthAction command examples

Batch mode example usage:

- Using Jacl:

```
$Admintask deleteHealthAction {-name myHealthAction}
```

- Using Jython:

```
AdminTask.deleteHealthAction('myHealthAction')
```

listHealthActions command

The listHealthActions command lists each of the defined health actions in the cell.

listHealthActions command examples

Batch mode example usage:

- Using Jacl:

```
$Admintask listHealthActions
```

- Using Jython:

executeHealthAction command

The executeHealthAction command runs the specified health action.

Table 19. executeHealthAction command arguments

Argument	Description
name (required)	Specifies the name of the health action that you want to run.
node (required)	Specifies the node on which to take the action.
server (required)	Specifies the server on which to take the action.
cell	Specifies the cell on which to run the health action.
timeout	Specifies a timeout value for running the health action.
variables	Specifies java.util.Properties values of variables.

executeHealthAction command examples**Batch mode example usage:**

- Using Jacl:

```
$AdminTask executeHealthAction {-name myHealthAction -node myNode -server myServer}
```

- Using Jython:

```
AdminTask.executeHealthAction ('[-name myHealthAction -node myNode -server myServer]')
```

Interactive mode example usage:

- Using Jacl:

```
$AdminTask executeHealthAction {-interactive}
```

- Using Jython:

```
AdminTask.executeHealthAction ('[interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Health management](#)

Related tasks:

[Configuring health management](#)

[Creating health policies](#)

[Creating health policy custom actions](#)

[Setting maintenance mode](#)

Related reference:

[Maintenance mode administrative tasks](#)

Maintenance mode administrative tasks

You can use the server maintenance mode commands to view or edit the maintenance mode of your servers without using the administrative console. To set maintenance mode on a node, you can use the NodeGroupManager MBean.

Use the following commands to set, disable, and check maintenance mode for your servers:

- [setMaintenanceMode](#)
- [unsetMaintenanceMode](#)
- [checkMode](#)

To set, disable, and check maintenance mode for a node:

- [Node maintenance mode](#)

setMaintenanceMode

The setMaintenanceMode command puts a server into maintenance mode.

Required parameters

-name

Specifies the name of the server that you want to put into maintenance mode.

Optional parameters

-mode

Mode is an optional parameter. If you do not specify a value, then the default is used.

Possible values:

break, affinity, or stop.

Specify break to break Hypertext Transfer Protocol (HTTP) or Session Initiation Protocol (SIP) affinity and keep the server running. Specify affinity to keep HTTP or SIP affinity active and keep the server running. Specify stop to stop the server immediately and put the server in affinity mode.

Default: affinity

Return value

The command returns void.

Batch mode example usage

- Using Jacl:

```
$AdminTask setMaintenanceMode xdnodel {-name test1 -mode break}
```

- Using Jython string:

```
AdminTask.setMaintenanceMode ('xdnodel', '[-name test1 -mode affinity]')
```

Interactive mode example usage

- Using Jacl:

```
AdminTask setMaintenanceMode {-interactive}
```

- Using Jython string:

```
AdminTask.setMaintenanceMode ('[-interactive]')
```

unsetMaintenanceMode

The unsetMaintenanceMode command takes the server out of maintenance mode.

Required parameters

-name

Specifies the name of the server that you want to take out of maintenance mode.

Return value

The command returns void.

Batch mode example usage

- Using Jacl:

```
$AdminTask unsetMaintenanceMode xdnodel {-name test1}
```

- Using Jython string:

```
AdminTask.unsetMaintenanceMode ('xdnodel','[-name test1]')
```

Interactive mode example usage

- Using Jacl:

```
AdminTask unsetMaintenanceMode {-interactive}
```

- Using Jython string:

```
AdminTask.unsetMaintenanceMode ('[-interactive]')
```

checkMode

The checkMode command checks whether a server is in maintenance mode.

Required parameters

-name

Specifies the name of the server that you want to check.

Return value

The command returns a string value that indicates the server mode.

Batch mode example usage

- Using Jacl:

```
$AdminTask checkMode xdnodel {-name test1}
```

- Using Jython string:

```
AdminTask.checkMode ('xdnodel','[-name test1]')
```

Interactive mode example usage

- Using Jacl:

```
AdminTask checkMode {-interactive}
```

- Using Jython string:

```
AdminTask.checkMode ('[-interactive]')
```

Node maintenance mode

You can use the NodeGroupManager MBean to set or unset maintenance mode for a node. See [NodeGroupManager MBean](#) for more information.

You can use the following example in wsadmin to set maintenance mode for a node:

```
set ngmMbean [$AdminControl queryNames WebSphere:*,type=NodeGroupManager,process=dmgr]  
$AdminControl invoke $ngmMbean setMaintenanceMode {nodeName true true}
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:
[Health management](#)

Related tasks:

[Setting maintenance mode](#)

[Creating health policies](#)

Related reference:

[Health policy administrative tasks](#)

Middleware application administrative tasks

You can use the Jacl or Jython scripting languages to create and manage middleware applications with the wsadmin tool.

The administrative tasks for middleware applications include the following commands:

- [addMiddlewareAppWebModule](#)
- [addMiddlewareTarget](#)
- [deployWasCEApp](#)
- [exportMiddlewareApp](#)
- [exportMiddlewareAppScript](#)
- [listMiddlewareAppEditions](#)
- [listMiddlewareApps](#)
- [listMiddlewareAppWebModules](#)
- [listMiddlewareTargets](#)
- [modifyMiddlewareAppWebModule](#)
- [modifyPHPApp](#)
- [modifyUnmanagedWebApp](#)
- [modifyWasCEApp](#)
- [removeMiddlewareAppWebModule](#)
- [removeMiddlewareTarget](#)
- [showMiddlewareApp](#)
- [installWasCEApp](#)
- [startWasCEApp](#)
- [stopWasCEApp](#)
- [undeployWasCEApp](#)
- [uninstallMiddlewareApp](#)
- [unregisterApp](#)

addMiddlewareAppWebModule

The `addMiddlewareAppWebModule` command adds a Web module to a middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

-moduleName

Specifies the name of a module. (String, required)

-virtualHost

Specifies the name of the virtual host, which enables a single host machine to resemble multiple host machines. (String, required)

-contextRoot

Specifies the path prefix that is associated with the application. Use this parameter to compose the URL that is needed to access the application files. (String, required)

Optional parameters

-edition

Specifies the name of the edition. This parameter is required if the specified application has more than one edition. (String, optional)

-cluster

Specifies the name of the cluster. This parameter adds a dynamic cluster as a deployment target. (String, optional)

-node

Specifies the name of the node. (String, optional)

-server

Specifies the name of the server. Issue both the server and node parameters to add a standalone server as a deployment target. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask addMiddlewareAppWebModule {-app myJ2EEApplication -moduleName myModule -contextRoot /MyJ2EEApp -virtualHost default_host -node AppServerNode1 -server tomcatServer1}
```

- Using Jython string:

```
AdminTask.addMiddlewareAppWebModule ('[-app myJ2EEApplication -moduleName myModule -contextRoot /MyJ2EEApp -virtualHost default_host -node AppServerNode1 -server tomcatServer1]')
```

- Using Jython list:

```
AdminTask.addMiddlewareAppWebModule (['-app', 'myJ2EEApplication', '-moduleName', 'myModule', '-contextRoot', '/MyJ2EEApp', '-virtualHost', 'default_host', '-node', 'AppServerNode1', '-server', 'tomcatServer1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addMiddlewareAppWebModule {-interactive}
```

- Using Jython string:

```
AdminTask.addMiddlewareAppWebModule ('[-interactive]')
```

- Using Jython list:

```
AdminTask.addMiddlewareAppWebModule (['-interactive'])
```

addMiddlewareTarget

The **addMiddlewareTarget** command adds a deployment target to a middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. This parameter is required if the specified application has more than one edition. (String, optional)

-module

Specifies the name of the module. Use this parameter to narrow the scope of a command to a single module. (String, optional)

-cluster

Specifies the name of the cluster. Use this parameter to add a dynamic cluster as a deployment target. (String, optional)

-node

Specifies the name of the node. (String, optional)

-server

Specifies the name of the server. Issue both the server and node parameters to add a standalone server as a deployment target. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask addMiddlewareTarget {-app myJ2EEApplication -node AppServerNode1 -server tomcatServer1}
```

- Using Jython string:

```
AdminTask.addMiddlewareTarget ('[-app myJ2EEApplication -node AppServerNode1 -server tomcatServer1]')
```

- Using Jython list:

```
AdminTask.addMiddlewareTarget (['-app', 'myJ2EEApplication', '-node', 'AppServerNode1', '-server', 'tomcatServer1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addMiddlewareTarget {-interactive}
```

- Using Jython string:

```
AdminTask.addMiddlewareTarget ('[-interactive]')
```

- Using Jython list:

```
AdminTask.addMiddlewareTarget (['-interactive'])
```

deployWASCEApp

The `deployWasCEApp` command adds a representation of a WebSphere® Application Server Community Edition application.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

-node

Specifies the name of the node. (String, required)

-server

Specifies the name of the server. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask deployWasCEApp {-app newSample -edition 2 -node nodeName -server WasCEServerRep}
```

- Using Jython string:

```
AdminTask.deployWasCEApp ('[-app newSample -edition 2 -node nodeName -server WasCEServerRep]')
```

- Using Jython list:

```
AdminTask.deployWasCEApp (['-app', 'newSample', '-edition', '2', '-node', 'nodeName', '-server', 'WasCEServerRep'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask deployWasCEApp {-interactive}
```

- Using Jython string:

```
AdminTask.deployWasCEApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.deployWasCEApp (['-interactive'])
```

exportMiddlewareApp

The **exportMiddlewareApp** command exports a middleware application to a directory.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

-dir

Specifies the directory in which the application is located. (String, required)

Optional parameters

-edition

Specifies the name of the edition. This parameter is required if the specified application has more than one edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask exportMiddlewareApp {-app sample -dir /home/user1}
```

- Using Jython string:

```
AdminTask.exportMiddlewareApp (['-app sample -dir /home/user1'])
```

- Using Jython list:

```
AdminTask.exportMiddlewareApp (['-app', 'sample', '-dir', '/home/user1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask exportMiddlewareApp {-interactive}
```

- Using Jython string:

```
AdminTask.exportMiddlewareApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.exportMiddlewareApp (['-interactive'])
```

exportMiddlewareAppScript

The **exportMiddlewareAppScript** command exports the scripts of a middleware application to a directory.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

-dir

Specifies the name of the directory in which the middleware application script is located. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

-type

Specifies the type of script. The valid types are SETUP and CLEANUP. Specify the type to narrow the scope of the command to a single script. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask exportMiddlewareAppScript {-app sample -dir /home/user1}
```

- Using Jython string:

```
AdminTask.exportMiddlewareAppScript ('[-app sample -dir /home/user1]')
```

- Using Jython list:

```
AdminTask.exportMiddlewareAppScript (['-app', 'sample', '-dir', '/home/user1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask exportMiddlewareAppScript {-interactive}
```

- Using Jython string:

```
AdminTask.exportMiddlewareAppScript ('[-interactive]')
```

- Using Jython list:

```
AdminTask.exportMiddlewareAppScript (['-interactive'])
```

listMiddlewareAppEditions

The `listMiddlewareAppEditions` command lists all the editions of a middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

Optional parameters

None.

Return value

The command returns a list of edition names.

Batch mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareAppEditions {-app sample}
```

- Using Jython string:

```
AdminTask.listMiddlewareAppEditions ('[-app sample]')
```

- Using Jython list:

```
AdminTask.listMiddlewareAppEditions (['-app', 'sample'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareAppEditions {-interactive}
```

- Using Jython string:

```
AdminTask.listMiddlewareAppEditions ('[-interactive]')
```

- Using Jython list:

```
AdminTask.listMiddlewareAppEditions (['-interactive'])
```

listMiddlewareApps

The **listMiddlewareApps** command list all of the middleware applications.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

Optional parameters

None.

Return value

The command returns a list of the middleware applications.

Batch mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareApps {-app sample}
```

- Using Jython string:

```
AdminTask.listMiddlewareApps ('[-app sample]')
```

- Using Jython list:

```
AdminTask.listMiddlewareApps (['-app', 'sample'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareApps {-interactive}
```

- Using Jython string:

```
AdminTask.listMiddlewareApps ('[-interactive]')
```

- Using Jython list:

```
AdminTask.listMiddlewareApps (['-interactive'])
```

listMiddlewareAppWebModules

The **listMiddlewareAppWebModules** command lists the Web modules of a middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

-edition

Specifies the name of the edition. (String, required)

Optional parameters

None.

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareAppWebModules {-app sample -edition 1}
```

- Using Jython string:

```
AdminTask.listMiddlewareAppWebModules (['-app sample -edition 1'])
```

- Using Jython list:

```
AdminTask.listMiddlewareAppWebModules (['-app', 'sample', '-edition', '1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareAppWebModules {-interactive}
```

Using Jython string:

```
AdminTask.listMiddlewareAppWebModules (['-interactive'])
```

- Using Jython list:

```
AdminTask.listMiddlewareAppWebModules (['-interactive'])
```

listMiddlewareTargets

The **listMiddlewareTargets** command lists the deployment targets for a middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

-module

Specifies the name of the module. (String, optional)

Return value

The command returns a list of deployment target objects.

Batch mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareTargets {-app sample}
```

- Using Jython string:

```
AdminTask.listMiddlewareTargets (['-app sample'])
```

- Using Jython list:

```
AdminTask.listMiddlewareTargets (['-app', 'sample'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareTargets {-interactive}
```

- Using Jython string:

```
AdminTask.listMiddlewareTargets (['-interactive'])
```

- Using Jython list:

```
AdminTask.listMiddlewareTargets (['-interactive'])
```

modifyMiddlewareAppWebModule

The **modifyMiddlewareAppWebModule** command modifies the virtual hosts of an application.

Target object

None.

Required parameters**-app**

Specifies the name of the application. (String, required)

Optional parameters**-edition**

Specifies the name of the edition. (String, optional)

-renameEdition

Specifies the new name of the edition. (String, optional)

-description

Specifies how the edition is modified. (String, optional)

Return value**Batch mode example usage**

- Using Jacl:

```
$AdminTask modifyMiddlewareAppWebModule {-app sample -edition 1 -renameEdition 2 -description Update Edition}
```

- Using Jython string:

```
AdminTask.modifyMiddlewareAppWebModule ('[-app sample -edition 1 -renameEdition 2 -description Update Edition]')
```

- Using Jython list:

```
AdminTask.modifyMiddlewareAppWebModule (['-app', 'sample', '-edition', '1', '-renameEdition', '2', '-description', 'Update Edition'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask modifyMiddlewareAppWebModule {-interactive}
```

- Using Jython string:

```
AdminTask.modifyMiddlewareAppWebModule ('[-interactive]')
```

- Using Jython list:

```
AdminTask.modifyMiddlewareAppWebModule (['-interactive'])
```

modifyPHPApp

The **modifyPHPApp** command updates a PHP application.

Target object

None.

Required parameters

-app

Specifies the name of the PHP application. (String, required)

Optional parameters

-edition

Specifies the edition of the PHP application. (String, optional)

-renameEdition

Specifies the new name of the edition. (String, optional)

-description

Specifies the description of the modified edition. (String, optional)

-archive

Specifies the path to the application archive file. (String, optional)

-setupScript

Specifies the path to the setup script file. (String, optional)

-cleanupScript

Specifies the path to the cleanup script file. (String, optional)

-contextRoot

Specifies the path prefix that is associated with the application. (String, optional)

-virtualHost

Specifies the name of the virtual host. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask modifyPHPApp {-app myPHPApplication -edition 1 -renameEdition 1.0.0}
```

- Using Jython string:

```
AdminTask.modifyPHPApp ('[-app myPHPApplication -edition 1 -renameEdition 1.0.0]')
```

- Using Jython list:

```
AdminTask.modifyPHPApp (['-app', 'myPHPApplication', '-edition', '1', '-renameEdition', '1.0.0'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask modifyPHPApp {-interactive}
```

- Using Jython string:

```
AdminTask.modifyPHPApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.modifyPHPApp (['-interactive'])
```

modifyUnmanagedWebApp

The **modifyUnmanagedWebApp** command updates an unmanaged Web application.

Target object

None.

Required parameters

-app

Specifies the name of the application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

-renameEdition

Specifies the new name of the edition. (String, optional)

-description

Specifies the description of the modified edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask modifyUnmanagedWebApp {-app sample -edition 1 -renameEdition 2 -description Update Edition}
```

- Using Jython string:

```
AdminTask.modifyUnmanagedWebApp (['-app sample -edition 1 -renameEdition 2 -description Update Edition'])
```

- Using Jython list:

```
AdminTask.modifyUnmanagedWebApp (['-app', 'sample', '-edition', '1', '-renameEdition', '2', '-description', 'Update Edition'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask modifyUnmanagedWebApp {-interactive}
```

- Using Jython string:

```
AdminTask.modifyUnmanagedWebApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.modifyUnmanagedWebApp (['-interactive'])
```

modifyWasCEApp

The **modifyWasCEApp** command updates a WebSphere Application Server Community Edition application.

Target object

None.

Required parameters

-app

Specifies the name of the application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

-renameEdition

Specifies the new name of the edition. (String, optional)

-description

Specifies the description of the edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask installWasCEApp {-app sample -edition 1 -renameEdition 2 -description Update Edition}
```

- Using Jython string:

```
AdminTask.modifyWasCEApp (['-app sample-edition 1 -renameEdition 2 -description Update Edition'])
```

- Using Jython list:

```
AdminTask.modifyWasCEApp (['-app', 'sample', '-edition', '1', 'renameEdition', '2', '-description', 'Update Edition'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask modifyWasCEApp {-interactive}
```

- Using Jython string:

```
AdminTask.modifyWasCEApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.modifyWasCEApp (['-interactive'])
```

removeMiddlewareAppWebModule

The **removeMiddlewareAppWebModule** command removes the Web module from the middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

-moduleName

Specifies the name of the Web module. (String, optional)

-cluster

Specifies the name of the cluster. (String, optional)

-node

Specifies the name of the node. (String, optional)

-server

Specifies the name of the server. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask removeMiddlewareAppWebModule {-app sample -edition 1 -moduleName myModule}
```

- Using Jython string:

```
AdminTask.removeMiddlewareAppWebModule (['-app sample -edition 1 -moduleName myModule'])
```

- Using Jython list:

```
AdminTask.removeMiddlewareAppWebModule (['-app', 'sample', '-edition', '1', '-moduleName', 'myModule'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeMiddlewareAppWebModule {-interactive}
```

- Using Jython string:

```
AdminTask.removeMiddlewareAppWebModule (['-interactive'])
```

- Using Jython list:

```
AdminTask.removeMiddlewareAppWebModule (['-interactive'])
```

removeMiddlewareTarget

The **removeMiddlewareTarget** command removes the deployment targets from the middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

-module

Specifies the name of the Web module. (String, optional)

-cluster

Specifies the name of the cluster. (String, optional)

-node

Specifies the name of the node. (String, optional)

-server

Specifies the name of the server. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask removeMiddlewareTarget {-app sample -cluster cluster1}
```

- Using Jython string:

```
AdminTask.removeMiddlewareTarget (['-app sample -cluster cluster1'])
```

- Using Jython list:

```
AdminTask.removeMiddlewareTarget (['-app', 'sample', '-cluster', 'cluster1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeMiddlewareTarget {-interactive}
```

- Using Jython string:

```
AdminTask.removeMiddlewareTarget (['-interactive'])
```

- Using Jython list:

```
AdminTask.removeMiddlewareTarget (['-interactive'])
```

showMiddlewareApp

The **showMiddlewareApp** command displays the attributes of the middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask showMiddlewareApp {-app sample}
```

- Using Jython string:

```
AdminTask.showMiddlewareApp (['-app sample'])
```

- Using Jython list:

```
AdminTask.showMiddlewareApp (['-app', 'sample'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask showMiddlewareApp {-interactive}
```

- Using Jython string:

```
AdminTask.showMiddlewareApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.showMiddlewareApp (['-interactive'])
```

installWasCEApp

The **installWasCEApp** command installs a WebSphere Application Server Community Edition application.

Target object

None.

Required parameters

-app

Specifies the name of the application. (String, required)

-archive

Specifies the location of the archive file. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

-description

Specifies the description of the edition.

-plan

Specifies the location of the external deployment plan.

-webModules: [[moduleName1 contextRoot1 virtualHost1][moduleName2 contextRoot2 virtualHost2]...]

Specifies the Web modules. Each **moduleName** value must match the name that is contained in the deployment descriptor file.

-clusterTargets

Specifies the cluster targets. Apply this parameter to the entire application.

-serverTargets

Specifies the server targets. Apply this parameter to the entire application.

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask installWasCEApp {-app sample -archive /tmp/sample.ear -edition 1 -serverTargets  
{{myserver01 WASCE_2.0.0.2_myserver01_DS_1}} -webModules {{samplemicrowebapp.war /B  
default_host}}
```

- Using Jython string:

```
AdminTask.installWasCEApp (['-app sample -archive /tmp/sample.ear -edition 1 -serverTargets  
[[myserver01 WASCE_2.0.0.2_myserver01_DS_1]] -webModules [[samplemicrowebapp.war /B  
default_host]]'])
```

- Using Jython list:

```
AdminTask.installWasCEApp (['-app', 'sample', '-archive', '/tmp/sample.ear', '-edition', '1',  
'-serverTargets', '[[myserver01 WASCE_2.0.0.2_myserver01_DS_1]]', '-webModules',
```

```
'[[samplemicrowebapp.war /B default_host]]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask installWasCEApp {-interactive}
```

- Using Jython string:

```
AdminTask.installWasCEApp ('[-interactive]')
```

- Using Jython list:

```
AdminTask.installWasCEApp (['-interactive'])
```

startWasCEApp

The **startWasCEApp** command starts a WebSphere Application Server Community Edition application.

Target object

None.

Required parameters

-app

Specifies the name of the application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask startWasCEApp {-app sample -edition 1}
```

- Using Jython string:

```
AdminTask.startWasCEApp ('[-app sample -edition 1]')
```

- Using Jython list:

```
AdminTask.startWasCEApp (['-app', 'sample', '-edition', '1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask startWasCEApp {-interactive}
```

- Using Jython string:

```
AdminTask.startWasCEApp ('[-interactive]')
```

- Using Jython list:

```
AdminTask.startWasCEApp (['-interactive'])
```

stopWasCEApp

The **stopWasCEApp** command stops a WebSphere Application Server Community Edition application.

Target object

None.

Required parameters

-app

Specifies the name of the application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask stopWasCEApp {-app sample -edition 1}
```

- Using Jython string:

```
AdminTask.stopWasCEApp (['-app sample -edition 1'])
```

- Using Jython list:

```
AdminTask.stopWasCEApp (['-app', 'sample', '-edition', '1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask stopWasCEApp {-interactive}
```

- Using Jython string:

```
AdminTask.stopWasCEApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.stopWasCEApp (['-interactive'])
```

undeployWasCEApp

The `undeployWasCEApp` command removes a representation of a WebSphere Application Server Community Edition application.

Target object

None.

Required parameters

-app

Specifies the name of the application. (String, required)

-node

Specifies the name of the node. (String, required)

-server

Specifies the name of the server. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask undeployWasCEApp {-app sample -edition 2 -node nodeName -server WasCEServerRep}
```

- Using Jython string:

```
AdminTask.undeployWasCEApp (['-app newSample -edition 2 -node nodeName -server WasCEServerRep'])
```

- Using Jython list:

```
AdminTask.undeployWasCEApp (['-app', 'newSample', '-edition', '2', '-node', 'nodeName', '-server', 'WasCEServerRep'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask undeployWasCEApp {-interactive}
```

- Using Jython string:

```
AdminTask.undeployWasCEApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.undeployWasCEApp (['-interactive'])
```

uninstallMiddlewareApp

The **uninstallMiddlewareApp** command uninstalls the middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask uninstallMiddlewareApp {-app sample -edition 1}
```

- Using Jython string:

```
AdminTask.uninstallMiddlewareApp (['-app sample -edition 1'])
```

- Using Jython list:

```
AdminTask.uninstallMiddlewareApp (['-app', 'sample', '-edition', '1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask uninstallMiddlewareApp {-interactive}
```

- Using Jython string:
-


```
AdminTask.uninstallMiddlewareApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.uninstallMiddlewareApp (['-interactive'])
```

unregisterApp

The **unregisterApp** command removes an unmanaged middleware application.

Target object

None.

Required parameters

-app

Specifies the name of the middleware application. (String, required)

Optional parameters

-edition

Specifies the name of the edition. (String, optional)

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask unregisterApp {-app sample -edition 1}
```

- Using Jython string:

```
AdminTask.unregisterApp (['-app sample -edition 1'])
```

- Using Jython list:

```
AdminTask.unregisterApp (['-app', 'sample', '-edition', '1'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask unregisterApp {-interactive}
```

- Using Jython string:

```
AdminTask.unregisterApp (['-interactive'])
```

- Using Jython list:

```
AdminTask.unregisterApp (['-interactive'])
```

Parent topic: [Administrator scripting interfaces](#)

Middleware descriptor administrative tasks

You can use the middleware descriptor administrative tasks to list the middleware descriptors and versions, display or change middleware descriptor information, and modify the middleware descriptor discovery interval.

Use the following commands to manage the middleware descriptors in your environment:

- [listMiddlewareDescriptors](#)
- [listMiddlewareDescriptorVersions](#)
- [showMiddlewareDescriptorInformation](#)
- [modifyMiddlewareDescriptorProperty](#)
- [modifyMiddlewareDescriptorDiscoveryInterval](#)

listMiddlewareDescriptors

The `listMiddlewareDescriptors` command lists the installed middleware descriptors in your cell.

Required parameters

None

Return value

The command returns a list of the middleware descriptors, for example:

```
jboss_server
application_server
phpRuntime
customhttp_server
wasceRuntime
apacheWebServerRuntime
apache_server
weblogic_server
tomcat_server
```

Batch mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareDescriptors
```

- Using Jython:

```
AdminTask.listMiddlewareDescriptors
```

listMiddlewareDescriptorVersions

The `listMiddlewareDescriptorVersions` command lists which versions have specific information provided in the middleware descriptors.

Required parameters

- **-name:** Specifies the name of the middleware descriptor. (String, required)

Return value

A list of versions that have specific information provided in the middleware descriptor. For example:

```
default
```

Batch mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareDescriptorVersions {-name wasceRuntime}
```

- Using Jython:

```
AdminTask.listMiddlewareDescriptorVersions ('[-name wasceRuntime]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareDescriptorVersions {-interactive}
```

- Using Jython:

```
AdminTask.listMiddlewareDescriptorVersions ([-interactive]')
```

showMiddlewareDescriptorInformation

The **showMiddlewareDescriptorInformation** command displays the contents of the specified middleware descriptor.

Required parameters

- **-name:** Specifies the name of the middleware descriptor. (String, required)
-

Return value

An example of the command output follows:

```
wasceRuntime:
  Discovery Interval: 4 min

  Version: default
    install.locations:win.install.loc="C:\\Program
Files\\IBM\\WebSphere\\AppServerCommunityEdition;C:\\Program Files
(x86)\\WebSphere\\AppServerCommunityEdition"

install.locations:unix.install.loc="/opt/IBM/WebSphere/AppServerCommunityEdition"

install.locations:foreign.discovery.class="com.ibm.ws.xd.agent.discovery.wasce.WASCEDiscoveryPlugin"
  timeOutGroup:startTimeOutValue="300000"
  timeOutGroup:stopTimeOutValue="300000"
```

Batch mode example usage

- Using Jacl:

```
$AdminTask showMiddlewareDescriptorInformation {-name wasceRuntime}
```

- Using Jython:

```
AdminTask.showMiddlewareDescriptorInformation ('[-name wasceRuntime]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask showMiddlewareDescriptorInformation {-interactive}
```

- Using Jython:

```
AdminTask.showMiddlewareDescriptorInformation ([-interactive]')
```

modifyMiddlewareDescriptorProperty

You can use the **modifyMiddlewareDescriptorProperty** command to edit the middleware descriptor properties.

Required parameters

- **-name:** Specifies the name of the middleware descriptor.
- **-version:** Specifies the middleware descriptor version.
- **-propName:** Specifies a property for the middleware descriptor. This property name can be any property that displays when you run the **showMiddlewareDescriptorInformation** command for the middleware descriptor.
- **-propValue:** Specifies a value for the middleware descriptor property.

Return value

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask modifyMiddlewareDescriptorProperty {-name wasceRuntime
-version default -propName install.locations:win.install.loc -propValue
```

```
c:\blade\server1 }
```

- Using Jython:

```
AdminTask.modifyMiddlewareDescriptorProperty ('[-name wasceRuntime -version default -propName  
install.locations:win.install.loc -propValue  
c:\blade\server1]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask modifyMiddlewareDescriptorProperty {-interactive}
```

- Using Jython:

```
AdminTask.modifyMiddlewareDescriptorProperty ([-interactive]')
```

modifyMiddlewareDescriptorDiscoveryInterval

You can use the **modifyMiddlewareDescriptorDiscoveryInterval** command to modify the discovery interval of the selected middleware descriptor.

Required parameters

- **-name:** Specifies the name of the middleware descriptor.
- **-interval:** Specifies the integer value of the discovery interval. You can use one of the following values:
 - **-1:** Disables automatic discovery.
 - **0 :** Performs middleware discovery when the middleware agent starts.
 - **value greater than 0:** Specifies that middleware discovery runs when the middleware agent starts and at the integer value interval.
- **-units:** Specifies the units for the discovery interval. You can use one of the following values:
 - **seconds**
 - **minutes**
 - **hours**
 - **days**

Return value

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask modifyMiddlewareDescriptorDiscoveryInterval {-name wasceRuntime  
-interval 5 -units seconds}
```

- Using Jython:

```
AdminTask.modifyMiddlewareDescriptorDiscoveryInterval ('[-name wasceRuntime -interval 5 -units  
seconds]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask modifyMiddlewareDescriptorProperty {-interactive}
```

- Using Jython:

```
AdminTask.modifyMiddlewareDescriptorProperty ([-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related tasks:

[Configuring middleware descriptors](#)

Middleware server creation administrative tasks

You can use administrative tasks to write a script that can recreate your middleware server configuration.

Use the following commands to create middleware servers:

- [createTomCatServer](#)
- [createWebLogicServer](#)
- [createJBossServer](#)
- [createForeignServer](#)
- [createWasCEServer](#)

createTomCatServer

The createTomCatServer command creates a representation of an Apache Tomcat server.

Target object

The node name.

Required parameters

-name

Specifies the name of the server to create. (String, required)

-templateName

Specifies the name of the template that is used to create the server.

-genUniquePorts

Specifies a parameter to generate unique HTTP ports for a server. (Boolean)

-templateLocation

Specifies the location where the template is stored. Use the system defined location if it is not specified. Using the system defined location is recommended.

z/OS

-specificShortName

Specifies the specific short name of the server. All servers should have unique specific short name. This parameter is optional and when it is not specified a unique specific short name is automatically assigned. The value must be eight characters or less and all upper case.

z/OS

-genericShortName

Specifies the generic short name of the server. All members of a cluster must have the same generic short name. Individual servers must have a unique generic short name. This parameter is optional. When this parameter is not specified, a unique generic short name is automatically assigned. The value must be eight characters or less and all upper case.

-clusterName

Specifies the name of the cluster for this server.

Return value

The command returns a list of the middleware servers for the specified type.

Batch mode example usage

- Using Jacl:

```
$AdminTask createTomCatServer nodename {-name ServerName}
```

- Using Jython:

```
AdminTask.createTomCatServer ('nodename', ['-name ServerName'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createTomCatServer {-interactive}
```

- Using Jython:

```
AdminTask.createTomCatServer ([-interactive])
```

createWebLogicServer

The createWebLogicServer command creates a representation of a BEA WebLogic Server.

Target object

The node name.

Required parameters

-name

Specifies the name of the server to create. (String, required)

-templateName

Specifies the name of the template that is used to create the server.

-genUniquePorts

Specifies a parameter to generate unique HTTP ports for a server. (Boolean)

-templateLocation

Specifies the location where the template is stored. Use the system defined location if it is not specified. Using the system defined location is recommended.

z/OS

-specificShortName

Specifies the specific short name of the server. All servers should have unique specific short name. This parameter is optional and when it is not specified a unique specific short name is automatically assigned. The value must be eight characters or less and all upper case.

z/OS

-genericShortName

Specifies the generic short name of the server. All members of a cluster must have the same generic short name. Individual servers must have a unique generic short name. This parameter is optional. When this parameter is not specified, a unique generic short name is automatically assigned. The value must be eight characters or less and all upper case.

-clusterName

Specifies the name of the cluster for this server.

Return value

The command returns a list of the middleware servers for the specified type.

Batch mode example usage

- Using Jacl:

```
$AdminTask createWebLogicServer nodename {-name ServerName}
```

- Using Jython:

```
AdminTask.createWebLogicServer ('nodename', ['-name ServerName'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createWebLogicServer {-interactive}
```

- Using Jython:

```
AdminTask.createWebLogicServer ([-interactive])
```

createJBossServer

The createJBossServer command creates a representation of a JBoss server.

Target object

The node name.

Required parameters

-name

Specifies the name of the server to create. (String, required)

-templateName

Specifies the name of the template that is used to create the server.

-genUniquePorts

Specifies a parameter to generate unique HTTP ports for a server. (Boolean)

-templateLocation

Specifies the location where the template is stored. Use the system defined location if it is not specified. Using the system defined location is recommended.

z/OS **-specificShortName**

Specifies the specific short name of the server. All servers should have unique specific short name. This parameter is optional and when it is not specified a unique specific short name is automatically assigned. The value must be eight characters or less and all upper case.

z/OS **-genericShortName**

Specifies the generic short name of the server. All members of a cluster must have the same generic short name. Individual servers must have a unique generic short name. This parameter is optional. When this parameter is not specified, a unique generic short name is automatically assigned. The value must be eight characters or less and all upper case.

-clusterName

Specifies the name of the cluster for this server.

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask createJBossServer nodename {-name ServerName}
```

- Using Jython:

```
AdminTask.createJBossServer ('nodename', ['-name ServerName'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createJBossServer (-interactive)
```

- Using Jython:

```
AdminTask.createJBossServer ([-interactive])
```

createForeignServer

The createForeignServer command creates a new foreign server.

Target object

The node name.

Required parameters

-name

Specifies the name of the server to create. (String, required)

-templateName

Specifies the name of the template that is used to create the server.

-genUniquePorts

Specifies a parameter to generate unique HTTP ports for a server. (Boolean)

-templateLocation

Specifies the location where the template is stored. Use the system defined location if it is not specified. Using the system defined location is recommended.

z/OS **-specificShortName**

Specifies the specific short name of the server. All servers should have unique specific short name. This parameter is optional and when it is not specified a unique specific short name is automatically assigned. The value must be eight characters or less and all upper case.

z/OS **-genericShortName**

Specifies the generic short name of the server. All members of a cluster must have the same generic short name. Individual servers must have a unique generic short name. This parameter is optional. When this parameter is not specified, a unique generic short name is automatically assigned. The value must be eight characters or less and all upper case.

-clusterName

Specifies the name of the cluster for this server.

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask createForeignServer nodename {-name ServerName}
```

- Using Jython:

```
AdminTask.createForeignServer ('nodename', '[-name ServerName]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createForeignServer {-interactive}
```

- Using Jython:

```
AdminTask.createForeignServer ([-interactive]')
```

createPHPServer

The createPHPServer command creates a new PHP server.

Required parameters

-name

Specifies the name of the server. (String, required)

Optional parameters

-templateName

Specifies the name of the server template to use. You can specify templateServerNode and templateServerName to use an existing server as a virtual template instead of a defined template. (String, optional)

-genUniquePorts

Specifies a boolean to generate unique ports for the server. (String, optional)

-templateLocation

Specifies the location in which the template is stored. Use the system defined location if the location is not specified. (String, optional)

-serverConfig

Specifies the configuration of the server definition properties. Specify phpServerRoot or apacheServerRoot for the runtime path. Specify phpVersion or apacheVersion for the version number. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask createPHPServer workstation_nameNode01 {-name myphpserver -serverConfig {-phpPath C:/PHP5 -apachePath C:/apache2}}
```

- Using Jython:

```
AdminTask.createPHPServer workstation_nameNode01 ('[-name myphpserver -serverConfig [-phpPath C:/PHP5 -apachePath C:/apache2]]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createPHPServer workstation_nameNode01 {-interactive}
```

- Using Jython:

```
AdminTask.createPHPServer workstation_nameNode01 ('[-interactive]')
```

createWasCEServer

The createWasCEServer command creates a representation of a WebSphere® Application Server Community Edition server.

Target object

None.

Required parameters

-name

Specifies the name of the server to create. (String, required)

-templateName

Specifies the name of the template that is used to create the server.

-genUniquePorts

Specifies a parameter to generate unique HTTP ports for a server. (Boolean)

-templateLocation

Specifies the location where the template is stored. Use the system defined location if it is not specified. Using the system defined location is recommended.

z/OS

-specificShortName

Specifies the specific short name of the server. All servers should have unique specific short name. This parameter is optional and when it is not specified a unique specific short name is automatically assigned. The value must be eight characters or less and all upper case.

z/OS

-genericShortName

Specifies the generic short name of the server. All members of a cluster must have the same generic short name. Individual servers must have a unique generic short name. This parameter is optional. When this parameter is not specified, a unique generic short name is automatically assigned. The value must be eight characters or less and all upper case.

-clusterName

Specifies the name of the cluster for this server.

-assistedServer

Specifies if the server that you create is an assisted life cycle server that is a representation of an existing server. The default is false. If you specify false, a complete life cycle server is created. If you specify true, an assisted life cycle server is created.

-templateServerNode

Specifies the name of the node that contains the server that you want to use as a template.

-templateServerName

Specifies the name of the existing server that you want to use as a template.

Return value

Batch mode example usage

- Using Jacl:

```
$AdminTask createWasCEServer nodename {-name ServerName -templateName wasce21}
```

- Using Jython:

```
AdminTask.createWasCEServer ('nodename', ['-name ServerName -templateName wasce21'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createWasCEServer {-interactive}
```

- Using Jython:

```
AdminTask.createWasCEServer ([-interactive])
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Adding middleware servers to configurations](#)

Middleware server management administrative tasks

You can use the Jython and Jacl scripting languages to manage middleware servers with the wsadmin tool. Use the commands and parameters in the MiddlewareServerManagement group.

Use the following commands to work with middleware servers:

- [getMiddlewareServerType](#)
- [listForeignServerTypes](#)
- [listMiddlewareServers](#)
- [listMiddlewareServerTypes](#)
- [listWASServerTypes](#)
- [modifyForeignServerProperty](#)
- [showMiddlewareServerInfo](#)
- [startMiddlewareServer](#)
- [stopMiddlewareServer](#)
- [executeMiddlewareServerOperation](#)

getMiddlewareServerType

The getMiddlewareServerType command lists the middleware server types.

Target object

None.

Required parameters

-serverName

Specifies the name of the server for which you want to display the type. (String, required)

-nodeName

Specifies the node on which the server runs. (String, required)

Return value

The command returns a list of the middleware servers for the specified type.

Batch mode example usage

- Using Jacl:

```
$AdminTask getMiddlewareServerType {-serverName myserver -nodeName mynode}
```

- Using Jython string:

```
AdminTask.getMiddlewareServerType(['-serverName myserver -nodeName mynode'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask stopMiddlewareServer {-interactive}
```

- Using Jython string:

```
AdminTask.stopMiddlewareServer (['-interactive'])
```

listForeignServerTypes

The listForeignServerTypes command lists all of the supported middleware server types, not including any middleware servers associated with WebSphere® Virtual Enterprise.

Target object

Return value

The command returns a list of middleware server types, for example:

```
PHP_SERVER
```

```
WASCE_SERVER
CUSTOMHTTP_SERVER
APACHE_SERVER
TOMCAT_SERVER
WEBLOGIC_SERVER
JBOSS_SERVER
WASAPP_SERVER
```

Batch mode example usage

- Using Jacl:

```
$AdminTask listForeignServerTypes
```

- Using Jython string:

```
AdminTask.listForeignServerTypes
```

listMiddlewareServers

The listMiddlewareServers command lists the middleware servers and the path to the configuration file for the servers. If you do not pass node or middleware server type parameters, then all of the servers are displayed.

Target object

None.

Optional parameters

-middlewareServerType

Specifies the type of middleware server to list. The middleware server types include: TOMCAT_SERVER, PHP_SERVER, WEBLOGIC_SERVER, JBOSS_SERVER, WASCE_SERVER, APACHE_SERVER, and CUSTOMHTTP_SERVER. (String, optional)

-nodeName

Specifies the node for which to list the servers. (String, optional)

Return value

The command returns the list of the middleware servers for the type that you indicated. For example, you might see the following output:

```
nodeagent(cells/xdblade02b07/nodes/xdblade01b08/servers/nodeagent|server.xml)
WASCE_2.0.0.0_xdblade01b08_DS_2(cells/xdblade02b07/nodes/xdblade01b08/servers/WASCE_2.0.0.0_xdblade01b08_DS_2|server.xml)
WASCE_2.0.0.0_xdblade01b08_DS_1(cells/xdblade02b07/nodes/xdblade01b08/servers/WASCE_2.0.0.0_xdblade01b08_DS_1|server.xml)
WASCE_2.0.0.0_xdblade01b08_DS_4(cells/xdblade02b07/nodes/xdblade01b08/servers/WASCE_2.0.0.0_xdblade01b08_DS_4|server.xml)
frucg(cells/xdblade02b07/nodes/xdblade01b08/servers/frucg|server.xml)
WASCE_2.0.0.0_xdblade01b08_DD_1(cells/xdblade02b07/nodes/xdblade01b08/servers/WASCE_2.0.0.0_xdblade01b08_DD_1|server.xml)
dmgr(cells/xdblade02b07/nodes/xdblade02b07/servers/dmgr|server.xml)
middlewareagent(cells/xdblade02b07/nodes/thebe14/servers/middlewareagent|server.xml)
tttomcat_server(cells/xdblade02b07/nodes/thebe14/servers/tttomcat_server|server.xml)
```

Batch mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareServers {-middlewareServerType TOMCAT_SERVER}
```

- Using Jython string:

```
AdminTask.listMiddlewareServers(['-middlewareServerType TOMCAT_SERVER'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareServers
```

- Using Jython string:

```
listMiddlewareServers( )
```

listMiddlewareServerTypes

The listMiddlewareServerTypes command lists all of the server types.

Target object

Return value

The command returns a list of all of the server types, for example:

```
APPLICATION_SERVER  
PHP_SERVER  
WASCE_SERVER  
CUSTOMHTTP_SERVER  
APACHE_SERVER  
TOMCAT_SERVER  
WEBLOGIC_SERVER  
ONDEMAND_ROUTER  
PROXY_SERVER  
WEB_SERVER  
JBOSS_SERVER  
WASAPP_SERVER  
GENERIC_SERVER
```

Batch mode example usage

- Using Jacl:

```
$AdminTask listMiddlewareServerTypes
```

- Using Jython string:

```
AdminTask.listMiddlewareServerTypes
```

Interactive mode example usage

listWASServerTypes

The listWASServerTypes command lists only the server types that are associated with WebSphere Application Server.

Target object

Return value

The command returns a list of server types, for example:

```
APPLICATION_SERVER  
ONDEMAND_ROUTER  
PROXY_SERVER  
WEB_SERVER  
GENERIC_SERVER
```

Batch mode example usage

- Using Jacl:

```
$AdminTask listWASServerTypes
```

- Using Jython string:

```
AdminTask.listWASServerTypes
```

modifyForeignServerProperty

The `modifyForeignServerProperty` command modifies a property on a middleware server.

Target object

None.

Required parameters

-serverName

Specifies the name of the server. (String, required)

-nodeName

Specifies the name of the node. (String, required)

-propKey

Specifies the property key that is associated with the `server.xml` file. (String, required)

-propValue

Specifies the property value that you want to set in the `server.xml` file. (String, required)

Return value

The command returns

Batch mode example usage

- Using Jacl:

```
$AdminTask modifyForeignServerProperty {-serverName WASCE_2.0.0.0_xdblade01b08_DS_4 -nodeName xdblade01b08 -propKey port -propValue 9090 }
```

- Using Jython string:

```
AdminTask.modifyForeignServerProperty(['-serverName WASCE_2.0.0.0_xdblade01b08_DS_4 -nodeName xdblade01b08 -propKey port -propValue 9090'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask modifyForeignServerProperty {-interactive}
```

- Using Jython string:

```
AdminTask.modifyForeignServerProperty(['-interactive'])
```

showMiddlewareServerInfo

The `showMiddlewareServerInfo` command displays a list of properties for the middleware server.

Return value

The command returns

Required parameters

server_name(path_to_configuration_file|configuration_file_name)

This parameter determines the server for which you want to display the information. For example, you might use the following value for this parameter:

```
odr2(cells/SVT61/nodes/oberon04/servers/odr2|server.xml)
```

You can determine the value of the `path_to_configuration_file` variable and the `configuration_file_name` value by running the `listMiddlewareServers` command. You can use the output of that command as the parameter value.

Batch mode example usage

- Using Jacl:

```
$AdminTask showMiddlewareServerInfo odr2(cells/SVT61/nodes/oberon04/servers/odr2|server.xml)
```

- Using Jython string:

```
AdminTask.showMiddlewareServerInfo(['odr2(cells/SVT61/nodes/oberon04/servers/odr2|server.xml)'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask showMiddlewareServerInfo {-interactive}
```

- Using Jython string:

```
AdminTask.showMiddlewareServerInfo(['-interactive'])
```

Example output

```
{cell SVT61}
{serverMiddlewareType ONDEMAND_ROUTER}
{com.ibm.websphere.baseProductVersion 6.1.0.11}
{node oberon04}
{server odr2}
```

startMiddlewareServer

The startMiddlewareServer command starts a specified middleware server.

Target object

None.

Required parameters

-serverName

Specifies the name of the server to start. (String, required)

-nodeName

Specifies the node on which the server that you want to start runs. (String, required)

Return value

The command returns

Batch mode example usage

- Using Jacl:

```
$AdminTask startMiddlewareServer {-serverName myserver -nodeName mynode}
```

- Using Jython string:

```
AdminTask.startMiddlewareServer(['-name myserver -node mynode'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask startMiddlewareServer{-interactive}
```

- Using Jython string:

```
AdminTask.startMiddlewareServer(['-interactive'])
```

stopMiddlewareServer

The stopMiddlewareServer command stops the specified middleware server.

Target object

None.

Required parameters

-serverName

Specifies the name of the middleware server to stop. (String, required)

-nodeName

Specifies the node on which the server that you want to stop runs. (String, required)

Batch mode example usage

- Using Jacl:

```
$AdminTask stopMiddlewareServer {-serverName myserver -nodeName mynode}
```

- Using Jython string:

```
AdminTask.stopMiddlewareServer('[-name myserver -node mynode]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask stopMiddlewareServer {-interactive}
```

- Using Jython string:

```
AdminTask.stopMiddlewareServer ('[-interactive]')
```

executeMiddlewareServerOperation

The executeMiddlewareServerOperation command runs an existing server operation on a specified server.

Target object

None.

Required parameters

-serverName

Specifies the name of the server on which to run the server operation. (String, required)

-nodeName

Specifies the name of the node on which the specified server runs. (String, required)

-operation

Specifies the name of the server operation that you want to run on the server. (String, required)

Return value

The command returns a status message for the server operation, for example:

```
CWMWS0420I: Start completed for middleware server  
"WASCE_2.0.0.0_xdblade01b08_DS_4" on node "xdblade01b08"
```

Batch mode example usage

- Using Jacl:

```
$AdminTask executeMiddlewareServerOperation {-serverName WASCE_2.0.0.0_xdblade01b08_DS_4 -  
nodeName xdblade01b08 -operation start }
```

- Using Jython string:

```
AdminTask.executeMiddlewareServerOperation('[-serverName WASCE_2.0.0.0_xdblade01b08_DS_4 -  
nodeName xdblade01b08 -operation start]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask executeMiddlewareServerOperation {-interactive}
```

- Using Jython string:

```
AdminTask.executeMiddlewareServerOperation('[-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Middleware server template administrative tasks

Use middleware server template administrative tasks to create server templates for middleware servers that you have created.

Use the following commands to create middleware server templates:

- [createApacheServerTemplate command](#)
- [createForeignServerTemplate command](#)
- [createBossServerTemplate command](#)
- [createPHPServerTemplate](#)
- [createTomCatServerTemplate command](#)
- [createWasCEServerTemplate command](#)
- [createWebLogicServerTemplate command](#)

createApacheServerTemplate command

The createApacheServerTemplate command creates a new server template that is based on an existing Apache server representation.

Target object

Required parameters

- templateName**
Specifies the name of the server template that you want to create. (String, required)
- serverName**
Specifies the server from which to base the template. (String, required)
- nodeName**
Specifies the node on which the specified server runs. (String, required)
- description**
Specifies a description for the template.
- templateLocation**
Specifies the location where you want to place the template.

Return value

This command returns the configuration ID of a new template.

Batch mode example usage

- Using Jacl:

```
$AdminTask createApacheServerTemplate{-templateName weblogic_new -serverName xdtest1 -nodeName xdnodel -description "My new template"}
```

- Using Jython String:

```
AdminTask.createApacheServerTemplate(['-templateName weblogic_new -serverName xdtest1 -nodeName xdnodel -description "My new template"'])
```

- Using Jython List:

```
AdminTask.createApacheServerTemplate(['-templateName', 'weblogic_new', '-serverName', 'xdtest1', '-nodeName', 'xdnodel', '-description', 'My new template'])
```

createForeignServerTemplate command

The createForeignServerTemplate command creates a new server template that is based on an existing custom HTTP server representation.

Target object

Required parameters

- templateName**
Specifies the name of the server template that you want to create. (String, required)
- serverName**
Specifies the server from which to base the template. (String, required)
- nodeName**

Specifies the node on which the specified server runs. (String, required)

-description

Specifies a description for the template.

-templateLocation

Specifies the location where you want to place the template.

Return value

This command returns the configuration ID of a new template.

Batch mode example usage

- Using Jacl:

```
$AdminTask createForeignServerTemplate{-templateName weblogic_new -serverName xdtest1 -
nodeName xdnodel -description "My new template"}
```

- Using Jython String:

```
AdminTask.createForeignServerTemplate(['-templateName weblogic_new -serverName xdtest1 -
nodeName xdnodel -description "My new template"'])
```

- Using Jython List:

```
AdminTask.createForeignServerTemplate(['-templateName','weblogic_new','-
serverName','xdtest1','-nodeName','xdnodel','-description','My new template'])
```

createJBossServerTemplate command

The createJBossServerTemplate command creates a new server template that is based on an existing JBoss server representation.

Target object

Required parameters

-templateName

Specifies the name of the server template that you want to create. (String, required)

-serverName

Specifies the server from which to base the template. (String, required)

-nodeName

Specifies the node on which the specified server runs. (String, required)

-description

Specifies a description for the template.

-templateLocation

Specifies the location where you want to place the template.

Return value

This command returns the configuration ID of a new template.

Batch mode example usage

- Using Jacl:

```
$AdminTask createJBossServerTemplate{-templateName jboss_new -serverName xdtest1 -nodeName
xdnodel -description "My new template"}
```

- Using Jython String:

```
AdminTask.createJBossServerTemplate(['-templateName jboss_new -serverName xdtest1 -nodeName
xdnodel -description "My new template"'])
```

- Using Jython List:

```
AdminTask.createJBossServerTemplate(['-templateName','jboss_new','-serverName','xdtest1','-
nodeName','xdnodel','-description','My new template'])
```


createPHPSTemplate

The createPHPSTemplate command creates a PHP server template.

Required parameters

-templateName

Specifies the name of the template to create. (String, required)

-serverName

Specifies the name of the server to use as a template. (String, required)

-nodeName

Specifies the name of the node on which the server is located. (String, required)

Optional parameters

-description

Specifies a description for the server template. (String, optional)

-templateLocation

Specifies the location in which the template is stored. Use the system defined location if the location is not specified. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask createPHPSTemplate {-templateName myphpTemplate -nodeName workstation_nameNode01 -serverName phpserver}
```

- Using Jython:

```
AdminTask.createPHPSTemplate ('[-templateName myphpTemplate -nodeName workstation_nameNode01 -serverName phpserver]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createPHPSTemplate {-interactive}
```

- Using Jython:

```
AdminTask.createPHPSTemplate ('[-interactive]')
```

createTomcatServerTemplate command

The createTomcatServerTemplate command creates a new server template that is based on an existing Apache Tomcat server representation.

Target object

Required parameters

-templateName

Specifies the name of the server template that you want to create. (String, required)

-serverName

Specifies the server from which to base the template. (String, required)

-nodeName

Specifies the node on which the specified server runs. (String, required)

-description

Specifies a description for the template.

-templateLocation

Specifies the location where you want to place the template.

Return value

This command returns the configuration ID of a new template.

Batch mode example usage

- Using Jacl:

```
$AdminTask createTomCatServerTemplate{-templateName tomcat_new -serverName xdtest1 -nodeName xdnodel -description "My new template"}
```

- Using Jython String:

```
AdminTask.createTomCatServerTemplate(['-templateName tomcat_new -serverName xdtest1 -nodeName xdnodel -description "My new template"'])
```

- Using Jython List:

```
AdminTask.createTomCatServerTemplate(['-templateName', 'tomcat_new', '-serverName', 'xdtest1', '-nodeName', 'xdnodel', '-description', 'My new template'])
```

createWasCEServerTemplate command

The createWasCEServerTemplate command creates a new server template that is based on an existing WebSphere® Application Server Community Edition server representation.

Target object

Required parameters

-templateName

Specifies the name of the server template that you want to create. (String, required)

-serverName

Specifies the server from which to base the template. (String, required)

-nodeName

Specifies the node on which the specified server runs. (String, required)

-description

Specifies a description for the template.

-templateLocation

Specifies the location where you want to place the template.

Return value

This command returns the configuration ID of a new template.

Batch mode example usage

- Using Jacl:

```
$AdminTask createWasCEServerTemplate{-templateName wasce_new -serverName xdtest1 -nodeName xdnodel -description "My new template"}
```

- Using Jython String:

```
AdminTask.createWasCEServerTemplate(['-templateName wasce_new -serverName xdtest1 -nodeName xdnodel -description "My new template"'])
```

- Using Jython List:

```
AdminTask.createWasCEServerTemplate(['-templateName', 'wasce_new', '-serverName', 'xdtest1', '-nodeName', 'xdnodel', '-description', 'My new template'])
```

createWebLogicServerTemplate command

The createWebLogicServerTemplate command creates a new server template that is based on an existing BEA WebLogic server representation.

Target object

Required parameters

-templateName

Specifies the name of the server template that you want to create. (String, required)

-serverName

Specifies the server from which to base the template. (String, required)

-nodeName

Specifies the node on which the specified server runs. (String, required)

-description

Specifies a description for the template.

-templateLocation

Specifies the location where you want to place the template.

Return value

This command returns the configuration ID of a new template.

Batch mode example usage

- Using Jacl:

```
$AdminTask createWebLogicServerTemplate{-templateName weblogic_new -serverName xdtest1 -nodeName xdnodel -description "My new template"}
```

- Using Jython String:

```
AdminTask.createWebLogicServerTemplate(['-templateName weblogic_new -serverName xdtest1 -nodeName xdnodel -description "My new template"'])
```

- Using Jython List:

```
AdminTask.createWebLogicServerTemplate(['-templateName','weblogic_new','-serverName','xdtest1','-nodeName','xdnodel','-description','My new template'])
```

Parent topic: [Administrator scripting interfaces](#)

Related tasks:

[Creating middleware server templates](#)

Related reference:

[Middleware server creation administrative tasks](#)

PHP server administrative tasks

Use the following commands to create and configure PHP servers and server templates.

The administrative tasks for creating and configuring PHP servers include the following commands:

- [createPHPServer](#)
- [listPHPServers](#)
- [startServer](#)
- [stopServer](#)
- [deleteServer](#)
- [createPHPServerTemplate](#)
- [deleteServerTemplate](#)

createPHPServer

The createPHPServer command creates a new PHP server.

Required parameters

-name

Specifies the name of the server. (String, required)

Optional parameters

-templateName

Specifies the name of the server template to use. You can specify `templateServerNode` and `templateServerName` to use an existing server as a virtual template instead of a defined template. (String, optional)

-genUniquePorts

Specifies a boolean to generate unique ports for the server. (String, optional)

-templateLocation

Specifies the location in which the template is stored. Use the system defined location if the location is not specified. (String, optional)

-serverConfig

Specifies the configuration of the server definition properties. Specify `phpServerRoot` or `apacheServerRoot` for the runtime path. Specify `phpVersion` or `apacheVersion` for the version number. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask createPHPServer workstation_nameNode01 {-name myphpserver -serverConfig {-phpPath C:/PHP5 -apachePath C:/apache2}}
```

- Using Jython:

```
AdminTask.createPHPServer workstation_nameNode01 ('[-name myphpserver -serverConfig [-phpPath C:/PHP5 -apachePath C:/apache2]]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createPHPServer workstation_nameNode01 {-interactive}
```

- Using Jython:

```
AdminTask.createPHPServer workstation_nameNode01 ('[-interactive]')
```

listPHPServers

The listPHPServers command lists all of the PHP servers.

Required parameters

None.

Optional parameters

-nodeName

Specifies the name of the node. Specify this parameter to narrowly scope the command. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask listPHPServers {}
```

- Using Jython:

```
AdminTask.listPHPServers ([''])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listPHPServers {-interactive}
```

- Using Jython:

```
AdminTask.listPHPServers (['-interactive'])
```

startServer

The startServer command starts a PHP server.

Required parameters

-serverName

Specifies the name of the server to start. (String, required)

-nodeName

Specifies the name of the node on which the server is located. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask startServer {-nodeName workstation_nameNode01 -serverName phpserver}
```

- Using Jython:

```
AdminTask.startServer (['-nodeName workstation_nameNode01 -serverName phpserver'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask startServer {-interactive}
```

- Using Jython:

```
AdminTask.startServer (['-interactive'])
```

stopServer

The stopServer command stops a PHP server.

Required parameters

-serverName

Specifies the name of the server to stop. (String, required)

-nodeName

Specifies the name of the node on which the server is located. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask stopServer {-nodeName workstation_nameNode01 -serverName phpserver}
```

- Using Jython:

```
AdminTask.stopServer ('[-nodeName workstation_nameNode01 -serverName phpserver]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask stopServer {-interactive}
```

- Using Jython:

```
AdminTask.stopServer ('[-interactive]')
```

deleteServer

The deleteServer command deletes a PHP server.

Required parameters**-serverName**

Specifies the name of the server to delete. (String, required)

-nodeName

Specifies the name of the node on which the server is located. (String, required)

Optional parameters

None.

Batch mode example usage

- Using Jacl:

```
$AdminTask deleteServer {-nodeName workstation_nameNode01 -serverName phpserver}
```

- Using Jython:

```
AdminTask.deleteServer ('[-nodeName workstation_nameNode01 -serverName phpserver]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask deleteServer {-interactive}
```

- Using Jython:

```
AdminTask.deleteServer ('[-interactive]')
```

createPHPServerTemplate

The createPHPServerTemplate command creates a PHP server template.

Required parameters

-templateName

Specifies the name of the template to create. (String, required)

-serverName

Specifies the name of the server to use as a template. (String, required)

-nodeName

Specifies the name of the node on which the server is located. (String, required)

Optional parameters**-description**

Specifies a description for the server template. (String, optional)

-templateLocation

Specifies the location in which the template is stored. Use the system defined location if the location is not specified. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask createPHPServerTemplate {-templateName myphptemplate -nodeName workstation_nameNode01 -serverName phpserver}
```

- Using Jython:

```
AdminTask.createPHPServerTemplate ('[-templateName myphptemplate -nodeName workstation_nameNode01 -serverName phpserver]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createPHPServerTemplate {-interactive}
```

- Using Jython:

```
AdminTask.createPHPServerTemplate ('[-interactive]')
```

deleteServerTemplate

The deleteServerTemplate command deletes a PHP server template.

Required parameters**-templateName**

Specifies the name of the template to delete. (String, required)

-serverName

Specifies the name of the server that uses the template. (String, required)

-nodeName

Specifies the name of the node on which the server is located. (String, required)

Optional parameters**-description**

Specifies a description for the server template. (String, optional)

-templateLocation

Specifies the location in which the template is stored. Use the system defined location if the location is not specified. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask deleteServerTemplate {-templateName myphptemplate -nodeName workstation_nameNode01 -serverName phpserver}
```

- Using Jython:

```
AdminTask.deleteServerTemplate ('[-templateName myphptemplate -nodeName workstation_nameNode01  
-serverName phpserver]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask deleteServerTemplate {-interactive}
```

- Using Jython:

```
AdminTask.deleteServerTemplate ('[-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Creating PHP servers and PHP dynamic clusters](#)

[Deploying PHP applications](#)

Related reference:

[Middleware application administrative tasks](#)

Rules for ODR routing policy administrative tasks

You can use administrative tasks to configure HTTP or Session Initiation Protocol (SIP) rules for the on demand router (ODR) routing policy.

The following rules are the preferred way to configure routing policies, but using the multi-cluster routing property (MCRP) custom property is another option. See the custom properties topic provided in the related links at the end of this topic. The advantage of using the following rules is that an expression can be used to determine which requests are affected by the policy, whereas MCRP custom properties only allow filtering by an application or and application's webmodule. The other advantage of these rules is that you can select the target (routingLocations) by cluster, server or webmodule, as opposed to just cluster.

You can specify SIP or HTTP protocols in the commands.

- [addRoutingRule](#)
- [changeRoutingDefaultRulesAction](#)
- [changeRoutingRuleAction](#)
- [changeRoutingRuleExpression](#)
- [changeRoutingRulePriority](#)
- [createRoutingRules](#)
- [listRoutingRules](#)
- [removeRoutingRule](#)

addRoutingRule

The **addRoutingRule** command adds a routing policy rule.

Note: If you have an existing application edition with a defined multi-cluster routing policy and you install a new edition, you must create a new multi-cluster routing policy for the new edition.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-priority

Positive integer value representing the priority of a rule. Zero is the highest priority. (String, required)

-expression

Specifies the rule expression. The expression must be enclosed in double quotes. For more information about specifying the parameters for the rule expression, refer to the SIP operands topic and the HTTP operands topic. (String, required)

-actionType

Specifies the type of action to associate with a rule. (String, required)

The following list contains the types of actions to associate with HTTP rules:

- **permit:** Permit routing to servers not in a maintenance mode.
- **redirect:** Redirect the request to the URL specified by the *redirectURL* option.
- **reject:** Reject routing with return code specified by the *errorcode* option.
- **permitsticky:** Permit routing to servers not in maintenance mode and perform active affinity; that is, affinity is always preserved even when not requested by the application.
- **permitMM:** Permit routing only to servers in maintenance mode.
- **permitstickyMM:** Permit routing only to servers in maintenance mode and perform active affinity.

The following list contains the types of actions to associate with SIP rules:

- **permit:** Permit routing to servers not in maintenance mode.
- **reject:** Reject routing with return code specified by the *errorcode* option.

Optional parameters

-odrname

Specifies the name of the ODR to which the routing policy work class applies. The **-odrname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

-dcname

Specifies the name of the dynamic cluster to which the rule applies. The **-dcname** parameter is required only if you modify an ODR cluster.

-multiclusterAction

Specifies the method to route requests if multiple routing location clusters are matched. The **-multiclusterAction** parameter applies to any of the permit actionTypes, and is required only if **actionType** is equal to permit, permitsticky, permitMM, or permitstickyMM.

- Failover: Find the first cluster with an available server and load balance across that cluster. The order of a dynamically generated list of clusters is undefined.
- WRR: **W**eighted **R**ound **R**obin load balance. For UDP retransmission, maintain affinity.
- WLOR: **W**eighted least outstanding request.

Note: The recommendation is to use the WLOR value instead of the WRR value.

The following list contains the possible values for SIP rules:

- Failover: Find the first cluster with an available server and load balance across that cluster. The order of a dynamically generated list of clusters is undefined.
- WRR: **W**eighted **R**ound **R**obin load balance. For UDP retransmission, maintain affinity.
- Error: If there are multiple clusters, to select from it throws an error. it expects one and only one cluster.

-routingLocations

Specifies a list of target locations to route requests. The **-routingLocations** parameter is required only if **actionType** is equal to any of the permit actionTypes.

Each operand in the list follows one of three formats, and can contain a wildcard * value, which matches any value:

- cluster=cellName/clusterName
- server=cellName/nodeName/serverName
- module=cellName/applicationName/applicationVersion/moduleName

With SIP routing rules only, you can alternatively define target clusters through a rule expression. The valid operators are AND, OR, NOT and parenthetical grouping. Format according to the following list:

- cluster=cellName/clusterName
- server=cellName/nodeName/serverName
- modules=cellName/applicationName/applicationVersion/moduleName
- server maintenance mode=true or false
- node maintenance mode=true or false
- protocol=PROTO_ *VALUE*:

PROTO_SIP = sip

SIP over TCP

PROTO_SIPS = sips

SIP over SSL and TCP

PROTO_SIPU = sipu

SIP over UDP

PROTO_SIPX = sipx

SIP over XMEM

Note: For applications that have no applicationVersion value, leave the applicationVersion value blank: module=cellName/application//moduleName.

-errorcode

Integer error code to reject request. The **-errorcode** parameter is required only if **actionType** is equal to reject.

Batch mode example usage

The following example shows a failover of all applications in a cell to a generic server cluster that points to another cell:

- Using Jacl:

```
$AdminTask addRoutingRule {-odrname odr -nodename node1 -protocol SIP -priority 0 -expression "request.method = 'getOperation'" -actionType permit -multiclusterAction Failover -routingLocations cluster=*/*}
```

- Using Jython string:

```
AdminTask.addRoutingRule('[-odrname odr -nodename node1 -protocol HTTP -priority 0 -expression "request.method = \'INVITE\'" -actionType permit -multiclusterAction Failover -routingLocations "module=*/*/*/*,cluster=myCell/myFailoverGSCThatPointsToAnotherCell"]')
```

The following example creates a multi-cluster routing policy for a new application edition:

- Using Jacl:

```
$AdminTask addRoutingRule {-odrname odr -nodename node1 -protocol HTTP -priority 0 -expression "uri LIKEIN {'/contextRoot','/contextRoot/%'}" -actionType permit -multicliAction Failover -routingLocations cluster=cellName/clusterName}
```

- Using Jython string:

```
AdminTask.addRoutingRule('[-odrname odr -nodename node1 -protocol HTTP -priority 0 -expression "uri LIKEIN ('/contextRoot','/contextRoot/%')" -actionType permit -multicliAction Failover -routingLocations cluster=cellName/clusterName]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addRoutingRule {-interactive}
```

- Using Jython string:

```
AdminTask.addRoutingRule ( '[-interactive] ' )
```

changeRoutingDefaultRulesAction

The **changeRoutingDefaultRulesAction** command changes the routing policy default action for a rule.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

Optional parameters

-odrname

Specifies the name of the ODR to which the routing policy work class applies. The **-odrname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

-dcname

Specifies the name of the dynamic cluster to which the rule applies. The **-dcname** parameter is required only if you modify an ODR cluster.

-multicliAction

Specifies the method to route requests if multiple routing location clusters are matched. The **-multicliAction** parameter applies to any of the permit actionTypes, and is required only if **actionType** is equal to permit, permitsticky, permitMM, or permitstickyMM.

- Failover: Find the first cluster with an available server and load balance across that cluster. The order of a dynamically generated list of clusters is undefined.
- WRR: Weighted Round Robin load balance. For UDP retransmission, maintain affinity.
- WLOR: Weighted least outstanding request.

Note: The recommendation is to use the WLOR value instead of the WRR value.

The following list contains the possible values for SIP rules:

- Failover: Find the first cluster with an available server and load balance across that cluster. The order of a dynamically generated list of clusters is undefined.
- WRR: Weighted Round Robin load balance. For UDP retransmission, maintain affinity.
- Error: If there are multiple clusters, to select from it throws an error. it expects one and only one cluster.

-routingLocations

Specifies a list of target locations to route requests. The **-routingLocations** parameter is required only if **actionType** is equal to any of the permit actionTypes.

Each operand in the list follows one of three formats, and can contain a wildcard * value, which matches any value:

- cluster=cellName/clusterName
- server=cellName/nodeName/serverName
- module=cellName/applicationName/applicationVersion/moduleName

With SIP routing rules only, you can alternatively define target clusters through a rule expression. The valid operators are AND, OR, NOT and parenthetical grouping. Format according to the following list:

- cluster=cellName/clusterName
- server=cellName/nodeName/serverName
- modules=cellName/applicationName/applicationVersion/moduleName
- server maintenance mode=true or false
- node maintenance mode=true or false
- protocol=PROTO_VALUE:

PROTO_SIP = sip

SIP over TCP

PROTO_SIPS = sips

SIP over SSL and TCP

PROTO_SIPU = sipu

SIP over UDP

PROTO_SIPX = sipx

SIP over XMEM

Note: For applications that have no **applicationVersion** value, leave the **applicationVersion** value blank: module=cellName/application//moduleName.

-errorcode

Integer error code to reject request. The **-errorcode** parameter is required only if **actionType** is equal to reject.

Batch mode example usage

- Using Jacl:

The following examples shows a failover of a single cluster to the failover generic server cluster:

```
$AdminTask changeRoutingDefaultRulesAction {-odrname odr -nodename node1 -protocol SIP -multiclusterAction Failover -routingLocations cluster=*/*}
```

- Using Jython string:

```
AdminTask.changeRoutingDefaultRulesAction(['-odrname odr -nodename node1 -protocol HTTP -multiclusterAction FAILOVER -routingLocations "cluster=myCell/myPrimaryCluster,cluster=myCell/myFailoverCluster"'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeRoutingDefaultRulesAction {-interactive}
```

- Using Jython string:

```
AdminTask.changeRoutingDefaultRulesAction (['-interactive'])
```

changeRoutingRuleAction

Use the **changeRoutingRuleAction** command to change a routing policy action for a rule.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-priority

Positive integer value representing the priority of a rule. Zero is the highest priority. (String, required)

Optional parameters

-odrname

Specifies the name of the ODR to which the routing policy work class applies. The **-odrname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

-dcname

Specifies the name of the dynamic cluster to which the rule applies. The **-dcname** parameter is required only if you modify an ODR cluster.

-multiclusterAction

Specifies the method to route requests if multiple routing location clusters are matched. The **-multiclusterAction** parameter applies to any of the permit actionTypes, and is required only if **actionType** is equal to permit, permitsticky, permitMM, or permitstickyMM.

- Failover: Find the first cluster with an available server and load balance across that cluster. The order of a dynamically generated list of clusters is undefined.
- WRR: Weighted Round Robin load balance. For UDP retransmission, maintain affinity.
- WLOR: Weighted least outstanding request.

Note: The recommendation is to use the WLOR value instead of the WRR value.

The following list contains the possible values for SIP rules:

- Failover: Find the first cluster with an available server and load balance across that cluster. The order of a dynamically generated list of clusters is undefined.
- WRR: Weighted Round Robin load balance. For UDP retransmission, maintain affinity.
- Error: If there are multiple clusters, to select from it throws an error. It expects one and only one cluster.

-routingLocations

Specifies a list of target locations to route requests. The **-routingLocations** parameter is required only if **actionType** is equal to any of the permit actionTypes.

Each operand in the list follows one of three formats, and can contain a wildcard * value, which matches any value:

- cluster=cellName/clusterName
- server=cellName/nodeName/serverName
- module=cellName/applicationName/applicationVersion/moduleName

With SIP routing rules only, you can alternatively define target clusters through a rule expression. The valid operators are AND, OR, NOT and parenthetical grouping. Format according to the following list:

- cluster=cellName/clusterName
- server=cellName/nodeName/serverName
- modules=cellName/applicationName/applicationVersion/moduleName
- server maintenance mode=true or false
- node maintenance mode=true or false
- protocol=PROTO_VALUE:

PROTO_SIP = sip

SIP over TCP

PROTO_SIPS = sips

SIP over SSL and TCP

PROTO_SIPU = sipu

SIP over UDP

PROTO_SIPX = sipx

SIP over XMEM

Note: For applications that have no **applicationVersion** value, leave the **applicationVersion** value blank: module=cellName/application//moduleName.

-errorcode

Integer error code to reject request. The **-errorcode** parameter is required only if **actionType** is equal to reject.

Batch mode example usage

- Using Jacl:

```
$AdminTask changeRoutingRuleAction {-odrname odr -nodename node1 -protocol SIP -priority 0 -multiclusterAction Failover -routingLocations cluster=*/*}
```


- Using Jython string:

```
AdminTask.changeRoutingRuleAction(['-odrname odr -nodename node1 -protocol HTTP -priority 0 -multiclusterAction WRR -routingLocations "cluster=myCell/*"]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeRoutingRuleAction {-interactive}
```

- Using Jython string:

```
AdminTask.changeRoutingRuleAction (['-interactive'])
```

changeRoutingRuleExpression

Use the **changeRoutingRuleExpression** command to change a routing policy rule expression.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-priority

Positive integer value representing the priority of a rule. Zero is the highest priority. (String, required)

-expression

Specifies the rule expression. The expression must be enclosed in double quotes. For more information about specifying the parameters for the rule expression, refer to the SIP operands topic and the HTTP operands topic. (String, required)

Required parameters

-odrname

Specifies the name of the ODR to which the routing policy work class applies. The **-odrname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

-dcname

Specifies the name of the dynamic cluster to which the rule applies. The **-dcname** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask changeRoutingRuleExpression {-odrname odr -nodename node1 -protocol SIP -priority 0 -expression "request.method = 'getOperation0'"}

```

- Using Jython string:

```
AdminTask.changeRoutingRuleExpression(['-odrname odr -nodename node1 -protocol HTTP -priority 0 -expression "request.method = \'INVITE\'"]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeRoutingRuleExpression {-interactive}
```

- Using Jython string:

```
AdminTask.changeRoutingRuleExpression (['-interactive'])
```

changeRoutingRulePriority

Use the **changeRoutingRulePriority** command to change a routing policy rule priority.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-priority

Positive integer value representing the priority of a rule. Zero is the highest priority. (String, required)

-expression

Specifies the rule expression. The expression must be enclosed in double quotes. For more information about specifying the parameters for the rule expression, refer to the SIP operands topic and the HTTP operands topic. (String, required)

Optional parameters

-odurname

Specifies the name of the ODR to which the routing policy work class applies. The **-odurname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

-dcname

Specifies the name of the dynamic cluster to which the rule applies. The **-dcname** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask changeRoutingRulePriority {-odurname odr -nodename node1 -protocol SIP -priority 0 -expression "request.method = 'getOperation0'"} 
```

- Using Jython string:

```
AdminTask.changeRoutingRulePriority(['-odurname odr -nodename node1 -protocol HTTP -priority 0 -expression "request.method = \'INVITE\'"]])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeRoutingRulePriority {-interactive}
```

- Using Jython string:

```
AdminTask.changeRoutingRulePriority (['-interactive'])
```

createRoutingRules

Use the **createRoutingRules** command to create a routing policy rule list.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

Optional parameters

-odurname

Specifies the name of the ODR to which the routing policy work class applies. The **-odurname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

-dcname

Specifies the name of the dynamic cluster to which the rule applies. The **-dcname** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask createRoutingRules {-odname odr -nodename node1 -protocol SIP}
```

- Using Jython string:

```
AdminTask.createRoutingRules('-odname odr -nodename node1 -protocol SIP')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createRoutingRules {-interactive}
```

- Using Jython string:

```
AdminTask.createRoutingRules (['-interactive'])
```

listRoutingRules

Use the **listRoutingRules** to delete a dynamic cluster from the configuration.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

Optional parameters

-odname

Specifies the name of the ODR to which the routing policy work class applies. The **-odname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

-dcname

Specifies the name of the dynamic cluster to which the rule applies. The **-dcname** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask listRoutingRules {-odname odr -nodename node1 -protocol SIP}
```

- Using Jython string:

```
AdminTask.listRoutingRules('-odname odr -nodename node1 -protocol SIP')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listRoutingRules {-interactive}
```

- Using Jython string:

```
AdminTask.listRoutingRules ('[-interactive]')
```

removeRoutingRule

Use the **removeRoutingRule** command to removes a routing policy rule.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-priority

Positive integer value representing the priority of a rule. Zero is the highest priority. (String, required)

-expression

Specifies the rule expression. The expression must be enclosed in double quotes. For more information about specifying the parameters for the rule expression, refer to the SIP operands topic and the HTTP operands topic. (String, required)

Optional parameters

-odname

Specifies the name of the ODR to which the routing policy work class applies. The **-odname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

-dcname

Specifies the name of the dynamic cluster to which the rule applies. The **-dcname** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask removeRoutingRule {-odname odr -nodename node1 -protocol SIP -expression "request.method = 'getOperation'"} 
```

- Using Jython string:

```
AdminTask.removeRoutingRule('[-odname odr -nodename node1 -protocol HTTP -expression "request.method = \'INVITE\'"]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeRoutingRule {-interactive}
```

- Using Jython string:

```
AdminTask.removeRoutingRule ('[-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Routing policy action types](#)

[Overview of request flow prioritization](#)

Related tasks:

[Configuring the on demand router for multi-cluster failover and load balancing routing](#)

[Configuring ODRs](#)

[Creating a static cluster of ODRs](#)

Related reference:

[SIP operands](#)

[HTTP operands](#)

Related information:

[Rules for ODR service policy administrative tasks](#)

Rules for ODR service policy administrative tasks

You can use administrative tasks to configure Session Initiation Protocol (SIP) or HTTP rules for an on demand router (ODR) service policy.

Command list

You can specify SIP or HTTP protocols in the commands. The following examples use SIP as the protocol:

- [addServiceRule](#)
- [removeServiceRule](#)
- [createServiceRules](#)
- [listServiceRules](#)
- [changeServiceRuleAction](#)
- [changeServiceDefaultRulesAction](#)
- [changeServiceRuleExpression](#)
- [changeServiceRulePriority](#)

addServiceRule

Use the **addServiceRule** command to add a service policy rule.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-priority

Specifies a positive integer value representing the priority of a rule. Zero is the highest priority. (String, required)

-expression

Specifies the rule expression. The expression must be enclosed in double quotes. (String, required)

-transactionClass

Specifies the transaction class to associate with a rule. (String, required)

Optional parameters

-odrname

Specifies the name of the ODR to which the service policy work class applies. The **-odrname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask addServiceRule {-odrname odr -nodename node1 -protocol SIP -priority 0 -expression "request.method = 'getOperation'" -transactionClass Default_TC}
```

- Using Jython string:

```
AdminTask.addServiceRule('-odrname odr -nodename node1 -protocol SIP -priority 0 -expression "request.method = \'getOperation\'' -transactionClass Default_TC')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addServiceRule {-interactive}
```

- Using Jython string:

```
AdminTask.addServiceRule ('[-interactive]')
```

removeServiceRule

Use the **removeServiceRule** command to remove a service policy rule.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-expression

Specifies the rule expression. The expression must be enclosed in double quotes. (String, required)

Optional parameters

-odrcode

Specifies the name of the ODR to which the service policy work class applies. The **-odrcode** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask removeServiceRule {-odrcode odr -nodename node1 -protocol SIP -expression "request.method = 'getOperation'"}
```

- Using Jython string:

```
AdminTask.removeServiceRule('-odrcode odr -nodename node1 -protocol SIP -expression "request.method = \'getOperation\'"')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeServiceRule {-interactive}
```

- Using Jython string:

```
AdminTask.removeServiceRule ('[-interactive]')
```

createServiceRules

Use the **createServiceRules** command to create a rule list for the service policy.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

Optional parameters

-odrcode

Specifies the name of the ODR to which the service policy work class applies. The **-odrcode** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask createServiceRules {-odrname odr -nodename node1 -protocol SIP}
```

- Using Jython string:

```
AdminTask.createServiceRules('-odrname odr -nodename node1 -protocol SIP')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createServiceRules {-interactive}
```

- Using Jython string:

```
AdminTask.createServiceRules ('[-interactive]')
```

listServiceRules

Use the **listServiceRules** command to list service policy rules.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

Optional parameters

-odrname

Specifies the name of the ODR to which the service policy work class applies. The **-odrname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask listServiceRules {-odrname odr -nodename node1 -protocol SIP}
```

- Using Jython string:

```
AdminTask.listServiceRules('-odrname odr -nodename node1 -protocol SIP')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listServiceRules {-interactive}
```

- Using Jython string:

```
AdminTask.listServiceRules ('[-interactive]')
```

changeServiceRuleAction

Use the **changeServiceRuleAction** command to change the service policy action for a rule.

Restriction: The use of the **changeServiceRuleAction** command with HTTP is not supported.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-priority

Specifies a positive integer value representing the priority of a rule. Zero is the highest priority. (String, required)

-transactionClass

Specifies the transaction class to associate with a rule. (String, required)

Optional parameters**-odrname**

Specifies the name of the ODR to which the service policy work class applies. The **-odrname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask changeServiceRuleAction {-odrname odr -nodename node1 -protocol SIP -priority 0 -transactionClass Default_TC}
```

- Using Jython string:

```
AdminTask.changeServiceRuleAction('-odrname odr -nodename node1 -protocol SIP -priority 0 -transactionClass Default_TC')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeServiceRuleAction {-interactive}
```

- Using Jython string:

```
AdminTask.changeServiceRuleAction ('[-interactive]')
```

changeServiceDefaultRulesAction

Use the **changeServiceDefaultRulesAction** command to change the default action of a service policy rule.

Restriction: The use of the **changeServiceDefaultRulesAction** command with HTTP is not supported.

Required parameters**-protocol**

Specifies the name of the protocol to associate with a rule. (String, required)

-transactionClass

Specifies the transaction class to associate with a rule. (String, required)

Optional parameters**-odrname**

Specifies the name of the ODR to which the service policy work class applies. The **-odrname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask changeServiceDefaultRulesAction {-odrname odr -nodename node1 -protocol SIP -transactionClass Default_TC}
```

- Using Jython string:

```
AdminTask.changeServiceDefaultRulesAction('-odrname odr -nodename node1 -protocol SIP -transactionClass Default_TC')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeServiceDefaultRulesAction {-interactive}
```

- Using Jython string:

```
AdminTask.changeServiceDefaultRulesAction ('[-interactive]')
```

changeServiceRuleExpression

Use the **changeServiceRuleExpression** command to change a rule expression of a service policy.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-priority

Specifies a positive integer value representing the priority of a rule. Zero is the highest priority. (String, required)

-expression

Specifies the rule expression. The expression must be enclosed in double quotes. (String, required)

-transactionClass

Specifies the transaction class to associate with a rule. (String, required)

Optional parameters

-odrname

Specifies the name of the ODR to which the service policy work class applies. The **-odrname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask changeServiceRuleExpression {-odrname odr -nodename node1 -protocol SIP -priority 0 -expression "request.method = 'getOperation'"} 
```

- Using Jython string:

```
AdminTask.changeServiceRuleExpression('-odrname odr -nodename node1 -protocol SIP -priority 0 -expression "request.method = \'getOperation\'")
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeServiceRuleExpression {-interactive}
```

- Using Jython string:

```
AdminTask.changeServiceRuleExpression ('[-interactive]')
```

changeServiceRulePriority

Use the **changeServiceRulePriority** command to change the priority of a service policy rule.

Required parameters

-protocol

Specifies the name of the protocol to associate with a rule. (String, required)

-priority

Specifies a positive integer value representing the priority of a rule. Zero is the highest priority. (String, required)

-expression

Specifies the rule expression. The expression must be enclosed in double quotes. (String, required)

Optional parameters

-odurname

Specifies the name of the ODR to which the service policy work class applies. The **-odurname** parameter is required only if you modify an ODR.

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR.

-clustername

Specifies the name of the cluster to which the rule applies. The **-clustername** parameter is required only if you modify an ODR cluster.

Batch mode example usage

- Using Jacl:

```
$AdminTask changeServiceRulePriority {-odurname odr -nodename node1 -protocol SIP -priority 0 -  
expression "request.method = 'getOperation'"} 
```

- Using Jython string:

```
AdminTask.changeServiceRulePriority('-odurname odr -nodename node1 -protocol SIP -priority 0 -  
expression "request.method = \'getOperation\'")
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeServiceRulePriority {-interactive}
```

- Using Jython string:

```
AdminTask.changeServiceRulePriority ('[-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring ODRs](#)

[Creating a static cluster of ODRs](#)

Related reference:

[Routing and service policies](#)

Rules for generic server cluster routing policy administrative tasks

You can use administrative tasks to configure routing rules for the generic server cluster routing policy.

Use these commands to configure routing policy work classes:

- [createRoutingPolicyWorkClass](#)
- [addRoutingPolicyRoutingRule](#)
- [deleteRoutingPolicyWorkClass](#)
- [removeRoutingPolicyRoutingRule](#)

createRoutingPolicyWorkClass

Use the **createRoutingPolicyWorkClass** command to create a routing policy work class.

Required parameters

-wcname

Specifies the unique work class name.

-type

Specifies the work class type. Must be HTTP, IIOP, or SOAP

-actionType

Specifies the default type of action to associate with a rule. (String, required)

The following list contains the types of actions to associate with routing rules:

- **permit**: Permit routing to servers not in a maintenance mode.
- **redirect**: Redirect the request to the URL specified by the *redirectURL* option.
- **reject**: Reject routing with return code specified by the *errorcode* option.
- **permitsticky**: Permit routing to servers not in maintenance mode and perform active affinity; that is, affinity is always preserved even when not requested by the application.
- **permitMM**: Permit routing to servers in maintenance mode.
- **permitstickyMM**: Permit routing to servers in maintenance mode and perform active affinity.

-action

Specifies the default type of action for the workclass. If the **actionType** is a permit action, provide the generic server cluster name to route to. If the **actionType** is reject, provide the error code to return. If the **actionType** is redirect, provide the URI to redirect to.

Optional parameters

-appname

Specifies the name of the application to which the routing policy work class applies. (Not edition qualified, for example MyApplication).

-odrname

Specifies the name of the ODR to which the routing policy work class applies.

-odrnode

Specifies the name of the node on which the ODR resides.

-dcname

Specifies the name of the ODR dynamic cluster resource related to the work class.

-members

Specifies the protocol-specific pattern:

- HTTP: /test1?/test2/*
- IIOP: <ejbName>:<ejbMethod>?<ejbName>:<ejbMethod>
- SOAP: <webService>:<operationName>?<webService>:<operationName>
- JMS: <bus>:<destination>?<bus>:<destination>

-vhost

Specifies the name of the virtual host to apply the workclass to.

If you use the command with the parameters **-odrnode** and **-odrname**, do not use the **-dcname** parameter.

If you use the command with the parameter **-dcname**, do not use the **-odrnode** and **-odrname** parameters.

Batch mode example usage

- Using Jacl:

```
$AdminTask createRoutingPolicyWorkClass {-dcname dynamic_cluster_1 -wcname work_class_1 -type HTTP -actiontype reject -action 55 -vhost admin_host -members "/test1?/test2/*"}
```

- Using Jython string:

```
AdminTask.createRoutingPolicyWorkClass ('[-dcname dynamic_cluster_1 -wcname work_class_1 -type HTTP -actiontype reject -action 55 -vhost admin_host -members "/test1?/test2/*"]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createRoutingPolicyWorkClass {-interactive}
```

- Using Jython string:

```
AdminTask.createRoutingPolicyWorkClass ('[-interactive]')
```

addRoutingPolicyRoutingRule

Use the **addRoutingPolicyRoutingRule** command to add a routing rule to an existing work class.

Required parameters

-wcname

Specifies the unique work class name.

-expression

Specifies the rule expression.

-actionType

Specifies the default type of action to associate with a rule. (String, required)

The following list contains the types of actions to associate with routing rules:

- permit: Permit routing to servers not in a maintenance mode.
- redirect: Redirect the request to the URL specified by the *redirectURL* option.
- reject: Reject routing with return code specified by the *errorcode* option.
- permitsticky: Permit routing to servers not in maintenance mode and perform active affinity; that is, affinity is always preserved even when not requested by the application.
- permitMM: Permit routing to servers in maintenance mode.
- permitstickyMM: Permit routing to servers in maintenance mode and perform active affinity.

-action

Specifies the default type of action for the workclass. If the **actionType** is a permit action, provide the generic server cluster name to route to. If the **actionType** is reject, provide the error code to return. If the **actionType** is redirect, provide the URI to redirect to.

Optional parameters

-odrname

Specifies the name of the ODR to which the routing policy work class applies.

-odrnnode

Specifies the name of the node on which the ODR resides.

-dcname

Specifies the name of the ODR dynamic cluster resource related to the work class.

-priority

Specifies a positive integer value representing the priority of the routing rule. Zero is the highest priority.

If you use the command with the parameters **-odrnnode** and **-odrname**, do not use the **-dcname** parameter.

If you use the command with the parameter **-dcname**, do not use the **-odrnnode** and **-odrname** parameters.

Batch mode example usage

- Using Jacl:

```
$AdminTask addRoutingPolicyRoutingRule {-dcname dynamic_cluster_1 -wcname work_class_1 -expression "serverType = 'WASAPP'" -actiontype redirect -action "/redirect/url/*" -priority 10}
```

- Using Jython string:

```
AdminTask.addRoutingPolicyRoutingRule ('[-dcname dynamic_cluster_1 -wcname work_class_1 -
expression "serverType = 'WASAPP'" -actiontype redirect -action "/redirect/url/*" -priority
10]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addRoutingPolicyRoutingRule {-interactive}
```

- Using Jython string:

```
AdminTask.addRoutingPolicyRoutingRule ('[-interactive]')
```

deleteRoutingPolicyWorkClass

Use the **deleteRoutingPolicyWorkClass** command to delete a routing policy work class.

Required parameters

-wcname

Specifies the unique work class name.

Optional parameters

-odrname

Specifies the name of the ODR to which the routing policy work class applies.

-odrnnode

Specifies the name of the node on which the ODR resides.

-dcname

Specifies the name of the ODR dynamic cluster resource related to the work class.

If you use the command with the parameters **-odrnnode** and **-odrname**, do not use the **-dcname** parameter.

If you use the command with the parameter **-dcname**, do not use the **-odrnnode** and **-odrname** parameters.

Batch mode example usage

- Using Jacl:

```
$AdminTask deleteRoutingPolicyWorkClass {-odrnnode node_1 -odrname odr_1 -wcname
odr_work_class}
$AdminTask deleteRoutingPolicyWorkClass {-dcname dyanmic_cluster -wcname work_class_1}
```

- Using Jython string:

```
AdminTask.deleteRoutingPolicyWorkClass ('[-odrnnode node_1 -odrname odr_1 -wcname
odr_work_class]')
AdminTask.deleteRoutingPolicyWorkClass ('[-dcname dyanmic_cluster -wcname work_class_1]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask deleteRoutingPolicyWorkClass {-interactive}
```

- Using Jython string:

```
AdminTask.deleteRoutingPolicyWorkClass ('[-interactive]')
```

removeRoutingPolicyRoutingRule

Use the **removeRoutingPolicyRoutingRule** command to remove a routing rule from an existing work class.

Required parameters

-wcname

Specifies the unique work class name.

-expression

Specifies the rule expression.

Optional parameters**-odrname**

Specifies the name of the ODR to which the routing policy work class applies.

-odrnnode

Specifies the name of the node on which the ODR resides.

-dcname

Specifies the name of the ODR dynamic cluster resource related to the work class.

If you use the command with the parameters **-odrnnode** and **-odrname**, do not use the **-dcname** parameter.

If you use the command with the parameter **-dcname**, do not use the **-odrnnode** and **-odrname** parameters.

Batch mode example usage

- Using Jacl:

```
$AdminTask removeRoutingPolicyRoutingRule {-dcname dynamic_cluster_1 -wcname work_class_1 -  
expression "serverType = 'WASAPP'"}}
```

- Using Jython string:

```
AdminTask.removeRoutingPolicyRoutingRule ('[-dcname dynamic_cluster_1 -wcname work_class_1 -  
expression "serverType = 'WASAPP'"]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeRoutingPolicyRoutingRule {-interactive}
```

- Using Jython string:

```
AdminTask.removeRoutingPolicyRoutingRule ('[-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Routing policy action types](#)

Related tasks:

[Defining routing policies for generic server clusters](#)

Related information:

[Rules for generic server cluster service policy administrative tasks](#)

[Rules for ODR service policy administrative tasks](#)

Rules for generic server cluster service policy administrative tasks

You can use administrative tasks to configure service rules for the generic server cluster routing policy.

Command list

Use these commands to configure service policy work classes:

- [createServicePolicyWorkClass](#)
- [addServicePolicyRoutingRule](#)
- [deleteServicePolicyWorkClass](#)
- [removeServicePolicyRoutingRule](#)

createServicePolicyWorkClass

Use the **createServicePolicyWorkClass** command to configure service policy work classes .

Required parameters

-wcname

Specifies the unique work class name.

-type

Specifies the work class type. Must be HTTP, IIOP, or SOAP

Optional parameters

-odrname

Specifies the name of the ODR to which the routing policy work class applies.

-odrnode

Specifies the name of the node on which the ODR resides.

-dcname

Specifies the name of the ODR dynamic cluster resource related to the work class.

-tcname

Specifies the name of the transaction class to map the workclass to. If not provided, this will default to the *Default_TC* of the *Default_SP*.

-members

Specifies the protocol-specific pattern:

- HTTP: /test1?/test2/*
- IIOP: <ejbName>:<ejbMethod>?<ejbName>:<ejbMethod>
- SOAP: <webService>:<operationName>?<webService>:<operationName>
- JMS: <bus>:<destination>?<bus>:<destination>

-vhost

Specifies the name of the virtual host to apply the workclass to.

If you use the command with the parameters **-odrnode** and **-odrname**, do not use the **-dcname** parameter.

If you use the command with the parameter **-dcname**, do not use the **-odrnode** and **-odrname** parameters.

Batch mode example usage

- Using Jacl:

```
$AdminTask createServicePolicyWorkClass {-dcname dynamic_cluster -wcname work_class_1 -type HTTP -vhost admin_host -tcname transaction_class_1 -members "/test1?/test2/*"}
```

- Using Jython string:

```
AdminTask.createServicePolicyWorkClass ('[-dcname dynamic_cluster -wcname work_class_1 -type HTTP -vhost admin_host -tcname transaction_class_1 -members "/test1?/test2/*"]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createServicePolicyWorkClass {-interactive}
```

- Using Jython string:

```
AdminTask.createServicePolicyWorkClass (['-interactive'])
```

addServicePolicyRoutingRule

Use the **addServicePolicyRoutingRule** command to add a routing rule to an existing workclass.

Required parameters

-wcname

Specifies the unique work class name.

-expression

Specifies the rule expression.

Optional parameters

-odrname

Specifies the name of the ODR to which the routing policy work class applies.

-odrnnode

Specifies the name of the node on which the ODR resides.

-dcname

Specifies the name of the ODR dynamic cluster resource related to the work class.

-tcname

Specifies the name of the transaction class to map the workclass to. If not provided, this will default to the *Default_TC* of the *Default_SP*.

-priority

Specifies a positive integer value representing the priority of the routing rule. Zero is the highest priority.

If you use the command with the parameters **-odrnnode** and **-odrname**, do not use the **-dcname** parameter.

If you use the command with the parameter **-dcname**, do not use the **-odrnnode** and **-odrname** parameters.

Batch mode example usage

- Using Jacl:

```
$AdminTask addServicePolicyRoutingRule {-dcname dynamic_cluster -wcname work_class_1 -  
expression "serverType = 'WASAPP'" -priority 10}
```

- Using Jython string:

```
AdminTask.addServicePolicyRoutingRule (['-dcname dynamic_cluster -wcname work_class_1 -  
expression "serverType = 'WASAPP'" -priority 10'])
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addServicePolicyRoutingRule {-interactive}
```

- Using Jython string:

```
AdminTask.addServicePolicyRoutingRule (['-interactive'])
```

deleteServicePolicyWorkClass

Use the **deleteServicePolicyWorkClass** command to delete a service policy work class.

Required parameters

-wcname

Specifies the unique work class name.

Optional parameters

-odrname

Specifies the name of the ODR to which the routing policy work class applies.

-odrnnode

Specifies the name of the node on which the ODR resides.

-dcname

Specifies the name of the ODR dynamic cluster resource related to the work class.

If you use the command with the parameters **-odrnnode** and **-odrname**, do not use the **-dcname** parameter.

If you use the command with the parameter **-dcname**, do not use the **-odrnnode** and **-odrname** parameters.

Batch mode example usage

- Using Jacl:

```
$AdminTask deleteServicePolicyWorkClass {-odrnnode node_1 -odrname odr_1 -wcname
odr_work_class}
$AdminTask deleteServicePolicyWorkClass {-dcname dyanmic_cluster -wcname work_class_1}
```

- Using Jython string:

```
AdminTask.deleteServicePolicyWorkClass ('[-odrnnode node_1 -odrname odr_1 -wcname
odr_work_class]')
AdminTask.deleteServicePolicyWorkClass ('[-dcname dyanmic_cluster -wcname work_class_1]')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask deleteServicePolicyWorkClass {-interactive}
```

- Using Jython string:

```
AdminTask.deleteServicePolicyWorkClass ('[-interactive]')
```

removeServicePolicyRoutingRule

Use the **removeServicePolicyRoutingRule** to remove a routing rule from an existing work class.

Required parameters**-wcname**

Specifies the unique work class name.

-expression

Specifies the rule expression.

Optional parameters**-odrname**

Specifies the name of the ODR to which the routing policy work class applies.

-odrnnode

Specifies the name of the node on which the ODR resides.

-dcname

Specifies the name of the ODR dynamic cluster resource related to the work class.

If you use the command with the parameters **-odrnnode** and **-odrname**, do not use the **-dcname** parameter.

If you use the command with the parameter **-dcname**, do not use the **-odrnnode** and **-odrname** parameters.

Batch mode example usage

- Using Jacl:

```
$AdminTask removeServicePolicyRoutingRule {-dcname dynamic_cluster -wcname work_class_1 -
expression "serverType = 'WASAPP'"}

```

- Using Jython string:

```
AdminTask.removeServicePolicyRoutingRule ('[-dcname dynamic_cluster -wcname work_class_1 -
```

```
expression "serverType = 'WASAPP']']')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeServicePolicyRoutingRule {-interactive}
```

- Using Jython string:

```
AdminTask.removeServicePolicyRoutingRule ('[-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Defining service policies for generic server clusters](#)

Related reference:

[Routing and service policies](#)

Related information:

[Rules for generic server cluster routing policy administrative tasks](#)

Ruleset administrative tasks

You can use the ruleset commands to add, remove, or modify rules and actions of the ruleset.xml file without using the administrative console.

Use the following commands to create ruleset lists, and add or remove rules and actions from the ruleset:

- [createRuleset](#)
- [listRuleset](#)
- [addRuleToRuleset](#)
- [removeRuleFromRuleset](#)
- [changeRulePriority](#)
- [changeRuleExpression](#)
- [addActionToRule](#)
- [removeActionFromRule](#)
- [addDefaultAction](#)
- [removeDefaultAction](#)

createRuleset

The createRuleset command creates a ruleset list.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

-rulesetType

Specifies the protocol type to associate with the ruleset. The only valid types are HTTP and SOAP. (String, required)

-defaultContinue

Specifies the setting for the default continue flag value. Specify True or False. (Boolean, required)

Optional parameters

-odrname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odrname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask createRuleset {-odrname odr -nodename node1 -rulesetName myRuleset -rulesetType HTTP -defaultContinue True}
```

- Using Jython string:

```
AdminTask.createRuleset ('-odrname odr -nodename node1 -rulesetName myRuleset -rulesetType HTTP -defaultContinue True')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask createRuleset {-interactive}
```

- Using Jython string:

```
AdminTask.createRuleset ( ['-interactive'] )
```

listRuleset

The listRuleset command lists ruleset rules and actions.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

Optional parameters

-odrname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odrname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask listRuleset {-odrname odr -nodename node1 -rulesetName myRuleset}
```

- Using Jython string:

```
AdminTask.listRuleset ('-odrname odr -nodename node1 -rulesetName myRuleset')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask listRuleset {-interactive}
```

- Using Jython string:

```
AdminTask.listRuleset ('[-interactive]')
```

addRuleToRuleset

The addRuleToRuleset command adds a rule to the ruleset.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

-ruleName

Specifies the name of the rule. The **-ruleName** parameter must be unique within the ruleset. (String, required)

-rulePriority

Specifies a positive integer value representing the priority of the rule. Zero is the highest priority. (String, required)

-expression

Specifies the rule expression. You must enclose the value in double quotes. (String, required)

Optional parameters

-odrname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odrname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only

if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask addRuleToRuleset {-odname odr -nodename node1 -rulesetName myRuleset -ruleName myRule -rulePriority 0 -expression "vhost = vhostA"}
```

- Using Jython string:

```
AdminTask.addRuleToRuleset('-odname odr -nodename node1 -rulesetName myRuleset -ruleName myRule -rulePriority 0 -expression "vhost = \'vhostA\'")
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addRuleToRuleset {-interactive}
```

- Using Jython string:

```
AdminTask.addRuleToRuleset ('[-interactive]')
```

removeRuleFromRuleset

The removeRuleFromRuleset command removes a rule from the ruleset.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

-ruleName

Specifies the name of the rule to remove. (String, required)

Optional parameters

-odname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask removeRuleFromRuleset {-odname odr -nodename node1 -rulesetName myRuleset -ruleName myRule}
```

- Using Jython string:

```
AdminTask.removeRuleFromRuleset('-odname odr -nodename node1 -rulesetName myRuleset -ruleName myRule')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeRulefromRuleset {-interactive}
```

- Using Jython string:

```
AdminTask.removeRulefromRuleset ('[-interactive]')
```

changeRulePriority

The changeRulePriority command modifies the priority of a rule.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

-ruleName

Specifies the name of the rule to modify. (String, required)

-rulePriority

Specifies a positive integer value representing the priority of the rule. Zero is the highest priority. (String, required)

Optional parameters

-odrname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odrname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask changeRulePriority {-odrname odr -nodename node1 -rulesetName myRuleset -ruleName myRule -rulePriority 1}
```

- Using Jython string:

```
AdminTask.changeRulePriority('-odrname odr-nodename node1 -rulesetName myRuleset -ruleName myRule -rulePriority 1')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeRulePriority {-interactive}
```

- Using Jython string:

```
AdminTask.changeRulePriority ('[-interactive]')
```

changeRuleExpression

The changeRuleExpression command modifies a rule expression.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

-ruleName

Specifies the name of the rule to modify. (String, required)

-expression

Specifies the new rule expression. (String, required)

Optional parameters

-odrname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odrname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask changeRuleExpression {-odname odr -nodename node1 -rulesetName myRuleset -ruleName myRule -expression "vhost = vhostB"}
```

- Using Jython string:

```
AdminTask.changeRuleExpression('-odname odr -nodename node1 -rulesetName myRuleset -ruleName myRule -expression "vhost = \'vhostB\'")
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask changeRuleExpression {-interactive}
```

- Using Jython string:

```
AdminTask.changeRuleExpression ('[-interactive]')
```

addActionToRule

The addActionToRule command adds an action to a rule.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

-ruleName

Specifies the name of the rule. The **-ruleName** parameter must be unique within the ruleset. (String, required)

-actionName

Specifies the name of the action. The **-actionName** parameter must be unique within the ruleset. (String, required)

-actionType

Specifies the type of action to associate with the rule. (String, required)

-actionValue

Specifies the action value, such as the format of the log file, to use. (String, required)

-actionContinue

Specifies the setting for the action continue flag value. Specify True or False. (Boolean, required)

Optional parameters

-odname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask addActionToRule {-odrname odr -nodename node1 -rulesetName myRuleset -ruleName myRule -actionName myAction -actionType log -actionValue "MyCustom.log %r %T %z %Z" -actionContinue true}
```

- Using Jython string:

```
AdminTask.addActionToRule('-odrname odr -nodename node1 -rulesetName myRuleset -ruleName myRule -actionName myAction -actionType log -actionValue "MyCustom.log %r %T %z %Z" -actionContinue true')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addActionToRule {-interactive}
```

- Using Jython string:

```
AdminTask.addActionToRule ('[-interactive]')
```

removeActionFromRule

The removeActionFromRule command removes an action from a rule.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

-ruleName

Specifies the name of the rule that contains the action. (String, required)

-actionName

Specifies the name of the action to remove. (String, required)

Optional parameters

-odrname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odrname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask removeActionFromRule {-odrname odr -nodename node1 -rulesetName myRuleset -ruleName myRule -actionName myAction}
```

- Using Jython string:

```
AdminTask.removeActionFromRule('-odrname odr -nodename node1 -rulesetName myRuleset -ruleName myRule -actionName myAction')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeActionFromRule {-interactive}
```

- Using Jython string:

```
AdminTask.removeActionFromRule ('[-interactive]')
```

addDefaultAction

The addDefaultAction command adds a default action to a ruleset.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

-actionName

Specifies the name of the action. The **-actionName** parameter must be unique within the ruleset. (String, required)

-actionType

Specifies the type of action to associate with the rule. (String, required)

-actionValue

Specifies the action value, such as the format of the log file, to use. (String, required)

-actionContinue

Specifies the setting for the action continue flag value. Specify True or False. (Boolean, required)

Optional parameters

-odrname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odrname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask addDefaultAction {-odrname odr -nodename node1 -rulesetName myRuleset -actionName defAction -actionType log -actionValue "default.log %a %d %D" -actionContinue True}
```

- Using Jython string:

```
AdminTask.addDefaultAction('-odrname odr -nodename node1 -rulesetName myRuleset -actionName defAction -actionType log -actionValue "default.log %a %d %D" -actionContinue True')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask addDefaultAction {-interactive}
```

- Using Jython string:

```
AdminTask.addDefaultAction ('[-interactive]')
```

removeDefaultAction

The removeDefaultAction command removes a default action from a ruleset.

Required parameters

-rulesetName

Specifies the name of the ruleset. (String, required)

-actionName

Specifies the name of the action to remove. (String, required)

Optional parameters

-odrname

Specifies the name of the on demand router (ODR) to which the ruleset applies. The **-odrname** parameter is required only if you modify an ODR. (String, optional)

-nodename

Specifies the name of the node on which the ODR resides. The **-nodename** parameter is required only if you modify an ODR. (String, optional)

-clustername

Specifies the name of the cluster to which the ruleset applies. The **-clusterName** parameter is required only if you modify an ODR cluster. (String, optional)

Batch mode example usage

- Using Jacl:

```
$AdminTask removeDefaultAction {-odrname odr -nodename node1 -rulesetName myRuleset -  
actionName defAction}
```

- Using Jython string:

```
AdminTask.removeDefaultAction('-odrname odr -nodename node1 -rulesetName myRuleset -actionName  
defAction')
```

Interactive mode example usage

- Using Jacl:

```
$AdminTask removeDefaultAction {-interactive}
```

- Using Jython string:

```
AdminTask.removeDefaultAction ('[-interactive]')
```

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Rules support](#)

[Rule-based request classification](#)

[Overview of request flow prioritization](#)

Related reference:

[Routing and service policies](#)

Runtime operations user preferences administrative tasks

You can use the Jacl and Jython scripting languages to clone, set, and get preferences with the wsadmin tool. Use the commands and parameters in the UserPreferences group.

User preferences are stored under the cell context or in the context for each resource type. The following list gives examples on the cell level and at each resource level:

- cells/myCell/preferences/User_1/preferenceset.xml
- cells/myCell/applications/TestAppA.ear/preferences/User_1/preferenceset.xml
- cells/myCell/nodes/myNode/servers/myServer/preferences/User_1/preferenceset.xml
- cells/myCell/clusters/myCluster/preferences/User_1/preferenceset.xml
- cells/myCell/serviceclasses/Platinum/preferences/User_1/preferenceset.xml

Use the following commands to work with the preferences for runtime operations:

- [clonePreference](#)
- [getPreferences](#)
- [setPreference](#)

For examples using Jython to interact with preferences objects, see [Jython preferences object commands](#).

clonePreference

The clonePreference command copies a preference from one user or role to another user or role.

Target object: The configID of the parent of where the preference is cloned.

Required parameters

- **-name: Specifies the name of the preference to clone. (String, required)**

Optional parameters

- **-scope: Specifies the scope of the preference to save when the preferences are cloned. If you do not specify a scope, the scope of the configID is used. (String, optional) The scope can be one of the following values:**
 - Applications
 - Servers
 - Clusters
 - ServiceClasses
- **-touser: Specifies the user for which to retrieve preferences. If you do not indicate this value, the default uses the user ID for the session in which the command is running. (String, optional)**
- **-torole: Specifies the role for which to retrieve preferences. This value is ignored if you specified the -touser parameter. (String, optional)**
- **-fromuser: Specifies the user that gets a copy the preferences. You can indicate multiple users by putting the users in a list, for example: user1,user2,user3. Do not put spaces in the list. (String, optional)**
- **-fromrole: Specifies the role that gets a copy of the preferences. (String, optional)**

Batch mode example usage

- Using Jython:

```
cellid= AdminConfig.getid("/Cell:myCell")
AdminTask.clonePreference(cellid, '[-name ChartGroups -fromuser user1 -touser
user2,user3,user4,user5]')
AdminConfig.save()
```

- Using Jacl:

```
AdminTask clonePreference cells/dabtcCell02|cell.xml#Cell_1 {-name ChartGroups -fromuser user1
-touser user2,user3,user4,user5}
```

Interactive mode example usage

- Using Jython:

```
AdminTask.clonePreference(['-interactive'])
```

- Using Jacl:

```
$AdminTask clonePreference {-interactive}
```


getPreferences

The getPreferences command returns a collection of preferences based on the preference document you specify by passing in a role or user name. You must have administrator privileges to get preferences for a certain user or user role.

Target object: the configID of the parent

Required parameters

- **-name: Specifies the name of the preference. (String, required)**

Optional parameters

- **-scope: Specifies a scope for the preferences. You can choose from one of the following valid values:**
 - Applications
 - Servers
 - Clusters
 - ServiceClasses

The default scope is empty, or undefined. If the scope is empty, the scope is derived from where the document is being read. For example, if the document is in the cell scope, then the scope is cell scoped. If the document is in a specific server context, then the scope is assumed to be applicable to that server. (String, optional)

- **-user: Specifies the user name for which to retrieve preferences. If you do not indicate a user, the default action is to use the user name that is running the script. (String, optional)**
- **-role: Specifies the user role for which to retrieve preferences. (String, optional)**

Return value:

The command returns a DescriptivePropertyGroup object that contains the preferences, or null if no preferences are found.

Batch mode example usage

- Using Jython string:

```
cellid= AdminConfig.getid("/Cell:myCell")
AdminTask.getPreferences(cellid, '[-name ChartGroups -user user1]')
```

- Using Jacl:

```
$AdminTask getPreferences cells/dabtcCell02|cell.xml#Cell_1 {-name Reports -user user1}
```

Interactive mode example usage

- Using Jython string:

```
AdminTask.getPreferences('[-interactive]')
```

- Using Jacl:

```
$AdminTask getPreferences {-interactive}
```

setPreference

The setPreference command sets a specific preference in a preference document. You must have administrator privileges to set preferences for a certain user or user role.

Target object: The configID of the parent

Required parameters

- **-name: Specifies the name of the preference. (String, required)**
- **-propertyLongName: Specifies the fully-qualified path to the property in the property group, delimited by a forward slash (/) character. If the property is directly contained in the root property group, then the short name of the property can be used because it would be exactly equal to the long name. (String, required)**
- **-propertyValue: Specifies the new value to set for the preference. (String, required)**

Optional parameters

- **-scope:** Specifies the scope of the preference to save when the preferences are cloned. If you do not specify a scope, the scope of the configID is used. (String, optional) The scope can be one of the following values:
 - Applications
 - Servers
 - Clusters
 - ServiceClasses
- **-user:** Specifies the user name for which to save preferences. If you do not indicate a user, the default action is to use the user name that is running the script. (String, optional)
- **-role:** Specifies the user role for which to save preferences. (String, optional)

Batch mode example usage

- Using Jython string:

```
AdminTask.setPreference(cellid, '[-name myChartGroup -propertyLongName defaultChartSize -propertyValue large]')
```

- Using Jacl:

```
$AdminTask setPreference cells/dabtcCell02|cell.xml#Cell_1 {-name Reports -user user1 -propertyLongName defaultChartSize -propertyValue large}
```

Interactive mode example usage

- Using Jython string:

```
AdminTask.setPreference('[-interactive]')
```

- Using Jacl:

```
$AdminTask setPreference {-interactive}
```

Jython preferences object commands

Use the following examples to interact with the preferences object with Jython commands:

To list preferences per user:

```
prefSets = AdminConfig.list("PreferenceSet")
print prefSets
```

To remove preferences for a user:

```
prefid = AdminConfig.getid("/PreferenceSet:User_1")
AdminConfig.remove(prefid)
AdminConfig.save()
```

To list preferences for all users, grouped by user:

```
prefs = AdminConfig.list("Preferences")
print prefs
```

Parent topic: [Administrator scripting interfaces](#)

Parent topic: [Administrator scripting interfaces](#)

Related concepts:

[Runtime operations overview](#)

Related tasks:

[Monitoring operations](#)

[Creating and managing reports](#)

Custom properties

Custom properties are unique settings in the administrative console. They are name-value pairs that you can enter on specific administrative console pages. Use the documentation for the specific custom property to determine where the property must be set.

Setting cell-wide custom properties

You can configure a cell-wide custom property by completing the following steps:

1. In the administrative console, select **System administration > Cell > Custom properties > New**.
2. Provide a name and value for the cell property.
3. Click **OK**.

Setting custom properties at other levels

Some custom properties can be set at other levels than the cell level. If the custom property documentation specifies another scope at which the custom property must be set, the steps are similar, but you first navigate to the indicated level before you set the custom property.

1. Navigate to the indicated configuration object. For example, if the scope indicated is the dynamic cluster, click **Servers > Dynamic clusters > *dynamic_cluster_name***.
2. Click **Custom properties > New**.
3. Provide a name and value for the cell property.
4. Click **OK**.

[Application edition manager custom properties](#)

By configuring custom properties for application edition manager, you can change the settings for your validation cluster. For example, you can set the operating mode of the validation cluster or create a custom property to keep the validation cluster after the rollout completes.

[Application placement custom properties](#)

Custom properties modify the application placement controller configuration. You can use these settings to tune the application placement controller behavior beyond the settings that are in the administrative console.

[Autonomic controllers custom properties](#)

You can use custom properties to configure the autonomic controllers to always start on a particular node or deployment manager process.

[Autonomic request flow manager custom properties](#)

You can use the following custom properties to change the behavior of the autonomic request flow manager (ARFM). Some custom properties are set on deployment targets.

[Autonomic request flow manager advanced custom properties](#)

You can use these properties to configure the autonomic request flow manager (ARFM).

[Binary trace facility custom properties](#)

You can modify trace dynamically and persistently by setting a cell custom property.

[Core group custom properties](#)

When you use this custom property, you can disable the default behavior of WebSphere® Virtual Enterprise that automatically configures each node agent as a core group bridge.

[Dynamic cluster custom properties](#)

You can use dynamic cluster custom properties to change the behavior of your dynamic clusters and application placement.

[Health controller custom properties](#)

You can use health controller custom properties to change the behavior of your health controller. For example, you can change the amount of time that passes before a runtime task that is generated for a health issue expires.

[Internal state reporting custom properties](#)

You can use internal state reporting custom properties to control the timing of when dumps of internal state occur, if dumps are enabled, if compression is applied to the messages from the internal state, and so on.

[Maintenance mode custom properties](#)

You can use maintenance mode custom properties to modify the behavior of your servers.

[Middleware agent custom properties](#)

You can use custom properties to modify the middleware agent configuration, and to configure timeout values for the middleware agent.

[Middleware server custom properties and variables](#)

Middleware server variables and custom properties define properties for middleware servers, such as the maximum heap size for assisted life-cycle servers.

On demand router system and custom properties

You can use on demand router (ODR) custom properties to change the behavior of your ODR. For example, you can change the error code that the ODR returns when messages are rejected because of processor or memory overload.

Operational alerts custom properties

You can use the operational alerts custom properties to improve the performance of the operational alerts on your runtime reporting and operational summary pages.

Runtime operations custom properties

You can use the runtime operations custom properties to modify runtime reports settings of your multi-cell environment.

Service policy custom properties

You can use service policy custom properties to provide service policy alerting for persistent service policy violations on a transaction class basis. When a violation is reported, a runtime task is generated by a runtime component within WebSphere Virtual Enterprise. A task provides information from which you can accept or deny the suggested action plan.

Runtime task custom properties

You can use runtime task custom properties to customize the e-mails that contain runtime task information.

VMware custom properties

You can use custom properties to configure the product to connect to the VMware Infrastructure SDK (VI SDK) web service endpoints.

Parent topic: [Reference](#)

Application edition manager custom properties

By configuring custom properties for application edition manager, you can change the settings for your validation cluster. For example, you can set the operating mode of the validation cluster or create a custom property to keep the validation cluster after the rollout completes.

Parent topic: [Custom properties](#)

deactivate.checkRoutingRules

You can enable the deactivate.checkRoutingRules custom property if you do not want an application edition deactivation to occur when routing policy work classes are still routing workload to the application edition. If you do not set the property, the deactivation still occurs, even if routing policy work class actions are found that reference the edition.

Table 1. deactivate.checkRoutingRules custom property values

Scope	Cell
Valid values	<p>No value: Deactivated, application editions can be deactivated even if routing policy work classes reference the edition.</p> <p>true: Enabled. If routing policy work classes reference an edition that is being deactivated, the deactivation does not occur until the actions are removed or changed.</p>

saveClonedCluster

You can set the saveClonedCluster custom property to maintain the validation cluster after you perform rollout to a new edition. If you do not set this custom property, the validation cluster is deleted after the rollout or after the validation is canceled. The saveClonedCluster custom property applies only to dynamic clusters.

Set this custom property on the validation cluster. To save the validation cluster, set the value to true.

Table 2. saveClonedCluster custom property values

Scope	Dynamic cluster
Valid values	<p>No value: Deactivated. The validation cluster is deleted after a rollout to a new edition.</p> <p>true: Enabled. The validation cluster is saved when the rollout to a new edition occurs.</p>

uninstall.checkRoutingRules

You can enable the uninstall.checkRoutingRules custom property if you do not want the uninstallation of an application edition to occur when routing policy work classes are still routing workload to the application edition. If you do not set the property, the uninstallation still occurs, even if routing policy work class actions are found that reference the edition.

Table 3. uninstall.checkRoutingRules custom property values

Scope	Cell
Valid values	<p>No value: Deactivated. Application editions can be uninstalled even if routing policy work classes reference the edition.</p> <p>true: Enabled. If routing policy work classes reference an edition that is being uninstalled, the uninstallation does not occur until the actions are removed or changed.</p>

VALIDATION_OPERATIONALMODE

To set the operational mode of the cloned validation cluster to a different mode from the production cluster, create the VALIDATION_OPERATIONALMODE custom property.

You can set this custom property on the production cluster before you start the validation process.

Table 4. VALIDATION_OPERATIONALMODE custom property values

Scope	Dynamic cluster
Valid values	automatic, manual, or supervised

If the value is unspecified or not valid, the validation cluster operational mode is set to the same operational

mode as the production cluster. If you specify any other value or you do not specify a value, then the validation dynamic cluster is set to manual mode.

Related concepts:

[Application edition manager concepts](#)

Related tasks:

[Validating an edition](#)

Related reference:

[Application edition manager states](#)

Application placement custom properties

Custom properties modify the application placement controller configuration. You can use these settings to tune the application placement controller behavior beyond the settings that are in the administrative console.

To set the application placement custom properties, expand **Operational policies > Autonomic managers > Application placement controller > Custom properties**.

Parent topic: [Custom properties](#)

CenterCell custom property

When you are configuring multi-cell performance management in your environment, you can use the CenterCell custom property to designate one cell as the center cell. You also set the CenterCell custom property individually for each cell that you want to designate as a point cell.

Note: One and only one custom property should be set to true.

Table 1. CenterCell custom property values

Scope	Cell
Valid values	true: Designates one cell as the center cell false: Designates one cell as a point cell

apcConcurrentStartSize

You can use the apcConcurrentStartSize custom property to define the maximum number of server instances the application placement controller should attempt to start concurrently on a particular node.

The default number of server instances that the application placement controller attempts to start on a node is one instance. Set this property to a value greater than 1 if you want the application placement controller to start more than one server instance at the same time on a particular node.

Table 2. apcConCurrentStartSize custom property values

Scope	Application placement controller
Valid values	An integer value that specifies the maximum number of instances to be started concurrently on a specific node.
Default	1

APC.XD.failedServerTimePeriod

Starting with version 7.0.0.3, use this custom property to adjust the application placement controller waiting time before removing a server from the list of server start failures.

When a server fails to start, the application placement controller records the server on the list of server start failures. The server is removed from the set of available servers. The application placement controller does not attempt to start the server as long as it appears on the list of server start failures. After a period of time (the default is 60 minutes), the application placement controller automatically removes the server from the list of server start failures. The server is included in the set of available servers. To adjust the application placement controller waiting time before removing a server from the list of server start failures, set the application placement controller custom property (in minutes): APC.XD.failedServerTimePeriod.

Table 3. APC.XD.failedServerTimePeriod custom property values

Value	Description
Default waiting time before automatically removing servers from the list of server start failures	60 minutes
Valid value	minutes

lazyStartMinInstances

You can use the lazyStartMinInstances custom property to configure multiple server instances to start when the on demand router (ODR) detects activity.

Prior to Version 6.1.1.2, only one server instance was started when a dynamic cluster was configured for application lazy start and the ODR received requests for the inactive dynamic cluster. By setting the lazyStartMinInstances custom property, however, multiple instances can be started.

When you set the custom property at the application placement level, the property applies to all of your dynamic clusters. Alternatively, you can set the custom property on a specific dynamic cluster so that multiple instances are started by that particular dynamic cluster. If other dynamic clusters configured for application lazy start exist, those clusters start only one instance each. The custom property value set at the dynamic cluster level overrides the custom property value set at the application placement level.

Table 4. lazyStartMinInstances custom property values

Table 4. lazyStartMinInstances custom property values

Scope	Dynamic cluster
Valid values	An integer value that specifies the minimum number of instances to be lazily started.
Default	1

apc.log.enablePlacementLog

You can use the apc.log.enablePlacementLog custom property to enable or disable logs from being saved to the apc.log log file. The apc.log log file contains information about placement decisions. IBM® support can use this log file to help understand the placement decisions that are being made by the application placement controller.

Table 5. apc.log.enablePlacementLog custom property values

Scope	Application placement controller
Valid values	<p>No value: The placement log is enabled.</p> <p>false: The placement log is disabled.</p>

apc.log.LogFileSize

You can use the apc.log.LogFileSize custom property to set the size of the application placement controller log file, the apc.log file, to a size in megabytes. The default size is 100 megabytes, but you can change the default value.

Table 6. apc.log.LogFileSize custom property values

S c o p e	Application placement controller
V a l i d v a l u e s	<p>An integer value of the number of megabytes for the log file. For example, if you enter 200, each log file will be about 200 megabytes.</p> <p>The Java™ virtual machine (JVM) attempts to limit the log file to a byte size of 200 * 1024 * 1024 or 209,715,200 bytes, which is a smaller limit used by the JVM and is an approximate value. The actual size will likely be larger and varies by each JVM vendor. The actual byte size on the disk depends on the sector and cluster size of the physical disk, which will likely be larger than the byte size of the file.</p>

apc.log.numHistoricalFiles

You can use the apc.log.numHistoricalFiles custom property to set the number of log files that are saved for the application placement controller. When the application placement controller log file reaches its maximum size, the historical file is saved if this value is greater than 1. If the value is equal to 1, the historical data appends to the existing file.

Table 7. apc.log.numHistoricalFiles custom property values

Scope	Application placement controller
Valid values	An integer value that specifies the number of historical files you want to save, for example, 2.

cpuUtilizationThreshold

You can use the cpuUtilizationThreshold custom property to define the percentage of processor power that you can use on each node in a cell.

The cpuUtilizationThreshold custom property defines the percentage of processor power that is used on the nodes in your environment. The default value is 100, which means that 100% of the processor can be used on each node in the cell. For configurations that involve z/OS® nodes, you must set this value to less than 100.

The application placement controller attempts to compute placements so that the utilization of any node does not exceed the defined threshold. The application placement controller considers all processes, including processes that are not related to WebSphere® Application Server or WebSphere Virtual Enterprise when making these calculations.

z/OS Important: This property must be defined and must be set to a value less than 100 for configurations that involve z/OS nodes.

Table 8. cpuUtilizationThreshold custom property values

Scope	Application placement controller: applies to all nodes in the cell.
--------------	---

	Node: applies to a specific node.
Valid values	An integer value that specifies the percentage of processor power to use on each node in the cell.

GenerateUniquePorts

You can use the GenerateUniquePorts custom property to generate unique ports for each dynamic cluster member.

z/OS By default, dynamic cluster members on distributed nodes are created with unique ports, while z/OS nodes do not have unique ports. On the z/OS platform, port numbers are usually set symmetrically according to specific range of numbers. The first dynamic cluster member always has unique ports, and these port numbers are applied to the other dynamic cluster members. You can set this property on the node group or on an individual node.

Table 9. GenerateUniquePorts custom property values

Scope	Node group, node
Valid values	true: Unique ports are generated for each dynamic cluster member. false: Unique ports are created for the first dynamic cluster member, and subsequent dynamic cluster members use the same ports.
Default	z/OS platforms: false Attention: If vertical stacking is enabled for a z/OS dynamic cluster, the port allocation strategy is forced to true, to generate unique ports for each dynamic cluster member. Distributed platforms: true

maintenanceModeOnOperationFail

You can use the maintenanceModeOnOperationFail custom property to move a server into maintenance mode when the server start fails. The default value is false, which, instead of placing the server into maintenance mode when a server start fails, generates a runtime task. This runtime task contains a notification that the application placement controller did not receive notification about the completion of the server start.

Table 10. maintenanceModeOnOperationFail custom property values

Scope	Application placement controller
Valid values	true: Places the server into maintenance mode when the server start fails. false: Generates a runtime task when a server start fails.
Default	false

memoryProfiler.isDisabled

You can use the memoryProfile.isDisabled custom property to disable the memory profiler.

Under the default operation, the application placement controller profiles dynamic clusters online to determine their memory usage. You can disable this default operation by setting the memoryProfiler.isDisabled custom property to true. When the memory profiler is disabled, the application placement controller assumes a default memory usage value. This memory usage value is computed as $64k + 1.2 \text{ maxHeapSize}$, where maxHeapSize is the maximum heap size setting in the server template of the dynamic cluster.

Table 11. memoryProfiler.isDisabled custom property values

Scope	Application placement controller
Valid values	true: Disables the memory profiler. false: Enables the memory profiler.
Default	false

OverlappingCells

You can use the OverlappingCells custom property when two WebSphere Virtual Enterprise cells share the same physical host. Configuring this property ensures that the application placement controller does not overuse resources on the shared host.

See [Configuring application placement for cells that share the same nodes](#) for more information about how to set this custom property.

Table 12. *OverlappingCells* custom property values

Scope	Application placement controller
Valid values	<p>Enter the custom property value in the following format:</p> <pre style="background-color: #f0f0f0; padding: 5px;"><code>user_name:password@deployment_manager_host:port.SOAP</code></pre> <p>If you are connecting three or more cells, separate the custom property values for each cell with a comma.</p> <ul style="list-style-type: none"> • <i>user_name</i>: Specifies the name of the user that can log into the other cell. • <i>password</i>: Specifies the password for that user. You can encode the password so that it does not display in plain text in the administrative console. To encode a password, use the encodePassword script. • <i>deployment_manager_host</i>: Specifies the name of the deployment manager host for the other cell. • <i>port</i>: Specifies the SOAP port that is associated with the deployment manager, for example, the SOAP_CONNECTOR_ADDRESS port.

reservedMemoryFixed and reservedMemoryPercent

You can use these custom properties to define how much memory on each node to reserve for processes that are not related to WebSphere Application Server or WebSphere Virtual Enterprise.

When these properties are defined, the total memory on a node for processes that are not related to WebSphere Application Server or WebSphere Virtual Enterprise can be calculated using the following formula:

$$\text{reservedMemoryFixed} + \text{reservedMemoryPercent} * (\text{totalNodeMemory} - \text{reservedMemoryFixed})$$

The reservedMemoryFixed custom property specifies a fixed amount of memory to reserve. The reservedMemoryPercent custom property specifies a percentage of memory to reserve.

Table 13. *reservedMemoryFixed* and *reservedMemoryPercent* custom property values

Scope	Application placement controller
Valid values	<p>To set the value for the entire cell: Type the name of the custom property as reservedMemoryFixed or reservedMemoryPercent.</p> <p>To set the value for a particular node: Type the name of the custom property, either reservedMemoryFixed.<i>node_name</i> or reservedMemoryPercent.<i>node_name</i>, where <i>node_name</i> is the name of your node. For example, if your node is named node1, the names of your custom properties are reservedMemoryFixed.node1 and reservedMemoryPercent.node1.</p>
Default	<p>reservedMemoryFixed: 64 MB</p> <p>reservedMemoryPercent: 5%</p>

unsetMaintenanceModeAfterStart

Use this custom property to take a server out of maintenance mode after the server starts. If the maintenanceModeOnOperationFail custom property is enabled, the server is placed into maintenance mode after the server start fails. When the server starts successfully, you can use this property to move the started server out of maintenance mode automatically.

Table 14. *unsetMaintenanceModeAfterStart* custom property values

Scope	Application placement controller
Valid values	<p>true: Moves the started server out of maintenance mode automatically.</p> <p>false: Leaves the server in maintenance mode after it is started.</p>
Default	false

useNodeFreeMemory

Use this custom property to define a replacement to the reservedMemory default custom property. This custom property uses the node free memory statistic, plus the memory calculated for any dynamic cluster instances that are already on the node, to calculate the memory available for launching additional dynamic cluster instances.

Table 15. useNodeFreeMemory custom property values

Scope	Application placement controller
Valid values	true: Automatically calculates available memory. false: Use reservedMemory default custom properties to define available memory.

Related tasks:

[Creating dynamic clusters](#)

[Configuring dynamic application placement](#)

[Configuring multi-cell performance management](#)

Related information:

[Dynamic cluster custom properties](#)

Autonomic controllers custom properties

You can use custom properties to configure the autonomic controllers to always start on a particular node or deployment manager process.

To force a controller to always start on a deployment manager or a node agent process, set the `HAManagedItemPreferred_component_variable` custom property on the specific deployment manager or node.

Complete the following steps to set the custom property on a node agent process.

1. In the administrative console, click **System administration > Nodes > *node_name* > *node_agent_name* > Java and process management > Process definition > Java virtual machine > Custom properties**.
2. Enter the name of the custom property.
3. Set the value of the custom property to `true`.
4. Click **Apply**, and save your changes.

Complete the following steps to set the custom property on a deployment manager process.

1. In the administrative console, click **System administration > Deployment manager > Java and process management > Process definition > Java virtual machine > Custom properties**.
2. Enter the name of the custom property.
3. Set the value of the custom property to `true`.
4. Click **Apply**, and save your changes.

Examples

To configure the application placement controller to start on a deployment manager or a node agent process, set the following custom property:

```
HAManagedItemPreferred_apc=true
```

Note: When you are using elasticity mode, you must set the `HAManagedItemPreferred_apc` custom property on the deployment manager or node that will not be removed. Otherwise, the application placement controller will not operate. Also, you must restart the current process in which the application placement controller is running after you set the custom property.

To configure the autonomic request flow manager to start on a deployment manager or a node agent process, set the following custom property:

```
HAManagedItemPreferred_arfm=true
```

To configure the health management controller to start on a deployment manager or a node agent process, set the following custom property:

```
HAManagedItemPreferred_hmm=true
```

To configure the high availability plugin-cfg generation service to start on a deployment manager or a node agent process, set the following custom property:

```
HAManagedItemPreferred_PluginCfgGenerator=true
```

To configure the VMware activity publisher to start on a deployment manager or a node agent process, set the following custom property:

```
HAManagedItemPreferred_vmshimactivitypublisher=true
```

When you use elasticity mode in an environment in which multi-cell performance management is configured, you must configure certain controllers to start on the deployment managers of the center cell and the point cells. To configure the application placement controller to start on the deployment manager of the center cell, set the following custom property:

```
HAManagedItemPreferred_apc=true
```

To configure the cell agent to start on the deployment managers of the point cells, set the following custom property:

```
HAManagedItemPreferred_cellagent=true
```

Note: You must restart the current process in which the controller is running after you set each custom property.

Parent topic: [Custom properties](#)

Related concepts:

[Dynamic operations](#)

Related tasks:

[Configuring the autonomic managers](#)

[Configuring multi-cell performance management](#)

[Generating the plug-in configuration in a high availability environment](#)

[Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

Autonomic request flow manager custom properties

You can use the following custom properties to change the behavior of the autonomic request flow manager (ARFM). Some custom properties are set on deployment targets.

Parent topic: [Custom properties](#)

CenterCell custom property

When you are configuring multi-cell performance management in your environment, you can use the CenterCell custom property to designate one cell as the center cell. You also set the CenterCell custom property individually for each cell that you want to designate as a point cell.

Note: One and only one custom property should be set to true.

Table 1. CenterCell custom property values

Scope	Cell
Valid values	true: Designates one cell as the center cell false: Designates one cell as a point cell

enableRateBasedARFM

ARFM uses a rate-based algorithm that results in a more consistent loading and protecting of application server resources. The rate-based ARFM feature is enabled by default. To disable the feature, set the enableRateBasedARFM custom property to false.

Table 2. AC5.BurstFactor custom property

Scope	Cell
Valid values	Boolean
Default	False

AC5.BurstFactor

You can set this value to change how tolerant memory overload protection (MOP) is to bursts of traffic. If the maximum rate is one unit per second and the burst factor is one, then one unit of non-affinity traffic is admitted if it did not admit one in the last second. If the burst factor is three however, it admits three units every three seconds.

Table 3. AC5.BurstFactor custom property

Scope	On demand router
Valid values	Integer
Default	3

arfmlgnoreHttpSessionsForCpu

You can set this custom property to true so that the ARFM ignores the HTTP dialog structure for the purposes of CPU overload protection. Every incoming HTTP message is subject to admission control.

Table 4. arfmlgnoreHttpSessionsForCpu custom property

Scope	Cell
Valid values	Boolean
Default	False

arfmlgnoreSipDialogsForCpu

You can set this custom property to true so that the ARFM ignores the Session Initiation Protocol (SIP) dialog structure for the purposes of processor overload protection. Every incoming SIP message is subject to admission control.

Table 5. arfmlgnoreSipDialogsForCpu custom property

Scope	Cell
Valid values	Boolean
Default	False

arfmManageCpu

CPU overload protection is on by default. Set the value of this custom property to false to disable CPU overload protection and request prioritization.

Table 6. arfmManageCpu custom property

--	--

Scope	Cell
Valid values	Boolean (true or false)
Default	True

arfmManualAllocation

You can specify this custom property on a deployment target to put ARFM into manual mode when the custom property `arfmMode` is not set to manual. The value of the property is a string in a syntax that allows expression of allocations.

Table 7. *arfmManualAllocation* custom property

Scope	Deployment target (server, dynamic cluster, and so on)
Valid values	String
Default	No default

arfmMode

You can use this custom property to specify the operating mode of the ARFM. In automatic mode, the ARFM has an autonomic controller that adjusts the dispatching priority of requests from the on demand routers to the servers and sends the placement controller information about the computational needs for the various dynamic clusters. In manual mode, you can override the autonomic controller with administrative settings.

Table 8. *arfmMode* custom property

Scope	Cell
Valid values	Manual, Automatic
Default	Automatic

CPUAdjustment

You can set this custom property on nodes that are on hyperthreaded computers. When the `CPUAdjustment` custom property is specified on a node, a correction function to apply to processor utilization readings is specified for that node. This function is applied to the processor utilization that is reported by the operating system, and yields a more accurate utilization measure.

Table 9. *CPUAdjustment* custom property

Scope	Node
Valid values	String
Default	No default

goodServiceTimeLimitSpec

Use a `goodServiceTimeLimitSpec` custom property when configuring the ARFM. The default value of one minute for determining timeouts is used if `goodServiceTimeLimitSpec` or `timeoutFactor` is not set. To override the default, set this custom property to customize the value for different levels of granularity. For example, service class, application, or module. To create a general rule of 5 minutes, create the custom property and set the value to `"*:*:*:*=300"` where the last part is the timeout value of 300 seconds. The timeout value is a decimal number, not necessarily a whole number, and is in units of seconds.

Table 10. *goodServiceTimeLimitSpec* custom property

Scope	Cell
Valid values	Decimal number in unit of seconds
Default	60 if <code>timeoutFactor</code> is not set. Otherwise, no default.

magicNMode

You can set this custom property when ARFM is in automatic mode. When in automatic mode, concurrency limiting is performed according to one of two criteria, depending if the `magicNMode` custom property is enabled. When the `magicNMode` custom property is enabled, the total number of requests that are running concurrently at any given time is limited to a certain total. In normal mode, the total number of requests depends on the use of processor power on the nodes.

Table 11. *magicNMode* custom property

Scope	Cell
Valid values	Boolean
Default	False

magicN

You can use this custom property when ARFM has the `magicNMode` custom property is enabled, to the limit on the total number of concurrent requests. If this property is not defined, a reasonable limit is estimated from other data.

If you are using the reasonable limit estimation, the limit on the total number of concurrent requests for a cell is the sum of a contribution from each node in the cell. The contribution from a node is the average, over the running application servers on that node, of the maximum size of the thread pool of the Web container on that application server.

Table 12. *magicN* custom property

Scope	Cell
Valid values	A positive integer defines the number of concurrent requests. To use the reasonable estimate, set the property value to a negative number or delete the custom property.
Default	-1

maxHttpLiveRequests

You can use this custom property to define the maximum number of requests that are targeted for a particular cell and could be anywhere in the cell. That is, either queued in an ODR or running on an application server at a given time.

Table 13. *maxHttpLiveRequests* custom property

Scope	Cell
Valid values	Integer
Default	No default

MOP.AveragingWindowInMS

The averaging window in milliseconds that is used by MOP. This time interval should be at least as long as the lifetime of 95 percent of the HTTP sessions, SIP dialogs, and application sessions. If the time interval is too short, MOP might not protect against memory overload. If the time interval is much longer than the lifetime of these objects, MOP still protects against memory overload but MOP takes a longer amount of time to ramp up to learn the maximum rate.

Table 14. *MOP.AveragingWindowInMS* custom property

Scope	Cell
Units	milliseconds
Default	180000 ms

MOP.InitialMaxRatePerSec

If no persisted maximum rate value exists, the value of this custom property is used as the initial maximum rate.

Table 15. *MOP.AveragingWindowInMS* custom property

Scope	Cell
Default	1

node.speed

You can use this property to define the node speed in MHz. Set the value to the processor speed, multiplied by the number of processors on a multi-processor node. Set this custom property on the node. When you are running a node agent and a middleware agent on the same node, the `node.memory` value can be overridden. Define the custom property for both agents and set them to equal values.

Table 16. *node.speed* custom property

Scope	Node
Valid values	A floating point number that represents the number of MHz of the reference instruction set.
Default	No default

node.memory

You can use this property to specify the available RAM defined in megabyte units on a node. Set this custom property on the node. Although this value is determined automatically, you can override the value.

Table 17. *node.memory* custom property

Scope	Node
Valid values	A number of megabytes
Default	No default

node.numCPUs

You can use this custom property to specify the number of processors on a multi-processor node. This value is automatically determined, but you can override the value. Set this custom property on the node.

Table 18. *node.numCPUs* custom property

Scope	Node
Valid values	Integer that represents the number of processors
Default	No default

timeoutFactor

Specify the `timeoutFactor` custom property to indicate a value that is multiplied by the response time threshold to determine the fine-grained timeout threshold. Use a `timeoutFactor` custom property when configuring the ARFM. The `timeoutFactor` value is multiplied by the response time threshold to yield the fine-grained timeout threshold.

This fine-grained timeout threshold is one of three different factors that the ARFM uses to determine that a running request has timed out. The running request has timed out when one of the following factors is true:

- The proxy framework indicates the request timed out.
- The request reply includes a special HTTP header that indicates that the internal application timed out.
- The service time exceeds the fine-grained timeout value.

Table 19. *timeoutFactor* custom property

Scope	Cell
Valid values	Decimal representation of a floating point number
Default	No default

useODRs

Specifies if the work profiler and placement controller rely on the services of the ODR servers.

Table 20. *useODRs* custom property

Scope	Cell
Valid values	Boolean
DefaultRT	True

Related concepts:

[Dynamic operations environment](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

[Configuring the autonomic managers](#)

[Configuring multi-cell performance management](#)

Related information:

[Autonomic request flow manager advanced custom properties](#)

Autonomic request flow manager advanced custom properties

You can use these properties to configure the autonomic request flow manager (ARFM).

Parent topic: [Custom properties](#)

Work profiler properties

profilerPeriod

Specifies a number of milliseconds, specified for each cell that specifies the length of the work profiler cycle.

Table 1. profilerPeriod custom property values

Scope	Cell
Valid values	Integer value for number of milliseconds
Default	60000 (1 minute)

profilerHalfLife

Specifies a number of minutes, specified for each cell. The work profiler discounts observations by an exponential function of time. The half life is the amount of time over which the discount changes by a factor of 2.

Table 2. profilerHalfLife custom property values

Scope	Cell
Valid values	Integer value for number of milliseconds
Default	600000 (10 minutes)

publishedAlphasPrintFrequency

Specifies the frequency at which work factors are printed to the SystemOut.log file. These work factors are printed to the log file during the work profiler cycle, which is a length of time that is specified with the profilerPeriod custom property. If you want the work profiler to print work factors to the log file during each of these cycles, then you can specify the value as 1. However, if you want to decrease the amount of text being printed to the log file, you can increase this value. For example, if you want the work factors to be printed after every 5 work profiler cycles, then you can specify the value as 5.

Table 3. publishedAlphasPrintFrequency custom property values

Scope	Cell
Valid values	Integer value greater than or equal to zero
Default	0 (Work factors are not printed to the SystemOut.log file.)

Work profiler output decay half life and smoothing weight function parameters

The work profiler works in two passes: first it fits the observations to a simple model to extract preliminary work factors, then it smooths the work factors by taking a weighted average. Each weight is the product of two factors, one of which diminishes the preliminary work factor importance with age and the other that varies with the goodness of the first pass fit. The age factor is an exponential decay; the half-life is the amount of time over which that factor drops by a factor of 2. This parameter is given in the cell with the profilerAlphaSmoothingHalfLife custom property, with a value that is the decimal notation for an integer, a number of milliseconds. The default is 10 minutes. To adjust the goodness level, two parameters are used, a threshold and a factor. The threshold is defined with the goodnessWeightThresholdcell custom property. The factor is given in the cell with the goodnessWeightFactor custom property.

Table 4. Work profiler output decay half life and smoothing weight function parameters

Property name	Value	Default
profilerAlphaSmoothingHalfLife	a decimal notation for an integer in a number of milliseconds	600000 (10 minutes)
goodnessWeightThreshold	a non-negative floating point number	20
goodnessWeightFactor	a non-negative floating point number	20

Work factor overrides

You can override the values that are computed by the work profiler. The work profiler computes a work factor for each transaction class and deployed Java™ Platform, Enterprise Edition (Java EE) module pair (TCM). The work factor is a floating point number that represents the number of megacycles of the reference instruction set.

You can override work factors by adding the custom property to the dynamic cluster.

For nodes that are not running WebSphere® Application Server, you can specify Magic N for the cell, specify the work factor for each related TCM pair, or put the autonomic request flow manager manual mode and specify the various control knob settings. Use the following grammar for defining work factor override specifications:

