Redbooks

Application Deployment in WebSphere z/OS V5.0.1

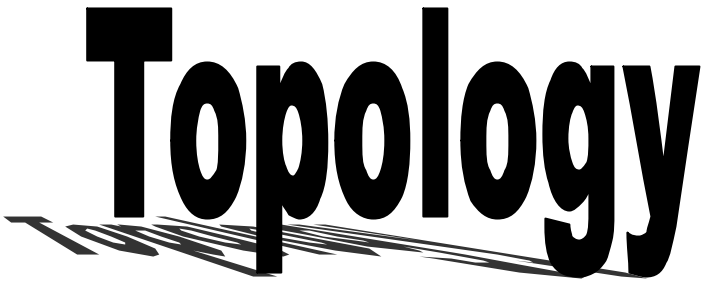
Sabine Holl	**sabine_holl@at.ibm.com**
Alex Louwe Kooijmans	**nl53347@nl.ibm.com**
Kevin J. Senior	**kev_senior@it.ibm.com**

© 2003 IBM Corporation

Agenda

- **Application Deployment for the WebSphere system administrator . . .**
- What is your complete target topology?
  - There will be multiple test environments as well as Production
- Developer Tools
  - WebSphere Studio Application Developer (WSAD)
  - WebSphere Studio Enterprise Developer (WSED)
- WebSphere z/OS: Application Deployment
  - Java Management EXtension (JMX) APIs
  - Application Assembly Tool (AAT)
  - WebSphere Administrative Console
  - wsadmin scripts and Ant
  - Hot Deployment and Dynamic Reload

© 2002 IBM Corporation



WebSphere z/OS V4 vs V5
Terminology
Planning a complete test and production infrastructure

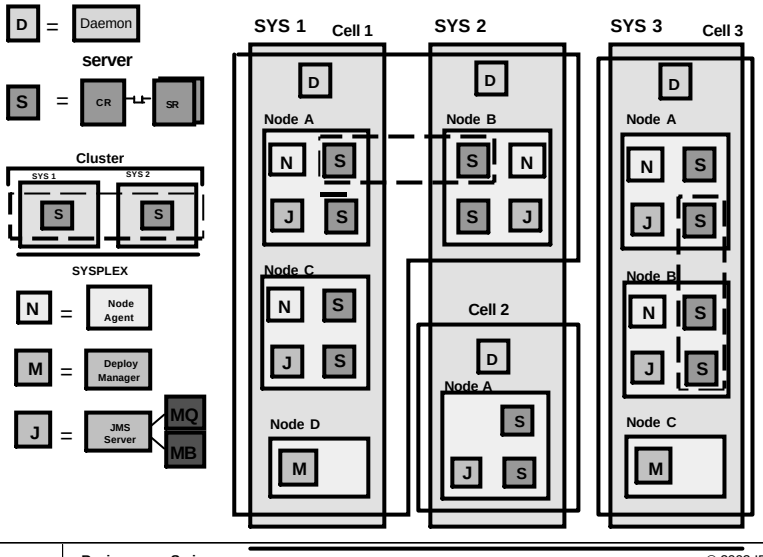
© 2002 IBM Corporation

Summary of Terminology

WAS V4	WAS V5
Node (approx)	Cell
Server	Cluster
Server Instance	Cluster member
Systems Management Server (approx)	Node Agent
Control Region	Control Region
Server Region	Server Region

© 2002 IBM Corporation

WebSphere z/OS V5 Topology



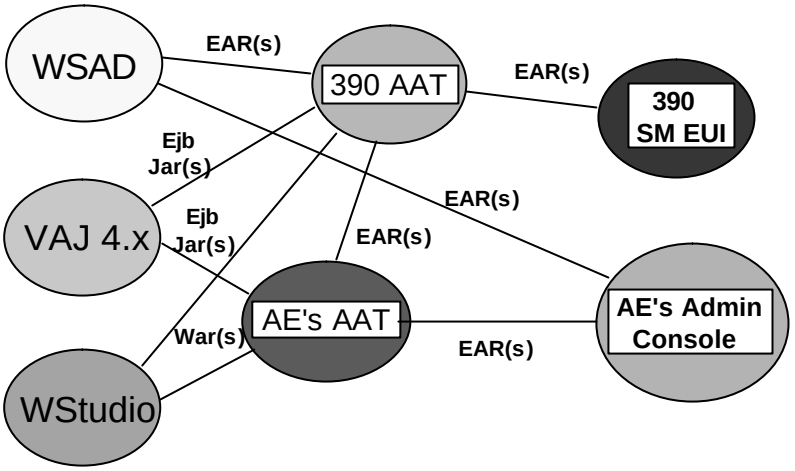
© 2002 IBM Corporation

WebSphere z/OS V5.0 compared with V4.01

- Complex Install/Bootstrap
- Replicated Server Topology
- Centralized Admin Repository
- "Fat" Admin Client
- SMAPI/REXX Scripting
- z/OS Proprietary SM Model
- Greatly simplified Install/Bootstrap
- Clustered Server Topology
- Distributed Admin Repository
- "Thin" Browser Admin Client
- wsadmin/JACL scripting
- WAS Family SM Model

