WebSphere® WebSphere Process Server for z/OS

Version 6.1.0

Troubleshooting and Support

WebSphere® WebSphere Process Server for z/OS

Version 6.1.0

Troubleshooting and Support

Note

Before using this information, be sure to read the general information in the Notices section at the end of this document.

31 March 2008

This edition applies to version 6, release 1, modification 0 of WebSphere Process Server for z/OS (product number 5655-N53) and to all subsequent releases and modifications until otherwise indicated in new editions.

To send us your comments about this document, send an e-mail message to doc-comments@us.ibm.com. We look forward to hearing from you.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Chapter 1. Overview of troubleshooting

Troubleshooting is a systematic approach to solving a problem. The goal is to determine why something does not work as expected and how to resolve the problem.

The first step in the troubleshooting process is to describe the problem completely. Without a problem description, neither you nor IBM can know where to start to find the cause of the problem. This step includes asking yourself basic questions, such as:

- What are the symptoms of the problem?
- Where does the problem occur?
- When does the problem occur?
- Under which conditions does the problem occur?
- Can the problem be reproduced?

The answers to these questions typically lead to a good description of the problem, and that is the best way to start down the path of problem resolution.

What are the symptoms of the problem?

When starting to describe a problem, the most obvious question is ″What is the problem?″ This might seem like a straightforward question; however, you can break it down into several more-focused questions that create a more descriptive picture of the problem. These questions can include:

- Who, or what, is reporting the problem?
- What are the error codes and messages?
- How does the system fail? For example, is it a loop, hang, crash, performance degradation, or incorrect result?
- What is the business impact of the problem?

Where does the problem occur?

Determining where the problem originates is not always easy, but it is one of the most important steps in resolving a problem. Many layers of technology can exist between the reporting and failing components. Networks, disks, and drivers are only a few components to be considered when you are investigating problems.

The following questions can help you to focus on where the problem occurs in order to isolate the problem layer.

- Is the problem specific to one platform or operating system, or is it common for multiple platforms or operating systems?
- Is the current environment and configuration supported?

Remember that if one layer reports the problem, the problem does not necessarily originate in that layer. Part of identifying where a problem originates is understanding the environment in which it exists. Take some time to completely describe the problem environment, including the operating system and version, all corresponding software and versions, and hardware information. Confirm that you are running within an environment that is a supported configuration; many

problems can be traced back to incompatible levels of software that are not intended to run together or have not been fully tested together.

When does the problem occur?

Develop a detailed timeline of events leading up to a failure, especially for those cases that are one-time occurrences. You can most easily do this by working backward: Start at the time an error was reported (as precisely as possible, even down to the millisecond), and work backward through the available logs and information. Typically, you need to look only as far as the first suspicious event that you find in a diagnostic log; however, this is not always easy to do and takes practice. Knowing when to stop looking is especially difficult when multiple layers of technology are involved, and when each has its own diagnostic information.

To develop a detailed timeline of events, answer the following questions:

- Does the problem happen only at a certain time of day or night?
- How often does the problem happen?
- What sequence of events leads up to the time that the problem is reported?
- Does the problem happen after an environment change, such as upgrading or installing software or hardware?

Responding to these types of questions can provide you with a frame of reference in which to investigate the problem.

Under which conditions does the problem occur?

Knowing what other systems and applications are running at the time that a problem occurs is an important part of troubleshooting. These and other questions about your environment can help you to identify the root cause of the problem:

- Does the problem always occur when the same task is being performed?
- Does a certain sequence of events need to occur for the problem to surface?
- Do any other applications fail at the same time?

Answering these types of questions can help you explain the environment in which the problem occurs and correlate any dependencies. Remember that just because multiple problems might have occurred around the same time, the problems are not necessarily related.

Can the problem be reproduced?

From a troubleshooting standpoint, the ″ideal″ problem is one that can be reproduced. Typically with problems that can be reproduced, you have a larger set of tools or procedures at your disposal to help you investigate. Consequently, problems that you can reproduce are often easier to debug and solve. However, problems that you can reproduce can have a disadvantage: If the problem is of significant business impact, you do not want it to recur! If possible, re-create the problem in a test or development environment, which typically offers you more flexibility and control during your investigation.

Tip: Simplify the scenario to isolate the problem to a suspected component.

The following questions can help you with reproducing the problem:

- Can the problem be re-created on a test machine?
- Are multiple users or applications encountering the same type of problem?

- Can the problem be re-created by running a single command, a set of commands, a particular application, or a stand-alone application?

Chapter 2. Troubleshooting checklist for WebSphere Process Server

Asking questions about hardware and software requirements, product fixes, specific problems, error messages, and diagnostic data can help you troubleshoot WebSphere Process Server.

The following questions can help you to identify the source of a problem that is occurring with WebSphere Process Server:

1. Is the configuration supported?

   Refer to the requirements for WebSphere Process Server to ensure that your system meets all hardware, operating system, and software requirements: WebSphere® Process Server system requirements Web site.
2. Have you applied the latest fixes?
3. What is the problem?
   - Installing and configuring WebSphere Process Server
   - Migrating existing applications and configuration information to WebSphere Process Server
   - Deploying applications on WebSphere Process Server
   - Administering applications and components on WebSphere Process Server
   - Using WebSphere Application Server capabilities in WebSphere Process Server
4. Have any error messages been issued?
5. For additional help in finding error and warning messages, interpreting messages, and configuring log files, see Diagnosing problems with message logs in the Websphere Application Server information center.
6. Difficult problems can require the use of tracing, which exposes the low-level flow of control and interactions between components. For help in understanding and using traces, see Working with trace in the Websphere Application Server information center.
7. If the checklist does not guide you to a resolution, you can collect additional diagnostic data. This data is necessary for IBM Support to effectively troubleshoot and assist you in resolving the problem. For more information, see Chapter 14, “Contacting IBM Software Support,” on page 77.

Chapter 3. Messages overview

When you receive a message from WebSphere Process Server, you can often resolve the problem by reading the entire message text and the recovery actions that are associated with the message.

You can find the full text of runtime messages, their explanations, and the recommended recovery actions by searching for the message identifier in the Messages section of the WebSphere Process Server Reference documentation.

Messages displayed during WebSphere Process Server product installation and profile creation are documented in Message reference for WebSphere Process Server for z/OS installation and configuration.

Runtime message identifiers consist of a four- or five-character message prefix, followed by a four- or five-character message number, followed by a single-letter message type code. For example, zzzzL1042C. The message type code describes the severity of the error message, as follows:

C Indicates a severe message.

E Indicates an urgent message.

I Indicates an informational message.

N Indicates an error message.

W Indicates a warning message.

Chapter 4. WebSphere Process Server log files

There are two distinct groups of log files in the installed product. Logs detailing the product installation, product updates, and profile management are one group. Logs detailing the characteristics and runtime activities of individual profiles comprise the second group.

Various log files are created during the installation and uninstallation of WebSphere Process Server and during profile creation, augmentation, and deletion. Examine these logs when problems occur during the product installation and configuration process. The log files and their locations within the product installation are detailed in Log files (installation and profile creation).

There are also a number of log files that are created for each profile. Some of these logs describe the parameters used for the creation of the profile. These types of log files generally remain unchanged once the profile is fully configured. Other profile-specific logs are continually updated to capture error, warning, and information messages emitted during runtime. Some of these log files are also used to capture a Common Base Event (that may include business object data) that is selected for monitoring. This set of logs is described in "Profile-specific log files" on page 49.

Chapter 5. Troubleshooting the installation and configuration

Use the information in the troubleshooting section to help you assess and correct problems that may occur if the product installation has not been successful.

Before you begin

The installer program logs installation records in multiple ways:

- Standard output messages

  Standard output messages display directly on the screen of server when you run the installation script zSMPInstall.sh. Standard output messages are also displayed when you run the augment script zWPSConfig.sh. You can choose to redirect these messages to a file by using redirect symbol > and a file name at the end of the command line. For example, adding `>run.log` to the end of the installation command will redirect the standard output messages to a file called **run.log** in the present working directory.
- Log file messages

  Log file messages for the installation process are written to the **zSMPInstall.log** file in the runtime directory. The default location for this file is /WebSphere/V6R1/AppServer/logs/wbi/zSMPInstall.log.

  Log messages for the configuration process are written to the **zWPSConfig.log** file in the runtime directory. The default location of this file is/WebSphere/V6R1/AppServer/logs/wbi/zWPSConfig.log.
- Trace file messages

  Trace messages for the installation process are written to the **zSMPInstall.trace** file in the runtime directory when you specify the -trace parameter on the command line when running the installation script zSMPInstall.sh. If you do not specify this parameter, the trace messages do not get written to the file. The default location of this file is /WebSphere/V6R1/AppServer/logs/wbi/zSMPInstall.trace.

  Trace messages for the configuration process are written to the **zWPSConfig.trace** file in the runtime directory. The default location of this file is/WebSphere/V6R1/AppServer/logs/wbi/zWPSConfig.trace.

Make sure that you have installed and configured WebSphere Application Server for z/OS® successfully. Refer to the installation troubleshooting information in the WebSphere Application Server for z/OS information center if you are having trouble installing and configuring WebSphere Application Server for z/OS.

About this task

For current information available from IBM® Support on known problems and their resolution, see the WebSphere Process Server Support page.

Procedure

1. Review the messages from Standard Out. There should be no error messages displayed. The standard output messages display on either the screen from which you ran the installation command or in a file that you specified by using the redirect (″>″) symbol on the command line.

The following is an example of a successful execution of the installation script with the **-install** option:

```
parsing command arguments...
parsing arguments complete
setting up configuration...
runtimeRootDirName is: /WebSphere/V6R0M0/AppServer
WAS_HOME is: /WebSphere/V6R0M0/AppServer
WBI_HOME is: /WebSphere/V6R0M0/AppServer
set up configuration complete
creating the symbolic links...
invokeSymLink
creation of symbolic links complete
doing post install file updates...
post install updates complete
running Configuration Manager update...
Configuration Manager update complete
augmenting profile(s)...
augmenting profile(s) complete
```

Error messages indicate an unsuccessful installation. Some errors as displayed in Standard Out will be self explanatory and can be easily corrected. If the installation completed to the point where the log and trace files were created, continue with the following steps.

2. Review the zSMPInstall.log (ASCII) file in the runtime directory. For a stand-alone configuration, the standard location for this file is /WebSphere/V6R1/AppServer/logs/wbi/zSMPInstall.log. For a deployment manager configuration, the standard location for this file is /WebSphere/V6R1/DeploymentManager/logs/wbi/zSMPInstall.log

   If there are error messages, determine which of the following tasks were in progress when the error occurred.

   - Create symbolic links
   - Create post installation file
   - Update codebase permissions
   - Update Configuration Manager

   Knowing the task that was in progress at the time of an error will help you assess the information in the trace file.

3. Review the zSMPInstall.trace or the zWPSConfig.trace / zWESBConfig.trace (ASCII) file in the runtime directory. For a stand-alone configuration, the standard location for these files are /WebSphere/V6R1/AppServer/logs/wbi/zSMPInstall.trace or /WebSphere/V6R1/AppServer/logs/wbi/zWPSConfig.trace For a deployment manager configuration, the standard location of this file is /WebSphere/V6R1/DeploymentManger/logs/wbi/zSMPInstall.trace or /WebSphere/V6R1/DeploymentManger/logs/wbi/zWPSConfig.trace

   On a successful installation, only informational messages (messages with a suffix of **I**, for example CWPIZ0044**I**) should be listed in the trace file.

   If warning messages (messages with a suffix of **W**) or error messages (messages with a suffix of **E**) are listed in the trace, further review is required.

   If the warning or error occurred during the create symbolic links, create post installation file, or update codebase permissions tasks, the trace message should contain information that will help you diagnose and correct the problem.

   If the warning or error occurred in the update Configuration Manager task, proceed to the next step.

   If the warning or error occurred in the augment profile(s) task, proceed to step 5.

4. Review the actions of the Update Configuration Manager task. These actions are recorded by writing to a log file (ASCII). The log file name is `installconfig.log`.

   Standard location for this file is in directory /WebSphere/V6R1/AppServer/logs/wbi/install.

   Search the Configuration Manager log for `>SEVERE<` or `>WARNING<` level messages to determine overall error in processing.

   Each Ant script run from the installation directory writes to its own log (ASCII).

   To determine what Ant file was running at the time of the error, you can look for 'Buildfile' previous.

   The default name for the installation directory that contains the ant scripts is: /WebSphere/V6R1/AppServer/properties/version/install.wbi/config/install.

   The resulting ant logs are written to the product log directory. The default name for this directory is /WebSphere/V6R1/AppServer/logs/wbi. Review these logs to determine errors in processing.

   If there were no problems a **BUILD SUCCESSFUL** message displays at the end of the file.
5. Review the actions of the WebSphere Application Server profile augment task. The augment profile(s) task records its actions by writing to a log file (ASCII). The log file name has the name **default_augment.log**. Standard location for this file is in the directory /WebSphere/V6R1/AppServer/logs/manageprofiles/default_augment.log.

   Search the Profile Augmentation log for `>SEVERE<` or `>WARNING<` level messages to determine overall error in processing.

   Individual Ant action logs are located in /WebSphere/V6R1/AppServer/profiles/default/logs/manageprofiles/default.

   To determine what Ant file was running at the time of the error, you can look in the log for last instance of 'Buildfile' prior to the message in which you are interested.

What to do next

After troubleshooting the problems that caused the installation errors, and after you have run the installation script successfully, you should perform the following steps:

1. Start the WebSphere Application Server.
2. Launch the Administrative Console and verify that the product components have been installed.

   For example, by installing WebSphere Process Server successfully, you should see evidence of Process Choreographer under the Enterprise Applications with names that start with `BPEContainer`, `BPCExplorer`, and `TaskContainer`.

WebSphere Process Server errors

If you experience a problem with one of the configuration tasks then there will be three main sources of information about the problem:

1. The error messages issued by the task
2. Error messages in the WebSphere deployment manager or application server job logs. If you are federating a node you might also find messages in the node agent job logs
3. Log files in the UNIX® file system

Wherever possible, the cause and solution to each problem is also documented with the symptoms. The problems described here were experienced when starting the server after completing the installation procedure for WebSphere Process Server. In the examples of error messages, the messages have been made easier to read by changing the places where line breaks occur. Therefore, if you see these errors in your system the messages will have a slightly different layout.

Failure in loading T2 native library db2jcct2zos

The following error happens when trying to use the DB2® Universal Type connector and WebSphere Application Server cannot load some external DB2 modules from SDSNLOAD or SDSNLOD2.

Error message: BBOO0220E:

```
error message: BBOO0220E:
[SCA.APPLICATION.mdcell.Bus:mdnodea.mdsr01a-SCA.APPLICATION.mdcell.Bus]
CWSIS0002E: The messaging engine encountered an exception while
starting.
Exception: com.ibm.ws.sib.msgstore.PersistenceException:
CWSIS1501E: The data source has produced an unexpected exception:
java.sql.SQLException: Failure in loading T2 native library
db2jcct2zos, reason: java.lang.UnsatisfiedLinkError:
/pp/db2v8/UK14852/jcc/lib/libdb2jcct2zos.so:
EDC5157I An internal error has occurred. (errno2=0x0BDF03B2)DSRA0010E:
SQL State = null, Error Code = -99,999DSRA0010E: SQL State = null,
Error Code = -99,999
com.ibm.ws.sib.utils.ras.SibMessage
com.ibm.ws.sib.utils.ras.SibMessage
```

There are a number of possible causes of a failure to load libdb2jcct2zos.so, but usually a failure like this is a symptom of a larger problem such as the DB2 Universal Driver not being been fully configured in the DB2 system you are accessing.

Check that all the steps for installing the DB2 Universal Driver have been performed for your DB2 system.

The installation instructions for the DB2 Universal Driver can be found in the DB2 Information Center at http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.db2.doc.java/install.htm#install?

DataSource has a null RelationalResourceAdapter property

The error shown in the example below is caused by a redundant datasource that is left behind after running the augment script zWPSConfig.sh. You can safely delete this datasource using the WebSphere Administration Console. Be careful not to delete the JDBC provider that has a very similar name.

```
error message: BBOO0222I: DSRA8208I: JDBC driver type : 2
com.ibm.ws.exception.RuntimeWarning:
com.ibm.ws.runtime.component.binder.ResourceBindingException:
invalid configuration passed to resource binding logic. REASON: Invalid
Configuration!
The DataSource: DB2 Universal JDBC Driver DataSource has a null
RelationalResourceAdapter property.
```

Perform the following steps to remove the redundant datasource:

1. Log in to the WebSphere Administration Console and navigate to Resources → JDBC Providers.
2. Set the scope to Server and click **Apply**.