```
spec ::= case ( "," case )*
case ::= pattern "=" value
pattern ::= service-class ":" txn-class ":" application ":" module
service-class ::= step
txn-class ::= step
application ::= step
module ::= step
step ::= name | "*"
value ::= number | "none"
```

Examples are provided in the following table:

Table 5. Work factor override specs

Example Spec	Description
::*:* = none	Specifies that every transaction class module (TCM) in the deployment target has no override. The deployment target has only one tier, and the value is calculated in the normal way for each case.
::*:* = 42	The deployment target has one tier. Every TCM in the deployment target has a work factor override for the tier, equal to 42 megacycles per request.
Platinum:*:*:* = 42, *:*:*:* = none	The deployment target has one tier. There is an override of 42 megacycles per request for transaction class modules that have a Platinum service class, and no overrides for transaction class modules that are assigned to any other service class in the deployment target.
*:tc_A: *:*=42, *:tc_B: Account Management:MicroWebApp.war=17, *:tc_B: *:*=none	There is an override of 42 megacycles per request for TCMs that have the tc_A transaction class. For any TCMs that have the tc_B transaction class, a deployed Java EE application named AccountManagement, and a Java EE module named MicroWebApp.war, there is an override of 17 megacycles per request. There is no override for any other TCMs that have the tc_B transaction class. This example does not consider transaction classes other than the tc_A or tc_B transaction classes and if it encounters another transaction class, an error message is displayed.
::*:* = none, *:*:*:* :./DbCell/CICS = 0.7	There is no override for the first tier. For the tier that is named CICS+1, a work factor override of 0.7 exists. The CICS+1 tier is the first tier in the CICS® deployment target, in the DbCell, regardless of the target TCM. The transaction class does not change from tier to tier, but the module might change.

External placement

An external cluster is used for foreign and generic servers that are not referenced by generic server clusters and are not monitored by the remote agent. Database servers are an example. An external cluster cannot be a target and cannot contain target servers. An external cluster server can run on any kind of machine. You must specify the placement of the external cluster servers, and the speeds of their unmonitored nodes by using a custom property on the ODR cell.

Use the following grammar to specify the value of the external.placement custom property:

```
spec ::= nodespec ( ";" nodespec )*
nodespec ::= nodeName ":" [ nodeSpeed ] ":" plmtlist
plmtlist ::= dtName ( "," dtName)*
nodeName ::= name ( "/" name )*
```

nodeSpeed ::= number

The grammar and interpretation for a dtName is the same as in the work factor override spec, except that in this situation, the relevant cell contains the node with a specified nodespec that contains the dtName. The grammar and interpretation for node names follows the same pattern. Following are some examples :

Table 6.

Spec value	Description
CicsNode: 8.6 : CICS	In addition to the WebSphere Application Server nodes and targets, there is one node that is not running WebSphere Application Server, named "CicsNode" in the same cell where this property appears. The computational power of that node is 8.6. There is one deployment target not associated with WebSphere Application Server on that node; named "CICS". One server process exists for that DT on that node.
../SysX /DBA:4 .7:DB1 ,DB2; ../SysX /DBB:2 .7:DB2	There are two nodes that are not running WebSphere Application Server, named DBA and DBB in the cell named SysX, with power of 4.7 and 2.7 respectively. Two deployment targets exist, named DB1 and DB2®, in cell SysX. DB1 is placed only on node DBA, and DB2 is placed on nodes.

Use per-process CPU readings

Set the useProcessCPU custom property to true to enable the ARFM controller and application placement controller to consider background work when computing the required resources and enable per-process CPU utilization statistics. When this property is set to false, the work profiler cannot estimate work factors as well because it uses CPU utilization readings for the whole node.

Table 7. useProcessCPU custom property for Version 6.0.2 and later

Name	Property Setting	Default	Valid Values
useProcessCPU	Set this custom property on the cell.	true	true or false

MustGather documents

Use the WebSphere Virtual Enterprise mustGather documents to troubleshoot the autonomic request flow manager and application placement. The Support team provides and maintains the [mustGather documents](#) for each version of WebSphere Virtual Enterprise.

Related tasks:

[Configuring the autonomic request flow manager](#)

Binary trace facility custom properties

You can modify trace dynamically and persistently by setting a cell custom property.

Set cell custom property with the following form for name and value:

Table 1. trace custom property

Scope	Cell
Name	trace.<scopeName>
Valid values	[<condition>;;<traceSpec>

Examples of changing trace dynamically

You can modify trace dynamically and persistently by setting a cell custom property. For example, to trace HTTP affinity processing and load balancing, use the following settings:

Table 2. trace HTTP affinity processing and load balancing

Scope	Cell
Name	trace.http
Valid values	http.request.affinity,http.request.loadBalance

Or, to trace server weight changes (which are traced by default), specify the following settings:

Table 3. trace HTTP affinity processing and load balancing

Scope	Cell
Name	trace.ODCEvent
Valid values	propertyType = 'server:weight';;ODC.event

The ODC language supports a nodeType operand, propertyType operand which returns <nodeName>: <propertyName>, and edgeType operand which returns <parentNodeType>:<childNodeType>. All ODC events are in the group ODC.event, so as long as they pass the condition test, they will be traced.

To enable all HTTP trace except load balancing, turn on all HTTP trace, then turn off http.request.loadBalance trace with a trace level of 0, as shown in the following table:

Table 4. trace HTTP without load balancing

Scope	Cell
Name	trace.http
Valid values	http,http.request.loadBalance=0

To trace all SIP messages that take longer than 100 milliseconds to be processed by the on demand router (ODR), specify the following settings:

Table 5. trace HTTP without load balancing

Scope	Cell
Name	trace.sip
Valid values	sip.message.slow=100

Specifying these settings traces all SIP messages that take longer than 100 milliseconds to be processed by the ODR. Note that in this case, the trace level field is used as a time threshold field that controls when the message is traced. The record description describes how the level value is used in this case: A SIP message took longer than the level (in milliseconds) to be processed by the ODR. The default level is 300. Record descriptions should be clear in describing how the level is used and what each level means for the record.

Parent topic: [Custom properties](#)

Related concepts:

[Options for filtering information in the trace-cfg.xml](#)

Related tasks:

[Using the binary trace facility](#)

Core group custom properties

When you use this custom property, you can disable the default behavior of WebSphere® Virtual Enterprise that automatically configures each node agent as a core group bridge.

Parent topic: [Custom properties](#)

xd.coregroup.cgb.disable

In the default configuration, each node agent is also a core group bridge in a multiple core group topology. With this default configuration, any time you create a new core group, the core group is automatically bridged with the other core groups in the cell. You can disable this behavior by setting the custom property to true. When you set the property to true, any existing core group bridge configuration remains. If you want to remove a core group bridge from a node agent, you must manually make the configuration change in the administrative console. By disabling the behavior, any additional node agents that are added to the cell are not core group bridges. In addition, any core groups that you add to the cell are not automatically bridged with the other core groups.

Table 1. *xd.coregroup.cgb.disable* custom property values

Scope	Cell
Valid values	true: Node agents are not automatically configured as core group bridges. false: Node agents are automatically configured as core group bridges. The coregroupsplit.py script sets this property to true.
Default	Not specified, which is the same behavior as setting the custom property to false.

Parent topic: [coregroupsplit.py script](#)

Parent topic: [Custom properties](#)

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Enabling cell affinity](#)

[Configuring ODRs](#)

[Preparing the hosting environment for dynamic operations](#)

[Creating and configuring ODRs](#)

[Configuring SSL offload for all HTTPS traffic](#)

Related reference:

[createodr.jacl script](#)

Dynamic cluster custom properties

You can use dynamic cluster custom properties to change the behavior of your dynamic clusters and application placement.

Parent topic: [Custom properties](#)

quiesceTimeoutMS custom property

You can use the **quiesceTimeoutMS** custom property to set the quiesce timeout value for dynamic cluster instances.

Set the value of the custom property to the amount of time, in milliseconds, to wait before a dynamic cluster is stopped. For example, if you want a dynamic cluster to stop after 1 minute of quiescing, set the value to 60000. If servers are being stopped by the application placement controller, the server operation timeout value is used by default. If servers are being stopped by health management monitoring, the restart timeout value is used by default.

Table 1. *quiesceTimeoutMS* custom property values

Scope	Dynamic cluster
Valid values	Integer

CenterCell custom property

When you configure multi-cell performance management in your environment, you can use the **CenterCell** custom property to designate one cell as the center cell. You also set the CenterCell custom property individually for each cell that you want to designate as a point cell.

Note: One and only one custom property should be set to true.

Table 2. *CenterCell* custom property values

Scope	Cell
Valid values	true: Designates one cell as the center cell false: Designates one cell as a point cell

lazyStartMinInstances custom property

You can use the **lazyStartMinInstances** custom property to configure multiple server instances to start when the on demand router (ODR) detects activity for an inactive dynamic cluster.

Prior to Version 6.1.1.2, only one server instance was started when a dynamic cluster was configured for application lazy start and the ODR received requests for the inactive dynamic cluster. By setting the **lazyStartMinInstances** custom property on a specific dynamic cluster, however, multiple instances can be started by that particular dynamic cluster. If other dynamic clusters configured for application lazy start exist, those clusters start only one instance each.

Alternatively, you can set the custom property on the application placement controller so that all of your dynamic clusters are affected. However, the custom property value set at the dynamic cluster level overrides the custom property value set at the application placement level.

Table 3. *lazyStartMinInstances* custom property values

Scope	Dynamic cluster
Valid values	An integer value that specifies the minimum number of instances to be lazily started.
Default	1

lazystart.http.status.code custom property

Use the **lazystart.http.status.code** custom property to specify a custom HTTP status code. You can specify this custom property at cell level or at cluster level. The on demand router returns the value of this custom property when a request is sent to a dynamic cluster, which is in automatic or supervised mode, and where lazy start is enabled. A custom HTTP status code is not returned for dynamic clusters in manual mode, even if lazy start is enabled. The custom property at cell level applies to all dynamic clusters. The custom property set at cluster level takes precedence over the custom property set at cell level. For **lazystart.http.status.code** property changes, it is not required to restart the on demand router.

Table 4. *lazystart.http.status.code* custom property

Scope	Cluster or cell
Value	Integer

equalCPUFactor custom property

You can use the **equalCPUFactor** custom property to tell to the dynamic workload manager (DWLM) how to equalize the performance of the servers in a dynamic cluster.

The DWLM calculates dynamic weights for the servers in a given dynamic cluster to equalize the performance of the servers. The two common measures of server performance are:

- Average service time for a request that is sent to the server.
- The processor utilization of the node.

With the custom property, you can specify the relevance of one of these measures over the other. For example, if you want consistency in how the users of your site perceive the site performance, you might choose to optimize average service time. If you are more concerned with the usage of your hardware, you might choose processor utilization as your measure of performance.

To place most priority on equalizing average service time, set the value of the custom property to 0. To equalize processor utilization of the node, set the value to 1. To use a blend of both measures with a relative weight to each measure, set the value to a fraction value between 0 and 1, for example 0.4. By setting the value to a fraction, you are placing a relative weight of 0.4 to the processor utilization and 1-0.4, or 0.6 relative weight to the average service time.

Equalizing both measures at the same time might not be possible at all times. For example, in an environment where servers are heterogeneous or have varying amount of unequal background load, equalizing the processor utilization might result in unequal average service time. A fast and a slow server running at the same processor utilization can result in short and long average request service time, respectively. A request that spends a significant amount of time in one of several servers in a deeper tier might result in a different average service time. This variance can occur depending on the server to which the request was sent, even if the servers in the deeper tier are homogeneous and have equal processor utilization. Other situations exist where the service time of a request depends on resources other than the processor. The value of the **equalCPUFactor** custom property helps the DWLM controller determine a weighted measure of both average service time and processor utilization to be equalized.

Even without the **equalCPUFactor** custom property, the processor utilization of the servers in a given dynamic cluster has an effect on the behavior of the DWLM controller. In general, when processor utilization is low, equal distribution of load takes precedence over equalizing the performance of service time or utilization. Gradually, as the utilization goes up, equalizing performance begins to take precedence. At very high processor utilization values, the weights tend to not change as much to avoid instability. When processor utilization is high, the sensitivity of performance on load distribution at that extreme operating point increases.

Table 5. *equalCPUFactor* custom property values

Scope	Cell, which applies to all dynamic clusters in the cell, or at the individual dynamic cluster level. If the custom property is specified at both the dynamic cluster and cell level, dynamic cluster level value overrides the value that is specified at the cell level.
Valid values	A fraction value between 0 and 1.
Default	0 for non-virtualized environments and 1 for virtualized environments.

HttpSessionRebalanceOff custom property

You can use the **HttpSessionRebalanceOff** custom property to disable HTTP session rebalancing.

HTTP session rebalancing is automatically enabled. You can use HTTP session rebalancing to reassign existing session affinities to new servers that become available for processing the given Web application. See [HTTP session rebalancing](#) for more information.

Use the **HttpSessionRebalanceOff** custom property if you want to return your configuration to the old HTTP session behavior, where session affinities are established with a particular application server and are not reassigned to any new servers that become available.

Session rebalancing is disabled by default on any dynamic clusters that consist of servers that are not running WebSphere® Application Server, such as PHP or Tomcat servers, because you might have another session clustering mechanism deployed for those servers.

You might consider disabling HTTP session rebalancing if your session sizes are large. If your sessions are large, the cost of moving the sessions to a new server might be more than the benefit of taking the workload off of the original server. You can use Performance Monitoring Infrastructure (PMI) data to make the decision to turn off session rebalancing. You might see in your PMI data that response time, memory utilization, and processor utilization increases on specific servers to transfer the session information. See the [WebSphere Application Server Network Deployment Information Center](#) for more information about analyzing PMI data and best practices for using HTTP sessions.

If you leave session rebalancing on, as sessions become more evenly distributed, and memory and processor utilization also become more evenly distributed across the servers in the cluster. If a cluster is more balanced, it is easier for WebSphere Virtual Enterprise to make autonomic decisions.

Table 6. *HttpSessionRebalanceOff* custom property values

Scope	Dynamic cluster
Valid values	true: Disables HTTP session rebalancing. false: Enables HTTP session rebalancing. If you want to disable HTTP session rebalancing for WebSphere Application Server application servers, you can delete the custom property.
Default	For dynamic clusters that consist of WebSphere Application Server application servers: false (enabled) For dynamic clusters that consist of servers that are not WebSphere Application Server application servers, such as PHP or Tomcat servers: true (disabled)

numVerticalInstances custom property

Use this custom property to define the number of dynamic cluster instances on a node.

Use this custom property only if the nodes in your dynamic cluster are heterogeneous and vary in power. If the nodes in your dynamic cluster are homogenous, you can define the number of dynamic cluster instances one time in the administrative console.

See [Configuring a dynamic cluster with heterogeneous nodes to support vertical stacking](#) for more information.

Table 7. *numVerticalInstances* custom property values

Scope	Dynamic cluster
Name format	Specify the name of the custom property as <code>numVerticalInstances.node_name</code> , where <code>node_name</code> is the name of your node.
Valid values	Integer value for the stacking number

proactiveIdleStop custom property

You can use the `proactiveIdleStop` custom property to stop dynamic cluster instances during periods of inactivity.

This custom property adds function to the **If other dynamic clusters need resources, stop all instances of this cluster during periods of inactivity** setting in the administrative console. This setting must be enabled with this custom property. With the administrative console setting, instances stop only if other clusters in the cell need the resources that are being used by the inactive instances. You also specify an amount of time to wait before stopping instances for the cluster. By setting this custom property, inactive instances stop even if the resources are not required elsewhere in the environment. The cluster instance goes idle in the amount of time that you specified in the administrative console setting.

The application placement controller stops the instance at some point in time between the amount of time that you specified in the administrative console setting plus the value specified for **Minimum time between placement changes** setting on the Application Placement Controller configuration panel. For example, if you set the `proactiveIdleStop` custom property to true, the value of **Minimum time between placement changes** is 15 minutes, and the **Time to wait before stopping instances** value is 5 minutes, the dynamic cluster instance is stopped sometime between 5 minutes and 20 minutes after the last request was routed to the instance.

Table 8. *proactiveIdleStop* custom property values

Scope	Dynamic cluster
Valid values	Boolean
Default	false (disabled)

z/OS

serverSpecificShortNames custom property

For z/OS® platforms, the `serverSpecificShortNames` custom property is specified on the dynamic cluster to indicate the specific short names of cluster members in a comma-separated list format, for example: SSN1,SSN2. Use commas to separate multiple short names. If you do not provide enough short names to be used for all of the cluster members, then the remaining cluster members are assigned generated generic short names, such as BB0S001, BB0S002, and so on.

Table 9. *serverSpecificShortNames* custom property values

Scope	Dynamic cluster
--------------	-----------------

scope	Dynamic cluster
Valid values	Comma-separated list of short names for dynamic cluster members, for example SSN1,SSN2

Related concepts:

[Dynamic clusters](#)

Related tasks:

[Configuring multi-cell performance management](#)

[Creating dynamic clusters](#)

[Monitoring and tuning the application placement controller](#)

[Monitoring and tuning health management](#)

Related reference:

[Dynamic cluster administrative tasks](#)

Related information:

[Application placement custom properties](#)

Health controller custom properties

You can use health controller custom properties to change the behavior of your health controller. For example, you can change the amount of time that passes before a runtime task that is generated for a health issue expires.

Parent topic: [Custom properties](#)

usexdHeapModule

You can set the usexdHeapModule custom property so that only the amount of free heap after a full garbage collection is sampled. Otherwise, the heap might include temporary Java™ objects.

In releases prior to Version 7.0, the excessive memory health policy calculated the amount of heap used by a Java virtual machine (JVM) by periodically sampling the amount of free heap.

Table 1. usexdHeapModule custom property valid value and default

Scope	Cell
Valid values	true: Only the amount of free heap after a full garbage collection is sampled. false: The amount of free heap is periodically sampled independent of the garbage collection cycle.
Default value	false

_CEE_DMPTARG

The _CEE_DMPTARG custom property specifies the location that is used for thread dumps on z/OS® systems.

You must set the _CEE_DMPTARG custom property value if you configure health policies that perform thread dumps on z/OS systems. This custom property sets the location where the thread dumps occur for a particular server.

Table 2. _CEE_DMPTARG custom property valid value and default

Scope	Application server
Valid value	A string that indicates the location where you want to save the thread dumps

com.ibm.ws.xd.hmm.controller.approvalTimeoutMinutes

The com.ibm.ws.xd.hmm.controller.approvalTimeoutMinutes custom property specifies the number of minutes for the approval timeout of runtime tasks for the health controller.

You can use the com.ibm.ws.xd.hmm.controller.approvalTimeoutMinutes custom property when you are using the supervise reaction mode. This property specifies the number of minutes that can pass before a runtime task for the health controller expires. If you do not take any actions on the runtime task, the task expires in the number of minutes that is specified in this property. If the runtime task expires when the health condition still exists, a new runtime task is generated.

Table 3. com.ibm.ws.xd.hmm.controller.approvalTimeoutMinutes custom property valid value and default

Scope	Cell
Valid value	Minutes
Default	30 minutes

Related concepts:

[Health management](#)

Related tasks:

[Configuring health management](#)

[Creating health policies](#)

[Configuring the health controller](#)

Related reference:

[Health policy administrative tasks](#)

[healthpolicy.py script](#)

Internal state reporting custom properties

You can use internal state reporting custom properties to control the timing of when dumps of internal state occur, if dumps are enabled, if compression is applied to the messages from the internal state, and so on.

When you change internal state reporting custom properties, you must restart your processes to use the most recently defined values.

Parent topic: [Custom properties](#)

bbson.DUMP_TIMEOUT custom property

Specifies the time, in minutes, between dumps of bulletin board over structured overlay network (BBSON) internal state. The default is to never perform a BBSON internal state dump. These state dumps go into the ProfileRoot/bbdumps/ directory. This property is only valid if you are using the WebSphere® Virtual Enterprise implementation of the bulletin board.

Table 1. *bbson.DUMP_TIMEOUT* custom property default, units, and scope

Scope	Cell
Default	0 (never dump)
Units	minutes

overlayNetStatsDumpPeriod custom property

Specifies the time, in seconds, between dumps of structured overlay network (SON) statistics. Set the value to 0 (the default) to disable dumps of SON statistics.

Table 2. *overlayNetStatsDumpPeriod* custom property default, units, and scope

Scope	Cell
Default	0 (do not perform dumps)
Units	seconds

son.compress custom property

Specifies if the compression of messages and network traffic from the SON is enabled.

Compression is enabled by default. To disable compression, set this custom property on the cell with a value of false.

Table 3. *son.compress* custom property default, valid values, and scope

Scope	Set this custom property at the cell-wide level.
Default	true (enabled)
Valid values	true or false

son.FailNeighborAfterLostkHeartbeats custom property

Defines the number of missed heartbeats after which a neighbor process is considered to be in a failed or missing state.

Table 4. *son.FailNeighborAfterLostkHeartbeats* custom property default, valid values, and scope

Scope	Set this custom property at the cell-wide level.
Default	6
Valid values	Integer value greater than 0

son.NeighborHeartbeatPeriod custom property

Defines the time interval, in milliseconds, between heartbeat packets that are sent to neighbor processes.

Table 5. *son.NeighborHeartbeatPeriod* custom property default, valid values, and scope

Scope	Set this custom property at the cell-wide level.
Default	20000 ms
Valid values	milliseconds (ms)

Related reference:

[Custom properties](#)

Maintenance mode custom properties

You can use maintenance mode custom properties to modify the behavior of your servers.

Parent topic: [Custom properties](#)

maintenanceModeSyncContainingNodeOnly

Use the `maintenanceModeSyncContainingNodeOnly` custom property to limit which nodes are synchronized.

When you set the custom property at the cell level and set maintenance mode on a server, only the node on which the server resides is synchronized. If this property is not set, all active nodes are synchronized.

Table 1. `maintenanceModeSyncContainingNodeOnly` custom property values

Scope	Cell
Valid values	true: Sets maintenance mode on a server and synchronizes only the node on which the server resides. false: All active nodes are synchronized.
Default	false

maintenanceModeOnOperationFail

Use the `maintenanceModeOnOperationFail` custom property to move a server into maintenance mode when the server start fails. The default value is `false`, which, instead of placing the server into maintenance mode when a server start fails, generates a runtime task. This runtime task contains a notification that the application placement controller did not receive notification about the completion of the server start.

Table 2. `maintenanceModeOnOperationFail` custom property values

Scope	Application placement controller
Valid values	true: Places the server into maintenance mode when the server start fails. false: Generates a runtime task when a server start fails.
Default	false

unsetMaintenanceModeAfterStart

Use this custom property to take a server out of maintenance mode after the server starts. If the `maintenanceModeOnOperationFail` custom property is enabled, the server is placed into maintenance mode after the server start fails. When the server starts successfully, you can use this property to move the started server out of maintenance mode automatically.

Table 3. `unsetMaintenanceModeAfterStart` custom property values

Scope	Application placement controller
Valid values	true: Moves the started server out of maintenance mode automatically. false: Leaves the server in maintenance mode after it is started.
Default	false

Related tasks:

[Setting maintenance mode](#)

Middleware agent custom properties

You can use custom properties to modify the middleware agent configuration, and to configure timeout values for the middleware agent.

Parent topic: [Custom properties](#)

ServerStartupTimeoutMillis

The ServerStartupTimeoutMillis custom property specifies the amount of time that the middleware agent waits, during the middleware server startup, before the middleware agent times out and stops trying to start the server. If the middleware server starts before the timeout value, the agent stops waiting and continues with the startup processing.

Table 1. ServerStartupTimeoutMillis custom property default, units, and scope

Scope	Cell, node, or server level
Default	75000 (75 seconds)
Units	milliseconds

ServerShutdownTimeoutMillis

The ServerShutdownTimeoutMillis custom property specifies the amount of time that the middleware agent uses to determine whether a server has successfully stopped.

Table 2. ServerShutdownTimeoutMillis custom property default, units, and scope

Scope	Cell, node, or server level
Default	75000 (75 seconds)
Units	milliseconds

MiddlewareAgentProcessScanIntervalMillis

The MiddlewareAgentProcessScanIntervalMillis custom property specifies a time interval at which the middleware agent scans the configured servers to reestablish monitors for all the running middleware servers. If you set the value to zero, the repeated scans are disabled.

Table 3. MiddlewareAgentProcessScanIntervalMillis custom property default, units, and scope

Scope	Cell, node, or server level
Default	60000 (60 seconds, or one minute)
Units	milliseconds

ServerInitializedTimeoutMillis

The ServerInitializedTimeoutMillis custom property is used to configure the middleware server initialization time. After a middleware server starts, additional time for the server to initialize is required. If the middleware agent reports that the middleware server has started too soon, HTTP requests that are directed to that middleware server might fail. You can use this custom property so that the middleware agent does not report that the server has started for a fixed period.

config: This custom property is supported for WebSphere® Virtual Enterprise servers only.

Table 4. ServerInitializedTimeoutMillis custom property default, units, and scope

Scope	Cell, node, or server level
Default	30000 (30 seconds)
Units	milliseconds

Related concepts:

[Middleware agent](#)
[Middleware nodes and servers](#)

Related tasks:

[Adding middleware servers to configurations](#)

Middleware server custom properties and variables

Middleware server variables and custom properties define properties for middleware servers, such as the maximum heap size for assisted life-cycle servers.

Parent topic: [Custom properties](#)

SERVER_MAX_HEAP_SIZE variable

The `SERVER_MAX_HEAP_SIZE` variable defines the maximum heap size for assisted life-cycle servers. If the variable is not defined, the default maximum heap size value for the server type is used.

To set the `SERVER_MAX_HEAP_SIZE` variable, click **Servers > All servers > server_name > Variables**. Specify the maximum heap size in megabytes.

Table 1. Default values for `SERVER_MAX_HEAP_SIZE` variable

Server type	Maximum heap size
JBoss	512 MB
Apache Tomcat	64 MB
WebSphere® Application Server Community Edition	64 MB
BEA WebLogic	400 MB

UseJMXSecureConnector custom property

The `UseJMXSecureConnector` custom property specifies if Java™ Management Extensions (JMX) security is enabled for a WebSphere Application Server Community Edition Version 2.1 server.

To enable JMX security, additional steps are required beyond setting the custom property. See [Configuring JMX security for WebSphere Application Server Community Edition Version 2.1 servers](#) for more information.

To set the `UseJMXSecureConnector` custom property, click **Servers > Other middleware servers > WebSphere Application Server Community Edition servers > wasce_server_name > Custom properties > New**. Set the value to true to enable JMX security.

Table 2. `UseJMXSecureConnector` custom property

Value	True or false
Default	false

Related concepts:

[Middleware nodes and servers](#)

Related tasks:

[Configuring assisted life-cycle WebSphere Application Server Community Edition servers](#)

[Configuring JMX security for WebSphere Application Server Community Edition Version 2.1 servers](#)

[Configuring JBoss servers](#)

[Configuring Apache Tomcat servers](#)

[Configuring BEA WebLogic servers](#)

On demand router system and custom properties

You can use on demand router (ODR) custom properties to change the behavior of your ODR. For example, you can change the error code that the ODR returns when messages are rejected because of processor or memory overload.

Parent topic: [Custom properties](#)

http.log.history custom property

Use the **http.log.history** custom property to specify the number of history server log files. The server log files are the on demand router (ODR) custom log files, as well as the proxy.log, local.log, and cache.log files. Without this custom property, you have one history file for the server log files. Setting this custom property does not affect the custom logs that are already initialized.

Table 1. http.log.history custom property values

Scope	ODR or ODR cluster
Value	Any positive integer greater than 0
Default	1

http.log.maxSize custom property

Use the **http.log.maxSize** custom property to specify the maximum log size in megabytes (MB). A value of UNLIMITED indicates unlimited. As with the **http.log.history** custom property, the **http.log.maxSize** custom property applies to the ODR custom log files, as well as the proxy.log, local.log, and cache.log files

Table 2. http.log.maxSize custom property values

Scope	ODR or ODR cluster
Value	Any positive integer greater than 0
Default	25 MB

CenterCell custom property

When you are configuring multi-cell performance management in your environment, you can use the **CenterCell** custom property to designate one cell as the center cell. You also set the **CenterCell** custom property individually for each cell that you want to designate as a point cell.

Note: One and only one custom property should be set to true.

Table 3. CenterCell custom property values

Scope	Cell
Valid values	true: Designates one cell as the center cell false: Designates one cell as a point cell

ODR.heapUsage.max

Use the **ODR.heapUsage.max** custom property to configure a percentage value that determines at what amount of heap usage the ODR rejects requests. If heap usage exceeds 90%, the ODR rejects any incoming request and a 503 error code is returned.

Table 4. ODR.heapUsage.max custom property

Scope	Cell
Value	Percentage
Default	90%

ODR.heapUsage.algorithm

Use the **ODR.heapUsage.algorithm** custom property to configure when heap usage is calculated. If you set the value to globalGC, the amount of heap usage is calculated only immediately after a global garbage collection cycle. If you set the value to request, the amount of heap usage is calculated with each request.

Table 5. ODR.heapUsage.algorithm custom property

Scope	Cell
Value	Algorithm
Default	globalGC

ODR.heapUsage.errorCode

Use the **ODR.heapUsage.errorCode** custom property to configure the type of error code that is returned when a request is received and the heap usage exceeds the maximum threshold.

Table 6. *ODR.heapUsage.errorCode* custom property

Scope	Cell
Value	Integer
Default	503

http.overload.error

Use the **http.overload.error** custom property to configure a custom error code for the autonomic request flow manager to return when an HTTP message over TCP or SSL is rejected due to processor or memory overload. If you do not configure this custom property, the default 503 error code is returned.

Table 7. *http.overload.error* custom property

Scope	On demand router
Value	Integer
Default	503

sip.overload.error

Use the **sip.overload.error** custom property to configure a custom error code for the autonomic request flow manager to return when a Session Initiation Protocol (SIP) message over Transmission Control Protocol (TCP) or Secure Sockets Layer (SSL) is rejected because of processor or memory overload. If you do not configure this custom property, the default 503 error code is returned.

Table 8. *sip.overload.error* custom property

Scope	On demand router
Value	Integer
Default	503

sipu.overload.error

Use the **sipu.overload.error** custom property to configure a custom error code for the autonomic request flow manager to return when a Session Initiation Protocol (SIP) message over User Datagram Protocol (UDP) is rejected due to processor or memory overload. If you do not configure this custom property, the default 503 error code is returned.

Table 9. *sipu.overload.error* custom property

Scope	On demand router
Value	Integer
Default	503

http.partialResponseBodyBufferSize

Use the **http.partialResponseBodyBufferSize** custom property to disable the ODR from buffering the size of the response payload when you perform a rollout on an application edition. Set the **http.partialResponseBodyBufferSize** custom property to a value of 0.

Table 10. *http.partialResponseBodyBufferSize* custom property

Scope	On demand router
Value	Integer
Default	0

System properties

Follow the instructions to set specific system properties for the ODR.

ODCLeftGroupDelay

Use the `ODCLeftGroupDelay =0` system property set to allow no delay for on demand configuration detection of down servers during failover when servers become unavailable.

From the administrative console, select **Servers > On Demand Routers > on_demand_router > Java and Process Management > Process Definition > Java Virtual Machine**.

ODR.traceRouteHeader

Use the `ODR.traceRouteHeader` system property on the ODR, and it will add a response header with the route that the request took.

From the administrative console, select **Servers > On demand routers > on_demand_router > Java and process management > Process definition > Java Virtual Machine**.

For example, `-DODR.traceRouteHeader=TraceRoute` adds a response header with the name `TraceRoute` to each

request. The value of this response header is the path that the request took.

Related concepts:

[Overview of request flow prioritization](#)

Related tasks:

[Configuring ODRs](#)

[Configuring multi-cell performance management](#)

[Performing a rollout on an edition](#)

[Preparing the hosting environment for dynamic operations](#)

[Creating and configuring ODRs](#)

[Creating ODRs](#)

[Configuring SSL offload for all HTTPS traffic](#)

Related reference:

[createodr.jacl script](#)

Operational alerts custom properties

You can use the operational alerts custom properties to improve the performance of the operational alerts on your runtime reporting and operational summary pages.

You can tune the performance of your operational alerts by setting any of the following custom properties on the cell. Set the properties that you want to a value that is different from the default value.

To set the operational alerts custom properties on the cell:

1. In the administrative console, click **System administration > Cell > Custom properties > New**.
2. Type the name of the custom property in the format, `opalert.property_name`, for example, `opalert.disableCache`.
3. Indicate a value for the custom property. Use the possible values in the following list of custom properties to determine the value that you use.
4. Save the custom property. The custom property is retrieved dynamically. You do not need to restart the deployment manager. The custom property is set for all the resources in the cell.

To set a custom property on a specific resource:

1. In the administrative console, click **System administration > Cell > Custom properties > New**.
2. Type the name of the custom property in the format, `opalert.resource_typeproperty_name`. The `resource_type` value is the name of the core group or core component on which you want to set the property. Use the name for the core component or core group that is listed in the administrative console. For example, you might use the name, `opalert.ARFMController.disableCache`, to set the property on the autonomic request flow manager (ARFM).
3. Indicate a value for the custom property. Use the possible values in the following list of custom properties to determine the value that you use.
4. Save the custom property. The custom property is retrieved dynamically. You do not need to restart the deployment manager. The custom property is used only for the resource that you indicated.

Parent topic: [Custom properties](#)

opalert.disableCache

When set to `true`, this custom property disables caching, which also disables any performance gains. You can use this custom property when you are debugging.

Table 1. `opalert.disableCache` custom property

Value	True or false
Default	false

opalert.disableAlerts

When set to `true`, this custom property disables the cacheable providers from adding operational alerts. To view detailed information about items such as core groups and core components, go to the operations view that is specific to the core group or core component.

Table 2. `opalert.disableAlerts` custom property

Value	True or false
Default	false

opalert.disableXDStability

When set to `true`, this custom property disables the cacheable provider from contributing to the overall status. To view detailed information about items such as core groups and core components, go to the operations view that is specific to the core group or core component.

Table 3. `opalert.disableXDStability` custom property

Value	True or false
Default	false

opalert.initPolicy

This custom property determines if the request waits for the initialization of the cache to complete before returning a status message. The first request for the information causes caching to occur. If the initialization policy is set to `blocking`, the initialization of the cache must complete before returning a status message. If the initialization policy is set to `nonblocking`, then an UNKNOWN status message is returned. You can refresh the operational alerts page or operations view when the cache has finished initializing. This property is only used when the cache initializes.

Table 4. `opalert.initPolicy` custom property

Value	blocking or nonblocking
Default	nonblocking

opalert.disableAutoRefresh

By default, the caches are automatically refreshed on a time interval. If this property is set to true, the caches are not automatically refreshed, and do not refresh until a new request comes in after the caches have been destroyed.

Table 5. *opalert.disableAutoRefresh* custom property

Value	true or false
Default	false

opalert.initPeriod

If the `opalert.disableAutoRefresh` property is enabled, this custom property defines how long to wait after an initialization is completed to start initialization again.

Table 6. *opalert.initPeriod* custom property

Value	Integer, in seconds
Default	10 seconds

opalert.destPolicy

The destruction policy specifies when to destroy the provider caches.

- Specify `inactivity` to destroy the caches if the cache is not accessed for the amount of time that is specified in the `opalert.destPeriod` custom property.
- Specify `fixedTimeout` to destroy the cache after the amount of time that is specified in the `opalert.destPeriod` custom property passes after the first initialization period. This value does not consider when the last cache access time is. If you have the `opalert.disableAutoRefresh` enabled, consider setting this custom property to `fixedTimeout`.

Table 7. *opalert.destPolicy* custom property

Value	<code>inactivity</code> or <code>fixedTimeout</code>
Default	<code>inactivity</code>

opalert.destPeriod

This custom property specifies the destruction period in seconds. The destruction period works with the `opalert.destPolicy` custom property to determine how long to wait, either after the first initialization or the last access, before destroying the cache.

Table 8. *opalert.destPeriod* custom property

Value	seconds
Default	600 seconds (10 minutes)

Related concepts:

[Runtime operations overview](#)

Related tasks:

[Monitoring operations](#)

[Creating and managing reports](#)

Related reference:

[Custom properties](#)

Runtime operations custom properties

You can use the runtime operations custom properties to modify runtime reports settings of your multi-cell environment.

You can modify how runtime operations reports for multiple cells are displayed by setting any of the following custom properties on the cell. If you want to use a value other than default, create a custom property.

Parent topic: [Custom properties](#)

cellExclusion

The cellExclusion custom property specifies a list of cells to exclude from the operations dashboard view. You cannot exclude the local cell. If you do not set the cellExclusion custom property, the default is to include all cells.

Table 1. cellExclusion custom property

Scope	Cell
Value	A list of cell names separated by commas.

maxApplicationSummaryDataSets

The maxApplicationSummaryDataSets custom property specifies the maximum number of data sets that are displayed in the application summary chart. If you do not set the maxApplicationSummaryDataSets custom property, the default number of data sets that are displayed is 5.

Table 2. maxApplicationSummaryDataSets

Scope	Cell
Value	Integer
Default	5

maxDeploymentTargetSummaryDataSets

The maxDeploymentTargetSummaryDataSets custom property specifies the maximum number of data sets that are displayed in the deployment target summary chart. If you do not set the maxDeploymentTargetSummaryDataSets custom property, the default number of data sets that are displayed is 5.

Table 3. maxDeploymentTargetSummaryDataSets custom property

Scope	Cell
Value	Integer
Default	5

maxServicePolicySummaryDataSets

The maxServicePolicySummaryDataSets custom property specifies the maximum number of data sets that are displayed in the service policy summary chart. If you do not set the maxServicePolicySummaryDataSets custom property, the default number of data sets that are displayed is 5.

Table 4. maxServicePolicySummaryDataSets custom property

Scope	Cell
Value	Integer
Default	5

topDataSetInterval

The topDataSetInterval custom property specifies how many milliseconds of statistics are used to calculate the top number of data sets to display in the summary charts. If you do not set the topDataSetInterval custom property, the last 120000 milliseconds, or 2 minutes, of statistics are used by default. If you set the custom property to 0 milliseconds, the data sets are returned in the order they are received by the on demand configuration (ODC).

Table 5. topDataSetInterval custom property

Scope	Cell
Value	Integer
Default	Last 120000 milliseconds

destroyStatisticCacheDelay

The destroyStatisticCacheDelay custom property specifies the delay in milliseconds between when WebSphere® Virtual Enterprise detects zero active sessions in the administrative console and when the statistic cache is destroyed. If no custom property exists, the default delay is set 0 milliseconds (0 seconds).

Table 6. *destroyStatisticCacheDelay* custom property

Scope	Cell
Value	Integer
Default	0 milliseconds

Related concepts:

[Runtime operations overview](#)

Related tasks:

[Monitoring operations](#)

[Creating and managing reports](#)

Related reference:

[Custom properties](#)

Service policy custom properties

You can use service policy custom properties to provide service policy alerting for persistent service policy violations on a transaction class basis. When a violation is reported, a runtime task is generated by a runtime component within WebSphere® Virtual Enterprise. A task provides information from which you can accept or deny the suggested action plan.

Parent topic: [Custom properties](#)

disableResponseTimeGoals

Use the `disableResponseTimeGoals` custom property to configure service policies without response time goals when multiple service policies are configured, and the response time goals cannot be accurately determined for any of the service policies.

Providing correct response time goals requires that load testing be performed to determine a reasonable value. However, you can configure service policies without response time goals to eliminate the need for performance testing. To determine when a new dynamic cluster instance should be started when there are no response time goals, a utility value is calculated based upon the fraction of time that requests wait. The utility value is a value between -1 and +1, based on the importance of a service class. Requests with zero wait time receive a +1 utility value, and requests with almost all time spent waiting receive a -1 utility value. A utility value of zero is obtained if the fraction of wait time equals the relative importance of the request service class, where relative importance is zero for very high importance and one for very low importance.

Table 1. `disableResponseTimeGoals` custom property

Scope	Cell
Value	true

detectTxcViolations

Use the `detectTxcViolations` custom property when you want persistent service policy violations to be shown at the transaction class level.

The autonomic request flow manager (ARFM) will use the properties you specify using this custom property. To enable the monitoring and reporting of persistent service policy violations by a transaction class (defined within a service class), set the `detectTxcViolations` custom property to `true`. For the runtime task to be generated and enabled for monitoring, this custom property must be set to `true`, and the service policy must be enabled through the administrative console.

Table 2. `detectTxcViolations` custom property

Scope	Cell
Value	true
Default	false

detectSvcViolations

You can use the `detectSvcViolations` custom property when you want persistent service policy violations to be shown at the service class level. You can set the value of `detectSvcViolations` to `false` in order to disable runtime tasks for persistent service policy violations at the service class level, while enabling the runtime tasks for violations at the transaction class level by setting `detectTxcViolations` to `true`.

The autonomic request flow manager (ARFM) uses the properties you specify using this custom property. Otherwise, it will monitor for only service class violations. To enable the monitoring and reporting of service policy violations within a service class, set the `detectSvcViolations` custom property to `true`, or allow it to keep its default value of `true`. For the runtime task to be generated and enabled for monitoring, this custom property must be set to `true`, and the service policy must be enabled through the administrative console.

Table 3. `detectSvcViolations` custom property

Scope	Cell
Value	true
Default	true

Related tasks:

[Managing runtime tasks](#)

[Monitoring operations](#)

[Defining a service policy](#)

Runtime task custom properties

You can use runtime task custom properties to customize the e-mails that contain runtime task information.

Requirement: To apply these custom properties, you must install the appropriate fix packs for your WebSphere® Virtual Enterprise release and your WebSphere Application Server Network Deployment release:

Table 1. Required interim fixes to customize e-mail notification

WebSphere Virtual Enterprise Version	WebSphere Application Server Network Deployment Version	Required interim fixes
6.0.2.1	6.0.2.x	PK55085
6.0.2.1	6.1.0.11 or 6.1.0.13	PK54881 and PK55085
6.0.2.1	6.1.0.15 or later	PK55085
6.0.2.7	6.0.2.35 or 6.0.2.37	No interim fix is required
6.1.0.0	6.1.0.11 or 6.1.0.13	PK54881 and PK55085
6.1.0.0	6.1.0.15 or later	PK55085
6.1.0.1	6.1.0.11 or 6.1.0.13	PK54881
6.1.0.1	6.1.0.15 or later	No interim fix is required

Parent topic: [Custom properties](#)

terminated.task.remove.timeout

You can use the `terminated.task.remove.timeout` custom property to change the time that expired task entries or actioned tasks stay in the runtime task list, which is 24 hours by default. The value is expressed in milliseconds. For example, 600000 milliseconds = 10 minutes.

Table 2. `terminated.task.remove.timeout` custom property values

Scope	Cell
Valid values	milliseconds
Default	24 hours

task.email.global.sender.id

You can use the `task.email.global.sender.id` custom property to change the e-mail address from which e-mail notifications are sent. Set the value to the specific user ID that you want to use to sent e-mails.

Table 3. `task.email.global.sender.id` custom property values

Scope	Cell
Valid values	user ID
Default	wasxd

task.email.custom.body

You can set the `task.email.custom.body` custom property to customize the details of task notifications in the generated runtime task e-mails.

Table 4. `task.email.custom.body` custom property values

S c o p e	Cell
V a l i d v a l u e s	<p>false: Deactivated. Customized information for task notifications is not provided in the generated runtime task e-mails.</p> <p>true: Enabled. The default value is overridden by a detailed list of task attributes. The attributes and formatting cannot be configured.</p> <p><i>custom_value</i>: Customized override. The default is overridden by a customized list of task attributes that you provide. If predefined variables exist within the custom string, the variables are replaced with actual task values. The following list contains the predefined variables:</p> <ul style="list-style-type: none"> • {TASK_ID}: The unique ID of the task • {TASK_TARGETS}: The targets of the task • {CELL_NAME}: The cell from which the task was generated • {TASK_REASON}: The reason the task was generated • {TASK_SUBMITTER}: The name of the entity that submitted the task • {TASK_SEVERITY}: The severity of the task • {EMPTY_STRING}: An empty string used for an empty subject or body to be defined

D e f a u l t	true
--	------

task.email.custom.subject

You can use the task.email.custom.subject custom property to enable a custom subject line for generated runtime task e-mails.

When enabled, the e-mail subjects are in the following format:

[target name, target name, ...] - submitter - originated time - severity

Table 5. task.email.custom.subject custom property values

S c o p e	Cell
V a l i d v a l u e s	<p>false: Deactivated. A custom subject for task notifications is not provided.</p> <p>true: Enabled. The default value is overridden by a detailed list of task attributes. The attributes and formatting cannot be configured.</p> <p><i>custom_value</i>: Customized override. The default is overridden by a customized list of task attributes that you provide. If predefined variables exist within the custom string, the variables are replaced with actual task values. The following list contains the predefined variables:</p> <ul style="list-style-type: none"> • {TASK_ID}: The unique ID of the task • {TASK_TARGETS}: The targets of the task • {CELL_NAME}: The cell from which the task was generated • {TASK_REASON}: The reason the task was generated • {TASK_SUBMITTER}: The name of the entity that submitted the task • {TASK_SEVERITY}: The severity of the task • {EMPTY_STRING}: An empty string used for an empty subject or body to be defined
D e f a u l t	false

Related concepts:

[Task management](#)

Related tasks:

[Defining email notification](#)

VMware custom properties

You can use custom properties to configure the product to connect to the VMware Infrastructure SDK (VI SDK) web service endpoints.

Parent topic: [Custom properties](#)

cpuLimitForCappedVMs

The `cpuLimitForCappedVMs` custom property defines a the target processor utilization percentage for WebSphere® Virtual Enterprise processes that run on virtual machines. This custom property value is maintained for virtual machines that have a defined processor utilization limit.

You can specify the `cpuLimitForCappedVMs` custom property on the cell with a percent value between 0 and 95%. If you specify 100%, the limit is equal to the limit that is placed on the virtual machine in the VMware software.

The custom property value is the target utilization that WebSphere Virtual Enterprise maintains for virtual machines that have processor utilization limits. This value should be the highest value that maintains virtual machine and system stability.

If your virtual machines are constantly being overloaded, lower the `cpuLimitForCappedVMs` value. To determine if a virtual machine is being overloaded, search for the following conditions in your environment:

- The virtual machine is running at 100% of its processor utilization limit for a long period of time.
- The average service time of requests is going up, indicating that requests in the application server are being queued.

vmware.service.unique_id.url

The `vmware.service.unique_id.url` custom property specifies the Uniform Resource Locator (URL) of the VMware web services endpoint.

For example, you might enter the following value for this property:

```
https://myserver.ibm.com/sdk
```

The `unique_id` value in the custom property name is optional. You can specify the `unique_id` value if you are specifying multiple sets of custom properties for each VMware server. For example, you might name the custom property as `vmware.service.vmwareserver1.url`. As a convention, you can specify the hostname of the vCenter or ESX server for the `unique_id` value.

vmware.service.unique_id.userid

The `vmware.service.unique_id.userid` custom property specifies the user ID to access the vCenter or ESX server.

For the `vmware.service.unique_id.userid` custom property, the following privileges are required by Virtual Enterprise to read certain properties and to perform various operations:

- System.Anonymous
- System.Read
- System.View
- Sessions.TerminateSession

For example, you might enter the following value for this property:

```
root
```

The `unique_id` value in the custom property name is optional. You can specify the `unique_id` value if you are specifying multiple sets of custom properties for each vCenter or ESX server. For example, you might name the custom property as `vmware.service.vmwareserver1.userid`. As a convention, you can specify the hostname of the vCenter or ESX server for the `unique_id` value.

vmware.service.unique_id.password

The `vmware.service.unique_id.password` custom property specifies the password to access the vCenter or ESX server.

For example, you might enter the following value for this property:

```
myPassword
```

The value can be an encoded password. To encode your password, you can use the `encodePassword.sh|bat` script. This script is in the `install_root/bin` directory.

The `unique_id` value in the custom property name is optional. You can specify the `unique_id` value if you are specifying multiple sets of custom properties for each vCenter or ESX server. For example, you might name the

custom property as `vmware.service.vmwareserver1.password`. As a convention, you can specify the hostname of the vCenter or ESX server for the `unique_id` value.

Related concepts:

[VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

Related tasks:

[Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)

Related reference:

[encodePassword.sh|bat script](#)

[☞ VMware Infrastructure SDK](#)

Logs

You can use log files to help with performance monitoring, accounting and problem determination.

You can examine the following log files:

- Visualization:
 - DeploymentTargetStatsHistoricCache
 - NodeStatsHistoricCache
 - ServerStatsCache
 - TCModuleInstanceStatsCache
 - TCModuleStatsCache
 - TierStatsCache
 - FineGrainedPowerConsumptionStatsCache
 - ServerPowerConsumptionStatsCache

[NodeStatsHistoricCache](#)

This log file contains historic information on the node statistics cache.

[ServerStatsCache](#)

This log file describes the server statistics cache.

[DeploymentTargetStatsHistoricCache](#)

This log file contains historic information on the deployment target statistics historic cache.

[TCModuleStatsCache](#)

This log file contains information about the transaction class module cache.

[TCModuleInstanceStatsCache](#)

This log file describes the transaction class module instance cache.

[TierStatsCache](#)

This log file describes the tier statistics cache.

[FineGrainedPowerConsumptionStatsCache](#)

This log file contains fine grained power and work consumption data. A record is written for every transaction class module and server instance. This action creates a record for every middleware application, module, transaction class, and server instance that has had work routed through an on demand router (ODR). Additional fields exist that hold relationship information such as the cluster to which the server belongs, the node group with which the cluster is associated, and the service policy with which the transaction class is associated.

[ServerPowerConsumptionStatsCache](#)

This file is a consolidation of the FineGrainedPowerConsumptionStatsCache at the server level with some additional server data.

[Strftime format conversions](#)

The format used when using the `%{format}t` log parameter is based on the non-extended BSD `strftime(3)` time conversion functions. The parameters that are specifically supported, and sample output, are listed in the following table.

[DeploymentTargetStatsHistoricCache](#)

This log file contains historic information on the deployment target statistics historic cache.

[NodeStatsHistoricCache](#)

This log file contains historic information on the node statistics cache.

[ServerStatsCache](#)

This log file describes the server statistics cache.

[TCModuleStatsCache](#)

This log file contains information about the transaction class module cache.

[TCModuleInstanceStatsCache](#)

This log file describes the transaction class module instance cache.

[TierStatsCache](#)

This log file describes the tier statistics cache.

[FineGrainedPowerConsumptionStatsCache](#)

This log file contains fine grained power and work consumption data. A record is written for every transaction class module and server instance. This action creates a record for every middleware application, module, transaction class, and server instance that has had work routed through an on demand router (ODR). Additional fields exist that hold relationship information such as the cluster to which the server belongs, the node group with which the cluster is associated, and the service policy with which the transaction class is associated.

[ServerPowerConsumptionStatsCache](#)

This file is a consolidation of the FineGrainedPowerConsumptionStatsCache at the server level with some

additional server data.

Parent topic: [Reference](#)

Related tasks:

[Configuring the visualization data service](#)

Strftime format conversions

The format used when using the `%{format}t` log parameter is based on the non-extended BSD `strftime(3)` time conversion functions. The parameters that are specifically supported, and sample output, are listed in the following table.

Selected locale-specific parameters are supported as American English only. Specifically, the order in which month, day, and year appear, and perhaps the ordering of other items, remain American English in all locales. The spelling of words such as the day of the week, or the name of the month, and the timezone will be locale-correct.

Semantics of conversion characters in format string:

Table 1. Conversion characters

Conversion string	Description	Example
<code>%A</code>	Replaced by national representation of the full weekday name.	'Thursday'
<code>%a</code>	Replaced by national representation of the abbreviated weekday.	'Thu'
<code>%B</code>	Replaced by national representation of the full month name.	'September'
<code>%b</code>	Replaced by national representation of the abbreviated month.	'Sep'
<code>%C</code>	Year, divided by 100 (i.e. the century).	'20'
<code>%c</code>	Replaced by national representation of time and date.	'Thu Sep 25 22:32:00 EDT'
<code>%D</code>	Equivalent to <code>%m/%d/%y</code>	'09/25/08'
<code>%d</code>	Replaced by the day of the month as a decimal number (01-31).	'25'
<code>%e</code>	Replaced by the day of the month as a decimal number (1-31).	'25'
<code>%F</code>	Equivalent to <code>%Y-%m-%d</code> .	'2008-09-25'
<code>%G*</code>	The ISO 8601 week-based year (see NOTES) with century as a decimal number. The 4-digit year corresponds to the ISO week number (see <code>%V</code>). This has the same format and value as <code>%Y</code> , except that if the ISO week number belongs to the previous or next year, that year is used instead.	'2008'
<code>%g*</code>	Replaced by the same year as in <code>%G</code> , but as a decimal	'08'

	number without century (00-99).	
%H	Replaced by the hour (24-hour clock) as a decimal number (00-23).	'22'
%h	Equivalent to %b.	'Sep'
%I	Replaced by the hour (12-hour clock) as a decimal number (01-12).	'10'
%j	Replaced by the day of the year as a decimal number (001-366).	'269'
%k	Replaced by the hour (24-hour clock) as a decimal number (0-23); single digits are preceded by a blank.	'22 (or '1' for 1am'
%l	Replaced by the hour (12-hour clock) as a decimal number (1-12); single digits are preceded by a blank.	'10'
%M	Replaced by the minute as a decimal number (00-59).	'32'
%m	Replaced by the month as a decimal number (01-12).	'09'
%n	Replaced by a newline character.	
%P	Replaced by AM or PM.	'PM'
%R	Equivalent to %H:%M.	'22:32'
%r	Equivalent to %I:%M:%S %p .	'10:32:00 pm'
%S	Replaced by the second as a decimal number (00-60).	'00'
%s	Replaced by the number of seconds since the Epoch, UTC (see mktime(3)).	'1222396320'
%T	Equivalent to %H:%M:%S	'22:32:00'
%t	Replaced by a tab character.	
%U	Replaced by the week number of the year (Sunday as the first day of the week) as a decimal number (00-53).	'38'
%u		

	Replaced by the weekday (Monday as the first day of the week) as a decimal number (1-7).	'4'
%V*	Replaced by the week number of the year (Monday as the first day of the week) as a decimal number (01-53). If the week containing January 1 has four or more days in the new year, then it is week 1; otherwise it is the last week of the previous year, and the next week is week 1.	'39'
%v	Equivalent to %e-%b-%Y.	'25-Sep-2008'
%W	Replaced by the week number of the year (Monday as the first day of the week) as a decimal number (00-53).	'38'
%w	Replaced by the weekday (Sunday as the first day of the week) as a decimal number (0-6).	'4'
%X	Replaced by national representation of the time.	'22:32:00'
%x	Replaced by national representation of the date.	'09/25/08'
%Y	Replaced by the year with century as a decimal number.	'2008'
%y	Replaced by the year without century as a decimal number.	'08'
%Z	Replaced by the time zone name.	'EDT'
%z	Replaced by the time zone offset from UTC; a leading plus sign stands for east of UTC, a minus sign or west of UTC, hours and minutes follow with two digits each and no delimiter between them (common form for RFC 822 date headers).	'-0500'
%+	Replaced by national representation of the date and time (the format is similar to that produced by date(1)).	'Thu Sep 25 22:32:00 EDT'
%l	Milliseconds, rounded to three places, with leading	'000'

	zeros. Note: this is a departure from the strftime format, which does not represent milliseconds.	
%%	Replaced by '%'	%

* %G, %g, and %V yield values [are] calculated from the week-based year defined by the ISO 8601 standard. In this system, weeks start on a Monday, and are numbered from 01, for the first week, up to 52 or 53, for the last week. Week 1 is the first week where four or more days fall within the new year. Or, synonymously, week 01 is: the first week of the year that contains a Thursday; or, the week that has 4 January in it. When three or fewer days of the first calendar week of the new year fall within that year, then the ISO 8601 week-based system counts those days as part of week 53 of the preceding year. For example, 1 January 2010 is a Friday, meaning that just three days of that calendar week fall in 2010. Thus, the ISO 8601 week-based system considers these days to be part of week 53 (%V) of the year 2009 (%G) ; week 01 of ISO 8601 year 2010 starts on Thursday, 4 January 2010.

