

*IBM WebSphere Application  
Server V6.1 Administration*

(Course code WA361 / VA361)

**Instructor Guide**

ERC 6.3

WebSphere Education

## Trademarks

IBM® is a registered trademark of International Business Machines Corporation.

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

AIX®	AIX 5L™	Cloudscape™
DB2®	developerWorks®	Domino®
e-business on demand®	HACMP™	Passport Advantage®
Rational®	Tivoli®	WebSphere®
z/OS®	400®	

Adobe and Acrobat and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

Intel and Pentium are trademarks or registered trademarks 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.

### October 2007 edition

The information contained in this document has not been submitted to any formal IBM test and is distributed on an “as is” basis without any warranty either express or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer’s ability to evaluate and integrate them into the customer’s operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will result elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

© Copyright International Business Machines Corporation 2001, 2007. All rights reserved.

**This document may not be reproduced in whole or in part without the prior written permission of IBM.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

# Contents

<b>Trademarks</b> .....	<b>xv</b>
<b>Instructor course overview</b> .....	<b>xvii</b>
<b>Course description</b> .....	<b>xix</b>
<b>Agenda</b> .....	<b>xxi</b>
<b>Unit 1. Course introduction</b> .....	<b>1-1</b>
WA361 IBM WebSphere Application Server V6.1 Administration .....	1-2
Introductions .....	1-4
Course description .....	1-6
Course prerequisites .....	1-8
Course objectives .....	1-10
Agenda – Day 1 .....	1-12
Agenda – Day 2 .....	1-14
Agenda – Day 3 .....	1-16
Agenda – Day 4 .....	1-18
Course information .....	1-20
Unit summary .....	1-22
<b>Unit 2. WebSphere product family overview</b> .....	<b>2-1</b>
WebSphere product family overview .....	2-2
Unit objectives .....	2-4
J2EE 1.4 platform architecture .....	2-6
WebSphere software platform .....	2-8
WebSphere Application Server evolution .....	2-11
What is an Application Server? .....	2-13
WebSphere Application Server V6.1 .....	2-16
Packaging .....	2-18
WebSphere V6.1 packaging .....	2-20
Product packaging bundles .....	2-22
WebSphere Application Server – Express/Base .....	2-24
Differences – Express and Base offerings .....	2-26
WebSphere Application Server – Network Deployment .....	2-28
Important points .....	2-30
J2EE 1.4 standards supported .....	2-32
Session Initiation Protocol (SIP) servlets .....	2-34
Portlet support .....	2-36
IBM HTTP Server .....	2-38
WebSphere Edge Components .....	2-40
IBM Rational products .....	2-42
Tooling .....	2-45
Checkpoint questions .....	2-47

Unit summary .....	2-49
<b>Unit 3. WebSphere Application Server architectural overview: Stand-alone . . .</b>	<b>3-1</b>
WebSphere Application Server architecture overview: Stand-alone .....	3-3
Unit objectives .....	3-5
Main themes .....	3-7
Version 6 packaging .....	3-9
WebSphere Application Server basics .....	3-11
WebSphere architecture runtime .....	3-13
WebSphere architecture runtime .....	3-16
WebSphere architecture runtime .....	3-19
WebSphere architecture runtime .....	3-21
WebSphere architecture runtime .....	3-23
WebSphere architecture runtime .....	3-25
WebSphere architecture runtime .....	3-27
WebSphere architecture runtime .....	3-29
WebSphere architecture runtime .....	3-31
WebSphere architecture runtime .....	3-33
JDBC providers .....	3-35
Data sources .....	3-37
Enhanced EAR .....	3-39
WebSphere architecture administration .....	3-41
WebSphere architecture administration .....	3-44
WebSphere architecture administration .....	3-46
WebSphere architecture administration .....	3-48
WebSphere profiles overview .....	3-50
WebSphere profiles benefits .....	3-52
WebSphere profiles benefits .....	3-54
Managing profiles .....	3-56
Checkpoint questions .....	3-58
Unit summary .....	3-60
<b>Unit 4. WebSphere Application Server architectural overview: Federated . . .</b>	<b>4-1</b>
Architectural overview: Federated .....	4-2
Unit objectives .....	4-4
Version 6 packaging .....	4-6
Network deployment runtime flow .....	4-8
Network deployment concepts .....	4-10
Managed versus unmanaged nodes .....	4-12
Network deployment administration flow .....	4-14
File synchronization .....	4-16
WebSphere Network Deployment profiles .....	4-18
Managing Web servers with WebSphere .....	4-20
Unmanaged Web server .....	4-22
IBM HTTP Server as unmanaged node (remote) .....	4-24
Managed Web server on a managed node (local) .....	4-26
IBM HTTP Server administration overview .....	4-28
IBM HTTP Server administration server .....	4-30

Web server custom plugin-cfg.xml . . . . .	4-32
Managing plugin-cfg.xml files . . . . .	4-34
Managing Web server plug-in properties . . . . .	4-36
Platform Messaging overview . . . . .	4-38
High availability overview . . . . .	4-40
Data replication service . . . . .	4-42
Failover of stateful session EJBs . . . . .	4-44
Node group overview . . . . .	4-46
Name service . . . . .	4-48
Virtual hosts . . . . .	4-50
Defining virtual hosts . . . . .	4-52
Edge Components . . . . .	4-54
Checkpoint questions . . . . .	4-56
Unit summary . . . . .	4-58
<b>Unit 5. WebSphere Application Server installation . . . . .</b>	<b>5-1</b>
WebSphere Application Server V6.1 installation . . . . .	5-2
Unit objectives . . . . .	5-4
Hardware requirements . . . . .	5-6
Software requirements – Operating systems . . . . .	5-8
Software requirements – Supported Web servers . . . . .	5-10
Software requirements – Database . . . . .	5-12
Installation overview . . . . .	5-14
Installation environments . . . . .	5-16
WebSphere Profiles: Types . . . . .	5-18
Creating profiles . . . . .	5-20
Pre-Installation tasks . . . . .	5-22
Launchpad (1 of 2) . . . . .	5-24
Launchpad (2 of 2) . . . . .	5-27
Installation – Startup (1 of 2) . . . . .	5-29
Installation – Startup (2 of 2) . . . . .	5-31
Profile Management Tool (1 of 3) . . . . .	5-33
Profile Management Tool (2 of 3) . . . . .	5-35
Profile Management Tool (3 of 3) . . . . .	5-37
Profile creation – Command line tool . . . . .	5-39
First steps . . . . .	5-41
Installation verification . . . . .	5-43
Silent installation . . . . .	5-45
Uninstall . . . . .	5-47
Applications installed during installation . . . . .	5-49
Directory structure . . . . .	5-53
Common command line tools . . . . .	5-55
Server commands . . . . .	5-57
IBM HTTP Server . . . . .	5-59
Web server plug-in Installer . . . . .	5-61
IBM HTTP Server and plug-in installation (1 of 3) . . . . .	5-63
IBM HTTP Server and plug-in installation (2 of 3) . . . . .	5-65
IBM HTTP Server and plug-in installation (3 of 3) . . . . .	5-67

IBM HTTP Server: GUI and silent installation . . . . .	5-69
Installation Factory - the problem . . . . .	5-71
Installation Factory: custom solutions . . . . .	5-73
Installation Factory creation-time capabilities . . . . .	5-75
Installation Factory example . . . . .	5-77
Installation Factory opening panel . . . . .	5-79
Checkpoint questions . . . . .	5-81
Unit summary . . . . .	5-83
Lab exercise . . . . .	5-85
<b>Unit 6. WebSphere Application Server administrative console . . . . .</b>	<b>6-1</b>
WebSphere Application Server administrative console . . . . .	6-3
Unit objectives . . . . .	6-5
Administrative console . . . . .	6-7
Starting the administrative console . . . . .	6-9
Administrative console in a cell topology . . . . .	6-11
Console login . . . . .	6-13
Administrative console areas . . . . .	6-16
Administrative console navigation tree (1 of 2) . . . . .	6-18
Administrative console navigation tree (2 of 2) . . . . .	6-20
Administrative console banner . . . . .	6-22
Administrative console Help (1 of 2) . . . . .	6-24
Administrative console Help (2 of 2) . . . . .	6-26
Administrative console Preferences . . . . .	6-28
Administrative console filters . . . . .	6-30
Guided activities . . . . .	6-32
My tasks . . . . .	6-34
Troubleshooting information . . . . .	6-36
Tivoli Performance Viewer . . . . .	6-38
Checkpoint questions . . . . .	6-40
Unit summary . . . . .	6-42
Lab exercise . . . . .	6-44
<b>Unit 7. J2EE overview . . . . .</b>	<b>7-1</b>
J2EE overview . . . . .	7-3
Unit objectives . . . . .	7-5
Application programming model support . . . . .	7-7
J2EE benefits . . . . .	7-9
Standards supported . . . . .	7-11
J2EE 1.4 platform architecture . . . . .	7-14
J2EE platform role overview . . . . .	7-16
J2EE platform roles (1 of 4) . . . . .	7-18
J2EE platform roles (2 of 4) . . . . .	7-20
J2EE platform roles (3 of 4) . . . . .	7-22
J2EE platform roles (4 of 4) . . . . .	7-24
J2EE 1.4 application components . . . . .	7-26
J2EE packaging files . . . . .	7-28
Checkpoint questions . . . . .	7-30

Unit summary	7-32
<b>Unit 8. Application assembly</b>	<b>8-1</b>
Application assembly	8-3
Unit objectives	8-5
Overview of application assembly and installation	8-7
J2EE packaging	8-9
WebSphere: Application packaging	8-11
WebSphere application contents	8-13
Application Server Toolkit (AST)	8-15
Application Server Toolkit functionality	8-17
Assembling an enterprise application	8-19
Import modules	8-21
J2EE Perspective	8-23
Application deployment descriptor (1 of 2)	8-25
Application deployment descriptor (2 of 2)	8-27
Application scope resources	8-29
Enhanced EAR	8-31
Un-enhancing an EAR file	8-33
Dealing with enhancements at deploy time	8-35
EJB module assembly	8-37
EJB module expanded	8-39
EJB deployment descriptor editor	8-41
Web module assembly	8-43
Web module	8-45
Web deployment descriptor editor	8-47
Application client assembly	8-49
Application Client module	8-51
Generating an EAR File for deployment	8-53
Checkpoint questions	8-55
Unit summary	8-57
<b>Unit 9. Introduction to the Trade family of applications</b>	<b>9-1</b>
Introduction to the Trade family of applications	9-2
Unit objectives	9-4
Trade application family overview	9-6
Trade application general functionality	9-8
Welcome page	9-10
Registration page	9-12
Login page and runtime server information	9-14
Other information on pages	9-16
Account Information page	9-18
Get Quote pop-up	9-20
Update Account Information page	9-22
Holdings page	9-24
Buy and Sell page	9-26
Transaction History page	9-28
Administrative Functions menu	9-30

Administration – Adjust user balance .....	9-32
Trade application modules .....	9-34
QuoteWS general functionality .....	9-36
QuoteWS modules .....	9-38
Trade Processor general functionality .....	9-40
Trade Ticker .....	9-42
Combined Ticker and Processor page .....	9-44
Trade Processor modules .....	9-46
Trade deployment architecture .....	9-48
Unit summary .....	9-50
Lab exercise .....	9-52
<b>Unit 10. Application installation .....</b>	<b>10-1</b>
Application installation .....	10-2
Unit objectives .....	10-4
Installing enterprise applications .....	10-6
Installation tasks .....	10-8
Creating a J2C authentication alias .....	10-10
Creating JDBC provider resources (1 of 2) .....	10-12
Creating JDBC provider resources (2 of 2) .....	10-14
Creating a data source (1 of 3) .....	10-16
Creating a data source (2 of 3) .....	10-19
Creating a data source (3 of 3) .....	10-21
Make application files available to server .....	10-23
Enterprise application installation (1 of 7) .....	10-25
Enterprise application installation (2 of 7) .....	10-27
Enterprise application installation (3 of 7) .....	10-29
Enterprise application installation (4 of 7) .....	10-31
Enterprise application installation (5 of 7) .....	10-33
Enterprise application installation (6 of 7) .....	10-35
Enterprise application installation (7 of 7) .....	10-37
Managing applications .....	10-39
Starting an application .....	10-41
Application auto start .....	10-43
Other application configuration options .....	10-45
Checkpoint questions .....	10-47
Unit summary .....	10-49
Lab exercise .....	10-51
<b>Unit 11. Problem determination .....</b>	<b>11-1</b>
Problem determination .....	11-2
Unit objectives .....	11-4
What is the IBM Support Assistant? .....	11-6
IBM Support Assistant console .....	11-9
IBM Support Assistant: Search component .....	11-11
IBM Support Assistant: Product information component .....	11-13
IBM Support Assistant: Tools component .....	11-16
IBM Support Assistant: Service component .....	11-18



IBM Support Assistant: Updater component	11-21
Problem determination – Divide and conquer	11-23
Searching on the Support page	11-25
WebSphere Support page	11-27
Searching the Information center	11-29
Product components and MustGather documents	11-31
WebSphere Application Server logs	11-33
Log files: Types and locations	11-36
Viewing logs	11-38
Configuring JVM logs	11-40
Viewing runtime messages in the console (1 of 2)	11-42
Viewing runtime messages in the console (2 of 2)	11-44
HTTP plug-in logs and tracing	11-46
Embedded HTTP Server logs	11-48
Traces	11-50
Enabling trace	11-52
Setting the log detail level	11-54
Reading a log or trace file	11-57
Generating an IBM Heap Dump	11-60
Hung thread detection	11-62
Connection leak diagnostics	11-64
Checking version levels and applying APARs	11-67
Install problems	11-69
Install exit codes	11-71
Installation log files (1 of 2)	11-73
Installation log files (2 of 2)	11-75
Problem determination tool availability	11-77
Problem determination tools (1 of 2)	11-79
Problem determination tools (2 of 2)	11-81
IBM Guided Activity Assistant	11-83
Collector tools	11-85
Configuration validator	11-87
Application Server Toolkit (AST)	11-89
AST – Profiling and logging perspective	11-91
Dumping the JNDI namespace	11-93
Dumping the JNDI namespace example	11-96
Checkpoint questions	11-98
Unit summary	11-100
Lab exercise	11-102
<b>Unit 12. Introduction to wsadmin and scripting</b>	<b>12-1</b>
Introduction to wsadmin and scripting	12-2
Unit objectives	12-4
Comparison of wsadmin and administration console	12-6
Why use scripting?	12-8
Introduction to wsadmin	12-10
Overview of wsadmin	12-12
Administrative functions using wsadmin	12-14

Administrative objects in wsadmin	12-16
Invoking wsadmin	12-18
Invoking wsadmin with security enabled	12-20
Connection type in wsadmin	12-22
Profile scripts	12-24
wsadmin properties	12-26
AdminConfig – Managing configurations	12-28
AdminApp – Managing applications	12-30
AdminControl – Managing running objects	12-32
AdminTask – Accessing administrative functions	12-34
Help within wsadmin	12-36
Help on specific commands	12-38
Important points to remember when using wsadmin	12-40
Scripting – Simple script	12-42
Scripting – Looping	12-44
Create, test, and debug scripts with AST	12-46
Launch Jython debugger	12-48
Debugger perspective	12-50
WebSphere Administration Command assist	12-52
WebSphere Administration Command assist	12-54
Build tool: Apache Ant	12-56
WebSphere Ant tasks	12-58
Some WebSphere Ant tasks	12-60
Checkpoint questions	12-62
Unit summary	12-64
Lab exercise	12-66
<b>Unit 13. Federating a cell</b>	<b>13-1</b>
Federating cells	13-2
Unit objectives	13-4
WebSphere cells	13-6
WebSphere Application Server types	13-8
Network deployment concepts	13-10
WebSphere profiles overview	13-12
WebSphere profiles benefits	13-14
WebSphere profiles types	13-16
Application server profile	13-18
Deployment manager profile	13-20
Custom profile	13-22
Creating profiles	13-24
Profile Management Tool (1 of 4)	13-26
Profile Management Tool (2 of 4)	13-28
Profile Management Tool (3 of 4)	13-30
Profile Management Tool (4 of 4)	13-32
Profile Creation – Command-line tool	13-34
First steps	13-36
Directory structure	13-38
Server commands	13-40

Common command-line tools . . . . .	13-42
WebSphere profiles Issues . . . . .	13-44
Adding a WebSphere node to a cell . . . . .	13-46
Cell topology . . . . .	13-48
Configuring synchronization . . . . .	13-50
Remove WebSphere node from a cell . . . . .	13-52
Checkpoint questions . . . . .	13-54
Unit summary . . . . .	13-56
Lab exercise . . . . .	13-58
<b>Unit 14. Workload management . . . . .</b>	<b>14-1</b>
Workload management . . . . .	14-2
Unit objectives . . . . .	14-4
What is workload management (WLM)? . . . . .	14-6
What can be workload managed? . . . . .	14-8
Clusters . . . . .	14-10
Clusters and cluster members . . . . .	14-12
Configurations: Vertical scaling . . . . .	14-14
Configurations: Horizontal scaling . . . . .	14-16
Configurations: Vertical and horizontal scaling . . . . .	14-18
Creating a cluster (1 of 3) . . . . .	14-20
Creating a cluster (2 of 3) . . . . .	14-22
Creating a cluster (3 of 3) . . . . .	14-24
Installing enterprise applications to a cluster . . . . .	14-26
Controlling a cluster . . . . .	14-28
Modification of clusters . . . . .	14-30
Basic WLM request routing . . . . .	14-32
Weighted routing example with no affinity . . . . .	14-34
Weighted routing example with affinity . . . . .	14-36
Edge components failover . . . . .	14-38
HTTP Server failover . . . . .	14-40
Web container failover . . . . .	14-42
EJB container failover . . . . .	14-44
HTTP session management . . . . .	14-46
Session affinity . . . . .	14-48
WebSphere session affinity . . . . .	14-50
Session persistence . . . . .	14-52
Session persistence tuning . . . . .	14-54
Session configuration: Memory-to-memory . . . . .	14-56
Session configuration: Replication domains . . . . .	14-58
Database persistence configuration . . . . .	14-60
Checkpoint questions . . . . .	14-62
Unit summary . . . . .	14-64
Lab exercise . . . . .	14-66
<b>Unit 15. Introduction to WebSphere Platform Messaging . . . . .</b>	<b>15-1</b>
Introduction to WebSphere Platform Messaging . . . . .	15-2
Unit objectives . . . . .	15-4

What is JMS . . . . .	15-6
Messaging – Basic flow . . . . .	15-8
WebSphere Platform Messaging . . . . .	15-10
Service Integration Bus (SIBus) . . . . .	15-12
Bus member . . . . .	15-15
Messaging engine (ME) . . . . .	15-17
Bus member and messaging engine . . . . .	15-19
Message store . . . . .	15-21
Messaging engine Data stores . . . . .	15-24
Bus destinations . . . . .	15-26
Linking destinations to bus members . . . . .	15-29
SIBus destination . . . . .	15-32
J2EE access to bus members . . . . .	15-34
JMS Destinations . . . . .	15-36
JMS Connection Factory . . . . .	15-38
JMS ActivationSpec . . . . .	15-40
Clustering for high availability . . . . .	15-42
Clustering for High Availability: Queue example . . . . .	15-44
Clustering for scalability: Queue example . . . . .	15-46
Clustering for scalability: Queue example . . . . .	15-48
Messaging engine topology . . . . .	15-50
SIBus and MEs in a stand-alone server . . . . .	15-52
Bus topology – Stand-alone servers . . . . .	15-54
SIBus and MEs in a Network Deployment Cell . . . . .	15-56
Network topologies – Network Deployment cell . . . . .	15-58
Destination quality of service for reliability . . . . .	15-61
Mediation . . . . .	15-63
Platform Messaging- Interoperability . . . . .	15-65
Relationship to WebSphere MQ . . . . .	15-67
Interoperability with MQ . . . . .	15-69
Usage scenarios for Platform Messaging . . . . .	15-71
V6 support for external JMS providers . . . . .	15-73
Service Integration as part of an ESB . . . . .	15-75
Checkpoint questions . . . . .	15-77
Unit summary . . . . .	15-79
Lab exercise . . . . .	15-81
<b>Unit 16. WebSphere security . . . . .</b>	<b>16-1</b>
WebSphere security . . . . .	16-2
Unit objectives . . . . .	16-4
Basic security end-to-end model . . . . .	16-6
WebSphere Application Server security overview . . . . .	16-8
WebSphere security service – Big picture . . . . .	16-11
Authentication and access control . . . . .	16-13
Authentication basic steps . . . . .	16-15
Authentication . . . . .	16-17
Basic authentication challenge types . . . . .	16-19
Registries and authentication mechanisms . . . . .	16-21

WebSphere administrative and application security	16-24
Federated repositories	16-26
Federated repositories: architecture overview	16-28
User registry support	16-30
Custom registry	16-32
Custom registry - Configuration	16-34
Authentication mechanisms	16-36
LTPA and LDAP	16-38
Delegation	16-40
Single sign-on (SSO)	16-42
Security wizard	16-44
Security wizard – step 1	16-46
Security wizard – step 2	16-48
Security wizard – step 3	16-50
Security wizard – step 4	16-52
Authorization	16-54
J2EE security roles: Application authorization	16-56
Securing J2EE application artifacts	16-58
Applying J2EE security with AST	16-60
Applying J2EE security roles using the console	16-62
Console security	16-64
Additional console security roles	16-66
Console security – creating users and groups	16-68
Console security – mapping users and groups	16-70
Java security model	16-72
Java 2 security overview	16-74
Enabling Java 2 security	16-76
What is SSL?	16-79
Symmetric key encryption	16-81
Asymmetric key encryption	16-83
How does SSL work?	16-85
Certificates and certificate authority (CA)	16-87
SSL within WebSphere Application Server	16-89
Creating key rings and certificates	16-91
SSL between plug-in and the application server	16-93
Cell key rings and truststores	16-96
Managing WebSphere key rings	16-98
Plug-in key rings	16-100
IBM HTTP Server key ring propagation	16-102
Checkpoint questions	16-104
Unit summary	16-106
Lab exercise	16-108
<b>Unit 17. Performance monitoring and tuning tools</b>	<b>17-1</b>
Performance monitoring and tuning tools	17-2
Unit objectives	17-5
What is the cause of the performance problem?	17-7
The need for performance monitoring and tuning	17-10

WebSphere performance tools .....	17-12
WebSphere performance technologies .....	17-14
Performance tool usage and data collection technology .....	17-16
Performance Monitoring Infrastructure (PMI) data collection technology .....	17-18
Performance Monitoring Infrastructure overview .....	17-20
PMI data modules and counters .....	17-22
Data collection settings .....	17-24
Enabling PMI .....	17-27
Specifying the data collection setting .....	17-29
Monitoring system health with Tivoli Performance Viewer .....	17-31
Tivoli Performance Viewer overview .....	17-33
TPV monitoring: First step .....	17-35
TPV user interface .....	17-37
TPV summary reports .....	17-39
Summary report example: Servlets .....	17-41
TPV performance modules .....	17-43
Performance module example: Web applications .....	17-45
Retrieving PMI data with the Performance servlet .....	17-47
Performance servlet overview .....	17-49
Performance servlet output example .....	17-51
Performance servlet features .....	17-53
Monitoring application flow with Request Metrics .....	17-56
Request Metrics overview .....	17-58
Request Metrics features .....	17-60
Using Request Metrics (1 of 2) .....	17-63
Using Request Metrics (2 of 2) .....	17-66
Using the Performance advisors to tune WebSphere .....	17-68
Performance advisors overview .....	17-70
Advisor architecture .....	17-72
TPV Advisor and Performance and Diagnostic Advisor comparison .....	17-74
Examples of advice .....	17-76
Using the TPV Performance Advisor .....	17-79
TPV Performance Advisor output example .....	17-82
Using the Performance and Diagnostic Advisor .....	17-84
Performance and Diagnostic Advisor output example (1 of 2) .....	17-87
Performance and Diagnostic Advisor output example (2 of 2) .....	17-89
Performance advisors best practices .....	17-91
WebSphere tuning best practice: Upstream queuing .....	17-94
Checkpoint .....	17-97
Unit summary .....	17-99
Lab exercise .....	17-101

<b>Unit 18. Course summary .....</b>	<b>18-1</b>
Course summary .....	18-2
Course objectives .....	18-4
Resources .....	18-6
Other WebSphere courses .....	18-8
Evaluations .....	18-10

# Trademarks

The reader should recognize that the following terms, which appear in the content of this training document, are official trademarks of IBM or other companies:

IBM® is a registered trademark of International Business Machines Corporation.