3. Click the JDBC provider called **DB2 Universal JDBC Driver Provider**.
4. Click the link to **Datasources** on the right.
5. You should see a list of three datasources. Check the box next to **DB2 Universal JDBC Driver Datasource** and click the **Delete** button.
6. Save your configuration changes and restart the WebSphere server.

SQLCODE = -471

This problem occurs when the Universal Driver has not been properly configured in a DB2 system.

The WLM_ENVIRONMENT for SYSIBM.SYSTABLES in the table SYSIBM.SYSROUTINES has a WLM name that does not match that being used in the stored procedure address space JCL. The installation instructions for the DB2 Universal Driver can be found in the DB2 Information Center at http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.db2.doc.java/install.htm#install?

```
ExtendedMessage: BBOO0220E:
[CommonEventInfrastructure_Bus:mdnodea.mdsr01a-CommonEventInfrastructur e_Bus]
CWSIS0002E: The messaging engine encountered an exception while
starting.
Exception: com.ibm.ws.sib.msgstore.PersistenceException:
CWSIS1501E: The data source has produced an unexpected exception:
com.ibm.db2.jcc.t2zos.y:[IBM/DB2][T2zos/2.9.32]
v.readExecuteCallInternal: nativeExecuteCall:5587:
DB2 engine SQL error, SQLCODE = -471, SQLSTATE = 55023,
error tokens = SYSIBM.SQLTABLES;00E7900C
```

SQL code -204 and -516

This error can be caused if the currentSchema property does not match the schema name of the tables and indexes that you created. The error messages show the J2C authentication alias that is being used.

```
error message: BBOO0220E: SCHD0125E: Unexpected exception
while processing the acquireLease operation:
com.ibm.ws.leasemanager.LeaseException: SCHD0300E:
Error during Database operation,
localized message is

_
:nativePrepareInto:1377:
DB2 engine SQL error, SQLCODE = -204, SQLSTATE = 42704,
error tokens = MDDBU.WSCH_LMGR,
Vendor Error Code is -204, ANSI-92 SQLState is 42704, cause:
[IBM/DB2][T2zos/2.9.32]T2zosPreparedStatement.readDescribeInput_
:nativeDescribeInput:2006:
DB2 engine SQL error, SQLCODE = -516, SQLSTATE = 26501,
error tokens =


..
..
com.ibm.db2.jcc.t2zos.y:
[IBM/DB2][T2zos/2.9.32]T2zosPreparedStatement.readDescribeInput:2006:
DB2 engine SQL error, SQLCODE = -516, SQLSTATE = 26501, ...
```

Repeated SIB messages about acquiring and losing locks

This error can occur after correcting the DB2 Universal Driver configuration and restarting the server. The error messages are repeated continuously in the adjunct region.

```
ExtendedMessage: BBOO0222I:
[CommonEventInfrastructure_Bus:mdnodea.mdsr01a-CommonEventInfrastructur e_Bus]
CWSIS1538I: The messaging engine, ME_UUID=68E9550CE7780888,
INC_UUID=5f244052b02f04b4,
is attempting to obtain an exclusive lock on the data store.
..
..
ExtendedMessage: BBOO0222I:
[CommonEventInfrastructure_Bus:mdnodea.mdsr01a-CommonEventInfrastructur
e_Bus]
CWSIS1546I: The messaging engine, ME_UUID=68E9550CE7780888,
INC_UUID=5f244052b02f04b4,
has lost an existing lock or failed to gain an initial lock on the database
```

These error messages indicate that there is a problem accessing the data store. Check that the fixWPSvars.jacl had created (jdbc/MEdatasource) in the datasource. Check that the datasource has an J2C authentication alias associated with it. If there is no J2C authentication alias associated with the datasource the database access defaults to the user ID of the servant region and tries to find tables called MKASRU which do not exist.

This error can occur because the -sibauth option has not been coded when running fixWPSvars.jacl. You can fix this in a number of ways:

- Associate the JDBC datasources used by the SIBs with the J2C authentication alias called WPSDBAlias.
- Create a new J2C Authentication Alias and associate that with the JDBC. Re-run fixWPSvars.jacl using the -sibauth option to specify WPSDBAuth as the J2C authentication alias, or use the WebSphere Administration console to make the change, specifying WPSDBAlias as the alias name.

The following steps explain how to create a new J2C authentication alias to be used by the SIBs to access DB2:

1. Pen theWebSphere Administration Console and navigate to **Security** → **Global security**.
2. Click the link to **J2C Authentication data** under **Additional Properties**.
3. Click the **New** button.
4. Enter a name for the alias, and enter the user ID and the password for the alias.
5. Click **OK**.
6. Click **Save**.

The following steps explain how to associate the Service Integration Buses with the authentication alias that you have created:

1. Open the WebSphere Administration Console and navigate to **Service integration** → **Buses**.
2. Click the first bus in the list.
3. On the next panel, click **Messaging engines**.
4. Click the hyperlink to the messaging engine.
5. Click the link to **Data store** under **Additional Properties**.
6. Expand the drop-down list box in the **Authentication alias** field and select the alias you created earlier.
7. Click **OK** and then save the change to the configuration.
8. Click the link at the top of the page to navigate back to the list of buses.

9. Select the next bus in the list and repeat the same procedure. Repeat for the remaining buses.
10. When all the buses have been updated to refer to a valid J2C authentication alias and stop and restart the server.

bpeconfig.jacl: An error occurred installing TaskContainer

A bpeconfig.jacl error normally occurs if you enter invalid input. The example below is displayed if you make a a mistake entering group names and if you use the delete key instead of the backspace key to make a correction. In the example below, the input appears to be MKHTSMG, but the value that was actually entered contained invalid characters ″MKSMG[D[D[D[D[CHTSMG″.

```
..
..
[] Install the task container [Yes/no]? Yes
[adminHTMUsers] User(s) to add to role TaskSystemAdministrator
(separator is pipe,
'|') []: MKADMIN
[adminHTMGroups] Group(s) to add to role TaskSystemAdministrator
(separator is pipe,
'|') []: MKSMADMG|MKCFG
[monitorHTMUsers] User(s) to add to role TaskSystemMonitor
(separator is pipe, '|')
[]:
[monitorHTMGroups] Group(s) to add to role TaskSystemMonitor
(separator is pipe, '|')
[]: MKSMG[D[D[D[D[CHTSMG
[jmsHTMRunAsUser] Run-as UserId for role EscalationUser
 [MKADMIN]: MKHTSM
[jmsHTMRunAsPwd] MKHTSM's password []: ********
[auto:mqType] Use WebSphere default messaging or
WebSphere MQ? WPM
task.ear install options: -appname "TaskContainer_mkcl01"
-usedefaultbindings
-defaultbinding.ejbjndi.prefix ejb/htm -cluster
"mkcl01"
-BindJndiForEJBMessageBinding {{"TaskContainer" "HTMScheduler"
"taskejb.jar,META-INF/ejb-jar.xml" ""
"eis/HTMInternalActivationSpec"
"jms/HTMIntQueue" ""}} -MapResRefToEJB {
{"TaskContainer"
"GenericHumanTaskManagerEJB" "taskejb.jar,META-INF/
ejb-jar.xml" "jdbc/BPEDB"
"javax.sql.DataSource" "jdbc/BPEDB_mkcl01"} {"TaskContainer"
"TaskContainerStartUpBean" "taskejb.jar,META-INF/ejb-jar.xml"
 "jdbc/BPEDB"
"javax.sql.DataSource" "jdbc/BPEDB_mkcl01"}} -MapResEnvRefToRes
{ {"TaskContainer"
"TaskContainerStartUpBean" "taskejb.jar,META-INF/ejb-jar.xml"
 "jms/HTMHoldQueue"
"javax.jms.Queue" "jms/HTMHldQueue"} {"TaskContainer"
 "TaskContainerStartUpBean"
"taskejb.jar,META-INF/ejb-jar.xml" "scheduler/BPCScheduler"
"com.ibm.websphere.scheduler.Scheduler" "BPEScheduler"}
} -MapRolesToUsers
{{"TaskSystemAdministrator" "AppDeploymentOption.No"
 "AppDeploymentOption.No"
"MKADMIN" "MKSMADMG|MKCFG"} {"TaskSystemMonitor"
 "AppDeploymentOption.No"
"AppDeploymentOption.No" "" "MKSMG[D[D[D[D[CHTSMG"}
 {"EscalationUser"
"AppDeploymentOption.No" "AppDeploymentOption.Yes"
 "" ""}} -MapRunAsRolesToUsers
{{"EscalationUser" "********" "MKHTSM"}}
An error occured installing TaskContainer_mkcl01:
```

```
..
..
com.ibm.ws.scripting.ScriptingException: WASX7132E: Application install for
/wasmkconfig/mkcell/mkdmnode/DeploymentManager/installableApps/task.ear failed: see
previous messages for details. Discarding changes.
```

If you look in the log of the servant job in the Deployment Manager you could also see a related error like that shown below:

```
error message: FFDC0010I: FFDC closed incident stream file
/wasmkconfig/mkcell/mkdmnode/DeploymentManager/profiles/default/logs/ff
dc/mkcell_mkdmnode_dmgr_STC12532_MKDMGRS_06.11.13_04.05.37_1.txt
com.ibm.etools.j2ee.commonarchivecore.exception.ResourceLoadException: IWAE0007E
Could not load resource "META-INF/ibm-application-bnd.xmi" in archive
"/wasmkconfig/mkcell/mkdmnode/DeploymentManager/profiles/default/temp/app35301.ear"
!Stack_trace_of_nested_exce!
com.ibm.etools.j2ee.exception.WrappedRuntimeException:
Exception occurred loading META-INF/ibm-application-bnd.xmi
!Stack_trace_of_nested_exce!
Wrapped exception
org.xml.sax.SAXParseException: An invalid XML character (Unicode: 0x1b) was found in
the value of attribute "name" and element is "groups".
```

Verification errors

When you verify the installation you may encountered some problems, which are described in this section.

Resources not seen in the administrative console

When you are checking that applications you have installed exist in the system, you may not see them listed under the installed applications section. If you do not see the applications listed log out of the administrative console and log back in.

If you do not see Service Integration Buses that you have configured, log out of the administrative console and log back in.

Troubleshooting the CBE Browser verification

There are a number of reasons why you might see an error when testing the CBE Browser in a Network Deployment configuration:

CWLCB0020E: Common Event Infrastructure is unavailable with NameNotFoundException

- A fully qualified JNDI name was not used on the CBE Browser.
- The schema name for the CEI tables is not equal to the user ID in the J2C Authentication Alias that is being used by the event and eventcat datasources.

The following is an example of messages that can be displayed when verifying the CBE browser in the Network Deployment configuration for the first time.

The following is an example of messages that can be displayed in the servant log of the WebSphere Application Server:

```
javax.naming.NameNotFoundException:
Context: mkcell/nodes/mkdmnode/servers/dmgr,
name: ejb/com/ibm/events/access/EventAccess:
First component in name com/ibm/events/access/EventAccess not found.
Root exception is org.omg.CosNaming.NamingContextPackage.NotFound:
IDL:omg.org/CosNaming/NamingContext/NotFound:1.0
```

A fully qualified JNDI name is needed in the Event Data Store when CEI is configured in a Network Deployment configuration. To resolve this problem, specify a fully qualified JNDI name for the EventAccess EJB.

CWLCB0020E: Common Event Infrastructure is unavailable with CEIDS0035E

The following error messages can also be displayed when verifying the CBE browser in the Network Deployment configuration for the first time:

```
error message: CEIDS0035E The implementation class that supports the
configured relational database system cannot be loaded.
Implementation class name:
com.ibm.events.datastore.impl.Db2UniveralDriverImpl
Relational database name: DB2
Database version: DSN08015
com.ibm.events.datastore.impl.DatabaseSpecificsFactory
handleCreateException(String, String, String, Exception)
```

These error messages indicate a problem loading a class, but the cause is usually a problem accessing the database. Sometimes there are also some error messages with DB2 return codes like -204 that can help you diagnose the problem, but sometimes you only see the CEIDS0035E message.

Solution 1

If you find a -551 error in the FFDC file, correct the authorization failure reported there by issuing appropriate GRANT statements in DB2.

For example, you could issue GRANT ALL ON TABLE MKCELL. TO MKDBU for all of the CEI tables, views, and indexes created in the CEI databases, then restart the WebSphere Application Server. You may still however receive error message CWLCB0020E on the CBE Browser and CEIDCS0035E in the servant log.

The following is an example of a different DB2 error reported in the FFDC log, and the same error reported in the servant message log:

```
Exception = com.ibm.db2.jcc.t2zos.y
Source = com.ibm.ws.rsadapter.jdbc.WSJdbcConnection.prepareStatement
probeid = 1584
Stack Dump = com.ibm.db2.jcc.t2zos.y:
[IBM/DB2][T2zos/2.9.32]T2zosPreparedStatement.readPrepareDescribeOutput
_:nativePrepareInto:1377:DB2 engine SQL error, SQLCODE = -204, SQLSTATE
= 42704, error tokens = MKDBU.CEI_T_CBE_MAP
```

A -204 code is resource allocation failure, that is, MKDBU.CEI_T_CBE_MAP not found. We had created all the CEI tables with a schema name of MKCELL, but the failure shows that the CBE Browser is attempting to access MKDBU.CEI_T_CBE_MAP. The user ID MKDBU is the one in the J2C Authentication Alias being used by the event and eventcat datasources.

The problem is that the CEI component interrogates the alias associated with the datasource and uses that to issue fully qualified SQL. This means that any value you set in the currentSchema custom property on the datasources is ignored. Unfortunately, at this time CEI must be configured so that the schema of all CEI objects in DB2 is equal to the user ID in the J2C Authentication Alias used by the event and eventcat datasources.

There are two ways to resolve this problem:

- Make the schema in the database match the user ID in the J2C Authentication Alias user ID being used on the event and eventcat datasources (solution 3a).
- Use a new J2C Authentication Alias for the event and eventcat datasources that has a user ID that matches the schema of the existing CEI tables (solution 3b). This is the approach we recommend if longer term you do not want to have the schema of CEI tables equal to the J2C Authentication Alias user ID.

Solution 2a

Drop the CEI databases, then recreate them. Then re-run the CEI DDL (which includes inserting the metadata and seeding the catalog), but specify a schema name equal to the J2C Authentication Alias user ID on all CREATE statements.

Solution 2b

Perform the following steps if you do not want to drop and recreate the CEI tables:

1. Create a RACF® user ID equal to the schema name you are currently using.
2. Using the WebSphere Administration Console, define a new J2C Authentication Alias and set the RACF user ID with a password in that alias. Set the schema of the CEI user ID and password in the J2C alias that you create.
3. Navigate to **Resources** → **JDDC Providers** and set the scope depending on whether CEI was deployed in a cluster or a server:
   - If the CEI EventServer application is deployed in a cluster, set the scope to the cluster.
   - If the CEI Event Server application is deployed in a server, set the scope to the server.

   Click **Event_DB2ZOS_JDBC_Provider**.
4. Click **Data sources** under **Additional properties**.
5. Click **Event**.
6. Scroll down and select the new J2C Authentication Alias you created from the drop-down list box in field Component-managed authentication alias.
7. Click **OK**.
8. Navigate to the eventcat datasource and make the same change.
9. Save your configuration changes then restart the server or cluster.

The advantage of using solution 3b rather than 3a is that the reason you experienced this problem in the first place is that you do not want to have tables in DB2 with a schema that matches the J2C Authentication Alias user ID. The CEI tables already have the schema you want to use, so it does not make sense to change the schema. When a fix is available to CEI that allows you to use the currentSchema property on the datasource, you can easily switch the datasources so that they return to using the original J2C Authentication Alias.

Of course, even with solution 3b you will be temporarily using a J2C Authentication Alias equal to the CEI schema. When there is a fix for CEI and you switch back to the original J2C Authentication Alias (the one that is not the same as the schema), you can delete the alias and the RACF user ID for the J2C Authentication Alias you created to temporarily match the CEI schema.

CWLCB0020E: Common Event Infrastructure is unavailable with CORBA NO MEMORY

In the servant long you may also see the following error:

```
java.rmi.RemoteException: CORBA NO_MEMORY exception
```

You are unlikely to experience this problem if you have just installed WebSphere Process Server, but it is useful to know that you can experience memory problems if you get events with a high number specified in the Maximum number of events field.