```
%{The time is: %X.%1%nThe date is: %x}t
```

This input yields the following output: The time is: 22:32:00.000 The date is: 09/25/08

Parent topic: [Logs](#)

Related concepts:
[Custom logs overview](#)

Related tasks:
[Configuring custom logs](#)

Related reference:
[Strftime format conversions](#)

DeploymentTargetStatsHistoricCache

This log file contains historic information on the deployment target statistics historic cache.

Location

This file is located in the *log_root/visualization* directory by default. You can configure the location of the visualization data service log files.

Usage notes

- **deploymentTargetName**: Deployment target name.
- **nodeName**: Node name.
- **deploymentTargetType**: Deployment target type.
- **speedReq**: MCycles per second.
- **highMemMark**: The highest amount of memory in MBytes consumed over the life span of the deployment target.

Parent topic: [Logs](#)

Parent topic: [Logs](#)

Related tasks:

[Configuring the visualization data service](#)

NodeStatsHistoricCache

This log file contains historic information on the node statistics cache.

Location

This file is in the `log_root/visualization` directory by default. You can configure the location of the visualization data service log files.

Usage notes

- **nodeName**: Node name.
- **nodeCPU**: Percentage of processor utilization.
- **nodeFreeMemory**: Free memory in kilobytes.
- **usedMemory**: Memory being used in kilobytes.
- **version**: Node version.
- **nodeSpeed**: Node speed in normalized MCycles/sec.
- **background**: Background processor use in normalized MCycles/sec.
- **wasBackground**: WebSphere® Application Server background processor use in normalized MCycles/sec.
- **isDBNode**: True, if this is a database node.
- **entitledCapacity**: The percentage of entitled capacity that is being used by a server partition. This metric is only available for AIX® shared-uncapped micro-partitions. If you are not using this configuration, the value displays as 0.
- **maxEntitledCapacity**: The maximum of entitled capacity that can be used by a server partition. This metric is only available for AIX shared-uncapped micro-partitions. If you are not using this configuration, the value displays as 0.
- **physicalCPUsConsumed**: The number of physical processors that are being used by the shared server partition. This metric is only available for AIX shared-uncapped micro-partitions. If you are not using this configuration, the value displays as 0.
- **availablePoolCPUs**: The number of processors available in the shared processor pool to which this server partition belongs. This metric is only available for AIX shared-uncapped micro-partitions. If you are not using this configuration, the value displays as 0.

Parent topic: [Logs](#)

Parent topic: [Logs](#)

Related tasks:

[Configuring the visualization data service](#)

ServerStatsCache

This log file describes the server statistics cache.

Location

This file is located in the *log_root/visualization* directory by default. You can configure the location of the visualization data service log files.

Usage notes

- **name:** Specifies the server name.
- **node:** Specifies the node name.
- **version:** Specifies the version of WebSphere® Virtual Enterprise that is running on the node.
- **weight:** Specifies the dynamic workload management (dWLM) weight of the server.
- **cpu:** Specifies the percentage of the processor that is being used.
- **usedMemory:** Specifies the number of kilobytes of memory that is used in the Java™ virtual machine (JVM) run time. This statistic is available only for server instances containing WebSphere Application Server.
- **uptime:** Specifies the number of milliseconds that JVM has been running. This statistic is available only for server instances containing WebSphere Application Server.
- **totalRequests:** Specifies the number of total requests for the server. This statistic is available only for server instances containing WebSphere Application Server. If your environment contains other servers, including external WebSphere application servers, you can manually calculate a value that is similar to the value of the totalRequests metric. Locate all entries for a particular server in the TCModuleInstanceStatsCache.log file. Using the beginning and end time stamps, calculate the sum of the serviced metric for the server. For more information, see [TCModuleInstanceStatsCache](#).
- **updateTime:** Specifies the time that the statistics were provided.
- **db_averageResponseTime:** Specifies the average response time for the database server.
- **db_throughput:** Specifies the throughput for the database server.
- **highMemMark:** The highest amount of memory in megabytes consumed over the life span of the server.
- **residentMemory:** The amount of physical memory not shared with other processes that the JVM is using in kilobytes.
- **totalMemory:** Specifies the total amount of memory that the JVM is using in kilobytes.
- **totalMethodCalls:** Specifies the total number of enterprise Java bean (EJB) module method calls.

Parent topic: [Logs](#)

Parent topic: [Logs](#)

Related tasks:

[Configuring the visualization data service](#)

TCModuleStatsCache

This log file contains information about the transaction class module cache.

Location

This file is located in the *log_root/visualization* directory by default. You can configure the location of the visualization data service log files.

Log information

- **timeStamp**: Time in milliseconds.
- **tcmodname**: Specifies the transaction class module name.
- **dtname**: Specifies the deployment target name.
- **gwid**: Specifies the Gateway ID.
- **j2eemodname**: Specifies the Java™ 2 Platform, Enterprise Edition (J2EE) module name.
- **appname**: Specifies the application name.
- **tcname**: Specifies the transaction class name.
- **scname**: Specifies the service policy name.
- **nodegroup**: Specifies the node group name.
- **cell**: Specifies the cell name.
- **proxy**: Specifies the proxy or on demand router name.
- **proxycell**: Specifies the cell where the autonomic request flow manager (ARFM) gateway that submitted these statistics resides.
- **proxynode**: Specifies the node on which the ARFM gateway that submitted these statistics resides.
- **proxyserver**: Specifies the server on which the ARFM gateway that submitted these statistics resides.
- **protocol**: Specifies the protocol family of the requests for these statistics: Hypertext Transfer Protocol (HTTP), SIP, Internet Inter-ORB Protocol (IIOP), or Java Message Service (JMS).
- **arrivals**: Specifies the number of requests that arrived during the reported interval.
- **executingInt**: Specifies the execution concurrency integral. This field is the sum, over each millisecond in the reported interval, of the number of requests that were running at the start of the millisecond.
- **lengthInt**: Specifies the queue length integral. This field is the sum, over each millisecond in the reported interval, of the number of requests in the queue at the start of the millisecond.
- **currentLen**: Specifies the number of requests in the queue at the end of the reported interval.
- **departs**: Specifies the number of messages that were dispatched to server during the reported interval and initiated a logical dialog. This field counts only the first reception of a certain message.
- **dropped**: Specifies the number of requests that were dropped because of a queue overflow.
- **waittm**: Specifies the sum in milliseconds, over all of the requests that are dispatched during the reported interval, of the time that each request spends waiting in the queue. The sum is specified in number of milliseconds. Divide the waittm value by the departs value to get the average amount of time that a request spends waiting in the queue.
- **resptm**: Specifies the sum in milliseconds of all of the response times of the requests that complete running during the reported interval. Divide the resptm value by the serviced value to get the average response time for the requests. The response time of a request is the sum of the queuing time of the request plus the service time of the request. The response time, however, does not include any time before the request reached the ARFM gateway queue. To calculate the sum, add the waittm value, which excludes time before enqueueing, and the servicetm value.
- **servicetm**: Specifies the sum of all the service times of the requests that completed running during the reported interval. The sum is in units of milliseconds. Divide the servicetm value by the serviced value to get the average service time for a request during the reported interval.
- **serviced**: Specifies the number of requests that completed during the reported interval.
- **begintm**: Specifies the start time for the statistics interval.
- **endtm**: Specifies the end time for the statistics interval.
- **qlen**: Specifies the sum, over the requests that arrived during the reported interval, of the queue length, before insertion, as seen at each request arrival.
- **abvgoal**: Specifies the number of requests that both returned during the reported interval had a response time above their service class threshold.
- **workFactors**: Specifies the average amount of work that is performed on a deployment target to serve an HTTP request, a Session Initiation Protocol (SIP) message, an Internet Inter-ORB Protocol (IIOP) call, or a Java Message Service (JMS) message. The work factor is represented in standard processor megacycles. You can override the work factor values by specifying the workFactorOverrideSpec custom property on the deployment target. For more information about the workFactorOverrideSpec custom property, see [Autonomic request flow manager advanced custom properties](#).
- **dlgRtmRefusals**: Specifies the number of dialog retransmissions that are refused admission to a queue.
- **dlgRtmAdmissions**: Specifies the number of dialog retransmissions that are admitted to a queue.
- **dlg1stRefusals**: Specifies the number of dialog initiating messages that are not retransmissions, which are refused admission to a queue.
- **dlg1stAdmissions**: Specifies the number of dialog initiating messages that are not retransmissions, which are admitted to a queue.

Parent topic: [Logs](#)

Parent topic: [Logs](#)

Related tasks:

[Configuring the autonomic request flow manager](#)

[Configuring the visualization data service](#)

Related information:

[Autonomic request flow manager custom properties](#)

TCModuleInstanceStatsCache

This log file describes the transaction class module instance cache.

Location

This file is located in the *log_root/visualization* directory by default. You can configure the location of the visualization data service log files.

Usage notes

- **tcmodname**: Transaction class module instance name.
- **gwid**: Gateway ID.
- **j2eemodname**: Java™ EE module name.
- **dtname**: Deployment target name.
- **scname**: Service policy name.
- **appname**: Application name.
- **tcname**: Transaction class name.
- **server**: Server name.
- **node**: Node name.
- **nodegroup**: Node group name cell: cell name.
- **proxy**: Name of the proxy or on demand router (ODR) hosting the ARFM gateway that submitted the statistics. Applicable if the protocol is HTTP or SIP.
- **proxycell**: The cell in which the proxy or ODR resides.
- **proxynode**: The node in which the proxy or ODR resides.
- **proxyserver**: Name of the server hosting the ARFM gateway that submits the statistics. Applicable if the protocol is JMS or IIOP.
- **protocol**: The protocol family of the requests that these statistics are for; that is, HTTP, IIOP, JMS, SIP.
- **arrivals**: No meaningful value.
- **respOutTime**: The sum in milliseconds, over the requests that completed during the reported interval due to service timeout, of the request's response time. Divide by serviced to compute average response time. Calculate the sum of the service time and queue time. The metric, however, does not include any time spent before the request reaches the ARFM gateway queues.
- **serviceOutTime**: The sum in milliseconds, over the requests that completed during the reported interval due to service timeout, of the request's service time. Divide by serviced to compute average service time.
- **numTimedOut**: The number of requests that completed, during the reported interval, due to service timeout.
- **numErrored**: The number of requests that completed, during the reported interval, with an error indicator.
- **firstErrorTime**: The time of the first error response, if any, otherwise 0.
- **lastErrorTime**: The time of the last error response, if any, otherwise 0.
- **lastTimeoutTime**: The time of the last timeout response, if any, otherwise 0.
- **currentLen**: No meaningful value.
- **lengthInt**: No meaningful value.
- **executingInt**: Integral over each millisecond in the reported interval, of the number of requests that were executing at the start of the interval.
- **execution**: Concurrency integral, in units of milliseconds * requests.
- **departs**: Number of requests dispatched to server.
- **dropped**: No meaningful value.
- **waittm**: Total wait time in queue of all requests in interval.
- **resptm**: Total response time for all requests in interval.
- **servicetm**: Total service time for all requests in interval serviced: number of requests serviced.
- **begintm**: Start time of interval.
- **endtm**: End time of interval.
- **qlen**: Total queue length over in the interval.
- **abvgoal**: Number of requests that both returned during the reported interval had a response time above their service class threshold.
- **workFactors**: The amount of work in MCycles/sec that each request is estimated to consume after the request is dispatched from the ARFM gateway.
- **dlgRtmRefusals**: Specifies the number of dialog retransmissions that are refused admission to a queue.
- **dlgRtmAdmissions**: Specifies the number of dialog retransmissions that are admitted to a queue.
- **dlg1stRefusals**: Specifies the number of dialog-initiating messages that are not retransmissions, which are refused admission to a queue.
- **dlg1stAdmissions**: Specifies the number of dialog-initiating messages that are not retransmissions, which are admitted to a queue.

Parent topic: [Logs](#)

Parent topic: [Logs](#)

Related tasks:

[Configuring the visualization data service](#)

TierStatsCache

This log file describes the tier statistics cache.

Location

This file is in the `log_root/visualization` directory.

Usage notes

- **proxy**: Proxy name.
- **dtName**: Server name.
- **svcclsName**: Service class name.
- **power**: Power® coefficient.

Parent topic: [Logs](#)

Parent topic: [Logs](#)

Related tasks:

[Configuring the visualization data service](#)

FineGrainedPowerConsumptionStatsCache

This log file contains fine grained power and work consumption data. A record is written for every transaction class module and server instance. This action creates a record for every middleware application, module, transaction class, and server instance that has had work routed through an on demand router (ODR). Additional fields exist that hold relationship information such as the cluster to which the server belongs, the node group with which the cluster is associated, and the service policy with which the transaction class is associated.

Location

This file is in the *log_root/visualization* directory.

Usage notes

- **timeStamp**: Specifies the time in milliseconds since January 1, 1970, 00:00:00 GMT.
- **tcmodname**: Specifies the transaction class module name. This value is a concatenation of the transaction class, the application name, and the module name.
- **gwid**: Specifies the gateway ID, which is a concatenation of the cluster name, cell name, node name, and ODR name.
- **cell**: Specifies the cell name.
- **appname**: Specifies the application name.
- **modulename**: Specifies the module name.
- **servicepolicy**: Specifies the service policy name.
- **tcname**: Specifies the transaction class name.
- **server**: Specifies the server name.
- **node**: Specifies the node name.
- **odr**: Specifies the ODR name.
- **cluster**: Specifies the cluster name.
- **nodegroup**: Specifies the node group name.
- **begintm**: Specifies the time that the interval began.
- **endtm**: Specifies the end time of the interval.
- **workfactor**: Specifies the estimated work factor from the work profiler for the request type.
- **numserviced**: Specifies the number of requests serviced of this type.
- **workcompleted**: Specifies the amount of work completed in the interval, calculated as $\text{numserviced} * \text{workfactor}$.
- **powerconsumed**: Specifies the power consumption, calculated as $\text{numserviced} * \text{workfactor} / (\text{endtm} - \text{begintm})$.
- **nodepower**: Specifies the total power that is available for the node.
- **nodeworkpotential**: Specifies the total amount of work that the node could accommodate over the interval ($\text{totalnodepower} * (\text{endtime} - \text{begintime})$).
- **cellpower**: Specifies the total amount of power available for consumption for the cell. This value is the sum of the **nodepower** value over all nodes in the cell.
- **cellworkpotential**: Specifies the total amount of work that the cell could accommodate over the interval ($\text{totalcellpower} * (\text{endtime} - \text{begintime})$).

Parent topic: [Logs](#)

Parent topic: [Logs](#)

Related tasks:

[Configuring the visualization data service](#)

ServerPowerConsumptionStatsCache

This file is a consolidation of the FineGrainedPowerConsumptionStatsCache at the server level with some additional server data.

Location

This file is in the *log_root/visualization* directory.

Usage notes

- **timeStamp**: Time in milliseconds since January 1, 1970, 00:00:00 GMT.
- **cell**: Cell name.
- **name**: Server name.
- **node**: Node name.
- **cluster**: Cluster name.
- **nodegroup**: Node group name.
- **begintimestamp**: The begin time of the interval.
- **endtimestamp**: The end time of the interval.
- **cpu**: The average server percentage of processor utilization over the *cputimeinterval* value.
- **workcompleted**: The amount of work completed by the server in the interval $(cpu * nodepower) * (endtime - begintime \text{ in seconds}) / 100$.
- **powerconsumed**: The power consumption by the server $(cpu * nodepower) / 100$.
- **nodepower**: The total power available for the node.
- **nodeworkpotential**: The total amount of work that the node could accommodate over the interval $(totalnodepower * (endtime - begintime))$.
- **cellpower**: The total amount of power available for consumption for the cell; a sum of the node power over all nodes in the cell.
- **cellworkpotential**: The total amount of work that the cell could accommodate over the interval $(totalcellpower * (endtime - begintime))$.

Parent topic: [Logs](#)

Parent topic: [Logs](#)

Related tasks:

[Configuring the visualization data service](#)

MBean Type List

[ApplicationEditionManager](#)
[AsyncPmiBridgeRegularEndOverlay](#)
[BBSO](#)
[CRRemoteTestProxy](#)
[CellAgentMBean](#)
[CentralizedInstallMgr](#)
[ConfigRepositoryProxy](#)
[DWLM](#)
[DistributedTrace](#)
[DynamicClusterConfigManager](#)
[DynamicClusterConfigUtil](#)
[DynamicClusterRuntimeManager](#)
[ExtendedConfigRepository](#)
[FileServiceMBean](#)
[HAManagedItems](#)
[HAPuginCfgGenerator](#)
[HAWsmmController](#)
[HealthConfigMonitor](#)
[HmControllerMBean](#)
[NodeDetectConsumer](#)
[NodeGroupManager](#)
[NodeGroupMonitor](#)
[OdrDebugMBean](#)
[OdrMBean](#)
[PlacementControllerMBean](#)
[ProcessMuckerMBean](#)
[QuiesceManager](#)
[SCPMBean](#)
[SIPServerQuiesceMBean](#)
[SmpMBean](#)
[TargetTreeMbean](#)
[TaskConfigMonitor](#)
[TrMBean](#)
[VESessionManagerMBean](#)
[VMPubControllerMBean](#)
[WorkProfilerMBean](#)
[WsmmProxyMBean](#)
[XDDWLMClientMBean](#)
[XDTestConfigHelper](#)
[XdCommMBean](#)
[XdStorageMBean](#)

WebSphere Virtual Enterprise (WVE) Release 7.0


Public JMX MBean Interfaces

This documentation provides information on the public APIs presented by the MBean that are part of the management system of WebSphere Virtual Enterprise (WVE) Release 7.0. These MBean are accessible programmatically and through the wsadmin command-line scripting tool. Click on an MBean Type at the left for detailed information about that MBean.

Copyright IBM Corp. 1996-2011

Messages

When you encounter a message in a log or other parts of the product interface, look up the message by its component prefix to find out more information.

To open the information center table of contents to the location of this reference information, click the **Show in Table of Contents** button () on your information center border.

Parent topic: [Reference](#)

ARFM

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[ARFM0001E](#)

RQF_NoMapReqFilt=ARFM0001E: MapRequestFilter not configured (request context was {0}).

[ARFM0002E](#)

RQF_UnkLocalHost=ARFM0002E: java.net.InetAddress.getLocalHost().getHostName() threw UnknownHostException: {0}

[ARFM0003E](#)

RQF_Read0=ARFM0003E: Cannot happen.

[ARFM0004I](#)

RQF_Init=ARFM0004I: ARFM HTTP Filters initialized.

[ARFM0005E](#)

RQF_RQMConsFailed=ARFM0005E: The ARFM Filter suffered unexpected exception {0} constructing a ReqQMgr to handle request {1} for deployment target {2} in cell {3}.

[ARFM0006E](#)

RQF_RQMConsFailedBefore=ARFM0006E: The ARFM Filter previously suffered some unexpected exception while constructing a ReqQMgr to handle a previous request, and will now pass without delay request {0} for deployment target {1} in cell {2}.

[ARFM0007I](#)

SIP_RQF_Init=ARFM0007I: ARFM SIP Filters initialized.

[ARFM0008I](#)

RQF_OvIRC=ARFM0008I: Response code {0} will be used to reject HTTP messages due to overload.

[ARFM0009I](#)

SIP_RQF_OvIRCs=ARFM0009I: Response code {0} will be used to reject SIP/TCP messages, and code {1} will be used to reject SIP/UDP messages, due to overload.

[ARFM0021E](#)

RSP_NoId=ARFM0021E: No ARFM ID attached to response {0}.

[ARFM0022E](#)

RSP_NullRqm=ARFM0022E: Null ReqQMgr attached to response {0}.

[ARFM0041E](#)

WOM_BuildFailed=ARFM0041E: unexpected Throwable aborted compilation of classification policy; {0}

[ARFM0042E](#)

WOM_MapFailed=ARFM0042E: unexpected exception aborted request classification; uriNode={0}, context={1}, exn={2}

[ARFM0043E](#)

WOM_ServerLacksNode=ARFM0043E: The ODC tree has no Node for Server {2}, which is the deployment target for the Web Module {1}, which handles URIs of the form {0}.

[ARFM0044E](#)

WOM_WebModuleNotDeployed=ARFM0044E: The ODC tree has no deployment target for Web Module {1}, which handles URIs of the form {0}.

[ARFM0045W](#)

WOM_BadOS=ARFM0045W: ARFM is unsure whether to do overload protection and queuing for traffic destined to node {0} because its operating system ({1}) is unexpected.

[ARFM0046E](#)

WOM_ModuleLacksClusters=ARFM0046E: The on-demand configuration tree contains a module, named {0}, that has no associated clusters.

[ARFM0047E](#)

WOM_GroupLacksNodes=ARFM0047E: The on-demand configuration tree contains a cluster (named {0}) that is deployed to a node group (named {1}) that contains no nodes.

[ARFM0061W](#)

WSV_getFmsFailed=ARFM0061W: Unexpected exception arose when looking up the FilterManagerService: {0}

[ARFM0062I](#)

WSV_noFMS=ARFM0062I: ARFM Gateway disabled because this is not an ODR process.

ARFM0063I

WSV_disabled=ARFM0063I: ARFM filters are disabled by ARFM configuration.

ARFM0081E

WTR_FindTreeFailed=ARFM0081E: Failed to get ODC Tree: {0}

ARFM0082E

WTR_NoCell=ARFM0082E: Failed to find cell {0} in the ODC Tree.

ARFM0085E

WTR_SvrUnkClust=ARFM0085E: ODC Tree changed while locked for read: in cell {0}, server {1} has previously unknown cluster {2}.

ARFM0096E

WTR_ReadClusterCpuFailed=ARFM0096E: While reading the CPU utilizations for clusters {1} in cell {0}, the ARFM Controller suffered an unexpected exception: {2}.

ARFM0097E

WTR_NodeAlarmFailed=ARFM0097E: Backstop stat production process for node {1} in cell {0} terminated unexpectedly due to failure {2}.

ARFM0099E

WTR_BadOdcExn=ARFM0099E: Unexpected ODCEXception while examining server {0} on node {1} in cell {2}: {3}

ARFM0100W

WTR_StaticClusterInGroupedNode=ARFM0100W: Ignoring cluster {2} on node {1} while estimating computing power of cell {0}.

ARFM0101W

NUI_NoUtil=ARFM0101W: Node {0} did not report CPU utilization within the last {1} seconds.

ARFM0121E

RqSt_LBExn=ARFM0121E: Exception aborted load balancing of request context {0}, with attributes {1}: {2}.

ARFM0122E

RqSt_ResumeFailed=ARFM0122E: Resume failed for context {0}, with attributes {1}: {2}.

ARFM0123E

RqSt_NoODC=ARFM0123E: ARFM failed to get an ODCHelper: {2}.

ARFM0141E

NSW_ListNodesFailed=ARFM0141E: Unexpected ODC Exception: {0}.

ARFM0142E

NSW_ConsFailed=ARFM0142E: Backstop NodeQStats production will not happen because of unexpected exception: {0}.

ARFM0161E

WTR_ExcessBkgnd=ARFM0161E: For the interval ending {2}, node {1} in cell {0} had a background CPU load of {3} percent, which exceeds the configured limit of {4}.

ARFM0162E

WTR_MultiBalancerState=ARFM0162E: Found multiple Balancer states for node {1} in cell {0}, one from {2} and another from {3}; using the first ({4}).

ARFM0163E

WTR_NoBBF=ARFM0163E: Unable to obtain BulletinBoardFactory at startup.

ARFM0164E

WTR_FindBbFailed=ARFM0164E: Attempt to find BulletinBoard named {0} failed: {1}.

ARFM0165E

WTR_FoundNoBb=ARFM0165E: Lookup of BulletinBoard named {0} returned null!

ARFM0166E

WTR_FindBbScopeFailed=ARFM0166E: Lookup of remote BulletinBoardScope for cell named {0} failed: {1}.

ARFM0167E

WTR_SubscribeFailed=ARFM0167E: An attempt to subscribe WebSphere Business Grid Balancer state for node {1} in cell {0} failed: {2}.

ARFM0168E

WTR_CloseSubscrFailed=ARFM0168E: An attempt to close subscription to WebSphere Business Grid Balancer state for node {1} in cell {0} failed: {2}.

ARFM0169E

WTR_NoNodeMon=ARFM0169E: Internal inconsistency, concerning node {1} of cell {0}.

ARFM0170W

WTR_NoBbScope=ARFM0170W: Unable to track WebSphere Business Grid Balancer state for node {1} in cell {0} due to some earlier failure.

ARFM0171E

WTR_InitFailed=ARFM0171E: Internal initialization failure: {0}.

ARFM0172E

WTR_SvrClusterFail=ARFM0172E: Internal failure regarding server {2} on node {1} in cell {0}: {3}.

ARFM0173W

WTR_NullCpuListnr=ARFM0173W: Internal failure (cell={0}, node={1}, toNotify={2}).

ARFM0201E

ENI_NoCell=ARFM0201E: Did not find the ODR cell (named {0}) in the ODC tree.

ARFM0202E

ENI_PropFail=ARFM0202E: Unexpected ODC Exception fetching external.nodes property from cell {0}: {1}.

ARFM0203E

ENI_SyntaxErr=ARFM0203E: Syntax error in external.placement custom property (value is "{1}") of the {0} cell: {2}.

ARFM0241E

RdCp_Failed=ARFM0241E: ARFM suffered unexpected problem reading the ODC tree for cell {0}: {1}

ARFM0261E

BPFA_SvcViol=ARFM0261E: Service policy {1} in cell {0} has suffered a serious violation since {2}.

ARFM0262E

BPFA_TaskSubmitExn=ARFM0262E: The ARFM Controller for cell {0} suffered an unexpected exception when submitting its current list of service policy violation tasks to the task manager.

ARFM0263W

BPFA_BigXss=ARFM0263W: The autonomic request flow management statistics for messages admissible to server {0} on node {1} in cell {2} for the sake of memory overload protection ({3} tests, {4} passes) were reduced by a factor of {5}.

ARFM0281E

MCT_UnkODR=ARFM0281E: An unexpected exception arose when trying to read the On-Demand Configuration property named ODR for the server {2} on the node {1} in the cell {0}: {3}.

ARFM0282E

MCT_TwoCenters=ARFM0282E: There are on-demand routers in two connected cells: one router is server {2} on node {1} in cell {0} and another is server {5} on node {4} in cell {3}.

ARFM0501E

TT1_err=ARFM0501E: Error with one parm, {0}

ARFM0998W

WFLT_GENLWRN=ARFM0998W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

ARFM0999E

WFLT_GENLERR=ARFM0999E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

ARFM1001E

RQM_UnkLocalHost=ARFM1001E: java.net.InetAddress.getLocalHost().getHostName() threw UnknownHostException: {0}

ARFM1002E

RQM_CfgloExn=ARFM1002E: IO problem reading ARFM Gateway config file {0}: {1}

ARFM1003E

RQM_CfgParseExn=ARFM1003E: Problem parsing ARFM Gateway config file {0}: {1}

ARFM1004E

RQM_WeightForUnkQueue=ARFM1004E: The ARFM Gateway {0} received a dequeuing weight for an unknown service class ({1}).

ARFM1005E

RQM_LogStatsCreateFailed=ARFM1005E: An ARFM Gateway failed to create a statistics log file named {0}: {1}

ARFM1006E

RQM_StatsReporterDied=ARFM1006E: The statistics generating process in ARFM Gateway {0} terminated due to failure {1}.

ARFM1007E

RQM_NqsWrongCell=ARFM1007E: The ARFM Gateway with ID {1} in cell {0} was asked to produce NodeQStats for node {3} in cell {2}.

ARFM1008E

RQM_BadGoalKind=ARFM1008E: RQM given unexpected goal kind {0} from ClassLoader {1}.

ARFM1009I

RQM_GwUp=ARFM1009I: The ARFM Gateway for protocol family {0} and target {1} has been initialized.

ARFM1081E

MgP_StartupDied=ARFM1081E: JMS MessagePuller {0} died at startup due to exception {1}

ARFM1082E

MgP_ClientFailed=ARFM1082E: JMS MessagePuller {0} caught exception from client: {1}

ARFM1083E

MgP_RcvFailed=ARFM1083E: JMS MessagePuller {0} caught exception while receiving a message: {1}

ARFM1101E

Ctl_BadCapMsgA=ARFM1101E: The ARFM Controller {0} received a malformed capacity setting {1}.

ARFM1102W

Ctl_DidntUseAll=ARFM1102W: The ARFM Controller {0} chose to use only {1} seats when {2} were available.

ARFM1103E

Ctl_OptFailed=ARFM1103E: The ARFM Controller {0} suffered an unexpected exception while solving the optimization (which was for {1} variables): {2}

ARFM1104E

Ctl_OptFailed1=ARFM1104E: Model {0} was {1}

ARFM1105E

Ctl_OptFailed2=ARFM1105E: Allocation {0} was {1}

ARFM1106E

Ctl_CreateLogFailed=ARFM1106E: ARFM Controller {0} tried to create statistics log file named {1} but failed: {2}

ARFM1107I

Ctl_Initted=ARFM1107I: The ARFM Controller for {0} has been activated on this machine; control cycle length is {1} milliseconds.

ARFM1108I

Ctl_Stopped=ARFM1108I: The ARFM Controller for {0} has been shutdown on this machine.

ARFM1109W

Ctl_NoRhos=ARFM1109W: The ARFM Controller for {0} suffered an unexpected internal failure (to produce CPU utilizations per cluster).

ARFM1110W

Ctl_BadRho=ARFM1110W: The ARFM Controller for {0} suffered an unexpected internal failure: the CPU utilization fraction for cluster {1} was found to be {2}.

ARFM1111W

Ctl_NoCap=ARFM1111W: The ARFM Controller skipped an optimization cycle due to complete lack of capacity information.

ARFM1112W

Ctl_CANotImplemented=ARFM1112W: The ARFM Controller was configured to use the obsolete functionality known as the Concurrency Adjuster --- ignored.

ARFM1113W

Ctl_CBNotImplemented=ARFM1113W: The ARFM Controller was configured to use the obsolete functionality known as the Concurrency Bounder --- ignored.

ARFM1114W

Ctl_NoEntryKappa=ARFM1114W: The Work Profiler supplied no speed factor for service class {2} and the entry tier {1} in cell {0}; using 1.

ARFM1115W

Ctl_NoEntryAlpha=ARFM1115W: The Work Profiler supplied no work factor for service class {2} and the entry tier {1} in cell {0}; using 1.

ARFM1116W

Ctl_NoKappaForRollup=ARFM1116W: The Work Profiler supplied no speed factor for the requests of service class {1} coming through gateway {0} to deployment target {2}.

ARFM1117W

Ctl_NoAlphaForRollup=ARFM1117W: The Work Profiler supplied no work factor for the requests of service class {1} coming through gateway {0} to deployment target {2}.

ARFM1118E

Ctl_ManyClasses=ARFM1118E: Too many service policies ({2}) from ARFM gateway {1} in cell {0}

ARFM1119I

Ctl_QsDrp=ARFM1119I: From {0} to {1} the autonomic request flow management gateway for the {2} protocol family in the {5} server on the {4} node in the {3} cell for messages destined for the {7} cluster in the {6} cell and having service policy {8} refused admission to {9} messages; {10} follow-up messages were not admitted; {11} of the first received transmissions of non-follow-up messages were not enqueued: {12} due to the limit on queue length, {13} due to the limit on the total number of active messages, {14} due to the logical dialog admission rate limit, {15} due to memory overload protection, and {16} due to other routing-related reasons; {17} retransmitted non-follow-up messages were also not enqueued; {18} messages were refused after spending some time in a queue.

ARFM1201E

FMM_ReadCfgFailed=ARFM1201E: Attempt to read config from file {0} failed: {1}

ARFM1221E

GWD_MultiDT=ARFM1221E: ARFM Controller for {0} got data from gateway {1} that mentions more than one deployment target; {2} and {3} are two of them.

ARFM1241W

DPSe_OldQStats=ARFM1241W: Speed request smoother for gateway {0} rejected QStats ending {1} because that is older than the smoother's previous QStats (which ended at {2}).

ARFM1242W

DPSe_WrongQStats=ARFM1242W: The speed request smoother for ARFM gateway {0} saw a gap in the series of statistics messages: one message reported on a period of time ending at {2} and the next message reported on a period of time beginning at {1}.

ARFM1261W

DPWU_NoWorkFactors=ARFM1261W: ARFM is unable to estimate the computational demands of the requests that arrive through gateway {0} due to missing work factors.

ARFM1262W

DPWU_NoSpeedFactors=ARFM1262W: ARFM is unable to estimate the computational demands of the requests that arrive through gateway {0} due to missing speed factors.

ARFM1981W

AlSq_RejectAckUnk=ARFM1981W: {0} ignoring allocation ack from unknown server {1}.

ARFM1982W

AlSq_RejectAllocUnk=ARFM1982W: {0} ignoring allocation for unknown server {1}.

ARFM1983W

AlSq_OtherDealloc=ARFM1983W: {0} noted deallocation from some other source (numInst={1}, desAlloc[i]={2}).

ARFM2061E

AIRO_HandleFailed=ARFM2061E: AllocReceiver {0} for cell {1} suffered unexpected exception handling received message: {2}

ARFM2241E

ESSI_SendFailed=ARFM2241E: BB-based send of an ARFM EvaldStats message for cell + Node Group {0} failed: {1}

ARFM2261E

ESSO_StartupDied=ARFM2261E: The JMS-based ARFM EvaldStats message sender for cell + Node Group {0} failed to initialize: {1}

ARFM2262E

ESSO_SendFailed=ARFM2262E: JMS-based send of an ARFM EvaldStats message for cell + Node Group {0} failed: {1}

ARFM2321E

MDS_SendFailed=ARFM2321E: Send of an ARFM mode message failed: {1}

ARFM2341E

MDSO_StartupDied=ARFM2341E: A JMS-based ARFM mode message sender failed to initialize: {0}

ARFM2342E

MDSO_SendFailed=ARFM2342E: Send of an ARFM mode message failed: {0}

ARFM2381E

NSRO_HandleFailed=ARFM2381E: {0} suffered unexpected exception handling received message: {1}

ARFM2401E

NSS_SendFailed=ARFM2401E: Send of an ARFM NodeQStats message failed: {0}

ARFM2421E

NSSO_StartupDied=ARFM2421E: A JMS-based ARFM NodeQStats message sender failed to initialize: {0}

ARFM2422E

NSSO_SendFailed=ARFM2422E: JMS-based send of an ARFM NodeQStats message failed: {0}

ARFM2581E

StSO_StartupDied=ARFM2581E: The JMS-based ARFM QStats message sender for cell + NodeGroup {0}, ODR {1}, failed to initialize: {2}

ARFM2582E

StSO_SendFailed=ARFM2582E: JMS-based send of an ARFM QStats message for cell + NodeGroup {0}, ODR {1}, failed: {2}

ARFM2621W

WtRO_RejectUnkObj=ARFM2621W: {0}, in cell + Node Group {1}, received unexpected type of object: {2}

ARFM2622W

WtRO_RejectUnkMsg=ARFM2622W: {0}, in cell + Node Group {1}, received unexpected type of JMS message: {2}

ARFM2623E

WtRO_HandleFailed=ARFM2623E: {0}, in cell + Node Group {1}, suffered unexpected exception handling received message: {2}

ARFM2661E

WtSO_StartupDied=ARFM2661E: The JMS-based ARFM weights message sender for cell + nodegroup {0} failed to initialize: {1}

ARFM2662E

WtSO_SetArgsLengths=ARFM2662E: The ARFM weights message sender was given inconsistent args: w.length={0}, classNames.size={1}

ARFM2663E

WtSO_SendJmsFailed=ARFM2663E: JMS-based send of an ARFM weights message in cell + NodeGroup {0} to ARFM gateway {1} suffered a JMS failure: {2}

ARFM2664E

WtSO_SendFailed=ARFM2664E: JMS-based send of an ARFM weights message in cell + NodeGroup {0} to ARFM gateway {1} failed: {2}

ARFM2701E

ApRO_HandleFailed=ARFM2701E: {0} suffered unexpected exception handling received message: {1}

ARFM2721E

AIS_SendFailed=ARFM2721E: Send of an ARFM work estimates message failed: {0}

ARFM2741E

AISO_StartupDied=ARFM2741E: A JMS-based ARFM work factor estimates message sender failed to initialize: {0}

ARFM2742E

AISO_SendFailed=ARFM2742E: JMS-based send of an ARFM work factor estimates message failed: {0}

ARFM2801E

WRSI_SendFailed=ARFM2801E: JMS-based send of an ARFM results message for cell + Node Group {0} failed: {1}

ARFM2821E

WRSO_StartupDied=ARFM2821E: The JMS-based ARFM results message sender for cell + nodegroup {0} failed to initialize: {1}

ARFM2822E

WRSO_SendFailed=ARFM2822E: JMS-based send of an ARFM results message in cell + NodeGroup {0} to ARFM gateway {1} failed: {2}

ARFM2841I

TC_NormalToEmergencyThrottle=ARFM2841I: {0} NORMAL to EMERGENCY_THROTTLE transition

ARFM2842I

TC_EmergencyUnthrottleToEmergencyThrottle=ARFM2842I: {0} EMERGENCY_UNTHROTTLE to EMERGENCY_THROTTLE transition

ARFM2843I

TC_EmergencyThrottleToEmergencyUnthrottle=ARFM2843I: {0} EMERGENCY_THROTTLE to EMERGENCY_UNTHROTTLE transition

ARFM2844I

TC_EmergencyUnthrottleToNormal=ARFM2844I: {0} EMERGENCY_UNTHROTTLE to NORMAL transition

ARFM2845E

TC_ArfmSensorFired=ARFM2845E: {0} ARFM emergency sensor fired.

ARFM2846E

TC_AsyncPmiDelaySensorFired=ARFM2846E: {0} Async PMI emergency sensor fired for node {1}.

ARFM2847E

TC_AsyncPmiHighUtilSensorFired=ARFM2847E: {0} High CPU utilization sensor fired for node {1}.

ARFM2848I

TC_ArfmSensorUnfired=ARFM2848I: {0} ARFM emergency sensor unfired.

ARFM2849I

TC_AsyncPmiDelaySensorUnfired=ARFM2849I: {0} Async PMI emergency sensor unfired for node {1}.

ARFM2850I

TC_AsyncPmiHighUtilSensorUnfired=ARFM2850I: {0} High CPU utilization sensor unfired for node {1}.

ARFM2881E

SSS_SendFailed=ARFM2881E: Send of an ARFM ServerStats message failed: {0}

ARFM2901E

Que_InitrAmongFollows=ARFM2901E: The {0} service policy of the {1} cell applies to traffic that is subject to admission control for CPU overload protection and includes both follow-up and non-follow-up messages; an example of the former was seen as recently as {2} and a current example of the latter is {3}.

ARFM2902E

Que_FollowAmongNitr=ARFM2902E: The {0} service policy of the {1} cell applies to traffic that is subject to admission control for CPU overload protection and includes both follow-up and non-follow-up messages; an example of the latter was seen as recently as {2} and a current example of the former is {3}.

ARFM4001E

QTI_SetQTIListenerFail=ARFM4001E: An Error occurs when setting QTI listener; data is {0}, and the exception is {1}.

ARFM4002E

QTI_ODCInitFailed=ARFM4002E: ODC initialization failure: {0}.

ARFM4021E

PQ_DlgAuxFail=ARFM4021E: An Error occurs when getting or setting DialogAux; data is {0}, and the exception is {1}.

ARFM4061E

WDFL_SetDlgAuxFail=ARFM4061E: An Error occurs when setting DialogAux; data is {0}, and the exception is {1}.

ARFM4081E

SCP_GetQTIFail1=ARFM4081E: An unexpected runtime exception occurred while initializing the SIP statistics gatherer for autonomic request flow management: {0}.

ARFM4082I

SCP_Enabled=ARFM4082I: Initialized the SIP statistics gatherer for autonomic request flow management.

ARFM4083I

SCP_Disabled=ARFM4083I: The SIP statistics gatherer for autonomic request flow management is disabled.

ARFM4084I

SCP_MemEnabled=ARFM4084I: Autonomic memory overload protection is enabled.

ARFM4085I

SCP_MemDisabled=ARFM4085I: Autonomic memory overload protection is disabled.

ARFM4101E

MOC_ODCInitFailed=ARFM4101E: ODC initialization failure: {0}.

ARFM4102W

MOC_IBMGcUnknown=ARFM4102W: Fail to load the JVM arguments on IBM JVM. {0}.

ARFM4103E

MOC_ThreReportErr=ARFM4103E: Importance threshold report failed. Server: {0}. Exception: {1}

ARFM4104I

MOC_Parm_GcDt=ARFM4104I: Absent an override via the explicitGcPeriod cluster custom property, autonomic memory overload protection will wait {0} seconds between explicitly triggered garbage collections.

ARFM4105I

MOC_Parm_GcDtOff=ARFM4105I: Absent an override via the explicitGcPeriod cluster custom property, autonomic memory overload protection will not explicitly trigger garbage collections.

ARFM4106I

MOC_Parm_GcDtRel=ARFM4106I: Absent an override via the explicitGcPeriod cluster custom property, the time between explicit garbage collections for the purpose of memory overload protection will be relative to the configured time between successive settings of the memory overload protection control value in the On-Demand Configuration facility.

ARFM4107I

MOC_Parm_MemBasic=ARFM4107I: The maximum heap size is {0} bytes; the low threshold for memory overload protection is {1} bytes and the high threshold is {2} bytes.

ARFM4108I

MOC_Parm_Nurs=ARFM4108I: Absent an override via the maxNonFullGcMemoryRelease cluster custom property, autonomic memory overload protection will figure a full garbage collection has happened when more than {0} kilobytes have been freed.

ARFM4109I

MOC_Parm_NursAlways=ARFM4109I: Absent an override via the maxNonFullGcMemoryRelease cluster custom property, autonomic memory overload protection will assume every garbage collection is a full one.

ARFM4110I

MOC_Parm_NursNever=ARFM4110I: Absent an override via the maxNonFullGcMemoryRelease cluster custom property, autonomic memory overload protection will assume no garbage collection is a full one.

ARFM4111I

MOC_UnklbmNurs=ARFM4111I: Generational garbage collection is in use in an IBM JRE, but the nursery size limit is not known exactly; assuming {0} kilobytes.

ARFM4112I

MOC_UnkVendor=ARFM4112I: Generational garbage collection is in use in a non-IBM JRE, but the nursery size limit is not known.

ARFM4113E

MOC_ODCErr=ARFM4113E: The memory overload protection controller suffered an unexpected error when looking up the local process in the On-Demand Configuration facility: {0}

ARFM4114I

MOC_GCTrig=ARFM4114I: The memory overload protection controller explicitly invoked a garbage collection (GC) at {0}; the previous detected major GC was at {1}, and the configured minimum time between explicit GCs was {2} seconds; the last time a heap usage sample was below {3} was {4}; the number of HTTP admissibility tests was {5} and {6} passed, and the choke fraction is {7}; the number of SIP admissibility tests was {8} and {9} passed, and the choke fraction is {10}; it is {11} that the garbage collector values throughput over pause times.

ARFM4115I

MOC_GCTrig2=ARFM4115I: The memory overload protection controller explicitly invoked a garbage collection (GC) at {0}; the previous detected major GC was at {1}, at which time the heap utilization became {2} percent, which is above the threshold of {3} percent; the configured minimum time between explicit GCs was {4} seconds; the number of HTTP admissibility tests was {5} and {6} passed, and the choke fraction is {7}; the number of SIP admissibility tests was {8} and {9} passed, and the choke fraction is {10}; it is {11} that the garbage collector values throughput over pause times.

ARFM4898W

WSCP_GENLWRN=ARFM4898W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

ARFM4899E

WSCP_GENLERR=ARFM4899E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

ARFM4998W

WCORE_GENLWRN=ARFM4998W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

ARFM4999E

WCORE_GENLERR=ARFM4999E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

ARFM5001E

WCF_FileErr=ARFM5001E: ARFM suffered unexpected exception parsing internal config from properties file {0}: {1}.

ARFM5002E

WCF_FindTreeFailed=ARFM5002E: Failed to get ODC Tree: {0}

ARFM5004E

WCF_NoCell=ARFM5004E: ARFM found no ODC vertex for the cell named {0}.

ARFM5006E

WCF_LoadFailed=ARFM5006E: ARFM for cell {0} suffered unexpected exception parsing config (some from ODC, some from file {1}): {2}.

ARFM5007I

WCF_Loaded=ARFM5007I: ARFM configuration for cell {0} is {1}.

ARFM5021E

TCT_OdcInitFailed=ARFM5021E: ARFM suffered unexpected failure during ODC setup: {0}

ARFM5022E

TCT_ReadFailed=ARFM5022E: ARFM suffered unexpected failure while looking up and reading configuration from ODC vertex for cell {0}: {1}.

ARFM5023E

TCT_BadEventType=ARFM5023E: ARFM got unexpected ODC event type.

ARFM5024E

TCT_EventErr=ARFM5024E: ARFM suffered unexpected exception while handling ODC event: {0}.

ARFM5041E

KOO_ReadFailed=ARFM5041E: ARFM suffered unexpected failure while looking up and reading configuration from ODC vertex for the cell {0}: {1}.

ARFM5998W

WMAN_GENLWRN=ARFM5998W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

ARFM5999E

WMAN_GENLERR=ARFM5999E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

ARFM6001E

OST_ConfigIoExn=ARFM6001E: Got I/O exception {2} while reading WSMM configuration from {1} for {0}; using defaults.

ARFM6002E

OST_Broken=ARFM6002E: {0} is broken by exception while parsing config info from {1}: {2}.

ARFM8001E

GRM_NoConfigFile=ARFM8001E: ARFM Controller configuration file {0} cannot be read.

ARFM8002E

GRM_Read0=ARFM8002E: Cannot happen.

ARFM8003W

GRM_GetFmsFailed=ARFM8003W: Unexpected exception arose when looking up the FilterManagerService: {0}

ARFM8004I

GRM_NoFMS=ARFM8004I: ARFM Controller disabled because this is not an ODR process.

ARFM8005I

GRM_NotStandAlone=ARFM8005I: ARFM Controller disabled because it is configured to run in the HTTP Proxy Filter.

ARFM8021E

HAC_FindTreeFailed=ARFM8021E: Failed to get ODC Tree: {0}

ARFM8022E

HAC_ReadTreeFailed=ARFM8022E: Failed to read ODC Tree: {0}

ARFM8023I

HAC_HamStarting=ARFM8023I: Starting ARFM controller for cell {0}, due to activation by HA-Manager.

ARFM8024I

HAC_HamStopping=ARFM8024I: Stopping ARFM controller for cell {0}, due to de-activation by HA-Manager.

ARFM8025W

HAC_UnknownEdge=ARFM8025W: Unknown edge change event {0}.

ARFM8026W

HAC_UnknownChange=ARFM8026W: Unknown topology change event {0}.

ARFM8027E

HAC_ControllerFailed=ARFM8027E: Attempt to instantiate ARFM Controller for node group {0} in cell {1} suffered unexpected failure: {2}.

ARFM8028I

HAC_OdcStarting=ARFM8028I: Starting ARFM controller for cell {0}, due to On-Demand Configuration (ODC) change.

ARFM8029I

HAC_OdcStopping=ARFM8029I: Stopping ARFM controller for cell {0}, due to On-Demand Configuration

(ODC) change.

[ARFM8030E](#)

HAC_GrpAbleFail=ARFM8030E: Unexpected exception arose while enabling or disabling ARFM controllers in the {0} cell (note: the central cell is {1} and the other connected cells are {2}): {3}.

[ARFM8031I](#)

HAC_ResetStarting=ARFM8031I: Starting ARFM controller for cell {0}, to implement a reset.

[ARFM8032I](#)

HAC_ResetStopping=ARFM8032I: Stopping ARFM controller for cell {0}, to implement a reset.

[ARFM8101E](#)

WCSP_LocErr=ARFM8101E: The ARFM for cell {0} suffered an unexpected exception/error when trying to find the location of the MBean holding its controllers; the exception/error is {1}.

[ARFM8102E](#)

WCSP_MBCallFail=ARFM8102E: The component status provider for ARFM suffered an unexpected exception when trying to query the ARFM controllers MBean: {0}.

[ARFM8501E](#)

TT1_err=ARFM8501E: Error with one parm, {0}

[ARFM8998W](#)

WCTL_GENLWRN=ARFM8998W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

[ARFM8999E](#)

WCTL_GENLERR=ARFM8999E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

ARFM0001E

RQF_NoMapReqFilt=ARFM0001E: MapRequestFilter not configured (request context was {0}).

Explanation

Internal configuration error: the Filter chain of the HTTP Proxy does not include the MapRequest Filter from wlm.client.

User response

Have IBM Services correct the error.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0002E

RQF_UnkLocalHost=ARFM0002E: java.net.InetAddress.getLocalHost().getHostName() threw
UnknownHostException: {}

Explanation

Code is unable to determine a hostname for the machine.

User response

Make sure the machine has an IP address.

Parent topic: [ARFM](#)

ARFM0003E

RQF_Read0=ARFM0003E: Cannot happen.

Explanation

Internal error.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0004I

RQF_Init=ARFM0004I: ARFM HTTP Filters initialized.

Explanation

The HTTP Proxy Filters that do Autonomic Request Flow Management have been initialized.

User response

Informational message only, no action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0005E

RQF_RQMConsFailed=ARFM0005E: The ARFM Filter suffered unexpected exception {0} constructing a ReqQMgr to handle request {1} for deployment target {2} in cell {3}.

Explanation

Something unexpected went wrong.

User response

Debug it.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0006E

RQF_RQMConsFailedBefore=ARFM0006E: The ARFM Filter previously suffered some unexpected exception while constructing a ReqQMgr to handle a previous request, and will now pass without delay request {0} for deployment target {1} in cell {2}.

Explanation

Something unexpected went wrong earlier.

User response

Find and debug it.

Parent topic: [ARFM](#)

ARFM0007I

SIP_RQF_Init=ARFM0007I: ARFM SIP Filters initialized.

Explanation

The SIP Proxy Filters that do Autonomic Request Flow Management have been initialized.

User response

Informational message only, no action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0008I

RQF_OvIRC=ARFM0008I: Response code {0} will be used to reject HTTP messages due to overload.

Explanation

The indicated response code is used when an HTTP message is rejected due to overload protection.

User response

Informational message only; a different code can be configured by setting the http.overload.error custom property of the On-Demand Router.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0009I

SIP_RQF_OvIRCs=ARFM0009I: Response code {0} will be used to reject SIP/TCP messages, and code {1} will be used to reject SIP/UDP messages, due to overload.

Explanation

The indicated response code is used when a SIP message is rejected due to overload protection.

User response

Informational message only; different codes can be configured by setting the sip.overload.error and sipu.overload.error custom properties of the On-Demand Router.

Parent topic: [ARFM](#)

ARFM0021E

RSP_NoId=ARFM0021E: No ARFM ID attached to response {0}.

Explanation

Internal error.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0022E

RSP_NullRqm=ARFM0022E: Null ReqQMgr attached to response {0}.

Explanation

This is usually a consequence of a failure when construction of the ReqQMgr was attempted.

User response

Find and debug the earlier failure.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0041E

WOM_BuildFailed=ARFM0041E: unexpected Throwable aborted compilation of classification policy; {0}

Explanation

Internal error.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0042E

WOM_MapFailed=ARFM0042E: unexpected exception aborted request classification; uriNode={0}, context={1}, exn={2}

Explanation

Internal error.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0043E

WOM_ServerLacksNode=ARFM0043E: The ODC tree has no Node for Server {2}, which is the deployment target for the Web Module {1}, which handles URIs of the form {0}.

Explanation

Internal error.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

ARFM0044E

WOM_WebModuleNotDeployed=ARFM0044E: The ODC tree has no deployment target for Web Module {1}, which handles URIs of the form {0}.

Explanation

Internal error.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

ARFM0045W

WOM_BadOS=ARFM0045W: ARFM is unsure whether to do overload protection and queuing for traffic destined to node {0} because its operating system ({1}) is unexpected.

Explanation

ARFM was built with a table that tells whether to apply overload protection and queuing for each of several target operating systems, but a request arrived for a target not running an operating system in that table; ARFM will apply overload protection and queuing.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0046E

WOM_ModuleLacksClusters=ARFM0046E: The on-demand configuration tree contains a module, named {0}, that has no associated clusters.

Explanation

This should never happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0047E

WOM_GroupLacksNodes=ARFM0047E: The on-demand configuration tree contains a cluster (named {0}) that is deployed to a node group (named {1}) that contains no nodes.

Explanation

Either something internal has failed or the cluster or node group is misconfigured.

User response

If something internal has failed, report this to IBM; if something is misconfigured, fix it.

Parent topic: [ARFM](#)

ARFM0061W

WSV_getFmsFailed=ARFM0061W: Unexpected exception arose when looking up the FilterManagerService: {0}

Explanation

Internal error.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0062I

WSV_noFMS=ARFM0062I: ARFM Gateway disabled because this is not an ODR process.

Explanation

ARFM Gateways run only in On-Demand Router processes.

User response

None. Informational entry.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0063I

WSV_disabled=ARFM0063I: ARFM filters are disabled by ARFM configuration.

Explanation

The EnableFilters entry in arfm.cfg is false, which prevents registration of the ARFM filters; such disablement is intended to be used only by developers in performance investigations.

User response

None. Informational entry.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0081E

WTR_FindTreeFailed=ARFM0081E: Failed to get ODC Tree: {0}

Explanation

Something went wrong in this ODR when ARFM tried to gain access to the ODC Tree.

User response

Check the propagation mechanism.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0082E

WTR_NoCell=ARFM0082E: Failed to find cell {0} in the ODC Tree.

Explanation

Internal inconsistency.

User response

Consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0085E

WTR_SvrUnkClust=ARFM0085E: ODC Tree changed while locked for read: in cell {0}, server {1} has previously unknown cluster {2}.

Explanation

This should not happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0096E

WTR_ReadClusterCpuFailed=ARFM0096E: While reading the CPU utilizations for clusters {1} in cell {0}, the ARFM Controller suffered an unexpected exception: {2}.

Explanation

This information is read from the ODC Tree and used by the ARFM Controller. A failure could be caused by an error in installation or configuration.

User response

Contact IBM Support.

Parent topic: [ARFM](#)

ARFM0097E

WTR_NodeAlarmFailed=ARFM0097E: Backstop stat production process for node {1} in cell {0} terminated unexpectedly due to failure {2}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0099E

WTR_BadOdcExn=ARFM0099E: Unexpected ODCEXception while examining server {0} on node {1} in cell {2}:
{3}

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0100W

WTR_StaticClusterInGroupedNode=ARFM0100W: Ignoring cluster {2} on node {1} while estimating computing power of cell {0}.

Explanation

This reports a condition that should not happen; autonomic request flow management will ignore the server for the named cluster on the named node.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0101W

NUI_NoUtil=ARFM0101W: Node {0} did not report CPU utilization within the last {1} seconds.

Explanation

The ARFM Controller did not receive a CPU utilization reading for the named node within the named amount of time; that data is normally delivered by Async PMI.

User response

Debug Async PMI config and operation.

Parent topic: [ARFM](#)

ARFM0121E

RqSt_LBExn=ARFM0121E: Exception aborted load balancing of request context {0}, with attributes {1}: {2}.

Explanation

Internal error.

User response

None (this error should be reported twice, hopefully the other report will be more helpful).

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0122E

RqSt_ResumeFailed=ARFM0122E: Resume failed for context {0}, with attributes {1}: {2}.

Explanation

Internal error.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0123E

RqSt_NoODC=ARFM0123E: ARFM failed to get an ODCHelper: {2}.

Explanation

Internal failure. Node and server level ARFM statistics will be unavailable. Charting, ARFM, and Placement functionality will be significantly degraded.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0141E

NSW_ListNodesFailed=ARFM0141E: Unexpected ODC Exception: {0}.

Explanation

Internal error.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0142E

NSW_ConsFailed=ARFM0142E: Backstop NodeQStats production will not happen because of unexpected exception: {0}.

Explanation

Internal error.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0161E

WTR_ExcessBkgnd=ARFM0161E: For the interval ending {2}, node {1} in cell {0} had a background CPU load of {3} percent, which exceeds the configured limit of {4}.

Explanation

The background load did not behave as expected, and thus the performance management will be impaired.

User response

Limit the background load to the configured value, or change the configured value.

Parent topic: [ARFM](#)

ARFM0162E

WTR_MultiBalancerState=ARFM0162E: Found multiple Balancer states for node {1} in cell {0}, one from {2} and another from {3}; using the first ({4}).

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0163E

WTR_NoBBF=ARFM0163E: Unable to obtain BulletinBoardFactory at startup.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0164E

WTR_FindBbFailed=ARFM0164E: Attempt to find BulletinBoard named {0} failed: {1}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0165E

WTR_FoundNoBb=ARFM0165E: Lookup of BulletinBoard named {0} returned null!

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0166E

WTR_FindBbScopeFailed=ARFM0166E: Lookup of remote BulletinBoardScope for cell named {0} failed: {1}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0167E

WTR_SubscribeFailed=ARFM0167E: An attempt to subscribe WebSphere Business Grid Balancer state for node {1} in cell {0} failed: {2}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0168E

WTR_CloseSubscrFailed=ARFM0168E: An attempt to close subscription to WebSphere Business Grid Balancer state for node {1} in cell {0} failed: {2}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0169E

WTR_NoNodeMon=ARFM0169E: Internal inconsistency, concerning node {1} of cell {0}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0170W

WTR_NoBbScope=ARFM0170W: Unable to track WebSphere Business Grid Balancer state for node {1} in cell {0} due to some earlier failure.

Explanation

This is a consequence of an earlier failure, probably one in the range ARFM0163--0166.

User response

Resolve the original problem.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0171E

WTR_InitFailed=ARFM0171E: Internal initialization failure: {0}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0172E

WTR_SvrClusterFail=ARFM0172E: Internal failure regarding server {2} on node {1} in cell {0}: {3}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0173W

WTR_NullCpuListnr=ARFM0173W: Internal failure (cell={0}, node={1}, toNotify={2}).

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0201E

ENI_NoCell=ARFM0201E: Did not find the ODR cell (named {0}) in the ODC tree.

Explanation

Internal error.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0202E

ENI_PropFail=ARFM0202E: Unexpected ODC Exception fetching external.nodes property from cell {0}: {1}.

Explanation

Internal error.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0203E

ENI_SyntaxErr=ARFM0203E: Syntax error in external.placement custom property (value is "{1}") of the {0} cell: {2}.

Explanation

There cell custom property named "external.placement" has a particular syntax, and the current value violates that syntax; while the value is malformed, ARFM ignores it.

User response

Delete the property or fix its value.

Parent topic: [ARFM](#)

ARFM0241E

RdCp_Failed=ARFM0241E: ARFM suffered unexpected problem reading the ODC tree for cell {0}: {1}

Explanation

Internal error.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0261E

BPFA_SvcViol=ARFM0261E: Service policy {1} in cell {0} has suffered a serious violation since {2}.

Explanation

The performance result for the named service policy has missed the target by at least the configured amount for at least the configured time.

User response

Diagnose and repair.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0262E

BPFA_TaskSubmitExn=ARFM0262E: The ARFM Controller for cell {0} suffered an unexpected exception when submitting its current list of service policy violation tasks to the task manager.

Explanation

This is an internal failure that should not happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM0263W

BPFA_BigXss=ARFM0263W: The autonomic request flow management statistics for messages admissible to server {0} on node {1} in cell {2} for the sake of memory overload protection ({3} tests, {4} passes) were reduced by a factor of {5}.

Explanation

The values reported by the workload management component were exceptionally large, and were reduced to avoid integer overflows in subsequent calculations; this will distort those calculations unless the corresponding statistics from all the other active ODRs were reduced by the same factor.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM0281E

MCT_UnkODR=ARFM0281E: An unexpected exception arose when trying to read the On-Demand Configuration property named ODR for the server {2} on the node {1} in the cell {0}: {3}.

Explanation

This is an internal failure that should not happen. For some purposes the server will be treated like it is not an on-demand router.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM0282E

MCT_TwoCenters=ARFM0282E: There are on-demand routers in two connected cells: one router is server {2} on node {1} in cell {0} and another is server {5} on node {4} in cell {3}.

Explanation

This is a configuration error; the only supported multi-cell topology is a star with on-demand routers in only the central cell. It is impossible to correctly instantiate ARFM controllers in the current configuration.

User response

Correct the configuration.

Parent topic: [ARFM](#)

ARFM0501E

TT1_err=ARFM0501E: Error with one parm, {0}

Explanation

exp

User response

ua

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0998W

WFLT_GENLWRN=ARFM0998W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something suspicious happened; this may or may not indicate an error.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM0999E

WFLT_GENLERR=ARFM0999E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something unexpected happened.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1001E

RQM_UnkLocalHost=ARFM1001E: java.net.InetAddress.getLocalHost().getHostName() threw
UnknownHostException: {0}

Explanation

Code is unable to determine a hostname for the machine.

User response

Make sure the machine has an IP address.

Parent topic: [ARFM](#)

ARFM1002E

RQM_CfgloExn=ARFM1002E: IO problem reading ARFM Gateway config file {0}: {1}

Explanation

This message should be gone by GA.

User response

Put a config file in the right place.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1003E

RQM_CfgParseExn=ARFM1003E: Problem parsing ARFM Gateway config file {0}: {1}

Explanation

This message should be gone by GA.

User response

Fix format error in config file.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1004E

RQM_WeightForUnkQueue=ARFM1004E: The ARFM Gateway {0} received a dequeuing weight for an unknown service class ({1}).

Explanation

Internal error - should never happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1005E

RQM_LogStatsCreateFailed=ARFM1005E: An ARFM Gateway failed to create a statistics log file named {0}: {1}

Explanation

There was some problem with the file IO; statistics will not be written to a log file.

User response

Check file name and file system state.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1006E

RQM_StatsReporterDied=ARFM1006E: The statistics generating process in ARFM Gateway {0} terminated due to failure {1}.

Explanation

That Gateway will generate no more statistics until the ODR containing it is started; ARFM functionality will be seriously degraded until then.

User response

Restart the ODR; report to IBM if the problem persists.

Parent topic: [ARFM](#)

ARFM1007E

RQM_NqsWrongCell=ARFM1007E: The ARFM Gateway with ID {1} in cell {0} was asked to produce NodeQStats for node {3} in cell {2}.

Explanation

Internal error - should never happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM1008E

RQM_BadGoalKind=ARFM1008E: RQM given unexpected goal kind {0} from ClassLoader {1}.

Explanation

Internal error - should never happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1009I

RQM_GwUp=ARFM1009I: The ARFM Gateway for protocol family {0} and target {1} has been initialized.

Explanation

This is a startup status report.

User response

No action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1081E

MgP_StartupDied=ARFM1081E: JMS MessagePuller {0} died at startup due to exception {1}

Explanation

The named ARFM component will not receive any internal messages; ARFM functionality will be severely degraded.

User response

Internal failure, consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1082E

MgP_ClientFailed=ARFM1082E: JMS MessagePuller {0} caught exception from client: {1}

Explanation

This is an unexpected internal failure.

User response

Consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1083E

MgP_RcvFailed=ARFM1083E: JMS MessagePuller {0} caught exception while receiving a message: {1}

Explanation

This is an unexpected internal failure.

User response

Consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1101E

Ctl_BadCapMsgA=ARFM1101E: The ARFM Controller {0} received a malformed capacity setting {1}.

Explanation

This is an internal inconsistency that should never arise.

User response

Internal error. Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1102W

Ctl_DidntUseAll=ARFM1102W: The ARFM Controller {0} chose to use only {1} seats when {2} were available.

Explanation

This usually indicates something is broken.

User response

Consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1103E

Ctl_OptFailed=ARFM1103E: The ARFM Controller {0} suffered an unexpected exception while solving the optimization (which was for {1} variables): {2}

Explanation

Internal failure.

User response

Consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1104E

Ctl_OptFailed1=ARFM1104E: Model {0} was {1}

Explanation

This message provides additional detail on the preceding ARFM1103E message.

User response

No additional action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1105E

Ctl_OptFailed2=ARFM1105E: Allocation {0} was {1}

Explanation

This message provides additional detail on the preceding ARFM1103E message.

User response

No additional action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1106E

Ctl_CreateLogFailed=ARFM1106E: ARFM Controller {0} tried to create statistics log file named {1} but failed: {2}

Explanation

Controller statistics will not be written to a file.

User response

Check for filesystem problems; consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1107I

Ctl_Initted=ARFM1107I: The ARFM Controller for {0} has been activated on this machine; control cycle length is {1} milliseconds.

Explanation

The local instance of the Autonomic Request Flow Management controllers MBean for the named cell has been activated (a cell has an ARFM Controllers MBean instance in each ODR, node agent, and deployment manager, for high availability; only one is activated at a time).

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1108I

Ctl_Stopped=ARFM1108I: The ARFM Controller for {0} has been shutdown on this machine.

Explanation

The local instance of the Autonomic Request Flow Management controllers MBean for the named cell has been deactivated (a cell has an ARFM Controllers MBean instance in each ODR, node agent, and deployment manager, for high availability; only one is activated at a time).

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1109W

Ctl_NoRhos=ARFM1109W: The ARFM Controller for {0} suffered an unexpected internal failure (to produce CPU utilizations per cluster).

Explanation

This should not happen. It will degrade the behavior of the Placement Controller.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1110W

Ctl_BadRho=ARFM1110W: The ARFM Controller for {0} suffered an unexpected internal failure: the CPU utilization fraction for cluster {1} was found to be {2}.

Explanation

This should not happen. It will degrade the behavior of the Placement Controller.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM1111W

Ctl_NoCap=ARFM1111W: The ARFM Controller skipped an optimization cycle due to complete lack of capacity information.

Explanation

Some other problem prevented the Controller from extracting the needed capacity information from the ODC tree.

User response

Find and fix the other problem.

Parent topic: [ARFM](#)

ARFM1112W

Ctl_CANotImplemented=ARFM1112W: The ARFM Controller was configured to use the obsolete functionality known as the Concurrency Adjuster --- ignored.

Explanation

This is an error in configuration that is not exposed to customers.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1113W

Ctl_CBNotImplemented=ARFM1113W: The ARFM Controller was configured to use the obsolete functionality known as the Concurrency Bounder --- ignored.

Explanation

This is an error in configuration that is not exposed to customers.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1114W

Ctl_NoEntryKappa=ARFM1114W: The Work Profiler supplied no speed factor for service class {2} and the entry tier {1} in cell {0}; using 1.

Explanation

This is a bug.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1115W

Ctl_NoEntryAlpha=ARFM1115W: The Work Profiler supplied no work factor for service class {2} and the entry tier {1} in cell {0}; using 1.

Explanation

This is a bug.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1116W

Ctl_NoKappaForRollup=ARFM1116W: The Work Profiler supplied no speed factor for the requests of service class {1} coming through gateway {0} to deployment target {2}.

Explanation

This is a bug.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1117W

Ctl_NoAlphaForRollup=ARFM1117W: The Work Profiler supplied no work factor for the requests of service class {1} coming through gateway {0} to deployment target {2}.

Explanation

This is a bug.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM1118E

Ctl_ManyClasses=ARFM1118E: Too many service policies ({2}) from ARFM gateway {1} in cell {0}

Explanation

The ARFM Controller can not handle more than 32,767 service policies, but received statistics for more than that many.

User response

If your installation really has more than 32,767 service policies, tell IBM that this limit needs to be raised, otherwise report a bug.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1119I

Ctl_QsDrp=ARFM1119I: From {0} to {1} the autonomic request flow management gateway for the {2} protocol family in the {5} server on the {4} node in the {3} cell for messages destined for the {7} cluster in the {6} cell and having service policy {8} refused admission to {9} messages; {10} follow-up messages were not admitted; {11} of the first received transmissions of non-follow-up messages were not enqueued: {12} due to the limit on queue length, {13} due to the limit on the total number of active messages, {14} due to the logical dialog admission rate limit, {15} due to memory overload protection, and {16} due to other routing-related reasons; {17} retransmitted non-follow-up messages were also not enqueued; {18} messages were refused after spending some time in a queue.

Explanation

This message gives information when messages are refused admission.

User response

None required.

Parent topic: [ARFM](#)

ARFM1201E

FMM_ReadCfgFailed=ARFM1201E: Attempt to read config from file {0} failed: {1}

Explanation

Internal error; default configuration will be used.

User response

Check for filesystem problems; consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1221E

GWD_MultiDT=ARFM1221E: ARFM Controller for {0} got data from gateway {1} that mentions more than one deployment target; {2} and {3} are two of them.

Explanation

Internal error; each gateway is supposed to serve only one deployment target.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1241W

DPS_e_OldQStats=ARFM1241W: Speed request smoother for gateway {0} rejected QStats ending {1} because that is older than the smoother's previous QStats (which ended at {2}).

Explanation

Internal error, should not happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM1242W

DPS_e_WrongQStats=ARFM1242W: The speed request smoother for ARFM gateway {0} saw a gap in the series of statistics messages: one message reported on a period of time ending at {2} and the next message reported on a period of time beginning at {1}.

Explanation

This is an internal problem that should not normally occur; sometimes it occurs as a consequence of other problems that disrupt the operation of the ARFM controller.

User response

Look for related problems, and report to IBM if you can not find any.

Parent topic: [ARFM](#)

ARFM1261W

DPWJ_NoWorkFactors=ARFM1261W: ARFM is unable to estimate the computational demands of the requests that arrive through gateway {0} due to missing work factors.

Explanation

This is an internal problem that should not occur.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM1262W

DPWJ_NoSpeedFactors=ARFM1262W: ARFM is unable to estimate the computational demands of the requests that arrive through gateway {0} due to missing speed factors.

Explanation

This is an internal problem that should not occur.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1981W

AlSq_RejectAckUnk=ARFM1981W: {0} ignoring allocation ack from unknown server {1}.

Explanation

Internal inconsistency. Should not happen.

User response

Consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1982W

AlSq_RejectAllocUnk=ARFM1982W: {0} ignoring allocation for unknown server {1}.

Explanation

Internal inconsistency. Should not happen.

User response

Consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM1983W

AlSq_OtherDealloc=ARFM1983W: {0} noted deallocation from some other source (numInst={1}, desAlloc[i]={2}).

Explanation

Internal inconsistency. Should not happen.

User response

Consult IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2061E

AIRO_HandleFailed=ARFM2061E: AllocReceiver {0} for cell {1} suffered unexpected exception handling received message: {2}

Explanation

Internal error.

User response

None; contact IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2241E

ESSI_SendFailed=ARFM2241E: BB-based send of an ARFM EvaldStats message for cell + Node Group {0} failed: {1}

Explanation

This is an internal communication failure. ARFM function will be significantly degraded.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2261E

ESSO_StartupDied=ARFM2261E: The JMS-based ARFM EvaldStats message sender for cell + Node Group {0} failed to initialize: {1}

Explanation

This is an internal communication failure. ARFM function will be significantly degraded.

User response

Check JMS provider setup for ARFM.

Parent topic: [ARFM](#)

ARFM2262E

ESSO_SendFailed=ARFM2262E: JMS-based send of an ARFM EvaldStats message for cell + Node Group {0} failed: {1}

Explanation

This is an internal communication failure. ARFM function will be significantly degraded.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2321E

MDS_SendFailed=ARFM2321E: Send of an ARFM mode message failed: {1}

Explanation

This is an internal communication failure.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

ARFM2341E

MDSO_StartupDied=ARFM2341E: A JMS-based ARFM mode message sender failed to initialize: {0}

Explanation

This is an internal communication failure.

User response

Check JMS provider setup for ARFM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2342E

MDSO_SendFailed=ARFM2342E: Send of an ARFM mode message failed: {0}

Explanation

This is an internal communication failure.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

ARFM2381E

NSRO_HandleFailed=ARFM2381E: {0} suffered unexpected exception handling received message: {1}

Explanation

Internal error.

User response

None; contact IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2401E

NSS_SendFailed=ARFM2401E: Send of an ARFM NodeQStats message failed: {0}

Explanation

This is an internal communication failure.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2421E

NSSO_StartupDied=ARFM2421E: A JMS-based ARFM NodeQStats message sender failed to initialize: {0}

Explanation

This is an internal communication failure.

User response

Check JMS provider setup for ARFM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2422E

NSSO_SendFailed=ARFM2422E: JMS-based send of an ARFM NodeQStats message failed: {0}

Explanation

This is an internal communication failure.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2581E

StSO_StartupDied=ARFM2581E: The JMS-based ARFM QStats message sender for cell + NodeGroup {0}, ODR {1}, failed to initialize: {2}

Explanation

This is an internal communication failure.

User response

Check JMS provider setup for ARFM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2582E

StSO_SendFailed=ARFM2582E: JMS-based send of an ARFM QStats message for cell + NodeGroup {0}, ODR {1}, failed: {2}

Explanation

This is an internal communication failure.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2621W

WtRO_RejectUnkObj=ARFM2621W: {0}, in cell + Node Group {1}, received unexpected type of object: {2}

Explanation

Internal inconsistency.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2622W

WtRO_RejectUnkMsg=ARFM2622W: {0}, in cell + Node Group {1}, received unexpected type of JMS message: {2}

Explanation

Internal inconsistency.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2623E

WtRO_HandleFailed=ARFM2623E: {0}, in cell + Node Group {1}, suffered unexpected exception handling received message: {2}

Explanation

Internal error.

User response

None; contact IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2661E

WtSO_StartupDied=ARFM2661E: The JMS-based ARFM weights message sender for cell + nodegroup {0} failed to initialize: {1}

Explanation

This is an internal communication failure.

User response

Check JMS provider setup for ARFM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2662E

WtSO_SetArgsLengths=ARFM2662E: The ARFM weights message sender was given inconsistent args: w.length={0}, classNamees.size={1}

Explanation

This is an internal inconsistency that should not arise.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2663E

WtSO_SendJmsFailed=ARFM2663E: JMS-based send of an ARFM weights message in cell + NodeGroup {0} to ARFM gateway {1} suffered a JMS failure: {2}

Explanation

This is an internal communication failure. ARFM functionality will be significantly degraded.

User response

Check JMS configuration for ARFM. Consult IBM for help.

Parent topic: [ARFM](#)

ARFM2664E

WtSO_SendFailed=ARFM2664E: JMS-based send of an ARFM weights message in cell + NodeGroup {0} to ARFM gateway {1} failed: {2}

Explanation

This is an internal communication failure. ARFM functionality will be significantly degraded.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2701E

ApRO_HandleFailed=ARFM2701E: {0} suffered unexpected exception handling received message: {1}

Explanation

Internal error.

User response

None; contact IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2721E

AIS_SendFailed=ARFM2721E: Send of an ARFM work estimates message failed: {0}

Explanation

This is an internal communication failure.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2741E

AISO_StartupDied=ARFM2741E: A JMS-based ARFM work factor estimates message sender failed to initialize: {0}

Explanation

This is an internal communication failure.

User response

Check JMS provider setup for ARFM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2742E

AISO_SendFailed=ARFM2742E: JMS-based send of an ARFM work factor estimates message failed: {0}

Explanation

This is an internal communication failure.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2801E

WRSI_SendFailed=ARFM2801E: JMS-based send of an ARFM results message for cell + Node Group {0} failed: {1}

Explanation

This is an internal communication failure. ARFM function will be significantly degraded.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2821E

WRSO_StartupDied=ARFM2821E: The JMS-based ARFM results message sender for cell + nodegroup {0} failed to initialize: {1}

Explanation

This is an internal communication failure.

User response

Check JMS provider setup for ARFM.

Parent topic: [ARFM](#)

ARFM2822E

WRSO_SendFailed=ARFM2822E: JMS-based send of an ARFM results message in cell + NodeGroup {0} to ARFM gateway {1} failed: {2}

Explanation

This is an internal communication failure. ARFM functionality will be significantly degraded.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2841I

TC_NormalToEmergencyThrottle=ARFM2841I: {0} NORMAL to EMERGENCY_THROTTLE transition

Explanation

This internal transition is in response to detecting an emergency condition.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2842I

TC_EmergencyUnthrottleToEmergencyThrottle=ARFM2842I: {0} EMERGENCY_UNTHROTTLER to EMERGENCY_THROTTLER transition

Explanation

This internal transition is in response to detecting an emergency condition.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2843I

TC_EmergencyThrottleToEmergencyUnthrottle=ARFM2843I: {0} EMERGENCY_THROTTLE to EMERGENCY_UNTHROTTLE transition

Explanation

This internal transition is in response to an emergency condition ending.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2844I

TC_EmergencyUnthrottleToNormal=ARFM2844I: {0} EMERGENCY_UNTHROTTLE to NORMAL transition

Explanation

This internal transition is in response to an emergency condition ending.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2845E

TC_ArfmSensorFired=ARFM2845E: {0} ARFM emergency sensor fired.

Explanation

ARFM control messages are not being received by the ODR.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2846E

TC_AsyncPmiDelaySensorFired=ARFM2846E: {0} Async PMI emergency sensor fired for node {1}.

Explanation

Async PMI messages are not being received by the ODR.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2847E

TC_AsyncPmiHighUtilSensorFired=ARFM2847E: {0} High CPU utilization sensor fired for node {1}.

Explanation

Extremely high node CPU utilization detected.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2848I

TC_ArfmSensorUnfired=ARFM2848I: {0} ARFM emergency sensor unfired.

Explanation

ARFM control messages resumed arrival at the ODR.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2849I

TC_AsyncPmiDelaySensorUnfired=ARFM2849I: {0} Async PMI emergency sensor unfired for node {1}.

Explanation

Async PMI messages resumed arrival at the ODR.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2850I

TC_AsyncPmiHighUtilSensorUnfired=ARFM2850I: {0} High CPU utilization sensor unfired for node {1}.

Explanation

Extremely high node CPU utilization is no longer detected.

User response

None.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2881E

SSS_SendFailed=ARFM2881E: Send of an ARFM ServerStats message failed: {0}

Explanation

This is an internal communication failure.

User response

Consult IBM for help.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2901E

Que_InitrAmongFollows=ARFM2901E: The {0} service policy of the {1} cell applies to traffic that is subject to admission control for CPU overload protection and includes both follow-up and non-follow-up messages; an example of the former was seen as recently as {2} and a current example of the latter is {3}.

Explanation

A service policy that applies to traffic subject to admission control for CPU overload protection should apply exclusively to follow-up messages or exclusively to non-follow-up messages; a follow-up message is one that is part of a pre-existing session or dialog.

User response

Revise the classification rules for the traffic currently classified into the named service policy.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM2902E

Que_FollowAmongIntrs=ARFM2902E: The {0} service policy of the {1} cell applies to traffic that is subject to admission control for CPU overload protection and includes both follow-up and non-follow-up messages; an example of the latter was seen as recently as {2} and a current example of the former is {3}.

Explanation

A service policy that applies to traffic subject to admission control for CPU overload protection should apply exclusively to follow-up messages or exclusively to non-follow-up messages; a follow-up message is one that is part of a pre-existing session or dialog.

User response

Revise the classification rules for the traffic currently classified into the named service policy.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4001E

QTI_SetQTIListenerFail=ARFM4001E: An Error occurs when setting QTI listener; data is {0}, and the exception is {1}.

Explanation

An Exception occurs when setting QTI listener.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

ARFM4002E

QTI_ODCInitFailed=ARFM4002E: ODC initialization failure: {0}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4021E

PQ_DlgAuxFail=ARFM4021E: An Error occurs when getting or setting DialogAux; data is {0}, and the exception is {1}.

Explanation

An Exception occurs when getting or setting DialogAux.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

ARFM4061E

WDFL_SetDlgAuxFail=ARFM4061E: An Error occurs when setting DialogAux; data is {0}, and the exception is {1}.

Explanation

An Exception occurs when setting DialogAux.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

ARFM4081E

SCP_GetQTIFail1=ARFM4081E: An unexpected runtime exception occurred while initializing the SIP statistics gatherer for autonomic request flow management: {0}.

Explanation

This should not happen.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

ARFM4082I

SCP_Enabled=ARFM4082I: Initialized the SIP statistics gatherer for autonomic request flow management.

Explanation

This reports that autonomic request flow management will be applied to SIP messages coming in to this process.

User response

No action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4083I

SCP_Disabled=ARFM4083I: The SIP statistics gatherer for autonomic request flow management is disabled.

Explanation

This reports that autonomic request flow management will NOT be applied to SIP messages coming in to this process.

User response

No action required --- unless that management was expected, in which case check the configuration and report to IBM if there is a problem.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4084I

SCP_MemEnabled=ARFM4084I: Autonomic memory overload protection is enabled.

Explanation

The memory protection parameter used by WLM will be set autonomically

User response

No action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4085I

SCP_MemDisabled=ARFM4085I: Autonomic memory overload protection is disabled.

Explanation

The memory protection parameter used by WLM will NOT be set autonomically

User response

No action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4101E

MOC_ODCInitFailed=ARFM4101E: ODC initialization failure: {0}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4102W

MOC_IBMGcUnknown=ARFM4102W: Fail to load the JVM arguments on IBM JVM. {0}.

Explanation

Some JVM internal call may be changed. Common memory overload protection method will be used.

User response

No action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4103E

MOC_ThreReportErr=ARFM4103E: Importance threshold report failed. Server: {0}. Exception: {1}

Explanation

ODC may occur some error. No new importance threshold will be transmitted till it recovers.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4104I

MOC_Parm_GcDt=ARFM4104I: Absent an override via the explicitGcPeriod cluster custom property, autonomic memory overload protection will wait {0} seconds between explicitly triggered garbage collections.

Explanation

Autonomic memory overload protection will explicitly trigger a garbage collection when the memory usage is high, unless this function is disabled or a full garbage collection has been done too recently.

User response

If the chosen interval is not good, set the explicitGcPeriod custom property on the cluster to indicate the desired interval; a value of zero disables this function.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4105I

MOC_Parm_GcDtOff=ARFM4105I: Absent an override via the explicitGcPeriod cluster custom property, autonomic memory overload protection will not explicitly trigger garbage collections.

Explanation

Autonomic memory overload protection will explicitly trigger a garbage collection when the memory usage is high, unless this function is disabled or a full garbage collection has been done too recently.

User response

If the chosen interval is not good, set the explicitGcPeriod custom property on the cluster to indicate the desired interval; a value of zero disables this function.

Parent topic: [ARFM](#)

ARFM4106I

MOC_Parm_GcDtRel=ARFM4106I: Absent an override via the explicitGcPeriod cluster custom property, the time between explicit garbage collections for the purpose of memory overload protection will be relative to the configured time between successive settings of the memory overload protection control value in the On-Demand Configuration facility.

Explanation

Autonomic memory overload protection will explicitly trigger a garbage collection when the memory usage is high, unless this function is disabled or a full garbage collection has been done too recently.

User response

If the chosen interval is not good, set the explicitGcPeriod custom property on the cluster to indicate the desired interval; a value of zero disables this function.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4107I

MOC_Parm_MemBasic=ARFM4107I: The maximum heap size is {0} bytes; the low threshold for memory overload protection is {1} bytes and the high threshold is {2} bytes.

Explanation

This message reports the basic parameters of the memory overload protection.

User response

Change the configuration if desired.

Parent topic: [ARFM](#)

ARFM4108I

MOC_Parm_Nurs=ARFM4108I: Absent an override via the maxNonFullGcMemoryRelease cluster custom property, autonomic memory overload protection will figure a full garbage collection has happened when more than {0} kilobytes have been freed.

Explanation

Autonomic memory overload protection will explicitly trigger a garbage collection when the memory usage is high, unless this function is disabled or a full garbage collection has been done too recently; when a generational garbage collection technique is in use, a full garbage collection is detected when the memory usage drops by more than a lesser collection can free.

User response

If the chosen value is not good, set the maxNonFullGcMemoryRelease custom property on the cluster to indicate the critical amount of freed memory; a value of zero means every collection is deemed full, and a huge value means no collection (except those explicitly initiated by the protection function) will be deemed full.

Parent topic: [ARFM](#)

ARFM4109I

MOC_Parm_NursAlways=ARFM4109I: Absent an override via the maxNonFullGcMemoryRelease cluster custom property, autonomic memory overload protection will assume every garbage collection is a full one.

Explanation

Autonomic memory overload protection will explicitly trigger a garbage collection when the memory usage is high, unless this function is disabled or a full garbage collection has been done too recently; when a generational garbage collection technique is in use, a full garbage collection is detected when the memory usage drops by more than a lesser collection can free.

User response

If the chosen value is not good, set the maxNonFullGcMemoryRelease custom property on the cluster to indicate the critical amount of freed memory; a value of zero means every collection is deemed full, and a huge value means no collection (except those explicitly initiated by the protection function) will be deemed full.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4110I

MOC_Parm_NursNever=ARFM4110I: Absent an override via the maxNonFullGcMemoryRelease cluster custom property, autonomic memory overload protection will assume no garbage collection is a full one.

Explanation

Autonomic memory overload protection will explicitly trigger a garbage collection when the memory usage is high, unless this function is disabled or a full garbage collection has been done too recently; when a generational garbage collection technique is in use, a full garbage collection is detected when the memory usage drops by more than a lesser collection can free.

User response

If the chosen value is not good, set the maxNonFullGcMemoryRelease custom property on the cluster to indicate the critical amount of freed memory; a value of zero means every collection is deemed full, and a huge value means no collection (except those explicitly initiated by the protection function) will be deemed full.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4111I

MOC_UnklbmNurs=ARFM4111I: Generational garbage collection is in use in an IBM JRE, but the nursery size limit is not known exactly; assuming {0} kilobytes.

Explanation

The autonomic memory overload protection controller is unable to determine the nursery size limit.

User response

Either accept the assumed value, set the maxNonFullGcMemoryRelease custom property on the cluster, or use a non-generational collector.

Parent topic: [ARFM](#)

ARFM4112I

MOC_UnkVendor=ARFM4112I: Generational garbage collection is in use in a non-IBM JRE, but the nursery size limit is not known.

Explanation

The autonomic memory overload protection controller is unable to determine the nursery size limit.

User response

Either accept that explicit garbage collections will not be suppressed by ones initiated by the collector, set the maxNonFullGcMemoryRelease custom property on the cluster, or use a non-generational collector.

Parent topic: [ARFM](#)

ARFM4113E

MOC_ODCErr=ARFM4113E: The memory overload protection controller suffered an unexpected error when looking up the local process in the On-Demand Configuration facility: {0}

Explanation

On-Demand Configuration is used to convey the memory overload protection control settings from the local process to the On-Demand Routers; because of this failure, the control settings will not be automatically updated.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM4114I

MOC_GCTrig=ARFM4114I: The memory overload protection controller explicitly invoked a garbage collection (GC) at {0}; the previous detected major GC was at {1}, and the configured minimum time between explicit GCs was {2} seconds; the last time a heap usage sample was below {3} was {4}; the number of HTTP admissibility tests was {5} and {6} passed, and the choke fraction is {7}; the number of SIP admissibility tests was {8} and {9} passed, and the choke fraction is {10}; it is {11} that the garbage collector values throughput over pause times.

Explanation

When collector values throughput over minimizing stop-the-world times, it improves the accuracy of the memory usage estimate. Otherwise, it only happens under urgent condition: the admission fraction of new dialogs is too small, so that it is hard for the collector itself to trigger a full GC to release the garbage in memory.

User response

No action required.

Parent topic: [ARFM](#)

ARFM4115I

MOC_GCTrig2=ARFM4115I: The memory overload protection controller explicitly invoked a garbage collection (GC) at {0}; the previous detected major GC was at {1}, at which time the heap utilization became {2} percent, which is above the threshold of {3} percent; the configured minimum time between explicit GCs was {4} seconds; the number of HTTP admissibility tests was {5} and {6} passed, and the choke fraction is {7}; the number of SIP admissibility tests was {8} and {9} passed, and the choke fraction is {10}; it is {11} that the garbage collector values throughput over pause times.

Explanation

When collector values throughput over minimizing stop-the-world times, it improves the accuracy of the memory usage estimate. Otherwise, it only happens under urgent condition: the admission fraction of new dialogs is too small, so that it is hard for the collector itself to trigger a full GC to release the garbage in memory.

User response

No action required.

Parent topic: [ARFM](#)

ARFM4898W

WSCP_GENLWRN=ARFM4898W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something suspicious happened; this may or may not indicate an error.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4899E

WSCP_GENLERR=ARFM4899E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something unexpected happened.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM4998W

WCORE_GENLWRN=ARFM4998W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something suspicious happened; this may or may not indicate an error.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

ARFM4999E

WCORE_GENLERR=ARFM4999E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something unexpected happened.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

ARFM5001E

WCF_FileErr=ARFM5001E: ARFM suffered unexpected exception parsing internal config from properties file {0}: {1}.

Explanation

Something went wrong reading the file or analyzing its contents.

User response

Check for file existence and correct contents.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5002E

WCF_FindTreeFailed=ARFM5002E: Failed to get ODC Tree: {0}

Explanation

Something went wrong in this ODR when ARFM tried to gain access to the ODC Tree.

User response

Check the propagation mechanism.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5004E

WCF_NoCell=ARFM5004E: ARFM found no ODC vertex for the cell named {0}.

Explanation

An ARFM subcomponent is being instantiated for the named cell, but that cell is not found in the ODC tree; this should not happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5006E

WCF_LoadFailed=ARFM5006E: ARFM for cell {0} suffered unexpected exception parsing config (some from ODC, some from file {1}): {2}.

Explanation

Something went wrong reading the file or analyzing its contents.

User response

Check for file existence and correct contents.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5007I

WCF_Loaded=ARFM5007I: ARFM configuration for cell {0} is {1}.

Explanation

Informational message only.

User response

No action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5021E

TCT_OdclnitFailed=ARFM5021E: ARFM suffered unexpected failure during ODC setup: {0}

Explanation

Internal error, should not happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5022E

TCT_ReadFailed=ARFM5022E: ARFM suffered unexpected failure while looking up and reading configuration from ODC vertex for cell {0}: {1}.

Explanation

Internal error, should not happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5023E

TCT_BadEventType=ARFM5023E: ARFM got unexpected ODC event type.

Explanation

Internal error, should not happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5024E

TCT_EventErr=ARFM5024E: ARFM suffered unexpected exception while handling ODC event: {0}.

Explanation

Internal error, should not happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5041E

KOO_ReadFailed=ARFM5041E: ARFM suffered unexpected failure while looking up and reading configuration from ODC vertex for the cell {0}: {1}.

Explanation

Internal error, should not happen.

User response

Report to IBM.

Parent topic: [ARFM](#)

ARFM5998W

WMAN_GENLWRN=ARFM5998W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something suspicious happened; this may or may not indicate an error.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM5999E

WMAN_GENLERR=ARFM5999E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something unexpected happened.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM6001E

OST_ConfigIoExn=ARFM6001E: Got I/O exception {2} while reading WSMM configuration from {1} for {0}; using defaults.

Explanation

Something went wrong while trying to read grm.cfg.

User response

Debug it.

Parent topic: [ARFM](#)

ARFM6002E

OST_Broken=ARFM6002E: {0} is broken by exception while parsing config info from {1}: {2}.

Explanation

Something went wrong while parsing grm.cfg.

User response

Debug it.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8001E

GRM_NoConfigFile=ARFM8001E: ARFM Controller configuration file {0} cannot be read.

Explanation

Internal configuration error: the ARFM Controller cannot start because the configuration file cannot be read.

User response

Have IBM Services correct the error.

Parent topic: [ARFM](#)

ARFM8002E

GRM_Read0=ARFM8002E: Cannot happen.

Explanation

Internal error.

User response

Report program defect to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8003W

GRM_GetFmsFailed=ARFM8003W: Unexpected exception arose when looking up the FilterManagerService: {0}

Explanation

Internal error.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8004I

GRM_NoFMS=ARFM8004I: ARFM Controller disabled because this is not an ODR process.

Explanation

ARFM Controllers run only in On-Demand Router processes.

User response

None. Informational entry.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8005I

GRM_NotStandAlone=ARFM8005I: ARFM Controller disabled because it is configured to run in the HTTP Proxy Filter.

Explanation

ARFM Controllers may run in either the HTTP Proxy Filter or stand alone, the configuration is not stand alone.

User response

None. Informational entry.

Parent topic: [ARFM](#)

ARFM8021E

HAC_FindTreeFailed=ARFM8021E: Failed to get ODC Tree: {0}

Explanation

Something went wrong when the ARFM Controller tried to gain access to the ODC Tree.

User response

Check the propagation mechanism, make sure target.xml exists.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8022E

HAC_ReadTreeFailed=ARFM8022E: Failed to read ODC Tree: {0}

Explanation

Something went wrong when the ARFM Controller tried to read the ODC Tree.

User response

Check the propagation mechanism.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8023I

HAC_HamStarting=ARFM8023I: Starting ARFM controller for cell {0}, due to activation by HA-Manager.

Explanation

The controller that does Autonomic Request Flow Management is being started.

User response

Informational message only, no action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8024I

HAC_HamStopping=ARFM8024I: Stopping ARFM controller for cell {0}, due to de-activation by HA-Manager.

Explanation

The controller that does Autonomic Request Flow Management is being stopped.

User response

Informational message only, no action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8025W

HAC_UnknownEdge=ARFM8025W: Unknown edge change event {0}.

Explanation

The controller received an unknown edge change event.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8026W

HAC_UnknownChange=ARFM8026W: Unknown topology change event {0}.

Explanation

The controller received an unknown topology change event.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8027E

HAC_ControllerFailed=ARFM8027E: Attempt to instantiate ARFM Controller for node group {0} in cell {1} suffered unexpected failure: {2}.

Explanation

Internal failure.

User response

Debug or report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8028I

HAC_OdcStarting=ARFM8028I: Starting ARFM controller for cell {0}, due to On-Demand Configuration (ODC) change.

Explanation

The controller that does Autonomic Request Flow Management is being started.

User response

Informational message only, no action required.

Parent topic: [ARFM](#)

ARFM8029I

HAC_OdcStopping=ARFM8029I: Stopping ARFM controller for cell {0}, due to On-Demand Configuration (ODC) change.

Explanation

The controller that does Autonomic Request Flow Management is being stopped.

User response

Informational message only, no action required.

Parent topic: [ARFM](#)

ARFM8030E

HAC_GrpAbleFail=ARFM8030E: Unexpected exception arose while enabling or disabling ARFM controllers in the {0} cell (note: the central cell is {1} and the other connected cells are {2}): {3}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8031I

HAC_ResetStarting=ARFM8031I: Starting ARFM controller for cell {0}, to implement a reset.

Explanation

The controller that does Autonomic Request Flow Management is being started.

User response

Informational message only, no action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8032I

HAC_ResetStopping=ARFM8032I: Stopping ARFM controller for cell {0}, to implement a reset.

Explanation

The controller that does Autonomic Request Flow Management is being stopped.

User response

Informational message only, no action required.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8101E

WCSP_LocErr=ARFM8101E: The ARFM for cell {0} suffered an unexpected exception/error when trying to find the location of the MBean holding its controllers; the exception/error is {1}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8102E

WCSP_MBCallFail=ARFM8102E: The component status provider for ARFM suffered an unexpected exception when trying to query the ARFM controllers MBean: {0}.

Explanation

Internal failure.

User response

Report to IBM.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8501E

TT1_err=ARFM8501E: Error with one parm, {0}

Explanation

exp

User response

ua

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ARFM8998W

WCTL_GENLWRN=ARFM8998W: Something suspicious happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something suspicious happened; this may or may not indicate an error.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

ARFM8999E

WCTL_GENLERR=ARFM8999E: Something unexpected happened in ARFM; data is {0}, and the exception (if any) is {1}.

Explanation

Something unexpected happened.

User response

Debug it if you can, using the information provided, or contact IBM support.

Parent topic: [ARFM](#)

[Terms of Use](#) | [Feedback](#)

ASND

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

ASND0001E

ASND0001=ASND0001E: Exception occurred in nodedetect: {0}

ASND0002I

ASND0002=ASND0002I: Detected server {0} started on node {1}

ASND0003I

ASND0003=ASND0003I: Detected server {0} stopped on node {1}

ASND0004I

ASND0004=ASND0004I: Detected {0} processor(s) operating at {1} MHz with {2} Mb of total system memory.

ASND0005E

ASND0005=ASND0005E: Unable to obtain information on processor speed. Please check that the appropriate system libraries are installed to enable WebSphere to query processor information.

ASND0006I

ASND0006=ASND0006I: Node detection of non WebSphere servers managed by Extended Deployment is operational

ASND0007I

ASND0007=ASND0007I: Node detection for Extended Deployment is operational

ASND0008E

ASND0008=ASND0008E: Node detection suspects erroneous node capabilities for node {0} and node characteristic {1}

ASND0009I

ASND0009=ASND0009I: The complete set of all node capabilities as detected by Extended Deployment is: {0}

ASND1001W

ASND1001=ASND1001W: Gathering of node data from middlewareagents is being skipped because processing of the previously gathered information began at {0} and is still in progress.

ASND1002W

ASND1002=ASND1002W: Gathered node data {0} from middlewareagents is being skipped because processing of the previously gathered information began at {1} and is still in progress.

ASND0001E

ASND0001=ASND0001E: Exception occurred in nodedetect: {0}

Explanation

None

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [ASND](#)

[Terms of Use](#) | [Feedback](#)

ASND0002I

ASND0002=ASND0002I: Detected server {0} started on node {1}

Explanation

An informational message that indicates that this server has detected the start of the indicated server.

User response

None.

Parent topic: [ASND](#)

[Terms of Use](#) | [Feedback](#)

ASND0003I

ASND0003=ASND0003I: Detected server {0} stopped on node {1}

Explanation

An informational message that indicates that this server has detected the stop of the indicated server.

User response

None.

Parent topic: [ASND](#)

[Terms of Use](#) | [Feedback](#)

ASND0004I

ASND0004=ASND0004I: Detected {0} processor(s) operating at {1} MHz with {2} Mb of total system memory.

Explanation

An informational message that indicates the detected characteristics of the machine.

User response

None.

Parent topic: [ASND](#)

[Terms of Use](#) | [Feedback](#)

ASND0005E

ASND0005=ASND0005E: Unable to obtain information on processor speed. Please check that the appropriate system libraries are installed to enable WebSphere to query processor information.

Explanation

WebSphere was unable to query information about processor speed on this system.

User response

On AIX, install the bos.perf.libperfstat package. Consult IBM service for other platforms.

Parent topic: [ASND](#)

ASND0006I

ASND0006=ASND0006I: Node detection of non WebSphere servers managed by Extended Deployment is operational

Explanation

an informational message that indicate overall subsystem status

User response

None.

Parent topic: [ASND](#)

[Terms of Use](#) | [Feedback](#)

ASND0007I

ASND0007=ASND0007I: Node detection for Extended Deployment is operational

Explanation

an informational message that indicate overall subsystem status

User response

None.

Parent topic: [ASND](#)

[Terms of Use](#) | [Feedback](#)

ASND0008E

ASND0008=ASND0008E: Node detection suspects erroneous node capabilities for node {0} and node characteristic {1}

Explanation

Some capability of the node could not be determined.

User response

Consult IBM service.

Parent topic: [ASND](#)

[Terms of Use](#) | [Feedback](#)

ASND0009I

ASND0009=ASND0009I: The complete set of all node capabilities as detected by Extended Deployment is: {0}

Explanation

A detailed dump of all detected node capabilities.

User response

None.

Parent topic: [ASND](#)

[Terms of Use](#) | [Feedback](#)

ASND1001W

ASND1001=ASND1001W: Gathering of node data from middlewareagents is being skipped because processing of the previously gathered information began at {0} and is still in progress.

Explanation

This happens when the local CPUs are overloaded or something gets stuck.

User response

Contact IBM support if CPU overload is unexpected or not present.

Parent topic: [ASND](#)

ASND1002W

ASND1002=ASND1002W: Gathered node data {0} from middlewareagents is being skipped because processing of the previously gathered information began at {1} and is still in progress.

Explanation

This happens when the local CPUs are overloaded or something gets stuck.

User response

Contact IBM support if CPU overload is unexpected or not present.

Parent topic: [ASND](#)

ASPC

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

ASPC0001E

aspc.unknown.collector=ASPC0001E: Unknown Collector type creation requested: {0}

ASPC0008E

aspc.throwable=ASPC0008E: Exception is: {0}

ASPC0009E

aspc.throwable.linked=ASPC0009E: Linked exception is: {0}

ASPC0001E

aspc.unknown.collector=ASPC0001E: Unknown Collector type creation requested: {0}

Explanation

Creation of a collector of unknown type was requested.

User response

None

Parent topic: [ASPC](#)

[Terms of Use](#) | [Feedback](#)

ASPC0008E

aspc.throwable=ASPC0008E: Exception is: {0}

Explanation

None

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [ASPC](#)

[Terms of Use](#) | [Feedback](#)

ASPC0009E

aspc.throwable.linked=ASPC0009E: Linked exception is: {0}

Explanation

None

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [ASPC](#)

[Terms of Use](#) | [Feedback](#)

ASPH

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

ASPH0008E

asph.throwable=ASPH0008E: Exception is: {0}

ASPH0009E

asph.throwable.linked=ASPH0009E: Linked exception is: {0}

ASPH0008E

asph.throwable=ASPH0008E: Exception is: {0}

Explanation

None

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [ASPH](#)

[Terms of Use](#) | [Feedback](#)

ASPH0009E

asph.throwable.linked=ASPH0009E: Linked exception is: {0}

Explanation

None

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [ASPH](#)

[Terms of Use](#) | [Feedback](#)

ASPS

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[ASPS0008E](#)

asps.throwable=ASPS0008E: Exception is: {0}

[ASPS0009E](#)

asps.throwable.linked=ASPS0009E: Linked exception is: {0}

[ASPS0011E](#)

asps.nopeer=ASPS0011E: Failed to join management overlay network, with name={0}, due to {1}

[ASPS0012I](#)

APMI_UP1=ASPS0012I: Statistic gathering of non WebSphere servers managed by Extended Deployment is operational

[ASPS0013I](#)

APMI_UP2=ASPS0013I: Statistic gathering for Extended Deployment is operational

[ASPS0014I](#)

APMI_CoreDump=ASPS0014I: The complete set of state for healthy Statistic gathering is: {0}

[ASPS0015E](#)

APMI_CoreDumpErr=ASPS0015E: The complete set of state for unhealthy Statistic gathering is: {0}

[ASPS0016E](#)

APMI_SpecificStatErr=ASPS0016E: Server {0} has not received statistic {1} from server {2} since {3}, where time of reporting is {4}

[ASPS0017E](#)

APMI_NoDefaultTP=ASPS0017E: ThreadPoolRepository {0} utterly fails to yield a Default thread pool.

[ASPS0018W](#)

APMI_SkipPoll=ASPS0018W: Skipping regular gathering of middlewareagent statistics because processing of the previous collection began at {0} and is still in progress.

[ASPS0019W](#)

APMI_DropStats=ASPS0019W: Middlewareagent statistics {0} are being ignored because the previous batch began at {1} and is still being processed.

[ASPS0022E](#)

APMI_XDSYSTEM_REG_ERR=ASPS0022E: Unable to register XdSystem PMI module.

[ASPS0023W](#)

APMI_XDSYSTEM_HW_INFO_WARN=ASPS0023W: The logical partition on which this node resides has a mode of shared, a type of uncapped, but performance collection is not enabled.

ASPS0008E

asps.throwable=ASPS0008E: Exception is: {0}

Explanation

None

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [ASPS](#)

[Terms of Use](#) | [Feedback](#)

ASPS0009E

asps.throwable.linked=ASPS0009E: Linked exception is: {0}

Explanation

None

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [ASPS](#)

[Terms of Use](#) | [Feedback](#)

ASPS0011E

asps.nopeer=ASPS0011E: Failed to join management overlay network, with name={0}, due to {1}

Explanation

This process failed to join the WebSphere management overlay network, using the given name for itself, because of the given exception

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [ASPS](#)

ASPS0012I

APMI_UP1=ASPS0012I: Statistic gathering of non WebSphere servers managed by Extended Deployment is operational

Explanation

an informational message that indicates overall subsystem status

User response

None.

Parent topic: [ASPS](#)

[Terms of Use](#) | [Feedback](#)

ASPS0013I

APMI_UP2=ASPS0013I: Statistic gathering for Extended Deployment is operational

Explanation

an informational message that indicates overall subsystem status

User response

None.

Parent topic: [ASPS](#)

[Terms of Use](#) | [Feedback](#)

ASPS0014I

APMI_CoreDump=ASPS0014I: The complete set of state for healthy Statistic gathering is: {0}

Explanation

A detailed dump of all Statistic gathering state.

User response

None.

Parent topic: [ASPS](#)

[Terms of Use](#) | [Feedback](#)

ASPS0015E

APMI_CoreDumpErr=ASPS0015E: The complete set of state for unhealthy Statistic gathering is: {0}

Explanation

A detailed dump of all Statistic gathering state.

User response

Review core group state, and then contact IBM support.

Parent topic: [ASPS](#)

[Terms of Use](#) | [Feedback](#)

ASPS0016E

APMI_SpecificStatErr=ASPS0016E: Server {0} has not received statistic {1} from server {2} since {3}, where time of reporting is {4}

Explanation

A specific statistic has not arrived as expected.

User response

Review core group state, and then contact IBM support.

Parent topic: [ASPS](#)

ASPS0017E

APMI_NoDefaultTP=ASPS0017E: ThreadPoolRepository {0} utterly fails to yield a Default thread pool.

Explanation

This should not happen.

User response

Contact IBM support.

Parent topic: [ASPS](#)

[Terms of Use](#) | [Feedback](#)

ASPS0018W

APMI_SkipPoll=ASPS0018W: Skipping regular gathering of middlewareagent statistics because processing of the previous collection began at {0} and is still in progress.

Explanation

This happens when the local CPUs are overloaded or something gets stuck.

User response

Report to IBM support if CPU overload is unexpected or not present.

Parent topic: [ASPS](#)

ASPS0019W

APMI_DropStats=ASPS0019W: Middlewareagent statistics {0} are being ignored because the previous batch began at {1} and is still being processed.

Explanation

This happens when the local CPUs are overloaded or something gets stuck.

User response

Report to IBM support if CPU overload is unexpected or not present.

Parent topic: [ASPS](#)

ASPS0022E

APMI_XDSYSTEM_REG_ERR=ASPS0022E: Unable to register XdSystem PMI module.

Explanation

Unable to register the XdSystem PMI module.

User response

Review and collect the system logs and then contact IBM support.

Parent topic: [ASPS](#)

[Terms of Use](#) | [Feedback](#)

ASPS0023W

APMI_XDSYSTEM_HW_INFO_WARN=ASPS0023W: The logical partition on which this node resides has a mode of shared, a type of uncapped, but performance collection is not enabled.

Explanation

The logical partition on which this node resides has a mode of shared, a type of uncapped, but performance collection is not enabled. Performance collection must be enabled for WebSphere Virtual Enterprise to recognize more than the entitled capacity of a logical partition. If performance collection is not enabled, WebSphere Virtual Enterprise will not recognize any additional capacity above the configured entitled capacity.

User response

Enable the Allow Performance Collection feature for the logical partition in the Hardware Management Console.

Parent topic: [ASPS](#)

[Terms of Use](#) | [Feedback](#)

CHKW

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

CHKW3000I

INFO_COMPONENT_NAME=CHKW3000I: IBM WebSphere Validation

CHKW3001I

INFO_SUBCOMPONENT_NAME=CHKW3001I: NodeGroup Validation

CHKW3002E

ERROR_NODEGROUP_MEMBER_NAME_REQUIRED=CHKW3002E: The name is a node group member beneath node group {0} is absent.

CHKW3003E

ERROR_NODEGROUP_NAME_DUPLICATION=CHKW3003E: The node group name {0} is duplicated.

CHKW3004E

ERROR_NODEGROUP_MEMBER_NAME_DUPLICATION=CHKW3004E: The node group member name {0} is duplicated.

CHKW3006E

ERROR_NODE_BELONGS_TO_MULTIPLE_NODE_GROUPS=CHKW3006E: The node group member {0} is a member of multiple node groups.

CHKW3000I

INFO_COMPONENT_NAME=CHKW3000I: IBM WebSphere Validation

Explanation

No additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [CHKW](#)

[Terms of Use](#) | [Feedback](#)

CHKW3001I

INFO_SUBCOMPONENT_NAME=CHKW3001I: NodeGroup Validation

Explanation

No additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [CHKW](#)

[Terms of Use](#) | [Feedback](#)

CHKW3002E

ERROR_NODEGROUP_MEMBER_NAME_REQUIRED=CHKW3002E: The name is a node group member beneath node group {0} is absent.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [CHKW](#)

CHKW3003E

ERROR_NODEGROUP_NAME_DUPLICATION=CHKW3003E: The node group name {0} is duplicated.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [CHKW](#)

[Terms of Use](#) | [Feedback](#)

CHKW3004E

ERROR_NODEGROUP_MEMBER_NAME_DUPLICATION=CHKW3004E: The node group member name {0} is duplicated.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [CHKW](#)

CHKW3006E

ERROR_NODE_BELONGS_TO_MULTIPLE_NODE_GROUPS=CHKW3006E: The node group member {0} is a member of multiple node groups.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [CHKW](#)

CLFY

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

CLFY0001E

CLFY_NoProtocolScope=CLFY0001E: The protocol scope has not been initialized.

CLFY0002E

CLFY_NotSupportedScope=CLFY0002E: The protocol scope {0} supplied is not supported.

CLFY0003E

CLFY_Exception=CLFY0003E: Exception: {0}.

CLFY0004E

CLFY_NoFMSFound=CLFY0004E: Exception: {0}.

CLFY0005I

CLFY_NoeWLMFound=CLFY0005I: eWLM has not been installed.

CLFY0006I

CLFY_RoutingDebug0=CLFY0006I: Starting per request routing debug for level routingOnly.

CLFY0007I

CLFY_RoutingDebug1=CLFY0007I: Starting per request routing debug for level selectionOnly.

CLFY0008I

CLFY_RoutingDebug2=CLFY0008I: Starting per request routing debug for level all.

CLFY0009E

CLFY_ValidationMsg0=CLFY0009E: Error found in classification expression: {0}.

CLFY0001E

CLFY_NoProtocolScope=CLFY0001E: The protocol scope has not been initialized.

Explanation

Internal system error.

User response

Internal system error, report program defect to customer support.

Parent topic: [CLFY](#)

[Terms of Use](#) | [Feedback](#)

CLFY0002E

CLFY_NotSupportedScope=CLFY0002E: The protocol scope {0} supplied is not supported.

Explanation

Internal system error.

User response

Internal system error, report program defect to customer support.

Parent topic: [CLFY](#)

[Terms of Use](#) | [Feedback](#)

CLFY0003E

CLFY_Exception=CLFY0003E: Exception: {0}.

Explanation

Internal system error. An unexpected exception occurred.

User response

Report program defect to customer support.

Parent topic: [CLFY](#)

[Terms of Use](#) | [Feedback](#)

CLFY0004E

CLFY_NoFMSFound=CLFY0004E: Exception: {0}.

Explanation

The filter management service was not found, server error.

User response

Maybe due to a failed install, if that is not the case, report problem to customer support.

Parent topic: [CLFY](#)

[Terms of Use](#) | [Feedback](#)

CLFY0005I

CLFY_NoewLMFound=CLFY0005I: eWLM has not been installed.

Explanation

eWLM was not found to have been installed on this server.

User response

None.

Parent topic: [CLFY](#)

[Terms of Use](#) | [Feedback](#)

CLFY0006I

CLFY_RoutingDebug0=CLFY0006I: Starting per request routing debug for level routingOnly.

Explanation

This is the start of the routing debug to show only routing policy information.

User response

None.

Parent topic: [CLFY](#)

[Terms of Use](#) | [Feedback](#)

CLFY0007I

CLFY_RoutingDebug1=CLFY0007I: Starting per request routing debug for level selectionOnly.

Explanation

This is the start of the routing debug to show only target selector information.

User response

None.

Parent topic: [CLFY](#)

[Terms of Use](#) | [Feedback](#)

CLFY0008I

CLFY_RoutingDebug2=CLFY0008I: Starting per request routing debug for level all.

Explanation

This is the start of the routing debug to show both routing policy and target selector information.

User response

None.

Parent topic: [CLFY](#)

[Terms of Use](#) | [Feedback](#)

CLFY0009E

CLFY_ValidationMsg0=CLFY0009E: Error found in classification expression: {0}.

Explanation

The input expression contains an error. Error details are shown in the message above.

User response

Fix problem indicated in the expression and try again.

Parent topic: [CLFY](#)

CWFOR

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

CWFOR1000E

CWFOR1000E=CWFOR1000E: The templateName parameter cannot be specified with either the templateServerNode or templateServerName parameters.

CWFOR1001E

CWFOR1001E=CWFOR1001E: The templateServerNode and templateServerName parameters must both be specified.

CWFOR1002E

CWFOR1002E=CWFOR1002E: Cannot find server "{0}" on node "{1}".

CWFOR1003E

CWFOR1003E=CWFOR1003E: Matching template {0} could not be found.

CWFOR1000E

CWFOR1000E=CWFOR1000E: The templateName parameter cannot be specified with either the templateServerNode or templateServerName parameters.

Explanation

The user must either enter the templateName or the templateServerNode and templateServerName parameters.

User response

Specify the command with either the templateName option or the templateServerNode and the templateServerName options.

Parent topic: [CWFOR](#)

CWFOR1001E

CWFOR1001E=CWFOR1001E: The templateServerNode and templateServerName parameters must both be specified.

Explanation

When using the templateServerNode parameter you must also specify the templateServerName.

User response

The templateServerNode parameter requires the templateServerName parameter to be entered as well.

Parent topic: [CWFOR](#)

CWFOR1002E

CWFOR1002E=CWFOR1002E: Cannot find server "{0}" on node "{1}".

Explanation

The server that you are using as a template is not located on the node that you provided.

User response

Enter the name of valid server to be used as a template and the name of the node in which that server resides.

Parent topic: [CWFOR](#)

[Terms of Use](#) | [Feedback](#)

CWFOR1003E

CWFOR1003E=CWFOR1003E: Matching template {0} could not be found.

Explanation

A template could not be located with the name {0}

User response

Enter the name of valid template.

Parent topic: [CWFOR](#)

[Terms of Use](#) | [Feedback](#)

CWLRS

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWLRS0400I](#)

GEPS0400I=CWLRS0400I: The WebSphere Business Grid Endpoint Selector has initialized successfully

[CWLRS0401I](#)

GEPS0401I=CWLRS0401I: The WebSphere Business Grid Endpoint Selector was not initialized because Extended Deployment is not enabled

[CWLRS0402I](#)

GEPS0402I=CWLRS0402I: The WebSphere Business Grid Endpoint Selector has started successfully

[CWLRS0420I](#)

GEPS0420I=CWLRS0420I: The WebSphere Business Grid Endpoint Selector is stopping dynamic cluster {1} in node group {2} on node {3}

[CWLRS0421I](#)

GEPS0421I=CWLRS0421I: The WebSphere Business Grid Endpoint Selector is starting dynamic cluster {1} in node group {2} on node {3}

[CWLRS0450E](#)

GEPS0450E=CWLRS0450E: Unable to load service policy

[CWLRS0451E](#)

GEPS0451E=CWLRS0451E: Error creating Business Grid GridSchedulerStatsMgr object

[CWLRS0452E](#)

GEPS0452E=CWLRS0452E: Error initializing Business Grid Endpoint Selector

[CWLRS0455E](#)

GEPS0455E=CWLRS0455E: Job {0} could not be dispatched due to a database exception: {1}.

[CWLRS0490I](#)

GEPS0490I=CWLRS0490I: registered for async pmi stats path: {0}

[CWLRS0491I](#)

GEPS_Task_StopDC=CWLRS0491I: Dynamic Cluster Member {0} needs to be stopped.

[CWLRS0492I](#)

GEPS_Task_StartDC=CWLRS0492I: Dynamic Cluster Member {0} needs to be started.

[CWLRS0493I](#)

GEPS0453I=CWLRS0493I: Job {0} cannot be dispatched at this time as its job class {1} is running with full capacity.

[CWLRS0494I](#)

GEPS0454W=CWLRS0494I: Job {0} is being canceled as it has been running longer than {1} seconds.

[CWLRS6000I](#)

GAP_COMPONENT_STARTED=CWLRS6000I: GAP (Grid Application Placement) Component has initialized successfully on process {0}.

[CWLRS6001I](#)

GAP_AGENT_STARTED=CWLRS6001I: GAP (Grid Application Placement) Agent has initialized successfully.

[CWLRS6002I](#)

GAP_AGENT_STARTUP_FAILED=CWLRS6002I: GAP Agent failed during startup process.

[CWLRS6003I](#)

GAP_MEDIATOR_FOUND=CWLRS6003I: GAP Mediator is running on node {0} server process {1}.

[CWLRS6004I](#)

USE_APC_PROPERTY=CWLRS6004I: APC Endpoint Selection is set to {0}.

[CWLRS6005I](#)

MAX_CONCURRENT_DISPATCHERS_PROPERTY=CWLRS6005I: Compute Grid maximum number of job dispatchers threads is set to {0}.

[CWLRS6006I](#)

JOB_CLASS_INFO=CWLRS6006I: Job class {0}, Importance {1}, Service Class {2}, Service Goal Type {3}, Application Type {4}, Submitter {5}.

[CWLRS6007I](#)

JOB_SERVICE_INFO=CWLRS6007I: Job Arrival Time {0}, Goal Max Completion Time {1}, Goal Max Queue Time {2}, Breach Time {3}.

CWLRS6008I

APC_WILL_SELECT_ENDPOINT=CWLRS6008I: APC (Application Placement Controller) will select an endpoint to execute this job.

CWLRS6009I

USER_CHOSEN_NOT_APC=CWLRS6009I: APC is active but user has chosen not to use it. GAP will make the endpoint selection.

CWLRS6010I

CLUSTER_TYPE_NOT_SUPPORT_BY_APC=CWLRS6010I: Unclustered servers are not supported by APC. GAP will make the endpoint selection.

CWLRS6011I

APC_NOT_ACTIVE=CWLRS6011I: APC is not active. GAP will make the endpoint selection.

CWLRS6012I

APC_CALLED_DISPATCH_JOB=CWLRS6012I: APC has selected endpoint {0} to execute the job.

CWLRS6013I

GAP_DISPATCHING_JOB_TO_ENDPOINT=CWLRS6013I: GAP is dispatching job {0}. Job queue time {1} seconds.

CWLRS6014I

BLOCKED_BY_APPLICATION_NOT_STARTED=CWLRS6014I: Job cannot be dispatched at this moment. Job application is not started. Job is waiting for the application to be started.

CWLRS6015I

BLOCKED_BY_PORT_NOT_ACTIVE=CWLRS6015I: Job cannot be dispatched at this moment. HTTP ports are not active. Job is waiting for the HTTP ports to become active.

CWLRS6016I

BLOCKED_BY_ENDPOINT_NOT_AVAILABLE=CWLRS6016I: Job cannot be dispatched at this moment. Make sure the endpoint is active and the job application is started and the HTTP ports are active. Job is waiting for an active endpoint.

CWLRS6017I

BLOCKED_BY_GEE_NOT_STARTED=CWLRS6017I: Job cannot be dispatched at this moment. GEE application is not started. Job is waiting for GEE application to be started.

CWLRS6018I

BLOCKED_BY_GEE_NOT_INSTALLED=CWLRS6018I: Job cannot be dispatched at this moment. GEE application is not installed.

CWLRS6019I

REMOVE_JOB=CWLRS6019I: Job has been removed from GAP queue.

CWLRS6020I

COMPLETE_JOB=CWLRS6020I: GAP has been notified that the job has completed.

CWLRS6021I

PERMITTED_NODES=CWLRS6021I: List of eligible endpoints to execute the job: {0}.

CWLRS6022I

INVALID_GOAL_TYPE=CWLRS6022I: Goal type {0} is not supported. Job will be assigned to Completion Time goal type.

CWLRS6023I

BLOCKED_BY_NO_JOBCLASS_CAPACITY=CWLRS6023I: Job {0} cannot be dispatched at this time as its job class {1} is running with full capacity.

CWLR0400I

GEPS0400I=CWLR0400I: The WebSphere Business Grid Endpoint Selector has initialized successfully

Explanation

Informational message

User response

No action required

Parent topic: [CWLR](#)

[Terms of Use](#) | [Feedback](#)

CWLRS0401I

GEPS0401I=CWLRS0401I: The WebSphere Business Grid Endpoint Selector was not initialized because Extended Deployment is not enabled

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLR0402I

GEPS0402I=CWLR0402I: The WebSphere Business Grid Endpoint Selector has started successfully

Explanation

Informational message

User response

No action required

Parent topic: [CWLR](#)

[Terms of Use](#) | [Feedback](#)

CWLRS0420I

GEPS0420I=CWLRS0420I: The WebSphere Business Grid Endpoint Selector is stopping dynamic cluster {1} in node group {2} on node {3}

Explanation

It was determined that it was necessary to stop the specified dynamic cluster in order to make room for a different one to start.

User response

No action required

Parent topic: [CWLRS](#)

CWLRS0421I

GEPS0421I=CWLRS0421I: The WebSphere Business Grid Endpoint Selector is starting dynamic cluster {1} in node group {2} on node {3}

Explanation

It was determined that there was a need to start an instance of the specified dynamic cluster to process a job in queue.

User response

No action required

Parent topic: [CWLRS](#)

CWLR0450E

GEPS0450E=CWLR0450E: Unable to load service policy

Explanation

Service policy relationships failed to load

User response

Ensure cell connectivity

Parent topic: [CWLR](#)

[Terms of Use](#) | [Feedback](#)

CWLRS0451E

GEPS0451E=CWLRS0451E: Error creating Business Grid GridSchedulerStatsMgr object

Explanation

A system or internal error has occurred

User response

Contact administrator

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLR0452E

GEPS0452E=CWLR0452E: Error initializing Business Grid Endpoint Selector

Explanation

A system or internal error has occurred

User response

Contact administrator

Parent topic: [CWLR](#)

CWLRS0455E

GEPS0455E=CWLRS0455E: Job {0} could not be dispatched due to a database exception: {1}.

Explanation

The Job {0} could not be dispatched. There was a database exception. Please see logs for details.

User response

Examine the system logs for the underlying failure.

Parent topic: [CWLRS](#)

CWLR0490I

GEPS0490I=CWLR0490I: registered for async pmi stats path: {0}

Explanation

Informational message

User response

No action required

Parent topic: [CWLR](#)

[Terms of Use](#) | [Feedback](#)

CWLR0491I

GEPS_Task_StopDC=CWLR0491I: Dynamic Cluster Member {0} needs to be stopped.

Explanation

The Business Grid Endpoint Selector has determined available resources will be better utilized by stopping the specified Dynamic Cluster Member.

User response

Approve or deny the change.

Parent topic: [CWLR0](#)

CWLRS0492I

GEPS_Task_StartDC=CWLRS0492I: Dynamic Cluster Member {0} needs to be started.

Explanation

The Business Grid Endpoint Selector has determined available resources will be better utilized by starting the specified Dynamic Cluster Member.

User response

Approve or deny the change.

Parent topic: [CWLRS](#)

CWLRS0493I

GEPS0453I=CWLRS0493I: Job {0} cannot be dispatched at this time as its job class {1} is running with full capacity.

Explanation

The total outstanding jobs belonging to this job class have exceeded the configured limit. New jobs belonging to {1} cannot be dispatched until some of the outstanding jobs complete.

User response

No action required

Parent topic: [CWLRS](#)

CWLRS0494I

GEPS0454W=CWLRS0494I: Job {0} is being canceled as it has been running longer than {1} seconds.

Explanation

The maximum execution time of job class {2} is {1} seconds. Job {0} has been canceled as it has been running longer than {1} seconds.

User response

Ensure that appropriate values are configured for this job class.

Parent topic: [CWLRS](#)

CWLRS6000I

GAP_COMPONENT_STARTED=CWLRS6000I: GAP (Grid Application Placement) Component has initialized successfully on process {0}.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6001I

GAP_AGENT_STARTED=CWLRS6001I: GAP (Grid Application Placement) Agent has initialized successfully.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6002I

GAP_AGENT_STARTUP_FAILED=CWLRS6002I: GAP Agent failed during startup process.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6003I

GAP_MEDIATOR_FOUND=CWLRS6003I: GAP Mediator is running on node {0} server process {1}.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6004I

USE_APC_PROPERTY=CWLRS6004I: APC Endpoint Selection is set to {0}.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6005I

MAX_CONCURRENT_DISPATCHERS_PROPERTY=CWLRS6005I: Compute Grid maximum number of job dispatchers threads is set to {0}.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6006I

JOB_CLASS_INFO=CWLRS6006I: Job class {0}, Importance {1}, Service Class {2}, Service Goal Type {3}, Application Type {4}, Submitter {5}.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6007I

JOB_SERVICE_INFO=CWLRS6007I: Job Arrival Time {0}, Goal Max Completion Time {1}, Goal Max Queue Time {2}, Breach Time {3}.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6008I

APC_WLL_SELECT_ENDPOINT=CWLRS6008I: APC (Application Placement Controller) will select an endpoint to execute this job.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6009I

USER_CHOOSEN_NOT_APC=CWLRS6009I: APC is active but user has chosen not to use it. GAP will make the endpoint selection.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6010I

CLUSTER_TYPE_NOT_SUPPORT_BY_APC=CWLRS6010I: Unclustered servers are not supported by APC. GAP will make the endpoint selection.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6011I

APC_NOT_ACTIVE=CWLRS6011I: APC is not active. GAP will make the endpoint selection.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6012I

APC_CALLED_DISPATCH_JOB=CWLRS6012I: APC has selected endpoint {0} to execute the job.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLR6013I

GAP_DISPATCHING_JOB_TO_ENDPOINT=CWLR6013I: GAP is dispatching job {0}. Job queue time {1} seconds.

Explanation

Informational message

User response

No action required

Parent topic: [CWLR](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6014I

BLOCKED_BY_APPLICATION_NOT_STARTED=CWLRS6014I: Job cannot be dispatched at this moment. Job application is not started. Job is waiting for the application to be started.

Explanation

Informational message.

User response

Start the job application.

Parent topic: [CWLRS](#)

CWLRS6015I

BLOCKED_BY_PORT_NOT_ACTIVE=CWLRS6015I: Job cannot be dispatched at this moment. HTTP ports are not active. Job is waiting for the HTTP ports to become active.

Explanation

Informational message

User response

Make sure the HTTP ports are active.

Parent topic: [CWLRS](#)

CWLRS6016I

BLOCKED_BY_ENDPOINT_NOT_AVAILABLE=CWLRS6016I: Job cannot be dispatched at this moment. Make sure the endpoint is active and the job application is started and the HTTP ports are active. Job is waiting for an active endpoint.

Explanation

Informational message

User response

Start the endpoint where the job application is installed.

Parent topic: [CWLRS](#)

CWLRS6017I

BLOCKED_BY_GEE_NOT_STARTED=CWLRS6017I: Job cannot be dispatched at this moment. GEE application is not started. Job is waiting for GEE application to be started.

Explanation

Informational message

User response

Start the endpoint where the grid execution environment is configured.

Parent topic: [CWLRS](#)

CWLRS6018I

BLOCKED_BY_GEE_NOT_INSTALLED=CWLRS6018I: Job cannot be dispatched at this moment. GEE application is not installed.

Explanation

Informational message

User response

Configure the job scheduler and grid execution environment.

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLR6019I

REMOVE_JOB=CWLR6019I: Job has been removed from GAP queue.

Explanation

Informational message

User response

No action required

Parent topic: [CWLR](#)

[Terms of Use](#) | [Feedback](#)

CWLR6020I

COMPLETE_JOB=CWLR6020I: GAP has been notified that the job has completed.

Explanation

Informational message

User response

No action required

Parent topic: [CWLR](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6021I

PERMITTED_NODES=CWLRS6021I: List of eligible endpoints to execute the job: {0}.

Explanation

Informational message

User response

No action required

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6022I

INVALID_GOAL_TYPE=CWLRS6022I: Goal type {0} is not supported. Job will be assigned to Completion Time goal type.

Explanation

Informational message

User response

The specified goal type is not supported. Supported goal types are Completion Time and Discretionary.

Parent topic: [CWLRS](#)

[Terms of Use](#) | [Feedback](#)

CWLRS6023I

BLOCKED_BY_NO_JOBCLASS_CAPACITY=CWLRS6023I: Job {0} cannot be dispatched at this time as its job class {1} is running with full capacity.

Explanation

The total outstanding jobs belonging to this job class have exceeded the configured limit. New jobs belonging to {1} cannot be dispatched until some of the outstanding jobs complete.

User response

No action required

Parent topic: [CWLRS](#)

CWOBB

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

CWOBB0001I

Cfg_Given=CWOBB0001I: The overlay-based bulletin board has been configured with a value of {1} for the parameter named {0} (the default, possibly dependent on other configuration settings, was {2}).

CWOBB0002E

Cfg_BadFloor=CWOBB0002E: The overlay-based bulletin board was configured with a minimum broadcast threshold of {0}, which is invalid; the bulletin board will use a minimum broadcast threshold of 1.

CWOBB0003E

Cfg_Bad1000=CWOBB0003E: The overlay-based bulletin board was configured with an invalid value ({0}) to use as the broadcast threshold when the system has 1000 processes; the threshold used in that case will be {1}.

CWOBB1000I

BBF_Init=CWOBB1000I: The overlay-based bulletin board has been initialized in process {0} in core group {1} in cell {2}.

CWOBB1001I

BBF_Broke=CWOBB1001I: The overlay-based bulletin board has failed to initialize in process {0} in core group {1} in cell {2}.

CWOBB1002W

BBF_ClockJerk=CWOBB1002W: The local machine clock has jumped backwards at least {0} milliseconds: a reading of {1} ({2}) was followed, some time later, by a reading of {3} ({4}).

CWOBB1003E

BBF_Callback_Throw=CWOBB1003E: The Bulletin Board subscriber {3} for board {0}, scope {1}, and subject {2} threw an exception or error {4}.

CWOBB1004E

BBF_NoPeer=CWOBB1004E: There was an unexpected exception while initializing the SON overlay; the exception is {0}.

CWOBB1005I

BBF_DUMPED=CWOBB1005I: The overlay-based bulletin board wrote its internal state, in concrete form to {0} and in abstract form to {1}.

CWOBB1006E

BBF_LoopbackConnection=CWOBB1006E: The overlay-based bulletin board on process {0} got TCP connection from itself.

CWOBB1007I

BBF_STATS=CWOBB1007I: Cumulative bulletin board factory statistics are {0}, rates since previous stats are {1}.

CWOBB1200W

Proc_SendQueueOrder=CWOBB1200W: The process queue of the bulletin board had a surprising entry (for the process named {3}, with send time {4} = {5}); this was discovered when inserting an entry for the process named {0} with send time {1} = {2}.

CWOBB2000I

BBS_ConfigSON=CWOBB2000I: The cell's configuration calls for the use of the overlay-based implementation of the Bulletin Board.

CWOBB2001I

BBS_ConfigHAM=CWOBB2001I: The cell's configuration calls for the use of the HAManager's implementation of the Bulletin Board.

CWOBB3000I

IMgr_Init=CWOBB3000I: The bulletin board interest aware membership service has initialized.

CWOBB3001E

IMgr_NodeLeft1=CWOBB3001E: The overlay reported the departure of an unknown process {0} with epoch {1}.

CWOBB3002E

IMgr_NodeLeft2=CWOBB3002E: The overlay reported multiple departures of process {0} with epoch {1}

CWOBB3003E

IMgr_NodeJoin=CWOBB3003E: The overlay reported the arrival of a already-present process {0} with epoch {1} (the process was already present with epoch {2}).

CWOBB3004W

IMgr_OldJoin=CWOBB3004W: The overlay reported the arrival of process {0} with epoch {1} when it was last seen with greater epoch {2}.

CWOBB9990E

bbson.unexpected.exception=CWOBB9990E: Method {0} caught exception: {1}

CWOBB9991E

bbson.exception=CWOBB9991E: Error happened:{0}

CWOBB9996E

MSG_ERR=CWOBB9996E: Something suspicious happened; the data (if any) is {0} and the exception (if any) is {1}.

CWOBB9998E

GENL_WARN=CWOBB9998E: Something suspicious happened; the data (if any) is {0} and the exception (if any) is {1}.

CWOBB9999E

GENL_ERR=CWOBB9999E: Something unexpected happened; the data (if any) is {0} and the exception (if any) is {1}.

CWOBB0001I

Cfg_Given=CWOBB0001I: The overlay-based bulletin board has been configured with a value of {1} for the parameter named {0} (the default, possibly dependent on other configuration settings, was {2}).

Explanation

This message reports on an explicit configuration setting from a cell custom property.

User response

Change or remove the setting if the current setting is not desired.

Parent topic: [CWOBB](#)

CWOBB0002E

Cfg_BadFloor=CWOBB0002E: The overlay-based bulletin board was configured with a minimum broadcast threshold of {0}, which is invalid; the bulletin board will use a minimum broadcast threshold of 1.

Explanation

The overlay-based bulletin board will disseminate an update via broadcast if the system size is at least the broadcast threshold; the minimum broadcast threshold is configured by a cell custom property named bbson.BROADCAST_FLOOR.

User response

Correct the configuration.

Parent topic: [CWOBB](#)

CWOBB0003E

Cfg_Bad1000=CWOBB0003E: The overlay-based bulletin board was configured with an invalid value ({0}) to use as the broadcast threshold when the system has 1000 processes; the threshold used in that case will be {1}.

Explanation

The overlay-based bulletin board will disseminate an update via broadcast if the system size is at least the broadcast threshold; the broadcast threshold is derived by geometrically interpolating between two given points, one of which is the broadcast threshold to use when the system contains 1000 processes. That threshold is configured by a cell custom property named `bbson.BROADCAST_THRESHOLD_FOR_1000`.

User response

Correct the configuration.

Parent topic: [CWOBB](#)

CWOBB1000I

BBF_Init=CWOBB1000I: The overlay-based bulletin board has been initialized in process {0} in core group {1} in cell {2}.

Explanation

Informational message only.

User response

No action required.

Parent topic: [CWOBB](#)

CWOBB1001I

BBF_Broke=CWOBB1001I: The overlay-based bulletin board has failed to initialize in process {0} in core group {1} in cell {2}.

Explanation

The bulletin board is not functional due to an initialization problem.

User response

See the earlier CWOBB1004E message.

Parent topic: [CWOBB](#)

CWOBB1002W

BBF_ClockJerk=CWOBB1002W: The local machine clock has jumped backwards at least {0} milliseconds: a reading of {1} ({2}) was followed, some time later, by a reading of {3} ({4}).

Explanation

A large backwards change in the clock is problematic for the Bulletin Board implementation, which will cope as best it can. The exact magnitude of the jump is unknown, and the coping will be approximate.

User response

Do not do that.

Parent topic: [CWOBB](#)

CWOBB1003E

BBF_Callback_Throw=CWOBB1003E: The Bulletin Board subscriber {3} for board {0}, scope {1}, and subject {2} threw an exception or error {4}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBB](#)

CWOBB1004E

BBF_NoPeer=CWOBB1004E: There was an unexpected exception while initializing the SON overlay; the exception is {0}.

Explanation

This should not happen. It can be caused by a problem with the configuration or a conflicting other process(es). This process will be unable to communicate in certain important ways unless and until the problem is corrected and this process is restarted.

User response

Consider the details in the exception. Correct any configuration problem, eliminate conflicting use of the configured port(s), or report to IBM.

Parent topic: [CWOBB](#)

CWOBB1005I

BBF_DUMPED=CWOBB1005I: The overlay-based bulletin board wrote its internal state, in concrete form to {0} and in abstract form to {1}.

Explanation

This is controlled by the cell custom property named bbson.DUMP_TIMEOUT; set it to a number of minutes or deleted it to have no dumps.

User response

Update the cell custom property if you want to change the behavior.

Parent topic: [CWOBB](#)

CWOBB1006E

BBF_LoopbackConnection=CWOBB1006E: The overlay-based bulletin board on process {0} got TCP connection from itself.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBB](#)

CWOBB1007I

BBF_STATS=CWOBB1007I: Cumulative bulletin board factory statistics are {0}, rates since previous stats are {1}.

Explanation

This is a periodic printout of the bulletin board factory statistics.

User response

Information only; no action required.

Parent topic: [CWOBB](#)

CWOBB1200W

Proc_SendQueueOrder=CWOBB1200W: The process queue of the bulletin board had a surprising entry (for the process named {3}, with send time {4} = {5}); this was discovered when inserting an entry for the process named {0} with send time {1} = {2}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBB](#)

CWOBB2000I

BBS_ConfigSON=CWOBB2000I: The cell's configuration calls for the use of the overlay-based implementation of the Bulletin Board.

Explanation

The alternative is the older implementation in the HAManager.

User response

If the older implementation is desired, remove the cell custom property named "WXDBulletinBoardProviderOption" or give it a value different from "WXD".

Parent topic: [CWOBB](#)

CWOBB2001I

BBS_ConfigHAM=CWOBB2001I: The cell's configuration calls for the use of the HAManager's implementation of the Bulletin Board.

Explanation

The alternative is the overlay-based implementation.

User response

If the overlay-based implementation is desired, create a cell custom property named "WXDBulletinBoardProviderOption" with a value of "WXD".

Parent topic: [CWOBB](#)

[Terms of Use](#) | [Feedback](#)

CWOBB3000I

IMgr_Init=CWOBB3000I: The bulletin board interest aware membership service has initialized.

Explanation

The service is part of the overlay-based implementation of the bulleting board and should be active in every process of a system using that implementation.

User response

Informational message only, no action required.

Parent topic: [CWOBB](#)

[Terms of Use](#) | [Feedback](#)

CWOBB3001E

IMgr_NodeLeft1=CWOBB3001E: The overlay reported the departure of an unknown process {0} with epoch {1}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBB](#)

CWOBB3002E

IMgr_NodeLeft2=CWOBB3002E: The overlay reported multiple departures of process {0} with epoch {1}

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBB](#)

CWOBB3003E

IMgr_NodeJoin=CWOBB3003E: The overlay reported the arrival of a already-present process {0} with epoch {1} (the process was already present with epoch {2}).

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBB](#)

CWOBB3004W

IMgr_OldJoin=CWOBB3004W: The overlay reported the arrival of process {0} with epoch {1} when it was last seen with greater epoch {2}.

Explanation

This should only happen under extremely rare circumstances when there are many problems with the network.

User response

Report to IBM if this message is appearing frequently or when there are not many problems in the network.

Parent topic: [CWOBB](#)

CWOBB9990E

bbson.unexpected.exception=CWOBB9990E: Method {0} caught exception: {1}

Explanation

An exception was caught in the specified method.

User response

None

Parent topic: [CWOBB](#)

[Terms of Use](#) | [Feedback](#)

CWOBB9991E

bbson.exception=CWOBB9991E: Error happened:{0}

Explanation

Unexpected Error happened.

User response

None

Parent topic: [CWOBB](#)

[Terms of Use](#) | [Feedback](#)

CWOBB9996E

MSG_ERR=CWOBB9996E: Something suspicious happened; the data (if any) is {0} and the exception (if any) is {1}.

Explanation

The overlay-based bulletin board received a network message that was malformed; this should not happen.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBB](#)

CWOBB9998E

GENL_WARN=CWOBB9998E: Something suspicious happened; the data (if any) is {0} and the exception (if any) is {1}.

Explanation

The overlay-based bulletin board encountered a condition that might or might not indicate a problem.

User response

If this happens a lot then report to IBM.

Parent topic: [CWOBB](#)

CWOBB9999E

GENL_ERR=CWOBB9999E: Something unexpected happened; the data (if any) is {0} and the exception (if any) is {1}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBB](#)

CWOBJ

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWOBJ0001E](#)

ILLEGAL_STATE_EXCEPTION_CWOBJ0001=CWOBJ0001E: Method, {0}, was called after initialization completed.

[CWOBJ0002W](#)

IGNORING_UNEXPECTED_EXCEPTION_CWOBJ0002=CWOBJ0002W: ObjectGrid component is ignoring an unexpected exception: {0}.

[CWOBJ0005W](#)

INTERRUPTED_EXCEPTION_CWOBJ0005=CWOBJ0005W: The thread created an InterruptedException: {0}

[CWOBJ0006W](#)

GENERAL_EXCEPTION_WARNING_CWOBJ0006=CWOBJ0006W: An exception occurred: {0}

[CWOBJ0007W](#)

NULL_VALUE_WARNING_CWOBJ0007=CWOBJ0007W: The value null was specified for {0}, a default value of {1} is used.

[CWOBJ0008E](#)

INVALID_VALUE_ERROR_CWOBJ0008=CWOBJ0008E: The value {0} provided for the property {1} is invalid.

[CWOBJ0010E](#)

MISSING_KEY_ERROR_CWOBJ0010=CWOBJ0010E: Message key {0} is missing.

[CWOBJ0011W](#)

DESERIALIZATION_FIELD_NOT_FOUND_CWOBJ0011=CWOBJ0011W: Cannot deserialize field {0} in class {1}; using the default value.

[CWOBJ0012E](#)

INVALID_LOGELEMENT_TYPE_CWOBJ0012=CWOBJ0012E: The LogElement type code, {0} ({1}), is not recognized for this operation.

[CWOBJ0013E](#)

EVICT_ENTRIES_EXCEPTION_CWOBJ0013=CWOBJ0013E: An exception occurred while attempting to evict entries from the cache: {0}

[CWOBJ0014E](#)

NESTED_TRANSACTION_CWOBJ0014=CWOBJ0014E: The ObjectGrid runtime detected an attempt to nest transactions.

[CWOBJ0015E](#)

TRANSACTION_EXCEPTION_CWOBJ0015=CWOBJ0015E: An exception occurred while attempting to process a transaction: {0}

[CWOBJ0016E](#)

NO_ACTIVE_TRANSACTION_CWOBJ0016=CWOBJ0016E: No active transaction is detected for the current operation.

[CWOBJ0017E](#)

DUPLICATE_KEY_EXCEPTION_CWOBJ0017=CWOBJ0017E: A duplicate key exception was detected during the processing of the ObjectMap operation: {0}

[CWOBJ0018E](#)

KEY_NOT_FOUND_EXCEPTION_CWOBJ0018=CWOBJ0018E: The key was not found during the processing of the ObjectMap operation: {0}

[CWOBJ0019W](#)

MISSING_CACHE_ENTRY_SLOT_DATA_CWOBJ0019=CWOBJ0019W: Did not find data in the cache entry slot reserved for {0} to use for ObjectMap name {1}.

[CWOBJ0020E](#)

CACHE_ENTRY_NOT_IN_MAP_CWOBJ0020=CWOBJ0020E: Cache entry is not in BackingMap {0}.

[CWOBJ0021E](#)

OBJECT_TRANSFORMER_NOT_FOUND_CWOBJ0021=CWOBJ0021E: A usable ObjectTransformer instance was not found during the deserialization of the LogSequence object for {0} ObjectGrid and {1} ObjectMap.

[CWOBJ0022E](#)

LOCK_MANAGER_INTERNAL_ERROR_CWOBJ0022=CWOBJ0022E: The caller does not own mutex: {0}.

CWOBJ0023E

UNRECOGNIZED_COPY_MODE_CWOBJ0023=CWOBJ0023E: The CopyMode ({0}) is not recognized for this operation.

CWOBJ0024E

REQUIRED_FIELD_NOT_FOUND_CWOBJ0024=CWOBJ0024E: Cannot deserialize field {0} in class {1}. Deserialization failed.

CWOBJ0025E

SERIALIZATION_FAILED_CWOBJ0025=CWOBJ0025E: The serialization of the LogSequence object failed. The number of serialized LogElement objects ({0}) does not match the number of read LogElement objects ({1}).

CWOBJ0026E

INVALID_JMX_CREDENTIAL_CWOBJ0026=CWOBJ0026E: The JMX credential type is not right. It should be of type {0}.

CWOBJ0027E

CLONE_METHOD_NOT_SUPPORTED_CWOBJ0027=CWOBJ0027E: Internal runtime error. Clone method not supported: {0}

CWOBJ0028E

KEY_NOT_FOUND_EVICTOR_CWOBJ0028=CWOBJ0028E: An error occurred in {0} for the map {1}. The key, {2}, was not found in the map. LogElement type is {3}.

CWOBJ0029E

MISSING_EVICTOR_DATA_CWOBJ0029=CWOBJ0029E: An error occurred in {0} for the map {1}. CacheEntry is missing a {2} object for key {3}. LogElement type is {4}.

CWOBJ0030I

OBJECTGRID_INSTRUMENTATION_ENABLED_CWOBJ0030=CWOBJ0030I: ObjectGrid entity class instrumentation is enabled. The instrumentation mode is {0}.

CWOBJ0031W

OPTIMISTIC_LOCKING_WITHOUT_CALLBACK_CWOBJ0031=CWOBJ0031W: Optimistic locking is enabled for map {0}, but a custom OptimisticCallback is not provided.

CWOBJ0032W

BUCKET_SIZE_MAY_BE_TOO_SMALL_CWOBJ0032=CWOBJ0032W: The number of buckets for map {0} may be too small. The current number of buckets is {1}. The current number of entries in the map is {2}. The following condition was observed: {3}.

CWOBJ0033I

CLASS_NOT_IMPLEMENT_CLONE_CWOBJ0033=CWOBJ0033I: Class, {0}, does not implement the clone() method. Using serialization instead for this Class in map {1}

CWOBJ0034I

TARGET_MEMORY_UTILIZATION_THRESHOLD_LEVEL_CWOBJ0034=CWOBJ0034I: Memory utilization threshold percentage is set to {0} %.

CWOBJ0035W

MEMORY_UTILIZATION_THRESHOLD_NOT_SUPPORTED_CWOBJ0035=CWOBJ0035W: Memory utilization threshold not supported for this JVM.

CWOBJ0036W

CHANGING_MEMORY_UTILIZATION_THRESHOLD_CWOBJ0036=CWOBJ0036W: Changing memory utilization threshold from {0} to {1} for {2} memory pool.

CWOBJ0037W

CHANGING_MEMORY_COLLECTION_UTILIZATION_THRESHOLD_CWOBJ0037=CWOBJ0037W: Changing memory collection utilization threshold from {0} to {1} for {2} memory pool.

CWOBJ0038W

MEMORY_THRESHOLD_EXCEEDED_CWOBJ0038=CWOBJ0038W: Memory threshold exceeded. Current heap memory usage: {0}.

CWOBJ0039W

MEMORY_COLLECTION_THRESHOLD_EXCEEDED_CWOBJ0039=CWOBJ0039W: Memory collection threshold exceeded. Current heap memory usage: {0}.

CWOBJ0040E

CWOBJ0040=CWOBJ0040E: Hash based data structure over run for {0} with {1} elements in the data structure. Examine the hashCode method on this class for better distribution.

CWOBJ0900I

CWOBJ0900=CWOBJ0900I: The ObjectGrid runtime component is started for server {0}.

CWOBJ0901E

CWOBJ0901=CWOBJ0901E: "{0}" system property is required to start ObjectGrid component for server {1}.

CWOBJ0902W

CWOBJ0902=CWOBJ0902W: Error prevented the ObjectGrid runtime component from starting for server {0}.

CWOBJ0910I

CWOBJ0910=CWOBJ0910I: The ObjectGrid runtime component is stopped for server {0}.

CWOBJ0911I

CWOBJ0911=CWOBJ0911I: Starting the ObjectGrid runtime component for server {0}.

CWOBJ0912E

CWOBJ0912=CWOBJ0912E: The application {0} has ObjectGrid configuration files that will not be used because application {1} is currently running an ObjectGrid server instance.

CWOBJ1001I

OPEN_FOR_BUSINESS_CWOBJ1001=CWOBJ1001I: ObjectGrid Server {0} is ready to process requests.

CWOBJ1002E

PORT_CONFLICT_CWOBJ1002=CWOBJ1002E: Server port {0} is already in use.

CWOBJ1003I

DCS_CWOBJ1003=CWOBJ1003I: DCS Adapter service is disabled by configuration, to enable it, please change your configuration with an endpoint defined.

CWOBJ1004E

SERVER_TOPIC_CWOBJ1004=CWOBJ1004E: Server topic is null

CWOBJ1005E

CLIENT_REQUESTQ_CWOBJ1005=CWOBJ1005E: The incoming request queue is null.

CWOBJ1006E

CLIENT_RESULTQ_CWOBJ1006=CWOBJ1006E: The outgoing result queue is null.

CWOBJ1007E

CLIENT_REQUEST_CWOBJ1007=CWOBJ1007E: ObjectGrid client request is null.

CWOBJ1008E

TXID_CWOBJ1008=CWOBJ1008E: ObjectGrid client request TxID is null.

CWOBJ1009E

SERVER_CWOBJ1009=CWOBJ1009E: ObjectGrid client received a null response from the server.

CWOBJ1010I

CLUSTER_SHUTDOWN_CWOBJ1010=CWOBJ1010I: Shutdown request is processing.

CWOBJ1011I

CLUSTER_SHUTDOWN_CWOBJ1011=CWOBJ1011I: Shutdown request is sending.

CWOBJ1012I

CLUSTER_SHUTDOWN_CWOBJ1012=CWOBJ1012I: Shutdown request is performed.

CWOBJ1013W

EXCEPTION_ON_SERVER_CWOBJ1013=CWOBJ1013W: An exception occurred on a remote server: {0}

CWOBJ1014I

CLASSPATH_PROBLEM_CWOBJ1014=CWOBJ1014I: Preceding {0} message may be caused by application classes missing from the classpath on the server.

CWOBJ1015I

OBJECTTRANSFORMER_PROBLEM_CWOBJ1015=CWOBJ1015I: Preceding {0} message may be caused by an incorrect application implementation of the ObjectTransformer or Serializable interface

CWOBJ1016E

PROPERTY_FILE_DOES_NOT_EXIST_CWOBJ1016E=CWOBJ1016E: The property file {0} does not exist: {1}.

CWOBJ1110I

DCS_CWOBJ1110=CWOBJ1110I: Starting the transport for ObjectGrid cluster {0} using IP Address {1}, port {2}, transport type {3}.

CWOBJ1111W

DCS_CWOBJ1111=CWOBJ1111W: Resolution of IP Addresses for host name {0} found only the loopback address. The loopback address will be used.

CWOBJ1112E

DCS_CWOBJ1112=CWOBJ1112E: An error was encountered while looking up the IP address for the host name of an ObjectGrid cluster member. The host name is {0} and the server name is {1}. The member will be excluded from the cluster.

CWOBJ1113E

DCS_CWOBJ1113=CWOBJ1113E: ObjectGrid cluster transport service on this process is not started. This cluster member is not defined in the configuration.

CWOBJ1114E

DCS_CWOBJ1114=CWOBJ1114E: ObjectGrid cluster transport service on this process could not process the incoming message. The message is {0} and the exception is {1}.

CWOBJ1115E

DCS_CWOBJ1115=CWOBJ1115E: An unrecognized view change event was received from the ObjectGrid cluster transport. The view identifier is {0} and the event is {1}.

CWOBJ1116E

DCS_CWOBJ1116=CWOBJ1116E: An attempt by another process to connect to this process via the ObjectGrid cluster transport has been rejected. The connecting process provided a name of {0}, a target of {1}, a member name of {2} and an IP address of {3}. The error message is {4}.

CWOBJ1117E

DCS_CWOBJ1117=CWOBJ1117E: An attempt to authenticate a connection has failed. The exception is {0}.

CWOBJ1118I

DCS_CWOBJ1118=CWOBJ1118I: ObjectGrid Server Initializing [Cluster: {0} Server: {1}].

CWOBJ1119I

CLIENT_CWOBJ1119=CWOBJ1119I: ObjectGrid client failed to connect to host: {0} port: {1}.

CWOBJ1120I

CLIENT_CWOBJ1120=CWOBJ1120I: ObjectGrid Client connected successfully to host: {0} port: {1}.

CWOBJ1201E

NO_CLIENT_ENDPOINT_DEFINED_CWOBJ1201E=CWOBJ1201E: No valid client access end points are defined.

CWOBJ1202E

SSL_SOCKET_FAILED_CWOBJ1202E=CWOBJ1202E: SSL Server Socket failed to initialize. The exception message is {0}

CWOBJ1203W

CLIENT_RESPONSE_TIMEOUT_CWOBJ1203W=CWOBJ1203W: Received a timeout event from the server for transaction: {0}

CWOBJ1204W

UNKNOWN_MESSAGE_TYPE_CWOBJ1204W=CWOBJ1204W: Received a message of unknown message type. The message is: {0}

CWOBJ1205E

SSL_INITIALIZATION_ERROR_CWOBJ1205E=CWOBJ1205E: SSL Initialization failed. The exception message is {0}

CWOBJ1206W

SSL_INITIALIZATION_WARNING_CWOBJ1206W=CWOBJ1206W: SSL Initialization failed. The exception message is {0}

CWOBJ1207W

CONFIG_PROPERTY_UNSUPPORTED_CWOBJ1207W=CWOBJ1207W: The property {0} on plug-in {1} is using an unsupported property type.

CWOBJ1208W

CONFIG_PLUGIN_UNSUPPORTED_CWOBJ1208W=CWOBJ1208W: The specified plug-in type, {0}, is not one of the supported plug-in types.

CWOBJ1211E

ERROR_OG_PMI_CREATE_FAILED_CWOBJ1211E=CWOBJ1211E: The Performance Monitoring Infrastructure (PMI) creation of {0} failed. The exception is {1}.

CWOBJ1212I

PMI_NOT_FOUND=CWOBJ1212I: The WebSphere Application Server Performance Monitoring Infrastructure (PMI) cannot be found.

CWOBJ1215I

TP_CWOBJ1215=CWOBJ1215I: ObjectGrid Transaction Propagation Event Listener is initializing [ObjectGrid {0}].

CWOBJ1216I

TP_CWOBJ1216=CWOBJ1216I: ObjectGrid Transaction Propagation Event Listener is initialized [ObjectGrid {0}].

CWOBJ1217I

TP_CWOBJ1217=CWOBJ1217I: ObjectGrid Transaction Propagation Service Point Initialized [ObjectGrid {0}].

CWOBJ1218E

TP_CWOBJ1218=CWOBJ1218E: ObjectGrid Transaction Propagation Event Listener failure occurred [ObjectGrid {0} Exception message {1}].

CWOBJ1219E

TP_CWOBJ1219=CWOBJ1219E: ObjectGrid Transaction Propagation Service End Point failure occurred [ObjectGrid {0} Exception message {1}].

CWOBJ1220E

TRANPROPLISTENER_UNSUPPORTED_CWOBJ1220=CWOBJ1220E: ObjectGrid Transaction Propagation Service is not supported in this environment.

CWOBJ1300I

PMA_CWOBJ1300=CWOBJ1300I: Adapter successfully initialized ObjectGrid.

CWOBJ1301E

PMA_CWOBJ1301=CWOBJ1301E: Adapter failed to initialize ObjectGrid. Exception occurred [Exception message {0}].

CWOBJ1302I

PMA_CWOBJ1302=CWOBJ1302I: Adapter stopped.

CWOBJ1303I

PMA_CWOBJ1303=CWOBJ1303I: Adapter started.

CWOBJ1304I

SECURITY_ENABLED_CWOBJ1304=CWOBJ1304I: ObjectGrid security is enabled.

CWOBJ1305I

SECURITY_DISABLED_CWOBJ1305=CWOBJ1305I: ObjectGrid security is disabled.

CWOBJ1306W

CANNOT_RETRIEVE_CLIENT_CERTS_CWOBJ1306=CWOBJ1306W: Cannot retrieve the client certificates from the SSL socket.

CWOBJ1307I

OBJECTGRID_SECURITY_ENABLED_CWOBJ1307=CWOBJ1307I: Security of the ObjectGrid instance {0} is enabled.

CWOBJ1308I

OBJECTGRID_SECURITY_DISABLED_CWOBJ1308=CWOBJ1308I: Security of the ObjectGrid instance {0} is disabled.

CWOBJ1309E

OBJECTGRID_CONNECT_TOKEN_CREATION_CWOBJ1309=CWOBJ1309E: Unexpected error occurred in the connect token creation: {0}.

CWOBJ1310E

OBJECTGRID_CONNECT_TOKEN_VALIDATION_CWOBJ1310=CWOBJ1310E: An attempt by another process to connect to this process via the core group transport has been rejected. The connecting process provided a source core group name of {0}, a target of {1}, a member name of {2} and an IP address of {3}. The error message is {4}.

CWOBJ1311W