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

AIX®	AIX 5L™	Cloudscape™
DB2®	developerWorks®	Domino®
e-business on demand®	HACMP™	Passport Advantage®
Rational®	Tivoli®	WebSphere®
z/OS®	400®	

Adobe and Acrobat and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

Intel and Pentium are trademarks or registered trademarks 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.





# Instructor course overview

This four-day course describes how to install and perform system administration for IBM WebSphere Application Server V6.1 Network Deployment. It covers the necessary skills for a stand-alone base WebSphere Application Server system and the additional skills to manage and configure a Network Deployment cell.

This course does not cover programming Java servlets, JSPs, or EJBs. These skills can be obtained by attending the appropriate IBM Rational Application Developer courses.

The target audience is customers, business partner and IBM personnel who need to install, configure and maintain systems using WebSphere Application Server V6.1.

Topics include:

- Overview of WebSphere Application Server and enterprise Java applications and services
- Installation and basic configuration of WebSphere Application Server and required components
- Installation and configuration of IBM HTTP Server V6.1
- Using the WebSphere administrative console
- Deploying J2EE 1.4 applications
- Security issues and configuring security for WebSphere Application Server and enterprise applications
- Network Deployment configuration and workload management
- Problem determination and resource analysis

The course covers WebSphere Application Server V6.1 Network Deployment for the Windows 2000, Linux and AIX platforms. Lab exercises are provided and support all of these platforms. Normally, a single class only uses one platform for the lab exercises but mixed operating system classes are not unusual.

In the class introduction be sure to set expectations as to the depth of the class. There is no way to make a person an expert in a product as rich in function as WebSphere Application Server in only four days.

Stress that this is only an introduction to the administrative functions of WebSphere Application Server and that true expertise can only come with experience. Also as part of the class introduction be sure to tell students that the class is taught at the V6.1 level.



---

# Course description

## IBM WebSphere Application Server V6.1 Administration

**Duration:** 4 days

### Purpose

This course enables students to install, configure, and maintain WebSphere Application Server V6.1 stand-alone and Network Deployment, install and configure IBM HTTP Server V6 and to deploy enterprise Java applications in a stand-alone or clustered configuration.

### Audience

The target audience of this course is administrators who install, configure, and manage Web-based applications on WebSphere Application Server.

### Prerequisites

Students should have:

- An understanding of the basic Internet concepts
- Experience using a Web browser
- Administration skills for a Web server (such as IBM HTTP Server or Apache)
- Basic operational skills for the Windows 2000 operating system

### Objectives

After completing this course, you should be able to:

- Relate WebSphere Application Server (WAS) to the WebSphere family of products
- Describe the architectural concepts related to WebSphere Application Server V6.1
- Install and configure WebSphere Application Server and IBM HTTP Server (IHS)
- Explain the basic concepts of the Java 2 Platform, Enterprise Edition (J2EE)

- Assemble and install server-side Java enterprise applications
- Use WebSphere Application Server administrative tools to configure and manage enterprise applications
- Configure security for server-side application resources
- Deploy applications in clustered environments
- View performance information about server and application components
- Use problem determination tools and log files to troubleshoot problems

# Agenda

## Day 1

- (0:30) Unit 1 - Course introduction
- (0:30) Unit 2 - WebSphere product family overview
- (1:00) Unit 3 - WebSphere Application Server architecture overview:  
Stand-alone
- (1:30) Unit 4 - Architectural overview: Federated
- (0:30) Unit 5 - WebSphere Application Server V6.1 installation
- (1:00) Exercise 1 - WebSphere Application Server installation
- (1:00) Exercise 2- IBM HTTP Server installation
- (0:30) Exercise 3- WebSphere Information Center installation (opt)
- (0:30) Unit 6 - WebSphere Application Server administrative console
- (0:30) Exercise 4- Explore WebSphere Application Server through the  
Administrative Console

## Day 2

- (0:30) Unit 7 - J2EE overview
- (0:30) Unit 8 - Application Assembly
- (0:30) Unit 9 - Introduction to the Trade family of applications
- (1:00) Exercise 5- Using AST to assemble the Trade application  
or Demonstration of AST part of exercise for AIX
- (0:30) Unit 10 - Application installation
- (0:45) Exercise 6- Installing Trade application
- (1:00) Unit 11 - Problem determination
- (1:15) Exercise 7- Problem determination (optional)  
or Demonstration of AST part of exercise for AIX

## Day 3

- (0:45) Unit 12 - Introduction to wsadmin and scripting
- (1:00) Exercise 8 - Experimenting with wsadmin  
or Demo of AST part of exercise for AIX
- (1:00) Unit 13 - Federating cells
- (1:00) Exercise 9 - Federating a cell
- (1:00) Unit 14 - Workload management
- (1:00) Exercise 10 - Clustering
- (1:00) Unit 15 - Introduction to WebSphere Platform Messaging  
(WPM)
- (1:00) Exercise 11 - Install Trade Processor Application

## Day 4

- (1:30) Unit 16 - WebSphere Security
- (1:00) Exercise 12 - Configuring SSL (optional)
- (1:00) Exercise 13 - Enabling WebSphere and J2EE security
- (1:00) Unit 17 - Performance monitoring and tuning tools
- (1:00) Exercise 14 - Integrated Performance Viewer (optional)
- (0:15) Unit 18 - Course summary

# Unit 1. Course introduction

## Estimated time

00:30

## What this unit is about

This unit provides an overview for the entire course.

## What you should be able to do

After completing this unit, you should be able to:

- Understand the order of the lectures and labs
- Understand the prerequisites for this course
- List the objectives for this course

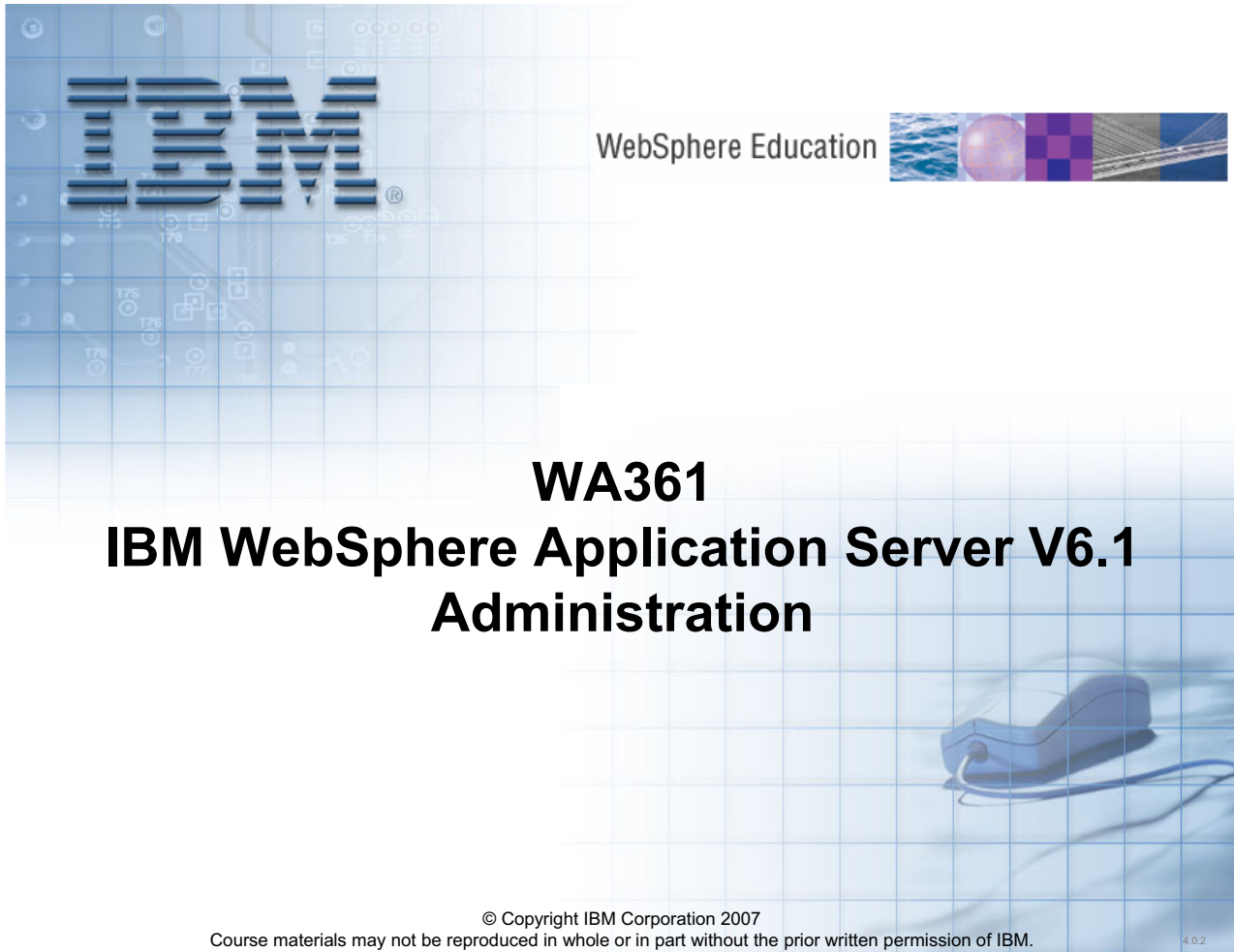


Figure 1-1. WA361 IBM WebSphere Application Server V6.1 Administration

WA3616.3

**Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

# Introductions

---

- Instructors
  
- Students
  - Introduce yourself
  - Name and organization
  - Internet knowledge
  - Java and Java Enterprise knowledge
  - IBM WebSphere Application Server knowledge
  - Internet and Web tools you use (or do not use)
  - What you hope to get from the class



© Copyright IBM Corporation 2007

Figure 1-2. Introductions

WA3616.3

## **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Course description

---

### WA361 – IBM WebSphere Application Server V6.1 Administration

**Duration:** 4 days

**Purpose:**

- This course enables students to install, configure, and maintain IBM WebSphere Application Server Network Deployment V6.1, and to deploy enterprise Java applications in a single server or clustered configuration
- Hands-on lab exercises are provided throughout the course for Windows 2000, Linux, and AIX platforms

© Copyright IBM Corporation 2007

---

Figure 1-3. Course description

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Course prerequisites

---

- To get the most out of this course offering, students should meet the following prerequisites:
  - An understanding of basic Internet concepts
  - Experience using a Web browser
  - Basic operational skills for the chosen lab environment:
    - Windows 2000
    - LINUX
    - AIX
  
- Students who do not meet these prerequisites may not be able to fully understand and utilize the materials presented in the course.

© Copyright IBM Corporation 2007

Figure 1-4. Course prerequisites

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Course objectives

---

- After completing this course, students should be able to:
  - Relate WebSphere Application Server to the WebSphere family of products
  - Describe the architectural concepts related to WebSphere Application Server V6.1
  - Install and configure WebSphere Application Server and IBM HTTP Server (IHS)
  - Explain the basic concepts of the Java 2 Platform, Enterprise Edition 1.4
  - Assemble and configure server-side Java enterprise applications using WebSphere Application Server Toolkit (AST)
  - Use WebSphere Application Server administrative tools to install, configure and manage enterprise applications
  - Configure WebSphere Platform Messaging (WPM)
  - Configure security for server-side application resources
  - Configure and deploy enterprise applications
  - View performance information about server and application components
  - Use problem determination tools and log files to troubleshoot problems

© Copyright IBM Corporation 2007

Figure 1-5. Course objectives

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Agenda – Day 1

---

- Course introduction
- WebSphere product family overview
- Architectural overview: Standalone
- Architectural overview: Federated
- WebSphere Application Server V6.1 installation
  - Exercise: WebSphere Application Server installation
  - Exercise: IBM HTTP Server installation
  - Exercise: WebSphere Information Center installation (optional)
- Administrative console
  - Exercise: Explore WebSphere Application Server through the administrative console

© Copyright IBM Corporation 2007

Figure 1-6. Agenda – Day 1

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Agenda – Day 2

---

- J2EE overview
  
- Application assembly using AST
  
- Introduction to the Trade family of applications
  - Exercise: Using AST to assemble the Trade application
  - Demo of AST part of exercise for AIX
  
- Application installation
  - Exercise: Installing the Trade application
  
- Problem determination
  - Exercise: Problem determination (optional)
  - Demo of AST part of exercise for AIX

© Copyright IBM Corporation 2007

Figure 1-7. Agenda – Day 2

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Agenda – Day 3

---

- Introduction to wsadmin and scripting
  - Exercise: Experimenting with wsadmin and scripting
  - Demo of AST part of exercise for AIX
  
- Profiles and federation
  - Exercise: Federating a cell
  
- Workload management
  - Exercise: Clustering

© Copyright IBM Corporation 2007

Figure 1-8. Agenda – Day 3

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Agenda – Day 4

---

- Introduction to WebSphere Platform Messaging (WPM)
  - Exercise: Set up the Service Integration Bus
- WebSphere security
  - Exercise: Configuring SSL (optional)
  - Exercise: Enabling WebSphere and J2EE security
- Performance monitoring and tuning tools
  - Exercise: Performance monitoring and tuning tools (optional)
- Course summary

© Copyright IBM Corporation 2007

Figure 1-9. Agenda – Day 4

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Course information

---

- Course is taught at WebSphere Application Server Network Deployment V6.1 level
- Screen captures are Windows centric
- Differences between operating systems will be explained in the materials
- Note: The course files are IBM copyrighted and are only for the use of those who attended this class. Use by non-attendees is not authorized.
  - No product software is included

© Copyright IBM Corporation 2007

Figure 1-10. Course information

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Unit summary

---



© Copyright IBM Corporation 2007

Figure 1-11. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**



---

## Unit 2. WebSphere product family overview

### Estimated time

00:30

### What this unit is about

This unit describes the products in the WebSphere product family.

### What you should be able to do

After completing this unit, you should be able to:

- Describe the WebSphere family of products and the functions of each product
- Relate development and deployment products in the WebSphere family
- Describe WebSphere Application Server V6.1 packages
- Describe WebSphere Application Server V6.1 features and functions

### How you will check your progress

Accountability:

- Checkpoint

### References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center



Figure 2-1. WebSphere product family overview

WA3616.3

**Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe the WebSphere family of products and the functions of each product
- Relate development and deployment products in the WebSphere family
- Describe WebSphere Application Server V6.1 packages
- Describe WebSphere Application Server V6.1 features and functions

© Copyright IBM Corporation 2007

Figure 2-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

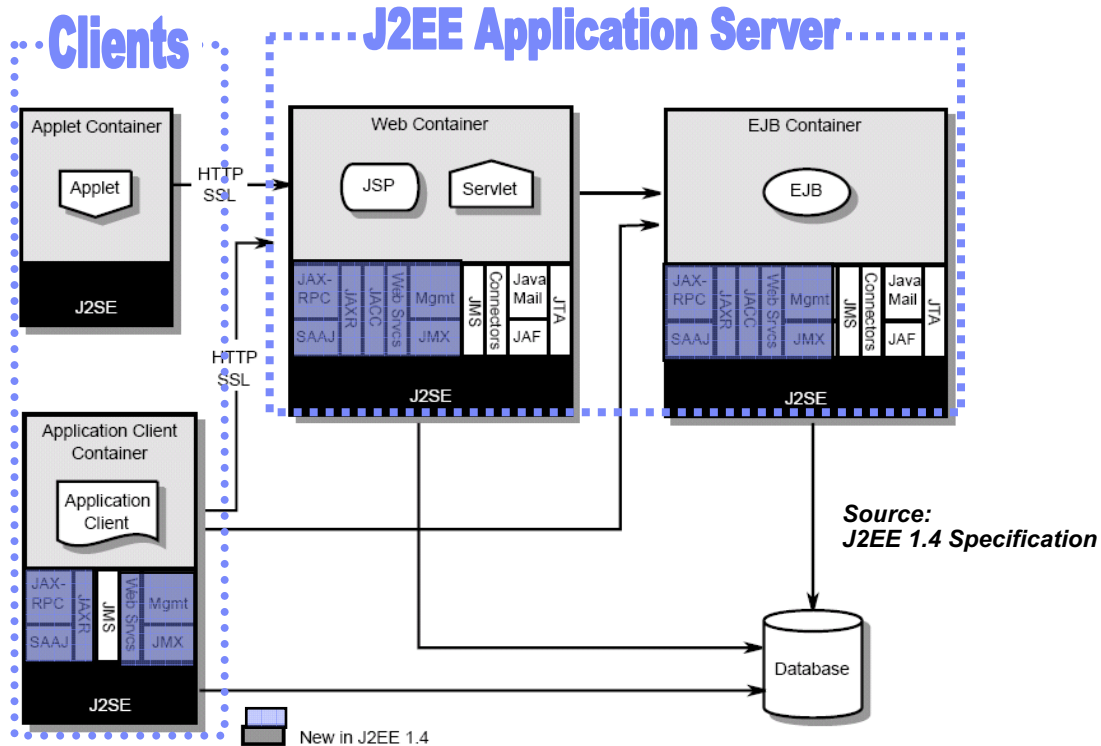
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: J2EE 1.4 platform architecture

# J2EE 1.4 platform architecture



© Copyright IBM Corporation 2007

Figure 2-3. J2EE 1.4 platform architecture

WA3616.3

## Notes:

The diagram shown on this slide appears in the J2EE 1.4 specification document found at the following URL <http://www.jcp.org/en/jsr/all>.

The J2EE platform offers a component-based model for building distributed, multi-tiered enterprise applications. Each application is typically made up of one or more J2EE components that work together in a distributed environment. The types of components that can make up a J2EE based application are:

Client tier components (Application Clients and Applets) – Run on a client machine

Web Components (Servlets and JSPs) – Run in the Web Container within the application server

Business Components (EJBs) – Run in the EJB Container within the Application Server

Each container shown in this diagram shows the type of application component that it supports, along with services that are available to the components by the container to support various application functions.

***Instructor notes:***

**Purpose —**

**Details —** Do not spend a lot of time discussing how Web applications work on this slide as it is coming up in the architecture topics.

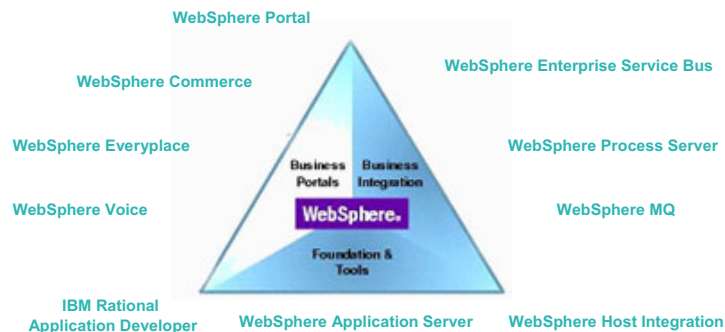
Also mention that in WebSphere Application Server V6.1 there are other containers which are not necessarily part of J2EE 1.4, such as the portlet and SIP containers.

**Additional information —**

**Transition statement —** Next: WebSphere software platform

## WebSphere software platform

- WebSphere is infrastructure software for dynamic e-business, delivering a secure and reliable software platform
- It provides:
  - Comprehensive e-business leadership
  - Tools to build, deploy and integrate your e-business
- It closes the gap between business strategy and information technology by allowing you to create and operate a dynamic e-business



© Copyright IBM Corporation 2007

Figure 2-4. WebSphere software platform

WA3616.3

### Notes:

IBM WebSphere is the leading software platform for e-business on demand. Successful industry leaders choose this adaptable, open platform to succeed in quickly and reliably delivering business results today knowing it gives them the freedom to grow their e-business in the future. Providing a full range of middleware software and support offerings, the WebSphere platform is infrastructure software that enables companies to develop, deploy, and integrate next-generation e-business applications from simple Web publishing through enterprise-scale transaction processing. WebSphere software transforms the way businesses manage customer, partner, and employee relationships. For example, you can use it to create compelling Web experiences, extend applications to incorporate mobile devices, and build electronic e-marketplaces. WebSphere software is all about the three fundamental aspects of e-business on demand:

**Foundation & Tools.** Rely upon a high quality foundation to rapidly build and deploy applications for high-performance e-business on demand.

**Business Portals.** Enhance customer, partner, employee, and supplier user experiences for optimal satisfaction.

Business Integration. Integrate applications and automate business processes for operational efficiency and business flexibility.

***Instructor notes:***

**Purpose** — The purpose of this visual is to present the WebSphere software platform.

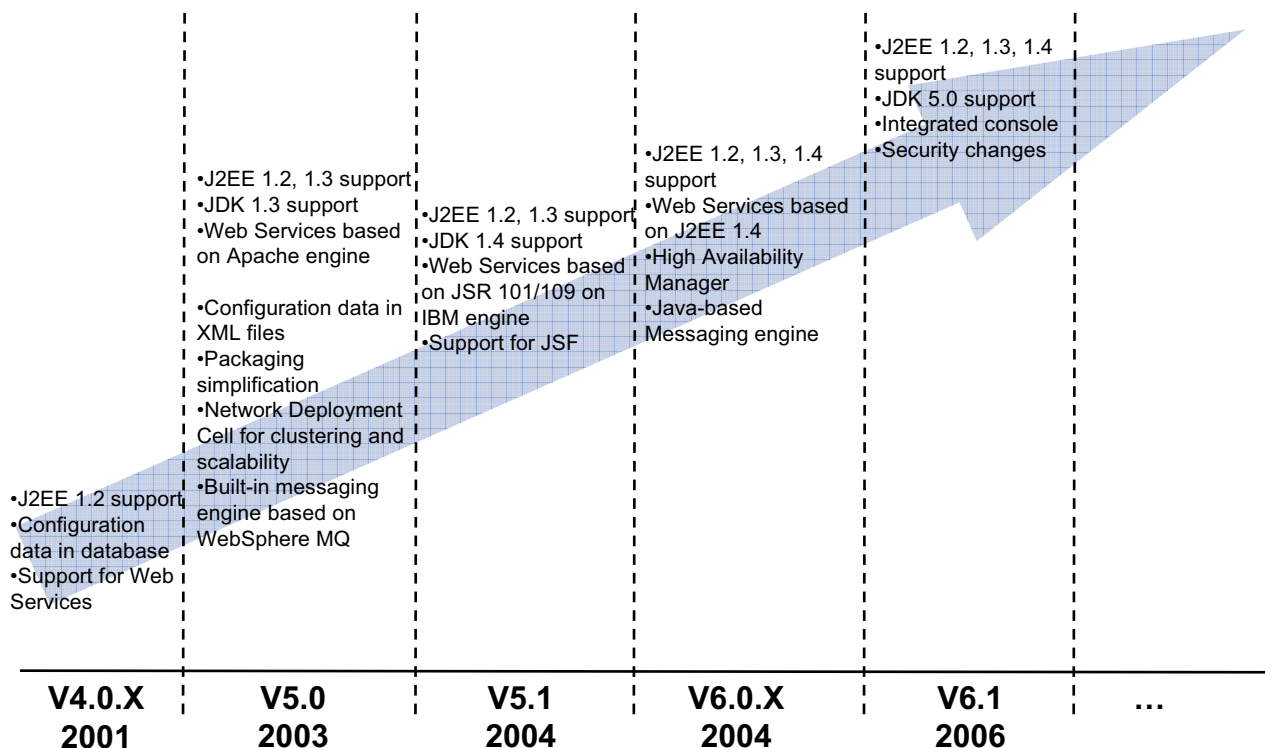
**Details** —

**Additional information** —

**Transition statement** — Next: WebSphere Application Server evolution



# WebSphere Application Server evolution



© Copyright IBM Corporation 2007

Figure 2-5. WebSphere Application Server evolution

WA3616.3

## Notes:

Reference: the WebSphere Application Server VC at <http://w3-03.ibm.com/software/xl/enablenow/en.nsf/doc/WHIL-6NDV35>

***Instructor notes:***

**Purpose —**

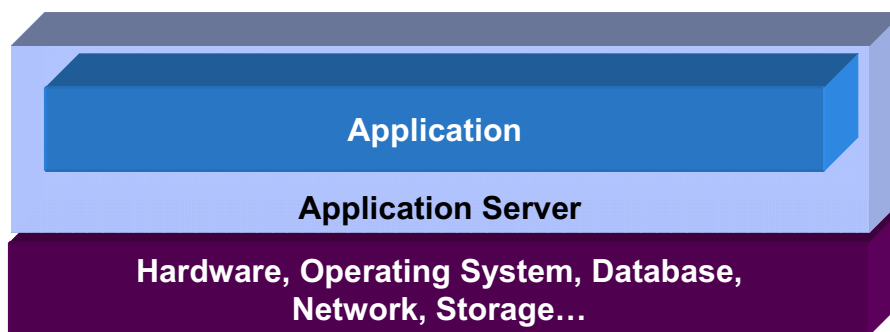
**Details —**

**Additional information —**

**Transition statement —** Next: What is an Application Server?

## What is an application server?

- Provides the infrastructure for running applications that run your business
  - Insulates applications from hardware, operating system, network...
  - Provides a common environment and programming model for your applications
    - Write once, run anywhere (J2EE)
    - Platform for developing and deploying Web Services
    - Platform for SOA based applications
  - Provides a scalable, reliable transaction engine for your enterprise



© Copyright IBM Corporation 2007

Figure 2-6. What is an Application Server?