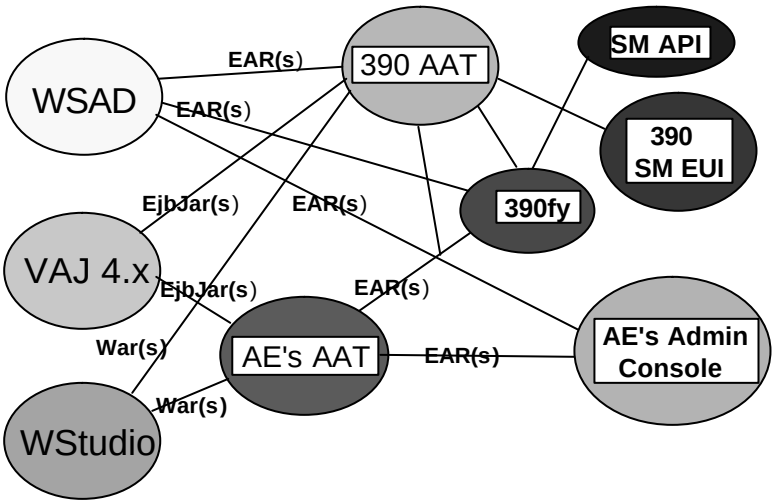
© 2002 IBM Corporation

Deployment in WebSphere before 4.0.4



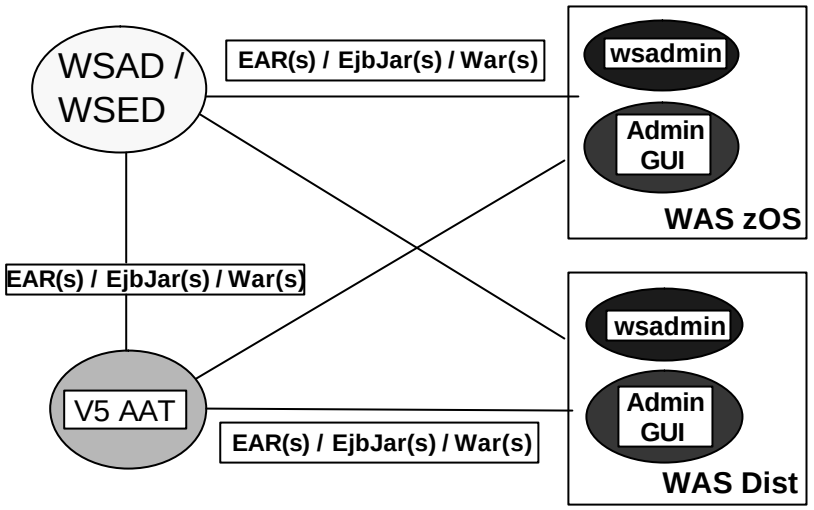
© 2002 IBM Corporation

Deployment in WebSphere V4.0.4



© 2002 IBM Corporation

Deployment with WebSphere V5



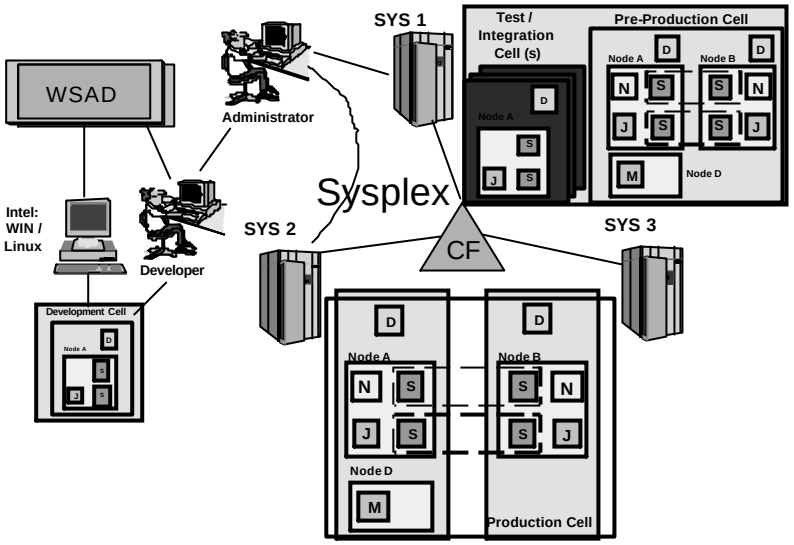
© 2002 IBM Corporation

Administrative Environments

- **Development Environment**
  - Developer is the administrator
  - Very tight integration with development tooling
  - Focus on one part of overall solution
  - Requires quick and easy admin. changes
  - WebSphere Base Application Server package
- **Test and Integration Environment**
  - Aggregates several development environments
  - Focus on total solution domain
- **Pre-Production**
  - Topology like production but smaller (re-sizable)
  - Test install / backout of changes
  - Test production security
  - Performance Test environment
- **Production Environment**
  - Greater scaling requirements
  - Administrative robustness and failure tolerance
  - Managed and restricted administration activities
  - Centralized administration
  - WebSphere Network Deployment package

© 2002 IBM Corporation

WebSphere Application Deployment Stages



© 2002 IBM Corporation

Application Deployment 'phases'

- Phase 1: Developer to Test/Integration
  - Install EAR file via GUI
- Phase 1x: Test / Integration to other clones environments such as:
  - Financial testing with closing controlled data
  - Education system with data which is reset after course
    - Use scripts
- Phase 2: Test / Integration to Pre-Production (and backout)
  - Scripts
- Phase 3: Pre-production to Production (and backout)
  - Scripts created / tested in Pre-production
- Phase E: Developer to Pre-production (Emergency fix path)
  - Manual process / script

© 2002 IBM Corporation

Developer Tools

WebSphere Studio Application Developer (WSAD)
WebSphere Studio Enterprise Developer (WSED)

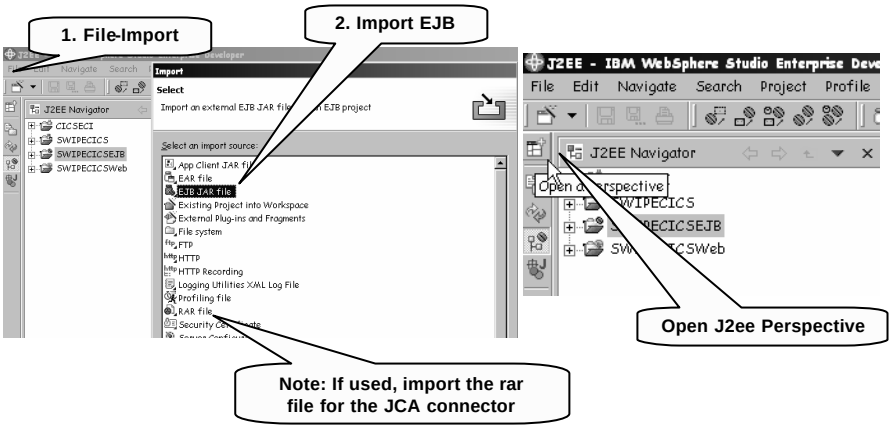
© 2002 IBM Corporation

WSAD and WSED

- Both WSAD and WSED are built on the Eclipse standard workbench so the 'look and feel' is similar.
- WSED contains support for procedural languages COBOL and PL/I in addition to Java.
- This is not a course in the use of WSAD. What we want to understand here is how the developer should prepare the deployment descriptors in the ear file so the application is ready to deploy.
- We will look at 2 aspects:
  - Resource Reference bindings for JCA adaptors
  - Security role-userid mappings

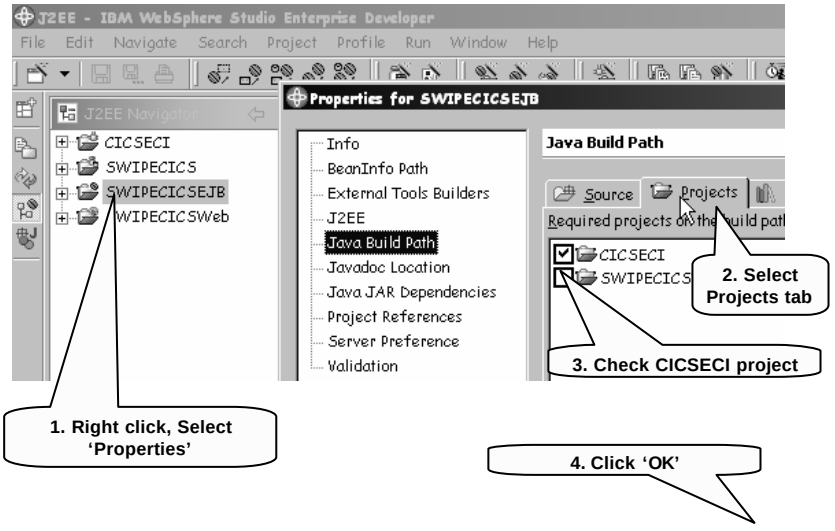
© 2002 IBM Corporation

WSAD: Importing an application and RAR file.