IGNORE_CREDENTIAL_GENERATOR_PROPS=CWOBJ1311W: The credentialGeneratorProps setting is ignored since the credentialGeneratorClass value is not provided.

CWOBJ1312W

EXPIRED_CREDENTIAL_EXCEPTION=CWOBJ1312W: The credential has been expired. The exception message is {0}.

CWOBJ1313W

CUSTOM_SECURE_TOKEN_MANAGER_CLASS_IGNORED=CWOBJ1313W: The customSecureTokenManagerClass setting is ignored since the provided customSecureTokenManagerType value is not "custom".

CWOBJ1400W

MULTIPLE_JAR_FILE_CWOBJ1400W=CWOBJ1400W: Detected multiple ObjectGrid runtime JARS files in the JVM. Using multiple ObjectGrid runtime JAR files may cause problems.

CWOBJ1401E

WRONG_JAR_FILE_CWOBJ1401E=CWOBJ1401E: Detected a wrong ObjectGrid runtime JAR file for this configuration. Detected configuration is {0}. Expected jar file is {1}.

CWOBJ1402E

MISSING_CONNECTION_LINK_CALLBACK_CWOBJ1402E=CWOBJ1402E: ObjectGrid connection link callback not found for id: {0}.

CWOBJ1500E

CANNOT_CREATE_GROUPNAME_CWOBJ1500=CWOBJ1500E: An exception occurred when attempting to create a GroupName for HA Group ({0}): {1}.

CWOBJ1501E

CANNOT_JOIN_GROUP_CWOBJ1501=CWOBJ1501E: An exception occurred when member ({0}) attempted to join HA Group ({1}): {2}.

CWOBJ1503E

CANNOT_GET_OBJECTGRID_CWOBJ1503=CWOBJ1503E: Cannot access ObjectGrid ({0}) for applying updates to replica member ({1}).

CWOBJ1504E

CANNOT_PROCESS_REPLICA_CHANGES_CWOBJ1504=CWOBJ1504E: An exception occurred when attempting to process the LogSequences for replica ({0}): {1}.

CWOBJ1505E

MORE_THAN_ONE_PRIMARY_RESPONSE_CWOBJ1505=CWOBJ1505E: More than one replication group member reported back as the primary. Only one primary can be active. ({0}).

CWOBJ1506E

POSSIBLE_NETWORK_PARTITION_CWOBJ1506=CWOBJ1506E: More than one primary replication group member exists in this group ({1}). Only one primary can be active. ({0}).

CWOBJ1507W

CANNOT_CLOSE_CHECKPOINT_CWOBJ1507=CWOBJ1507W: An exception occurred when attempting to end the replication process for BackingMap ({0}): {1}.

CWOBJ1508E

CANNOT_SEND_MESSAGE_CWOBJ1508=CWOBJ1508E: An exception occurred when attempting to send message ({0}) from sender ({1}) to receiver ({2}): {3}.

CWOBJ1509E

CANNOT_SERIALIZE_MESSAGE_CWOBJ1509=CWOBJ1509E: An exception occurred when attempting to serialize message ({0}): {1}.

CWOBJ1510E

CANNOT_DESERIALIZE_MESSAGE_CWOBJ1510=CWOBJ1510E: An exception occurred when attempting to inflate message ({0}): {1}.

CWOBJ1511I

OPEN_FOR_BUSINESS_CWOBJ1511=CWOBJ1511I: {0} ({1}) is open for business.

CWOBJ1512W

GROUP_MEMBER_ALREADY_EXISTS_CWOBJ1512=CWOBJ1512W: {0} already exists in replication group {1}.

CWOBJ1513E

SYNCH_REPLICATION_FAILED_CWOBJ1513=CWOBJ1513E: Synchronous replication failed on {0} ({1}). This member is no longer active.

CWOBJ1514I

PRIMARY_DOWNGRADED_CWOBJ1514=CWOBJ1514I: Primary ({0}) is being downgraded to either a replica or standby.

CWOBJ1515I

MIN_CONFIG_NOT_MET_CWOBJ1515=CWOBJ1515I: Minimum configuration requirements not satisfied for replication group ({0}).

CWOBJ1516E

CANNOT_ACTIVATE_OBJECTGRID_CWOBJ1516=CWOBJ1516E: An exception occurred when attempting to activate the replication process for ObjectGrid ({0}): {1}.

CWOBJ1517E

SYNCH_REPLICATION_TRAN_FAILED_CWOBJ1517=CWOBJ1517E: Synchronous replication failed for transaction {2} on {0} ({1}). This member is no longer active.

CWOBJ1518E

CANNOT_COMMIT_REPLICA_CHANGES_CWOBJ1518=CWOBJ1518E: An exception occurred when attempting to commit replica transaction ({0}) for primary transaction ({1}) on Replica ({2}): {3}.

CWOBJ1519E

CANNOT_ROLLBACK_REPLICA_CHANGES_CWOBJ1519=CWOBJ1519E: An exception occurred when attempting to rollback the LogSequences for replica ({0}): {1}.

CWOBJ1520I

TRAN_LAG_THRESHOLD_MET_LOST_CWOBJ1520=CWOBJ1520I: The transaction lag threshold, {0}, was met. Current lag: {1}. Transaction {2} may be lost.

CWOBJ1521I

TRAN_LAG_THRESHOLD_MET_SLOW_CWOBJ1521=CWOBJ1521I: The transaction lag threshold, {0}, was met. Current lag: {1}.

CWOBJ1522I

TRAN_WARNING_MET_CWOBJ1522=CWOBJ1522I: The transaction lag warning, {0}, was met. Current lag: {1}. Beginning check for lost transaction. Possible lost transaction is {2}.

CWOBJ1523I

DUP_TRANSACTION_CWOBJ1523=CWOBJ1523I: Received a duplicate transaction, {0}. Transaction will be

thrown away.

CWOBJ1524I

LISTENER_REREGISTER_CWOBJ1524=CWOBJ1524I: Replica listener {0} must reregister with the primary.

CWOBJ1525I

CHECKPRELOADSTATE_EXCEPTION_CWOBJ1525=CWOBJ1525I: A ReplicaPreloadController ({0}) for map {1} threw an unexpected exception in method checkPreloadState {2}

CWOBJ1526I

ENTERING_PEER_MODE_CWOBJ1526=CWOBJ1526I: Replica {0} entering peer mode after {1} seconds

CWOBJ1527W

FAILED_ENTERING_PEER_MODE_CWOBJ1527=CWOBJ1527W: Replica {0} failed to enter peer mode after {1} seconds

CWOBJ1528E

FAILED_ENTERING_PEER_MODE_TIMEOUT_CWOBJ1528=CWOBJ1528E: Replica {0} failed to enter peer mode: Waiting for data copy from the primary to complete timed out. Current timeout (ms): {1}

CWOBJ1529E

FAILED_ENTERING_PEER_MODE_COPY_CWOBJ1529=CWOBJ1529E: Replica {0} failed to enter peer mode: A transaction threw an error while copying data from the primary.

CWOBJ1530E

FAILED_ENTERING_PEER_MODE_ORDER_CWOBJ1530=CWOBJ1530E: Replica {0} failed to enter peer mode: Received incorrect ordering data from the primary, data copy cannot complete.

CWOBJ1600I

GATEWAY_STARTED_CWOBJ1600=CWOBJ1600I: ManagementGateway service started on port ({0}).

CWOBJ1601E

GATEWAY_SERVICE_FAILED_CWOBJ1601=CWOBJ1601E: ManagementGateway service failed to start on port ({0}).

CWOBJ1602E

GATEWAY_CLIENT_CONNECT_FAILED_CWOBJ1602=CWOBJ1602E: ManagementGateway service failed to connect to server at ({0}):({1}).

CWOBJ1603E

MANAGEMENT_SERVICE_RESPONSE_FAILED_CWOBJ1603=CWOBJ1603E: Management service failed to respond to ({0}) remote request: {1}.

CWOBJ1604I

MANAGEMENT_GATEWAY_STOP_FAILED_CWOBJ1604=CWOBJ1604I: ManagementGateway service failed to stop connector due to Throwable {0}. Exiting.

CWOBJ1605I

MANAGEMENT_GATEWAY_REFRESH_FAILED_CWOBJ1605=CWOBJ1605I: ManagementGateway caught Throwable {0} while refreshing attributes. Exiting.

CWOBJ1606I

NO_RESPONSE_FROM_SERVER_CWOBJ1606=CWOBJ1606I: {0} - Unable to get response from server {1}. Returning false.

CWOBJ1607I

USE_WSADMIN_CWOBJ1607=CWOBJ1607I: {0} - When an ObjectGrid server is colocated with a WebSphere Application Server, use WSADMIN to stop server {1}. Returning false.

CWOBJ1608I

SERVER_NOT_RESPONDING_NULL_CWOBJ1608=CWOBJ1608I: {0} - Unable to get response from server {1}. Ensure server is running. Returning null.

CWOBJ1609I

NO_ROUTING_TABLE_CWOBJ1609=CWOBJ1609I: {0} - Unable to get routing table. Please wait several seconds and retry operation. Returning null.

CWOBJ1610W

RESET_NULL_CLUSTER_CWOBJ1610=CWOBJ1610W: Try to reset a null cluster for {0}.

CWOBJ1611I

OPEN_CLUSTER_CWOBJ1611=CWOBJ1611I: Replication group cluster {0} is open for business.

CWOBJ1612I

CLOSE_CLUSTER_CWOBJ1612=CWOBJ1612I: Replication group cluster {0} is closed for business.

CWOBJ1615E

CREDENTIAL_EXPIRED_CWOBJ1615=CWOBJ1615E: The JMX credential has expired: {0}.

CWOBJ1616I

JMX_SECURITY_NOT_FOUND_CWOBJ1616=CWOBJ1616I: JMX Security implementation not found.

CWOBJ1620I

REPLACE_SERVER_CWOBJ1620=CWOBJ1620I: Replacing target for wrongly routed request due to changes in the server. The new target is {0}.

CWOBJ1630I

DOMINO_MODE_CWOBJ1630=CWOBJ1630I: Replication group cannot serve this request {0}.

CWOBJ1632E

NULL_ID_CWOBJ1632=CWOBJ1632E: Original request does not have a valid ID; no way to forward this request.

CWOBJ1634I

BLIND_FORWARD_CWOBJ1634=CWOBJ1634I: Router cannot find the forwarding target; using blind forwarding.

CWOBJ1660I

SERVER_NOT_RIGHT_CWOBJ1660=CWOBJ1660I: Replication group member has changed. This server does not host what is requested anymore. The original request is {0}.

CWOBJ1661I

SERVER_CLUSTER_DATA_CWOBJ1661=CWOBJ1661I: Cluster data are updated for replication group: {0}

CWOBJ1663E

VERIFY_NULL_CLUSTER_CWOBJ1663=CWOBJ1663E: Server router cannot verify server routing for {0}, because cluster data for this replication group are null in the server.

CWOBJ1668W

NOT_STARTED_CWOBJ1668=CWOBJ1668W: Request is coming to the server that has not completely started.

CWOBJ1680W

PER_SERVER_CWOBJ1680=CWOBJ1680W: The configured TCP connection timeout is smaller than $\text{retryInterval} * \max(\text{startupRetries}, \text{maxRetries})$, so there is possibility that connection will time out.

CWOBJ1682W

TOTAL_CWOBJ1682=CWOBJ1682W: The configured transaction timeout is smaller than $\text{maxForwards} * \text{retryInterval} * \max(\text{startupRetries}, \text{maxRetries})$, so there is possibility that transaction will time out.

CWOBJ1700I

STANDLAONE_HAMANAGER_INITIALIZED_CWOBJ1700=CWOBJ1700I: Standalone HAManager is initialized with coregroup {0}.

CWOBJ1702E

STANDLAONE_HAMANAGER_NOT_INITIALIZED_CWOBJ1702=CWOBJ1702E: Standalone HAManager is not initialized, so it cannot be started.

CWOBJ1710I

STANDLAONE_HAMANAGER_STARTED_CWOBJ1710=CWOBJ1710I: Standalone HAManager is started successfully.

CWOBJ1711I

STANDLAONE_HAMANAGER_ALREADY_STARTED_CWOBJ1711=CWOBJ1711I: Standalone HAManager is already started successfully.

CWOBJ1712E

STANDLAONE_HAMANAGER_NOT_STARTED_CWOBJ1712=CWOBJ1712E: Standalone HAManager is not started.

CWOBJ1713E

STANDLAONE_HAMANAGER_START_FAIL_CWOBJ1713=CWOBJ1713E: Standalone HAManager failed to start.

CWOBJ1720I

HAMANAGER_CONTROLLER_CWOBJ1720=CWOBJ1720I: HAManager Controller detected that ObjectGrid server is in the WebSphere environment, using WebSphere HAManager instead of initializing and starting standalone HAManager.

CWOBJ1730I

HAMANAGER_CONTROLLER_EXTERNALHAM_NULL_CWOBJ1730=CWOBJ1730I: HAManager Controller detected that the WebSphere external HAManager is null.

CWOBJ1760I

HEARTBEAT_SLIDEBAR_SET_CWOBJ1760=CWOBJ1760I: Heartbeat frequency level is set to {0}.

CWOBJ1761E

HEARTBEAT_SLIDEBAR_SET_CWOBJ1761=CWOBJ1761E: Heartbeat frequency level {0} is beyond the allowable range.

CWOBJ1762I

HEARTBEAT_SLIDEBAR_SET_CWOBJ1762=CWOBJ1762I: The custom details of heart beating parameters is set to {0}.

CWOBJ1763I

HEARTBEAT_SLIDEBAR_SET_CWOBJ1763=CWOBJ1763I: The heartbeating parameters are {0}.

CWOBJ1764I

LEADERMANAGER_SLIDEBAR_SET_CWOBJ1764=CWOBJ1764I: The leader manager heartbeating interval is {0}.

CWOBJ1765I

LEADERMANAGER_SLIDEBAR_SET_CWOBJ1765=CWOBJ1765I: The leader manager heartbeating timeout is {0}.

CWOBJ1766I

LEADERMANAGER_SLIDEBAR_SET_CWOBJ1766=CWOBJ1766I: The number of leader manager heartbeating threads is {0}.

CWOBJ1767I

DCS_SLIDEBAR_SET_CWOBJ1767=CWOBJ1767I: The DCS heartbeating interval is {0}.

CWOBJ1768I

DCS_SLIDEBAR_SET_CWOBJ1768=CWOBJ1768I: The DCS heartbeating timeout is {0}.

CWOBJ1769I

DCS_SLIDEBAR_SET_CWOBJ1769=CWOBJ1769I: The number of DCS heartbeating threads is {0}.

CWOBJ1790I

HAMANAGER_CONTROLLER_NEED_STANDALONE_HAM_CWOBJ1790=CWOBJ1790I: Need to initialize and start the standalone HAManager.

CWOBJ1792I

TM_CWOBJ1792=CWOBJ1792I: The maximum number of threads is {0} and the minimum number of threads is {1}.

CWOBJ1800I

CLIENT_RECEIVED_RESPONSE_HAVING_FORWARD_REQUIREMENT_CWOBJ1800=CWOBJ1800I: Forwarding is required for request {0} with response of {1}.

CWOBJ1810I

CLIENT_FORWARDING_CWOBJ1810=CWOBJ1810I: Forwarding is required for response {0}.

CWOBJ1811E

FORWARDING_NOT_FOUND_REQUEST_CWOBJ1811=CWOBJ1811E: Forwarding is required, but the original request cannot be found.

CWOBJ1820E

FORWARD_NULL_RGID_CWOBJ1820=CWOBJ1820E: Forwarding request does not have a replication group identifier.

CWOBJ1870I

CLIENT_DOMINO_CWOBJ1870=CWOBJ1870I: Server service is not available for response {0}.

CWOBJ1871E

NULL_DOMINO_CWOBJ1871=CWOBJ1871E: Detected unavailable service, received null response, no way to retry.

CWOBJ1872I

CLIENT_DOMINO_TIMEOUT_CWOBJ1872=CWOBJ1872I: Service is unavailable with response of {0}.

CWOBJ1890I

DEAD_SERVER_REROUTING_CWOBJ1890=CWOBJ1890I: Re-routing request {0} due to an un-responsive server.

CWOBJ1891E

NO_SERVER_REROUTING_CWOBJ1891=CWOBJ1891E: All servers are not available in replication group {0}.

CWOBJ1898W

FORWARD_NULL_TARGET_CWOBJ1898=CWOBJ1898W: Forwarding is required, but router cannot find new available target for response {0}

CWOBJ1899W

FORWARD_NULL_RGID_CWOBJ1899=CWOBJ1899W: Forwarding is required, but router cannot find right replication group for response {0}

CWOBJ1900I

RPC_SERVICE_INIT_CWOBJ1900=CWOBJ1900I: Client server remote procedure call service is initialized.

CWOBJ1901I

RPC_SERVICE_START_CWOBJ1901=CWOBJ1901I: Client server remote procedure call service is started.

CWOBJ1902I

RPC_HANDLER_THREADS_START_CWOBJ1902=CWOBJ1902I: Client server remote procedure call handler threads are started.

CWOBJ1903I

CONFIG_NETWORK_SERVICE_INIT_CWOBJ1903=CWOBJ1903I: Configuration network service is initialized.

CWOBJ1904I

CONFIG_NETWORK_SERVICE_START_CWOBJ1904=CWOBJ1904I: Configuration network service is started.

CWOBJ1905I

CONFIG_NETWORK_HANDLER_START_CWOBJ1905=CWOBJ1905I: Configuration handler is started.

CWOBJ1913I

SYSADMIN_NETWORK_SERVICE_INIT_CWOBJ1913=CWOBJ1913I: System administration network service is initialized.

CWOBJ1914I

SYSADMIN_NETWORK_SERVICE_START_CWOBJ1914=CWOBJ1914I: System administration network service is started.

CWOBJ1915I

SYSADMIN_NETWORK_HANDLER_START_CWOBJ1915=CWOBJ1915I: System administration handler is started.

CWOBJ1921W

Cannot_Find_host_name=CWOBJ1921W: Cannot find host name

CWOBJ1922E

Cannot_Lookup_IP=CWOBJ1922E: Cannot lookup IP for this host ({0}).

CWOBJ1927E

HostNotMatchReal=CWOBJ1927E: Configured host IP {0} host name {1} does not match the host IP of this machine {2} host name {3}, so ObjectGrid cannot be started.

CWOBJ1929W

LocalHostUsed=CWOBJ1929W: LOCALHOST is used in the configuration that may lose server identity in multiple machine environment

CWOBJ1931I

ServerSupport=CWOBJ1931I: The configuration for {0} does not support either an ObjectGrid replication group member or a client/server transaction processor. This server will provide bootstrap support to peer ObjectGrid servers and clients only.

CWOBJ1932I

ThreadPoolMinMax=CWOBJ1932I: Client thread pool minimum size is {0} maximum size {1}.

CWOBJ1993I

HEARTBEAT_SERVICE_INIT_CWOBJ1993=CWOBJ1993I: Health monitor heart beat service is initialized.

CWOBJ1994I

HEARTBEAT_SERVICE_START_CWOBJ1994=CWOBJ1994I: Health monitor heart beat service is started.

CWOBJ1995I

HEARTBEAT_HANDLER_START_CWOBJ1995=CWOBJ1995I: Health monitor heart beat handler is started.

CWOBJ2000E

NO_RGM_CWOBJ2000=CWOBJ2000E: No member in this replication group {0}.

CWOBJ2001W

NO_AVAILABLE_RGM_CWOBJ2001=CWOBJ2001W: No available member in this replication group {0}.

CWOBJ2002W

NO_AVAILABLE_RT_CWOBJ2002=CWOBJ2002W: No available routing table for this replication group {0}.

CWOBJ2003I

RT_CACHE_CWOBJ2003=CWOBJ2003I: Cannot find routing cache for cache key {0}, creating new routing cache.

CWOBJ2010E

NULL_TARGET_CWOBJ2010=CWOBJ2010E: Target for this request is null.

CWOBJ2020I

ClientProperty_CWOBJ2020=CWOBJ2020I: Client properties are {0}.

CWOBJ2021I

PreferLocalJVM_CWOBJ2021=CWOBJ2021I: Prefer local JVM is enabled.

CWOBJ2022I

PreferLocalHost_CWOBJ2022=CWOBJ2022I: Prefer local host routing is enabled.

CWOBJ2023I

PreferZones_CWOBJ2023=CWOBJ2023I: Preferred zone routing is enabled for zones: {0}.

CWOBJ2060I

NEW_RT_CHANGE_CWOB2060=CWOB2060I: Client received new version of replication group cluster {0}.

CWOB2068I

REACHABILITY_CWOB2068=CWOB2068I: Reachability control detected problem in replication group member {0}.

CWOB2069I

REACHABILITY_TIMER_CWOB2069=CWOB2069I: Reachability control timer releases replication group member {0}.

CWOB2086I

ROUTING_THREAD_CONTROL_CWOB2086=CWOB2086I: Routing thread control is activated due to overload for replication group {0}.

CWOB2088I

REACHABILITY_CWOB2088=CWOB2088I: Reachability control is activated to regulate the server availability for replication group {0}.

CWOB2090W

NULL_RT_CWOB2090=CWOB2090W: Cannot find routing table for replication group {0}.

CWOB2091W

EMPTY_RT_CWOB2091=CWOB2091W: Routing table is not null, but it does not contain any servers for replication group {0}.

CWOB2092I

NULL_RUNTIME_CWOB2092=CWOB2092I: Routing table is null in runtime for replication group {0}.

CWOB2093I

NULL_RGCLUSTERSTORE_CWOB2093=CWOB2093I: Routing table is not null in replication group cluster store for replication group {0}

CWOB2096I

GET_RT_RGCLUSTERSTORE_CWOB2096=CWOB2096I: Routing table was obtained from replication group cluster store for replication group {0}.

CWOB2097I

ROUND_ROBIN_CWOB2097=CWOB2097I: Routing is based on round robin algorithm for replication group {0}.

CWOB2098I

ROUTING_RANDOM_CWOB2098=CWOB2098I: Routing is based on random selection for replication group {0}.

CWOB2100I

STALECONN_CWOB2100=CWOB2100I: Connection ({0}) is stale, it cannot be reused.

CWOB2101W

MAXWAIT_CWOB2101=CWOB2101W: Connection cannot be acquired after the maximum wait time.

CWOB2400E

INVALID_MAP_SET_CONFIGURATION_CWOB2400=CWOB2400E: Invalid Configuration: backingMap {0} is a member of more than one mapSet.

CWOB2401E

BACKING_MAP_WO_MAPSET_CWOB2401=CWOB2401E: Invalid Configuration: backingMap {0} in distributed ObjectGrid {1} is not in a mapSet.

CWOB2402E

MAPSET_REF_NONEXISTENT_BMAP_CWOB2402=CWOB2402E: Invalid Configuration: mapSet has a reference to a {0} map. This backingMap does not exist in the ObjectGrid XML file.

CWOB2403E

INVALID_XML_FILE_CWOB2403=CWOB2403E: The XML file is invalid. A problem has been detected with {0} at line {1}. The error message is {2}.

CWOB2404W

INVALID_CONFIG_VALUE_CWOB2404=CWOB2404W: The value specified for {0} is {1}. This is an invalid value. {0} will not be set.

CWOB2405E

OG_BINDING_REF_NONEXISTENT_OG_CWOB2405=CWOB2405E: The objectgridBinding ref {0} in the Cluster XML file does not reference a valid objectGrid from the ObjectGrid XML file.

CWOB2407W

PLUGIN_PROPERTY_INVALID_CWOB2407=CWOB2407W: The {0} property on the {1} plug-in class could not be set. The exception is {2}.

CWOB2408E

INVALID_ARGUMENT_CWOB2408=CWOB2408E: The following argument is invalid: {0}

CWOBJ2409E

SERVER_STARTUP_EXCEPTION_CWOBJ2409=CWOBJ2409E: An exception occurred during server startup: {0}

CWOBJ2410E

ACTIVATION_FAILURE_CWOBJ2410=CWOBJ2410E: ObjectGrid server failed to activate.

CWOBJ2411E

INITIALIZATION_FAILURE_CWOBJ2411=CWOBJ2411E: ObjectGrid server failed to initialize.

CWOBJ2412E

BOOTSTRAP_FAILURE_CWOBJ2412=CWOBJ2412E: ObjectGrid server failed to bootstrap.

CWOBJ2413E

SERVER_STOP_UNSUCCESSFUL_CWOBJ2413=CWOBJ2413E: The attempt to stop the ObjectGrid server was unsuccessful.

CWOBJ2414E

FORCEFUL_TERMINATION_CWOBJ2414=CWOBJ2414E: The ObjectGrid server will be forcefully terminated.

CWOBJ2415I

SCRIPT_CREATION_CWOBJ2415=CWOBJ2415I: Creating script file {0}

CWOBJ2416E

PLUGIN_INSTANTIATION_ERROR_CWOBJ2416=CWOBJ2416E: Plugin {0} could not be instantiated and will not be set. The exception is {1}

CWOBJ2417W

DEPRECATED_CLUSTER_XML_ATTRIBUTE_CWOBJ2417=CWOBJ2417W: The {0} attribute on the objectgridBinding element has been deprecated in the cluster XML. Use the {0} attribute on the serverDefinition element.

CWOBJ2418E

SERVER_LAUNCH_FAILED_CWOBJ2418=CWOBJ2418E: ObjectGrid server was not successfully launched.

CWOBJ2419W

MIN_THREADPOOL_SIZE_WARNING_CWOBJ2419=CWOBJ2419W: minThreadPoolSize cannot be less than 1. The default value of {0} will be used.

CWOBJ2420W

MAX_THREADPOOL_SIZE_WARNING_CWOBJ2420=CWOBJ2420W: minThreadPoolSize is set to {0}. maxThreadPoolSize is set to {1}. maxThreadPoolSize must be greater than minThreadPoolSize. The default values will be used. minThreadPoolSize = {2}, maxThreadPoolSize = {3}.

CWOBJ2421W

OVERRIDE_WARNING_CWOBJ2421=CWOBJ2421W: The List that was supplied to override client side ObjectGrid settings for cluster {0} contains an element that is not an ObjectGridConfiguration object. This element will be removed from the List: {1}.

CWOBJ2422I

CHECKSUM_DIFFERENCE_CWOBJ2422=CWOBJ2422I: Configuration version on client may not be the same as configuration version used by this server. Client side: host = {0}, , port = {1}, , Server side: host = {2}, port = {3}.

CWOBJ2423I

CLIENT_OVERRIDE_URL_CWOBJ2423=CWOBJ2423I: Client side ObjectGrid settings will be overridden for cluster {0} using the URL {1}

CWOBJ2424I

CLIENT_OVERRIDE_MAP_CWOBJ2424=CWOBJ2424I: Client side ObjectGrid settings will be overridden for cluster {0} using an entry supplied in the overrideMap.

CWOBJ2425E

CLIENT_OVERRIDE_MAP_ERROR_CWOBJ2425=CWOBJ2425E: The Map provided to override client side ObjectGrid settings for cluster {0} contains a value that is not of type java.util.List. Client side ObjectGrid settings will not be overridden for this cluster.

CWOBJ2426E

CONTAINER_WITHOUT_ZONE_INVALID_CWOBJ2426=CWOBJ2426E: This ObjectGrid container has been started without a zone association. This ObjectGrid container must be started within a zone since one or more containers in the domain already have been started within one or more zones.

CWOBJ2427E

CONTAINER_WITH_ZONE_INVALID_CWOBJ2427=CWOBJ2427E: This ObjectGrid container has been started with a zone association. This ObjectGrid container must be started without a zone since one or more containers in the domain already have been started without a zone.

CWOBJ2428W

ZONE_CONFIG_DEFAULT_INVALID_CWOBJ2428=CWOBJ2428W: The ObjectGrid container {0} has started without an association to a zone, but other containers have already started within zones. {0} will be torn

down.

CWOBJ2429W

ZONE_CONFIG_CUSTOM_INVALID_CWOBJ2429=CWOBJ2429W: The ObjectGrid container {0} has started with an association to a zone, but other containers have already started without zone associations. {0} will be torn down.

CWOBJ2430E

ZONE_RULE_TOO_FEW_ZONES_CWOBJ2430=CWOBJ2430E: The zoneRule {0} contains too few zones ({1}) for the number of shardMappings ({2}) that use the zoneRule.

CWOBJ2431E

MAP_SET_NOT_CONFIGURED_FOR_ZONE_CWOBJ2431=CWOBJ2431E: The container was started in zone {0}, but the {1} mapSet for ObjectGrid {2} is not configured to run within this zone.

CWOBJ2432E

WRONG_NUMBER_SHARD_MAPPINGS_CWOBJ2432=CWOBJ2432E: The wrong number of {0} shardMappings were found for the {1} mapSet in the {2} ObjectGrid. Expected {3} shardMappings, but found {4}.

CWOBJ2500E

SERVER_STARTUP_ERROR_CWOBJ2500=CWOBJ2500E: Failed to start ObjectGrid server {0}.

CWOBJ2501I

LAUNCHING_SERVER_CWOBJ2501=CWOBJ2501I: Launching ObjectGrid server {0}.

CWOBJ2502I

LAUNCHING_SERVER_XML_CWOBJ2502=CWOBJ2502I: Starting ObjectGrid server using ObjectGrid XML file URL "{0}" and Cluster XML file URL "{1}".

CWOBJ2503I

SERVER_BOOTSTRAP_PEER_CWOBJ2503=CWOBJ2503I: Bootstrapping to a peer Objectgrid server on host {0} and port {1}.

CWOBJ2504I

SERVER_BOOTSTRAP_LIST_CWOBJ2504=CWOBJ2504I: Attempting to bootstrap to a peer ObjectGrid server using the following host(s) and port(s) "{0}".

CWOBJ2505I

SERVER_BOOTSTRAP_FILE_CWOBJ2505=CWOBJ2505I: Attempting to bootstrap to a peer ObjectGrid server using the Cluster XML file URL "{0}".

CWOBJ2506I

COMMAND_LINE_TRACE_FILE_CWOBJ2506=CWOBJ2506I: Trace is being logged to {0}.

CWOBJ2507I

COMMAND_LINE_TRACE_SPEC_CWOBJ2507=CWOBJ2507I: Trace specification is set to {0}.

CWOBJ2508I

LAUNCHING_SERVER_SECURITY_CWOBJ2508=CWOBJ2508I: A security properties file "{0}" has been specified and will be used to start the server.

CWOBJ2509E

SERVER_STARTUP_TIMEOUT_CWOBJ2509=CWOBJ2509E: Timed out after waiting {0} seconds for the server to start.

CWOBJ2510I

SERVER_STOP_CWOBJ2510=CWOBJ2510I: Stopping ObjectGrid server {0}.

CWOBJ2511I

SERVER_STOP_WAITING_CWOBJ2511=CWOBJ2511I: Waiting for the server to stop.

CWOBJ2512I

SERVER_STOPPED_CWOBJ2512=CWOBJ2512I: ObjectGrid server {0} stopped.

CWOBJ2513E

SERVER_STOP_TIMEOUT_CWOBJ2513=CWOBJ2513E: Timed out after waiting {0} seconds for the server to stop.

CWOBJ2514I

SERVER_START_WAITING_CWOBJ2514=CWOBJ2514I: Waiting for ObjectGrid server activation to complete.

CWOBJ2515E

INVALID_ARGS_CWOBJ2515=CWOBJ2515E: The arguments provided are invalid. Here are the valid arguments. {0}{1}

CWOBJ2516I

ACTIVATION_COMPLETED_CWOBJ2516=CWOBJ2516I: ObjectGrid server has completed activation.

CWOBJ2517I

SUCCESS_SERVER_BOOTSTRAP_PEER_CWOBJ2517=CWOBJ2517I: Successfully bootstrapped to peer

Objectgrid server on host {0} and port {1}.

CWOBJ2518I

LAUNCHING_CATALOGED_OG_SERVER_CWOBJ2518=CWOBJ2518I: Launching a cataloged ObjectGrid server

CWOBJ2519I

CWOBJ2519=CWOBJ2519I: The client interceptor has not been registered. Security will not be enabled.

CWOBJ2601I

ADD_SUFFIX_TO_VIEW_NAME=CWOBJ2601I: Add suffix {0} to stream query views deployed in partition {1}.

CWOBJ2602W

VIEW_TRANSFORMER_EXISTS=CWOBJ2602W: The view transformer {0} has already existed.

CWOBJ2603W

STREAM_DESCRIPTOR_EXISTS=CWOBJ2603W: The stream descriptor {0} has already existed.

CWOBJ2604I

STREAM_QUERY_JAR_NOT_IN_CLASSPATH=CWOBJ2604I: The stream query jar file is not in the class path.

CWOBJ2605E

STREAM_QUERY_LOGGER_ERROR=CWOBJ2605E: The stream query logger setting method introspection or invocation error: {0}

CWOBJ2606W

VIEW_REMOVE_NON_EXISTING_ENTRY=CWOBJ2606W: Try to remove a non-existing entry for key {0}

CWOBJ2607E

STREAM_QUERY_SET_ACROSS_MAP_SET=CWOBJ2607E: The stream query set with name {0} contains maps from different map sets.

CWOBJ3001I

EM_SERVICE_STARTED_CWOBJ3001I=CWOBJ3001I: The ObjectGrid EntityManager service is available to process requests for ObjectGrid: {0} and container or server: {1}

CWOBJ3002I

EM_INIT_ENTITIES_CWOBJ3002I=CWOBJ3002I: Initializing entity metadata for ObjectGrid: {0}

CWOBJ3003I

EM_REGISTERED_CWOBJ3003I=CWOBJ3003I: Entity registered: {0}

CWOBJ3004E

EM_REGISTER_EXCEPTION_CWOBJ3004E=CWOBJ3004E: An exception occurred while registering an entity: {0}

CWOBJ3005I

EM_CREATING_INDEX_CWOBJ3005I=CWOBJ3005I: Creating index {0} for entity BackingMap {1}, attribute {2}

CWOBJ3006E

EM_UNSUPPORTED_INDEX_TYPE_CWOBJ3006E=CWOBJ3006E: The defined MapIndexPlugin type is unsupported for index: {0}, BackingMap: {1}, attribute: {2}.

CWOBJ3007E

EM_LATE_REGISTRATION_CWOBJ3007E=CWOBJ3007E: Unable to register new entity {0} after ObjectGrid initialization has completed.

CWOBJ3008E

EM_BACKINGMAP_REASSOCIATION_CWOBJ3008E=CWOBJ3008E: BackingMap {0} is associated with entity {1} and cannot be reassociated with entity {2}

CWOBJ3009E

EM_REPOSITORY_EXCEPTION_CWOBJ3009E=CWOBJ3009E: An exception occurred while communicating with the entity metadata repository: {0}

CWOBJ3010E

EM_INVALID_MAPSET_CWOBJ3010E=CWOBJ3010E: All entity BackingMaps must be members of a MapSet with the name: "ENTITY_MAPSET".

CWOBJ3011E

EM_METADATA_LISTENER_EXCEPTION_CWOBJ3011E=CWOBJ3011E: Error creating entity metadata for entity {0} ({1}): {2}

CWOBJ3012E

EM_MULTIPLE_MAPSETS_CWOBJ3012E=CWOBJ3012E: Invalid entity MapSet configuration. Only one Entity MapSet is allowed. Invalid MapSets: {0}

CWOBJ3013E

EM_MULTIPLE_MAPSETS_CWOBJ3013E=CWOBJ3013E: The EntityMetadata repository is not available.

Timeout threshold reached when trying to register the entity: {0}.

CWOB3014I

AVAILABILITY_STATE_CHANGED_CWOB3014=CWOB3014I: The availability state has changed for {0}. The state is now {1}. It was previously {2}.

CWOB3015E

EM_MISSING_MAPSET_CWOB3015E=CWOB3015E: Invalid Entity MapSet configuration. Unable to find MapSet that contains a BackingMap for {0}

CWOB3016E

EM_SCHEMA_MAPSET_CROSSOVER_CWOB3016E=CWOB3016E: Invalid entity MapSet configuration. Entity {0} should be present in MapSet {1} but is already exists in MapSet {2}.

CWOB3101E

WB_LOADER_INITIALIZATION_FAILED_CWOB3101E=CWOB3101E: The write-behind loader of map {0} partition {1} initialization fails with exception {2}.

CWOB3102E

WB_LOADER_FAILED_CWOB3102E=CWOB3102E: Loader fails to do a write-behind update to the database for map {0} partition {1}. The exception is {2}.

CWOB3103E

WB_LOADER_FAILED_CWOB3103E=CWOB3103E: The write-behind loader of map {0} partition {1} fails to complete a transaction. The exception is {2}.

CWOB3104W

WB_LOADER_LOCKTIMEOUT_CWOB3104W=CWOB3104W: The write-behind loader of map {0} partition {1} gets a lock timeout exception when trying to flip the queue maps: {2}

CWOB3111E

CLIENT_LOADER_AGENT_FAIL_CWOB3111E=CWOB3111E: The client loader agent {0} execution fails with exception: {1}.

CWOB3112I

DEFAULT_PERSISTENCE_UNIT_CWOB3112I=CWOB3112I: The JPA persistence unit name is not specified. The first persistence unit {0} defined in the persistence.xml is used.

CWOB3113E

AGENT_FAIL_CWOB3113E=CWOB3113E: The DataGrid agent {0} execution fails with a fatal exception: {1}.

CWOB3114E

AGENT_FAIL_RETRYABLE_CWOB3114E=CWOB3114E: The DataGrid agent {0} execution fails with a retryable exception: {1}.

CWOB3115E

UNEXPECTED_SHARD_STATE_CWOB3115E=CWOB3115E: The shard is expected in {0} state, but currently it is in {1} state. It might take a while for a shard to move to the target state.

CWOB3121E

TIME_BASED_DBUPDATE_AGENT_FAIL_CWOB3121E=CWOB3121E: The time-based database update agent fails with exception: {0}.

CWOB3122E

TIME_BASED_DBUPDATE_FAIL_CWOB3122E=CWOB3122E: The time-based database update fails with exception: {0}.

CWOB3131E

JPA_TX_CALLBACK_NOT_FOUND_CWOB3131E=CWOB3131E: The JPATxCallback transaction callback plugin cannot be found.

CWOB3141W

NODEGROUP_NOT_SET_FOR_ZONE_SUPPORT_CWOB3141W=CWOB3141W: This ObjectGrid server is not associated with a zone. In order to start the ObjectGrid server in a zone, ensure that the server's node is within a node group whose name begins with the string ReplicationZone.

CWOB6090E

EmptyIOR=CWOB6090E: IOR is null for this shard of partition ({0}).

CWOB6091E

NullSelectedTarget=CWOB6091E: Selected target for partition ({0}) is null.

CWOB6092E

NullOrbContext=CWOB6092E: No orb context associated with this request for partition ({0}) with transaction ID ({1}), so routing cannot be made.

CWOB6093E

NullRequest=CWOB6093E: Request is null.

CWOB6094E

NullResponse=CWOBJ6094E: Response is null.

CWOB6096E

SystemException=CWOB6096E: Client encountered system exception.

CWOB6200I

RoutingTablePropagated=CWOB6200I: Server routing table is propagated among servers for partition ({0}).

CWOB6600I

NoTargetAvailable=CWOB6600I: No target is available for partition ({0}).

CWOB6602I

NoRoutingTableAvailable=CWOB6602I: No routing table is available for partition ({0}).

CWOB6620I

MapSetBootstrap=CWOB6620I: Bootstrap to get MapSet and Map for routing.

CWOB6621E

FailedMapSet=CWOB6621E: Cannot look up map set name for this map ({0}).

CWOB6622E

InvalidPartition=CWOB6622E: Partition ({0}) is invalid.

CWOB6661I

ForwardNeeded=CWOB6661I: Forward is needed for partition ({0}) since this shard {1} does not have capacity to serve this request.

CWOB6662I

MaxForward=CWOB6662I: Maximum number of forwards has reached for ({0}).

CWOB6663I

RetryNeeded=CWOB6663I: Retry is needed for partition ({0}) since this shard {1} does not have capacity to serve this request.

CWOB6664I

MaxRetries=CWOB6664I: Maximum number of retries has reached for ({0}).

CWOB6666I

ConfigMaxForward=CWOB6666I: The configured maximum number of forwards is {0}.

CWOB6668I

ConfigMaxRetries=CWOB6668I: The configured maximum number of retries is {0}.

CWOB6800I

ClientRTUpdated=CWOB6800I: Client received new routing table for partition ({0}) with epoch ({1}).

CWOB6860I

DQNotMatch=CWOB6860I: The data quality of this shard does not meet the requirement of this request for partition ({0}). The router will automatically forward this request to shards with better data quality.

CWOB6866I

ClientDQ=CWOB6866I: Client required quality of data for this request is {0}

CWOB6900I

RemoteZoneLink=CWOB6900I: Remote zone ({0}) joined; routing table is linked.

CWOB6902I

RemoteZoneLeave=CWOB6902I: Remote zone ({0}) left; routing table is updated.

CWOB6904I

PrimaryZone=CWOB6904I: Zone ({0}) is set as the primary zone of partition ({1}).

CWOB6906I

PrimaryZoneChange=CWOB6906I: Primary zone changed from Zone ({0}) to Zone ({1}) for partition ({2}).

CWOB7000I

ServerAgentStart=CWOB7000I: ObjectGrid server agent started successfully for server ({0}).

CWOB7006I

DynamicPort=CWOB7006I: ObjectGrid server agent generated dynamic port {0}.

CWOB7008I

DynamicVerifyPort=CWOB7008I: ObjectGrid server agent verified dynamically generated port {0}.

CWOB7010E

ServerAgentMalfunction=CWOB7010E: ObjectGrid server agent encountered failure.

CWOB7200I

DeadServer=CWOB7200I: Detected the failure of server ({0}) in core group ({1}).

CWOBJ7201I

NewServer=CWOBJ7201I: Detected the addition of new server ({0}) in core group ({1}).

CWOBJ7203I

NewLeader=CWOBJ7203I: Leader changed. New leader ({0}) is elected in core group ({1}) and reported to catalog server.

CWOBJ7204I

CurrLeader=CWOBJ7204I: The current leader for core group ({1}) is ({0}).

CWOBJ7206I

ChangeLeader=CWOBJ7206I: New leader is ({0}). Old leader was ({1}).

CWOBJ7210I

RoutingTableUpdate=CWOBJ7210I: Updated routing table for partition ({0}).

CWOBJ7700I

PeerManagerStart=CWOBJ7700I: Peer Manager service started successfully in server ({0}) with core group ({1}).

CWOBJ7701I

PeerManagerStop=CWOBJ7701I: Peer Manager service stopped.

CWOBJ7710I

PeerManagerLeaderToAll=CWOBJ7710I: Peer Manager received broadcasting message from leader.

CWOBJ7712I

PeerManagerAllToLeader=CWOBJ7712I: Leader gathers messages from all members.

CWOBJ7800I

Start_HAController=CWOBJ7800I: Start ObjectGrid HA Controller with core group ({0}), host ({1}), and port ({2}).

CWOBJ7801I

AddMember_HAController=CWOBJ7801I: Add new server member ({0}) as instructed by catalog server.

CWOBJ7802I

removeMember_HAController=CWOBJ7802I: Remove server member ({0}) as instructed by catalog server.

CWOBJ7810E

Failed_Start_HAController=CWOBJ7810E: Failed to start ObjectGrid HA Controller.

CWOBJ7811E

Failed_AddMember_HAController=CWOBJ7811E: Failed to add new server member ({0}) as instructed by catalog server.

CWOBJ7812E

Failed_removeMember_HAController=CWOBJ7812E: Failed to remove server member ({0}) as instructed by catalog server.

CWOBJ7820E

InconsistentView_HAController=CWOBJ7820E: Members are in inconsistent view for core group ({0}).

CWOBJ8000I

Register_CWOBJ8000=CWOBJ8000I: Registration is successful with zone ({0}) and coregroup of ({1}).

CWOBJ8009E

Failed_Register_CWOBJ8009=CWOBJ8009E: Registration failed with zone ({0}).

CWOBJ8101I

StandbyCatalogServerCreated_CWOBJ8101=CWOBJ8101I: Notify that standby catalog server is created with domain= {0} and with IOR= {1}

CWOBJ8102I

MasterCatalogServerCreated_CWOBJ8102=CWOBJ8102I: Notify that master catalog server is created with domain= {0} and with IOR= {1}

CWOBJ8103I

NullORB_CWOBJ8103=CWOBJ8103I: ORB is null at this time, automatic retry later, Catalog server cluster is {0}

CWOBJ8106I

MasterCatalogServerActivated_CWOBJ8106=CWOBJ8106I: master catalog server cluster activated with cluster {0}

CWOBJ8108I

ResentStandbyCatalogServer_CWOBJ8108=CWOBJ8108I: Re-send standby catalog server on the request of master catalog server with domain= {0} and IOR= {1}

CWOBJ8109I

UpdateCatalogServerCluster_CWOBj8109=CWOBj8109I: Updated catalog server cluster {0} from server {1} with entry {2}

[CWOBj8201I](#)

BatchInfo_CWOBj8201=CWOBj8201I: BatchProcessor is processing jobs for {0}

[CWOBj8401I](#)

WaitForReplica_CWOBj8401=CWOBj8401I: Waiting for a server replica to be started. Start another server(s) immediately.

[CWOBj8601I](#)

PeerServers_CWOBj8601=CWOBj8601I: PeerManager found peers of size {0}

[CWOBj8700E](#)

OutOfRange_CWOBj8700=CWOBj8700E: Custom partitioning is out of range for map ({0}).

[CWOBj9000I](#)

ENGLISH_ONLY_INFO_MESSAGE_CWOBj9000=CWOBj9000I: This message is an English-only Informational message: {0}.

[CWOBj9001W](#)

ENGLISH_ONLY_WARN_MESSAGE_CWOBj9001=CWOBj9001W: This message is an English-only Warning message: {0}.

[CWOBj9002E](#)

ENGLISH_ONLY_ERROR_MESSAGE_CWOBj9002=CWOBj9002E: This message is an English only Error message: {0}.

[CWOBj9999I](#)

RegistrationSuccess_CWOBj9999=CWOBj9999I: This jvm ({0}) is registered successfully with CatalogServer. The CatalogServer will instruct what functions this jvm will take.

CWOBJ0001E

ILLEGAL_STATE_EXCEPTION_CWOBJ0001=CWOBJ0001E: Method, {0}, was called after initialization completed.

Explanation

After initialization completes, certain method invocations are no longer accepted.

User response

Restructure your code so that the configuration completes before use of the runtime is initiated.

Parent topic: [CWOBJ](#)

CWOBJ0002W

IGNORING_UNEXPECTED_EXCEPTION_CWOBJ0002=CWOBJ0002W: ObjectGrid component is ignoring an unexpected exception: {0}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0005W

INTERRUPTED_EXCEPTION_CWOBJ0005=CWOBJ0005W: The thread created an InterruptedException: {0}

Explanation

An InterruptedException occurred.

User response

Check the exception message to see whether this interruption is expected.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ0006W

GENERAL_EXCEPTION_WARNING_CWOBJ0006=CWOBJ0006W: An exception occurred: {0}

Explanation

An exception occurred during the runtime.

User response

Check the exception message to see whether this is an expected exception.

Parent topic: [CWOBJ](#)

CWOBJ0007W

NULL_VALUE_WARNING_CWOBJ0007=CWOBJ0007W: The value null was specified for {0}, a default value of {1} is used.

Explanation

An null value was specified for the variable. A default value is used.

User response

Set the appropriate value. Please refer to ObjectGrid documentation to know the valid values for the variables or properties.

Parent topic: [CWOBJ](#)

CWOBJ0008E

INVALID_VALUE_ERROR_CWOBJ0008=CWOBJ0008E: The value {0} provided for the property {1} is invalid.

Explanation

An invalid value was specified for the variable.

User response

Set the appropriate value. Please refer to ObjectGrid document to know the valid values for the variables or properties.

Parent topic: [CWOBJ](#)

CWOBJ0010E

MISSING_KEY_ERROR_CWOBJ0010=CWOBJ0010E: Message key {0} is missing.

Explanation

A message key is missing in the message resource bundle

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ0011W

DESERIALIZATION_FIELD_NOT_FOUND_CWOBJ0011=CWOBJ0011W: Cannot deserialize field {0} in class {1}; using the default value.

Explanation

During the deserialization of an object, an expected field was not found. This field was probably not found because the object was deserialized by a different version of the class than the one that serialized it.

User response

This warning indicates a potential problem. No user action is required unless further errors arise.

Parent topic: [CWOBJ](#)

CWOBJ0012E

INVALID_LOGELEMENT_TYPE_CWOBJ0012=CWOBJ0012E: The LogElement type code, {0} ({1}), is not recognized for this operation.

Explanation

An internal error occurred in the ObjectGrid runtime.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0013E

EVICT_ENTRIES_EXCEPTION_CWOBJ0013=CWOBJ0013E: An exception occurred while attempting to evict entries from the cache: {0}

Explanation

A problem occurred while attempting to apply the eviction entries to the cache.

User response

Check the exception message to see whether this is an expected exception.

Parent topic: [CWOBJ](#)

CWOBJ0014E

NESTED_TRANSACTION_CWOBJ0014=CWOBJ0014E: The ObjectGrid runtime detected an attempt to nest transactions.

Explanation

The nesting of transactions is not permitted.

User response

Modify the code to avoid the nesting of transactions.

Parent topic: [CWOBJ](#)

CWOBJ0015E

TRANSACTION_EXCEPTION_CWOBJ0015=CWOBJ0015E: An exception occurred while attempting to process a transaction: {0}

Explanation

A problem occurred during transaction processing.

User response

Check the exception message to see whether this exception is expected.

Parent topic: [CWOBJ](#)

CWOBJ0016E

NO_ACTIVE_TRANSACTION_CWOBJ0016=CWOBJ0016E: No active transaction is detected for the current operation.

Explanation

An active transaction is necessary to perform this operation.

User response

Modify the code to start a transaction before performing this operation.

Parent topic: [CWOBJ](#)

CWOBJ0017E

DUPLICATE_KEY_EXCEPTION_CWOBJ0017=CWOBJ0017E: A duplicate key exception was detected during the processing of the ObjectMap operation: {0}

Explanation

The key for the entry already exists in the cache.

User response

Modify the code to avoid inserting the same key more than once.

Parent topic: [CWOBJ](#)

CWOBJ0018E

KEY_NOT_FOUND_EXCEPTION_CWOBJ0018=CWOBJ0018E: The key was not found during the processing of the ObjectMap operation: {0}

Explanation

The key for the entry does not exist in the cache.

User response

Modify the code to ensure that the entry exists before attempting the operation.

Parent topic: [CWOBJ](#)

CWOBJ0019W

MISSING_CACHE_ENTRY_SLOT_DATA_CWOBJ0019=CWOBJ0019W: Did not find data in the cache entry slot reserved for {0} to use for ObjectMap name {1}.

Explanation

An internal error occurred in the ObjectGrid runtime.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0020E

CACHE_ENTRY_NOT_IN_MAP_CWOBJ0020=CWOBJ0020E: Cache entry is not in BackingMap {0}.

Explanation

Internal error in ObjectGrid runtime.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0021E

OBJECT_TRANSFORMER_NOT_FOUND_CWOBJ0021=CWOBJ0021E: A usable ObjectTransformer instance was not found during the deserialization of the LogSequence object for {0} ObjectGrid and {1} ObjectMap.

Explanation

The receiving side of a LogSequence object does not have the proper configuration to support the required ObjectTransformer instance.

User response

Verify the configuration of the ObjectGrid instances for both the sending and receiving sides of the LogSequence object.

Parent topic: [CWOBJ](#)

CWOBJ0022E

LOCK_MANAGER_INTERNAL_ERROR_CWOBJ0022=CWOBJ0022E: The caller does not own mutex: {0}.

Explanation

An internal error occurred in the ObjectGrid runtime.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0023E

UNRECOGNIZED_COPY_MODE_CWOBJ0023=CWOBJ0023E: The CopyMode ({0}) is not recognized for this operation.

Explanation

An internal error occurred in the ObjectGrid runtime.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0024E

REQUIRED_FIELD_NOT_FOUND_CWOBJ0024=CWOBJ0024E: Cannot deserialize field {0} in class {1}.
Deserialization failed.

Explanation

During deserialization of an object, a required field was not found. This problem is likely an ObjectGrid runtime error.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0025E

SERIALIZATION_FAILED_CWOBJ0025=CWOBJ0025E: The serialization of the LogSequence object failed. The number of serialized LogElement objects ({0}) does not match the number of read LogElement objects ({1}).

Explanation

An internal error occurred in the ObjectGrid runtime.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0026E

INVALID_JMX_CREDENTIAL_CWOBJ0026=CWOBJ0026E: The JMX credential type is not right. It should be of type {0}.

Explanation

The JMX credential type is not right. If basic authentication is used, the expected type is String[] with the first element being user name and the second being password. If the client certificate is used, the expected type is Certificate[].

User response

Use the right credentials.

Parent topic: [CWOBJ](#)

CWOBJ0027E

CLONE_METHOD_NOT_SUPPORTED_CWOBJ0027=CWOBJ0027E: Internal runtime error. Clone method not supported: {0}

Explanation

An internal error occurred in the ObjectGrid runtime.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0028E

KEY_NOT_FOUND_EVICTOR_CWOBJ0028=CWOBJ0028E: An error occurred in {0} for the map {1}. The key, {2}, was not found in the map. LogElement type is {3}.

Explanation

An internal error occurred when trying to evict an entry.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0029E

MISSING_EVICTOR_DATA_CWOBJ0029=CWOBJ0029E: An error occurred in {0} for the map {1}. CacheEntry is missing a {2} object for key {3}. LogElement type is {4}.

Explanation

An internal error occurred when trying to evict an entry.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ0030I

OBJECTGRID_INSTRUMENTATION_ENABLED_CWOBJ0030=CWOBJ0030I: ObjectGrid entity class instrumentation is enabled. The instrumentation mode is {0}.

Explanation

ObjectGrid entity class instrumentation is enabled. Java classes in the configured transformation domain may be transformed to support field-access entities.

User response

None.

Parent topic: [CWOBJ](#)

CWOBJ0031W

OPTIMISTIC_LOCKING_WITHOUT_CALLBACK_CWOBJ0031=CWOBJ0031W: Optimistic locking is enabled for map {0}, but a custom OptimisticCallback is not provided.

Explanation

When using optimistic locking, data integrity is preserved using an OptimisticCallback.

User response

Provide a OptimisticCallback implementation to the BackingMap configuration.

Parent topic: [CWOBJ](#)

CWOBJ0032W

BUCKET_SIZE_MAY_BE_TOO_SMALL_CWOBJ0032=CWOBJ0032W: The number of buckets for map {0} may be too small. The current number of buckets is {1}. The current number of entries in the map is {2}. The following condition was observed: {3}.

Explanation

The number of buckets should be the estimated number of entries in a map. This message may also result if the hashCode() method of the object type is not well formed. Performance of your application may be impacted.

User response

Examine the current number of buckets and determine if it needs to be increased. Also examine the hashCode() algorithm to determine if it needs to be modified to better distribute the entries across the buckets.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ0033I

CLASS_NOT_IMPLEMENT_CLONE_CWOBJ0033=CWOBJ0033I: Class, {0}, does not implement the clone() method. Using serialization instead for this Class in map {1}

Explanation

If using the default the default ObjectTransformer and the object type does not implement the clone() method, the objectgrid is serialized and deserialized to create a copy of the object.

User response

If using a copy mode such as COPY_ON_READ or COPY_ON_READ_AND_COMMIT and your performance is slow, consider adding a clone() method to your object type or provide in a custom ObjectTransformer implementation for your object type.

Parent topic: [CWOBJ](#)

CWOBJ0034I

TARGET_MEMORY_UTILIZATION_THRESHOLD_LEVEL_CWOBJ0034=CWOBJ0034I: Memory utilization threshold percentage is set to {0} %.

Explanation

The target memory utilization threshold percentage is set to the target level.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

CWOBJ0035W

MEMORY_UTILIZATION_THRESHOLD_NOT_SUPPORTED_CWOBJ0035=CWOBJ0035W: Memory utilization threshold not supported for this JVM.

Explanation

The JVM does not support memory utilization threshold.

User response

None. Evictors and other ObjectGrid components can not rely on memory utilization threshold. They should run as normal.

Parent topic: [CWOBJ](#)

CWOBJ0036W

CHANGING_MEMORY_UTILIZATION_THRESHOLD_CWOBJ0036=CWOBJ0036W: Changing memory utilization threshold from {0} to {1} for {2} memory pool.

Explanation

The Java heap memory utilization threshold is changed. This may have impact on other system components that rely on the memory utilization threshold setting.

User response

Ensure the new memory utilization threshold is acceptable.

Parent topic: [CWOBJ](#)

CWOBJ0037W

CHANGING_MEMORY_COLLECTION_UTILIZATION_THRESHOLD_CWOBJ0037=CWOBJ0037W: Changing memory collection utilization threshold from {0} to {1} for {2} memory pool.

Explanation

The Java heap memory collection utilization threshold is changed. This may have impact on other system components that rely on the memory collection utilization threshold setting.

User response

Ensure the new memory collection utilization threshold is acceptable.

Parent topic: [CWOBJ](#)

CWOBJ0038W

MEMORY_THRESHOLD_EXCEEDED_CWOBJ0038=CWOBJ0038W: Memory threshold exceeded. Current heap memory usage: {0}.

Explanation

The Java heap memory threshold exceeded target usage threshold.

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ0039W

MEMORY_COLLECTION_THRESHOLD_EXCEEDED_CWOBJ0039=CWOBJ0039W: Memory collection threshold exceeded. Current heap memory usage: {0}.

Explanation

The Java heap memory collection threshold exceeded target usage threshold.

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ0040E

CWOBJ0040=CWOBJ0040E: Hash based data structure over run for {0} with {1} elements in the data structure. Examine the hashCode method on this class for better distribution.

Explanation

The hash based data structure holding elements in the grid is getting too many collisions. This is likely because the hashCode method on the key class has not been implemented effectively.

User response

Examine the hashCode algorithm of the class to determine if a more distributed result is possible.

Parent topic: [CWOBJ](#)

CWOBJ0900I

CWOBJ0900=CWOBJ0900I: The ObjectGrid runtime component is started for server {0}.

Explanation

The ObjectGrid component is started.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ0901E

CWOBJ0901=CWOBJ0901E: "{0}" system property is required to start ObjectGrid component for server {1}.

Explanation

ObjectGrid runtime component requires "{0}" to be specified as a Java Virtual Machine system property.

User response

See Information Center for using WebSphere Administrator Console for providing ObjectGrid required custom properties.

Parent topic: [CWOBJ](#)

CWOBJ0902W

CWOBJ0902=CWOBJ0902W: Error prevented the ObjectGrid runtime component from starting for server {0}.

Explanation

A prior error prevented the ObjectGrid component from starting.

User response

See prior error messages to determine what prevented ObjectGrid component from starting.

Parent topic: [CWOBJ](#)

CWOBJ0910I

CWOBJ0910=CWOBJ0910I: The ObjectGrid runtime component is stopped for server {0}.

Explanation

The ObjectGrid component is stopped.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

CWOBJ0911I

CWOBJ0911=CWOBJ0911I: Starting the ObjectGrid runtime component for server {0}.

Explanation

The ObjectGrid component is starting.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

CWOBJ0912E

CWOBJ0912=CWOBJ0912E: The application {0} has ObjectGrid configuration files that will not be used because application {1} is currently running an ObjectGrid server instance.

Explanation

Two applications with ObjectGrid server configuration files with this server name are deployed to this application server. Only one ObjectGrid server configuration is allowed in a WebSphere Application Server.

User response

Ensure there is only one ObjectGrid server application is deployed on this server.

Parent topic: [CWOBJ](#)

CWOBJ1001I

OPEN_FOR_BUSINESS_CWOBJ1001=CWOBJ1001I: ObjectGrid Server {0} is ready to process requests.

Explanation

ObjectGrid Server is ready to process requests.

User response

The services for this ObjectGrid Server are available.

Parent topic: [CWOBJ](#)

CWOBJ1002E

PORT_CONFLICT_CWOBJ1002=CWOBJ1002E: Server port {0} is already in use.

Explanation

ObjectGrid server cannot be started due to port conflict.

User response

Users need to choose another port.

Parent topic: [CWOBJ](#)

CWOBJ1003I

DCS_CWOBJ1003=CWOBJ1003I: DCS Adapter service is disabled by configuration, to enable it, please change your configuration with an endpoint defined.

Explanation

DCS adapter is turned off.

User response

Users can turn on DCS adapter by changing the configuration.

Parent topic: [CWOBJ](#)

CWOBJ1004E

SERVER_TOPIC_CWOBJ1004=CWOBJ1004E: Server topic is null

Explanation

Server topic is null

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1005E

CLIENT_REQUESTQ_CWOBJ1005=CWOBJ1005E: The incoming request queue is null.

Explanation

Client request handler cannot retrieve requests.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1006E

CLIENT_RESULTQ_CWOBJ1006=CWOBJ1006E: The outgoing result queue is null.

Explanation

Client request handler cannot give result to client.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1007E

CLIENT_REQUEST_CWOBJ1007=CWOBJ1007E: ObjectGrid client request is null.

Explanation

Client request handler cannot handle request that does not contain any information about the request.

User response

Check your request

Parent topic: [CWOBJ](#)

CWOBJ1008E

TXID_CWOBJ1008=CWOBJ1008E: ObjectGrid client request TxID is null.

Explanation

We use TXID to match connections and do pooling, TXID cannot be null.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1009E

SERVER_CWOBJ1009=CWOBJ1009E: ObjectGrid client received a null response from the server.

Explanation

Encountered a null response from server.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1010I

CLUSTER_SHUTDOWN_CWOBJ1010=CWOBJ1010I: Shutdown request is processing.

Explanation

Cluster servers are processing the shutdown request.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1011I

CLUSTER_SHUTDOWN_CWOBJ1011=CWOBJ1011I: Shutdown request is sending.

Explanation

Cluster servers are processing the shutdown request

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1012I

CLUSTER_SHUTDOWN_CWOBJ1012=CWOBJ1012I: Shutdown request is performed.

Explanation

Cluster servers are processing the shutdown request.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1013W

EXCEPTION_ON_SERVER_CWOBJ1013=CWOBJ1013W: An exception occurred on a remote server: {0}

Explanation

An exception occurred during the server runtime processing of a request from the client.

User response

Check the exception message to see whether this is an expected exception.

Parent topic: [CWOBJ](#)

CWOBJ1014I

CLASSPATH_PROBLEM_CWOBJ1014=CWOBJ1014I: Preceding {0} message may be caused by application classes missing from the classpath on the server.

Explanation

If an application class is not in the classpath on the server a serialization error will occur on the server when processing a message from a client.

User response

Check the exception message to determine which class is missing on the server. If the class is in the classpath, contact IBM support for assistance.

Parent topic: [CWOBJ](#)

CWOBJ1015I

OBJECTTRANSFORMER_PROBLEM_CWOBJ1015=CWOBJ1015I: Preceeding {0} message may be caused by an incorrect application implementation of the ObjectTransformer or Serializable interface

Explanation

If an application implementation of ObjectTransformer or Serializable interface is incorrect, a serialization error will occur on the server when processing a message from a client.

User response

Check the exception message to determine the problem.

Parent topic: [CWOBJ](#)

CWOBJ1016E

PROPERTY_FILE_DOES_NOT_EXIST_CWOBJ1016E=CWOBJ1016E: The property file {0} does not exist: {1}.

Explanation

The property file does not exist in the system. It will be ignored.

User response

Specify a valid property file.

Parent topic: [CWOBJ](#)

CWOBJ1110I

DCS_CWOBJ1110=CWOBJ1110I: Starting the transport for ObjectGrid cluster {0} using IP Address {1}, port {2}, transport type {3}.

Explanation

The ObjectGrid cluster member transport is starting.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

CWOBJ1111W

DCS_CWOBJ1111=CWOBJ1111W: Resolution of IP Addresses for host name {0} found only the loopback address. The loopback address will be used.

Explanation

There may be a problem with the host name or DNS resolution. For production related implementation, a non-loopback address is normally expected.

User response

Modify the host name or determine if a DNS problem exists.

Parent topic: [CWOBJ](#)

CWOBJ1112E

DCS_CWOBJ1112=CWOBJ1112E: An error was encountered while looking up the IP address for the host name of an ObjectGrid cluster member. The host name is {0} and the server name is {1}. The member will be excluded from the cluster.

Explanation

Unable to resolve the IP address for the indicated host. The ObjectGrid cluster member for the specified host will be excluded.

User response

Correct the host name lookup problem and retry.

Parent topic: [CWOBJ](#)

CWOBJ1113E

DCS_CWOBJ1113=CWOBJ1113E: ObjectGrid cluster transport service on this process is not started. This cluster member is not defined in the configuration.

Explanation

This ObjectGrid cluster member is not a configured member of the cluster. If this cluster member should be a member of a ObjectGrid cluster, repair the configuration.

User response

Review the current configuration.

Parent topic: [CWOBJ](#)

CWOBJ1114E

DCS_CWOBJ1114=CWOBJ1114E: ObjectGrid cluster transport service on this process could not process the incoming message. The message is {0} and the exception is {1}.

Explanation

An unexpected internal error has been detected.

User response

Review the IBM ObjectGrid internet support web site for a similar problem or contact IBM service.

Parent topic: [CWOBJ](#)

CWOBJ1115E

DCS_CWOBJ1115=CWOBJ1115E: An unrecognized view change event was received from the ObjectGrid cluster transport. The view identifier is {0} and the event is {1}.

Explanation

The type of the event is not recognized. The HA Manager does not know how to respond to the event.

User response

Review the IBM ObjectGrid internet support web site for a similar problem or contact IBM service.

Parent topic: [CWOBJ](#)

CWOBJ1116E

DCS_CWOBJ1116=CWOBJ1116E: An attempt by another process to connect to this process via the ObjectGrid cluster transport has been rejected. The connecting process provided a name of {0}, a target of {1}, a member name of {2} and an IP address of {3}. The error message is {4}.

Explanation

The ObjectGrid cluster transport has rejected the connection attempt.

User response

This may be a connection attempt from an unauthorized party.

Parent topic: [CWOBJ](#)

CWOBJ1117E

DCS_CWOBJ1117=CWOBJ1117E: An attempt to authenticate a connection has failed. The exception is {0}.

Explanation

The ObjectGrid cluster transport has rejected the connection attempt.

User response

This may be a connection attempt from an unauthorized party.

Parent topic: [CWOBJ](#)

CWOBJ1118I

DCS_CWOBJ1118=CWOBJ1118I: ObjectGrid Server Initializing [Cluster: {0} Server: {1}].

Explanation

The ObjectGrid cluster member is initializing.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

CWOBJ1119I

CLIENT_CWOBJ1119=CWOBJ1119I: ObjectGrid client failed to connect to host: {0} port: {1}.

Explanation

ObjectGrid client failed to connect.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1120I

CLIENT_CWOBJ1120=CWOBJ1120I: ObjectGrid Client connected successfully to host: {0} port: {1}.

Explanation

ObjectGrid Client connected successfully.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1201E

NO_CLIENT_ENDPOINT_DEFINED_CWOBJ1201E=CWOBJ1201E: No valid client access end points are defined.

Explanation

No valid client access end points are defined.

User response

Define a valid client access end point.

Parent topic: [CWOBJ](#)

CWOBJ1202E

SSL_SOCKET_FAILED_CWOBJ1202E=CWOBJ1202E: SSL Server Socket failed to initialize. The exception message is {0}

Explanation

SSL Server Socket fails to initialize. The SSL settings might be wrong or the port number is already in use.

User response

Please examine the exception to see what went wrong.

Parent topic: [CWOBJ](#)

CWOBJ1203W

CLIENT_RESPONSE_TIMEOUT_CWOBJ1203W=CWOBJ1203W: Received a timeout event from the server for transaction: {0}

Explanation

Client did not receive expected response message from the server within a configured timeout limit.

User response

Look for prior messages that may explain the timeout. If none found, try increasing the timeout limit.

Parent topic: [CWOBJ](#)

CWOBJ1204W

UNKNOWN_MESSAGE_TYPE_CWOBJ1204W=CWOBJ1204W: Received a message of unknown message type. The message is: {0}

Explanation

An unexpected internal error has been detected.

User response

Review the IBM ObjectGrid internet support web site for a similar problem or contact IBM service.

Parent topic: [CWOBJ](#)

CWOBJ1205E

SSL_INITIALIZATION_ERROR_CWOBJ1205E=CWOBJ1205E: SSL Initialization failed. The exception message is {0}

Explanation

SSL Initialization failed. The SSL settings might be wrong.

User response

Please examine the exception to see what went wrong.

Parent topic: [CWOBJ](#)

CWOBJ1206W

SSL_INITIALIZATION_WARNING_CWOBJ1206W=CWOBJ1206W: SSL Initialization failed. The exception message is {0}

Explanation

SSL Initialization failed. The SSL settings might be wrong.

User response

Please examine the exception to see what went wrong.

Parent topic: [CWOBJ](#)

CWOBJ1207W

CONFIG_PROPERTY_UNSUPPORTED_CWOBJ1207W=CWOBJ1207W: The property {0} on plug-in {1} is using an unsupported property type.

Explanation

Java primitives and their java.lang counterparts are the only supported property types. java.lang.String is also supported.

User response

Check the property type and change it to one of the supported types.

Parent topic: [CWOBJ](#)

CWOBJ1208W

CONFIG_PLUGIN_UNSUPPORTED_CWOBJ1208W=CWOBJ1208W: The specified plug-in type, {0}, is not one of the supported plug-in types.

Explanation

This type of plug-in is unsupported.

User response

Add one of the supported plug-in types.

Parent topic: [CWOBJ](#)

CWOBJ1211E

ERROR_OG_PMI_CREATE_FAILED_CWOBJ1211E=CWOBJ1211E: The Performance Monitoring Infrastructure (PMI) creation of {0} failed. The exception is {1}.

Explanation

An attempt to create ObjectGrid PMI failed.

User response

Examine the exception message and the first failure data capture (FFDC) log.

Parent topic: [CWOBJ](#)

CWOBJ1212I

PMI_NOT_FOUND=CWOBJ1212I: The WebSphere Application Server Performance Monitoring Infrastructure (PMI) cannot be found.

Explanation

The WebSphere Application Server PMI cannot be found.

User response

This is expected if ObjectGrid is not running in WebSphere Application Server. If ObjectGrid is running in WebSphere Application Server, contact IBM support.

Parent topic: [CWOBJ](#)

CWOBJ1215I

TP_CWOBJ1215=CWOBJ1215I: ObjectGrid Transaction Propagation Event Listener is initializing [ObjectGrid {0}].

Explanation

This informational message indicates that the ObjectGrid Transaction Propagation Event Listener is initializing.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

CWOBJ1216I

TP_CWOBJ1216=CWOBJ1216I: ObjectGrid Transaction Propagation Event Listener is initialized [ObjectGrid {0}].

Explanation

ObjectGrid Transaction Propagation Event Listener Initialized.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

CWOBJ1217I

TP_CWOBJ1217=CWOBJ1217I: ObjectGrid Transaction Propagation Service Point Initialized [ObjectGrid {0}].

Explanation

This informational message indicates that the ObjectGrid Transaction Propagation Event Listener is initialized.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1218E

TP_CWOBJ1218=CWOBJ1218E: ObjectGrid Transaction Propagation Event Listener failure occurred [ObjectGrid {0} Exception message {1}].

Explanation

ObjectGrid runtime encountered an ObjectGrid Transaction Propagation failure.

User response

Examine the exception to determine the failure.

Parent topic: [CWOBJ](#)

CWOBJ1219E

TP_CWOBJ1219=CWOBJ1219E: ObjectGrid Transaction Propagation Service End Point failure occurred [ObjectGrid {0} Exception message {1}].

Explanation

ObjectGrid runtime encountered an ObjectGrid Transaction Propagation Service End Point failure.

User response

Examine the exception to determine the failure.

Parent topic: [CWOBJ](#)

CWOBJ1220E

TRANPROPLISTENER_UNsupported_CWObj1220=CWObj1220E: ObjectGrid Transaction Propagation Service is not supported in this environment.

Explanation

ObjectGrid Transaction Propagation Service is not supported on z/OS or the standalone ObjectGrid server environment.

User response

Do not use ObjectGrid Transaction Propagation Service on z/OS or in the standalone ObjectGrid server environment

Parent topic: [CWObj](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1300I

PMA_CWOBJ1300=CWOBJ1300I: Adapter successfully initialized ObjectGrid.

Explanation

Adapter successfully initialized ObjectGrid.

User response

None. Informational entry.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1301E

PMA_CWOBJ1301=CWOBJ1301E: Adapter failed to initialize ObjectGrid. Exception occurred [Exception message {0}].

Explanation

Adapters attempt to initialize ObjectGrid failed.

User response

Please examine the exception to determine the failure.

Parent topic: [CWOBJ](#)

CWOBJ1302I

PMA_CWOBJ1302=CWOBJ1302I: Adapter stopped.

Explanation

Adapter stopped.

User response

None. Informational Only.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1303I

PMA_CWOBJ1303=CWOBJ1303I: Adapter started.

Explanation

Adapter started.

User response

None. Informational Only.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1304I

SECURITY_ENABLED_CWOBJ1304=CWOBJ1304I: ObjectGrid security is enabled.

Explanation

ObjectGrid security is enabled.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1305I

SECURITY_DISABLED_CWOBJ1305=CWOBJ1305I: ObjectGrid security is disabled.

Explanation

ObjectGrid security is disabled.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1306W

CANNOT_RETRIEVE_CLIENT_CERTS_CWOBJ1306=CWOBJ1306W: Cannot retrieve the client certificates from the SSL socket.

Explanation

For some reason, the runtime cannot retrieve the client certificates from the SSL socket.

User response

Check your SSL configurations.

Parent topic: [CWOBJ](#)

CWOBJ1307I

OBJECTGRID_SECURITY_ENABLED_CWOBJ1307=CWOBJ1307I: Security of the ObjectGrid instance {0} is enabled.

Explanation

Security of the ObjectGrid instance {0} is enabled.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1308I

OBJECTGRID_SECURITY_DISABLED_CWOBJ1308=CWOBJ1308I: Security of the ObjectGrid instance {0} is disabled.

Explanation

Security of the ObjectGrid instance {0} is disabled.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1309E

OBJECTGRID_CONNECT_TOKEN_CREATION_CWOBJ1309=CWOBJ1309E: Unexpected error occurred in the connect token creation: {0}.

Explanation

An unexpected error occurs in the connection token creation.

User response

Check the security configuration

Parent topic: [CWOBJ](#)

CWOBJ1310E

OBJECTGRID_CONNECT_TOKEN_VALIDATION_CWOBJ1310=CWOBJ1310E: An attempt by another process to connect to this process via the core group transport has been rejected. The connecting process provided a source core group name of {0}, a target of {1}, a member name of {2} and an IP address of {3}. The error message is {4}.

Explanation

The High Availability Manager has rejected a connection attempt.

User response

This may be a connection attempt from an unauthorized party.

Parent topic: [CWOBJ](#)

CWOBJ1311W

IGNORE_CREDENTIAL_GENERATOR_PROPS=CWOBJ1311W: The credentialGeneratorProps setting is ignored since the credentialGeneratorClass value is not provided.

Explanation

The credentialGeneratorProps setting is only used if the credentialGeneratorClass value is provided.

User response

Set the credentialGeneratorClass if you plan to customize the credential generator.

Parent topic: [CWOBJ](#)

CWOBJ1312W

EXPIRED_CREDENTIAL_EXCEPTION=CWOBJ1312W: The credential has been expired. The exception message is {0}.

Explanation

The credential has been expired. Please check the exception message for why it expires.

User response

ObjectGrid will try to re-generate a credential. If the problem persists, please check the exception messages for the reason why the credential expires.

Parent topic: [CWOBJ](#)

CWOBJ1313W

CUSTOM_SECURE_TOKEN_MANAGER_CLASS_IGNORED=CWOBJ1313W: The customSecureTokenManagerClass setting is ignored since the provided customSecureTokenManagerType value is not "custom".

Explanation

In order to use the custom secure token manager, the customSecureTokenManagerType property has to be set to "custom".

User response

Set the customSecureTokenManagerType value to "custom" to use the custom secure token manager; or remove the customSecureTokenManagerClass value to get rid of this warning.

Parent topic: [CWOBJ](#)

CWOBJ1400W

MULTIPLE_JAR_FILE_CWOBJ1400W=CWOBJ1400W: Detected multiple ObjectGrid runtime JARS files in the JVM. Using multiple ObjectGrid runtime JAR files may cause problems.

Explanation

Usually only one ObjectGrid runtime JAR should be found in a JVM.

User response

Use the appropriate ObjectGrid runtime JAR for your configuration.

Parent topic: [CWOBJ](#)

CWOBJ1401E

WRONG_JAR_FILE_CWOBJ1401E=CWOBJ1401E: Detected a wrong ObjectGrid runtime JAR file for this configuration. Detected configuration is {0}. Expected Jar file is {1}.

Explanation

Each ObjectGrid runtime JAR file corresponds to a particular supported configuration.

User response

Use the appropriate ObjectGrid runtime JAR for your configuration.

Parent topic: [CWOBJ](#)

CWOBJ1402E

MISSING_CONNECTION_LINK_CALLBACK_CWOBJ1402E=CWOBJ1402E: ObjectGrid connection link callback not found for id: {0}.

Explanation

Internal error in ObjectGrid runtime.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1500E

CANNOT_CREATE_GROUPNAME_CWOBJ1500=CWOBJ1500E: An exception occurred when attempting to create a GroupName for HA Group ({0}): {1}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1501E

CANNOT_JOIN_GROUP_CWOBJ1501=CWOBJ1501E: An exception occurred when member ({0}) attempted to join HA Group ({1}): {2}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1503E

CANNOT_GET_OBJECTGRID_CWOBJ1503=CWOBJ1503E: Cannot access ObjectGrid ({0}) for applying updates to replica member ({1}).

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1504E

CANNOT_PROCESS_REPLICA_CHANGES_CWOBJ1504=CWOBJ1504E: An exception occurred when attempting to process the LogSequences for replica ({0}): {1}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1505E

MORE_THAN_ONE_PRIMARY_RESPONSE_CWOBJ1505=CWOBJ1505E: More than one replication group member reported back as the primary. Only one primary can be active. ({0}).

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1506E

POSSIBLE_NETWORK_PARTITION_CWOBJ1506=CWOBJ1506E: More than one primary replication group member exists in this group ({1}). Only one primary can be active. ({0}).

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1507W

CANNOT_CLOSE_CHECKPOINT_CWOBJ1507=CWOBJ1507W: An exception occurred when attempting to end the replication process for BackingMap ({0}): {1}.

Explanation

While attempting to shut down a primary replication group member, an exception occurred during the clean up processing.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1508E

CANNOT_SEND_MESSAGE_CWOBJ1508=CWOBJ1508E: An exception occurred when attempting to send message ({0}) from sender ({1}) to receiver ({2}): {3}.

Explanation

A problem occurred while attempting to send a message between replication group members.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1509E

CANNOT_SERIALIZE_MESSAGE_CWOBJ1509=CWOBJ1509E: An exception occurred when attempting to serialize message ({0}): {1}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1510E

CANNOT_DESERIALIZE_MESSAGE_CWOBJ1510=CWOBJ1510E: An exception occurred when attempting to inflate message ({0}): {1}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1511I

OPEN_FOR_BUSINESS_CWOBJ1511=CWOBJ1511I: {0} ({1}) is open for business.

Explanation

Specified replication group member is now ready to accept requests.

User response

None.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1512W

GROUP_MEMBER_ALREADY_EXISTS_CWOBJ1512=CWOBJ1512W: {0} already exists in replication group {1}.

Explanation

The specified replication group member is already active in this replication group.

User response

None.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1513E

SYNCH_REPLICATION_FAILED_CWOBJ1513=CWOBJ1513E: Synchronous replication failed on {0} ({1}). This member is no longer active.

Explanation

A problem was encountered that prevented synchronous replication from successfully completing.

User response

Review previous messages in the log to help diagnose the problem. Stopping and restarting the specified server may be required.

Parent topic: [CWOBJ](#)

CWOBJ1514I

PRIMARY_DOWNGRADED_CWOBJ1514=CWOBJ1514I: Primary ({0}) is being downgraded to either a replica or standby.

Explanation

This is not a normal operation, but ObjectGrid processing can continue.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1515I

MIN_CONFIG_NOT_MET_CWOBJ1515=CWOBJ1515I: Minimum configuration requirements not satisfied for replication group ({0}).

Explanation

The necessary primary and replica configuration requirements were not met with the recent replication group member change.

User response

Wait for additional resources to be started and recognized for this configuration.

Parent topic: [CWOBJ](#)

CWOBJ1516E

CANNOT_ACTIVATE_OBJECTGRID_CWOBJ1516=CWOBJ1516E: An exception occurred when attempting to activate the replication process for ObjectGrid ({0}): {1}.

Explanation

While attempting to start a primary replication group member, an exception occurred during the activation processing.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1517E

SYNCH_REPLICATION_TRAN_FAILED_CWOBJ1517=CWOBJ1517E: Synchronous replication failed for transaction {2} on {0} ({1}). This member is no longer active.

Explanation

A problem was encountered that prevented synchronous replication from successfully completing.

User response

Review previous messages in the log to help diagnose the problem. Stopping and restarting the specified server may be required.

Parent topic: [CWOBJ](#)

CWOBJ1518E

CANNOT_COMMIT_REPLICA_CHANGES_CWOBJ1518=CWOBJ1518E: An exception occurred when attempting to commit replica transaction ({0}) for primary transaction ({1}) on Replica ({2}): {3}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1519E

CANNOT_ROLLBACK_REPLICA_CHANGES_CWOBJ1519=CWOBJ1519E: An exception occurred when attempting to rollback the LogSequences for replica ({0}): {1}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1520I

TRAN_LAG_THRESHOLD_MET_LOST_CWOBJ1520=CWOBJ1520I: The transaction lag threshold, {0}, was met. Current lag: {1}. Transaction {2} may be lost.

Explanation

The number of transactions waiting to be processed by the replica exceeded the transaction lag threshold. The replica is still waiting to process the same transaction that was flagged at the transaction lag warning level which indicates a possible lost transaction. The transaction may be lost due to a network failure or it may be slow in arriving.

User response

The transaction will be automatically requested from the primary. If this message appears frequently and is followed by CWOBJ523I messages, consider raising the transaction lag threshold as the transactions are arriving, but later than expected.

Parent topic: [CWOBJ](#)

CWOBJ1521I

TRAN_LAG_THRESHOLD_MET_SLOW_CWOBJ1521=CWOBJ1521I: The transaction lag threshold, {0}, was met. Current lag: {1}.

Explanation

The queue of transactions waiting to be processed by the replica exceeded the transaction lag threshold. The replica processes transactions slowly according to the transaction lag threshold.

User response

If the replica processes transactions at an acceptable rate, raise the transaction lag threshold. Or the server running the replica may be under too much load and is unable to keep up with the number of incoming transactions. This may be followed by a CWOBJ523I message.

Parent topic: [CWOBJ](#)

CWOBJ1522I

TRAN_WARNING_MET_CWOBJ1522=CWOBJ1522I: The transaction lag warning, {0}, was met. Current lag: {1}. Beginning check for lost transaction. Possible lost transaction is {2}.

Explanation

The queue of transactions waiting to be processed by the replica exceeded the transaction warning threshold. This could happen if the replica processes transactions slowly or if a transaction is missing. If the replica falls further behind, the transaction lag threshold will check whether it is still waiting for a specific transaction or if it is processing slowly.

User response

If transactions are processed at an acceptable rate, the transaction lag warning could be raised. Watch for a subsequent CWOBJ520I or CWOBJ521I messages to indicate if further action is required.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1523I

DUP_TRANSACTION_CWOBJ1523=CWOBJ1523I: Received a duplicate transaction, {0}. Transaction will be thrown away.

Explanation

The replica received a transaction that was already processed. This could happen if the replica thought it was missing a transaction, requested it again and the original transaction came in later.

User response

If this happens frequently, consider raising the transaction lag threshold to allow more time before a transaction is labeled as missing and requested.

Parent topic: [CWOBJ](#)

CWOBJ1524I

LISTENER_REREGISTER_CWOBJ1524=CWOBJ1524I: Replica listener {0} must reregister with the primary.

Explanation

The replica will deregister and reregister the primary. It will get a new snapshot of the data and then continue processing new transactions. This could happen if the replica fell behind on transaction processing or if a lost transaction was requested from the primary, but the primary could not resend it.

User response

If this happens frequently, verify that the server can handle the current load. Consider increasing the transaction lag threshold to allow the replica to have more waiting transactions.

Parent topic: [CWOBJ](#)

CWOBJ1525I

CHECKPRELOADSTATE_EXCEPTION_CWOBJ1525=CWOBJ1525I: A ReplicaPreloadController ({0}) for map {1} threw an unexpected exception in method checkPreloadState {2}

Explanation

When promoting from replica to primary an exception occurred when the ReplicaPreloadController was called to determine the state of the replica. The exception is ignored and preload is performed on the map.

User response

Examine the stack trace to determine the cause of the problem. Fix the problem in your implementation or contact IBM support if the problem does not appear to be in your implementation.

Parent topic: [CWOBJ](#)

CWOBJ1526I

ENTERING_PEER_MODE_CWOBJ1526=CWOBJ1526I: Replica {0} entering peer mode after {1} seconds

Explanation

This is an informational message on how long it took for a replica to enter peer mode where both primary and replica have the same data.

User response

None.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1527W

FAILED_ENTERING_PEER_MODE_CWOBJ1527=CWOBJ1527W: Replica {0} failed to enter peer mode after {1} seconds

Explanation

The replica failed to enter peer mode. Look for additional messages that point to the specific cause of the failure. Possible reasons may include a timeout or data failing to copy from the primary.

User response

Review the action recommended by the specific message for timeout, bad data copy or other reason.

Parent topic: [CWOBJ](#)

CWOBJ1528E

FAILED_ENTERING_PEER_MODE_TIMEOUT_CWOBJ1528=CWOBJ1528E: Replica {0} failed to enter peer mode:
Waiting for data copy from the primary to complete timed out. Current timeout (ms): {1}

Explanation

The replica was unable to copy all of the data from the primary before the timeout.

User response

Check for other error messages that could indicate a general communication error or other transactional errors.

Parent topic: [CWOBJ](#)

CWOBJ1529E

FAILED_ENTERING_PEER_MODE_COPY_CWOBJ1529=CWOBJ1529E: Replica {0} failed to enter peer mode: A transaction threw an error while copying data from the primary.

Explanation

A transaction throw an error while committing on the replica. Look for duplicate key, key not found or other transactional errors, or errors inflating the key.

User response

If there is an error inflating the key, verify that the server for the replica has the correct classpath.

Parent topic: [CWOBJ](#)

CWOBJ1530E

FAILED_ENTERING_PEER_MODE_ORDER_CWOBJ1530=CWOBJ1530E: Replica {0} failed to enter peer mode: Received incorrect ordering data from the primary, data copy cannot complete.

Explanation

The replica is unable to complete the data copy based on the information it received from the primary.

User response

Contact IBM support.

Parent topic: [CWOBJ](#)

CWOBJ1600I

GATEWAY_STARTED_CWOBJ1600=CWOBJ1600I: ManagementGateway service started on port ({0}).

Explanation

ManagementGateway service is ready to process requests.

User response

ManagementGateway service is available.

Parent topic: [CWOBJ](#)

CWOBJ1601E

GATEWAY_SERVICE_FAILED_CWOBJ1601=CWOBJ1601E: ManagementGateway service failed to start on port ({0}).

Explanation

ManagementGateway service failed to start.

User response

Ensure specified port is not already in use.

Parent topic: [CWOBJ](#)

CWOBJ1602E

GATEWAY_CLIENT_CONNECT_FAILED_CWOBJ1602=CWOBJ1602E: ManagementGateway service failed to connect to server at ({0}):({1}).

Explanation

ManagementGateway service failed to connect to server.

User response

Ensure server is running.

Parent topic: [CWOBJ](#)

CWOBJ1603E

MANAGEMENT_SERVICE_RESPONSE_FAILED_CWOBJ1603=CWOBJ1603E: Management service failed to respond to ({0}) remote request: {1}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWOBJ](#)

CWOBJ1604I

MANAGEMENT_GATEWAY_STOP_FAILED_CWOBJ1604=CWOBJ1604I: ManagementGateway service failed to stop connector due to Throwable {0}. Exiting.

Explanation

ManagementGateway service failed to stop connector.

User response

Try again at a later time.

Parent topic: [CWOBJ](#)

CWOBJ1605I

MANAGEMENT_GATEWAY_REFRESH_FAILED_CWOBJ1605=CWOBJ1605I: ManagementGateway caught Throwable {0} while refreshing attributes. Exiting.

Explanation

ManagementGateway service failed while refreshing attributes.

User response

Try again at a later time.

Parent topic: [CWOBJ](#)

CWOBJ1606I

NO_RESPONSE_FROM_SERVER_CWOBJ1606=CWOBJ1606I: {0} - Unable to get response from server {1}.
Returning false.

Explanation

Unable to get response from server.

User response

No action required.

Parent topic: [CWOBJ](#)

CWOBJ1607I

USE_WSADMIN_CWOBJ1607=CWOBJ1607I: {0} - When an ObjectGrid server is colocated with a WebSphere Application Server, use WSADMIN to stop server {1}. Returning false.

Explanation

When in a WebSphere Application Server environment, use WSADMIN to stop server.

User response

When in WebSphere Application Server environment, use WSADMIN to stop server.

Parent topic: [CWOBJ](#)

CWOBJ1608I

SERVER_NOT_RESPONDING_NULL_CWOBJ1608=CWOBJ1608I: {0} - Unable to get response from server {1}.
Ensure server is running. Returning null.

Explanation

Unable to get response from server.

User response

Please wait several seconds and retry operation.

Parent topic: [CWOBJ](#)

CWOBJ1609I

NO_ROUTING_TABLE_CWOBJ1609=CWOBJ1609I: {0} - Unable to get routing table. Please wait several seconds and retry operation. Returning null.

Explanation

Unable to get routing table.

User response

Please wait several seconds and retry operation.

Parent topic: [CWOBJ](#)

CWOBJ1610W

RESET_NULL_CLUSTER_CWOBJ1610=CWOBJ1610W: Try to reset a null cluster for {0}.

Explanation

Replication group cluster data are not available.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1611I

OPEN_CLUSTER_CWOBJ1611=CWOBJ1611I: Replication group cluster {0} is open for business.

Explanation

Now replication group cluster can accept requests.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1612I

CLOSE_CLUSTER_CWOBJ1612=CWOBJ1612I: Replication group cluster {0} is closed for business.

Explanation

Now replication group cluster cannot accept requests.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1615E

CREDENTIAL_EXPIRED_CWOBJ1615=CWOBJ1615E: The JMX credential has expired: {0}.

Explanation

The JMX credential has expired.

User response

Retry with new credential.

Parent topic: [CWOBJ](#)

CWOBJ1616I

JMX_SECURITY_NOT_FOUND_CWOBJ1616=CWOBJ1616I: JMX Security implementation not found.

Explanation

MX4J or Java version 5.0 or above is not available.

User response

If JMX security is desired, add MX4J to the classpath or use a level of Java that supports JMX.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1620I

REPLACE_SERVER_CWOBJ1620=CWOBJ1620I: Replacing target for wrongly routed request due to changes in the server. The new target is {0}.

Explanation

Old routing target replaced with new target.

User response

If the intended replication group is out of service, you need to bring it back.

Parent topic: [CWOBJ](#)

CWOBJ1630I

DOMINO_MODE_CWOBJ1630=CWOBJ1630I: Replication group cannot serve this request {0}.

Explanation

Routing is refused due to the unavailable service such as the Domino effect

User response

Information only.

Parent topic: [CWOBJ](#)

CWOBJ1632E

NULL_ID_CWOBJ1632=CWOBJ1632E: Original request does not have a valid ID; no way to forward this request.

Explanation

No way to forward this request because the original request does not have a valid ID.

User response

Report to IBM support

Parent topic: [CWOBJ](#)

CWOBJ1634I

BLIND_FORWARD_CWOBJ1634=CWOBJ1634I: Router cannot find the forwarding target; using blind forwarding.

Explanation

Router cannot find the forwarding target.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1660I

SERVER_NOT_RIGHT_CWOBJ1660=CWOBJ1660I: Replication group member has changed. This server does not host what is requested anymore. The original request is {0}.

Explanation

Replication group member has changed.

User response

If the intended replication group is out of service, you need to bring it back.

Parent topic: [CWOBJ](#)

CWOBJ1661I

SERVER_CLUSTER_DATA_CWOBJ1661=CWOBJ1661I: Cluster data are updated for replication group: {0}

Explanation

Cluster data are updated

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1663E

VERIFY_NULL_CLUSTER_CWOBJ1663=CWOBJ1663E: Server router cannot verify server routing for {0}, because cluster data for this replication group are null in the server.

Explanation

No replication group cluster data are available to verify.

User response

Report to IBM support

Parent topic: [CWOBJ](#)

CWOBJ1668W

NOT_STARTED_CWOBJ1668=CWOBJ1668W: Request is coming to the server that has not completely started.

Explanation

Server startup takes 60-120 seconds. Request will be automatically retried if you have configured so (by default the request will be automatically retried).

User response

Adjust your configuration or start your clients 60-120 seconds after you start your servers.

Parent topic: [CWOBJ](#)

CWOBJ1680W

PER_SERVER_CWOBJ1680=CWOBJ1680W: The configured TCP connection timeout is smaller than $\text{retryInterval} * \text{max}(\text{startupRetries}, \text{maxRetries})$, so there is possibility that connection will time out.

Explanation

The configured TCP connection timeout should be larger than $\text{retryInterval} * \text{max}(\text{startupRetries}, \text{maxRetries})$.

User response

Adjust your configuration.

Parent topic: [CWOBJ](#)

CWOBJ1682W

TOTAL_CWOBJ1682=CWOBJ1682W: The configured transaction timeout is smaller than $\text{maxForwards} * \text{retryInterval} * \text{max}(\text{startupRetries}, \text{maxRetries})$, so there is possibility that transaction will time out.

Explanation

The configured transaction timeout should be larger than $\text{maxForwards} * \text{retryInterval} * \text{max}(\text{startupRetries}, \text{maxRetries})$.

User response

Adjust your configuration.

Parent topic: [CWOBJ](#)

CWOBJ1700I

STANDLAONE_HAMANAGER_INITIALIZED_CWOBJ1700=CWOBJ1700I: Standalone HAManager is initialized with coregroup {0}.

Explanation

standalone HAManager is initialized successfully.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1702E

STANDLAONE_HAMANAGER_NOT_INITIALIZED_CWOBJ1702=CWOBJ1702E: Standalone HAManager is not initialized, so it cannot be started.

Explanation

Standalone HAManager is not initialized.

User response

Initialize it before starting it.

Parent topic: [CWOBJ](#)

CWOBJ1710I

STANDLAONE_HAMANAGER_STARTED_CWOBJ1710=CWOBJ1710I: Standalone HAManager is started successfully.

Explanation

Standalone HAManager is started successfully.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1711I

STANDLAONE_HAMANAGER_ALREADY_STARTED_CWOBJ1711=CWOBJ1711I: Standalone HAManager is already started successfully.

Explanation

Standalone HAManager is already started successfully.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1712E

STANDLAONE_HAMANAGER_NOT_STARTED_CWOBJ1712=CWOBJ1712E: Standalone HAManager is not started.

Explanation

Standalone HAManager is not started.

User response

Initialize and start it before using it.

Parent topic: [CWOBJ](#)

CWOBJ1713E

STANDLAONE_HAMANAGER_START_FAIL_CWOBJ1713=CWOBJ1713E: Standalone HAManager failed to start.

Explanation

Standalone HAManager failed to start.

User response

Check if ports are used already.

Parent topic: [CWOBJ](#)

CWOBJ1720I

HAMANAGER_CONTROLLER_CWOBJ1720=CWOBJ1720I: HAManager Controller detected that ObjectGrid server is in the WebSphere environment, using WebSphere HAManager instead of initializing and starting standalone HAManager.

Explanation

ObjectGrid server is running in the WebSphere environment.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1730I

HAMANAGER_CONTROLLER_EXTERNALHAM__NULL_CWOBJ1730=CWOBJ1730I: HAManager Controller detected that the WebSphere external HAManager is null.

Explanation

Cannot get the external HAManager from WebSphere.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1760I

HEARTBEAT_SLIDEBAR_SET_CWOBJ1760=CWOBJ1760I: Heartbeat frequency level is set to {0}.

Explanation

The parameters bag corresponding to the heartbeat frequency level is used.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1761E

HEARTBEAT_SLIDEBAR_SET_CWOBJ1761=CWOBJ1761E: Heartbeat frequency level {0} is beyond the allowable range.

Explanation

The heartbeat frequency level was set beyond the allowable range.

User response

Set the heartbeat frequency level within the allowable range.

Parent topic: [CWOBJ](#)

CWOBJ1762I

HEARTBEAT_SLIDEBAR_SET_CWOBJ1762=CWOBJ1762I: The custom details of heart beating parameters is set to {0}.

Explanation

The customized parameters are used.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1763I

HEARTBEAT_SLIDEBAR_SET_CWOBJ1763=CWOBJ1763I: The heartbeating parameters are {0}.

Explanation

The parameters are used.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1764I

LEADERMANAGER_SLIDEBAR_SET_CWOBJ1764=CWOBJ1764I: The leader manager heartbeating interval is {0}.

Explanation

The leader manager heartbeating interval is used.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1765I

LEADERMANAGER_SLIDEBAR_SET_CWOBJ1765=CWOBJ1765I: The leader manager heartbeating timeout is {0}.

Explanation

The leader manager heartbeating timeout is used.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1766I

LEADERMANAGER_SLIDEBAR_SET_CWOBJ1766=CWOBJ1766I: The number of leader manager heartbeating threads is {0}.

Explanation

The number of leader manager heartbeating threads is used.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1767I

DCS_SLIDEBAR_SET_CWOBJ1767=CWOBJ1767I: The DCS heartbeating interval is {0}.

Explanation

The DCS heartbeating interval is used

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1768I

DCS_SLIDEBAR_SET_CWOBJ1768=CWOBJ1768I: The DCS heartbeating timeout is {0}.

Explanation

The DCS heartbeating timeout is used

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1769I

DCS_SLIDEBAR_SET_CWOBJ1769=CWOBJ1769I: The number of DCS heartbeating threads is {0}.

Explanation

The number of DCS heartbeating threads is used

User response

none

Parent topic: [CWOBJ](#)

CWOBJ1790I

HAMANAGER_CONTROLLER_NEED_STANDALONE_HAM_CWOBJ1790=CWOBJ1790I: Need to initialize and start the standalone HAManager.

Explanation

Cannot get the external HAManager from WebSphere. Need to initialize and start the standalone HAManager.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1792I

TM_CWOBJ1792=CWOBJ1792I: The maximum number of threads is {0} and the minimum number of threads is {1}.

Explanation

Configure thread pool.

User response

Information only.

Parent topic: [CWOBJ](#)

CWOBJ1800I

CLIENT_RECEIVED_RESPONSE_HAVING_FORWARD_REQUIREMENT_CWOBJ1800=CWOBJ1800I: Forwarding is required for request {0} with response of {1}.

Explanation

Forward routing is required.

User response

None. Handled automatically

Parent topic: [CWOBJ](#)

CWOBJ1810I

CLIENT_FORWARDING_CWOBJ1810=CWOBJ1810I: Forwarding is required for response {0}.

Explanation

Forwarding is required for response.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1811E

FORWARDING_NOT_FOUND_REQUEST_CWOBJ1811=CWOBJ1811E: Forwarding is required, but the original request cannot be found.

Explanation

Forwarding is required, but the original request cannot be found.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1820E

FORWARD_NULL_RGID_CWOBJ1820=CWOBJ1820E: Forwarding request does not have a replication group identifier.

Explanation

There is not any replication group identifier in this forwarding request.

User response

Contact IBM Support

Parent topic: [CWOBJ](#)

CWOBJ1870I

CLIENT_DOMINO_CWOBJ1870=CWOBJ1870I: Server service is not available for response {0}.

Explanation

Server service is not available due to the Domino effect or other events.

User response

Bring at least the minimum number of servers up.

Parent topic: [CWOBJ](#)

CWOBJ1871E

NULL_DOMINO_CWOBJ1871=CWOBJ1871E: Detected unavailable service, received null response, no way to retry.

Explanation

Null response from the unavailable service

User response

Contact IBM support

Parent topic: [CWOBJ](#)

CWOBJ1872I

CLIENT_DOMINO_TIMEOUT_CWOBJ1872=CWOBJ1872I: Service is unavailable with response of {0}.

Explanation

Service is not available

User response

Bring at least the minimum number of servers up or check if server startup is successful.

Parent topic: [CWOBJ](#)

CWOBJ1890I

DEAD_SERVER_REROUTING_CWOBJ1890=CWOBJ1890I: Re-routing request {0} due to an un-responsive server.

Explanation

The request for intended server failed to complete. Request was re-routed to another server.

User response

None. Handled automatically. If the intended replication group is out of service, you need to bring it back.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1891E

NO_SERVER_REROUTING_CWOBJ1891=CWOBJ1891E: All servers are not available in replication group {0}.

Explanation

All servers were either not started or have failed. They are not available

User response

If the intended replication group is out of service, you need to bring it back.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1898W

FORWARD_NULL_TARGET_CWOBJ1898=CWOBJ1898W: Forwarding is required, but router cannot find new available target for response {0}

Explanation

Service is not available.

User response

Make service available.

Parent topic: [CWOBJ](#)

CWOBJ1899W

FORWARD_NULL_RGID_CWOBJ1899=CWOBJ1899W: Forwarding is required, but router cannot find right replication group for response {0}

Explanation

Replication group ID is lost.

User response

Contact IBM Support

Parent topic: [CWOBJ](#)

CWOBJ1900I

RPC_SERVICE_INIT_CWOBJ1900=CWOBJ1900I: Client server remote procedure call service is initialized.

Explanation

Client server remote procedure call service is initialized.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1901I

RPC_SERVICE_START_CWOBJ1901=CWOBJ1901I: Client server remote procedure call service is started.

Explanation

Client server remote procedure call service is started.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1902I

RPC_HANDLER_THREADS_START_CWOBJ1902=CWOBJ1902I: Client server remote procedure call handler threads are started.

Explanation

Client server remote procedure call handler threads are started.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1903I

CONFIG_NETWORK_SERVICE_INIT_CWOBJ1903=CWOBJ1903I: Configuration network service is initialized.

Explanation

Configuration network service is initialized.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1904I

CONFIG_NETWORK_SERVICE_START_CWOBJ1904=CWOBJ1904I: Configuration network service is started.

Explanation

Configuration network service is started.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1905I

CONFIG_NETWORK_HANDLER_START_CWOBJ1905=CWOBJ1905I: Configuration handler is started.

Explanation

Configuration handler is started.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1913I

SYSADMIN_NETWORK_SERVICE_INIT_CWOBJ1913=CWOBJ1913I: System administration network service is initialized.

Explanation

System administration network service is initialized.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1914I

SYSADMIN_NETWORK_SERVICE_START_CWOBJ1914=CWOBJ1914I: System administration network service is started.

Explanation

System administration network service is started.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1915I

SYSADMIN_NETWORK_HANDLER_START_CWOBJ1915=CWOBJ1915I: System administration handler is started.

Explanation

System administration handler is started.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ1921W

Cannot_Find_host_name=CWOBJ1921W: Cannot find host name

Explanation

Cannot get host name.

User response

System will use localhost to replace hostname.

Parent topic: [CWOBJ](#)

CWOBJ1922E

Cannot_Lookup_IP=CWOBJ1922E: Cannot lookup IP for this host ({0}).

Explanation

Host name you configured is not correct.

User response

Change your configuration file.

Parent topic: [CWOBJ](#)

CWOBJ1927E

HostNotMatchReal=CWOBJ1927E: Configured host IP {0} host name {1} does not match the host IP of this machine {2} host name {3}, so ObjectGrid cannot be started.

Explanation

Machine configuration and ObjectGrid configuration do not match.

User response

Change configuration.

Parent topic: [CWOBJ](#)

CWOBJ1929W

LocalHostUsed=CWOBJ1929W: LOCALHOST is used in the configuration that may lose server identity in multiple machine environment

Explanation

In multiple computer systems with remote actions, localhost cannot be used.

User response

Change localhost to real host name or IP address.

Parent topic: [CWOBJ](#)

CWOBJ1931I

ServerSupport=CWOBJ1931I: The configuration for {0} does not support either an ObjectGrid replication group member or a client/server transaction processor. This server will provide bootstrap support to peer ObjectGrid servers and clients only.

Explanation

The configuration for this server does not support either an ObjectGrid replication group member or a client/server transaction processor.

User response

Check your configuration.

Parent topic: [CWOBJ](#)

CWOBJ1932I

ThreadPoolMinMax=CWOBJ1932I: Client thread pool minimum size is {0} maximum size {1}.

Explanation

Thread pool size

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1993I

HEARTBEAT_SERVICE_INIT_CWOBJ1993=CWOBJ1993I: Health monitor heart beat service is initialized.

Explanation

Health monitor heart beat service is initialized.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1994I

HEARTBEAT_SERVICE_START_CWOBJ1994=CWOBJ1994I: Health monitor heart beat service is started.

Explanation

Health monitor heart beat service is started.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ1995I

HEARTBEAT_HANDLER_START_CWOBJ1995=CWOBJ1995I: Health monitor heart beat handler is started.

Explanation

Health monitor heart beat handler is started.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2000E

NO_RGM_CWOBJ2000=CWOBJ2000E: No member in this replication group {0}.

Explanation

No member can be found in this replication group.

User response

Check if servers are started or data are available

Parent topic: [CWOBJ](#)

CWOBJ2001W

NO_AVAILABLE_RGM_CWOBJ2001=CWOBJ2001W: No available member in this replication group {0}.

Explanation

No available member can be found in this replication group.

User response

Check if server service is available

Parent topic: [CWOBJ](#)

CWOBJ2002W

NO_AVAILABLE_RT_CWOBJ2002=CWOBJ2002W: No available routing table for this replication group {0}.

Explanation

No available routing table for this replication group.

User response

Check if clients have brought in routing table

Parent topic: [CWOBJ](#)

CWOBJ2003I

RT_CACHE_CWOBJ2003=CWOBJ2003I: Cannot find routing cache for cache key {0}, creating new routing cache.

Explanation

First time routing or cluster changes.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2010E

NULL_TARGET_CWOBJ2010=CWOBJ2010E: Target for this request is null.

Explanation

Request did not come with target information.

User response

contact IBM support.

Parent topic: [CWOBJ](#)

CWOBJ2020I

ClientProperty_CWOBJ2020=CWOBJ2020I: Client properties are {}.

Explanation

Client properties are loaded.

User response

None.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2021I

PreferLocalJVM_CWOBJ2021=CWOBJ2021I: Prefer local JVM is enabled.

Explanation

Prefer local JVM is enabled.

User response

None.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2022I

PreferLocalHost_CWOBJ2022=CWOBJ2022I: Prefer local host routing is enabled.

Explanation

Prefer local host is enabled.

User response

None.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2023I

PreferZones_CWOBJ2023=CWOBJ2023I: Preferred zone routing is enabled for zones: {0}.

Explanation

Prefer zone routing is enabled.

User response

None.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2060I

NEW_RT_CHANGE_CWOBJ2060=CWOBJ2060I: Client received new version of replication group cluster {0}.

Explanation

Client received new version of replication group cluster

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2068I

REACHABILITY_CWOBJ2068=CWOBJ2068I: Reachability control detected problem in replication group member {0}.

Explanation

Some server cannot be reached, reachability mechanism will handle it.

User response

None.

Parent topic: [CWOBJ](#)

CWOBJ2069I

REACHABILITY_TIMER_CWOBJ2069=CWOBJ2069I: Reachability control timer releases replication group member {0}.

Explanation

This member is available for routing.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2086I

ROUTING_THREAD_CONTROL_CWOBJ2086=CWOBJ2086I: Routing thread control is activated due to overload for replication group {0}.

Explanation

Thread control is in action.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2088I

REACHABILITY_CWOBJ2088=CWOBJ2088I: Reachability control is activated to regulate the server availability for replication group {0}.

Explanation

Reachability is in action.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2090W

NULL_RT_CWOBJ2090=CWOBJ2090W: Cannot find routing table for replication group {0}.

Explanation

Replication group cluster is null.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2091W

EMPTY_RT_CWOBJ2091=CWOBJ2091W: Routing table is not null, but it does not contain any servers for replication group {0}.

Explanation

Replication group cluster is empty.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2092I

NULL_RUNTIME_CWOBJ2092=CWOBJ2092I: Routing table is null in runtime for replication group {0}.

Explanation

Getting routing table from runtime.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2093I

NULL_RGCLUSTERSTORE_CWOBJ2093=CWOBJ2093I: Routing table is not null in replication group cluster store for replication group {0}

Explanation

Getting routing table from cluster store.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2096I

GET_RT_RGCLUSTERSTORE_CWOBJ2096=CWOBJ2096I: Routing table was obtained from replication group cluster store for replication group {0}.

Explanation

Obtained replication group cluster from replication group cluster store.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2097I

ROUND_ROBIN_CWOBJ2097=CWOBJ2097I: Routing is based on round robin algorithm for replication group {0}.

Explanation

Routing is based on round robin algorithm.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2098I

ROUTING_RANDOM_CWOBJ2098=CWOBJ2098I: Routing is based on random selection for replication group {0}.

Explanation

Routing is based on random selection.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2100I

STALECONN_CWOBJ2100=CWOBJ2100I: Connection ({0}) is stale, it cannot be reused.

Explanation

Connection is stale.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2101W

MAXWAIT_CWOBJ2101=CWOBJ2101W: Connection cannot be acquired after the maximum wait time.

Explanation

There are not any connections left in the pool.

User response

Increase the maximum number of connections in the configuration.

Parent topic: [CWOBJ](#)

CWOBJ2400E

INVALID_MAP_SET_CONFIGURATION_CWOBJ2400=CWOBJ2400E: Invalid Configuration: backingMap {0} is a member of more than one mapSet.

Explanation

A backingMap can belong to only one mapSet.

User response

Edit the cluster XML file so that each backingMap belongs to only one mapSet.

Parent topic: [CWOBJ](#)

CWOBJ2401E

BACKING_MAP_WO_MAPSET_CWOBJ2401=CWOBJ2401E: Invalid Configuration: backingMap {0} in distributed ObjectGrid {1} is not in a mapSet.

Explanation

Each backingMap of a distributed ObjectGrid must be placed in a mapSet.

User response

Edit the cluster XML file so that each backingMap in a distributed ObjectGrid belongs to a mapSet.

Parent topic: [CWOBJ](#)

CWOBJ2402E

MAPSET_REF_NONEXISTENT_BMAP_CWOBJ2402=CWOBJ2402E: Invalid Configuration: mapSet has a reference to a {0} map. This backingMap does not exist in the ObjectGrid XML file.

Explanation

Each map within a mapSet must reference a backingMap from the ObjectGrid XML file.

User response

Edit the XML file(s) so that each map within the mapSet references a backingMap from the ObjectGrid XML file.

Parent topic: [CWOBJ](#)

CWOBJ2403E

INVALID_XML_FILE_CWOBJ2403=CWOBJ2403E: The XML file is invalid. A problem has been detected with {0} at line {1}. The error message is {2}.

Explanation

The XML file does not conform to the schema.

User response

Edit the XML file so that it conforms to the schema.

Parent topic: [CWOBJ](#)

CWOBJ2404W

INVALID_CONFIG_VALUE_CWOBJ2404=CWOBJ2404W: The value specified for {0} is {1}. This is an invalid value. {0} will not be set.

Explanation

The value for this configuration attribute is not valid.

User response

Set the configuration attribute to a proper value in the XML file.

Parent topic: [CWOBJ](#)

CWOBJ2405E

OG_BINDING_REF_NONEXISTENT_OG_CWOBJ2405=CWOBJ2405E: The objectgridBinding ref {0} in the Cluster XML file does not reference a valid objectGrid from the ObjectGrid XML file.

Explanation

Each of the objectgridBindings must reference an objectGrid from the ObjectGrid XML file.

User response

Edit the XML files so that the objectgridBinding in the Cluster XML references a valid objectGrid in the ObjectGrid XML.

Parent topic: [CWOBJ](#)

CWOBJ2407W

PLUGIN_PROPERTY_INVALID_CWOBJ2407=CWOBJ2407W: The {0} property on the {1} plug-in class could not be set. The exception is {2}.

Explanation

The property for this plug-in could not be set.

User response

See the exception for more information.

Parent topic: [CWOBJ](#)

CWOBJ2408E

INVALID_ARGUMENT_CWOBJ2408=CWOBJ2408E: The following argument is invalid: {0}

Explanation

This argument is not a valid command line argument.

User response

Only use valid arguments on the command line.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2409E

SERVER_STARTUP_EXCEPTION_CWOBJ2409=CWOBJ2409E: An exception occurred during server startup: {0}

Explanation

An exception occurred during server startup.

User response

Check the exception message.

Parent topic: [CWOBJ](#)

CWOBJ2410E

ACTIVATION_FAILURE_CWOBJ2410=CWOBJ2410E: ObjectGrid server failed to activate.

Explanation

A problem occurred which caused server activation to fail.

User response

Check the log for exceptions.

Parent topic: [CWOBJ](#)

CWOBJ2411E

INITIALIZATION_FAILURE_CWOBJ2411=CWOBJ2411E: ObjectGrid server failed to initialize.

Explanation

A problem occurred which caused server initialization to fail.

User response

Check the log for exceptions.

Parent topic: [CWOBJ](#)

CWOBJ2412E

BOOTSTRAP_FAILURE_CWOBJ2412=CWOBJ2412E: ObjectGrid server failed to bootstrap.

Explanation

A problem occurred which caused server bootstrap to fail.

User response

Check the log for exceptions.

Parent topic: [CWOBJ](#)

CWOBJ2413E

SERVER_STOP_UNSUCCESSFUL_CWOBJ2413=CWOBJ2413E: The attempt to stop the ObjectGrid server was unsuccessful.

Explanation

A problem occurred during server shutdown.

User response

Check the log for exceptions.

Parent topic: [CWOBJ](#)

CWOBJ2414E

FORCEFUL_TERMINATION_CWOBJ2414=CWOBJ2414E: The ObjectGrid server will be forcefully terminated.

Explanation

A problem occurred during server shutdown.

User response

Check the log for exceptions.

Parent topic: [CWOBJ](#)

CWOBJ2415I

SCRIPT_CREATION_CWOBJ2415=CWOBJ2415I: Creating script file {0}

Explanation

A script file will be created in the OBJECTGRID_HOME directory.

User response

See the OBJECTGRID_HOME directory for the script file.

Parent topic: [CWOBJ](#)

CWOBJ2416E

PLUGIN_INSTANTIATION_ERROR_CWOBJ2416=CWOBJ2416E: Plugin {0} could not be instantiated and will not be set. The exception is {1}

Explanation

Plugin instantiation was not completed successfully.

User response

See the accompanying exception.

Parent topic: [CWOBJ](#)

CWOBJ2417W

DEPRECATED_CLUSTER_XML_ATTRIBUTE_CWOBJ2417=CWOBJ2417W: The {0} attribute on the objectgridBinding element has been deprecated in the cluster XML. Use the {0} attribute on the serverDefinition element.

Explanation

This is no longer a valid attribute

User response

Do not use the deprecated attribute. Use the attribute on the serverDefinition element.

Parent topic: [CWOBJ](#)

CWOBJ2418E

SERVER_LAUNCH_FAILED_CWOBJ2418=CWOBJ2418E: ObjectGrid server was not successfully launched.

Explanation

A problem occurred during server startup.

User response

Check the log for exceptions.

Parent topic: [CWOBJ](#)

CWOBJ2419W

MIN_THREADPOOL_SIZE_WARNING_CWOBJ2419=CWOBJ2419W: minThreadPoolSize cannot be less than 1. The default value of {0} will be used.

Explanation

The minThreadPoolSize was set to a value less than 1.

User response

Set the minThreadPoolSize to a value equal to or greater than 1.

Parent topic: [CWOBJ](#)

CWOBJ2420W

MAX_THREADPOOL_SIZE_WARNING_CWOBJ2420=CWOBJ2420W: minThreadPoolSize is set to {0}. maxThreadPoolSize is set to {1}. maxThreadPoolSize must be greater than minThreadPoolSize. The default values will be used. minThreadPoolSize = {2}, maxThreadPoolSize = {3}.

Explanation

maxThreadPoolSize must be greater than minThreadPoolSize.

User response

Set maxThreadPoolSize to a value greater than the minThreadPoolSize value.

Parent topic: [CWOBJ](#)

CWOBJ2421W

OVERRIDE_WARNING_CWOBJ2421=CWOBJ2421W: The List that was supplied to override client side ObjectGrid settings for cluster {0} contains an element that is not an ObjectGridConfiguration object. This element will be removed from the List: {1}.

Explanation

Client-side overriding will take place using only the objects in the List that are of type `com.ibm.websphere.objectgrid.config.ObjectGridConfiguration`.

User response

Remove objects from the client-side override List that are not of type `com.ibm.websphere.objectgrid.config.ObjectGridConfiguration`.

Parent topic: [CWOBJ](#)

CWOBJ2422I

CHECKSUM_DIFFERENCE_CWOBJ2422=CWOBJ2422I: Configuration version on client may not be the same as configuration version used by this server. Client side: host = {0}, , port = {1}, , Server side: host = {2}, port = {3}.

Explanation

An ObjectGrid client has connected to this server with a configuration version that is different than this server configuration version. This can occur when an ObjectGrid client bootstraps from one ObjectGrid server, and then contacts another server that was started with a different configuration file or the same configuration file with changes.

User response

Users should have administrators compare the configuration files provided by each server to determine if the differences are incompatible.

Parent topic: [CWOBJ](#)

CWOBJ2423I

CLIENT_OVERRIDE_URL_CWOBJ2423=CWOBJ2423I: Client side ObjectGrid settings will be overridden for cluster {0} using the URL {1}

Explanation

Overriding ObjectGrids on the client side using ObjectGrids found in the URL.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2424I

CLIENT_OVERRIDE_MAP_CWOBJ2424=CWOBJ2424I: Client side ObjectGrid settings will be overridden for cluster {0} using an entry supplied in the overrideMap.

Explanation

ObjectGridConfigurations will be used to override client side settings for the cluster specified.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2425E

CLIENT_OVERRIDE_MAP_ERROR_CWOBJ2425=CWOBJ2425E: The Map provided to override client side ObjectGrid settings for cluster {0} contains a value that is not of type java.util.List. Client side ObjectGrid settings will not be overridden for this cluster.

Explanation

Each value in the overrideMap must be of type java.util.List that contains ObjectGridConfiguration objects.

User response

Make this value of type java.util.List

Parent topic: [CWOBJ](#)

CWOBJ2426E

CONTAINER_WITHOUT_ZONE_INVALID_CWOBJ2426=CWOBJ2426E: This ObjectGrid container has been started without a zone association. This ObjectGrid container must be started within a zone since one or more containers in the domain already have been started within one or more zones.

Explanation

If ObjectGrid containers have already been started within zones in this domain, then no subsequent container can be started without an association to a zone in the domain.

User response

Start this container within a zone.

Parent topic: [CWOBJ](#)

CWOBJ2427E

CONTAINER_WITH_ZONE_INVALID_CWOBJ2427=CWOBJ2427E: This ObjectGrid container has been started with a zone association. This ObjectGrid container must be started without a zone since one or more containers in the domain already have been started without a zone.

Explanation

If ObjectGrid containers have already been started without zones in this domain, then no subsequent container can be started with an association to a zone in the domain.

User response

Start the container without a zone.

Parent topic: [CWOBJ](#)

CWOBJ2428W

ZONE_CONFIG_DEFAULT_INVALID_CWOBJ2428=CWOBJ2428W: The ObjectGrid container {0} has started without an association to a zone, but other containers have already started within zones. {0} will be torn down.

Explanation

If ObjectGrid containers have already been started within zones in this domain, then no subsequent container can be started without an association to a zone in the domain.

User response

Start the container within a zone.

Parent topic: [CWOBJ](#)

CWOBJ2429W

ZONE_CONFIG_CUSTOM_INVALID_CWOBJ2429=CWOBJ2429W: The ObjectGrid container {0} has started with an association to a zone, but other containers have already started without zone associations. {0} will be torn down.

Explanation

If ObjectGrid containers have already been started without zones in this domain, then no subsequent container can be started within a zone in the domain.

User response

Start the container without a zone.

Parent topic: [CWOBJ](#)

CWOBJ2430E

ZONE_RULE_TOO_FEW_ZONES_CWOBJ2430=CWOBJ2430E: The zoneRule {0} contains too few zones ({1}) for the number of shardMappings ({2}) that use the zoneRule.

Explanation

If a zoneRule contains more than one zone, it must have at least as many zones as shardMappings that make use of the zoneRule.

User response

Add zones to the zoneRule or move shardMappings to different zoneRules.

Parent topic: [CWOBJ](#)

CWOBJ2431E

MAP_SET_NOT_CONFIGURED_FOR_ZONE_CWOBJ2431=CWOBJ2431E: The container was started in zone {0}, but the {1} mapSet for ObjectGrid {2} is not configured to run within this zone.

Explanation

The zone name that is used to start the container must be within a zoneRule used by one of the shardMappings for the mapSet.

User response

Ensure that the zone is used by each mapSet within the deploymentPolicy that was used to start the container.

Parent topic: [CWOBJ](#)

CWOBJ2432E

WRONG_NUMBER_SHARD_MAPPINGS_CWOBJ2432=CWOBJ2432E: The wrong number of {0} shardMappings were found for the {1} mapSet in the {2} ObjectGrid. Expected {3} shardMappings, but found {4}.

Explanation

If the shard type is a replica, the number of shardMappings for the type should match the maximum number of replicas specified on the mapSet. There should be only 1 primary shardMapping.

User response

Adjust the number of shardMappings for the shardType.

Parent topic: [CWOBJ](#)

CWOBJ2500E

SERVER_STARTUP_ERROR_CWOBJ2500=CWOBJ2500E: Failed to start ObjectGrid server {0}.

Explanation

The ObjectGrid server failed to start properly.

User response

Check the log for exceptions.

Parent topic: [CWOBJ](#)

CWOBJ2501I

LAUNCHING_SERVER_CWOBJ2501=CWOBJ2501I: Launching ObjectGrid server {0}.

Explanation

An ObjectGrid server is starting up.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2502I

LAUNCHING_SERVER_XML_CWOBJ2502=CWOBJ2502I: Starting ObjectGrid server using ObjectGrid XML file URL "{0}" and Cluster XML file URL "{1}".

Explanation

An ObjectGrid server is starting using a cluster XML file and an ObjectGrid xml file.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2503I

SERVER_BOOTSTRAP_PEER_CWOBJ2503=CWOBJ2503I: Bootstrapping to a peer Objectgrid server on host {0} and port {1}.

Explanation

This ObjectGrid server will bootstrap to a peer server to retrieve information required to start.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2504I

SERVER_BOOTSTRAP_LIST_CWOBJ2504=CWOBJ2504I: Attempting to bootstrap to a peer ObjectGrid server using the following host(s) and port(s) "{0}".

Explanation

This ObjectGrid server will use the list of hosts and ports provided in an attempt to connect to a peer ObjectGrid server.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2505I

SERVER_BOOTSTRAP_FILE_CWOBJ2505=CWOBJ2505I: Attempting to bootstrap to a peer ObjectGrid server using the Cluster XML file URL "{0}".

Explanation

This ObjectGrid server will use the list of servers in the Cluster XML file in an attempt to connect to a peer ObjectGrid server.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2506I

COMMAND_LINE_TRACE_FILE_CWOBJ2506=CWOBJ2506I: Trace is being logged to {0}.

Explanation

The trace file has been set on the command line.

User response

See the specified trace file for ObjectGrid server start-up trace.

Parent topic: [CWOBJ](#)

CWOBJ2507I

COMMAND_LINE_TRACE_SPEC_CWOBJ2507=CWOBJ2507I: Trace specification is set to {0}.

Explanation

The trace specification has been set on the command line.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2508I

LAUNCHING_SERVER_SECURITY_CWOBJ2508=CWOBJ2508I: A security properties file "{0}" has been specified and will be used to start the server.

Explanation

A security properties file has been provided to start a secure server.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2509E

SERVER_STARTUP_TIMEOUT_CWOBJ2509=CWOBJ2509E: Timed out after waiting {0} seconds for the server to start.

Explanation

The ObjectGrid server did not start within the timeout interval.

User response

Check the log for exceptions.

Parent topic: [CWOBJ](#)

CWOBJ2510I

SERVER_STOP_CWOBJ2510=CWOBJ2510I: Stopping ObjectGrid server {0}.

Explanation

Stopping ObjectGrid server.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2511I

SERVER_STOP_WAITING_CWOBJ2511=CWOBJ2511I: Waiting for the server to stop.

Explanation

Waiting for the ObjectGrid server to stop.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2512I

SERVER_STOPPED_CWOBJ2512=CWOBJ2512I: ObjectGrid server {0} stopped.

Explanation

ObjectGrid server stopped.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2513E

SERVER_STOP_TIMEOUT_CWOBJ2513=CWOBJ2513E: Timed out after waiting {0} seconds for the server to stop.

Explanation

The ObjectGrid server did not stop within the timeout interval.

User response

Check the log for exceptions.

Parent topic: [CWOBJ](#)

CWOBJ2514I

SERVER_START_WAITING_CWOBJ2514=CWOBJ2514I: Waiting for ObjectGrid server activation to complete.

Explanation

The ObjectGrid server has launched. Waiting for the server to complete activation.

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ2515E

INVALID_ARGS_CWOBJ2515=CWOBJ2515E: The arguments provided are invalid. Here are the valid arguments.
{0}{1}

Explanation

The arguments provided to this script are invalid.

User response

Enter valid arguments.

Parent topic: [CWOBJ](#)

CWOBJ2516I

ACTIVATION_COMPLETED_CWOBJ2516=CWOBJ2516I: ObjectGrid server has completed activation.

Explanation

The ObjectGrid server is active and ready to process requests.

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ2517I

SUCCESS_SERVER_BOOTSTRAP_PEER_CWOBJ2517=CWOBJ2517I: Successfully bootstrapped to peer Objectgrid server on host {0} and port {1}.

Explanation

This ObjectGrid server has successfully bootstrapped to a peer server to retrieve information required to start this server.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2518I

LAUNCHING_CATALOGED_OG_SERVER_CWOBJ2518=CWOBJ2518I: Launching a cataloged ObjectGrid server

Explanation

A cataloged ObjectGrid server is starting up.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ2519I

CWOBJ2519=CWOBJ2519I: The client interceptor has not been registered. Security will not be enabled.

Explanation

The client is running without the ObjectGrid initializer specified in the orb.properties file and/or does not have the ObjectGrid binaries available to the root classloader.

User response

If ObjectGrid authentication and authorization are desired, then a full ObjectGrid install is required.

Parent topic: [CWOBJ](#)

CWOBJ2601I

ADD_SUFFIX_TO_VIEW_NAME=CWOBJ2601I: Add suffix {0} to stream query views deployed in partition {1}.

Explanation

The stream query is executed to a partitioned map set, so we need to add partition name suffix to view names.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ2602W

VIEW_TRANSFORMER_EXISTS=CWOBJ2602W: The view transformer {0} has already existed.

Explanation

The view transformer has already been added. This case should not happen.

User response

Please report to support team.

Parent topic: [CWOBJ](#)

CWOBJ2603W

STREAM_DESCRIPTOR_EXISTS=CWOBJ2603W: The stream descriptor {0} has already existed.

Explanation

The stream descriptor has already been added. This case should not happen.

User response

Please report to support team.

Parent topic: [CWOBJ](#)

CWOBJ2604I

STREAM_QUERY_JAR_NOT_IN_CLASSPATH=CWOBJ2604I: The stream query jar file is not in the class path.

Explanation

The stream query class is shipped in a separate jar file. The jar file is probably not in the classpath.

User response

If you intend to use stream query functions, add the stream query jar file in the classpath.

Parent topic: [CWOBJ](#)

CWOBJ2605E

STREAM_QUERY_LOGGER_ERROR=CWOBJ2605E: The stream query logger setting method introspection or invocation error: {0}

Explanation

The stream query logger setting method cannot be introspected or invoked.

User response

Please report to support team.

Parent topic: [CWOBJ](#)

CWOBJ2606W

VIEW_REMOVE_NON_EXISTING_ENTRY=CWOBJ2606W: Try to remove a non-existing entry for key {0}

Explanation

The entry does not exist in the stream query view map

User response

Please report to support team.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ2607E

STREAM_QUERY_SET_ACROSS_MAP_SET=CWOBJ2607E: The stream query set with name {0} contains maps from different map sets.

Explanation

A stream query set can only contain maps from one map set.

User response

Make sure a stream query set contain maps from only one map set.

Parent topic: [CWOBJ](#)

CWOBJ3001I

EM_SERVICE_STARTED_CWOBJ3001I=CWOBJ3001I: The ObjectGrid EntityManager service is available to process requests for ObjectGrid: {0} and container or server: {1}

Explanation

The EntityManager service can now accept requests.

User response

None

Parent topic: [CWOBJ](#)

CWOBJ3002I

EM_INIT_ENTITIES_CWOBJ3002I=CWOBJ3002I: Initializing entity metadata for ObjectGrid: {0}

Explanation

Entity metadata is being discovered and cached for use.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ3003I

EM_REGISTERED_CWOBJ3003I=CWOBJ3003I: Entity registered: {0}

Explanation

The specified entity metadata has been successfully bound to the ObjectGrid infrastructure.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ3004E

EM_REGISTER_EXCEPTION_CWOBJ3004E=CWOBJ3004E: An exception occurred while registering an entity: {0}

Explanation

An exception was detected when attempting to register an entity with the EntityManager service.

User response

Review the exception, resolve the error and retry the operation.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ3005I

EM_CREATING_INDEX_CWOBJ3005I=CWOBJ3005I: Creating index {0} for entity BackingMap {1}, attribute {2}

Explanation

An index was not explicitly defined for an entity BackingMap and was automatically created.

User response

None

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ3006E

EM_UNSUPPORTED_INDEX_TYPE_CWOBJ3006E=CWOBJ3006E: The defined MapIndexPlugin type is unsupported for index: {0}, BackingMap: {1}, attribute: {2}.

Explanation

An index was created for an entity BackingMap but is not compatible with the EntityManager service

User response

Change the MapIndexPlugin to use a supported index configuration.

Parent topic: [CWOBJ](#)

CWOBJ3007E

EM_LATE_REGISTRATION_CWOBJ3007E=CWOBJ3007E: Unable to register new entity {0} after ObjectGrid initialization has completed.

Explanation

New, non-subset entities must be defined prior to ObjectGrid initialization.

User response

Register the entities prior to initializing the ObjectGrid instance.

Parent topic: [CWOBJ](#)

CWOBJ3008E

EM_BACKINGMAP_REASSOCIATION_CWOBJ3008E=CWOBJ3008E: BackingMap {0} is associated with entity {1} and cannot be reassociated with entity {2}

Explanation

A BackingMap can only be associated with a single entity type and cannot be reassigned.

User response

Review the entity metadata definitions and choose a name that is not in use.

Parent topic: [CWOBJ](#)

CWOBJ3009E

EM_REPOSITORY_EXCEPTION_CWOBJ3009E=CWOBJ3009E: An exception occurred while communicating with the entity metadata repository: {0}

Explanation

An exception occurred while communicating with the entity metadata repository.

User response

Review the exception, resolve the error and retry the operation.

Parent topic: [CWOBJ](#)

CWOBJ3010E

EM_INVALID_MAPSET_CWOBJ3010E=CWOBJ3010E: All entity BackingMaps must be members of a MapSet with the name: "ENTITY_MAPSET".

Explanation

All entity BackingMaps must be defined in a single MapSet named ENTITY_MAPSET when using the EntityManager service in a clustered ObjectGrid.

User response

Identify the entity BackingMaps and add them to the ENTITY_MAPSET MapSet.

Parent topic: [CWOBJ](#)

CWOBJ3011E

EM_METADATA_LISTENER_EXCEPTION_CWOBJ3011E=CWOBJ3011E: Error creating entity metadata for entity {0} ({1}): {2}

Explanation

The specified entity metadata could not be created due to the specified exception.

User response

Review the exception, resolve the error and retry the operation.

Parent topic: [CWOBJ](#)

CWOBJ3012E

EM_MULTIPLE_MAPSETS_CWOBJ3012E=CWOBJ3012E: Invalid entity MapSet configuration. Only one Entity MapSet is allowed. Invalid MapSets: {0}

Explanation

Only one MapSet with entities is allowed per ObjectGrid configuration

User response

Consolidate all entities into a single MapSet and restart the server.

Parent topic: [CWOBJ](#)

CWOBJ3013E

EM_MULTIPLE_MAPSETS_CWOBJ3013E=CWOBJ3013E: The EntityMetadata repository is not available. Timeout threshold reached when trying to register the entity: {0}.

Explanation

The runtime could not register the defined entities with the metadata repository.

User response

Verify that there is at least one primary shard activated for the entity manager.

Parent topic: [CWOBJ](#)

CWOBJ3014I

AVAILABILITY_STATE_CHANGED_CWOBJ3014=CWOBJ3014I: The availability state has changed for {0}. The state is now {1}. It was previously {2}.

Explanation

The availability state for a shard has changed.

User response

None.

Parent topic: [CWOBJ](#)

CWOBJ3015E

EM_MISSING_MAPSET_CWOBJ3015E=CWOBJ3015E: Invalid Entity MapSet configuration. Unable to find MapSet that contains a BackingMap for {0}

Explanation

Each Entity needs a BackingMap of the same name.

User response

Verify that a BackingMap named {0} is defined in your configuration.

Parent topic: [CWOBJ](#)

CWOBJ3016E

EM_SCHEMA_MAPSET_CROSSOVER_CWOBJ3016E=CWOBJ3016E: Invalid entity MapSet configuration. Entity {0} should be present in MapSet {1} but is already exists in MapSet {2}.

Explanation

The entities for a logical schema must be contained in a single mapset.

User response

Make sure that entities do not contain references to other entities that exist in another mapset and restart the server.

Parent topic: [CWOBJ](#)

CWOB3101E

WB_LOADER_INITIALIZATION_FAILED_CWOB3101E=CWOB3101E: The write-behind loader of map {0} partition {1} initialization fails with exception {2}.

Explanation

The write-behind loader initialization fails. It could be the session cannot initialize or the write behind queue map does not exist.

User response

Please report to support team.

Parent topic: [CWOB](#)

CWOB3102E

WB_LOADER_FAILED_CWOB3102E=CWOB3102E: Loader fails to do a write-behind update to the database for map {0} partition {1}. The exception is {2}.

Explanation

Loader fails to do a write-behind update to the database. It could be that the database is updated by other applications.

User response

Look at the exception and take appropriate actions to compensate this failure.

Parent topic: [CWOB](#)

CWOBJ3103E

WB_LOADER_FAILED_CWOBJ3103E=CWOBJ3103E: The write-behind loader of map {0} partition {1} fails to complete a transaction. The exception is {2}.

Explanation

The write-behind loader fails to complete a transaction. See exception for more details.

User response

Look at the exception and take appropriate actions to compensate this failure.

Parent topic: [CWOBJ](#)

CWOBJ3104W

WB_LOADER_LOCKTIMEOUT_CWOBJ3104W=CWOBJ3104W: The write-behind loader of map {0} partition {1} gets a lock timeout exception when trying to flip the queue maps: {2}

Explanation

The write-behind loader gets a lock timeout exception when trying to flip the queue map states. See exception for more details.

User response

Please report to support team.

Parent topic: [CWOBJ](#)

CWOBJ3111E

CLIENT_LOADER_AGENT_FAIL_CWOBJ3111E=CWOBJ3111E: The client loader agent {0} execution fails with exception: {1}.

Explanation

The client loader agent execution fails. The agent will be re-executed for a certain number of times.

User response

None.

Parent topic: [CWOBJ](#)

CWOBJ3112I

DEFAULT_PERSISTENCE_UNIT_CWOBJ3112I=CWOBJ3112I: The JPA persistence unit name is not specified. The first persistence unit {0} defined in the persistence.xml is used.

Explanation

The user does not specify the JPA persistence unit name. ObjectGrid uses the first persistence unit defined in the persistence.xml

User response

If you do not like this default behavior, please specify a persistence unit name.

Parent topic: [CWOBJ](#)

CWOBJ3113E

AGENT_FAIL_CWOBJ3113E=CWOBJ3113E: The DataGrid agent {0} execution fails with a fatal exception: {1}.

Explanation

The DataGrid agent execution fails with a fatal non-retryable exception.

User response

Examine the exception and agent implementation for possible causes.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ3114E

AGENT_FAIL_RETRYABLE_CWOBJ3114E=CWOBJ3114E: The DataGrid agent {0} execution fails with a retryable exception: {1}.

Explanation

The DataGrid agent execution fails with a retryable exception. The agent will be re-executed for a certain number of times.

User response

None.

Parent topic: [CWOBJ](#)

CWOBJ3115E

UNEXPECTED_SHARD_STATE_CWOBJ3115E=CWOBJ3115E: The shard is expected in {0} state, but currently it is in {1} state. It might take a while for a shard to move to the target state.

Explanation

The shard state is not expected. It might take a while for a shard to move to the target state.

User response

None.

Parent topic: [CWOBJ](#)

CWOB3121E

TIME_BASED_DBUPDATE_AGENT_FAIL_CWOB3121E=CWOB3121E: The time-based database update agent fails with exception: {0}.

Explanation

The time-based database update agent execution fails. The agent will be re-executed for a certain number of times.

User response

None.

Parent topic: [CWOB](#)

CWOBJ3122E

TIME_BASED_DBUPDATE_FAIL_CWOBJ3122E=CWOBJ3122E: The time-based database update fails with exception: {0}.

Explanation

The time-based database update fails. The thread will retry later.

User response

None.

Parent topic: [CWOBJ](#)

CWOBJ3131E

JPA_TX_CALLBACK_NOT_FOUND_CWOBJ3131E=CWOBJ3131E: The JPATxCallback transaction callback plug-in cannot be found.

Explanation

When using JPALoader or JPAEntityLoader, the JPATxCallback transaction callback plug-in is expected to be configured on the ObjectGrid.

User response

Configure the JPATxCallback transaction callback plug-in on the ObjectGrid.

Parent topic: [CWOBJ](#)

CWOBJ3141W

NODEGROUP_NOT_SET_FOR_ZONE_SUPPORT_CWOBJ3141W=CWOBJ3141W: This ObjectGrid server is not associated with a zone. In order to start the ObjectGrid server in a zone, ensure that the server's node is within a node group whose name begins with the string ReplicationZone.

Explanation

A server's node must be within a nodegroup whose name begins with ReplicationZone in order for that server to be placed into an ObjectGrid zone.

User response

In order to start the ObjectGrid server in a zone, ensure that the server's node is within a node group whose name begins with the string ReplicationZone.

Parent topic: [CWOBJ](#)

CWOBJ6090E

EmptyIOR=CWOBJ6090E: IOR is null for this shard of partition ({0}).

Explanation

IOR is null for this shard.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6091E

NullSelectedTarget=CWOBJ6091E: Selected target for partition ({0}) is null.

Explanation

Selected target is null.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6092E

NullOrbContext=CWOBJ6092E: No orb context associated with this request for partition ({0}) with transaction ID ({1}), so routing cannot be made.

Explanation

The request does not have ObjectGrid orb context with it.

User response

contact IBM support

Parent topic: [CWOBJ](#)

CWOBJ6093E

NullRequest=CWOBJ6093E: Request is null.

Explanation

The request is null.

User response

contact IBM support

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6094E

NullResponse=CWOBJ6094E: Response is null.

Explanation

The response is null.

User response

contact IBM support

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6096E

SystemException=CWOBJ6096E: Client encountered system exception.

Explanation

Client encountered system exception; server may be down or network may break.

User response

System will do automatic retries.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6200I

RoutingTablePropagated=CWOBJ6200I: Server routing table is propagated among servers for partition ({0}).

Explanation

Server routing table is propagated among servers for the specified partition.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ6600I

NoTargetAvailable=CWOBJ6600I: No target is available for partition ({0}).

Explanation

No target is available.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6602I

NoRoutingTableAvailable=CWOBJ6602I: No routing table is available for partition ({0}).

Explanation

No routing table is available.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6620I

MapSetBootstrap=CWOBJ6620I: Bootstrap to get MapSet and Map for routing.

Explanation

Bootstrap to get MapSet and Map for routing.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ6621E

FailedMapSet=CWOBJ6621E: Cannot look up map set name for this map ({0}).

Explanation

Cannot look up map set name for this map.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6622E

InvalidPartition=CWOBJ6622E: Partition ({0}) is invalid.

Explanation

Partition is invalid.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6661I

ForwardNeeded=CWOBJ6661I: Forward is needed for partition ({0}) since this shard {1} does not have capacity to serve this request.

Explanation

Primary shard changed or data quality changed or previous routing is wrong.

User response

Router will automatically forward this request to correct server.

Parent topic: [CWOBJ](#)

CWOBJ6662I

MaxForward=CWOBJ6662I: Maximum number of forwards has reached for ({0}).

Explanation

System has done the maximum number of forwards but cannot find right target.

User response

Check your server status.

Parent topic: [CWOBJ](#)

CWOBJ6663I

RetryNeeded=CWOBJ6663I: Retry is needed for partition ({0}) since this shard {1} does not have capacity to serve this request.

Explanation

Shard is busy or shard is temporarily down or network is busy/down.

User response

System will do automatic retries.

Parent topic: [CWOBJ](#)

CWOBJ6664I

MaxRetries=CWOBJ6664I: Maximum number of retries has reached for ({0}).

Explanation

System has done the maximum number of retries but cannot get this request serviced.

User response

Check your server status.

Parent topic: [CWOBJ](#)

CWOBJ6666I

ConfigMaxForward=CWOBJ6666I: The configured maximum number of forwards is {0}.

Explanation

configured maximum number of forwards.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6668I

ConfigMaxRetries=CWOBJ6668I: The configured maximum number of retries is {0}.

Explanation

configured maximum number of retries.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6800I

ClientRTUpdated=CWOBJ6800I: Client received new routing table for partition ({0}) with epoch ({1}).

Explanation

Client received new routing table with new epoch.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6860I

DQNotMatch=CWOBJ6860I: The data quality of this shard does not meet the requirement of this request for partition ({0}). The router will automatically forward this request to shards with better data quality.

Explanation

The data quality of this shard does not meet the requirement of this request. The router will automatically forward this request to shards with better data quality.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ6866I

ClientDQ=CWOBJ6866I: Client required quality of data for this request is {0}

Explanation

Client required data quality.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6900I

RemoteZoneLink=CWOBJ6900I: Remote zone ({0}) joined; routing table is linked.

Explanation

Received request to join remote zone into this zone; routing table is linked.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6902I

RemoteZoneLeave=CWOBJ6902I: Remote zone ({0}) left; routing table is updated.

Explanation

Received request to remove remote zone; routing table is updated.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6904I

PrimaryZone=CWOBJ6904I: Zone ({0}) is set as the primary zone of partition ({1}).

Explanation

This zone is set as the primary zone for the specified partition.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ6906I

PrimaryZoneChange=CWOBJ6906I: Primary zone changed from Zone ({0}) to Zone ({1}) for partition ({2}).

Explanation

The primary zone changed for the specified partition.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7000I

ServerAgentStart=CWOBJ7000I: ObjectGrid server agent started successfully for server ({0}).

Explanation

ObjectGrid server agent started successfully.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7006I

DynamicPort=CWOBJ7006I: ObjectGrid server agent generated dynamic port {0}.

Explanation

ObjectGrid server agent generated dynamic port for ha manager.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ7008I

DynamicVerifyPort=CWOBJ7008I: ObjectGrid server agent verified dynamically generated port {0}.

Explanation

ObjectGrid server agent verified dynamically generated port for ha manager.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7010E

ServerAgentMalfunction=CWOBJ7010E: ObjectGrid server agent encountered failure.

Explanation

ObjectGrid server agent encountered failures.

User response

turn on trace and find the source of failures.

Parent topic: [CWOBJ](#)

CWOBJ7200I

DeadServer=CWOBJ7200I: Detected the failure of server ({0}) in core group ({1}).

Explanation

Server failure is detected and reported to catalog server.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7201I

NewServer=CWOBJ7201I: Detected the addition of new server ({0}) in core group ({1}).

Explanation

New server is detected and reported to catalog server.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7203I

NewLeader=CWOBJ7203I: Leader changed. New leader ({0}) is elected in core group ({1}) and reported to catalog server.

Explanation

New leader is elected and reported to catalog server.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7204I

CurrLeader=CWOBJ7204I: The current leader for core group ({1}) is ({0}).

Explanation

Report the current leader.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ7206I

ChangeLeader=CWOBJ7206I: New leader is ({0}). Old leader was ({1}).

Explanation

New leader is elected to replace old leader.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ7210I

RoutingTableUpdate=CWOBJ7210I: Updated routing table for partition ({0}).

Explanation

Routing table changed; reported this change.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7700I

PeerManagerStart=CWOBJ7700I: Peer Manager service started successfully in server ({0}) with core group ({1}).

Explanation

Peer Manager service started successfully.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7701I

PeerManagerStop=CWOBJ7701I: Peer Manager service stopped.

Explanation

Peer Manager service stopped successfully.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ7710I

PeerManagerLeaderToAll=CWOBJ7710I: Peer Manager received broadcasting message from leader.

Explanation

Peer Manager received message from leader.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ7712I

PeerManagerAllToLeader=CWOBJ7712I: Leader gathers messages from all members.

Explanation

Leader gathers messages from all members.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ7800I

Start_HAController=CWOBJ7800I: Start ObjectGrid HA Controller with core group ({0}), host ({1}), and port ({2}).

Explanation

Start ObjectGrid High Availability Controller.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7801I

AddMember_HAController=CWOBJ7801I: Add new server member ({0}) as instructed by catalog server.

Explanation

Add new server member.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ7802I

removeMember_HAController=CWOBJ7802I: Remove server member ({0}) as instructed by catalog server.

Explanation

Remove server member.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ7810E

Failed_Start_HAController=CWOBJ7810E: Failed to start ObjectGrid HA Controller.

Explanation

Failed to start ObjectGrid High Availability Controller.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ7811E

Failed_AddMember_HAController=CWOBJ7811E: Failed to add new server member ({0}) as instructed by catalog server.

Explanation

Failed to add new server member.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7812E

Failed_removeMember_HAController=CWOBJ7812E: Failed to remove server member ({0}) as instructed by catalog server.

Explanation

Failed to remove new server member.

User response

none

Parent topic: [CWOBJ](#)

CWOBJ7820E

InconsistentView_HAController=CWOBJ7820E: Members are in inconsistent view for core group ({0}).

Explanation

Servers have different views in this core group, caused by synchronization failure.

User response

none

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ8000I

Register_CWOBJ8000=CWOBJ8000I: Registration is successful with zone ({0}) and coregroup of ({1}).

Explanation

This jvm is successfully registered with zone and catalog server successfully allocates a coregroup for this jvm.

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ8009E

Failed_Register_CWOBJ8009=CWOBJ8009E: Registration failed with zone ({0}).

Explanation

This jvm failed to register with Catalog server.

User response

None.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ8101I

StandbyCatalogServerCreated_CWOBJ8101=CWOBJ8101I: Notify that standby catalog server is created with domain= {0} and with IOR= {1}

Explanation

Standby Catalog Server is created and notified.

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ8102I

MasterCatalogServerCreated_CWOBJ8102=CWOBJ8102I: Notify that master catalog server is created with domain= {0} and with IOR= {1}

Explanation

Master Catalog Server is created and notified.

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ8103I

NullORB_CWOBJ8103=CWOBJ8103I: ORB is null at this time, automatic retry later, Catalog server cluster is {0}

Explanation

Master Catalog Server is created and notified.

User response

none.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ8106I

MasterCatalogServerActivated_CWOBJ8106=CWOBJ8106I: master catalog server cluster activated with cluster {0}

Explanation

Master catalog server is activated.

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ8108I

ResentStandbyCatalogServer_CWOBJ8108=CWOBJ8108I: Re-send standby catalog server on the request of master catalog server with domain= {0} and IOR= {1}

Explanation

Tell master catalog server who are replica catalog servers

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ8109I

UpdateCatalogServerCluster_CWOBJ8109=CWOBJ8109I: Updated catalog server cluster {0} from server {1} with entry {2}

Explanation

Tell master catalog server who are replica catalog servers

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ8201I

BatchInfo_CWOBJ8201=CWOBJ8201I: BatchProcessor is processing jobs for {0}

Explanation

Processing batch jobs for this coregroup.

User response

none.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ8401I

WaitForReplica_CWOBJ8401=CWOBJ8401I: Waiting for a server replica to be started. Start another server(s) immediately.

Explanation

Waiting for a server replica to be started. Start another server(s) immediately.

User response

none.

Parent topic: [CWOBJ](#)

CWOBJ8601I

PeerServers_CWOBJ8601=CWOBJ8601I: PeerManager found peers of size {0}

Explanation

Processing batch jobs for this coregroup.

User response

none.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ8700E

OutOfRange_CWOBJ8700=CWOBJ8700E: Custom partitioning is out of range for map ({0}).

Explanation

Custom partitioning must return an integer between 0 and the maximum number of partition for this map.

User response

Change CustomPartitionID implementation of custom application to ensure that the return value of partition id is within the range.

Parent topic: [CWOBJ](#)

[Terms of Use](#) | [Feedback](#)

CWOBJ9000I

ENGLISH_ONLY_INFO_MESSAGE_CWOBJ9000=CWOBJ9000I: This message is an English-only Informational message: {0}.

Explanation

This informational message is not translated.

User response

See message for details.

Parent topic: [CWOBJ](#)

CWOBJ9001W

ENGLISH_ONLY_WARN_MESSAGE_CWOBJ9001=CWOBJ9001W: This message is an English-only Warning message: {0}.

Explanation

This warning message is not translated.

User response

See message for details.

Parent topic: [CWOBJ](#)

CWOBJ9002E

ENGLISH_ONLY_ERROR_MESSAGE_CWOBJ9002=CWOBJ9002E: This message is an English only Error message: {0}.

Explanation

This error message is not translated.

User response

See message for details.

Parent topic: [CWOBJ](#)

CWOBJ9999I

RegistrationSuccess_CWOBJ9999=CWOBJ9999I: This jvm ({0}) is registered successfully with CatalogServer. The CatalogServer will instruct what functions this jvm will take.

Explanation

Catalogserver successfully gets the information for this jvm and will instruct this jvm to function.

User response

none

Parent topic: [CWOBJ](#)

CWODB

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWODB0001I](#)

WDBO_JDBCStatsAggInitialized=CWODB0001I: JDBC Stats Aggregator initialized

[CWODB0002E](#)

WDBO_JDBCStatsAggInitializeError=CWODB0002E: Exception caught while initializing the JDBC Stats Aggregator: {0}

[CWODB0003E](#)

WDBO_JDBCStatsAggComputeError=CWODB0003E: Exception caught while computing in the JDBC Stats Aggregator: {0}

[CWODB0004I](#)

WDBO_JDBCStatsAggRefreshed=CWODB0004I: JDBC Stats Aggregator refreshed

[CWODB0005E](#)

WDBO_JDBCStatsAggRefreshError=CWODB0005E: Exception caught while refreshing the JDBC Stats Aggregator: {0}

[CWODB0010E](#)

ERR_ConfigSvcInitServer=CWODB0010E: Unable to get config service

[CWODB0011E](#)

ERR_ConfigSvcInitClient=CWODB0011E: Unable to get config service: {0}

[CWODB0012E](#)

ERR_URLParsing=CWODB0012E: Could not parse URL: {0}

[CWODB0013W](#)

WARN_MultipleCells=CWODB0013W: Deployment manager is managing multiple cells

[CWODB0014E](#)

ERR_ConfigSvcAccess=CWODB0014E: Error accessing Config Service: {0}

[CWODB0015E](#)

ERR_ConfigSvcLookup=CWODB0015E: Error during config lookup: {0}

[CWODB0016E](#)

ERR_CellMapNotBuilt=CWODB0016E: Cell map not built

[CWODB0017I](#)

INFO_CellMapOk=CWODB0017I: Cell map initialized

[CWODB0018E](#)

ERR_CellMapNull=CWODB0018E: Cell map is uninitialized

[CWODB0019E](#)

ERR_NodeNotFound=CWODB0019E: Node {0} not found

[CWODB0020E](#)

ERR_ServerNotFound=CWODB0020E: Server {0} not found

[CWODB0021E](#)

ERR_DBNotFound=CWODB0021E: Database {0} not found

CWODB0001I

WDBO_JDBCStatsAggInitialized=CWODB0001I: JDBC Stats Aggregator initialized

Explanation

The JDBC Stats Aggregator was initialized successfully.

User response

No action required.

Parent topic: [CWODB](#)

CWODB0002E

WDBO_JDBCStatsAggInitializeError=CWODB0002E: Exception caught while initializing the JDBC Stats Aggregator: {0}

Explanation

Something went wrong when while initializing the JDBC Stats Aggregator.

User response

Report program defect to IBM.

Parent topic: [CWODB](#)

CWODB0003E

WDBO_JDBCStatsAggComputeError=CWODB0003E: Exception caught while computing in the JDBC Stats Aggregator: {0}

Explanation

Something went wrong when computing the aggregate in the JDBC Stats Aggregator.

User response

Investigate the cause of the failure.

Parent topic: [CWODB](#)

CWODB0004I

WDBO_JDBCStatsAggRefreshed=CWODB0004I: JDBC Stats Aggregator refreshed

Explanation

The JDBC Stats Aggregator was refreshed successfully.

User response

No action required.

Parent topic: [CWODB](#)

[Terms of Use](#) | [Feedback](#)

CWODB0005E

WDBO_JDBCStatsAggRefreshError=CWODB0005E: Exception caught while refreshing the JDBC Stats Aggregator:
{0}

Explanation

Something went wrong when while refreshing the JDBC Stats Aggregator.

User response

Report program defect to IBM.

Parent topic: [CWODB](#)

CWODB0010E

ERR_ConfigSvcInitServer=CWODB0010E: Unable to get config service

Explanation

Could not obtain config service object. ConfigServiceFactory may not be initialized.

User response

Report program defect to IBM.

Parent topic: [CWODB](#)

[Terms of Use](#) | [Feedback](#)

CWODB0011E

ERR_ConfigSvcInitClient=CWODB0011E: Unable to get config service: {0}

Explanation

Could not obtain the config service object from the AdminClient proxy. There may be an error communicating to the deployment manager.

User response

Report program defect to IBM.

Parent topic: [CWODB](#)

CWODB0012E

ERR_URLParsing=CWODB0012E: Could not parse URL: {0}

Explanation

There was an error parsing the specified URL.

User response

Check the syntax of the specified URL. If it is correct, report the problem to IBM.

Parent topic: [CWODB](#)

[Terms of Use](#) | [Feedback](#)

CWODB0013W

WARN_MultipleCells=CWODB0013W: Deployment manager is managing multiple cells

Explanation

The deployment manager configuration contains information for multiple cells. Only the first cell will be mapped.

User response

Contact technical support.

Parent topic: [CWODB](#)

[Terms of Use](#) | [Feedback](#)

CWODB0014E

ERR_ConfigSvcAccess=CWODB0014E: Error accessing Config Service: {0}

Explanation

Could not access the config service needed to process the cell configuration. There may be an error communicating to the deployment manager.

User response

Contact technical support.

Parent topic: [CWODB](#)

CWODB0015E

ERR_ConfigSvcLookup=CWODB0015E: Error during config lookup: {0}

Explanation

There was an unexpected error reading the cell configuration.

User response

Contact technical support.

Parent topic: [CWODB](#)

[Terms of Use](#) | [Feedback](#)

CWODB0016E

ERR_CellMapNotBuilt=CWODB0016E: Cell map not built

Explanation

The cell configuration was either not accessible or empty.

User response

Contact technical support.

Parent topic: [CWODB](#)

[Terms of Use](#) | [Feedback](#)

CWODB0017I

INFO_CellMapOk=CWODB0017I: Cell map initialized

Explanation

The cell configuration was read successfully.

User response

No action required.

Parent topic: [CWODB](#)

[Terms of Use](#) | [Feedback](#)

CWODB0018E

ERR_CellMapNull=CWODB0018E: Cell map is uninitialized

Explanation

The cell configuration has not been initialized. Must do a buildCellMap() first.

User response

Contact technical support.

Parent topic: [CWODB](#)

[Terms of Use](#) | [Feedback](#)

CWODB0019E

ERR_NodeNotFound=CWODB0019E: Node {0} not found

Explanation

Attempt was made to find information on the named node, but it is not a known database node in this cell.

User response

No action required.

Parent topic: [CWODB](#)

CWODB0020E

ERR_ServerNotFound=CWODB0020E: Server {0} not found

Explanation

Attempt was made to find information on the named server, but it is not a known database server in this cell.

User response

No action required.

Parent topic: [CWODB](#)

[Terms of Use](#) | [Feedback](#)

CWODB0021E

ERR_DBNotFound=CWODB0021E: Database {0} not found

Explanation

Attempt was made to find information on the named database, but it is not a known database in this cell.

User response

No action required.

Parent topic: [CWODB](#)

CWODR

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWODR0001E](#)

CWODR_Exception=CWODR0001E: Exception: {0}.

[CWODR0002E](#)

CWODR_NoFMSFound=CWODR0002E: The filter management service was not found.

[CWODR0003E](#)

CWODR_NoTargetAvail=CWODR0003E: No servers were available for selection satisfying conditions: {0}

[CWODR0004E](#)

CWODR_MultiTargets=CWODR0004E: Multiple clusters are available for selection: {0}

[CWODR0005E](#)

CWODR_StartQuiesce=CWODR0005E: On Demand Router quiesce started.

[CWODR0007E](#)

CWODR_EndQuiesce=CWODR0007E: On Demand Router quiesce completed.

[CWODR0008E](#)

CWODR_StartServerQuiesce=CWODR0008E: Quiesce started for server {0} on node {1}.

[CWODR0009E](#)

CWODR_EndServerQuiesce=CWODR0009E: Quiesce completed for server {0} on node {1}.

[CWODR0010E](#)

CWODR_CannotCancelQuiesce=CWODR0010E: Quiesce cannot be cancelled for server {0} on node {1}.

[CWODR0011E](#)

CWODR_QuiesceInProgress=CWODR0011E: Quiesce is already in progress for server {0} on node {1}.

CWODR0001E

CWODR_Exception=CWODR0001E: Exception: {0}.

Explanation

An unexpected exception occurred.

User response

Report program defect to IBM.

Parent topic: [CWODR](#)

[Terms of Use](#) | [Feedback](#)

CWODR0002E

CWODR_NoFMSFound=CWODR0002E: The filter management service was not found.

Explanation

The filter management service was not found, server error.

User response

Maybe due to a failed install, if that is not the case, report problem to customer support.

Parent topic: [CWODR](#)

[Terms of Use](#) | [Feedback](#)

CWODR0003E

CWODR_NoTargetAvail=CWODR0003E: No servers were available for selection satisfying conditions: {0}

Explanation

Servers or applications may not have been started or failed to start which could service the request.

User response

Check that the appropriate servers and applications have been started.

Parent topic: [CWODR](#)

CWODR0004E

CWODR_MultiTargets=CWODR0004E: Multiple clusters are available for selection: {0}

Explanation

Multiple clusters are available to service a request and the multi-cluster load balance policy is "Error" if multiple clusters are available.

User response

Either change the multi-cluster load balance policy or ensure that multiple clusters can not service a request or message.

Parent topic: [CWODR](#)

[Terms of Use](#) | [Feedback](#)

CWODR0005E

CWODR_StartQuiesce=CWODR0005E: On Demand Router quiesce started.

Explanation

The On Demand Router received a start quiesce request.

User response

None.

Parent topic: [CWODR](#)

[Terms of Use](#) | [Feedback](#)

CWODR0007E

CWODR_EndQuiesce=CWODR0007E: On Demand Router quiesce completed.

Explanation

The On Demand Router quiesce is completed.

User response

None.

Parent topic: [CWODR](#)

CWODR0008E

CWODR_StartServerQuiesce=CWODR0008E: Quiesce started for server {0} on node {1}.

Explanation

The server received a start quiesce request.

User response

None.

Parent topic: [CWODR](#)

[Terms of Use](#) | [Feedback](#)

CWODR0009E

CWODR_EndServerQuiesce=CWODR0009E: Quiesce completed for server {0} on node {1}.

Explanation

Server quiesce is completed.

User response

None.

Parent topic: [CWODR](#)

[Terms of Use](#) | [Feedback](#)

CWODR0010E

CWODR_CannotCancelQuiesce=CWODR0010E: Quiesce cannot be cancelled for server {0} on node {1}.

Explanation

Phase one of server quiesce is completed and quiesce cannot be cancelled.

User response

None.

Parent topic: [CWODR](#)

[Terms of Use](#) | [Feedback](#)

CWODR0011E

CWODR_QuiesceInProgress=CWODR0011E: Quiesce is already in progress for server {0} on node {1}.

Explanation

Server quiesce has already been started.

User response

None.

Parent topic: [CWODR](#)

[Terms of Use](#) | [Feedback](#)

CWPFC

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWPFC0001E](#)

ERR_MISSING_KEY=CWPFC0001E: Message key {0} was not found in any searched resource bundles.

[CWPFC0003E](#)

ERR_NAMINGEX=CWPFC0003E: NamingException looking up home for startup bean with jndi name {0}, exception was {1}

[CWPFC0004E](#)

ERR_CREATEERROR=CWPFC0004E: Cannot create partition bean named {0}, exception was {1}

[CWPFC0005E](#)

ERR_FORCESTOP=CWPFC0005E: Partition bean named {0} forces app stop

[CWPFC0006E](#)

ERR_INVOKE=CWPFC0006E: Exception calling partition bean named {0}, exception was {1}

[CWPFC0007E](#)

ERR_INITERR=CWPFC0007E: Problem initializing Partitioning Facility service {0}

[CWPFC0009E](#)

FATAL_NO_HAM_POLICY=CWPFC0009E: No HAManager policy defined for partition groups

[CWPFC0011E](#)

FATAL_NO_SERVICE=CWPFC0011E: Service {0} not found

[CWPFC0028I](#)

ERROR_PROBLEM_ACTIVATING_PARTITION=CWPFC0028I: Application {0} Partition {1} Cannot activate, exception: {2}

[CWPFC0029E](#)

ERROR_NULL_PARTITION_NAME=CWPFC0029E: Application {0} tried to make a partition with either a null name or classification.

[CWPFC0030E](#)

ERROR_NULL_PARTITION=CWPFC0030E: Application {0} returned a NULL PartitionDefinition from its getPartitions method.

[CWPFC0031E](#)

ERROR_NULL_PARTITION_LIST=CWPFC0031E: Application {0} returned a NULL PartitionDefinition[] from its getPartitions method.

[CWPFC0032E](#)

ERROR_APP_PROBLEM_ACTIVATING_PARTITION=CWPFC0032E: Application {0} Partition {1} Application cannot activate.

[CWPFC0033E](#)

ERROR_APP_PROBLEM_TO_MANY_PARTITIONED_BEANS=CWPFC0033E: Partitions for {1} in Application {0} will not be activated. Only one partitioned bean per J2EE Application is allowed.

[CWPFC9999E](#)

FATAL_UNEX_EXCEPT=CWPFC9999E: Unexpected Exception Occurred: {0}

CWPFC0001E

ERR_MISSING_KEY=CWPFC0001E: Message key {0} was not found in any searched resource bundles.

Explanation

A key was passed into the Messages class to resolve to a string, but a properties file for the locale could not be found and the appropriate message could not be retrieved.

User response

Determine why the message catalog could not be found.

Parent topic: [CWPFC](#)

CWPFC0003E

ERR_NAMINGEX=CWPFC0003E: NamingException looking up home for startup bean with jndi name {0}, exception was {1}

Explanation

An unexpected exception occurred looking up the home for the startup bean.

User response

Verify the EAR deploy worked correctly and that the beans were deployed.

Parent topic: [CWPFC](#)

CWPFC0004E

ERR_CREATEERROR=CWPFC0004E: Cannot create partition bean named {0}, exception was {1}

Explanation

An unexpected exception occurred creating an instance of the partition bean.

User response

Verify the EAR containing the partition bean deployed correctly.

Parent topic: [CWPFC](#)

[Terms of Use](#) | [Feedback](#)

CWPFC0005E

ERR_FORCESTOP=CWPFC0005E: Partition bean named {0} forces app stop

Explanation

The specified partition bean start method returned false causing the EAR start to be aborted.

User response

Determine why the partition bean returned false and correct the problem allowing the application to start.

Parent topic: [CWPFC](#)

[Terms of Use](#) | [Feedback](#)

CWPFC0006E

ERR_INVOKE=CWPFC0006E: Exception calling partition bean named {0}, exception was {1}

Explanation

An unexpected exception occurred calling the start or stop method of the startup bean.

User response

Verify why the method throws an exception.

Parent topic: [CWPFC](#)

[Terms of Use](#) | [Feedback](#)

CWPFC0007E

ERR_INITERR=CWPFC0007E: Problem initializing Partitioning Facility service {0}

Explanation

The Partitioning Facility service failed to initialize.

User response

Verify the application server installation.

Parent topic: [CWPFC](#)

[Terms of Use](#) | [Feedback](#)

CWPFC0009E

FATAL_NO_HAM_POLICY=CWPFC0009E: No HAManager policy defined for partition groups

Explanation

Add a 1 of N rule for type=WPF_partition to rules.xml

User response

Please add the policy to the rules.xml file on all cluster nodes.

Parent topic: [CWPFC](#)

[Terms of Use](#) | [Feedback](#)

CWPFC0011E

FATAL_NO_SERVICE=CWPFC0011E: Service {0} not found

Explanation

WebSphere was not installed correctly, certain system services cannot be found. Please reinstall.

User response

Please reinstall WebSphere.

Parent topic: [CWPFC](#)

[Terms of Use](#) | [Feedback](#)

CWPFC0028I

ERROR_PROBLEM_ACTIVATING_PARTITION=CWPFC0028I: Application {0} Partition {1} Cannot activate, exception: {2}

Explanation

The partitionLoadEvent threw an exception.

User response

Debug your applicatio and fix the problem.

Parent topic: [CWPFC](#)

CWPFC0029E

ERROR_NULL_PARTITION_NAME=CWPFC0029E: Application {0} tried to make a partition with either a null name or classification.

Explanation

A null string was passed to PartitionManager.createPartitionDefinition

User response

Fix application.

Parent topic: [CWPFC](#)

CWPFC0030E

ERROR_NULL_PARTITION=CWPFC0030E: Application {0} returned a NULL PartitionDefinition from its getPartitions method.

Explanation

A null element is present in the array returned from getPartitions.

User response

Fix application.

Parent topic: [CWPFC](#)

CWPFC0031E

ERROR_NULL_PARTITION_LIST=CWPFC0031E: Application {0} returned a NULL PartitionDefinition[] from its getPartitions method.

Explanation

A null return value was returned from getPartitions.

User response

Fix application.

Parent topic: [CWPFC](#)

CWPFC0032E

ERROR_APP_PROBLEM_ACTIVATING_PARTITION=CWPFC0032E: Application {0} Partition {1} Application cannot activate.

Explanation

The partitionLoadEvent returned false

User response

Verify why the application returned false.

Parent topic: [CWPFC](#)

CWPFC0033E

ERROR_APP_PROBLEM_TO_MANY_PARTITIONED_BEANS=CWPFC0033E: Partitions for {1} in Application {0} will not be activated. Only one partitioned bean per J2EE Application is allowed.

Explanation

Partitioned J2EE Application can contain only a single partitioned bean.

User response

The bean should be removed from the application.

Parent topic: [CWPFC](#)

CWPFC9999E

FATAL_UNEX_EXCEPT=CWPFC9999E: Unexpected Exception Occurred: {0}

Explanation

An unexpected exception occurred.

User response

Verify the installation and try again. If this persists then report to IBM.

Parent topic: [CWPFC](#)

[Terms of Use](#) | [Feedback](#)

CWPFR

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

CWPFR0001I