WA3616.3

### Notes:

So what does an application server provide to make these challenges easier?

The primary role of an application server is to provide the infrastructure for running the applications that run your business. The infrastructure WebSphere Application Server provides insulates your applications from the hardware, operating systems, databases, etc. that they are running on. It enables developers to be shielded from the intricacies of the hardware or operating system, so they can focus on writing new business logic/applications that give you competitive advantage.

The application server also provides a secure, scalable and reliable environment for your applications to execute in, so developers do not have to spend time writing code to make each application secure, scalable and reliable. Developers can focus on building the business logic, and then by deploying their code to the application server, they automatically get those qualities of service because they are inherent in the application server infrastructure itself.

Fundamentally, an application server provides a common environment and programming model so developers can write their applications and have them run consistently, reliably,

and with efficient use of resources. It is this common environment for applications that is provided by the J2EE programming model. Applications written to comply with J2EE will run on WebSphere Application Server or other J2EE application servers unchanged – on whatever operating system the application server supports.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere Application Server V6.1

## WebSphere Application Server V6.1

---

- Provides the foundation for the WebSphere family of products
- Delivers a high performance and extremely scalable transaction engine for dynamic e-business applications
- Available in multiple packages
  - [WebSphere Application Server V6.1 Express](#)
  - [WebSphere Application Server V6.1 \(Base\)](#)
  - [WebSphere Application Server V6.1 Network Deployment](#)
- All product offerings provide the **same** core/base application server functionality

© Copyright IBM Corporation 2007

Figure 2-7. WebSphere Application Server V6.1

WA3616.3

### **Notes:**

The WebSphere software platform for e-business starts with a foundation that is formed from Web application serving and integration. IBM WebSphere Application Server software provides the core software to deploy, integrate, and manage your e-business applications.

WebSphere Application Server software supports custom-built applications that are either based on integrated WebSphere software platform products or on other third-party products. Such applications can range from dynamic Web presentations to sophisticated transaction processing systems.

IBM WebSphere Application Server Express, WebSphere Application Server, and WebSphere Application Server Network Deployment are incrementally related. As you scale up your e-business demand, you can also scale up your e-business capability by moving from one product to the next.

***Instructor notes:***

**Purpose** — The purpose of this visual is to describe the family of WebSphere Application Server products and the ship package for each product.

**Details** —

**Additional information** —

**Transition statement** — Next: Packaging

# Packaging

Platform	CD Image	Includes
Distributed	Primary	<ul style="list-style-type: none"> <li>• WebSphere Application Server</li> </ul>
	Installation Tools	<ul style="list-style-type: none"> <li>• IBM HTTP Server</li> <li>• Web server plug-ins</li> <li>• Application client</li> <li>• Update installer</li> <li>• Installation Factory</li> <li>• Migration tools</li> <li>• IBM Support Assistant</li> <li>• World Type fonts (Linux only)</li> </ul>
	WebSphere Application Server Toolkit (AST)	<ul style="list-style-type: none"> <li>• Eclipse-based tool for building, deploying, and debugging applications (Windows and Linux only)</li> <li>• Used for building and debugging administrative scripts</li> </ul>
© Copyright IBM Corporation 2007		

Figure 2-8. Packaging

WA3616.3

**Notes:**

For WebSphere Application Server V6.1 GA customers will have choice of CD or DVD media pack.



***Instructor notes:***

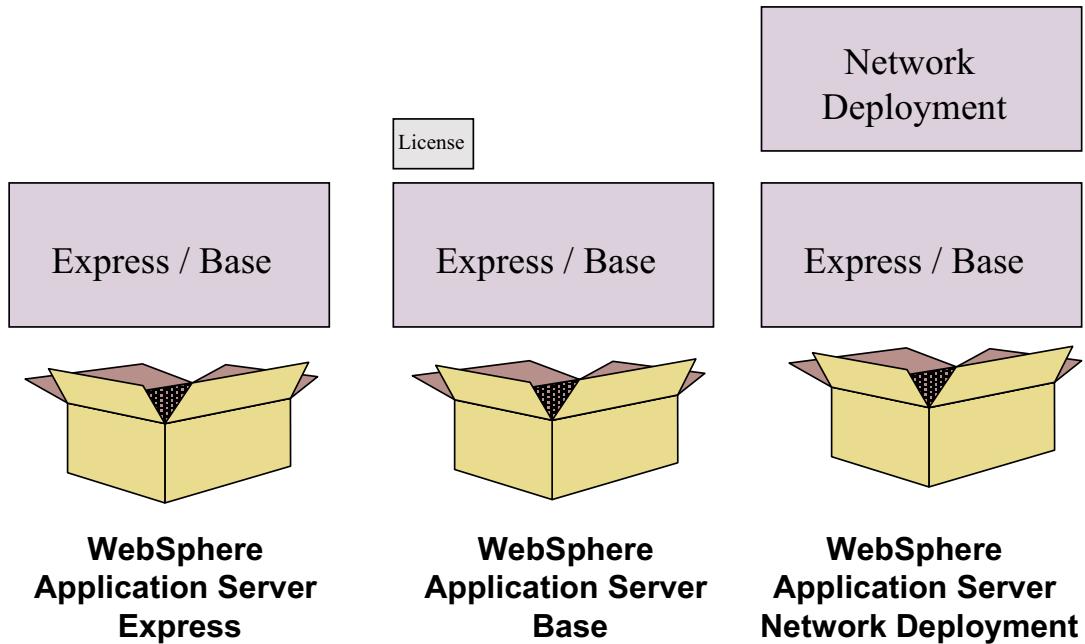
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere V6.1 packaging

## WebSphere V6.1 packaging



© Copyright IBM Corporation 2007

Figure 2-9. WebSphere V6.1 packaging

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Product packaging bundles

## Product packaging bundles

	WebSphere Application Server Express	WebSphere Application Server (Base)	WebSphere Application Server Network Deployment
Core Application Server	Standalone Node	Standalone Node	Deployment Manager, Standalone Node, Managed Node
IBM HTTP Server V6.1 Web Server plug-ins	x	x	x
Application Client	x	x	x
Data Direct JDBC Drivers	x	x	x
Development and Deployment Tools	Rational Web Developer AST	Rational Application Developer *Trial* AST	Rational Application Developer *Trial* AST
Database included in the package	DB2 Express (Dev. Use only)	DB2 Express (Dev. Use only)	DB2
Edge Components			x
IBM Tivoli Directory Server (LDAP server)			x
Tivoli Access Manager Server			x

© Copyright IBM Corporation 2007

Figure 2-10. Product packaging bundles

WA3616.3

### Notes:

This chart summarizes the various WebSphere Application Server V6.1 packages for production use.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere Application Server – Express/Base

## WebSphere Application Server – Express/Base

---

- Single server model
  - No clustering or multiserver management
- Full J2EE 1.4 support
  - Web services
  - Programming model extensions
  - J2SE 1.5 is supported for applications
- Ease of use
  - Provides quick deployment of new applications using WebSphere Rapid Deployment
- Quick installation
  - Only one CD per platform and single launch for runtime and tools
- Includes tight integration with IBM Rational tools
- Two pricing models
  - Per-CPU server pricing
  - Per-user pricing

© Copyright IBM Corporation 2007

Figure 2-11. WebSphere Application Server – Express/Base

WA3616.3

### **Notes:**

This slide summarizes the various WebSphere Application Server V6.1 packages for production use. WebSphere Application Server Express is geared to those who need to get started quickly with e-business. It is specifically targeted at medium-sized businesses or departments of a large corporation. It provides easy use and application development. It contains full J2EE 1.4 support but is limited to a single server environment. The WebSphere Application Server Express offering is unique in that it is bundled with an application development tool. Although there are Rational Web Developer and Rational Application Developer products that are designed to support each WebSphere Application Server package, these products are normally ordered independently of the server. WebSphere Application Server Express includes the Rational Web Developer application development tool. It provides a development environment geared toward Web developers and includes support for most J2EE 1.4 features with the exception of EJB and JCA development environments. However, keep in mind that WebSphere Application Server Express does contain full support for EJB and JCA, so you can deploy applications with them.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Differences – Express and Base offerings

## Differences – Express and Base offerings

Features	WebSphere Application Server V6.1 Express	WebSphere Application Server V6.1 (Base)
Licensing Terms	Limited to maximum of two CPUs	Unlimited CPUs
Development Tools	Rational Web Developer, AST	Rational Application Developer Trial, AST
Cosmetic changes	Launch pad, License terms, Install menus, product ID, and so on refer to WebSphere Application Server V6.1 - Express	Launch pad, License terms, Install menus, product ID, and so on. refer to WebSphere Application Server V6.1
Migration	V5 Express -> V6.1 Express	V5.x -> V6.1
Federating to Network Deployment	No software upgrade required, although it requires paper license upgrade	No software upgrade required, although it requires paper license upgrade
Edition Upgrade Path	Can be upgraded to WebSphere Application Server V6.1	N/A

© Copyright IBM Corporation 2007

Figure 2-12. Differences – Express and Base offerings

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere Application Server – Network Deployment

## WebSphere Application Server – Network Deployment

---

- Targets multiple servers, multiple nodes environment
  - Focus is on scalability, availability, and performance
  - Clustering and failover support
  - Web server plug-in supports weighted workload management
- Distributed administration
  - XML-based repository
  - Browser based administration console for the entire topology
- Edge Components
  - Load Balancer
  - Switch consultants for CISCO and Nortel switches
    - Site selector for scaling network dispatcher
  - Caching proxy
    - Reverse proxy, content-based routing
    - Dynamic content caching: integration with servlet caching
- Enhanced Web services functions
  - UDDI registry - for private UDDI registry
  - Web services gateway

© Copyright IBM Corporation 2007

Figure 2-13. WebSphere Application Server – Network Deployment

WA3616.3

### **Notes:**

Adding WebSphere Application Server V6.1 or WebSphere Application Server V6.1 Express to a network deployment cell is just a license upgrade – no additional code needed.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Important points

## Important points

---

- All V6.1 application servers versions have the same core and base functionality
- WebSphere Application Server Express V6.1 node can be upgraded to a WebSphere Application Server V6.1 node
  - Can be done without any software upgrades
  - Only a paper license upgrade is required
- WebSphere Application Server Express V6.1 and WebSphere Application Server V6.1 node can be federated into a WebSphere Application Server Network Deployment V6.1 cell
  - Can be done without any software upgrades
  - Only a paper license upgrade is required

© Copyright IBM Corporation 2007

Figure 2-14. Important points

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: J2EE 1.4 standards supported

## J2EE 1.4 standards supported

Standards	Level	WebSphere Application Server V6.1
J2EE	1.4	✓ Fully certified and part of Sun's JCEE list
EJB	2.1	✓ EJB 2.1 and EJB 2.0 support
J2SE	1.5	✓ J2SE V5.0
Servlet	2.4	✓ Servlet 2.4
JSP	2.0	✓ JSP 2.0
JTS/JTA	1.0.1	✓ With distributed transactions
JMS	1.1	✓ With Native Provider, and MQ plug-in
JDBC	3.0	✓ 2PC across heterogeneous databases
JNDI	1.2.1	✓ JNDI 1.2 for EJB lookup and CosNaming
RMI/IIOP	1.0	✓ Fully supported
JavaMail/JAF	1.3/1.0.2	✓ Plus Domino support
SSL Security	2.0	✓ JSSE and JCE
XML JAXP	1.2	✓ XML in EJBs
J-IDL/CORBA		✓ IIOP 1.2
J2C	1.5	✓ Bean and container managed
LDAP		✓ IBM Directory Server, iPlanet, ActiveDirectory
HTTP	1.1	✓ Yes, plus across multiple Web servers
SOAP	2.2.2	✓ SOAP support for Web services.
SOAP-SEC	1.0	
COM/ASP Support		✓ With Java wrapping and proxy
JMX	1.0	✓ JMX
XML4J	4.0	✓ XML support
XSL	2.3	✓ XSL parser

© Copyright IBM Corporation 2007

Figure 2-15. J2EE 1.4 standards supported

WA3616.3

### Notes:

For supported platforms and databases see the IBM WebSphere product Web page:  
<http://www.ibm.com/software/webservers/appserv/>

***Instructor notes:***

**Purpose —**

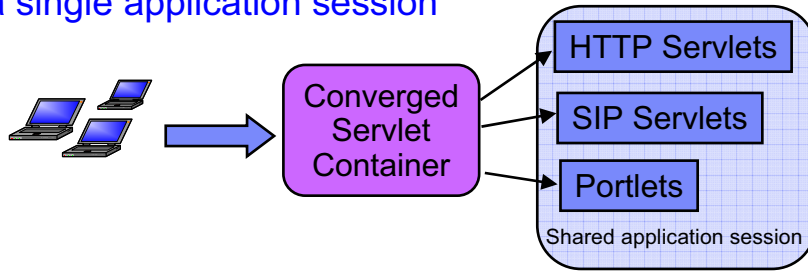
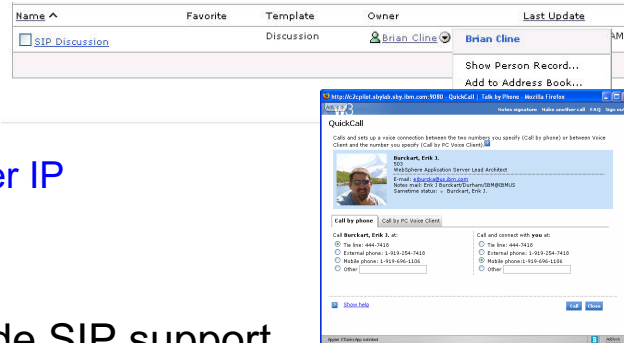
**Details —**

**Additional information —**

**Transition statement —** Next: Session Initiation Protocol (SIP) servlets

# Session Initiation Protocol (SIP) servlets

- The next generation of applications are being built on SIP
  - Chat and instant messaging
  - VoIP is built using SIP
  - Collaborative applications
  - Entertainment and gaming
  - Video conferencing, video over IP
- JSR 116 SIP Servlets provide SIP support
  - Servlets can share a single application session across protocols
  - HTTP servlets and portlets can create SIP messages and tie state together with a single application session



© Copyright IBM Corporation 2007

Figure 2-16. Session Initiation Protocol (SIP) servlets

WA3616.3

## Notes:



***Instructor notes:***

**Purpose —**

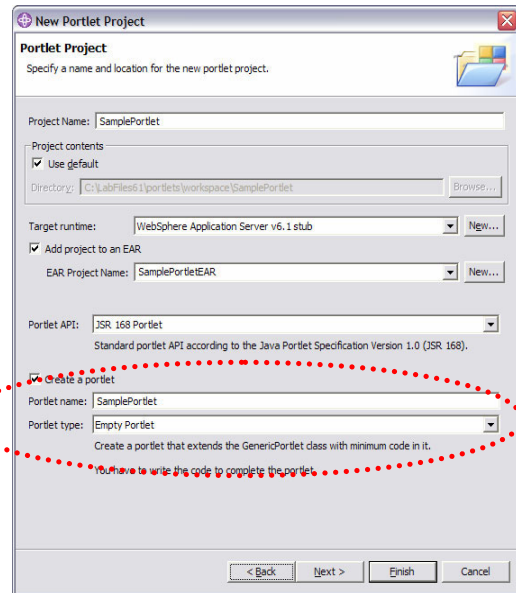
**Details —**

**Additional information —**

**Transition statement —** Next: Portlet support

# Portlet support

- Supports JSR 168 portlet programming model
  - Makes portlet programming model available in all WAS-based servers
- Tools support includes
  - Wizards to create
    - Portlet projects
    - Portlets
  - Editor to manage the portlet deployment descriptor (portlet.xml)
  - Ability to import portlet WAR file
- Limited aggregation through JSP tag library
  - Proprietary interface
  - Provides migration for aggregating servlets
- Limited remote request dispatch support provided
  - RemoteRequestDispatcher



The portlet project creation wizard will create an empty portlet for you and add it to the project

© Copyright IBM Corporation 2007

Figure 2-17. Portlet support

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: IBM HTTP Server

## IBM HTTP Server

---

- Based on Apache Web Server 2.0.47
  
- IBM enhancements
  - SSL support provides secure sessions
  - Advanced Cache speeds up serving static content
  - Machine Translation dynamically translates HTML files to another language
  - LDAP supports a common directory for user authentication and authorization
  - SNMP support monitors the IBM HTTP server from a network management console

© Copyright IBM Corporation 2007

Figure 2-18. IBM HTTP Server

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

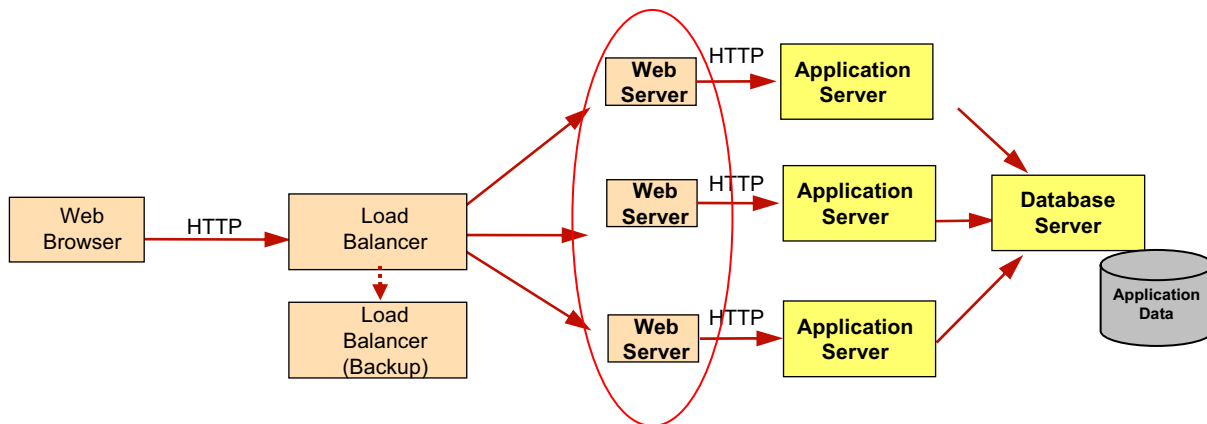
**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere Edge Components

## WebSphere Edge Components

- Addresses problems Web sites have with response time, scalability and reliability
- Edge Components include:
  - Load Balancer
  - Switch consultants for Cisco and Nortel switches
  - Caching proxy
  - Dynamic content caching



© Copyright IBM Corporation 2007

Figure 2-19. WebSphere Edge Components

WA3616.3

### Notes:

The Edge Components are geared for Internet Service Providers (ISP), whether they are in the ISP business or in the business of providing access to internal users of corporate information technology (IT). The Edge Components are designed to help you provide Internet access and content hosting. The main components are:

Caching proxy (formerly Web Traffic Express), a caching proxy server with Platform for Internet Content Selection (PICS) filtering. This provides highly scalable caching and filtering functions associated with receiving requests and serving URLs. With tunable caching capable of supporting high cache hit rates, the server can reduce bandwidth costs and provide more consistent rapid customer response times.

Load Balancer (formerly Network Dispatcher), load-balancing software with high availability features. It can dynamically monitor and balance requests to available TCP/IP servers and applications in real time. Load balancing allows heavily accessed Web sites to increase capacity by linking many individual servers to a single logical server.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: IBM Rational products

## IBM Rational products

- IBM Rational Web Developer
  - Replaces WebSphere Studio Site Developer (WSSD)
  - Bundled with WebSphere Application Server V6.1 – Express
- IBM Rational Application Developer
  - Replaces WebSphere Studio Application Developer (WSAD)
  - Trial version included with WebSphere Application Server V6.1 and WebSphere Application Server V6.1 Network Deployment

IBM Rational Tool	Supported J2EE Application	Target V6.1 Application Server Package	Comments
IBM Rational Application Developer	Full J2EE 1.4, 1.3 and 1.2 applications	All V6.1 packages	
IBM Rational Web Developer	J2EE 1.4, 1.3 and 1.2 applications, except no EJB and J2C development environment	All V6.1 packages	Subset of IRAD Can import J2EE 1.4 apps containing EJB, J2C and deploy

© Copyright IBM Corporation 2007

Figure 2-20. IBM Rational products

WA3616.3

### Notes:

Support for WebSphere Application Server V6.1 runtime will be available later with Rational products V7.

When IBM acquired Rational Development Corporation, IBM complemented its software portfolio with a world class set of design, model, development, and test tools. It was only natural to move the WebSphere Studio family of tools under the IBM Rational brand and begin integrating the great capabilities of WebSphere Studio with the quality tools in Rational and provide complete seamless development environment. IBM Rational Application/Web Developer is the follow-on product for WebSphere Studio Application/Site Developer and includes many of the best development features of the Rational products. IBM Rational Application/Web Developer is the new development tool for the applications destined for WebSphere Application Server as support is provided for building J2EE 1.2, 1.3, and 1.4 applications. WebSphere Application Server V6.1.0 is the platform which fully supports J2EE 1.4 applications and IBM Rational Application/Web Developer supports deployment to WebSphere Application Server V6.1.0 as well as deployment to WebSphere Application Server V4.0 and V5.0.



WebSphere Studio Site Developer and WebSphere Studio Application Developer renamed to Rational Web Developer and Rational Application Developer. No impact to underlying technology or design of the products. Rational is the “sole source” for core developer tools relating to the IBM SDP enabling IBM to streamline and simplify communications and offerings to better address customer needs

Rational owns IBM's strategy and execution for application development life cycle tools. IBM Software Development Platform is the basis of the strategy. Eclipse is the underlying framework for building and integrating the SDP. Rational Developer products will continue, as WebSphere Studio has in the past, to provide the best possible tools for developing and deploying applications to WebSphere.