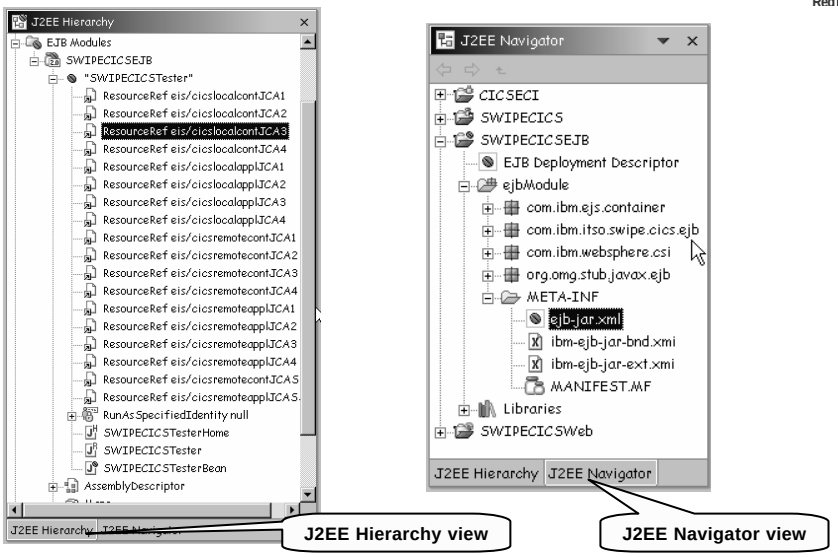
© 2002 IBM Corporation

WSAD: Add RAR to SWIPECICSEJB Build path



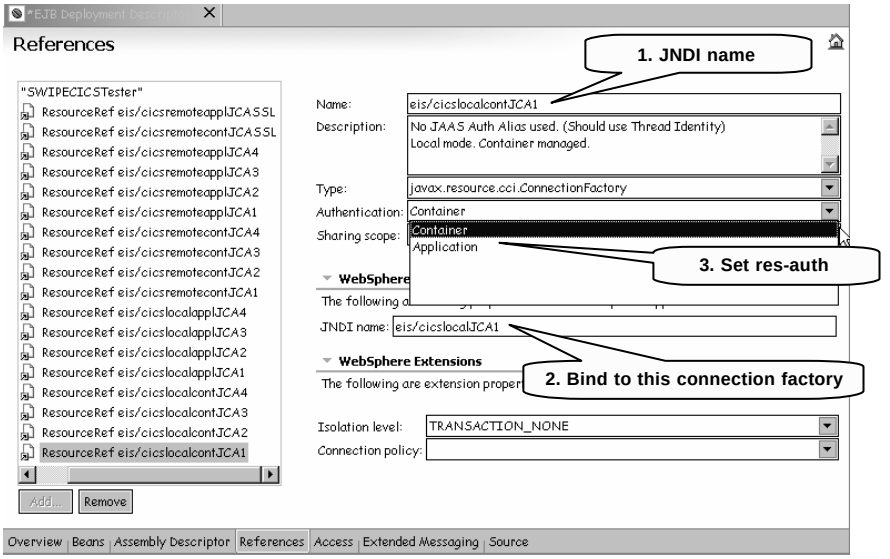
© 2002 IBM Corporation

WSAD: Resource Reference bindings (1)



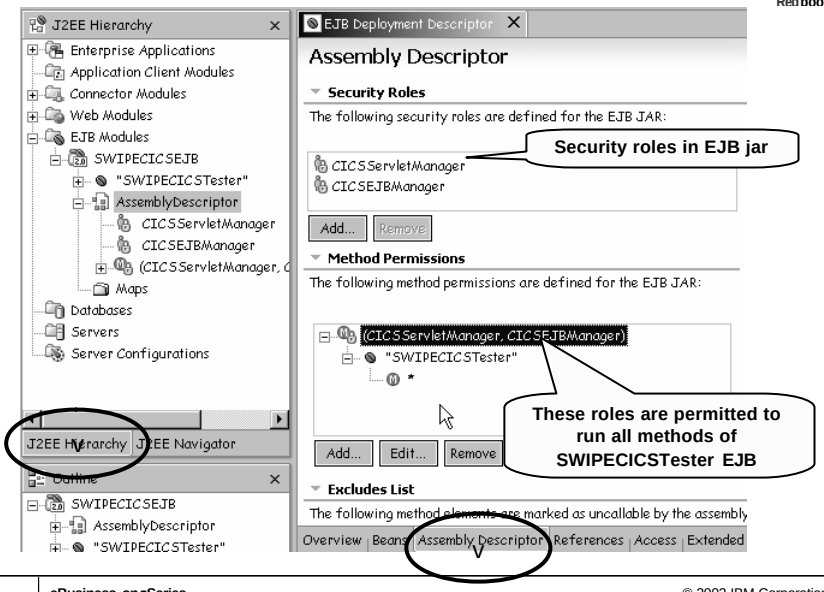
© 2002 IBM Corporation

WSAD: Resource Reference bindings (2)



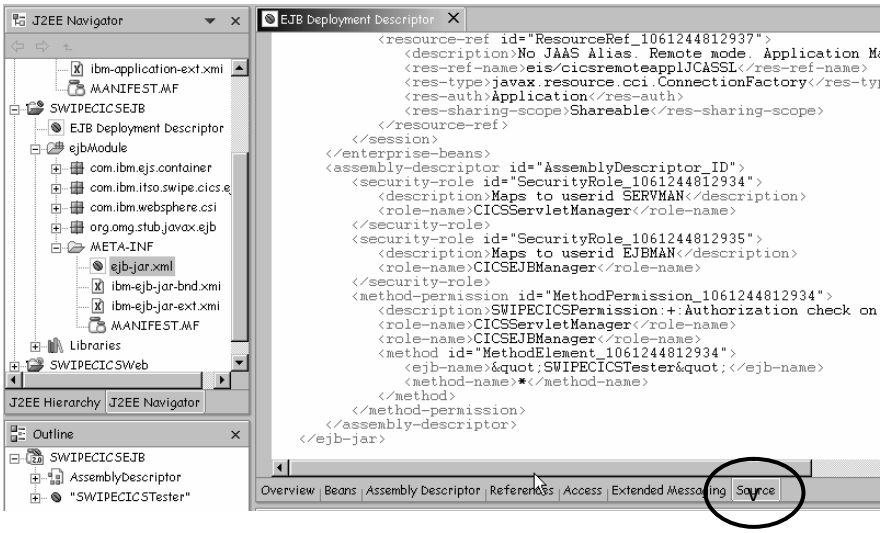
© 2002 IBM Corporation

WSAD: Security Roles



© 2002 IBM Corporation

WSAD: XML view of deployment descriptor



© 2002 IBM Corporation

WebSphere z/OS V5

Java Management Extensions (JMX)
Application Assembly Tool (AAT)
WebSphere Administrative Console
Deploying application with MDBs
wsadmin / JACL scripts and Ant
Hot deployment and Dynamic Reload
Classloaders

© 2002 IBM Corporation

WebSphere Management Facility Characteristics

- **Full Function, Single API**
  - Based on JMX core framework
  - Configuration, Operations, Events, Performance, Tuning
  - Accessible through Java APIs, scripting language, web console GUI
- **Standards Based Implementation**
  - Instrumentation - JMX, JSR77
  - Connectors - SOAP, RMI/IIOP, SNMP (later)
  - Configuration - UML/MOF, XML
- **Supports Multiple Administrators**
  - Update concurrency
  - J2EE Role-based Security
- **Customizable, Interoperable**
  - Extensible by adding JMX MBeans
  - File-based, distributed SM repository
  - Same JMX code, MicroEdition in WAS z/OS

© 2002 IBM Corporation

WebSphere JMX Components

- **WebSphere MBeans**
  - JMX ModelMBean (ManagedBean) Interface
  - JSR-077 ManagedObject Interface, Event Notification
  - extended WebSphere properties, operations, notifications
  - all MBeans defined by XML MBeanDescriptor
- **WebSphere JMX Agent**
  - MBeanServer, plus Agent Services (Monitor, Timer, Relation)
  - TMX4J (externally available on AlphaWorks)
- **JMX Connectors and Adapters**
  - XML/HTTP (SOAP) JMX Connector
  - RMI JMX Connector (uses WebSphere ORB)
  - Later ...
    - SNMP Adapter
    - Asynchronous JMS Connector

© 2002 IBM Corporation

WebSphere Admin Tools

- **Command Line Tools**
  - single node utilities to start server, perform simple functions
  - located in /bin directory
- **Admin Console**
  - Web application - runs in some application server
  - browser is the client environment (connects to app server)
- **Wsadmin Scripting program**
  - non-graphic admin client
  - 3 modes - interactive, batch, command line
- **Java Admin APIs**
  - allows custom admin programs to be written in Java
  - public classes and methods documented in the product javadoc , InfoCenter

© 2002 IBM Corporation

WebSphere z/OS V5

Application Deployment

Java Management Extensions (JMX)
Application Assembly Tool (AAT)
WebSphere Administrative Console
Deploying application with MDBs
wsadmin / JACL scripts and Ant
Hot deployment and Dynamic Reload
Classloaders

© 2002 IBM Corporation

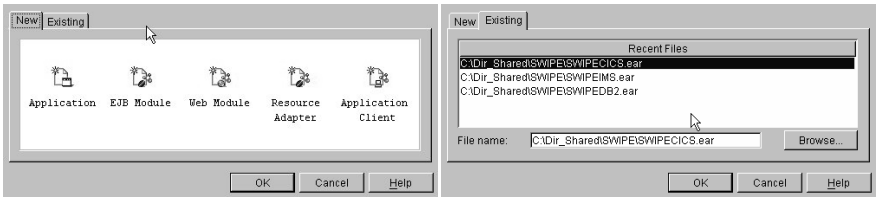
Application Assembly Tool (AAT)