INFO_CREATING_PARTITION_CLUSTER=CWPFR0001I: Creating partition routing cluster for partition {0} .

CWPFR0002I

INFO_CREATED_PARTITION_CLUSTER=CWPFR0002I: Partition routing cluster for partition {0} is successfully created and open for e-Business at {1}.

CWPFR0037E

ERR_CANNOT_FIND_PARTITION_CLUSTER=CWPFR0037E: Partition cluster cannot be found for partition {0}.

CWPFR0038E

ERR_REMOVE_PARTITION_CLUSTER=CWPFR0038E: Cannot remove partition cluster for partition {0}.

CWPFR0039E

ERR_REMOVE_PARTITION_IOR=CWPFR0039E: Error when remove partition member IOR for partition {0}.

CWPFR0073E

ERR_PARTITION_IOR_NULL=CWPFR0073E: Partition routing member gets null IOR.

CWPFR0081E

FATAL_WPFWLM_ADAPTER=CWPFR0081E: WPFWLM adapter cannot be started

CWPFR0097E

FATAL_WPFWLM_SERVICE=CWPFR0097E: Component WPFWLM Service cannot be started

CWPFR1111E

ERR_MISSING_KEY=CWPFR1111E: Message key {0} was not found in any searched resource bundles.

CWPFR2001E

FATAL_WPF_INVALID_PARTITION_ID=CWPFR2001E: WPF invalid partition id.

CWPFR2002E

FATAL_WPF_INVALID_CONTEXT_NAME=CWPFR2002E: WPF invalid context name

CWPFR6010E

FATAL_PARTITION_ADAPTER_CANNOT_FOUND=CWPFR6010E: Partition WLM cannot find adapter.

CWPFR9999E

ERR_PARTITION_CLUSTER_CREATION=CWPFR9999E: Partition cluster creation error {0}

CWPFR0001I

INFO_CREATING_PARTITION_CLUSTER=CWPFR0001I: Creating partition routing cluster for partition {0} .

Explanation

Routable partition cluster is in creating.

User response

User doesn't need to do anything.

Parent topic: [CWPFR](#)

[Terms of Use](#) | [Feedback](#)

CWPFR0002I

INFO_CREATED_PARTITION_CLUSTER=CWPFR0002I: Partition routing cluster for partition {0} is successfully created and open for e-Business at {1}.

Explanation

Routable partition cluster is created.

User response

This partitioned cluster is routable now.

Parent topic: [CWPFR](#)

CWPFR0037E

ERR_CANNOT_FIND_PARTITION_CLUSTER=CWPFR0037E: Partition cluster cannot be found for partition {0}.

Explanation

Cannot find partition cluster.

User response

trace

Parent topic: [CWPFR](#)

[Terms of Use](#) | [Feedback](#)

CWPFR0038E

ERR_REMOVE_PARTITION_CLUSTER=CWPFR0038E: Cannot remove partition cluster for partition {0}.

Explanation

Error occurs when remove partition cluster.

User response

trace.

Parent topic: [CWPFR](#)

CWPFR0039E

ERR_REMOVE_PARTITION_IOR=CWPFR0039E: Error when remove partition member IOR for partition {0}.

Explanation

Error occurs when remove partition member IOR.

User response

trace.

Parent topic: [CWPFR](#)

CWPFR0073E

ERR_PARTITION_IOR_NULL=CWPFR0073E: Partition routing member gets null IOR.

Explanation

Partition routing member gets null ior

User response

Trace orb

Parent topic: [CWPFR](#)

[Terms of Use](#) | [Feedback](#)

CWPFR0081E

FATAL_WPFWLM_ADAPTER=CWPFR0081E: WPFWLM adapter cannot be started

Explanation

Startup error for WPFWLM adapter

User response

trace to see if object adapter or stub have problems

Parent topic: [CWPFR](#)

CWPFR0097E

FATAL_WPFWLM_SERVICE=CWPFR0097E: Component WPFWLM Service cannot be started

Explanation

Component startup error for WPFWLMService

User response

check if other components fail and check if XDClusterAdapter is started.

Parent topic: [CWPFR](#)

[Terms of Use](#) | [Feedback](#)

CWPFR1111E

ERR_MISSING_KEY=CWPFR1111E: Message key {0} was not found in any searched resource bundles.

Explanation

A key was passed into the Messages class to resolve to a string, but a properties file for the locale could not be found and the appropriate message could not be retrieved.

User response

Determine why the message catalog could not be found.

Parent topic: [CWPFR](#)

CWPFR2001E

FATAL_WPF_INVALID_PARTITION_ID=CWPFR2001E: WPF invalid partition id.

Explanation

Partition id is invalid.

User response

report to IBM support.

Parent topic: [CWPFR](#)

CWPFR2002E

FATAL_WPF_INVALID_CONTEXT_NAME=CWPFR2002E: WPF invalid context name

Explanation

Context name is invalid.

User response

none.

Parent topic: [CWPFR](#)

[Terms of Use](#) | [Feedback](#)

CWPFR6010E

FATAL_PARTITION_ADAPTER_CANNOT_FOUND=CWPFR6010E: Partition WLM cannot find adapter.

Explanation

Partition pWLM Adapter is not started or WPFWLMSERVICE component is not started.

User response

none.

Parent topic: [CWPFR](#)

CWPFR9999E

ERR_PARTITION_CLUSTER_CREATION=CWPFR9999E: Partition cluster creation error {0}

Explanation

Partition cluster is not created right.

User response

none.

Parent topic: [CWPFR](#)

[Terms of Use](#) | [Feedback](#)

CWPKI

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

CWPKI0001I

ssl.init.startinit.CWPKI0001I=CWPKI0001I: SSL service is initializing the configuration

CWPKI0002I

ssl.init.startcomplete.CWPKI0002I=CWPKI0002I: SSL service initialization completed successfully

CWPKI0003I

ssl.init.svcstart.CWPKI0003I=CWPKI0003I: SSL service is starting

CWPKI0004I

ssl.init.svcstartcomplete.CWPKI0004I=CWPKI0004I: SSL service started successfully

CWPKI0005I

ssl.init.startfail.CWPKI0005I=CWPKI0005I: SSL service initialization failed

CWPKI0006E

ssl.init.mbeanerror.CWPKI0006E=CWPKI0006E: Error creating or registering {0} mBean. The exception is {1}

CWPKI0007I

ssl.init.svcstartfail.CWPKI0007I=CWPKI0007I: SSL service failed to start successfully

CWPKI0008E

ssl.init.error.CWPKI0008E=CWPKI0008E: Error during SSL initialization. The exception is {0}.

CWPKI0009E

ssl.init.nullsecobject.CWPKI0009E=CWPKI0009E: Cannot create security object during initialization.

CWPKI0010E

ssl.init.nullprocesstype.CWPKI0010E=CWPKI0010E: Cannot obtain the WAS process type during initialization.

CWPKI0011E

ssl.loadresource.error.CWPKI0011E=CWPKI0011E: Failed to load {0} resource from cell. The exception is {1}

CWPKI0012I

ssl.fips.enabled.CWPKI0012I=CWPKI0012I: FIPS is enabled.

CWPKI0013W

ssl.checkProviderList.warning.CWPKI0013W=CWPKI0013W: FIPS is enabled but the IBMJCEFIPS provider is not active in the java.security file. To ensure FIPS algorithms usage for all WAS process types, uncomment the IBMJCEFIPS provider in the java.security file, ahead of the IBMJCE, and renumber the provider list in sequential order.

CWPKI0014I

ssl.init.ssldm.init.CWPKI0014I=CWPKI0014I: The SSL component's FFDC Diagnostic Module {0} registered successfully: {1}.

CWPKI0015E

ssl.stop.error.CWPKI0015E=CWPKI0015E: Error stopping SSL component. The exception is {0}.

CWPKI0016W

ssl.expiration.warning.CWPKI0016W=CWPKI0016W: The certificate with alias {0} from keyStore {1} will be expired in {2} days.

CWPKI0017E

ssl.expiration.expired.CWPKI0017E=CWPKI0017E: The certificate with alias {1} from keyStore {2} is expired.

CWPKI0018W

ssl.keystore.type.invalid.CWPKI0018W=CWPKI0018W: The keystore type of {0} is not valid for SSL config alias {1}.

CWPKI0019E

ssl.client.config.parse.CWPKI0019E=CWPKI0019E: Error parsing the SSL client configuration file {0}. The error returned is {1}.

CWPKI0020E

ssl.load.trustmanager.error.CWPKI0020E=CWPKI0020E: Error loading custom trust manager class {0}. The exception message is {1}.

CWPKI0021E

ssl.load.keymanager.error.CWPKI0021E=CWPKI0021E: Error loading custom key manager class {0}. The exception message is {1}.

CWPKI0022E

ssl.client.handshake.error.CWPKI0022E=CWPKI0022E: SSL HANDSHAKE FAILURE: A signer with SubjectDN "{0}" was sent from target host:port "{1}". The signer may need to be added to local trust store "{2}" located in SSL configuration alias "{3}" loaded from SSL configuration file "{4}". The extended error message from the SSL handshake exception is: "{5}".

CWPKI0023E

ssl.client.alias.not.found.CWPKI0023E=CWPKI0023E: The certificate alias "{0}" specified by the property com.ibm.ssl.keyStoreClientAlias is not found in KeyStore "{1}".

CWPKI0024E

ssl.server.alias.not.found.CWPKI0024E=CWPKI0024E: The certificate alias "{0}" specified by the property com.ibm.ssl.keyStoreServerAlias is not found in KeyStore "{1}".

CWPKI0025E

ssl.load.https.stream.handler.CWPKI0025E=CWPKI0025E: Could not load the https Handler class "{0}". The extended error message is {1}.

CWPKI0026E

ssl.reinitialize.config.error.CWPKI0026E=CWPKI0026E: Error reinitializing the SSL configuration after a change to security.xml. The extended error message is "{0}".

CWPKI0027I

ssl.disable.url.hostname.verification.CWPKI0027I=CWPKI0027I: Disabling default hostname verification for HTTPS URL connections.

CWPKI0028E

ssl.no.such.algorithm.CWPKI0028E=CWPKI0028E: SSL handshake protocol "{0}" is not valid. This protocol is specified in the SSL configuration alias "{1}" loaded from SSL configuration file "{2}". The extended error message is: "{3}".

CWPKI0029E

ssl.invalid.context.provider.CWPKI0029E=CWPKI0029E: SSL context provider "{0}" is not valid. This provider is specified in the SSL configuration alias "{1}" loaded from SSL configuration file "{2}". The extended error message is: "{3}".

CWPKI0030E

ssl.signer.exchange.error.CWPKI0030E=CWPKI0030E: Error occurred exchanging signers between cell and node. The exception that occurred is: {0}.

CWPKI0031E

ssl.client.keystore.create.error.CWPKI0031E=CWPKI0031E: Error creating a client keystore or truststore during initialization. The exception that occurred is: {0}.

CWPKI0032E

ssl.self.signed.create.error.CWPKI0032E=CWPKI0032E: Error creating a self-signed certificate. The exception that occurred is: {0}.

CWPKI0033E

ssl.keystore.load.error.CWPKI0033E=CWPKI0033E: The keystore located at "{0}" failed to load due to the following error: {1}.

CWPKI0034E

ssl.schedule.init.error.CWPKI0034E=CWPKI0034E: Schedule "{0}" could not be initialized because of the following error: "{1}".

CWPKI0035E

ssl.schedule.read.error.CWPKI0035E=CWPKI0035E: Schedule "{0}" could not read the next scheduled date. Initializing alarm for the following date: {1}.

CWPKI0036E

ssl.notification.send.error.CWPKI0036E=CWPKI0036E: Error sending email to "{0}" using smtp server "{1}". The exception message is: "{2}".

CWPKI0037I

ssl.notification.send.info.CWPKI0037I=CWPKI0037I: Expiration monitor reports the following information: {0}.

CWPKI0038E

ssl.expiration.monitor.start.error.CWPKI0038E=CWPKI0038E: Expiration monitor failed to start with the following error: {0}.

CWPKI0039E

ssl.cannot.find.node.from.hostlist.CWPKI0039E=CWPKI0039E: Cannot find Node connector properties for the hostname {0} in the hostlist for keystore {1}.

CWPKI0040I

ssl.handshake.failure.info.CWPKI0040I=CWPKI0040I: An SSL handshake failure occurred from a secure client. The server's SSL signer has to be added to the client's trust store. A retrieveSigners utility is provided to download signers from the server but requires administrative permission. Check with your administrator to have this utility run to setup the secure environment before running the client. Alternatively, the com.ibm.ssl.enableSignerExchangePrompt can be enabled in ssl.client.props for "DefaultSSLSettings" in order to allow acceptance of the signer during the connection attempt.

CWPKI0041W

ssl.default.password.in.use.CWPKI0041W=CWPKI0041W: One or more key stores are using the default password.

CWPKI0200E

crypto.key.generate.configuration.error.CWPKI0200E=CWPKI0200E: An attempt to generate keys using KeySet {0} occurred when the KeySet is not configured to generate keys. The detailed message is: {1}.

CWPKI0201E

crypto.key.getkey.error.CWPKI0201E=CWPKI0201E: Error retrieving key alias {0} from KeySet {1}. The exception that occurred is: {2}.

CWPKI0202E

crypto.key.generate.class.load.error.CWPKI0202E=CWPKI0202E: An error occurred trying to instantiate the key generation class {0} configured in KeySet {1}. The detailed message is: {2}.

CWPKI0203E

crypto.key.import.error.CWPKI0203E=CWPKI0203E: An attempt to import keys to KeySet {0} failed. The detailed message is: {1}.

CWPKI0204E

crypto.key.auto.generate.error.CWPKI0204E=CWPKI0204E: An error occurred during a scheduled key generation for KeySetGroup {0}. The detailed error message is: {1}.

CWPKI0300I

ssl.signer.retrieval.usage.CWPKI0300I=CWPKI0300I: Use the -listRemoteKeyStoreNames and -listLocalKeyStoreNames options to get list of names for <remoteKeyStoreName> and <localKeyStoreName>, respectively. Usage: retrieveSigners <remoteKeyStoreName> <localKeyStoreName> [options] options: [-profileName <profileName>] [-remoteAlias <aliasFromRemoteStore>] [-localAlias <storeAsAlias>] [-listRemoteKeyStoreNames] [-listLocalKeyStoreNames] [-autoAcceptBootstrapSigner] [-uploadSigners] [-host <host>] [-port <port>] [-conntype <RMI|SOAP>] [-user <user>] [-password <password>] [-trace] [-logfile <filename>] [-replacelog] [-quiet] [-help]

CWPKI0301I

ssl.signer.retrieval.tracemodeon.CWPKI0301I=CWPKI0301I: Trace mode is on.

CWPKI0302E

ssl.signer.retrieval.logfile.error.CWPKI0302E=CWPKI0302E: Cannot write to the trace logfile at the following location: {0}

CWPKI0303I

ssl.signer.retrieval.logfile.location.CWPKI0303I=CWPKI0303I: Trace is being logged to the following location: {0}

CWPKI0304E

ssl.signer.remote.truststore.not.found.CWPKI0304E=CWPKI0304E: The <remoteKeyStoreName> specified as "{0}" was not found on the server.

CWPKI0305E

ssl.signer.alias.not.found.CWPKI0305E=CWPKI0305E: The <aliasFromRemoteStore> specified as "{0}" was not found in truststore "{1}" on the server.

CWPKI0306I

ssl.signer.remote.truststore.list.CWPKI0306I=CWPKI0306I: The following remote keystores exist on the specified server: {0}

CWPKI0307I

ssl.signer.local.truststore.list.CWPKI0307I=CWPKI0307I: The following local keystores exist on the client: {0}

CWPKI0308I

ssl.signer.add.to.local.truststore.CWPKI0308I=CWPKI0308I: Adding signer alias "{0}" to local keystore "{1}" with the following SHA digest: {2}

CWPKI0309I

ssl.signer.no.signers.added.CWPKI0309I=CWPKI0309I: All signers from remote keystore already exist in local keystore.

CWPKI0310E

ssl.signer.local.truststore.not.found.CWPKI0310E=CWPKI0310E: The <localKeyStoreName> specified as "{0}" was not found on the client.

[CWPKI0311E](#)

ssl.certificate.before.date.invalid.CWPKI0311E=CWPKI0311E: The certificate with subject DN {0} has a start date {1} which is valid after the current date/time. This will can happen if the client's clock is set earlier than the server's clock. Please verify the clocks are in sync between this client and server and retry the request.

[CWPKI0312E](#)

ssl.certificate.end.date.invalid.CWPKI0312E=CWPKI0312E: The certificate with subject DN {0} has an end date {1} which is no longer valid.

[CWPKI0313W](#)

ssl.signer.invalid.option.CWPKI0313W=CWPKI0313W: The following option is not valid: {0}

[CWPKI0314E](#)

ssl.signer.general.error.CWPKI0314E=CWPKI0314E: The following error is returned from an exception: {0}

CWPKI0001I

ssl.init.startinit.CWPKI0001I=CWPKI0001I: SSL service is initializing the configuration

Explanation

SSL service is initializing the configuration.

User response

None. Informational only

Parent topic: [CWPKI](#)

CWPKI0002I

ssl.init.startcomplete.CWPKI0002I=CWPKI0002I: SSL service initialization completed successfully

Explanation

SSL service initialization completed successfully..

User response

None. Informational only

Parent topic: [CWPKI](#)

CWPKI0003I

ssl.init.svcstart.CWPKI0003I=CWPKI0003I: SSL service is starting

Explanation

SSL service is starting.

User response

None. Informational only

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0004I

ssl.init.svcstartcomplete.CWPKI0004I=CWPKI0004I: SSL service started successfully

Explanation

SSL service started.

User response

None. Informational only

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0005I

ssl.init.startfail.CWPKI0005I=CWPKI0005I: SSL service initialization failed

Explanation

SSL service initialization failed

User response

None. Informational only

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0006E

ssl.init.mbeanerror.CWPKI0006E=CWPKI0006E: Error creating or registering {0} mBean. The exception is {1}

Explanation

An unexpected exception occurred when trying to create or register an mBean.

User response

There may be a problem with the configuration. The exception may include details.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0007I

ssl.init.svcstartfail.CWPKI0007I=CWPKI0007I: SSL service failed to start successfully

Explanation

SSL service did not start.

User response

None. Informational only

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0008E

ssl.init.error.CWPKI0008E=CWPKI0008E: Error during SSL initialization. The exception is {0}.

Explanation

An unexpected error occurred during security initialization.

User response

This is a general error. Look for previous messages that may be related to the failure or a configuration problem. Enabling SSL=all=enabled debug trace may yield additional information.

Parent topic: [CWPKI](#)

CWPKI0009E

ssl.init.nullsecobject.CWPKI0009E=CWPKI0009E: Cannot create security object during initialization.

Explanation

Cannot create the security object from repository. Internal Error.

User response

The security.xml might be corrupted or missing. Contact your service representative.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0010E

ssl.init.nullprocesstype.CWPKI0010E=CWPKI0010E: Cannot obtain the WAS process type during initialization.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWPKI](#)

CWPKI0011E

ssl.loadresource.error.CWPKI0011E=CWPKI0011E: Failed to load {0} resource from cell. The exception is {1}

Explanation

The specified resource could not be loaded due to an exception.

User response

The failure may be related to a configuration problem related to the resource.

Parent topic: [CWPKI](#)

CWPKI0012I

ssl.fips.enabled.CWPKI0012I=CWPKI0012I: FIPS is enabled.

Explanation

The server is running in FIPS mode, using the IBMJCEFIPS provider.

User response

No user action is required.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0013W

ssl.checkProviderList.warning.CWPKI0013W=CWPKI0013W: FIPS is enabled but the IBMJCEFIPS provider is not active in the java.security file. To ensure FIPS algorithms usage for all WAS process types, uncomment the IBMJCEFIPS provider in the java.security file, ahead of the IBMJCE, and renumber the provider list in sequential order.

Explanation

When the server is running in FIPS mode the IBMJCEFIPS provider should be in the java.security file.

User response

The java.security file needs to be changed to include the IBMJCEFIPS provider in the provider list before the IBMJCE provider.

Parent topic: [CWPKI](#)

CWPKI0014I

ssl.init.ssldm.init.CWPKI0014I=CWPKI0014I: The SSL component's FFDC Diagnostic Module {0} registered successfully: {1}.

Explanation

Describes whether or not the SSL component's FFDC Diagnostic module was successfully registered.

User response

None. Informational only.

Parent topic: [CWPKI](#)

CWPKI0015E

ssl.stop.error.CWPKI0015E=CWPKI0015E: Error stopping SSL component. The exception is {0}.

Explanation

An unexpected error occurred stopping the SSL component.

User response

This is a general error. Look for previous messages that may be related to the failure or a configuration problem. Enabling SSL=all=enabled debug trace may yield additional information.

Parent topic: [CWPKI](#)

CWPKI0016W

ssl.expiration.warning.CWPKI0016W=CWPKI0016W: The certificate with alias {0} from keyStore {1} will be expired in {2} days.

Explanation

A certificate is about to expire in the keystore.

User response

Open the keystore and validate the expiration dates on all certificates in the keystore. Prepare to generate new certificates, if necessary.

Parent topic: [CWPKI](#)

CWPKI0017E

ssl.expiration.expired.CWPKI0017E=CWPKI0017E: The certificate with alias {1} from keyStore {2} is expired.

Explanation

A certificate is expired in the keystore.

User response

Open the keystore and validate the expiration dates on all certificates in the keystore. Remove any expired certs.

Parent topic: [CWPKI](#)

CWPKI0018W

ssl.keystore.type.invalid.CWPKI0018W=CWPKI0018W: The keystore type of {0} is not valid for SSL config alias {1}.

Explanation

The keystore type configured is not correct.

User response

Change the keystore type in the SSL configuration.

Parent topic: [CWPKI](#)

CWPKI0019E

ssl.client.config.parse.CWPKI0019E=CWPKI0019E: Error parsing the SSL client configuration file {0}. The error returned is {1}.

Explanation

There may be a problem with the syntax of the ssl.client.props file or the location of the file is invalid.

User response

Review the error returned and check the syntax and location of the ssl.client.props file.

Parent topic: [CWPKI](#)

CWPKI0020E

ssl.load.trustmanager.error.CWPKI0020E=CWPKI0020E: Error loading custom trust manager class {0}. The exception message is {1}.

Explanation

A class loading error occurred loading the custom trust manager configured.

User response

Ensure the class can be found in the environment.

Parent topic: [CWPKI](#)

CWPKI0021E

ssl.load.keymanager.error.CWPKI0021E=CWPKI0021E: Error loading custom key manager class {0}. The exception message is {1}.

Explanation

A class loading error occurred loading the custom key manager configured.

User response

Ensure the class can be found in the environment.

Parent topic: [CWPKI](#)

CWPKI0022E

ssl.client.handshake.error.CWPKI0022E=CWPKI0022E: SSL HANDSHAKE FAILURE: A signer with SubjectDN "{0}" was sent from target host:port "{1}". The signer may need to be added to local trust store "{2}" located in SSL configuration alias "{3}" loaded from SSL configuration file "{4}". The extended error message from the SSL handshake exception is: "{5}".

Explanation

An error occurred during the SSL handshake. It may require a signer export/import from the target host to the client TrustStore.

User response

Review the extended error message coming from the TrustManager to help determine what needs to change between the target SSL configuration and the client SSL configuration.

Parent topic: [CWPKI](#)

CWPKI0023E

ssl.client.alias.not.found.CWPKI0023E=CWPKI0023E: The certificate alias "{0}" specified by the property com.ibm.ssl.keyStoreClientAlias is not found in KeyStore "{1}".

Explanation

The certificate alias specified for this SSL configuration is not in the specified KeyStore.

User response

Either add a certificate into the KeyStore with the specified certificate alias or change the specified certificate alias to match an alias found in the client KeyStore.

Parent topic: [CWPKI](#)

CWPKI0024E

ssl.server.alias.not.found.CWPKI0024E=CWPKI0024E: The certificate alias "{0}" specified by the property com.ibm.ssl.keyStoreServerAlias is not found in KeyStore "{1}".

Explanation

The certificate alias specified for this SSL configuration is not in the specified KeyStore.

User response

Either add a certificate into the KeyStore with the specified certificate alias or change the specified certificate alias to match an alias found in the server KeyStore.

Parent topic: [CWPKI](#)

CWPKI0025E

ssl.load.https.stream.handler.CWPKI0025E=CWPKI0025E: Could not load the https Handler class "{0}". The extended error message is {1}.

Explanation

There was a classloading error trying to load the HTTPS URLStreamHandler class.

User response

Check the SSL configuration to ensure the context provider is correct for the platform.

Parent topic: [CWPKI](#)

CWPKI0026E

ssl.reinitialize.config.error.CWPKI0026E=CWPKI0026E: Error reinitializing the SSL configuration after a change to security.xml. The extended error message is "{0}".

Explanation

An exception occurred reading the SSL configurations from the security.xml after a change occurred.

User response

Review the exception message text and verify the SSL configuration parameters are valid.

Parent topic: [CWPKI](#)

CWPKI0027I

ssl.disable.url.hostname.verification.CWPKI0027I=CWPKI0027I: Disabling default hostname verification for HTTPS URL connections.

Explanation

Hostname verification will be disabled by default for URL connections. Hostname verification checks that the X509 Certificate Common Name (CN) matches the hostname it is from.

User response

To enable default JSSE URL hostname verification, set the `com.ibm.ssl.performURLHostNameVerification` property to true.

Parent topic: [CWPKI](#)

CWPKI0028E

ssl.no.such.algorithm.CWPKI0028E=CWPKI0028E: SSL handshake protocol "{0}" is not valid. This protocol is specified in the SSL configuration alias "{1}" loaded from SSL configuration file "{2}". The extended error message is: "{3}".

Explanation

The handshake protocol specified is not recognized as a valid handshake protocol.

User response

Check the SSL configuration to ensure the right handshake protocol is specified.

Parent topic: [CWPKI](#)

CWPKI0029E

ssl.invalid.context.provider.CWPKI0029E=CWPKI0029E: SSL context provider "{0}" is not valid. This provider is specified in the SSL configuration alias "{1}" loaded from SSL configuration file "{2}". The extended error message is: "{3}".

Explanation

The SSL context provider specified is not recognized as a valid context provider.

User response

Check the SSL configuration to ensure the correct SSL context provider is specified.

Parent topic: [CWPKI](#)

CWPKI0030E

ssl.signer.exchange.error.CWPKI0030E=CWPKI0030E: Error occurred exchanging signers between cell and node. The exception that occurred is: {0}.

Explanation

The DefaultKeyStores between cell and node will have exchange signers with corresponding DefaultTrustStores. An error occurred during this process.

User response

A manual signer exchange may need to take place.

Parent topic: [CWPKI](#)

CWPKI0031E

ssl.client.keystore.create.error.CWPKI0031E=CWPKI0031E: Error creating a client keystore or truststore during initialization. The exception that occurred is: {0}.

Explanation

An error occurred while creating the file-based keystore or truststore during process initialization. Check that the keystore or truststore settings are valid.

User response

Verify the keystore or truststore settings in the ssl.client.props.

Parent topic: [CWPKI](#)

CWPKI0032E

ssl.self.signed.create.error.CWPKI0032E=CWPKI0032E: Error creating a self-signed certificate. The exception that occurred is: {0}.

Explanation

An error occurred while creating a self-signed certificate during process startup.

User response

Check that the default self-signed certificate property values (com.ibm.ssl.defaultCertReq*) are valid.

Parent topic: [CWPKI](#)

CWPKI0033E

ssl.keystore.load.error.CWPKI0033E=CWPKI0033E: The keystore located at "{0}" failed to load due to the following error: {1}.

Explanation

An error occurred while creating or opening the keystore.

User response

Check that the properties in the keystore configuration and ensure the keystore exists.

Parent topic: [CWPKI](#)

CWPKI0034E

ssl.schedule.init.error.CWPKI0034E=CWPKI0034E: Schedule "{0}" could not be initialized because of the following error: "{1}".

Explanation

An error occurred initializing the schedule.

User response

Check that the properties for the scheduler are valid. Ensure the /etc directory is writable.

Parent topic: [CWPKI](#)

CWPKI0035E

ssl.schedule.read.error.CWPKI0035E=CWPKI0035E: Schedule "{0}" could not read the next scheduled date. Initializing alarm for the following date: {1}.

Explanation

An error occurred reading the date from the schedule file in /etc.

User response

Ensure the /etc directory is writable or the file has not been modified.

Parent topic: [CWPKI](#)

CWPKI0036E

ssl.notification.send.error.CWPKI0036E=CWPKI0036E: Error sending email to "{0}" using smtp server "{1}". The exception message is: "{2}".

Explanation

An error occurring sending email to the specified SMTP server.

User response

Ensure the SMTP server specified is valid and that your companies firewall policy allows sending to SMTP ports.

Parent topic: [CWPKI](#)

CWPKI0037I

ssl.notification.send.info.CWPKI0037I=CWPKI0037I: Expiration monitor reports the following information: {0}.

Explanation

This is information regarding certificate expiration.

User response

May need to manage certificates to resolve the reported problems.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0038E

ssl.expiration.monitor.start.error.CWPKI0038E=CWPKI0038E: Expiration monitor failed to start with the following error: {0}.

Explanation

A problem occurred starting the expiration monitor command task.

User response

Try starting the expiration monitor explicitly to determine more information about the error.

Parent topic: [CWPKI](#)

CWPKI0039E

ssl.cannot.find.node.from.hostlist.CWPKI0039E=CWPKI0039E: Cannot find Node connector properties for the hostname {0} in the hostlist for keystore {1}.

Explanation

Make sure the hostname entered is in the canonical format as it appears in serverindex.xml.

User response

Edit the hostlist to convert it to the proper canonical format.

Parent topic: [CWPKI](#)

CWPKI0040I

ssl.handshake.failure.info.CWPKI0040I=CWPKI0040I: An SSL handshake failure occurred from a secure client. The server's SSL signer has to be added to the client's trust store. A retrieveSigners utility is provided to download signers from the server but requires administrative permission. Check with your administrator to have this utility run to setup the secure environment before running the client. Alternatively, the com.ibm.ssl.enableSignerExchangePrompt can be enabled in ssl.client.props for "DefaultSSLSettings" in order to allow acceptance of the signer during the connection attempt.

Explanation

This message is for providing options for the client to retrieve signers needed for a successful SSL connection.

User response

Either run retrieveSigners or enable the signer exchange prompt to correct the problem.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0041W

ssl.default.password.in.use.CWPKI0041W=CWPKI0041W: One or more key stores are using the default password.

Explanation

Default key store passwords should be changed when possible.

User response

Change all key store passwords either using AdminConsole or IKeyMan.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0200E

crypto.key.generate.configuration.error.CWPKI0200E=CWPKI0200E: An attempt to generate keys using KeySet {0} occurred when the KeySet is not configured to generate keys. The detailed message is: {1}.

Explanation

The KeySet either does not have a keyGenerationClass defined, it cannot find the keyGenerationClass, a read-only KeyStore is associated with the KeySet, or the KeyStore does not allow the writing of secret keys.

User response

Modify the configuration so a proper keyGenerationClass is configured and a KeyStore type is configured with allows the writing of secret keys.

Parent topic: [CWPKI](#)

CWPKI0201E

crypto.key.getkey.error.CWPKI0201E=CWPKI0201E: Error retrieving key alias {0} from KeySet {1}. The exception that occurred is: {2}.

Explanation

An error occurred while retrieving keys from the KeyStore for the specified KeySet.

User response

Check that the KeySet configuration is correct.

Parent topic: [CWPKI](#)

CWPKI0202E

crypto.key.generate.class.load.error.CWPKI0202E=CWPKI0202E: An error occurred trying to instantiate the key generation class {0} configured in KeySet {1}. The detailed message is: {2}.

Explanation

Either the runtime could not find the key generation class configured for the KeySet or the class does not implement `com.ibm.websphere.crypto.KeyGenerator` or `com.ibm.websphere.crypto.KeyPairGenerator`.

User response

Check to ensure the key generation class configured is specified in a location which can be found by the WebSphere runtime. Check the InfoCenter on where to specify custom classes for the runtime to find them.

Parent topic: [CWPKI](#)

CWPKI0203E

crypto.key.import.error.CWPKI0203E=CWPKI0203E: An attempt to import keys to KeySet {0} failed. The detailed message is: {1}.

Explanation

The keys passed in may not have been correctly formed or the keystore could not be accessed to store them.

User response

Attempt to determine the cause based on the exception and adjust the configuration as needed.

Parent topic: [CWPKI](#)

CWPKI0204E

crypto.key.auto.generate.error.CWPKI0204E=CWPKI0204E: An error occurred during a scheduled key generation for KeySetGroup {0}. The detailed error message is: {1}.

Explanation

A problem occurred while a new key reference was created for the KeySetGroup specified. After the key reference was created in the configuration the key was generated. One of these steps failed.

User response

Attempt to determine the cause based on the exception and adjust the configuration as needed.

Parent topic: [CWPKI](#)

CWPKI0300I

ssl.signer.retrieval.usage.CWPKI0300I=CWPKI0300I: Use the -listRemoteKeyStoreNames and -listLocalKeyStoreNames options to get list of names for <remoteKeyStoreName> and <localKeyStoreName>, respectively. Usage: retrieveSigners <remoteKeyStoreName> <localKeyStoreName> [options] options: [-profileName <profileName>] [-remoteAlias <aliasFromRemoteStore>] [-localAlias <storeAsAlias>] [-listRemoteKeyStoreNames] [-listLocalKeyStoreNames] [-autoAcceptBootstrapSigner] [-uploadSigners] [-host <host>] [-port <port>] [-conntype <RMI|SOAP>] [-user <user>] [-password <password>] [-trace] [-logfile <filename>] [-replacelog] [-quiet] [-help]

Explanation

Usage information on the parameters for executing this script.

User response

None.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0301I

ssl.signer.retrieval.tracemodeon.CWPKI0301I=CWPKI0301I: Trace mode is on.

Explanation

Indicates trace mode is on.

User response

None.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0302E

ssl.signer.retrieval.logfile.error.CWPKI0302E=CWPKI0302E: Cannot write to the trace logfile at the following location: {0}

Explanation

Indicates there's a problem writing to the specified logfile.

User response

Change the logfile path or make sure the file specified is not in use.

Parent topic: [CWPKI](#)

CWPKI0303I

ssl.signer.retrieval.logfile.location.CWPKI0303I=CWPKI0303I: Trace is being logged to the following location: {0}

Explanation

Indicates where the mode is being logged.

User response

None.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0304E

ssl.signer.remote.truststore.not.found.CWPKI0304E=CWPKI0304E: The <remoteKeyStoreName> specified as "{0}" was not found on the server.

Explanation

The remote truststore is not found.

User response

Try issuing -listRemoteKeyStoreNames command to get the list of names.

Parent topic: [CWPKI](#)

CWPKI0305E

ssl.signer.alias.not.found.CWPKI0305E=CWPKI0305E: The <aliasFromRemoteStore> specified as "{0}" was not found in truststore "{1}" on the server.

Explanation

The alias specified was not found in the truststore.

User response

Try issuing -listRemoteKeyStoreNames command to get the list of names.

Parent topic: [CWPKI](#)

CWPKI0306I

ssl.signer.remote.truststore.list.CWPKI0306I=CWPKI0306I: The following remote keystores exist on the specified server: {0}

Explanation

Indicates a list of the remote keystores.

User response

None.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0307I

ssl.signer.local.truststore.list.CWPKI0307I=CWPKI0307I: The following local keystores exist on the client: {0}

Explanation

Indicates a list of the local keystores.

User response

None.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0308I

ssl.signer.add.to.local.truststore.CWPKI0308I=CWPKI0308I: Adding signer alias "{0}" to local keystore "{1}" with the following SHA digest: {2}

Explanation

Indicates the signer being added to the local keystore.

User response

None.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0309I

ssl.signer.no.signers.added.CWPKI0309I=CWPKI0309I: All signers from remote keystore already exist in local keystore.

Explanation

Indicates no signers needed to be added to the local keystore.

User response

None.

Parent topic: [CWPKI](#)

CWPKI0310E

ssl.signer.local.truststore.not.found.CWPKI0310E=CWPKI0310E: The <localKeyStoreName> specified as "{0}" was not found on the client.

Explanation

The local truststore is not found.

User response

Try issuing -listLocalKeyStoreNames command to get the list of names.

Parent topic: [CWPKI](#)

CWPKI0311E

ssl.certificate.before.date.invalid.CWPKI0311E=CWPKI0311E: The certificate with subject DN {0} has a start date {1} which is valid after the current date/time. This will can happen if the client's clock is set earlier than the server's clock. Please verify the clocks are in sync between this client and server and retry the request.

Explanation

The start date of the certificate is not valid.

User response

Ensure that the client's clock matches up with the server's clock. Otherwise, create a certificate with the proper start date.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0312E

ssl.certificate.end.date.invalid.CWPKI0312E=CWPKI0312E: The certificate with subject DN {0} has an end date {1} which is no longer valid.

Explanation

The certificate has expired.

User response

Replace the certificate with a valid certificate.

Parent topic: [CWPKI](#)

CWPKI0313W

ssl.signer.invalid.option.CWPKI0313W=CWPKI0313W: The following option is not valid: {0}

Explanation

Check the command line to ensure the options are correct.

User response

Check the usage help and retry after correcting the option.

Parent topic: [CWPKI](#)

[Terms of Use](#) | [Feedback](#)

CWPKI0314E

ssl.signer.general.error.CWPKI0314E=CWPKI0314E: The following error is returned from an exception: {0}

Explanation

Check the command line to ensure the options are correct.

User response

Check the usage help and retry after correcting the option.

Parent topic: [CWPKI](#)

CWPRJ

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWPRJ0001E](#)

INVALID_ARGUMENT_NULL_CWPRJ0001E=CWPRJ0001E: Invalid value detected. The value for field {0} must be a valid non-null value.

[CWPRJ1001E](#)

INVALID_SUBSET_ATTRIBUTE_MISSING_CWPRJ1001E=CWPRJ1001E: Invalid subset attribute. Attribute does not exist on superset entity. Entity={0}, Entity Class={1}, Attribute={2}

[CWPRJ1002E](#)

INVALID_SUBSET_ATTRIBUTE_EXTRA_CWPRJ1002E=CWPRJ1002E: Invalid subset attribute. Extra attribute defined: Entity={0}, Entity Class={1}, Attribute={2}

[CWPRJ1003E](#)

INVALID_SUBSET_ATTRIBUTE_TYEMISMATCH_CWPRJ1003E=CWPRJ1003E: Invalid subset attribute. Attribute or association type mismatch: Entity={0}, Entity Class={1}, Attribute={2}

[CWPRJ1004E](#)

INVALID_ASSOCIATION_KEY_CARDINALITY_CWPRJ1004E=CWPRJ1004E: Invalid association. Associations that are also keys can only have uni-directional OneToOne or ManyToOne relationships.

[CWPRJ1005E](#)

ERROR_RESOLVING_ASSOCIATION_CWPRJ1005E=CWPRJ1005E: Error resolving entity association. Entity={0}, association={1}.

[CWPRJ1006E](#)

ERROR_LOADING_ENTITY_CLASS_CWPRJ1006E=CWPRJ1006E: Error loading class {0} for entity {1}

[CWPRJ1007E](#)

INVALID_ENTITY_DEFINITION_CWPRJ1007E=CWPRJ1007E: Invalid entity definition for class: {0}. {1}

[CWPRJ1008E](#)

INVALID_ENTITY_DEFINITION_TOPLEVEL_CWPRJ1008E=CWPRJ1008E: {0} class must be defined as a top level class.

[CWPRJ1009E](#)

INVALID_ENTITY_DEFINITION_FINAL_CWPRJ1009E=CWPRJ1009E: {0} class must not be final

[CWPRJ1010E](#)

INVALID_ENTITY_DEFINITION_CONSTRUCTOR_CWPRJ1010E=CWPRJ1010E: {0} class must define a default, public or protected, no-argument constructor.

[CWPRJ1011E](#)

INVALID_ENTITY_DEFINITION_PUBLIC_CWPRJ1011E=CWPRJ1011E: {0} class must be public.

[CWPRJ1012E](#)

INVALID_ENTITY_DEFINITION_COMPARABLE_CWPRJ1012E=CWPRJ1012E: IdClass {0} must define equals() and hashCode() methods.

[CWPRJ1013E](#)

INVALID_ENTITY_DEFINITION_NOMETADATA_CWPRJ1013E=CWPRJ1013E: No metadata information was defined for the entity {0}.

[CWPRJ1014E](#)

INVALID_SUPERSET_ENTITY_NOMETADATA_CWPRJ1014E=CWPRJ1014E: No superset metadata information was defined for entity {0} with superset {1}.

[CWPRJ1015E](#)

ENTITY_DEFINITION_EXCEPTION_CWPRJ1015E=CWPRJ1015E: An exception occurred while creating the entity configuration for an annotated class or XML metadata for entity: {0}.

[CWPRJ1016E](#)

INVALID_IDCLASS_DEFINITION_MISSINGATTRIBUTE_CWPRJ1016E=CWPRJ1016E: Key class "{0}" must define attribute: "{1}".

[CWPRJ1017E](#)

INVALID_IDCLASS_DEFINITION_INVALIDATTRIBUTETYPE_CWPRJ1017E=CWPRJ1017E: Key class "{0}" attribute "{1}" type is incorrect. Declared type is: "{2}". Required type is: "{3}".

[CWPRJ1020E](#)

INVALID_ORDERBY_SPECIFIED_CWPRJ1020E=CWPRJ1020E: Error in OrderBy configuration. Field "{3}" in entity "{1}" does not exist. Check attribute "{2}" of entity "{0}".

CWPRJ1021E

INVALID_VERSION_TYPE_SPECIFIED_CWPRJ1021E=CWPRJ1021E: Error in version type. Specified version type of "{0}" is not supported.

CWPRJ1022W

INVALID_ASSOCIATION_REFERENCE_CWPRJ1022W=CWPRJ1022W: Association reference from entity: "{0}" attribute "{1}", to entity: "{2}" for attribute key(s) "{3}", association key "{4}" could not be found.

CWPRJ1023E

INVALID_ENTITY_ACCESTYPE_CWPRJ1023E=CWPRJ1023E: Invalid entity access-type specified: {0}

CWPRJ1024E

INVALID_XML_FILE_CWPRJ1024E=CWPRJ1024E: The XML file is invalid. A problem has been detected with {0} at line {1}. The error message is {2}.

CWPRJ1025E

MULTIPLE_ENTITY_ACCESTYPE_CWPRJ1025E=CWPRJ1025E: Unable to determine entity access type. Both fields and properties are annotated.

CWPRJ1026E

MISSING_ATTRIBUTES_CWPRJ1026E=CWPRJ1026E: No attributes or associations defined.

CWPRJ1027E

MISSING_ENTITYCLASS_XML_CWPRJ1027E=CWPRJ1027E: Entity class: "{0}" does not exist in entity descriptor file: "{1}".

CWPRJ1028E

MULTIPLE_SCHEMAROOTS_CWPRJ1028E=CWPRJ1028E: Multiple schema root entities detected. Schema root already defined for entity: {0}.

CWPRJ1029E

MULTIPLE_SCHEMAROOTS_CWPRJ1029E=CWPRJ1029E: Multiple schema root references detected for Entity class: "{0}", First Root Class: "{1}", Second Root Class: "{2}".

CWPRJ1100E

INVALID_ATTRIBUTE_CWPRJ1100E=CWPRJ1100E: Invalid attribute or association: "{0}".

CWPRJ1101E

MISSING_FIELD_CWPRJ1101E=CWPRJ1101E: Field is undefined for class: "{0}".

CWPRJ1102E

MISSING_PROPERTY_CWPRJ1102E=CWPRJ1102E: Property is undefined for class: "{0}".

CWPRJ1103E

MISSING_ASSOCIATION_TARGET_ENTITY_CWPRJ1103E=CWPRJ1103E: The target entity is undefined.

CWPRJ1104E

DUPLICATE_FIELDPROPERTY_CWPRJ1104E=CWPRJ1104E: Attribute is defined more than once.

CWPRJ1105E

MULTIPLE_INVERSE_ASSOCIATIONS_CWPRJ1105E=CWPRJ1105E: The target association has more than one inverse relationship to this entity and is missing the MappedBy relationship definition. Source entity: {0}, Inverse entity: {1}, Duplicate attributes: {2}

CWPRJ1108E

INVALID_INVERSE_ASSOCIATION_TYPE_CWPRJ1108E=CWPRJ1108E: The inverse target association references an invalid entity type. Inverse, target entity: {0}, association name: {1}

CWPRJ1109E

INVALID_ENTITY_DEFINITION_MISSING_TARGET_CWPRJ1109E=CWPRJ1109E: The target entity type of {0} is not defined.

CWPRJ1110E

INVALID_INVERSE_KEY_ASSOCIATION_CWPRJ1110E=CWPRJ1110E: An association that is also a key must not have an inverse association. Inverse, target entity: {0}, association name: {1}

CWPRJ1111E

INVERSE_ASSOCIATION_TYPE_MISMATCH_CWPRJ1111E=CWPRJ1111E: The inverse target association must match the source target type. Inverse, target entity: {0}, association name: {1}

CWPRJ1112E

TARGET_ENTITY_NOT_DEFINED_CWPRJ1112E=CWPRJ1112E: Target entity not defined for field or property: "{0}".

CWPRJ1113E

ENTITY_ATTRIBUTE_MISSING_CARDINALITY_CWPRJ1113E=CWPRJ1113E: Attribute "{0}" is an entity type but a relationship to the entity is not defined.

CWPRJ1114E

ENTITY_ATTRIBUTE_NOT_SERIALIZABLE_CWPRJ1114E=CWPRJ1114E: Attribute "{0}" of type "{1}" is not serializable.

[CWPRJ1115E](#)

INVALID_MANY_ASSOCIATION_TYPE_CWPRJ1115E=CWPRJ1115E: Invalid OneToMany or ManyToMany association type of: "{0}".

[CWPRJ1200I](#)

PROJECTOR_INSTRUMENTATION_ENABLED_CWPRJ1200I=CWPRJ1200I: Projector entity class instrumentation is enabled. The instrumentation mode is {0}.

[CWPRJ1201E](#)

FIELD_ACCESS_ENTITY_NOT_INSTRUMENTED_CWPRJ1201E=CWPRJ1201E: Field-access entity not instrumented. Entity class={0}.

[CWPRJ1202W](#)

PROXY_UNAVAILABLE_CWPRJ1202W=CWPRJ1202W: Entity proxy support is unavailable.

[CWPRJ1300E](#)

MULTIPLE_METHODS_PER_TYPE_CWPRJ1300E=CWPRJ1300E: Multiple methods listen to the same event {0} in class {1}.

[CWPRJ1301E](#)

ENTITY_CALLBACK_NO_PARAM_CWPRJ1301E=CWPRJ1301E: Entity lifecycle callback method, {0}, defined in the entity class {1} must have no parameters.

[CWPRJ1302E](#)

LISTENER_CALLBACK_ONE_PARAM_CWPRJ1302E=CWPRJ1302E: Entity lifecycle callback method, {0}, defined in the entity listener class {1} must have only one parameter.

[CWPRJ1303E](#)

LISTENER_UNASSIGNABLE_TYPE_CWPRJ1303E=CWPRJ1303E: The parameter of the entity callback method, {0}, is not assignable to entity {1}.

[CWPRJ1304E](#)

NO_SUCH_CALLBACK_METHOD_CWPRJ1304E=CWPRJ1304E: The method name {0} with parameter {1} does not exist in the class {2}.

[CWPRJ1305E](#)

UNEXPECTED_CALLBACK_EXCEPTION_CWPRJ1305E=CWPRJ1305E: Unexpected exception encountered when invoking the lifecycle callback method {0}: {1}

[CWPRJ5000I](#)

INSTRUMENTATION_ENABLED_CWPRJ5000I=CWPRJ5000I: Java instrumentation mechanism is enabled. The instrumentation mode is {0}.

[CWPRJ9000I](#)

ENGLISH_ONLY_INFO_MESSAGE_CWPRJ9000=CWPRJ9000I: This message is an English-only Informational message: {0}.

[CWPRJ9001W](#)

ENGLISH_ONLY_WARN_MESSAGE_CWPRJ9001=CWPRJ9001W: This message is an English-only Warning message: {0}.

[CWPRJ9002E](#)

ENGLISH_ONLY_ERROR_MESSAGE_CWPRJ9002=CWPRJ9002E: This message is an English only Error message: {0}.

CWPRJ0001E

INVALID_ARGUMENT_NULL_CWPRJ0001E=CWPRJ0001E: Invalid value detected. The value for field {0} must be a valid non-null value.

Explanation

The specified field or argument must not be null.

User response

Change the method or object call to set the appropriate argument or verify that the field value is properly set.

Parent topic: [CWPRJ](#)

CWPRJ1001E

INVALID_SUBSET_ATTRIBUTE_MISSING_CWPRJ1001E=CWPRJ1001E: Invalid subset attribute. Attribute does not exist on superset entity. Entity={0}, Entity Class={1}, Attribute={2}

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWPRJ](#)

CWPRJ1002E

INVALID_SUBSET_ATTRIBUTE_EXTRA_CWPRJ1002E=CWPRJ1002E: Invalid subset attribute. Extra attribute defined:
Entity={0}, Entity Class={1}, Attribute={2}

Explanation

The specified entity attribute or association does not exist on the superset entity.

User response

Review the metadata for the entity attribute and verify that the name and type match the superset entity.

Parent topic: [CWPRJ](#)

CWPRJ1003E

INVALID_SUBSET_ATTRIBUTE_TYPEMISMATCH_CWPRJ1003E=CWPRJ1003E: Invalid subset attribute. Attribute or association type mismatch: Entity={0}, Entity Class={1}, Attribute={2}

Explanation

The specified entity attribute or association exists on the superset entity, but is of a different type or cardinality.

User response

Review the metadata for the entity attribute and verify that the name and type match the superset entity.

Parent topic: [CWPRJ](#)

CWPRJ1004E

INVALID_ASSOCIATION_KEY_CARDINALITY_CWPRJ1004E=CWPRJ1004E: Invalid association. Associations that are also keys can only have uni-directional OneToOne or ManyToOne relationships.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [CWPRJ](#)

CWPRJ1005E

ERROR_RESOLVING_ASSOCIATION_CWPRJ1005E=CWPRJ1005E: Error resolving entity association. Entity={0}, association={1}.

Explanation

The specified entity association is not valid. The metadata processor was unable to create the entity because of this error.

User response

Check the nested exception message for details on how to resolve the error and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1006E

ERROR_LOADING_ENTITY_CLASS_CWPRJ1006E=CWPRJ1006E: Error loading class {0} for entity {1}

Explanation

The specified entity class or related class could not be loaded from the classpath.

User response

Verify that the class definition exists on the classpath.

Parent topic: [CWPRJ](#)

[Terms of Use](#) | [Feedback](#)

CWPRJ1007E

INVALID_ENTITY_DEFINITION_CWPRJ1007E=CWPRJ1007E: Invalid entity definition for class: {0}. {1}

Explanation

The specified entity definition is invalid.

User response

Review the exception text, resolve the error and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1008E

INVALID_ENTITY_DEFINITION_TOPLEVEL_CWPRJ1008E=CWPRJ1008E: {0} class must be defined as a top level class.

Explanation

All entities must be top-level classes. They cannot be interfaces.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1009E

INVALID_ENTITY_DEFINITION_FINAL_CWPRJ1009E=CWPRJ1009E: {0} class must not be final

Explanation

Entity classes must not have the final modifier specified.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

[Terms of Use](#) | [Feedback](#)

CWPRJ1010E

INVALID_ENTITY_DEFINITION_CONSTRUCTOR_CWPRJ1010E=CWPRJ1010E: {0} class must define a default, public or protected, no-argument constructor.

Explanation

All entities must have a default, public or protected, no-argument constructor.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1011E

INVALID_ENTITY_DEFINITION_PUBLIC_CWPRJ1011E=CWPRJ1011E: {0} class must be public.

Explanation

All entities must have a public modifier.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

[Terms of Use](#) | [Feedback](#)

CWPRJ1012E

INVALID_ENTITY_DEFINITION_COMPARABLE_CWPRJ1012E=CWPRJ1012E: IdClass {0} must define equals() and hashCode() methods.

Explanation

Classes that are used as composite entity key classes must implement the equals() and hashCode() methods.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1013E

INVALID_ENTITY_DEFINITION_NOMETADATA_CWPRJ1013E=CWPRJ1013E: No metadata information was defined for the entity {0}.

Explanation

An entity was referenced in XML, but no metadata exists for that entity.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1014E

INVALID_SUPERSET_ENTITY_NOMETADATA_CWPRJ1014E=CWPRJ1014E: No superset metadata information was defined for entity {0} with superset {1}.

Explanation

An entity was defined with a superset entity, but no entity metadata was defined for the superset entity.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1015E

ENTITY_DEFINITION_EXCEPTION_CWPRJ1015E=CWPRJ1015E: An exception occurred while creating the entity configuration for an annotated class or XML metadata for entity: {0}.

Explanation

There is an error in the metadata information that describes the entity characteristics.

User response

Check the nested exception message for details on how to resolve the error and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1016E

INVALID_IDCLASS_DEFINITION_MISSINGATTRIBUTE_CWPRJ1016E=CWPRJ1016E: Key class "{0}" must define attribute: "{1}".

Explanation

Key classes must define all of the attributes from the referencing entity class.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1017E

INVALID_IDCLASS_DEFINITION_INVALIDATTRIBUTETYPE_CWPRJ1017E=CWPRJ1017E: Key class "{0}" attribute "{1}" type is incorrect. Declared type is: "{2}". Required type is: "{3}".

Explanation

Key classes must define all of the attributes with the same type from the referencing entity class.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1020E

INVALID_ORDERBY_SPECIFIED_CWPRJ1020E=CWPRJ1020E: Error in OrderBy configuration. Field "{3}" in entity "{1}" does not exist. Check attribute "{2}" of entity "{0}".

Explanation

The specified field does not exist in the target entity.

User response

Verify the field exists.

Parent topic: [CWPRJ](#)

CWPRJ1021E

INVALID_VERSION_TYPE_SPECIFIED_CWPRJ1021E=CWPRJ1021E: Error in version type. Specified version type of "{0}" is not supported.

Explanation

Specified version type is not supported.

User response

Check the documentation for the types which are supported.

Parent topic: [CWPRJ](#)

CWPRJ1022W

INVALID_ASSOCIATION_REFERENCE_CWPRJ1022W=CWPRJ1022W: Association reference from entity: "{0}" attribute "{1}", to entity: "{2}" for attribute key(s) "{3}", association key "{4}" could not be found.

Explanation

Unable to construct object due to one or more missing associations.

User response

Entities to which the source entity has a relationship with has been removed by the application.

Parent topic: [CWPRJ](#)

CWPRJ1023E

INVALID_ENTITY_ACCESTYPE_CWPRJ1023E=CWPRJ1023E: Invalid entity access-type specified: {0}

Explanation

Entities and entity identification classes must have an access-type of FIELD or PROPERTY.

User response

Review the entity metadata and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1024E

INVALID_XML_FILE_CWPRJ1024E=CWPRJ1024E: The XML file is invalid. A problem has been detected with {0} at line {1}. The error message is {2}.

Explanation

The XML file does not conform to the schema.

User response

Edit the XML file so that it conforms to the schema.

Parent topic: [CWPRJ](#)

CWPRJ1025E

MULTIPLE_ENTITY_ACESSTYPE_CWPRJ1025E=CWPRJ1025E: Unable to determine entity access type. Both fields and properties are annotated.

Explanation

Annotated entities can only have annotations specified on the field or property methods, but not both.

User response

Review the entity annotations and verify that the entity metadata annotations are only specified on the fields or get method of each attribute.

Parent topic: [CWPRJ](#)

CWPRJ1026E

MISSING_ATTRIBUTES_CWPRJ1026E=CWPRJ1026E: No attributes or associations defined.

Explanation

All entities must have at least 1 attribute or association defined.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

[Terms of Use](#) | [Feedback](#)

CWPRJ1027E

MISSING_ENTITYCLASS_XML_CWPRJ1027E=CWPRJ1027E: Entity class: "{0}" does not exist in entity descriptor file: "{1}".

Explanation

The specified entity class could not be found.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1028E

MULTIPLE_SCHEMAROOTCS_CWPRJ1028E=CWPRJ1028E: Multiple schema root entities detected. Schema root already defined for entity: {0}.

Explanation

The specified entity class could not be found.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1029E

MULTIPLE_SCHEMAROOTS_CWPRJ1029E=CWPRJ1029E: Multiple schema root references detected for Entity class: "{0}", First Root Class: "{1}", Second Root Class: "{2}".

Explanation

Entities may only reference one schema root entity explicitly or implicitly via its composite key.

User response

Verify that only one schema root entity is reachable from the key associations of the entity class and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1100E

INVALID_ATTRIBUTE_CWPRJ1100E=CWPRJ1100E: Invalid attribute or association: "{0}".

Explanation

The specified attribute is invalid.

User response

Check the nested exception message for details on how to resolve the error and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1101E

MISSING_FIELD_CWPRJ1101E=CWPRJ1101E: Field is undefined for class: "{0}".

Explanation

The specified field does not exist on the specified class.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1102E

MISSING_PROPERTY_CWPRJ1102E=CWPRJ1102E: Property is undefined for class: "{0}".

Explanation

The specified property does not exist on the specified class.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

[Terms of Use](#) | [Feedback](#)

CWPRJ1103E

MISSING_ASSOCIATION_TARGET_ENTITY_CWPRJ1103E=CWPRJ1103E: The target entity is undefined.

Explanation

The target entity type could not be determined.

User response

The target entity type of an association must be explicitly set on the association metadata if it cannot be determined by the field or property type.

Parent topic: [CWPRJ](#)

CWPRJ1104E

DUPLICATE_FIELDPROPERTY_CWPRJ1104E=CWPRJ1104E: Attribute is defined more than once.

Explanation

An attribute or association must have a unique name and cannot be defined more than one time for a single entity.

User response

Review the XML definition for the entity and verify that there is a different name for each element: basic, one-to-one, many-to-one, one-to-many or many-to-many.

Parent topic: [CWPRJ](#)

CWPRJ1105E

MULTIPLE_INVERSE_ASSOCIATIONS_CWPRJ1105E=CWPRJ1105E: The target association has more than one inverse relationship to this entity and is missing the MappedBy relationship definition. Source entity: {0}, Inverse entity: {1}, Duplicate attributes: {2}

Explanation

A bi-directional relationship between two entities must have one field or property defined with the mappedBy annotation parameter or XML attribute.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1108E

INVALID_INVERSE_ASSOCIATION_TYPE_CWPRJ1108E=CWPRJ1108E: The inverse target association references an invalid entity type. Inverse, target entity: {0}, association name: {1}

Explanation

The inverse side of a bi-directional relationship between two entities has an invalid target entity type defined.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1109E

INVALID_ENTITY_DEFINITION_MISSING_TARGET_CWPRJ1109E=CWPRJ1109E: The target entity type of {0} is not defined.

Explanation

An entity has an association to another entity, but the target entity does not exist in the metadata repository.

User response

Review the entity metadata and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1110E

INVALID_INVERSE_KEY_ASSOCIATION_CWPRJ1110E=CWPRJ1110E: An association that is also a key must not have an inverse association. Inverse, target entity: {0}, association name: {1}

Explanation

An entity key may have a one-to-one or many-to-one association to another entity, but the target entity must not have an bi-directional relationship to the same entity.

User response

Review the entity metadata and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ1111E

INVERSE_ASSOCIATION_TYPE_MISMATCH_CWPRJ1111E=CWPRJ1111E: The inverse target association must match the source target type. Inverse, target entity: {0}, association name: {1}

Explanation

The inverse side of a bi-directional relationship between two entities has a target type that does not match the source type.

User response

Verify that the class or entity metadata definition is valid and resubmit the operation.

Parent topic: [CWPRJ](#)

CWPRJ112E

TARGET_ENTITY_NOT_DEFINED_CWPRJ112E=CWPRJ112E: Target entity not defined for field or property: "{0}".

Explanation

The target Entity for the association is not defined.

User response

Verify that the target entity is defined.

Parent topic: [CWPRJ](#)

CWPRJ1113E

ENTITY_ATTRIBUTE_MISSING_CARDINALITY_CWPRJ1113E=CWPRJ1113E: Attribute "{0}" is an entity type but a relationship to the entity is not defined.

Explanation

The attribute type must declare the relationship cardinality.

User response

Define the cardinality of the entity relationship using the OneToOne or ManyToOne annotation or in the entity descriptor file.

Parent topic: [CWPRJ](#)

CWPRJ114E

ENTITY_ATTRIBUTE_NOT_SERIALIZABLE_CWPRJ114E=CWPRJ114E: Attribute "{0}" of type "{1}" is not serializable.

Explanation

All attribute types must be serializable.

User response

Verify that the attribute implements the java.io.Serializable or java.io.Externalizable interfaces.

Parent topic: [CWPRJ](#)

CWPRJ1115E

INVALID_MANY_ASSOCIATION_TYPE_CWPRJ1115E=CWPRJ1115E: Invalid OneToMany or ManyToMany association type of: "{0}".

Explanation

All multi-valued association types must be of type: java.util.Collection, java.util.List or java.util.Set.

User response

Change the attribute type to one of the supported data types.

Parent topic: [CWPRJ](#)

CWPRJ1200I

PROJECTOR_INSTRUMENTATION_ENABLED_CWPRJ1200I=CWPRJ1200I: Projector entity class instrumentation is enabled. The instrumentation mode is {0}.

Explanation

Projector entity class instrumentation is enabled. Java classes in the configured transformation domain may be transformed to support field-access entities.

User response

None.

Parent topic: [CWPRJ](#)

CWPRJ1201E

FIELD_ACCESS_ENTITY_NOT_INSTRUMENTED_CWPRJ1201E=CWPRJ1201E: Field-access entity not instrumented.
Entity class={0}.

Explanation

This error occurs when the Projector Java instrumentation agent is enabled but the field-access entity class is not included in instrumentation domain, or the field-access entity class is included in instrumentation domain, but is not in the field-access entity domain. The field-access entity domain is an optional configuration. When specified, all field-access entity classes must be included in the field-access entity domain.

User response

Verify that the field-access entity class is included in instrumentation domain if no field-access entity domain is defined. If field-access entity domain is defined, ensure the entity class is included.

Parent topic: [CWPRJ](#)

[Terms of Use](#) | [Feedback](#)

CWPRJ1202W

PROXY_UNAVAILABLE_CWPRJ1202W=CWPRJ1202W: Entity proxy support is unavailable.

Explanation

Proxies cannot be created for managed entities because the CGLIB library is not installed or is incompatible. The absence of proxy support may degrade performance.

User response

Review the FFDC logs for details on why the CGLIB was unable to load and verify that a supported CGLIB library version is installed in the classpath.

Parent topic: [CWPRJ](#)

CWPRJ1300E

MULTIPLE_METHODS_PER_TYPE_CWPRJ1300E=CWPRJ1300E: Multiple methods listen to the same event {0} in class {1}.

Explanation

Only one method can listen to one particular lifecycle event in one entity or entity listener class definition.

User response

Remove extra methods.

Parent topic: [CWPRJ](#)

CWPRJ1301E

ENTITY_CALLBACK_NO_PARAM_CWPRJ1301E=CWPRJ1301E: Entity lifecycle callback method, {0}, defined in the entity class {1} must have no parameters.

Explanation

The entity lifecycle callback method defined in an entity class should not have parameters.

User response

Redefine the entity lifecycle callback method to remove the parameters.

Parent topic: [CWPRJ](#)

CWPRJ1302E

LISTENER_CALLBACK_ONE_PARAM_CWPRJ1302E=CWPRJ1302E: Entity lifecycle callback method, {0}, defined in the entity listener class {1} must have only one parameter.

Explanation

The lifecycle callback method defined in an entity listener class should have just one parameter.

User response

Redefine the entity lifecycle callback method in the listener class to have just one parameter.

Parent topic: [CWPRJ](#)

CWPRJ1303E

LISTENER_UNASSIGNABLE_TYPE_CWPRJ1303E=CWPRJ1303E: The parameter of the entity callback method, {0}, is not assignable to entity {1}.

Explanation

The parameter of a lifecycle callback method defined in an entity listener should be the same type as the entity type or a super type of the entity type.

User response

Redefine the entity lifecycle callback method to make sure the parameter type is the entity type or its super type.

Parent topic: [CWPRJ](#)

[Terms of Use](#) | [Feedback](#)

CWPRJ1304E

NO_SUCH_CALLBACK_METHOD_CWPRJ1304E=CWPRJ1304E: The method name {0} with parameter {1} does not exist in the class {2}.

Explanation

The configured callback method name does not exist in the specified entity or entity listener class.

User response

Fix the configuration by choosing a valid method name.

Parent topic: [CWPRJ](#)

CWPRJ1305E

UNEXPECTED_CALLBACK_EXCEPTION_CWPRJ1305E=CWPRJ1305E: Unexpected exception encountered when invoking the lifecycle callback method {0}: {1}

Explanation

A runtime exception is caught during the invocation of the lifecycle callback method.

User response

Inspect the exception and fix the root cause.

Parent topic: [CWPRJ](#)

CWPRJ5000I

INSTRUMENTATION_ENABLED_CWPRJ5000I=CWPRJ5000I: Java instrumentation mechanism is enabled. The instrumentation mode is {0}.

Explanation

Java instrumentation mechanism is enabled. Java classes in the configured transformation domain may be transformed to support field-access entities.

User response

None.

Parent topic: [CWPRJ](#)

CWPRJ9000I

ENGLISH_ONLY_INFO_MESSAGE_CWPRJ9000=CWPRJ9000I: This message is an English-only Informational message: {0}.

Explanation

This informational message is not translated.

User response

See message for details.

Parent topic: [CWPRJ](#)

CWPRJ9001W

ENGLISH_ONLY_WARN_MESSAGE_CWPRJ9001=CWPRJ9001W: This message is an English-only Warning message: {0}.

Explanation

This warning message is not translated.

User response

See message for details.

Parent topic: [CWPRJ](#)

CWPRJ9002E

ENGLISH_ONLY_ERROR_MESSAGE_CWPRJ9002=CWPRJ9002E: This message is an English only Error message: {0}.

Explanation

This error message is not translated.

User response

See message for details.

Parent topic: [CWPRJ](#)

CWPTF

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

CWPTF0001E

CWPTF_ExcErr=CWPTF0001E: Exception: {0}.

CWPTF0002W

CWPTF_ExcWarn=CWPTF0002W: Exception: {0}.

CWPTF0003W

CWPTF_InvCellCustomProp=CWPTF0003W: Invalid trace cell custom property: name={0}, value={1}, reason={2}.

CWPTF0001E

CWPTF_ExcErr=CWPTF0001E: Exception: {0}.

Explanation

An unexpected exception occurred.

User response

Report program defect to IBM.

Parent topic: [CWPTF](#)

[Terms of Use](#) | [Feedback](#)

CWPTF0002W

CWPTF_ExcWarn=CWPTF0002W: Exception: {0}.

Explanation

An unexpected exception occurred.

User response

Report program defect to IBM.

Parent topic: [CWPTF](#)

[Terms of Use](#) | [Feedback](#)

CWPTF0003W

CWPTF_InvCellCustomProp=CWPTF0003W: Invalid trace cell custom property: name={0}, value={1}, reason={2}.

Explanation

An invalid trace cell custom property was found as indicated by the reason.

User response

Correct the name and/or value of the cell custom property as indicated by the reason, or delete the custom property.

Parent topic: [CWPTF](#)

CWSPT

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWSPT0001E](#)

SHOULD_NOT_OCCUR_REPORT_TO_DEVELOPER=CWSPT0001E: Should not occur: report to developer; message= {0}

[CWSPT0004E](#)

CANNOT_CREATE_SOFT_CHECKPOINTING_LOG_FOR_TRANSFORM=CWSPT0004E: Cannot create soft checkpointing log for transform: {0} : {1}

[CWSPT0005E](#)

LOGGING_ERROR_EXCEPTION_RAISED=CWSPT0005E: Logging error. Exception raised: {0}

[CWSPT0006E](#)

NO_BUFFERED_LOG_MESSAGE=CWSPT0006E: There is no buffered log message in the topic. The local log index is {0}.

[CWSPT0007E](#)

MSG_LOGGING_TURNED_OFF=CWSPT0007E: Msg logging turned off

[CWSPT0008E](#)

CANNOT_REPLAY_MESSAGE=CWSPT0008E: Cannot replay msg from {0} to {1} because {2}

[CWSPT0009E](#)

CANNOT_SET_HOST_SERVICES=CWSPT0009E: Cannot set host services: {0}

[CWSPT0010W](#)

NO_AUX_INFO=CWSPT0010W: The dummy Soft Checkpointing service could not find auxiliary information for a transform

[CWSPT0011W](#)

CANNOT_RECLAIM_SC_LOG_OF_TRANSFORM=CWSPT0011W: Cannot reclaim Soft Checkpointing log of transform with ID {0}. The exception message is {1}.

[CWSPT0012W](#)

CANNOT_PUT_SOFT_CHECKPOINT_RECORDS=CWSPT0012W: Cannot put soft checkpointing log records! {0}

[CWSPT0013W](#)

SMILETIMER_REQUESTTIMEOUT_RAISED_EXCEPTION=CWSPT0013W: Exception encountered when scheduling a timer with interval {0}. The exception message is {1}.

[CWSPT0014W](#)

CANNOT_FIND_WRAPPER_MAKER_FOR_TOPIC=CWSPT0014W: Severe Runtime error: "Cannot find wrapper maker for topic {0}, methodID {1}"

[CWSPT0015I](#)

RECOVERING_TRANSFORM=CWSPT0015I: Recovering transform {0}.

[CWSPT0103E](#)

INSTANTIATION_EXCEPTION=CWSPT0103E: Class: {0} can not be instantiated.

[CWSPT0104E](#)

ILLEGALACCESS_EXCEPTION=CWSPT0104E: Class: {0} can not be instantiated due to IllegalAccessException exception.

[CWSPT0105E](#)

CLASSNOTFOUND_EXCEPTION=CWSPT0105E: Class: {0} can not be found.

[CWSPT0106E](#)

MALFORMEDURL_EXCEPTION=CWSPT0106E: Class: {0} can not be instantiated due to a malformed URL.

[CWSPT0108E](#)

VIEW_NOT_FOUND=CWSPT0108E: View: {0} can not be retrieved.

[CWSPT0112I](#)

DELETION_SUCCEED=CWSPT0112I: Deployed jar files or directory {0} have been successfully deleted.

[CWSPT0113I](#)

DELETION_FAIL=CWSPT0113I: Deletion of the deployed jar files or directory {0} failed.

[CWSPT0201E](#)

CLASSCAST_EXCEPTION=CWSPT0201E: Connecting to server through RMI, The code: {0} is not a valid

script. Exception: {1}.

CWSPT0202E

CLASSNOTFOUND_EXCEPTION=CWSPT0202E: Connecting to server through RMI, Class: {0} can not be found. Exception: {1}.

CWSPT0203E

ILLEGALACCESS_EXCEPTION=CWSPT0203E: Connecting to server through RMI, Class: {0} can not be instantiated due to IllegalAccessException exception. Exception: {1}.

CWSPT0204E

INSTANTIATION_EXCEPTION=CWSPT0204E: Connecting to server through RMI, Class: {0} can not be instantiated. Exception: {1}.

CWSPT0205E

REMOTE_EXCEPTION=CWSPT0205E: Class: {0} fails during the execution of a remote method call. Exception: {1}.

CWSPT0206E

NOT_BOUND_EXCEPTION=CWSPT0206E: URL: {0} can not be looked up. Exception: {1}.

CWSPT0207E

MALFORMEDURL_EXCEPTION=CWSPT0207E: Class: {0} can not be instantiated due to a malformed URL. Exception: {1}.

CWSPT0208E

NUM_OF_VALUES_MISMATCH_WITH_NUM_OF_COLUMNS=CWSPT0208E: The number of values {0} does not match with that of columns: {1}.

CWSPT0209E

COLUMN_VALUE_MISMATCH_COLUMN_TYPE=CWSPT0209E: The column value {0} at position {1} does not match with the type {2} of the corresponding column {3}.

CWSPT0210E

NO_SUCH_INTEGER_VALUE=CWSPT0210E: There is no such integer value at position: {0}.

CWSPT0211E

NO_SUCH_FLOAT_VALUE=CWSPT0211E: There is no such float value at position: {0}.

CWSPT0212E

NOT_SUCH_COLUMN_NAME=CWSPT0212E: There is no such a column with a name: {0}.

CWSPT0213E

NO_SUCH_LONG_VALUE=CWSPT0213E: There is no such long value at position: {0}.

CWSPT0214E

NO_SUCH_BOOLEAN_VALUE=CWSPT0214E: There is no such boolean value at position: {0}.

CWSPT0215E

NO_SUCH_STRING_VALUE=CWSPT0215E: There is no such string value at position: {0}.

CWSPT0301E

UNKNOWNHOST_EXCEPTION=CWSPT0301E: DownstreamBBConnection failed because of UnknownHostException with hostname {0}. Exception: {1}.

CWSPT0302E

IO_EXCEPTION=CWSPT0302E: DownstreamBBConnection failed because an I/O exception of some sort has occurred with hostname {0}. Exception: {1}.

CWSPT0303E

IO_EXCEPTION_2=CWSPT0303E: DownstreamBBConnection: write failed because an I/O exception of some sort has occurred with hostname {0}. Exception: {1}. Retrying.

CWSPT0304E

IO_EXCEPTION_3=CWSPT0304E: DownstreamBBConnection: cannot close because an I/O exception of some sort has occurred with hostname {0}. Exception: {1}. Retrying.

CWSPT0305E

ALREADY_BOUND_EXCEPTION=CWSPT0305E: Cannot set host services: Exception: {0}.

CWSPT0306E

RUNTIME_EXCEPTION=CWSPT0306E: LocalClientConnection caught exception: Exception: {0}.

CWSPT0307E

IO_EXCEPTION_4=CWSPT0307E: Listener start() caught exception with a broker name {0} and port {1}. Exception: {2}.

CWSPT0308E

BROKERTOHOST_NULL=CWSPT0308E: Broker-to-Host map is null and it is not a multi-broker setup.

CWSPT0309E

UNDEFINED_HOSTNAME=CWSPT0309E: The host name for broker {0} cannot be found.

CWSPT0310E

BROKERTOPORT_NULL=CWSPT0310E: The Broker to port mapping is not initialized.

CWSPT0311E

UNDEFINED_PORT=CWSPT0311E: The port number for broker {0} is not defined.

CWSPT0312E

GENERAL_EXCEPTION=CWSPT0312E: Caught Exception: {0}.

CWSPT0313E

GRAPH_BUILD_FAILED_EXCEPTION=CWSPT0313E: An exception is encountered when loading the graph: {0}.

CWSPT0315E

DUPLICATE_NAME_EXCEPTION=CWSPT0315E: Cannot set host services because of deuplicated names. The exception message is {0}.

CWSPT0501E

RELATION_NOT_EXIST=CWSPT0501E: The view name: {0} is not found.

CWSPT0502E

RELATION_EXIST=CWSPT0502E: The relation or table: {0} exists.

CWSPT0503E

VIEW_OR_RELATION_EXIST=CWSPT0503E: The view or relation: {0} around line : {1} exists.

CWSPT0504E

TABLE_NOT_EXIST_IN_FROM_CLAUSE=CWSPT0504E: The view: {0} referred around line : {1} does not exist in FROM clause.

CWSPT0505E

TABLE_IN_FROM_CLAUSE_NOT_EXIST=CWSPT0505E: The table: {0} referred around line : {1} in FROM clause does not exist.

CWSPT0506E

COLUMN_NOT_EXIST_IN_A_TABLE=CWSPT0506E: The column: {0} referred to around line : {1} does not exist in the view with a name or alias: {2}.

CWSPT0507E

COLUMN_NOT_EXIST_IN_BOTH_TABLES=CWSPT0507E: The column: {0} referred does not exist in both relations: {1} and {2} around line : {3}.

CWSPT0508E

COLUMN_NOT_EXIST_IN_ANY_TABLE=CWSPT0508E: The column: {0} referred around line : {1} does not exist in any view.

CWSPT0509E

COLUMN_REF_AMBIGUITY=CWSPT0509E: The column: {0} referred to around line : {1} is ambiguous in the current context.

CWSPT0510E

NOT_A_STREAM=CWSPT0510E: The name {0} does not refer to a stream.

CWSPT0511E

NOT_A_VIEW=CWSPT0511E: The name {0} does not refer to a derived view.

CWSPT0512E

NO_TOPOLOGY_PROVIDED=CWSPT0512E: No brokers are provided for deployment.

CWSPT0513E

NO_VIEW_PROVIDED=CWSPT0513E: No views are provided for deployment.

CWSPT0514E

RELATION_OR_VIEW_USED=CWSPT0514E: The relation or view: {0} is referenced by other views.

CWSPT0515E

INVALID_SELECTED_ITEMS=CWSPT0515E: The view: {0} contains an invalid expression in selection list after name resolution: {1} with either wild card (*) or fully-qualified column name without a correlation name around line : {2}.

CWSPT0516E

LOWER_BOUND_GREATER_THAN_UPPERBOUND=CWSPT0516E: The lower bound: {0} is greater than the upper bound: {1} around line : {2}.

CWSPT0517E

LOWER_BOUND_SMALLER_THAN_LIMIT=CWSPT0517E: The lower bound: {0} is smaller than the lower bound limit of the type: {1} around line : {2}.

CWSPT0518E

UPPER_BOUND_GREATER_THAN_LIMIT=CWSPT0518E: The upper bound: {0} is greater than the upper bound limit of the type: {1} around line : {2}.

CWSPT0519E

NAME_EXIST=CWSPT0519E: The name: {0} exists and it can not be used again in the same query.

CWSPT0551E

NOT_SMILE_MESSAGE=CWSPT0551E: This is not a message with a type of either PublishedMessage or SubscribedMessage.

CWSPT0552E

NOT_SUCH_KEY_COLUMN_NAME=CWSPT0552E: There is no such a key column with a name: {0} in a view: {1}.

CWSPT0553E

NOT_SUCH_KEY_COLUMN_INDEX=CWSPT0553E: There is no such a key column with an index: {0} in a view: {1}.

CWSPT0554E

NOT_SUCH_NON_KEY_COLUMN_NAME=CWSPT0554E: There is no such a non-key column with a name: {0} in a view: {1}.

CWSPT0555E

NOT_SUCH_NON_KEY_COLUMN_INDEX=CWSPT0555E: There is no such a non-key column with an index: {0} in a view: {1}.

CWSPT0556E

VIEW_NAME_WRONG=CWSPT0556E: The view name {0} in the message is either null, empty, or wrong.

CWSPT0557E

KEY_COLUMN_NOT_EXIST=CWSPT0557E: The key column {0} in the view {1} did not get set up.

CWSPT0558E

BROKER_EXIST=CWSPT0558E: A broker with name {0} has already existed.

CWSPT0559E

VIEW_NOT_DEPLOYED=CWSPT0559E: The view with name {0} has not been deployed.

CWSPT0601E

APPLYASPECT_CAUGHT_EXCEPTION=CWSPT0601E: applyAspect() caught exception: {0}

CWSPT0602E

UNSUPPORTED_STATEMENT=CWSPT0602E: The statement {0} is not supported by SMILE SQL.

CWSPT0611E

COMPILE_CAUGHT_EXCEPTION=CWSPT0611E: "compile()" caught exception: {0}

CWSPT0612E

CHECKTYPEANDSET_CAUGHT_EXCEPTION=CWSPT0612E: "checkTypeAndSet()" caught exception {0}

CWSPT0613E

FATAL_ERROR_DURING_PLACEMENT=CWSPT0613E: Fatal error during placement. Exception = {0}

CWSPT0614E

INCOMPATIBLE_TYPES_ARE_COMPARED=CWSPT0614E: Incompatible types are compared: {0}

CWSPT0615E

FAILED_TO_PERFORM_JOIN_OPERATION=CWSPT0615E: Fatal error: "Failed to perform join operation : {0}"

CWSPT0616E

COULDNT_LOAD_UNITS_FILE=CWSPT0616E: Fatal error: "Could not load units file: {0}"

CWSPT0618E

MMPL_COMPILE_EXCEPTION=CWSPT0618E: Fatal error: "mmpl compile exception: {0}"

CWSPT0619E

UNKNOWN_ERROR_CYCLICAL_DEPENDENCY_DETECTED=CWSPT0619E: Compiler Error. "Cyclical dependency detected."

CWSPT0620E

SYNTAX_ERROR_IN_SQL_INPUT_FILE=CWSPT0620E: There is a syntax error: {0}

CWSPT0621E

EXCEPTION_CAUGHT=CWSPT0621E: Fatal error. Exception caught: {0}

CWSPT0622E

FATAL_ERROR_ENCOUNTERED_RUNNING_PLACEMENT_SERVICE=CWSPT0622E: There is a fatal error encountered when running the placement service: {0}

CWSPT0623E

RELATION_NOT_EXISTEXCEPTION=CWSPT0623E: Fatal error: "The stream or view does not exist: {0}"

CWSPT0624W

CANNOT_FIND_FIELD_AT_INDEX=CWSPT0624W: Fatal error: "The field with index {0} cannot be found."

CWSPT0626W

STATELESS_TRANSFORM_TIMEOUT_COUNTER_GREATER_THAN_ZERO=CWSPT0626W: Fatal error: "Stateless transform timeout counter: {0} > 0"

CWSPT0630W

CANNOT_FULLY_ERASE=CWSPT0630W: Cannot fully erase the directory {0} : {1}

CWSPT0633I

GETHASHTABLESIGNATURESTRRAW_HASHTABLE_IS_NULL=CWSPT0633I: Fatal error: "getHashTableSignatureStrRaw() hashTable is null"

CWSPT0634W

CANNOT_CREATE_DIRECTORY_TRY_PROCEEDING_FORWARD=CWSPT0634W: Cannot create directory. Continuing.

CWSPT0641I

PLACEMENTSERVICE_CREATED=CWSPT0641I: "PlacementService created"

CWSPT0642E

PLACEMENT_INITCOMMUNICATIONS_ERROR=CWSPT0642E: Placement error: {0}

CWSPT0643E

LINK_PLACEMENT_FAILED=CWSPT0643E: Link placement failed.

CWSPT0645I

CURRENT_DIRECTORY=CWSPT0645I: The smile repository directory is {0}

CWSPT0750E

FEWER_THAN_TWO_JOIN=CWSPT0750E: Severe Error: "Query rewriter generated a join with fewer than 2 operands".

CWSPT0751E

INCOMPATIBLE_MERGE=CWSPT0751E: Error: Merged relations have incompatible maps.

CWSPT0752E

INCONSISTENT_COMBINE=CWSPT0752E: Severe Error: Query rewriter detected compiler error in COMBINE at MMPL line {0} position {1}.

CWSPT0753E

MISMATCHED_RENAMES=CWSPT0753E: Severe Error: Query rewriter detected compile error in COMBINE at MMPL line {0} position {1}.

CWSPT0754E

TWO_MAPS_SHARE_KEY=CWSPT0754E: Severe Error: Query rewriter detected compile error in JOIN.

CWSPT0755E

MISSING_ACTION=CWSPT0755E: Severe Error: "Query rewriter could not find action for column named {0}".

CWSPT0756E

MULTIPLE_MAPS_CONTAIN_COLUMNS=CWSPT0756E: Severe Error: "Query rewriter found same set of columns in multiple maps."

CWSPT0757E

COMBINE_NON_INTEGER=CWSPT0757E: Columns {2} and {3} in view {4} cannot be combined into an integer type.

CWSPT0758E

COMBINE_KEY_AND_NONKEY=CWSPT0758E: Columns {2} and {3} in view {4} cannot be combined.

CWSPT0759E

COMBINE_GENERATES_DUPLICATE=CWSPT0759E: Operation {2} in relation {4} generates a duplicate column {3} near line {0} position {1}.

CWSPT0760E

DUPLICATE_JOIN_COLUMN=CWSPT0760E: Operation {2} in relation {4} has a column {3} appearing in both operands but not merged, near line {0} position {1}.

CWSPT0761E

DUPLICATE_RELATION=CWSPT0761E: Statement redefines a view named {3} near line {0} position {1}.

CWSPT0762E

RENAME_GENERATES_DUPLICATE=CWSPT0762E: Renaming a column {3} in view {4} generates a duplicate name.

CWSPT0763E

SPLIT_GENERATES_DUPLICATE=CWSPT0763E: Splitting a column generates a duplicate column name {3}.

CWSPT0764E

MERGE_NON_TIMED=CWSPT0764E: Merge of non-time-keyed views.

CWSPT0765E

MERGE_INCOMPATIBLE_SCHEMAS=CWSPT0765E: Attempt to merge views with incompatible schemas.

CWSPT0766E

SELF_MERGE_ERROR=CWSPT0766E: Attempt to merge view {2} with itself.

CWSPT0767E

NOT_SPLITTABLE_COLUMN=CWSPT0767E: Cannot split column {3} in view {2}.

CWSPT0768E

MMPL_SYNTAX_EXCEPTION=CWSPT0768E: Syntax exception in MMPL. Unexpected token is {2}.

CWSPT0769E

SQL_SYNTAX_EXCEPTION=CWSPT0769E: Syntax exception in SQL query. Unexpected token is {2}.

CWSPT0770E

NO_TIME_COLUMN=CWSPT0770E: No time column generated for stream {2}.

CWSPT0771E

CANNOT_RESOLVE_COLUMN_NAME=CWSPT0771E: Reference to an undefined column {2} in view {3}.

CWSPT0772E

CANNOT_RESOLVE_MERGED_COLUMN_NAME=CWSPT0772E: Reference to an undefined JOIN USING column {2} in view {3}

CWSPT0773E

UNDEFINED_VIEW_NAME_IN_SUBSCRIPTION=CWSPT0773E: An MMPL Subscription statement refers to an undefined view {2}.

CWSPT0774E

UNDEFINED_REFERENCE_IN_ALIAS=CWSPT0774E: An MMPL Assignment statement refers to an undefined view {2}.

CWSPT0775E

UNDEFINED_PROJECTION_COLUMN=CWSPT0775E: A projection operation is referring to an undefined column {2}.

CWSPT0776E

RENAMED_COLUMN_NOT_FOUND=CWSPT0776E: The column {2} being renamed in view {3} does not exist.

CWSPT0777E

WINDOW_SIZE_NOT_GE_0=CWSPT0777E: The window size in view {2} is not greater than 0.

CWSPT0778E

UNKNOWN_EXPRESSION=CWSPT0778E: An unknown expression appeared in an MMPL WHERE or ADD clause at line {0} , position {1}.

CWSPT0779E

ILLEGAL_RANGE=CWSPT0779E: Illegal numeric range at line {0} , position {1}.

CWSPT0780E

NON_SCALAR_SELECT=CWSPT0780E: Expression in WHERE clause is not a scalar expression at line {0} , position {1}.

CWSPT0781E

NON_RELATIONAL_ALIAS=CWSPT0781E: An MMPL assignment statement has a right-hand side which is not a relational expression at line {0} , position {1}.

CWSPT0782E

MISPLACED_AGGREGATION_EXPRESSION=CWSPT0782E: An MMPL aggregation expression appeared elsewhere than in a SELECT or EXTEND at line {0} , position {1}.

CWSPT0783E

UNRESOLVABLE_EXPRESSION=CWSPT0783E: The expression {2} cannot be resolved at line {0} , position {1}.

CWSPT0784E

UNRESOLVABLE_ARITHMETIC_TERM=CWSPT0784E: The operand to the operator {2} cannot be resolved at line {0}, position {1}.

CWSPT0785E

ILLEGAL_PROJECTION=CWSPT0785E: Column {3} of view {2} depends on a column that is being projected away.

CWSPT0786E

ILLEGAL_STRING_SIZE=CWSPT0786E: String size operator was applied to non-string.

CWSPT0787E

ILLEGAL_ENUMERATION=CWSPT0787E: ENUMERATION is not supported.

CWSPT0788E

CANNOT_BOOL_AGGREGATES=CWSPT0788E: Cannot apply boolean operations to aggregates.

CWSPT0789E

BAD_INPUTS_FOR_COMBINE=CWSPT0789E: Operation {7} in view {6} combines column {2} of type {4} with column {3} of type {5}.

CWSPT0790E

INCOMPATIBLE_TYPES_FOR_COMPARISON=CWSPT0790E: Expression in view {6} compares column {2} of type {4} with column {3} of type {5}.

CWSPT0791E

EXPECTED_NUMERIC_TYPE=CWSPT0791E: Column {2} of type {4} in view {6} was expected to be numeric.

CWSPT0792E

EXPECTED_BOOLEAN_TYPE=CWSPT0792E: Column {2} of type {4} in view {6} was expected to be boolean.

CWSPT0793E

ILLEGAL_LATEST=CWSPT0793E: Column {2} in view {3} does not depend upon a time key.

CWSPT0794E

ILLEGAL_MERGE_OPERANDS=CWSPT0794E: View {6} attempts to merge views that are not time-keyed.

CWSPT0795E

ILLEGAL_WHERE_EXPRESSION=CWSPT0795E: View {6} contains WHERE clause of type {4}.

CWSPT0796E

INCOMPATIBLE_JOIN_COLUMNS=CWSPT0796E: View {6} joins column {2} of type {4} with column {3} of type {5}.

CWSPT0797E

ILLEGAL_SPLIT_TYPE=CWSPT0797E: Column {2} of type {4} in view {6} is of illegal type for SPLIT.

CWSPT0798E

ILLEGAL_WINDOW_OPERAND=CWSPT0798E: View {6} applies WINDOW to a view that is not time-keyed.

CWSPT0799E

CANNOT_COMBINE_INCOMPATIBLE_UNITS=CWSPT0799E: View {6} attempts to combine column {2} with units {4} with column {3} with units {5}.

CWSPT0800E

INCOMPATIBLE_UNITS=CWSPT0800E: View {6} uses incompatible units {4} and {5}.

CWSPT0801E

UNKNOWN_AGGREGATION_OPERATOR=CWSPT0801E: Unknown aggregation operator at line {0}, position {1}.

CWSPT0802E

UNKNOWN_TYPE_FOR_AGGREGATION=CWSPT0802E: Unknown type for aggregation.

CWSPT0803E

DIFFERENT_SIGNATURE_FOR_MERGE=CWSPT0803E: Views {2} and {3} have different types.

CWSPT0804E

OPERATOR_NEEDS_TWO_INPUTS=CWSPT0804E: Operator needs at least two operands.

CWSPT0805E

UNKNOWN_OPERATOR_TYPE=CWSPT0805E: Unknown operator type {0}.

CWSPT0806E

NOT_SHOULD_HAVE_1_OPERAND=CWSPT0806E: NOT operator should have one operand.

CWSPT0807E

UNSUPPORTED_JOIN_RULE=CWSPT0807E: Severe error: "Query rewriter failed to decompose a JOIN: Environment = "{2}"."

CWSPT0808E

SPLIT_COLUMN_SMALLER_THAN_DIVISOR=CWSPT0808E: SPLIT operation has a divisor larger than the range of the column being split.

CWSPT0809E

SPLIT_COLUMN_HAS_ZERO_RANGE=CWSPT0809E: SPLIT column has zero range.

CWSPT0810E

CONSTANTS_WITH_NONNUMERICS=CWSPT0810E: Constant relation has non-numeric columns.

CWSPT0811E

CONSTANTS_WITH_NONKEYS=CWSPT0811E: Constant relation should not have non-key columns

CWSPT0812E

COLUMN_LACKS_TYPE=CWSPT0812E: Compiler error: "Column {2} in view {3} lacks a type."

CWSPT0813E

INVALID_MERGE=CWSPT0813E: Compiler error: "Invalid Merge: Merge has more than one visibility column."

CWSPT0814E

TYPE_COMPARE_FAILED=CWSPT0814E: Compiler error: "Type compare failed: exception data: {2}."

CWSPT0815E

UNITS_FOR_NONNUMERICS=CWSPT0815E: compiler error: "Attempt to extract units for non-numeric fields."

CWSPT0816E

INVALID_TRANSFORM=CWSPT0816E: compiler error: "An invalid transform {2} was attempted."

CWSPT0817E

UNIT_NOT_TIME=CWSPT0817E: Compiler error: "A unit {2} is expected to be a time unit and is not."

CWSPT0818E

INVALID_OP_ON_UNIT=CWSPT0818E: Compiler error: "An invalid operation was attempted on unit {2}."

CWSPT0819E

INVALID_UNIT=CWSPT0819E: Compiler error: "An invalid unit {2} was detected."

CWSPT0820E

UNKNOWN_BASIC_TYPE=CWSPT0820E: Compiler error: "An invalid MMPL type {2} was detected."

CWSPT0821E

NO_RULE_FOR_CLASS=CWSPT0821E: Compiler error: "The transform with class {2} has no rule."

CWSPT0822E

UNKNOWN_EXPRESSION_INPUT=CWSPT0822E: Probable compiler error: "The expression has an unknown form."

CWSPT0823E

NO_TYPES_IN_ADD=CWSPT0823E: Compiler error: "An EXTEND has no types in its ADD list."

CWSPT0824E

UNSUPPORTED_TYPE=CWSPT0824E: Compiler error: "Type {2} is unsupported in MMPL."

CWSPT0825E

NO_TYPE_FOR_SUBSCRIPTION=CWSPT0825E: Subscription {2} could not be assigned a type.

CWSPT0826E

RENAME_ENTRY_GENERATES_DUPLICATE=CWSPT0826E: Trying to rename a query from {0} to {1} caused a duplication.

CWSPT0827E

CANNOT_REMOVE_BECAUSE_USED_BY=CWSPT0827E: Cannot remove view {0} because it is used by view(s) {1}.

CWSPT0828E

CYCLIC_DEPENDENCY=CWSPT0828E: The update would create a cyclic dependency of views

CWSPT0829E

ADDITION_CREATES_CONFLICT=CWSPT0829E: The view {0} already exists. Trying to add it again fails.

CWSPT0830E

MULTIPLE_VISIBILITY_COLUMNS_APPLY=CWSPT0830E: Severe compiler error: "Multiple visibility columns apply. Invariant violated in {0}."

CWSPT0831E

CANNOT_COMPILE_GENERATED_JAVA_CODE=CWSPT0831E: Cannot compile generated Java code. See Log.

CWSPT0832E

COULD_NOT_RUN_PROCESS=CWSPT0832E: Probable compiler error: "Could not run process: {0} ({1})."

CWSPT0833E

NON_KEY_COLUMN_MUST_BE_OF_EVOLVABLE_TYPE=CWSPT0833E: Severe compiler error: "Non-key column {0} must be of evolvable type."

CWSPT0834E

KEY_COLUMN_MUST_BE_OF_NON_EVOLVABLE_TYPE=CWSPT0834E: Compiler error: "Key column: {0} must be of non-evolvable type."

CWSPT0835E

PARAMETER_NULL_IS_INVALID=CWSPT0835E: Compiler error: "Parameter null is invalid."

CWSPT0836E

TYPE_NAME_CANNOT_BE_NULL=CWSPT0836E: Compiler error: "Type name cannot be null."

CWSPT0837E

MORE_THAN_TWO_MESSAGE_JOIN_NOT_ALLOWED=CWSPT0837E: There must be two message streams to join:{0}.

CWSPT0838E

INTERNAL_COMPILER_ERROR_FOR_LOGGING=CWSPT0838E: Compiler Error: "In {0} with a message : {1}."

CWSPT0839E

INTERNAL_COMPILER_ERROR=CWSPT0839E: Compiler Error: "In {0}."

CWSPT0840E

CAN_NOT_FIND_PARENT_NODE=CWSPT0840E: Compiler error: "Cannot find parent node in {0}."

CWSPT0843E

ADS_EXCEPTION_1=CWSPT0843E: Compiler error: "Multiple messages apply to the same equi-column command."

CWSPT0844E

RUNTIME_EXCEPTION=CWSPT0844E: Compiler error: "Method not implemented."

CWSPT0845E

ASSERTION_ERROR=CWSPT0845E: Compiler error: "Simple join has more than one joined result:{0}."

CWSPT0846E

ADS_EXCEPTION_2=CWSPT0846E: Compiler error: "Incompatible inputs for merge in ADS stage. Some checks failed earlier in line {0} and position {1}."

CWSPT0847E

ILLEGAL_STATE_EXCEPTION_1=CWSPT0847E: Compiler error: "This case should be handled in rewriter: {0}."

CWSPT0848E

ADS_EXCEPTION_3=CWSPT0848E: Compiler error: "Message is effected by multiple split operations. Invalid state in line {0} and position {1}."

CWSPT0849E

ADS_EXCEPTION_4=CWSPT0849E: Severe compiler error: "Aggregation should have been factored stand-alone expression in line {0} and position {1}."

CWSPT0850E

ADS_EXCEPTION_5=CWSPT0850E: Compiler error: "Duplicate message found in line {0} and position {1}."

CWSPT0851E

ADS_EXCEPTION_6=CWSPT0851E: Compiler error: "Message not found with columns: {0} in line {1} and position {2}."

CWSPT0852E

CODEGEN_EXCEPTION=CWSPT0852E: Compiler error: "Cannot find next node process:in line {0} and position {1}."

CWSPT0853E

CODEGEN_EXCEPTION_1=CWSPT0853E: Compiler error: "Unknown type".

CWSPT0854E

CODEGEN_EXCEPTION_2=CWSPT0854E: Compiler Error: "Cannot compute index for column {0} in Hash Table {1}."

CWSPT0855E

CODEGEN_EXCEPTION_3=CWSPT0855E: Compiler error: "Unknown column {0} and type {1}."

CWSPT0856E

CODEGEN_EXCEPTION_4=CWSPT0856E: Compiler error: "Unsupported equality statement at line {0} and position {1}."

CWSPT0857E

CODEGEN_EXCEPTION_5=CWSPT0857E: Compiler error: "Should have been factored out by splitter component in line: {0} and position: {1}."

CWSPT0858E

CODEGEN_EXCEPTION_6=CWSPT0858E: Compiler error: "Not supported: {0} in {1}."

CWSPT0859E

CODEGEN_EXCEPTION_7=CWSPT0859E: Compiler error: "The operator {0} is not supported on String."

CWSPT0860E

CODEGEN_EXCEPTION_8=CWSPT0860E: Compiler error: "Unknown case".

CWSPT0861E

CODEGEN_EXCEPTION_9=CWSPT0861E: Compiler error: "Storing silence is not supported for client hashes for now."

CWSPT0862E

CODEGEN_EXCEPTION_10=CWSPT0862E: Compiler error: "Compiler generated code is not designed to get data from client table."

CWSPT0863E

ADS_EXCEPTION_7=CWSPT0863E: Compiler error: "node: {0} should have had broker assignment in line {1} and position {2}."

CWSPT0864E

JOIN_FAILED=CWSPT0864E: Compiler error: "Join failed: {2}".

CWSPT0865E

CANNOT_FIND_TOOLJAR_EXCEPTION=CWSPT0865E: Cannot find class com.sun.tools.javac.Main. The tools.jar is probably not in the class path.

CWSPT0867E

ILLEGAL_STATE_EXCEPTION=CWSPT0867E: Compiler error: "There is a name conflict."

CWSPT0001E

SHOULD_NOT_OCCUR_REPORT_TO_DEVELOPER=CWSPT0001E: Should not occur: report to developer;
message= {0}

Explanation

An unexpected error condition has occurred.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0004E

CANNOT_CREATE_SOFT_CHECKPOINTING_LOG_FOR_TRANSFORM=CWSPT0004E: Cannot create soft checkpointing log for transform: {0} : {1}

Explanation

An IO error condition occurred while creating a soft checkpointing service for the identified transform. Query processing continues. However, error recovery will not be possible.

User response

Check that database used for message logging is properly configured. If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0005E

LOGGING_ERROR_EXCEPTION_RAISED=CWSPT0005E: Logging error. Exception raised: {0}

Explanation

An error condition occurred while logging a message. Query processing continues. However, error recovery will not be possible.

User response

Check that database used for message logging is properly configured. If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0006E

NO_BUFFERED_LOG_MESSAGE=CWSPT0006E: There is no buffered log message in the topic. The local log index is {0}.

Explanation

An error condition occurred while logging a message. Query processing continues. However, error recovery will not be possible.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0007E

MSG_LOGGING_TURNED_OFF=CWSPT0007E: Msg logging turned off

Explanation

An error condition occurred while replaying logged messages. Query processing continues. However, error recovery will not be possible.

User response

Check that database used for message logging is properly configured. If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0008E

CANNOT_REPLAY_MESSAGE=CWSPT0008E: Cannot replay msg from {0} to {1} because {2}

Explanation

Replay of the specified range of messages cannot be completed due to the referenced Exception. Query processing continues. However, error recovery will not be possible.

User response

Check that database used for message logging is properly configured. If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0009E

CANNOT_SET_HOST_SERVICES=CWSPT0009E: Cannot set host services: {0}

Explanation

The SMILE Runtime cannot be started due to the referenced exception. This generally happens because of attempt to start SMILE when it is already running.

User response

Restart the SMILE Runtime. If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0010W

NO_AUX_INFO=CWSPT0010W: The dummy Soft Checkpointing service could not find auxiliary information for a transform

Explanation

The dummy Soft Checkpointing service could not find auxiliary information for a transform.

User response

none.

Parent topic: [CWSPT](#)

CWSPT0011W

CANNOT_RECLAIM_SC_LOG_OF_TRANSFORM=CWSPT0011W: Cannot reclaim Soft Checkpointing log of transform with ID {0}. The exception message is {1}.

Explanation

The Soft Checkpointing log of the referenced transform could not be reclaimed.

User response

Check that database used for message logging is properly configured.

Parent topic: [CWSPT](#)

CWSPT0012W

CANNOT_PUT_SOFT_CHECKPOINT_RECORDS=CWSPT0012W: Cannot put soft checkpointing log records! {0}

Explanation

Soft Checkpointing records cannot be written to the log due to the referenced exception.

User response

Check that database used for message logging is properly configured.

Parent topic: [CWSPT](#)

CWSPT0013W

SMILETIMER_REQUESTTIMEOUT_RAISED_EXCEPTION=CWSPT0013W: Exception encountered when scheduling a timer with interval {0}. The exception message is {1}.

Explanation

An exception is caught when schedule a timer.

User response

Check the exception message for details

Parent topic: [CWSPT](#)

CWSPT0014W

CANNOT_FIND_WRAPPER_MAKER_FOR_TOPIC=CWSPT0014W: Severe Runtime error: "Cannot find wrapper maker for topic {0}, methodID {1}"

Explanation

An error was encountered at runtime: "A SMILE WrapperMaker could not be found for the referenced topic and method."

User response

This is a system bug. Report this message to IBM support team.

Parent topic: [CWSPT](#)

CWSPT0015I

RECOVERING_TRANSFORM=CWSPT0015I: Recovering transform {0}.

Explanation

The Soft Checkpointing Service is recovering the referenced transform.

User response

none.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0103E

INSTANTIATION_EXCEPTION=CWSPT0103E: Class: {0} can not be instantiated.

Explanation

The specified class object cannot be instantiated because it is an interface or is an abstract class.

User response

Check your generated code to see if the class is there.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0104E

ILLEGALACCESS_EXCEPTION=CWSPT0104E: Class: {0} can not be instantiated due to IllegalAccessException exception.

Explanation

The currently executing method does not have access to the definition of constructor.

User response

Check the security configuration of the execution environment.

Parent topic: [CWSPT](#)

CWSPT0105E

CLASSNOTFOUND_EXCEPTION=CWSPT0105E: Class: {0} can not be found.

Explanation

No definition for the class with the specified name could be found.

User response

Check your generated code and its claspath.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0106E

MALFORMEDURL_EXCEPTION=CWSPT0106E: Class: {0} can not be instantiated due to a malformed URL.

Explanation

Either no legal protocol could be found in a specification string or the string could not be parsed.

User response

Check your URL to see whether it is correct.

Parent topic: [CWSPT](#)

CWSPT0108E

VIEW_NOT_FOUND=CWSPT0108E: View: {0} can not be retrieved.

Explanation

Either the view does not exist or there is a network connection problem to prevent the view from being retrieved from the view repository.

User response

Check your connection to the repository server or whether the view exists.

Parent topic: [CWSPT](#)

CWSPT0112I

DELETION_SUCCEED=CWSPT0112I: Deployed jar files or directory {0} have been successfully deleted.

Explanation

Deployed jar files or directory have been successfully deleted.

User response

none

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0113I

DELETION_FAIL=CWSPT0113I: Deletion of the deployed jar files or directory {0} failed.

Explanation

Cannot delete the deployed jar files or directory.

User response

Please report this problem to IBM support team

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0201E

CLASSCAST_EXCEPTION=CWSPT0201E: Connecting to server through RMI, The code: {0} is not a valid script.
Exception: {1}.

Explanation

While connecting to a SMILE server through RMI, No definition for the class with the specified name could be found.

User response

Check your generated code and its claspath.

Parent topic: [CWSPT](#)

CWSPT0202E

CLASSNOTFOUND_EXCEPTION=CWSPT0202E: Connecting to server through RMI, Class: {0} can not be found.
Exception: {1}.

Explanation

While connecting to a SMILE server through RMI, No definition for the class with the specified name could be found.

User response

Check your generated code and its claspath.

Parent topic: [CWSPT](#)

CWSPT0203E

ILLEGALACCESS_EXCEPTION=CWSPT0203E: Connecting to server through RMI, Class: {0} can not be instantiated due to IllegalAccessException exception. Exception: {1}.

Explanation

While connecting to a SMILE server through RMI, The currently executing method does not have access to the definition of constructor.

User response

Check the security configuration of the execution environment.

Parent topic: [CWSPT](#)

CWSPT0204E

INSTANTIATION_EXCEPTION=CWSPT0204E: Connecting to server through RMI, Class: {0} can not be instantiated. Exception: {1}.

Explanation

While connecting to a SMILE server through RMI, The specified class object cannot be instantiated because it is an interface or is an abstract class.

User response

Check the classpath of the execution environment to be sure required RMI support classes are present.

Parent topic: [CWSPT](#)

CWSPT0205E

REMOTE_EXCEPTION=CWSPT0205E: Class: {0} fails during the execution of a remote method call. Exception: {1}.

Explanation

The exception may be caused by a broken connection to a remote server or the remote server is down.

User response

Check your remote server for hosting the SMILE services and make sure that it is running.

Parent topic: [CWSPT](#)

CWSPT0206E

NOT_BOUND_EXCEPTION=CWSPT0206E: URL: {0} can not be be looked up. Exception: {1}.

Explanation

An attempt is made to lookup or unbind in the registry the specified URL that has no associated binding. .

User response

Make sure the remote RMI SMILE service is running.

Parent topic: [CWSPT](#)

CWSPT0207E

MALFORMEDURL_EXCEPTION=CWSPT0207E: Class: {0} can not be instantiated due to a malformed URL.
Exception: {1}.

Explanation

Either no legal protocol could be found in a specification string or the string could not be parsed.

User response

Check the URL to see if it is correct.

Parent topic: [CWSPT](#)

CWSPT0208E

NUM_OF_VALUES_MISMATCH_WITH_NUM_OF_COLUMNS=CWSPT0208E: The number of values {0} does not match with that of columns: {1}.

Explanation

The number of values in a message does not match with that of columns.

User response

Check the message and its schema.

Parent topic: [CWSPT](#)

CWSPT0209E

COLUMN_VALUE_MISMATCH_COLUMN_TYPE=CWSPT0209E: The column value {0} at position {1} does not match with the type {2} of the corresponding column {3}.

Explanation

The column value at position does not match with the type of the corresponding column.

User response

Check the message and its schema.

Parent topic: [CWSPT](#)

CWSPT0210E

NO_SUCH_INTEGER_VALUE=CWSPT0210E: There is no such integer value at position: {0}.

Explanation

There is no such integer value at position.

User response

Check the message and its schema.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0211E

NO_SUCH_FLOAT_VALUE=CWSPT0211E: There is no such float value at position: {0}.

Explanation

There is no such float value at position.

User response

Check the message and its schema.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0212E

NOT_SUCH_COLUMN_NAME=CWSPT0212E: There is no such a column with a name: {0}.

Explanation

There is no such a column with a name.

User response

Check the message and its schema.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0213E

NO_SUCH_LONG_VALUE=CWSPT0213E: There is no such long value at position: {0}.

Explanation

There is no such long value at position.

User response

Check the message and its schema.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0214E

NO_SUCH_BOOLEAN_VALUE=CWSPT0214E: There is no such boolean value at position: {0}.

Explanation

There is no such boolean value at position.

User response

Check the message and its schema.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0215E

NO_SUCH_STRING_VALUE=CWSPT0215E: There is no such string value at position: {0}.

Explanation

There is no such string value at position.

User response

Check the message and its schema.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0301E

UNKNOWNHOST_EXCEPTION=CWSPT0301E: DownstreamBBConnection failed because of UnknownHostException with hostname {0}. Exception: {1}.

Explanation

The IP address of a host could not be determined.

User response

Check your host IP address.

Parent topic: [CWSPT](#)

CWSPT0302E

IO_EXCEPTION=CWSPT0302E: DownstreamBBConnection failed because an I/O exception of some sort has occurred with hostname {0}. Exception: {1}.

Explanation

This was produced by failed or interrupted I/O operations.

User response

Check your network connection.

Parent topic: [CWSPT](#)

CWSPT0303E

IO_EXCEPTION_2=CWSPT0303E: DownstreamBBConnection: write failed because an I/O exception of some sort has occurred with hostname {0}. Exception: {1}. Retrying.

Explanation

This was produced by failed or interrupted I/O operations.

User response

Check your network connection.

Parent topic: [CWSPT](#)

CWSPT0304E

IO_EXCEPTION_3=CWSPT0304E: DownstreamBBConnection: cannot close because an I/O exception of some sort has occurred with hostname {0}. Exception: {1}. Retrying.

Explanation

This was produced by failed or interrupted I/O operations.

User response

Check your network connection.

Parent topic: [CWSPT](#)

CWSPT0305E

ALREADY_BOUND_EXCEPTION=CWSPT0305E: Cannot set host services: Exception: {0}.

Explanation

Host services can not be set up because an existing binding of a queue.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0306E

RUNTIME_EXCEPTION=CWSPT0306E: LocalClientConnection caught exception: Exception: {0}.

Explanation

LocalClientConnection caught exception.

User response

Check your message client.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0307E

IO_EXCEPTION_4=CWSPT0307E: Listener start() caught exception with a broker name {0} and port {1}.
Exception: {2}.

Explanation

Listener start() caught exception.

User response

contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0308E

BROKERTOHOST_NULL=CWSPT0308E: Broker-to-Host map is null and it is not a multi-broker setup.

Explanation

OGHostServices.getHostNamForBroker: not a multi-broker setup.

User response

This should be a single broker setup.

Parent topic: [CWSPT](#)

CWSPT0309E

UNDEFINED_HOSTNAME=CWSPT0309E: The host name for broker {0} cannot be found.

Explanation

The broker host name for broker cannot be found. The host name might be invalid.

User response

Check your configuration to see whether the host name is configured right.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0310E

BROKERTOPORT_NULL=CWSPT0310E: The Broker to port mapping is not initialized.

Explanation

This might be a single broker setup.

User response

Check your configuration to see whether it is a single broker setup.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0311E

UNDEFINED_PORT=CWSPT0311E: The port number for broker {0} is not defined.

Explanation

The port number is not configured for the broker.

User response

Check your deployment to see whether the port number is provided or not.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0312E

GENERAL_EXCEPTION=CWSPT0312E: Caught Exception: {0}.

Explanation

Runtime exception related to socket connection.

User response

Check for network.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0313E

GRAPH_BUILD_FAILED_EXCEPTION=CWSPT0313E: An exception is encountered when loading the graph: {0}.

Explanation

The graph loading failed due to an exception.

User response

Check for the generated code for the runtime, or report to IBM customer support.

Parent topic: [CWSPT](#)

CWSPT0315E

DUPLICATE_NAME_EXCEPTION=CWSPT0315E: Cannot set host services because of deuplicated names. The exception message is {0}.

Explanation

Host services can not be set up because of duplicate names.

User response

If the problem persists,contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0501E

RELATION_NOT_EXIST=CWSPT0501E: The view name: {0} is not found.

Explanation

The view is not found.

User response

Check that spelling of intended view name matches the spelling used in its definition.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0502E

RELATION_EXIST=CWSPT0502E: The relation or table: {0} exists.

Explanation

The relation or table exists.

User response

Choose a different name to avoid duplication.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0503E

VIEW_OR_RELATION_EXIST=CWSPT0503E: The view or relation: {0} around line : {1} exists.

Explanation

The view or relation exists.

User response

Choose a different name to avoid duplication

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0504E

TABLE_NOT_EXIST_IN_FROM_CLAUSE=CWSPT0504E: The view: {0} referred around line : {1} does not exist in FROM clause.

Explanation

The view referred does not exist in FROM clause.

User response

Check that the FROM clause includes the table name and that the reference uses the same spelling.

Parent topic: [CWSPT](#)

CWSPT0505E

TABLE_IN_FROM_CLAUSE_NOT_EXIST=CWSPT0505E: The table: {0} referred around line : {1} in FROM clause does not exist.

Explanation

The view referred in the FROM clause does not exist.

User response

Check that the FROM clause includes the view name and that the reference uses the same spelling.

Parent topic: [CWSPT](#)

CWSPT0506E

COLUMN_NOT_EXIST_IN_A_TABLE=CWSPT0506E: The column: {0} referred to around line : {1} does not exist in the view with a name or alias: {2}.

Explanation

The column referred to does not exist in the view with the name or alias.

User response

Check that the referenced view has that column and that the reference uses the same spelling.

Parent topic: [CWSPT](#)

CWSPT0507E

COLUMN_NOT_EXIST_IN_BOTH_TABLES=CWSPT0507E: The column: {0} referred does not exist in both relations: {1} and {2} around line : {3}.

Explanation

The column referred to does not exist in both views.

User response

Rewrite the query or rename columns to use a column name that exists in both views

Parent topic: [CWSPT](#)

CWSPT0508E

COLUMN_NOT_EXIST_IN_ANY_TABLE=CWSPT0508E: The column: {0} referred around line : {1} does not exist in any view.

Explanation

The column referred to does not exist in any view.

User response

Rewrite the query to use a correct column reference.

Parent topic: [CWSPT](#)

CWSPT0509E

COLUMN_REF_AMBIGUITY=CWSPT0509E: The column: {0} referred to around line : {1} is ambiguous in the current context.

Explanation

The column referred to around line is ambiguous in the current context.

User response

Rename a column or attach a view name qualifier to the column name to eliminate the ambiguity.

Parent topic: [CWSPT](#)

CWSPT0510E

NOT_A_STREAM=CWSPT0510E: The name {0} does not refer to a stream.

Explanation

The name does not refer to a stream.

User response

Make sure that the name refers to a stream.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0511E

NOT_A_VIEW=CWSPT0511E: The name {0} does not refer to a derived view.

Explanation

The name does not refer to a derived view.

User response

Make sure that the name refers to a derived view.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0512E

NO_TOPOLOGY_PROVIDED=CWSPT0512E: No brokers are provided for deployment.

Explanation

No brokers are provided for deployment.

User response

Make sure that brokers are provided for deployment.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0513E

NO_VIEW_PROVIDED=CWSPT0513E: No views are provided for deployment.

Explanation

No views are provided for deployment.

User response

Make sure that views are provided for deployment.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0514E

RELATION_OR_VIEW_USED=CWSPT0514E: The relation or view: {0} is referenced by other views.

Explanation

The relation or view is referenced by other views.

User response

These views must be compiled together -- possible implementation bug.

Parent topic: [CWSPT](#)

CWSPT0515E

INVALID_SELECTED_ITEMS=CWSPT0515E: The view: {0} contains an invalid expression in selection list after name resolution: {1} with either wild card (*) or fully-qualified column name without a correlation name around line : {2}.

Explanation

The view contains an invalid expression in selection list after name resolution with either wild card (*) or fully-qualified column name without a correlation name.

User response

Rewrite the expression.

Parent topic: [CWSPT](#)

CWSPT0516E

LOWER_BOUND_GREATER_THAN_UPPERBOUND=CWSPT0516E: The lower bound: {0} is greater than the upper bound: {1} around line : {2}.

Explanation

The lower bound is greater than the upper bound.

User response

Respecify the lower and/or upper bound.

Parent topic: [CWSPT](#)

CWSPT0517E

LOWER_BOUND_SMALLER_THAN_LIMIT=CWSPT0517E: The lower bound: {0} is smaller than the lower bound limit of the type: {1} around line : {2}.

Explanation

The lower bound is smaller than the lower bound limit of the type.

User response

Either change the type or the bound.

Parent topic: [CWSPT](#)

CWSPT0518E

UPPER_BOUND_GREATER_THAN_LIMIT=CWSPT0518E: The upper bound: {0} is greater than the upper bound limit of the type: {1} around line : {2}.

Explanation

The upper bound is greater than the upper bound limit of the type.

User response

Either change the type or the bound.

Parent topic: [CWSPT](#)

CWSPT0519E

NAME_EXIST=CWSPT0519E: The name: {0} exists and it can not be used again in the same query.

Explanation

The name exists and it can not be used again in the same query.

User response

Choose a different name

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0551E

NOT_SMILE_MESSAGE=CWSPT0551E: This is not a message with a type of either PublishedMessage or SubscribedMessage.

Explanation

This is not a message with a type of either PublishedMessage or SubscribedMessage.

User response

Make sure that your message has a type of either PublishedMessage or SubscribedMessage.

Parent topic: [CWSPT](#)

CWSPT0552E

NOT_SUCH_KEY_COLUMN_NAME=CWSPT0552E: There is no such a key column with a name: {0} in a view: {1}.

Explanation

There is no such a key column with the name.

User response

Make sure that a right key column name is provided.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0553E

NOT_SUCH_KEY_COLUMN_INDEX=CWSPT0553E: There is no such a key column with an index: {0} in a view: {1}.

Explanation

There is no such a key column given the index.

User response

Make sure that a right index for the key column is provided.

Parent topic: [CWSPT](#)

CWSPT0554E

NOT_SUCH_NON_KEY_COLUMN_NAME=CWSPT0554E: There is no such a non-key column with a name: {0} in a view: {1}.

Explanation

There is no such a non-key column with the name

User response

Make sure that a right non-key column name is provided.

Parent topic: [CWSPT](#)

CWSPT0555E

NOT_SUCH_NON_KEY_COLUMN_INDEX=CWSPT0555E: There is no such a non-key column with an index: {0} in a view: {1}.

Explanation

There is no such a non-key column with the name.

User response

Make sure that a right index for the non-key column is provided.

Parent topic: [CWSPT](#)

CWSPT0556E

VIEW_NAME_WRONG=CWSPT0556E: The view name {0} in the message is either null, empty, or wrong.

Explanation

The view name in the message is either null, empty, or wrong.

User response

Make sure that a view name is specified.

Parent topic: [CWSPT](#)

CWSPT0557E

KEY_COLUMN_NOT_EXIST=CWSPT0557E: The key column {0} in the view {1} did not get set up.

Explanation

The key column in the view did not get set up.

User response

Check the message schema.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0558E

BROKER_EXIST=CWSPT0558E: A broker with name {0} has already existed.

Explanation

The broker with the same name exists and can not be used again.

User response

Use a new broker name.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0559E

VIEW_NOT_DEPLOYED=CWSPT0559E: The view with name {0} has not been deployed.

Explanation

The view has not been deployed before it can be subscribed.

User response

Make sure that the view is deployed before it can be subscribed.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0601E

APPLYASPECT_CAUGHT_EXCEPTION=CWSPT0601E: applyAspect() caught exception: {0}

Explanation

This is an internal error.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0602E

UNSUPPORTED_STATEMENT=CWSPT0602E: The statement {0} is not supported by SMILE SQL.

Explanation

The referenced statement is not part of SMILE SQL.

User response

Change the query to use supported facilities.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0611E

COMPILE_CAUGHT_EXCEPTION=CWSPT0611E: "compile()" caught exception: {0}

Explanation

"compile()" caught the referenced exception. Use the exception message to resolve the problem.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0612E

CHECKTYPEANDSET_CAUGHT_EXCEPTION=CWSPT0612E: "checkTypeAndSet()" caught exception {0}

Explanation

"checkTypeAndSet()" caught the referenced exception. Use the exception message to resolve the problem.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0613E

FATAL_ERROR_DURING_PLACEMENT=CWSPT0613E: Fatal error during placement. Exception = {0}

Explanation

Fatal error during placement.

User response

contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0614E

INCOMPATIBLE_TYPES_ARE_COMPARED=CWSPT0614E: Incompatible types are compared: {0}

Explanation

Incompatible types are compared.

User response

Correct query.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0615E

FAILED_TO_PERFORM_JOIN_OPERATION=CWSPT0615E: Fatal error: "Failed to perform join operation : {0}"

Explanation

Fatal error: "joinHelper() caught the referenced exception." Use the exception message to report the problem.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0616E

COULDNT_LOAD_UNITS_FILE=CWSPT0616E: Fatal error: "Could not load units file: {0}"

Explanation

Fatal error. Report the problem to IBM.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0618E

MMPL_COMPILE_EXCEPTION=CWSPT0618E: Fatal error: "mmpl compile exception: {0}"

Explanation

Fatal error: "mmpl compile exception."

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0619E

UNKNOWN_ERROR_CYCLICAL_DEPENDENCY_DETECTED=CWSPT0619E: Compiler Error. "Cyclical dependency detected."

Explanation

This is a compiler error

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0620E

SYNTAX_ERROR_IN_SQL_INPUT_FILE=CWSPT0620E: There is a syntax error: {0}

Explanation

There is a syntax error in the SQL input file.

User response

Correct the referenced syntax error and resubmit.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0621E

EXCEPTION_CAUGHT=CWSPT0621E: Fatal error. Exception caught: {0}

Explanation

The referenced exception was caught.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0622E

FATAL_ERROR_ENCOUNTERED_RUNNING_PLACEMENT_SERVICE=CWSPT0622E: There is a fatal error encountered when running the placement service: {0}

Explanation

There is a fatal error encountered when running the placement service. Exception data is included .

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0623E

RELATION_NOT_EXISTEXCEPTION=CWSPT0623E: Fatal error: "The stream or view does not exist: {0}"

Explanation

Fatal error. Report the error to IBM.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0624W

CANNOT_FIND_FIELD_AT_INDEX=CWSPT0624W: Fatal error: "The field with index {0} cannot be found."

Explanation

Fatal error: Report the problem to IBM.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0626W

STATELESS_TRANSFORM_TIMEOUT_COUNTER_GREATER_THAN_ZERO=CWSPT0626W: Fatal error: "Stateless transform timeout counter: {0} > 0"

Explanation

Fatal error. Report the problem to IBM.

User response

If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0630W

CANNOT_FULLY_ERASE=CWSPT0630W: Cannot fully erase the directory {0} : {1}

Explanation

The compiler cannot delete its output directory.

User response

The directory is probably opened in another program. Make sure all other programs using the directory have exited. If the problem persists, contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

CWSPT0633I

GETHASHTABLESIGNATURESTRRAW_HASHTABLE_IS_NULL=CWSPT0633I: Fatal error:
"getHashTableSignatureStrRaw() hashTable is null"

Explanation

Compiler error.

User response

Contact your IBM Customer Service Center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0634W

CANNOT_CREATE_DIRECTORY_TRY_PROCEEDING_FORWARD=CWSPT0634W: Cannot create directory. Continuing.

Explanation

SMILE could not create a directory, but will still attempt to deploy.

User response

None.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0641I

PLACEMENTSERVICE_CREATED=CWSPT0641I: "PlacementService created"

Explanation

None.

User response

None.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0642E

PLACEMENT_INITCOMMUNICATIONS_ERROR=CWSPT0642E: Placement error: {0}

Explanation

Severe error: Placement cannot communicate with the SMILE Server

User response

If the problem persists, contact IBM Customer Service Center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0643E

LINK_PLACEMENT_FAILED=CWSPT0643E: Link placement failed.

Explanation

Link placement failed.

User response

None.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0645I

CURRENT_DIRECTORY=CWSPT0645I: The smile repository directory is {0}

Explanation

This message reports the folder where the SMILE repository is found.

User response

None.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0750E

FEWER_THAN_TWO_JOIN=CWSPT0750E: Severe Error: "Query rewriter generated a join with fewer than 2 operands".

Explanation

This is a compiler bug.

User response

This is a SMILE compiler error. Report query and message to IBM support.

Parent topic: [CWSPT](#)

CWSPT0751E

INCOMPATIBLE_MERGE=CWSPT0751E: Error: Merged relations have incompatible maps.

Explanation

The query rewriter detected that the two merged relations have incompatible schemas.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0752E

INCONSISTENT_COMBINE=CWSPT0752E: Severe Error: Query rewriter detected compiler error in COMBINE at MMPL line {0} position {1}.

Explanation

The compiler generated the message "The number of LHS expressions does not equal the number of RHS expressions".

User response

This is a SMILE compiler error. Report query and message to IBM support.

Parent topic: [CWSPT](#)

CWSPT0753E

MISMATCHED_RENAMES=CWSPT0753E: Severe Error: Query rewriter detected compile error in COMBINE at MMPL line {0} position {1}.

Explanation

The compiler generated the error message "The number of renames for LHS and RHS do not match."

User response

This is a SMILE compiler error. Report query and message to IBM support.

Parent topic: [CWSPT](#)

CWSPT0754E

TWO_MAPS_SHARE_KEY=CWSPT0754E: Severe Error: Query rewriter detected compile error in JOIN.

Explanation

Two maps share the same key dependency.

User response

This is a SMILE compiler error. Report to IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0755E

MISSING_ACTION=CWSPT0755E: Severe Error: "Query rewriter could not find action for column named {0}".

Explanation

This is a bug in the compiler.

User response

This is a SMILE compiler error. Report to IBM support.

Parent topic: [CWSPT](#)

CWSPT0756E

MULTIPLE_MAPS_CONTAIN_COLUMNS=CWSPT0756E: Severe Error: "Query rewriter found same set of columns in multiple maps."

Explanation

This is a SMILE compiler error

User response

This is a SMILE compiler error. Report to IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0757E

COMBINE_NON_INTEGER=CWSPT0757E: Columns {2} and {3} in view {4} cannot be combined into an integer type.

Explanation

The columns should be of integer or long type.

User response

Rewrite the query to use integer or long types.

Parent topic: [CWSPT](#)

CWSPT0758E

COMBINE_KEY_AND_NONKEY=CWSPT0758E: Columns {2} and {3} in view {4} cannot be combined.

Explanation

Columns being combined must be either both keys or both non-keys.

User response

Rewrite the query.

Parent topic: [CWSPT](#)

CWSPT0759E

COMBINE_GENERATES_DUPLICATE=CWSPT0759E: Operation {2} in relation {4} generates a duplicate column {3} near line {0} position {1}.

Explanation

There is another column in the result with the same name.

User response

Rewrite to use a unique name.

Parent topic: [CWSPT](#)

CWSPT0760E

DUPLICATE_JOIN_COLUMN=CWSPT0760E: Operation {2} in relation {4} has a column {3} appearing in both operands but not merged, near line {0} position {1}.

Explanation

This is illegal, since it would result in a view with two identically named columns.

User response

Rewrite either to merge the same-named columns, or to rename one of the columns.

Parent topic: [CWSPT](#)

CWSPT0761E

DUPLICATE_RELATION=CWSPT0761E: Statement redefines a view named {3} near line {0} position {1}.

Explanation

A view name is defined twice.

User response

Rewrite to use a unique name.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0762E

RENAME_GENERATES_DUPLICATE=CWSPT0762E: Renaming a column {3} in view {4} generates a duplicate name.

Explanation

The resulting view would have two columns with the same name.

User response

Rewrite to use a unique name.

Parent topic: [CWSPT](#)

CWSPT0763E

SPLIT_GENERATES_DUPLICATE=CWSPT0763E: Splitting a column generates a duplicate column name {3}.

Explanation

The resulting view would have two columns with the same name by splitting.

User response

Rewrite to make sure splitting results unique names.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0764E

MERGE_NON_TIMED=CWSPT0764E: Merge of non-time-keyed views.

Explanation

SMILE currently supports MERGE only for time-keyed views.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0765E

MERGE_INCOMPATIBLE_SCHEMAS=CWSPT0765E: Attempt to merge views with incompatible schemas.

Explanation

Views can only be merged if the schemas are identical.

User response

Rewrite query to make sure merge only applies to identical schemas.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0766E

SELF_MERGE_ERROR=CWSPT0766E: Attempt to merge view {2} with itself.

Explanation

You may not merge a view with itself.

User response

Do not merge view with itself; eliminate the merge operation.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0767E

NOT_SPLITTABLE_COLUMN=CWSPT0767E: Cannot split column {3} in view {2}.

Explanation

A column must be integer or long type to be split.

User response

Rewrite query to make sure either the split column is integer or long type, or do not use split

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0768E

MMPL_SYNTAX_EXCEPTION=CWSPT0768E: Syntax exception in MMPL. Unexpected token is {2}.

Explanation

The MMPL intermediate language generated from your query had a syntax error.

User response

Report to IBM support

Parent topic: [CWSPT](#)

CWSPT0769E

SQL_SYNTAX_EXCEPTION=CWSPT0769E: Syntax exception in SQL query. Unexpected token is {2}.

Explanation

Your SQL query had a syntax error.

User response

Rewrite your SQL query.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0770E

NO_TIME_COLUMN=CWSPT0770E: No time column generated for stream {2}.

Explanation

The generated MMPL did not specify a time column.

User response

Report to IBM support

Parent topic: [CWSPT](#)

CWSPT0771E

CANNOT_RESOLVE_COLUMN_NAME=CWSPT0771E: Reference to an undefined column {2} in view {3}.

Explanation

You are referring to a column that has not been defined.

User response

Either change the column definition or the column reference so that they match.

Parent topic: [CWSPT](#)

CWSPT0772E

CANNOT_RESOLVE_MERGED_COLUMN_NAME=CWSPT0772E: Reference to an undefined JOIN USING column {2} in view {3}

Explanation

You are referring to a JOIN USING column that does not exist.

User response

Check reference and rewrite query.

Parent topic: [CWSPT](#)

CWSPT0773E

UNDEFINED_VIEW_NAME_IN_SUBSCRIPTION=CWSPT0773E: An MMPL Subscription statement refers to an undefined view {2}.

Explanation

The view name is undefined.

User response

If you did not write this MMPL yourself, this is a SQL to MMPL translation error. Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0774E

UNDEFINED_REFERENCE_IN_ALIAS=CWSPT0774E: An MMPL Assignment statement refers to an undefined view {2}.

Explanation

The view named in the right side of the assignment statement is undefined.

User response

If you did not write this MMPL yourself, this is a SQL to MMPL translation error. Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0775E

UNDEFINED_PROJECTION_COLUMN=CWSPT0775E: A projection operation is referring to an undefined column {2}.

Explanation

The column name does not exist so it cannot be projected.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0776E

RENAMED_COLUMN_NOT_FOUND=CWSPT0776E: The column {2} being renamed in view {3} does not exist.

Explanation

The column name does not exist.

User response

Check that the defined name and the reference match.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0777E

WINDOW_SIZE_NOT_GE_0=CWSPT0777E: The window size in view {2} is not greater than 0.

Explanation

A negative or zero window size is illegal.

User response

Rewrite query.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0778E

UNKNOWN_EXPRESSION=CWSPT0778E: An unknown expression appeared in an MMPL WHERE or ADD clause at line {0} , position {1}.

Explanation

The expression cannot be recognized.

User response

If you did not write this MMPL yourself, this is a SQL to MMPL translation error. Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0779E

ILLEGAL_RANGE=CWSPT0779E: Illegal numeric range at line {0} , position {1}.

Explanation

The upper bound must be greater than or equal to the lower bound.

User response

Correct numeric ranges.

Parent topic: [CWSPT](#)

CWSPT0780E

NON_SCALAR_SELECT=CWSPT0780E: Expression in WHERE clause is not a scalar expression at line {0} , position {1}.

Explanation

The expression must evaluate to a scalar boolean value.

User response

Correct the expression in the WHERE clause to make sure it is scalar.

Parent topic: [CWSPT](#)

CWSPT0781E

NON_RELATIONAL_ALIAS=CWSPT0781E: An MMPL assignment statement has a right-hand side which is not a relational expression at line {0} , position {1}.

Explanation

The right hand side must resolve to a relational expression.

User response

If you did not write this MMPL yourself, this is a SQL to MMPL translation error. Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0782E

MISPLACED_AGGREGATION_EXPRESSION=CWSPT0782E: An MMPL aggregation expression appeared elsewhere than in a SELECT or EXTEND at line {0} , position {1}.

Explanation

Aggregation operators should appear only in SELECT or EXTEND.

User response

If you did not write this MMPL yourself, this is a SQL to MMPL translation error. Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0783E

UNRESOLVABLE_EXPRESSION=CWSPT0783E: The expression {2} cannot be resolved at line {0} , position {1}.

Explanation

It can neither be resolved to a view or to an operation on columns.

User response

Rewrite expression.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0784E

UNRESOLVABLE_ARITHMETIC_TERM=CWSPT0784E: The operand to the operator {2} cannot be resolved at line {0}, position{1}.

Explanation

An operand must be a constant, scalar expression, aggregate, or variable.

User response

Rewrite expression to make sure the operand is a constant, scalar expression, aggregate, or variable.

Parent topic: [CWSPT](#)

CWSPT0785E

ILLEGAL_PROJECTION=CWSPT0785E: Column {3} of view {2} depends on a column that is being projected away.

Explanation

A column may not be retained in a view if it is functionally dependent on a column that is not kept.

User response

Either drop both columns or retain the column that the given column depends upon.

Parent topic: [CWSPT](#)

CWSPT0786E

ILLEGAL_STRING_SIZE=CWSPT0786E: String size operator was applied to non-string.

Explanation

String size operator was applied to non-string.

User response

Probable compiler error. Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0787E

ILLEGAL_ENUMERATION=CWSPT0787E: ENUMERATION is not supported.

Explanation

ENUMERATION type not currently supported.

User response

Should not occur. Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0788E

CANNOT_BOOL_AGGREGATES=CWSPT0788E: Cannot apply boolean operations to aggregates.

Explanation

Aggregates can only be used for numbers, not booleans.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0789E

BAD_INPUTS_FOR_COMBINE=CWSPT0789E: Operation {7} in view {6} combines column {2} of type {4} with column {3} of type {5}.

Explanation

The operation requires that the inputs be integer or long types to be combined.

User response

Rewrite query to make sure the inputs to combine are integer or long types

Parent topic: [CWSPT](#)

CWSPT0790E

INCOMPATIBLE_TYPES_FOR_COMPARISON=CWSPT0790E: Expression in view {6} compares column {2} of type {4} with column {3} of type {5}.

Explanation

In order for two columns to be comparable, they must be two numerics, two booleans, or two strings.

User response

Rewrite query to make sure columns are comparable

Parent topic: [CWSPT](#)

CWSPT0791E

EXPECTED_NUMERIC_TYPE=CWSPT0791E: Column {2} of type {4} in view {6} was expected to be numeric.

Explanation

The type was something other than numeric.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0792E

EXPECTED_BOOLEAN_TYPE=CWSPT0792E: Column {2} of type {4} in view {6} was expected to be boolean.

Explanation

The type was something other than boolean.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0793E

ILLEGAL_LATEST=CWSPT0793E: Column {2} in view {3} does not depend upon a time key.

Explanation

The LATEST operator only applies to columns that depend upon a time key.

User response

Rewrite query.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0794E

ILLEGAL_MERGE_OPERANDS=CWSPT0794E: View {6} attempts to merge views that are not time-keyed.

Explanation

The MERGE operator only applies to time-keyed streams or views.

User response

Rewrite query.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0795E

ILLEGAL_WHERE_EXPRESSION=CWSPT0795E: View {6} contains WHERE clause of type {4}.

Explanation

A WHERE clause should be of type boolean.

User response

Rewrite query.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0796E

INCOMPATIBLE_JOIN_COLUMNS=CWSPT0796E: View {6} joins column {2} of type {4} with column {3} of type {5}.

Explanation

Joined columns must be of comparable type.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0797E

ILLEGAL_SPLIT_TYPE=CWSPT0797E: Column {2} of type {4} in view {6} is of illegal type for SPLIT.

Explanation

the column to split must be numeric.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0798E

ILLEGAL_WINDOW_OPERAND=CWSPT0798E: View {6} applies WINDOW to a view that is not time-keyed.

Explanation

The WINDOW operator only applies to time-keyed streams or views.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0799E

CANNOT_COMBINE_INCOMPATIBLE_UNITS=CWSPT0799E: View {6} attempts to combine column {2} with units {4} with column {3} with units {5}.

Explanation

The units are incompatible.

User response

Rewrite query.

Parent topic: [CWSPT](#)

CWSPT0800E

INCOMPATIBLE_UNITS=CWSPT0800E: View {6} uses incompatible units {4} and {5}.

Explanation

The units are incompatible.

User response

Rewrite query.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0801E

UNKNOWN_AGGREGATION_OPERATOR=CWSPT0801E: Unknown aggregation operator at line {0}, position {1}.

Explanation

The MMPL contains an unknown aggregation operator.

User response

This is probably a SQL to MMPL translation error. Report to IBM support.

Parent topic: [CWSPT](#)

CWSPT0802E

UNKNOWN_TYPE_FOR_AGGREGATION=CWSPT0802E: Unknown type for aggregation.

Explanation

An unknown type is being converted to an aggregate type.

User response

This is probably a compiler error. Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0803E

DIFFERENT_SIGNATURE_FOR_MERGE=CWSPT0803E: Views {2} and {3} have different types.

Explanation

Views being merged must have the same type.

User response

Rewrite query.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0804E

OPERATOR_NEEDS_TWO_INPUTS=CWSPT0804E: Operator needs at least two operands.

Explanation

Operator has too few operands.

User response

This is probably an SQL to MMPL translation error Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0805E

UNKNOWN_OPERATOR_TYPE=CWSPT0805E: Unknown operator type {0}.

Explanation

MMPL code has an operation that is not defined.

User response

This is probably an SQL to MMPL translation error Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0806E

NOT_SHOULD_HAVE_1_OPERAND=CWSPT0806E: NOT operator should have one operand.

Explanation

MMPL generate a NOT operator with wrong number of operands.

User response

This is probably an SQL to MMPL translation error Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0807E

UNSUPPORTED_JOIN_RULE=CWSPT0807E: Severe error: "Query rewriter failed to decompose a JOIN:
Environment = "{2}":."

Explanation

This is probably a compiler error.

User response

Take the query and message and report to IBM support team.

Parent topic: [CWSPT](#)

CWSPT0808E

SPLIT_COLUMN_SMALLER_THAN_DIVISOR=CWSPT0808E: SPLIT operation has a divisor larger than the range of the column being split.

Explanation

SPLIT operation has a divisor larger than the range of the column being split, so the quotient will always be 0.

User response

Rewrite the query without SPLIT, or use a different divisor.

Parent topic: [CWSPT](#)

CWSPT0809E

SPLIT_COLUMN_HAS_ZERO_RANGE=CWSPT0809E: SPLIT column has zero range.

Explanation

The SPLIT cannot be performed.

User response

Rewrite the query.

Parent topic: [CWSPT](#)

CWSPT0810E

CONSTANTS_WITH_NONNUMERICS=CWSPT0810E: Constant relation has non-numeric columns.

Explanation

An MMPL constant relation with non-numeric columns was entered.

User response

This is probably an SQL to MMPL translation error Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0811E

CONSTANTS_WITH_NONKEYS=CWSPT0811E: Constant relation should not have non-key columns

Explanation

An MMPL constant relation with non-key columns was entered.

User response

This is probably an SQL to MMPL translation error Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0812E

COLUMN_LACKS_TYPE=CWSPT0812E: Compiler error: "Column {2} in view {3} lacks a type."

Explanation

This is probably a compiler error.

User response

Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0813E

INVALID_MERGE=CWSPT0813E: Compiler error: "Invalid Merge: Merge has more than one visibility column."

Explanation

This is probably a compiler error.

User response

Probable compiler error. Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0814E

TYPE_COMPARE_FAILED=CWSPT0814E: Compiler error: "Type compare failed: exception data: {2}."

Explanation

This is probably a compiler error

User response

Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0815E

UNITS_FOR_NONNUMERICS=CWSPT0815E: compiler error: "Attempt to extract units for non-numeric fields."

Explanation

This is probably a compiler error.

User response

Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0816E

INVALID_TRANSFORM=CWSPT0816E: compiler error: "An invalid transform {2} was attempted."

Explanation

This should not occur.

User response

This is probably a compiler error. Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0817E

UNIT_NOT_TIME=CWSPT0817E: Compiler error: "A unit {2} is expected to be a time unit and is not."

Explanation

Should not occur in SQL. Compiler error.

User response

Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0818E

INVALID_OP_ON_UNIT=CWSPT0818E: Compiler error: "An invalid operation was attempted on unit {2}."

Explanation

Should not occur in SQL. Compiler error.

User response

Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0819E

INVALID_UNIT=CWSPT0819E: Compiler error: "An invalid unit {2} was detected."

Explanation

Should not occur in SQL. Compiler error.

User response

Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0820E

UNKNOWN_BASIC_TYPE=CWSPT0820E: Compiler error: "An invalid MMPL type {2} was detected."

Explanation

Should not occur in SQL. Compiler error.

User response

Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0821E

NO_RULE_FOR_CLASS=CWSPT0821E: Compiler error: "The transform with class {2} has no rule."

Explanation

This is probably a compiler error.

User response

Report to IBM support center.

Parent topic: [CWSPT](#)

CWSPT0822E

UNKNOWN_EXPRESSION_INPUT=CWSPT0822E: Probable compiler error: "The expression has an unknown form."

Explanation

This is probably a compiler error.

User response

Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0823E

NO_TYPES_IN_ADD=CWSPT0823E: Compiler error: "An EXTEND has no types in its ADD list."

Explanation

The compiler generated malformed MMPL.

User response

This is probably an SQL to MMPL translation error Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0824E

UNSUPPORTED_TYPE=CWSPT0824E: Compiler error: "Type {2} is unsupported in MMPL."

Explanation

The compiler generated malformed MMPL.

User response

This is probably an SQL to MMPL translation error Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0825E

NO_TYPE_FOR_SUBSCRIPTION=CWSPT0825E: Subscription {2} could not be assigned a type.

Explanation

Probable MMPL error.

User response

This is probably an SQL to MMPL translation error. Report to IBM support center.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0826E

RENAME_ENTRY_GENERATES_DUPLICATE=CWSPT0826E: Trying to rename a query from {0} to {1} caused a duplication.

Explanation

The repository may not have two queries with the same view name.

User response

Choose different names for different queries.

Parent topic: [CWSPT](#)

CWSPT0827E

CANNOT_REMOVE_BECAUSE_USED_BY=CWSPT0827E: Cannot remove view {0} because it is used by view(s) {1}.

Explanation

If a view refers to a second view, you may not remove the second view without removing the first.

User response

You must either not remove the view, or else remove the dependent views in the same transaction.

Parent topic: [CWSPT](#)

CWSPT0828E

CYCLIC_DEPENDENCY=CWSPT0828E: The update would create a cyclic dependency of views

Explanation

A first view cannot depend on a second view that directly or indirectly depends on the first view.

User response

Rewrite the queries to eliminate the cyclic dependency.

Parent topic: [CWSPT](#)

CWSPT0829E

ADDITION_CREATES_CONFLICT=CWSPT0829E: The view {0} already exists. Trying to add it again fails.

Explanation

The repository may not have two queries with the same name.

User response

Choose a different name.

Parent topic: [CWSPT](#)

CWSPT0830E

MULTIPLE_VISIBILITY_COLUMNS_APPLY=CWSPT0830E: Severe compiler error: "Multiple visibility columns apply. Invariant violated in {0}".

Explanation

This is a compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0831E

CANNOT_COMPILE_GENERATED_JAVA_CODE=CWSPT0831E: Cannot compile generated Java code. See Log.

Explanation

This is probably a compiler error.

User response

Please look at the log. or take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0832E

COULD_NOT_RUN_PROCESS=CWSPT0832E: Probable compiler error: "Could not run process: {0} ({1})."

Explanation

This is probably a compiler error.

User response

Check your Java environment.

Parent topic: [CWSPT](#)

CWSPT0833E

NON_KEY_COLUMN_MUST_BE_OF_EVOLVABLE_TYPE=CWSPT0833E: Severe compiler error: "Non-key column {0} must be of evolvable type."

Explanation

This is probably a compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0834E

KEY_COLUMN_MUST_BE_OF_NON_EVOLVABLE_TYPE=CWSPT0834E: Compiler error: "Key column: {0} must be of non-evolvable type."

Explanation

This is probably a compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0835E

PARAMETER_NULL_IS_INVALID=CWSPT0835E: Compiler error: "Parameter null is invalid."

Explanation

This is a compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0836E

TYPE_NAME_CANNOT_BE_NULL=CWSPT0836E: Compiler error: "Type name cannot be null."

Explanation

This is a compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0837E

MORE_THAN_TWO_MESSAGE_JOIN_NOT_ALLOWED=CWSPT0837E: There must be two message streams to join: {0}.

Explanation

Joining more than two message streams is not allowed at this point.

User response

Change your query to have a join with only two message streams.

Parent topic: [CWSPT](#)

CWSPT0838E

INTERNAL_COMPILER_ERROR_FOR_LOGGING=CWSPT0838E: Compiler Error: "In {0} with a message : {1}."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0839E

INTERNAL_COMPILER_ERROR=CWSPT0839E: Compiler Error: "In {0}."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0840E

CAN_NOT_FIND_PARENT_NODE=CWSPT0840E: Compiler error: "Cannot find parent node in {0}."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0843E

ADS_EXCEPTION_1=CWSPT0843E: Compiler error: "Multiple messages apply to the same equi-column command."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0844E

RUNTIME_EXCEPTION=CWSPT0844E: Compiler error: "Method not implemented."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0845E

ASSERTION_ERROR=CWSPT0845E: Compiler error: "Simple join has more than one joined result:{0}."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0846E

ADS_EXCEPTION_2=CWSPT0846E: Compiler error: "Incompatible inputs for merge in ADS stage. Some checks failed earlier in line {0} and position {1}."

Explanation

Compiler error: Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0847E

ILLEGAL_STATE_EXCEPTION_1=CWSPT0847E: Compiler error: "This case should be handled in rewriter: {0}."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0848E

ADS_EXCEPTION_3=CWSPT0848E: Compiler error: "Message is effected by multiple split operations. Invalid state in line {0} and posistion {1}."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0849E

ADS_EXCEPTION_4=CWSPT0849E: Severe compiler error: "Aggregation should have been factored stand-alone expression in line {0} and position {1}".

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0850E

ADS_EXCEPTION_5=CWSPT0850E: Compiler error: "Duplicate message found in line {0} and position {1}."

Explanation

Compiler error: Duplicate message found.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0851E

ADS_EXCEPTION_6=CWSPT0851E: Compiler error: "Message not found with columns: {0} in line {1} and position {2}."

Explanation

Compiler error: Message not found.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0852E

CODEGEN_EXCEPTION=CWSPT0852E: Compiler error: "Cannot find next node process:in line {0} and position {1}."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0853E

CODEGEN_EXCEPTION_1=CWSPT0853E: Compiler error: "Unknown type".

Explanation

The compiler generated the error message "Unknown type".

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0854E

CODEGEN_EXCEPTION_2=CWSPT0854E: Compiler Error: "Cannot compute index for column {0} in Hash Table {1}."

Explanation

Compiler error

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0855E

CODEGEN_EXCEPTION_3=CWSPT0855E: Compiler error: "Unknown column {0} and type {1}."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0856E

CODEGEN_EXCEPTION_4=CWSPT0856E: Compiler error: "Unsupported equality statement at line {0} and position {1}."

Explanation

Compiler error: Unsupported equality statement.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0857E

CODEGEN_EXCEPTION_5=CWSPT0857E: Compiler error: "Should have been factored out by splitter component in line: {0} and position: {1}."

Explanation

This is an internal compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0858E

CODEGEN_EXCEPTION_6=CWSPT0858E: Compiler error: "Not supported: {0} in {1}".

Explanation

This is an internal compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0859E

CODEGEN_EXCEPTION_7=CWSPT0859E: Compiler error: "The operator {0} is not supported on String."

Explanation

Compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0860E

CODEGEN_EXCEPTION_8=CWSPT0860E: Compiler error: "Unknown case".

Explanation

This is an internal compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0861E

CODEGEN_EXCEPTION_9=CWSPT0861E: Compiler error: "Storing silence is not supported for client hashes for now."

Explanation

This is an internal compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0862E

CODEGEN_EXCEPTION_10=CWSPT0862E: Compiler error: "Compiler generated code is not designed to get data from client table."

Explanation

This is an internal compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0863E

ADS_EXCEPTION_7=CWSPT0863E: Compiler error: "node: {0} should have had broker assignment in line {1} and position {2}."

Explanation

Compiler error: node should have had broker assignment.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

CWSPT0864E

JOIN_FAILED=CWSPT0864E: Compiler error: "Join failed: {2}".

Explanation

This is probably a compiler error.

User response

Take the query and message and report to IBM support team.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWSPT0865E

CANNOT_FIND_TOOLJAR_EXCEPTION=CWSPT0865E: Cannot find class com.sun.tools.javac.Main. The tools.jar is probably not in the class path.

Explanation

The class com.sun.tools.javac.Main is not found. The tools.jar is probably not in the class path.

User response

Add the tools.jar in the classpath.

Parent topic: [CWSPT](#)

CWSPT0867E

ILLEGAL_STATE_EXCEPTION=CWSPT0867E: Compiler error: "There is a name conflict."

Explanation

This is a compiler error.

User response

Take the query and message and contact IBM support.

Parent topic: [CWSPT](#)

[Terms of Use](#) | [Feedback](#)

CWWSM

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWWSM0001E](#)

session.parseXML=CWWSM0001E: An exception occurred when parsing the web.xml : {0}

[CWWSM0002E](#)

session.lineXML=CWWSM0002E: An exception occurred at line {0} when parsing the web.xml.

[CWWSM0003E](#)

session.columnXML=CWWSM0003E: An exception occurred at column {0} when parsing the web.xml.

[CWWSM0004E](#)

session.DGException=CWWSM0004E: Caught a datagrid exception {0} when performing a Datagrid operation.

[CWWSM0005E](#)

session.throwable=CWWSM0005E: Caught a runtime exception {0}.

[CWWSM0006E](#)

session.affinityManager=CWWSM0006E: Caught exception when instantiating the affinity manager: {0}.

[CWWSM0007I](#)

session.objectgrid=CWWSM0007I: Using the ObjectGrid based Session Manager.

[CWWSM0008I](#)

session.webapp=CWWSM0008I: Web application {0} contains declaration for HttpSessionFilter.

[CWWSM0009I](#)

session.filter.override=CWWSM0009I: Session Override did not contain any catalog end points. Checking servlet context.

[CWWSM0010I](#)

session.filter.conn.failed=CWWSM0010I: Connection attempt to ObjectGrid failed: Exception is {0}.

[CWWSM0020E](#)

ERR_MISSING_KEY=CWWSM0020E: Message key {0} was not found in any searched resource bundles.

CWWSM0001E

session.parseXML=CWWSM0001E: An exception occurred when parsing the web.xml : {0}

Explanation

A parse exception occurred while analyzing the web.xml.

User response

Check the integrity of the web.xml as well as its contents for conformance with the web.xml schema.

Parent topic: [CWWSM](#)

[Terms of Use](#) | [Feedback](#)

CWWSM0002E

session.lineXML=CWWSM0002E: An exception occurred at line {0} when parsing the web.xml.

Explanation

A parse exception occurred while analyzing the web.xml.

User response

Check the integrity of the web.xml as well as its contents for conformance with the web.xml schema.

Parent topic: [CWWSM](#)

[Terms of Use](#) | [Feedback](#)

CWWSM0003E

session.columnXML=CWWSM0003E: An exception occurred at column {0} when parsing the web.xml.

Explanation

A parse exception occurred while analyzing the web.xml.

User response

Check the integrity of the web.xml as well as its contents for conformance with the web.xml schema.

Parent topic: [CWWSM](#)

[Terms of Use](#) | [Feedback](#)

CWWSM0004E

session.DGException=CWWSM0004E: Caught a datagrid exception {0} when performing a Datagrid operation.

Explanation

A datagrid exception was caught when attempting to perform a datagrid operation.

User response

The exception itself may contain additional details of the problem causing the exception.

Parent topic: [CWWSM](#)

[Terms of Use](#) | [Feedback](#)

CWWSM0005E

session.throwable=CWWSM0005E: Caught a runtime exception {0}.

Explanation

none.

User response

none.

Parent topic: [CWWSM](#)

[Terms of Use](#) | [Feedback](#)

CWWSM0006E

session.affinityManager=CWWSM0006E: Caught exception when instantiating the affinity manager: {0}.

Explanation

none.

User response

none.

Parent topic: [CWWSM](#)

[Terms of Use](#) | [Feedback](#)

CWWSM0007I

session.objectgrid=CWWSM0007I: Using the ObjectGrid based Session Manager.

Explanation

none.

User response

none.

Parent topic: [CWWSM](#)

[Terms of Use](#) | [Feedback](#)

CWWSM0008I

session.webapp=CWWSM0008I: Web application {0} contains declaration for HttpSessionFilter.

Explanation

The user application was found to have been instrumented with a declaration for a HttpSessionFilter.

User response

none.

Parent topic: [CWWSM](#)

CWWSM0009I

session.filter.override=CWWSM0009I: Session Override did not contain any catalog end points. Checking servlet context.

Explanation

none.

User response

none.

Parent topic: [CWWSM](#)

[Terms of Use](#) | [Feedback](#)

CWWSM0010I

session.filter.conn.failed=CWWSM0010I: Connection attempt to ObjectGrid failed: Exception is {0}.

Explanation

none.

User response

none.

Parent topic: [CWWSM](#)

[Terms of Use](#) | [Feedback](#)

CWWSM0020E

ERR_MISSING_KEY=CWWSM0020E: Message key {0} was not found in any searched resource bundles.

Explanation

A key was passed into the Messages class to resolve to a string, but a properties file for the locale could not be found and the appropriate message could not be retrieved.

User response

Determine why the message catalog could not be found.

Parent topic: [CWWSM](#)

CWXCS

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

CWXCS0001I

css.servant.initialized=CWXCS0001I: Servant {0} has initialized successfully.

CWXCS0002I

css.servant.terminated=CWXCS0002I: Servant {0} has terminated.

CWXCS0003I

css.service.initialized=CWXCS0003I: CSS service {0} initialized successfully.

CWXCS0004I

css.service.started=CWXCS0004I: CSS service {0} started successfully.

CWXCS0200E

css.unexpected.exception=CWXCS0200E: Method {0} caught exception: {1}

CWXCS0201E

css.unexpected.remote.exception=CWXCS0201E: Method {0} caught remote exception: {1}

CWXCS0202E

css.factory.initialization.failure=CWXCS0202E: Method {0} was unable to initialize factory {1}

CWXCS0001I

css.servant.initialized=CWXCS0001I: Servant {0} has initialized successfully.

Explanation

The designated servant has successfully completed initialization.

User response

None

Parent topic: [CWXCS](#)

[Terms of Use](#) | [Feedback](#)

CWXCS0002I

css.servant.terminated=CWXCS0002I: Servant {0} has terminated.

Explanation

The designated servant has terminated.

User response

None

Parent topic: [CWXCS](#)

[Terms of Use](#) | [Feedback](#)

CWXCS0003I

css.service.initialized=CWXCS0003I: CSS service {0} initialized successfully.

Explanation

The CSS service has successfully completed initialization.

User response

None

Parent topic: [CWXCS](#)

[Terms of Use](#) | [Feedback](#)

CWXCS0004I

css.service.started=CWXCS0004I: CSS service {0} started successfully.

Explanation

The CSS service has successfully started.

User response

None

Parent topic: [CWXCS](#)

[Terms of Use](#) | [Feedback](#)

CWXCS0200E

css.unexpected.exception=CWXCS0200E: Method {0} caught exception: {1}

Explanation

An exception was caught in the specified method.

User response

None

Parent topic: [CWXCS](#)

[Terms of Use](#) | [Feedback](#)

CWXCS0201E

css.unexpected.remote.exception=CWXCS0201E: Method {0} caught remote exception: {1}

Explanation

A remote exception was caught in the specified method.

User response

None

Parent topic: [CWXCS](#)

[Terms of Use](#) | [Feedback](#)

CWXCS0202E

css.factory.initialization.failure=CWXCS0202E: Method {0} was unable to initialize factory {1}

Explanation

The specified factory could not be initialized.

User response

None

Parent topic: [CWXCS](#)

[Terms of Use](#) | [Feedback](#)

CWXDA

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWXDA0001I](#)

xda.service.initialized=CWXDA0001I: XDA service {0} initialized successfully.

[CWXDA0002I](#)

xda.service.started=CWXDA0002I: XDA service {0} started successfully.

[CWXDA0003I](#)

xda.service.federate=CWXDA0003I: Agent with node name {0} is being federated to host {1}.

[CWXDA0004I](#)

xda.service.nodeexists=CWXDA0004I: Agent with node name {0} already exists in the cell.

[CWXDA0005I](#)

xda.service.nodeupdated=CWXDA0005I: Agent node name is updated to {0} as node name with same name detected.

[CWXDA0006I](#)

xda.service.federatesuccess=CWXDA0006I: Agent {0} is successfully federated into the cell.

[CWXDA0007E](#)

xda.service.federatecontacterror=CWXDA0007E: Unable to contact {0} to federate.

[CWXDA0008E](#)

xda.service.federateerror=CWXDA0008E: Error happened while federating, check logs/addAgent.log file for errors.

[CWXDA0012I](#)

xda.service.disabled=CWXDA0012I: XDA service {0} is disabled.

[CWXDA0013I](#)

xda.service.stopped=CWXDA0013I: XDA service {0} is stopped.

[CWXDA0014I](#)

xda.service.changedefault=CWXDA0014I: Changing agent name to local host name from Default.

[CWXDA0015I](#)

xda.service.federatestart=CWXDA0015I: Federating the agent, Tool information is logged in file {0}.

[CWXDA0016I](#)

xda.service.federateusage=CWXDA0016I: Usage: addAgent -host <dmgr_hostname> -port <adminconsole_port> [[-username <username>] [-password <password>]] [-nodename <nodename>] [-noadmin] [-trace [<trace_spec>]] [-help]

[CWXDA0017E](#)

xda.service.unabletowrite=CWXDA0017E: Unable to write to the log file {0}.

[CWXDA0018I](#)

xda.service.actionslogged=CWXDA0018I: Tool actions are being logged in file {0}.

[CWXDA0019E](#)

xda.service.alreadyfederated=CWXDA0019E: Node is already federate into {0}, cannot be federated again.

[CWXDA0020E](#)

xda.service.discoveryprefererror=CWXDA0020E: Discovery service {0} threw BackingStoreException {1}, so ignore processing of earlier discovered elements.

[CWXDA0021I](#)

xda.agent.stopping=CWXDA0021I: Stopping the agent, check the log file {0} for log messages.

[CWXDA0022E](#)

xda.agent.errstopping=CWXDA0022E: Error Stopping the agent, check the log file {0} for stop errors.

[CWXDA0023I](#)

xda.agent.stopsuccess=CWXDA0023I: Stop command is issued successfully, check the log file {0} for messages.

[CWXDA0024E](#)

xda.service.filemissing=CWXDA0024E: File {0} missing to proceed further.

[CWXDA0025E](#)

xda.service.erroraddingnode=CWXDA0025E: Error while adding node {0} to the cell.

CWXDA0026E

xda.service.notfederated=CWXDA0026E: Node is not federated into any cell.

CWXDA0027E

xda.service.unfederateerror=CWXDA0027E: Error while removing a node from cell.

CWXDA0028E

xda.service.mastercontacterror=CWXDA0028E: Error contacting master repository.

CWXDA0029E

xda.service.unfederatestart=CWXDA0029E: Removing agent from the master repository.

CWXDA0030I

xda.service.removeagentsuccess=CWXDA0030I: Removed agent successfully from the cell.

CWXDA0031E

xda.security.configreaderror=CWXDA0031E: Unable to find security configuration.

CWXDA0032I

xda.nodesync.changedetectedinrepo=CWXDA0032I: Newer version of repository {0} detected on master node.

CWXDA0033I

xda.agent.launch=CWXDA0033I: Launching the agent, check the log file {0} for launch errors.

CWXDA0034I

xda.agent.launchsuccess=CWXDA0034I: Agent launched successfully. Check logs for messages.

CWXDA0035W

xda.agent.launchwitherr=CWXDA0035W: Agent launched with errors. Check logs for errors.

CWXDA0036I

xda.node.portchange=CWXDA0036I: Port {0} is updated to {1} for the node {2}.

CWXDA0037E

xda.agent.agentalreadyrunning=CWXDA0037E: An instance of the agent may be already running.

CWXDA0038E

xda.service.nouserspecified=CWXDA0038E: Username not specified. Requires -username <username> when security is enabled on deployment manager.

CWXDA0039E

xda.service.nopasswdspecified=CWXDA0039E: Password not specified. Requires -password <password> when security is enabled on Deployment Manager.

CWXDA0040E

xda.agent.agentalreadystopped=CWXDA0040E: The agent cannot be reached. It appears to be stopped.

CWXDA0041I

xda.channel.securemode=CWXDA0041I: XDA channels will be started in ssl mode

CWXDA0042E

xda.service.invalidcreds=CWXDA0042E: Unable to federate using the username and the password provided.

CWXDA0043I

xda.federate.rollback=CWXDA0043I: Rolling back the federation due to error during federation.

CWXDA0044E

xda.federate.errorrollback=CWXDA0044E: Error during rolling back federate operations.

CWXDA0045I

xda.agent.startedpid=CWXDA0045I: Agent started successfully; process id is {2}

CWXDA0046I

xda.readports.localconfig=CWXDA0046I: Missing host and port information. Reading the information from local configuration

CWXDA0047I

xda.service.startMWserverusage=CWXDA0047I: Usage: startMWServer -server <server_name> [-trace [<trace_spec>]] [-help]

CWXDA0048I

xda.service.stopMWserverusage=CWXDA0048I: Usage: stopMWServer -server <server_name> [-trace [<trace_spec>]] [-help]

CWXDA0049I

xda.service.mwserverstatususage=CWXDA0049I: Usage: mwServerStatus -server <server_name> [-trace [<trace_spec>]] [-help]

CWXDA0200E

xda.unexpected.exception=CWXDA0200E: Method {0} caught exception: {1}

CWXDA0201E

xda.agent.exception=CWXDA0201E: Error happened:{0}

CWXDA1001I

xda.hacontroller.initialized=CWXDA1001I: XDA HAController on Agent {0} initialized successfully.

CWXDA1002I

xda.hacontroller.started=CWXDA1002I: XDA HAController on Agent {0} started successfully.

CWXDA1003I

xda.hacontroller.already.started=CWXDA1003I: XDA HAController on Agent is already started.

CWXDA1004I

xda.hacontroller.websphere.environment=CWXDA1004I: XDA HAController detects that this agent is in the websphere environment and imports websphere HA properties.

CWXDA1010I

xda.hacontroller.not.initialized=CWXDA1010I: XDA HAController on Agent is not initialized successfully.

CWXDA1022E

xda.hacontroller.failed.start=CWXDA1022E: XDA HAController on Agent failed to start.

CWXDA1023E

xda.hacontroller.not.start=CWXDA1023E: XDA HAController on Agent is not started.

CWXDA2000I

xda.hacontroller.add.new.member=CWXDA2000I: XDA HAController on Agent adds new member {0}.

CWXDA2001I

xda.catalogserver.initialized=CWXDA2001I: XDA CatalogServer cluster on Agent {0} initialized successfully.

CWXDA2002I

xda.catalogserver.started=CWXDA2002I: XDA CatalogServer Cluster on Agent {0} started successfully.

CWXDA6000I

xda.catalognode.started=CWXDA6000I: XD Agent {0} is a CatalogServer node.

CWXDA6001I

xda.nocatalognode.started=CWXDA6001I: XD Agent {0} is not a CatalogServer node.

CWXDA8000I

xda.xdagent.domain=CWXDA8000I: This XD Agent is associated with the domain of {0}

CWXDA9000I

xda.xdagent.process.started=CWXDA9000I: Server {0} type {1} on node {2} cell {3} was successfully started.

CWXDA9001I

xda.xdagent.process.alreadystarted=CWXDA9001I: Server {0} type {1} on node {2} cell {3} is already running.

CWXDA9002I

xda.xdagent.process.stopped=CWXDA9002I: Server {0} type {1} on node {2} cell {3} was successfully stopped.

CWXDA9003I

xda.xdagent.process.alreadystopped=CWXDA9003I: Server {0} type {1} on node {2} cell {3} is not running.

CWXDA9004I

xda.xdagent.process.terminated=CWXDA9004I: Server {0} type {1} on node {2} cell {3} was successfully terminated.

CWXDA9005I

xda.xdagent.process.ended=CWXDA9005I: Agent has detected that Server {0} type {1} on node {2} cell {3} has terminated.

CWXDA9006I

xda.xdagent.process.attach.started.server=CWXDA9006I: Agent found server {0} type {1} on node {2} cell {3} started and has reattached successfully.

CWXDA9007I

xda.xdagent.process.attach.not.started.server=CWXDA9007I: Agent found server {0} type {1} on node {2} cell {3} not started and has abandoned an attempt to reattach.

CWXDA9008I

xda.xdagent.process.attach.failed=CWXDA9008I: Agent attempt to reattach to server {0} type {1} on

node {2} cell {3} failed with return code {4}.

CWXDA9011I

xda.xdagent.process.running=CWXDA9011I: Server {0} type {1} on node {2} cell {3} is running.

CWXDA9012I

xda.xdagent.process.notrunning=CWXDA9012I: Server {0} type {1} on node {2} cell {3} is not running.

CWXDA9013E

xda.xdagent.process.actionerror=CWXDA9013E: Error invoking action on Server {0} type {1} on node {2} cell {3}.

CWXDA9014I

xda.agent.process.issuecmd=CWXDA9014I: Invoking action {0} on Server {1}. Check {2} file for log messages.

CWXDA9015E

xda.agent.process.servernotfound=CWXDA9015E: Server {0} not found to invoke the operation.

CWXDA9016E

xda.process.stopagenttoremove=CWXDA9016E: Active Agent cannot be removed from the cell. Stop the agent first.

CWXDA9017I

xda.xdagent.process.startedpid=CWXDA9017I: Server {0} type {1} on node {2} cell {3} was successfully started; process id is {4}.

CWXDA9018I

xda.xdagent.process.pid=CWXDA9018I: Server {0} process id is {1}.

CWXDA9019I

xda.xdagent.processattach.pid=CWXDA9019I: Found process id {0} for the server {1} from an earlier start; Attempt will be made to attach.

CWXDA9020I

xda.xdagent.processattach.info=CWXDA9020I: Unable to attach to the process {0}.

CWXDA9021E

xda.xdagent.processlaunch.error=CWXDA9021E: Error launching server {0}. Check server logs for error messages.

CWXDA9022I

xda.service.syncstart=CWXDA9022I: Synchronizing configuration of the node with the master repository, Tool information is logged in file {0}.

CWXDA9023I

xda.service.syncusage=CWXDA9023I: Usage: syncAgent [[-host <dmgr_hostname>] [-port <adminconsole_port>]] [[-username <username>] [-password <password>]] [-trace [<trace_spec>]] [-help]

CWXDA9024E

xda.service.syncerror=CWXDA9024E: Error happened while synchronizing with master repository, check logs/syncAgent.log file for errors.

CWXDA9025I

xda.service.syncsuccess=CWXDA9025I: Agent successfully synchronized configuration with the cell manager.

CWXDA9070I

xda.xdagent.discovery.service.opchange=CWXDA9070I: Operation was changed from {0} to {1} for object {3}

CWXDA9071I

xda.xdagent.discovery.service.newchanges=CWXDA9071I: Agent for node {0} is sending new config information to the deployment manager at node {1}

CWXDA9072I

xda.xdagent.discovery.service.discoveredobject=CWXDA9072I: Remote agent has detected new {0} with name {1}

CWXDA0001I

xda.service.initialized=CWXDA0001I: XDA service {0} initialized successfully.

Explanation

The XD Agent service has successfully completed initialization.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0002I

xda.service.started=CWXDA0002I: XDA service {0} started successfully.

Explanation

The XD Agent service has successfully started.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0003I

xda.service.federate=CWXDA0003I: Agent with node name {0} is being federated to host {1}.

Explanation

The Agent is being federated to deployment manager.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0004I

xda.service.nodeexists=CWXDA0004I: Agent with node name {0} already exists in the cell.

Explanation

The Agent with same node name already exists in the cell.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0005I