WebSphere Rapid Deploy feature of Rational Application Developer V6.1 will make it easier than ever before to system test and deploy applications to WebSphere Application Server.

***Instructor notes:***

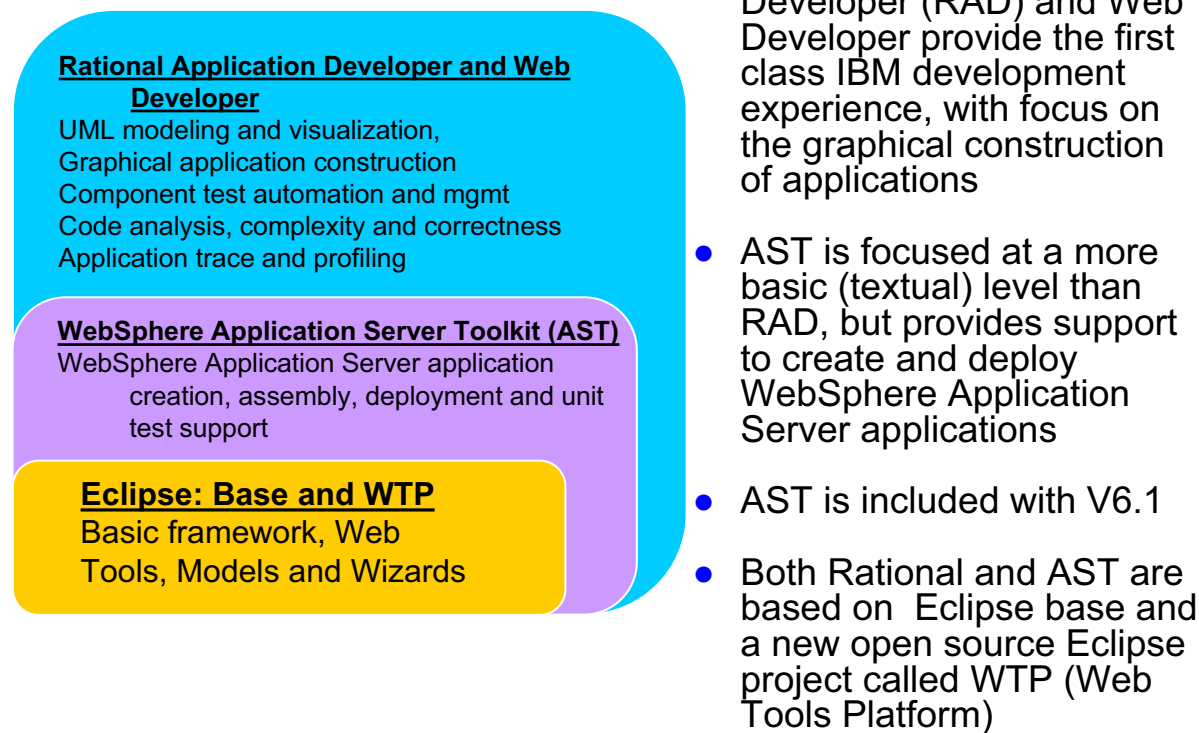
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Tooling

# Tooling



© Copyright IBM Corporation 2007

Figure 2-21. Tooling

WA3616.3

## Notes:

Rational V7.0 is available for support of WebSphere Application Server V6.1.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Checkpoint questions

---

## Checkpoint questions

---

1. WebSphere Application Server V6.1 currently supports which version of J2EE?
  - A. J2EE 1.2
  - B. J2EE 1.3
  - C. J2EE 1.4
  
2. Which products are included in the Network Deployment edition?
  - A. WebSphere Application Server
  - B. WebSphere Process Server
  - C. Edge Components
  - D. IBM HTTP Server
  
3. All application servers editions (Express, Base and ND) have the same base functionality.
  - A. True
  - B. False

© Copyright IBM Corporation 2007

Figure 2-22. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.

**Instructor notes:**

Purpose —

## Checkpoint solutions

---

1. WebSphere Application Server V6.1 currently supports which version of J2EE?  
C. J2EE 1.4
  
2. Which products are included in the Network Deployment edition?  
A. WebSphere Application Server  
C. Edge Components  
D. IBM HTTP Server
  
3. All application servers editions (Express, Base and ND) have the same base functionality.  
A. True

© Copyright IBM Corporation 2007

**Additional information** —

**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe the WebSphere family of products and the functions of each product
- Relate development and deployment products in the WebSphere family
- Describe WebSphere Application Server V6.1 packages
- Describe WebSphere Application Server V6.1 features and functions

© Copyright IBM Corporation 2007

Figure 2-23. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**



---

## Unit 3. WebSphere Application Server architectural overview: Stand-alone

### Estimated time

01:00

### What this unit is about

This unit provides an architectural overview of WebSphere Application Server V6.1 at run time.

### What you should be able to do

After completing this unit, you should be able to:

- Describe a typical e-business application architecture
- Describe the architectural differences between WebSphere Application Server packages
- Describe what is running in a WebSphere Application Server node
- Describe the architectural implications of the Web server plug-in
- Describe the use of JDBC providers and data sources
- Describe System Integration Bus, WebSphere Platform Messaging, and JMS
- Describe Managed Servers (Web servers such as IHS and others)
- Describe administrative interfaces and tools
- Describe High Availability Manager

### How you will check your progress

Accountability:

- Checkpoint

### References

SG24-7304 -00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center



## WebSphere Application Server architecture: Stand-alone

Figure 3-1. WebSphere Application Server architecture overview: Stand-alone

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

After completing this unit, you should be able to:

- Describe a typical e-business application architecture
- Describe the architectural differences between WebSphere Application Server packages
- Describe what is running in a WebSphere Application Server node
- Describe the architectural implications of the Web server plug-in
- Describe the use of JDBC providers and data sources
- Describe System Integration Bus, WebSphere Platform Messaging, and JMS
- Describe Managed Servers (Web servers such as IHS and others)
- Describe administrative interfaces and tools
- Describe High Availability Manager

© Copyright IBM Corporation 2007

Figure 3-2. Unit objectives

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Main themes

---

## Main themes

---

- Platform enablement
  - Developing the e-business operating system for enterprise integration
  - Foundation for other middleware products
  
- Ease of use
  - Reduced complexity, and increased integration of roles
    - Simplify development and deployment of applications with new WebSphere Rapid Deployment
  
- Enterprise class deployment
  - Building the next generation application server with the qualities of service required by enterprise class deployment
  
- Standards-based architecture and programming model
  - J2EE 1.4, Web services, Service Data Objects (SDO), and so forth

© Copyright IBM Corporation 2007

Figure 3-3. Main themes

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose** — You should set the stage for a discussion of architecture and reinforce the purpose of using WebSphere Application Server.

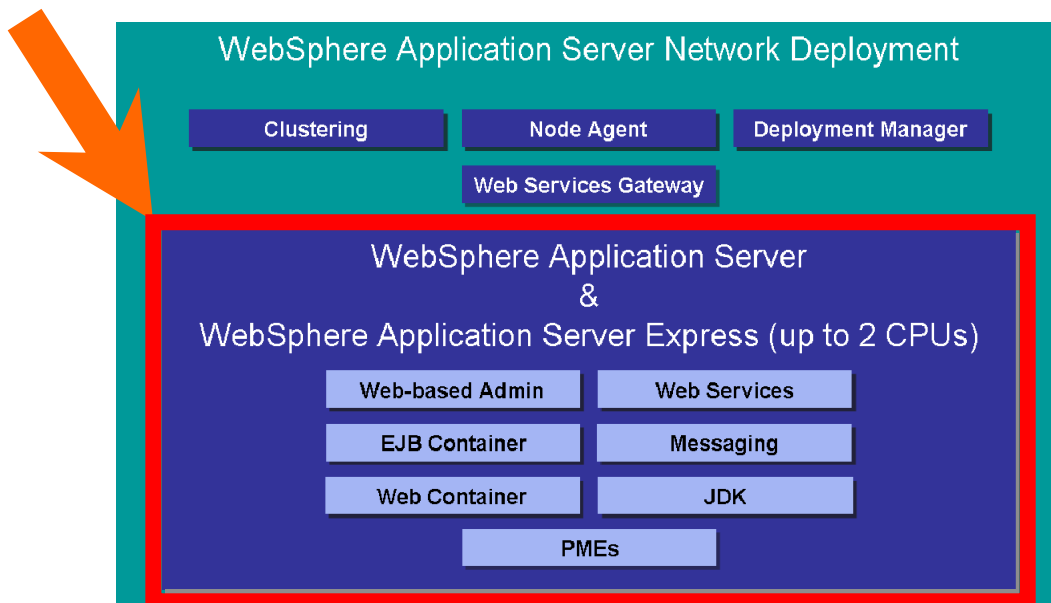
**Details** —

**Additional information** —

**Transition statement** — Next: Version 6 packaging



## Version 6 packaging



© Copyright IBM Corporation 2007

Figure 3-4. Version 6 packaging

WA3616.3

### Notes:

In this unit the focus will be on the Express and Base versions of WebSphere Application Server. All of the concepts that are discussed within this unit are applicable to all three versions: Express, Base, and Network Deployment.

***Instructor notes:***

**Purpose** — To set up the discussion of working with Express and Base versions of the product.

**Details** — You may wish to point out that all topics within this unit apply to Network Deployment as well.

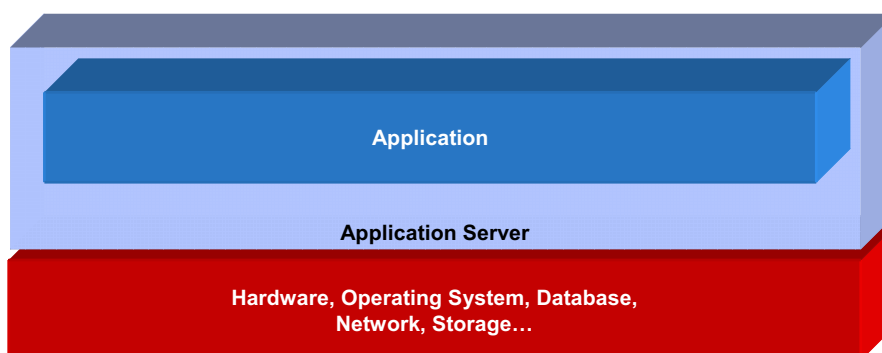
**Additional information** —

**Transition statement** — Next: WebSphere Application Server basics

## WebSphere Application Server basics

---

- WebSphere Application Server
  - Is a platform on which Java-based business applications run
  - Is an implementation of the Java 2 Enterprise Edition (J2EE) specification
  - Provides services (database connectivity, threading, workload management, and so forth) that can be used by the business applications



© Copyright IBM Corporation 2007

Figure 3-5. WebSphere Application Server basics

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

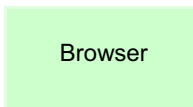
**Details** — You may wish to point out that if students have experience with other J2EE application servers that this is IBM's implementation of the J2EE specifications.

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime

## WebSphere architecture runtime

---



1 of 10

© Copyright IBM Corporation 2007

Figure 3-6. WebSphere architecture runtime

WA3616.3

### Notes:

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

The main element is the application server, a Java process that encapsulates many services, including the containers, where business logic executes. If you are familiar with J2EE, you will recognize the Web Container and the EJB container. The Web container executes Servlets and JavaServer Pages (JSPs), both of which are Java classes that generate markup to be viewed by a Web browser. Traffic into and out of the Web container travels through the embedded HTTP Server. While Servlets and JSPs can act independently, they most commonly make calls to Enterprise Java Beans (EJBs) to execute business logic or access data. EJBs, which run in the EJB container, are easily reusable Java classes. They most commonly communicate with a relational database or other external source of application data, either returning that data to the Web container or making changes to the data on behalf of the servlet or JSP.

The JMS messaging engine is built into the application server. This is a pure-Java messaging engine. JMS destinations, known as queues and topics provide asynchronous

messaging services to the code running inside the containers. JMS will be covered in more depth later in this course.

As you will see in more detail later on, the Web services engine enables application components to be exposed as Web services, which can be accessed using Simple Object Access Protocol (SOAP).

Several other services run within the application server, including the dynamic cache, data replication, security, and others. These will be covered later in the course.

There are also some important components outside of the application server process.

WebSphere Application Server also provides a plug-in for HTTP servers that determines what HTTP traffic is intended to be handled by WebSphere, and routes the requests to the appropriate server. The plug-in is also a critical player in workload management of HTTP requests, as it can distribute the load to multiple application servers, as well as steer traffic away from unavailable servers. It too reads its configuration from a special XML file.

**Instructor notes:**

**Purpose** — To show the runtime architecture of a WebSphere Application Server.

**Details** — 1 of 10: The browser is the main interaction mechanism that users will use.

2 of 10: A browser communicates with a Web server (HTTP Server).

3 of 10: The way the request gets into the WebSphere Application Server is from the HTTP Server Plug-in that is loaded with the HTTP Server. This request is forwarded to the Embedded HTTP server within the application server. The embedded server forwards the request into the Web container to either a servlet or a JSP.

4 of 10: If these servlets or JSPs wish to access distributed business logic or a database, the J2EE way to do this is via EJBs within the EJB Container.

5 of 10: EJBs (entity in this case) can communicate with the database to store, retrieve, query and delete data. JDBC is one way that this communication can occur.

6 of 10: There are many other services that are provided within the WebSphere Application Server. Some of those services are listed here.

7 of 10: The browser may communicate directly with the embedded HTTP server (bypassing the external Web server) this should only be used for testing and development purposes. This is the way that you will access your application servers in many of the lab exercises.

8 of 10: Browsers are not the only clients, a pure Java client can access EJBs directly via RMI/IIOP.

9 of 10: Web services clients can also access your application server. There are essentially two ways that this can happen: 1) via SOAP over HTTP and pass through the embedded HTTP server, and 2) communicating directly to the messaging engine within the application server.

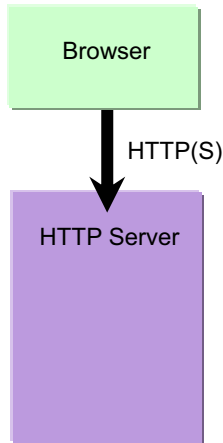
10 of 10: Finally, you can have a JMS client that directly communicates with the messaging engine.

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime

## WebSphere architecture runtime

---



**2 of 10**

© Copyright IBM Corporation 2007

Figure 3-7. WebSphere architecture runtime

WA3616.3

### **Notes:**

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

1 of 10: The browser is the main interaction mechanism that users will use.

2 of 10: A browser communicates with a Web server (HTTP Server).

3 of 10: The way the request gets into the WebSphere Application Server is from the HTTP Server Plug-in that is loaded with the HTTP Server. This request is forwarded to the Embedded HTTP server within the application server. The embedded server forwards the request into the Web container to either a servlet or a JSP.

4 of 10: If these servlets or JSPs wish to access distributed business logic or a database, the J2EE way to do this is via EJBs within the EJB Container.

5 of 10: EJBs (entity in this case) can communicate with the database to store, retrieve, query and delete data. JDBC is one way that this communication can occur.

6 of 10: There are many other services that are provided within the WebSphere Application Server. Some of those services are listed here.



7 of 10: The browser may communicate directly with the embedded HTTP server (bypassing the external Web server) this should only be used for testing and development purposes. This is the way that you will access your application servers in many of the lab exercises.

8 of 10: Browsers are not the only clients, a pure Java client can access EJBs directly via RMI/IIOP.

9 of 10: Web services clients can also access your application server. There are essentially two ways that this can happen: 1) via SOAP over HTTP and pass through the embedded HTTP server, and 2) communicating directly to the messaging engine within the application server.

10 of 10: Finally, you can have a JMS client that directly communicates with the messaging engine.

***Instructor notes:***

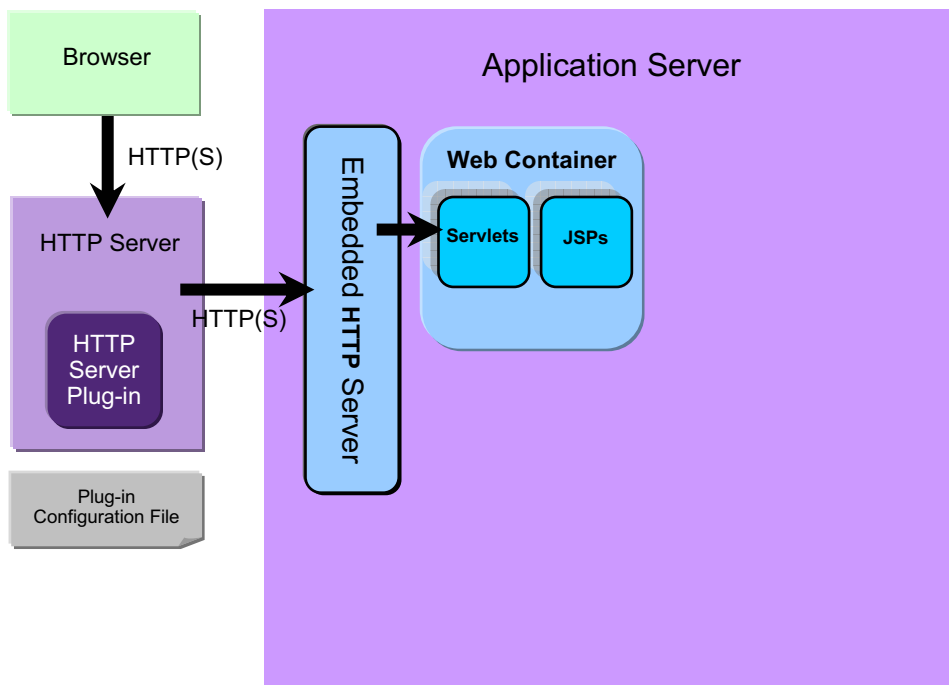
**Purpose** — To show the runtime architecture of a WebSphere Application Server.

**Details** —

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime

## WebSphere architecture runtime



**3 of 10**

© Copyright IBM Corporation 2007

Figure 3-8. WebSphere architecture runtime

WA3616.3

### Notes:

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

3 of 10: The way the request gets into the WebSphere Application Server is from the HTTP Server Plug-in that is loaded with the HTTP Server. This request is forwarded to the Embedded HTTP server within the application server. The embedded server forwards the request into the Web container to either a servlet or a JSP.

***Instructor notes:***

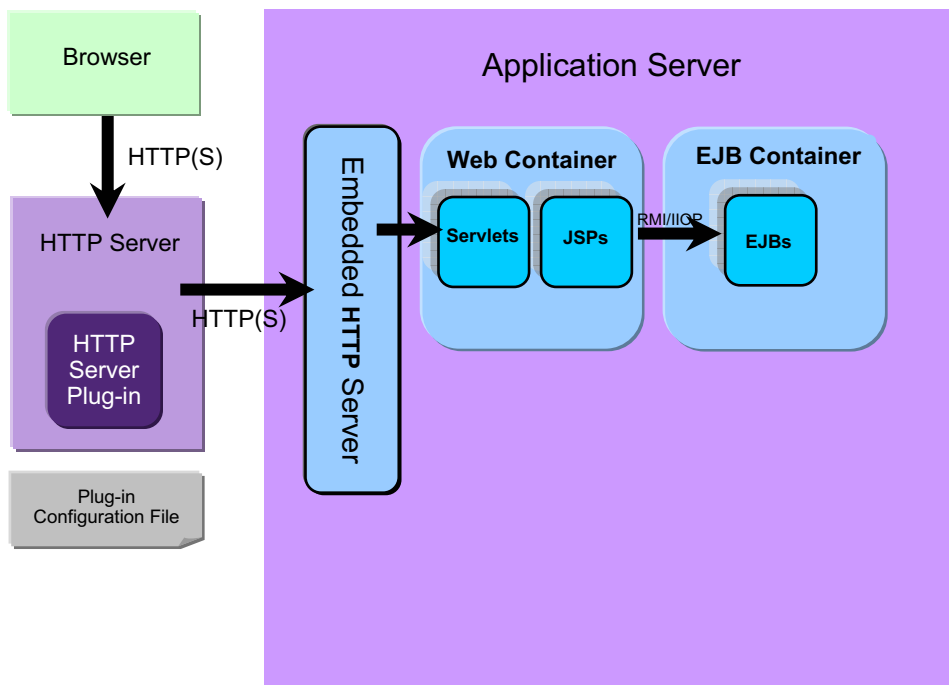
**Purpose** — To show the runtime architecture of a WebSphere Application Server.

**Details** —

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime

## WebSphere architecture runtime



**4 of 10**

© Copyright IBM Corporation 2007

Figure 3-9. WebSphere architecture runtime

WA3616.3

### Notes:

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

4 of 10: If these servlets or JSPs wish to access distributed business logic or a database, the J2EE way to do this is via EJBs within the EJB Container.

***Instructor notes:***

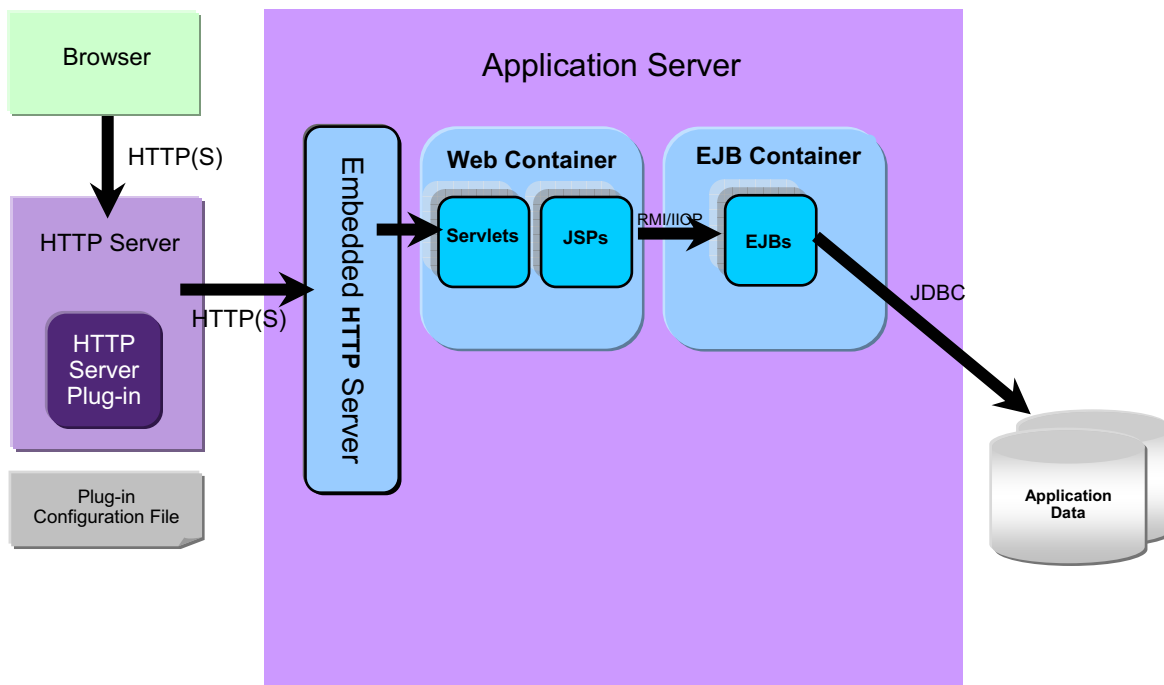
**Purpose** — To show the runtime architecture of a WebSphere Application Server.