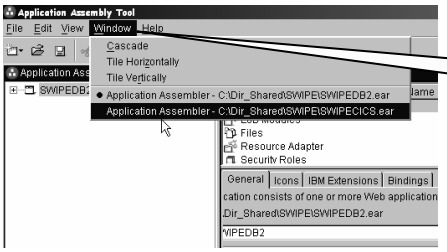
- The AAT in WebSphere z/OS V5 is now like the AAT used on other platforms.
- Part of WebSphere Client Development Kit for zOS
- Download from /<websphere_install_root>/**lib/setup.exe**

- Import the application ear file
- Each application is in a separate window
- View / Edit deployment descriptor
- Export modified ear

© 2002 IBM Corporation

AAT: Selecting your ear file

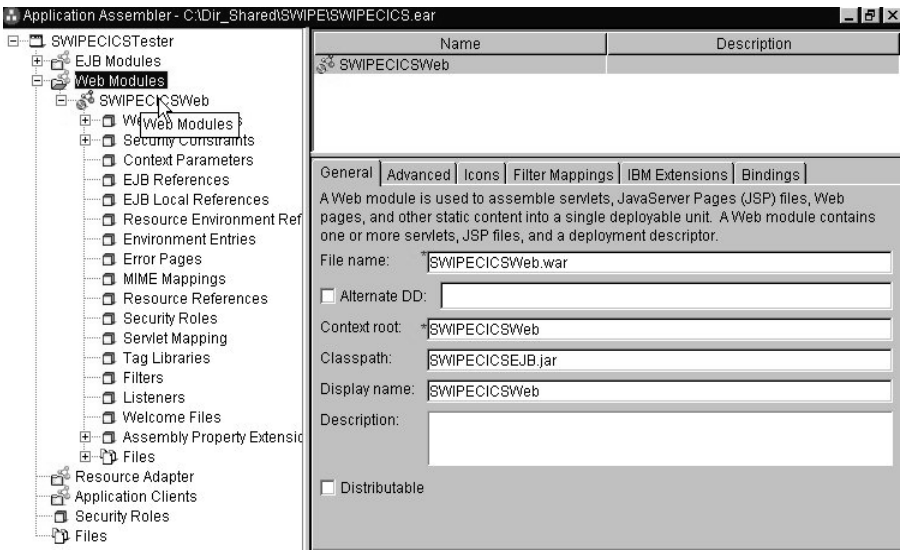




Each open application is selectable from 'Window' in the tool bar

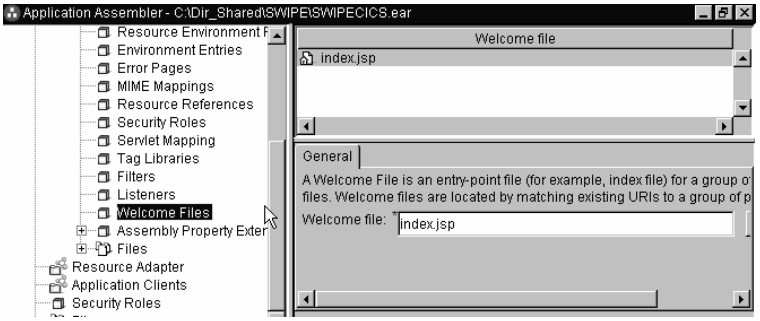
© 2002 IBM Corporation

AAT: Overview



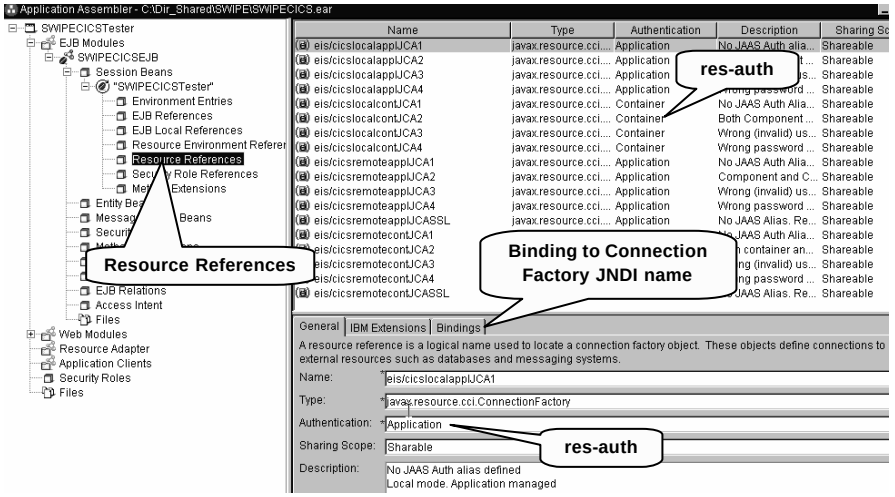
© 2002 IBM Corporation

AAT: Welcome Files



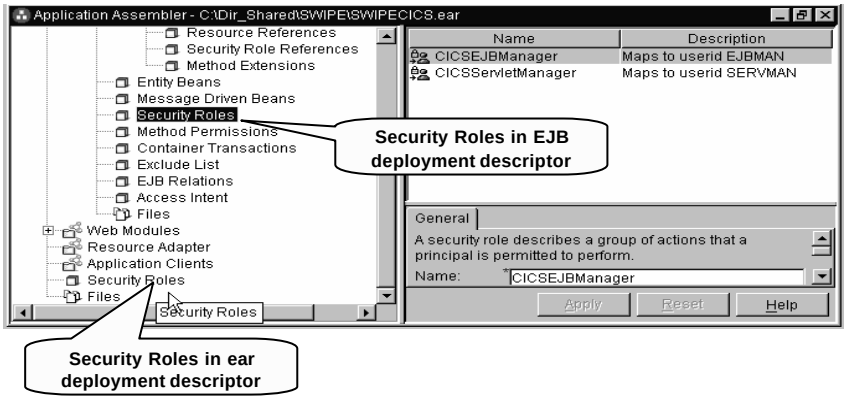
© 2002 IBM Corporation

AAT: Resource Reference Binding



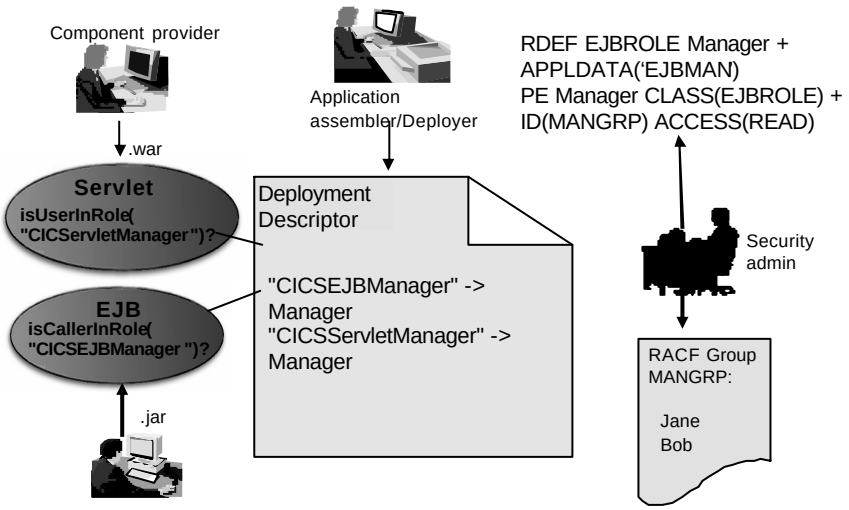
© 2002 IBM Corporation

AAT: Security Roles



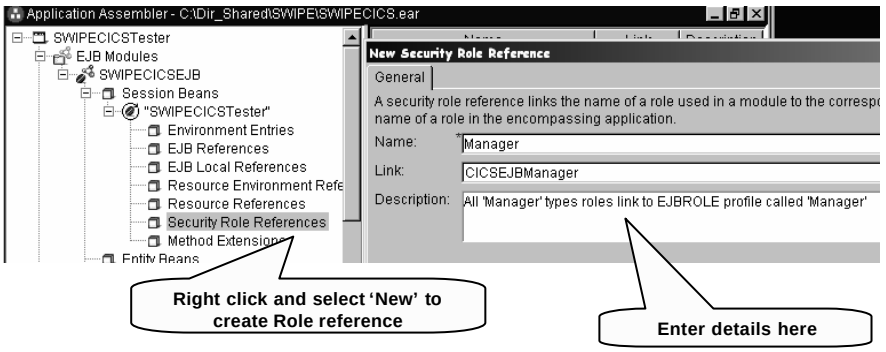
© 2002 IBM Corporation

AAT: Security Role References (1)