You can resolve this problem in either of two ways:

- Reduce the maximum number of events to retrieve from 500 to 100, then retrieve the next blocks of events using the time and date query.
- Increase the minimum and maximum heap sizes for the JVM of the servant region in the server running the EventServer application.

Message reference for WebSphere Process Server for z/OS installation and configuration

The message reference for WebSphere Process Server for z/OS lists the message codes that display while running the install script or when running the configuration script.

About the installation error messages

Use the data in the Explanation and user response fields to troubleshoot the WebSphere Process Server for z/OS message codes.

The message code displays as CWPIZyyyyz, where:

- CWPIZ = The WebSphere Process Server for z/OS message prefix
- yyyy = The numeric identifier assigned to the number
- z = Descriptor (E, I or W) for the type of message, where:
  - E = Error message
  - I = Informational message
  - W = Warning message

For a listing of the WebSphere Process Server for z/OS installation error messages, see CWPIZ in the Messages portion of the Reference documentation.

The WebSphere Process Server for z/OS installation error messages are written to the zSMPInstall.log file in the run-time directory. The standard default location for the log file is /WebSphere/V6R1/AppServer/logs/wbi/zSMPInstall.log.

The WebSphere Process Server for z/OS configuration error messages are written to the zWPSConfig.log file and the zWESBConfig.log file in the run-time directory. The standard default location for these log files are /WebSphere/V6R1/AppServer/logs/wbi/zWESBConfig.log and /WebSphere/V6R1/AppServer/logs/wbi/zWPSConfig.log respectively.

Log files

Various log files are created during the product installation and configuration process.

Purpose

Consult the applicable logs if problems occur during the product installation and configuration process.

Standard out messages redirected to log file

Standard out messages report high-level actions such as the starting and completing of the action that verifies the command line arguments.

By default, these messages display directly on the screen from which you run the product installation script. However, you can *redirect* these messages to a file by using the redirect symbol and a file name at the end of the command line. For example, specifying `>run.log` at the end of the installation command redirects the standard out messages to a file named `run.log` in the present working directory.

Standard out messages also report severe errors that occur prior to the Log and Trace File being opened. For instance, the following message block displays if a required keyword (-runtime) was not included in the installation command.

```
parsing command arguments...
CWPIZ0101E -runtime keyword and value not specified on command line.
com.ibm.ws390.installer.InstallFailureException: -runtime keyword and value not specified
CWPIZ0017E  install task failed.
```

Log file

These messages include the messages written to Standard Out, but provide additional information and settings that were used by the installer program.

For instance, the following log portion shows the response properties and their values being used. It also shows the source and target directories being used during the creation of the symbolic links.

```
response property: profilePath=/WebSphere/V6R1M0/AppServer/profiles/default
response property: nodeName=SY1
response property: scaSecurityPassword=ibmuser
response property: dbType=Derby
response property: ceiSampleJmsUser=ibmuser
response property: scaSecurityUserId=ibmuser
response property: configureScaSecurity=true
response property: mqUser=ibmuser
response property: serverName=server1
response property: adminBFMGroups=ibmuser
response property: profileName=default
response property: dbCreateNew=true
response property: ceiSampleJmsPwd=ibmuser
response property: cellName=SY1
response property: dbLocation=/WebSphere/V6R1M0/AppServer/derby/databases/WBIDB
response property: mqPwd=ibmuser
response property: was.install.root=/WebSphere/V6R1M0/AppServer
response property: augment=
response property: ceiDbProduct=CLOUDSCAPE_V51_1
response property: wbi.install.root=/WebSphere/V6R1M0/AppServer
response property: ceiSampleServerName=server1
response property: templatePath=/WebSphere/V6R1M0/AppServer/profileTemplates/default.*
response property: dbName=WBIDB
set up configuration complete
creating the symbolic links...
Source=/usr/lpp/zWPS/V6R1M0

Target=/WebSphere/V6R1M0/AppServer
```

```
creation of symbolic links complete
doing post install file updates...
post install updates complete
running Configuration Manager update...
Configuration Manager update complete
```

Trace file

These messages are written to the **zSMPInstall.trace** file in the run-time directory.

The example below shows some preliminary informational messages and then a **CWPIZ0322E** error indicating that the required profileName property was not found in the response file that the user specified on the installation script command line (nor was provided as a -Z override).

The subsequent **CWPIZ0017E** error message is a general message indicating the final outcome of the zSMPInstall.sh run.

```
[8/16/05 17:00:45:380 EDT] 0000000a ManagerAdmin  I   BBOO0222I:
TRAS0017I: The startup trace state is *=info.

[8/16/05 17:00:48:230 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0044I Begin install task.

[8/16/05 17:00:48:273 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0117I WPS installer log data will be written to
/WebSphere/V6R1M0/AppServer/logs/wbi/zSMPInstall.log.

[8/16/05 17:00:48:282 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0024I WPS installer trace data will be written to
/WebSphere/V6R1M0/AppServer/logs/wbi/zSMPInstall.trace.

[8/16/05 17:00:48:292 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0014I Trace specification is "*=all=disabled".

[8/16/05 17:00:48:298 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0045I WPS SMP/E root directory is /zrockuser/wbi/Install.

[8/16/05 17:00:48:302 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0052I WAS SMP/E root directory is /web/usr/lpp/zWebSphere/V6R0.

[8/16/05 17:00:48:307 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0046I Destination application server root directory is
/WebSphere/V6R1M0/AppServer.

[8/16/05 17:00:48:314 EDT] 0000000a WPSInstaller  E   BBOO0220E:
CWPIZ0322E profileName property not specified in Response File.

[8/16/05 17:00:48:318 EDT] 0000000a WPSInstaller  E   BBOO0220E:
CWPIZ0017E install task failed.

A trace file from a zSMPInstall.sh executed with the trace specification argument
set to "*=all=enabled" provides additional debugging information. It may contain
information that is meaningful only to a developer. The following is a partial
trace using "*=all=enabled":

************ Start Display Current Environment ************
Host Operating System is z/OS, version 01.04.00
Java version = J2RE 1.4.2 IBM z/OS Persistent Reusable VM build cm142-20050623
(JIT enabled: jitc), Java
Compiler = jitc, Java VM name = Classic VM
was.install.root = /WebSphere/V6R1M0/AppServer
user.install.root = /WebSphere/V6R1M0/AppServer/profiles/default
Java Home = /web/usr/lpp/zWebSphere/V6R0/java/J1.4
ws.ext.dirs = /WebSphere/V6R1M0/AppServer/java/lib:/WebSphere/V6R1M0/AppServer/java/lib/
ext:/WebSphere/V6R1M0/AppServer/classes:/WebSphere/V6R1M0/AppServer/lib:/WebSphere/V6R1M0/AppServer/
installedChannels:/WebSphere/V6R1M0/AppServer/lib/ext:/WebSphere/V6R1M0/AppServer/deploytool/itp
/plugins/com.ibm.etools.ejbdeploy/runtime:/WebSphere/V6R1M0/AppServer/MQSeries/pubsubroot/lib
Classpath = /zrockuser/bbzconfig.jar:/WebSphere/V6R1M0/AppServer/lib/admin.jar:/WebSphere/V6R1M0
/AppServer/lib/ant.jar:/WebSphere/V6R1M0/AppServer/lib/bootstrapws390.jar:/WebSphere/V6R1M0
/AppServer/lib/bootstrap.jar:/WebSphere/V6R1M0/AppServer/lib/configmanager.jar:/WebSphere
/V6R1M0/AppServer/lib/emf.jar:/WebSphere/V6R1M0/AppServer/lib/ras.jar:/WebSphere/V6R1M0
/AppServer/lib/runtimefw.jar:/WebSphere/V6R1M0/AppServer/lib/utils.jar:/WebSphere/V6R1M0
/AppServer/lib/wasjmx.jar:/WebSphere/V6R1M0/AppServer/lib/wasproduct.jar:/WebSphere/V6R1M0
/AppServer/lib/wccm_base.jar:/WebSphere/V6R1M0/AppServer/lib/wjmxapp.jar:/WebSphere/V6R1M0
/AppServer/lib/wsanttasks.jar:/WebSphere/V6R1M0/AppServer/lib/wsexception.jar:/WebSphere
/V6R1M0/AppServer/lib/wsprofile.jar:/WebSphere/V6R1M0/AppServer/profiles/default/properties:
/WebSphere/V6R1M0/AppServer/properties:/WebSphere/V6R1M0/AppServer/lib/bootstrap.jar:/WebSphere
/V6R1M0/AppServer/lib/j2ee.jar:/WebSphere/V6R1M0/AppServer/lib/lmproxy.jar:/WebSphere/V6R1M0
/AppServer/lib/urlprotocols.jar:/WebSphere/V6R1M0/AppServer/lib/bootstrapws390.jar
Java Library path = /web/usr/lpp/zWebSphere/V6R0/java/J1.4/bin/classic/libjvm.so:/web/usr
/lpp/zWebSphere/V6R0/java/J1.4/bin/classic:/web/usr/lpp/zWebSphere/V6R0/java/J1.4/bin/:
/WebSphere/V6R1M0/AppServer/lib:/WebSphere/V6R1M0/AppServer/lib:/WebSphere/V6R1M0/AppServer
```

```
/MQSeries/pubsubroot/lib:/mqm/java/bin:/mqm/java/lib:/db2810/lib:/db2beta/db2710/lib:
/web/usr/lpp/WebSphere/lib:/lib:/usr/lib:/java/J1.3/bin:/java/J1.4/bin:/java/J5.0/bin:
/staf/lib:/WebSphere/V6R1M0/AppServer/lib:/usr/lib
Current trace specification = *=all
************* End Display Current Environment *************
[10/3/05 16:35:05:709 EDT] 0000000a ManagerAdmin  I   BBOO0222I: TRAS0017I:
The startup trace state is *=all.
[10/3/05 16:35:08:638 EDT] 0000000a WPSInstaller  >  setup Entry
                                /web/usr/wbi/zWebSphere/V6R0
                                APPSERVER
                                zSMPInstall.sh
                                -smproot
                                /web/usr/wbi/zWPS/V6R0
                                -runtime
                                /WebSphere/V6R1M0/AppServer
                                -response
                                /web/usr/wbi/zWPS/V6R0/zos.config/standAloneProfile.rsp
                                -prereqonly
                                -trace
                                *=all=enabled
[10/3/05 16:35:08:640 EDT] 0000000a WPSInstaller  3   logFileDeleted
                                true
[10/3/05 16:35:08:660 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0044I:
Begin install task.
[10/3/05 16:35:08:702 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0117I:
WPS installer log data will be written to /WebSphere/V6R1M0/AppServer/logs/wbi/zSMPInstall.log.
[10/3/05 16:35:08:712 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0024I:
WPS installer trace data will be written to /WebSphere/V6R1M0/AppServer/logs/wbi/zSMPInstall.trace.
[10/3/05 16:35:08:722 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0014I:
Trace specification is "*=all=enabled".
[10/3/05 16:35:08:726 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0052I:
WAS SMP/E root directory is /web/usr/lpp/zWebSphere/V6R0.
[10/3/05 16:35:08:730 EDT] 0000000a WPSInstaller  >  checkPathName Entry
                                /web/usr/wbi/zWPS/V6R0
[10/3/05 16:35:08:731 EDT] 0000000a WPSInstaller  <  checkPathName Exit
[10/3/05 16:35:08:732 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0045I:
WPS SMP/E root directory is /web/usr/wbi/zWPS/V6R0.
[10/3/05 16:35:08:736 EDT] 0000000a Symlink       >  isSymlink Entry
                                /web/usr/wbi/zWPS/V6R0
[10/3/05 16:35:08:737 EDT] 0000000a Symlink       3   absolute path
                                /web/usr/wbi/zWPS/V6R0
[10/3/05 16:35:08:737 EDT] 0000000a Symlink       3   canonical path
                                /web/usr/wbi/zWPS/V6R0
[10/3/05 16:35:08:738 EDT] 0000000a Symlink       <  isSymlink Exit
                                false
[10/3/05 16:35:08:738 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0046I:
Destination application server root directory is /WebSphere/V6R1M0/AppServer.
[10/3/05 16:35:08:744 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0247I:
Response file is /web/usr/wbi/zWPS/V6R0/zos.config/sample.rsp.
[10/3/05 16:35:08:764 EDT] 0000000a WPSInstaller  3   response property
                                profilePath=/WebSphere/V6R1M0/AppServer/profiles/default
[10/3/05 16:35:08:765 EDT] 0000000a WPSInstaller  3   response property
                                nodeName=SY1
```

Chapter 6. Troubleshooting migration

If you encounter problems during migration, the information described here could help.

Troubleshooting version-to-version migration

Review this page for troubleshooting tips if you encounter problems while migrating from an older version of WebSphere Process Server.

- If you encounter a problem when you are migrating from a previous version of WebSphere Process Server to version 6.1, check your log files and other available information.
  1. Look for diagnostic information in the two files that were generated when you ran the migration job. The files were written to the JESOUT data set during the WROUT and WRERR steps; browse them from SDSF.
  2. Look in the following log files, which are in ASCII format (if you are viewing them on z/OS, you must convert them to EBCDIC first):
     - *migration_backup_directory*/base_backup/WASPreUpgrade.*time_stamp*.log
     - *migration_backup_directory*/base_backup/WBIPostUpgrade.*time_stamp*.log
     - *migration_backup_directory*/base_backup/WBIPreUpgrade.*time_stamp*.log
     - /WebSphere/V6R1/DeploymentManager/profiles/default/logs/WASPreUpgradeSummary.log
     - /WebSphere/V6R1/DeploymentManager/profiles/default/logs/WASPostUpgradeSummary.log
     - /WebSphere/V6R1/DeploymentManager/profiles/default/logs/WASPostUpgrade.*time_stamp*.log
     - /WebSphere/V6R1/AppServer/profiles/default/logs/WASPreUpgradeSummary.log
     - /WebSphere/V6R1/AppServer/profiles/default/logs/WASPostUpgradeSummary.log
     - /WebSphere/V6R1/AppServer/profiles/default/logs/WASPostUpgrade.*time_stamp*.log
  3. Look for the following messages in the logs:

     ```
     MIGR0259I: The migration has successfully completed.
     ```

     ```
     MIGR0271W: The migration completed with warnings.
     ```
  4. Open the Log and Trace Analyzer built into the Application Server Toolkit (AST) on the service log of the server that is hosting the resource that you are trying to access, and use it to browse error and warning messages.

     See Debugging components in the Application Server Toolkit.
  5. With WebSphere Process Server, run the dumpNameSpace command and pipe, redirect, or "more" the output so that it can be easily viewed.

     This command results in a display of all objects in WebSphere Process Server namespace, including the directory path and object name.
  6. If the object a client needs to access does not appear, use the administrative console to verify the following conditions.
     - The server hosting the target resource is started.

- The Web module or Enterprise JavaBean container hosting the target resource is running.
- The JNDI name of the target resource is properly specified.

If none of these steps solves the problem, see Troubleshooting and support for additional troubleshooting resources, including information about how to contact IBM Support.

- During the migration process, problems might occur during the WBIPreUpgrade or WBIPostUpgrade steps.
  - Problems can occur during the WBIPreUpgrade step.
    - A ″Not found″ or ″No such file or directory″ message is returned.

      This problem can occur if the WBIPreUpgrade script is not in the correct version 6.1 bin directory; for example: /WebSphere/V6R1/DeploymentManager/profiles/default/bin. Verify that the WBIPreUpgrade script is in the correct directory so that the migration job can run the script.
    - The DB2 JDBC driver and DB2 JDBC driver (XA) cannot be found in the drop-down list of supported JDBC providers in the administrative console.

      The administrative console no longer displays deprecated JDBC provider names. The new JDBC provider names used in the administrative console are more descriptive and less confusing. The new providers will differ only by name from the deprecated ones.

      The deprecated names will continue to exist in the jdbc-resource-provider-templates.xml file for migration reasons (for existing JACL scripts for example); however, you are encouraged to use the new JDBC provider names in your JACL scripts.
    - You receive the following message:

      ```
      MIGR0108E: The specified WebSphere directory does not contain a WebSphere
      version that can be upgraded.
      ```

      This can occur if an incorrect directory was used in the WBIPreUpgrade step of the migration job.
  - Problems can occur during the WBIPostUpgrade step.
    - A ″Not found″ or ″No such file or directory″ message is returned.