**Details** —

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime

## WebSphere architecture runtime



5 of 10

© Copyright IBM Corporation 2007

Figure 3-10. WebSphere architecture runtime

WA3616.3

### Notes:

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

5 of 10: EJBs (entity in this case) can communicate with the database to store, retrieve, query and delete data. JDBC is one way that this communication can occur.

***Instructor notes:***

**Purpose** — To show the runtime architecture of a WebSphere Application Server.

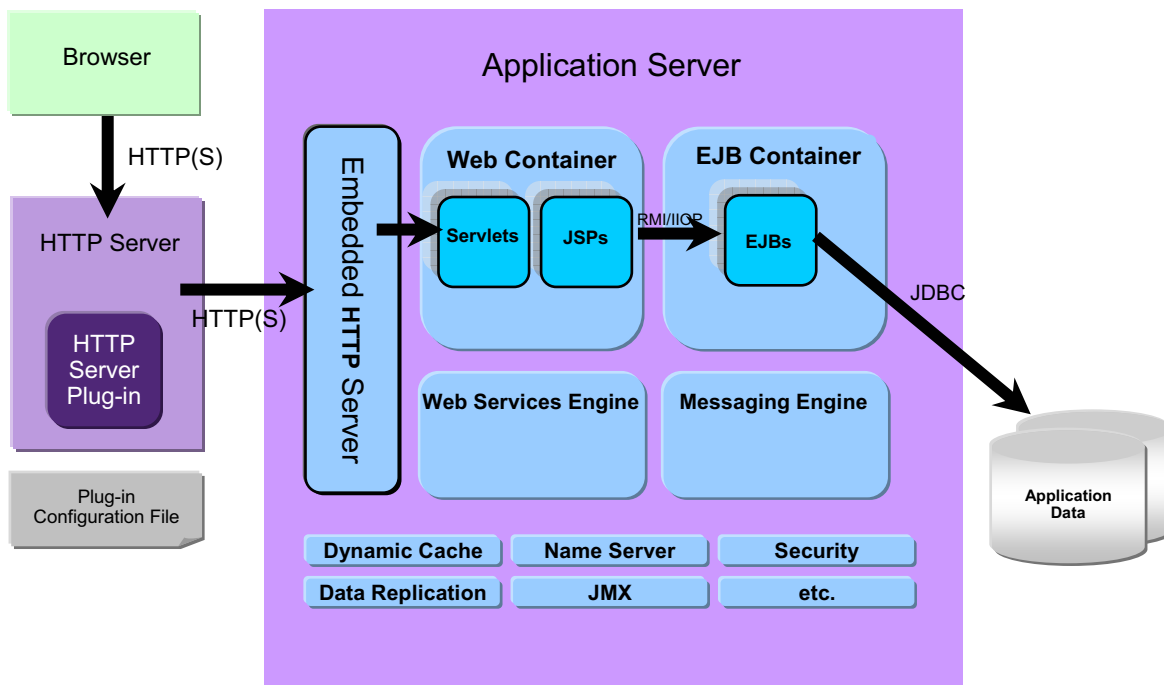
**Details** —

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime



## WebSphere architecture runtime



6 of 10

© Copyright IBM Corporation 2007

Figure 3-11. WebSphere architecture runtime

WA3616.3

### Notes:

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

6 of 10: There are many other services that are provided within the WebSphere Application Server. Some of those services are listed here.

***Instructor notes:***

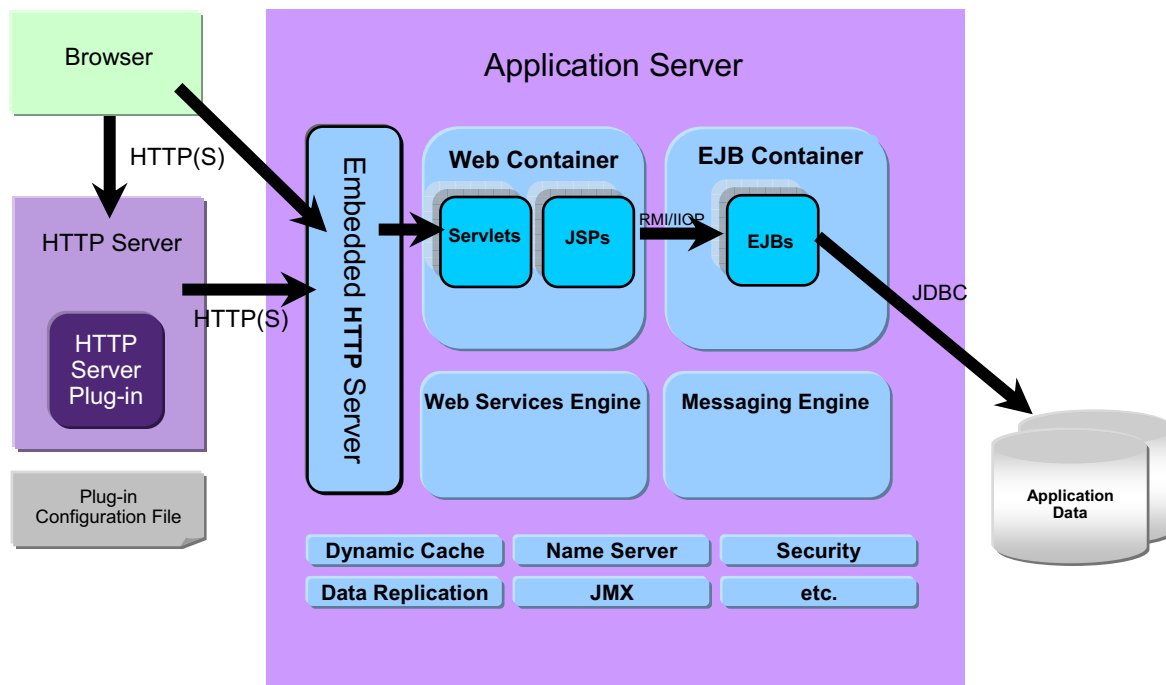
**Purpose** — To show the runtime architecture of a WebSphere Application Server.

**Details** —

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime

## WebSphere architecture runtime



**7 of 10**

© Copyright IBM Corporation 2007

Figure 3-12. WebSphere architecture runtime

WA3616.3

### Notes:

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

7 of 10: The browser may communicate directly with the embedded HTTP server (bypassing the external Web server) this should only be used for testing and development purposes. This is the way that you will access your application servers in many of the lab exercises.

***Instructor notes:***

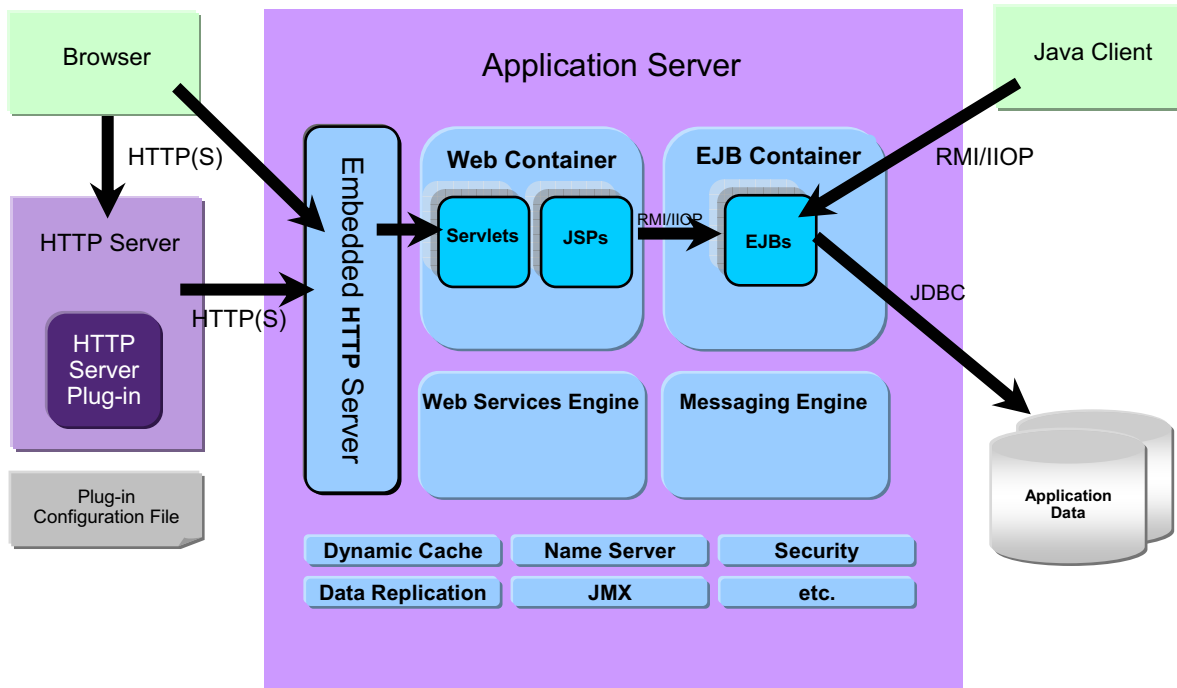
**Purpose** — To show the runtime architecture of a WebSphere Application Server.

**Details** —

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime

# WebSphere architecture runtime



8 of 10

© Copyright IBM Corporation 2007

Figure 3-13. WebSphere architecture runtime

WA3616.3

**Notes:**

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

8 of 10: Browsers are not the only clients, a pure Java client can access EJBs directly via RMI/IIOP.

***Instructor notes:***

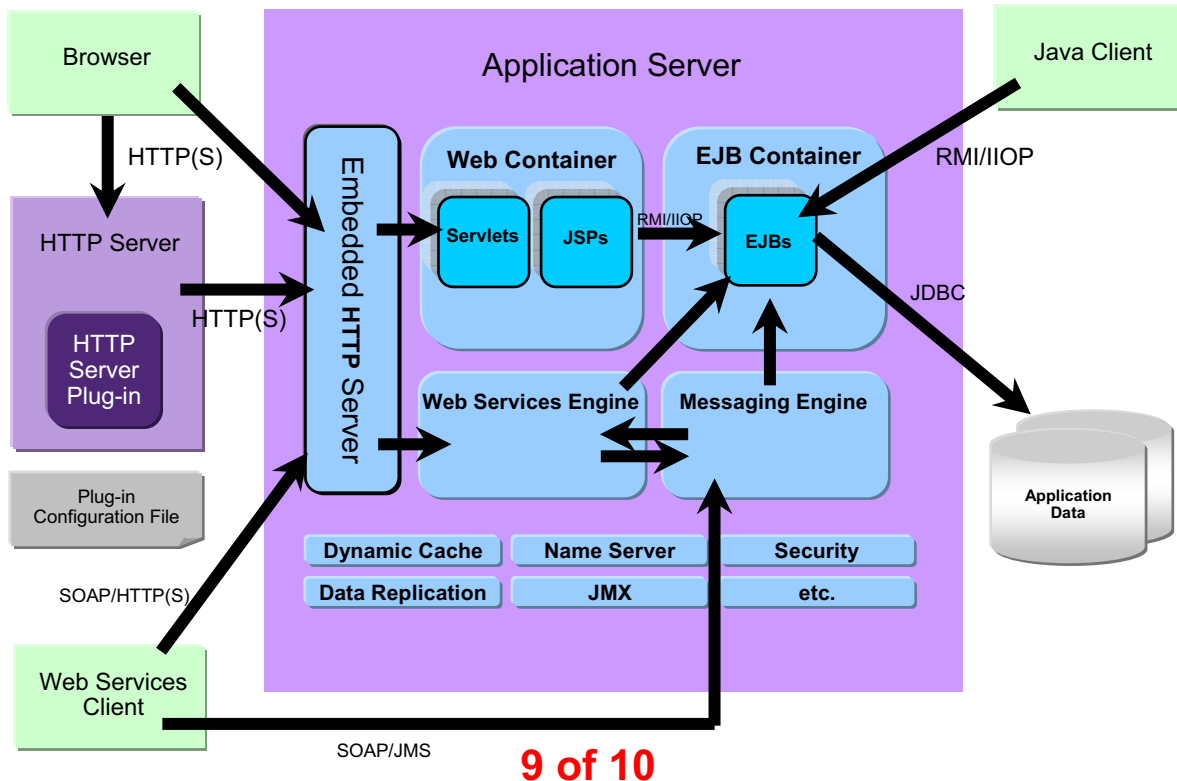
**Purpose** — To show the runtime architecture of a WebSphere Application Server.

**Details** —

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime

# WebSphere architecture runtime



© Copyright IBM Corporation 2007

Figure 3-14. WebSphere architecture runtime

WA3616.3

### Notes:

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

9 of 10: Web services clients can also access your application server. There are essentially two ways that this can happen: 1) via SOAP over HTTP and pass through the embedded HTTP server, and 2) communicating directly to the messaging engine within the application server.

***Instructor notes:***

**Purpose** — To show the runtime architecture of a WebSphere Application Server.

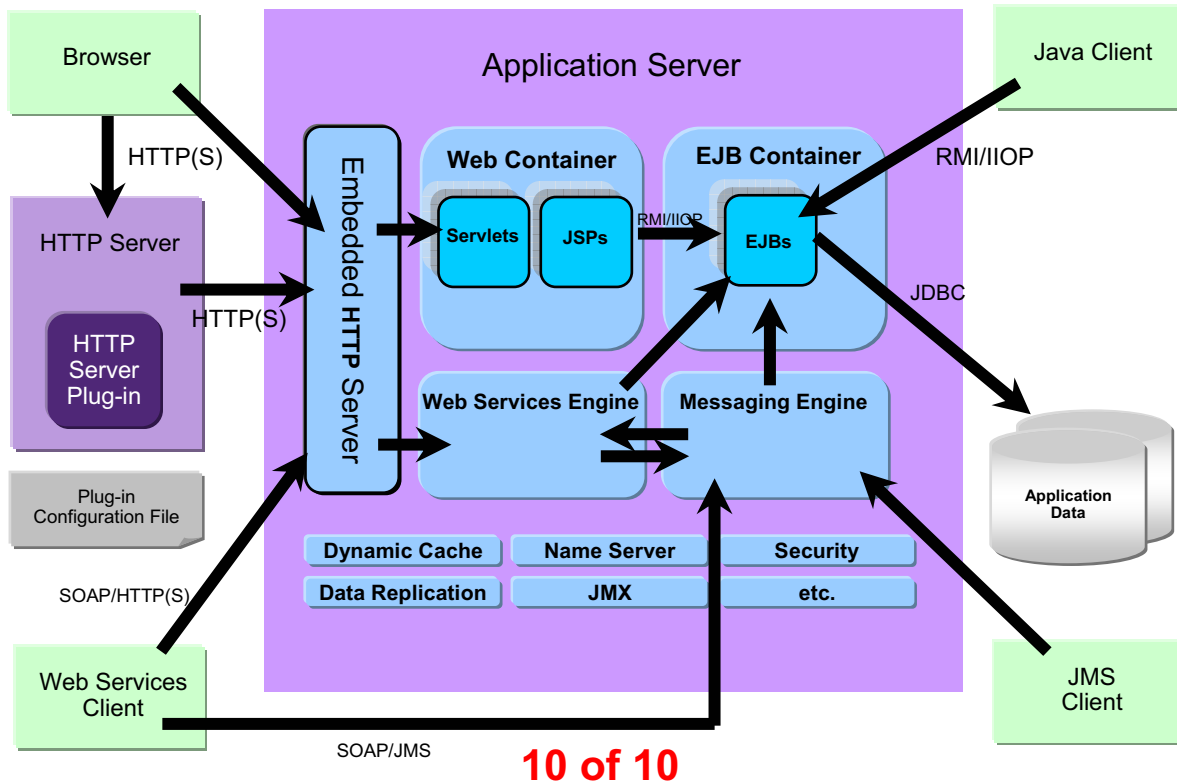
**Details** —

**Additional information** —

**Transition statement** — Next: WebSphere architecture runtime



# WebSphere architecture runtime



© Copyright IBM Corporation 2007

Figure 3-15. WebSphere architecture runtime

WA3616.3

## Notes:

This diagram illustrates the basic architecture of WebSphere Application Server, including several of the larger components.

10 of 10: Finally, you can have a JMS client that directly communicates with the messaging engine.

***Instructor notes:***

**Purpose** — To show the runtime architecture of a WebSphere Application Server.

**Details** —

**Additional information** —

**Transition statement** — Next: JDBC providers

## JDBC providers

- Provide the JDBC driver implementation for database access
  - Type 2 JDBC drivers (Thick)
    - Require the database client software on the client node to connect to the database server
  - Type 3 JDBC drivers (Net protocol)
    - Require server side code to map net protocol to native database
  - Type 4 JDBC drivers (Native protocol)
    - Connect directly to the database using its native protocol
  
- XA drivers support transaction recovery

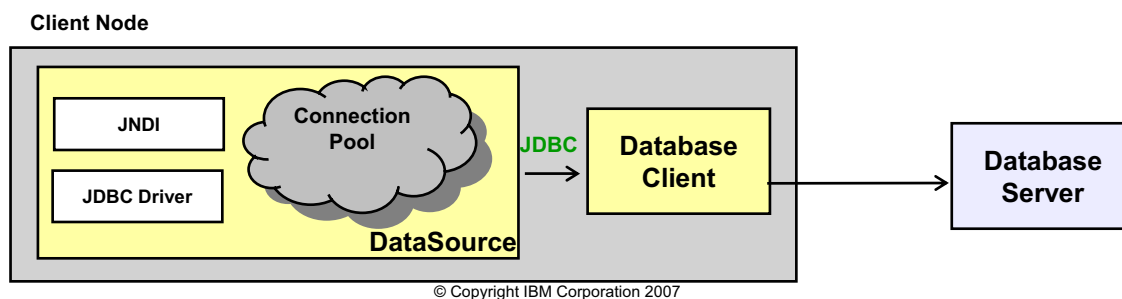


Figure 3-16. JDBC providers

WA3616.3

### Notes:

In order to access a database from with an application server, a JDBC provider is necessary. The JDBC provider allows access to the database.

***Instructor notes:***

**Purpose** — To point out that in order to communicate with a database, a JDBC provider is necessary.

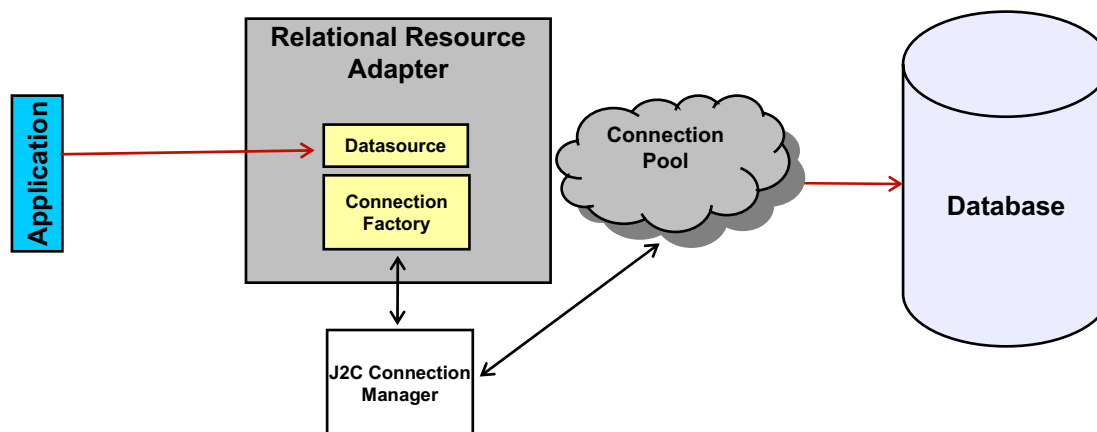
**Details** — Do not dwell on this, students will have the opportunity to create these within their labs. This is here because it is an important aspect of communication that is needed within the application server.

**Additional information** —

**Transition statement** — Next: Data sources

## Data sources

- Data sources can improve performance and portability for database access
  - Standard and XA data sources
- Connection pooling is provided by two parts:
  - J2C connection manager
  - Relational resource adapter



© Copyright IBM Corporation 2007

Figure 3-17. Data sources

WA3616.3

### Notes:

Rather than having the JDBC drivers directly communicate with the database, the communication is abstracted into a data source.

***Instructor notes:***

**Purpose** —

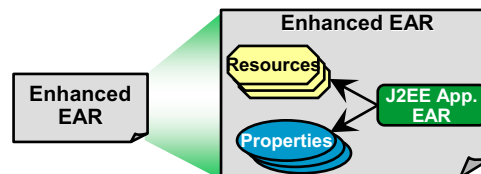
**Details** — Again, do not dwell here, merely point out some of the basic advantages of separating the interaction with the database from the JDBC drivers.

**Additional information** —

**Transition statement** — Next: Enhanced EAR

## Enhanced EAR

- Enterprise archive containing J2EE artifacts plus resource information needed to install on the WebSphere Application Server V6.
  - JDBC resources (datasources)
  - Class loader
  - JAAS authentication aliases
  - Shared libraries
  - Virtual host information
- Benefits: Improved productivity.
  - Application resources and properties come with the application.
  - Application install process creates the necessary resources within the server or cluster.
  - Moving application from one server to another also moves the resources.
- Support integrated with the IBM Rational Tooling (IRAD) and AST.
  - Found on Deployment page of application deployment descriptor
- **Warning:** Can possibly cause problems if unintended application scoped resources are used in production.
  - Enhancements can be removed or ignored during application installation.



© Copyright IBM Corporation 2007

Figure 3-18. Enhanced EAR

WA3616.3

### Notes:

This is IBM WebSphere specific. This is not part of J2EE. However, those artifacts that are defined within an enhanced EAR will be ignored if this EAR file is installed on a non-WebSphere application server.

Developers/administrators can define resources/properties within an enterprise application in tools and import/export Enhanced EAR.

Some resources will still need to be defined in the Application Server – like JMS, JavaMail, etc. Settings are defined in IBM tooling and stored on deployment.xml and packaged with EAR file. Resources applied at the new Application scope.

***Instructor notes:***

**Purpose** —

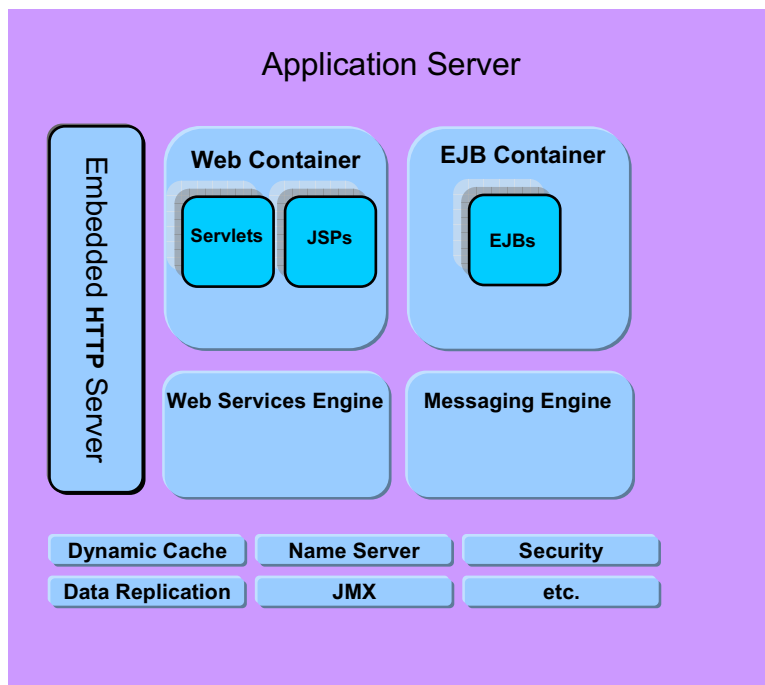
**Details** — Do not spend a lot of time here, students will work with EARs later. This is here to point out that the JDBC resources can be defined directly in the EAR file now.