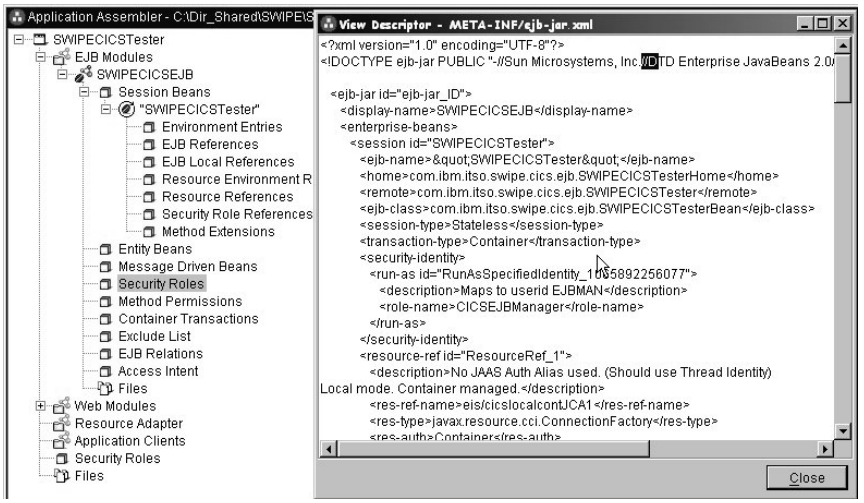
© 2002 IBM Corporation

AAT: Security Role References (2)



© 2002 IBM Corporation

AAT: View XML file



© 2002 IBM Corporation

AAT: Save

- When changes are finished use 'File' –'Save' to save the changes or click the diskette icon in the tool bar.
- There is no specific 'Export' option as there was in AAT for V4.
- You can now use the WebSphere Administration Console or wsadmin to deploy the application

© 2002 IBM Corporation

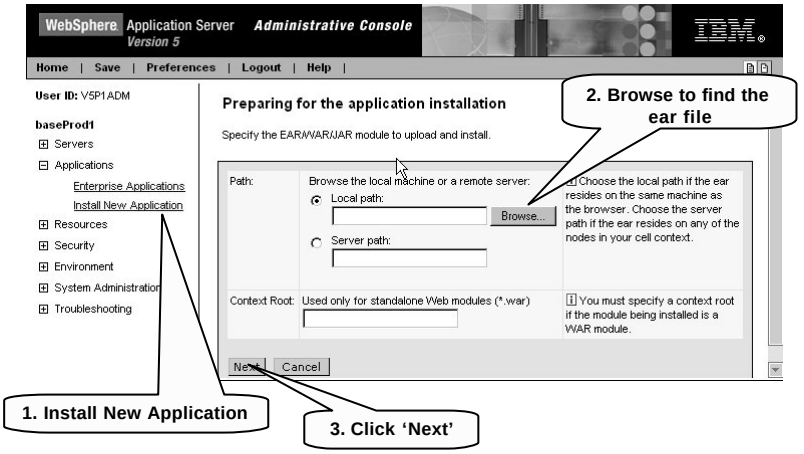
WebSphere z/OS V5

Application Deployment

Java Management Extensions (JMX)
Application Assembly Tool (AAT)
WebSphere Administrative Console
Deploying application with MDBs
wsadmin / JACL scripts and Ant
Hot deployment and Dynamic Reload
Classloaders