      This problem can occur if the WBIPostUpgrade script is not in the correct version 6.1 bin directory; for example: /WebSphere/V6R1/DeploymentManager/profiles/default/bin. Verify that the WBIPostUpgrade script is in the correct directory so that the migration job can run the script.
    - When you migrate the federated nodes in a cell, you receive the following error messages:

      ```
      MIGR0304I: The previous WebSphere environment is being restored.
       com.ibm.websphere.management.exception.RepositoryException:
       com.ibm.websphere.management.exception.ConnectorException: ADMC0009E:
         The system failed to make the SOAP RPC call: invoke
      MIGR0286E: The migration failed to complete.
      ```

      A connection timeout occurs when the federated node tries to retrieve configuration updates from the deployment manager during the WBIPostUpgrade migration step for the federated node. Copying the entire configuration might take more than the connection timeout if the configuration that you are migrating to version 6.1 contains any of the following elements:

      - Many small applications
      - A few large applications
      - One very large application

If this occurs, modify the timeout value before running the migration job.

1. Change to the properties directory for the version 6.1 profile to which you are migrating your federated node; for example:

   ```
   /WebSphere/V6R1/AppServer/profiles/default/properties
   ```

2. Open the soap.client.props file in that directory and find the value for the com.ibm.SOAP.requestTimeout property. This is the timeout value in seconds. The default value is 180 seconds.
3. Change the value of com.ibm.SOAP.requestTimeout to make it large enough to migrate your configuration. For example, the following entry would give you a timeout value of a half of an hour:

   ```
   com.ibm.SOAP.requestTimeout=1800
   ```

   Note: Select the smallest timeout value that will meet your needs. Be prepared to wait for at least three times the timeout that you select—once to download files to the backup directory, once to upload the migrated files to the deployment manager, and once to synchronize the deployment manager with the migrated node agent.
4. Go to the following location in the backup directory that was created by the WBIPreUpgrade step of the migration job:

   migration_backup_directory/profiles/default/properties
5. Open the soap.client.props file in that directory and find the value for the com.ibm.SOAP.requestTimeout property.
6. Change the value of com.ibm.SOAP.requestTimeout to the same value that you used in the version 6.1 file.

- You receive the "Unable to copy document to temp file" error message. Here is an example:

  ```
  MIGR0304I: The previous WebSphere environment is being restored.
  com.ibm.websphere.management.exception.DocumentIOException: Unable to copy
  document to temp file:
    cells/sunblade1Network/applications/LARGEApp.ear/LARGEApp.ear
  ```

  Your file system might be full. If your file system is full, clear some space and rerun the WBIPostUpgrade command.
- You receive the following message:

  ```
  MIGR0108E: The specified WebSphere directory does not contain a WebSphere
  version that can be upgraded.
  ```

  The following possible reasons for this error exist:

  - An incorrect directory might have been used when running the WBIPreUpgrade or WBIPostUpgrade steps
  - The WBIPreUpgrade command was not run.
- You receive the following error message:

  MIGR0253E: The backup directory *migration_backup_directory* does not exist.

  The following possible reasons for this error exist:

  - An incorrect backup directory might be specified.

    For example, the directory might have been a subdirectory of the version 6.0.x tree that was deleted after the WBIPreUpgrade command was run and the older version of the product was uninstalled but before the WBIPostUpgrade command was run.

    1. Determine whether or not the full directory structure specified in the error message exists.
    2. If possible, rerun the WBIPreUpgrade command, specifying the correct full migration backup directory.

3. If the backup directory does not exist and the older version it came from is gone, rebuild the older version from a backup repository or XML configuration file.
4. Rerun the WBIPreUpgrade command.

- You decide that you need to run WBIPreUpgrade again after you have already run the WBIPostUpgrade command.

  During the course of a deployment manager or a managed node migration, WBIPostUpgrade might disable the old environment. If after running WBIPostUpgrade you want to run WBIPreUpgrade again against the old installation, you must run the migrationDisablementReversal.jacl script located in the old *install_root*/bin directory. After running this JACL script, your version 6.0.x environment will be in a valid state again, allowing you to run WBIPreUpgrade to produce valid results.

  For more information on scripting, see Getting started with scripting. Scripting, as described there, is available for WebSphere Process Server.

- A federated migration fails with message MIGR0405E.

  The migration that has taken place on your deployment manager as part of your federated migration has failed. For a more detailed reason for why this error has occurred, open the folder *your_node_name*_migration_temp located on your deployment manager node under the ...DeploymentManagerProfile/temp directory. For example:

  ```
  /websphere61/procserver/profiles/dm_profile/temp/nodeX
  _migration_temp
  ```

  The logs and everything else involved in the migration for this node on the deployment manager node are located in this folder. This folder will also be required for IBM support related to this scenario.

- WebSphere Process Server version 6.1 applications are lost during migration.

  If any of the version 6.1 applications fail to install during a federated migration, they will be lost during the synchronizing of the configurations. The reason that this happens is that one of the final steps of WBIPostUpgrade is to run a syncNode command. This has the result of downloading the configuration on the deployment manager node and overwriting the configuration on the federated node. If the applications fail to install, they will not be in the configuration located on the deployment manager node. To resolve this issue, manually install the applications after migration. If they are standard version 6.1 applications, they will be located in the *install_root*/installableApps directory.

  To manually install an application that was lost during migration, use the wsadmin command to run the install_*application_name*.jacl script that the migration tools created in the backup directory.

  See the Wsadmin tool.

- WebSphere Process Server version 6.1 applications fail to install.

  Manually install the applications using the wsadmin command after WBIPostUpgrade has completed.

  To manually install an application that failed to install during migration, use the wsadmin command to run the install_*application_name*.jacl script that the migration tools created in the backup directory.

  See the Wsadmin tool, or see the WBIPostUpgrade command.

- If you select the option for the migration process to install the enterprise applications that exist in the version 6.0.x configuration into the new version 6.1

configuration, you might encounter some error messages during the application-installation phase of migration.

The applications that exist in the version 6.0.x configuration might have incorrect deployment information—usually, incorrect XML documents that were not validated sufficiently in previous WebSphere Process Server runtimes. The runtime now has an improved application-installation validation process and will fail to install these malformed EAR files. This results in a failure during the application-installation phase of WBIPostUpgrade and produces an ″E:″ error message. This is considered a ″fatal″ migration error.

If migration fails in this way during application installation, you can do one of the following:

– Fix the problems in the version 6.0.x applications, and then remigrate.

– Proceed with the migration and ignore these errors.

  In this case, the migration process does not install the failing applications but does complete all of the other migration steps.

  Later, you can fix the problems in the applications and then manually install them in the new version 6.1 configuration using the administrative console or an install script.

- After migrating to a version 6.1 cell that contains or interoperates with version 6.0.x nodes that are not at WebSphere Process Server version 6.0.1.3 or later, the cluster function might fail.

  When starting these version 6.0.x servers, you might see the following problems:

  – You might see a first failure data capture (FFDC) log that shows a ClassNotFoundException error message. This exception is thrown from the RuleEtiquette.runRules method and looks something like the following example:

```
Exception = java.lang.ClassNotFoundException
Source = com.ibm.ws.cluster.selection.SelectionAdvisor.<init>
probeid = 133
Stack Dump = java.lang.ClassNotFoundException: rule.local.server
at java.net.URLClassLoader.findClass(URLClassLoader.java(Compiled Code))
at com.ibm.ws.bootstrap.ExtClassLoader.findClass(ExtClassLoader.java:106)
at java.lang.ClassLoader.loadClass(ClassLoader.java(Compiled Code))
at java.lang.ClassLoader.loadClass(ClassLoader.java(Compiled Code))
at java.lang.Class.forName1(Native Method)
at java.lang.Class.forName(Class.java(Compiled Code))
at com.ibm.ws.cluster.selection.rule.RuleEtiquette.runRules(RuleEtiquette.java
:154)at com.ibm.ws.cluster.selection.SelectionAdvisor.handleNotification
(SelectionAdvisor.java:153)
at com.ibm.websphere.cluster.topography.DescriptionFactory$Notifier.run
(DescriptionFactory.java:257)
at com.ibm.ws.util.ThreadPool$Worker.run(ThreadPool.java:1462)
```

  – You might see a java.io.IOException that looks something like the following example:

```
Exception = java.io.IOException
Source = com.ibm.ws.cluster.topography.DescriptionManagerA. update probeid
   = 362
Stack Dump = java.io.IOException
at com.ibm.ws.cluster.topography.ClusterDescriptionImpl.importFromStream
(ClusterDescriptionImpl.java:916)
at com.ibm.ws.cluster.topography.DescriptionManagerA.update
(DescriptionManagerA.java:360)
Caused by: java.io.EOFException
at java.io.DataInputStream.readFully(DataInputStream.java(Compiled Code))
at java.io.DataInputStream.readUTF(DataInputStream.java(Compiled Code))
at com.ibm.ws.cluster.topography.KeyRepositoryImpl.importFromStream
(KeyRepositoryImpl.java:193)
```

During migration, version 6.1 cluster information is distributed throughout the cell. WebSphere Process Server version version 6.0.x nodes that are not at version 6.0.1.3 or later fail to read this information.

To avoid this problem, upgrade all version 6.0.x nodes that will be contained in or interoperating with a version 6.1 cell to version 6.0.1.3 or later before migrating your deployment managers to version 6.1.

- After you migrate a managed node to version 6.1, the application server might not start.

  When you try to start the application server, you might see errors similar to those in the following example:

```
[5/11/06 15:41:23:190 CDT] 0000000a SystemErr  R
    com.ibm.ws.exception.RuntimeError:
com.ibm.ws.exception.RuntimeError:  org.omg.CORBA.INTERNAL:
   CREATE_LISTENER_FAILED_4
vmcid: 0x49421000  minor code: 56  completed: No
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr  R  at
com.ibm.ws.runtime.WsServerImpl.bootServerContainer(WsServerImpl.java:198)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr  R  at
com.ibm.ws.runtime.WsServerImpl.start(WsServerImpl.java:139)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr  R  at
com.ibm.ws.runtime.WsServerImpl.main(WsServerImpl.java:460)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr  R  at
com.ibm.ws.runtime.WsServer.main(WsServer.java:59)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr  R  at
sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr  R  at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:64)
[5/11/06 15:41:23:197 CDT] 0000000a SystemErr  R  at
sun.reflect.DelegatingMethodAccessorImpl.invoke
    (DelegatingMethodAccessorImpl.java:43)
```

  Change the port number at which the managed node's server is listening. If the deployment manager is listening at port 9101 for ORB_LISTENER_ADDRESS, for example, the server of the managed node should not be listening at port 9101 for its ORB_LISTENER_ADDRESS. To resolve the problem in this example, perform the following steps:

  1. On the administrative console, click **Application servers** → *server_name* → **Ports** → **ORB_LISTENER_ADDRESS**.
  2. Change the ORB_LISTENER_ADDRESS port number to one that is not used.

- If synchronization fails when you migrate a managed node to version 6.1, the server might not start.

  You might receive messages similar to the following when you migrate a managed node to version 6.1:

```
ADMU0016I: Synchronizing configuration between node and cell.
ADMU0111E: Program exiting with error:
           com.ibm.websphere.management.exception.AdminException: ADMU0005E:
           Error synchronizing repositories
ADMU0211I: Error details may be seen in the file:
           /opt/WebSphere/61AppServer/profiles/AppSrv02/logs/syncNode.log
MIGR0350W: Synchronization with the deployment manager using the SOAP protocol
 failed.
MIGR0307I: The restoration of the previous WebSphere Application Server
 environment is complete.
MIGR0271W: Migration completed successfully, with one or more warnings.
```

  These messages indicate the following:

  – Your deployment manager is at a version 6.1 configuration level.

– The managed node that you are trying to migrate is at a version 6.1 configuration level on the deployment manager's repository (including applications).
– The managed node itself is not quite complete given that you did not complete the syncNode operation.

Perform the following actions to resolve this issue:

1. Rerun the syncNode command on the node to synchronize it with the deployment manager.

   See the syncNode command .
2. Run the GenPluginCfg command.

   See the GenPluginCfg command .

- If you encounter a problem when you are migrating from a previous version of WebSphere Process Server to version 6.1, check your log files and other available information.
  – If the migration job fails before the WBIPostUpgrade step, re-run the migration job.
  – If the migration job fails in the WBIPostUpgrade step, the configuration of the new 6.1 server is partially updated so re-create (or restore from backup) the new 6.1 server before re-running the migration job.
  – Problems occur with a managed (federated) node migration.

    A federated node is the most complex node to migrate because it is essentially two migrations rolled into one. A federated node requires a migration of the node configuration information contained in the deployment manager's master repository as well as the configuration information contained in the federated node. Federated node migration requires an active connection to the deployment manager. If you have security enabled, it is essential that you follow the instructions that were generated when you created the migration jobs. The migration job must be submitted with a WebSphere Administrator's user ID that has been properly configured for obtaining secure connections.

    version 6.x node agents might display as not synchronized or not available when you change the deployment manager node name in a mixed cell during migration to the version 6.1 deployment manager. version 6.x node agents maintain a link to the version 6.x deployment manager until they are restarted; therefore, they might fail to synchronize with the new deployment manager. The discovery problem, which prevents automatic synchronization, occurs because the node agent is not yet aware of the deployment manager name change that occurred during the migration. If you experience this problem, perform these steps on the node.

    1. Stop the node.
    2. Run the **syncNode** command.
    3. Restart the node.
  – Job fails during the application-installation phase of migration.

    If you select the option for the migration process to install the enterprise applications that exist in the version 6.0.x configuration into the new version 6.1 configuration, you might encounter error messages during the application-installation phase of migration.

    The applications that exist in the version 6.0.x configuration might have incorrect deployment information—usually, invalid XML documents that were not validated sufficiently in previous WebSphere Process Server runtimes. The runtime now has an improved application-installation validation process and

will fail to install these malformed EAR files. This results in a failure during the application-installation phase of WBIPostUpgrade and produces an "E:" error message. This is considered a "fatal" migration error.

If migration fails in this way during application installation, you can do one of the following:

- Fix the problems in the version 6.0.x applications, and then remigrate.
- Proceed with the migration and ignore these errors.
  1. Restart the migration job in the FINISHUP step to allow the remaining migration functions to be performed.

     Do this by adding the RESTART=FINISHUP parameter to the job card and resubmitting the job.
  2. Later, fix the problems in the applications and then manually install them in the new version 6.1 configuration using the administrative console or an install script.

– Out of space errors occur.

  The migration logs are located in *temporary_directory_location*/*nnnnn*, where *temporary_directory_location* is the value that you specified when you created the migration jobs (where the default is /tmp/migrate) and *nnnnn* is a unique number that was generated during the creation of your migration jobs. Normally, the space requirements for the migration logs are small. If you enable tracing, however, the log files can be quite large. The best practice is to enable tracing only after problems have been found. If tracing is required, try to only enable tracing related to the step in the process that is being debugged. This will help to reduce the space requirements.

  Tracing can be enabled when you create the migration jobs or by changing the variables in the migration JCL from disabled to enabled:

  ```
  TraceState=enabled
  profileTrace=disabled
  preUpGradeTrace=disabled
  postUpGradeTrace=enabled
  ```

  During migration, a backup copy of your version 6.0.x configuration is made. This backup becomes the source of the information being migrated. The default backup location is /tmp/migrate/*nnnnn*. This location can be changed when you create the migration jobs. Depending on the size of the node being migrated, this backup can be quite large. If your temporary space is inadequate, then you will need to relocate this backup.

– Batch job time is exceeded.

  Each z/OS installation is different with respect to job classes and time limitations. Make sure you have specified appropriate job classes and timeout values on your job card.

– Failures occur during server startup after migration.

  Review the instructions that were generated when you created the migration jobs. Verify that the JCL procedures have been copied over correctly to your PROCLIB, the RACF definitions have been created, the version 6.1 libraries have been authorized, and, if required, your STEPLIB statements to the version 6.1 libraries have been specified. Make sure that the daemon process associated with your cell is at the appropriate level. The daemon process must be at the highest WebSphere Process Server for z/OS version level of all servers that it manages within the cell.