**Additional information** —

**Transition statement** — Next: WebSphere architecture administration



# WebSphere architecture administration



1 of 4

© Copyright IBM Corporation 2007

Figure 3-19. WebSphere architecture administration

WA3616.3

## Notes:

Earlier, you saw the runtime depiction of a WebSphere Application Server, this diagram illustrates the basic architecture of administering WebSphere Application Server.

There are two main tools used to administer WebSphere Application Server: 1) the administrative console, and 2) wsadmin command line tool.

The server's configuration is stored in a set of XML files, often referred to as the configuration repository. These files define the server itself, as well as resources and services that it provides.

1 of 4: This is a standard application server as previously discussed.

2 of 4: One of the services provided within the application server is the admin service. This service allows for the ability to configure the application server. The files necessary for configuration are stored outside of the actual application server in a set of XML configuration files. There is an application that runs within the Web container that provides users the ability to administer the application server via a Web application – the admin console.

3 of 4: Here you see the communication from the browser all the way back to the XML configuration files.

4 of 4: wsadmin can be used to administer the application server in two ways:

1. Via SOAP by communicating with the embedded HTTP server, and
2. By using RMI (the default) to communicate directly with the admin service

***Instructor notes:***

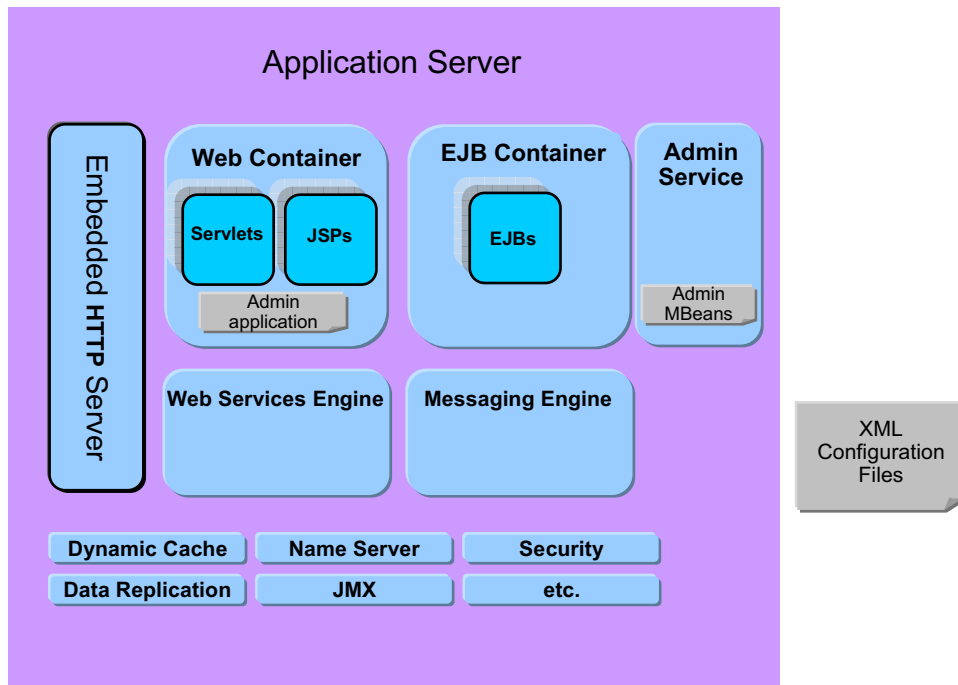
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere architecture administration

# WebSphere architecture administration



## 2 of 4

© Copyright IBM Corporation 2007

Figure 3-20. WebSphere architecture administration

WA3616.3

### Notes:

2 of 4: One of the services provided within the application server is the admin service. This service allows for the ability to configure the application server. The files necessary for configuration are stored outside of the actual application server in a set of XML configuration files. There is an application that runs within the Web container that provides users the ability to administer the application server via a Web application – the admin console.

***Instructor notes:***

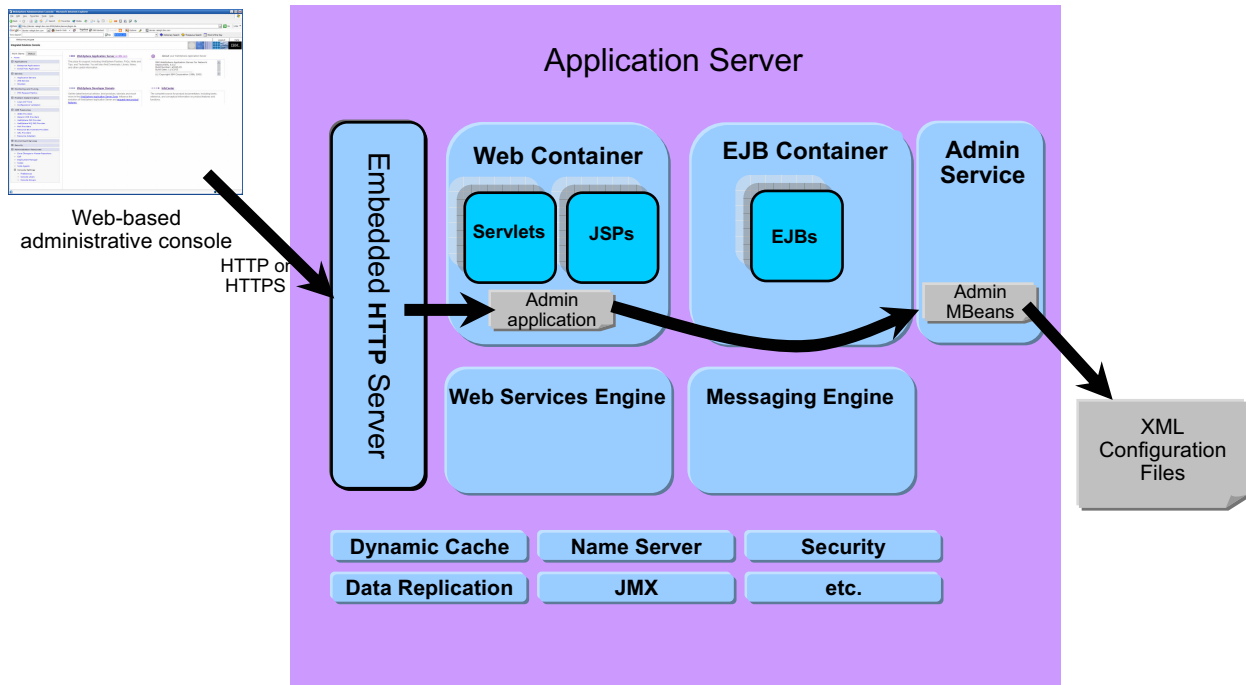
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere architecture administration

# WebSphere architecture administration



## 3 of 4

© Copyright IBM Corporation 2007

Figure 3-21. WebSphere architecture administration

WA3616.3

### Notes:

3 of 4: Here you see the communication from the browser all the way back to the XML configuration files.

***Instructor notes:***

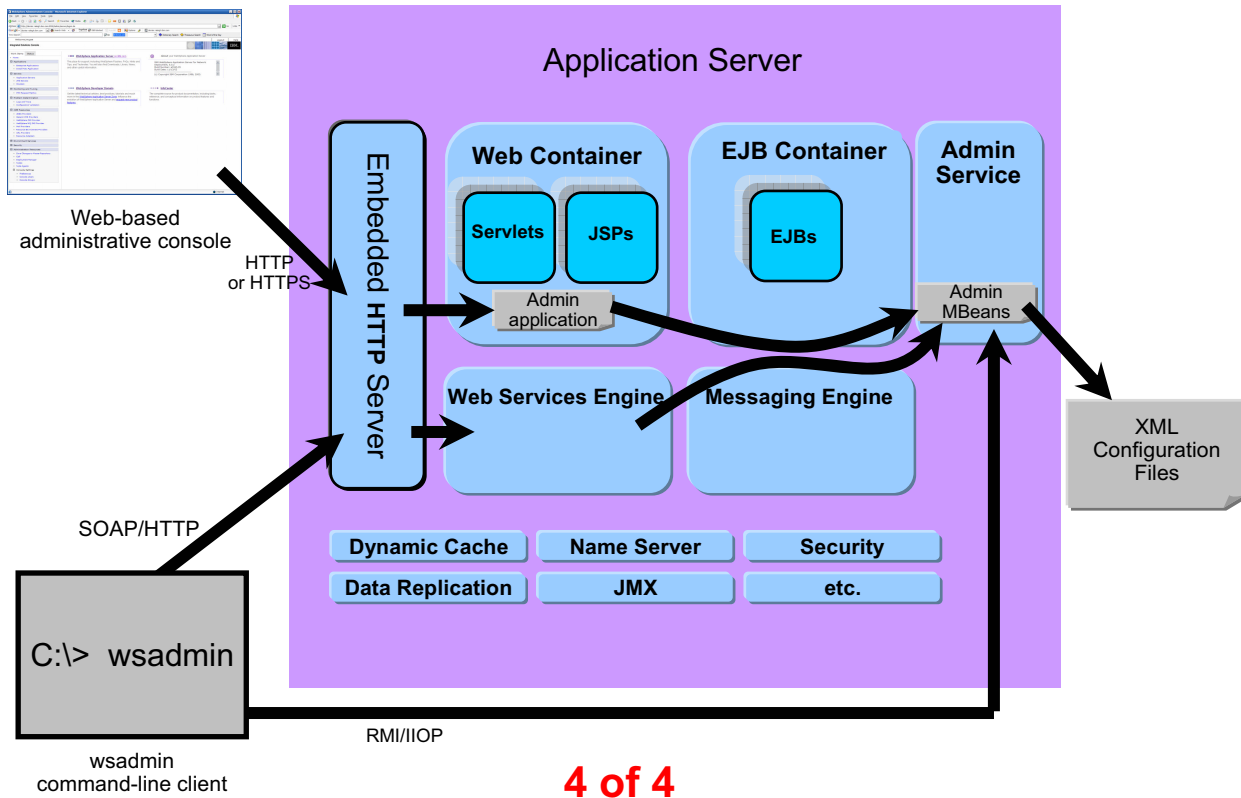
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere architecture administration

# WebSphere architecture administration



**4 of 4**

© Copyright IBM Corporation 2007

Figure 3-22. WebSphere architecture administration

WA3616.3

## Notes:

4 of 4: wsadmin can be used to administer the application server in two ways:

1. Via SOAP by communicating with the embedded HTTP server, and
2. By using RMI (the default) to communicate directly with the admin service



***Instructor notes:***

**Purpose —**

**Details —**

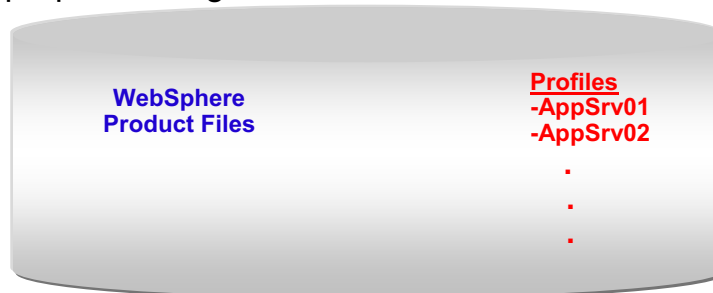
**Additional information —**

**Transition statement —** Next: WebSphere profiles overview

## WebSphere profiles overview

---

- Profiles are sets of files that represent a WebSphere Application Server configuration.
- WebSphere Application Server files are split into two categories:
  - **Product files**
    - Set of shared read-only static files or product binaries shared by any instances of the WebSphere Application Server product
  - **Configuration files (profiles)**
    - Set of user-customizable data files
    - Files include: WebSphere configuration, installed applications, resource adapters, properties, log files, and so forth



© Copyright IBM Corporation 2007

Figure 3-23. WebSphere profiles overview

WA3616.3

### **Notes:**

Profiles are the way that you are allowed to run more than one application server on a single installation of WebSphere product files.

***Instructor notes:***

**Purpose —**

**Details —** The main purpose here is to for students to understand that the WebSphere product files (<was\_root>\...) only need to be installed once, and then in order to create an application server, they will be creating profiles.

**Additional information —**

**Transition statement —** Next: WebSphere profiles benefits

## WebSphere profiles benefits

- Benefits of profiles:
  - Each profile uses the same product files.
  - Simpler than multiple WebSphere installations.
    - Less disk space
    - Simplifies application of product updates

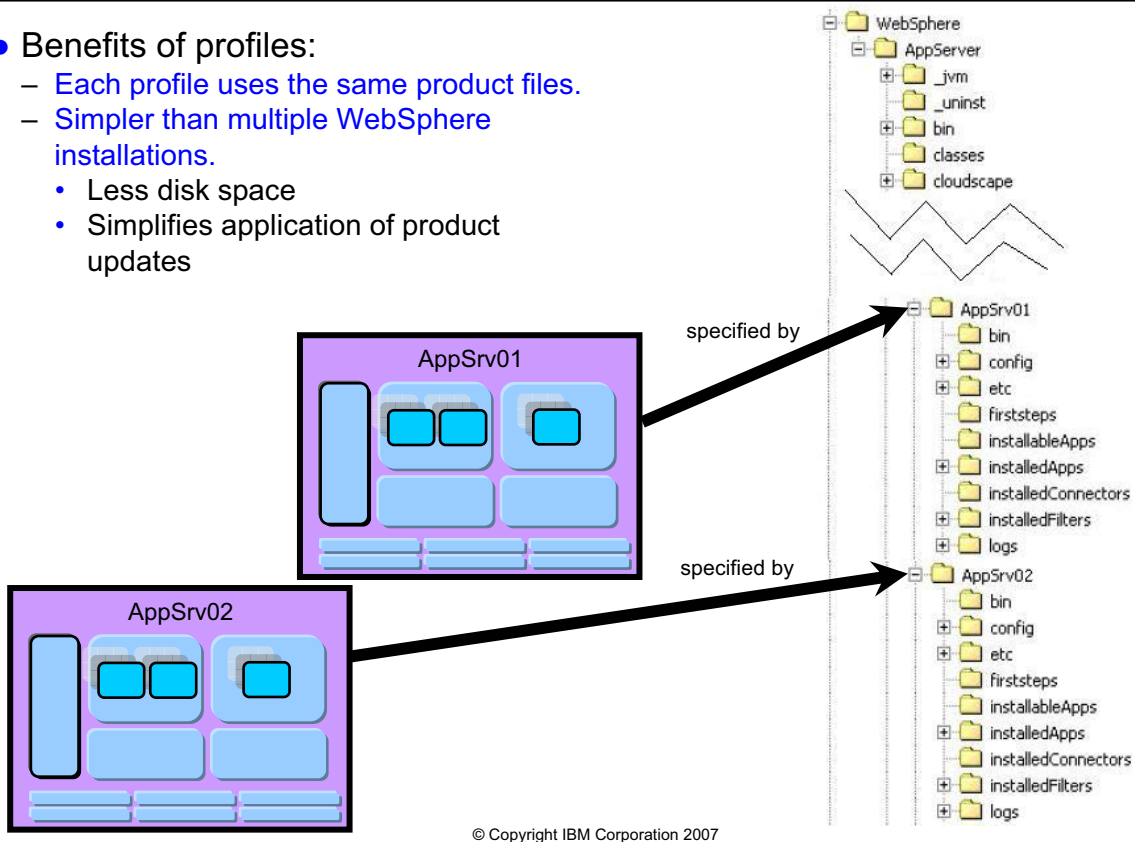


Figure 3-24. WebSphere profiles benefits

WA3616.3

### Notes:

Notice that under the WebSphere installation directory (<was\_root>) there are subdirectories for each profile. In the example above there are two application servers running that are each configure by the files that exist within their own profile directory.

***Instructor notes:***

**Purpose** —

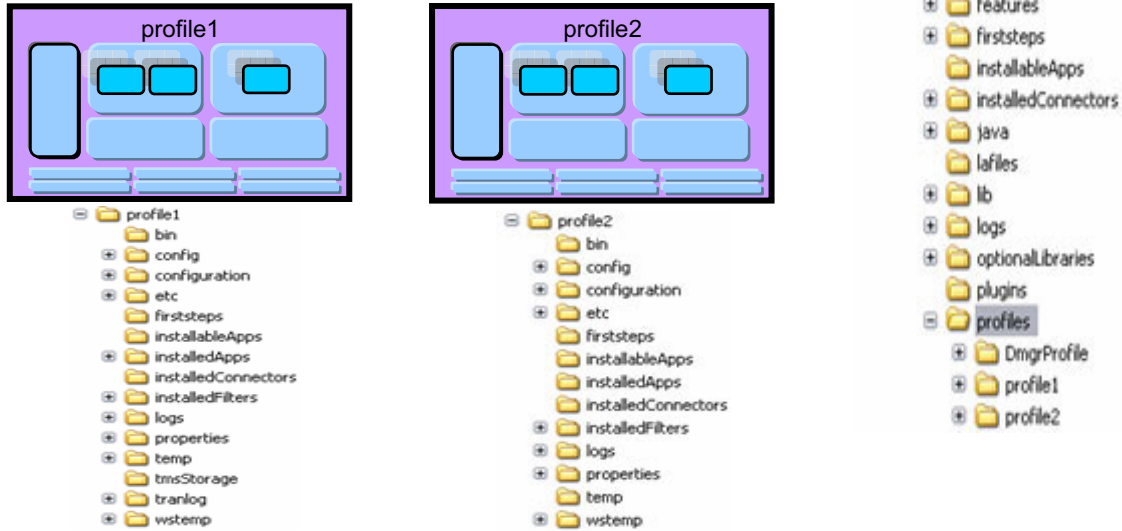
**Details** — To pictorially represent the relationship between a running application server and the files that specify their runtime configuration.

**Additional information** —

**Transition statement** — Next: WebSphere profiles benefits

# WebSphere profiles benefits

- Benefits of profiles:
  - Each profile uses the same product files.
  - Simpler than multiple WebSphere installations.
    - Less disk space
    - Simplifies application of product updates



© Copyright IBM Corporation 2007

Figure 3-25. WebSphere profiles benefits

WA3616.3

## Notes:

Notice that under the WebSphere installation directory (<was\_root>) there are subdirectories for each profile. In the example above there are two application servers running that are each configure by the files that exist within their own profile directory.

***Instructor notes:***

**Purpose** —

**Details** — To pictorially represent the relationship between a running application server and the files that specify their runtime configuration.

**Additional information** —

**Transition statement** — Next: Managing profiles

## Managing profiles

---

- Profiles should be managed through one of the tools provided.
  - **Profile Management tool (PMT) Wizard**
    - Eclipse-based GUI tool for creating profiles
    - Gathers user input and invokes the “manageprofiles” command line tool to create the profiles
    - Replaces v6.0 Profile Creation Tool
  - **manageprofiles script**
    - Command line interface for profile management functions.
    - V6.0 wasprofile command line tool has been enhanced and renamed to manageprofiles script in V6.1.

© Copyright IBM Corporation 2007

Figure 3-26. Managing profiles

WA3616.3

### **Notes:**

These are the two ways of creating and managing profiles.



***Instructor notes:***

**Purpose —**

**Details —** Note that the types of profiles that can be created within the wizard are not mentioned here because you are just currently discussing Express and Base. The deployment manager and custom profiles will be discussed in the next architecture unit.

**Additional information —**

**Transition statement —** Next: Checkpoint questions

## Checkpoint questions

---

1. Which of the following provides an environment for running servlets?
  - A. Client module
  - B. Web container
  - C. EJB module
  
2. Which type of JDBC driver is considered a “thick” driver?
  - A. Type 2
  - B. Type 3
  - C. Type 4
  
3. Which of the following are components contained within the application server’s JVM?
  - A. HTTP Server plug-in
  - B. Embedded HTTP Server
  - C. DB2 database

© Copyright IBM Corporation 2007

---

Figure 3-27. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.

**Instructor notes:**

Purpose —

Details —

## Checkpoint solutions

---

1. Which of the following provides an environment for running servlets?  
B. Web container
2. Which type of JDBC driver is considered a “thick” driver?  
A. Type 2
3. Which of the following are components contained within the application server’s JVM?  
B. Embedded HTTP Server

© Copyright IBM Corporation 2007

**Additional information** —**Transition statement** — Next: Unit summary

## Unit summary

---

After completing this unit, you should be able to:

- Describe a typical e-business application architecture
- Describe the architectural differences between WebSphere Application Server packages
- Describe what is running in a WebSphere Application Server node
- Describe the architectural implications of the Web server plug-in
- Describe the use of JDBC providers and data sources
- Describe System Integration Bus, WebSphere Platform Messaging, and JMS
- Describe Managed Servers (Web servers such as IHS and others)
- Describe administrative interfaces and tools
- Describe High Availability Manager

© Copyright IBM Corporation 2007

Figure 3-28. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**



# Unit 4. WebSphere Application Server architectural overview: Federated

## Estimated time

01:30

## What this unit is about

This unit describes the concepts and terminology of a Network Deployment installation.