xda.service.nodeupdated=CWXDA0005I: Agent node name is updated to {0} as node name with same name detected.

Explanation

The Agent name is changed to new name to avoid conflict with already existing nodes.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0006I

xda.service.federatesuccess=CWXDA0006I: Agent {0} is successfully federated into the cell.

Explanation

The Agent is successfully federated into the cell.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0007E

xda.service.federatecontacterror=CWXDA0007E: Unable to contact {0} to federate.

Explanation

Unable to contact the server to federate. Server may be down.

User response

Make sure deployment manager is running before running addAgent step.

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0008E

xda.service.federateerror=CWXDA0008E: Error happened while federating, check logs/addAgent.log file for errors.

Explanation

Error happened while federating the agent into deployment manager.

User response

Check the addAgent.log file in logs folder for errors.

Parent topic: [CWXDA](#)

CWXDA0012I

xda.service.disabled=CWXDA0012I: XDA service {0} is disabled.

Explanation

The XD Agent service is disabled.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0013I

xda.service.stopped=CWXDA0013I: XDA service {0} is stopped.

Explanation

The XD Agent service is stopped.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0014I

xda.service.changedefault=CWXDA0014I: Changing agent name to local host name from Default.

Explanation

Current name of the agent is Default, it will be changed to host name.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0015I

xda.service.federatestart=CWXDA0015I: Federating the agent, Tool information is logged in file {0}.

Explanation

Agent is federated into the cell. Check the logs for errors.

User response

Check the logs for errors and take appropriate action.

Parent topic: [CWXDA](#)

CWXDA0016I

xda.service.federateusage=CWXDA0016I: Usage: addAgent -host <dmgr_hostname> -port <adminconsole_port> [[-username <username>] [-password <password>]] [-nodename <nodename>] [-noadmin] [-trace [<trace_spec>]] [-help]

Explanation

addAgent script was run with wrong arguments.

User response

Specify correct arguments.

Parent topic: [CWXDA](#)

CWXDA0017E

xda.service.unabletowrite=CWXDA0017E: Unable to write to the log file {0}.

Explanation

Error while writing to the log file.

User response

Make sure log file location is writable.

Parent topic: [CWXDA](#)

CWXDA0018I

xda.service.actionslogged=CWXDA0018I: Tool actions are being logged in file {0}.

Explanation

Trace information is logged to the specified file.

User response

Check the logs for informational messages and errors.

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0019E

xda.service.alreadyfederated=CWXDA0019E: Node is already federate into {0}, cannot be federated again.

Explanation

Current node is already federated into cell, so it cannot be federate again.

User response

Run removeNode first.

Parent topic: [CWXDA](#)

CWXDA0020E

xda.service.discoverypreferror=CWXDA0020E: Discovery service {0} threw BackingStoreException {1}, so ignore processing of earlier discovered elements.

Explanation

Discovery service throws BackingStoreException to indicate there has been some error doing discovery, so retry again.

User response

None

Parent topic: [CWXDA](#)

CWXDA0021I

xda.agent.stopping=CWXDA0021I: Stopping the agent, check the log file {0} for log messages.

Explanation

Agent is being stopped. Check the logs for errors.

User response

Check the logs for errors and take appropriate action.

Parent topic: [CWXDA](#)

CWXDA0022E

xda.agent.errstopping=CWXDA0022E: Error Stopping the agent, check the log file {0} for stop errors.

Explanation

Error while stopping the agent. Check the logs for errors.

User response

Check the logs for errors and take appropriate action.

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0023I

xda.agent.stopsuccess=CWXDA0023I: Stop command is issued successfully, check the log file {0} for messages.

Explanation

Stop command is issued successfully. Check the logs for errors.

User response

Check the logs for errors and take appropriate action.

Parent topic: [CWXDA](#)

CWXDA0024E

xda.service.filemissing=CWXDA0024E: File {0} missing to proceed further.

Explanation

Unable to locate the file that is required.

User response

Contact IBM support.

Parent topic: [CWXDA](#)

CWXDA0025E

xda.service.erroraddingnode=CWXDA0025E: Error while adding node {0} to the cell.

Explanation

Error while adding node to the cell.

User response

Check the logs for errors.

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0026E

xda.service.notfederated=CWXDA0026E: Node is not federated into any cell.

Explanation

Node is not part of any cell, so removeNode will not have any affect.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0027E

xda.service.unfederateerror=CWXDA0027E: Error while removing a node from cell.

Explanation

Error happened while removing node from cell, check logs for error.

User response

Check the logs for errors.

Parent topic: [CWXDA](#)

CWXDA0028E

xda.service.mastercontacterror=CWXDA0028E: Error contacting master repository.

Explanation

Error happened while contacting master repository. Check logs for errors.

User response

Check the logs for errors.

Parent topic: [CWXDA](#)

CWXDA0029E

xda.service.unfederatestart=CWXDA0029E: Removing agent from the master repository.

Explanation

Removing federated agent from the master repository.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0030I

xda.service.removeagentsuccess=CWXDA0030I: Removed agent successfully from the cell.

Explanation

Removed the agent successfully from the cell.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0031E

xda.security.configreaderror=CWXDA0031E: Unable to find security configuration.

Explanation

Security configuration is missing on the agent.

User response

Check if security xml exists on the agent.

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0032I

xda.nodesync.changedetectedinrepo=CWXDA0032I: Newer version of repository {0} detected on master node.

Explanation

Newer version of the repository is found on the master node.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0033I

xda.agent.launch=CWXDA0033I: Launching the agent, check the log file {0} for launch errors.

Explanation

Agent is launched. Check log file for any errors.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0034I

xda.agent.launchsuccess=CWXDA0034I: Agent launched successfully. Check logs for messages.

Explanation

Agent launched successfully. Check logs for useful messages.

User response

None

Parent topic: [CWXDA](#)

CWXDA0035W

xda.agent.launchwitherr=CWXDA0035W: Agent launched with errors. Check logs for errors.

Explanation

Agent launched with errors. Check log files for more information.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0036I

xda.node.portchange=CWXDA0036I: Port {0} is updated to {1} for the node {2}.

Explanation

Port is updated to handle port conflicts of agent.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0037E

xda.agent.agentalreadyrunning=CWXDA0037E: An instance of the agent may be already running.

Explanation

An instance of the agent may be already running on the port.

User response

Stop the running agent to issue start again.

Parent topic: [CWXDA](#)

CWXDA0038E

xda.service.nouserspecified=CWXDA0038E: Username not specified. Requires -username <username> when security is enabled on deployment manager.

Explanation

Security is enabled on deployment manager, requires username to successfully federate/unfederate/sync the agent.

User response

Specify -username <username> to the addAgent/removeAgent/syncAgent script.

Parent topic: [CWXDA](#)

CWXDA0039E

xda.service.nopasswdspecified=CWXDA0039E: Password not specified. Requires -password <password> when security is enabled on Deployment Manager.

Explanation

Security is enabled on deployment manager, requires password to successfully federate/unfederate/sync the agent.

User response

Specify -password <password> to the addAgent/removeAgent/syncAgent script.

Parent topic: [CWXDA](#)

CWXDA0040E

xda.agent.agentalreadystopped=CWXDA0040E: The agent cannot be reached. It appears to be stopped.

Explanation

An instance of the agent may not be running on the port.

User response

Start the agent to issue stop again.

Parent topic: [CWXDA](#)

CWXDA0041I

xda.channel.securemode=CWXDA0041I: XDA channels will be started in ssl mode

Explanation

Security is enabled, so channels will be started in ssl mode.

User response

None.

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0042E

xda.service.invalidcreds=CWXDA0042E: Unable to federate using the username and the password provided.

Explanation

Security is enabled, so channels will be started in ssl mode.

User response

None.

Parent topic: [CWXDA](#)

CWXDA0043I

xda.federate.rollback=CWXDA0043I: Rolling back the federation due to error during federation.

Explanation

Error happened during federation, so rolling back the federation process.

User response

Check the logs for error messages.

Parent topic: [CWXDA](#)

CWXDA0044E

xda.federate.errorrollback=CWXDA0044E: Error during rolling back federate operations.

Explanation

Error during roll back of federation steps.

User response

Check the logs for error messages.

Parent topic: [CWXDA](#)

CWXDA0045I

xda.agent.startedpid=CWXDA0045I: Agent started successfully; process id is {2}

Explanation

The XD Agent service has successfully started.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0046I

xda.readports.localconfig=CWXDA0046I: Missing host and port information. Reading the information from local configuration

Explanation

Host and port information not specified to the command. Information will be read from local configuration repository.

User response

None.

Parent topic: [CWXDA](#)

CWXDA0047I

xda.service.startMWserverusage=CWXDA0047I: Usage: startMWServer -server <server_name> [-trace
[<trace_spec>]] [-help]

Explanation

startMWServer script was run with wrong arguments.

User response

Specify correct arguments.

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0048I

xda.service.stopMWserverusage=CWXDA0048I: Usage: stopMWServer -server <server_name> [-trace
[<trace_spec>]] [-help]

Explanation

stopMWServer script was run with wrong arguments.

User response

Specify correct arguments.

Parent topic: [CWXDA](#)

CWXDA0049I

xda.service.mwserverstatususage=CWXDA0049I: Usage: mwServerStatus -server <server_name> [-trace
[<trace_spec>]] [-help]

Explanation

mwServerStatus script was run with wrong arguments.

User response

Specify correct arguments.

Parent topic: [CWXDA](#)

CWXDA0200E

xda.unexpected.exception=CWXDA0200E: Method {0} caught exception: {1}

Explanation

An exception was caught in the specified method.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA0201E

xda.agent.exception=CWXDA0201E: Error happened:{0}

Explanation

Unexpected Error happened.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA1001I

xda.hacontroller.initialized=CWXDA1001I: XDA HAController on Agent {0} initialized successfully.

Explanation

The XD Agent HAController has successfully completed initialization.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA1002I

xda.hacontroller.started=CWXDA1002I: XDA HAController on Agent {0} started successfully.

Explanation

The XD Agent HAController has successfully started.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA1003I

xda.hacontroller.already.started=CWXDA1003I: XDA HAController on Agent is already started.

Explanation

The XD Agent HAController has been started before.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA1004I

xda.hacontroller.websphere.environment=CWXDA1004I: XDA HAController detects that this agent is in the websphere environment and imports websphere HA properties.

Explanation

XDA HAController detects that this agent is in the websphere environment and imports websphere HA properties.

User response

None

Parent topic: [CWXDA](#)

CWXDA1010I

xda.hacontroller.not.initialized=CWXDA1010I: XDA HAController on Agent is not initialized successfully.

Explanation

The XD Agent HAController cannot start standalone HAManager because it is not initialized.

User response

turn on trace, contact IBM support

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA1022E

xda.hacontroller.failed.start=CWXDA1022E: XDA HAController on Agent failed to start.

Explanation

The XD Agent HAController failed to start standalone HAManager.

User response

Turn on trace, contact IBM support

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA1023E

xda.hacontroller.not.start=CWXDA1023E: XDA HAController on Agent is not started.

Explanation

The XD Agent HAController is not started..

User response

Turn on trace, contact IBM support

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA2000I

xda.hacontroller.add.new.member=CWXDA2000I: XDA HAController on Agent adds new member {0}.

Explanation

The XD Agent HAController adds new member.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA2001I

xda.catalogserver.initialized=CWXDA2001I: XDA CatalogServer cluster on Agent {0} initialized successfully.

Explanation

The XD Agent CatalogServer Cluster has successfully completed initialization.

User response

None

Parent topic: [CWXDA](#)

CWXDA2002I

xda.catalogserver.started=CWXDA2002I: XDA CatalogServer Cluster on Agent {0} started successfully.

Explanation

The XD Agent CatalogServer cluster has successfully started.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA6000I

xda.catalognode.started=CWXDA6000I: XD Agent {0} is a CatalogServer node.

Explanation

This node manages CatalogServer cluster

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA6001I

xda.nocatalognode.started=CWXDA6001I: XD Agent {0} is not a CatalogServer node.

Explanation

This node does not manage CatalogServer cluster

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA8000I

xda.xdagent.domain=CWXDA8000I: This XD Agent is associated with the domain of {0}

Explanation

The XD Agent HAController has successfully started.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9000I

xda.xdagent.process.started=CWXDA9000I: Server {0} type {1} on node {2} cell {3} was successfully started.

Explanation

The specified server was successfully started.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9001I

xda.xdagent.process.alreadystarted=CWXDA9001I: Server {0} type {1} on node {2} cell {3} is already running.

Explanation

The specified server is already running.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9002I

xda.xdagent.process.stopped=CWXDA9002I: Server {0} type {1} on node {2} cell {3} was successfully stopped.

Explanation

The specified server was successfully stopped.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9003I

xda.xdagent.process.alreadystopped=CWXDA9003I: Server {0} type {1} on node {2} cell {3} is not running.

Explanation

The specified server is not running.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9004I

xda.xdagent.process.terminated=CWXDA9004I: Server {0} type {1} on node {2} cell {3} was successfully terminated.

Explanation

The specified server was successfully terminated.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9005I

xda.xdagent.process.ended=CWXDA9005I: Agent has detected that Server {0} type {1} on node {2} cell {3} has terminated.

Explanation

The specified server was ended.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9006I

xda.xdagent.process.attach.started.server=CWXDA9006I: Agent found server {0} type {1} on node {2} cell {3} started and has reattached successfully.

Explanation

The agent reattached to the specified server successfully.

User response

None

Parent topic: [CWXDA](#)

CWXDA9007I

xda.xdagent.process.attach.not.started.server=CWXDA9007I: Agent found server {0} type {1} on node {2} cell {3} not started and has abandoned an attempt to reattach.

Explanation

When attempting to reattach, the agent found to the specified server was not started.

User response

None

Parent topic: [CWXDA](#)

CWXDA9008I

xda.xdagent.process.attach.failed=CWXDA9008I: Agent attempt to reattach to server {0} type {1} on node {2} cell {3} failed with return code {4}.

Explanation

The agent's attempt to reattach to the specified server failed.

User response

None

Parent topic: [CWXDA](#)

CWXDA9011I

xda.xdagent.process.running=CWXDA9011I: Server {0} type {1} on node {2} cell {3} is running.

Explanation

The specified server is running.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9012I

xda.xdagent.process.notrunning=CWXDA9012I: Server {0} type {1} on node {2} cell {3} is not running.

Explanation

The specified server is not running.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9013E

xda.xdagent.process.actionerror=CWXDA9013E: Error invoking action on Server {0} type {1} on node {2} cell {3}.

Explanation

Error while trying to invoke action on the server.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9014I

xda.agent.process.issuecmd=CWXDA9014I: Invoking action {0} on Server {1}. Check {2} file for log messages.

Explanation

Error while trying to invoke action on the server.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9015E

xda.agent.process.servernotfound=CWXDA9015E: Server {0} not found to invoke the operation.

Explanation

Server not found to invoke the operation.

User response

Make sure server exists before issuing a command.

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9016E

xda.process.stopagenttoremove=CWXDA9016E: Active Agent cannot be removed from the cell. Stop the agent first.

Explanation

Tried to remove agent while agent is still running.

User response

Stop the agent to remove it from the cell.

Parent topic: [CWXDA](#)

CWXDA9017I

xda.xdagent.process.startedpid=CWXDA9017I: Server {0} type {1} on node {2} cell {3} was successfully started; process id is {4}.

Explanation

The specified server was successfully started.

User response

None

Parent topic: [CWXDA](#)

CWXDA9018I

xda.xdagent.process.pid=CWXDA9018I: Server {0} process id is {1}.

Explanation

Server process id is displayed.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9019I

xda.xdagent.processattach.pid=CWXDA9019I: Found process id {0} for the server {1} from an earlier start; Attempt will be made to attach.

Explanation

Attach to a process using the process id from an earlier start

User response

None

Parent topic: [CWXDA](#)

CWXDA9020I

xda.xdagent.processattach.info=CWXDA9020I: Unable to attach to the process {0}.

Explanation

Unable to attach to the process

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9021E

xda.xdagent.processlaunch.error=CWXDA9021E: Error launching server {0}. Check server logs for error messages.

Explanation

Error happened launching the server. Check logs for error messages.

User response

Check the server logs and take necessary action to correct the problem.

Parent topic: [CWXDA](#)

CWXDA9022I

xda.service.syncstart=CWXDA9022I: Synchronizing configuration of the node with the master repository, Tool information is logged in file {0}.

Explanation

Node is being synchronized with the master repository. Check the logs for errors.

User response

Check the logs for errors and take appropriate action.

Parent topic: [CWXDA](#)

CWXDA9023I

xda.service.syncusage=CWXDA9023I: Usage: syncAgent [[-host <dmgr_hostname>] [-port <adminconsole_port>]] [[-username <username>] [-password <password>]] [-trace [<trace_spec>]] [-help]

Explanation

syncAgent script was run with wrong arguments.

User response

Specify correct arguments.

Parent topic: [CWXDA](#)

CWXDA9024E

xda.service.syncerror=CWXDA9024E: Error happened while synchronizing with master repository, check logs/syncAgent.log file for errors.

Explanation

Error happened while synchronizing configuration of the agent with deployment manager.

User response

Check the syncAgent.log file in logs folder for errors.

Parent topic: [CWXDA](#)

CWXDA9025I

xda.service.syncsuccess=CWXDA9025I: Agent successfully synchronized configuration with the cell manager.

Explanation

The Agent successfully synchronized configuration with the cell manager.

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9070I

xda.xdagent.discoveryservice.opchange=CWXDA9070I: Operation was changed from {0} to {1} for object {3}

Explanation

The deployment manager logic decided that a different operation is required

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9071I

xda.xdagent.discovery.service.newchanges=CWXDA9071I: Agent for node {0} is sending new config information to the deployment manager at node {1}

Explanation

Remote agent discovered new config information

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDA9072I

xda.xdagent.discovery.service.discoveredobject=CWXDA9072I: Remote agent has detected new {0} with name {1}

Explanation

Remote agent discovered new foreign element

User response

None

Parent topic: [CWXDA](#)

[Terms of Use](#) | [Feedback](#)

CWXDC

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

CWXDC0401E

CWXDC0401E=CWXDC0401E: Cannot find dynamic cluster {0} in the configuration repository.

CWXDC0402E

CWXDC0402E=CWXDC0402E: Cannot find cluster {0} in the configuration repository.

CWXDC0403E

CWXDC0403E=CWXDC0403E: A dynamic cluster with the name {0} already exists in the configuration repository.

CWXDC0404E

CWXDC0404E=CWXDC0404E: A cluster with the name {0} already exists in the configuration repository.

CWXDC0401E

CWXDC0401E=CWXDC0401E: Cannot find dynamic cluster {0} in the configuration repository.

Explanation

The specified dynamic cluster cannot be found in the configuration repository.

User response

Determine if the specified dynamic cluster exists, and if the correct dynamic cluster name is specified. Specify a valid dynamic cluster name and try the operation again.

Parent topic: [CWXDC](#)

CWXDC0402E

CWXDC0402E=CWXDC0402E: Cannot find cluster {0} in the configuration repository.

Explanation

The specified cluster cannot be found in the configuration repository.

User response

Determine if the specified cluster exists, and if the correct cluster name is specified. Specify a valid cluster name and try the operation again.

Parent topic: [CWXDC](#)

[Terms of Use](#) | [Feedback](#)

CWXDC0403E

CWXDC0403E=CWXDC0403E: A dynamic cluster with the name {0} already exists in the configuration repository.

Explanation

The specified dynamic cluster already exists in the configuration repository.

User response

Determine if the dynamic cluster name already exists in the configuration repository. Specify a different dynamic cluster name and try the operation again.

Parent topic: [CWXDC](#)

CWXDC0404E

CWXDC0404E=CWXDC0404E: A cluster with the name {0} already exists in the configuration repository.

Explanation

The specified cluster already exists in the configuration repository.

User response

Determine if the cluster name already exists in the configuration repository. Specify a different cluster name and try the operation again.

Parent topic: [CWXDC](#)

[Terms of Use](#) | [Feedback](#)

CWXDG

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWXDG1000E](#)

GS_CONFIG_ACCESS_FAILED=CWXDG1000E: Unable to access job scheduler configuration. Exception = {0}.

[CWXDG1010E](#)

HA_SENSOR_START_FAILED=CWXDG1010E: Attempt to start the grid endpoint sensor component on {0} has failed. Exception = {1}.

[CWXDG1030I](#)

HA_SENSOR_ACTIVATED=CWXDG1030I: The grid endpoint sensor component has successfully been activated.

[CWXDG1040I](#)

HA_SENSOR_DEACTIVATED=CWXDG1040I: The grid endpoint sensor component has successfully been deactivated.

[CWXDG1050E](#)

ERROR_PROCESSING_JOBS_FROM_ENDPOINT=CWXDG1050E: Unexpected errors encountered while processing job data received from endpoints.

[CWXDG1060E](#)

ERROR_REGISTERING_SERVERSTATUSLISTENER=CWXDG1060E: Attempt to register a server status listener has failed. Exception = {0}.

[CWXDG1070E](#)

ERROR_REGISTERING_NODESTATUSLISTENER=CWXDG1070E: Attempt to register a node status listener has failed. Exception = {0}.

[CWXDG1080E](#)

GRIDSCHEDULER_MBEAN_UNAVAILABLE=CWXDG1080E: The job scheduler MBean is currently unavailable. Exception = {0}.

[CWXDG1090E](#)

GRIDSCHEDULER_MBEAN_OP_ERROR=CWXDG1090E: Attempt to call job scheduler MBean operation {0} has failed. Exception = {0}.

[CWXDG1100E](#)

ERROR_UPDATING_USAGE_DATA=CWXDG1100E: Attempt to update usage accounting data has failed. Exception = {0}.

[CWXDG1110E](#)

UNEXPECTED_ERROR=CWXDG1110E: Unexpected runtime errors encountered. Exception = {0}.

[CWXDG5001I](#)

CWXDG_I_InitComplete=CWXDG5001I: WebSphere Compute Grid native batch container initialization complete.

[CWXDG5002W](#)

CWXDG_W_EXCEPTION=CWXDG5002W: Non-fatal exception occurred: {0}

[CWXDG5003E](#)

CWXDG_E_EXCEPTION=CWXDG5003E: Fatal exception occurred: {0}

[CWXDG5004E](#)

CWXDG_E_PROCESS_EXCEPTION=CWXDG5004E: Error creating native process: {0}. Job step {1} of job {2} did not run.

[CWXDG5005I](#)

CWXDG5005I=CWXDG5005I: [{0}] Setting up Grid Utility job {1} for execution: {2} {3} {4} {5}

[CWXDG5006I](#)

CWXDG5006I=CWXDG5006I: [{0}] Grid Utility Job {1} ended.

[CWXDG5007I](#)

CWXDG5007I=CWXDG5007I: [{0}] Cancel request received for Grid Utility Job {1}.

[CWXDG5008I](#)

CWXDG5008I=CWXDG5008I: [{0}] Beginning Grid Utility Job {1} execution.

[CWXDG5009I](#)

CWXDG5009I=CWXDG5009I: [{0}] Beginning step {1} execution.

CWXDG5010I

CWXDG5010I=CWXDG5010I: [{0}] Grid Utility Job step {1} ended: cancelled.

CWXDG5011I

CWXDG5011I=CWXDG5011I: [{0}] Grid Utility Job step {1} ended.

CWXDG5012I

CWXDG5012I=CWXDG5012I: [{0}] Grid Utility Job {1} ended: cancelled.

CWXDG5013I

CWXDG5013I=CWXDG5013I: [{0}] Grid Utility Job {1} ended.

CWXDG5014E

CWXDG5014E=CWXDG5014E: [{0}] Unable to read Grid Utility Job {1} {2} file {3}: {4}.

CWXDG5015E

CWXDG5015E=CWXDG5015E: [{0}] Unable to close Grid Utility Job {1} {2} file {3}: {4}.

CWXDG5016E

CWXDG5016E=CWXDG5016E: [{0}] Unable to delete Grid Utility Job {1} {2} file {3}: {4}.

CWXDG5017I

CWXDG5017I=CWXDG5017I: Cannot cancel Grid Utility Job {0}: Job not found.

CWXDG5018I

CWXDG5018I=CWXDG5018I: [{0}] Cancelling Grid Utility Job {1}.

CWXDG5019I

CWXDG5019I=CWXDG5019I: [{0}] {1}

CWXDG5020E

CWXDG5020E=CWXDG5020E: Null jobid passed to {0}.

CWXDG5021I

CWXDG5021I=CWXDG5021I: [{0}] Grid Utility Job [{1}] job log output is quiesced: output limit is exceeded.

CWXDG5022E

CWXDG5022E=CWXDG5022E: Null timestamp passed to {0}.

CWXDG5023E

CWXDG5023E=CWXDG5023E: Compute Grid Native Container method {0} attempted to cache a job log information object, but the object already exists.

CWXDG5024E

CWXDG5024E=CWXDG5024E: Compute Grid Native Container failed while creating the job log directory {0}: the exception {1} was caught when java.io.File.getCanonicalPath() was applied against abstract path {2}.

CWXDG5025E

CWXDG5025E=CWXDG5025E: Compute Grid Native Container failed while creating the job log directory {0}: {0} but is a regular file.

CWXDG5026E

CWXDG5026E=CWXDG5026E: Compute Grid Native Container failed while creating the job log directory {0}: java.io.File.mkdirs() returned false.

CWXDG5027E

CWXDG5027E=CWXDG5027E: Compute Grid Native Container failed while creating the job log directory {0}: {1}

CWXDG5028E

CWXDG5028E=CWXDG5028E: Compute Grid Native Container failed while creating the job log directory {0}: the exception {1} was caught when java.io.FilePermission() of "read,write" was applied to the directory.

CWXDG5029E

CWXDG5029E=CWXDG5029E: Compute Grid Native Container failed while creating the job log directory {0}: a file with the name {0} exists.

CWXDG5030E

CWXDG5030E=CWXDG5030E: Compute Grid Native Container failed while validating the job log directory {0}: the directory {0} is not writable.

CWXDG5031E

CWXDG5031E=CWXDG5031E: Compute Grid Native Container failed while validating the job log directory {0}: the directory {0} is not readable.

CWXDG5032E

CWXDG5032E=CWXDG5032E: Compute Grid Native Container failed while creating the job log directory {0}: the exception {1} was caught.

CWXDG5033E

CWXDG5033E=CWXDG5033E: Null jobclass passed to {0}.

CWXDG5034E

CWXDG5034E=CWXDG5034E: Illegal parameter(s) passed to {0}.

CWXDG5035E

CWXDG5035E=CWXDG5035E: Compute Grid Native Container File Logger {0} failed: directory {1} does not exist.

CWXDG5036E

CWXDG5036E=CWXDG5036E: Compute Grid Native Container File Logger {0} failed: file {1} does not exist.

CWXDG5037E

CWXDG5037E=CWXDG5037E: Compute Grid Native Container File Logger {0} failed: caught exception {1} while reading file {2}.

CWXDG5038E

CWXDG5038E=CWXDG5038E: Compute Grid Native Container File Logger is unable to capture System Stream: {0} exists but is a regular file.

CWXDG5039E

CWXDG5039E=CWXDG5039E: Compute Grid Native Container File Logger is unable to capture System Stream: unable to create directory {0}.

CWXDG5040E

CWXDG5040E=CWXDG5040E: Compute Grid Native Container File Logger is unable to capture System Stream: unable to create regular file {0}: {1}

CWXDG5041E

CWXDG5041E=CWXDG5041E: Compute Grid Native Container File Logger is unable to capture System Stream {0}: {1}

CWXDG5042E

CWXDG5042E=CWXDG5042E: Compute Grid Native Container File Logger write {0} failed: {1}

CWXDG5043E

CWXDG5043E=CWXDG5043E: Compute Grid Native Container failed while writing to the job log file {0}: the exception {1} was caught.

CWXDG5044E

CWXDG5044E=CWXDG5044E: Compute Grid Native Container failed while attempting to set file permissions for the job log file {0}: the exception {1} was caught when java.io.FilePermission() of "read,write" was applied to the directory.

CWXDG5045E

CWXDG5045E=CWXDG5045E: Compute Grid Native Container failed while attempting to write to the job log file {0}: the file {0} is not writable.

CWXDG5046E

CWXDG5046E=CWXDG5046E: Compute Grid Native Container failed while attempting to read the job log file {0}: the file {0} is not readable.

CWXDG5047E

CWXDG5047E=CWXDG5047E: Compute Grid Native Container failed while removing the job log file {0}: java.io.File.delete() returned false.

CWXDG5048E

CWXDG5048E=CWXDG5048E: Compute Grid Native Container failed while removing the job log file {0}: the exception {1} was caught.

CWXDG5049E

CWXDG5049E=CWXDG5049E: Compute Grid Native Container failed while removing the job log file {0}: {0} is not a regular file, it is a directory.

CWXDG5050E

CWXDG5050E=CWXDG5050E: Compute Grid Native Container failed while flushing the job log file stream {0}: the exception {1} was caught.

CWXDG5051E

CWXDG5051E=CWXDG5051E: Compute Grid Native Container failed while flushing the job log file writer {0}: the exception {1} was caught.

CWXDG5052E

CWXDG5052E=CWXDG5052E: Compute Grid Native Container failed while writing to the job log file {0}: the exception {1} was caught.

CWXDG5053E

CWXDG5053E=CWXDG5053E: Compute Grid Native Container failed while writing to the job log class file {0}: the exception {1} was caught.

CWXDG5054E

CWXDG5054E=CWXDG5054E: Compute Grid Native Container failed while writing to the job log class file {0}: java.io.File.createFile() returned false.

CWXDG5055E

CWXDG5055E=CWXDG5055E: Compute Grid Native Container failed while writing to the job log class file {0}: the exception {1} was caught.

CWXDG5056E

CWXDG5056E=CWXDG5056E: Null xJCL passed to {0}.

CWXDG5057E

CWXDG5057E=CWXDG5057E: Null jobClass passed to {0}.

CWXDG5058E

CWXDG5058E=CWXDG5058E: Null log directory passed to {0}.

CWXDG5059E

CWXDG5059E=CWXDG5059E: Compute Grid Native Container is unable to update the job log class list file {0}.

CWXDG5060E

CWXDG5060E=CWXDG5060E: Compute Grid Native Container failed while removing the job class list lock file {0}: java.io.File.delete() returned false.

CWXDG5061E

CWXDG5061E=CWXDG5061E: Compute Grid Native Container failed while removing the job class list lock file {0}: the exception {1} was caught.

CWXDG5062I

CWXDG5062I=CWXDG5062I: Compute Grid Native Container job {0} log file(s) are purged: {1}.

CWXDG5063I

CWXDG5063I=CWXDG5063I: Compute Grid Native Container job {0} log file(s) are purged: maximum job log age limit exceeded.

CWXDG5064I

CWXDG5064I=CWXDG5064I: Compute Grid Native Container job {0} log file(s) are purged: maximum file system log size limit is exceeded.

CWXDG5065I

CWXDG5065I=CWXDG5065I: Compute Grid Native Container job {0} log file(s) are purged: purged by user request.

CWXDG5066E

CWXDG5066E=CWXDG5066E: Compute Grid Native Container failed while writing to the job log file {0}: the exception {1} was caught.

CWXDG5067I

CWXDG5067I=CWXDG5067I: [{0}] Applying [property {1}] with [value {2}] obtained from {3} to the job xJCL.

CWXDG5068W

CWXDG5068W=CWXDG5068W: [{0}] System property {1} not found.

CWXDG5069I

CWXDG5069I=CWXDG5069I: [{0}] Job xJCL after all symbolic variable substitution(s) performed: {1}

CWXDG8250I

CWXDG_I_XDGridConfig_SchedConfigSuccess=CWXDG8250I: Job scheduler configuration completed successfully.

CWXDG8251E

CWXDG_I_XDGridConfig_SchedConfigFailed=CWXDG8251E: Job scheduler configuration failed.

CWXDG8252I

CWXDG_I_XDGridConfig_EndpointConfigSuccess=CWXDG8252I: Grid endpoint configuration completed successfully.

CWXDG8253E

CWXDG_I_XDGridConfig_EndpointConfigFailed=CWXDG8253E: Grid endpoint configuration failed.

CWXDG8254E

CWXDG_I_XDGridConfig_SchedException=CWXDG8254E: Job scheduler configuration failed due to error {0}.

CWXDG8255E

CWXDG_I_XDGridConfig_EndpointException=CWXDG8255E: Grid endpoint configuration failed due to error {0}.

CWXDG8256E

CWXDG_E_XDGridConfig_ConfigOption=CWXDG8256E: Configuration option {0} not supported.

[Terms of Use](#) | [Feedback](#)

CWXDG1000E

GS_CONFIG_ACCESS_FAILED=CWXDG1000E: Unable to access job scheduler configuration. Exception = {0}.

Explanation

The job scheduler may not have been configured.

User response

Verify that the job scheduler has been configured successfully.

Parent topic: [CWXDG](#)

CWXDG1010E

HA_SENSOR_START_FAILED=CWXDG1010E: Attempt to start the grid endpoint sensor component on {0} has failed. Exception = {1}.

Explanation

Internal error.

User response

Contact IBM Support.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG1030I

HA_SENSOR_ACTIVATED=CWXDG1030I: The grid endpoint sensor component has successfully been activated.

Explanation

The grid endpoint sensor high availability component has successfully been activated.

User response

None

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG1040I

HA_SENSOR_DEACTIVATED=CWXDG1040I: The grid endpoint sensor component has successfully been deactivated.

Explanation

The grid endpoint sensor high availability component has successfully been deactivated.

User response

None

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG1050E

ERROR_PROCESSING_JOBS_FROM_ENDPOINT=CWXDG1050E: Unexpected errors encountered while processing job data received from endpoints.

Explanation

Internal error.

User response

Contact IBM Support.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG1060E

ERROR_REGISTERING_SERVERSTATUSLISTENER=CWXDG1060E: Attempt to register a server status listener has failed. Exception = {0}.

Explanation

This is an internal problem.

User response

Contact IBM Support.

Parent topic: [CWXDG](#)

CWXDG1070E

ERROR_REGISTERING_NODESTATUSLISTENER=CWXDG1070E: Attempt to register a node status listener has failed. Exception = {0}.

Explanation

This is an internal problem.

User response

Contact IBM Support.

Parent topic: [CWXDG](#)

CWXDG1080E

GRIDSCHEDULER_MBEAN_UNAVAILABLE=CWXDG1080E: The job scheduler MBean is currently unavailable.
Exception = {0}.

Explanation

Errors detected during MBean lookup.

User response

Verify that the job scheduler is running.

Parent topic: [CWXDG](#)

CWXDG1090E

GRIDSCHEDULER_MBEAN_OP_ERROR=CWXDG1090E: Attempt to call job scheduler MBean operation {0} has failed. Exception = {0}.

Explanation

Errors detected during the specified MBean operation.

User response

Verify that the job scheduler is running.

Parent topic: [CWXDG](#)

CWXDG1100E

ERROR_UPDATING_USAGE_DATA=CWXDG1100E: Attempt to update usage accounting data has failed.
Exception = {0}.

Explanation

Internal error.

User response

Contact IBM Support.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG1110E

UNEXPECTED_ERROR=CWXDG1110E: Unexpected runtime errors encountered. Exception = {0}.

Explanation

Internal error.

User response

Contact IBM Support.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5001I

CWXDG_I_InitComplete=CWXDG5001I: WebSphere Compute Grid native batch container initialization complete.

Explanation

Indicates normal completion of container initialization.

User response

No user action required.

Parent topic: [CWXDG](#)

CWXDG5002W

CWXDG_W_EXCEPTION=CWXDG5002W: Non-fatal exception occurred: {0}

Explanation

Indicates an unexpected non-fatal exception occurred during container processing. The container is able to continue normal function.

User response

Optionally report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5003E

CWXDG_E_EXCEPTION=CWXDG5003E: Fatal exception occurred: {0}

Explanation

Indicates an unexpected fatal exception occurred during container processing. The container is not able to continue normal function.

User response

Report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5004E

CWXDG_E_PROCESS_EXCEPTION=CWXDG5004E: Error creating native process: {0}. Job step {1} of job {2} did not run.

Explanation

Indicates an unexpected error occurred while creating native process in which to run the indicated job step of the indicated job.

User response

This is most likely a configuration or job specification error. Based on the specified error, consider the native environment setup and ensure necessary conditions are met, such as PATH, LIBPATH, or CLASSPATH settings; that there is sufficient memory to create the process; and that the specified program is actually available on the machine. This is not the exhaustive list of possible causes for error, but should provide starting point to enable problem resolution.

Parent topic: [CWXDG](#)

CWXDG5005I

CWXDG5005I=CWXDG5005I: [{0}] Setting up Grid Utility job {1} for execution: {2} {3} {4} {5}

Explanation

The Grid Utility job named {1} is being setup for execution at time {0}. {2} {3} {4} {5} are properties of the job.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5006I

CWXDG5006I=CWXDG5006I: [{0}] Grid Utility Job {1} ended.

Explanation

The Grid Utility Job named {1} ended at time {0}.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5007I

CWXDG5007I=CWXDG5007I: [{0}] Cancel request received for Grid Utility Job {1}.

Explanation

A cancel request was received for the Grid Utility Job named {1} at time {0}.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5008I

CWXDG5008I=CWXDG5008I: [{0}] Beginning Grid Utility Job {1} execution.

Explanation

The Grid Utility Job named {1} is beginning execution at time {0}.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5009I

CWXDG5009I=CWXDG5009I: [{0}] Beginning step {1} execution.

Explanation

The Grid Utility Job step named {1} is beginning execution at time {0}.

User response

None, information message only.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5010I

CWXDG5010I=CWXDG5010I: [{0}] Grid Utility Job step {1} ended: cancelled.

Explanation

The Grid Utility Job step named {1} ended at time {0}. It was cancelled.

User response

None, information message only.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5011I

CWXDG5011I=CWXDG5011I: [{0}] Grid Utility Job step {1} ended.

Explanation

The Grid Utility Job step named {1} ended at time {0}.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5012I

CWXDG5012I=CWXDG5012I: [{0}] Grid Utility Job {1} ended: cancelled.

Explanation

The Grid Utility Job named {1} ended at time {0}. It was cancelled.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5013I

CWXDG5013I=CWXDG5013I: [{0}] Grid Utility Job {1} ended.

Explanation

The job named {1} ended at time {0}.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5014E

CWXDG5014E=CWXDG5014E: [{0}] Unable to read Grid Utility Job {1} {2} file {3}: {4}.

Explanation

The Compute Grid Native Container was unable to read a file for the Grid Utility Job named {1} at time {0}. The file is of type {2}, which may be STDOUT or STDERR. The file is named {3}. The exception {4} was caught.

User response

Varies by reported cause {4}.

Parent topic: [CWXDG](#)

CWXDG5015E

CWXDG5015E=CWXDG5015E: [{0}] Unable to close Grid Utility Job {1} {2} file {3}: {4}.

Explanation

The Compute Grid Native Container was unable to close a file for the Grid Utility Job named {1} at time {0}. The file is of type {2}, which may be STDOUT or STDERR. The file is named {3}. The exception {4} was caught.

User response

Varies by reported cause {4}.

Parent topic: [CWXDG](#)

CWXDG5016E

CWXDG5016E=CWXDG5016E: [{0}] Unable to delete Grid Utility Job {1} {2} file {3}: {4}.

Explanation

The Compute Grid Native Container was unable to delete a file for the Grid Utility Job named {1} at time {0}. The file is of type {2}, which may be STDOUT or STDERR. The file is named {3}. The exception {4} was caught.

User response

Varies by reported cause {4}.

Parent topic: [CWXDG](#)

CWXDG5017I

CWXDG5017I=CWXDG5017I: Cannot cancel Grid Utility Job {0}: Job not found.

Explanation

The Compute Grid Native Container received a request to cancel the Grid Utility Job named {0}. The Grid Utility Job named {0} could not be cancelled. It was not found.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5018I

CWXDG5018I=CWXDG5018I: [{0}] Cancelling Grid Utility Job {1}.

Explanation

The Compute Grid Native Container is cancelling the Grid Utility Job named {1} at time {0}.

User response

None, information message only.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5019I

CWXDG5019I=CWXDG5019I: [{0}] {1}

Explanation

The Compute Grid Native Container is logging the application message {1} at time {0}.

User response

None, information message only.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5020E

CWXDG5020E=CWXDG5020E: Null jobid passed to {0}.

Explanation

The Compute Grid Native Container method {0} was passed a null jobid parameter.

User response

If the caller of {0} is a Grid Utility application, call {0} with a valid job identifier. If it is not, report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5021I

CWXDG5021I=CWXDG5021I: [{0}] Grid Utility Job [{1}] job log output is quiesced: output limit is exceeded.

Explanation

The Compute Grid Native Container has quiesced job log output for the Grid Utility Job named {1} at time {0}. The job log output has been exceeded.

User response

Ensure that the Grid Utility Job {1} is not generating excessive output. If not, adjust the job log output limits for the job class.

Parent topic: [CWXDG](#)

CWXDG5022E

CWXDG5022E=CWXDG5022E: Null timestamp passed to {0}.

Explanation

The Compute Grid Native Container method {0} was passed a null timestamp parameter.

User response

Report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5023E

CWXDG5023E=CWXDG5023E: Compute Grid Native Container method {0} attempted to cache a job log information object, but the object already exists.

Explanation

The Compute Grid Native Container method {0} attempted to cache a job log information object. The attempt failed because the object already exists.

User response

Report problem to IBM Service.

Parent topic: [CWXDG](#)

CWXDG5024E

CWXDG5024E=CWXDG5024E: Compute Grid Native Container failed while creating the job log directory {0}: the exception {1} was caught when java.io.File.getCanonicalPath() was applied against abstract path {2}.

Explanation

The Compute Grid Native Container attempted to create the job log directory named {0}. The creation failed because the exception {1} was caught when java.io.File.getCanonicalPath() was applied against abstract path {2}.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5025E

CWXDG5025E=CWXDG5025E: Compute Grid Native Container failed while creating the job log directory {0}: {0} but is a regular file.

Explanation

The Compute Grid Native Container attempted to create the job log directory named {0}. The creation failed because a regular file with the name {0} already exists.

User response

Ensure that {0} identifies a directory in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5026E

CWXDG5026E=CWXDG5026E: Compute Grid Native Container failed while creating the job log directory {0}:
java.io.File.mkdirs() returned false.

Explanation

The Compute Grid Native Container attempted to create the job log directory named {0}. The directory could not be created: class java.io.File.mkdirs() returned false.

User response

Ensure that {0} identifies a directory in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5027E

CWXDG5027E=CWXDG5027E: Compute Grid Native Container failed while creating the job log directory {0}:
{1}

Explanation

The Compute Grid Native Container attempted to create the job log directory named {0}. The creation attempt failed. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5028E

CWXDG5028E=CWXDG5028E: Compute Grid Native Container failed while creating the job log directory {0}: the exception {1} was caught when java.io.FilePermission() of "read,write" was applied to the directory.

Explanation

The Compute Grid Native Container attempted to create the job log directory named {0}. The creation failed. The exception {1} was caught when java.io.FilePermission() of "read,write" was applied to the directory {0}.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5029E

CWXDG5029E=CWXDG5029E: Compute Grid Native Container failed while creating the job log directory {0}: a file with the name {0} exists.

Explanation

The Compute Grid Native Container attempted to create the job log directory named {0}. The creation failed. A file with the name {0} exists.

User response

Ensure that {0} identifies a directory in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5030E

CWXDG5030E=CWXDG5030E: Compute Grid Native Container failed while validating the job log directory {0}: the directory {0} is not writable.

Explanation

The Compute Grid Native Container was unable to validate the job log directory named {0}. The validation failed. The directory {0} exists but is not writable.

User response

Ensure that {0} identifies a directory in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5031E

CWXDG5031E=CWXDG5031E: Compute Grid Native Container failed while validating the job log directory {0}: the directory {0} is not readable.

Explanation

The Compute Grid Native Container was unable to validate the job log directory named {0}. The validation failed. The directory {0} exists but is not readable.

User response

Ensure that {0} identifies a directory in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5032E

CWXDG5032E=CWXDG5032E: Compute Grid Native Container failed while creating the job log directory {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container attempted to create the job log directory named {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5033E

CWXDG5033E=CWXDG5033E: Null jobclass passed to {0}.

Explanation

The Compute Grid Native Container method {0} was passed a null jobclass parameter.

User response

If the caller of {0} is a Grid Utility application, call {0} with a valid jobclass parameter. If it is not, report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5034E

CWXDG5034E=CWXDG5034E: Illegal parameter(s) passed to {0}.

Explanation

The Compute Grid Native Container method {0} was passed illegal parameter(s).

User response

If the caller of {0} is a Grid Utility application, call {0} with valid parameters. If it is not, report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5035E

CWXDG5035E=CWXDG5035E: Compute Grid Native Container File Logger {0} failed: directory {1} does not exist.

Explanation

The Compute Grid Native Container File Logger method {0} failed: the directory named {1} does not exist.

User response

If the caller of {0} is a Grid Utility application, call {0} with valid parameters. If it is not, report problem to IBM Service.

Parent topic: [CWXDG](#)

CWXDG5036E

CWXDG5036E=CWXDG5036E: Compute Grid Native Container File Logger {0} failed: file {1} does not exist.

Explanation

The Compute Grid Native Container File Logger method {0} failed. The file named {1} does not exist.

User response

If the caller of {0} is a Grid Utility application, call {0} with valid parameters. If it is not, report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5037E

CWXDG5037E=CWXDG5037E: Compute Grid Native Container File Logger {0} failed: caught exception {1} while reading file {2}.

Explanation

The Compute Grid Native Container File Logger method {0} failed. The exception {1} was caught while reading the file {2}.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5038E

CWXDG5038E=CWXDG5038E: Compute Grid Native Container File Logger is unable to capture System Stream: {0} exists but is a regular file.

Explanation

The Compute Grid Native Container File Logger is unable to capture the System Stream (STDOUT or STDERR). The file {0} is not a directory but is a regular file.

User response

Ensure that {0} identifies a directory in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5039E

CWXDG5039E=CWXDG5039E: Compute Grid Native Container File Logger is unable to capture System Stream: unable to create directory {0}.

Explanation

The Compute Grid Native Container File Logger is unable to capture the System Stream (STDOUT or STDERR). The directory {0} could not be created.

User response

Ensure that {0} identifies a directory in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5040E

CWXDG5040E=CWXDG5040E: Compute Grid Native Container File Logger is unable to capture System Stream: unable to create regular file {0}: {1}

Explanation

The Compute Grid Native Container File Logger is unable to capture the System Stream (STDOUT or STDERR). The regular file {0} could not be created. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5041E

CWXDG5041E=CWXDG5041E: Compute Grid Native Container File Logger is unable to capture System Stream {0}: {1}

Explanation

The Compute Grid Native Container File Logger is unable to capture the System Stream {0} (System.out or System.err). The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5042E

CWXDG5042E=CWXDG5042E: Compute Grid Native Container File Logger write {0} failed: {1}

Explanation

The Compute Grid Native Container File Logger is unable to write to the file {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5043E

CWXDG5043E=CWXDG5043E: Compute Grid Native Container failed while writing to the job log file {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container attempted to write to the job log file {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5044E

CWXDG5044E=CWXDG5044E: Compute Grid Native Container failed while attempting to set file permissions for the job log file {0}: the exception {1} was caught when java.io.FilePermission() of "read,write" was applied to the directory.

Explanation

The Compute Grid Native Container attempted to set file permissions for the job log {0}. The attempt failed. The exception {1} was caught when java.io.FilePermission() of "read,write" was applied to the file {0}.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5045E

CWXDG5045E=CWXDG5045E: Compute Grid Native Container failed while attempting to write to the job log file {0}: the file {0} is not writable.

Explanation

The Compute Grid Native Container was unable to write to the job log file {0}. The file {0} is not writable.

User response

Ensure that the parent directory of {0} identifies a directory in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5046E

CWXDG5046E=CWXDG5046E: Compute Grid Native Container failed while attempting to read the job log file {0}: the file {0} is not readable.

Explanation

The Compute Grid Native Container was unable to read the job log file {0}. The file {0} is not readable.

User response

Ensure that the parent directory of {0} identifies a directory in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5047E

CWXDG5047E=CWXDG5047E: Compute Grid Native Container failed while removing the job log file {0}:
java.io.File.delete() returned false.

Explanation

The Compute Grid Native Container attempted to remove the job log file {0}. The file could not be removed:
class java.io.File.delete() returned false.

User response

Ensure that {0} identifies a file in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5048E

CWXDG5048E=CWXDG5048E: Compute Grid Native Container failed while removing the job log file {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container attempted to remove the job log file {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5049E

CWXDG5049E=CWXDG5049E: Compute Grid Native Container failed while removing the job log file {0}: {0} is not a regular file, it is a directory.

Explanation

The Compute Grid Native Container attempted to remove the job log file {0}. {0} is not a regular file. It is a directory.

User response

Report problem to IBM Service.

Parent topic: [CWXDG](#)

CWXDG5050E

CWXDG5050E=CWXDG5050E: Compute Grid Native Container failed while flushing the job log file stream {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container failed while flushing the job log file stream {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5051E

CWXDG5051E=CWXDG5051E: Compute Grid Native Container failed while flushing the job log file writer {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container failed while flushing the job log file writer {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5052E

CWXDG5052E=CWXDG5052E: Compute Grid Native Container failed while writing to the job log file {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container failed while writing to the job log file {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5053E

CWXDG5053E=CWXDG5053E: Compute Grid Native Container failed while writing to the job log class file {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container failed while writing to the job class log file {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5054E

CWXDG5054E=CWXDG5054E: Compute Grid Native Container failed while writing to the job log class file {0}:
java.io.File.createFile() returned false.

Explanation

The Compute Grid Native Container failed while writing to the job log class file {0}. The job log class file could not be created: class java.io.File.createFile() returned false.

User response

Ensure that {0} identifies a file in the file system to which the Compute Grid Native Container has read and write permissions.

Parent topic: [CWXDG](#)

CWXDG5055E

CWXDG5055E=CWXDG5055E: Compute Grid Native Container failed while writing to the job log class file {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container failed while writing to the job log class file {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5056E

CWXDG5056E=CWXDG5056E: Null xJCL passed to {0}.

Explanation

The Compute Grid Native Container method {0} was passed a null xJCL parameter.

User response

If the caller of {0} is a Grid Utility application, call {0} with a valid xJCL parameter. If it is not, report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5057E

CWXDG5057E=CWXDG5057E: Null jobClass passed to {0}.

Explanation

The Compute Grid Native Container method {0} was passed a null job class parameter.

User response

If the caller of {0} is a Grid Utility application, call {0} with a valid job class parameter. If it is not, report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5058E

CWXDG5058E=CWXDG5058E: Null log directory passed to {0}.

Explanation

The Compute Grid Native Container method {0} was passed a null log directory parameter.

User response

If the caller of {0} is a Grid Utility application, call {0} with a valid log directory parameter. If it is not, report problem to IBM Service.

Parent topic: [CWXDG](#)

CWXDG5059E

CWXDG5059E=CWXDG5059E: Compute Grid Native Container is unable to update the job log class list file {0}.

Explanation

The Compute Grid Native Container failed while attempting to update the job log class list file {0}. The file {0} could not be locked.

User response

Report problem to IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5060E

CWXDG5060E=CWXDG5060E: Compute Grid Native Container failed while removing the job class list lock file {0}: java.io.File.delete() returned false.

Explanation

The Compute Grid Native Container attempted to remove the job class list lock file {0}. The file could not be removed: class java.io.File.delete() returned false.

User response

Report problem to IBM Service.

Parent topic: [CWXDG](#)

CWXDG5061E

CWXDG5061E=CWXDG5061E: Compute Grid Native Container failed while removing the job class list lock file {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container attempted to remove the job class list lock file {0}. The exception {1} was caught.

User response

Report problem to IBM Service.

Parent topic: [CWXDG](#)

CWXDG5062I

CWXDG5062I=CWXDG5062I: Compute Grid Native Container job {0} log file(s) are purged: {1}.

Explanation

The Compute Grid Native Container job log file(s) for job {0} were purged. They were purged for reason {1}, which is a message identifier.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5063I

CWXDG5063I=CWXDG5063I: Compute Grid Native Container job {0} log file(s) are purged: maximum job log age limit exceeded.

Explanation

The Compute Grid Native Container job log file(s) for job {0} were purged. They were purged because the maximum job log age limit has been exceeded.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5064I

CWXDG5064I=CWXDG5064I: Compute Grid Native Container job {0} log file(s) are purged: maximum file system log size limit is exceeded.

Explanation

The Compute Grid Native Container job log file(s) for job {0} were purged. They were purged because the maximum file system log size limit was exceeded.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5065I

CWXDG5065I=CWXDG5065I: Compute Grid Native Container job {0} log file(s) are purged: purged by user request.

Explanation

The Compute Grid Native Container job log file(s) for job {0} were purged. They were purged by user request.

User response

None, information message only.

Parent topic: [CWXDG](#)

CWXDG5066E

CWXDG5066E=CWXDG5066E: Compute Grid Native Container failed while writing to the job log file {0}: the exception {1} was caught.

Explanation

The Compute Grid Native Container failed while writing to the job log file {0}. The exception {1} was caught.

User response

Varies by reported cause {1}.

Parent topic: [CWXDG](#)

CWXDG5067I

CWXDG5067I=CWXDG5067I: [{0}] Applying [property {1}] with [value {2}] obtained from {3} to the job xJCL.

Explanation

The Compute Grid Native Container is applying the value {2} for the symbolic variable {1} found in the job xJCL. The value was obtained from {3} (i.e., System.properties). {0} is a timestamp.

User response

None, information message only.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG5068W

CWXDG5068W=CWXDG5068W: [{0}] System property {1} not found.

Explanation

The Compute Grid Native Container attempted to resolved the symbolic variable {1} at time {0}. No System property with that name was found.

User response

Ensure that {1} is a defined System property.

Parent topic: [CWXDG](#)

CWXDG5069I

CWXDG5069I=CWXDG5069I: [{0}] Job xJCL after all symbolic variable substitution(s) performed: {1}

Explanation

The job xJCL {1} after all symbolic variable substitution was performed is displayed at time {0}.

User response

None, information message only.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG8250I

CWXDG_I_XDGridConfig_SchedConfigSuccess=CWXDG8250I: Job scheduler configuration completed successfully.

Explanation

Indicates job scheduler environment successfully configured.

User response

No user action required.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG8251E

CWXDG_I_XDGridConfig_SchedConfigFailed=CWXDG8251E: Job scheduler configuration failed.

Explanation

Indicates job scheduler environment failed to configure correctly.

User response

Contact IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG8252I

CWXDG_I_XDGridConfig_EndpointConfigSuccess=CWXDG8252I: Grid endpoint configuration completed successfully.

Explanation

Indicates grid endpoint environment successfully configured.

User response

No user action required.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG8253E

CWXDG_I_XDGridConfig_EndpointConfigFailed=CWXDG8253E: Grid endpoint configuration failed.

Explanation

Indicates grid endpoint environment failed to configure correctly.

User response

Contact IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG8254E

CWXDG_I_XDGridConfig_SchedException=CWXDG8254E: Job scheduler configuration failed due to error {0}.

Explanation

Indicates job scheduler environment was not successfully configured due to the specified error.

User response

Contact IBM Service.

Parent topic: [CWXDG](#)

CWXDG8255E

CWXDG_I_XDGridConfig_EndpointException=CWXDG8255E: Grid endpoint configuration failed due to error {0}.

Explanation

Indicates grid endpoint environment was not successfully configured due to the specified error.

User response

Contact IBM Service.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXDG8256E

CWXDG_E_XDGridConfig_ConfigOption=CWXDG8256E: Configuration option {0} not supported.

Explanation

Indicates unsupported configuration option specified.

User response

Specify valid configuration option.

Parent topic: [CWXDG](#)

[Terms of Use](#) | [Feedback](#)

CWXMA

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWXMA0001E](#)

CWXMA0001E=CWXMA0001E: The middleware application named "{0}" could not be found.

[CWXMA0002E](#)

CWXMA0002E=CWXMA0002E: The edition named "{0}" could not be found.

[CWXMA0003E](#)

CWXMA0003E=CWXMA0003E: The dynamic cluster named "{0}" could not be found.

[CWXMA0004E](#)

CWXMA0004E=CWXMA0004E: The node named "{0}" could not be found.

[CWXMA0005E](#)

CWXMA0005E=CWXMA0005E: The server named "{0}" could not be found on the node named "{1}".

[CWXMA0006E](#)

CWXMA0006E=CWXMA0006E: The virtual host named "{0}" could not be found.

[CWXMA0007E](#)

CWXMA0007E=CWXMA0007E: The module named "{0}" could not be found.

[CWXMA0008E](#)

CWXMA0008E=CWXMA0008E: The target server or dynamic cluster could not be found.

[CWXMA0009E](#)

CWXMA0009E=CWXMA0009E: The script with type "{0}" could not be found.

[CWXMA0010E](#)

CWXMA0010E=CWXMA0010E: The module named "{0}" could not be found in the archive.

[CWXMA0020E](#)

CWXMA0020E=CWXMA0020E: The edition named "{0}" already exists.

[CWXMA0021E](#)

CWXMA0021E=CWXMA0021E: An application with name "{0}" already exists. Select a different name.

[CWXMA0022E](#)

CWXMA0022E=CWXMA0022E: The context root "{0}" already exists. Specify a different context root.

[CWXMA0040E](#)

CWXMA0040E=CWXMA0040E: The value of context root is invalid. The context root must start with a "/" character.

[CWXMA0041E](#)

CWXMA0041E=CWXMA0041E: The value of context root is invalid. The context root cannot be only "/".

[CWXMA0042E](#)

CWXMA0042E=CWXMA0042E: A validation error occurred in task {0}. Application name, {1}, is not valid. An application name cannot begin with a leading dot, cannot have leading or trailing spaces and cannot contain any of the following characters: / , # \$ @ : ; " * ? < > | = + & % '' ;]

[CWXMA0060E](#)

CWXMA0060E=CWXMA0060E: The file "{0}" could not be found.

[CWXMA0061E](#)

CWXMA0061E=CWXMA0061E: The file "{0}" has an invalid extension. Valid extensions for this file include: "{1}".

[CWXMA0062E](#)

CWXMA0062E=CWXMA0062E: The length of file "{0}" is zero.

[CWXMA0080E](#)

CWXMA0080E=CWXMA0080E: The required cluster parameter or node and server parameters are not specified.

[CWXMA0081E](#)

CWXMA0081E=CWXMA0081E: Both the node and server parameters must be specified.

[CWXMA0082E](#)

CWXMA0082E=CWXMA0082E: The cluster parameter cannot be specified in conjunction with the node or server parameters.

CWXMA0083E

CWXMA0083E=CWXMA0083E: The application "{0}" has no editions.

CWXMA0084E

CWXMA0084E=CWXMA0084E: The edition parameter is required, as the application "{0}" has several editions.

CWXMA0085E

CWXMA0085E=CWXMA0085E: The server named "{0}" on the node named "{1}" is not a standalone server.

CWXMA0086E

CWXMA0086E=CWXMA0086E: The server named "{0}" on the node named "{1}" is of type "{2}", which is not a server type supported by the application. The application is only supported by the following server types: "{3}".

CWXMA0087E

CWXMA0087E=CWXMA0087E: The application named "{0}" is of type "{1}". An application of type "{2}" is expected.

CWXMA0088E

CWXMA0088E=CWXMA0088E: The cluster named "{0}" is of type "{1}", which is not a cluster type supported by the application. The application is only supported by the following cluster types: "{2}".

CWXMA0089E

CWXMA0089E=CWXMA0089E: The module named "{0}" is not specified.

CWXMA0090E

CWXMA0090E=CWXMA0090E: The required moduleName parameter must be specified and cannot be empty.

CWXMA0101E

CWXMA0101E=CWXMA0101E: The application named "{0}" cannot be started because it is not deployed to any targets.

CWXMA0102E

CWXMA0102E=CWXMA0102E: The application named "{0}" cannot be stopped because it is not deployed to any targets.

CWXMA0103E

CWXMA0103E=CWXMA0103E: The application named "{0}" failed to be started.

CWXMA0104E

CWXMA0104E=CWXMA0104E: The application named "{0}" failed to be stopped.

CWXMA0105E

CWXMA0105E=CWXMA0105E: The application named "{0}" failed to be started on server "{1}" on node "{2}".

CWXMA0106E

CWXMA0106E=CWXMA0106E: The application named "{0}" failed to be stopped on server "{1}" on node "{2}".

CWXMA0107E

CWXMA0107E=CWXMA0107E: The application named "{0}" does not exist on server "{1}" on node "{2}".

CWXMA0108E

CWXMA0108E=CWXMA0108E: Edition "{0}" of application "{1}" failed to be deployed onto server "{2}" on node "{3}".

CWXMA0109E

CWXMA0109E=CWXMA0109E: Edition "{0}" of application "{1}" failed to be undeployed from server "{2}" on node "{3}".

CWXMA0110E

CWXMA0110E=CWXMA0110E: Edition "{0}" of application "{1}" does not exist on server "{2}" on node "{3}".

CWXMA0121I

CWXMA0121I=CWXMA0121I: The application named "{0}" is started on server "{1}" on node "{2}".

CWXMA0122I

CWXMA0122I=CWXMA0122I: The application named "{0}" is stopped on server "{1}" on node "{2}".

CWXMA0123I

CWXMA0123I=CWXMA0123I: Edition "{0}" of application "{1}" has been deployed onto server "{2}" on node "{3}".

CWXMA0124I

CWXMA0124I=CWXMA0124I: Edition "{0}" of application "{1}" has been undeployed from server "{2}" on node "{3}".

CWXMA0001E

CWXMA0001E=CWXMA0001E: The middleware application named "{0}" could not be found.

Explanation

The specified middleware application name could not be located.

User response

Ensure the specified application name is a valid name for an existing middleware application.

Parent topic: [CWXMA](#)

CWXMA0002E

CWXMA0002E=CWXMA0002E: The edition named "{0}" could not be found.

Explanation

The specified edition name could not be located.

User response

Ensure the specified edition name is a valid name for an existing edition.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0003E

CWXMA0003E=CWXMA0003E: The dynamic cluster named "{0}" could not be found.

Explanation

The specified dynamic cluster name could not be located.

User response

Ensure the specified cluster name is a valid name for an existing dynamic cluster.

Parent topic: [CWXMA](#)

CWXMA0004E

CWXMA0004E=CWXMA0004E: The node named "{0}" could not be found.

Explanation

The specified node name could not be located.

User response

Ensure the specified node name is a valid name for an existing node.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0005E

CWXMA0005E=CWXMA0005E: The server named "{0}" could not be found on the node named "{1}".

Explanation

The specified server name could not be located.

User response

Ensure the specified server name is a valid name for an existing server.

Parent topic: [CWXMA](#)

CWXMA0006E

CWXMA0006E=CWXMA0006E: The virtual host named "{0}" could not be found.

Explanation

The specified virtual host name could not be located.

User response

Ensure the specified virtual host name is a valid name for an existing virtual host.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0007E

CWXMA0007E=CWXMA0007E: The module named "{0}" could not be found.

Explanation

The specified module name could not be located.

User response

Ensure the specified module name is a valid name for an existing module.

Parent topic: [CWXMA](#)

CWXMA0008E

CWXMA0008E=CWXMA0008E: The target server or dynamic cluster could not be found.

Explanation

The specified target could not be located.

User response

Ensure the specified target is a valid target for an existing module.

Parent topic: [CWXMA](#)

CWXMA0009E

CWXMA0009E=CWXMA0009E: The script with type "{0}" could not be found.

Explanation

The specified script could not be located.

User response

Ensure the specified script type is a valid type for an existing script.

Parent topic: [CWXMA](#)

CWXMA0010E

CWXMA0010E=CWXMA0010E: The module named "{0}" could not be found in the archive.

Explanation

The specified module could not be located inside the application archive.

User response

Ensure the specified module is a valid module in the archive.

Parent topic: [CWXMA](#)

CWXMA0020E

CWXMA0020E=CWXMA0020E: The edition named "{0}" already exists.

Explanation

Edition names must be unique for an application.

User response

Specify a unique edition name that is not already in existence for the application.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0021E

CWXMA0021E=CWXMA0021E: An application with name "{0}" already exists. Select a different name.

Explanation

Application names must be unique.

User response

Specify a unique application name that is not already in existence.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0022E

CWXMA0022E=CWXMA0022E: The context root "{0}" already exists. Specify a different context root.

Explanation

Context roots must be unique.

User response

Specify a unique context root that is not already in existence.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0040E

CWXMA0040E=CWXMA0040E: The value of context root is invalid. The context root must start with a "/" character.

Explanation

The context root must start with a "/" character.

User response

Specify a context root that starts with a "/" character.

Parent topic: [CWXMA](#)

CWXMA0041E

CWXMA0041E=CWXMA0041E: The value of context root is invalid. The context root cannot be only "/".

Explanation

The context root cannot be only "/".

User response

Specify a context root that starts with "/" and follows by other alphanumeric characters.

Parent topic: [CWXMA](#)

CWXMA0042E

CWXMA0042E=CWXMA0042E: A validation error occurred in task {0}. Application name, {1}, is not valid. An application name cannot begin with a leading dot, cannot have leading or trailing spaces and cannot contain any of the following characters: / , # \$ @ : ; " * ? < > | = + & % '']]

Explanation

The specified application name contains an unacceptable character. An application name cannot begin with a leading dot, cannot contain leading or trailing spaces and cannot contain any of the following characters: backward slash (\), slash (/), number sign (#), dollar (\$), at sign (@), colon (:), asterisk (*), question mark (?), quotation marks ("), greater than sign (>), less than sign (<), pipe (|), semicolon (;), or comma (,), equal sign (=), plus (+), percent (%), ampersand (&), single quotation mark ('), square brackets (]).

User response

Specify an application name that does not contain any of these unacceptable characters.

Parent topic: [CWXMA](#)

CWXMA0060E

CWXMA0060E=CWXMA0060E: The file "{0}" could not be found.

Explanation

The specified file could not be located.

User response

Ensure the specified file name is a valid name for an existing file.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0061E

CWXMA0061E=CWXMA0061E: The file "{0}" has an invalid extension. Valid extensions for this file include: "{1}".

Explanation

The specified file has an invalid extension.

User response

Ensure the specified file name extension is valid.

Parent topic: [CWXMA](#)

CWXMA0062E

CWXMA0062E=CWXMA0062E: The length of file "{0}" is zero.

Explanation

The specified file is zero bytes in length.

User response

Specify a file with a length greater than zero bytes.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0080E

CWXMA0080E=CWXMA0080E: The required cluster parameter or node and server parameters are not specified.

Explanation

Either the cluster parameter or node and server parameters are not specified.

User response

Ensure either the cluster parameter or node and server parameters are specified.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0081E

CWXMA0081E=CWXMA0081E: Both the node and server parameters must be specified.

Explanation

The node and server parameters are mutually dependent parameters.

User response

Ensure both the node and server parameters are specified.

Parent topic: [CWXMA](#)

CWXMA0082E

CWXMA0082E=CWXMA0082E: The cluster parameter cannot be specified in conjunction with the node or server parameters.

Explanation

The cluster parameter and node and server parameters are mutually exclusive parameters.

User response

Ensure only the cluster parameter or node and server parameters are specified.

Parent topic: [CWXMA](#)

CWXMA0083E

CWXMA0083E=CWXMA0083E: The application "{0}" has no editions.

Explanation

The application has no editions defined in the configuration repository.

User response

Ensure the application has an edition defined in the configuration repository, or uninstall the application.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0084E

CWXMA0084E=CWXMA0084E: The edition parameter is required, as the application "{0}" has several editions.

Explanation

The command requires the edition parameter for applications with several editions.

User response

Specify the edition parameter.

Parent topic: [CWXMA](#)

CWXMA0085E

CWXMA0085E=CWXMA0085E: The server named "{0}" on the node named "{1}" is not a standalone server.

Explanation

The specified server is not a standalone server.

User response

Ensure the specified server name is a valid name for an existing standalone server.

Parent topic: [CWXMA](#)

CWXMA0086E

CWXMA0086E=CWXMA0086E: The server named "{0}" on the node named "{1}" is of type "{2}", which is not a server type supported by the application. The application is only supported by the following server types: "{3}".

Explanation

The specified server is not a type of server that supports the application.

User response

Ensure the specified server name is a valid type for the application.

Parent topic: [CWXMA](#)

CWXMA0087E

CWXMA0087E=CWXMA0087E: The application named "{0}" is of type "{1}". An application of type "{2}" is expected.

Explanation

The specified application is not of the expected type.

User response

Specify an application of the expected type.

Parent topic: [CWXMA](#)

CWXMA0088E

CWXMA0088E=CWXMA0088E: The cluster named "{0}" is of type "{1}", which is not a cluster type supported by the application. The application is only supported by the following cluster types: "{2}".

Explanation

The specified cluster is not a type of cluster that supports the application.

User response

Ensure the specified cluster name is a valid type for the application.

Parent topic: [CWXMA](#)

CWXMA0089E

CWXMA0089E=CWXMA0089E: The module named "{0}" is not specified.

Explanation

The required module that is in the application archive is not specified.

User response

Specify the module and the necessary configuration parameters.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0090E

CWXMA0090E=CWXMA0090E: The required moduleName parameter must be specified and cannot be empty.

Explanation

The moduleName parameter must be specified and cannot be empty.

User response

Ensure the moduleName parameter is specified and is not empty.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0101E

CWXMA0101E=CWXMA0101E: The application named "{0}" cannot be started because it is not deployed to any targets.

Explanation

The specified application needs to be deployed to at least one deployment target before it can be started.

User response

Associate the application with the necessary deployment targets.

Parent topic: [CWXMA](#)

CWXMA0102E

CWXMA0102E=CWXMA0102E: The application named "{0}" cannot be stopped because it is not deployed to any targets.

Explanation

The specified application needs to be deployed to at least one deployment target before it can be stopped.

User response

Associate the application with the necessary deployment targets.

Parent topic: [CWXMA](#)

CWXMA0103E

CWXMA0103E=CWXMA0103E: The application named "{0}" failed to be started.

Explanation

Error(s) might have occurred when the specified application was attempted to be started.

User response

Check the server logs for details, resolve any issues, then retry.

Parent topic: [CWXMA](#)

CWXMA0104E

CWXMA0104E=CWXMA0104E: The application named "{0}" failed to be stopped.

Explanation

Error(s) might have occurred when the specified application was attempted to be stopped.

User response

Check the server logs for details, resolve any issues, then retry.

Parent topic: [CWXMA](#)

CWXMA0105E

CWXMA0105E=CWXMA0105E: The application named "{0}" failed to be started on server "{1}" on node "{2}".

Explanation

Error(s) occurred when the specified application was attempted to be started on the particular server.

User response

Check the server logs for details, resolve any issues, then retry.

Parent topic: [CWXMA](#)

CWXMA0106E

CWXMA0106E=CWXMA0106E: The application named "{0}" failed to be stopped on server "{1}" on node "{2}".

Explanation

Error(s) occurred when the specified application was attempted to be stopped on the particular server.

User response

Check the server logs for details, resolve any issues, then retry.

Parent topic: [CWXMA](#)

CWXMA0107E

CWXMA0107E=CWXMA0107E: The application named "{0}" does not exist on server "{1}" on node "{2}".

Explanation

The specified application is absent on the particular server.

User response

Ensure that the application is properly installed, check the server logs for details, resolve any issues, then retry.

Parent topic: [CWXMA](#)

CWXMA0108E

CWXMA0108E=CWXMA0108E: Edition "{0}" of application "{1}" failed to be deployed onto server "{2}" on node "{3}".

Explanation

Error(s) occurred when the specified application edition was attempted to be deployed onto the particular server.

User response

Check the server logs for details, resolve any issues, then retry.

Parent topic: [CWXMA](#)

CWXMA0109E

CWXMA0109E=CWXMA0109E: Edition "{0}" of application "{1}" failed to be undeployed from server "{2}" on node "{3}".

Explanation

Error(s) occurred when the specified application edition was attempted to be undeployed from the particular server.

User response

Check the server logs for details, resolve any issues, then retry.

Parent topic: [CWXMA](#)

CWXMA0110E

CWXMA0110E=CWXMA0110E: Edition "{0}" of application "{1}" does not exist on server "{2}" on node "{3}".

Explanation

The specified application edition is absent on the particular server.

User response

Check the server logs for details, resolve any issues, then retry.

Parent topic: [CWXMA](#)

CWXMA0121I

CWXMA0121I=CWXMA0121I: The application named "{0}" is started on server "{1}" on node "{2}".

Explanation

The specified application started successfully on the particular server.

User response

No user action is required.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0122I

CWXMA0122I=CWXMA0122I: The application named "{0}" is stopped on server "{1}" on node "{2}".

Explanation

The specified application stopped successfully on the particular server.

User response

No user action is required.

Parent topic: [CWXMA](#)

[Terms of Use](#) | [Feedback](#)

CWXMA0123I

CWXMA0123I=CWXMA0123I: Edition "{0}" of application "{1}" has been deployed onto server "{2}" on node "{3}".

Explanation

The specified application is deployed successfully onto the particular server.

User response

No user action is required.

Parent topic: [CWXMA](#)

CWXMA0124I

CWXMA0124I=CWXMA0124I: Edition "{0}" of application "{1}" has been undeployed from server "{2}" on node "{3}".

Explanation

The specified application is undeployed successfully from the particular server.

User response

No user action is required.

Parent topic: [CWXMA](#)

CWXQY

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[CWXQY1200E](#)

FAILTOPARSE=CWXQY1200E: Syntax error. Encountered {0} at line {1}, column {2}.

[CWXQY1201E](#)

INVAGGINWHERE=CWXQY1201E: Aggregate functions are not allowed in where clause.

[CWXQY1202E](#)

FAILINITCONFIG=CWXQY1202E: Failed to initialize ObjectMap configuration {0}.

[CWXQY1203E](#)

UNDEFINEDALIAS=CWXQY1203E: {0} can not be resolved.

[CWXQY1204E](#)

UNDEFINEDNAME=CWXQY1204E: {0} can not be resolved.

[CWXQY1205E](#)

UNDEFINEDCONSTR=CWXQY1205E: The constructor {0} is not defined.

[CWXQY1206E](#)

AMBIGUOUSCONSTR=CWXQY1206E: Ambiguous constructors.

[CWXQY1207E](#)

BADNAVIGATIONCMP=CWXQY1207E: Can not apply navigation operation operator to {0}

[CWXQY1208E](#)

BADNAVIGATION=CWXQY1208E: Can not apply navigation operation to {0}

[CWXQY1209E](#)

BADRETURN=CWXQY1209E: Invalid select clause {0}

[CWXQY1210E](#)

INTERNAL=CWXQY1210E: Internal Error. {0}

[CWXQY1211E](#)

RESOLVEERRORS=CWXQY1211E: One or more resolution errors. {0}

[CWXQY1212E](#)

SUBQRNUMPROP=CWXQY1212E: Subquery can only return a single property.

[CWXQY1213E](#)

INVALIDGRPBY=CWXQY1213E: The property type {0} used in the group by clause does not support grouping.

[CWXQY1214E](#)

INVALIDORDBY=CWXQY1214E: The property type {0} used in order by clause does not support ordering.

[CWXQY1215E](#)

ARGMUSTBASIC=CWXQY1215E: Operator is not supported on given property type.

[CWXQY1217E](#)

CANTCOMPARI=CWXQY1217E: Comparison operator not supported on given types.

[CWXQY1220E](#)

EQNOTSUPPORTED=CWXQY1220E: Equal operator is not supported on given property types.

[CWXQY1221E](#)

EQONLONG=CWXQY1221E: Equal operator is not supported on long types.

[CWXQY1222E](#)

COMPENTITYCOLL=CWXQY1222E: Comparison not supported on entity collections.

[CWXQY1223E](#)

INVALIDCMP=CWXQY1223E: Comparison operator used incorrectly.

[CWXQY1225E](#)

LIKENONCHAR=CWXQY1225E: Operator LIKE is not supported on given property types.

[CWXQY1227E](#)

MAXMINBADARG=CWXQY1227E: Function min or max has invalid argument.

[CWXQY1229E](#)

ARGMUSTBOOLEA=CWXQY1229E: The operand has to be of boolean type.

CWXQY1230E

ALLOWONLYONEO=CWXQY1230E: Only one operand is allowed.

CWXQY1231E

INVALIDARGTYP=CWXQY1231E: Invalid argument type.

CWXQY1232E

ONETOTHREEARG=CWXQY1232E: Function requires at least one argument and no more than three arguments.

CWXQY1233E

TWOTOTHREEARG=CWXQY1233E: Function requires at least two arguments and no more than three arguments.

CWXQY1234E

TWOARG=CWXQY1234E: Function requires two arguments.

CWXQY1235E

ONEARG=CWXQY1235E: Function requires one argument.

CWXQY1236E

SUMARGNUMERIC=CWXQY1236E: Function sum argument must be a type of numeric.

CWXQY1238E

AVGARGNUMERIC=CWXQY1238E: Function avg argument must be numeric.

CWXQY1240E

ASSINGDIFFTYP=CWXQY1240E: Assignment on different types not supported.

CWXQY1241E

ASSIGNCONVFAL=CWXQY1241E: Conversion of a numeric type failed on assignment.

CWXQY1242E

ARITHNONNUMER=CWXQY1242E: Arithmetic operation not supported on nonnumeric types.

CWXQY1243E

UNKNOWNSCALAR=CWXQY1243E: Function {0} is not a supported function.

CWXQY1244E

TYPECHECKERRORS=CWXQY1244E: One or more properties used incorrectly. {0}

CWXQY1246E

ALIASDUP=CWXQY1246E: Identifier "{0}" is already defined.

CWXQY1247E

OPERATORNOTSUPPORTED=CWXQY1247E: {0} operation is not supported.

CWXQY1248E

INVALIDENTITYCOMP=CWXQY1248E: Comparison between entity beans of different types not allowed.

CWXQY1249E

REQUIRECOLLECTION=CWXQY1249E: Empty predicate can only be applied to a valued relationship.

CWXQY1250E

INVALIDMEMBEROF=CWXQY1250E: Member predicate can not be applied to given property types.

CWXQY1251E

UNDEFINEDCONSTT=CWXQY1251E: Internal error. Invalid constant {0}.

CWXQY1252E

MISSINGORDERBYTERM=CWXQY1252E: ORDER BY term does not appear in SELECT.

CWXQY1253E

MISSINGIDEXOBJ=CWXQY1253E: No index available for ObjectMap {0}.

CWXQY1254E

MISSINGALIASCAT=CWXQY1254E: Internal error MISSINGALIASCAT.

CWXQY1255E

WRONGTERMFORGP=CWXQY1255E: The field {0} appears in a SELECT or HAVING clause without an aggregate function but is not specified in the GROUP BY clause.

CWXQY1256E

NONESTEDAGGFUNC=CWXQY1256E: Nested aggregate functions are not allowed.

CWXQY1257E

AGGHASMOREDISTIN=CWXQY1257E: DISTINCT is specified more than once in aggregate functions.

CWXQY1258E

INVALIDNEXTSTATE=CWXQY1258E: Internal error. Invalid state on call to next.

CWXQY1259E

ARITHMETICOPFAIL=CWXQY1259E: An exception occurred while evaluating the arithmetic expression {0}.

CWXQY1260E

ARITHMETICOVERFLOW=CWXQY1260E: Underflow or overflow occurred while evaluating the arithmetic expression {0}.

CWXQY1261E

ARITHMETICDIVBYZERO=CWXQY1261E: An Arithmetic exception occurred due to division by zero.

CWXQY1262E

NOTFOUNDINMAP=CWXQY1262E: ObjectMap {0} not found.

CWXQY1263E

NOTFOUNDINDEX=CWXQY1263E: An [{0}] occurred because the ObjectMap [{1}] does not have index [{2}].

CWXQY1264E

NOOBJECTININDEX=CWXQY1264E: An [{0}] occurred because the index [{1}] does not contain the object [{2}].

CWXQY1265E

EVALINTERNALERROR=CWXQY1265E: An internal error found in [{0}].

CWXQY1266E

INTROSPMETHOD=CWXQY1266E: [{0}] occurred while introspecting method [{1}] of class [{2}].

CWXQY1267E

FIELDGETOBJECTFAILED=CWXQY1267E: [{0}] occurred because the specified object [{1}] is not an instance of the class or interface declaring the underlying field [{2}].

CWXQY1268E

INVOKMETHODFAIL=CWXQY1268E: [{0}] occurred while invoking method [{1}] on object [{2}].

CWXQY1269E

FIELDACCESSFAILED=CWXQY1269E: [{0}] occurred because the field [{1}] is inaccessible.

CWXQY1270E

DATEWRONGJDBCESCAPE=CWXQY1270E: Date given [{0}] is not in the JDBC date escape format[yyyy-mm-dd].

CWXQY1271E

TIMEWRONGJDBCESCAPE=CWXQY1271E: Time given [{0}] is not in the JDBC time escape format[hh:mm:ss].

CWXQY1272E

TIMESTAMPWRONGJDBCESCAPE=CWXQY1272E: Timestamp given [{0}] is not in the JDBC timestamp escape format[yyyy-mm-dd hh:mm:ss.ffffff].

CWXQY1282E

SUBSTRWRONGRANGE=CWXQY1282E: The second or third argument of the SUBSTR function is out of range.

CWXQY1283E

NOTNEEDEDPARAMETER=CWXQY1283E: Parameter {0} is not used in the query.

CWXQY1285E

OVERFLOWAVG=CWXQY1285E: Counter overflow occurred computing AVG.

CWXQY1286E

OVERFLOWCOUNT=CWXQY1286E: Counter overflow occurred computing COUNT.

CWXQY1287E

TOOMANYPROJTIONITEMS=CWXQY1287E: Exceeds the maximum number of elements [{0}] allowed in Tuple object fail to add the element [{1}].

CWXQY1288E

FEWPARAMETER=CWXQY1288E: The query uses {1} parameters but only {0} were passed.

CWXQY1289E

NOPARAMETER=CWXQY1289E: No parameters were passed to a query that required parameters.

CWXQY1290E

NOTDEFINEDPARAMETER=CWXQY1290E: Parameter {0} is not defined.

CWXQY1291E

INVALIDPARAMETERATYPE=CWXQY1291E: Parameter {0} passed in is a type of {1} which is not the expected type {2}.

CWXQY1292E

SCALARSUBQNODEDATE=CWXQY1292E: Scalar subquery returned no data.

[CWXQY1293E](#)

SCALARSUBQMORCOL=CWXQY1293E: Scalar subquery returns more than one column.

[CWXQY1294E](#)

SCALARSUBQMOREROW=CWXQY1294E: Scalar subquery returned more than one row.

[CWXQY1296E](#)

INVALIDINDEXTYPE=CWXQY1296E: Internal Error. Undefined type [{0}] for index field.

[CWXQY1297E](#)

ONECHARACTERONLY=CWXQY1297E: Character can only be compared to Character or String of length 1. {0} has more than one character.

[CWXQY1298E](#)

CONFLICTNAME=CWXQY1298E: Duplicate name {0} in select expressions.

[CWXQY1299E](#)

INVALIDTOKEN=CWXQY1299E: Invalid token is found in query. {0} in "{1}".

[CWXQY1300E](#)

PARSEERROR=CWXQY1300E: Query parser encountered an error. {0}.

[CWXQY1301E](#)

INVALIDAS=CWXQY1301E: {0} of an aggregate function is not valid in the context where it is used.

[CWXQY1302E](#)

INVALIDMEMBEROFMISPK=CWXQY1302E: Member operation failed either because a primary key is not contained in the object or the primary key is not identified in the object's metadata.

[CWXQY1304E](#)

INVALIDEQMISPK=CWXQY1304E: Equal operation failed either because a primary key is not contained in the object or the primary key is not identified in the object's metadata.

CWXQY1200E

FAILTOPARSE=CWXQY1200E: Syntax error. Encountered {0} at line {1}, column {2}.

Explanation

Syntax error. The query string may contain syntax error or a numeric literal that is out of range.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1201E

INVAGGINWHERE=CWXQY1201E: Aggregate functions are not allowed in where clause.

Explanation

The query does not satisfy the syntax rule.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1202E

FAILINITCONFIG=CWXQY1202E: Failed to initialize ObjectMap configuration {0}.

Explanation

The ObjectMap is not configured properly.

User response

Correct the ObjectMap configuration.

Parent topic: [CWXQY](#)

CWXQY1203E

UNDEFINEDALIAS=CWXQY1203E: {0} can not be resolved.

Explanation

The identification variable is not defined.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1204E

UNDEFINEDNAME=CWXQY1204E: {0} can not be resolved.

Explanation

The property is not defined.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1205E

UNDEFINEDCONSTR=CWXQY1205E: The constructor {0} is not defined.

Explanation

Undefined constructor {0}.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1206E

AMBIGUOUSCONSTR=CWXQY1206E: Ambiguous constructors.

Explanation

Ambiguous constructors.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1207E

BADNAVIGATIONCMP=CWXQY1207E: Can not apply navigation operation operator to {0}

Explanation

A navigation operation can not be applied to a multi valued relationship or property.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1208E

BADNAVIGATION=CWXQY1208E: Can not apply navigation operation to {0}

Explanation

A navigation operation can not be applied to a multi valued relationship or property.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1209E

BADRETURN=CWXQY1209E: Invalid select clause {0}

Explanation

SELECT clause cannot project a multi valued relationship.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1210E

INTERNAL=CWXQY1210E: Internal Error. {0}

Explanation

Internal Error.

User response

Report this problem to IBM service.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1211E

RESOLVEERRORS=CWXQY1211E: One or more resolution errors. {0}

Explanation

One or more resolution errors.

User response

Read details followed by this message.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1212E

SUBQRYNUMPROP=CWXQY1212E: Subquery can only return a single property.

Explanation

A subquery contains multiple elements in the select clause.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1213E

INVALIDGRPBY=CWXQY1213E: The property type {0} used in the group by clause does not support grouping.

Explanation

The property type used in the group by clause does not support grouping.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1214E

INVALIDORDBY=CWXQY1214E: The property type {0} used in order by clause does not support ordering.

Explanation

The property type used in order by clause does not support ordering.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1215E

ARGMUSTBASIC=CWXQY1215E: Operator is not supported on given property type.

Explanation

Operator is not supported on given property type.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1217E

CANTCOMPARI=CWXQY1217E: Comparison operator not supported on given types.

Explanation

Comparison operator not supported on given types.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1220E

EQNOTSUPPORTED=CWXQY1220E: Equal operator is not supported on given property types.

Explanation

Equal operator is not supported on given property types.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1221E

EQONLONG=CWXQY1221E: Equal operator is not supported on long types.

Explanation

Equal operator is not supported on long types.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1222E

COMPENTITYCOLL=CWXQY1222E: Comparison not supported on entity collections.

Explanation

Comparison not supported on entity collections.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1223E

INVALIDCMP=CWXQY1223E: Comparison operator used incorrectly.

Explanation

Comparison operator used incorrectly.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1225E

LIKENONCHAR=CWXQY1225E: Operator LIKE is not supported on given property types.

Explanation

Operator LIKE is not supported on non character types.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1227E

MAXMINBADARG=CWXQY1227E: Function min or max has invalid argument.

Explanation

Function min or max has invalid argument.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1229E

ARGMUSTBOOLEA=CWXQY1229E: The operand has to be of boolean type.

Explanation

The operand has to be of boolean type.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1230E

ALLOWONLYONEO=CWXQY1230E: Only one operand is allowed.

Explanation

Only one operand is allowed.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1231E

INVALIDARGTYP=CWXQY1231E: Invalid argument type.

Explanation

Invalid argument type.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1232E

ONETOTHREEARG=CWXQY1232E: Function requires at least one argument and no more than three arguments.

Explanation

Function requires at least one argument and no more than three arguments.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1233E

TWOTOTHREEARG=CWXQY1233E: Function requires at least two arguments and no more than three arguments.

Explanation

Function requires at least two arguments and no more than three arguments.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1234E

TWOARG=CWXQY1234E: Function requires two arguments.

Explanation

Function requires two arguments.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1235E

ONEARG=CWXQY1235E: Function requires one argument.

Explanation

Function requires one argument.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1236E

SUMARGNUMERIC=CWXQY1236E: Function sum argument must be a type of numeric.

Explanation

Function sum argument must be a type of numeric.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1238E

AVGARGNUMERIC=CWXQY1238E: Function avg argument must be numeric.

Explanation

Function avg argument must be numeric.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1240E

ASSINGDIFFTYP=CWXQY1240E: Assignment on different types not supported.

Explanation

Assignment on different types not supported.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1241E

ASSIGNCONVFAI=CWXQY1241E: Conversion of a numeric type failed on assignment.

Explanation

Conversion of a numeric type failed on assignment.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1242E

ARITHNONNUMER=CWXQY1242E: Arithmetic operation not supported on nonnumeric types.

Explanation

Arithmetic operation not supported on nonnumeric types.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1243E

UNKNOWNSCALAR=CWXQY1243E: Function {0} is not a supported function.

Explanation

Function is not supported in EJB query language.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1244E

TYPECHECKERRORS=CWXQY1244E: One or more properties used incorrectly. {0}

Explanation

One or more properties used incorrectly.

User response

Read details followed by this message.

Parent topic: [CWXQY](#)

CWXQY1246E

ALIASDUP=CWXQY1246E: Identifier "{0}" is already defined.

Explanation

Identifier is already defined.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1247E

OPERATORNOTSUPPORTED=CWXQY1247E: {0} operation is not supported.

Explanation

operation is not supported.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1248E

INVALIDENTITYCOMP=CWXQY1248E: Comparison between entity beans of different types not allowed.

Explanation

Comparison between entity beans of different types not allowed.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1249E

REQUIRECOLLECTION=CWXQY1249E: Empty predicate can only be applied to a valued relationship.

Explanation

Empty predicate can only be applied to a valued relationship.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1250E

INVALIDMEMBEROF=CWXQY1250E: Member predicate can not be applied to given property types.

Explanation

The Member predicate can only compare an entity bean to a collection of beans of a compatible type.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1251E

UNDEFINEDCONSTT=CWXQY1251E: Internal error. Invalid constant {0}.

Explanation

Internal error. Invalid constant {0}.

User response

Report this problem to IBM service.

Parent topic: [CWXQY](#)

CWXQY1252E

MISSINGORDERBYTERM=CWXQY1252E: ORDER BY term does not appear in SELECT.

Explanation

If the ORDER BY clause is used, the element being ordered by must appear in the SELECT clause.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1253E

MISSINGIDEXOBJ=CWXQY1253E: No index available for ObjectMap {0}.

Explanation

At least one index should be defined for each ObjectMap.

User response

Correct the metadata.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1254E

MISSINGALIASCAT=CWXQY1254E: Internal error MISSINGALIASCAT.

Explanation

Internal error.

User response

Report this problem to IBM service.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1255E

WRONGTERMFORGP=CWXQY1255E: The field {0} appears in a SELECT or HAVING clause without an aggregate function but is not specified in the GROUP BY clause.

Explanation

When performing a grouping operation, elements of the SELECT clause must either be aggregation functions or be grouping elements. The field indicated by token {0} is used in the SELECT clause and does not appear in an aggregation function but is not a grouping element.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1256E

NONESTEDAGGFUNC=CWXQY1256E: Nested aggregate functions are not allowed.

Explanation

An aggregate function such as SUM cannot contain another aggregate function in the argument expression.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1257E

AGGHASMOREDISTIN=CWXQY1257E: DISTINCT is specified more than once in aggregate functions.

Explanation

You cannot use DISTINCT more than once in aggregate functions in a query.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1258E

INVALIDNEXTSTATE=CWXQY1258E: Internal error. Invalid state on call to next.

Explanation

Internal error.

User response

Report this problem to IBM.

Parent topic: [CWXQY](#)

CWXQY1259E

ARITHMETICOPFAIL=CWXQY1259E: An exception occurred while evaluating the arithmetic expression {0}.

Explanation

An exception occurred while evaluating the given expression.

User response

Correct the arithmetic expression.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1260E

ARITHMETICOVERFLOW=CWXQY1260E: Underflow or overflow occurred while evaluating the arithmetic expression {0}.

Explanation

Underflow or overflow occurred while evaluating the arithmetic expression {0}.

User response

Correct the arithmetic expression.

Parent topic: [CWXQY](#)

CWXQY1261E

ARITHMETICDIVBYZERO=CWXQY1261E: An Arithmetic exception occurred due to division by zero.

Explanation

An Arithmetic exception occurred due to division by zero.

User response

Correct the arithmetic expression.

Parent topic: [CWXQY](#)

CWXQY1262E

NOTFOUNDINMAP=CWXQY1262E: ObjectMap {0} not found.

Explanation

ObjectMap {0} is either not defined or it is defined but it is not found in this server.

User response

Correct the query statement and make sure all ObjectMap referenced in the query statement are defined and can be found in the same server.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1263E

NOTFOUNDINDEX=CWXQY1263E: An [{0}] occurred because the ObjectMap [{1}] does not have index [{2}].

Explanation

The ObjectGrid Entity Manager catalog is out of sync with the actual ObjectMap configuration.

User response

Check the ObjectMap configuration.

Parent topic: [CWXQY](#)

CWXQY1264E

NOBJECTININDEX=CWXQY1264E: An [{0}] occurred because the index [{1}] does not contain the object [{2}].

Explanation

An object {2} returned from the ObjectMap using index {1} is not the expected type.

User response

Check the class definition and the ObjectMap and index configuration.

Parent topic: [CWXQY](#)

CWXQY1265E

EVALINTERNALERROR=CWXQY1265E: An internal error found in [{0}].

Explanation

An internal error found in [{0}].

User response

Report this problem to IBM service.

Parent topic: [CWXQY](#)

CWXQY1266E

INTROSPMETHOD=CWXQY1266E: [{0}] occurred while introspecting method [{1}] of class [{2}].

Explanation

Methods should be defined as public, must not be void and have no arguments.

User response

Correct the class definition. Methods should be defined as public and must not be void and have no arguments.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1267E

FIELDGETOBJECTFAILED=CWXQY1267E: [{0}] occurred because the specified object [{1}] is not an instance of the class or interface declaring the underlying field [{2}]

Explanation

An error detected while attempting to retrieve a given field.

User response

correct the given class definition or the field name

Parent topic: [CWXQY](#)

CWXQY1268E

INVOKMETHODFAIL=CWXQY1268E: [{0}] occurred while invoking method [{1}] on object [{2}].

Explanation

The invoked method threw an exception.

User response

correct the invoked method in the application model.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1269E

FIELDACCESSFAILED=CWXQY1269E: [{0}] occurred because the field [{1}] is inaccessible.

Explanation

Cannot access the field {1}

User response

Correct the application model.

Parent topic: [CWXQY](#)

CWXQY1270E

DATEWRONGJDBCESCAPE=CWXQY1270E: Date given [{0}] is not in the JDBC date escape format[yyyy-mm-dd].

Explanation

The correct literal constant for date is yyyy-mm-dd.

User response

Correct the literal constant {0}

Parent topic: [CWXQY](#)

CWXQY1271E

TIMEWRONGJDBCESCAPE=CWXQY1271E: Time given [{0}] is not in the JDBC time escape format[hh:mm:ss].

Explanation

The correct literal constant for time is hh:mm:ss.

User response

Correct the literal constant {0}

Parent topic: [CWXQY](#)

CWXQY1272E

TIMESTAMPWRONGJDBCESCAPE=CWXQY1272E: Timestamp given [{0}] is not in the JDBC timestamp escape format[yyyy-mm-dd hh:mm:ss.fffffffff].

Explanation

The correct literal constant for timestamp is yyyy-mm-dd hh:mm:ss.fffffffff.

User response

Correct the literal constant {0}

Parent topic: [CWXQY](#)

CWXQY1282E

SUBSTRWRONGRANGE=CWXQY1282E: The second or third argument of the SUBSTR function is out of range.

Explanation

The sum of the second and third arguments is greater than the length of the first argument.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1283E

NOTNEEDEDPARAMETER=CWXQY1283E: Parameter {0} is not used in the query.

Explanation

The parameter indicated by token {0} is not used in the query.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1285E

OVERFLOWAVG=CWXQY1285E: Counter overflow occurred computing AVG.

Explanation

Overflow occurred while computing AVG aggregate function.

User response

Correct the query statement to avoid the overflow.

Parent topic: [CWXQY](#)

CWXQY1286E

OVERFLOWCOUNT=CWXQY1286E: Counter overflow occurred computing COUNT.

Explanation

Overflow occurred while computing COUNT aggregate function.

User response

Correct the query statement to avoid the overflow.

Parent topic: [CWXQY](#)

CWXQY1287E

TOOMANYPROJTIONITEMS=CWXQY1287E: Exceeds the maximum number of elements [{0}] allowed in Tuple object fail to add the element [{1}].

Explanation

A maximum of {0} identification variables can be used in the query plan which is generated by the query statement.

User response

Reduce the complexity of the query statement.

Parent topic: [CWXQY](#)

CWXQY1288E

FEWPARAMETER=CWXQY1288E: The query uses {1} parameters but only {0} were passed.

Explanation

The number of parameters passed in the query engine is less than expected.

User response

Correct the number of parameter passed in.

Parent topic: [CWXQY](#)

CWXQY1289E

NOPARAMETER=CWXQY1289E: No parameters were passed to a query that required parameters.

Explanation

No parameters were passed to a query that required parameters.

User response

Correct the parameter passed in.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1290E

NOTDEFINEDPARAMETER=CWXQY1290E: Parameter {0} is not defined.

Explanation

The parameter indicated by token {0} is not defined.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1291E

INVALIDPARAMETERTYPE=CWXQY1291E: Parameter {0} passed in is a type of {1} which is not the expected type {2}.

Explanation

Parameter {0} passed in is a type of {1} which is not the expected type {2}.

User response

Correct the parameter passed in.

Parent topic: [CWXQY](#)

CWXQY1292E

SCALARSUBQNODATE=CWXQY1292E: Scalar subquery returned no data.

Explanation

A subquery used with a basic predicate must return a single value. No values were returned at runtime when the subquery was evaluated.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1293E

SCALARSUBQMORECOL=CWXQY1293E: Scalar subquery returns more than one column.

Explanation

The SELECT clause of a subquery is invalid because it specifies multiple columns.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1294E

SCALARSUBQMOREROW=CWXQY1294E: Scalar subquery returned more than one row.

Explanation

A subquery used with a basic predicate must only return a single value. Multiple values were returned at runtime when the subquery was evaluated.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1296E

INVALIDINDEXTYPE=CWXQY1296E: Internal Error. Undefined type [{0}] for index field.

Explanation

Internal error.

User response

Report this problem to IBM service.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1297E

ONECHARACTERONLY=CWXQY1297E: Character can only be compared to Character or String of length 1. {0} has more than one character.

Explanation

Character can only be compared to Character or String of length 1.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1298E

CONFLICTNAME=CWXQY1298E: Duplicate name {0} in select expressions.

Explanation

There is name conflict in selection list, use alias to make sure the output column names are unique.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1299E

INVALIDTOKEN=CWXQY1299E: Invalid token is found in query. {0} in "{1}".

Explanation

The query statement contains an invalid token.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1300E

PARSEERROR=CWXQY1300E: Query parser encountered an error. {0}.

Explanation

Internal error.

User response

Report this problem to IBM service.

Parent topic: [CWXQY](#)

[Terms of Use](#) | [Feedback](#)

CWXQY1301E

INVALIDAS=CWXQY1301E: {0} of an aggregate function is not valid in the context where it it used.

Explanation

The select alias of an aggregate function is not valid in group by clause.

User response

Correct the query statement.

Parent topic: [CWXQY](#)

CWXQY1302E

INVALIDMEMBEROFMISPK=CWXQY1302E: Member operation failed either because a primary key is not contained in the object or the primary key is not identified in the object's metadata.

Explanation

The Member operation requires the object to have a primary key and also identify the primary key.

User response

Correct the object definition.

Parent topic: [CWXQY](#)

CWXQY1304E

INVALIDEQMISPK=CWXQY1304E: Equal operation failed either because a primary key is not contained in the object or the primary key is not identified in the object's metadata.

Explanation

The Equal operation requires the object to have a primary key and also identify the primary key.

User response

Correct the object definition.

Parent topic: [CWXQY](#)

DCPC

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

DCPC0001I

PC_Enabled=DCPC0001I: The Application Placement Controller is now enabled.

DCPC0002I

PC_Disabled=DCPC0002I: The Application Placement Controller is now disabled.

DCPC0003E

PC_EnableErr=DCPC0003E: Application Placement Controller enable error: {0}.

DCPC0005I

PC_PlacementChange=DCPC0005I: Application Placement Controller - placement change; newPlacement={0}.

DCPC0006E

PC_LoopErr=DCPC0006E: Application Placement Controller: unexpected error: {0}.

DCPC0009I

PC_Config1I=DCPC0009I: Application Placement Controller: minTimeBetweenPlacementChanges has been set to {0} min.

DCPC0010I

PC_Config2I=DCPC0010I: Application Placement Controller: serverStateChangeTimeout has been set to {0} min.

DCPC0011I

PC_Config3I=DCPC0011I: Application Placement Controller: approvalTimeout has been set to {0} min.

DCPC0012I

PC_RECOMPUTEI=DCPC0012I: Application Placement Controller: extra placement recomputation has been initiated by user or another system component.

DCPC0013E

PC_BBFindErr=DCPC0013E: Application Placement Controller: unable to find Bulletin Board for {0}. Error message is {1}.

DCPC0014E

PC_BBFErr=DCPC0014E: Application Placement Controller: unable to find Bulletin Board Factory.

DCPC0015E

BB_SubscribeFailed=DCPC0015E: Application Placement Controller: unable to subscribe to topic: cell = {0}, node = {1}, message = {2}.

DCPC0016E

PC_EXCEPTION=DCPC0016E: Application Placement Controller caught unexpected exception. Message = {0}. Stack Trace = {1}.

DCPC0017I

PC_SERVER_MM_KEY=DCPC0017I: Application Placement Controller has set server {0} on node {1} into maintenance mode.

DCPC0018I

PC_SERVER_UNMM_KEY=DCPC0018I: Application Placement Controller has taken server {0} on node {1} out of maintenance mode.

DCPC0019I

PC_SERVER_NO_START_CONFIRMATION_KEY=DCPC0019I: Application Placement Controller has not received confirmation for the start of server {0} on node {1} within the Server operation timeout value specified on the Application Placement Controller configuration. The Application Placement Controller will not attempt to start this server until it receives confirmation of its successful start, but will attempt to start other instances if appropriate. Administrative action is needed to investigate any problems with the failed startup attempt.

DCPC0045I

PC_GATE_CONNECT_SUCCESS=DCPC0045I: The Application Placement Controller has successfully connected to a remote controller {0}.

DCPC0046W

PC_GATE_CONNECT_FAIL=DCPC0046W: The Application Placement Controller has not successfully connected to a remote controller {0}.

DCPC0050E

PC_MINPOLICYErr=DCPC0050E: Application Placement Controller: unable to satisfy minInstances policy for dynamic cluster {0}; minInstances = {1}, allocated {2}.

DCPC0051E

PC_NOSOLUTIONErr=DCPC0051E: Application Placement Controller: unable to find a solution.

DCPC0101E

PC_AP_STATUS_TIMEOUT_KEY=DCPC0101E: Placement plan was not approved on time.

DCPC0102I

PC_AP_STATUS_COMPLETE_KEY=DCPC0102I: Placement completed.

DCPC0103E

PC_AP_STATUS_ABORTED_KEY=DCPC0103E: Placement aborted.

DCPC0104E

PC_StartInstance_Err=DCPC0104E: Application Placement Controller: Failed to launch; DynamicCluster={0}, Node={1}, Member={2} error={3}.

DCPC0105E

PC_StopInstance_Err=DCPC0105E: Application Placement Controller: Failed to launch; DynamicCluster={0}, Node={1}, Member={2} error={3}.

DCPC0106E

PC_TaskSubmit_Err=DCPC0106E: Application Placement Controller: Failed to submit task to Task Controller; error={0}.

DCPC0107E

PC_TaskStatusSubmit_Err=DCPC0107E: Application Placement Controller: Failed to submit task status to Task Controller; error={0}.

DCPC0119I

PC_RealTimeWakeUp=DCPC0119I: The Application Placement Controller has been activated to review a dynamic event.

DCPC0120I

PC_ServerRecommendation=DCPC0120I: The Application Placement Controller recommends server {0} be changed to state {1} on node {2}. Dynamic Cluster {3} operating mode is {4}.

DCPC0121I

PC_RecommendationExpire=DCPC0121I: The Application Placement Controller has submitted a task to the Task Management subsystem that will expire at {0}. The operating mode is {1}. The task ID is {2}.

DCPC0122I

PC_TaskDenied=DCPC0122I: The Application Placement Controller has submitted a task that was denied. The task ID is {0}.

DCPC0123I

PC_TaskApproved=DCPC0123I: The Application Placement Controller has received approval for task ID {0}.

DCPC0124I

PC_TaskExpired=DCPC0124I: The Application Placement Controller has submitted a task that has expired. The task ID is {0}.

DCPC0125I

PC_TaskSource=DCPC0125I: The Application Placement Controller will submit tasks from the active location {0}.

DCPC0126I

PC_TaskComplete=DCPC0126I: The Application Placement Controller has completed a change for task ID {0}. Outstanding changes remaining:{1}. Outstanding tasks remaining:{2}.

DCPC0127W

PC_TaskCompleteErr=DCPC0127W: The Application Placement Controller has completed a change for task ID {0} with errors. Outstanding changes remaining:{1}. Outstanding tasks remaining:{2}.

DCPC0128W

PC_FindBestNoVacancy=DCPC0128W: The Application Placement Controller has been asked to find an alternate location to run server {0} and a replacement location is not available.

DCPC0129I

PC_FindBestVacancy=DCPC0129I: The Application Placement Controller has been asked to find an alternate location to run server {0} and the available replacement location is {1}.

DCPC0130I

PC_FindBestNotNeeded=DCPC0130I: The Application Placement Controller has been asked to find an alternate location to run server {0} and a replacement location is not needed. The server can be stopped or placed into maintenance mode.

DCPC0131I

PC_DCConfigChanged=DCPC0131I: The Application Placement Controller has detected a configuration change for Dynamic Cluster {0}. PropertyName={1} OldValue={2} NewValue={3}

DCPC0132I

PC_DCChangeCommanCall=DCPC0132I: The Application Placement Controller has issued command {0} for task ID {1} on server {2} that will expire at {3}.

DCPC0133I

PC_DCChangeCommanReturn=DCPC0133I: The Application Placement Controller has received status for command {0} for task ID {1} on server {2}. success={3}

DCPC0301I

PC_ChangeGoal=DCPC0301I: The Application Placement Controller detected that service policy goals have been or soon will be breached for these service classes: "{0}". Modify the placement of dynamic cluster instances to maintain desired service level performance. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0302I

PC_ChangeBalance=DCPC0302I: The Application Placement Controller detected an imbalance in service level performance among service classes. Modify the placement of dynamic cluster instances to balance service level performance among the service classes that need additional resources: "{1}", and those that can spare resources: "{0}". Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0303I

PC_ChangeCapacity=DCPC0303I: The Application Placement Controller detected that additional resources can be allocated to these service classes: "{0}". Though none of the service classes are projected to breach service policy goals, the placement of dynamic cluster instances can be modified to improve service level performance. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0304I

PC_ChangeMinMaxPolicy=DCPC0304I: The Application Placement Controller detected that these dynamic clusters "{0}" are not running the required number of dynamic cluster instances as specified in the configuration. Modify the placement of dynamic cluster instances to ensure that the specified minimum and maximum number of instances is met. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0305I

PC_ChangeNodeUtilization=DCPC0305I: The Application Placement Controller detected that the CPU utilization values exceed the specified thresholds on nodes: "{0}". Modify the placement of dynamic cluster instances to transfer workload to other nodes. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0306I

PC_ChangeGoalAut=DCPC0306I: The Application Placement Controller detected that service policy goals have been or soon will be breached for these service classes: "{0}". Modify the placement of dynamic cluster instances to maintain desired service level performance. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

DCPC0307I

PC_ChangeBalanceAut=DCPC0307I: The Application Placement Controller detected an imbalance in service level performance among service classes. Modify the placement of dynamic cluster instances to balance service level performance among the service classes that need additional resources: "{1}", and those that can spare resources: "{0}". Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

DCPC0308I

PC_ChangeCapacityAut=DCPC0308I: The Application Placement Controller detected that additional resources can be allocated to these service classes: "{0}". Though none of the service classes are projected to breach service policy goals, the placement of dynamic cluster instances can be modified to improve service level performance. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

DCPC0309I

PC_ChangeMinMaxPolicyAut=DCPC0309I: The Application Placement Controller detected that these dynamic clusters "{0}" are not running the required number of dynamic cluster instances as specified in the configuration. Modify the placement of dynamic cluster instances to ensure that the specified minimum and maximum number of instances is met. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

DCPC0310I

PC_ChangeNodeUtilizationAut=DCPC0310I: The Application Placement Controller detected that the CPU utilization values exceed the specified thresholds on nodes: "{0}". Modify the placement of dynamic cluster instances to transfer workload to other nodes. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is

automatic.

DCPC0311I

PC_ChangeNodeMemory=DCPC0311I: The Application Placement Controller for detected that memory utilization values exceed thresholds on nodes: "{0}". Modify the placement of dynamic cluster instances to balance memory usage among nodes. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0312I

PC_ChangeNodeMemoryAut=DCPC0312I: The Application Placement Controller detected that memory utilization values exceed thresholds on nodes: "{0}". Modify the placement of dynamic cluster instances to balance memory usage among nodes. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

DCPC0315I

PC_ChangeLazyStartStop=DCPC0315I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation / deactivation needs to be acted upon because of recent activity or inactivity: "{0}". The dynamic cluster operating mode is supervised.

DCPC0316I

PC_ChangeLazyStartStopAut=DCPC0316I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation / deactivation needs to be acted upon because of recent activity or inactivity: "{0}". The dynamic cluster operating mode is automatic.

DCPC0317I

PC_ChangeIsolation=DCPC0317I: The Application Placement Controller detected that these dynamic clusters "{0}" are in violation of a configured isolation policy. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0318I

PC_ChangeIsolationAut=DCPC0318I: The Application Placement Controller detected that these dynamic clusters "{0}" are in violation of a configured isolation policy. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

DCPC0319I

PC_ChangeAllocation=DCPC0319I: The Application Placement Controller detected that instances of dynamic cluster "{0}" are incorrectly running on a node. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0320I

PC_ChangeAllocationAut=DCPC0320I: The Application Placement Controller detected that instances of dynamic cluster "{0}" are incorrectly running on a node. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

DCPC0321I

PC_ChangePerf=DCPC0321I: The Application Placement Controller detected that the system performance can be improved by a reallocation of dynamic cluster instances. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0322I

PC_ChangePerfAut=DCPC0322I: The Application Placement Controller detected that the system performance can be improved by a reallocation of dynamic cluster instances. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

DCPC0323I

PC_ChangeMaxAbove=DCPC0323I: The Application Placement Controller detected that these dynamic clusters "{0}" are violating restrictions associated with collocation of long running jobs and transactional workloads. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

DCPC0324I

PC_ChangeMaxAboveAut=DCPC0324I: The Application Placement Controller detected that these dynamic clusters "{0}" are violating restrictions associated with collocation of long running jobs and transactional workloads. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

DCPC0325I

PC_ChangeLazyStart=DCPC0325I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation needs to be acted upon because of recent activity. "{0}". The dynamic cluster operating mode is supervised.

DCPC0326I

PC_ChangeLazyStartAut=DCPC0326I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation needs to be acted upon because of recent activity. "{0}".

The dynamic cluster operating mode is automatic.

DCPC0327I

PC_ChangeLazyStop=DCPC0327I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation needs to be acted upon because of recent resource shortages. "{0}". The dynamic cluster operating mode is supervised.

DCPC0328I

PC_ChangeLazyStopAut=DCPC0328I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation needs to be acted upon because of recent resource shortages. "{0}". The dynamic cluster operating mode is automatic.

DCPC0329I

PC_ChangeIdleStop=DCPC0329I: The Application Placement Controller detected that the following dynamic cluster configured for idle stop needs to be acted upon. "{0}". The DC operating mode is supervised.

DCPC0330I

PC_ChangeIdleStopAut=DCPC0330I: The Application Placement Controller detected that the following dynamic cluster configured for idle stop needs to be acted upon. "{0}". The dynamic cluster operating mode is automatic.

DCPC0331I

PC_PLACEMAP_DUMP=DCPC0331I: Application Placement Controller {0} is {1}

DCPC0399W

PC_NO_EXPLANATION=DCPC0399W: Application Placement Controller is unable to explain the reason for the placement change. Additional data={0}.

DCPC0400I

LZ_ActivatedDC=DCPC0400I: Application Lazy Start Controller has detected activity for dynamic cluster {0}.

DCPC0401I

LZ_DeactivateDC=DCPC0401I: Application Lazy Start Controller will allow deactivation of dynamic cluster {0} instances if resources are required by other dynamic clusters.

DCPC0402E

LZ_DCStateNotFound=DCPC0402E: The Lazy Start Controller could not locate and update the state of dynamic cluster {0}.

DCPC0403I

LZ_DCMinIdleTime=DCPC0403I: Dynamic cluster {0} can be deactivated if idle time exceeds {1} min.

DCPC0404I

LZ_DCMustBeActive=DCPC0404I: Dynamic cluster {0} can no longer be deactivated due to inactivity.

DCPC0500I

APC_UP=DCPC0500I: Application Placement Controller for Extended Deployment is operational

DCPC0501I

APC_CoreDump=DCPC0501I: The complete set of state for a healthy Application Placement Controller is: {0}

DCPC0502E

APC_CoreDumpErr=DCPC0502E: The complete set of state for an unhealthy Application Placement Controller is: {0}

DCPC0001I

PC_Enabled=DCPC0001I: The Application Placement Controller is now enabled.

Explanation

The Application Placement Controller object is enabled.

User response

No action required.

Parent topic: [DCPC](#)

DCPC0002I

PC_Disabled=DCPC0002I: The Application Placement Controller is now disabled.

Explanation

Application Placement Controller object is disabled.

User response

No action required.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0003E

PC_EnableErr=DCPC0003E: Application Placement Controller enable error: {0}.

Explanation

Application Placement Controller could not be enabled due to an unexpected error.

User response

Report problem to IBM.

Parent topic: [DCPC](#)

DCPC0005I

PC_PlacementChange=DCPC0005I: Application Placement Controller - placement change; newPlacement={0}.

Explanation

To better serve application demand Application Placement Controller proposed a new placement.

User response

No action required.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0006E

PC_LoopErr=DCPC0006E: Application Placement Controller: unexpected error: {0}.

Explanation

Placement Controller experiences an error while executing the control loop.

User response

Report problem to IBM.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0009I

PC_Config1I=DCPC0009I: Application Placement Controller: minTimeBetweenPlacementChanges has been set to {0} min.

Explanation

Placement Controller configuration has been changed by a user.

User response

No action required.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0010I

PC_Config2I=DCPC0010I: Application Placement Controller: serverStateChangeTimeout has been set to {0} min.

Explanation

Placement Controller configuration has been changed by a user.

User response

No action required.

Parent topic: [DCPC](#)

DCPC0011I

PC_Config3I=DCPC0011I: Application Placement Controller: approvalTimeout has been set to {0} min.

Explanation

Placement Controller configuration has been changed by a user.

User response

No action required.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0012I

PC_RECOMPUTEI=DCPC0012I: Application Placement Controller: extra placement recomputation has been initiated by user or another system component.

Explanation

Placement Controller has started an out-of-schedule placement cycle.

User response

No action required.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0013E

PC_BBFindErr=DCPC0013E: Application Placement Controller: unable to find Bulletin Board for {0}. Error message is {1}.

Explanation

Application Placement Controller could not find the Bulletin Board.

User response

Report problem to IBM.

Parent topic: [DCPC](#)

DCPC0014E

PC_BBFErr=DCPC0014E: Application Placement Controller: unable to find Bulletin Board Factory.

Explanation

Application Placement Controller could not find the Bulletin Board Factory.

User response

Report problem to IBM.

Parent topic: [DCPC](#)

DCPC0015E

BB_SubscribeFailed=DCPC0015E: Application Placement Controller: unable to subscribe to topic: cell = {0}, node = {1}, message = {2}.

Explanation

Application Placement Controller could not subscribe to BB topic.

User response

Report problem to IBM.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0016E

PC_EXCEPTION=DCPC0016E: Application Placement Controller caught unexpected exception. Message = {0}.
Stack Trace = {1}.

Explanation

This error message is generated when an exception or an unexpected parameter value occurs within the APC code. It is generated to aid problem diagnosis.

User response

Report problem to IBM.

Parent topic: [DCPC](#)

DCPC0017I

PC_SERVER_MM_KEY=DCPC0017I: Application Placement Controller has set server {0} on node {1} into maintenance mode.

Explanation

The server has been placed into maintenance mode. Typical causes are the server failing to start within the Server operation timeout value specified on the Application Placement Controller configuration.

User response

Investigate and resolve any problems relating to server start and remove the server from maintenance mode when corrected.

Parent topic: [DCPC](#)

DCPC0018I

PC_SERVER_UNMM_KEY=DCPC0018I: Application Placement Controller has taken server {0} on node {1} out of maintenance mode.

Explanation

The Application Placement Controller has detected that the server came up in maintenance mode after it previously failed to start.

User response

No action required.

Parent topic: [DCPC](#)

DCPC0019I

PC_SERVER_NO_START_CONFIRMATION_KEY=DCPC0019I: Application Placement Controller has not received confirmation for the start of server {0} on node {1} within the Server operation timeout value specified on the Application Placement Controller configuration. The Application Placement Controller will not attempt to start this server until it receives confirmation of its successful start, but will attempt to start other instances if appropriate. Administrative action is needed to investigate any problems with the failed startup attempt.

Explanation

The server failed to start or timed out while starting.

User response

Investigate and resolve any problems relating to the failed start attempt.

Parent topic: [DCPC](#)

DCPC0045I

PC_GATE_CONNECT_SUCCESS=DCPC0045I: The Application Placement Controller has successfully connected to a remote controller {0}.

Explanation

Another Application Placement Controller is configured and the local controller has connected to the remote controller.

User response

No action required.

Parent topic: [DCPC](#)

DCPC0046W

PC_GATE_CONNECT_FAIL=DCPC0046W: The Application Placement Controller has not successfully connected to a remote controller {0}.

Explanation

Another Application Placement Controller is configured and the local controller was unable to connect to the remote controller.

User response

Use parameter 0 as shown in the message along with the FFDC logs to debug the problem.

Parent topic: [DCPC](#)

DCPC0050E

PC_MINPOLICYErr=DCPC0050E: Application Placement Controller: unable to satisfy minInstances policy for dynamic cluster {0}; minInstances = {1}, allocated {2}.

Explanation

Application Placement Controller is unable to satisfy minInstances policy for a dynamic cluster due to insufficient memory.

User response

Increase total memory in the node group or lower minInstances policy values.

Parent topic: [DCPC](#)

DCPC0051E

PC_NOSOLUTIONErr=DCPC0051E: Application Placement Controller: unable to find a solution.

Explanation

Application Placement Controller is unable to find a placement that satisfies all memory and CPU speed requirements.

User response

Increase node group capacity.

Parent topic: [DCPC](#)

DCPC0101E

PC_AP_STATUS_TIMEOUT_KEY=DCPC0101E: Placement plan was not approved on time.

Explanation

Placement plan sent for approval was not approved on time. The approval was ignored.

User response

No action is required.

Parent topic: [DCPC](#)

DCPC0102I

PC_AP_STATUS_COMPLETE_KEY=DCPC0102I: Placement completed.

Explanation

Approved placement plan was successfully executed.

User response

No action is required.

Parent topic: [DCPC](#)

DCPC0103E

PC_AP_STATUS_ABORTED_KEY=DCPC0103E: Placement aborted.

Explanation

Approved placement plan was not executed.

User response

No action is required.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0104E

PC_StartInstance_Err=DCPC0104E: Application Placement Controller: Failed to launch; DynamicCluster={0}, Node={1}, Member={2} error={3}.

Explanation

Placement Executer experienced an error while starting a DynamicCluster member on a Node.

User response

The Application Placement Controller was unable to communicate with the node agent on the specified node. Ensure that the node agent process is started and that there are no network connectivity issues.

Parent topic: [DCPC](#)

DCPC0105E

PC_StopInstance_Err=DCPC0105E: Application Placement Controller: Failed to launch; DynamicCluster={0}, Node={1}, Member={2} error={3}.

Explanation

Placement Executer experienced an error while stopping a DynamicCluster member on a Node.

User response

The Application Placement Controller was unable to communicate with the node agent on the specified node. Ensure that the node agent process is started and that there are no network connectivity issues.

Parent topic: [DCPC](#)

DCPC0106E

PC_TaskSubmit_Err=DCPC0106E: Application Placement Controller: Failed to submit task to Task Controller; error={0}.

Explanation

Placement Executer experienced an error while submitting task.

User response

The Application Placement Controller was unable to communicate with the deployment manager. Ensure that the deployment manager process is started and that there are no network connectivity issues.

Parent topic: [DCPC](#)

DCPC0107E

PC_TaskStatusSubmit_Err=DCPC0107E: Application Placement Controller: Failed to submit task status to Task Controller; error={0}.

Explanation

Placement Executer experienced an error while submitting task.

User response

The Application Placement Controller was unable to communicate with the deployment manager. Ensure that the deployment manager process is started and that there are no network connectivity issues.

Parent topic: [DCPC](#)

DCPC0119I

PC_RealTimeWakeUp=DCPC0119I: The Application Placement Controller has been activated to review a dynamic event.

Explanation

A dynamic event has occurred that requires the Application Placement Controller to review the event and potentially reconfigure the system.

User response

No user action is required at this time, additional messages will be logged if a change is required.

Parent topic: [DCPC](#)

DCPC0120I

PC_ServerRecommendation=DCPC0120I: The Application Placement Controller recommends server {0} be changed to state {1} on node {2}. Dynamic Cluster {3} operating mode is {4}.

Explanation

Application Placement Controller has determined a server instance needs to be stopped or started. See message DCPC0121I for additional information.

User response

If the Dynamic Cluster operating mode is supervised, use the Admin Console Task Management panel to accept or deny the recommendation. If the Dynamic Cluster operating mode is automatic, no user action is required.

Parent topic: [DCPC](#)

DCPC0121I

PC_RecomendationExpire=DCPC0121I: The Application Placement Controller has submitted a task to the Task Management subsystem that will expire at {0}. The operating mode is {1}. The task ID is {2}.

Explanation

Application Placement Controller has submitted a task to the Task Management subsystem. See message DCPC0120I for additional information.

User response

If the Dynamic Cluster operating mode is supervised, use the Admin Console Task Management panel to accept the recommendation prior to the expiration time. If the Dynamic Cluster operating mode is automatic, no user action is required.

Parent topic: [DCPC](#)

DCPC0122I

PC_TaskDenied=DCPC0122I: The Application Placement Controller has submitted a task that was denied. The task ID is {0}.

Explanation

Application Placement Controller has submitted a task to the Task Management subsystem and the task was denied. If there is a reoccurrence of the conditions that originally caused the task to be generated, a new task will be generated.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

DCPC0123I

PC_TaskApproved=DCPC0123I: The Application Placement Controller has received approval for task ID {0}.

Explanation

Application Placement Controller has received user approval for previously submitted task. The task will be executed.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0124I

PC_TaskExpired=DCPC0124I: The Application Placement Controller has submitted a task that has expired. The task ID is {0}.

Explanation

Application Placement Controller has submitted a task to the Task Management subsystem and the task has expired. If there is a reoccurrence of the conditions that originally caused the task to be generated, a new task will be generated.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0125I

PC_TaskSource=DCPC0125I: The Application Placement Controller will submit tasks from the active location {0}.

Explanation

Application Placement Controller will submit tasks to the Task Management subsystem from the indicated cell/node/server.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

DCPC0126I

PC_TaskComplete=DCPC0126I: The Application Placement Controller has completed a change for task ID {0}. Outstanding changes remaining:{1}. Outstanding tasks remaining:{2}.

Explanation

Application Placement Controller has been notified that a change has completed.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

DCPC0127W

PC_TaskCompleteErr=DCPC0127W: The Application Placement Controller has completed a change for task ID {0} with errors. Outstanding changes remaining:{1}. Outstanding tasks remaining:{2}.

Explanation

Application Placement Controller has been notified that a change has completed with errors.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

DCPC0128W

PC_FindBestNoVacancy=DCPC0128W: The Application Placement Controller has been asked to find an alternate location to run server {0} and a replacement location is not available.

Explanation

This is usually caused by a user or system component requesting the shutdown or setting of maintenance mode for a server or node.

User response

Use the Admin Console to modify the placement of dynamic cluster instances to maintain desired service level performance.

Parent topic: [DCPC](#)

DCPC0129I

PC_FindBestVacancy=DCPC0129I: The Application Placement Controller has been asked to find an alternate location to run server {0} and the available replacement location is {1}.

Explanation

This is usually caused by a user or system component requesting the shutdown or setting of maintenance mode for a server or node.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

DCPC0130I

PC_FindBestNotNeeded=DCPC0130I: The Application Placement Controller has been asked to find an alternate location to run server {0} and a replacement location is not needed. The server can be stopped or placed into maintenance mode.

Explanation

This is usually caused by a user or system component requesting the shutdown or setting of maintenance mode for a server or node.

User response

Use the Admin Console to modify the placement of dynamic cluster instances to maintain desired service level performance.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0131I

PC_DCConfigChanged=DCPC0131I: The Application Placement Controller has detected a configuration change for Dynamic Cluster {0}. PropertyName={1} OldValue={2} NewValue={3}

Explanation

This is usually caused by a user or system component requesting a configuration change for a dynamic cluster.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

DCPC0132I

PC_DCChangeCommanCall=DCPC0132I: The Application Placement Controller has issued command {0} for task ID {1} on server {2} that will expire at {3}.

Explanation

Application Placement Controller is attempting to start or stop a server instance.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0133I

PC_DCChangeCommanReturn=DCPC0133I: The Application Placement Controller has received status for command {0} for task ID {1} on server {2}. success={3}

Explanation

Application Placement Controller is attempting to start or stop a server instance.

User response

No user action is required at this time.

Parent topic: [DCPC](#)

DCPC0301I

PC_ChangeGoal=DCPC0301I: The Application Placement Controller detected that service policy goals have been or soon will be breached for these service classes: "{0}". Modify the placement of dynamic cluster instances to maintain desired service level performance. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller has requested placement changes in order to maintain service levels for the specified service classes.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

DCPC0302I

PC_ChangeBalance=DCPC0302I: The Application Placement Controller detected an imbalance in service level performance among service classes. Modify the placement of dynamic cluster instances to balance service level performance among the service classes that need additional resources: "{1}", and those that can spare resources: "{0}". Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller has determined that there is a more optimal placement of dynamic cluster instances and has requested a placement change.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0303I

PC_ChangeCapacity=DCPC0303I: The Application Placement Controller detected that additional resources can be allocated to these service classes: "{0}". Though none of the service classes are projected to breach service policy goals, the placement of dynamic cluster instances can be modified to improve service level performance. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller has determined that there is a more optimal placement of dynamic cluster instances and has requested a placement change.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0304I

PC_ChangeMinMaxPolicy=DCPC0304I: The Application Placement Controller detected that these dynamic clusters "{0}" are not running the required number of dynamic cluster instances as specified in the configuration. Modify the placement of dynamic cluster instances to ensure that the specified minimum and maximum number of instances is met. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

The minimum and maximum dynamic cluster instances is specified in the dynamic cluster configuration properties. The Application Placement Controller compares the number of active instances against the configured values and requests placement changes if necessary.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

DCPC0305I

PC_ChangeNodeUtilization=DCPC0305I: The Application Placement Controller detected that the CPU utilization values exceed the specified thresholds on nodes: "{0}". Modify the placement of dynamic cluster instances to transfer workload to other nodes. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

Maximum CPU utilization is specified in the Autonomic Request Flow Manager configuration. The Application Placement Controller monitors the CPU utilization per node and requests a placement change when the specified threshold is breached.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0306I

PC_ChangeGoalAut=DCPC0306I: The Application Placement Controller detected that service policy goals have been or soon will be breached for these service classes: "{0}". Modify the placement of dynamic cluster instances to maintain desired service level performance. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller has requested placement changes in order to maintain service levels for the specified service classes.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0307I

PC_ChangeBalanceAut=DCPC0307I: The Application Placement Controller detected an imbalance in service level performance among service classes. Modify the placement of dynamic cluster instances to balance service level performance among the service classes that need additional resources: "{1}", and those that can spare resources: "{0}". Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller has determined that there is a more optimal placement of dynamic cluster instances and has requested a placement change.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0308I

PC_ChangeCapacityAut=DCPC0308I: The Application Placement Controller detected that additional resources can be allocated to these service classes: "{0}". Though none of the service classes are projected to breach service policy goals, the placement of dynamic cluster instances can be modified to improve service level performance. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller has determined that there is a more optimal placement of dynamic cluster instances and has requested a placement change.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0309I

PC_ChangeMinMaxPolicyAut=DCPC0309I: The Application Placement Controller detected that these dynamic clusters "{0}" are not running the required number of dynamic cluster instances as specified in the configuration. Modify the placement of dynamic cluster instances to ensure that the specified minimum and maximum number of instances is met. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

The minimum and maximum dynamic cluster instances is specified in the dynamic cluster configuration properties. The Application Placement Controller compares the number of active instances against the configured values and requests placement changes if necessary.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

DCPC0310I

PC_ChangeNodeUtilizationAut=DCPC0310I: The Application Placement Controller detected that the CPU utilization values exceed the specified thresholds on nodes: "{0}". Modify the placement of dynamic cluster instances to transfer workload to other nodes. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

Maximum CPU utilization is specified in the Autonomic Request Flow Manager configuration. The Application Placement Controller monitors the CPU utilization per node and requests a placement change when the specified threshold is breached.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0311I

PC_ChangeNodeMemory=DCPC0311I: The Application Placement Controller for detected that memory utilization values exceed thresholds on nodes: "{0}". Modify the placement of dynamic cluster instances to balance memory usage among nodes. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller senses and monitors total memory utilization and memory capacity on a node. If the sum of memory consumed by all servers on a node exceeds the capacity of a node, application placement change is requested.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0312I

PC_ChangeNodeMemoryAut=DCPC0312I: The Application Placement Controller detected that memory utilization values exceed thresholds on nodes: "{0}". Modify the placement of dynamic cluster instances to balance memory usage among nodes. Review the strategy used to automatically modify the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller senses and monitors total memory utilization and memory capacity on a node. If the sum of memory consumed by all servers on a node exceeds the capacity of a node, application placement change is requested.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0315I

PC_ChangeLazyStartStop=DCPC0315I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation / deactivation needs to be acted upon because of recent activity or inactivity: "{0}". The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller receives dynamic cluster activity information from the On Demand Router. That information has changed such that activation / deactivation is required.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

DCPC0316I

PC_ChangeLazyStartStopAut=DCPC0316I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation / deactivation needs to be acted upon because of recent activity or inactivity: "{0}". The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller detected that the following dynamic cluster configured for lazy activation / deactivation needs to be acted upon because of recent activity or inactivity: "{0}".

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

DCPC0317I

PC_ChangeIsolation=DCPC0317I: The Application Placement Controller detected that these dynamic clusters "{0}" are in violation of a configured isolation policy. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller detected that these dynamic clusters are in violation of a configured isolation policy. Review the strategy for modifying the placement of dynamic cluster instances in the action plan.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0318I

PC_ChangeIsolationAut=DCPC0318I: The Application Placement Controller detected that these dynamic clusters "{0}" are in violation of a configured isolation policy. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller detected that these dynamic clusters are in violation of a configured isolation policy. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0319I

PC_ChangeAllocation=DCPC0319I: The Application Placement Controller detected that instances of dynamic cluster "{0}" are incorrectly running on a node. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller detected that instances of dynamic cluster are incorrectly running on a node. Review the strategy for modifying the placement of dynamic cluster instances in the action plan.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

DCPC0320I

PC_ChangeAllocationAut=DCPC0320I: The Application Placement Controller detected that instances of dynamic cluster "{0}" are incorrectly running on a node. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller detected that instances of dynamic cluster are incorrectly running on a node. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0321I

PC_ChangePerf=DCPC0321I: The Application Placement Controller detected that the system performance can be improved by a reallocation of dynamic cluster instances. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller detected that the system performance can be improved by a reallocation of dynamic cluster instances. Review the strategy for modifying the placement of dynamic cluster instances in the action plan.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0322I

PC_ChangePerfAut=DCPC0322I: The Application Placement Controller detected that the system performance can be improved by a reallocation of dynamic cluster instances. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller detected that the system performance can be improved by a reallocation of dynamic cluster instances. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0323I

PC_ChangeMaxAbove=DCPC0323I: The Application Placement Controller detected that these dynamic clusters "{0}" are violating restrictions associated with collocation of long running jobs and transactional workloads. Review the strategy for modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller detected that these dynamic clusters "{0}" are violating restrictions associated with collocation of long running jobs and transactional workloads. Review the strategy for modifying the placement of dynamic cluster instances in the action plan.

User response

Review the action plan provided by the system and approve the proposed placement changes.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0324I

PC_ChangeMaxAboveAut=DCPC0324I: The Application Placement Controller detected that these dynamic clusters "{0}" are violating restrictions associated with collocation of long running jobs and transactional workloads. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan. The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller detected that these dynamic clusters "{0}" are violating restrictions associated with collocation of long running jobs and transactional workloads. Review the strategy for automatically modifying the placement of dynamic cluster instances in the action plan.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0325I

PC_ChangeLazyStart=DCPC0325I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation needs to be acted upon because of recent activity. "{0}". The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller receives dynamic cluster activity information from the On Demand Router. That information has changed such that activation is required.

User response

Since the Dynamic Cluster operating mode is supervised, use the Admin Console Task Management panel to accept or deny the recommendation.

Parent topic: [DCPC](#)

DCPC0326I

PC_ChangeLazyStartAut=DCPC0326I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation needs to be acted upon because of recent activity. "{0}". The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller receives dynamic cluster activity information from the On Demand Router. That information has changed such that activation is required.

User response

Since the Dynamic Cluster operating mode is automatic, no user action is required.

Parent topic: [DCPC](#)

DCPC0327I

PC_ChangeLazyStop=DCPC0327I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation needs to be acted upon because of recent resource shortages. "{0}". The dynamic cluster operating mode is supervised.

Explanation

The Application Placement Controller receives dynamic cluster activity information from the On Demand Router. That information has changed such that deactivation is required.

User response

Since the Dynamic Cluster operating mode is supervised, use the Admin Console Task Management panel to accept or deny the recommendation.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0328I

PC_ChangeLazyStopAut=DCPC0328I: The Application Placement Controller detected that the following dynamic cluster configured for lazy activation needs to be acted upon because of recent resource shortages. "{0}". The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller receives dynamic cluster activity information from the On Demand Router. That information has changed such that deactivation is required.

User response

Since the Dynamic Cluster operating mode is automatic, no user action is required.

Parent topic: [DCPC](#)

DCPC0329I

PC_ChangeIdleStop=DCPC0329I: The Application Placement Controller detected that the following dynamic cluster configured for idle stop needs to be acted upon. "{0}". The DC operating mode is supervised.

Explanation

The Application Placement Controller receives dynamic cluster activity information from the On Demand Router. That information has changed such that deactivation is required.

User response

Since the Dynamic Cluster operating mode is supervised, use the Admin Console Task Management panel to accept or deny the recommendation.

Parent topic: [DCPC](#)

DCPC0330I

PC_ChangeIdleStopAut=DCPC0330I: The Application Placement Controller detected that the following dynamic cluster configured for idle stop needs to be acted upon. "{0}". The dynamic cluster operating mode is automatic.

Explanation

The Application Placement Controller receives dynamic cluster activity information from the On Demand Router. That information has changed such that deactivation is required.

User response

Since the Dynamic Cluster operating mode is automatic, no user action is required.

Parent topic: [DCPC](#)

DCPC0331I

PC_PLACEMAP_DUMP=DCPC0331I: Application Placement Controller {0} is {1}

Explanation

The Application Placement Controller has ran and information about the input and output has been logged.

User response

None. The Application Placement Controller has ran and information about the input and output has been logged.

Parent topic: [DCPC](#)

DCPC0399W

PC_NO_EXPLANATION=DCPC0399W: Application Placement Controller is unable to explain the reason for the placement change. Additional data={0}.

Explanation

This warning message is generated when an unexplainable condition caused a placement change.

User response

None. A placement change is pending or in progress.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0400I

LZ_ActivateDC=DCPC0400I: Application Lazy Start Controller has detected activity for dynamic cluster {0}.

Explanation

Application Lazy Start Controller detected incoming requests for the dynamic cluster and will start at least one server instance of this dynamic cluster.

User response

No action is required, placement changes have been initiated.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0401I

LZ_DeactivateDC=DCPC0401I: Application Lazy Start Controller will allow deactivation of dynamic cluster {0} instances if resources are required by other dynamic clusters.

Explanation

Application Lazy Start Controller detected that this dynamic cluster has been idle for the specified wait period and will allow all its server instances to be stopped if resources consumed by them are needed by other dynamic clusters.

User response

None.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0402E

LZ_DCStateNotFound=DCPC0402E: The Lazy Start Controller could not locate and update the state of dynamic cluster {0}.

Explanation

The Lazy Start Controller attempted to update the state of the dynamic cluster but had no prior representation of that dynamic cluster.

User response

None.

Parent topic: [DCPC](#)

DCPC0403I

LZ_DCMIdleTime=DCPC0403I: Dynamic cluster {0} can be deactivated if idle time exceeds {1} min.

Explanation

Application Lazy Start Controller will permit deactivation of the dynamic cluster if its idle time exceeds the specified time.

User response

None.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0404I

LZ_DCMustBeActive=DCPC0404I: Dynamic cluster {0} can no longer be deactivated due to inactivity.

Explanation

Application Lazy Start Controller is not permitted to deactivate this dynamic cluster in the future.

User response

None.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0500I

APC_UP=DCPC0500I: Application Placement Controller for Extended Deployment is operational

Explanation

an informational message that indicate overall subsystem status

User response

None.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0501I

APC_CoreDump=DCPC0501I: The complete set of state for a healthy Application Placement Controller is: {0}

Explanation

A detailed dump of all Application Placement Controller state.

User response

None.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCPC0502E

APC_CoreDumpErr=DCPC0502E: The complete set of state for an unhealthy Application Placement Controller is: {0}

Explanation

A detailed dump of all Application Placement Controller state.

User response

First review task management for potential remedies, otherwise contact IBM support.

Parent topic: [DCPC](#)

[Terms of Use](#) | [Feedback](#)

DCWM

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

DCWM0000I

INFO_SUBCOMPONENT_NAME=DCWM0000I: Dynamic Cluster Validation

DCWM0001E

DCWM_UNKNOWN_ERROR=DCWM0001E: Error occurred during dynamic cluster validation. Exception was {0}.

DCWM0002E

ERROR_DYNAMIC_CLUSTER_NAME_REQUIRED=DCWM0002E: The name of a dynamic cluster in {0} is absent.

DCWM0003E

ERROR_DYNAMIC_CLUSTER_NAME_INVALID_CHARS=DCWM0003E: The name of the dynamic cluster {0} contains invalid characters.

DCWM0004E

ERROR_DYNAMIC_CLUSTER_MININSTANCES_INVALID=DCWM0004E: The number of minimum instances for dynamic cluster {0} is invalid.

DCWM0005E

ERROR_DYNAMIC_CLUSTER_MAXINSTANCES_INVALID=DCWM0005E: The number of maximum instances for dynamic cluster {0} is invalid.

DCWM0006E

ERROR_DYNAMIC_CLUSTER_MIN_GREATER_THAN_MAX=DCWM0006E: The number of minimum instances is greater than the number of maximum instances for dynamic cluster {0}.

DCWM0007E

ERROR_DYNAMIC_CLUSTER_OPMODE_REQUIRED=DCWM0007E: The operational mode for dynamic cluster {0} is absent.

DCWM0008E

ERROR_SERVER_CLUSTER_DOES_NOT_EXIST=DCWM0008E: The corresponding server cluster does not exist for dynamic cluster {0}.

DCWM0009E

ERROR_DYNAMIC_CLUSTER_NODEGROUP_REQUIRED=DCWM0009E: The node group for dynamic cluster {0} is absent.

DCWM0010E

ERROR_DYNAMIC_CLUSTER_TEMPLATE_REQUIRED=DCWM0010E: The server template for dynamic cluster {0} is absent.

DCWM0011E

ERROR_DYNAMIC_CLUSTER_SERVERINACTIVITYTIME_INVALID=DCWM0011E: The server inactivity time for dynamic cluster {0} is invalid.

DCWM0012E

ERROR_DYNAMIC_CLUSTER_VERTICALSTACKINGINSTANCES_INVALID=DCWM0012E: The number of vertical stacking instances for dynamic cluster {0} is invalid.

DCWM0013E

ERROR_DYNAMIC_CLUSTER_VERTICALSTACKING_GREATER_THAN_MAX=DCWM0013E: The number of vertical stacking instances is greater than the number of maximum instances for dynamic cluster {0}.

DCWM0000I

INFO_SUBCOMPONENT_NAME=DCWM0000I: Dynamic Cluster Validation

Explanation

No additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

[Terms of Use](#) | [Feedback](#)

DCWM0001E

DCWM_UNKNOWN_ERROR=DCWM0001E: Error occurred during dynamic cluster validation. Exception was {0}.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

[Terms of Use](#) | [Feedback](#)

DCWM0002E

ERROR_DYNAMIC_CLUSTER_NAME_REQUIRED=DCWM0002E: The name of a dynamic cluster in {0} is absent.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

[Terms of Use](#) | [Feedback](#)

DCWM0003E

ERROR_DYNAMIC_CLUSTER_NAME_INVALID_CHARS=DCWM0003E: The name of the dynamic cluster {0} contains invalid characters.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0004E

ERROR_DYNAMIC_CLUSTER_MININSTANCES_INVALID=DCWM0004E: The number of minimum instances for dynamic cluster {0} is invalid.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0005E

ERROR_DYNAMIC_CLUSTER_MAXINSTANCES_INVALID=DCWM0005E: The number of maximum instances for dynamic cluster {0} is invalid.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0006E

ERROR_DYNAMIC_CLUSTER_MIN_GREATER_THAN_MAX=DCWM0006E: The number of minimum instances is greater than the number of maximum instances for dynamic cluster {0}.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0007E

ERROR_DYNAMIC_CLUSTER_OPMODE_REQUIRED=DCWM0007E: The operational mode for dynamic cluster {0} is absent.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0008E

ERROR_SERVER_CLUSTER_DOES_NOT_EXIST=DCWM0008E: The corresponding server cluster does not exist for dynamic cluster {0}.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0009E

ERROR_DYNAMIC_CLUSTER_NODEGROUP_REQUIRED=DCWM0009E: The node group for dynamic cluster {0} is absent.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0010E

ERROR_DYNAMIC_CLUSTER_TEMPLATE_REQUIRED=DCWM0010E: The server template for dynamic cluster {0} is absent.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0011E

ERROR_DYNAMIC_CLUSTER_SERVERINACTIVITYTIME_INVALID=DCWM0011E: The server inactivity time for dynamic cluster {0} is invalid.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0012E

ERROR_DYNAMIC_CLUSTER_VERTICALSTACKINGINSTANCES_INVALID=DCWM0012E: The number of vertical stacking instances for dynamic cluster {0} is invalid.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWM0013E

ERROR_DYNAMIC_CLUSTER_VERTICALSTACKING_GREATER_THAN_MAX=DCWM0013E: The number of vertical stacking instances is greater than the number of maximum instances for dynamic cluster {0}.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [DCWM](#)

DCWT

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

DWCT0008E

DWCT_MCRPAppNotFound=DWCT0008E: No application has been deployed on the target cluster.

DWCT0009I

DWCT_MCRPDefault=DWCT0009I: Cross cluster routing policy has not been defined for the on demand router.

DWCT0010E

DWCT_MCRPConfigErr=DWCT0010E: Cross cluster routing policy configuration cell or generic cluster {0} is incorrect.

DWCT0008E

DWCT_MCRPAppNotFound=DWCT0008E: No application has been deployed on the target cluster.

Explanation

Configured multi-cluster routing policy for an application, but the application has not been deployed on the cluster.

User response

Check your on demand router customization and deploy the appropriate application.

Parent topic: [DCWT](#)

DWCT0009I

DWCT_MCRPDefault=DWCT0009I: Cross cluster routing policy has not been defined for the on demand router.

Explanation

No multi-cluster routing policy for the on demand router exists.

User response

No action.

Parent topic: [DCWT](#)

[Terms of Use](#) | [Feedback](#)

DWCT0010E

DWCT_MCRPConfigErr=DWCT0010E: Cross cluster routing policy configuration cell or generic cluster {0} is incorrect.