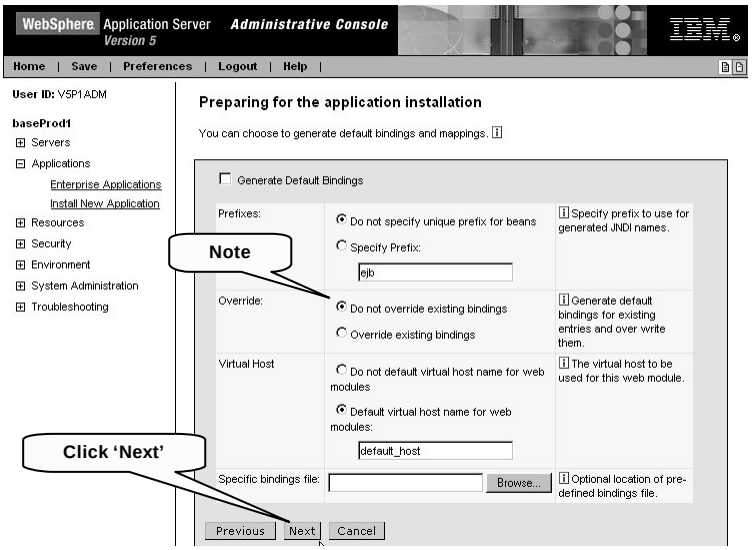
© 2002 IBM Corporation

Admin Console: Locate the ear file to install



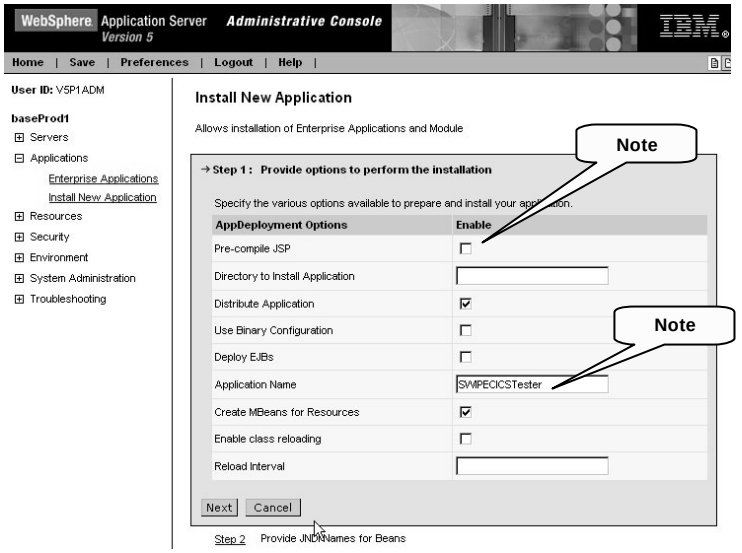
© 2002 IBM Corporation

Admin Console: Preparing for installation



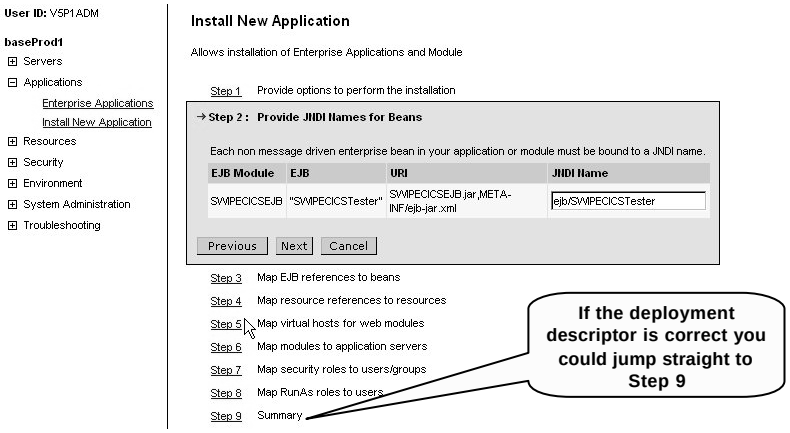
© 2002 IBM Corporation

Admin Console: Step 1: Application Options



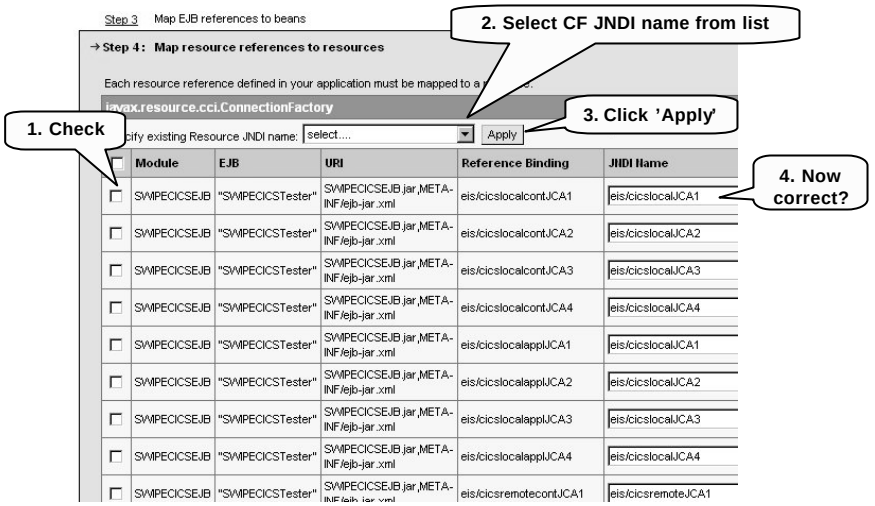
© 2002 IBM Corporation

Admin Console: Step 2: Bean JNDI names



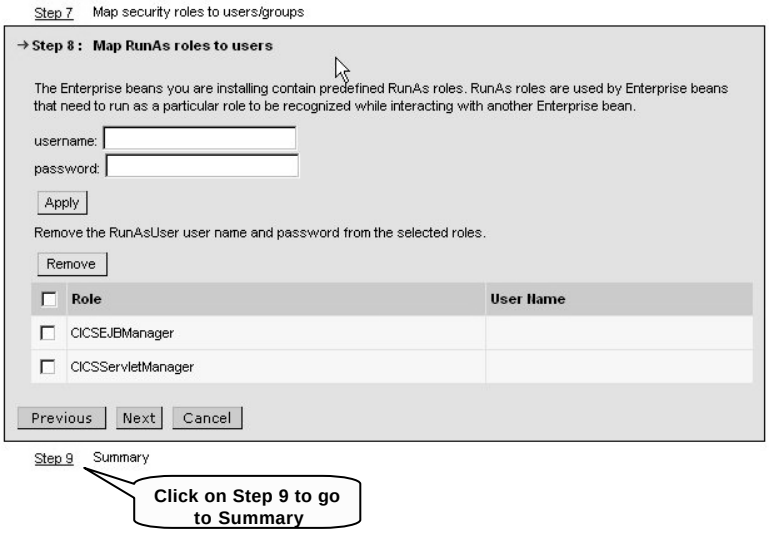
© 2002 IBM Corporation

Admin Console: Step 4: Resource References



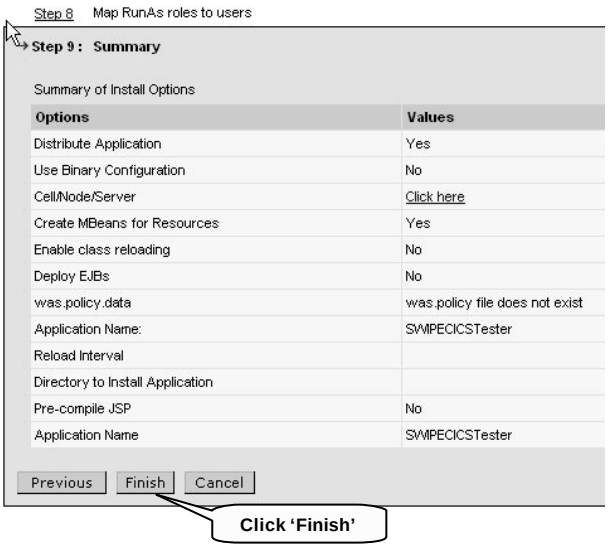
© 2002 IBM Corporation

Admin Console: Step 8: Map RunAs roles to users



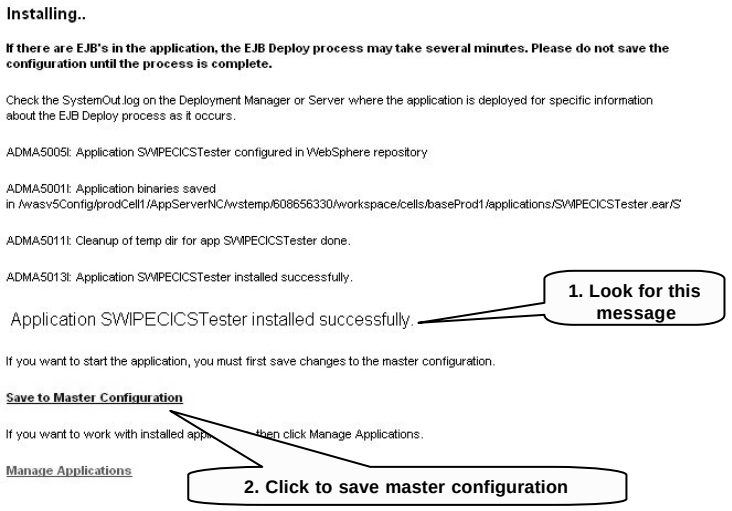
© 2002 IBM Corporation

Admin Console: Step 9: Summary



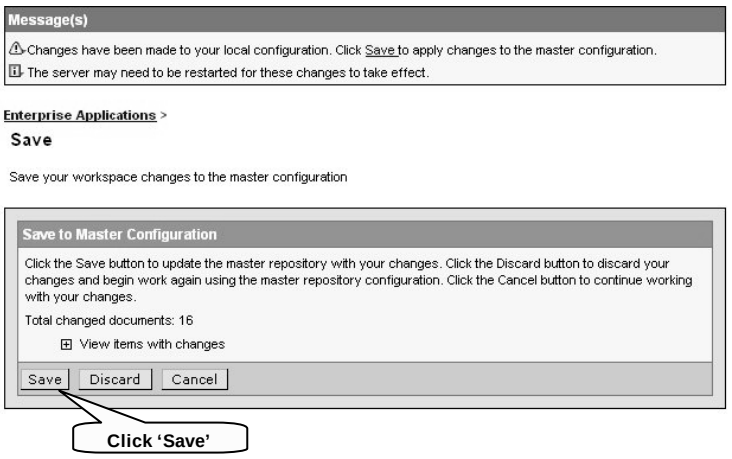
© 2002 IBM Corporation

Admin Console: Installation messages



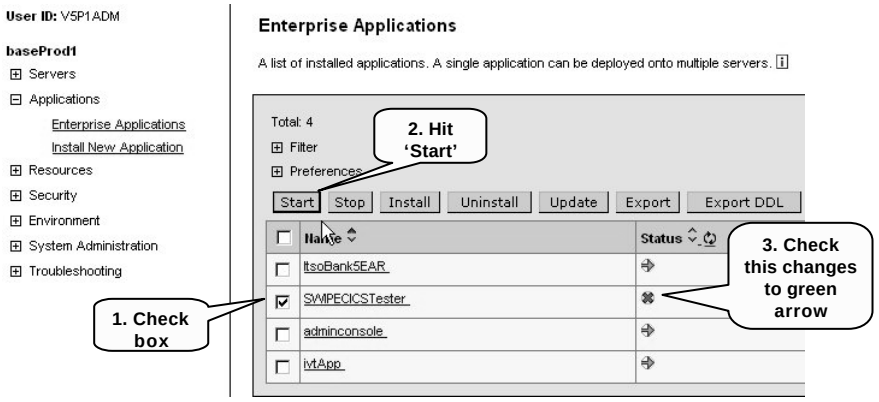
© 2002 IBM Corporation

Admin Console: Save master configuration



© 2002 IBM Corporation

Admin Console: Start the application



© 2002 IBM Corporation

WebSphere z/OS V5

Java Management Extensions (JMX)
Application Assembly Tool (AAT)
WebSphere Administrative Console
Deploying applications with MDBs
wsadmin / JACL scripts and Ant
Hot deployment and Dynamic Reload
Classloaders