## What you should be able to do

After completing this unit, you should be able to:

- Describe Network Deployment runtime flow
- Describe Network Deployment concepts and terminology such as cell, node, node agent, and deployment manager
- Describe Network Deployment administration flow
- Explain how to manage Web servers with WebSphere
- Describe high availability concepts and functions
- Describe data replication service concepts and functions
- Describe name service concepts and functions
- Describe platform messaging

## How you will check your progress

Accountability:

- Checkpoint

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

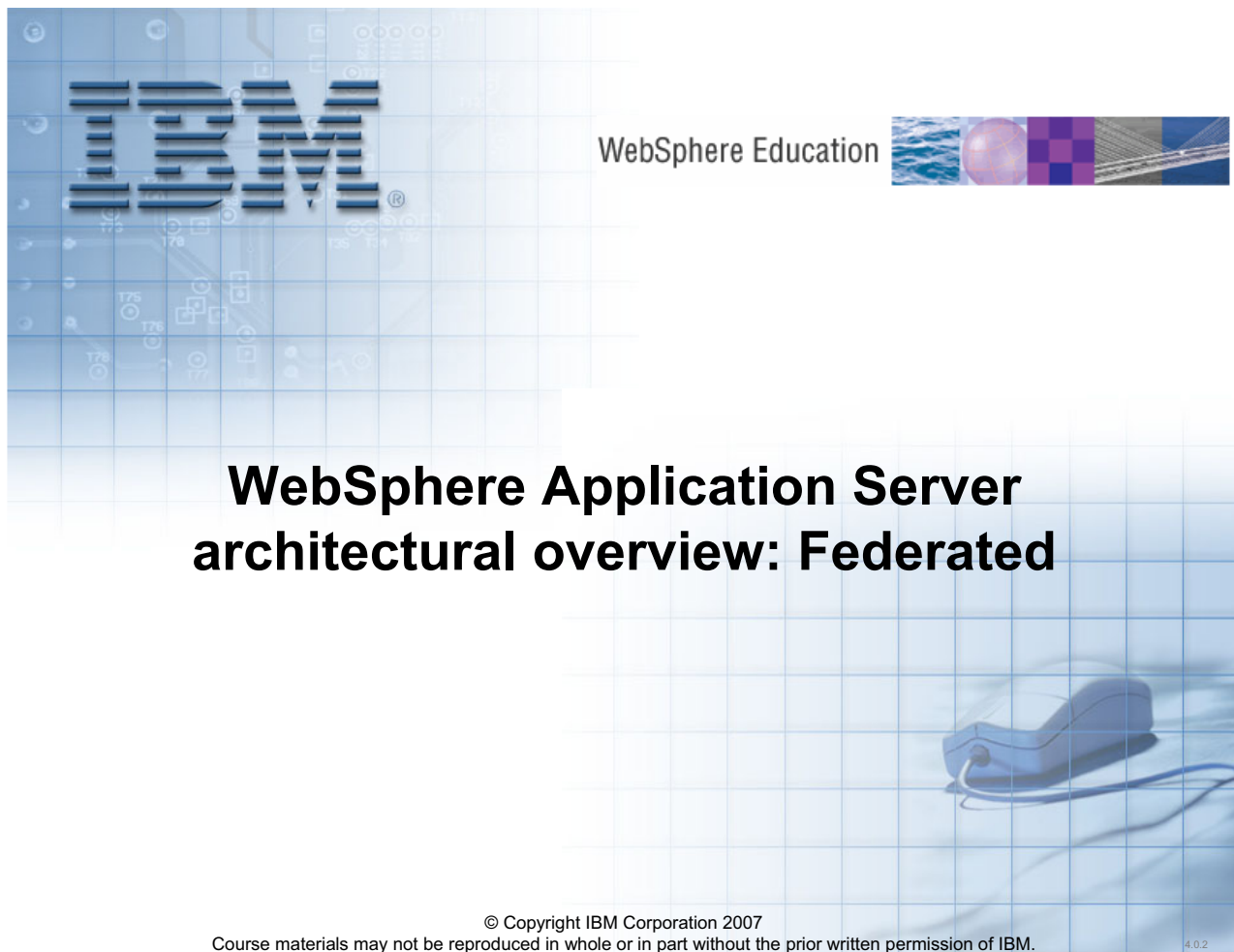


Figure 4-1. Architectural overview: Federated

WA3616.3

**Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Unit objectives

---

After completing this unit, you should be able to:

- Describe Network Deployment runtime flow
- Describe Network Deployment concepts and terminology such as cell, node, node agent, and deployment manager
- Describe Network Deployment administration flow
- Explain how to manage Web servers with WebSphere
- Describe high availability concepts and functions
- Describe data replication service concepts and functions
- Describe name service concepts and functions
- Describe platform messaging

© Copyright IBM Corporation 2007

Figure 4-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

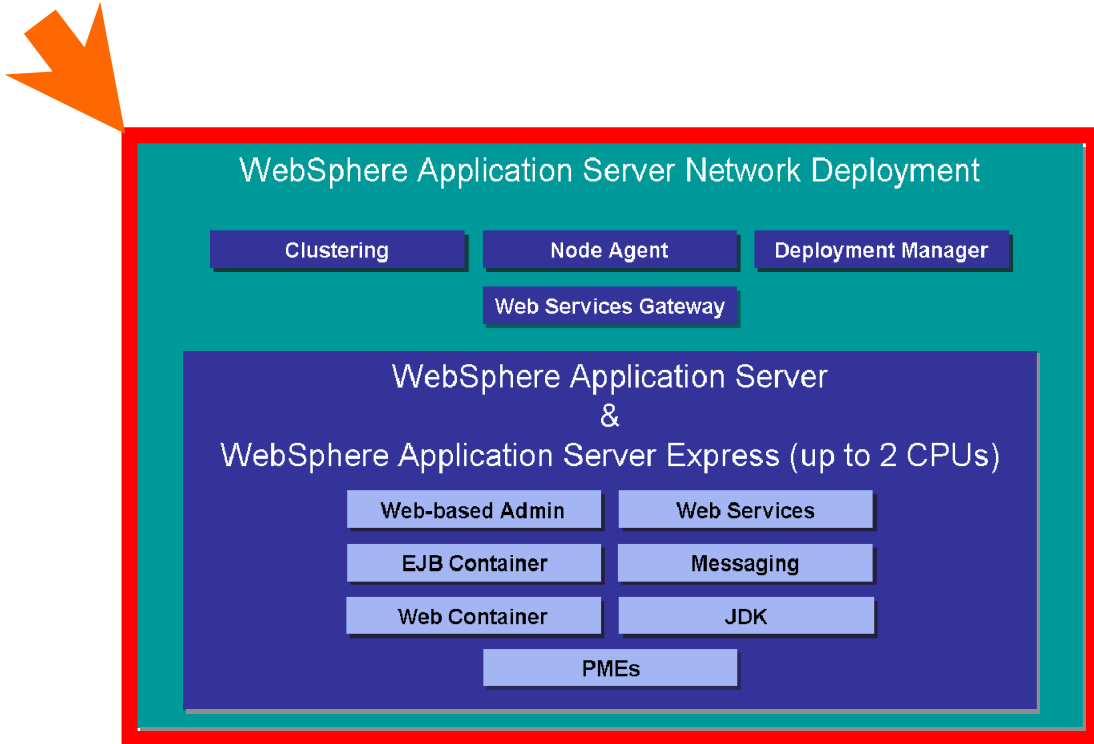
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Version 6 packaging



© Copyright IBM Corporation 2007

Figure 4-3. Version 6 packaging

WA3616.3

### **Notes:**

This unit puts the focus on Network Deployment version of WebSphere Application Server.

***Instructor notes:***

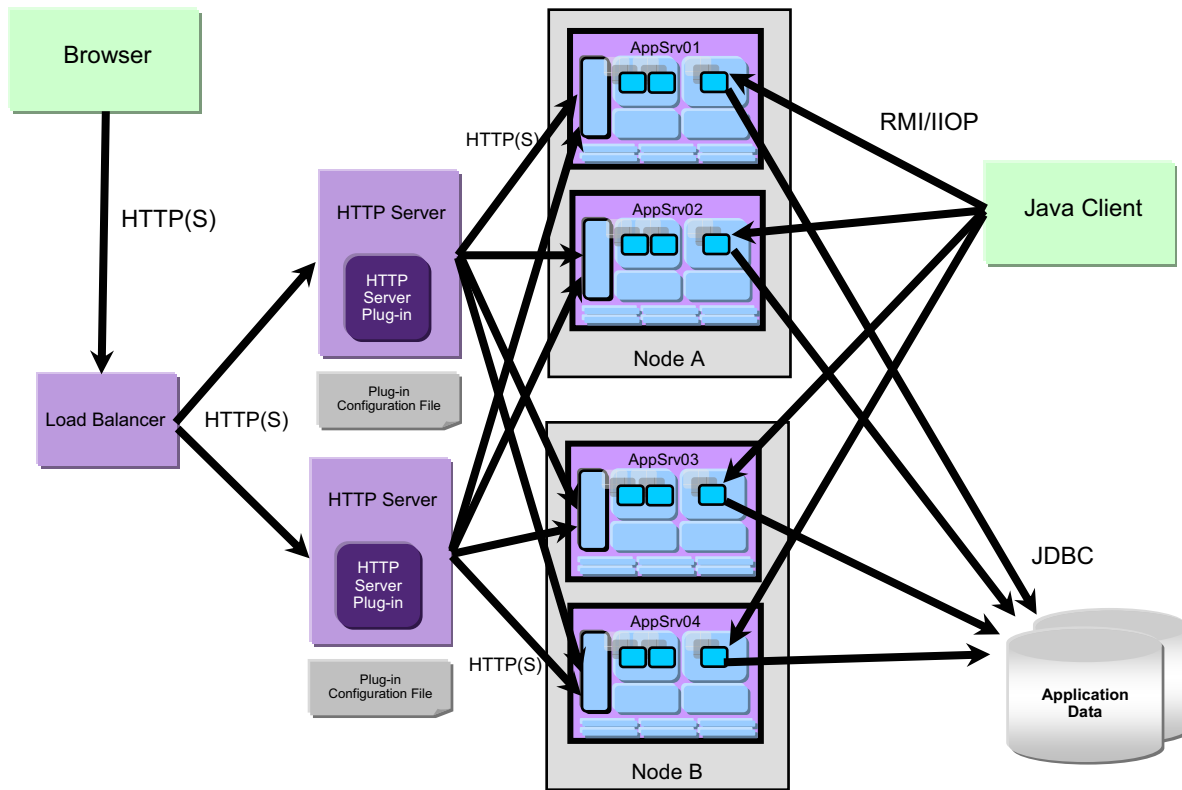
**Purpose** —

**Details** — Reiterate that everything contained in the last architecture unit still applies. This is separated into two units to group concepts and terminology better.

**Additional information** —

**Transition statement** —

## Network deployment runtime flow



© Copyright IBM Corporation 2007

Figure 4-4. Network deployment runtime flow

WA3616.3

### Notes:

The main theme with network deployment is distributed applications. While the “flow” of an application remains the same, there are significant additions to runtime of an application. Note the “Load balancer” this allows for multiple HTTP servers, users point their browsers to the load balancer and their request will be work load managed to an HTTP Server. Once the request hits one of these HTTP Servers, the HTTP Server plug-in will load balance the request between the application servers that it is configured to serve. Once the request enters the application server, the flow is identical to how it was in Express and Base.

The Java clients requests to EJBs can also be work load managed so that the requests do not all hit one application server.

***Instructor notes:***

**Purpose** —

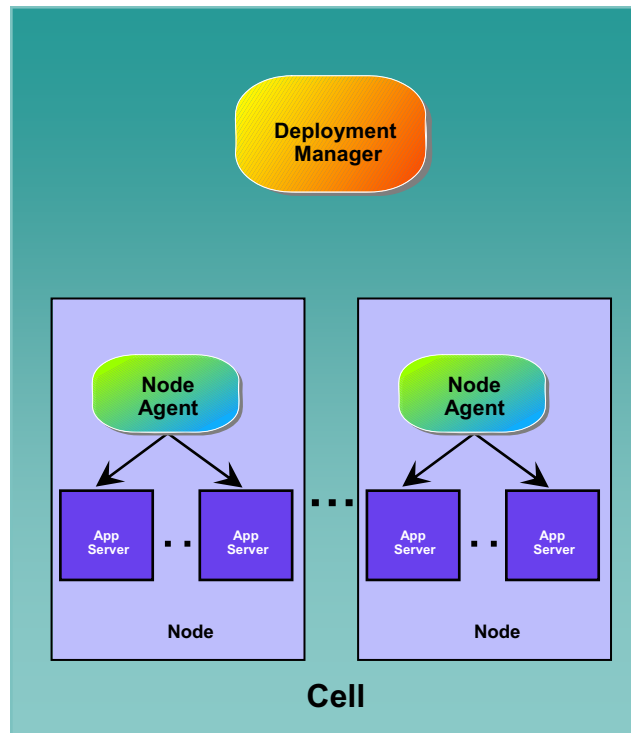
**Details** — While this slide is not animated like it was in the previous architecture unit, you should walk through it with the same process and point out where workload management occurs.

**Additional information** —

**Transition statement** —

## Network deployment concepts

- A **node** is a logical grouping of application servers
  - Each node is managed by a single **node agent** process
  - Multiple nodes can exist on a single machine through the use of profiles
- A **deployment manager** (DMgr) process manages the node agents
  - Holds the configuration repository for the entire management domain, called a **cell**
  - Within a cell, the administrative console runs inside the DMgr



© Copyright IBM Corporation 2007

Figure 4-5. Network deployment concepts

WA3616.3

### Notes:

The deployment manager here is an application server that manages the administrative environment within a cell. You will see later in this unit that a node is represented by a profile. The node agent is a very important process that allows for communication of administrative information (commands and configuration files) to reach the applications servers.



**Instructor notes:****Purpose —**

**Details —** Spend some time explaining the terms here. Make sure that you point out that the main concepts here are administrative in purpose only – not runtime!

At this time you also want to make sure students understand the fact that *node* does not equal *machine*.

**Additional information —****Transition statement —**

## Managed versus unmanaged nodes

---

- A managed node is a node that contains a node agent
- An unmanaged node is a node in the cell without a node agent
  - Enables the rest of the environment to be aware of the node
    - Useful for defining HTTP servers as part of the topology
    - Enables creation of different plug-in configurations for different HTTP servers

© Copyright IBM Corporation 2007

Figure 4-6. Managed versus unmanaged nodes

WA3616.3

### **Notes:**

A node agent is a process that handles communications with the resources within that node.

***Instructor notes:***

**Purpose** —

**Details** — This is really foreshadowing for the discussion on managed and unmanaged nodes with Web servers and to discuss the nature of node agents.

**Additional information** —

**Transition statement** —

## Network deployment administration flow

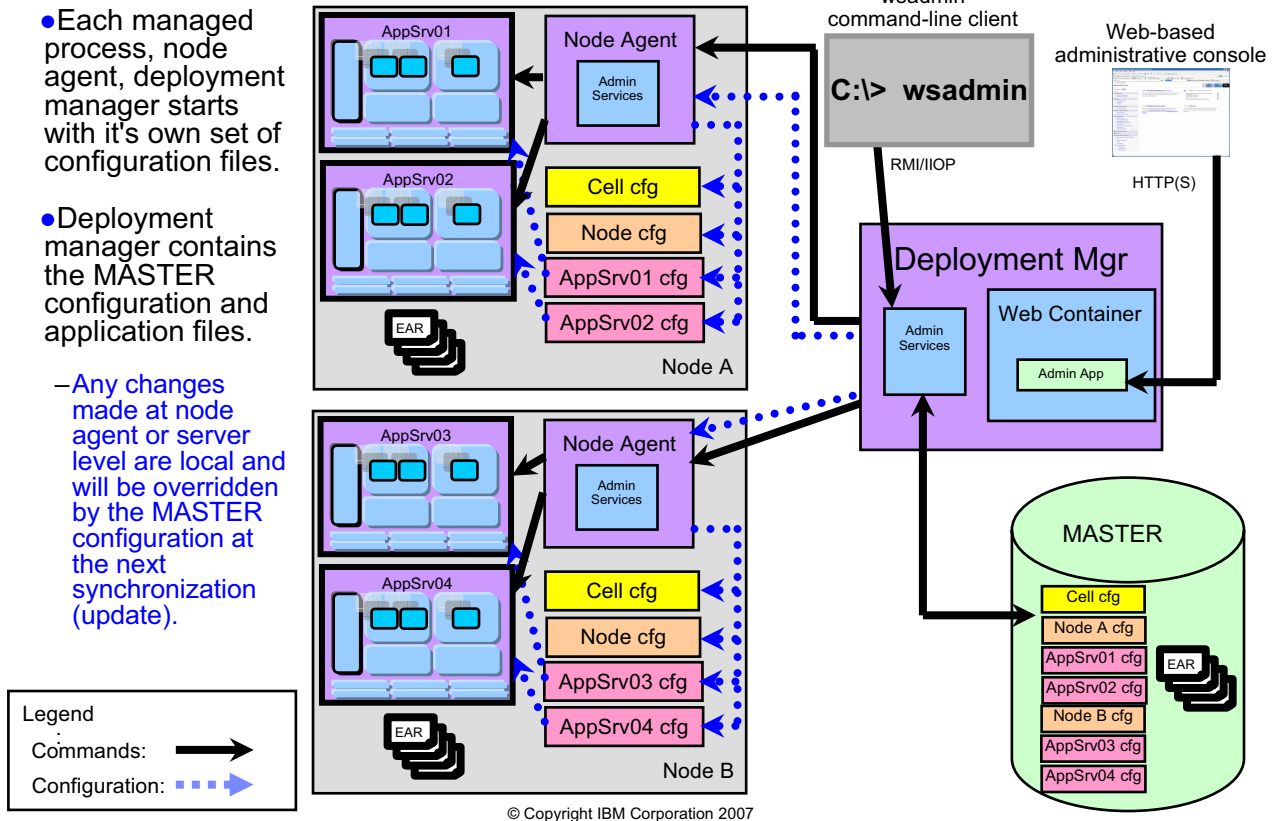


Figure 4-7. Network deployment administration flow

WA3616.3

### Notes:

The administrative console and wsadmin are still the two ways that the environment is administered. However, take note that these tools now talk to the deployment manager and **NOT** to the application servers directly. The communication of these commands flows from the tools to the deployment manager to the node agents, to the application servers. This allows administration of multiple nodes (each possibly containing multiple application servers) from a single focal point (the deployment manager).

There is **ONE** main repository for the configuration files within a cell, and those are associated with the deployment manager. All updates to the configuration files should go through the deployment manager. You will see in a moment how this process works. You should be very careful in connecting to an application server directly with wsadmin or the administrative console as any changes that are made to the configuration files are only temporary, they will be overwritten with the configuration files from the MASTER files.

***Instructor notes:***

**Purpose** —

**Details** — This slide is not animated. However, you may wish to walk through the administrative process like was done with the animated slide in the last architecture unit.

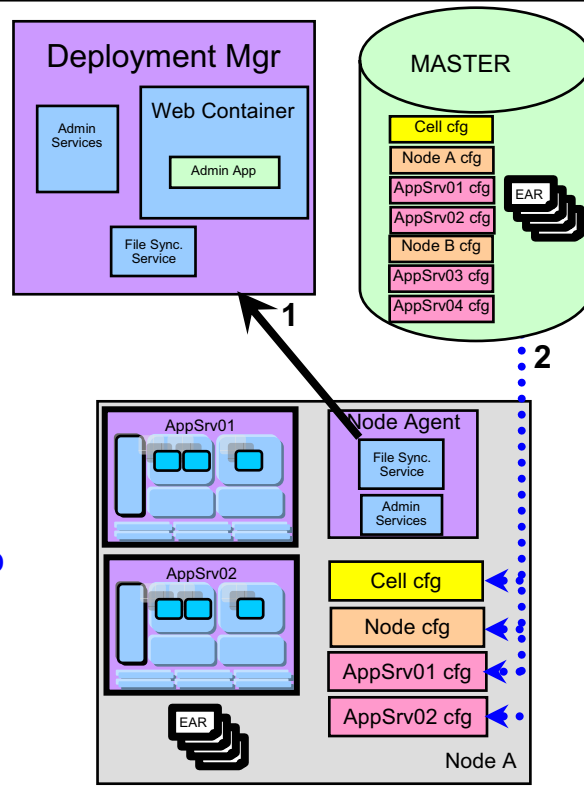
Make sure to point out all the information in the student notes.

**Additional information** —

**Transition statement** —

# File synchronization

- Deployment manager contains the master configuration
- Node agents synchronize their files with the master copy
  - **Automatically**
    - At start up
    - Periodically
  - **Manually**
    - Administrative console
    - Command line
- During synchronization
  - 1. Node agent asks for changes to master configuration
  - 2. New or updated files are copied to the node



© Copyright IBM Corporation 2007

Figure 4-8. File synchronization

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose** —

**Details** — Once again reinforce that this process will overwrite changes made directly on the node.

**Additional information** —

**Transition statement** —

## WebSphere Network Deployment profiles

---

- Benefits of profiles in network deployment:
  - Think of profiles as representing a node
    - Can install multiple profiles on a single machine
  
  - All profiles use the same product files
    - **Application server** profile (stand-alone)
      - Equivalent to Base or Express application server
      - Has a node name and a cell name property, and corresponding directories
      - Cell directory is overwritten upon federation
    - **Deployment manager** profile
      - Creates a deployment manager
    - **Custom** profile (managed)
      - Creates a managed node which, by default, is federated into a cell
      - Creates a node agent, but no application servers
    - **Cell** profile
      - Creates both a deployment manager and a federated node

© Copyright IBM Corporation 2007

Figure 4-9. WebSphere Network Deployment profiles

WA3616.3

### **Notes:**

The definition of profile has not changed with the addition of network deployment. However, now you see that what was defined previously as a profile is listed here as a “stand-alone node”, as it is just that – and application server. There is also the addition of the deployment manager which is a special kind of node that manages the administrative domain of a cell.



***Instructor notes:***

**Purpose** —

**Details** — Point out three types of profiles that can be created and contrast this to what was listed within the first architecture unit.

**Additional information** —

**Transition statement** —

## Managing Web servers with WebSphere

---

- WebSphere V6: DMgr can manage external Web servers
  - IBM HTTP Server V6 (special case – no node agent needed)
    - Deployment manager can distribute plugin-cfg.xml files to Web server machines
    - Can be started and stopped
    - Can edit the httpd.conf
  - Other Web servers (node agent needed)
    - Can have plugin-cfg.xml files automatically distributed to them
    - Can be started and stopped
  
- Web servers can be defined within WebSphere cell topologies
  - Managed node (local) or unmanaged node (remote)
    - Managed nodes use a node agent to control the Web server
    - Unmanaged nodes use the IBM HTTP Server Admin Service instead of a node agent to control the Web server

© Copyright IBM Corporation 2007

Figure 4-10. Managing Web servers with WebSphere

WA3616.3

### **Notes:**

Over the next several slides three scenarios will be presented:

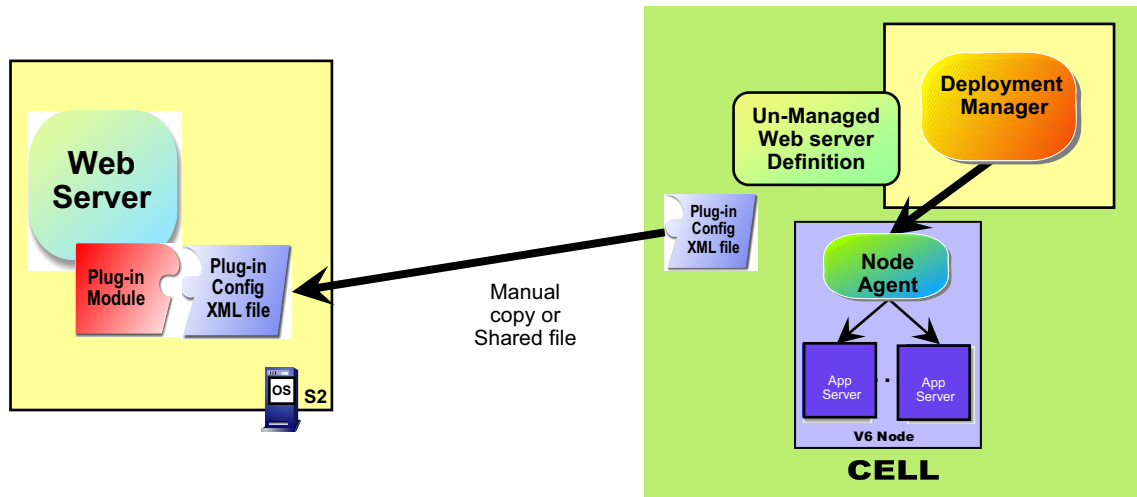
- Using a Web server as an unmanaged node
- Using IBM HTTP Server as an unmanaged node
- Using a Web server as a managed node

***Instructor notes:*****Purpose —**

**Details —** This slide is merely setting up for the discussion of different ways that Web servers can be used with network deployment. Do not dwell on the topics here as the pictures over the next three slides will make these concepts more clearly. You may wish to quickly discuss these three types and proceed to the pictures and then flip back to this page to explain the details.

**Additional information —****Transition statement —**

## Unmanaged Web server



- Web server (other than IBM HTTP Server) not managed by WebSphere node agent
- Allows WebSphere system administrator to create custom plug-in files for a specific Web server
- Manually ftp/copy the plug-in configuration file from the DMgr machine to the Web server machine

© Copyright IBM Corporation 2007

Figure 4-11. Unmanaged Web server

WA3616.3

### Notes:

Web server is registered as unmanaged node in WebSphere configuration. This scenario is common for Web servers installed outside your firewall or in a DMZ where no WebSphere Application Server exists.

The implication with this scenario is that all management of the Web server has to happen manually (outside the control of WebSphere).