Explanation

The configured cell and/or generic server cluster is not correct.

User response

Correct the configuration in the on demand router and continue.

Parent topic: [DCWT](#)

DWLM

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[DWLM0001E](#)

DWLM_CM_ERR=DWLM0001E: Error getting cluster members for; clusterMembername={0}.

[DWLM0002E](#)

DWLM_PMI_ERR=DWLM0002E: Exception while configuring PMI for cluster, clusterName={0}. Exception: {1}

[DWLM0003E](#)

DWLM_APMI_ERR=DWLM0003E: Error de-registering for async PMI, serverPath={0}. Exception: {1}

[DWLM0004E](#)

DWLM_GET_CLSTRS_ERR=DWLM0004E: Error retrieving clusters from the the cluster MBean.

[DWLM0005E](#)

DWLM_GET_CLSTR_ERR=DWLM0005E: Error retrieving a cluster from the the cluster MBean; clusterName={0}.

[DWLM0006E](#)

DWLM_GET_CLSTR_MGR_ERR=DWLM0006E: Error finding Cluster Manager MBean.

[DWLM0007E](#)

DWLM_INV_ATT_ERR=DWLM0007E: Invalid Attribute Exception while adding clusters to cluster balance bean.

[DWLM0008E](#)

DWLM_ENBL_NT_ERR=DWLM0008E: Error while enabling notifications.

[DWLM0009E](#)

DWLM_APMI2_ERR=DWLM0009E: Error registering for async PMI, serverPath={0}. Exception: {1}

[DWLM0010E](#)

DWLM_WLMC_RGSTR_ERR=DWLM0010E: Error registering as a WLM controller

[DWLM0011E](#)

DWLM_SET_WGHT_ERR=DWLM0011E: Error while setting cluster member weight

[DWLM0012E](#)

DWLM_WCCM_ERROR=DWLM0012E: Exception getting Top Level WCCM Object from Repository with URI={0}. Exception = {1}

[DWLM0013E](#)

DWLM_ADMCL_ERROR=DWLM0013E: Error getting AdminClient for node= {0}. Exception = {1}

[DWLM0014E](#)

DWLM_MAL_ERROR=DWLM0014E: Malformed exception while creating ObjectName with queryString = {0}. Exception = {1}

[DWLM0015E](#)

DWLM_CON_ERROR=DWLM0015E: Connector exception while querying Object Name. AdminClient connector properties: {0}. Exception = {1}

[DWLM0016E](#)

DWLM_CONS_THROW=DWLM0016E: Caught Throwable while notifying consumer = {0}. Throwable = {1}

[DWLM0017E](#)

DWLM_DISABLE_GROUP_EXCEPTION=DWLM0017E: Caught Exception while disabling group. Exception = {0}

[DWLM0018E](#)

DWLM_ENABLE_GROUP_EXCEPTION=DWLM0018E: Caught Exception while enabling group. Exception = {0}

[DWLM0019E](#)

DWLM_NO_PMI_DATA=DWLM0019E: PMI monitored data not supplied during DWLM control cycle: node={0} server= {1}

[DWLM0020E](#)

DWLM_FIND_CLUSTER_ERROR=DWLM0020E: Exception while collecting cluster information from ODC. Exception = {0}

[DWLM0021E](#)

DWLM_GET_CLUSTER_MODE_ERROR=DWLM0021E: Exception while getting cluster mode property from ODC for cluster {0}. Exception = {1}

[DWLM0022E](#)

DWLM_LOCATE_ACTIVE_CTLR_ERROR=DWLM0022E: Exception while invoking the getActiveServerName MBean operation for cluster {0}. Exception = {1}

[DWLM0023E](#)

DWLM_IS_ENABLED_ERROR=DWLM0023E: Exception while invoking the isEnabled MBean operation for cluster {0}. Exception = {1}

[DWLM0024E](#)

DWLM_GET_NUMBER_OF_SERVERS_ERROR=DWLM0024E: Exception while querying for the number of servers in cluster {0}. Exception = {1}

DWLM0001E

DWLM_CM_ERR=DWLM0001E: Error getting cluster members for; clusterMembername={0}.

Explanation

DWLM Controller not load the cluster members from the repository.

User response

Report problem to IBM.

Parent topic: [DWLM](#)

DWLM0002E

DWLM_PMI_ERR=DWLM0002E: Exception while configuring PMI for cluster, clusterName={0}. Exception: {1}

Explanation

There was exception will trying to configure PMI.

User response

Report problem to IBM.

Parent topic: [DWLM](#)

DWLM0003E

DWLM_APMI_ERR=DWLM0003E: Error de-registering for async PMI, serverPath={0}. Exception: {1}

Explanation

While trying to de-register a server for async pmi an exception was encountered.

User response

Report problem to IBM.

Parent topic: [DWLM](#)

DWLM0004E

DWLM_GET_CLSTRS_ERR=DWLM0004E: Error retrieving clusters from the the cluster MBean.

Explanation

There was an exception thrown while invoking retrieveClusters method on the Cluster Manager MBean.

User response

Report problem to IBM.

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0005E

DWLM_GET_CLSTR_ERR=DWLM0005E: Error retrieving a cluster from the the cluster MBean; clusterName={0}.

Explanation

There was an exception thrown while invoking retrieveCluster method on the Cluster Manager MBean.

User response

Report problem to IBM.

Parent topic: [DWLM](#)

DWLM0006E

DWLM_GET_CLSTR_MGR_ERR=DWLM0006E: Error finding Cluster Manager MBean.

Explanation

While querying for the Cluster Manager MBean an exception was encountered.

User response

Report problem to IBM.

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0007E

DWLM_INV_ATT_ERR=DWLM0007E: Invalid Attribute Exception while adding clusters to cluster balance bean.

Explanation

During initialization, while adding a cluster to the cluster balance bean an invalid attribute exception was thrown.

User response

Report problem to IBM.

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0008E

DWLM_ENBL_NT_ERR=DWLM0008E: Error while enabling notifications.

Explanation

An error was encountered while enabling notifications

User response

Report problem to IBM.

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0009E

DWLM_APMI2_ERR=DWLM0009E: Error registering for async PMI, serverPath={0}. Exception: {1}

Explanation

While trying to register a server for async pmi an exception was encountered.

User response

Report problem to IBM.

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0010E

DWLM_WLMC_RGSTR_ERR=DWLM0010E: Error registering as a WLM controller

Explanation

An Error was encountered while registering as a WLM Controller.

User response

Report problem to IBM.

Parent topic: [DWLM](#)

DWLM0011E

DWLM_SET_WGHT_ERR=DWLM0011E: Error while setting cluster member weight

Explanation

An Error was encountered setting a cluster member weight

User response

Report problem to IBM.

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0012E

DWLM_WCCM_ERROR=DWLM0012E: Exception getting Top Level WCCM Object from Repository with URI={0}.
Exception = {1}

Explanation

Exception getting Top Level WCCM Object from Repository

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0013E

DWLM_ADMCL_ERROR=DWLM0013E: Error getting AdminClient for node= {0}. Exception = {1}

Explanation

Error getting AdminClient for node= {0}. Exception = {1}

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0014E

DWLM_MAL_ERROR=DWLM0014E: Malformed exception while creating ObjectName with queryString = {0}.
Exception = {1}

Explanation

Malformed exception while creating ObjectName with queryString = {0}. Exception = {1}

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0015E

DWLM_CON_ERROR=DWLM0015E: Connector exception while querying Object Name. AdminClient connector properties: {0}. Exception = {1}

Explanation

Caught an unexpected Connector exception while querying Object Name.

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0016E

DWLM_CONS_THROW=DWLM0016E: Caught Throwable while notifying consumer = {0}. Throwable = {1}

Explanation

Caught Throwable while notifying consumer = {0}. Throwable = {1}

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0017E

DWLM_DISABLE_GROUP_EXCEPTION=DWLM0017E: Caught Exception while disabling group. Exception = {0}

Explanation

Caught Exception while disabling group. Exception = {0}

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0018E

DWLM_ENABLE_GROUP_EXCEPTION=DWLM0018E: Caught Exception while enabling group. Exception = {0}

Explanation

Caught Exception while enabling group. Exception = {0}

User response

Report problem To IBM

Parent topic: [DWLM](#)

DWLM0019E

DWLM_NO_PMI_DATA=DWLM0019E: PMI monitored data not supplied during DWLM control cycle: node= {0}
server= {1}

Explanation

PMI monitored data not supplied during DWLM control cycle: node= {0} server= {1}

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0020E

DWLM_FIND_CLUSTER_ERROR=DWLM0020E: Exception while collecting cluster information from ODC. Exception = {0}

Explanation

Caught an unexpected exception while collecting cluster information from ODC.

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0021E

DWLM_GET_CLUSTER_MODE_ERROR=DWLM0021E: Exception while getting cluster mode property from ODC for cluster {0}. Exception = {1}

Explanation

Caught an unexpected exception while collecting the cluster mode property for the specified cluster.

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0022E

DWLM_LOCATE_ACTIVE_CTLR_ERROR=DWLM0022E: Exception while invoking the getActiveServerName MBean operation for cluster {0}. Exception = {1}

Explanation

Caught an unexpected exception while invoking the getActiveServerName operation on the HAManagedItems MBean.

User response

Report problem To IBM

Parent topic: [DWLM](#)

DWLM0023E

DWLM_IS_ENABLED_ERROR=DWLM0023E: Exception while invoking the isEnabled MBean operation for cluster {0}. Exception = {1}

Explanation

Caught an unexpected exception while invoking the isEnabled operation on the DWLM MBean.

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

DWLM0024E

DWLM_GET_NUMBER_OF_SERVERS_ERROR=DWLM0024E: Exception while querying for the number of servers in cluster {0}. Exception = {1}

Explanation

Caught an unexpected exception while collecting cluster member information from ODC.

User response

Report problem To IBM

Parent topic: [DWLM](#)

[Terms of Use](#) | [Feedback](#)

OCSP

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

OCSP0001E

OCSP_NodeIn2Grps=OCSP0001E: Node {0} is in two node groups, {1} and {2}, and will therefore not be used.

OCSP0002E

OCSP_Exception=OCSP0002E: Exception: {0}.

OCSP0003E

OCSP_UnknownNode=OCSP0003E: Node {0} was not found. Invalid custom property value of "node.name" in endpoint of generic server cluster {1}.

OCSP0004E

OCSP_JMS_NoDestDefined=OCSP0004E: No matching destination defined in cell for resource reference binding {0} with destination {1} and Bus {2}

OCSP0005E

OCSP_JMS_NoDestDefinedModule=OCSP0005E: No matching destination defined in cell for resource reference binding {0} defined by module {1}

OCSP0006I

OCSP_JMS_DESTSWITCH=OCSP0006I: Destination {0} is being updated to be part of work class {1} in application {2}. Earlier it was part of default workclass of {3}

OCSP0007I

OCSP_JMS_DestAlreadyMember=OCSP0007I: Destination {0} is already part of default work class {1} in application {2}.

OCSP0001E

OCSP_NodeIn2Grps=OCSP0001E: Node {0} is in two node groups, {1} and {2}, and will therefore not be used.

Explanation

A node was detected to be a member of multiple node groups and will therefore not be used until the condition is resolved.

User response

Make sure each node is a member of at most one node group and restart all processes.

Parent topic: [OCSP](#)

[Terms of Use](#) | [Feedback](#)

OCSP0002E

OCSP_Exception=OCSP0002E: Exception: {0}.

Explanation

An unexpected exception occurred.

User response

Report program defect to IBM.

Parent topic: [OCSP](#)

[Terms of Use](#) | [Feedback](#)

OCSP0003E

OCSP_UnknownNode=OCSP0003E: Node {0} was not found. Invalid custom property value of "node.name" in endpoint of generic server cluster {1}.

Explanation

The "node.name" custom property of a generic server cluster endpoint refers to a non-existent node.

User response

Make sure that the node which is specified by the "node.name" property exists.

Parent topic: [OCSP](#)

OCSP0004E

OCSP_JMS_NoDestDefined=OCSP0004E: No matching destination defined in cell for resource reference binding {0} with destination {1} and Bus {2}

Explanation

Unable to map jms resource jndi binding to destination in the current cell.

User response

Make sure destination and bus name defined by the resource exist in the current cell.

Parent topic: [OCSP](#)

[Terms of Use](#) | [Feedback](#)

OCSP0005E

OCSP_JMS_NoDestDefinedModule=OCSP0005E: No matching destination defined in cell for resource reference binding {0} defined by module {1}

Explanation

Unable to map jms resource jndi binding to destination in the current cell.

User response

Make sure destination and bus name defined by the resource exist in the current cell.

Parent topic: [OCSP](#)

[Terms of Use](#) | [Feedback](#)

OCSP0006I

OCSP_JMS_DESTSWTCH=OCSP0006I: Destination {0} is being updated to be part of work class {1} in application {2}. Earlier it was part of default workclass of {3}

Explanation

Work class membership of destination changed.

User response

none

Parent topic: [OCSP](#)

OCSP0007I

OCSP_JMS_DestAlreadyMember=OCSP0007I: Destination {0} is already part of default work class {1} in application {2}.

Explanation

Work class membership of destination will not be changed.

User response

For optimal results, a given destination cannot be consumed by more than one module.

Parent topic: [OCSP](#)

ODCF

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[ODCF0001E](#)

ODC_UnknownNotify=ODCF0001E: Unknown notification; notification type={0}.

[ODCF0002E](#)

ODC_Exception=ODCF0002E: Exception: {0}.

[ODCF0003E](#)

ODC_AppNotFound=ODCF0003E: Application {0} not found; ignoring event.

[ODCF0004E](#)

ODC_VHostNotFound=ODCF0004E: Virtual host {0} not found for module {1}.

[ODCF0005E](#)

ODC_NoWCThreadPool=ODCF0005E: No thread pool is associated with the webcontainer for server {0}.

[ODCF0006I](#)

ODC_WaitingForInfo=ODCF0006I: Waiting up to {0} seconds to receive routing information.

[ODCF0007I](#)

ODC_CompleteInfoReceived=ODCF0007I: Routing information was received.

[ODCF0008I](#)

ODC_InCompleteInfoReceived=ODCF0008I: Partial Routing information was received after waiting for {0} seconds.

[ODCF0009I](#)

ODC_ServerStartStop=ODCF0009I: Marked server {0} {1} due to transaction: {2}

[ODCF8005E](#)

P2PBase_NoPeer=ODCF8005E: There was an unexpected exception while initializing the SON overlay; the exception is {0}.

[ODCF8010I](#)

P2P_Setup=ODCF8010I: Peer layer setting up.

[ODCF8012I](#)

P2P_NoDMgr=ODCF8012I: Unable to find Deployment Manager host name; running standalone.

[ODCF8014E](#)

P2P_ConfigError=ODCF8014E: Configuration error in peer infrastructure: {0}.

[ODCF8015I](#)

P2P_TestConectFail=ODCF8015I: The unstructured overlay is configured for testing; connection failures will be generated with a probability of {0}.

[ODCF8016I](#)

P2P_TestWriteDelay=ODCF8016I: The unstructured overlay is configured for testing; TCP writes will be delayed {0} milliseconds.

[ODCF8017I](#)

P2P_KeepBug15832=ODCF8017I: The unstructured overlay is configured for testing; the fix to bug 15832 is disabled.

[ODCF8020I](#)

P2P_Starting=ODCF8020I: Peer layer starting; process={0}; boot host={1}.

[ODCF8030I](#)

P2P_Started=ODCF8030I: Peer layer started; process={0}.

[ODCF8040I](#)

ODCF8040=ODCF8040I: Detected process {0} started.

[ODCF8041I](#)

ODCF8041=ODCF8041I: Detected process {0} stopped.

[ODCF8101E](#)

P2P_Throw=ODCF8101E: Unexpected error: {0}.

[ODCF8102W](#)

P2P_WThrow=ODCF8102W: Unexpected warning: {0}.

[ODCF8510I](#)

SON_Setup=ODCF8510I: The unstructured overlay is setting up: {0}.

ODCF8516I

SON_BootedUnsec=ODCF8516I: The unstructured overlay is operational, without security: {0}.

ODCF8517I

SON_BootedSec=ODCF8517I: The unstructured overlay is operational, with security: {0}.

ODCF8525I

SON_StopInboundTcp=ODCF8525I: The unstructured overlay is stopping its inbound TCP channel.

ODCF8526I

SON_StopOutboundTcp=ODCF8526I: The unstructured overlay is stopping its outbound TCP channel.

ODCF8530I

SON_NetStats=ODCF8530I: Cumulative network statistics for {0} are {1}, rates since previous stats are {2}.

ODCF8531I

SON_NbrAddS=ODCF8531I: Added neighbor {0}, neighbor set size is now {1}.

ODCF8532I

SON_NbrAddL=ODCF8532I: Added neighbor {0}, neighbor set is now {1}.

ODCF8533I

SON_NbrSubS=ODCF8533I: Removed neighbor {0}, neighbor set size is now {1}.

ODCF8534I

SON_NbrSubL=ODCF8534I: Removed neighbor {0}, neighbor set is now {1}.

ODCF8535I

SON_MemMgr_Nums=ODCF8535I: At {0}, the cumulative counts for the overlay member manager are {1}; the previous report was at {2} and the changes since then are {3}.

ODCF8540W

SON_MeGone=ODCF8540W: This process was told it is not part of the SON overlay.

ODCF8550I

SON_BuffStats=ODCF8550I: The current {1} protocol statistics for the unstructured overlay member {0} are {2}.

ODCF8601E

SON_EThrow=ODCF8601E: Unexpected error: {0}.

ODCF8602W

SON_WThrow=ODCF8602W: Unexpected warning: {0}.

ODCF0001E

ODC_UnknownNotify=ODCF0001E: Unknown notification; notification type={0}.

Explanation

An unexpected notification was received.

User response

Report program defect to IBM.

Parent topic: [ODCF](#)

ODCF0002E

ODC_Exception=ODCF0002E: Exception: {0}.

Explanation

An unexpected exception occurred.

User response

Report program defect to IBM.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF0003E

ODC_AppNotFound=ODCF0003E: Application {0} not found; ignoring event.

Explanation

An application event was received for a non-existent application.

User response

Report program defect to IBM.

Parent topic: [ODCF](#)

ODCF0004E

ODC_VHostNotFound=ODCF0004E: Virtual host {0} not found for module {1}.

Explanation

A web module is deployed into a virtual host which was not found.

User response

Report program defect to IBM.

Parent topic: [ODCF](#)

ODCF0005E

ODC_NoWCThreadPool=ODCF0005E: No thread pool is associated with the webcontainer for server {0}.

Explanation

An application server's web container did not have an associated thread pool.

User response

Report program defect to IBM.

Parent topic: [ODCF](#)

ODCF0006I

ODC_WaitingForInfo=ODCF0006I: Waiting up to {0} seconds to receive routing information.

Explanation

The Proxy is waiting to receive current routing information. The wait will be a timeout in the specified number of seconds.

User response

None

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF0007I

ODC_CompleteInfoReceived=ODCF0007I: Routing information was received.

Explanation

The Proxy has received current routing information.

User response

None

Parent topic: [ODCF](#)

ODCF0008I

ODC_InCompleteInfoReceived=ODCF0008I: Partial Routing information was received after waiting for {0} seconds.

Explanation

The Proxy Server has received partial routing information, some information may be missing.

User response

If repeated partial information is received, try increasing the timeout value by setting "routing.info.timeout".

Parent topic: [ODCF](#)

ODCF0009I

ODC_ServerStartStop=ODCF0009I: Marked server {0} {1} due to transaction: {2}

Explanation

A server has been marked started or stopped in the local ODC tree

User response

None

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8005E

P2PBase_NoPeer=ODCF8005E: There was an unexpected exception while initializing the SON overlay; the exception is {0}.

Explanation

This should not happen. It can be caused by a problem with the configuration or a conflicting other process(es). This process will be unable to communicate in certain important ways unless and until the problem is corrected and this process is restarted.

User response

Consider the details in the exception. Correct any configuration problem, eliminate conflicting use of the configured port(s), or report to IBM.

Parent topic: [ODCF](#)

ODCF8010I

P2P_Setup=ODCF8010I: Peer layer setting up.

Explanation

The peer communication layer for On Demand Configuration is setting up.

User response

Information only; no action required.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8012I

P2P_NoDMgr=ODCF8012I: Unable to find Deployment Manager host name; running standalone.

Explanation

The peer communication layer for On Demand Configuration could not find the host name of the Deployment Manager for this cell; this process will operate as a standalone process.

User response

Ensure that this node is federated into a cell, unless you intend it to operate in standalone mode.

Parent topic: [ODCF](#)

ODCF8014E

P2P_ConfigError=ODCF8014E: Configuration error in peer infrastructure: {0}.

Explanation

The peer communication layer for On Demand Configuration encountered a configuration error.

User response

Report error condition to IBM.

Parent topic: [ODCF](#)

ODCF8015I

P2P_TestConectFail=ODCF8015I: The unstructured overlay is configured for testing; connection failures will be generated with a probability of {0}.

Explanation

This is a setting used to test failure handling, not to be used in normal operations.

User response

If this testing is not desired, remove the cell custom property named son.connectFailProbability.

Parent topic: [ODCF](#)

ODCF8016I

P2P_TestWriteDelay=ODCF8016I: The unstructured overlay is configured for testing; TCP writes will be delayed {0} milliseconds.

Explanation

This is a setting used to test flow control, not to be used in normal operations.

User response

If this testing is not desired, remove the cell custom property named son.writeDelay.

Parent topic: [ODCF](#)

ODCF8017I

P2P_KeepBug15832=ODCF8017I: The unstructured overlay is configured for testing; the fix to bug 15832 is disabled.

Explanation

This is a setting used to expose problems that are exacerbated by bug 15832, not to be used in normal operations.

User response

If this testing is not desired, remove the cell custom property named son.keepBug15832.

Parent topic: [ODCF](#)

ODCF8020I

P2P_Starting=ODCF8020I: Peer layer starting; process={0}; boot host={1}.

Explanation

The peer communication layer for On Demand Configuration is starting.

User response

Information only; no action required.

Parent topic: [ODCF](#)

ODCF8030I

P2P_Started=ODCF8030I: Peer layer started; process={0}.

Explanation

The peer communication layer for On Demand Configuration has started.

User response

Information only; no action required.

Parent topic: [ODCF](#)

ODCF8040I

ODCF8040=ODCF8040I: Detected process {0} started.

Explanation

An informational message that indicates that this server has detected the start of the indicated process.

User response

None.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8041I

ODCF8041=ODCF8041I: Detected process {0} stopped.

Explanation

An informational message that indicates that this server has detected the stop of the indicated process.

User response

None.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8101E

P2P_EThrow=ODCF8101E: Unexpected error: {0}.

Explanation

An internal error occurred.

User response

Report to IBM.

Parent topic: [ODCF](#)

ODCF8102W

P2P_WThrow=ODCF8102W: Unexpected warning: {0}.

Explanation

A suspicious internal condition arose.

User response

Report to IBM if bad symptoms result.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8510I

SON_Setup=ODCF8510I: The unstructured overlay is setting up: {0}.

Explanation

The peer infrastructure is setting up.

User response

Information only; no action required.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8516I

SON_BootedUnsec=ODCF8516I: The unstructured overlay is operational, without security: {0}.

Explanation

The unstructured overlay has completed its initialization, and is not using SSL.

User response

Information only; no action required.

Parent topic: [ODCF](#)

ODCF8517I

SON_BootedSec=ODCF8517I: The unstructured overlay is operational, with security: {0}.

Explanation

The unstructured overlay has completed its initialization, and is using SSL.

User response

Information only; no action required.

Parent topic: [ODCF](#)

ODCF8525I

SON_StopInboundTcp=ODCF8525I: The unstructured overlay is stopping its inbound TCP channel.

Explanation

This is one step in shutting down the host process.

User response

Information only; no action required.

Parent topic: [ODCF](#)

ODCF8526I

SON_StopOutboundTcp=ODCF8526I: The unstructured overlay is stopping its outbound TCP channel.

Explanation

This is one step in shutting down the host process.

User response

Information only; no action required.

Parent topic: [ODCF](#)

ODCF8530I

SON_NetStats=ODCF8530I: Cumulative network statistics for {0} are {1}, rates since previous stats are {2}.

Explanation

This is a periodic printout of network statistics for the overlay.

User response

Information only; no action required.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8531I

SON_NbrAddS=ODCF8531I: Added neighbor {0}, neighbor set size is now {1}.

Explanation

A new edge was added in the overlay.

User response

Information only; no action required.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8532I

SON_NbrAddL=ODCF8532I: Added neighbor {0}, neighbor set is now {1}.

Explanation

A new edge was added in the overlay.

User response

Information only; no action required.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8533I

SON_NbrSubS=ODCF8533I: Removed neighbor {0}, neighbor set size is now {1}.

Explanation

An edge was removed from the overlay.

User response

Information only; no action required.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8534I

SON_NbrSubL=ODCF8534I: Removed neighbor {0}, neighbor set is now {1}.

Explanation

An edge was removed from the overlay.

User response

Information only; no action required.

Parent topic: [ODCF](#)

ODCF8535I

SON_MemMgr_Nums=ODCF8535I: At {0}, the cumulative counts for the overlay member manager are {1}; the previous report was at {2} and the changes since then are {3}.

Explanation

This is a periodic printout of operational statistics for the overlay.

User response

Information only; no action required.

Parent topic: [ODCF](#)

ODCF8540W

SON_MeGone=ODCF8540W: This process was told it is not part of the SON overlay.

Explanation

This should not happen, and indicates a temporary or continuing problem with connectivity or load.

User response

Report to IBM if bad symptoms are associated.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8550I

SON_BuffStats=ODCF8550I: The current {1} protocol statistics for the unstructured overlay member {0} are {2}.

Explanation

This is a status message; large numbers of active connections or buffers indicate there may be something going wrong.

User response

Report to IBM if this is associated with bad symptoms.

Parent topic: [ODCF](#)

ODCF8601E

SON_Throw=ODCF8601E: Unexpected error: {0}.

Explanation

An internal error occurred.

User response

Report to IBM.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

ODCF8602W

SON_WThrow=ODCF8602W: Unexpected warning: {0}.

Explanation

A suspicious internal condition arose.

User response

Report to IBM if bad symptoms result.

Parent topic: [ODCF](#)

[Terms of Use](#) | [Feedback](#)

SESN

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[SESN0006E](#)

SessionContext.createWhenStop=SESN0006E: Attempted to create a session while the WebSphere Application Server was stopping.

[SESN0007E](#)

SessionContext.accessWhenStop=SESN0007E: Attempted to access a session while the WebSphere Application Server was stopping.

[SESN0008E](#)

SessionContext.unauthAccessError=SESN0008E: A user authenticated as {0} has attempted to access a session owned by {1}.

[SESN0012E](#)

SessionData.putValErr1=SESN0012E: Null key entered. The HttpSession.putValue or HttpSession.setAttribute method was called from a servlet or JSP with a null key.

[SESN0013E](#)

SessionData.putValErr2=SESN0013E: Null value entered for key {0}. The HttpSession.putValue method was called from a servlet or JSP with a null value.

[SESN0066E](#)

SessionContext.responseAlreadyCommitted=SESN0066E: The response is already committed to the client. The session cookie cannot be set.

[SESN0116W](#)

SessionContext.maxSessionIdLengthExceeded=SESN0116W: Session identifier {0} has exceeded the max length limit of {1}.

[SESN0117I](#)

SessionContextRegistry.globalSessionsEnabled=SESN0117I: Global sessions is enabled for Web modules running with the Web container-level session management configuration.

[SESN0118W](#)

SessionContextRegistry.globalSessionTBWarning=SESN0118W: Time-based write is enabled with global sessions. Accessing a global session from more than one server or cluster may result in loss of session data integrity.

[SESN0119W](#)

SessionContextRegistry.globalSessionM2MWarning=SESN0119W: Memory-to-memory replication is enabled with global sessions. Accessing a global session from more than one server or cluster may result in loss of session data integrity.

[SESN0120I](#)

SessionContextRegistry.SessionNotGlobalForWebApp=SESN0120I: Web module {0} will not participate in global sessions because the Web container-level session management configuration has been overridden.

[SESN0121E](#)

SessionContext.CrossoverOnRetrieve=SESN0121E: Session crossover detected in Web application {0}. Session {1} was retrieved when session {2} was expected -

[SESN0122E](#)

SessionContext.CrossoverOnReference=SESN0122E: Session crossover detected in Web application {0}. Session {1} was referenced by method {2} when session {3} was expected -

[SESN0123E](#)

SessionContext.CrossoverOnReturn=SESN0123E: Session crossover detected in Web application {0}. Session {1} was returned to the client when session {2} was expected -

[SESN0124W](#)

SessionContext.DebugCrossoverEnabled=SESN0124W: Session crossover detection is enabled.

[SESN0169I](#)

SessionContext.propertyFound=SESN0169I: Session Manager found the custom property {0} with value {1}.

[SESN0170W](#)

SessionContext.invalidPropertyFound=SESN0170W: Session Manager found the custom property {0} with a non-numeric value {1} so it will be ignored.

[SESN0171W](#)

SessionContext.valueOutOfRange=SESN0171W: Session Manager found the custom property {0} with out-of-range value {1} so it will use {2}.

SESN0172I

SessionIdGeneratorImpl.UsingDefaultSecureRandom=SESN0172I: Using the default SecureRandom implementation for ID generation.

SESN0175I

SessionContextRegistry.existingContext=SESN0175I: Will use an existing session context for application key {0}

SESN0176I

SessionContextRegistry.newContext=SESN0176I: Will create a new session context for application key {0}

SESN0194W

SessionProperties.serverLevelConfigOnly=SESN0194W: Session Manager found custom property {0} with value {1}. It cannot override server level configuration with value {2}. It will be ignored.

SESN0195I

SessionProperties.propertyFoundButAlreadySet=SESN0195I: Session Manager found custom property {0} with value {1}. Because it is the same as the server level configuration property, it will be used.

SESN0196W

Store.createSessionIdAlreadyExists=SESN0196W: The Id Generator generated an id that already exists.

SESN0006E

SessionContext.createWhenStop=SESN0006E: Attempted to create a session while the WebSphere Application Server was stopping.

Explanation

This error occurs when a session request is received while the Application Server is stopping.

User response

Restart the Application Server.

Parent topic: [SESN](#)

SESN0007E

SessionContext.accessWhenStop=SESN0007E: Attempted to access a session while the WebSphere Application Server was stopping.

Explanation

This error occurs when a session request is received while the Application Server is stopping.

User response

Retart the Application Server.

Parent topic: [SESN](#)

SESN0008E

SessionContext.unauthAccessError=SESN0008E: A user authenticated as {0} has attempted to access a session owned by {1}.

Explanation

The Session Security Integration feature has detected an attempt to access a session by an unauthorized user.

User response

No user action is required.

Parent topic: [SESN](#)

SESN0012E

SessionData.putValErr1=SESN0012E: Null key entered. The HttpSession.putValue or HttpSession.setAttribute method was called from a servlet or JSP with a null key.

Explanation

The HttpSession.putValue or HttpSession.setAttribute method cannot be called with a null key.

User response

Fix the servlet or JSP to pass a non-null key.

Parent topic: [SESN](#)

SESN0013E

SessionData.putValErr2=SESN0013E: Null value entered for key {0}. The HttpSession.putValue method was called from a servlet or JSP with a null value.

Explanation

The HttpSession.putValue method cannot be called with a null value.

User response

Fix the servlet or JSP to pass a non-null value.

Parent topic: [SESN](#)

SESN0066E

SessionContext.responseAlreadyCommitted=SESN0066E: The response is already committed to the client. The session cookie cannot be set.

Explanation

The response is already committed to client so the session cookie cannot be sent to client.

User response

Correct the application to access the HTTP session before writing anything to the response.

Parent topic: [SESN](#)

SESN0116W

SessionContext.maxSessionIdLengthExceeded=SESN0116W: Session identifier {0} has exceeded the max length limit of {1}.

Explanation

The value specified for the SessionIdentifierMaxLength property has been exceeded.

User response

Only set this property if absolutely necessary. If this property is required, set it to the largest value that your installation can tolerate. If still experiencing this problem, it is likely due to repeated failovers. Investigate and fix the root cause of the failovers.

Parent topic: [SESN](#)

[Terms of Use](#) | [Feedback](#)

SESN0117I

SessionContextRegistry.globalSessionsEnabled=SESN0117I: Global sessions is enabled for Web modules running with the Web container-level session management configuration.

Explanation

This message is for informational purposes only.

User response

If global sessions is not desired, disable global sessions by setting the Servlet21SessionCompatibility property to false.

Parent topic: [SESN](#)

[Terms of Use](#) | [Feedback](#)

SESN0118W

SessionContextRegistry.globalSessionTBWarning=SESN0118W: Time-based write is enabled with global sessions. Accessing a global session from more than one server or cluster may result in loss of session data integrity.

Explanation

The time-based write feature is enabled with global sessions. Unless all Web modules that access the session are in the same server or cluster, session data integrity may be lost.

User response

Disable the time-based write if Web modules that access global sessions are split across servers or clusters.

Parent topic: [SESN](#)

SESN0119W

SessionContextRegistry.globalSessionM2MWarning=SESN0119W: Memory-to-memory replication is enabled with global sessions. Accessing a global session from more than one server or cluster may result in loss of session data integrity.

Explanation

The memory-to-memory replication feature is enabled with global sessions. Unless all Web modules that access the session are in the same server or cluster, session data integrity may be lost.

User response

Disable memory-to-memory replication if Web Modules that access global sessions are split across servers or clusters.

Parent topic: [SESN](#)

[Terms of Use](#) | [Feedback](#)

SESN0120I

SessionContextRegistry.SessionNotGlobalForWebApp=SESN0120I: Web module {0} will not participate in global sessions because the Web container-level session management configuration has been overridden.

Explanation

The specified Web module will not participate in global sessions because the Web container-level session management configuration has been overridden either at the enterprise application- or Web module-level.

User response

If you want the Web module to participate in global sessions, disable the session management configuration that is specified at the enterprise application- or the Web module-level and restart the server.

Parent topic: [SESN](#)

SESN0121E

SessionContext.CrossoverOnRetrieve=SESN0121E: Session crossover detected in Web application {0}. Session {1} was retrieved when session {2} was expected -

Explanation

A call to the request.getSession method returned a session other than the requested session.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [SESN](#)

SESN0122E

SessionContext.CrossoverOnReference=SESN0122E: Session crossover detected in Web application {0}.
Session {1} was referenced by method {2} when session {3} was expected -

Explanation

An application referenced a session other than the session associated with the request.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [SESN](#)

SESN0123E

SessionContext.CrossoverOnReturn=SESN0123E: Session crossover detected in Web application {0}. Session {1} was returned to the client when session {2} was expected -

Explanation

A cookie or URL was returned to the client containing a session ID that is not associated with the request.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [SESN](#)

SESN0124W

SessionContext.DebugCrossoverEnabled=SESN0124W: Session crossover detection is enabled.

Explanation

Checks for session crossover are being initiated.

User response

For better performance, you may choose to disable these checks by setting the Webcontainer custom property DebugSessionCrossover=false.

Parent topic: [SESN](#)

[Terms of Use](#) | [Feedback](#)

SESN0169I

SessionContext.propertyFound=SESN0169I: Session Manager found the custom property {0} with value {1}.

Explanation

Session Manager will use the specified property and value to control behavior.

User response

Verify that the specified property and value will result in the desired behavior.

Parent topic: [SESN](#)

[Terms of Use](#) | [Feedback](#)

SESN0170W

SessionContext.invalidPropertyFound=SESN0170W: Session Manager found the custom property {0} with a non-numeric value {1} so it will be ignored.

Explanation

Session Manager expected the specified property to contain a numeric value. The property will be ignored.

User response

Correct the specified property value to make it a valid number.

Parent topic: [SESN](#)

SESN0171W

SessionContext.valueOutOfRange=SESN0171W: Session Manager found the custom property {0} with out-of-range value {1} so it will use {2}.

Explanation

Session Manager expected the specified property to have a value within a certain range. Session Manager will use the closest valid value.

User response

Correct the specified property value to make it within the documented range.

Parent topic: [SESN](#)

SESN0172I

SessionIdGeneratorImpl.UsingDefaultSecureRandom=SESN0172I: Using the default SecureRandom implementation for ID generation.

Explanation

Session Manager is using the java default SecureRandom implementation for session ID generation.

User response

No action is required.

Parent topic: [SESN](#)

SESN0175I

SessionContextRegistry.existingContext=SESN0175I: Will use an existing session context for application key {0}

Explanation

An existing session context is going to be shared for this application key.

User response

No action is required.

Parent topic: [SESN](#)

SESN0176I

SessionContextRegistry.newContext=SESN0176I: Will create a new session context for application key {0}

Explanation

A new session context will be created for this application key.

User response

No action is required.

Parent topic: [SESN](#)

SESN0194W

SessionProperties.serverLevelConfigOnly=SESN0194W: Session Manager found custom property {0} with value {1}. It cannot override server level configuration with value {2}. It will be ignored.

Explanation

The custom property can only be set at the server level configuration. The property will be ignored.

User response

Remove the custom property from the application/module level configuration. Change the server level configuration if applicable.

Parent topic: [SESN](#)

SESN0195I

SessionProperties.propertyFoundButAlreadySet=SESN0195I: Session Manager found custom property {0} with value {1}. Because it is the same as the server level configuration property, it will be used.

Explanation

The custom property can only be set at the server level configuration. Since it is the same as the server level configuration property, it will be used.

User response

No action is required.

Parent topic: [SESN](#)

SESN0196W

Store.createSessionIdAlreadyExists=SESN0196W: The Id Generator generated an id that already exists.

Explanation

The Id Generator generated an id that already exists. We will create another id.

User response

No action is required.

Parent topic: [SESN](#)

SRVI

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[SRVI0001E](#)

SRVI_CM_ERR=SRVI0001E: Error getting cluster members for cluster, clusterName={0}.

[SRVI0002E](#)

SRVI_PMI_ERR=SRVI0002E: Exception while configuring PMI for cluster, clusterName={0}.

[SRVI0003E](#)

SRVI_APMI_ERR=SRVI0003E: Error deRegistering for async PMI, serverPath={0}.

[SRVI0004E](#)

SRVI_GET_CLSTRS_ERR=SRVI0004E: Error Retrieving clusters from the the cluster MBean.

[SRVI0005E](#)

SRVI_GET_CLSTR_ERR=SRVI0005E: Error Retrieving a cluster from the the cluster MBean; clusterName={0}.

[SRVI0006E](#)

SRVI_GET_CLSTR_MGR_ERR=SRVI0006E: Error finding Cluster Manager MBean.

[SRVI0007E](#)

SRVI_INV_ATT_ERR=SRVI0007E: Invalid Attribute Exception while adding clusters to cluster balance bean.

[SRVI0008E](#)

SRVI_ENBL_NT_ERR=SRVI0008E: Error while enabling notifications. Exception = {0}

[SRVI0009E](#)

SRVI_APMI2_ERR=SRVI0009E: Error deRegistering for async PMI, serverPath={0}.

[SRVI0010E](#)

SRVI_WLMC_RGSTR_ERR=SRVI0010E: Error registering as a WLM controller

[SRVI0011E](#)

SRVI_SET_WGHT_ERR=SRVI0011E: Error while setting cluster member weight

[SRVI0012E](#)

SRVI_WCCM_ERROR=SRVI0012E: Exception Getting Top Level WCCM Object from Repository with URI={0}.
Exception = {1}

[SRVI0013E](#)

SRVI_ADMCL_ERROR=SRVI0013E: Error getting AdminClient for node= {0}. Exception = {1}

[SRVI0014E](#)

SRVI_MAL_ERROR=SRVI0014E: Malformed Exception while creating ObjectName with queryString = {0}.
Exception = {1}

[SRVI0015E](#)

SRVI_CON_ERROR=SRVI0015E: Connector Exception while querying Object Name ac = {0}. Exception = {1}

[SRVI0016E](#)

SRVI_CONS_THROW=SRVI0016E: Caught Throwable while notifying consumer = {0}. Throwable = {1}

SRVI0001E

SRVI_CM_ERR=SRVI0001E: Error getting cluster members for cluster, clusterName={0}.

Explanation

SRVI Controller could not load the cluster members from the repository.

User response

Report problem to IBM.

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0002E

SRVI_PMI_ERR=SRVI0002E: Exception while configuring PMI for cluster, clusterName={0}.

Explanation

There was exception will trying to configure PMI.

User response

Report problem to IBM.

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0003E

SRVI_APMI_ERR=SRVI0003E: Error deRegistering for async PMI, serverPath={0}.

Explanation

While trying to deregister a server for async pmi, an exception was encountered.

User response

Report problem to IBM.

Parent topic: [SRVI](#)

SRVI0004E

SRVI_GET_CLSTRS_ERR=SRVI0004E: Error Retrieving clusters from the the cluster MBean.

Explanation

There was an exception thrown while invoking retrieveClusters method on the Cluster Manager MBean.

User response

Report problem to IBM.

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0005E

SRVI_GET_CLSTR_ERR=SRVI0005E: Error Retrieving a cluster from the the cluster MBean; clusterName={0}.

Explanation

There was an exception thrown while invoking retrieveCluster method on the Cluster Manager MBean.

User response

Report problem to IBM.

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0006E

SRVI_GET_CLSTR_MGR_ERR=SRVI0006E: Error finding Cluster Manager MBean.

Explanation

While querying for the Cluster Manager MBean an exception was encountered.

User response

Report problem to IBM.

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0007E

SRVI_INV_ATT_ERR=SRVI0007E: Invalid Attribute Exception while adding clusters to cluster balance bean.

Explanation

During initialization, while adding a cluster to the cluster balance bean, an invalid attribute exception was thrown.

User response

Report problem to IBM.

Parent topic: [SRVI](#)

SRVI0008E

SRVI_ENBL_NT_ERR=SRVI0008E: Error while enabling notifications. Exception = {0}

Explanation

An error was encountered while enabling notifications

User response

Report problem to IBM.

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0009E

SRVI_APMI2_ERR=SRVI0009E: Error degistering for async PMI, serverPath={0}.

Explanation

While trying to register a server for async pmi an exception was encountered.

User response

Report problem to IBM.

Parent topic: [SRVI](#)

SRVI0010E

SRVI_WLMC_RGSTR_ERR=SRVI0010E: Error registering as a WLM controller

Explanation

An Error was encountered while registering as a WLM Controller.

User response

Report problem to IBM.

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0011E

SRVI_SET_WGHT_ERR=SRVI0011E: Error while setting cluster member weight

Explanation

An Error was encountered setting a cluster member weight

User response

Report problem to IBM.

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0012E

SRVI_WCCM_ERROR=SRVI0012E: Exception Getting Top Level WCCM Object from Repository with URI={0}.
Exception = {1}

Explanation

Exception Getting Top Level WCCM Object from Repository

User response

Report Problem To IBM

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0013E

SRVI_ADMCL_ERROR=SRVI0013E: Error getting AdminClient for node= {0}. Exception = {1}

Explanation

Error getting AdminClient for node= {0}. Exception = {1}

User response

Report Problem To IBM

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0014E

SRVI_MAL_ERROR=SRVI0014E: Malformed Exception while creating ObjectName with queryString = {0}.
Exception = {1}

Explanation

Malformed Exception while creating ObjectName with queryString = {0}. Exception = {1}

User response

Report Problem To IBM

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0015E

SRVI_CON_ERROR=SRVI0015E: Connector Exception while querying Object Name ac = {0}. Exception = {1}

Explanation

Connector Exception while querying Object Name ac = {0}. Exception = {1}

User response

Report Problem To IBM

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

SRVI0016E

SRVI_CONS_THROW=SRVI0016E: Caught Throwable while notifying consumer = {0}. Throwable = {1}

Explanation

Caught Throwable while notifying consumer = {0}. Throwable = {1}

User response

Report Problem To IBM

Parent topic: [SRVI](#)

[Terms of Use](#) | [Feedback](#)

VMPB

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

VMPB0001I

VC_Enabled=VMPB0001I: VMware Activity Publisher Controller enabled.

VMPB0002I

VC_Disabled=VMPB0002I: VMware Activity Publisher Controller disabled.

VMPB0003E

VC_EnableErr=VMPB0003E: VMware Activity Publisher Controller enable error: {0}.

VMPB0004E

VC_LoginErr=VMPB0004E: VMware Activity Publisher failed to login to the VMware Hyper-Visor.

VMPB0001I

VC_Enabled=VMPB0001I: VMware Activity Publisher Controller enabled.

Explanation

VMware Activity Publisher Controller object is enabled.

User response

No action required.

Parent topic: [VMPB](#)

[Terms of Use](#) | [Feedback](#)

VMPB0002I

VC_Disabled=VMPB0002I: VMware Activity Publisher Controller disabled.

Explanation

VMware Activity Publisher Controller object is disabled.

User response

No action required.

Parent topic: [VMPB](#)

[Terms of Use](#) | [Feedback](#)

VMPB0003E

VC_EnableErr=VMPB0003E: VMware Activity Publisher Controller enable error: {0}.

Explanation

VMware Activity Publisher Controller could not be enabled due to an unexpected error.

User response

Gather the logs, if possible reproduce with at trace specification of vm.pub=all, and send the logs to IBM.

Parent topic: [VMPB](#)

VMPB0004E

VC_LoginErr=VMPB0004E: VMware Activity Publisher failed to login to the VMware Hyper-Visor.

Explanation

VMware Activity Publisher failed to login to the VMWare Hyper-Visor.

User response

Please ensure the VMware user name and password are correct.

Parent topic: [VMPB](#)

[Terms of Use](#) | [Feedback](#)

WHWC

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

WHWC0001E

WHWC_Exception=WHWC0001E: Exception: {0}.

WHWC1600I

INFO_COMPONENT_NAME=WHWC1600I: IBM WebSphere XD Partitions Validation

WHWC0001E

WHWC_Exception=WHWC0001E: Exception: {0}.

Explanation

An unexpected exception occurred.

User response

Report program defect to IBM.

Parent topic: [WHWC](#)

[Terms of Use](#) | [Feedback](#)

WHWC1600I

INFO_COMPONENT_NAME=WHWC1600I: IBM WebSphere XD Partitions Validation

Explanation

No additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [WHWC](#)

[Terms of Use](#) | [Feedback](#)

WHWP

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

WHWP0001E

WHWP_Exception=WHWP0001E: Exception: {0}.

WHWP0002E

WHWP_NotActive=WHWP0002E: Partition {0} is not active on this server.

WHWP0001E

WHWP_Exception=WHWP0001E: Exception: {0}.

Explanation

An unexpected exception occurred.

User response

Report program defect to IBM.

Parent topic: [WHWP](#)

[Terms of Use](#) | [Feedback](#)

WHWP0002E

WHWP_NotActive=WHWP0002E: Partition {0} is not active on this server.

Explanation

A server received an HTTP request for a partition that is not active on this server. This occurs if a partition has been unloaded from a server and has not yet been loaded on another.

User response

No action.

Parent topic: [WHWP](#)

WKPF

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

WKPF2001I

NWMS_SwitchFirstWithCpu=WKPF2001I: For node {0}, initially using a speed factor estimation technique that requires CPU information.

WKPF2002I

NWMS_SwitchFirstSansCpu=WKPF2002I: For node {0}, initially using a speed factor estimation technique that does not require CPU information.

WKPF2003I

NWMS_SwitchLaterWithCpu=WKPF2003I: For node {0}, switching to a speed factor estimation technique that requires CPU information.

WKPF2004I

NWMS_SwitchLaterSansCpu=WKPF2004I: For node {0}, switching a speed factor estimation technique that does not require CPU information.

WKPF2021E

NWMSM_MissedCpu=WKPF2021E: Unable to estimate speed factors for node {0} because of lack of CPU information

WKPF2041E

PCPC_EmptyStats=WKPF2041E: A collector of per-process CPU statistics was given an empty set of statistics, for cell {0}, node {1}, server {2}, path {3}.

WKPF2042E

PCPC_StatsSansCpu=WKPF2042E: A collector of per-process CPU statistics was given a set of statistics that did not include average CPU utilization, for cell {0}, node {1}, server {2}, path {3}.

WKPF2061E

ENCC_StatsSansCpu=WKPF2061E: A collector of external CPU statistics was given a set of statistics that did not include average CPU utilization, for cell {0}, node {1}, server {2}, path {3}.

WKPF2081E

AvCC_EmptyStats=WKPF2081E: A collector of average CPU statistics was given an empty set of statistics, for cell {0}, node {1}, server {2}, path {3}.

WKPF2082E

AvCC_StatsSansCpu=WKPF2082E: A collector of average CPU statistics was given a set of statistics that did not include average CPU utilization, for cell {0}, node {1}, server {2}, path {3}.

WKPF2200I

WP_UP=WKPF2200I: work profiler for Extended Deployment is operational

WKPF2201I

WP_CoreDump=WKPF2201I: The complete set of state for a healthy work profiler is: {0}

WKPF2202E

WP_CoreDumpErr=WKPF2202E: The complete set of state for an unhealthy work profiler is: {0}

WKPF2001I

NWMS_SwitchFirstWithCpu=WKPF2001I: For node {0}, initially using a speed factor estimation technique that requires CPU information.

Explanation

The work profiler has two ways to estimate speed factors involving a given node. The better one requires CPU information about the node, and is used when that information is available.

User response

No user action required.

Parent topic: [WKPF](#)

WKPF2002I

NWMS_SwitchFirstSansCpu=WKPF2002I: For node {0}, initially using a speed factor estimation technique that does not require CPU information.

Explanation

The work profiler has two ways to estimate speed factors involving a given node. The better one requires CPU information about the node, and is used when that information is available.

User response

No user action required.

Parent topic: [WKPF](#)

WKPF2003I

NWMS_SwitchLaterWithCpu=WKPF2003I: For node {0}, switching to a speed factor estimation technique that requires CPU information.

Explanation

The work profiler has two ways to estimate speed factors involving a given node. The better one requires CPU information about the node, and is used when that information is available.

User response

No user action required.

Parent topic: [WKPF](#)

[Terms of Use](#) | [Feedback](#)

WKPF2004I

NWMS_SwitchLaterSansCpu=WKPF2004I: For node {0}, switching a speed factor estimation technique that does not require CPU information.

Explanation

The work profiler has two ways to estimate speed factors involving a given node. The better one requires CPU information about the node, and is used when that information is available.

User response

No user action required.

Parent topic: [WKPF](#)

WKPF2021E

NWMSM_MissedCpu=WKPF2021E: Unable to estimate speed factors for node {0} because of lack of CPU information

Explanation

The work profiler has been configured to use a speed factor estimation technique that requires CPU information but that information is now unavailable for the named node.

User response

Configure a different choice of speed factor estimation techniques, or fix the problem that cause a lack of CPU information.

Parent topic: [WKPF](#)

WKPF2041E

PCPC_EmptyStats=WKPF2041E: A collector of per-process CPU statistics was given an empty set of statistics, for cell {0}, node {1}, server {2}, path {3}.

Explanation

This is an internal error that should never arise.

User response

Report to IBM.

Parent topic: [WKPF](#)

WKPF2042E

PCPC_StatsSansCpu=WKPF2042E: A collector of per-process CPU statistics was given a set of statistics that did not include average CPU utilization, for cell {0}, node {1}, server {2}, path {3}.

Explanation

This is an internal error that should never arise.

User response

Report to IBM.

Parent topic: [WKPF](#)

WKPF2061E

ENCC_StatsSansCpu=WKPF2061E: A collector of external CPU statistics was given an set of statistics that did not include average CPU utilization, for cell {0}, node {1}, server {2}, path {3}.

Explanation

This is an internal error that should never arise.

User response

Report to IBM.

Parent topic: [WKPF](#)

[Terms of Use](#) | [Feedback](#)

WKPF2081E

AvCC_EmptyStats=WKPF2081E: A collector of average CPU statistics was given an empty set of statistics, for cell {0}, node {1}, server {2}, path {3}.

Explanation

This is an internal error that should never arise.

User response

Report to IBM.

Parent topic: [WKPF](#)

WKPF2082E

AvCC_StatsSansCpu=WKPF2082E: A collector of average CPU statistics was given a set of statistics that did not include average CPU utilization, for cell {0}, node {1}, server {2}, path {3}.

Explanation

This is an internal error that should never arise.

User response

Report to IBM.

Parent topic: [WKPF](#)

[Terms of Use](#) | [Feedback](#)

WKPF2200I

WP_UP=WKPF2200I: work profiler for Extended Deployment is operational

Explanation

an informational message that indicate overall subsystem status

User response

None.

Parent topic: [WKPF](#)

[Terms of Use](#) | [Feedback](#)

WKPF2201I

WP_CoreDump=WKPF2201I: The complete set of state for a healthy work profiler is: {0}

Explanation

A detailed dump of all work profiler state.

User response

None.

Parent topic: [WKPF](#)

[Terms of Use](#) | [Feedback](#)

WKPF2202E

WP_CoreDumpErr=WKPF2202E: The complete set of state for an unhealthy work profiler is: {0}

Explanation

A detailed dump of all work profiler state.

User response

First review task management for potential remedies, otherwise contact IBM support.

Parent topic: [WKPF](#)

[Terms of Use](#) | [Feedback](#)

WPVR

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[WPVR0001I](#)

WPVR0001I=WPVR0001I: Application {0}, Edition {1} - edition state set to {2}.

[WPVR0002E](#)

WPVR0002E=WPVR0002E: Application {0}, Edition {1} - cannot activate an edition in the {2} state.

[WPVR0003E](#)

WPVR0003E=WPVR0003E: Application {0}, Edition {1} - cannot deactivate an edition in the {2} state.

[WPVR0004I](#)

WPVR0004I=WPVR0004I: Activating edition {0} of application {1}.

[WPVR0005I](#)

WPVR0005I=WPVR0005I: Activation completed for edition {0} of application {1}.

[WPVR0006E](#)

WPVR0006E=WPVR0006E: Activation failed for edition {0} of application {1}.

[WPVR0007I](#)

WPVR0007I=WPVR0007I: Deactivating edition {0} of application {1}.

[WPVR0008I](#)

WPVR0008I=WPVR0008I: Deactivation completed for edition {0} of application {1}.

[WPVR0009E](#)

WPVR0009E=WPVR0009E: Deactivation failed for edition {0} of application {1}.

[WPVR0010I](#)

WPVR0010I=WPVR0010I: Rollout started for edition {0} of application {1}.

[WPVR0011E](#)

WPVR0011E=WPVR0011E: Rollout of edition {0} of application {1} failed. Check the log for details.

[WPVR0012I](#)

WPVR0012I=WPVR0012I: Rollout for edition {0} of application {1} completed successfully.

[WPVR0013E](#)

WPVR0013E=WPVR0013E: Rollout failed because no target or previous edition information found for edition {0} of application {1}.

[WPVR0014I](#)

WPVR0014I=WPVR0014I: Rollout: Edition {0} of application {1} deactivated. Edition {2} activated.

[WPVR0015I](#)

WPVR0015I=WPVR0015I: Rollout: Processing server {0}.

[WPVR0016I](#)

WPVR0016I=WPVR0016I: Rollout: Quiescing server/application at {0}.

[WPVR0017W](#)

WPVR0017W=WPVR0017W: Rollout: Failed to quiesce {0}. It may be stopped.

[WPVR0018I](#)

WPVR0018I=WPVR0018I: Rollout: Stopping {0}.

[WPVR0019W](#)

WPVR0019W=WPVR0019W: Rollout: Failed to stop {0}. It may be stopped.

[WPVR0020I](#)

WPVR0020I=WPVR0020I: Rollout: Synchronizing node {0}.

[WPVR0021W](#)

WPVR0021W=WPVR0021W: Rollout: Failed to synchronize node {0}.

[WPVR0022I](#)

WPVR0022I=WPVR0022I: Rollout: Starting {0}.

[WPVR0023W](#)

WPVR0023W=WPVR0023W: Rollout: Failed to start {0}.

[WPVR0024E](#)

WPVR0024E=WPVR0024E: Rollout: Cannot rollout an active edition - edition {0} of application {1}.

WPVR0025I

WPVR0025I=WPVR0025I: Rollout: Draining {0}.

WPVR0026W

WPVR0026W=WPVR0026W: Rollout: Not all servers of the cluster {0} are running. Application {1} may become unavailable during rollout.

WPVR0027E

WPVR0027E=WPVR0027E: Activate: Edition {0} is already active on the same target. You can have only one edition active on a target.

WPVR0028I

WPVR0028I=WPVR0028I: Validation started for edition {0} of application {1}.

WPVR0029E

WPVR0029E=WPVR0029E: Validation failed for edition {0} of application {1}. Check the log for details.

WPVR0030I

WPVR0030I=WPVR0030I: Validation completed for edition {0} of application {1}.

WPVR0031I

WPVR0031I=WPVR0031I: Mapping module {1} to new targets for {0}.

WPVR0032I

WPVR0032I=WPVR0032I: Mapping JNDI name for edition {0} of application {1}.

WPVR0033I

WPVR0033I=WPVR0033I: Validation cleanup in progress for edition {0} of application {1}.

WPVR0034I

WPVR0034I=WPVR0034I: Validation cleanup of edition {0} of application {1} completed successfully.

WPVR0035I

WPVR0035I=WPVR0035I: Preparing to rollout from validation mode in progress for edition {0} of application {1}.

WPVR0036E

WPVR0036E=WPVR0036E: Error encountered during rollout from validation mode for edition {0} of application {1}. Check the logs for details.

WPVR0037I

WPVR0037I=WPVR0037I: Message: {0}.

WPVR0038I

WPVR0038I=WPVR0038I: Validation: Cloned cluster {0} as {1}.

WPVR0039E

WPVR0039E=WPVR0039E: Validation: Cloning cluster {0} failed. Check the log for details.

WPVR0042I

WPVR0042I=WPVR0042I: Validation: Cloned server {0} as {1}.

WPVR0043E

WPVR0043E=WPVR0043E: Validation: Cloning server {0} failed. Check the logs for details.

WPVR0047I

WPVR0047I=WPVR0047I: Validation: Deleted cluster {0}.

WPVR0048I

WPVR0048I=WPVR0048I: Validation: Deleted server {0}.

WPVR0049E

WPVR0049E=WPVR0049E: Validation: Deleting cluster {0} failed. Check the logs for details.

WPVR0050E

WPVR0050E=WPVR0050E: Validation: Deleting server {0} failed. Check the logs for details.

WPVR0051E

WPVR0051E=WPVR0051E: Application {0} - Cannot validate an edition in {1} state.

WPVR0052E

WPVR0052E=WPVR0052E: Unsaved changes. Save changes before continuing.

WPVR0053I

WPVR0053I=WPVR0053I: Rollout: Drain completed for {0}.

WPVR0054I

WPVR0054I=WPVR0054I: Rollout: Draining {0} ({1} seconds)

WPVR0055W

WPVR0055W=WPVR0055W: Rollout completes with errors. Check the logs for details.

WPVR0056E

WPVR0056E=WPVR0056E: The application named "{0}" could not be found.

WPVR0057E

WPVR0057E=WPVR0057E: The edition named "{0}" could not be found.

WPVR0058E

WPVR0058E=WPVR0058E: The enterprise application contains more than two modules. This is not supported by Application Edition.

WPVR0059E

WPVR0059E=WPVR0059E: The application is a SIP application. Invalid parameters specified.

WPVR0060E

WPVR0060E=WPVR0060E: The application is not a SIP application. Invalid parameters specified.

WPVR0061I

WPVR0061I=WPVR0061I: {0} started.

WPVR0062E

WPVR0062E=WPVR0062E: Validation target(s) already exists. Another edition of the application may already be in validation mode.

WPVR0063I

WPVR0063I=WPVR0063I: Cannot validate the only available edition of the application.

WPVR0064I

WPVR0064I=WPVR0064I: This action is not available to unmanaged web applications.

WPVR0065I

WPVR0065I=WPVR0065I: Rollout: Cancelling compute grid jobs on server {0}.

WPVR0066I

WPVR0066I=WPVR0066I: Rollout: No compute grid jobs were cancelled on server {0}.

WPVR0067W

WPVR0067W=WPVR0067W: Rollout: Cancelled compute grid job(s) {0} on server {1}.

WPVR0101I

WPVR0101I=WPVR0101I: This is a rollout of a Session Initiation Protocol (SIP) application.

WPVR0111I

WPVR0111I=WPVR0111I: Server or cluster member {0} with edition {1} of application {2} created.

WPVR0112I

WPVR0112I=WPVR0112I: Server or cluster member {0} with edition {1} of application {2} started.

WPVR0113I

WPVR0113I=WPVR0113I: Route new traffic to servers or cluster members with edition {0} of application {1}.

WPVR0114I

WPVR0114I=WPVR0114I: Server or cluster member {0} deleted.

WPVR0115I

WPVR0115I=WPVR0115I: Server or cluster member {0} quiesced.

WPVR0121E

WPVR0121E=WPVR0121E: Failed to create server or cluster member {0} with edition {1} of application {2}.

WPVR0122E

WPVR0122E=WPVR0122E: Failed to start server or cluster member {0} with edition {1} of application {2}.

WPVR0123E

WPVR0123E=WPVR0123E: Failed to route new traffic to servers or cluster members with edition {0} of application {1}.

WPVR0124W

WPVR0124W=WPVR0124W: Failed to delete server or cluster member {0}.

WPVR0125W

WPVR0125W=WPVR0125W: Failed to quiesce server or cluster member {0}.

WPVR0201E

WPVR0201E=WPVR0201E: Failed to deploy edition {0} of application {1} onto server {2} on node {3}.

WPVR0202E

WPVR0202E=WPVR0202E: Failed to undeploy edition {0} of application {1} from server {2} on node {3}.

[WPVR0203E](#)

WPVR0203E=WPVR0203E: Failed to start edition {0} of application {1} on server {2} of node {3}.

[WPVR0204E](#)

WPVR0204E=WPVR0204E: Failed to stop edition {0} of application {1} on server {2} of node {3}.

[WPVR0500E](#)

WPVR0500E=WPVR0500E: The parameter, {0}, cannot be null or empty.

[WPVR0501E](#)

WPVR0501E=WPVR0501E: The specified node group cannot be found.

[WPVR0502E](#)

WPVR0502E=WPVR0502E: The specified operational mode is invalid.

[WPVR0503I](#)

WPVR0503I=WPVR0503I: Cloned dynamic cluster {0} from dynamic cluster {1} successfully.

[WPVR0504E](#)

WPVR0504E=WPVR0504E: Failed to clone dynamic cluster {0} from dynamic cluster {1}.

[WPVR0505I](#)

WPVR0505I=WPVR0505I: Canceling validation of edition {0} of application {1}.

[WPVR0506E](#)

WPVR0506E=WPVR0506E: Application {0}, Edition {1} - cannot cancel validation on an edition in the {2} state.

[WPVR0507I](#)

WPVR0507I=WPVR0507I: Canceling the validation completed for edition {0} of application {1}.

[WPVR0508E](#)

WPVR0508E=WPVR0508E: Canceling the validation failed for edition {0} of application {1}.

[WPVR0514W](#)

WPVR0514W=WPVR0514W: Starting nodeserver {0} to improve availability of application {1} during rollout since only 1 server is found running currently.

[WPVR0515I](#)

WPVR0515I=WPVR0515I: Rollout: Disallowing new requests w/o affinity to server/application at {0}.

[WPVR0517E](#)

WPVR0517E=WPVR0517E: Cannot deactivate edition {0} of application {1} as {2} routing rules reference it, please refer to the logs for the list of rules to be removed.

WPVR0001I

WPVR0001I=WPVR0001I: Application {0}, Edition {1} - edition state set to {2}.

Explanation

An informational message that indicates that an edition has been put into a particular state.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0002E

WPVR0002E=WPVR0002E: Application {0}, Edition {1} - cannot activate an edition in the {2} state.

Explanation

The system cannot activate an edition in the particular state.

User response

Ensure that the particular edition of the application is in the correct state for activation.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0003E

WPVR0003E=WPVR0003E: Application {0}, Edition {1} - cannot deactivate an edition in the {2} state.

Explanation

The system cannot deactivate an edition in the particular state.

User response

Ensure that the particular edition of the application is in the correct state for deactivation.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0004I

WPVR0004I=WPVR0004I: Activating edition {0} of application {1}.

Explanation

An informational message that indicates that the system is activating an edition.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0005I

WPVR0005I=WPVR0005I: Activation completed for edition {0} of application {1}.

Explanation

An informational message that indicates that activation has completed for an edition.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0006E

WPVR0006E=WPVR0006E: Activation failed for edition {0} of application {1}.

Explanation

The system failed to activate the particular edition of the application.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0007I

WPVR0007I=WPVR0007I: Deactivating edition {0} of application {1}.

Explanation

An informational message that indicates that the system is deactivating an edition.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0008I

WPVR0008I=WPVR0008I: Deactivation completed for edition {0} of application {1}.

Explanation

An informational message that indicates that deactivation has completed for an edition.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0009E

WPVR0009E=WPVR0009E: Deactivation failed for edition {0} of application {1}.

Explanation

The system failed to deactivate the particular edition of the application.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0010I

WPVR0010I=WPVR0010I: Rollout started for edition {0} of application {1}.

Explanation

An informational message that indicates that rollout has started for an edition.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0011E

WPVR0011E=WPVR0011E: Rollout of edition {0} of application {1} failed. Check the log for details.

Explanation

The system failed to rollout the particular edition of the application.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0012I

WPVR0012I=WPVR0012I: Rollout for edition {0} of application {1} completed successfully.

Explanation

An informational message that indicates that the system successfully rolled out an edition.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0013E

WPVR0013E=WPVR0013E: Rollout failed because no target or previous edition information found for edition {0} of application {1}.

Explanation

The system failed to rollout an edition due to missing target or edition information.

User response

Ensure that the edition information is proper and the edition is properly deployed to a target.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0014I

WPVR0014I=WPVR0014I: Rollout: Edition {0} of application {1} deactivated. Edition {2} activated.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0015I

WPVR0015I=WPVR0015I: Rollout: Processing server {0}.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0016I

WPVR0016I=WPVR0016I: Rollout: Quiescing server/application at {0}.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0017W

WPVR0017W=WPVR0017W: Rollout: Failed to quiesce {0}. It may be stopped.

Explanation

The system failed to quiesce an application or server during rollout.

User response

Check the logs for details and recycle the application or server if necessary.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0018I

WPVR0018I=WPVR0018I: Rollout: Stopping {0}.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0019W

WPVR0019W=WPVR0019W: Rollout: Failed to stop {0}. It may be stopped.

Explanation

The system failed to stop an application or server, which may already be stopped, during rollout.

User response

Check the logs for details and recycle the application or server if necessary.

Parent topic: [WPVR](#)

WPVR0020I

WPVR0020I=WPVR0020I: Rollout: Synchronizing node {0}.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0021W

WPVR0021W=WPVR0021W: Rollout: Failed to synchronize node {0}.

Explanation

The system failed to synchronize a node during rollout.

User response

Check the logs for details and synchronize the node if necessary.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0022I

WPVR0022I=WPVR0022I: Rollout: Starting {0}.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0023W

WPVR0023W=WPVR0023W: Rollout: Failed to start {0}.

Explanation

The system failed to start an application or server during rollout.

User response

Check the logs for details and restart the application or server if necessary.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0024E

WPVR0024E=WPVR0024E: Rollout: Cannot rollout an active edition - edition {0} of application {1}.

Explanation

The system failed to rollout an edition.

User response

Check the logs for details.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0025I

WPVR0025I=WPVR0025I: Rollout: Draining {0}.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0026W

WPVR0026W=WPVR0026W: Rollout: Not all servers of the cluster {0} are running. Application {1} may become unavailable during rollout.

Explanation

A warning that continuous availability cannot be guaranteed for the particular rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0027E

WPVR0027E=WPVR0027E: Activate: Edition {0} is already active on the same target. You can have only one edition active on a target.

Explanation

An edition is already active on the particular target.

User response

Deactivate the currently active edition before activating another.

Parent topic: [WPVR](#)

WPVR0028I

WPVR0028I=WPVR0028I: Validation started for edition {0} of application {1}.

Explanation

An informational message that indicates the progress of putting an edition into validation mode.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0029E

WPVR0029E=WPVR0029E: Validation failed for edition {0} of application {1}. Check the log for details.

Explanation

The system failed to put an edition into validation mode.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0030I

WPVR0030I=WPVR0030I: Validation completed for edition {0} of application {1}.

Explanation

An informational message that indicates the progress of putting an edition into validation mode.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0031I

WPVR0031I=WPVR0031I: Mapping module {1} to new targets for {0}.

Explanation

An informational progress message.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0032I

WPVR0032I=WPVR0032I: Mapping JNDI name for edition {0} of application {1}.

Explanation

An informational progress message.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0033I

WPVR0033I=WPVR0033I: Validation cleanup in progress for edition {0} of application {1}.

Explanation

An informational message that indicates that cleanup of the validation target of an edition has started.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0034I

WPVR0034I=WPVR0034I: Validation cleanup of edition {0} of application {1} completed successfully.

Explanation

An informational message that indicates that cleanup of the validation target of an edition completed successfully.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0035I

WPVR0035I=WPVR0035I: Preparing to rollout from validation mode in progress for edition {0} of application {1}.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0036E

WPVR0036E=WPVR0036E: Error encountered during rollout from validation mode for edition {0} of application {1}. Check the logs for details.

Explanation

The system failed to rollout an edition from validation mode.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0037I

WPVR0037I=WPVR0037I: Message: {0}.

Explanation

An informational system message.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0038I

WPVR0038I=WPVR0038I: Validation: Cloned cluster {0} as {1}.

Explanation

An informational message that indicates the progress of putting an edition into validation mode.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0039E

WPVR0039E=WPVR0039E: Validation: Cloning cluster {0} failed. Check the log for details.

Explanation

The system failed to clone a cluster for validation.

User response

Check the logs for details.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0042I

WPVR0042I=WPVR0042I: Validation: Cloned server {0} as {1}.

Explanation

An informational message that indicates the progress of putting an edition into validation mode.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0043E

WPVR0043E=WPVR0043E: Validation: Cloning server {0} failed. Check the logs for details.

Explanation

The system failed to clone a server for validation.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0047I

WPVR0047I=WPVR0047I: Validation: Deleted cluster {0}.

Explanation

An informational message that indicates that the system has deleted a validation cluster.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0048I

WPVR0048I=WPVR0048I: Validation: Deleted server {0}.

Explanation

An informational message that indicates that the system has deleted a validation server.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0049E

WPVR0049E=WPVR0049E: Validation: Deleting cluster {0} failed. Check the logs for details.

Explanation

The system failed to delete a validation cluster.

User response

Check the logs for details.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0050E

WPVR0050E=WPVR0050E: Validation: Deleting server {0} failed. Check the logs for details.

Explanation

The system failed to delete a validation server.

User response

Check the logs for details.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0051E

WPVR0051E=WPVR0051E: Application {0} - Cannot validate an edition in {1} state.

Explanation

The system cannot validate an edition in the particular state.

User response

Ensure that the edition is in the proper state for validation.

Parent topic: [WPVR](#)

WPVR0052E

WPVR0052E=WPVR0052E: Unsaved changes. Save changes before continuing.

Explanation

There is unsaved changes in the workspace. Changes should be saved before work continues.

User response

Save or undo the changes in the workspace before continuing.

Parent topic: [WPVR](#)

WPVR0053I

WPVR0053I=WPVR0053I: Rollout: Drain completed for {0}.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0054I

WPVR0054I=WPVR0054I: Rollout: Draining {0} ({1} seconds)

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0055W

WPVR0055W=WPVR0055W: Rollout completes with errors. Check the logs for details.

Explanation

The rollout process completed but non-fatal errors were encountered. Check the logs for details.

User response

Check the logs for details and perform any corrective action if necessary.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0056E

WPVR0056E=WPVR0056E: The application named "{0}" could not be found.

Explanation

The system failed to find the named application.

User response

Ensure that the named application is properly installed.

Parent topic: [WPVR](#)

WPVR0057E

WPVR0057E=WPVR0057E: The edition named "{0}" could not be found.

Explanation

The system failed to find the particular edition.

User response

Ensure that the particular edition of the application is properly installed.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0058E

WPVR0058E=WPVR0058E: The enterprise application contains more than two modules. This is not supported by Application Edition.

Explanation

Application Edition Management supports EAR with two modules only.

User response

Application Edition Management is not available for such enterprise applications.

Parent topic: [WPVR](#)

WPVR0059E

WPVR0059E=WPVR0059E: The application is a SIP application. Invalid parameters specified.

Explanation

The specified parameters are not applicable for a SIP application.

User response

Specify the correct parameters required for a SIP application and retry.

Parent topic: [WPVR](#)

WPVR0060E

WPVR0060E=WPVR0060E: The application is not a SIP application. Invalid parameters specified.

Explanation

SIP application-specific parameters specified for a non-SIP application.

User response

Specify the correct parameters required for a non-SIP application and retry.

Parent topic: [WPVR](#)

WPVR0061I

WPVR0061I=WPVR0061I: {0} started.

Explanation

An informational message that indicates that a server or an application has started.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0062E

WPVR0062E=WPVR0062E: Validation target(s) already exists. Another edition of the application may already be in validation mode.

Explanation

Deployment target(s) of the same name(s) as the validation target(s) to be created already exists.

User response

Ensure that no other edition is in validation mode or remove the duplicate target(s) and retry.

Parent topic: [WPVR](#)

WPVR0063I

WPVR0063I=WPVR0063I: Cannot validate the only available edition of the application.

Explanation

The only available edition of an application cannot be placed into validation mode.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0064I

WPVR0064I=WPVR0064I: This action is not available to unmanaged web applications.

Explanation

Unmanaged web applications cannot be rolled out or validated.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0065I

WPVR0065I=WPVR0065I: Rollout: Cancelling compute grid jobs on server {0}.

Explanation

An informational message that indicates that compute grid jobs are being cancelled.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0066I

WPVR0066I=WPVR0066I: Rollout: No compute grid jobs were cancelled on server {0}.

Explanation

An informational message that indicates that no compute grid jobs were cancelled.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0067W

WPVR0067W=WPVR0067W: Rollout: Cancelled compute grid job(s) {0} on server {1}.

Explanation

Compute grid jobs were cancelled before stopping application or server.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0101I

WPVR0101I=WPVR0101I: This is a rollout of a Session Initiation Protocol (SIP) application.

Explanation

An informational message that indicates that the particular rollout is a rollout of a SIP application.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR011I

WPVR011I=WPVR011I: Server or cluster member {0} with edition {1} of application {2} created.

Explanation

An informational message that indicates that a server or cluster member with the particular edition of the named application has been created.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0112I

WPVR0112I=WPVR0112I: Server or cluster member {0} with edition {1} of application {2} started.

Explanation

An informational message that indicates that a server or cluster member with the particular edition of the named application has been started.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0113I

WPVR0113I=WPVR0113I: Route new traffic to servers or cluster members with edition {0} of application {1}.

Explanation

An informational message that indicates that new traffic will be routed to servers or cluster members with the particular edition of the application.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0114I

WPVR0114I=WPVR0114I: Server or cluster member {0} deleted.

Explanation

An informational message that indicates that a server or cluster member has been deleted.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0115I

WPVR0115I=WPVR0115I: Server or cluster member {0} quiesced.

Explanation

An informational message that indicates that a server or cluster member has been quiesced.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0121E

WPVR0121E=WPVR0121E: Failed to create server or cluster member {0} with edition {1} of application {2}.

Explanation

The system failed to create a particular server or cluster member.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0122E

WPVR0122E=WPVR0122E: Failed to start server or cluster member {0} with edition {1} of application {2}.

Explanation

The system failed to start a particular server or cluster member.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0123E

WPVR0123E=WPVR0123E: Failed to route new traffic to servers or cluster members with edition {0} of application {1}.

Explanation

The system failed to re-route traffic to certain servers or cluster members.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0124W

WPVR0124W=WPVR0124W: Failed to delete server or cluster member {0}.

Explanation

The system failed to delete a particular server or cluster member.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0125W

WPVR0125W=WPVR0125W: Failed to quiesce server or cluster member {0}.

Explanation

The system failed to quiesce a particular server or cluster member.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0201E

WPVR0201E=WPVR0201E: Failed to deploy edition {0} of application {1} onto server {2} on node {3}.

Explanation

The system failed to deploy the particular application edition onto the named server.

User response

Check the server logs for details, resolve the issues, and retry.

Parent topic: [WPVR](#)

WPVR0202E

WPVR0202E=WPVR0202E: Failed to undeploy edition {0} of application {1} from server {2} on node {3}.

Explanation

The system failed to undeploy the particular application edition onto the named server.

User response

Check the server logs for details, resolve the issues, and retry.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0203E

WPVR0203E=WPVR0203E: Failed to start edition {0} of application {1} on server {2} of node {3}.

Explanation

The system failed to start the particular application edition on the named server.

User response

Check the server logs for details, resolve the issues, and retry.

Parent topic: [WPVR](#)

WPVR0204E

WPVR0204E=WPVR0204E: Failed to stop edition {0} of application {1} on server {2} of node {3}.

Explanation

The system failed to stop the particular application edition on the named server.

User response

Check the server logs for details, resolve the issues, and retry.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0500E

WPVR0500E=WPVR0500E: The parameter, {0}, cannot be null or empty.

Explanation

The particular parameter cannot have a null or empty value.

User response

Specify a correct value for the parameter that is not null or empty.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0501E

WPVR0501E=WPVR0501E: The specified node group cannot be found.

Explanation

Node group of the specified name cannot be found.

User response

Specify a node group that exists.

Parent topic: [WPVR](#)

WPVR0502E

WPVR0502E=WPVR0502E: The specified operational mode is invalid.

Explanation

The operational mode specified is an invalid operational mode. Valid values are "automatic", "supervised", and "manual".

User response

Specify a correct operational mode. Valid values are "automatic", "supervised", and "manual".

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0503I

WPVR0503I=WPVR0503I: Cloned dynamic cluster {0} from dynamic cluster {1} successfully.

Explanation

A dynamic cluster has been successfully cloned.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0504E

WPVR0504E=WPVR0504E: Failed to clone dynamic cluster {0} from dynamic cluster {1}.

Explanation

A dynamic cluster failed to be cloned.

User response

Check the logs for details and retry as appropriate.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0505I

WPVR0505I=WPVR0505I: Canceling validation of edition {0} of application {1}.

Explanation

An informational message that indicates that the system is canceling the validation an edition.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0506E

WPVR0506E=WPVR0506E: Application {0}, Edition {1} - cannot cancel validation on an edition in the {2} state.

Explanation

The system cannot cancel validation on an edition in the particular state.

User response

Ensure that the particular edition of the application is in the correct state for canceling validation.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0507I

WPVR0507I=WPVR0507I: Canceling the validation completed for edition {0} of application {1}.

Explanation

An informational message that indicates that canceling the validation has completed for an edition.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0508E

WPVR0508E=WPVR0508E: Canceling the validation failed for edition {0} of application {1}.

Explanation

The system failed to cancel the validation of the particular edition of the application.

User response

Check the logs for details.

Parent topic: [WPVR](#)

WPVR0514W

WPVR0514W=WPVR0514W: Starting nodeserver {0} to improve availability of application {1} during rollout since only 1 server is found running currently.

Explanation

During the rollout only one server serving the application is found and so another server instance is being started to provide availability during the rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

WPVR0515I

WPVR0515I=WPVR0515I: Rollout: Disallowing new requests w/o affinity to server/application at {0}.

Explanation

An informational message that indicates the progress of a rollout.

User response

No user action is required.

Parent topic: [WPVR](#)

[Terms of Use](#) | [Feedback](#)

WPVR0517E

WPVR0517E=WPVR0517E: Cannot deactivate edition {0} of application {1} as {2} routing rules reference it, please refer to the logs for the list of rules to be removed.

Explanation

The system cannot deactivate an edition since there are routing rules which reference this edition and they need to be removed before deactivate can be performed.

User response

Ensure that the particular edition of the application does not have any routing rules referencing it.

Parent topic: [WPVR](#)

WUTM

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[WUTM1000E](#)

WUTM_UNKNOWN_ERROR=WUTM1000E: Exception occurred in the Runtime Task UI. Exception was {0}.

[WUTM1001E](#)

ERROR_MBEAN_ACCESS=WUTM1001E: Exception occurred while trying to access the TaskManagement MBean. Exception was {0}.

[WUTM1002E](#)

ERROR_INVOKE_OPERATION=WUTM1002E: Exception occurred while invoking an operation on the TaskManagement MBean. Exception was {0}.

[WUTM1003E](#)

ERROR_EXECUTE_ACTION=WUTM1003E: Exception occurred while executing an action. Exception was {0}.

[WUTM1004E](#)

INVALID_TASK_ID=WUTM1004E: Task ID was either null or invalid. Exception was {0}.

[WUTM1005E](#)

ERROR_MONITOR_LINK=WUTM1005E: Exception occurred while processing task target monitor link. Exception was {0}.

[WUTM1006E](#)

ERROR_IBM_MESSAGE=WUTM1006E: Exception occurred while processing IBM error messages. Exception was {0}.

WUTM1000E

WUTM_UNKNOWN_ERROR=WUTM1000E: Exception occurred in the Runtime Task UI. Exception was {0}.

Explanation

None.

User response

None.

Parent topic: [WUTM](#)

[Terms of Use](#) | [Feedback](#)

WUTM1001E

ERROR_MBEAN_ACCESS=WUTM1001E: Exception occurred while trying to access the TaskManagement MBean. Exception was {0}.

Explanation

None.

User response

None.

Parent topic: [WUTM](#)

[Terms of Use](#) | [Feedback](#)

WUTM1002E

ERROR_INVOKE_OPERATION=WUTM1002E: Exception occurred while invoking an operation on the TaskManagement MBean. Exception was {0}.

Explanation

None.

User response

None.

Parent topic: [WUTM](#)

[Terms of Use](#) | [Feedback](#)

WUTM1003E

ERROR_EXECUTE_ACTION=WUTM1003E: Exception occurred while executing an action. Exception was {0}.

Explanation

None.

User response

None.

Parent topic: [WUTM](#)

[Terms of Use](#) | [Feedback](#)

WUTM1004E

INVALID_TASK_ID=WUTM1004E: Task ID was either null or invalid. Exception was {0}.

Explanation

None.

User response

None.

Parent topic: [WUTM](#)

[Terms of Use](#) | [Feedback](#)

WUTM1005E

ERROR_MONITOR_LINK=WUTM1005E: Exception occurred while processing task target monitor link. Exception was {0}.

Explanation

None.

User response

None.

Parent topic: [WUTM](#)

[Terms of Use](#) | [Feedback](#)

WUTM1006E

ERROR_IBM_MESSAGE=WUTM1006E: Exception occurred while processing IBM error messages. Exception was {0}.

Explanation

None.

User response

None.

Parent topic: [WUTM](#)

WVCS

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[WVCS0002W](#)

OLD_WASBG=WVCS0002W: WebSphere background work statistic for node {0} in cell {1} is stale. The WebSphere background work value was last received at {2}, its submission interval is {3} seconds, and is being queried for use at {4}.

[WVCS0003I](#)

CP_FOUND_TIME_FORMAT=WVCS0003I: A cell custom property was found for specifying a format for the timestamps for the data logs, "{0}".

[WVCS0004W](#)

CP_INVALID_TIME_FORMAT=WVCS0004W: A cell custom property was found for specifying a format for the timestamps for the data logs, however the format provided, "{0}", is invalid and can not be used.

[WVCS0005W](#)

INVALID_TIME_FORMAT=WVCS0005W: Specified timestamp for the data logs is invalid, "{0}". Unformatted timestamp will be used.

[WVCS0006I](#)

CP_NEW_TIME_FORMAT=WVCS0006I: Custom cell property was either changed or added, changing the timestamp format for the visualization data logs to {0}

[WVCS0007I](#)

NEW_TIME_FORMAT=WVCS0007I: A new timestamp format for the visualization data logs was specified, changing the timestamp format to {0}

[WVCS0008W](#)

CP_REMOVED_TIME_FORMAT=WVCS0008W: The cell level custom property xd.visengine.timestampformat was either removed or had its value cleared, timestamp formatting will be removed for the visualization data logs.

[WVCS0009W](#)

REMOVED_TIME_FORMAT=WVCS0009W: The timestamp configuration for the visualization data logs was cleared, timestamp formatting will be removed for the visualization data logs.

[WVCS0010I](#)

APPLY_CONFIG_FORMAT=WVCS0010I: Timestamp configuration for the visualization data logs was located, {0}, and will be applied to the visualization data logs.

[WVCS0011I](#)

CP_FORMAT_PRECEDENCE=WVCS0011I: Timestamp configuration for the visualization data logs was located, {0}, however a cell level custom property, xd.visengine.timestampformat was also present and will take precedence unless its value was empty.

[WVCS0012I](#)

UPDATING_WRITER_PROPS=WVCS0012I: Configuration for visualization data service has been modified, updating properties for the data writer writing log {0}.

[WVCS0013I](#)

UPDATING_WRITER_PROP_DT=WVCS0013I: Configuration for visualization data service has been modified and a new data transformer has been specified, updating data transformer for the data writer writing log {0}. Please note that the transformation for the first data logged to the file after this change may not be accurate if data logging is already enabled.

[WVCS0022W](#)

ALPHA_INT_ODC_READ_ERROR=WVCS0022W: An exception, {0}, occurred when trying to read the work profiler interval from the odc with message {1}. Will use default the default interval of {2}.

[WVCS0023W](#)

OLD_BG=WVCS0023W: Background work statistic for node {0} in cell {1} is stale. The background work value was last received at {2}, its submission interval is {3} seconds, and is being queried for use at {4}.

[WVCS0024I](#)

DISABLENONXD=WVCS0024I: Custom property {0} located with a value of {1}. Will not monitor any metrics in the {2} cache pertaining to non-XD nodes or their hosted servers.

[WVCS1000E](#)

WVCS_UNKNOWN_ERROR=WVCS1000E: Exception occurred in the visualization engine. Exception was {0}.

[WVCS1001E](#)

ERROR_CACHE_ACCESS=WVCS1001E: Exception occurred while trying to access the cache service.
Exception was {0}.

WVCS1002E

ERROR_ADMINSERVICE_ACCESS=WVCS1002E: Could not get a handle to the process" administrative service, AdminService instance was null.

WVCS0002W

OLD_WASBG=WVCS0002W: WebSphere background work statistic for node {0} in cell {1} is stale. The WebSphere background work value was last received at {2}, its submission interval is {3} seconds, and is being queried for use at {4}.

Explanation

This could be caused by communication problems between the XD work profiler and the Deployment Manager.

User response

Please monitor the situation and call support if problem persists.

Parent topic: [WVCS](#)

WVCS0003I

CP_FOUND_TIME_FORMAT=WVCS0003I: A cell custom property was found for specifying a format for the timestamps for the data logs, "{0}".

Explanation

None

User response

None

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS0004W

CP_INVALID_TIME_FORMAT=WVCS0004W: A cell custom property was found for specifying a format for the timestamps for the data logs, however the format provided, "{0}", is invalid and can not be used.

Explanation

Please see the javadoc for SimpleDateFormat for valid patterns.

User response

Please correct the pattern and save, the new format will dynamically be used once saved to the master repository.

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS0005W

INVALID_TIME_FORMAT=WVCS0005W: Specified timestamp for the data logs is invalid, "{0}". Unformatted timestamp will be used.

Explanation

Please see the javadoc for SimpleDateFormat for valid patterns.

User response

Please correct the pattern and save, the new format will dynamically be used once saved to the master repository.

Parent topic: [WVCS](#)

WVCS0006I

CP_NEW_TIME_FORMAT=WVCS0006I: Custom cell property was either changed or added, changing the timestamp format for the visualization data logs to {0}

Explanation

None

User response

None

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS0007I

NEW_TIME_FORMAT=WVCS0007I: A new timestamp format for the visualization data logs was specified, changing the timestamp format to {0}

Explanation

None

User response

None

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS0008W

CP_REMOVED_TIME_FORMAT=WVCS0008W: The cell level custom property xd.visengine.timestampformat was either removed or had its value cleared, timestamp formatting will be removed for the visualization data logs.

Explanation

None

User response

None

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS0009W

REMOVED_TIME_FORMAT=WVCS0009W: The timestamp configuration for the visualization data logs was cleared, timestamp formatting will be removed for the visualization data logs.

Explanation

None

User response

None

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS0010I

APPLY_CONFIG_FORMAT=WVCS0010I: Timestamp configuration for the visualization data logs was located, {0}, and will be applied to the visualization data logs.

Explanation

None

User response

None

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS0011I

CP_FORMAT_PRECEDENCE=WVCS0011I: Timestamp configuration for the visualization data logs was located, {0}, however a cell level custom property, xd.visengine.timestampformat was also present and will take precedence unless its value was empty.

Explanation

None

User response

None

Parent topic: [WVCS](#)

WVCS0012I

UPDATING_WRITER_PROPS=WVCS0012I: Configuration for visualization data service has been modified, updating properties for the data writer writing log {0}.

Explanation

None

User response

None

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS0013I

UPDATING_WRITER_PROP_DT=WVCS0013I: Configuration for visualization data service has been modified and a new data transformer has been specified, updating data transformer for the data writer writing log {0}. Please note that the transformation for the first data logged to the file after this change may not be accurate if data logging is already enabled.

Explanation

When a new data transformer is specified while data is actively being logged, the transformer could miss data some data from when the last data was logged and its instantiation.

User response

None

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS0022W

ALPHA_INT_ODC_READ_ERROR=WVCS0022W: An exception, {0}, occurred when trying to read the work profiler interval from the odc with message {1}. Will use default the default interval of {2}.

Explanation

The odc tree could be malformed.

User response

Please enable trace on the dmgr com.ibm.ws.odc.*=all.

Parent topic: [WVCS](#)

WVCS0023W

OLD_BG=WVCS0023W: Background work statistic for node {0} in cell {1} is stale. The background work value was last received at {2}, its submission interval is {3} seconds, and is being queried for use at {4}.

Explanation

This could be caused by communication problems between the XD work profiler and the Deployment Manager.

User response

Please monitor the situation and call support if problem persists.

Parent topic: [WVCS](#)

WVCS0024I

DISABLENONXD=WVCS0024I: Custom property {0} located with a value of {1}. Will not monitor any metrics in the {2} cache pertaining to non-XD nodes or their hosted servers.

Explanation

None

User response

None

Parent topic: [WVCS](#)

[Terms of Use](#) | [Feedback](#)

WVCS1000E

WVCS_UNKNOWN_ERROR=WVCS1000E: Exception occurred in the visualization engine. Exception was {0}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [WVCS](#)

WVCS1001E

ERROR_CACHE_ACCESS=WVCS1001E: Exception occurred while trying to access the cache service. Exception was {0}.

Explanation

The cache service may not have fully initialized.

User response

Please try accessing the cache service again.

Parent topic: [WVCS](#)

WVCS1002E

ERROR_ADMINSERVICE_ACCESS=WVCS1002E: Could not get a handle to the process" administrative service, AdminService instance was null.

Explanation

The AdminService is a core component of the process, the process may not have started up without problems.

User response

Please check the logs for additional errors, call support if problem persists.

Parent topic: [WVCS](#)

WVDS

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

WVDS1000E

WVDS_UNKNOWN_ERROR=WVDS1000E: Exception occurred in the Visualization Data Service. Exception was {0}.

WVDS1001E

ERROR_SAVING_PROPERTIES=WVDS1001E: Encountered an error saving the Visualization Data Service configuration. Exception was {1}.

WVDS1000E

WVDS_UNKNOWN_ERROR=WVDS1000E: Exception occurred in the Visualization Data Service. Exception was {0}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [WVDS](#)

WVDS1001E

ERROR_SAVING_PROPERTIES=WVDS1001E: Encountered an error saving the Visualization Data Service configuration. Exception was {1}.

Explanation

The configuration for the Visualizaton Data Service may contain invalid values.

User response

Please verify the configuration and try saving again.

Parent topic: [WVDS](#)

WXCS

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[WXCS0001E](#)

PC_BBFindErr=WXCS0001E: Container State: unable to find Bulletin Board for {0}. Error message is {1}.

[WXCS0002E](#)

BB_SubscribeFailed=WXCS0002E: Container State: unable to subscribe to topic: cell = {0}, node = {1}, message = {2}.

[WXCS0003E](#)

PC_StartInstance_Err=WXCS0003E: Container State: Failed to launch; DynamicCluster={0}, Node={1}, Member={2} error={3}.

[WXCS0004E](#)

PC_StopInstance_Err=WXCS0004E: Container State: Failed to stop; DynamicCluster={0}, Node={1}, Member={2} error={3}.

WXCS0001E

PC_BBFindErr=WXCS0001E: Container State: unable to find Bulletin Board for {0}. Error message is {1}.

Explanation

Container State could not find the Bulletin Board.

User response

Report problem to IBM.

Parent topic: [WXCS](#)

WXCS0002E

BB_SubscribeFailed=WXCS0002E: Container State: unable to subscribe to topic: cell = {0}, node = {1}, message = {2}.

Explanation

Container State could not subscribe to BB topic.

User response

Report problem to IBM.

Parent topic: [WXCS](#)

WXCS0003E

PC_StartInstance_Err=WXCS0003E: Container State: Failed to launch; DynamicCluster={0}, Node={1}, Member={2} error={3}.

Explanation

Container State experienced an error while starting a server.

User response

Check connectivity, nodeagent may not be started.

Parent topic: [WXCS](#)

WXCS0004E

PC_StopInstance_Err=WXCS0004E: Container State: Failed to stop; DynamicCluster={0}, Node={1}, Member={2} error={3}.

Explanation

Container State experienced an error while stopping a server.

User response

Check connectivity, nodeagent may not be started.

Parent topic: [WXCS](#)

WXDH

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[WXDH0002I](#)

HMC_Starting=WXDH0002I: Starting health controller for {0}.

[WXDH0003I](#)

HMC_Stopping=WXDH0003I: Stopping health controller for {0}.

[WXDH0004I](#)

HMC_AlreadyStarted=WXDH0004I: health controller already started.

[WXDH0005E](#)

HMC_EnableError=WXDH0005E: Exception caught while enabling controller: {0}

[WXDH0006E](#)

HMC_ClassNotFound=WXDH0006E: Health policy not found: {0}

[WXDH0007W](#)

HMC_RmWarn=WXDH0007W: Member {0} not found for health policy {1}.

[WXDH0008E](#)

HMC_LoopErr=WXDH0008E: Error during control loop execution: {0}

[WXDH0009E](#)

HMC_StopErr=WXDH0009E: Error stopping the health control loop: {0}

[WXDH0010W](#)

HMC_SuspectRS=WXDH0010W: Suspect restart state {0} for server {1}.

[WXDH0011W](#)

HMC_MaxVFail=WXDH0011W: Server {0} exceeded max verification failures: disabling restarts.

[WXDH0012W](#)

HMC_ServerDFail=WXDH0012W: Server {0} with restarts disabled failed health check.

[WXDH0014E](#)

HMC_NGNotFound=WXDH0014E: Cannot find node group for node {0}, {1}.

[WXDH0015E](#)

HMC_TgtNotFound=WXDH0015E: Cannot find target member {0} for health policy {1}.

[WXDH0016W](#)

HMC_ConfigErr=WXDH0016W: Error reading configuration repository {0}.

[WXDH0017E](#)

HMC_NoCfgUpd=WXDH0017E: Could not locate the HealthConfigMonitor on the Deployment Manager.

[WXDH0019E](#)

HMC_NoSrvListen=WXDH0019E: Could not start server configuration listener

[WXDH0021E](#)

HMC_CLError=WXDH0021E: Error starting configuration listener: {0}

[WXDH0022E](#)

HMC_CNError=WXDH0022E: Error handling configuration notification: {0}

[WXDH0028E](#)

HMC_DisableError=WXDH0028E: Exception caught while disabling controller: {0}

[WXDH0029I](#)

HMC_RestartsProhibited=WXDH0029I: Server {0} cannot be restarted, restarts are prohibited at this time.

[WXDH0030W](#)

HMC_RestartsDisabled=WXDH0030W: Server {0} cannot be restarted, restarts are disabled.

[WXDH0031I](#)

HMC_MaintenanceMode=WXDH0031I: Server {0} cannot be restarted, the node is in maintenance mode.

[WXDH0032I](#)

HMC_RestartInterval=WXDH0032I: Server {0} cannot be restarted, the last restart occurred too recently.

[WXDH0033W](#)

HMC_ClusterMin=WXDH0033W: Server {0} cannot be restarted, it is the only running member of a cluster.

WXDH0034I

HMC_AddPolicy=WXDH0034I: Health policy {0} has been added.

WXDH0035E

HMC_AddPolicyError=WXDH0035E: Health policy {0} could not be added.

WXDH0036I

HMC_RemovePolicy=WXDH0036I: Health policy {0} has been removed.

WXDH0037I

HMC_ModifyPolicy=WXDH0037I: The {0} for health policy {1} has changed to {2}.

WXDH0038E

HMC_ModifyPolicyError=WXDH0038E: The {0} of health policy {1} could not be changed.

WXDH0039I

HMC_AddMember=WXDH0039I: Member {0} has been added to health policy {1}.

WXDH0040E

HMC_AddMemberError=WXDH0040E: Could not add member {0} to health policy {1}.

WXDH0041I

HMC_RemoveMember=WXDH0041I: Member {0} has been removed from health policy {1}.

WXDH0042I

HMC_AddReaction=WXDH0042I: The {0} reaction has been added to health policy {1}.

WXDH0043I

HMC_RemoveReaction=WXDH0043I: The {0} reaction has been removed from health policy {1}.

WXDH0044W

HMC_RequiresODRData=WXDH0044W: Member {0} of health policy {1} is not adjacent to the ODR.

WXDH0045E

HMC_BBError=WXDH0045E: An error occurred locating the {0} bulletin board: {1}

WXDH0046E

HMC_StartError=WXDH0046E: An error occurred starting the health controller: {0}

WXDH0047E

HMC_BGError=WXDH0047E: An error occurred locating the batch grid scheduler: {0}

WXDH0048I

HMC_BGNoRestart=WXDH0048I: Server {0} cannot be restarted, long-running work cannot be interrupted.

WXDH0049I

HMC_NodeInTransition=WXDH0049I: Server {0} cannot be restarted, its node is moving between long-running and transactional work.

WXDH0050W

HMC_BGUnsupportedPolicy=WXDH0050W: The health condition of policy {0} does not apply to servers performing long-running work.

WXDH0051W

HMC_NonWasUnsupportedPolicy=WXDH0051W: Health policy {0} is not supported for non-WebSphere server {1}.

WXDH1001E

HMA_NoStop=WXDH1001E: Failed to stop server {0}, {1}.

WXDH1002E

HMA_NADown=WXDH1002E: Cannot contact node agent on {0}.

WXDH1003E

HMA_NoStart=WXDH1003E: Failed to start server {0}.

WXDH1004E

HMA_SrvNotFnd=WXDH1004E: Could not find server {0}.

WXDH1005E

HMA_NoHeapDumps=WXDH1005E: Heap dumps are not supported on server {0}.

WXDH1006I

HMA_HeapDumpGenerated=WXDH1006I: A heap dump was automatically generated on server {0} in file {1}.

WXDH1007I

HMA_Restart=WXDH1007I: Issuing restart for server {0}.

WXDH1008I

HMA_RestartSuccess=WXDH1008I: The restart operation for server {0} succeeded.

WXDH1009E

HMA_RestartFailure=WXDH1009E: The restart operation for server {0} failed.

WXDH1010I

HMA_ThreadDump=WXDH1010I: Issuing thread dump for server {0}.

WXDH1011I

HMA_ThreadDumped=WXDH1011I: Issued thread dump for server {0}.

WXDH1012E

HMA_ThreadDumpFailed=WXDH1012E: Thread dump failed for server {0}.

WXDH1013E

HMA_ThreadDumpFailedNoServer=WXDH1013E: Thread dump failed for server {0}, server could not be contacted.

WXDH1014I

HMA_HeapDump=WXDH1014I: Issuing heap dumps for server {0}.

WXDH1015E

HMA_HeapDumpFailed=WXDH1015E: Heap dump failed for server {0}.

WXDH1016E

HMA_HeapDumpFailedNoServer=WXDH1016E: Heap dump failed for server {0}, server could not be contacted.

WXDH1017I

HMA_HeapDumpSuccess=WXDH1017I: The following heap dumps were generated for server {0}: {1}

WXDH1018E

HMA_APCNotFnd=WXDH1018E: Could not contact the placement controller: {0}

WXDH1019E

HMA_APCDisabled=WXDH1019E: Server {0} cannot be restarted, APC disabled and minimum cannot be maintained.

WXDH1020E

HMA_APCNoRestart=WXDH1020E: Server {0} cannot be restarted, cluster minimum cannot be maintained.

WXDH1021W

HMA_PlanInvalid=WXDH1021W: The plan generated for server {0} and condition {1} is not longer valid.

WXDH1022I

HMA_Custom=WXDH1022I: Executing custom task {0} on target {1}.

WXDH1023I

HMA_CustomSuccess=WXDH1023I: Execution of custom task {0} succeeded.

WXDH1024E

HMA_CustomFailed=WXDH1024E: Execution of custom task {0} failed.

WXDH1025I

HMA_MaintMode=WXDH1025I: Issuing setting of maintenance mode for server {0}.

WXDH1026I

HMA_MaintModeSuccess=WXDH1026I: The maintenance mode operation for server {0} succeeded.

WXDH1027E

HMA_MaintModeFailure=WXDH1027E: The maintenance mode operation for server {0} failed.

WXDH2001W

HMS_NoHangDetect=WXDH2001W: Thread hang detection is disabled or its status cannot be obtained for server {0}.

WXDH3001E

HMS_NoData=WXDH3001E: No sensor data received during control cycle from server {0} for health policy {1}.

WXDH3002W

HMS_Age=WXDH3002W: The maximum server age specified by policy {0} was reached by server {1} on node {2}. The limit is {3} hours and the current age is {4} hours.

WXDH3003W

HMS_Wrk=WXDH3003W: The maximum number of requests specified by policy {0} was reached by server {1} on node {2}. The limit is {3} requests and the current work is {4} requests.

WXDH3004W

HMS_Memory=WXDH3004W: The memory consumption limit specified by policy {0} was exceeded by server {1} on node {2}. The limit is {3} % and the current heap size is {4} % of the maximum of {5} KB.

WXDH3005W

HMS_Response=WXDH3005W: The average response time limit specified by policy {0} was exceeded by server {1} on node {2}. The limit is {3} ms and the current value is {4} ms.

WXDH3006W

HMS_Storm_Drain=WXDH3006W: A storm drain condition is suspected by policy {0} for server {1} on node {2}. The average response time has dropped to {3} ms.

WXDH3007W

HMS_Timeout=WXDH3007W: The request timeout limit specified by policy {0} was exceeded by server {1} on node {2}. The limit is {3} % and the current timeout fraction is {4} %.

WXDH3008W

HMS_Leak=WXDH3008W: A memory leak is suspected by policy {0} for server {1} on node {2}.

WXDH3009W

HMS_Servlet_Response=WXDH3009W: A servlet response time change point has been detected by health policy {0} for server {1} on node {2}: Suspected servlets are: {3}.

WXDH3010W

HMS_Response_Warning=WXDH3010W: A server response time change point has been detected for server {0} on node {1}. The average response time has changed from {2} ms to {3} ms.

WXDH3011W

HMS_EJB_Response=WXDH3011W: A response time change point has been detected by health policy {0} for server {1} on node {2} for the average EJB method response time: Suspected EJBs are: {3}.

WXDH3012W

HMS_Timeout_Hang=WXDH3012W: The request timeout limit specified by policy {0} was exceeded by server {1} on node {2}. The limit is {3} % and the current timeout fraction is {4} %. A total of {5} threads may be hung.

WXDH3013W

HMS_Custom=WXDH3013W: A custom health condition violation is suspected by health policy {0} for server {1} on node {2}.

WXDH4001E

TT1_err=WXDH4001E: Error with one parm, {0}

WXDH5000I

HMM_UP=WXDH5000I: Health Management Controller for Extended Deployment is operational

WXDH5001I

HMM_CoreDump=WXDH5001I: The complete set of state for a healthy Health Management Controller is: {0}

WXDH5002E

HMM_CoreDumpErr=WXDH5002E: The complete set of state for an unhealthy Health Management Controller is: {0}

WXDH9999E

HMM_GENLERR=WXDH9999E: Health Controller internal error: data is {0}, and the exception (if any) is {1}.

WXDH0002I

HMC_Starting=WXDH0002I: Starting health controller for {0}.

Explanation

The health management controller is being started.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0003I

HMC_Stopping=WXDH0003I: Stopping health controller for {0}.

Explanation

The health management controller is being stopped.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0004I

HMC_AlreadyStarted=WXDH0004I: health controller already started.

Explanation

Attempt to start an already started health controller.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0005E

HMC_EnableError=WXDH0005E: Exception caught while enabling controller: {0}

Explanation

Something went wrong when the health controller was enabled.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0006E

HMC_ClassNotFound=WXDH0006E: Health policy not found: {0}

Explanation

A health policy was not found.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0007W

HMC_RmWarn=WXDH0007W: Member {0} not found for health policy {1}.

Explanation

Could not find health policy member to remove.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

WXDH0008E

HMC_LoopErr=WXDH0008E: Error during control loop execution: {0}

Explanation

An error occurred during the health control loop execution.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0009E

HMC_StopErr=WXDH0009E: Error stopping the health control loop: {0}

Explanation

An error occurred while stopping the health controller.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0010W

HMC_SuspectRS=WXDH0010W: Suspect restart state {0} for server {1}.

Explanation

The health controller encountered an unexpected state.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

WXDH0011W

HMC_MaxVFail=WXDH0011W: Server {0} exceeded max verification failures: disabling restarts.

Explanation

The server failed verification on restart too many times.

User response

Investigate the cause of the failure.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0012W

HMC_ServerDFail=WXDH0012W: Server {0} with restarts disabled failed health check.

Explanation

The server failed a health check, but restarts are disabled.

User response

Investigate the cause of the failure.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0014E

HMC_NGNotFnd=WXDH0014E: Cannot find node group for node {0}, {1}.

Explanation

Response time cannot be monitored.

User response

Report to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0015E

HMC_TgtNotFound=WXDH0015E: Cannot find target member {0} for health policy {1}.

Explanation

The target was not added to the health policy because it could not be found.

User response

Report configuration problem to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0016W

HMC_ConfigErr=WXDH0016W: Error reading configuration repository {0}.

Explanation

The health controller could not read the configuration repository.

User response

Report configuration problem to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0017E

HMC_NoCfgUpd=WXDH0017E: Could not locate the HealthConfigMonitor on the Deployment Manager.

Explanation

The health controller cannot receive health policy updates.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0019E

HMC_NoSrvListen=WXDH0019E: Could not start server configuration listener

Explanation

The health controller could not enable listening for server configuration changes in {0}.

User response

Internal error; report to IBM.

Parent topic: [WXDH](#)

WXDH0021E

HMC_CLError=WXDH0021E: Error starting configuration listener: {0}

Explanation

An error occurred starting the health configuration listener.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

WXDH0022E

HMC_CNError=WXDH0022E: Error handling configuration notification: {0}

Explanation

An error occurred processing a configuration change.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0028E

HMC_DisableError=WXDH0028E: Exception caught while disabling controller: {0}

Explanation

Something went wrong when the health controller was disabled.

User response

Report program defect to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0029I

HMC_RestartsProhibited=WXDH0029I: Server {0} cannot be restarted, restarts are prohibited at this time.

Explanation

The health controller is configured not to perform restarts at this time.

User response

Information message only, no action required.

Parent topic: [WXDH](#)

WXDH0030W

HMC_RestartsDisabled=WXDH0030W: Server {0} cannot be restarted, restarts are disabled.

Explanation

The server has been restarted the maximum number times consecutively, and a health condition continues to be violated.

User response

If restarts are desired, re-enable them for the server and/or change the configured maximum.

Parent topic: [WXDH](#)

WXDH0031I

HMC_MaintenanceMode=WXDH0031I: Server {0} cannot be restarted, the node is in maintenance mode.

Explanation

The health controller will not restart servers on nodes in maintenance mode.

User response

None required.

Parent topic: [WXDH](#)

WXDH0032I

HMC_RestartInterval=WXDH0032I: Server {0} cannot be restarted, the last restart occurred too recently.

Explanation

The health controller is configured maintain a minimum restart interval.

User response

None required.

Parent topic: [WXDH](#)

WXDH0033W

HMC_ClusterMin=WXDH0033W: Server {0} cannot be restarted, it is the only running member of a cluster.

Explanation

The health controller will not restart the sole running member of a cluster.

User response

Start another server in the cluster.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0034I

HMC_AddPolicy=WXDH0034I: Health policy {0} has been added.

Explanation

A health policy has been added to the configuration repository and read by the health controller.

User response

None required.

Parent topic: [WXDH](#)

WXDH0035E

HMC_AddPolicyError=WXDH0035E: Health policy {0} could not be added.

Explanation

A health policy has been added to the configuration repository, but the health controller could not read it.

User response

Synchronize the configuration repository with all nodes, then disable and re-enable the health controller.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0036I

HMC_RemovePolicy=WXDH0036I: Health policy {0} has been removed.

Explanation

A health policy has been removed from the configuration repository and the health controller.

User response

None required.

Parent topic: [WXDH](#)

WXDH0037I

HMC_ModifyPolicy=WXDH0037I: The {0} for health policy {1} has changed to {2}.

Explanation

The attribute specified for the health policy was modified.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0038E

HMC_ModifyPolicyError=WXDH0038E: The {0} of health policy {1} could not be changed.

Explanation

The attribute of the health policy was modified, but a problem occurred processing the new value.

User response

Synchronize the configuration repository with all nodes, then disable and re-enable the health controller.

Parent topic: [WXDH](#)

WXDH0039I

HMC_AddMember=WXDH0039I: Member {0} has been added to health policy {1}.

Explanation

A member has been added to the health policy.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0040E

HMC_AddMemberError=WXDH0040E: Could not add member {0} to health policy {1}.

Explanation

A member was added to the health policy, but a problem occurred processing the addition.

User response

Synchronize the configuration repository with all nodes, then disable and re-enable the health controller.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0041I

HMC_RemoveMember=WXDH0041I: Member {0} has been removed from health policy {1}.

Explanation

A member has been removed from the health policy.

User response

None required.

Parent topic: [WXDH](#)

WXDH0042I

HMC_AddReaction=WXDH0042I: The {0} reaction has been added to health policy {1}.

Explanation

A reaction has been added to the health policy.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0043I

HMC_RemoveReaction=WXDH0043I: The {0} reaction has been removed from health policy {1}.

Explanation

A reaction has been removed from the health policy.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0044W

HMC_RequiresODRData=WXDH0044W: Member {0} of health policy {1} is not adjacent to the ODR.

Explanation

The health policy requires data generated by the ODR. Only servers in an adjacent tier to the ODR will be represented in its data.

User response

Do not use health policies based on ODR data for this member.

Parent topic: [WXDH](#)

WXDH0045E

HMC_BBError=WXDH0045E: An error occurred locating the {0} bulletin board: {1}

Explanation

The health controller could not locate the bulletin board.

User response

Report problem to IBM.

Parent topic: [WXDH](#)

WXDH0046E

HMC_StartError=WXDH0046E: An error occurred starting the health controller: {0}

Explanation

The health controller could not be started.

User response

Report problem to IBM.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH0047E

HMC_BGError=WXDH0047E: An error occurred locating the batch grid scheduler: {0}

Explanation

The health controller could not locate the scheduler for long running work.

User response

Report problem to IBM.

Parent topic: [WXDH](#)

WXDH0048I

HMC_BGNoRestart=WXDH0048I: Server {0} cannot be restarted, long-running work cannot be interrupted.

Explanation

The server is running long-running work, and the scheduler for this work does not permit the restart.

User response

None required.

Parent topic: [WXDH](#)

WXDH0049I

HMC_NodeInTransition=WXDH0049I: Server {0} cannot be restarted, its node is moving between long-running and transactional work.

Explanation

The node on which the server is running is being reallocated for another type of work, servers will be stopped.

User response

None required.

Parent topic: [WXDH](#)

WXDH0050W

HMC_BGUnsupportedPolicy=WXDH0050W: The health condition of policy {0} does not apply to servers performing long-running work.

Explanation

A policy for transactional work is being applied to a server on a node allocated for long-running work.

User response

Check policy definitions, and ensure that transactional and long-running applications are not deployed to the same targets.

Parent topic: [WXDH](#)

WXDH0051W

HMC_NonWasUnsupportedPolicy=WXDH0051W: Health policy {0} is not supported for non-WebSphere server {1}.

Explanation

A policy that is only supported for WebSphere servers is being applied to a non WebSphere server.

User response

Check policy definitions, and ensure that unsupported health policies are not created for non WebSphere servers.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1001E

HMA_NoStop=WXDH1001E: Failed to stop server {0}, {1}.

Explanation

A server could not be stopped.

User response

Investigate the cause of the failure.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1002E

HMA_NADown=WXDH1002E: Cannot contact node agent on {0}.

Explanation

A node agent cannot be contacted; it may be down.

User response

Check if the node agent is running and restart if necessary.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1003E

HMA_NoStart=WXDH1003E: Failed to start server {0}.

Explanation

A server could not be started.

User response

Investigate the cause of the failure.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1004E

HMA_SrvNotFnd=WXDH1004E: Could not find server {0}.

Explanation

The server instance could not be found.

User response

Investigate the cause of the failure.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1005E

HMA_NoHeapDumps=WXDH1005E: Heap dumps are not supported on server {0}.

Explanation

The server does not support automated heap dumps.

User response

The server must run an IBM JDK on WAS version 6.02 or later.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1006I

HMA_HeapDumpGenerated=WXDH1006I: A heap dump was automatically generated on server {0} in file {1}.

Explanation

A heap dump may be found at the specified location.

User response

Retain this file for off-line analysis and problem determination.

Parent topic: [WXDH](#)

WXDH1007I

HMA_Restart=WXDH1007I: Issuing restart for server {0}.

Explanation

The health controller is automatically restarting the server.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1008I

HMA_RestartSuccess=WXDH1008I: The restart operation for server {0} succeeded.

Explanation

The health controller restarted the server successfully.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1009E

HMA_RestartFailure=WXDH1009E: The restart operation for server {0} failed.

Explanation

The health controller was unable to restart the server.

User response

Check logs for error messages.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1010I

HMA_ThreadDump=WXDH1010I: Issuing thread dump for server {0}.

Explanation

The health controller is automatically taking a thread dump.

User response

None required.

Parent topic: [WXDH](#)

WXDH1011I

HMA_ThreadDumped=WXDH1011I: Issued thread dump for server {0}.

Explanation

The health controller requested a thread dump.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1012E

HMA_ThreadDumpFailed=WXDH1012E: Thread dump failed for server {0}.

Explanation

An error occurred attempting to take a thread dump from the server.

User response

Check logs for further error messages.

Parent topic: [WXDH](#)

WXDH1013E

HMA_ThreadDumpFailedNoServer=WXDH1013E: Thread dump failed for server {0}, server could not be contacted.

Explanation

The server could not be contacted.

User response

Check logs for further error messages.

Parent topic: [WXDH](#)

WXDH1014I

HMA_HeapDump=WXDH1014I: Issuing heap dumps for server {0}.

Explanation

The health controller is automatically taking a series of heap dumps.

User response

None required.

Parent topic: [WXDH](#)

WXDH1015E

HMA_HeapDumpFailed=WXDH1015E: Heap dump failed for server {0}.

Explanation

Either the server could not be contacted, the JVM does not support the heap dump command or the maximum number of heap dumps already exists.

User response

Check logs for further error messages.

Parent topic: [WXDH](#)

WXDH1016E

HMA_HeapDumpFailedNoServer=WXDH1016E: Heap dump failed for server {0}, server could not be contacted.

Explanation

The server could not be contacted.

User response

Check logs for further error messages.

Parent topic: [WXDH](#)

WXDH1017I

HMA_HeapDumpSuccess=WXDH1017I: The following heap dumps were generated for server {0}: {1}

Explanation

The process of taking heap dumps for the server succeeded.

User response

Retain dump files for off-line analysis and problem determination.

Parent topic: [WXDH](#)

WXDH1018E

HMA_APCNotFnd=WXDH1018E: Could not contact the placement controller: {0}

Explanation

The dynamic cluster member cannot be restarted because the placement controller cannot be contacted.

User response

Make sure the placement controller is enabled.

Parent topic: [WXDH](#)

WXDH1019E

HMA_APCDisabled=WXDH1019E: Server {0} cannot be restarted, APC disabled and minimum cannot be maintained.

Explanation

The placement controller is disabled and the minimum number of instances cannot be guaranteed.

User response

Start the placement controller or additional instances.

Parent topic: [WXDH](#)

WXDH1020E

HMA_APCNoRestart=WXDH1020E: Server {0} cannot be restarted, cluster minimum cannot be maintained.

Explanation

The placement controller recommendation would leave too few running server instances.

User response

Start another instance in the dynamic cluster.

Parent topic: [WXDH](#)

WXDH1021W

HMA_PlanInvalid=WXDH1021W: The plan generated for server {0} and condition {1} is not longer valid.

Explanation

Conditions have changed since the plan was generated for the server and condition.

User response

A new plan will be submitted to the task management system for approval.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1022I

HMA_Custom=WXDH1022I: Executing custom task {0} on target {1}.

Explanation

The health controller is executing a custom task.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1023I

HMA_CustomSuccess=WXDH1023I: Execution of custom task {0} succeeded.

Explanation

The execution of the custom task completed successfully.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1024E

HMA_CustomFailed=WXDH1024E: Execution of custom task {0} failed.

Explanation

The execution of the custom task failed to complete successfully.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1025I

HMA_MaintMode=WXDH1025I: Issuing setting of maintenance mode for server {0}.

Explanation

The health controller is automatically placing the server in maintenance mode.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1026I

HMA_MaintModeSuccess=WXDH1026I: The maintenance mode operation for server {0} succeeded.

Explanation

The health controller placed the server into maintenance mode successfully.

User response

None required.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH1027E

HMA_MaintModeFailure=WXDH1027E: The maintenance mode operation for server {0} failed.

Explanation

The health controller was unable to place the server into maintenance mode.

User response

Check logs for error messages.

Parent topic: [WXDH](#)

WXDH2001W

HMS_NoHangDetect=WXDH2001W: Thread hang detection is disabled or its status cannot be obtained for server {0}.

Explanation

Thread hang detection must be enabled to provide advisory information on request timeout policies.

User response

If advisory information regarding thread status is desired, enable hang detection for the server, remove it from the request timeout policy, and add it again.

Parent topic: [WXDH](#)

WXDH3001E

HMS_NoData=WXDH3001E: No sensor data received during control cycle from server {0} for health policy {1}.

Explanation

No sensor data was received from the server.

User response

Make sure PMI is enabled on the server.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH3002W

HMS_Age=WXDH3002W: The maximum server age specified by policy {0} was reached by server {1} on node {2}. The limit is {3} hours and the current age is {4} hours.

Explanation

The maximum allowed server age has been reached.

User response

If the policy is supervised, check runtime tasks to approve configured response.

Parent topic: [WXDH](#)

WXDH3003W

HMS_Work=WXDH3003W: The maximum number of requests specified by policy {0} was reached by server {1} on node {2}. The limit is {3} requests and the current work is {4} requests.

Explanation

The server has reached the maximum number of allowed requests.

User response

If the policy is supervised, check runtime tasks to approve configured response.

Parent topic: [WXDH](#)

WXDH3004W

HMS_Memory=WXDH3004W: The memory consumption limit specified by policy {0} was exceeded by server {1} on node {2}. The limit is {3} % and the current heap size is {4} % of the maximum of {5} KB.

Explanation

The server is approaching its maximum memory limit.

User response

If the policy is supervised, check runtime tasks to approve configured response.

Parent topic: [WXDH](#)

WXDH3005W

HMS_Response=WXDH3005W: The average response time limit specified by policy {0} was exceeded by server {1} on node {2}. The limit is {3} ms and the current value is {4} ms.

Explanation

The server is responding too slowly to incoming requests.

User response

If the policy is supervised, check runtime tasks to approve configured response.

Parent topic: [WXDH](#)

WXDH3006W

HMS_Storm_Drain=WXDH3006W: A storm drain condition is suspected by policy {0} for server {1} on node {2}. The average response time has dropped to {3} ms.

Explanation

A significant decrease in response time and increase in routing weight has been observed on this server

User response

Investigate the cause of the symptoms.

Parent topic: [WXDH](#)

WXDH3007W

HMS_Timeout=WXDH3007W: The request timeout limit specified by policy {0} was exceeded by server {1} on node {2}. The limit is {3} % and the current timeout fraction is {4} %.

Explanation

The server is experiencing a large number of request timeouts.

User response

If the policy is supervised, check runtime tasks to approve configured response.

Parent topic: [WXDH](#)

WXDH3008W

HMS_Leak=WXDH3008W: A memory leak is suspected by policy {0} for server {1} on node {2}.

Explanation

Based on memory usage trends, a memory leak is suspected.

User response

If the policy is supervised, check runtime tasks to approve configured response.

Parent topic: [WXDH](#)

WXDH3009W

HMS_Servlet_Response=WXDH3009W: A servlet response time change point has been detected by health policy {0} for server {1} on node {2}: Suspected servlets are: {3}.

Explanation

A significant change in response time has been observed for one or more servlets

User response

Investigate the cause of the problem.

Parent topic: [WXDH](#)

WXDH3010W

HMS_Response_Warning=WXDH3010W: A server response time change point has been detected for server {0} on node {1}. The average response time has changed from {2} ms to {3} ms.

Explanation

A significant increase in response time has been observed for the server

User response

Investigate the cause of the increase.

Parent topic: [WXDH](#)

WXDH3011W

HMS_EJB_Response=WXDH3011W: A response time change point has been detected by health policy {0} for server {1} on node {2} for the average EJB method response time: Suspected EJBs are: {3}.

Explanation

A significant change in average EJB method response time has been observed for one or more EJBs

User response

Investigate the cause of the problem.

Parent topic: [WXDH](#)

WXDH3012W

HMS_Timeout_Hang=WXDH3012W: The request timeout limit specified by policy {0} was exceeded by server {1} on node {2}. The limit is {3} % and the current timeout fraction is {4} %. A total of {5} threads may be hung.

Explanation

The server is experiencing a large number of request timeouts, and threads may be hung.

User response

If the policy is supervised, check runtime tasks to approve configured response.

Parent topic: [WXDH](#)

WXDH3013W

HMS_Custom=WXDH3013W: A custom health condition violation is suspected by health policy {0} for server {1} on node {2}.

Explanation

A custom health condition violation has been detected.

User response

Restart the server or specify a custom action.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH4001E

TT1_err=WXDH4001E: Error with one parm, {0}

Explanation

exp

User response

ua

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH5000I

HMM_UP=WXDH5000I: Health Management Controller for Extended Deployment is operational

Explanation

an informational message that indicate overall subsystem status

User response

None.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH5001I

HMM_CoreDump=WXDH5001I: The complete set of state for a healthy Health Management Controller is: {0}

Explanation

A detailed dump of all Health Management Controller state.

User response

None.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH5002E

HMM_CoreDumpErr=WXDH5002E: The complete set of state for an unhealthy Health Management Controller is: {0}

Explanation

A detailed dump of all Health Management Controller state.

User response

First review task management for potential remedies, otherwise contact IBM support.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDH9999E

HMM_GENLERR=WXDH9999E: Health Controller internal error: data is {0}, and the exception (if any) is {1}.

Explanation

An unexpected error occurred.

User response

Debug if you can, using the information provided, or contact IBM support.

Parent topic: [WXDH](#)

[Terms of Use](#) | [Feedback](#)

WXDO

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[WXDO0106W](#)

WARNING_HOST_REMOVED_WORKCLASS=WXDO0106W: The removed {0} virtual host is referenced in work classes for the {1} on demand router on the {2} node. Update the following work classes: {3}

[WXDO4023I](#)

INFO_COMPONENT_NAME=WXDO4023I: IBM WebSphere XD Policy Validation

[WXDO5026I](#)

INFO_COMPONENT_NAME=WXDO5026I: IBM WebSphere XD Health Class Validation

[WXDO7901E](#)

WXDO7901E=WXDO7901E: The node has not been added because {0}={1} on the deployment manager node is incompatible with {0}={2} on the node.

[WXDO9006E](#)

WXDO9006E=WXDO9006E: The system failed to uninstall the application because work class actions are configured to permit requests to this application.

WXDO0106W

WARNING_HOST_REMOVED_WORKCLASS=WXDO0106W: The removed {0} virtual host is referenced in work classes for the {1} on demand router on the {2} node. Update the following work classes: {3}

Explanation

The removed virtual host is referenced in on demand router work classes.

User response

Update the offending work classes to reference an existing virtual host.

Parent topic: [WXDO](#)

WXDO4023I

INFO_COMPONENT_NAME=WXDO4023I: IBM WebSphere XD Policy Validation

Explanation

No additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [WXDO](#)

[Terms of Use](#) | [Feedback](#)

WXDO5026I

INFO_COMPONENT_NAME=WXDO5026I: IBM WebSphere XD Health Class Validation

Explanation

No additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [WXDO](#)

[Terms of Use](#) | [Feedback](#)

WXDO7901E

WXDO7901E=WXDO7901E: The node has not been added because {0}={1} on the deployment manager node is incompatible with {0}={2} on the node.

Explanation

The node has a product installed which is either not on the deployment manager node, or is at a lower level on the deployment manager node.

User response

Install the same level of the product on the deployment manager node before attempting to add the managed node.

Parent topic: [WXDO](#)

WXDO9006E

WXDO9006E=WXDO9006E: The system failed to uninstall the application because work class actions are configured to permit requests to this application.

Explanation

The system failed to uninstall the application because work class actions are configured to permit requests to this application.

User response

Reconfigure work class actions to avoid permitting requests to this application.

Parent topic: [WXDO](#)

WXDV

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

WXDV1001E

ERROR_UPDATE_VISCONFIG=WXDV1001E: Unexpected error encountered during update of the Visualization Data Service configuration. The error was {0}.

WXDV1002E

ERROR_LOADING_VISCONFIG=WXDV1002E: Unexpected error encountered during loading of the Visualization Data Service configuration. The error was {0}.

WXDV1001E

ERROR_UPDATE_VISCONFIG=WXDV1001E: Unexpected error encountered during update of the Visualization Data Service configuration. The error was {0}.

Explanation

The configuration for the Visualizaton Data Service may contain invalid values.

User response

Please verify the configuration and try saving again.

Parent topic: [WXDV](#)

WXDV1002E

ERROR_LOADING_VISCONFIG=WXDV1002E: Unexpected error encountered during loading of the Visualization Data Service configuration. The error was {0}.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

Please log out of the administrative console and try accessing the configuration again.

Parent topic: [WXDV](#)

WXMI

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[WXMI0000E](#)

WXMI0000E=WXMI0000E: Could not find text for message ID {0}.

[WXMI0001I](#)

WXMI0001I=WXMI0001I: Migrating {0} configuration from {1} to {2}.

[WXMI0002I](#)

WXMI0002I=WXMI0002I: Migrating from {0}

[WXMI0003I](#)

WXMI0003I=WXMI0003I: Migrating to {0}

[WXMI0004I](#)

WXMI0004I=WXMI0004I: Reading actions from {0}

[WXMI0005I](#)

WXMI0005I=WXMI0005I: Logging level is {0}

[WXMI0006I](#)

WXMI0006I=WXMI0006I: Logging to {0}

[WXMI0007I](#)

WXMI0007I=WXMI0007I: Set variable {0} = {1}

[WXMI0008I](#)

WXMI0008I=WXMI0008I: Executing {0} - action {1} of {2}

[WXMI0009I](#)

WXMI0009I=WXMI0009I: End of {0} migration. RC={1}

[WXMI0010I](#)

WXMI0010I=WXMI0010I: {0}

[WXMI0011I](#)

WXMI0011I=WXMI0011I: {0} has already been migrated in profile {1}

[WXMI0012I](#)

WXMI0012I=WXMI0012I: No pending migration actions found.

[WXMI2001I](#)

WXMI2001I=WXMI2001I: Backing up configuration located at {0} for profile {1}.

[WXMI2002I](#)

WXMI2002I=WXMI2002I: Removing backup of configuration for profile {0} .

[WXMI2003I](#)

WXMI2003I=WXMI2003I: {0}

[WXMI2004I](#)

WXMI2004I=WXMI2004I: XDPreUpgrade Setting variable {0} = {1}

[WXMI2005I](#)

WXMI2005I=WXMI2005I: Executing {0} - action {1} of {2}

[WXMI2006I](#)

WXMI2006I=WXMI2006I: Reading actions from {0}

[WXMI2007I](#)

WXMI2007I=WXMI2007I: Logging level is {0}

[WXMI2008I](#)

WXMI2008I=WXMI2008I: Logging to {0}

[WXMI2009I](#)

WXMI2009I=WXMI2009I: End of backup of configuration for profile. RC={1}

[WXMI2010I](#)

WXMI2010I=WXMI2010I: End of removal of configuration backup for profile. RC={1}

[WXMI9000E](#)

WXMI9000E=WXMI9000E: An error occurred during migration. See {0}.

[WXMI9001E](#)

WXMI9001E=WXMI9001E: Option {0} is unknown.

[WXMI9002E](#)

WXMI9002E=WXMI9002E: Parameter required for option: {0}

[WXMI9003E](#)

WXMI9003E=WXMI9003E: Parameter on {0} option is not valid.

[WXMI9004E](#)

WXMI9004E=WXMI9004E: Directory {0} is not a valid WebSphere profile.

[WXMI9005E](#)

WXMI9005E=WXMI9005E: Profile {0} contains more than one cell.

[WXMI9006E](#)

WXMI9006E=WXMI9006E: Profile {0} does not contain a WebSphere cell.

[WXMI9007E](#)

WXMI9007E=WXMI9007E: Profile {0} does not contain a dmgr node.

[WXMI9008E](#)

WXMI9008E=WXMI9008E: An error occurred during backup of configuration. See {0}.

[WXMI9009E](#)

WXMI9009E=WXMI9009E: An error occurred during the removal of a configuration backup. See {0}.

[WXMI9010E](#)

WXMI9010E=WXMI9010E: Source cell name {0} does not match target cell name {1}.

[WXMI9011E](#)

WXMI9011E=WXMI9011E: Source node name {0} does not match target node name {1}.

[WXMI9012E](#)

WXMI9012E=WXMI9012E: Source Node {0} contains a DMGR and target node {1} does not.

[WXMI9013E](#)

WXMI9013E=WXMI9013E: Target Node {0} contains a DMGR and source node {1} does not.

[WXMI9014E](#)

WXMI9014E=WXMI9014E: WebSphere Extended Deployment is not installed in the Source install {0}.

[WXMI9015E](#)

WXMI9015E=WXMI9015E: {0} does not contain a valid WebSphere installation.

[WXMI9016E](#)

WXMI9016E=WXMI9016E: When entering userid and password, both parameters must be entered.

WXMI0000E

WXMI0000E=WXMI0000E: Could not find text for message ID {0}.

Explanation

WebSphere could not find the message text for the indicated message ID.

User response

Open a problem report with IBM service.

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0001I

WXMI0001I=WXMI0001I: Migrating {0} configuration from {1} to {2}.

Explanation

Migration of a WebSphere Extended Deployment configuration has been initiated by the user.

User response

None

Parent topic: [WXM](#)

WXMI0002I

WXMI0002I=WXMI0002I: Migrating from {0}

Explanation

Indicates the install directory from which a WebSphere Extended Deployment configuration is being migrated.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0003I

WXMI0003I=WXMI0003I: Migrating to {0}

Explanation

Indicates the profile directory to which the WebSphere Extended Deployment configuration is being migrated.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0004I

WXMI0004I=WXMI0004I: Reading actions from {0}

Explanation

Indicates the directory from which migration actions are being read.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0005I

WXMI0005I=WXMI0005I: Logging level is {0}

Explanation

Indicates the logging level being used.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0006I

WXMI0006I=WXMI0006I: Logging to {0}

Explanation

Indicates location and name of XDUpgrade log.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0007I

WXMI0007I=WXMI0007I: Set variable {0} = {1}

Explanation

The indicated XDUpgrade variable {0} is set to the indicated value {1}

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0008I

WXMI0008I=WXMI0008I: Executing {0} - action {1} of {2}

Explanation

XDUpgrade is executing action {0}, which is action {1} of a total of {2}.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0009I

WXMI0009I=WXMI0009I: End of {0} migration. RC={1}

Explanation

Migration of a WebSphere Extended Deployment configuration has completed with the indicated return code.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0010I

WXMI0010I=WXMI0010I: {0}

Explanation

Displays correct syntax for XDUpgrade command.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI0011I

WXMI0011I=WXMI0011I: {0} has already been migrated in profile {1}

Explanation

The specified package has already been migrated.

User response

None

Parent topic: [WXM](#)

[Terms of Use](#) | [Feedback](#)

WXMI0012I

WXMI0012I=WXMI0012I: No pending migration actions found.

Explanation

XDUgrade was invoked, but no packages that require migration were found.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI2001I

WXMI2001I=WXMI2001I: Backing up configuration located at {0} for profile {1}.

Explanation

Backup of configuration been initiated by the user.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI2002I

WXMI2002I=WXMI2002I: Removing backup of configuration for profile {0} .

Explanation

Removal of a backup of configuration been initiated by the user.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI2003I

WXMI2003I=WXMI2003I: {0}

Explanation

Displays correct syntax for XDPreUpgrade command.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI2004I

WXMI2004I=WXMI2004I: XDPreUpgrade Setting variable {0} = {1}

Explanation

XDPreUpgrade has set the specified variable to the specified value

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI2005I

WXMI2005I=WXMI2005I: Executing {0} - action {1} of {2}

Explanation

XDPreUpgrade is executing action {0}, which is action {1} of a total of {2}.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI2006I

WXMI2006I=WXMI2006I: Reading actions from {0}

Explanation

Indicates the directory from which migration actions are being read.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI2007I

WXMI2007I=WXMI2007I: Logging level is {0}

Explanation

Indicates the logging level being used.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI2008I

WXMI2008I=WXMI2008I: Logging to {0}

Explanation

Indicates location and name of XDUpgrade log.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI2009I

WXMI2009I=WXMI2009I: End of backup of configuration for profile. RC={1}

Explanation

Backup of a WebSphere Extended Deployment configuration has completed with the indicated return code.

User response

None

Parent topic: [WXMI](#)

WXMI2010I

WXMI2010I=WXMI2010I: End of removal of configuration backup for profile. RC={1}

Explanation

Removal of a WebSphere Extended Deployment configuration backup has completed with the indicated return code.

User response

None

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI9000E

WXMI9000E=WXMI9000E: An error occurred during migration. See {0}.

Explanation

An error occurred during execution of XDUpgrade. Details of the problem should show in the XDUpgrade log file.

User response

Contact IBM Service.

Parent topic: [WXMI](#)

WXMI9001E

WXMI9001E=WXMI9001E: Option {0} is unknown.

Explanation

Command issued with incorrect options.

User response

Reissue command with correct options.

Parent topic: [WXMJ](#)

WXMI9002E

WXMI9002E=WXMI9002E: Parameter required for option: {0}

Explanation

Required parameter not specified for indicated option.

User response

Reissue command with required parameter.

Parent topic: [WXMJ](#)

[Terms of Use](#) | [Feedback](#)

WXMI9003E

WXMI9003E=WXMI9003E: Parameter on {0} option is not valid.

Explanation

Command issued with parameter that was not valid.

User response

Reissue command with valid parameter.

Parent topic: [WXMI](#)

WXMI9004E

WXMI9004E=WXMI9004E: Directory {0} is not a valid WebSphere profile.

Explanation

Specified profile name is not an actual WebSphere profile.

User response

Review available profile names and reissue command with a valid profile name.

Parent topic: [WXMJ](#)

[Terms of Use](#) | [Feedback](#)

WXMI9005E

WXMI9005E=WXMI9005E: Profile {0} contains more than one cell.

Explanation

Specified profile contains more than one cell.

User response

This is not a valid WebSphere configuration. Remove cell directories from profile until there is only one cell and reissue command.

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI9006E

WXMI9006E=WXMI9006E: Profile {0} does not contain a WebSphere cell.

Explanation

Specified profile does not contain a WebSphere cell.

User response

Review available profile names and reissue command with a valid profile name.

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI9007E

WXMI9007E=WXMI9007E: Profile {0} does not contain a dmgr node.

Explanation

Specified profile is not a deployment manager profile.

User response

Review available profile names and reissue command with a valid deployment manager profile name.

Parent topic: [WXMI](#)

WXMI9008E

WXMI9008E=WXMI9008E: An error occurred during backup of configuration. See {0}.

Explanation

An error occurred during execution of XDPreUpgrade. Details of the problem should show in the XDPreUpgrade log file.

User response

Contact IBM Service.

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI9009E

WXMI9009E=WXMI9009E: An error occurred during the removal of a configuration backup. See {0}.

Explanation

An error occurred during execution of XDPreUpgrade. Details of the problem should show in the XDPreUpgrade log file.

User response

Contact IBM Service.

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI9010E

WXMI9010E=WXMI9010E: Source cell name {0} does not match target cell name {1}.

Explanation

The target and source cell names are different.

User response

Determine correct cell and reissue command.

Parent topic: [WXMJ](#)

[Terms of Use](#) | [Feedback](#)

WXMI9011E

WXMI9011E=WXMI9011E: Source node name {0} does not match target node name {1}.

Explanation

The target and source node names are different.

User response

Determine correct nodes and reissue command.

Parent topic: [WXMJ](#)

[Terms of Use](#) | [Feedback](#)

WXMI9012E

WXMI9012E=WXMI9012E: Source Node {0} contains a DMGR and target node {1} does not.

Explanation

The target and source node names are different.

User response

Determine correct nodes and reissue command.

Parent topic: [WXMJ](#)

[Terms of Use](#) | [Feedback](#)

WXMI9013E

WXMI9013E=WXMI9013E: Target Node {0} contains a DMGR and source node {1} does not.

Explanation

The target and source node names are different.

User response

Determine correct nodes and reissue command.

Parent topic: [WXMJ](#)

[Terms of Use](#) | [Feedback](#)

WXMI9014E

WXMI9014E=WXMI9014E: WebSphere Extended Deployment is not installed in the Source install {0}.

Explanation

WebSphere Extended Deployment is not installed in the Source install.

User response

Determine correct source install location.

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI9015E

WXMI9015E=WXMI9015E: {0} does not contain a valid WebSphere installation.

Explanation

WebSphere Extended Deployment is not installed in the Source install.

User response

Determine correct source install location.

Parent topic: [WXMI](#)

[Terms of Use](#) | [Feedback](#)

WXMI9016E

WXMI9016E=WXMI9016E: When entering userid and password, both parameters must be entered.

Explanation

Either the userid or the password parameter was not found.

User response

Enter correct userid and password parameters.

Parent topic: [WXMI](#)

XADC

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[XADC0001E](#)

XADC0001=XADC0001E: Unrecognized property: http://apache.org/xml/properties/schema/external-schemaLocation. Exception = {0}

[XADC0002E](#)

XADC0002=XADC0002E: Error while parsing file. Exception = {0}

[XADC0003E](#)

XADC0003=XADC0003E: Error while adding Server to DefaultCoreGroup after server additon. Exception = {0}

[XADC0004E](#)

XADC0004=XADC0004E: Core Group Type Not found, initialization of CoreGroupMonitor failed

[XADC0005E](#)

XADC0005=XADC0005E: Exception while handling server addition. Exception = {0}

[XADC0006E](#)

XADC0006=XADC0006E: Exception while handling server deletion. Exception = {0}

[XADC0007E](#)

XADC0007=XADC0007E: Exception while creating default core group during initalization. Exception = {0}

[XADC0008E](#)

XADC0008=XADC0008E: Exception while intializing default core group during initalization. Exception = {0}

[XADC0009E](#)

XADC0009=XADC0009E: Dyanmic Cluster with supplied node group name and dynamic cluster name pair was not found. Supplied node group name = {0}. Supplied dynamic cluster name = {1}.

[XADC0010E](#)

XADC0010=XADC0010E: Exception while creating cluster member on node. Exception = {0}

[XADC0011E](#)

XADC0011=XADC0011E: Exception while creating Dynamic Cluster. Exception = {0}

[XADC0012E](#)

XADC0012=XADC0012E: Exception encountered while deleting Dynamic Cluster. Exception = {0}

[XADC0013E](#)

XADC0013=XADC0013E: Exception while deleting cluster member on node. Exception = {0}

[XADC0014E](#)

XADC0014=XADC0014E: Exception while Retrieving Dynamic Cluster resource in workspace. Exception = {0}

[XADC0015E](#)

XADC0015=XADC0015E: Exception Activating MBean. Exception = {0}

[XADC0016E](#)

XADC0016=XADC0016E: Exception Sending Notification. Exception = {0}

[XADC0017E](#)

XADC0017=XADC0017E: Exception Getting Top Level WCCM Object from Repository with URI={0}. Exception = {1}

[XADC0018E](#)

XADC0018=XADC0018E: Exception while syncing master repository with active node agents. Exception = {0}

[XADC0019E](#)

XADC0019=XADC0019E: Exception while getting dynamic cluster members for deletion. Exception = {0}

[XADC0020E](#)

XADC0020=XADC0020E: Repository Exception while deleting Repository documents. Exception = {0}

[XADC0021E](#)

XADC0021=XADC0021E: Exception creating dynamic cluster template. Exception = {0}

[XADC0022E](#)

XADC0022=XADC0022E: Exception creating static cluster for dynamic cluster. Exception = {0}

XADC0023E

XADC0023=XADC0023E: Error disabling Placement Controller. Exception = {0}

XADC0024E

XADC0024=XADC0024E: Error merging cluster template with cluster member document. Exception = {0}

XADC0025E

XADC0025=XADC0025E: Error getting MBean ObjectName from adminclient={1}. Exception = {0}

XADC0026E

XADC0026=XADC0026E: While handling Cluster added notification an Exception occurred. Exception = {0}

XADC0027E

XADC0027=XADC0027E: While handling Cluster modification notification an Exception occurred. Exception = {0}

XADC0028E

XADC0028=XADC0028E: Caught Throwable exception while notifying consumers. Throwable = {0}

XADC0029E

XADC0029=XADC0029E: MalformedObjectNameException creating ObjectName = WebSphere:*,type=Server,processType=ManagedProcess,node= {0}

XADC0030E

XADC0030=XADC0030E: Error getting AdminClient URI for node= {0}. Exception = {1}

XADC0031I

XADC0031=XADC0031I: Registered Listener for ConfigRepository Changes.

XADC0032E

XADC0032=XADC0032E: Dynamic cluster {0} cannot be deleted because it is not in manual mode. Change the operational mode to Manual and try this operation again

XADC0033E

XADC0033=XADC0033E: Dynamic cluster {0} cannot be deleted because applications are installed on it. Uninstall the applications and try this operation again

XADC0034E

XADC0034=XADC0034E: Dynamic cluster {0} cannot be deleted because it has active members instances. Stop dynamic cluster members instances and try this operation again

XADC0035I

DISABLECGBCONFIG=XADC0035I: Custom property {0} located with a value of {1}. Core group bridge auto-configuration is now disabled.

XADC0001E

XADC0001=XADC0001E: Unrecognized property: http://apache.org/xml/properties/schema/external-schemaLocation. Exception = {0}

Explanation

Error while setting up document

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0002E

XADC0002=XADC0002E: Error while parsing file. Exception = {0}

Explanation

Error while parsing xml file

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0003E

XADC0003=XADC0003E: Error while adding Server to DefaultCoreGroup after server additon. Exception = {0}

Explanation

PackageNotFoundException while handling addition of server to cell.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0004E

XADC0004=XADC0004E: Core Group Type Not found, initialization of CoreGroupMonitor failed

Explanation

Core Group Type Not found, initialization CoreGroupMonitor failed

User response

Report Problem To IBM

Parent topic: [XADC](#)

XADC0005E

XADC0005=XADC0005E: Exception while handling server addition. Exception = {0}

Explanation

A server has been added to the config repository and while handling that event an exception occurred

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0006E

XADC0006=XADC0006E: Exception while handling server deletion. Exception = {0}

Explanation

A server has been deleted from the config repository and while handling that event an exception occurred

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0007E

XADC0007=XADC0007E: Exception while creating default core group during initialization. Exception = {0}

Explanation

During initialization no default core group was found. While attempting to create it and an exception was thrown.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0008E

XADC0008=XADC0008E: Exception while intializing default core group during initalization. Exception = {0}

Explanation

Exception while intializing default core group during initalization

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0009E

XADC0009=XADC0009E: Dyanmic Cluster with supplied node group name and dynamic cluster name pair was not found. Supplied node group name = {0}. Supplied dynamic cluster name = {1}.

Explanation

Could not found dynamic cluster in repository

User response

Ensure that dynamic cluster exists

Parent topic: [XADC](#)

XADC0010E

XADC0010=XADC0010E: Exception while creating cluster member on node. Exception = {0}

Explanation

Exception while creating cluster member on node.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0011E

XADC0011=XADC0011E: Exception while creating Dynamic Cluster. Exception = {0}

Explanation

Exception while creating Dynamic Cluster.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0012E

XADC0012=XADC0012E: Exception encountered while deleting Dynamic Cluster. Exception = {0}

Explanation

Exception encountered while deleting Dynamic Cluster.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0013E

XADC0013=XADC0013E: Exception while deleting cluster member on node. Exception = {0}

Explanation

Exception while deleting cluster member on node.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0014E

XADC0014=XADC0014E: Exception while Retrieving Dynamic Cluster resource in workspace. Exception = {0}

Explanation

Exception while Retrieving Dynamic Cluster resource in workspace.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0015E

XADC0015=XADC0015E: Exception Activating MBean. Exception = {0}

Explanation

During Dynamic Cluster Configuration Manager initialization, there was an error activating MBean.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0016E

XADC0016=XADC0016E: Exception Sending Notification. Exception = {0}

Explanation

Exception while Sending Dynamic Cluster Notification

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0017E

XADC0017=XADC0017E: Exception Getting Top Level WCCM Object from Repository with URI={0}. Exception = {1}

Explanation

Exception Getting Top Level WCCM Object from Repository

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0018E

XADC0018=XADC0018E: Exception while syncing master repository with active node agents. Exception = {0}

Explanation

Exception while syncing master repository with active node agents.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0019E

XADC0019=XADC0019E: Exception while getting dynamic cluster members for deletion. Exception = {0}

Explanation

Exception while getting dynamic cluster members for deletion

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0020E

XADC0020=XADC0020E: Repository Exception while deleting Repository documents. Exception = {0}

Explanation

Repository Exception while deleting Repository documents

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0021E

XADC0021=XADC0021E: Exception creating dynamic cluster template. Exception = {0}

Explanation

Exception creating dynamic cluster template.

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0022E

XADC0022=XADC0022E: Exception creating static cluster for dynamic cluster. Exception = {0}

Explanation

Exception creating static cluster for dynamic cluster

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0023E

XADC0023=XADC0023E: Error disabling Placement Controller. Exception = {0}

Explanation

Error disabling Placement Controller

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0024E

XADC0024=XADC0024E: Error merging cluster template with cluster member document. Exception = {0}

Explanation

Error merging cluster template with cluster member document

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0025E

XADC0025=XADC0025E: Error getting MBean ObjectName from adminclient={1}. Exception = {0}

Explanation

Error getting MBean ObjectName from adminclient

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0026E

XADC0026=XADC0026E: While handling Cluster added notification an Exception occurred. Exception = {0}

Explanation

While handling Cluster added notification an Exception occurred

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0027E

XADC0027=XADC0027E: While handling Cluster modification notification an Exception occurred. Exception = {0}

Explanation

While handling Cluster modification notification an Exception occurred

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0028E

XADC0028=XADC0028E: Caught Throwable exception while notifying consumers. Throwable = {0}

Explanation

Caught Throwable exception while notifying consumers

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0029E

XADC0029=XADC0029E: MalformedObjectNameException creating ObjectName =
WebSphere:*,type=Server,processType=ManagedProcess,node= {0}

Explanation

MalformedObjectNameException creating ObjectName =
WebSphere:*,type=Server,processType=ManagedProcess,node= {0}

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0030E

XADC0030=XADC0030E: Error getting AdminClient URI for node= {0}. Exception = {1}

Explanation

Error getting AdminClient URI for node= {0}. Exception = {1}

User response

Report Problem To IBM

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0031I

XADC0031=XADC0031I: Registered Listener for ConfigRepository Changes.

Explanation

Registered Listener for ConfigRepository Changes.

User response

None

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XADC0032E

XADC0032=XADC0032E: Dynamic cluster {0} cannot be deleted because it is not in manual mode. Change the operational mode to Manual and try this operation again

Explanation

Dynamic cluster cannot be deleted because it is not in manual mode.

User response

Change the operational mode to Manual before deleting the dynamic cluster.

Parent topic: [XADC](#)

XADC0033E

XADC0033=XADC0033E: Dynamic cluster {0} cannot be deleted because applications are installed on it. Uninstall the applications and try this operation again

Explanation

If a dynamic cluster is deleted before the installed applications are uninstalled, those applications fall into an inconsistent state and might not run normally.

User response

Uninstall the applications on the dynamic cluster before deleting the dynamic cluster.

Parent topic: [XADC](#)

XADC0034E

XADC0034=XADC0034E: Dynamic cluster {0} cannot be deleted because it has active members instances. Stop dynamic cluster members instances and try this operation again

Explanation

Dynamic cluster cannot be deleted because it has active members instances.

User response

Stop dynamic cluster members instances before deleting the dynamic cluster.

Parent topic: [XADC](#)

XADC0035I

DISABLECGBCONFIG=XADC0035I: Custom property {0} located with a value of {1}. Core group bridge auto-configuration is now disabled.

Explanation

Core group bridge auto configuration has been disabled via a custom property.

User response

None.

Parent topic: [XADC](#)

[Terms of Use](#) | [Feedback](#)

XBGA

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

XBGA0100E

ERROR_INVALID_DATASOURCE_JNDI_NAME=XBGA0100E: The datasource JNDIName specified does not exist.

XBGA0101E

ERROR_INVALID_DATASOURCE_ALIAS=XBGA0101E: The datasource alias specified does not exist.

XBGA0102E

ERROR_CROSSVALIDATING_DATA=XBGA0102E: Unable to cross-validate data specified for long-running scheduler.

XBGA0450E

XBGA0450E=XBGA0450E: The long-running scheduler has not been configured yet.

XBGA0451E

XBGA0451E=XBGA0451E: The attribute name {0} is not a valid long-running scheduler attribute.

XBGA0452E

XBGA0452E=XBGA0452E: The property name {0} is not a valid long-running scheduler custom property.

XBGA0453E

XBGA0453E=XBGA0453E: Custom property {0} already exists for long-running scheduler.

XBGA0454E

XBGA0454E=XBGA0454E: Encountered error in getting {0} value. Exception = {1}

XBGA0700I

XBGA0700I=XBGA0700I: The job scheduler has successfully been deployed to {0} using the datasource {1}.

XBGA0710E

XBGA0710E=XBGA0710E: Error encountered when attempting to deploy the job scheduler to {0} using the datasource {1}. Exception = {2}

XBGA0720E

XBGA0720E=XBGA0720E: An attempt has been made to deploy the job scheduler to the cluster {1} using the embedded Derby datasource {0}.

XBGA0730E