  After migrating to a version 6.1 cell that contains or interoperates with version 6.0.x nodes that are not at version 6.0.1.3 or later, the cluster function might fail. When starting these version 6.0.x application servers, you might see the following problems:

- You might see a first failure data capture (FFDC) log that shows a ClassNotFoundException error message. This exception is thrown from the RuleEtiquette.runRules method and looks something like the following example:

```
Exception = java.lang.ClassNotFoundException
Source = com.ibm.ws.cluster.selection.SelectionAdvisor.<init>
probeid = 133
Stack Dump = java.lang.ClassNotFoundException: rule.local.server
at java.net.URLClassLoader.findClass(URLClassLoader.java(Compiled Code))
at com.ibm.ws.bootstrap.ExtClassLoader.findClass(ExtClassLoader.java:106)
at java.lang.ClassLoader.loadClass(ClassLoader.java(Compiled Code))
at java.lang.ClassLoader.loadClass(ClassLoader.java(Compiled Code))
at java.lang.Class.forName1(Native Method)
at java.lang.Class.forName(Class.java(Compiled Code))
at com.ibm.ws.cluster.selection.rule.RuleEtiquette.runRules
(RuleEtiquette.java:154)
at com.ibm.ws.cluster.selection.SelectionAdvisor.handleNotification
(SelectionAdvisor.java:153)
at com.ibm.websphere.cluster.topography.DescriptionFactory$Notifier.run
(DescriptionFactory.java:257)
at com.ibm.ws.util.ThreadPool$Worker.run(ThreadPool.java:1462)
```

- You might see a java.io.IOException that looks something like the following example:

```
Exception = java.io.IOException
Source = com.ibm.ws.cluster.topography.DescriptionManagerA. update probeid
 = 362
Stack Dump = java.io.IOException
at com.ibm.ws.cluster.topography.ClusterDescriptionImpl.importFromStream
(ClusterDescriptionImpl.java:916)
at com.ibm.ws.cluster.topography.DescriptionManagerA.update
(DescriptionManagerA.java:360)
Caused by: java.io.EOFException
at java.io.DataInputStream.readFully(DataInputStream.java(Compiled Code))
at java.io.DataInputStream.readUTF(DataInputStream.java(Compiled Code))
at com.ibm.ws.cluster.topography.KeyRepositoryImpl.importFromStream
(KeyRepositoryImpl.java:193)
```

  During migration, version 6.1 cluster information is distributed throughout the cell.version 6.0.x nodes that are not at version 6.0.1.3 or later fail to read this information. To avoid this problem, upgrade all version 6.0.x nodes that will be contained in or interoperating with a version 6.1 cell to version 6.0.1.3 or later before migrating your deployment managers to version 6.1.

After migration, carefully review the job output and log files for errors.

Note: WebSphere Process Server provides an interactive problem control system (IPCS) verb exit to help you to format information from dumps of WebSphere Process Server processes. This verb exit was named CBDATA, which was an alias of the real module name, in version 6.0.x and earlier. In version 6.1, that alias was removed. In version 6.1 and later, therefore, you must use the real name of this verb exit, BBORDATA, instead of the alias.

If you migrate a node to version 6.1 then discover that you need to revert back to version 6.0.x, see Rolling back your environment.

If none of these steps solves the problem, see Troubleshooting and support for additional troubleshooting resources, including information about how to contact IBM Support.

What to do next

If you did not find your problem listed, contact IBM support.

Chapter 7. Troubleshooting a failed deployment

This topic describes the steps to take to determine the cause of a problem when deploying an application. It also presents some possible solutions.

Before you begin

This topic assumes the following things:
- You have a basic understanding of debugging a module.
- Logging and tracing is active while the module is being deployed.

About this task

The task of troubleshooting a deployment begins after you receive notification of an error. There are various symptoms of a failed deployment that you have to inspect before taking action.

Procedure

1. Determine if the application installation failed.

   Examine the SystemOut.log file for messages that specify the cause of failure. Some of the reasons an application might not install include the following:
   - You are attempting to install an application on multiple servers in the same Network Deployment cell.
   - An application has the same name as an existing module on the Network Deployment cell to which you are installing the application.
   - You are attempting to deploy J2EE modules within an EAR file to different target servers.

   Important: If the installation has failed and the application contains services, you must remove any SIBus destinations or J2C activation specifications created prior to the failure before attempting to reinstall the application. The simplest way to remove these artifacts is to click **Save > Discard all** after the failure. If you inadvertently save the changes, you must manually remove the SIBus destinations and J2C activation specifications (see Deleting SIBus destinations and Deleting J2C activation specifications in the Administering section).
2. If the application is installed correctly, examine it to determine if it started successfully.

   If the application did not start successfully, the failure occurred when the server attempted to initiate the resources for the application.
   a. Examine the SystemOut.log file for messages that will direct you on how to proceed.
   b. Determine if resources required by the application are available and/or have started successfully.

      Resources that are not started prevent an application from running. This protects against lost information. The reasons for a resource not starting include:
      - Bindings are specified incorrectly
      - Resources are not configured correctly
      - Resources are not included in the resource archive (RAR) file
      - Web resources not included in the Web services archive (WAR) file

c. Determine if any components are missing.

   The reason for missing a component is an incorrectly built enterprise archive (EAR) file. Make sure that the all of the components required by the module are in the correct folders on the test system on which you built the Java™ archive (JAR) file. “Preparing to deploy to a server” contains additional information.

3. Examine the application to see if there is information flowing through it.

   Even a running application can fail to process information. Reasons for this are similar to those mentioned in step 2b on page 35.

   a. Determine if the application uses any services contained in another application. Make sure that the other application is installed and has started successfully.

   b. Determine if the import and export bindings for devices contained in other applications used by the failing application are configured correctly. Use the administrative console to examine and correct the bindings.

4. Correct the problem and restart the application.

Chapter 8. Troubleshooting WebSphere Process Server administration

Troubleshooting is the process of finding and eliminating the cause of a problem. This group of topics helps you identify and resolve problems that can occur during typical administration tasks.

For information on troubleshooting Business Process Choreographer or Common Event Infrastructure components, see one of the following locations:

- The WebSphere Process Server for z/OS, version 6.1, information center
- The *Business Process Choreographer* PDF
- The *Common Event Infrastructure* PDF

Troubleshooting the failed event manager

This topic discusses problems that you can encounter while using the failed event manager.

Note: This topic does not discuss how to use the failed event manager to find, modify, resubmit, or delete failed events on the system. For information about managing failed events, see *Managing WebSphere Process Server failed events* in the information center.

Select the problem you are experiencing from the table below:

Problem	Refer to the following
I am having trouble with reduced performance during an advanced search	"Advanced search feature is not optimized"
I am having trouble entering values in the Search page's **By Date** tab	"Values in the By Date tab automatically change to default if entered incorrectly"
I am having trouble deleting expired events	"Using the Delete Expired Events function appears to suspend the failed event manager" on page 38
I am having trouble with failed events not being created	"Failed events are not being created" on page 38

Advanced search feature is not optimized

The failed event manager's advanced search feature is not optimized. Therefore, you may experience reduced performance when using the Advanced search tab with a large set of failed events.

Values in the By Date tab automatically change to default if entered incorrectly

The Search page's **By Date** tab contains two fields: **From Date** and **To Date**. Both fields are required. The values are locale-dependendent, and they must be formatted exactly as shown in the example above the field. Any inconsistency in the value's format (for example, including four digits in the year instead of 2, or

omitting the time) will cause the failed event manager to issue the following warning and substitute a default value in the field:

```
CWMAN0017E: The date entered could not be parsed correctly:
your_incorrectly_formatted_date. Date: default_date is being used.
```

The default value of the **From Date** field is defined as `January 1, 1970, 00:00:00 GMT`.

Important: The actual default value shown in your failed event manager implementation will vary depending on your locale and time zone. For example, the From Date field defaults to `12/31/69 7:00 PM` for a machine with an en_US locale in the Eastern Standard Time (EST) time zone.
The default value for the **To Date** field is always the current date and time, formatted for your locale and time zone.

To avoid this problem, always enter your dates and times carefully, following the example provided above each field.

Using the Delete Expired Events function appears to suspend the failed event manager

If you use the Delete Expired Events button in situations where there are many failed events in the current search results, or where those events contain a large amount of business data, the failed event manager can appear to be suspended indefinitely.

In this situation, the failed event manager is not actually suspended: it is working through the large data set, and will refresh the results set as soon as the command completes.

Failed events are not being created

If the Recovery subsystem is not creating failed events, go through the following checklist of potential causes:

- Ensure that the wpsFEMgr application is running. If necessary, restart it.
- Ensure that the failed event manager's database has been created, and that the connection has been tested.
- Ensure that the necessary failed event destination has been created on the SCA system bus. There should be one failed event destination for each deployment target.
- Ensure that the Quality of Service (QoS) **Reliability** qualifier has been set to `Assured` for any Service Component Architecture (SCA) implementation, interface, or partner reference that participates in events you want the Recovery service to handle.

Troubleshooting Service Component Architecture and WebSphere MQ communications

Communication between Service Component Architecture (SCA) modules and WebSphere MQ queue managers depends on the binding between the imports and exports within the SCA module and the queues in WebSphere MQ servers. Use this information to determine the servers that are not processing WebSphere MQ messages.

Before you begin

This task assumes that you have noticed requests dependant on WebSphere MQ are not being processed and that you have access to the administrative console. You should also either have the ability to make changes to the WebSphere MQ queue manager or be in contact with the WebSphere MQ administrator.

About this task

Service Component Architecture (SCA) modules depend on the bindings between the server and the WebSphere MQ queue manager. Communications between the two entities could keep messages from processing completely. The following steps should help you discover the cause of the disruption and what to do to get the messages processed again.

Procedure

1. Display the SCA module communicating with WebSphere MQ to make sure it is still processing. Navigate to this page using **Applications > SCA Modules**.
2. Display the queue manager to make sure it is still operational. Perform this task at the WebSphere MQ administration console.
3. Display the bindings between the SCA module and the queue manager to make sure the binding is correct. If the binding is incorrect, change the binding. Navigate to this page using **Applications** → **SCA modules** >*moduleName* > **Imports | Exports** > *importName | exportName* > **Bindings** > *bindingName* [type].
4. Locate any messages that may indicate failed transactions. You will have to investigate system, SCA-specific message areas, WebSphere MQ-specific message areas, the failed event queue and other locations to determine what has failed.
   a. Examine SystemOut.log for any messages that would indicate processing failures.

      If there is an WebSphere MQ error, there will be an MQException linked somewhere in the stack trace with a WebSphere MQ reason code (for example, 2059 is “queue manager unavailable”).
   b. Check AMQERR*xx*.LOG and the WebSphere MQ FFDC files to determine the cause of a WebSphere MQ error.
   c. Examine the application queues to determine if there are any unprocessed messages. Make sure you examine both WebSphere MQ and Service Integration Bus (SIB) queues.
   d. Examine the WebSphere MQ dead letter queue and the SIB exception destination.
   e. Examine the failed event queue to determine if there are any messages related to the applications of interest. See Finding failed events for information about locating the failed events. See “Managing WebSphere Process Server failed events” for information about locating the failed events.

Troubleshooting event sequencing

Refer to the information in this topic if you are experiencing difficulty with event sequencing.

Problems with the event sequencing qualifier

Ensure that your component definition is correct:

- Is the event sequencing qualifier set on the method? Event sequencing validation fails if the qualifier is erroneously set on the interface.
- Is the parameter name valid?
- Is the xpath element valid, and does it correctly resolve to a primitive?
- Is there a single eventSequencing element for the method? Each method supports only one eventSequencing element.
- Is there a single keySpecification element for the method? Each method supports only one keySpecification element.

Deadlocks

Deadlocks occur when an invoked operation with a lock invokes another operation on the same component using the same event sequencing key and group. You can resolve a deadlock by using the esAdmin command to list and release the current lock.

To avoid deadlocks, carefully consider dependencies when implementing event sequencing. Ensure that operations with circular dependencies are in different event sequencing groups.

Deadlocks with a BPEL process

Deadlocks can occur when event sequencing is used with Business Process Execution Language (BPEL) processes. Deadlocks are caused by setting event sequencing qualifiers on operations that correspond to both of the following activities:

- Multiple instantiating receive or pick activities, where the `createInstance` attribute is set to `yes`
- Correlation set specifications with an `initiation` attribute set to `join`

Resolve this type of deadlock by using the esAdmin command to list and release the current lock. To prevent further deadlocks, ensure that these types of dependent operations are put into different event sequencing groups.

Performance issues

If you are experiencing memory problems on the messaging engine server used for event sequencing components, try modifying the runtime event sequencing property maxActiveMessages in the *install_root*/properties/eventsequencing.properties file.

The maxActiveMessages property defines the number of messages currently locked on a component destination; too many large messages can negatively affect performance and cause memory problems. Note that a value of `0` (zero) means that an unlimited number of messages are allowed. By default, the maxActiveMessages property is set to 100.

To modify the maxActiveMessages property, perform the following steps.

1. Open the eventsequencing.properties file in a text editor.
2. Make the appropriate modifications for your environment.

3. Save and close the file.
4. Stop and restart any applications that are part of the event sequencing component in order for the changes to take effect.

Troubleshooting your deployment environment

When processing appears sluggish or requests fail, use a focused approach to determine the source of the problem in the environment. The approach described is for non-standalone server environments.

Before you begin

You must be logged into the administrative console of the deployment manager to perform this task.

Required security role for this task: When security and role-base authorization are enabled, you must log in as an administrator or operator to perform this task.

About this task

Investigate the state of your deployment environment if you notice any of the following symptoms:
- Unavailable applications
- Sluggish applications
- Stopped applications
- Decreased throughput
- Sluggish performance

Procedure

1. Display the topology layout that describes this deployment environment to determine the status of the topology.
2. Display the topology to determine the state of the various roles in the topology. Note the roles with unexpected states or warning for further investigation.
3. Locate the nodes that are causing the error state for each role.
4. Make sure all nodes are synchronized.

   On the Nodes page in the administrative console select any unsynchronized nodes and click **Synchronize**.
5. Make sure that the messaging engines associated with all the buses are running.

   If they are not running, stop and start the messaging engines.
6. Locate the logs associated with the nodes in error and view the logs for error messages.
7. Take any actions prescribed by the error messages to affect the correction.
8. Correct any errors and restart the affected nodes.

Results

The nodes previously in error start and the status of the topology becomes "running."

What to do next

Restart any affected applications

Troubleshooting the business rules manager

Some of the problems you may encounter using the business rules manager are login errors, login conflicts, and access conflicts.

You can take various steps to troubleshoot these problems.

Resolving login errors

A log in error occurs upon logging in.

About this task

The login error message is as follows:

```
Unable to process login. Please check User ID and password and try again.
```

Note: Login errors occur only when global security is enabled and either the user ID, password, or both, are incorrect.

To resolve login errors, perform the following steps.

Procedure

1. Click **OK** on the error message to return to the Login page.
2. Enter the valid **User ID** and **Password**.
   - If passwords are case sensitive, make sure that Caps Lock key is not on.
   - Make sure the user ID and password are spelled correctly.
   - Check with the system administrator to be sure that the user ID and password are correct.
3. Click **Login**.

What to do next

If you resolve the login error, you will now be able to log in to the business rules manager. If the error is not resolved, contact your system administrator.

Resolving login conflict errors

A login conflict error occurs when another user with the same user ID is already logged in to the application.

About this task

The login conflict message is as follows:

```
Another user is currently logged in with the same User ID. Select from the
following options:
```

Usually this error occurs when a user closed the browser without logging out. When this condition occurs, the next attempted login before the session timeout expires results in a login conflict.

Note: Login conflict errors occur only when global security is enabled.

To resolve login conflict errors, select from the following three options:

- Return to the Login page.
  Use this option if you want to open the application with a different user ID.
- Log out the other user with the same user ID.
  Use this option to log out the other user and start a new session.

  Note: Any unpublished local changes made in the other session will be lost.
- Inherit the context of the other user with the same user ID and log out that user.
  Use this option to continue work already in progress. All unpublished local changes in the previous session that have been saved will not be lost. The business rules manager will open to the last page displayed in the previous session.

Resolving access conflict errors

An access conflict error occurs when a business rule is updated in the data source by one user at the same time another user is updating the same rule.

Before you begin

This error is reported when you publish your local changes to the repository.

About this task

To correct access conflict errors, perform the following actions:

- Find the source of the business rule that is causing the error and check if your changes on the local machine are still valid. Your change may no longer be required after the changes done by another user.
- If you choose to continue working in the business rules manager, you must reload those business rule groups and rule schedules in error from the data source as your local changes of business rule groups and rule schedules in error are no longer usable. Reload a business rule group or rule schedule page, by clicking **Reload** in the Publish and Revert page of the rule for which the error was reported. You can still use local changes in other business rule groups and rule schedules that are not in error.