© 2002 IBM Corporation

Deploying Applications with MDBs

- Use the WebSphere Administrative Console to define the listener ports

  Servers – Application Servers - <server name> - Additional Properties
  Select : Message Listener Service and set the properties.
  Open Custom Properties and review/set properties of listener service

- For each MDB, use the AAT to configure the deployment attributes to match the listener port

  Expand EJB module. Click on Message-driven beans.
  Open General Deployment tab. Set properties e.g. transaction type
  Open Advanced Deployment tab. Set properties e.g. Acknowledge mode
  Open Bindings tab. Set properties e.g. MDB listener port.

- Use the WebSphere Administrative Console to install the application
  - This stage is a standard WebSphere Application Server task
  - You are prompted to specify the name of the listener port that the application is to use for late responses. Select the listener port, then click **OK**.

© 2002 IBM Corporation

WebSphere z/OS V5

Java Management Extensions (JMX)
Application Assembly Tool (AAT)
WebSphere Administrative Console
Deploying application with MDBs
wsadmin / JACL scripts and Ant
Hot deployment and Dynamic Reload
Classloaders

© 2002 IBM Corporation

wsadmin overview

- wsadmin
  - Actually a specialized Java AdminClient application
  - Several possible languages (Tcl, Jython, Javascript) but only Jacl on z/OS
  - 3 modes - interactive, command line (-c) , script file (-f)
  - 3 objects to represent admin functions - AdminControl, AdminConfig, AdminApp
  - Primarily meant to allow rapid creation of small control programs using existing administration function
  - Considerable documentation in InfoCenter section **(200 pages or more!)**
- AdminClient Java API
  - Represents the remote client side of WAS admin programming interface
  - Used by all internal admin tools shipped with WAS 5.0
  - Based on JMX programming model, with distributed extensions
  - Allows more sophisticated admin program logic, better exception handling, registration for event notification
  - Public javadoc shipped in install image

© 2002 IBM Corporation

wsadmin scripting environment

- Refer to the Infocentre or Systems Administration manual GA22-7962
- Properties files define properties relating to a connection between a wsadmin client and a target server
- Three levels of properties files:
  - Level 1. Default properties: from <WAS_HOME>/properties/wsadmin.properties
  - Level 2. In the Java user.home property/wsadmin.properties
  - Level 3. WSADMIN_PROPERTIES environment variable -> wsadmin.properties
  - -p command line option points to a file with a properties
    These properties override others from the other sources which are themselves processed in order, each overriding the previous properties encountered. Keep it simple!
- Profiles can be used to customize the scripting environment
  - -profile can point to a properties file (which can point to another profile)

© 2002 IBM Corporation

wsadmin tracing

- Trace options can be set in a properties file (default wsadmin.properties)
  - Or –profile point to a properties file
- com.ibm.ws.scripting.traceString=true switches on trace to . . .
- com.ibm.ws.scripting.traceFile=/<filepath>/tracefile.name
  - If not set, trace / logging goes to the admin console
- Enabling trace from command line invocations:

  $AdminControl trace
  com.ibm.*=all=enabled

© 2002 IBM Corporation

Installing a new application using wsadmin

- wsadmin can be used to install a new application
- For interactive and command line, use Telnet to execute wsadmin or a batch job, not OMVS because the command line is too short
- Launch wsadmin by running wsadmin.sh file from <WAS_HOME>/bin
- Interactive install
  - <WAS_HOME>/bin/wsadmin.sh
  - wsadmin>$AdminApp **installInteractive** /<wasconfig_root>/<cell>/<node>/installableApps/SWIPECICS.ear
- Non-interactive invocation
  - <WAS_HOME>/bin/wsadmin.sh
  - wsadmin>$AdminApp **install** /<wasconfig_root>/<cell>/<node>/installableApps/SWIPECICS.ear

© 2002 IBM Corporation

Installing a new application using wsadmin

- Use -conntype NONE option, if the server stopped
  - wsadmin.sh –conntype NONE –c $AdminApp install /<wasconfig_root>/<cell>/<node>/installableApps/SWIPECICS.ear
- To list all installed applications (you need the server up to do this .... no local mode)
  - wsadmin>$AdminApp list
- Invoking help text
  - wsadmin>$Help help
  - wsadmin>$AdminApp help
  - wsadmin>$AdminApp help installInteractive

© 2002 IBM Corporation

Batch execution

- The customized CNTL library contains BBOIAPP which execute a batch job that invokes wsadmin to install the adminconsole and ivt applications.

```
//*********************************************************************/
//*  STEP 2 - Install the IVT app.                                    */
//*********************************************************************/
//INST2 EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD  SYSOUT=*
//SYSTSIN  DD  *
 BPXBATCH SH +
   /wasv5Config/prodCell1+
   /AppServerNC+
   /bin/wsadmin.sh -conntype none +
   -c '$AdminApp install +
   /wasv5Config/prodCell1+
   /AppServerNC+
   /installableApps/ivtApp.ear +
   {-appname ivtApp -usedefaultbindings +
   -server server2 +
   -node nodeC  +
   -cell baseProd1}' +
   1>> /tmp/bbowiapp.out +
   2>> /tmp/bbowiapp.err
/*
```


© 2002 IBM Corporation

Application Update

- From end-user perspective, the update process will be very similar to install
  - It will additionally take the name of the existing app that is to be upgraded
- No continuous availability of the application. The reinstall process:-
  - Stops the current application if it is running
  - Uninstalls the current application
  - Installs the new application j
- Continuous availability requires more planning
  - ND configuration
    - AutoSynch off to prevent automatic install across the cell
    - Use of the distributed plugin for initial balancing and affinities (implied by ND)
  - A way to control distribution of IP messages (dynamic VIPA and Sysplex distributor

© 2002 IBM Corporation

Application Update (continued)

- Using Admin Console
  - The user interface will guide the users through the same install steps for update
  - Applications ->Enterprise Applications. Select the Application and click "Update"
- Using wsadmin
  - Use update option in AdminApp
  - Example:

    $AdminApp install
    <wasconfig_root>/<cell>/<node>/installableApps/SWIPECICS.ear
    { -appname SWIPECICSTester -update -update.ignore.new}
    - -update option specifies that the application named SWIPECICS is to be redeployed using the supplied EAR file
    - -update.ignore.new option specifies that the bindings from the new version are ignored

© 2002 IBM Corporation

Ant

- Ant (ANother Tool) is now supported in WebSphere z/OS V5.0.1 (Service level W501000)
- Very popular with developers
- Apache Ant is a Java-based build tool.
  - Similar to Make, but extended by Java classes rather than shell scripts.
  - Instead of shell, XML files are used.
  - These XML files reference a target tree in which various tasks are run.
  - Each task is run by an object that implements a particular Task interface.
- For more detailed information about Ant, refer to the Apache Web site: http://jakarta.apache.org/ant/index.html
- Also see WebSphere distributed Redbook for usage information:
  SG24-6993 WebSphere Version 5 Application Development Handbook
  There are samples for wsadmin and Ant supplied with this Redbook

© 2002 IBM Corporation

Ant

- The tasks included with IBM WebSphere Application Server enable you to:
  - Install and uninstall applications
  - Run EJB deployment and JSP precompilation tools
  - Start and stop servers
  - Run administrative scripts or commands
- By combining these tasks with those provided by Ant, you can create build scripts that compile, package, install, and test your application on IBM WebSphere Application Server.
- WebSphere Ant is supplied in <WAS_HOME>/lib/ant.jar
  Javadoc for package: com.ibm.websphere.ant.tasks is at:

  http://publib7b.boulder.ibm.com/wasinfo1/en/info/aes/javadoc/ae/com/ibm/websphere/ant/tasks/package-summary.html
- Launched using: <was_install_root>/bin/ws_ant.sh
- Note that Ant build XML file and properties files must be in ascii.