XBGA0730E=XBGA0730E: An attempt has been made to deploy the job scheduler to {1} without the specification of a datasource.

XBGA0740I

XBGA0740I=XBGA0740I: The job scheduler has successfully been removed from {0}.

XBGA0750E

XBGA0750E=XBGA0750E: Error encountered when attempting to remove the job scheduler from {0}. Exception = {1}.

XBGA0751E

XBGA0751E=XBGA0751E: Error encountered when attempting to remove the job scheduler from {0}. Exception = {1}.

XBGA0760I

XBGA0760I=XBGA0760I: The WebSphere endpoint {0} has successfully been redeployed using the datasource {1}.

XBGA0770E

XBGA0770E=XBGA0770E: Error encountered when attempting to redeploy WebSphere endpoint {0} to use datasource {1}. Exception = {1}.

XBGA0780E

XBGA0780E=XBGA0780E: An attempt has been made to redeploy the WebSphere endpoint cluster {1} using the embedded Derby datasource {0}.

XBGA0790E

XBGA0790E=XBGA0790E: Error encountered when attempting to redeploy WebSphere endpoint {0} to use datasource {1}. Exception = {1}.

XBGA0800E

XBGA0800E=XBGA0800E: Error encountered when attempting to configure the job scheduler.

[XBGA0810E](#)

XBGA0810E=XBGA0810E: Error encountered when attempting to redeploy the WebSphere endpoint {0}.

[Terms of Use](#) | [Feedback](#)

XBGA0100E

ERROR_INVALID_DATASOURCE_JNDI_NAME=XBGA0100E: The datasource JNDIName specified does not exists.

Explanation

The datasource JNDIName specified is not a valid resource on the system.

User response

Make sure the datasource is created before configuring the long-running scheduler.

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGA0101E

ERROR_INVALID_DATASOURCE_ALIAS=XBGA0101E: The datasource alias specified does not exists.

Explanation

The datasource alias specified is not a valid resource on the system.

User response

Make sure the alias is created before configuring the long-running scheduler.

Parent topic: [XBGA](#)

XBGA0102E

ERROR_CROSSVALIDATING_DATA=XBGA0102E: Unable to cross-validate data specified for long-running scheduler.

Explanation

User response

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGA0450E

XBGA0450E=XBGA0450E: The long-running scheduler has not been configured yet.

Explanation

No long-running scheduler attributes or properties have been set.

User response

Configure long-running scheduler by using `modifyLongRunningSchedulerAttribute` to set attribute or `createLongRunningSchedulerProperty` to create custom property.

Parent topic: [XBGA](#)

XBGA0451E

XBGA0451E=XBGA0451E: The attribute name {0} is not a valid long-running scheduler attribute.

Explanation

The attribute name specified is not a valid long-running scheduler attribute.

User response

Try the operation again by selecting a valid attribute to create or modify. Valid attributes are datasourceJNDIName and datasourceAlias.

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGA0452E

XBGA0452E=XBGA0452E: The property name {0} is not a valid long-running scheduler custom property.

Explanation

The property name specified is not a valid long-running scheduler property.

User response

Try the operation again by selecting a valid property. Use `listLongRunningSchedulerProperties` to list all available properties.

Parent topic: [XBGA](#)

XBGA0453E

XBGA0453E=XBGA0453E: Custom property {0} already exists for long-running scheduler.

Explanation

A long-running scheduler property with the specified name is already created.

User response

Choose another name for the long-running scheduler property, and try the operation again.

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGA0454E

XBGA0454E=XBGA0454E: Encountered error in getting {0} value. Exception = {1}

Explanation

Unable to retrieve {0} value due to some errors.

User response

Check to see if the long-running scheduler has been configured.

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGA0700I

XBGA0700I=XBGA0700I: The job scheduler has successfully been deployed to {0} using the datasource {1}.

Explanation

The job scheduler has been configured to run on the specified deployment target using the datasource selected.

User response

Recycle job scheduler deployment target.

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGA0710E

XBGA0710E=XBGA0710E: Error encountered when attempting to deploy the job scheduler to {0} using the datasource {1}. Exception = {2}

Explanation

This is an internal configuration problem.

User response

Contact IBM Support.

Parent topic: [XBGA](#)

XBGA0720E

XBGA0720E=XBGA0720E: An attempt has been made to deploy the job scheduler to the cluster {1} using the embedded Derby datasource {0}.

Explanation

A network datasource is required to deploy the job scheduler to a cluster.

User response

Create and select a datasource for a network database.

Parent topic: [XBGA](#)

XBGA0730E

XBGA0730E=XBGA0730E: An attempt has been made to deploy the job scheduler to {1} without the specification of a datasource.

Explanation

A datasource selection is required to deploy the job scheduler.

User response

Create and select a valid datasource.

Parent topic: [XBGA](#)

XBGA0740I

XBGA0740I=XBGA0740I: The job scheduler has successfully been removed from {0}.

Explanation

The job scheduler has been uninstalled and taken out of service.

User response

Recycle all job scheduler servers to complete the uninstall operation.

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGA0750E

XBGA0750E=XBGA0750E: Error encountered when attempting to remove the job scheduler from {0}.
Exception = {1}.

Explanation

This is an internal configuration problem.

User response

Contact IBM Support.

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGA0751E

XBGA0751E=XBGA0751E: Error encountered when attempting to remove the job scheduler from {0}.
Exception = {1}.

Explanation

The job scheduler uninstall operation has reported error(s).

User response

Ensure that the server or cluster hosting the job scheduler has stopped.

Parent topic: [XBGA](#)

XBGA0760I

XBGA0760I=XBGA0760I: The WebSphere endpoint {0} has successfully been redeployed using the datasource {1}.

Explanation

The WebSphere endpoint has been re-configured to use the selected datasource.

User response

Recycle redeployed endpoint.

Parent topic: [XBGA](#)

XBGA0770E

XBGA0770E=XBGA0770E: Error encountered when attempting to redeploy WebSphere endpoint {0} to use datasource {1}. Exception = {1}.

Explanation

This is an internal configuration problem.

User response

Contact IBM Support.

Parent topic: [XBGA](#)

XBGA0780E

XBGA0780E=XBGA0780E: An attempt has been made to redeploy the WebSphere endpoint cluster {1} using the embedded Derby datasource {0}.

Explanation

A network datasource is required for a WebSphere endpoint cluster.

User response

Create and select a datasource for a network database.

Parent topic: [XBGA](#)

XBGA0790E

XBGA0790E=XBGA0790E: Error encountered when attempting to redeploy WebSphere endpoint {0} to use datasource {1}. Exception = {1}.

Explanation

The WebSphere endpoint redeployment operation has reported error(s).

User response

Ensure that the server or cluster hosting the endpoint has stopped.

Parent topic: [XBGA](#)

XBGA0800E

XBGA0800E=XBGA0800E: Error encountered when attempting to configure the job scheduler.

Explanation

Errors have been detected during the configuration of the job scheduler.

User response

View JVM logs or trace.log for further details.

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGA0810E

XBGA0810E=XBGA0810E: Error encountered when attempting to redeploy the WebSphere endpoint {0}.

Explanation

Errors have been detected during the redeployment of the WebSphere endpoint.

User response

View JVM logs or trace.log for further details.

Parent topic: [XBGA](#)

[Terms of Use](#) | [Feedback](#)

XBGS

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

XBGS0100E

XBGS0100=XBGS0100E: Datasource JNDI Name not configured.

XBGS0110E

XBGS0110=XBGS0110E: Startup bean service not enabled.

XBGS0200E

XBGS0200=XBGS0200E: Job scheduler not initialized.

XBGS0210E

XBGS0210=XBGS0210E: No database connection detected.

XBGS0220E

XBGS0220=XBGS0220E: Job scheduler MBeans may not be available.

XBGS0100E

XBGS0100=XBGS0100E: Datasource JNDI Name not configured.

Explanation

The datasource JNDI name for the job scheduler has not been set.

User response

Configure the job scheduler datasource JNDI Name at the configuration panel.

Parent topic: [XBGS](#)

XBGS0110E

XBGS0110=XBGS0110E: Startup bean service not enabled.

Explanation

The startup bean service has not been enabled on the job scheduler server.

User response

Enable startup bean service at the job scheduler application server panel.

Parent topic: [XBGS](#)

[Terms of Use](#) | [Feedback](#)

XBGS0200E

XBGS0200=XBGS0200E: Job scheduler not initialized.

Explanation

Unable to get job scheduler identity from the Bulletin Board.

User response

Verify job scheduler set up and configuration steps.

Parent topic: [XBGS](#)

[Terms of Use](#) | [Feedback](#)

XBGS0210E

XBGS0210=XBGS0210E: No database connection detected.

Explanation

Unable to get a database connection through the job scheduler datasource.

User response

Verify that all setup steps have been completed and the database has started.

Parent topic: [XBGS](#)

[Terms of Use](#) | [Feedback](#)

XBGS0220E

XBGS0220=XBGS0220E: Job scheduler MBeans may not be available.

Explanation

Unable to query job scheduler MBeans.

User response

Verify job scheduler set up and configuration steps.

Parent topic: [XBGS](#)

[Terms of Use](#) | [Feedback](#)

XCIM

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[XCIM0001I](#)

controller.started.successfully=XCIM0001I: The centralized installation manager started successfully.

[XCIM0002E](#)

create.work.record.dir.failed=XCIM0002E: A failure occurred while attempting to create the work record directory {0}.

[XCIM0003E](#)

error.detected.in.descriptor=XCIM0003E: An error occurred while parsing the installation package descriptor {0}. Ignoring the descriptor.

[XCIM0004E](#)

exception.parsing.descriptor=XCIM0004E: An exception occurred while parsing the installation package descriptor {0}. Ignoring the descriptor. Exception: {1}; Cause: {2}

[XCIM0005E](#)

key.file.does.not.exist=XCIM0005E: The specified key file {0} does not exist.

[XCIM0006E](#)

cannot.access.key.file=XCIM0006E: The system cannot access key file {0} using the ID "{1}".

[XCIM0007E](#)

error.installing.public.key=XCIM0007E: An exception occurred while installing the public key on host {0}. Exception: {1} Cause: {2}

[XCIM0008E](#)

error.installing.public.key.uimsg=XCIM0008E: A failure occurred while installing public key on host {0}. Cause: {1}

[XCIM0009E](#)

error.connecting.to.target=XCIM0009E: An error occurred while connecting to the remote target {0}. Exception: {1} Cause: {2}

[XCIM0010E](#)

error.connecting.to.target.uimsg=XCIM0010E: An error occurred while connecting to the remote target {0}. Cause: {1}

[XCIM0011E](#)

privateKey.required.if.no.password=XCIM0011E: Parameter {0} is required if parameter {1} is not specified.

[XCIM0012E](#)

request.cancelled.duplicate.target=XCIM0012E: The request is cancelled due to duplicate installation targets.

[XCIM0013E](#)

install.request.cancelled.duplicate.target=XCIM0013E: The installation request that is submitted on {0} is cancelled due to duplicate installation targets. Target host: {1}, Package name: {2}

[XCIM0014E](#)

uninstall.request.cancelled.duplicate.target=XCIM0014E: The uninstallation request that is submitted on {0} is cancelled due to duplicate installation targets. Target host: {1}, Package name: {2}

[XCIM0015E](#)

node.platformType.not.known=XCIM0015E: The platform type of node {0} is not known. Exception: {1}

[XCIM0016E](#)

node.hostName.not.known=XCIM0016E: The hostName of node {0} is not known. Exception: {1}

[XCIM0017E](#)

package.name.not.known=XCIM0017E: The installation package descriptor with the package name of "{0}" does not exist.

[XCIM0018E](#)

feature.name.not.known=XCIM0018E: The specified feature named "{0}" is not a valid feature of the installation package "{1}".

[XCIM0019E](#)

platform.type.invalid=XCIM0019E: The platformType "{0}" is invalid or not supported.

[XCIM0020E](#)

must.accept.license.to.proceed=XCIM0020E: Accept the license to proceed with the installation command.

XCIM0021I

excpn.checking.if.host.valid.uninstall.target=XCIM0021I: An exception occurred while checking if host {0} is a valid target for the uninstallation of package "{1}". Exception: {2}

XCIM0022E

package.not.valid.for.node.install=XCIM0022E: The software package "{0}" is not a valid package to install on host {1} with configured node "{2}".

XCIM0023E

package.not.valid.for.host.install=XCIM0023E: The software package "{0}" is not a valid package to install on host {1}.

XCIM0024E

installLoc.not.known=XCIM0024E: The installation location for package "{0}" cannot be determined.

XCIM0025E

file.not.found.on.host=XCIM0025E: File "{0}" could not be found on host {1}.

XCIM0026E

insufficient.freespace.for.workdir=XCIM0026E: There is an insufficient amount of space for the working directory "{0}" on host {1}. Package "{2}" requires {3} kilobytes of free space.

XCIM0027E

error.allocating.workdir.on.target=XCIM0027E: An error occurred while allocating directory "{0}" on host {1}. Exception: {2} Cause: {3}

XCIM0028E

cell.var.undefined=XCIM0028E: Variable {0} is undefined.

XCIM0029E

error.sending.file.to.target=XCIM0029E: An error occurred while sending file {0} to directory "{1}" on host {2}. Exception: {3} Cause: {4}

XCIM0030E

no.installer.for.platform=XCIM0030E: The installation package "{1}" does not contain a defined installer for platform type {0}.

XCIM0031E

no.matching.installer.for.machine.arch=XCIM0031E: The installation package "{2}" does not contain a defined installer for hardware type "{0}" and platform type {1}.

XCIM0032E

cell.has.no.managed.node.on.host=XCIM0032E: The current deployment manager cell contains no managed nodes that is defined on host {0}.

XCIM0033E

cell.has.no.managed.node.on.host.in.loc=XCIM0033E: The current deployment manager cell contains no managed nodes that is defined on host {0} at the specified installation location "{1}".

XCIM0034E

nodes.not.managed.on.host=XCIM0034E: Invalid nodes are specified for host {0}. The following nodes are considered invalid: {1}.

XCIM0035E

nodes.not.managed.on.host.in.loc=XCIM0035E: Invalid nodes are specified for host {0} at installation location {1}. The following nodes are considered invalid: {2}.

XCIM0036E

missing.install.root.for.node=XCIM0036E: The value of the WAS_INSTALL_ROOT variable for node {0} cannot be retrieved.

XCIM0037E

missing.user.install.root.for.node=XCIM0037E: The value of the USER_INSTALL_ROOT variable for node {0} cannot be retrieved.

XCIM0038E

nodes.with.diff.install.root=XCIM0038E: Nodes that have a different installation root path on host {0} from "{2}" of reference node {1} cannot be specified in the same command. The following nodes are considered invalid: {3}

XCIM0039E

error.running.manageprofiles.cmd=XCIM0039E: An error occurred while running the manageprofiles command: {0} on host {1}. Message text: {2}

XCIM0040E

error.accessing.remote.host.for.profile.name=XCIM0040E: An error occurred while accessing host {0} to retrieve the profile name. Exception: {1} Cause: {2}

XCIM0041E

error.during.install=XCIM0041E: An error occurred while installing package "{0}" on host {1}. Error detail: {2}

XCIM0042E

exception.checking.if.package.valid.for.node=XCIM0042E: An exception occurred while checking if package "{0}" is valid for installation on node "{1}". Exception: {2}

XCIM0043E

exception.caught.retrieving.install.roots=XCIM0043E: An exception occurred while retrieving the installation root variable values for node "{0}". Exception: {1}

XCIM0044E

error.running.cmd.on.host=XCIM0044E: An error occurred while running command [{0}] on the target host {1}. Exception: {2} Cause: {3}

XCIM0045E

install.failed.with.nonzero.rc=XCIM0045E: The installation command [{0}] failed with the return code {1}.

XCIM0046E

install.failed.with.stderr=XCIM0046E: The installation command [{0}] failed. The following error message was returned: "{1}"

XCIM0047E

install.failed.details.in.log=XCIM0047E: The installation command [{0}] failed. For more information, refer to "{1}".

XCIM0048E

post.install.failed.with.stderr=XCIM0048E: The post-installation command [{0}] failed. The following error message was returned: "{1}"

XCIM0049E

post.install.failed.with.nonzero.rc=XCIM0049E: The post-installation command [{0}] failed with the return code {1}.

XCIM0050E

post.install.failed.with.stdout=XCIM0050E: The post-installation command [{0}] failed. The following output message was returned: {1}

XCIM0051E

exception.retrieving.remote.log=XCIM0051E: An exception occurred while retrieving log "{0}" from host {1}. Exception: {2}

XCIM0052E

missing.license.agreement.for.package=XCIM0052E: The license agreement file for package "{0}" cannot be located.

XCIM0053E

missing.license.information.for.package=XCIM0053E: The license information file for package "{0}" cannot be located.

XCIM0054E

multiple.installLoc.on.host=XCIM0054E: The controller detected more than one installation location on the host {0}. Installation locations: {1}.

XCIM0055E

unzip.installer.only.take.zipfiles=XCIM0055E: The installation package contains a file name with an invalid file type: {0}

XCIM0056E

file.does.not.exist.in.dir=XCIM0056E: File {0} is not found in directory "{1}".

XCIM0057E

install.failed.as.no.files.were.specified=XCIM0057E: The installation failed, because no files were specified in the installation package.

XCIM0058E

error.instantiating.helper=XCIM0058E: An error occurred while accessing helper class "{0}" specified by installation package "{1}". Error detail: {2}

XCIM0059E

unknown.parameter.for.helper=XCIM0059E: "{0}" is not a valid parameter name for the helper class {1}.

XCIM0060E

non.string.type.supplied.as.adminId=XCIM0060E: A non-string value was specified for the user name.

XCIM0061E

zero.length.or.blank.adminId.not.allowed=XCIM0061E: An empty string was specified as a value for the user name parameter.

XCIM0062E

non.string.type.supplied.as.password=XCIM0062E: A non-string value was specified for the password.

XCIM0063E

zero.length.or.blank.password.not.allowed=XCIM0063E: An empty string was specified for the password.

XCIM0064E

error.persisting.file=XCIM0064E: An exception occurred while writing data to file "{0}". Exception: {1}

XCIM0065E

error.while.sending.notification=XCIM0065E: An error occurred while sending a notification. Exception: {0}

XCIM0066E

failed.to.run.startNode.due.to.error=XCIM0066E: Unable to initiate the startNode command on host {0} due to an error. Exception: {1}

XCIM0067E

installed.updi.version.not.known=XCIM0067E: The version of the Update Installer that is present on host {0} in location "{1}" is unknown.

XCIM0068E

installed.updi.version.incompatible=XCIM0068E: The version of the Update Installer that is present on host {0} in location "{1}" is {2}. The minimum version required by the maintenance package "{3}" is {4}.

XCIM0069E

error.retrieving.file.from.host=XCIM0069E: An error occurred while retrieving file "{0}" from host {1}. Exception: {2}

XCIM0070E

post.install.failed.details.in.log=XCIM0070E: The post-installation command [{0}] failed. For more information, refer to "{1}".

XCIM0071E

error.during.uninstall=XCIM0071E: An error occurred while uninstalling package "{0}" on host {1}. Error detail: {2}

XCIM0072E

host.install.loc.invalid.as.uninstall.target=XCIM0072E: Installation location "{0}" on host {1} does not have the correct version of the package "{2}" required for the remote uninstallation process. Package version: {3}

XCIM0073E

uninstall.failed.with.stderr=XCIM0073E: The uninstallation command [{0}] failed. The following error message was returned: "{1}"

XCIM0074E

uninstall.failed.with.nonzero.rc=XCIM0074E: The uninstallation command [{0}] failed. The following error code was returned: {1}.

XCIM0075E

uninstall.failed.details.in.log=XCIM0075E: The uninstallation command [{0}] failed. For more information, refer to "{1}".

XCIM0076E

uninstall.results.unknown.log.missing=XCIM0076E: The status of the uninstallation command [{0}] cannot be determined, because the output log is missing. Log path: "{1}". Host name: {2}

XCIM0077E

unknown.product.name=XCIM0077E: The product short name "{0}" specified in the installation package descriptor "{1}" is not supported.

XCIM0078E

pak.file.not.applicable.to.package=XCIM0078E: The .pak file "{0}" cannot be used to update the product specified by the installation package "{1}".

XCIM0079E

invalid.version.string.format=XCIM0079E: An invalid version string "{0}" for product {1} is located in file "{2}".

XCIM0080E

error.checking.if.maintenance.dependency.met=XCIM0080E: An exception occurred while checking if node "{0}" meets the maintenance dependency. Exception: {1}

XCIM0081E

cannot.mix.updates.for.different.platforms=XCIM0081E: Pak files "{0}" and "{1}" are incompatible, because they might be applicable to different operating systems.

XCIM0082E

maintenance.descriptor.file.missing=XCIM0082E: Pak file "{0}" is missing a required descriptor: {1}.

XCIM0083E

cell.has.no.defined.node.on.host=XCIM0083E: The current deployment manager cell contains no node with the WAS_INSTALL_ROOT variable defined on host {0}.

XCIM0084E

cell.has.defined.node.on.host.in.loc=XCIM0084E: The current deployment manager cell has no node with the WAS_INSTALL_ROOT variable set to "{1}" on host {0}.

XCIM0085E

only.boolean.value.is.allowed=XCIM0085E: Only boolean value is allowed for parameter {0}.

XCIM0086E

fromPrereq.true.when.prereq.package.is.null=XCIM0086E: A conflicting attribute value for an installation descriptor is detected.

XCIM0087E

license.file.in.unknown.prereq.package=XCIM0087E: The prerequisite package that contains the license agreement file for package "{0}" cannot be determined.

XCIM0088E

license.file.not.found.in.prereq.package=XCIM0088E: The license file for package "{0}" cannot be located in the directory of the prerequisite package "{1}".

XCIM0089E

no.zip.file.with.specified.prefix=XCIM0089E: The .zip file that contains the license file for package "{0}" cannot be located. The prefix for the .zip file name is: {1}.

XCIM0090E

file.matching.pattern.not.found=XCIM0090E: A file that matches the name pattern "{0}" is not found in directory "{1}".

XCIM0091E

file.list.cannot.be.empty=XCIM0091E: The parameter "fileList" is required for a maintenance installation and must not be empty.

XCIM0092E

no.maintenance.file.specified=XCIM0092E: The maintenance installation request was not successful, because no files were selected for installation.

XCIM0093E

selected.maintenance.not.valid.for.node=XCIM0093E: The list of selected maintenance is not valid to install on host {0}. List of selected maintenance: {1}

XCIM0094E

prereq.updi.version.not.known=XCIM0094E: The minimum version of the Update Installer required for installing the selected maintenance cannot be determined. Selected maintenance: {0}

XCIM0095E

updi.version.txt.missing=XCIM0095E: The Update Installer installation image file "{0}" that is located in directory "{1}" is missing the entry {2}.

XCIM0096E

updi.version.unknown=XCIM0096E: The version of the Update Installer in file "{0}" that is located in directory "{1}" cannot be determined.

XCIM0097E

updi.version.not.compatible=XCIM0097E: The Update Installer in file "{0}" of directory "{1}" is version {2}. This version does not meet the minimum required version of {3}.

XCIM0098E

error.while.parsing.maintenance.xml=XCIM0098E: An error occurred while parsing the maintenance.xml file in file "{0}". Exception: {1}

XCIM0099E

cimgr.controller.not.initialized=XCIM0099E: The centralized installation manager did not initialize, or has not yet completed initialization.

XCIM0100I

operation.completed.successfully=XCIM0100I: Operation completed successfully.

XCIM0102E

ERROR_INSTALL_TARGET_HOSTNAME_REQUIRED=XCIM0102E: The host name of the install target {0} is absent.

XCIM0103E

ERROR_INSTALL_TARGET_HOSTNAME_INVALID=XCIM0103E: The host name of the install target {0} contains invalid characters.

XCIM0104E

ERROR_INSTALL_TARGET_PLATFORM_INVALID=XCIM0104E: The platform of the install target {0} is invalid.

XCIM0121E

uninstall.updi.failed.with.stderr=XCIM0121E: The uninstall of the Update Installer failed with the error message "{1}". Command: {0}

XCIM0122E

uninstall.updi.failed.with.nonzero.rc=XCIM0122E: The uninstall of the Update Installer failed with the return code "{1}". Command: {0}

XCIM0123E

uninstall.updi.failed.details.in.log=XCIM0123E: The uninstall of the Update Installer failed. For more information, consult "{0}".

XCIM0124E

uninstall.updi.results.unknown.log.missing=XCIM0124E: The successful uninstall of the Update Installer cannot be determined as the output log is missing. Log path: "{0}". Host name: {1}

XCIM0125E

file.download.urls.is.empty.but.referenced=XCIM0125E: There is no "FileDownloadURLs" specified but element {0} has a non-empty "DownloadDir" attribute.

XCIM0126E

file.download.url.index.out.of.bound=XCIM0126E: The index number specified by the "DownloadURL" attribute of element {0} exceeds the number of file download URLs specified in the descriptor.

XCIM0127E

maintenance.cmd.operates.only.on.maintenance.pkg=XCIM0127E: The {0} command is only valid for maintenance packages.

XCIM0128E

start.node.failed.with.stderr=XCIM0128E: The "startNode" command failed with the error message "{0}"

XCIM0129E

start.node.failed.with.nonzero.rc=XCIM0129E: The "startNode" command failed with the return code {0}.

XCIM0130E

start.node.failed.with.stdout=XCIM0130E: The "startNode" command failed with the output message: {0}

XCIM0131E

post.uninstall.failed.with.stderr=XCIM0131E: The post-uninstall command [{0}] failed with the error message "{1}"

XCIM0132E

post.uninstall.failed.with.nonzero.rc=XCIM0132E: The post-uninstall command [{0}] failed with the return code {1}.

XCIM0133E

post.uninstall.failed.with.stdout=XCIM0133E: The post-uninstall command [{0}] failed with the output message: {1}

XCIM0134E

no.uninstall.maintenance.file.specified=XCIM0134E: No maintenance file were selected for uninstall.

XCIM0135E

repository.root.undefined=XCIM0135E: The centralized installation repository root is undefined.

XCIM0136E

error.opening.remoteaccess.log=XCIM0136E: Error opening log file {0}. Logging disabled. Exception: {1}

XCIM0137E

maintenance.xml.missing.prereq.updi.ver=XCIM0137E: The maintenance.xml in {0} does not specify the required Update Installer version information.

XCIM0138E

package.prereq.missing=XCIM0138E: The Install Package Descriptor with the package name of "{0}" is missing a PrereqPackage attribute.

XCIM0139E

package.prereq.name.not.known=XCIM0139E: The prerequisite package named by the Install Package Descriptor "{0}" cannot be located. The name of the prerequisite package is {1}.

XCIM0140E

remote.cmd.failed.with.stderr=XCIM0140E: Command [{0}] failed with the error message "{1}"

XCIM0141E

remote.cmd.failed.with.nonzero.rc=XCIM0141E: Command [{0}] failed with the return code {1}.

XCIM0142E

remote.cmd.failed.with.stdout=XCIM0142E: Command [{0}] failed with the output message: {1}

XCIM0143E

product.short.name.missing=XCIM0143E: Either a ProductShortName or a ProductShortNames element must be present. Descriptor filename: {0}.

XCIM0144E

pak.entry.name.mandatory.for.zip.bundle=XCIM0144E: "PakEntryName" element is required if element {0} specifies a zip file for its "PakFilename" attribute.

XCIM0145E

pak.entry.name.has.wrong.filetype=XCIM0145E: Only filename with a ".pak" suffix is allowed in "PakEntryName" elements.

XCIM0146E

pak.entry.info.for.zip.missing=XCIM0146E: The MaintenancePak info for zip file {0} cannot be found.

XCIM0147E

unexpected.error.locating.node.for.host=XCIM0147E: An unexpected error occurred while locating the middleware node on host {0} with the install location {1}. Exception = {2}

XCIM0148E

maintenance.pak.missing.from.bundle=XCIM0148E: The maintenance pak file {0} is not found in {1}.

XCIM0149E

require.special.parm.missing=XCIM0149E: Value for the special parameter {0} is undefined. Specify a value for the specified parameter using the "specialParms" command argument.

XCIM0150E

maintenance.pkg.not.valid.for.cmd=XCIM0150E: Maintenance package "{0}" is not valid for the "{1}" command.

XCIM0151E

cannot.mix.updates.for.different.products=XCIM0151E: Pak file "{0}" is incompatible with the other pak files selected as they target different software packages.

XCIM0152E

cannot.have.both.product.short.name.elements=XCIM0152E: Either a ProductShortName or a ProductShortNames element must be present but not both. Descriptor filename: {0}.

XCIM0153E

ifix.found.nodcentric.uninstall.aborted=XCIM0153E: Maintenance packages for the software product must be uninstalled first before uninstalling the product. Maintenance for product {0} detected on host {1}: {2}.

XCIM0154E

ifix.found.maintenance.uninstall.aborted=XCIM0154E: {0} version {1} cannot be uninstalled without first uninstalling these interim fixes from host {2}: {3}.

XCIM0155E

exception.parsing.xml.on.host=XCIM0155E: An unexpected error occurred accessing and parsing {0} from host {1}. Exception = {2}

XCIM0156E

exception.parsing.maintenance.xml.from.pakfile.on.host=XCIM0156E: An unexpected error occurred while parsing the maintenance.xml file in {0} from host {1}. Exception = {2}

XCIM0157E

installer.element.not.allowed=XCIM0157E: Installer element is not allowed outside of an InstallerGroup when there are feature-dependent InstallerGroup. Descriptor filename: {0}.

XCIM0158E

missing.installer.element=XCIM0158E: The install package descriptor is missing the mandatory Installer element. Descriptor filename: {0}.

XCIM0159E

create.cmd.log.failed=XCIM0159E: Failed to create output file for command logging.

XCIM0160E

error.creating.file.writer=XCIM0160E: An error was encountered creating a FileWriter for log file {0}. Exception: {1}

XCIM0161I

rmtcmd.started.running.cmdgrp=XCIM0161I: Started running command group on host {0}

XCIM0162I

rmtcmd.sending.file.to.host=XCIM0162I: Sending script file {0} to host.

XCIM0163I

rmtcmd.resolved.cmd.path=XCIM0163I: Resolved command path: {0}

XCIM0164I

rmtcmd.running.cmd=XCIM0164I: Running command: {0}

XCIM0165I

rmtcmd.stdout.from.cmd=XCIM0165I: Standard output from command:

XCIM0166I

rmtcmd.stderr.from.cmd=XCIM0166I: Standard error from command:

XCIM0167I

rmtcmd.rtncode.from.cmd=XCIM0167I: Return code from command: {0}

XCIM0168I

rmtcmd.check.stdout.for.success.msg=XCIM0168I: Checking standard output from command for presence of success message: {0}. Check type is "{1}".

XCIM0169I

rmtcmd.completed=XCIM0169I: Command group completed successfully.

XCIM0170E

rmtcmd.failed=XCIM0170E: Command group failed as specified success criteria was not met.

XCIM0171E

rmtcmd.failed.see.log=XCIM0171E: Command group failed. For details refer to log file {0}.

XCIM0175E

only.one.feature.is.allowed=XCIM0175E: Only one feature can be specified for package "{0}".

XCIM0176E

specify.at.least.one.feature=XCIM0176E: At least one feature must be specified for package "{0}".

XCIM0177E

remote.cmd.called.in.the.wrong.state=XCIM0177E: The processing of a remote command occurred out of order. Processing has been halted.

XCIM0178E

remote.file.matching.pattern.not.found=XCIM0178E: A file matching the pattern "{0}" cannot be found in directory "{1}" of host {2}.

XCIM0179E

cannot.resolve.java.install.dir=XCIM0179E: Unable to resolve "{0}" to the actual installation directory for the IBM software developer kit (SDK) on host {1}.

XCIM0180E

error.resolving.file.or.dir.on.host=XCIM0180E: An error was encountered while resolving the name pattern "{0}" to a file or directory in directory "{1}" of the remote host {2}. Exception: {3} Cause: {4}

XCIM0181E

unknown.java.install.loc.for.platform=XCIM0181E: The install location for the IBM Software Developer Kit (SDK), Java 2 Technology Edition is not known for "{0}".

XCIM0182E

no.middleware.nodes.defined.on.host=XCIM0182E: Target host {0} has no middleware nodes defined.

XCIM0183E

node.not.middleware.node.on.host=XCIM0183E: The specified node "{0}" is not a middleware node on host {1}.

XCIM0184E

node.invalid.as.wasce.manager=XCIM0184E: The specified node "{0}" is invalid as manager of the WebSphere Application Server Community Edition installation.

XCIM0185E

optional.parm.missing.for.cmd.helper=XCIM0185E: A required OptionalCommandParms element is missing from the command definition for install package "{0}" and helper class {1}.

XCIM0186E

unexpected.error.renaming.remote.logfile=XCIM0186E: An error was encountered renaming log file {0} on host {1}. Exception: {2} Cause: {3}

XCIM0187E

node.not.found.for.name=XCIM0187E: The Node object cannot be located for the middleware node {0}.

XCIM0188E

node.metadata.service.not.available=XCIM0188E: The NodeMetaData service is not available for middleware node {0}. The middleware agent or node agent may have stopped.

XCIM0189E

unexpected.error.updating.node.metadata=XCIM0189E: An error was encountered updating the node-metadata for middleware node {0}. Exception: {1} Cause: {2}

XCIM0190E

ibm.java.pkg.not.found.with.rpm=XCIM0190E: Cannot find package using command "{0}" on host {1}.

XCIM0191I

license.agreement.not.avail.for.viewing=XCIM0191I: The license agreement file for package "{0}" is not available for direct viewing from the command.

[XCIM0192E](#)

install.failed.with.nonzero.rc.and.stdout=XCIM0192E: The install command [{0}] failed with the return code {1}. Console output: {2}

[XCIM0193I](#)

java.sdk.cannot.be.uninstalled.remotely=XCIM0193I: The IBM software developer kit (SDK) for Windows cannot be uninstalled remotely using the centralized installation manager. Use the Windows "Add or Remove programs" locally to uninstall.

[XCIM0194E](#)

ibm.java2.sdk.already.installed=XCIM0194E: The IBM software developer kit (SDK), Java 2 Technology Edition is already installed on host {0}.

[XCIM0195E](#)

ibm.java2.may.already.be.installed.on.host=XCIM0195E: The IBM software developer kit (SDK) may already be installed on host {1} as the directory "{0}" is not empty.

[XCIM0196E](#)

unexpected.error.checking.dir.empty=XCIM0196E: An error was encountered checking if the directory {0} on host {1} was empty. Exception: {2} Cause: {3}

[XCIM0197E](#)

package.invalid.for.uninstall.of.non-ifix.pak.type=XCIM0197E: Package "{0}" cannot be used to uninstall maintenance {1} of type {2}.

[XCIM0198E](#)

installed.java.home.invalid=XCIM0198E: The specified installation directory "{0}" for a supported Java environment on host {1} is invalid.

[XCIM0199W](#)

install.completed.with.errors=XCIM0199W: Installation completed but errors were detected. Check logs for error messages.

[XCIM0200W](#)

uninstall.completed.with.errors=XCIM0200W: Uninstallation completed but errors were detected. Check logs for error messages.

[XCIM0201E](#)

no.middleware.nodes.on.host.as.wasce.mgr=XCIM0201E: No middleware nodes on host {0} meet the minimum requirement of having WebSphere Extended Deployment Version {1} installed.

[XCIM0202E](#)

install.failed.cmd.timed.out.with.rc=XCIM0202E: The installation command [{0}] timed out. Return code = {1}.

[XCIM0203E](#)

install.failed.cmd.timed.out=XCIM0203E: The installation command [{0}] timed out.

[XCIM0204E](#)

unexpected.error.stopping.middleware.agent=XCIM0204E: An error was encountered while stopping the middleware agent on host {0}. Installation location = "{1}". Exception: {2}

[XCIM0205E](#)

exception.checking.remote.path=XCIM0205E: An error was encountered checking for the existence of file {0} on host {1}. Exception: {2} Cause: {3}

[XCIM0216E](#)

user.not.configured.to.sudo.slibclean.with.nopasswd=XCIM0216E: Unable to run "sudo slibclean" as user {0} without a password prompt on target host {1}.

[XCIM0217E](#)

user.unable.to.sudo.slibclean=XCIM0217E: Unable to run "sudo slibclean" as user {0} on target host {1}. Error message: {2}

[XCIM0226E](#)

prereq.ifix.missing=XCIM0226E: Interim Fix {0} must be applied before installing {1} if the target host has product {2} installed.

XCIM0001I

controller.started.successfully=XCIM0001I: The centralized installation manager started successfully.

Explanation

The centralized installation manager started successfully and is available to service requests.

User response

No user action is required.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0002E

create.work.record.dir.failed=XCIM0002E: A failure occurred while attempting to create the work record directory {0}.

Explanation

The centralized installation manager encountered an error while creating the specified directory for work records.

User response

Check the log for any error messages that precede this one. Correct the error if possible and restart the deployment manager.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0003E

error.detected.in.descriptor=XCIM0003E: An error occurred while parsing the installation package descriptor {0}. Ignoring the descriptor.

Explanation

An error is detected in the specified install package descriptor.

User response

Check the log for any error messages that precede this one. Correct the error in the descriptor file as indicated or contact an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0004E

exception.parsing.descriptor=XCIM0004E: An exception occurred while parsing the installation package descriptor {0}. Ignoring the descriptor. Exception: {1}; Cause: {2}

Explanation

An error is detected in the specified installation package descriptor.

User response

Correct the error in the descriptor file as indicated or contact an IBM service representative.

Parent topic: [XCIM](#)

XCIM0005E

key.file.does.not.exist=XCIM0005E: The specified key file {0} does not exist.

Explanation

The specified key file cannot be found at the specified location.

User response

Specify the correct location for the key file and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0006E

cannot.access.key.file=XCIM0006E: The system cannot access key file {0} using the ID "{1}".

Explanation

The server running under the specified user ID does not have the authority to read the specified key file.

User response

Grant read access authority to the specified user ID and retry the operation.

Parent topic: [XCIM](#)

XCIM0007E

error.installing.public.key=XCIM0007E: An exception occurred while installing the public key on host {0}.
Exception: {1} Cause: {2}

Explanation

The controller did not install the public key on the target host due to error.

User response

Correct the error as indicated by the message and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0008E

error.installing.public.key.uimsg=XCIM0008E: A failure occurred while installing public key on host {0}. Cause: {1}

Explanation

The controller did not install the public key on the target host because of the specified error.

User response

Correct the error as indicated by the message and retry the operation.

Parent topic: [XCIM](#)

XCIM0009E

error.connecting.to.target=XCIM0009E: An error occurred while connecting to the remote target {0}. Exception: {1} Cause: {2}

Explanation

The controller did not establish a connection with the target host for centralized installation or uninstallation operations because of the specified error.

User response

Correct the error as indicated by the message and retry the operation.

Parent topic: [XCIM](#)

XCIM0010E

error.connecting.to.target.uimsg=XCIM0010E: An error occurred while connecting to the remote target {0}.
Cause: {1}

Explanation

The controller did not establish a connection with the target host for centralized installation or uninstallation operations because of the specified error.

User response

Correct the error as indicated by the message and retry the operation.

Parent topic: [XCIM](#)

XCIM0011E

privateKey.required.if.no.password=XCIM0011E: Parameter {0} is required if parameter {1} is not specified.

Explanation

At least one of the two parameters must be specified with this command.

User response

Specify at least one of the two parameters and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0012E

request.cancelled.duplicate.target=XCIM0012E: The request is cancelled due to duplicate installation targets.

Explanation

The installation or uninstallation request to which this message is attached is cancelled, because another request to the same target is still being processed.

User response

Wait until the other installation or uninstallation request completes, and then resubmit your request.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0013E

install.request.cancelled.duplicate.target=XCIM0013E: The installation request that is submitted on {0} is cancelled due to duplicate installation targets. Target host: {1}, Package name: {2}

Explanation

The specified installation request is cancelled, because another request to the same target is still being processed.

User response

Wait until the other installation or uninstallation request completes, and then resubmit your request.

Parent topic: [XCIM](#)

XCIM0014E

uninstall.request.cancelled.duplicate.target=XCIM0014E: The uninstallation request that is submitted on {0} is cancelled due to duplicate installation targets. Target host: {1}, Package name: {2}

Explanation

The specified uninstallation request is cancelled, because another request to the same target is still being processed.

User response

Wait until the other installation or uninstallation request completes, and then resubmit your request.

Parent topic: [XCIM](#)

XCIM0015E

node.platformType.not.known=XCIM0015E: The platform type of node {0} is not known. Exception: {1}

Explanation

The controller cannot retrieve the platform type information of the node.

User response

Make sure the node name is correctly entered and retry the operation, or contact an IBM support representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0016E

node.hostName.not.known=XCIM0016E: The hostName of node {0} is not known. Exception: {1}

Explanation

The controller cannot retrieve the host name of the node.

User response

Make sure the node name is correctly entered and retry the operation, or contact an IBM support representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0017E

package.name.not.known=XCIM0017E: The installation package descriptor with the package name of "{0}" does not exist.

Explanation

The controller cannot identify the named installation package.

User response

Ensure that the name is entered correctly, or issue the listPackagesForInstall AdminTask command to list the names of all the installation packages that are known to the controller.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0018E

feature.name.not.known=XCIM0018E: The specified feature named "{0}" is not a valid feature of the installation package "{1}".

Explanation

The installation package does not contain any feature with the specified name.

User response

Ensure that the name of the package and the feature is correctly entered and retry the command, or issue the listFeaturesForInstall AdminTask command to list the names of all the features that are contained within this package.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0019E

platform.type.invalid=XCIM0019E: The platformType "{0}" is invalid or not supported.

Explanation

The specified platformType is either misspelled or not a supported platformType for centralized installation

User response

Ensure that the platformType parameter value is spelled correctly or specify a supported platformType and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0020E

must.accept.license.to.proceed=XCIM0020E: Accept the license to proceed with the installation command.

Explanation

The value "true" must be specified for the acceptLicense parameter.

User response

Specify a value of "true" for the parameter and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0021I

excptn.checking.if.host.valid.uninstall.target=XCIM0021I: An exception occurred while checking if host {0} is a valid target for the uninstallation of package "{1}". Exception: {2}

Explanation

The controller encountered an error while performing the validity check.

User response

Review the error that accompanies this message. If possible, correct the error and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0022E

package.not.valid.for.node.install=XCIM0022E: The software package "{0}" is not a valid package to install on host {1} with configured node "{2}".

Explanation

The specified software package is not a valid package to install on the specified host.

User response

Verify that the target host meets the dependencies as stipulated by the software package. Then, contact the provider of the software package.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0023E

package.not.valid.for.host.install=XCIM0023E: The software package "{0}" is not a valid package to install on host {1}.

Explanation

The specified software package is a not valid package to install on the specified host.

User response

Verify that the target host meets the dependencies as stipulated by the software package. Then, contact the provider of the software package.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0024E

installLoc.not.known=XCIM0024E: The installation location for package "{0}" cannot be determined.

Explanation

The directory on the target host must be specified.

User response

Provide the installation location and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0025E

file.not.found.on.host=XCIM0025E: File "{0}" could not be found on host {1}.

Explanation

The specified file is not found on the target host.

User response

The specified file might have been removed or deleted from the target host. If possible, contact the administrator of the target workstation to restore the file, and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0026E

insufficient.freespace.for.workdir=XCIM0026E: There is an insufficient amount of space for the working directory "{0}" on host {1}. Package "{2}" requires {3} kilobytes of free space.

Explanation

The target directory does not contain enough free space to hold the files that are required for the installation.

User response

Ensure that the specified directory contains the minimum amount of free space, or specify a different directory, and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0027E

error.allocating.workdir.on.target=XCIM0027E: An error occurred while allocating directory "{0}" on host {1}.
Exception: {2} Cause: {3}

Explanation

An error occurred while allocation the directory on the target host.

User response

Correct the error and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0028E

cell.var.undefined=XCIM0028E: Variable {0} is undefined.

Explanation

The named variable is not yet defined within the WebSphere Application Server Deployment Manager cell.

User response

Define the named variable with the appropriate value in the administrative console, and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0029E

error.sending.file.to.target=XCIM0029E: An error occurred while sending file {0} to directory "{1}" on host {2}.
Exception: {3} Cause: {4}

Explanation

The controller did not send the specified file to the target host due to error.

User response

Correct the problem that caused the exception and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0030E

no.installer.for.platform=XCIM0030E: The installation package "{1}" does not contain a defined installer for platform type {0}.

Explanation

The controller cannot locate an appropriate installation program for the target platform type.

User response

Contact the provider of the installation package.

Parent topic: [XCIM](#)

XCIM0031E

no.matching.installer.for.machine.arch=XCIM0031E: The installation package "{2}" does not contain a defined installer for hardware type "{0}" and platform type {1}.

Explanation

The controller cannot locate an appropriate installation program for the target hardware type and platform type.

User response

If the hardware type needs to be supported, contact the provider of the installation package.

Parent topic: [XCIM](#)

XCIM0032E

cell.has.no.managed.node.on.host=XCIM0032E: The current deployment manager cell contains no managed nodes that is defined on host {0}.

Explanation

The current deployment manager cell does not include any managed nodes from the specified host.

User response

Install WebSphere Application Server Network Deployment on the specified host with a custom profile. Federate the node to the deployment manager cell, and retry installing WebSphere Extended Deployment.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0033E

cell.has.no.managed.node.on.host.in.loc=XCIM0033E: The current deployment manager cell contains no managed nodes that is defined on host {0} at the specified installation location "{1}".

Explanation

The current deployment manager cell does not include any managed nodes from the remote host at the specified installation location.

User response

Ensure that the installation location is correct by issuing the listManagedNodesOnHost command with the -showInstallLocation option, and retry the installation command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0034E

nodes.not.managed.on.host=XCIM0034E: Invalid nodes are specified for host {0}. The following nodes are considered invalid: {1}.

Explanation

One or more of the specified nodes are not defined as managed nodes from the specified host in the current deployment manager cell.

User response

Specify only the managed nodes that are defined on the specified host, or federate the nodes to the current deployment manager cell. Issue the command again.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0035E

nodes.not.managed.on.host.in.loc=XCIM0035E: Invalid nodes are specified for host {0} at installation location {1}. The following nodes are considered invalid: {2}.

Explanation

One or more of the specified nodes are not defined as managed nodes from the host at the specified location in the current deployment manager cell.

User response

Specify only the managed nodes that are defined on the specified host, or federate the nodes to the current deployment manager cell. Issue the command again.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0036E

missing.install.root.for.node=XCIM0036E: The value of the WAS_INSTALL_ROOT variable for node {0} cannot be retrieved.

Explanation

The WAS_INSTALL_ROOT variable is defined for all the managed nodes. The configuration might be corrupted.

User response

Verify that the WAS_INSTALL_ROOT variable for the node has not been deleted inadvertently. If so, add the variable with the appropriate value again.

Parent topic: [XCIM](#)

XCIM0037E

missing.user.install.root.for.node=XCIM0037E: The value of the USER_INSTALL_ROOT variable for node {0} cannot be retrieved.

Explanation

The USER_INSTALL_ROOT variable is defined for all the managed nodes. The configuration might be corrupted.

User response

Verify that the USER_INSTALL_ROOT variable for the node has not been deleted inadvertently. If so, add the variable with the appropriate value again.

Parent topic: [XCIM](#)

XCIM0038E

nodes.with.diff.install.root=XCIM0038E: Nodes that have a different installation root path on host {0} from "{2}" of reference node {1} cannot be specified in the same command. The following nodes are considered invalid: {3}

Explanation

Nodes with different installation root paths on the same host must not be specified in the same command.

User response

Issue the installation command again by specifying only those nodes that have the same installation root path.

Parent topic: [XCIM](#)

XCIM0039E

error.running.manageprofiles.cmd=XCIM0039E: An error occurred while running the manageprofiles command: {0} on host {1}. Message text: {2}

Explanation

There was an error with running the manageprofiles command on the remote host.

User response

Correct the error and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0040E

error.accessing.remote.host.for.profile.name=XCIM0040E: An error occurred while accessing host {0} to retrieve the profile name. Exception: {1} Cause: {2}

Explanation

The controller encountered an error while running a remote command to retrieve the name of a node profile.

User response

Review the error message, correct the error, and retry the command.

Parent topic: [XCIM](#)

XCIM0041E

error.during.install=XCIM0041E: An error occurred while installing package "{0}" on host {1}. Error detail: {2}

Explanation

The installation operation was stopped due to an unexpected error.

User response

Review the error message, correct the error, and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0042E

exception.checking.if.package.valid.for.node=XCIM0042E: An exception occurred while checking if package "{0}" is valid for installation on node "{1}". Exception: {2}

Explanation

The controller encountered an error while performing the validity check.

User response

Review the error message, correct the error, and retry the operation.

Parent topic: [XCIM](#)

XCIM0043E

exception.caught.retrieve.install.roots=XCIM0043E: An exception occurred while retrieving the installation root variable values for node "{0}". Exception: {1}

Explanation

The controller encountered an error while retrieving the installation root values for the node.

User response

Ensure that the name of the node is entered correctly and retry the operation.

Parent topic: [XCIM](#)

XCIM0044E

error.running.cmd.on.host=XCIM0044E: An error occurred while running command [{0}] on the target host {1}.
Exception: {2} Cause: {3}

Explanation

The controller failed to run the command on the target host due to error.

User response

Review the error message, correct the error, and run the command manually on the target host.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0045E

install.failed.with.nonzero.rc=XCIM0045E: The installation command [{0}] failed with the return code {1}.

Explanation

The specified installation command was issued, but a non-zero code indicating an error was returned.

User response

Review the documentation provided by the provider of the installation package for the details about the return code, correct the error, and retry the command.

Parent topic: [XCIM](#)

XCIM0046E

install.failed.with.stderr=XCIM0046E: The installation command [{0}] failed. The following error message was returned: "{1}"

Explanation

The specified installation command was issued but an error was reported.

User response

Review the error message, correct the error, and retry the operation.

Parent topic: [XCIM](#)

XCIM0047E

install.failed.details.in.log=XCIM0047E: The installation command [{0}] failed. For more information, refer to "{1}".

Explanation

The specified installation command was issued, but a failure was reported. For details, refer to the specified log file.

User response

Review the messages in the specified log, correct the error, and retry the operation.

Parent topic: [XCIM](#)

XCIM0048E

post.install.failed.with.stderr=XCIM0048E: The post-installation command [{0}] failed. The following error message was returned: "{1}"

Explanation

The specified post-installation command was issued, but an error was reported.

User response

Review the error message, correct the error, and run the command manually on the target host.

Parent topic: [XCIM](#)

XCIM0049E

post.install.failed.with.nonzero.rc=XCIM0049E: The post-installation command [{0}] failed with the return code {1}.

Explanation

The specified post-installation command was issued, but a non-zero code was returned.

User response

Review the documentation provided by the provider of the installation package for details about the return code, correct the error, and run the command manually on the target host.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0050E

post.install.failed.with.stdout=XCIM0050E: The post-installation command [{0}] failed. The following output message was returned: {1}

Explanation

The specified post-installation command was issued, but an error was reported.

User response

Review the error message, correct the error, and run the command manually on the target host.

Parent topic: [XCIM](#)

XCIM0051E

exception.retrieve.remote.log=XCIM0051E: An exception occurred while retrieving log "{0}" from host {1}.
Exception: {2}

Explanation

The controller was unable to retrieve the specified log file from the target host due to an error.

User response

Review the error message, correct the error, and retry the operation.

Parent topic: [XCIM](#)

XCIM0052E

missing.license.agreement.for.package=XCIM0052E: The license agreement file for package "{0}" cannot be located.

Explanation

The license agreement file for the package cannot be located in the centralized installation repository.

User response

Contact the provider of the package for the license agreement, or ensure that all of the required files were downloaded successfully and retry the operation.

Parent topic: [XCIM](#)

XCIM0053E

missing.license.information.for.package=XCIM0053E: The license information file for package "{0}" cannot be located.

Explanation

The license information file for the package cannot be located in the centralized installation repository.

User response

Contact the provider of the package for the license information, ensure that all of the required files were downloaded successfully and retry the operation.

Parent topic: [XCIM](#)

XCIM0054E

multiple.installLoc.on.host=XCIM0054E: The controller detected more than one installation location on the host {0}. Installation locations: {1}.

Explanation

The specified host contains more than one installation location of WebSphere Application Server. At least one of the locations must be specified as the installation target.

User response

Specify at least one of the installation locations and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0055E

unzip.installer.only.take.zipfiles=XCIM0055E: The installation package contains a file name with an invalid file type: {0}

Explanation

An installation package that uses the process of extracting files as the package installer must specify .zip as the only valid file type.

User response

Contact the provider of the software package to verify the package content is correct.

Parent topic: [XCIM](#)

XCIM0056E

file.does.not.exist.in.dir=XCIM0056E: File {0} is not found in directory "{1}".

Explanation

The specified file cannot be found in the specified directory

User response

Ensure that the specified directory contains the file.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0057E

install.failed.as.no.files.were.specified=XCIM0057E: The installation failed, because no files were specified in the installation package.

Explanation

The installation process cannot proceed, because no files were specified in the installation package.

User response

Contact the provider of the software package to verify that the package descriptor associated with the software package is correct.

Parent topic: [XCIM](#)

XCIM0058E

error.instantiating.helper=XCIM0058E: An error occurred while accessing helper class "{0}" specified by installation package "{1}". Error detail: {2}

Explanation

There was a problem with accessing the specified helper class. The included error detail might provide further details.

User response

Contact the provider of the software package to verify the package descriptor associated with the software package is correct.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0059E

unknown.parameter.for.helper=XCIM0059E: "{0}" is not a valid parameter name for the helper class {1}.

Explanation

An invalid parameter name was passed the helper function.

User response

Contact an IBM support representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0060E

non.string.type.supplied.as.adminId=XCIM0060E: A non-string value was specified for the user name.

Explanation

Only a string value type can be specified as the user name.

User response

Enter a valid user name and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0061E

zero.length.or.blank.adminId.not.allowed=XCIM0061E: An empty string was specified as a value for the user name parameter.

Explanation

An empty string is an invalid value for the the user name parameter.

User response

Enter a valid user name and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0062E

non.string.type.supplied.as.password=XCIM0062E: A non-string value was specified for the password.

Explanation

Only a string value type can be specified.

User response

Enter a valid password and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0063E

zero.length.or.blank.password.not.allowed=XCIM0063E: An empty string was specified for the password.

Explanation

An empty string is an invalid value for the password parameter.

User response

Enter a valid password and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0064E

error.persisting.file=XCIM0064E: An exception occurred while writing data to file "{0}". Exception: {1}

Explanation

An unexpected error occurred while writing data to the file.

User response

Refer to the exception details to correct the problem, or contact an IBM support representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0065E

error.while.sending.notification=XCIM0065E: An error occurred while sending a notification. Exception: {0}

Explanation

An unexpected error occurred while sending a notification.

User response

Report the problem to an IBM support representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0066E

failed.to.run.startNode.due.to.error=XCIM0066E: Unable to initiate the startNode command on host {0} due to an error. Exception: {1}

Explanation

The centralized installation manager was unable to initiate the startNode command because of the specified error.

User response

Manually run the startNode command on the specified host by logging into the remote host.

Parent topic: [XCIM](#)

XCIM0067E

installed.updi.version.not.known=XCIM0067E: The version of the Update Installer that is present on host {0} in location "{1}" is unknown.

Explanation

The controller cannot determine the version of the Update Installer that is present on the target host due to a missing UPDI.product file.

User response

The Update Installer on the target host might be corrupted. Consider uninstalling the current version and installing a new version of the Update Installer.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0068E

installed.updi.version.incompatible=XCIM0068E: The version of the Update Installer that is present on host {0} in location "{1}" is {2}. The minimum version required by the maintenance package "{3}" is {4}.

Explanation

The version of the Update Installer that is present on the target host is not at the required level.

User response

Install a version of the Update Installer that meets the minimum requirement, and retry the operation.

Parent topic: [XCIM](#)

XCIM0069E

error.retrieving.file.from.host=XCIM0069E: An error occurred while retrieving file "{0}" from host {1}. Exception: {2}

Explanation

An error occurred during the retrieval of a file from the remote host.

User response

Review the exception details, correct the problem, and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0070E

post.install.failed.details.in.log=XCIM0070E: The post-installation command [{0}] failed. For more information, refer to "{1}".

Explanation

The specified post-installation command was issued, but a failure was reported. For details, refer to the specified log file.

User response

Review the messages in the specified log, correct the error, and retry the operation.

Parent topic: [XCIM](#)

XCIM0071E

error.during.uninstall=XCIM0071E: An error occurred while uninstalling package "{0}" on host {1}. Error detail: {2}

Explanation

The uninstallation process was stopped due to an unexpected error.

User response

Review the error message, correct the error, and retry the operation.

Parent topic: [XCIM](#)

XCIM0072E

host.install.loc.invalid.as.uninstall.target=XCIM0072E: Installation location "{0}" on host {1} does not have the correct version of the package "{2}" required for the remote uninstallation process. Package version: {3}

Explanation

The specified location on the target host does not contain the same level of the software specified by the package. The uninstallation process was stopped.

User response

If the software version has changed due to maintenance updates, uninstall the maintenance to return the software package to the original version, and restart the uninstallation process.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0073E

uninstall.failed.with.stderr=XCIM0073E: The uninstallation command [{0}] failed. The following error message was returned: "{1}"

Explanation

The specified uninstallation command was issued, but an error was reported.

User response

Review the error message, correct the error, and restart the process.

Parent topic: [XCIM](#)

XCIM0074E

uninstall.failed.with.nonzero.rc=XCIM0074E: The uninstallation command [{0}] failed. The following error code was returned: {1}.

Explanation

The specified uninstallation command was issued, but a non-zero code was returned.

User response

Review the documentations provided by the provider of the software package for details about the return code. Correct the error and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0075E

uninstall.failed.details.in.log=XCIM0075E: The uninstallation command [{0}] failed. For more information, refer to "{1}".

Explanation

The specified uninstallation command was issued, but a failure was reported. For details, refer to the specified log file.

User response

Review the messages in the specified log, correct the error, and restart the process.

Parent topic: [XCIM](#)

XCIM0076E

uninstall.results.unknown.log.missing=XCIM0076E: The status of the uninstallation command [{0}] cannot be determined, because the output log is missing. Log path: "{1}". Host name: {2}

Explanation

The controller cannot determine the outcome of the uninstallation process, because the output log file is missing.

User response

Report the problem to the provider of the software package.

Parent topic: [XCIM](#)

XCIM0077E

unknown.product.name=XCIM0077E: The product short name "{0}" specified in the installation package descriptor "{1}" is not supported.

Explanation

The installation package descriptor specifies an unsupported product short name.

User response

Report the problem to the package provider.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0078E

pak.file.not.applicable.to.package=XCIM0078E: The .pak file "{0}" cannot be used to update the product specified by the installation package "{1}".

Explanation

The .pak file does not list the product in the descriptor as an installation target. Either the .pak file does not belong to the package, or the .pak file is invalid.

User response

Contact an IBM support representative.

Parent topic: [XCIM](#)

XCIM0079E

invalid.version.string.format=XCIM0079E: An invalid version string "{0}" for product {1} is located in file "{2}".

Explanation

The maintenance.xml file in the specified .pak file has an invalid version string format for the target product.

User response

Contact an IBM support representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0080E

error.checking.if.maintenance.dependency.met=XCIM0080E: An exception occurred while checking if node "{0}" meets the maintenance dependency. Exception: {1}

Explanation

The controller encountered an error while performing the dependency check.

User response

Review the error message, correct the problem, and restart the process.

Parent topic: [XCIM](#)

XCIM0081E

cannot.mix.updates.for.different.platforms=XCIM0081E: Pak files "{0}" and "{1}" are incompatible, because they might be applicable to different operating systems.

Explanation

The names of the two .pak files indicate that they might be applicable to different operating systems. Maintenance updates for different operating systems cannot be installed on the same target host.

User response

The updates must be applied separately on different host. Select one of the two and restart the process.

Parent topic: [XCIM](#)

XCIM0082E

maintenance.descriptor.file.missing=XCIM0082E: Pak file "{0}" is missing a required descriptor: {1}.

Explanation

The specified .pak file is missing the descriptor file that is included in all .pak files for Version 6.0 and above.

User response

Specify a correct .pak file for the Version 6 system.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0083E

cell.has.no.defined.node.on.host=XCIM0083E: The current deployment manager cell contains no node with the WAS_INSTALL_ROOT variable defined on host {0}.

Explanation

The target host does not have a node with the WAS_INSTALL_ROOT variable defined.

User response

Ensure that the host name is entered correctly. The host name must match the host name of a node in the current deployment manager cell with the node that has the WAS_INSTALL_ROOT variable defined.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0084E

cell.has.no.defined.node.on.host.in.loc=XCIM0084E: The current deployment manager cell has no node with the WAS_INSTALL_ROOT variable set to "{1}" on host {0}.

Explanation

The target host does not have a node with the specified WAS_INSTALL_ROOT value.

User response

Ensure that the host name and the installation location are entered correctly.

Parent topic: [XCIM](#)

XCIM0085E

only.boolean.value.is.allowed=XCIM0085E: Only boolean value is allowed for parameter {0}.

Explanation

The specified parameter takes only boolean true or false as values.

User response

Specify either true or false as the values for the specified parameter.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0086E

fromPrereq.true.when.prereq.package.is.null=XCIM0086E: A conflicting attribute value for an installation descriptor is detected.

Explanation

The attribute "FromPrereq" cannot be set to true when the "PrereqPackage" attribute is not set.

User response

Contact the provider of the installation package descriptor to update the descriptor

Parent topic: [XCIM](#)

XCIM0087E

license.file.in.unknown.prereq.package=XCIM0087E: The prerequisite package that contains the license agreement file for package "{0}" cannot be determined.

Explanation

The specified installation package descriptor is defective. It is missing a PrereqPackage attribute that is referenced by other attributes or elements.

User response

Contact the provider of the installation package descriptor to update the descriptor.

Parent topic: [XCIM](#)

XCIM0088E

license.file.not.found.in.prereq.package=XCIM0088E: The license file for package "{0}" cannot be located in the directory of the prerequisite package "{1}".

Explanation

The license agreement and information files for this package cannot be located in the directory of the prerequisite package.

User response

Ensure that the files for the prerequisite package are stored in the centralized installation repository, or ensure that all of the required files were successfully downloaded.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0089E

no.zip.file.with.specified.prefix=XCIM0089E: The .zip file that contains the license file for package "{0}" cannot be located. The prefix for the .zip file name is: {1}.

Explanation

The license file for this package is contained in a .zip file with the specified prefix. No .zip file with that name prefix for the package is found in the centralized installation repository.

User response

Contact the provider of the package, or ensure that all of the required files were successfully downloaded.

Parent topic: [XCIM](#)

XCIM0090E

file.matching.pattern.not.found=XCIM0090E: A file that matches the name pattern "{0}" is not found in directory "{1}".

Explanation

There is no file in the specified directory that matches the name pattern.

User response

Ensure that a file that matches the name pattern exists in the directory specified. You might need to download additional files from the IBM support Web site to the specified directory.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0091E

file.list.cannot.be.empty=XCIM0091E: The parameter "fileList" is required for a maintenance installation and must not be empty.

Explanation

An empty fileList parameter is invalid for a package type that specifies user-selected maintenance.

User response

Specify one or more maintenance .pak files, separated by commas, as the value for the fileList parameter.

Parent topic: [XCIM](#)

XCIM0092E

no.maintenance.file.specified=XCIM0092E: The maintenance installation request was not successful, because no files were selected for installation.

Explanation

The maintenance installation request failed, because the list of maintenance files in the request is empty.

User response

Specify one or more maintenance .pak files, and resubmit the request.

Parent topic: [XCIM](#)

XCIM0093E

selected.maintenance.not.valid.for.node=XCIM0093E: The list of selected maintenance is not valid to install on host {0}. List of selected maintenance: {1}

Explanation

The list of selected maintenance is not valid, because the product installed on the host does not meet the combined dependency.

User response

If the list consists of more than one maintenance files, remove some the files. Alternatively, specify a different host.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0094E

prereq.updi.version.not.known=XCIM0094E: The minimum version of the Update Installer required for installing the selected maintenance cannot be determined. Selected maintenance: {0}

Explanation

The controller cannot determine the minimum version requirement by examining the selected maintenance files.

User response

Contact an IBM support representative.

Parent topic: [XCIM](#)

XCIM0095E

updi.version.txt.missing=XCIM0095E: The Update Installer installation image file "{0}" that is located in directory "{1}" is missing the entry {2}.

Explanation

The Update Installer installation image file is missing a required entry. The entry might have been deleted.

User response

Download the Update Installer from the IBM support Web site again.

Parent topic: [XCIM](#)

XCIM0096E

updi.version.unknown=XCIM0096E: The version of the Update Installer in file "{0}" that is located in directory "{1}" cannot be determined.

Explanation

The controller cannot determine the version of the Update Installer that is located at the specified path.

User response

Download the Update Installer from the IBM support Web site again.

Parent topic: [XCIM](#)

XCIM0097E

updi.version.not.compatible=XCIM0097E: The Update Installer in file "{0}" of directory "{1}" is version {2}. This version does not meet the minimum required version of {3}.

Explanation

The version of the Update Installer in the centralized installation repository does not meet the required version.

User response

Download the minimum required version of the Update Installer from the IBM support Web site.

Parent topic: [XCIM](#)

XCIM0098E

error.while.parsing.maintenance.xml=XCIM0098E: An error occurred while parsing the maintenance.xml file in file "{0}". Exception: {1}

Explanation

An unexpected error occurred while parsing the .xml file in the specifield file.

User response

Contact an IBM support representative.

Parent topic: [XCIM](#)

XCIM0099E

cimgr.controller.not.initialized=XCIM0099E: The centralized installation manager did not initialize, or has not yet completed initialization.

Explanation

The controller did not initialize, or has not yet completed initialization, due to a possible error during the start up process.

User response

Check the deployment manager output log for any errors that are related to the start up of the CIMGrComponentImpl. Correct the error, and restart the deployment manager.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0100I

operation.completed.successfully=XCIM0100I: Operation completed successfully.

Explanation

The operation was completed in a successful manner.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0102E

ERROR_INSTALL_TARGET_HOSTNAME_REQUIRED=XCIM0102E: The host name of the install target {0} is absent.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0103E

ERROR_INSTALL_TARGET_HOSTNAME_INVALID=XCIM0103E: The host name of the install target {0} contains invalid characters.

Explanation

No Additional information is available.

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0104E

ERROR_INSTALL_TARGET_PLATFORM_INVALID=XCIM0104E: The platform of the install target {0} is invalid.

Explanation

Valid platforms are "aix", "hpux", "linux", "solaris", "windows", "os390" and "os400".

User response

Please use either the admin application or the command line scripting tool to update the part of the configuration which is in error.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0121E

uninstall.updi.failed.with.stderr=XCIM0121E: The uninstall of the Update Installer failed with the error message "{1}". Command: {0}

Explanation

The uninstall of the Update Installer was unsuccessful due to error.

User response

Review the error that accompanies this error message. If possible, correct the error and retry the operation.

Parent topic: [XCIM](#)

XCIM0122E

uninstall.updi.failed.with.nonzero.rc=XCIM0122E: The uninstall of the Update Installer failed with the return code "{1}". Command: {0}

Explanation

The uninstall of the Update Installer was unsuccessful due to error.

User response

Review the documentations provided with the Update Installer for the meaning of the return code during uninstall operation. If possible, correct the error and retry the command.

Parent topic: [XCIM](#)

XCIM0123E

uninstall.updi.failed.details.in.log=XCIM0123E: The uninstall of the Update Installer failed. For more information, consult "{0}".

Explanation

The specified uninstall command was run but reported a failure. For detail, consult the specified log file.

User response

Review the messages in the specified log. If possible, correct the error and retry the operation.

Parent topic: [XCIM](#)

XCIM0124E

uninstall.updi.results.unknown.log.missing=XCIM0124E: The successful uninstall of the Update Installer cannot be determined as the output log is missing. Log path: "{0}". Host name: {1}

Explanation

The controller cannot determine the outcome of the uninstall operation as the output log file from the uninstall command is missing.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0125E

file.download.urls.is.empty.but.referenced=XCIM0125E: There is no "FileDownloadURLs" specified but element {0} has a non-empty "DownloadDir" attribute.

Explanation

An install package descriptor has an element with an invalid DownloadDir attribute.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0126E

file.download.url.index.out.of.bound=XCIM0126E: The index number specified by the "DownloadURL" attribute of element {0} exceeds the number of file download URLs specified in the descriptor.

Explanation

An install package descriptor has an element with an invalid DownloadURL attribute.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0127E

maintenance.cmd.operates.only.on.maintenance.pkg=XCIM0127E: The {0} command is only valid for maintenance packages.

Explanation

The specified command can only operate on maintenance packages. Other type of install packages are invalid for this command.

User response

Specify a maintenance package with the command.

Parent topic: [XCIM](#)

XCIM0128E

start.node.failed.with.stderr=XCIM0128E: The "startNode" command failed with the error message "{0}"

Explanation

The "startNode" command was run on the target node but reported an error.

User response

Review the error that accompanies this error message. If possible, correct the error and re-run the command manually on the target host.

Parent topic: [XCIM](#)

XCIM0129E

start.node.failed.with.nonzero.rc=XCIM0129E: The "startNode" command failed with the return code {0}.

Explanation

The "startNode" command was run on the target node but returned with a non-zero return code.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0130E

start.node.failed.with.stdout=XCIM0130E: The "startNode" command failed with the output message: {0}

Explanation

The "startNode" command was run on the target node but reported an error.

User response

Review the error that accompanies this error message. If possible, correct the error and re-run the command manually on the target host.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0131E

post.uninstall.failed.with.stderr=XCIM0131E: The post-uninstall command [{0}] failed with the error message "{1}"

Explanation

The specified post-uninstall command was run but reported an error.

User response

Review the error that accompanies this error message. If possible, correct the error and re-run the command manually on the target host.

Parent topic: [XCIM](#)

XCIM0132E

post.uninstall.failed.with.nonzero.rc=XCIM0132E: The post-uninstall command [{0}] failed with the return code {1}.

Explanation

The specified post-uninstall command was run but returned with a non-zero return code.

User response

Review the documentation provided by the provider of the install package for the meaning of of the return code. If possible, correct the error and re-run the command manually on the target host.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0133E

post.uninstall.failed.with.stdout=XCIM0133E: The post-uninstall command [{0}] failed with the output message: {1}

Explanation

The specified post-uninstall command was run but reported an error.

User response

Review the error that accompanies this error message. If possible, correct the error and re-run the command manually on the target host.

Parent topic: [XCIM](#)

XCIM0134E

no.uninstall.maintenance.file.specified=XCIM0134E: No maintenance file were selected for uninstall.

Explanation

The maintenance uninstall request failed because the user-selected list of maintenance files in the work request is empty.

User response

Specify one or more maintenance pak files and resubmit the request.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0135E

repository.root.undefined=XCIM0135E: The centralized installation repository root is undefined.

Explanation

The path to the centralized installation repository is unknown. The property file defining the repository path may be corrupted.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0136E

error.opening.remoteaccess.log=XCIM0136E: Error opening log file {0}. Logging disabled. Exception: {1}

Explanation

An unexpected error was encountered opening the specified log file. As a result, logging was disabled.

User response

Try to correct the problem indicated by the exception or report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0137E

maintenance.xml.missing.prereq.updi.ver=XCIM0137E: The maintenance.xml in {0} does not specify the required Update Installer version information.

Explanation

The maintenance.xml file in the specified pak file is missing the required version information.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0138E

package.prereq.missing=XCIM0138E: The Install Package Descriptor with the package name of "{0}" is missing a PrereqPackage attribute.

Explanation

Install Package Descriptor for maintenance packages must have the PrereqPackage attribute specified. This attribute is missing from the referenced descriptor.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0139E

package.prereq.name.not.known=XCIM0139E: The prerequisite package named by the Install Package Descriptor "{0}" cannot be located. The name of the prerequisite package is {1}.

Explanation

The PrereqPackage attribute in the specified Package Descriptor refers to another package which does not have a descriptor in place.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0140E

remote.cmd.failed.with.stderr=XCIM0140E: Command [{0}] failed with the error message "{1}"

Explanation

The specified command was run but reported an error.

User response

Review the error that accompanies this error message. If possible, correct the error and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0141E

remote.cmd.failed.with.nonzero.rc=XCIM0141E: Command [{0}] failed with the return code {1}.

Explanation

The specified command was run but returned a non-zero return code indicating error.

User response

Review the documentations provided by the provider of the software package for the meaning of the return code for the particular command. If possible, correct the error and retry the command.

Parent topic: [XCIM](#)

XCIM0142E

remote.cmd.failed.with.stdout=XCIM0142E: Command [{0}] failed with the output message: {1}

Explanation

The specified command was run but reported an error.

User response

Review the error that accompanies this error message. If possible, correct the error and re-run the command manually on the target host.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0143E

product.short.name.missing=XCIM0143E: Either a ProductShortName or a ProductShortNames element must be present. Descriptor filename: {0}.

Explanation

The install package descriptor does not include either a ProductShortName or ProductShortNames element.

User response

Report the problem to the provider of the install package and have the descriptor replaced.

Parent topic: [XCIM](#)

XCIM0144E

pak.entry.name.mandatory.for.zip.bundle=XCIM0144E: "PakEntryName" element is required if element {0} specifies a zip file for its "PakFilename" attribute.

Explanation

PakEntryName elements are missing in an install package descriptor with zip filename as PakFilename attribute value.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0145E

pak.entry.name.has.wrong.filetype=XCIM0145E: Only filename with a ".pak" suffix is allowed in "PakEntryName" elements.

Explanation

One or more PakEntryName elements in the install package descriptor have incorrect filetype.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0146E

pak.entry.info.for.zip.missing=XCIM0146E: The MaintenancePak info for zip file {0} cannot be found.

Explanation

The MaintenancePak info for the specified zip file is missing likely as a result of bad descriptor data.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0147E

unexpected.error.locating.node.for.host=XCIM0147E: An unexpected error occurred while locating the middleware node on host {0} with the install location {1}. Exception = {2}

Explanation

An unexpected error occurred locating configuration information on a remote node.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0148E

maintenance.pak.missing.from.bundle=XCIM0148E: The maintenance pak file {0} is not found in {1}.

Explanation

The named maintenance pak file is not found in the specified zip file.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0149E

require.special.parm.missing=XCIM0149E: Value for the special parameter {0} is undefined. Specify a value for the specified parameter using the "specialParms" command argument.

Explanation

The user did not specify a value for the specified parameter and there is no default value defined.

User response

Retry the command and specify a value for the required parameter using the "specialParms" command argument.

Parent topic: [XCIM](#)

XCIM0150E

maintenance.pkg.not.valid.for.cmd=XCIM0150E: Maintenance package "{0}" is not valid for the "{1}" command.

Explanation

The specified command is designed for non-maintenance packages only and a maintenance package was specified.

User response

Specify a non-maintenance package and retry the command.

Parent topic: [XCIM](#)

XCIM0151E

cannot.mix.updates.for.different.products=XCIM0151E: Pak file "{0}" is incompatible with the other pak files selected as they target different software packages.

Explanation

Cannot apply fixes targeted for different software packages in the same operation.

User response

Maintenance updates for different software packages cannot be installed on the same host in the same operation. The updates must be applied separately. Remove the specified pak file from your selection and retry.

Parent topic: [XCIM](#)

XCIM0152E

cannot.have.both.product.short.name.elements=XCIM0152E: Either a ProductShortName or a ProductShortNames element must be present but not both. Descriptor filename: {0}.

Explanation

The install package descriptor must specify either a ProductShortName element or a ProductShortNames element but not both.

User response

Report the problem to the provider of the install package and have the descriptor replaced.

Parent topic: [XCIM](#)

XCIM0153E

ifix.found.nodcentric.uninstall.aborted=XCIM0153E: Maintenance packages for the software product must be uninstalled first before uninstalling the product. Maintenance for product {0} detected on host {1}: {2}.

Explanation

The software product cannot be uninstalled cleanly without first uninstalling all maintenance packages that have been applied.

User response

Uninstall the maintenance packages listed and then retry uninstalling the product.

Parent topic: [XCIM](#)

XCIM0154E

ifix.found.maintenance.uninstall.aborted=XCIM0154E: {0} version {1} cannot be uninstalled without first uninstalling these interim fixes from host {2}: {3}.

Explanation

The maintenance specified cannot be uninstalled cleanly without first uninstalling the listed interim fixes that have been applied.

User response

Uninstall the interim fixes listed in this error message and then retry uninstalling the maintenance.

Parent topic: [XCIM](#)

XCIM0155E

exception.parsing.xml.on.host=XCIM0155E: An unexpected error occurred accessing and parsing {0} from host {1}. Exception = {2}

Explanation

An unexpected error occurred while accessing and parsing a file from the remote host.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0156E

exception.parsing.maintenance.xml.from.pakfile.on.host=XCIM0156E: An unexpected error occurred while parsing the maintenance.xml file in {0} from host {1}. Exception = {2}

Explanation

An unexpected error occurred while accessing and parsing a file from the remote host.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0157E

installer.element.not.allowed=XCIM0157E: Installer element is not allowed outside of an InstallerGroup when there are feature-dependent InstallerGroup. Descriptor filename: {0}.

Explanation

The install package descriptor contains an illegal Installer element.

User response

Report the problem to the provider of the install package and have the descriptor replaced.

Parent topic: [XCIM](#)

XCIM0158E

missing.installer.element=XCIM0158E: The install package descriptor is missing the mandatory Installer element. Descriptor filename: {0}.

Explanation

The install package descriptor must include one or more Installer elements.

User response

Report the problem to the provider of the install package and have the descriptor replaced.

Parent topic: [XCIM](#)

XCIM0159E

create.cmd.log.failed=XCIM0159E: Failed to create output file for command logging.

Explanation

The centralized installation manager was unable to create an output file for command logging.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0160E

error.creating.file.writer=XCIM0160E: An error was encountered creating a FileWriter for log file {0}. Exception: {1}

Explanation

An unexpected error occurred while creating a FileWriter for logging.

User response

Try to correct the problem indicated by the exception or report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0161I

rmtcmd.started.running.cmdgrp=XCIM0161I: Started running command group on host {0}

Explanation

An informational message to show when and where (at which host) the command group was started.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0162I

rmtcmd.sending.file.to.host=XCIM0162I: Sending script file {0} to host.

Explanation

An informational message to show which script file was sent to the target host.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0163I

rmtcmd.resolved.cmd.path=XCIM0163I: Resolved command path: {0}

Explanation

An informational message to show the resolved command path on the target workstation.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0164I

rmtcmd.running.cmd=XCIM0164I: Running command: {0}

Explanation

An informational message to show when the said command was run.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0165I

rmtcmd.stdout.from.cmd=XCIM0165I: Standard output from command:

Explanation

An informational message to show the standard output from running the command.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0166I

rmtcmd.stderr.from.cmd=XCIM0166I: Standard error from command:

Explanation

An informational message to show the standard error output from running the command.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0167I

rmtcmd.rtncode.from.cmd=XCIM0167I: Return code from command: {0}

Explanation

An informational message to show the return code from running the command.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0168I

rmtcmd.check.stdout.for.success.msg=XCIM0168I: Checking standard output from command for presence of success message: {0}. Check type is "{1}".

Explanation

An informational message to show the success message the runtime was looking for in the standard output from the command.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0169I

rmtcmd.completed=XCIM0169I: Command group completed successfully.

Explanation

An informational message indicating that the command group was run successfully.

User response

None.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0170E

rmtcmd.failed=XCIM0170E: Command group failed as specified success criteria was not met.

Explanation

The run of the command group was considered a failure as the success criteria the runtime was looking for was not met.

User response

Look for clues from the command log file containing this message. Correct the problem if possible and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0171E

rmtcmd.failed.see.log=XCIM0171E: Command group failed. For details refer to log file {0}.

Explanation

The run of the command group was considered a failure. For details look for information contained in the specified log file.

User response

Open the log file with a text editor or view its content from the Administrator Console, Installation history panel.

Parent topic: [XCIM](#)

XCIM0175E

only.one.feature.is.allowed=XCIM0175E: Only one feature can be specified for package "{0}".

Explanation

Only one feature can be specified for installation as the features defined are mutually exclusive features.

User response

Specify only one of the features and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0176E

specify.at.least.one.feature=XCIM0176E: At least one feature must be specified for package "{0}".

Explanation

You must specify at least one of the features defined for the package for installation.

User response

Specify a feature and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0177E

remote.cmd.called.in.the.wrong.state=XCIM0177E: The processing of a remote command occurred out of order. Processing has been halted.

Explanation

The processing of a remote command occurred out of order.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0178E

remote.file.matching.pattern.not.found=XCIM0178E: A file matching the pattern "{0}" cannot be found in directory "{1}" of host {2}.

Explanation

A file matching the specified name pattern cannot be found on the remote host. The name pattern or source directory may be incorrect or the target host is missing a file in the specified directory.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0179E

cannot.resolve.java.install.dir=XCIM0179E: Unable to resolve "{0}" to the actual installation directory for the IBM software developer kit (SDK) on host {1}.

Explanation

The centralized installation manager is unable to resolve the specified string to the actual installation directory for the IBM SDK.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0180E

error.resolving.file.or.dir.on.host=XCIM0180E: An error was encountered while resolving the name pattern "{0}" to a file or directory in directory "{1}" of the remote host {2}. Exception: {3} Cause: {4}

Explanation

An unexpected error was encountered while resolving the specified name pattern to the actual file or directory name on the remote host.

User response

Try to correct the problem indicated by the exception or report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0181E

unknown.java.install.loc.for.platform=XCIM0181E: The install location for the IBM Software Developer Kit (SDK), Java 2 Technology Edition is not known for "{0}".

Explanation

The installation location for the IBM SDK is not known for the specified platform.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0182E

no.middleware.nodes.defined.on.host=XCIM0182E: Target host {0} has no middleware nodes defined.

Explanation

The specified host has no middleware nodes defined. A middleware node is a managed or unmanaged node that is not a deployment manager node.

User response

Define a middleware node on the specified host and then retry the operation or specify a different host.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0183E

node.not.middleware.node.on.host=XCIM0183E: The specified node "{0}" is not a middleware node on host {1}.

Explanation

The specified host does not have a middleware node by that name.

User response

Specify the correct name for the middleware node and make sure the host name is correctly entered.

Parent topic: [XCIM](#)

XCIM0184E

node.invalid.as.wasce.manager=XCIM0184E: The specified node "{0}" is invalid as manager of the WebSphere Application Server Community Edition installation.

Explanation

The specified node does not have the WebSphere Extended Deployment extension to manage the WebSphere Application Server Community Edition installation.

User response

Specify a different node or augment the node with the necessary WebSphere Extended Deployment extension and then retry the operation.

Parent topic: [XCIM](#)

XCIM0185E

optional.parm.missing.for.cmd.helper=XCIM0185E: A required OptionalCommandParms element is missing from the command definition for install package "{0}" and helper class {1}.

Explanation

The install package descriptor for the specified package is missing an optional command parameter element.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0186E

unexpected.error.renaming.remote.logfile=XCIM0186E: An error was encountered renaming log file {0} on host {1}. Exception: {2} Cause: {3}

Explanation

An unexpected error was encountered rename a log file on the target host.

User response

Try to correct the problem indicated by the exception or report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0187E

node.not.found.for.name=XCIM0187E: The Node object cannot be located for the middleware node {0}.

Explanation

The centralized installation manager failed to locate the Node object with the specified node name.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0188E

node.metadata.service.not.available=XCIM0188E: The NodeMetaData service is not available for middleware node {0}. The middleware agent or node agent may have stopped.

Explanation

The centralized installation manager failed to obtain the NodeMetaData service for the specified node.

User response

The middleware agent or node agent for the node may have problem starting. Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0189E

unexpected.error.updating.node.metadata=XCIM0189E: An error was encountered updating the node-metadata for middleware node {0}. Exception: {1} Cause: {2}

Explanation

An unexpected error was encountered updating the node-metadata for the specified node.

User response

Report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0190E

ibm.java.pkg.not.found.with.rpm=XCIM0190E: Cannot find package using command "{0}" on host {1}.

Explanation

The package is no longer registered with the RPM Package Manager on the target host. The package may have been deleted or removed from the system or the command is incorrect.

User response

The package may have been deleted or removed from the system or the command is incorrect. If it is the former no action is needed. If it is the latter report the problem to an IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0191I

license.agreement.not.avail.for.viewing=XCIM0191I: The license agreement file for package "{0}" is not available for direct viewing from the command.

Explanation

The license agreement file for the package is not available for viewing or display from the command line.

User response

If you downloaded the package, the license agreement file may be available for viewing from the download site. If not, contact the provider of the package for assistance.

Parent topic: [XCIM](#)

XCIM0192E

install.failed.with.nonzero.rc.and.stdout=XCIM0192E: The install command [{0}] failed with the return code {1}.
Console output: {2}

Explanation

The specified install command was run but returned a non-zero return code indicating error.

User response

Review the console output and documentation provided by the provider of the install package for the meaning of the return code. If possible, correct the error and retry the command.

Parent topic: [XCIM](#)

XCIM0193I

java.sdk.cannot.be.uninstalled.remotely=XCIM0193I: The IBM software developer kit (SDK) for Windows cannot be uninstalled remotely using the centralized installation manager. Use the Windows "Add or Remove programs" locally to uninstall.

Explanation

Just an informational message to indicate that the IBM SDK was not uninstalled by the centralized installation manager.

User response

If appropriate, use the Windows "Add or Remove programs" locally to uninstall the IBM software developer kit.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0194E

ibm.java2.sdk.already.installed=XCIM0194E: The IBM software developer kit (SDK), Java 2 Technology Edition is already installed on host {0}.

Explanation

The IBM SDK is already installed on the target host.

User response

Do not re-install the SDK on top of an existing installation. Uninstall it first and clean up the directory before proceeding with the installation of the SDK.

Parent topic: [XCIM](#)

XCIM0195E

ibm.java2.may.already.be.installed.on.host=XCIM0195E: The IBM software developer kit (SDK) may already be installed on host {1} as the directory "{0}" is not empty.

Explanation

The specified directory on the target host is not empty. The installation of the SDK was aborted.

User response

Make sure the SDK is uninstalled and the directory is empty. Retry the installation.

Parent topic: [XCIM](#)

XCIM0196E

unexpected.error.checking.dir.empty=XCIM0196E: An error was encountered checking if the directory {0} on host {1} was empty. Exception: {2} Cause: {3}

Explanation

An unexpected error was encountered checking if a directory was empty on the target host.

User response

Try to correct the problem indicated by the exception or report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0197E

package.invalid.for.uninstall.of.non-ifix.pak.type=XCIM0197E: Package "{0}" cannot be used to uninstall maintenance {1} of type {2}.

Explanation

The specified package can only be used to uninstall maintenance of type IFIX.

User response

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0198E

installed.java.home.invalid=XCIM0198E: The specified installation directory "{0}" for a supported Java environment on host {1} is invalid.

Explanation

A supported Java environment cannot be located on the target host at the location specified.

User response

Enter the correct installation directory for a supported Java environment on the target host and resubmit the installation request.

Parent topic: [XCIM](#)

XCIM0199W

install.completed.with.errors=XCIM0199W: Installation completed but errors were detected. Check logs for error messages.

Explanation

The installation completed but there were some errors.

User response

Check the logs for any error and act accordingly.

Parent topic: [XCIM](#)

XCIM0200W

uninstall.completed.with.errors=XCIM0200W: Uninstallation completed but errors were detected. Check logs for error messages.

Explanation

The uninstallation completed but there were some errors.

User response

Check the logs for any error and act accordingly.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0201E

no.middleware.nodes.on.host.as.wasce.mgr=XCIM0201E: No middleware nodes on host {0} meet the minimum requirement of having WebSphere Extended Deployment Version {1} installed.

Explanation

No middleware nodes can be found on the host that meet the stated requirement.

User response

Install or update the WebSphere Extended Deployment product on the host and retry the operation.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0202E

install.failed.cmd.timed.out.with.rc=XCIM0202E: The installation command [{0}] timed out. Return code = {1}.

Explanation

The specified installation command was issued but failed to complete within a specified time. A non-zero return code was also returned.

User response

Review the documentation provided by the provider of the installation package for probable cause. Correct the error, and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0203E

install.failed.cmd.timed.out=XCIM0203E: The installation command [{0}] timed out.

Explanation

The specified installation command was issued but failed to complete within a specified time.

User response

Review the documentation provided by the provider of the installation package for probable cause. Correct the error, and retry the command.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0204E

unexpected.error.stopping.middleware.agent=XCIM0204E: An error was encountered while stopping the middleware agent on host {0}. Installation location = "{1}". Exception: {2}

Explanation

An unexpected error was detected while stopping the middleware agent on the target host.

User response

Review the error message for probable cause and correct if appropriate. If the error persists report the problem to the IBM service representative.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0205E

exception.checking.remote.path=XCIM0205E: An error was encountered checking for the existence of file {0} on host {1}. Exception: {2} Cause: {3}

Explanation

An unexpected error was encountered while checking for the existence of the specified file on the target host.

User response

Correct the problem as indicated by the exception or report the problem to an IBM service representative.

Parent topic: [XCIM](#)

XCIM0216E

user.not.configured.to.sudo.slibclean.with.nopasswd=XCIM0216E: Unable to run "sudo slibclean" as user {0} without a password prompt on target host {1}.

Explanation

A remote operation requires the use of the "sudo" command on the target host in order allow the non-root user to run "slibclean" as root on the target. Also, the "sudo" command must be configured to allow the specified user to run the "slibclean" command without a password prompt.

User response

Configure "sudo" accordingly and resubmit the request.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0217E

user.unable.to.sudo.slibclean=XCIM0217E: Unable to run "sudo slibclean" as user {0} on target host {1}. Error message: {2}

Explanation

A remote operation requires the use of the "sudo" command on the target host in order allow the non-root user to run "slibclean" as root on the target. The included error message explains why the centralized installation manager was unable to run the "sudo slibclean" command on the target.

User response

Correct the problem indicated by the included message and resubmit the request. If the error message indicates that the "sudo" command is not found then most likely it is because "sudo" has not been installed. Check the InfoCenter to see where you can download a copy of "sudo" for installation on your target host.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XCIM0226E

prereq.ifix.missing=XCIM0226E: Interim Fix {0} must be applied before installing {1} if the target host has product {2} installed.

Explanation

The specified Interim Fix must be present on the target installation before upgrading to the new maintenance level.

User response

Download and install the Interim Fix on the target host before retrying the upgrade to the new maintenance level.

Parent topic: [XCIM](#)

[Terms of Use](#) | [Feedback](#)

XHAD

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[XHAD0001I](#)

XHAD0001I=XHAD0001I: The Deployment Manager process is in standby mode.

[XHAD0002I](#)

XHAD0002I=XHAD0002I: The Deployment Manager process is in active mode.

[XHAD0003I](#)

XHAD0003I=XHAD0003I: The JMX SOAP proxy is available at port {0}

[XHAD0004E](#)

XHAD0004E=XHAD0004E: The JMX SOAP proxy failed to start with exception: {0}.

[XHAD0005E](#)

XHAD0005E=XHAD0005E: A Deployment Manager process running in active mode cannot be found.

[XHAD0006E](#)

XHAD0006E=XHAD0006E: This Deployment Manager process is in active mode but does not hold the lock on the file {0}. The Deployment Manager process is being stopped.

[XHAD0007W](#)

XHAD0007W=XHAD0007W: Problem occurred during reloading of tasks.

[XHAD0008I](#)

XHAD0008I=XHAD0008I: Deployment manager node {0} has been set to active mode.

[XHAD0009I](#)

XHAD0009I=XHAD0009I: Deployment manager node {0} has been removed from active mode.

[XHAD1000I](#)

XHAD1000I=XHAD1000I: Usage: xd_hadmgrAdd -hostname <primary_dmgr_host> [-port <primary_dmgr_soap_port>] -configRoot <fully_qualified_path_to_shared_configuration> -workspaceRoot <fully_qualified_path_to_workspace> [-proxyServerJmxSoapAddress <JMX_SOAP_host>: <JMX_SOAP_port>] [-proxyServerHttpPort <HTTP_port> <HTTP_secure_port>] [-uniquePort] [-user <uid>] [-password <pwd>] [-quiet] [-logfile <filename>] [-replacelog] [-trace] [-help]

[XHAD1001I](#)

XHAD1001I=XHAD1001I: Connecting to primary deployment manager host {0} using port {1}

[XHAD1002I](#)

XHAD1002I=XHAD1002I: Begin addition of standby deployment manager node {0} to cell {1}

[XHAD1003I](#)

XHAD1003I=XHAD1003I: Standby deployment manager node {0} has been added successfully to cell {1}

[XHAD1004I](#)

XHAD1004I=XHAD1004I: Please restart the active deployment manager in node {0}

[XHAD1005I](#)

XHAD1005I=XHAD1005I: Please restart the On Demand Router {0}

[XHAD1006I](#)

XHAD1006I=XHAD1006I: Usage: xd_hadmgrRemove -hostname <primary_dmgr_host> [-port <primary_dmgr_soap_port>] [-user <uid>] [-password <pwd>] [-quiet] [-logfile <filename>] [-replacelog] [-trace] [-help]

[XHAD1007I](#)

XHAD1007I=XHAD1007I: Begin removal of standby deployment manager node {0} from cell {1}

[XHAD1008I](#)

XHAD1008I=XHAD1008I: Standby deployment manager node {0} has been removed successfully from cell {1}

[XHAD1011E](#)

XHAD1011E=XHAD1011E: Mandatory parameters missing

[XHAD1012E](#)

XHAD1012E=XHAD1012E: Another xd_hadmgrAdd, xd_hadmgrRemove, addNode or removeNode operation is currently in progress. Please attempt this operation later.

[XHAD1013E](#)

XHAD1013E=XHAD1013E: Could not find text for message ID {0}.

XHAD1014E

XHAD1014E=XHAD1014E: Given host and port do not point to a deployment manager.

XHAD1015E

XHAD1015E=XHAD1015E: Node {0} already exists in primary deployment manager cell {1}.

XHAD1016E

XHAD1016E=XHAD1016E: Node {0} does not meet the minimum version of {1}

XHAD1017E

XHAD1017E=XHAD1017E: The core group that the primary deployment manager belongs to does not contain an On Demand Router

XHAD1018E

XHAD1018E=XHAD1018E: An error occurred during xd_hadmgrAdd.

XHAD1019E

XHAD1019E=XHAD1019E: Cannot read configuration at {0}

XHAD1020E

XHAD1020E=XHAD1020E: Problem reading primary deployment manager configuration: {0}

XHAD1021E

XHAD1021E=XHAD1021E: Node {0} does not exist in primary deployment manager cell {1}.

XHAD1022E

XHAD1022E=XHAD1022E: Cannot remove the last deployment manager from cell {0}.

XHAD1023E

XHAD1023E=XHAD1023E: Cannot remove the primary deployment manager from cell {0}.

XHAD1024E

XHAD1024E=XHAD1024E: An error occurred during xd_hadmgrRemove.

XHAD1025I

XHAD1025I=XHAD1025I: Rolling back to the original configuration.

XHAD1026E

XHAD1026E=XHAD1026E: An error occurred during rollback.

XHAD1027E

XHAD1027E=XHAD1027E: {0} {1} conflicts with an existing port in {2}

XHAD1028I

XHAD1028I=XHAD1028I: {0} is assigned to port {1}

XHAD1029I

XHAD1029I=XHAD1029I: Updating endpoints in On Demand Router {0}

XHAD1030I

XHAD1030I=XHAD1030I: Updating endpoints in deployment manager of node {0}

XHAD1031E

XHAD1031E=XHAD1031E: Option {0} required for creating the first standby deployment manager

XHAD1032E

XHAD1032E=XHAD1032E: The standby deployment manager is running. Stop the standby deployment manager, and run the command again.

XHAD1033E

XHAD1033E=XHAD1033E: Problem reading standby deployment manager configuration: {0}

XHAD0001I

XHAD0001I=XHAD0001I: The Deployment Manager process is in standby mode.

Explanation

An informational message that indicates that this Deployment Manager process is in standby mode.

User response

None.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD0002I

XHAD0002I=XHAD0002I: The Deployment Manager process is in active mode.

Explanation

An informational message that indicates that this Deployment Manager process is in active mode.

User response

None.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD0003I

XHAD0003I=XHAD0003I: The JMX SOAP proxy is available at port {0}

Explanation

An informational message that indicates the availability of the JMX SOAP proxy.

User response

None.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD0004E

XHAD0004E=XHAD0004E: The JMX SOAP proxy failed to start with exception: {0}.

Explanation

The server cannot start the JMX SOAP proxy because of the stated exception.

User response

Examine the exception in this message for problem determination.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD0005E

XHAD0005E=XHAD0005E: A Deployment Manager process running in active mode cannot be found.

Explanation

The JMX SOAP proxy is unable to find a Deployment Manager process running in active mode.

User response

Verify that a Deployment Manager process is running and in active mode.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD0006E

XHAD0006E=XHAD0006E: This Deployment Manager process is in active mode but does not hold the lock on the file {0}. The Deployment Manager process is being stopped.

Explanation

This Deployment Manager process is in active mode but does not hold the lock on the file {0}. The active Deployment Manager process must hold a lock on this file, and if it does not, then it must be stopped.

User response

Verify that the connection between all Deployment Manager processes and the shared device is stable, and then restart this deployment manager.

Parent topic: [XHAD](#)

XHAD0007W

XHAD0007W=XHAD0007W: Problem occurred during reloading of tasks.

Explanation

There was problems during reloading of tasks from harddisk.

User response

None.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD0008I

XHAD0008I=XHAD0008I: Deployment manager node {0} has been set to active mode.

Explanation

An informational message that indicates that Deployment Manager node {0} has been set to active mode.

User response

None

Parent topic: [XHAD](#)

XHAD0009I

XHAD0009I=XHAD0009I: Deployment manager node {0} has been removed from active mode.

Explanation

An informational message that indicates that Deployment Manager node {0} has been removed from active mode. The Deployment Manager process is being stopped.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1000I

XHAD1000I=XHAD1000I: Usage: xd_hadmgrAdd -hostname <primary_dmgr_host> [-port <primary_dmgr_soap_port>] -configRoot <fully_qualified_path_to_shared_configuration> -workspaceRoot <fully_qualified_path_to_workspace> [-proxyServerJmxSoapAddress <JMX_SOAP_host>:<JMX_SOAP_port>] [-proxyServerHttpPort <HTTP_port> <HTTP_secure_port>] [-uniquePort] [-user <uid>] [-password <pwd>] [-quiet] [-logfile <filename>] [-replacelog] [-trace] [-help]

Explanation

This is the usage message.

User response

Specify the parameters according to the usage.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1001I

XHAD1001I=XHAD1001I: Connecting to primary deployment manager host {0} using port {1}

Explanation

xd_hadmgrAdd/xd_hadmgrRemove is trying to establish a connection to primary deployment manager.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1002I

XHAD1002I=XHAD1002I: Begin addition of standby deployment manager node {0} to cell {1}

Explanation

Adding the standby deployment manager to the primary cell.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1003I

XHAD1003I=XHAD1003I: Standby deployment manager node {0} has been added successfully to cell {1}

Explanation

xd_hadmgrAdd program has completed successfully

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1004I

XHAD1004I=XHAD1004I: Please restart the active deployment manager in node {0}

Explanation

Restart the active deployment manager is required for the highly available deployment manager to function correctly.

User response

Please restart the active deployment manager.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1005I

XHAD1005I=XHAD1005I: Please restart the On Demand Router {0}

Explanation

Restart the On Demand Router is required for the highly available deployment manager to function correctly.

User response

Please restart the specified On Demand Router.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1006I

XHAD1006I=XHAD1006I: Usage: xd_hadmgrRemove -hostname <primary_dmgr_host> [-port <primary_dmgr_soap_port>] [-user <uid>] [-password <pwd>] [-quiet] [-logfile <filename>] [-replacelog] [-trace] [-help]

Explanation

This is the usage message.

User response

Specify the parameters according to the usage.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1007I

XHAD1007I=XHAD1007I: Begin removal of standby deployment manager node {0} from cell {1}

Explanation

xd_hadmgrRemove program has started.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1008I

XHAD1008I=XHAD1008I: Standby deployment manager node {0} has been removed successfully from cell {1}

Explanation

xd_hadmgrRemove program has completed successfully.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1011E

XHAD1011E=XHAD1011E: Mandatory parameters missing

Explanation

One or more mandatory parameters are missing.

User response

Please see usage.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1012E

XHAD1012E=XHAD1012E: Another xd_hadmgrAdd, xd_hadmgrRemove, addNode or removeNode operation is currently in progress. Please attempt this operation later.

Explanation

Concurrent xd_hadmgrAdd, xd_hadmgrRemove, addNode and removeNode operations are not allowed.

User response

Wait until the other operation completes, and then retry this operation.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1013E

XHAD1013E=XHAD1013E: Could not find text for message ID {0}.

Explanation

The specified message ID could not be found in the message file.

User response

Please contact IBM support.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1014E

XHAD1014E=XHAD1014E: Given host and port do not point to a deployment manager.

Explanation

The connected process does not have a DeploymentManager MBean. The most common cause is that the host/port specified is not that for the Deployment Manager.

User response

Verify that the host and port passed to the tool are correct.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1015E

XHAD1015E=XHAD1015E: Node {0} already exists in primary deployment manager cell {1}.

Explanation

The node is already exist in the primary deployment manager cell. Node name must be unique.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1016E

XHAD1016E=XHAD1016E: Node {0} does not meet the minimum version of {1}

Explanation

All nodes must meet the minimum version in order for the highly available deployment manager to work correctly.

User response

Upgrade the node to the specified version.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1017E

XHAD1017E=XHAD1017E: The core group that the primary deployment manager belongs to does not contain an On Demand Router

Explanation

At least one On Demand Router must be present in the primary deployment manager's core group.

User response

Define an On Demand Router in the primary deployment manager's core group.

Parent topic: [XHAD](#)

XHAD1018E

XHAD1018E=XHAD1018E: An error occurred during xd_hadmgrAdd.

Explanation

A fatal error occurred during xd_hadmgrAdd and cannot continue.

User response

Please check the log.

Parent topic: [XHAD](#)

XHAD1019E

XHAD1019E=XHAD1019E: Cannot read configuration at {0}

Explanation

Cannot read the configuration documents at the specified location.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1020E

XHAD1020E=XHAD1020E: Problem reading primary deployment manager configuration: {0}

Explanation

Cannot read or find the primary deployment manager configuration documents at the specified location.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1021E

XHAD1021E=XHAD1021E: Node {0} does not exist in primary deployment manager cell {1}.

Explanation

The node does not exist in the primary deployment manager cell. No removal is required.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1022E

XHAD1022E=XHAD1022E: Cannot remove the last deployment manager from cell {0}.

Explanation

The last deployment manager cannot be deleted.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1023E

XHAD1023E=XHAD1023E: Cannot remove the primary deployment manager from cell {0}.

Explanation

The primary deployment manager that is running cannot be removed.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1024E

XHAD1024E=XHAD1024E: An error occurred during xd_hadmgrRemove.

Explanation

A fatal error occurred during xd_hadmgrRemove and cannot continue.

User response

Please check the log.

Parent topic: [XHAD](#)

XHAD1025I

XHAD1025I=XHAD1025I: Rolling back to the original configuration.

Explanation

The original configuration is being restored.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1026E

XHAD1026E=XHAD1026E: An error occurred during rollback.

Explanation

A fatal error occurred during rollback of xd_hadmgrAdd/xd_hadmgrRemove and cannot continue.

User response

Please check the log.

Parent topic: [XHAD](#)

XHAD1027E

XHAD1027E=XHAD1027E: {0} {1} conflicts with an existing port in {2}

Explanation

The new port conflicts with existing ports.

User response

Please use another free port.

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1028I

XHAD1028I=XHAD1028I: {0} is assigned to port {1}

Explanation

This is an information message indicating which port will be used.

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1029I

XHAD1029I=XHAD1029I: Updating endpoints in On Demand Router {0}

Explanation

Endpoints are being updated on this On Demand Router

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1030I

XHAD1030I=XHAD1030I: Updating endpoints in deployment manager of node {0}

Explanation

Endpoints are being updated on this deployment manager

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1031E

XHAD1031E=XHAD1031E: Option {0} required for creating the first standby deployment manager

Explanation

This option is required for creating the first standby deployment manager

User response

Specify this option according to the usage

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1032E

XHAD1032E=XHAD1032E: The standby deployment manager is running. Stop the standby deployment manager, and run the command again.

Explanation

None

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XHAD1033E

XHAD1033E=XHAD1033E: Problem reading standby deployment manager configuration: {0}

Explanation

None

User response

None

Parent topic: [XHAD](#)

[Terms of Use](#) | [Feedback](#)

XJMS

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[XJMS0001E](#)

XJMS_XDFunctionDisable=XJMS0001E: XD functionality disabled.

[XJMS0002I](#)

XJMS_NOJMSServiceInNonServer=XJMS0002I: XD JMS Proxy cannot run in dmgr or nodeagent process.

[XJMS0003E](#)

XJMS_JMSXDMessagePacingAlreadyActive=XJMS0003E: XD JMS message pacing cannot be started.

[XJMS0004I](#)

XJMS_XJMSroxyStarted=XJMS0004I: XD JMS message pacing controller successfully started.

[XJMS0005I](#)

XJMS_XJMSroxyStopped=XJMS0005I: XD JMS message pacing controller successfully stopped.

[XJMS0006E](#)

XJMS_ErrorInvokingRequestChain=XJMS0006E: Error while invoking request filters.

[XJMS0007E](#)

XJMS_ErrorInvokingResponseChain=XJMS0007E: Error while invoking response filters.

[XJMS0008I](#)

XJMS_XDJMSBatchSize=XJMS0008I: XD JMS batch size set to {0}

[XJMS0009E](#)

XJMS_XDJMSFunctionDisable=XJMS0009E: XD JMS functionality disabled.

[XJMS0010E](#)

XJMS_XDJMSError=XJMS0010E: XD JMS System error

[XJMS0011E](#)

XJMS_RequiredServiceNotStarted=XJMS0011E: XD JMS services did not start.

[XJMS0012I](#)

XJMS_XDJMSClassificationStarted=XJMS0012I: XD JMS classification service started.

[XJMS0013I](#)

XJMS_XDJMSClassificationStopped=XJMS0013I: XD JMS classification service stopped.

[XJMS0014I](#)

XJMS_XDJMSARFMStarted=XJMS0014I: XD JMS request control service started.

[XJMS0015I](#)

XJMS_XDJMSARFMStopped=XJMS0015I: XD JMS request control service stopped.

[XJMS0016I](#)

XJMS_XDJMSWSMMServiceStarted=XJMS0016I: XD JMS WSMM service started.

[XJMS0017E](#)

XJMS_JMSReqQFilterODCTreeNull=XJMS0017E: XD JMS request queue filter failed to obtain odc tree.

[XJMS0018E](#)

XJMS_JMSReqQFilterNoClassifyInfo=XJMS0018E: XD JMS classification information is missing for bus {0} and destination {1}.

[XJMS0019E](#)

XJMS_JMSReqStateResumeError=XJMS0019E: XD JMS wsmm filter failed to resume the request for context {0} with attributes {1}

[XJMS0020W](#)

XJMS_XDJMSRequestReceivedBeforeInitialization=XJMS0020W: Message is received before XD JMS message pacer initialization completed. Message correlator is {0}.

[XJMS0021E](#)

XJMS_XDJMSClassifyFailed=XJMS0021E: Unable to map destination and bus to workclass. Aborting the xd classification filter for the current request. Request context information is {0}.

[XJMS0022W](#)

XJMS_XDJMSDestinationMultiModUse=XJMS0022W: Destination {0} is being used by more than one module. XD mapping will be associated with module {1}.

XJMS0001E

XJMS_XDFunctionDisable=XJMS0001E: XD functionality disabled.

Explanation

XD functionality is disabled on the server.

User response

Enable XD functionality.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0002I

XJMS_NOJMSServiceInNonServer=XJMS0002I: XD JMS Proxy cannot run in dmgr or nodeagent process.

Explanation

XD JMS Proxy cannot run in dmgr or nodeagent process.

User response

none.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0003E

XJMS_JMSXDMessagingPacingAlreadyActive=XJMS0003E: XD JMS message pacing cannot be started.

Explanation

Error while starting XD JMS message pacing controller. Instance of controller already registered with Messaging Engine.

User response

Report program defect to IBM.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0004I

XJMS_XJMSroxyStarted=XJMS0004I: XD JMS message pacing controller successfully started.

Explanation

XD JMS message pacing controller successfully started on the server.

User response

none.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0005I

XJMS_XJMSroxyStopped=XJMS0005I: XD JMS message pacing controller successfully stopped.

Explanation

XD JMS message pacing controller successfully stopped on the server.

User response

none.

Parent topic: [XJMS](#)

XJMS0006E

XJMS_ErrorInvokingRequestChain=XJMS0006E: Error while invoking request filters.

Explanation

Internal error happened while handling incoming request.

User response

Report program defect to IBM.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0007E

XJMS_ErrorInvokingResponseChain=XJMS0007E: Error while invoking response filters.

Explanation

Internal error happened in post processing the request.

User response

Report program defect to IBM.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0008I

XJMS_XDJMSBatchSize=XJMS0008I: XD JMS batch size set to {0}

Explanation

JMS message batch size is set by XD JMS message pacing controller.

User response

none

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0009E

XJMS_XDJMSFunctionDisable=XJMS0009E: XD JMS functionality disabled.

Explanation

XD JMS functionality is disabled on the server.

User response

Enable XD JMS functionality.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0010E

XJMS_XDJMSError=XJMS0010E: XD JMS System error

Explanation

XD JMS System internal error occurred.

User response

Report program defect to IBM.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0011E

XJMS_RequiredServiceNotStarted=XJMS0011E: XD JMS services did not start.

Explanation

Services needed for XD JMS functionality are not currently running.

User response

Report program defect to IBM.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0012I

XJMS_XDJMSClassificationStarted=XJMS0012I: XD JMS classification service started.

Explanation

XD JMS classification service started successfully on the server.

User response

none.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0013I

XJMS_XDJMSClassificationStopped=XJMS0013I: XD JMS classification service stopped.

Explanation

XD JMS classification service stopped successfully on the server.

User response

none.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0014I

XJMS_XDJMSARFMStarted=XJMS0014I: XD JMS request control service started.

Explanation

XD JMS request control service started successfully on the server.

User response

none.

Parent topic: [XJMS](#)

XJMS0015I

XJMS_XDJMSARFMStopped=XJMS0015I: XD JMS request control service stopped.

Explanation

XD JMS request control service stopped successfully on the server.

User response

none.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0016I

XJMS_XDJMSWSMMServiceStarted=XJMS0016I: XD JMS WSMM service started.

Explanation

XD JMS WSMM service started successfully on the server.

User response

none.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0017E

XJMS_JMSReqQFilterODCTreeNull=XJMS0017E: XD JMS request queue filter failed to obtain odc tree.

Explanation

Error while initializing odc tree inside request queue filter.

User response

Report program defect to IBM.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0018E

XJMS_JMSReqQFilterNoClassifyInfo=XJMS0018E: XD JMS classification information is missing for bus {0} and destination {1}.

Explanation

Classification information for the destination is missing in the current context.

User response

Report program defect to IBM.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0019E

XJMS_JMSReqStateResumeError=XJMS0019E: XD JMS wsmm filter failed to resume the request for context {0} with attributes {1}

Explanation

Error while resuming the suspended request.

User response

Report program defect to IBM.

Parent topic: [XJMS](#)

[Terms of Use](#) | [Feedback](#)

XJMS0020W

XJMS_XDJMSRequestReceivedBeforeInitialization=XJMS0020W: Message is received before XD JMS message pacer initialization completed. Message correlator is {0}.

Explanation

Work is delivered before XD JMS message pacer initialization completed. Work will not be throttled by XD.

User response

Deliver messages after server initialization completes.

Parent topic: [XJMS](#)

XJMS0021E

XJMS_XDJMSClassifyFailed=XJMS0021E: Unable to map destination and bus to workclass. Aborting the xd classification filter for the current request. Request context information is {0}.

Explanation

Unable to determine module using the current destination and bus.

User response

Make sure destination and bus name being consumed is configured in work class of the application.

Parent topic: [XJMS](#)

XJMS0022W

XJMS_XDJMSDestinationMultiModUse=XJMS0022W: Destination {0} is being used by more than one module. XD mapping will be associated with module {1}.

Explanation

Destination is being used by more than one module. Optimal solution is provided when a destination is consumed by a single module.

User response

Make sure a given destination is consumed by a single module for optimal results.

Parent topic: [XJMS](#)

XNMI

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[XNMI0001E](#)

node.speed.invalid=XNMI0001E: NodeSpeed "{0}" received from remote agent is not of the correct format. Default value of {1} is used instead.

[XNMI0002E](#)

number.of.cpus.invalid=XNMI0002E: NumberOfCPUs "{0}" received from remote agent is not of the correct format. Default value of {1} is used instead.

[XNMI0003E](#)

remote.agent.port.invalid=XNMI0003E: Remote agent port number "{0}" is invalid. Default port number {1} used instead.

[XNMI0004E](#)

bad.response.from.remote.agent=XNMI0004E: Bad response from remote agent: {0}

[XNMI0010I](#)

service.started.successfully=XNMI0010I: The external node stats service started successfully.

[XNMI0011E](#)

node.speed.property.value.invalid=XNMI0011E: The "{0}" custom property value is not of the correct format. Default value of {1} is used instead.

[XNMI0012E](#)

number.of.cpus.property.value.invalid=XNMI0012E: The "{0}" custom property value is not of the correct format. Default value of {1} is used instead.

[XNMI0013E](#)

error.encountered.loading.plugin=XNMI0013E: An error was encountered loading the class "{0}". Cause: {1}.

[XNMI0014E](#)

stats.collector.impl.class.invalid=XNMI0014E: Class "{0}" identified by the custom property "{1}" does not implement the "{2}" interface.

[XNMI0015E](#)

named.unmanaged.node.not.found=XNMI0015E: There is no node with name "{0}" in cell "{2}". No statistics will be reported by the node.

XNMI0001E

node.speed.invalid=XNMI0001E: NodeSpeed "{0}" received from remote agent is not of the correct format. Default value of {1} is used instead.

Explanation

A value other than a floating point number is received from the remote agent as the nodeSpeed.

User response

If the remote agent is supplied by IBM, report the error to IBM support.

Parent topic: [XNMI](#)

XNMI0002E

number.of.cpus.invalid=XNMI0002E: NumberOfCPUs "{0}" received from remote agent is not of the correct format. Default value of {1} is used instead.

Explanation

A value other than an integer number is received from the remote agent as the number of CPUs running in the remote node.

User response

If the remote agent is supplied by IBM, report the error to IBM support.

Parent topic: [XNMI](#)

XNMI0003E

remote.agent.port.invalid=XNMI0003E: Remote agent port number "{0}" is invalid. Default port number {1} used instead.

Explanation

A value other than an integer number is specified as the value for the custom property "stats.collector.remote.agent.port".

User response

Specify an integer number as the value for the custom property "stats.collector.remote.agent.port" of the unmanaged node.

Parent topic: [XNMI](#)

[Terms of Use](#) | [Feedback](#)

XNMI0004E

bad.response.from.remote.agent=XNMI0004E: Bad response from remote agent: {0}

Explanation

An unexpected response is received from the remote agent.

User response

If the remote agent is supplied by IBM, report the error to IBM support.

Parent topic: [XNMI](#)

[Terms of Use](#) | [Feedback](#)

XNMI0010I

service.started.successfully=XNMI0010I: The external node stats service started successfully.

Explanation

The external node stats service started successfully and is available to service requests.

User response

No user action is required.

Parent topic: [XNMI](#)

XNMI0011E

node.speed.property.value.invalid=XNMI0011E: The "{0}" custom property value is not of the correct format. Default value of {1} is used instead.

Explanation

A value other than a floating point number is specified as the custom property value.

User response

Specify a floating point number as the value for the custom property.

Parent topic: [XNMI](#)

[Terms of Use](#) | [Feedback](#)

XNMI0012E

number.of.cpus.property.value.invalid=XNMI0012E: The "{0}" custom property value is not of the correct format. Default value of {1} is used instead.

Explanation

A value other than an integer number is specified as the custom property value.

User response

Specify an integer number as the value for the custom property.

Parent topic: [XNMI](#)

[Terms of Use](#) | [Feedback](#)

XNMI0013E

error.encountered.loading.plugin=XNMI0013E: An error was encountered loading the class "{0}". Cause: {1}.

Explanation

The system was not able to load the named class because of the problem identified in the message.

User response

Remedy the problem according to the message. If the cause of the problem is ClassNotFoundException, make sure that the name of the class is spelled correctly and that the class is in the Application Server classpath.

Parent topic: [XNMI](#)

[Terms of Use](#) | [Feedback](#)

XNMI0014E

stats.collector.impl.class.invalid=XNMI0014E: Class "{0}" identified by the custom property "{1}" does not implement the "{2}" interface.

Explanation

The class named by the particular custom property of the unmanaged node must implements the specified interface.

User response

Specify the correct class or remove the custom property from the unmanaged node so the system will use a default class for this operation.

Parent topic: [XNMI](#)

[Terms of Use](#) | [Feedback](#)

XNMI0015E

named.unmanaged.node.not.found=XNMI0015E: There is no node with name "{0}" in cell "{2}". No statistics will be reported by the node.

Explanation

The node custom property of a Generic Server Cluster endpoint specifies a node that is not defined in the specified cell.

User response

Either remove or correct the node name in the node custom property of the Generic Server Cluster endpoint or define a node with that name in the specified cell.

Parent topic: [XNMI](#)

[Terms of Use](#) | [Feedback](#)

XREP

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

[XREP0001I](#)

CHECKPOINT_INITIATED=XREP0001I: Creation of full checkpoint {0} initiated.

[XREP0002I](#)

CHECKPOINT_COMPLETED=XREP0002I: Creation of full checkpoint {0} completed.

[XREP0003I](#)

RESTORE_INITIATED=XREP0003I: Restoration of checkpoint {0} initiated.

[XREP0004I](#)

RESTORE_COMPLETED=XREP0004I: Restoration of checkpoint {0} completed.

[XREP0005I](#)

DELTA_INITIATED=XREP0005I: Creation of delta checkpoint {0} initiated.

[XREP0006I](#)

DELTA_COMPLETED=XREP0006I: Creation of delta checkpoint {0} completed.

[XREP0007I](#)

DELETE_INITIATED=XREP0007I: Deletion of checkpoint {0} initiated.

[XREP0008I](#)

DELETE_COMPLETED=XREP0008I: Deletion of checkpoint {0} completed.

[XREP0009E](#)

CHECKPOINT_ERROR=XREP0009E: An error occurred while creating the full checkpoint {0}: {1}.

[XREP0010W](#)

DELTA_ERROR=XREP0010W: An error occurred while creating the delta checkpoint {0}: {1}.

[XREP0011E](#)

RESTORE_ERROR=XREP0011E: An error occurred while restoring the checkpoint {0}: {1}.

[XREP0012E](#)

DELETE_ERROR=XREP0012E: An error occurred while deleting the checkpoint {0}: {1}.

[XREP0013E](#)

REPOSITORY_LOCKED=XREP0013E: The repository is locked by another user.

[XREP0014W](#)

ATTR_GET_FAILED=XREP0014W: Failed to retrieve the attribute {0}: {1}.

[XREP0015W](#)

ATTR_SET_FAILED=XREP0015W: Failed to set the attribute {0}: {1}.

[XREP0016E](#)

COMPONENT_START_ERROR=XREP0016E: Error(s) occurred when starting the repository checkpoint component.

[XREP0017I](#)

CHECKPOINT_PROGRESS=XREP0017I: {0} of {1} documents saved.

[XREP0018I](#)

RESTORE_PROGRESS=XREP0018I: {0} of {1} documents restored.

[XREP0019I](#)

RESTORE_PREPARATION=XREP0019I: Preparing {0} documents for restoration.

[XREP0020I](#)

RESTORE_RESTORATION=XREP0020I: Restoring {0} documents.

[XREP0021E](#)

INVALID_CHECKPOINT_NAME=XREP0021E: Invalid checkpoint name. Checkpoint name should not contain | ? * < " : > /.

XREP0001I

CHECKPOINT_INITIATED=XREP0001I: Creation of full checkpoint {0} initiated.

Explanation

An informational message that indicates that a full checkpoint operation has been initiated.

User response

No user action is required.

Parent topic: [XREP](#)

XREP0002I

CHECKPOINT_COMPLETED=XREP0002I: Creation of full checkpoint {0} completed.

Explanation

An informational message that indicates that a full checkpoint operation has been completed.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0003I

RESTORE_INITIATED=XREP0003I: Restoration of checkpoint {0} initiated.

Explanation

An informational message that indicates that a checkpoint restoration has been initiated.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0004I

RESTORE_COMPLETED=XREP0004I: Restoration of checkpoint {0} completed.

Explanation

An informational message that indicates that a checkpoint restoration has been completed.

User response

No user action is required.

Parent topic: [XREP](#)

XREP0005I

DELTA_INITIATED=XREP0005I: Creation of delta checkpoint {0} initiated.

Explanation

An informational message that indicates that a delta checkpoint operation has been initiated.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0006I

DELTA_COMPLETED=XREP0006I: Creation of delta checkpoint {0} completed.

Explanation

An informational message that indicates that a delta checkpoint operation has been completed.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0007I

DELETE_INITIATED=XREP0007I: Deletion of checkpoint {0} initiated.

Explanation

An informational message that indicates that a checkpoint deletion has been initiated.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0008I

DELETE_COMPLETED=XREP0008I: Deletion of checkpoint {0} completed.

Explanation

An informational message that indicates that a checkpoint deletion has been completed.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0009E

CHECKPOINT_ERROR=XREP0009E: An error occurred while creating the full checkpoint {0}: {1}.

Explanation

The system cannot create the full checkpoint successfully due to the exception recorded.

User response

Please follow the nested exception to determine the cause of the problem.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0010W

DELTA_ERROR=XREP0010W: An error occurred while creating the delta checkpoint {0}: {1}.

Explanation

The system cannot create the delta checkpoint successfully due to the exception recorded.

User response

Please follow the nested exception to determine the cause of the problem.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0011E

RESTORE_ERROR=XREP0011E: An error occurred while restoring the checkpoint {0}: {1}.

Explanation

The system cannot restore the checkpoint successfully due to the exception recorded.

User response

Please follow the nested exception to determine the cause of the problem.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0012E

DELETE_ERROR=XREP0012E: An error occurred while deleting the checkpoint {0}: {1}.

Explanation

The system cannot delete the checkpoint successfully due to the exception recorded.

User response

Please follow the nested exception to determine the cause of the problem.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0013E

REPOSITORY_LOCKED=XREP0013E: The repository is locked by another user.

Explanation

The current operation cannot be completed because the configuration repository is locked by another user.

User response

Please retry at a later time.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0014W

ATTR_GET_FAILED=XREP0014W: Failed to retrieve the attribute {0}: {1}.

Explanation

The system failed to retrieve the specified attribute due to the exception recorded.

User response

Please follow the nested exception to determine the cause of the problem or retry at a later time.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0015W

ATTR_SET_FAILED=XREP0015W: Failed to set the attribute {0}: {1}.

Explanation

The system failed to set the specified attribute due to the exception recorded.

User response

Please follow the nested exception to determine the cause of the problem or retry at a later time.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0016E

COMPONENT_START_ERROR=XREP0016E: Error(s) occurred when starting the repository checkpoint component.

Explanation

This exception is unexpected. The cause is not immediately known.

User response

If the problem persists, see problem determination information on the WebSphere Application Server Support page at <http://www.ibm.com/software/webservers/appserv/was/support/>.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0017I

CHECKPOINT_PROGRESS=XREP0017I: {0} of {1} documents saved.

Explanation

An informational message that indicates the progress of a checkpoint creation.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0018I

RESTORE_PROGRESS=XREP0018I: {0} of {1} documents restored.

Explanation

An informational message that indicates the progress of a checkpoint restoration.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0019I

RESTORE_PREPARATION=XREP0019I: Preparing {0} documents for restoration.

Explanation

An informational message that indicates the progress of a checkpoint restoration.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0020I

RESTORE_RESTORATION=XREP0020I: Restoring {0} documents.

Explanation

An informational message that indicates the progress of a checkpoint restoration.

User response

No user action is required.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XREP0021E

INVALID_CHECKPOINT_NAME=XREP0021E: Invalid checkpoint name. Checkpoint name should not contain | ? * < " : > /.

Explanation

The checkpoint name specified contains invalid characters.

User response

Specify a checkpoint name without the invalid characters.

Parent topic: [XREP](#)

[Terms of Use](#) | [Feedback](#)

XXXX

When you encounter a message in a log or other parts of the product interface, look up the message by its message ID to find out more information.

XXXXX0001E

giop.filterInitException=XXXXX0001E: An exception occurred when constructing the filterChain : {0}

XXXXX0001E

giop.filterInitException=XXXXX0001E: An exception occurred when constructing the filterChain : {0}

Explanation

An exception was caught when initializing the filter chain.

User response

Investigate the cause of the filter chain exception.

Parent topic: [XXXX](#)

- [IBM WebSphere Virtual Enterprise Version 7.0](#)
- [Product overview](#)
 - [What is new in this release](#)
 - [z/OS considerations](#)
 - [Virtualization and WebSphere Virtual Enterprise](#)
 - [Configuring AIX 5.3 and AIX 6.1 on POWER5 and POWER6 \(Micro-Partitioning\)](#)
 - [WebSphere Virtual Enterprise in a micro-partitioned environment](#)
 - [Supported server virtualization environments](#)
 - [Virtualization options](#)
 - [Deploying the product on Solaris 10 Zones with shared CPU](#)
 - [Dynamic operations](#)
 - [Overview of dynamic operations](#)
 - [Overview of request flow prioritization](#)
 - [Dynamic operations environment](#)
 - [Components of dynamic operations](#)
 - [Operational policies](#)
 - [Elasticity mode](#)
 - [Extended manageability](#)
 - [Supported middleware server types](#)
 - [Middleware nodes and servers](#)
 - [Middleware agent](#)
- [Installing and customizing the product](#)
 - [Hardware and software requirements](#)
 - [Distributed operating systems](#)
 - [Using Installation Manager to install the product](#)
 - [Using a response file to install the product](#)
 - [Using ISMP programs to install the product](#)
 - [Installation scenarios](#)
 - [Virtual Enterprise topology](#)
 - [Installing in a WebSphere Application Server Network Deployment environment](#)
 - [Installing the middleware agent](#)
 - [Installing the product silently](#)
 - [Installation parameters](#)
 - [Using the centralized installation manager repository tool to add products](#)
 - [Using the Installation Factory plug-in to create and install customized packages](#)
 - [Build definition file](#)
 - [Creating a build definition file and generating a CIP](#)
 - [Installing a CIP](#)
 - [Installing a CIP to apply maintenance to an existing product installation](#)
 - [Uninstalling CIP updates from an existing product installation](#)
 - [Creating a build definition file and generating an IIP](#)
 - [Installing an IIP](#)
 - [Modifying an existing build definition file for an IIP](#)
 - [Silently installing a CIP or an IIP](#)
 - [Creating and augmenting profiles](#)
 - [Using the graphical user interface to create profiles](#)
 - [Creating profiles for multiple products](#)
 - [Creating profiles for use with other IBM products](#)
 - [Using the graphical user interface to augment profiles](#)
 - [manageprofiles command](#)
 - [Non-root profile creation and augmentation](#)
 - [Uninstalling the product](#)
- [z/OS](#)
 - [Installing the WebSphere Customization Toolbox](#)
 - [Generating customization definitions](#)

- [Uploading and running customized jobs](#)
 - [Uninstalling the product on z/OS](#)
- [Migrating WebSphere Virtual Enterprise](#)
 - [Distributed operating systems](#)
 - [VEUpgrade command](#)
 - [Migrating large topologies of external WebSphere application servers](#)
 - [Migration toolkit](#)
 - [Using the interim fix tool to check for maintenance packages](#)
 - [Using the Update Installer to install maintenance packages](#)
 - [Updating the middleware agent](#)
 - [z/OS](#)
 - [Migrating the product from Version 6.1.0.5 or Version 6.1.1 to Version 7.0](#)
 - [Migrating to a Version 7.0 for z/OS cell](#)
 - [Migrating the deployment manager for z/OS](#)
 - [Migrating an application server node for z/OS](#)
 - [VEUpgrade command](#)
- [Preparing the hosting environment for dynamic operations](#)
 - [Configuring VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)
 - [VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)
 - [Advanced configuration for VMware Infrastructure 3 platforms and WebSphere Virtual Enterprise](#)
 - [Federating middleware nodes](#)
 - [Creating and configuring ODRs](#)
 - [Creating ODRs](#)
 - [Configuring ODRs](#)
 - [Configuring a Web server as a trusted proxy server](#)
 - [Enabling cell affinity](#)
 - [Cell affinity function](#)
 - [Defining generic server clusters for remote ODR cells](#)
 - [Configuring cell affinity in a multi-tiered environment](#)
 - [Configuring the multi-cluster routing policy](#)
 - [Generating a plugin-cfg.xml](#)
 - [Cell affinity when an ODR fails](#)
 - [Using the binary trace facility](#)
 - [Options for filtering information in the trace-cfg.xml](#)
 - [Tuning the on demand router \(ODR\)](#)
 - [Modifying the JVM heap size for the on demand router](#)
 - [Tuning ODR persistent connections](#)
 - [Tuning ODR maximum connections](#)
 - [Configuring custom logs](#)
 - [Custom logs overview](#)
 - [Configuring SSL offload for all HTTPS traffic](#)
 - [Configuring SSL offload for partial HTTPS traffic](#)
 - [Integrating the SIP on demand router with Load Balancer](#)
 - [Defining routing policies for generic server clusters](#)
 - [Configuring rewriting rules](#)
 - [Defining service policies for generic server clusters](#)
 - [Configuring communication between core groups that are in the same cell](#)
 - [BBSO bulletin board](#)
 - [Configuring WebSphere Virtual Enterprise for cross-cell communication](#)
 - [Enabling communication between cells that have security enabled](#)
 - [Configuring an ODR to dynamically update the web server plug-in configuration](#)
 - [Generating the plug-in configuration in a high availability environment](#)
 - [Segregating HTTP traffic by ODR clusters](#)
 - [Controlling the generation of the plugin-cfg.xml file](#)
 - [Propagating the plugin-cfg.xml file to web servers](#)
 - [Routing requests directly from a web server to a back-end application server](#)
 - [Configuring passive HTTP session affinity in the on demand router](#)

- [Configuring the on demand router for multi-cluster failover and load balancing routing](#)
 - [Configuring ODR OutOfMemory prevention](#)
 - [Creating dynamic clusters](#)
 - [Creating a static cluster of ODRs](#)
 - [Creating a dynamic cluster of ODRs](#)
 - [Dynamic clusters](#)
 - [Static clusters versus dynamic clusters](#)
 - [Dynamic cluster isolation](#)
 - [Configuring application lazy start](#)
 - [Application lazy start](#)
 - [The lazy start controller](#)
 - [Configuring vertical stacking](#)
 - [Determining the stacking number for a dynamic cluster](#)
 - [Configuring a dynamic cluster with homogeneous nodes to support vertical stacking](#)
 - [Configuring a dynamic cluster with heterogeneous nodes to support vertical stacking](#)
 - [HTTP session rebalancing](#)
 - [Adding middleware servers to configurations](#)
 - [Adding complete life-cycle middleware servers](#)
 - [PHP servers and dynamic clusters](#)
 - [WebSphere Application Server Community Edition servers](#)
 - [WebSphere Application Server Community Edition servers](#)
 - [Configuring JMX security for WebSphere Application Server Community Edition Version 2.1 servers](#)
 - [Adding assisted life-cycle middleware servers](#)
 - [WebSphere Application Server Community Edition servers](#)
 - [Apache Tomcat servers](#)
 - [BEA WebLogic servers](#)
 - [JBoss servers](#)
 - [Custom HTTP servers](#)
 - [External WebSphere application servers](#)
 - [Managing middleware servers](#)
 - [Configuring middleware descriptors](#)
 - [Creating middleware server templates](#)
 - [Configuring middleware server operations](#)
 - [Viewing middleware server configuration files](#)
 - [Configuring the external log viewing service](#)
 - [Deploying applications with defined service levels](#)
 - [Deploying enterprise applications](#)
 - [Deploying unmanaged Web applications](#)
 - [Deploying PHP applications](#)
 - [Deploying WebSphere Application Server Community Edition applications](#)
 - [Routing and service policies](#)
 - [Defining a service policy](#)
 - [Configuring service policies without response time goals](#)
 - [Subexpression builder operands for routing and service policies](#)
 - [SOAP operands](#)
 - [SIP operands](#)
 - [IIOP operands](#)
 - [HTTP operands](#)
 - [Service policy rules](#)
 - [Rules support](#)
 - [Routing policy action types](#)
 - [Overview of work classes](#)
 - [Work class types](#)
 - [Request classification operators](#)
 - [Rule-based request classification](#)
- [Deploying and managing application editions without loss of service](#)
 - [Application edition manager concepts](#)

- [Application edition manager](#)
 - [Application edition manager states](#)
 - [Operational environment](#)
 - [Edition compatibility](#)
- [Installing an application edition](#)
- [Activating an edition](#)
 - [Activating concurrent application editions](#)
- [Creating routing policies for application editions](#)
- [Validating an edition](#)
 - [Canceling an application validation](#)
- [Performing a rollout on an edition](#)
 - [Algorithm for performing a rollout](#)
- [Performing a rollback on an edition](#)
- [Deleting an installed edition](#)
- [Troubleshooting application edition manager](#)
 - [Configuring timeout values for application edition manager on z/OS systems](#)
 - [Application edition manager frequently asked questions](#)
- [Managing the WebSphere Virtual Enterprise environment](#)
 - [Setting maintenance mode](#)
 - [Routing to servers that are in maintenance mode](#)
 - [Manually replacing SSL certificates for the middleware agent](#)
 - [Fine-grained security](#)
 - [Generating Simple Network Management Protocol \(SNMP\) traps](#)
 - [Configuring the autonomic managers](#)
 - [Configuring the autonomic request flow manager](#)
 - [Rate-based autonomic request flow manager \(ARFM\)](#)
 - [Configuring emergency throttle](#)
 - [Memory overload protection](#)
 - [Configuring memory overload protection](#)
 - [Trace settings for autonomic request flow manager and application placement](#)
 - [Request prioritization problems](#)
 - [Configuring dynamic application placement](#)
 - [Overview of application placement](#)
 - [Dynamic application placement](#)
 - [Configuring multi-cell performance management](#)
 - [Manually disabling communication between multiple cells](#)
 - [Configuring elasticity mode](#)
 - [Elasticity mode](#)
 - [Monitoring and tuning the application placement controller](#)
 - [Application placement frequently asked questions](#)
 - [Application placement controller logs](#)
 - [Overview of IIOP and JMS request flow prioritization](#)
 - [Overview of JMS support in WebSphere Virtual Enterprise](#)
 - [Optimizing the database tier for performance monitoring](#)
 - [Configuring the database tier](#)
 - [Monitoring the database tier](#)
 - [Configuring health management](#)
 - [Enabling and disabling health management](#)
 - [Health management](#)
 - [Excessive request timeout health policy target timeout value](#)
 - [Creating health policies](#)
 - [Custom health condition subexpression builder](#)
 - [Creating health policy custom actions](#)
 - [Modifying the JVM heap size](#)
 - [Monitoring and tuning health management](#)
 - [Troubleshooting health management](#)
 - [Tuning the WebSphere Virtual Enterprise cell](#)
 - [Using the centralized installation manager to manage your environment](#)
 - [Centralized installation manager overview](#)
 - [Installing packages](#)

- [Downloading the Update Installer for WebSphere Software](#)
 - [Downloading package descriptors and the associated binary files](#)
 - [Installing interim fixes](#)
 - [Installing refresh packs or fix packs](#)
 - [Manually adding files to the repository](#)
 - [Downloading and installing WebSphere Application Server Community Edition](#)
 - [Installing sudo on target AIX operating systems to apply maintenance as a non-root user](#)
 - [Uninstalling packages](#)
 - [Monitoring requests](#)
 - [Managing installation targets](#)
 - [Using Remote Execution and Access to access remote installation targets](#)
 - [Installing the Secure Shell public key to access installation targets](#)
 - [Running Cygwin as root to use SSH on target Windows operating systems](#)
 - [Updating Secure Shell public key installation records](#)
 - [Routing requests to nodes that are not running WebSphere Virtual Enterprise](#)
 - [Routing requests to external nodes with generic server clusters](#)
 - [Enabling external monitoring with the remote agent](#)
 - [Multiple tiers of processing](#)
 - [WebSphere Virtual Enterprise topology terminology](#)
 - [Configuring work factors in multiple tier configurations](#)
 - [Overriding work factor estimates](#)
 - [Configuring node computing power](#)
 - [Configuring a high availability deployment manager environment](#)
 - [Adding the signer certificate from the secondary deployment manager to the local trust store](#)
 - [The high availability deployment manager](#)
 - [xd_hadmgrAdd command](#)
 - [xd_hadmgrRemove command](#)
 - [Configuring a checkpoint](#)
 - [The repository checkpoint and restore function](#)
 - [Restoring a checkpoint](#)
 - [Using centralized logging to diagnose problems](#)
- [Monitoring operations](#)
 - [Runtime operations overview](#)
 - [Creating and managing reports](#)
 - [Configuring the visualization data service](#)
 - [Task management](#)
 - [Managing runtime tasks](#)
 - [Task management service event logger](#)
 - [Defining email notification](#)
 - [Troubleshooting extended administration](#)
- [Configuring WebSphere Virtual Enterprise to work with other IBM products](#)
 - [Enabling the on demand router to work with IBM Enterprise Workload Manager](#)
 - [WebSphere Virtual Enterprise and IBM Enterprise Workload Manager](#)
- [Reference](#)
 - [Release notes](#)
 - [Port number settings](#)
 - [Scripts](#)
 - [arfmController.py script](#)
 - [checkHmmLocation.jacl](#)
 - [checkPlacementLocation.jacl script](#)
 - [CreateDynamicCluster.py script](#)
 - [createDynamicCluster.jacl script](#)
 - [createodr.jacl script](#)
 - [coregroupsplit.py script](#)
 - [DataReaderWrapper class](#)
 - [deleteDynamicCluster.jacl script](#)

- [deleteodr.jacl script](#)
- [dumpOdrState.jacl script](#)
- [dumpXdState.py script](#)
- [Health controller commands with the AdminConfig object](#)
- [HmmControllerProcs.jacl](#)
- [ifixtool.sh|.bat script](#)
- [importOverlayConfig.py script](#)
- [linkCells|linkCellsZOS script](#)
- [ve_manageODC.py script](#)
- [manageODR.py script](#)
- [manageWVEBB.py script](#)
- [MirrorCell.bat|.sh script](#)
- [odrDebug.py script](#)
- [PlacementControllerProcs.jacl script](#)
- [pluginmerge.bat|.sh script](#)
- [propagatePluginCfg.py](#)
- [serverQuiesce.py script](#)
- [servicepolicy.py script](#)
- [unlinkCells|unlinkCellsZOS script](#)
- [useWVEBB.py](#)
- [wve_encodePassword.sh|.bat script](#)
- [workclassoperations.py script](#)
- [xd_APCconfig.jacl script](#)
- [Administrative roles and privileges](#)
- [Administrator scripting interfaces](#)
 - [Application edition management administrative tasks](#)
 - [Centralized installation manager administrative tasks](#)
 - [Dynamic cluster administrative tasks](#)
 - [Default work class administrative tasks](#)
 - [Health policy administrative tasks](#)
 - [Maintenance mode administrative tasks](#)
 - [Middleware application administrative tasks](#)
 - [Middleware descriptor administrative tasks](#)
 - [Middleware server creation administrative tasks](#)
 - [Middleware server management administrative tasks](#)
 - [Middleware server template administrative tasks](#)
 - [PHP server administrative tasks](#)
 - [Rules for ODR routing policy administrative tasks](#)
 - [Rules for ODR service policy administrative tasks](#)
 - [Rules for generic server cluster routing policy administrative tasks](#)
 - [Rules for generic server cluster service policy administrative tasks](#)
 - [Ruleset administrative tasks](#)
 - [Runtime operations user preferences administrative tasks](#)
- [Custom properties](#)
 - [Application edition manager custom properties](#)
 - [Application placement custom properties](#)
 - [Autonomic controllers custom properties](#)
 - [Autonomic request flow manager custom properties](#)
 - [Autonomic request flow manager advanced custom properties](#)
 - [Binary trace facility custom properties](#)
 - [Core group custom properties](#)
 - [Dynamic cluster custom properties](#)
 - [Health controller custom properties](#)
 - [Internal state reporting custom properties](#)
 - [Maintenance mode custom properties](#)
 - [Middleware agent custom properties](#)
 - [Middleware server custom properties and variables](#)
 - [On demand router system and custom properties](#)
 - [Operational alerts custom properties](#)
 - [Runtime operations custom properties](#)
 - [Service policy custom properties](#)

- [Runtime task custom properties](#)
- [VMware custom properties](#)
- [Logs](#)
 - [StrfTime format conversions](#)
 - [DeploymentTargetStatsHistoricCache](#)
 - [NodeStatsHistoricCache](#)
 - [ServerStatsCache](#)
 - [TCModuleStatsCache](#)
 - [TCModuleInstanceStatsCache](#)
 - [TierStatsCache](#)
 - [FineGrainedPowerConsumptionStatsCache](#)
 - [ServerPowerConsumptionStatsCache](#)
- [Messages](#)
- [Site map](#)