Troubleshooting bindings

Various error conditions can occur with bindings that are specific to the type of binding.

About this task

The manner in which error conditions are handled depends upon the type of binding concerned.

Troubleshooting JMS and WebSphere MQ JMS bindings

You can diagnose and fix problems with JMS and WebSphere MQ JMS bindings.

Implementation exceptions

About this task

In response to various error conditions, the JMS and MQ JMS import and export implementation can return one of two types of exceptions:

- ServiceBusinessException: this exception is returned if exception specified on the service business interface (WSDL port type or Java interface) occurred.
- ServiceRuntimeException: raised in all other cases. In most cases, the cause exception will contain the original exception. In the J2C case that would be ResourceException and in the JMS case, it would be JMSException.

When an XML schema has a type defined without a global element, the JMS bindings (JMSDataBindingImplXML and JMSDataBindingImplJava) cannot resolve the type to an element.

Schema:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.ibm.com" xmlns:tns="http://www.ibm.com">
 <!-- global element required but missing -->

<complexType name="Quote">
  <sequence>
   <element name="symbol" type="string"></element>
  <element name="price" type="float"></element>
  </sequence>
 </complexType>
</schema>
```

If you receive one of the following exceptions:

```
com.ibm.websphere.sca.ServiceRuntimeException:
caused by: java.lang.IllegalArgumentException:
{Quote}Quote is not corresponding to a global element.
```

Or

```
[8/25/06 10:20:40:938 PDT] 00000054 FFDC          Z
com.ibm.ws.sca.databinding.impl.DataBindingImplXML
com.ibm.ws.sca.databinding.impl.DataBindingImplXML#002 Exception:

org.eclipse.emf.ecore.xmi.FeatureNotFoundException:
Feature 'Quote' not found. (sca:/dataObject.xml, 2, 126)
```

This might mean you need to define a global element:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.ibm.com" xmlns:tns="http://www.ibm.com">

<element name="Quote" type="tns:Quote"></element> <!-- global element required -->

<complexType name="Quote">
  <sequence>
   <element name="symbol" type="string"></element>
   <element name="price" type="float"></element>
  </sequence>
 </complexType>

</schema>
```

JMS-based SCA messages not appearing in the failed event manager

About this task

If SCA messages originated through a JMS interaction fail, you would expect to find these messages in the failed event manager. If such messages are not appearing in the failed event manager, ensure that the underlying SIB destination

of the JMS Destination has a maximum failed deliveries value greater than 1. Setting this value to 2 or more enables interaction with the failed event manager during SCA invocations for the JMS bindings.

WebSphere MQ JMS-based SCA messages not appearing in the failed event manager

About this task

If SCA messages originated through a WebSphere MQ JMS interaction fail, you would expect to find these messages in the failed event manager. If such messages are not appearing in the failed event manager, ensure that the value of the maximum retries property on the underlying listener port is equal to or greater than 1. Setting this value to 1 or more enables interaction with the failed event manager during SCA invocations for the MQ JMS bindings.

Faults

About this task

The argument that is expected to be passed to the JMSDataBinding and JMSObjectBinding depends on the interface operation and the input, output and fault types.

For faults, the outDataBindingType specified on the method binding is used. If none is specified, the binding level dataBindingType is used for all serialization and deserialization.

If the fault type is simple, a string is set on the JMS DataBinding representing the fault message. In addition IsBusinessException is set to true.

If the fault type is a data object, then a data object is set on the JMS DataBinding which represents the fault message. This scenario requires the use of JMSDataBinding.

Messages containing faults are handled by the JMS data bindings. A Boolean header property IsBusinessException is intercepted by the data binding. If the value is true, the data binding informs the runtime that the payload contains fault data.

If you are working with a custom data binding, you need to take the following steps to handle faults correctly. Default implementations handle faults without user intervention.

Procedure

1. For JMS exports, use the setBusinessException (boolean isBusinessException) method on the JMSDataBinding interface to indicate that data object or object specified on the data binding is a fault object and the message created by the binding should be constructed accordingly. The Data Binding is then responsible to specify the isBusinessException appropriately.
2. For JMS imports use the isBusinessException() method on the JMSDataBinding interface to indicate whether the message contains a fault.

   The data binding gets the value of the header property indicating a fault defined in the payload. After the runtime passes the JMS message to the data binding, it invokes isBusinessException() on the data binding. If returned value is false, the message is processed normally, otherwise, the

ServiceBusinessException is returned to the caller. The data object or object produced by the binding is set on a ServiceBusinessException and it is returned to the caller.

Misusage scenarios: comparison with WebSphere MQ bindings

About this task

The WebSphere MQ JMS binding is designed to interoperate with JMS applications deployed against WebSphere MQ, which exposes messages according to the JMS message model. The WebSphere MQ import and export, however, are principally designed to interoperate with native WebSphere MQ applications, and expose the full content of the WebSphere MQ message body to mediations.

The following scenarios should be built using the WebSphere MQ JMS binding, not the WebSphere MQ binding:

- Invoking a JMS message-driven bean (MDB) from an SCA module, where the MDB is deployed against the WebSphere MQ JMS provider. This would be implemented using a WebSphere MQ JMS import.
- Allowing the SCA module to be called from a J2EE component servlet or EJB via JMS. This would be implemented using a WebSphere MQ JMS export.
- Mediating the contents of a JMS MapMessage, in transit across WebSphere MQ. This would use WebSphere MQ JMS export and import in conjunction with the appropriate data binding.

There are situations in which the WebSphere MQ binding and WebSphere MQ JMS binding might be expected to interoperate. In particular, when you are bridging between J2EE and non-J2EE WebSphere MQ applications, use a WebSphere MQ export and WebSphere MQ JMS import (or vice versa) in conjunction with appropriate data bindings, mediation module, or both.

Troubleshooting generic JMS bindings

Certain failure conditions can occur with a generic JMS binding.

About this task

Various error conditions can occur with generic JMS bindings.

Troubleshooting generic JMS exceptions

In response to various error conditions, the generic JMS import and export implementation can return an exception.

About this task

In response to various error conditions, the generic JMS import and export implementation can return one of two types of exceptions:

- ServiceBusinessException – this exception is returned if the exception specified on the service business interface (WSDL port type or Java interface) occurred.
- ServiceRuntimeException – raised in all other cases. In most cases, the *cause* exception will contain the original exception. In the case of JMS, it would be JMSException.

Troubleshooting generic JMS message expiry

A request message by the JMS provider is subject to expiration.

About this task

Request expiry refers to the expiration of a request message by the JMS provider when the JMSExpiration time on the request message is reached. As with other JMS bindings, the generic JMS binding handles the request expiry by setting expiration on the callback message placed by the import to be the same as for the outgoing request. Notification of the expiration of the callback message will indicate that the request message has expired and the client should be notified by means of a business exception.

If the callback destination is moved to the third-party provider, however, this type of request expiry is not supported.

Response expiry refers to the expiration of a response message by the JMS provider when the JMSExpiration time on the response message is reached.

Response expiry for the generic JMS binding is not supported, because the exact expiry behavior of a third-party JMS provider is not defined. You can, however, check that the response is not expired if and when it is received.

For outbound request messages, the JMSExpiration value will be calculated from the time waited and from the requestExpiration values carried in the asyncHeader, if set.

Troubleshooting generic JMS connection factory errors

When you define certain types of Connection Factory in your generic JMS provider, you may receive an error message when you try to start an application. You can modify the external Connection Factory to avoid this problem.

About this task

When launching an application you may receive the following error message: "MDB Listener Port JMSConnectionFactory type does not match JMSDestination type"

This problem can arise when you are defining external connection factories. Specifically the exception can be thrown when you create a JMS 1.0.2 Topic Connection Factory, instead of a JMS 1.1 (unified) Connection Factory (i.e., one that is able to support both point-to-point and publish/subscribe communication).

To resolve this issue take the following steps:

Procedure

1. Access the generic JMS provider that you are using.
2. Replace the JMS 1.0.2 Topic Connection Factory that you defined with a JMS 1.1 (unified) Connection Factory.

Results

When you launch the application with the newly defined JMS 1.1 Connection Factory you should no longer receive an error message.

Troubleshooting WebSphere MQ bindings

You can diagnose and fix faults and failure conditions that occur with WebSphere MQ bindings.

About this task

The primary failure conditions of WebSphere MQ bindings are determined by transactional semantics, by WebSphere MQ configuration, or by reference to existing behavior in other components. The primary failure conditions include:

- Failure to connect to WebSphere MQ queue manager or queue: a failure to connect to WebSphere MQ, to receive messages, will result in the MDB ListenerPort failing to start. This condition will be logged in the WebSphere Application Server log. Persistent messages will remain on the WebSphere MQ queue until they are successfully retrieved (or expired by WebSphere MQ). A failure to connect to WebSphere MQ, to send outbound messages, will cause rollback of the transaction controlling the send.
- Failure to parse inbound message, or to construct outbound message: a failure in the data binding will cause rollback of the transaction controlling the work.
- Failure to send outbound message: a failure to send a message will cause rollback of the relevant transaction
- Multiple or unexpected response messages: the import expects only one response message for each request message; when a response has arrived, the record is deleted. If response messages arrive unexpectedly, they will be discarded as with the JMS import.

Misusage scenarios: comparison with WebSphere MQ JMS bindings

About this task

The WebSphere MQ import and export are principally designed to interoperate with native WebSphere MQ applications, and expose the full content of the WebSphere MQ message body to mediations. The WebSphere MQ JMS binding, however, is designed to interoperate with JMS applications deployed against WebSphere MQ, which exposes messages according to the JMS message model.

The following scenarios should be built using the WebSphere MQ JMS binding, not the WebSphere MQ binding:

- Invoking a JMS message-driven bean (MDB) from an SCA module, where the MDB is deployed against the WebSphere MQ JMS provider. This would be implemented using a WebSphere MQ JMS import.
- Allowing the SCA module to be called from a J2EE component servlet or EJB via JMS. This would be implemented using a WebSphere MQ JMS export.
- Mediating the contents of a JMS MapMessage, in transit across WebSphere MQ. This would use WebSphere MQ JMS export and import in conjunction with the appropriate data binding.

There are situations in which the WebSphere MQ binding and WebSphere MQ JMS binding might be expected to interoperate. In particular, when you are bridging between J2EE and non-J2EE WebSphere MQ applications, use a WebSphere MQ export and WebSphere MQ JMS import (or vice versa) in conjunction with appropriate data bindings and/or mediation module.

If WebSphere MQ cannot deliver a message to its intended destination, usually due to configuration errors, it will send messages instead to a nominated dead-letter

queue. In doing so it adds a dead-letter header to the start of the message body; this header contains failure reasons, the original destination, and other information.

Profile-specific log files

There are log files detailing the characteristics and runtime activities of individual profiles. These log files are located within the logs directory of the profile path.

There are a number of log files that are created for each profile. Some of these logs describe the parameters used for the creation of the profile. These types of log files generally remain unchanged once the profile is fully configured. Other profile-specific logs are continually updated to capture error, warning, and information messages emitted during runtime. Some of these log files are also used to capture a Common Base Event (that may include business object data) that is selected for monitoring.

The table below specifies the different types of profile-specific log files and the locations where you can find them within the product. Within the table, the variable *install_root* represents the installation directory of WebSphere Process Server. The variable *profile_root* represents the root location of a profile.

Table 1. Profile-specific log files updated during runtime

Log	Contents
First failure data capture (ffdc) log and exception files (common to all profile types) are found in *profile_root*/logs/ffdc	Contains the ffdc log and exception files for individual profiles. There are two types of ffdc logs: a single log file with a compilation of all the errors encountered during the profile runtime, and numerous text files with details such as stack traces and other information. The naming conventions for the different types of profiles are given for both files, as follows: • **Deployment manager profile:** – Log file — *deployment_manager_name*_exception.log. – Text files — *deployment_manager_name_hex_id_date_time*.txt. • – Log file(s) — *node_agent_name*_exception.log and *server_name*_exception.log . – Text files — *node_agent_name*(or)*server_name_hex_id_date_time*.txt. • **Managed node profile:** – Log file(s) — *node_agent_name*_exception.log and *server_name*_exception.log . – Text files — *node_agent_name*(or)*server_name_hex_id_date_time*.txt. • **Stand-alone profile:** – Log file — *server_name*_exception.log. – Text files — *server_name_hex_id_date_time*.txt.

Table 1. Profile-specific log files updated during runtime (continued)

Log	Contents
Deployment manager logs SystemErr.log and SystemOut.log are found in deployment manager job logs unless customized to write to a file. The start and stop server logs are found in *profile_root*/logs/dmgr if started in USS.	You will work primarily with two log files in this directory: • startServer.log — Contains the system parameters detected on the system and the messages emitted by the deployment manager during the start process • stopServer.log — Contains the system parameters detected on the system and the messages emitted when the deployment manager is shut down.
Node agent logs SystemErr.log and SystemOut.log are found in node agent job logs unless customized to write to a file. The start and stop server logs are found in *profile_root*/logs/nodeagent if started in USS.	You will work primarily with two log files in this directory: • startServer.log — Contains the system parameters detected on the system and the messages emitted by the node agent during the start process • stopServer.log — Contains the system parameters detected on the system and the messages emitted when the node agent is shut down.
Server logs SystemErr.log and SystemOut.log are found in server job logs unless customized to write to a file. The start and stop server logs are found in *profile_root*/logs/<servername> if started in USS.	You will work primarily with two log files in this directory: • startServer.log — Contains the system parameters detected on the system and the messages emitted by the server during the start process • stopServer.log — Contains the system parameters detected on the system and the messages emitted when the server is shut down.
Node federation log files are found in the *profile_root*/logs/ directory.	Two log files are generated when you attempt to federate a node to a deployment manager: • addNode.log — contains the pertinent server environment information and messages generated when you attempt to federate the profile. • isFederated.log — lists the commands used by the deployment manager to federate the profile.
The location of the Integrated Solutions Console application deployment log file is listed here (only for deployment manager and stand-alone profiles):The location of the Integrated Solutions Console application deployment log file is in the <*profilepath*>/logs directory.	The iscinstall.log file contains information regarding the deployment of the administration console application in a deployment manager or stand-alone profile.
The location of the log files for a profile creation is the *install_root*/logs/ directory.	These files contain the output from profile creation. All profile types will contain this file.

Chapter 9. Troubleshooting WebSphere Application Server

Because IBM WebSphere Process Server is built on IBM WebSphere Application Server, the function that you are having problems with may be provided by the underlying WebSphere Application Server. You might want to consult troubleshooting information in the WebSphere Application Server documentation.

WebSphere Process Server is built on WebSphere Application Server, version 6.1.

For more information about troubleshooting in WebSphere Application Server, see Troubleshooting and support in the WebSphere Application Server Information Center.

Chapter 10. Tools for troubleshooting your applications

WebSphere Process Server and WebSphere Integration Developer include several tools you can use to troubleshoot your applications that you develop and deploy on the server.

During development of your applications, you can use debugging tools in WebSphere Integration Developer. You can implement runtime troubleshooting capabilities into your applications using logging, tracing, and service component event monitoring. Administrators of running applications can use the failed event manager to view, modify, resubmit, and delete failed operations between Service Component Architecture (SCA) components.

Debugging applications in WebSphere Integration Developer

In order to debug applications that are running on WebSphere Process Server, you must use your application development tool, such as IBM WebSphere Integration Developer.

About this task

For more information about debugging applications, see **Debugging components** in the IBM WebSphere Business Process Management information center or in the online documentation installed with WebSphere Integration Developer.

Using logging, tracing, and monitoring in your applications

Designers and developers of applications that run on WebSphere Process Server can use capabilities such as monitoring and logging that add troubleshooting features to applications.

About this task

WebSphere Process Server is built on IBM WebSphere Application Server, Network Deployment, version 6.1. For more information, see Adding logging and tracing to your application in the WebSphere Application Server Information Center.

To use logging, tracing, and monitoring with your applications, perform the following steps.

- You can set up service component event monitoring for applications running on WebSphere Process Server. For more information, see Monitoring service component events.
- You can add logging and tracing to your applications using WebSphere Application Server. For more information, see Adding logging and tracing to your application.

Managing failed events

The WebSphere Process Server Recovery service monitors for failed operations between Service Component Architecture (SCA) components. If an operation fails, the Recovery service captures data about the event and the failure. You can then use the failed event manager to view, modify, resubmit, or delete the failed event.