© 2002 IBM Corporation

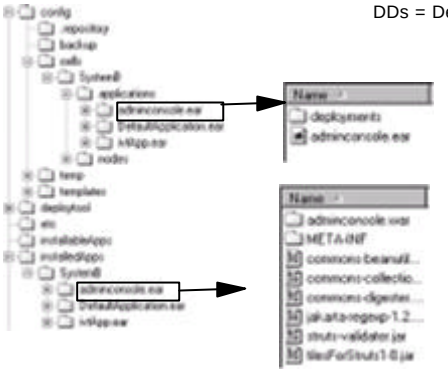
WebSphere z/OS V5

Application Deployment

Where did it go?

© 2002 IBM Corporation

Repository - Base Application Server



DDs = Deployment Descriptors

- deployments folder contains only the DDs with deployment modifications
- Copy of ear file here is not exploded and not executed

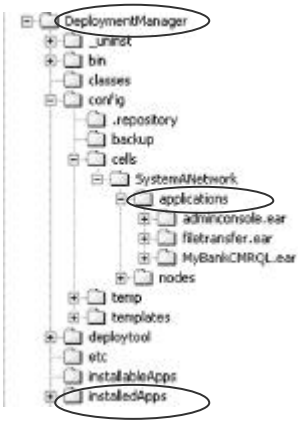
- Contains the exploded ear file
- Original DDs
- All application binaries

- DDs in installedApps dir. are the original DDs in the EAR file (unmodified DDs)
- App Server takes the modified DDs (from config dir.) to start the application
- DDs in config dir. contain all updates provided after installation of the Application ear.

Example: Could have provided a different JNDI name for an EJB or resource

© 2002 IBM Corporation

Repository – Network Deployment



Node configuration for all the Nodes are saved in machine configured with Deployment Manager

"applications" directory contain Application DDs

"installedApps"directory contains Application binaries

Deployment Manager Machine	Node Machine
Configuration files for all the Nodes are saved	Only configuration files specific to the Node will be saved

© 2002 IBM Corporation

WebSphere z/OS V5

Java Management Extensions (JMX)
Application Assembly Tool (AAT)
WebSphere Administrative Console
Deploying application with MDBs
wsadmin / JACL scripts
Hot Deployment and Dynamic Reloading
Classloaders

© 2002 IBM Corporation

Hot Deployment and Dynamic Reloading

- Hot deployment
  - The process of adding new components (such as WAR files, EJB Jar files, enterprise Java beans, servlets, and JSP files) to a running server without having to stop the application server process and start it again.
- Dynamic reloading
  - The ability to change an existing component without needing to restart the server in order for the change to take effect. Dynamic reloading involves:
    - Changes to the implementation of a component of an application, such as changing the implementation of a servlet
    - Changes to the settings of the application, such as changing the deployment descriptor for a Web module

© 2002 IBM Corporation

Hot Deployment Overview

- Locate your expanded ear file.
  If you did not override the location in Step 1 of the installation it will be in:
  install_root/installedApps/*cell_name*/*application_name*.ear.
- Locate the application metadata files.
- If running ND configuration with multiple nodes disable AutoSynch :
  System Administration – Node Agents – File Synchronization Service
  Uncheck the option: Automatic Synchronization
- Change or add the following components or modules as needed:
  - Application files
  - WAR files
  - EJB Jar files
  - HTTP plug-in configuration files

© 2002 IBM Corporation

Hot Deployment: General rules

- Web Modules can be reloaded independently in WebSphere distributed but not in WebSphere z/OS
- EJB Modules
  - Cannot be stopped/restarted independently in Version 5
  - The entire application has to be restarted
- Best practice: stop/restart the Enterprise Application
  - This will ensure that all the dependencies are also reloaded

© 2002 IBM Corporation

Dynamic Reload

- Set as an option on the application or on the deployment descriptor (META-INF/ibm-application-ext.xmi)

  (Reload Enabled and Reload Interval options)

- Performance penalty so avoid, especially avoid low reload intervals.

© 2002 IBM Corporation

WebSphere z/OS V5

Application Deployment

Java Management Extensions (JMX)
Application Assembly Tool (AAT)
WebSphere Administrative Console
Deploying application with MDBs
wsadmin / JACL scripts
Hot Deployment and Dynamic Reloading Classloaders

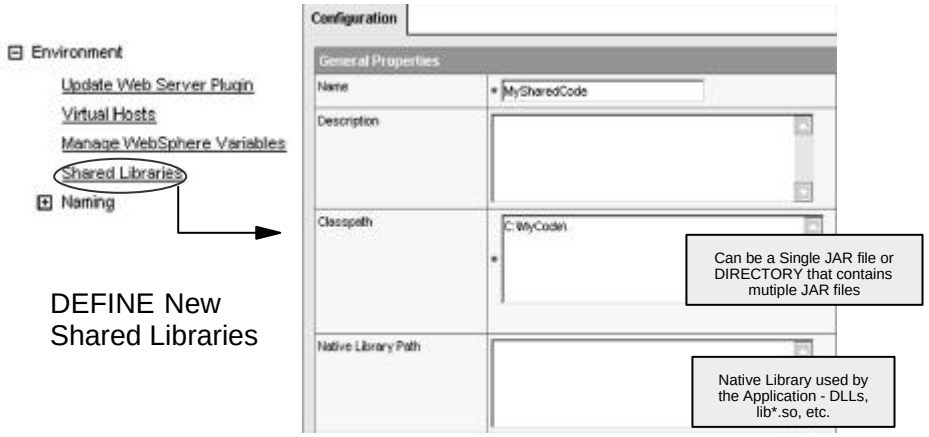
© 2002 IBM Corporation

v4 vs. v5 Classloading Policies - Comparison

v4	v5	
4.0 Classloader Policy	**v5 Application Classloader Policy**	**v5 WAR Classloader Policy**
SERVER	SINGLE	APPLICATION
COMPATIBILITY	SINGLE	MODULE
APPLICATION	MULTIPLE	APPLICATION
MODULE	N/A	N/A
J2EE Application Mode	MULTIPLE	MODULE

© 2002 IBM Corporation

Shared Libraries



DEFINE New Shared Libraries

© 2002 IBM Corporation

References

© 2002 IBM Corporation

References

- WebSphere z/OS V5 InfoCenter:

  http://www.ibm.com/software/webservers/appserv/zos_os390/library.html
  and follow link or,
  http://publib.boulder.ibm.com/infocenter/wasinfo/index.jsp
- WebSphere z/OS PDFs available from the zos_os390/library.html site.
- Redbooks and Redpieces (http://ibm.com/redbooks)
  - SG24-6086 z/OS WebSphere Application Server V5 and J2EE 1.3 Security Handbook
  - SG24-7044 Migrating applications from WebSphere for z/OS V4 and V3.5 to V5
  - SG24-6993 WebSphere Version 5 Application Development Handbook
    (useful for information on wsadmin and Ant but written for distributed WebSphere)

© 2002 IBM Corporation