What is a failed event?

In the context of WebSphere Process Server, an event is a request that is received by a WebSphere Process Server application. It can come from an external source (such as an inbound application adapter) or an external invocation to a Web service. The event is comprised of a reference to the business logic it wants to operate and its data, stored in a Service Data Object (a business object). When an event is received, it is processed by the appropriate WebSphere Process Server application business logic.

A single thread of execution can branch off into multiple branches (or threads); the individual branches are linked to the main invoking event by the same session context.

If this business logic in one of these branches cannot execute completely due to system failure, component failure, or component unavailability, the event moves into the failed state. If multiple branches fail, a failed event is created for each. The WebSphere Process Server Recovery service handles the following types of failed events:

- Event failures that occur during an asynchronous invocation of a Service Component Architecture (SCA) operation
- Event failures that are caused by a runtime exception (in other words, any exception that is not declared in the methods used by the business logic)

The Recovery service does not handle failures from synchronous invocations or from Business Process Execution Language (BPEL) asynchronous request/reply invocations.

Failed events typically have source and destination information associated with them. The source and destination are based on the failure point (the location where the invocation fails), regardless of the type of interaction. Consider the following example, where Component A is asynchronously invoking Component B. The request message is sent from A to B, and the response message is sent from B to A.

- If the exception occurs during the initial request, Component A is the source and Component B is the destination for the purposes of the failed event manager.
- If the exception occurs during the response, Component B is the source and Component A is the destination for the purposes of the failed event manager.

This is true for all asynchronous invocations.

How are failed events managed?

The Recovery service sends failed asynchronous interactions to failed event destinations that have been created on the SCA system bus (SCA.SYSTEM.*cell_name*.Bus). The data for failed events is stored in the failed event database (by default, WPCRSDB) and is made available for administrative purposes through the failed event manager interface.

An administrator uses the failed event manager to browse and manage all WebSphere Process Server failed events. Common tasks for managing failed events include:

- Browsing all failed events
- Searching for failed events by specific criteria
- Editing data for a failed event
- Resubmitting failed events

- Deleting failed events

To access the failed event manager, click **Integration Applications → Failed Event Manager**.

Security considerations for recovery

If you have enabled security for your WebSphere Process Server applications and environment, it is important to understand how role-based access and user identity affect the Recovery subsystem.

Role-based access for the failed event manager

The failed event manager uses role-based access control for the failed event data and tasks. Only the administrator and operator roles are authorized to perform tasks within the failed event manager. Users logged in as either administrator or operator can view all data associated with failed events and can perform all tasks.

Event identity and user permissions

A failed event encapsulates information about the user who originated the request. If a failed event is resubmitted, its identity information is updated to reflect the user who resubmitted the event. Because different users logged in as administrator or operator can resubmit events, these users must be given permissions to the downstream components required to process the event.

For more information about implementing security, see Securing applications and their environment.

Finding failed events

Failed events are stored in the failed event database and are retrieved through the search functionality of the failed event manager. You can search for all failed events on the all the servers within the cell, or for a specific subset of events.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

This topic describes how to find all failed events on the server, with references to topics for conducting other searches based on session ID, source, destination, date, business object type, exception text, or a combination of those criteria.

To retrieve a complete list of failed events, use the following procedure.

Procedure

1. Ensure the administrative console is running.
2. Click **Integration Applications → Failed Event Manager** to enter the failed event manager.
3. Click **Failed events on this server → Get all failed events**.

Results

The Search Results page opens, displaying a list of all the WebSphere Process Server failed events in the cell.

What to do next

You can now view and modify data in a failed event, resubmit it, or delete it.

Searching for failed events by session ID

Use the Search page's **By Session** tab to find only those events that failed within a specific session.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

Every event executes within a session; if that event fails, the failed event manager encapsulates specific session information for the failed execution branch in the Session ID parameter. The same session ID is given to all resources and processes that are part of a session, including Common Base Events and business processes.

To search for failed events by session ID, use the following procedure.

Procedure

1. Ensure the administrative console is running and then click **Integration Applications → Failed Event Manager**, to enter the failed event manager.
2. From the main failed event manager page, click **Search by session**.
3. Use the **Session ID** field to specify the session ID to search against.
4. Click **OK** to begin the search.

Results

The Search Results page opens, listing all failed events that originated during the specified session.

Searching for failed events by destination

Use the **By Destination** tab on the Search page to find only those failed events that are associated with a specific destination module, component, or method. The failed event manager determines the destination based on the point of failure, regardless of the type of interaction.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

When performing a search, note the following:

- The values for the fields are case sensitive.
- The fields accept the asterisk (*) wildcard character.
- If you leave any field on this tab blank, the blank field is treated as a wild card. The failed event manager will search in all components, modules, or methods.
- You can search on a single destination criteria or on multiple criteria. Searching on two or more of the destination criteria provides a more refined list of failed events.

To search for failed events by destination, use the following procedure.

Procedure

1. Ensure the administrative console is running, and then click **Integration Applications → Failed Event Manager** to enter the failed event manager.
2. From the main failed event manager page, click **Search by destination**.

   The Search page opens with the **By Destination** tab selected.
3. Specify the search criteria you want to use. You can use any combination of the following fields to customize your search:
   - **Destination module**: use this field to specify the failed event's destination module.
   - **Destination component**: use this field to specify the failed event's destination component.
   - **Destination method**: use this field to specify the failed event's destination method.
4. Click **OK** to begin the search.

Results

The Search Results page opens and displays a list of all failed events that were destined for the specified module, component, or method.

Searching for failed events by source

Use the **By Source** tab on the Search page to find only those failed events that originated from a specific source module, component, or both. The failed event manager determines the source based on the point of failure, regardless of the type of interaction.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

When performing a search, note the following:

- The values for the fields are case sensitive.
- The fields accept the asterisk (*) wildcard character.
- If you leave either field on this tab blank, the blank field is treated as a wild card. The failed event manager will search in all components or modules.
- To get the most refined list of failed events, use both the **Source module** and **Source component** fields.

To search for failed events by source, use the following procedure.

Procedure

1. Ensure the administrative console is running, and then click **Integration Applications → Failed Event Manager** to enter the failed event manager.
2. From the main failed event manager page, click **Search by source**.

   The Search page opens with the **By Source** tab selected.
3. Specify the search criteria. You can use one or both of the following fields:
   - **Source module**: use this field to specify the module that the failed event originated from.

- **Source component**: use this field to specify the component that the failed event originated from.

4. Click **OK** to begin the search.

Results

The Search Results page opens and displays a list of all failed events that originated from the specified module, component, or both.

Searching for failed events by date

Use the **By Date** tab on the Search page to find only those events that failed during a specific time period.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

When performing a search by date, note the following:

- The format for the date and time are locale-specific. An example of the appropriate format is provided with each field.

  Note: The values you supply must match the required format exactly. If you provide an incorrectly formatted value, the failed event manager displays a warning and substitutes the default value for that field.
- The time is always local to the server. It is not updated to reflect the local time of individual machines running the administrative console.
- You must specify a value for both fields on this tab.

To search for failed events by date, use the following procedure.

Procedure

1. Ensure the administrative console is running, and then click **Integration Applications** → **Failed Event Manager** to enter the failed event manager.
2. From the main failed event manager page, click **Search by date**.
3. Use the **From Date** field to specify the starting date and time. Because the required format for the value varies by locale, the failed event manager provides a locale-appropriate example above this field. Ensure the value you enter is formatted in the same manner as the example provided. (For instance, the required format for the en_US locale is *MM/DD/YY HH:MM Meridiem*; therefore, a correctly formatted value for this field looks like `11/10/05 4:30 PM`.)
4. Use the **To Date** field to specify the ending date and time. Because the required format for the value varies by locale, the failed event manager provides a locale-appropriate example above this field. Ensure the value you enter is formatted in the same manner as the example provided. (For instance, the required format for the en_US locale is *MM/DD/YY HH:MM Meridiem*; therefore, a correctly formatted value for this field looks like `11/17/05 4:30 PM`.)
5. Click **OK** to begin the search.

   The Search Results page opens and displays a list of all failed events that originated during the specified time period.

Searching for failed events by business object type

Use the **By Type** tab of the Search page to find only those failed events that are associated with a specific business object.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

To search for failed events by business object type, use the following procedure.

Procedure

1. Ensure the administrative console is running, and then click **Integration Applications → Failed Event Manager** to enter the failed event manager.
2. From the main failed event manager page, click **Search by business object type**.

   The Search page opens with the **By Type** tab selected.
3. Specify the business object type you want to search against by using one of the following items:
   - **Select the business object type**: use this drop-down menu to select the type of business object associated with the failed events. This menu contains a list of all business object types found in the failed events on the server.
   - **Other business object type**: use this field to specify the type of business object associated with the failed events. The field accepts the asterisk (*) wildcard character. All values are case sensitive.
4. Click **OK** to begin the search.

Results

The Search Results page opens and displays a list of all failed events that are associated with the specified business object type.

Searching for failed events by exception

Use the **By Exception** tab of the Search page to find only those failed events that are associated with a specific exception. You can specify part or all of the exception text.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

To search for failed events by exception type, use the following procedure.

Procedure

1. Ensure the administrative console is running, and then click **Integration Applications → Failed Event Manager** to enter the failed event manager.
2. From the main failed event manager page, click **Search by exception text**.

   The Search page opens with the **By exception** tab selected.
3. In the **Exception text** field, type the text associated with the exception you want to search against.

You can specify all or part of the exception text, as well as the asterisk (*) wildcard character to make the search easier. The values in this field are case sensitive.

Note: If you leave the **Exception text** field blank, it is treated as a wild card; all failed events are returned.

4. Click **OK** to begin the search.

Results

The Search Results page opens and displays a list of all failed events that are associated with the specified exception text.

Performing an advanced search for failed events

Use the **Advanced** tab of the Search page to perform a more refined search for failed events by using a combination of the criteria found on the other search tabs.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

Note the following:

- Unless otherwise noted below, all fields accept the asterisk (*) wildcard character.
- Leaving a field blank causes it to be treated as a wild card.
- The advanced search is not optimized; executing an advanced search on a large set of failed events can reduce performance.

To perform an advanced search, use the following procedure.

Procedure

1. Ensure the administrative console is running, and then click **Integration Applications → Failed Event Manager** to enter the failed event manager.
2. From the main failed event manager page, click **Advanced search**.

   The Search page opens with the **Advanced** tab selected.
3. Specify the search criteria you want to use. You can use any combination of the following fields to customize your search:
   - **Destination module**: use this field to specify the failed event's destination module.
   - **Destination component**: use this field to specify the failed event's destination component.
   - **Session ID**: use this field to specify the session in which the event executed. This field does not accept the asterisk (*) wildcard character.
   - **Destination method**: use this field to specify the failed event's destination method.
   - **Source module**: use this field to specify the module that the failed event originated from.
   - **Source component**: use this field to specify the component that the failed event originated from.

- **From Date**: use this field to specify the starting date and time if you want to search within a specific time period. This field does not accept the asterisk (*) wildcard character.
- **To Date**: use this field to specify the ending date and time if you want to search within a specific time period. This field does not accept the asterisk (*) wildcard character.
- **Business object type**: use this field to specify the type of business object associated with the failed events.
- **Exception text**: use this field to specify the text associated with the exception you want to search against.

4. Click **OK** to begin the search.

Results

The Search Results page opens and displays a list of all failed events that meet the specified criteria.

Working with data in failed events

Each failed event has data associated with it; often, that data can be edited before an event is resubmitted. There are two basic types of data for a failed event: data about the event, and business data.

Data about the failed event

Each failed event has the following data associated with it:

- The unique message ID and session ID for the event
- The service invocation type between SCA components
- The names of the module and component from which the event originated (the source). The failed event manager determines the source of an event based on the location where the invocation failed.
- The names of the destination module, component and method for the event. The failed event manager determines the event's destination based on the location where the invocation failed.
- The time the event failed
- The exception thrown when the event failed

This data cannot be edited. In addition, failed events can have associated trace and expiration data, both of which can be edited.

Business data

Events typically include business data. Business data can be encapsulated in a business object, or it can be simple data that is not part of a business object. Business data is edited with the business data editor available in the failed event manager.

Browsing data in failed events

Use the failed event manager to view failed event data and any business data associated with the event.

Before you begin

You must be logged as administrator or operator to perform this task.

About this task

Each failed event has two types of data associated with it:

- Failed event data: information about the failed event itself, including the source and destination for the event, the time it failed, the exception it failed with, its message and session IDs, and its trace and expiration settings.
- Business data: information contained in the event. The business data can be encapsulated in a business object, or it can be simple data that is not part of a business object.

To browse failed event data, use the following procedure.

Procedure

1. Ensure that the failed event manager is open and that you have retrieved a list of the failed events on your system.
2. From the Search Results page of the failed event manager, click the ID (found in the Message ID column) of the failed event whose data you want to browse.

   The Failed Event Details page opens and displays all of the information about the event.
3. If you want to browse the business data associated with the failed event, click **Edit business data**.

   The Business Data Editor collection page opens, displaying the business data associated with the failed event. Each parameter name in the hierarchy is a link. If the parameter is a simple data type, clicking its name will open up a form so you can edit the parameter's value. If the parameter is a complex data type, clicking its name will expand the hierarchy further.

Editing trace or expiration data in a failed event

The Failed Event Details page enables you to set or modify values for the trace control and expiration date associated with a failed event.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

Important: Any edits you make to the trace or expiration data are only saved locally until you resubmit the event. If you perform any other action before resubmitting the event, all edits are lost.

Failed events can be resubmitted with trace to help you monitor the event processing. Tracing can be set for a service or a component, and it can be sent to a log or to the Common Event Infrastructure (CEI) server. When you view the failed event data on the Failed Event Details page, the default trace value `SCA.LOG.INFO;COMP.LOG.INFO` is shown for the event. If you resubmit the event with this default setting, no trace occurs when the session calls an SCA service or executes a component.

Some failed events also have an expiration. If a user has specified an expiration with the asynchronous call that sends the event, that data persists even if the event fails, and the expiration time appears in the **Resubmit Expiration Time** field on the Failed Event Details page. Expired failed events cannot be resubmitted

successfully. To prevent a second failure, you can edit the expiration date for the event to ensure that it is not expired when it is resubmitted.

To edit trace or expiration data in a failed event, use the following procedure.

Procedure

1. Ensure that the failed event manager is open and that you have retrieved a list of the failed events on your system.
2. From the failed event manager's Search Results page, click the ID (found in the Message ID column) of the failed event whose data you want to edit.

   The Failed Event Details page opens.
3. If the event has an expiration date that causes it to expire before it is resubmitted, edit the expiration in the **Resubmit expiration time** field.

   The expiration time shown is local to the server. The value for this field must be formatted according to your specified locale. An example of the correct format for your locale is provided above the field.
4. If you want to enable tracing for the failed event, specify a new value in the **Trace Control** field. For detailed information about trace values, see the Monitoring topics in the WebSphere Process Server Information Center.
5. Do one of the following:
   - If the edited data is correct and you want to resubmit the event, click **Resubmit** to make the changes at a server level.
   - If you want to remove the changes you made, click **Undo local changes**.

   The edited failed event is resubmitted for processing and is removed from the failed event manager.

   Related tasks

   "Finding failed events" on page 55
   Failed events are stored in the failed event database and are retrieved through the search functionality of the failed event manager. You can search for all failed events on the all the servers within the cell, or for a specific subset of events.

Editing business data in a failed event

Business data can be encapsulated into a business object, or it can be simple data that is not part of a business object. A failed event can have both simple data and a business object associated with it. Use the business data editor to edit the business data associated with a failed event before you resubmit it.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

For each failed event, the editor displays the associated business data in a hierarchical format; the navigation tree at the top of the table is updated as you navigate through the parameters to give you a clear picture of where you are in the hierarchy.

You can edit only simple data types (for example, String, Long, Integer, Date, Boolean). If a data type is complex (for example, an array or a business object), you must navigate through the business data hierarchy until you reach the simple data

types that make up the array or business object. Complex data is denoted by an ellipsis (...) in the Parameter Value column.

Important: Any edits you make to business data are saved locally. Changes are not made to the corresponding business data in the server until you resubmit the failed event.

To edit business data associated with a failed event, use the following procedure.

Procedure

1. Ensure that the failed event manager is open and that you have retrieved a list of the failed events on your system.
2. From the failed event manager's Search Results page, click the ID (found in the Message ID column) of the failed event whose data you want to edit.

   The Failed Event Details page opens.
3. From the Failed Event Details page, click **Edit business data** to access the Business Data Editor collection page.

   This page displays a hierarchical view of all of the data associated with the failed event.
4. Navigate through the business data hierarchy by clicking on the name of each parameter (these appear as links in the Parameter Name column). When you have located the parameter whose value you want to edit, click its name.

   If the parameter has an editable value, the Business Data Editor page opens.
5. In the **Parameter value** field, specify the new value for the parameter.
6. Click **OK**.

   The change is saved locally and you are returned to the Business Data Editor collection page.
7. If you want to remove the changes you made, click **Undo local business data changes**.

   All of the edits are removed and the business data is returned to its original state.
8. If the edited business data is correct, click **Resubmit** to make the changes at a server level.

   The edited failed event is resubmitted for processing and is removed from the failed event manager.

Resubmitting failed events

If you want to try to send an event again, you must resubmit it from the failed event manager. You can resubmit an event without changes, or you can edit the business data parameters before resubmitting it.

When a failed event is resubmitted, the processing resumes only for the failed branch, not for the entire event.

Tracing is available for resubmitted events to help monitor the event's processing. Tracing can be set for a service or a component, and its output can be sent to a log or to the Common Event Infrastructure (CEI) server.

You can also use the event's unique message ID to track its success or failure. If a resubmitted event fails again, it is returned to the failed event manager with its original message ID and an updated failure time.

Resubmitting an unchanged failed event

You can resubmit one or more unchanged failed events to be processed again. Processing resumes only for the failed branch, not for the entire event.

About this task

You must be logged in as administrator or operator to perform this task.

Procedure

1. Ensure that the failed event manager is open and that you have retrieved a list of the failed events on your system.
2. From the Search Results page, select the check box next to each failed event you want to resubmit.
3. Click **Resubmit**.

Results

Each selected event is resubmitted for processing and is removed from the failed event manager.

Resubmitting a failed event with trace

You can monitor the resubmission of a failed event to determine whether it now succeeds. The failed event manager provides optional tracing for all failed events.

About this task

Tracing can be set for a service or a component, and it can be output to a log or to the Common Event Infrastructure (CEI) server. For detailed information about setting and viewing trace, see the Monitoring topics in the information center.

You must be logged in as administrator or operator to perform this task.

Procedure

1. Ensure that the failed event manager is open and that you have retrieved a list of the failed events on your system.
2. From the Search Results page, select the check box next to each failed event you want to resubmit.
3. Click **Resubmit with trace**.
4. From the Resubmit with Trace page, specify the level of trace you want to use in the **Trace control** field.

   By default, the value is SCA.LOG.INFO;COMP.LOG.INFO. With this setting, no trace occurs when the session calls an SCA service or executes a component.
5. Click **OK** to resubmit the failed event and return to the Search Results page.

What to do next

To view the trace log for a resubmitted event, open the corresponding component logger or use the CEI log viewer.

Finding Common Base Events related to a failed event

A failed event can be related to one or more Common Base Events. The failed event manager provides a link to view related Common Base Events in the Common Base Event Browser.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

Examining related Common Base Events can give you additional information about how or why the original event failed. The failed event and any related Common Base Events are linked by the same session ID.

To find and view related Common Base Events, use the following procedure.

Procedure

1. From within the administrative console, use the failed event manager to locate the failed event you want to investigate. See "Finding failed events" on page 55 for instructions on how to search for failed events.
2. From the Failed Event Details page for that event, click **Browse Related Common Base Events**.

Results

The Common Base Event Browser opens in a new browser window and lists any Common Base Events related to the original failed event.

Finding business process instances related to a failed event

If a failed event is generated from a business process, the failed event manager provides a link to view that business process instance in Business Process Choreographer Explorer.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

Examining the business process instance that generated the failed event can give you additional information about how or why the event failed. The business process instance and the failed event are linked by a common session ID.

Note: Not all failed events are generated from a business process instance.

To find and examine a business process instance related to a failed event, use the following procedure.

Procedure

1. From within the administrative console, use the failed event manager to locate the failed event you want to investigate. See "Finding failed events" on page 55 for instructions on how to search for failed events.
2. From the Failed Event Details page for that event, click **Open calling process in Business Process Choreographer Explorer**.

Results

The Business Process Choreographer Explorer opens in a new browser window and displays information about the related process instance.

Deleting failed events

If you do not want to resubmit a failed event, or if you have failed events that have expired, use the failed event manager to delete them from the server. The failed event manager provides three options for deleting failed events.

Before you begin

You must be logged in as administrator or operator to perform this task.

About this task

To delete one or more failed events, use the following procedure.

Procedure

1. Ensure that the failed event manager is open and that you have retrieved a list of the failed events on your system.
2. From the failed event manager's Search Results page, do one of the following:
   - If you want to delete one or more specific failed events, select the check box next to each event and then click **Delete**.
   - If you want to delete only those failed events that have expired, click **Delete expired events**. Note that this deletes only the expired events in the current set of search results.
   - If you want to delete all failed events on the server, click **Clear all on server**.

Troubleshooting the failed event manager

This topic discusses problems that you can encounter while using the failed event manager.

Note: This topic does not discuss how to use the failed event manager to find, modify, resubmit, or delete failed events on the system. For information about managing failed events, see *Managing WebSphere Process Server failed events* in the information center.

Select the problem you are experiencing from the table below:

Problem	Refer to the following
I am having trouble with reduced performance during an advanced search	"Advanced search feature is not optimized" on page 37
I am having trouble entering values in the Search page's **By Date** tab	"Values in the By Date tab automatically change to default if entered incorrectly" on page 37
I am having trouble deleting expired events	"Using the Delete Expired Events function appears to suspend the failed event manager" on page 38
I am having trouble with failed events not being created	"Failed events are not being created" on page 38

Advanced search feature is not optimized

The failed event manager's advanced search feature is not optimized. Therefore, you may experience reduced performance when using the Advanced search tab with a large set of failed events.

Values in the By Date tab automatically change to default if entered incorrectly

The Search page's **By Date** tab contains two fields: **From Date** and **To Date**. Both fields are required. The values are locale-dependendent, and they must be formatted exactly as shown in the example above the field. Any inconsistency in the value's format (for example, including four digits in the year instead of 2, or omitting the time) will cause the failed event manager to issue the following warning and substitute a default value in the field:

```
CWMAN0017E: The date entered could not be parsed correctly:
your_incorrectly_formatted_date. Date: default_date is being used.
```

The default value of the **From Date** field is defined as `January 1, 1970, 00:00:00 GMT`.

Important: The actual default value shown in your failed event manager implementation will vary depending on your locale and time zone. For example, the From Date field defaults to `12/31/69 7:00 PM` for a machine with an en_US locale in the Eastern Standard Time (EST) time zone.
The default value for the **To Date** field is always the current date and time, formatted for your locale and time zone.

To avoid this problem, always enter your dates and times carefully, following the example provided above each field.

Using the Delete Expired Events function appears to suspend the failed event manager

If you use the Delete Expired Events button in situations where there are many failed events in the current search results, or where those events contain a large amount of business data, the failed event manager can appear to be suspended indefinitely.

In this situation, the failed event manager is not actually suspended: it is working through the large data set, and will refresh the results set as soon as the command completes.

Failed events are not being created

If the Recovery subsystem is not creating failed events, go through the following checklist of potential causes:

- Ensure that the wpsFEMgr application is running. If necessary, restart it.
- Ensure that the failed event manager's database has been created, and that the connection has been tested.
- Ensure that the necessary failed event destination has been created on the SCA system bus. There should be one failed event destination for each deployment target.

- Ensure that the Quality of Service (QoS) **Reliability** qualifier has been set to `Assured` for any Service Component Architecture (SCA) implementation, interface, or partner reference that participates in events you want the Recovery service to handle.

Chapter 11. Searching knowledge bases

You can often find solutions to problems by searching IBM knowledge bases. Optimize your results by using available resources, support tools, and search methods.

About this task

To search for solutions to your problems in IBM knowledge bases, perform the following steps.

Procedure

1. Search the information center.

   IBM provides extensive documentation in the form of online information centers. An information center can be installed on your local machine or on a local intranet. An information center can also be viewed on the IBM Web site. You can use the powerful search function of the information center to query conceptual and reference information and detailed instructions for completing tasks.
2. Search available technical resources. In addition to this information center, the following technical resources are available to help you answer questions and resolve problems:
   - WebSphere Process Server technotes
   - WebSphere Process Server Authorized Program Analysis Reports (APARs)
   - WebSphere Process Server support Web site
   - WebSphere Redbooks® Domain
   - IBM Education Assistant
   - WebSphere Process Server forums and newsgroups
3. Search with IBM Support Assistant. IBM Support Assistant (ISA) is a free software serviceability workbench that helps you resolve questions and problems with IBM software products. For more information, see the IBM Support Assistant.

   To search multiple Internet resources for your product, open the ISA and click **Search**. From this page, you can search a variety of resources including:
   - IBM Software Support Documents
   - IBM developerWorks®
   - IBM newsgroups and forums
   - Google
   - IBM product information centers

   Note: These free newsgroups and forums do not offer any formal IBM product support. They are intended for user-to-user communication. IBM will not be actively participating in these discussions. However, IBM does review these newsgroups periodically to maintain a free flow of accurate information.
4. Search with the IBM Software Support Toolbar. IBM Software Support Toolbar is a browser plug-in that provides you with a mechanism to easily search IBM support sites. You can download the toolbar at www.ibm.com/software/support/toolbar/.

What to do next

Tip:

The following resources describe how to optimize your search results:

- Searching the IBM Support Web site
- Using the Google search engine
- IBM Software Support RSS feeds
- My Support e-mail updates

Chapter 12. IBM Support Assistant

The IBM Support Assistant is a tool that helps you use various IBM Support resources.

Note: The IBM Support Assistant is supported on Microsoft® Windows® and Linux® systems.

The IBM Support Assistant offers four components to help you with software questions:

- a Search component, which helps you access pertinent Support information in multiple locations.
- a Product Information component, which helps you find the right IBM site for your product questions.
- a Tools component, which provides specialized analysis tools to investigate product problems.
- a Service component, which helps you submit an enhanced problem report that includes key system data to IBM.

Using the IBM Support Assistant with WebSphere Process Server, requires installing IBM Support Assistant and then installing plug-ins for WebSphere Process Server. The plug-ins for WebSphere Process Server include an automated way to gather information about a problem and send it to IBM, and tools that help you set trace levels.

For more information and to install the latest version of IBM Support Assistant, see the IBM Support Assistant Web page.

IBM Support Assistant also is included on the *WebSphere Application Server Network Deployment Supplements V6.1* disk that is included with WebSphere Process Server and can be installed from the WebSphere Process Server launchpad.

After the IBM Support Assistant is installed, you can start it with the **Start** menu option on Windows operating systems or with the startisa.sh shell script on all other platforms. On Windows operating systems, the IBM Support Assistant opens in its own window. On all other platforms, it opens in a Web browser.

When you have IBM Support Assistant open, you can view available plug-ins for WebSphere Process Server by clicking **Updater**, clicking **New Plug-ins** and then expanding **WebSphere**. When you select the check box for the WebSphere Process Server plug-in, and click **Install**, the download page opens.

To learn more about how to use the IBM Support Assistant, click **Help** in the IBM Support Assistant window.

Related concepts

Chapter 14, “Contacting IBM Software Support,” on page 77
IBM Software Support provides assistance with product defects.

Related tasks

Chapter 13, “Getting fixes,” on page 75
A product fix might be available to resolve your problem.

Chapter 13. Getting fixes

A product fix might be available to resolve your problem.

About this task

To get product fixes, perform the following steps.

Procedure

1. Obtain the tools required to get the fix.
2. Determine which fix you need. Check the list of WebSphere Process Server recommended fixes to confirm that your software is at the latest maintenance level. Check the list of problems fixed in the IBM WebSphere Process Server fix readme documentation that is available for each listed fix pack and refresh pack to see if IBM has already published an individual fix to resolve your problem. To determine what fixes are available using IBM Support Assistant, run a query on `fix` from the search page.

   Individual fixes are published as often as necessary to resolve defects in WebSphere Process Server. In addition, two kinds of cumulative collections of fixes, called fix packs and refresh packs, are published periodically for WebSphere Process Server, in order to bring users up to the latest maintenance level. You should install these update packages as early as possible in order to prevent problems.

   Note: Fixes specific to the underlying WebSphere Application Server product may also be obtained from the WebSphere Application Server Support Site or from the WebSphere Application Server Support team. Fixes for individual APARs for WebSphere Application Server generally can be applied without affecting WebSphere Process Server. However, consult with the software requirements page before updating WebSphere Application Server with cumulative collections of fixes (fix packs). First check to see that the cumulative fix has passed certification, or contact the Support team for verification.
3. Download the fix. Open the download document and follow the link in the **Download package** section.
4. Apply the fix. Follow the instructions in the **Installation Instructions** section of the download document. For more information, see Installing fix packs and refresh packs with the Update Installer in the Installing WebSphere Process Server documentation.
5. Optional: To receive weekly notification of fixes and updates, subscribe to My Support e-mail updates.

Related concepts

Chapter 12, “IBM Support Assistant,” on page 73
The IBM Support Assistant is a tool that helps you use various IBM Support resources.

Chapter 14. Contacting IBM Software Support

IBM Software Support provides assistance with product defects.

Before you begin

To take advantage of unique Support features, see the WebSphere Process Server support page. The Support Page contains the latest information on fixes and downloads, educational resources, and commonly encountered problems and their solutions.

Before contacting IBM Software Support, your company must have an active IBM software maintenance contract, and you must be authorized to submit problems to IBM. The type of software maintenance contract that you need depends on the type of product you have. For information about the types of maintenance contracts available, see "Enhanced Support" in the *Software Support Handbook* at techsupport.services.ibm.com/guides/services.html.

To contact IBM Software Support with a problem, perform the following steps.

Procedure

1. Define the problem, gather background information, and determine the severity of the problem. For help, see the "Contacting IBM" in the *Software Support Handbook* at techsupport.services.ibm.com/guides/beforecontacting.html.
2. Gather diagnostic information. When explaining a problem to IBM, be as specific as possible. Include all relevant background information so that IBM Software Support specialists can help you solve the problem efficiently. For information that IBM Support needs in order to help you solve a problem, see the WebSphere Process Server MustGather technote.

   Tip: You can use the WebSphere Process Server plug-in for the IBM Support Assistant to capture data and send it to IBM.

   Note: If you are able to determine that the problem is purely with underlying WebSphere Application Server functionality, consider requesting assistance specifically from the WebSphere Application Server Support team rather than the WebSphere Process Server team. For information that IBM Support needs in order to help you solve a WebSphere Application Server problem, see the WebSphere Application Server MustGather Technote.
3. Submit your problem to IBM Software Support in one of the following ways:
   - Using IBM Support Assistant: See IBM Support Assistant.
   - Online: Open a service request on the IBM Software Support site using the Electronic Service Request (ESR) tool.
   - By telephone: For the telephone number to call in your country or region, go to the contacts page of the IBM Software Support Handbook on the Web and click the name of your geographic region.

What to do next

If the problem you submit is for a software defect or for missing or inaccurate documentation, IBM Software Support creates an Authorized Program Analysis

Report (APAR). The APAR describes the problem in detail. Whenever possible, IBM Software Support provides a workaround that you can implement until the APAR is resolved and a fix is delivered. IBM publishes resolved APARs on the Software Support Web site daily, so that other users who experience the same problem can benefit from the same resolution.

Related concepts

Chapter 12, “IBM Support Assistant,” on page 73
The IBM Support Assistant is a tool that helps you use various IBM Support resources.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
577 Airport Blvd., Suite 800
Burlingame, CA 94010
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows: (c) (your company name) (year). Portions of

this code are derived from IBM Corp. Sample Programs. (c) Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

IBM, the IBM logo, developerWorks, i5/OS, Redbooks, WebSphere, z/OS, and zSeries are registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Adobe is a registered trademark of Adobe Systems Incorporated in the United States, and/or other countries.

Microsoft and Windows are registered trademark of Microsoft Corporation in the United States, other countries, or both.

Intel is a registered trademark of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

This product includes software developed by the Eclipse Project (http://www.eclipse.org).

IBM WebSphere Process Server for z/OS, Version 6.1.0

Printed in USA