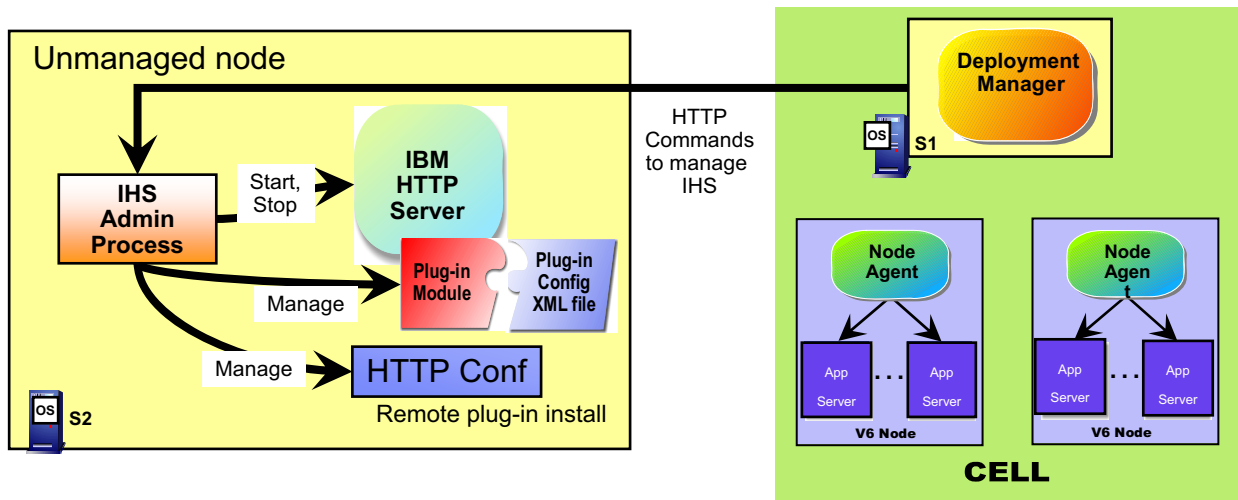
This functionality is consistent with previous versions of WebSphere Application Server.

**Instructor notes:****Purpose —**

**Details —** The details are in the student notes. The main thing to point out here is that everything with this Web server has to happen manually, there is no automatic propagation of plugin-cfg.xml files, and no way to start and stop the Web server from with the WebSphere administration tools.

**Additional information —****Transition statement —**

# IBM HTTP Server as unmanaged node (remote)



- WebSphere V6 and IBM HTTP Server have special enhancements
  - IHS admin process provides administrative functions for IBM HTTP Server within WebSphere
  - Provides ability to start, stop IBM HTTP Server, make configuration changes to httpd.conf and automatically push the plug-in configuration file to IBM HTTP Server machine
  - Does not need node agent on the Web server machine

© Copyright IBM Corporation 2007

Figure 4-12. IBM HTTP Server as unmanaged node (remote)

WA3616.3

## Notes:

IBM HTTP Server can be managed completely from the Deployment Manager (DMgr). The DMgr communicates with the IBM HTTP Server admin process that runs on the node with the IBM HTTP Server server.

UNIX NOTE: On UNIX platforms the userid under which the IHS admin process runs must have write permissions to the plugin-cfg.xml file. By default, at IBM HTTP Server installation, the plugin-config.xml file is owned by root, and only root has write access to it. This will cause a failure with propagation of the plugin-cfg.xml file when run from the administration tools.

***Instructor notes:***

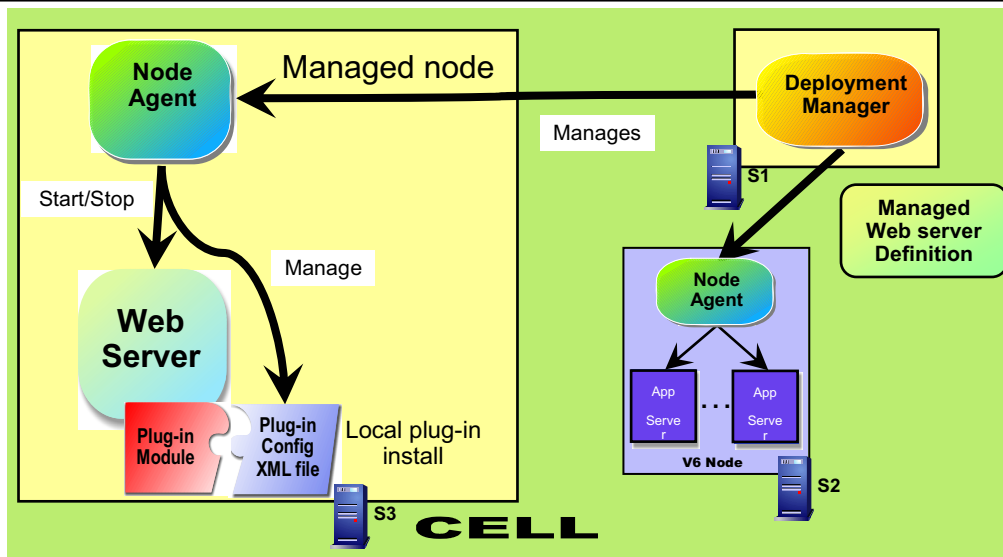
**Purpose** —

**Details** — Again, the details are in the student notes.

**Additional information** —

**Transition statement** —

## Managed Web server on a managed node (local)



- Install a Web server on a managed node.
- Create a Web server definition within the DMgr.
- Node agent receives commands from DMgr to administer the Web server.
- Plugin-cfg.xml file is propagated through the file synchronization service and lives under the config directory.

© Copyright IBM Corporation 2007

Figure 4-13. Managed Web server on a managed node (local)

WA3616.3

### Notes:

The Web server is managed via the deployment manager through the node agent. This provides ability to start, stop the Web server and automatically push the plug-in configuration file to the Web server from the DMgr. This can be used when the Web server is on the same machine where WebSphere Application Server is installed. This is a common scenario for behind a firewall where a WebSphere node can be installed.

There is a node agent that is performing the communication with the Web server from the admin tools of WebSphere.

It may be undesirable to use this configuration, since access to the node agent in a DMZ could compromise security.



***Instructor notes:***

**Purpose —**

**Details —**

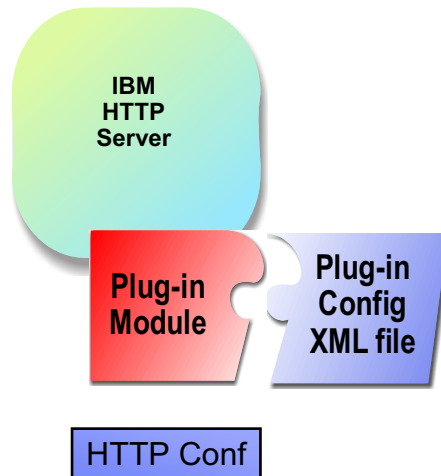
**Additional information —**

**Transition statement —**

## IBM HTTP Server administration overview

---

- Direct administration of IBM HTTP Server V6.x is done by manually editing `httpd.conf`
  - There is no Web-based console for IBM HTTP Server as there was in previous versions.



© Copyright IBM Corporation 2007

Figure 4-14. IBM HTTP Server administration overview

WA3616.3

### **Notes:**

The previous user interface to administer IBM HTTP Server no longer exists. You must either manually make updates to the `httpd.conf` file, or use the WebSphere administrative console.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## IBM HTTP Server administration server

---

- IBM HTTP Server administration server runs as a separate instance of IBM HTTP Server.
- Admin component for IBM HTTP Server V6 includes:
  - IBM HTTP Server admin configuration file (admin.conf).
  - Default port for the IBM HTTP Server admin server is 8008.
- IBM HTTP Server admin authentication password file (htpasswd.admin).
  - Initially blank, which prohibits access to IBM HTTP Server administration
  - Administrator updates IBM HTTP Server admin password file using:  
> `htpasswd -cm ..\conf\admin.passwd <user_name>`
- To start and stop the administration server:
  - `<ihs_root>/bin/adminctl start`
  - `<ihs_root>/bin/adminctl stop`
  - Or Windows Service

© Copyright IBM Corporation 2007

Figure 4-15. IBM HTTP Server administration server

WA3616.3

### **Notes:**

This is no different than in previous versions of IBM HTTP Server on UNIX platforms. The functionality in V6.x has been extended to Windows platforms.

***Instructor notes:***

**Purpose —**

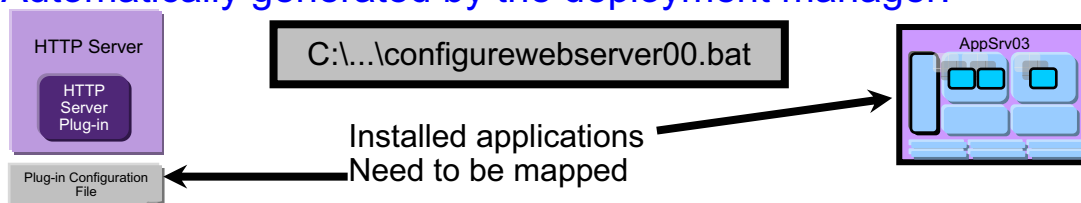
**Details —**

**Additional information —**

**Transition statement —**

## Web server custom plugin-cfg.xml

- Web server definitions are created to allow the mapping of J2EE enterprise applications to specific Web servers.
  - Can be done through the administrative console
  - Alternately use the script generated during the installation of the plug-in which can automate the mapping of all the applications to the Web server
    - `configure<Web_server_name>.bat` in `<plugin_root>\bin`
- Mapping the applications to specific Web servers will cause the custom plugin-cfg.xml files for only those Web servers to include the information for those applications.
  - Web servers target specific applications running in a cell.
  - Automatically generated by the deployment manager.



© Copyright IBM Corporation 2007

Figure 4-16. Web server custom plugin-cfg.xml

WA3616.3

### Notes:

Just as modules for an enterprise application need to be mapped to one or more application servers, they also need to be mapped to one or more Web servers.

During the upcoming lab exercise you will be doing this through the administrative console rather than the generated script.

***Instructor notes:***

**Purpose —**

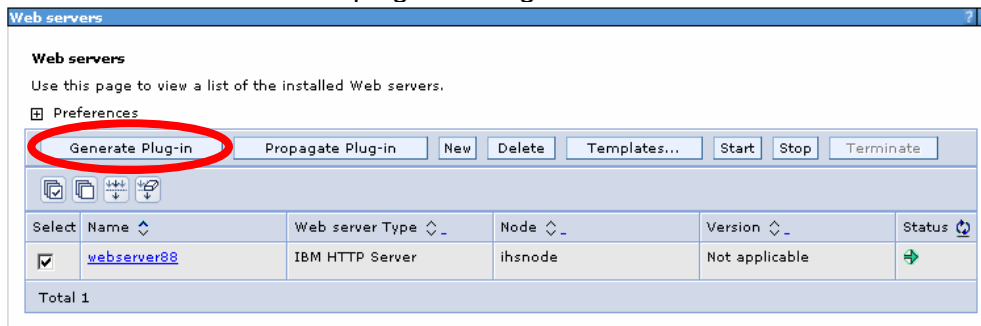
**Details —**

**Additional information —**

**Transition statement —**

## Managing plugin-cfg.xml files

- plugin-cfg.xml files are automatically generated and propagated.
  - This behavior is the default, but can be changed.
  - This behavior is configurable through the console.
- plugin-cfg.xml files can be generic to a cell or custom to Web server.
  - [Generating a cell generic plugin-cfg.xml file](#)
    - Use the command line script `<was_root>\bin\GenPluginCfg.bat`
    - Not available through the console
  - [Generating a Web server custom plugin-cfg.xml file](#)
    - Use the administrative console
    - Need to map applications to Web servers
    - Can customize each Web server's plug-in settings



© Copyright IBM Corporation 2007

Figure 4-17. Managing plugin-cfg.xml files

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

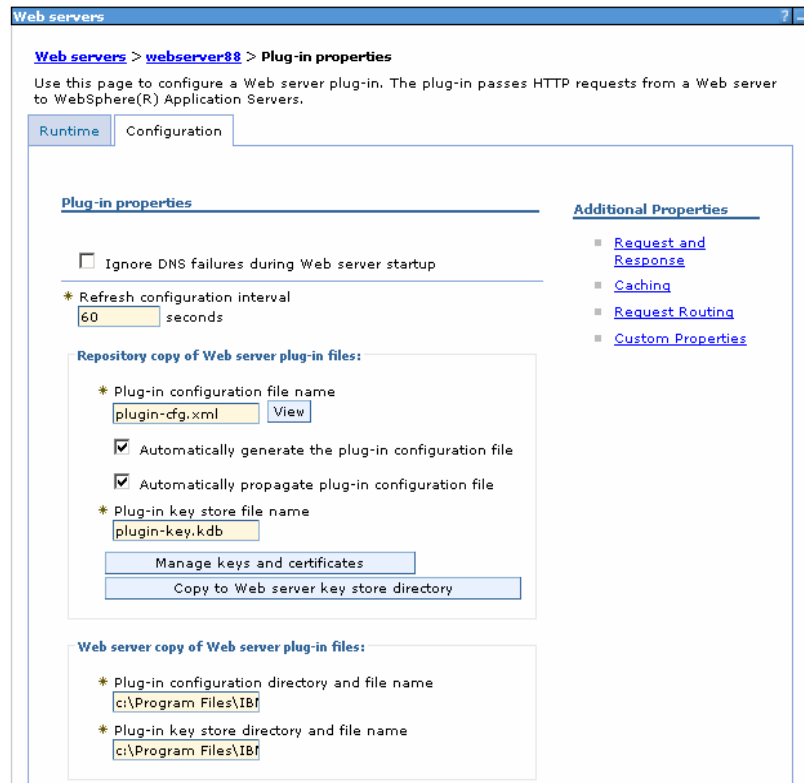
**Details —**

**Additional information —**

**Transition statement —**

## Managing Web server plug-in properties

- Each Web server can have customized plugin-cfg settings.
  - Not just application mappings



© Copyright IBM Corporation 2007

Figure 4-18. Managing Web server plug-in properties

WA3616.3

### Notes:

The links on the right side of the page under Additional Properties provide access to further pages for changing the plug-in properties.

***Instructor notes:***

**Purpose —**

**Details —**

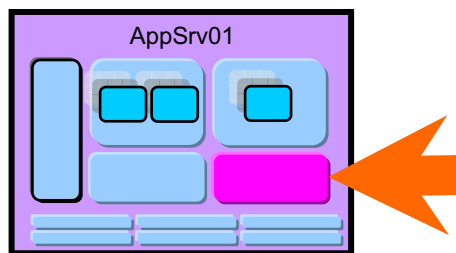
**Additional information —**

**Transition statement —**

## Platform Messaging overview

---

- Integrated asynchronous capabilities for the WebSphere platform
  - Integral JMS messaging service for WebSphere Application Server
  - Fully compliant JMS 1.1 provider
- Service integration bus
  - Intelligent infrastructure for service-oriented integration
  - Unifies SOA, messaging, message brokering and publish/subscribe
- Compliment and extend WebSphere MQ and application server
  - Share and extend messaging family capabilities



© Copyright IBM Corporation 2007

Figure 4-19. Platform Messaging overview

WA3616.3

### Notes:

The service integration functionality within WebSphere Application Server provides a highly-flexible messaging fabric that supports a service-oriented architecture with a wide spectrum of quality of service options, supported protocols, and messaging patterns. It supports both message-oriented and service-oriented applications.

SOA is an architectural style whose goal is to achieve loose coupling among interacting software agents. A service is a unit of work done by a service provider to achieve desired end results for a service consumer. Both provider and consumer are roles played by software agents on behalf of their owners.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## High availability overview

---

- High availability (HA) manager is used to eliminate single points of failure.
- High availability manager is responsible for running key services on available servers rather than on a dedicated one (such as the DMgr).
- Can take advantage of fault-tolerant storage technologies such as Network Attached Storage (NAS).
- Hot standby and peer failover for critical singleton services.
  - WLM routing, PMI aggregation, JMS messaging, transaction manager, and so forth.
  - Failed singleton starts up on an already-running JVM.
  - Planned failover takes < 1 second.

© Copyright IBM Corporation 2007

Figure 4-20. High availability overview

WA3616.3

### **Notes:**

High availability is a very complex topic and will not be covered within this course.

**Instructor notes:****Purpose —**

**Details —** As the student notes point out, this is a very complex topic. The intent with this slide is just to make students aware of high availability, not to explore the topic in any depth.

The HA Manager runs as a service within each application server process that monitors the healthiness of WebSphere clusters. In the event of a server failure, the HA manager will failover the singleton service and recover any in-flight transactions. Each application server process runs a HA manager component and shares information through an underlying communication infrastructure Distribution and Consistency Services (DCS) such that no single point of failure would exist in the topology. Every member in a WebSphere cluster would know where singleton services are running.

There is an excellent graphic in the **WebSphere Application Server V6 Scalability and Performance Handbook** red book (SG24-6392-00).

**Additional information —****Transition statement —**

## Data replication service

---

- Data replication service (DRS) is responsible for replicating in-memory data among WebSphere processes.
  - Helps allow for high availability and failover recovery
  - Improves performance and scalability
- What uses this service?
  - Stateful session EJB persistence and failover
  - HTTP session persistence and failover
  - Dynamic cache replication
- Uses either peer-to-peer or client/server replication techniques.

© Copyright IBM Corporation 2007

Figure 4-21. Data replication service

WA3616.3

### **Notes:**

Data replication service is very easy to configure and use. This service is very important when dealing with failover in a work load managed environment.



***Instructor notes:*****Purpose —**

**Details —** Make sure to point out the importance of this in a distributed environment. You may wish to go back to the runtime slide and point out why this is so important in that environment.

The next slide goes into details of this functionality for stateful session beans.

**Additional information —****Transition statement —**

## Failover of stateful session EJBs

---

- Uses DRS, similar to HTTP session failover.
- Always enabled.
- WLM fails beans over to a server that already has a copy of the session data in memory if possible.
- Ability to collocate stateful session bean replicas with HTTP session replicas with hot failover.
  - J2EE 1.4 specification requires HTTP session state objects to be able to contain local references to EJBs.

© Copyright IBM Corporation 2007

Figure 4-22. Failover of stateful session EJBs

WA3616.3

### **Notes:**

This is significant new functionality within J2EE 1.4. In previous versions of the specification this was not defined, therefore stateful session beans were restricted as they could not easily perform in a work load managed with fail over environment.

***Instructor notes:***

**Purpose —**

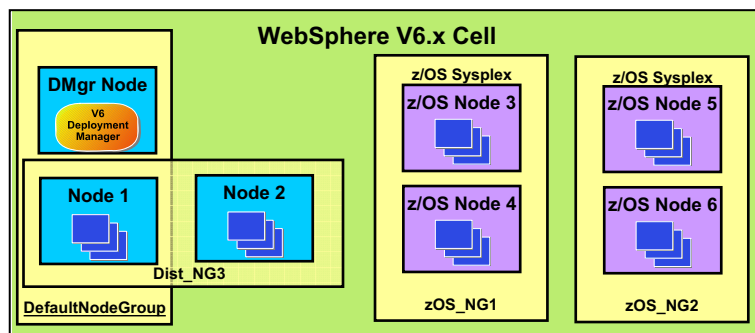
**Details —**

**Additional information —**

**Transition statement —**

## Node group overview

- Enables mixing nodes with different capabilities within the same cell for administration purposes
  - z/OS and distributed nodes
  - WBI nodes and base nodes
  - Mechanism that allows validation of the node capability before perform certain functions
    - Example: Creating a cluster of nodes – cannot mix servers from z/OS and distributed nodes within a cluster
- Default configuration with single node group is sufficient unless you want to mix platforms within cell



© Copyright IBM Corporation 2007

Figure 4-23. Node group overview

WA3616.3

### Notes:

Node groups are not used for work load management, like cells they only serve an administrative purpose.

In the picture above, there are four distinct node groups. The first is the “DefaultNodeGroup” that was created when the deployment manager was created. By default all nodes created within the cell get assigned to this group (thus why Node 1 is there). The second is “Dist\_NG3” which contains Node1 and Node2 (it overlaps with DefaultNodeGroup). The third and fourth node groups are the z/OS NG1 and NG2.

***Instructor notes:***

**Purpose** —

**Details** — See student notes.

**Additional information** —

**Transition statement** — Not for work load management.

## Name service

- Provides a JNDI name space.
- Registers all EJB and J2EE resources (example: JDBC Providers, JMS, J2C, URL and JavaMail) that are hosted by the application server.
- There is one name server per application server.
- Configured bindings can map resources to remote locations.

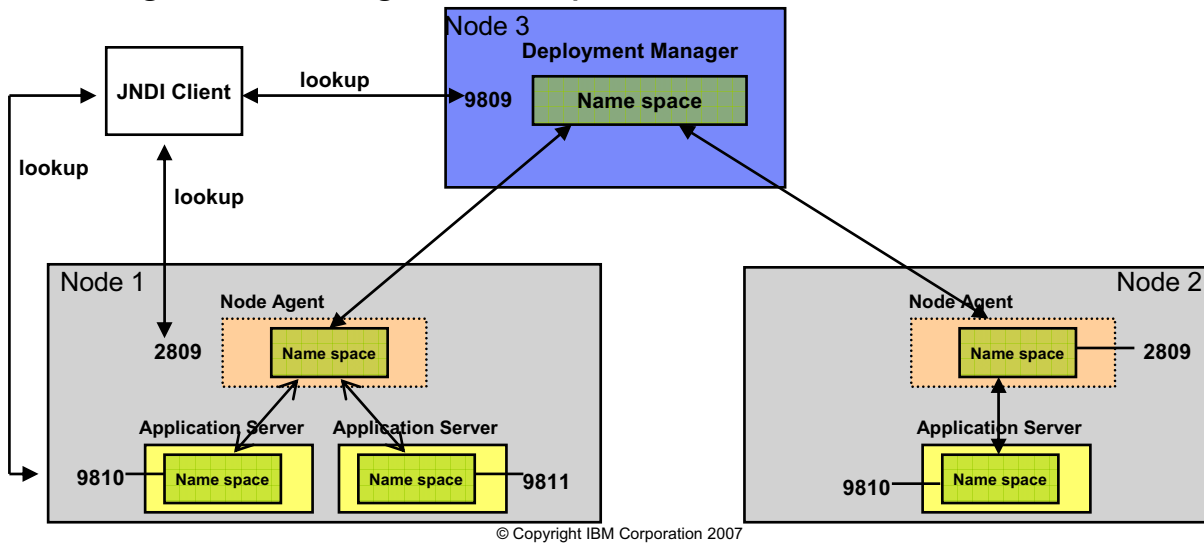


Figure 4-24. Name service

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

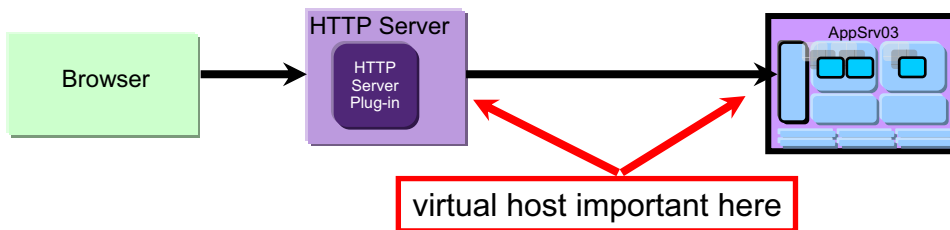
**Details** — Configured bindings: configured binding to map a resource within the same cell but in a different application server, to things in a different cell, or use fully qualified names (bad idea).

**Additional information** —

**Transition statement** —

## Virtual hosts

- Configuration that enables a host machine to resemble multiple host machines
  - Allows one machine to support multiple applications
  - Associated with the cell not a single node
  - Enables plug-in to route requests to the correct servers
- Each virtual host has a logical name and:
  - One or more host aliases
    - Each alias is a host name and port combination (allows wildcards)
    - For example: \*:80, \*:443, \*:9080, \*:9060
- There are several default virtual hosts, including:
  - default\_host - Used for accessing the default applications
    - Example: http://localhost:9080/snoop
  - admin\_host - Used for accessing the administrative console
    - Example: http://localhost:9060/ibm/console



© Copyright IBM Corporation 2007

Figure 4-25. Virtual hosts

WA3616.3

### Notes:



***Instructor notes:***

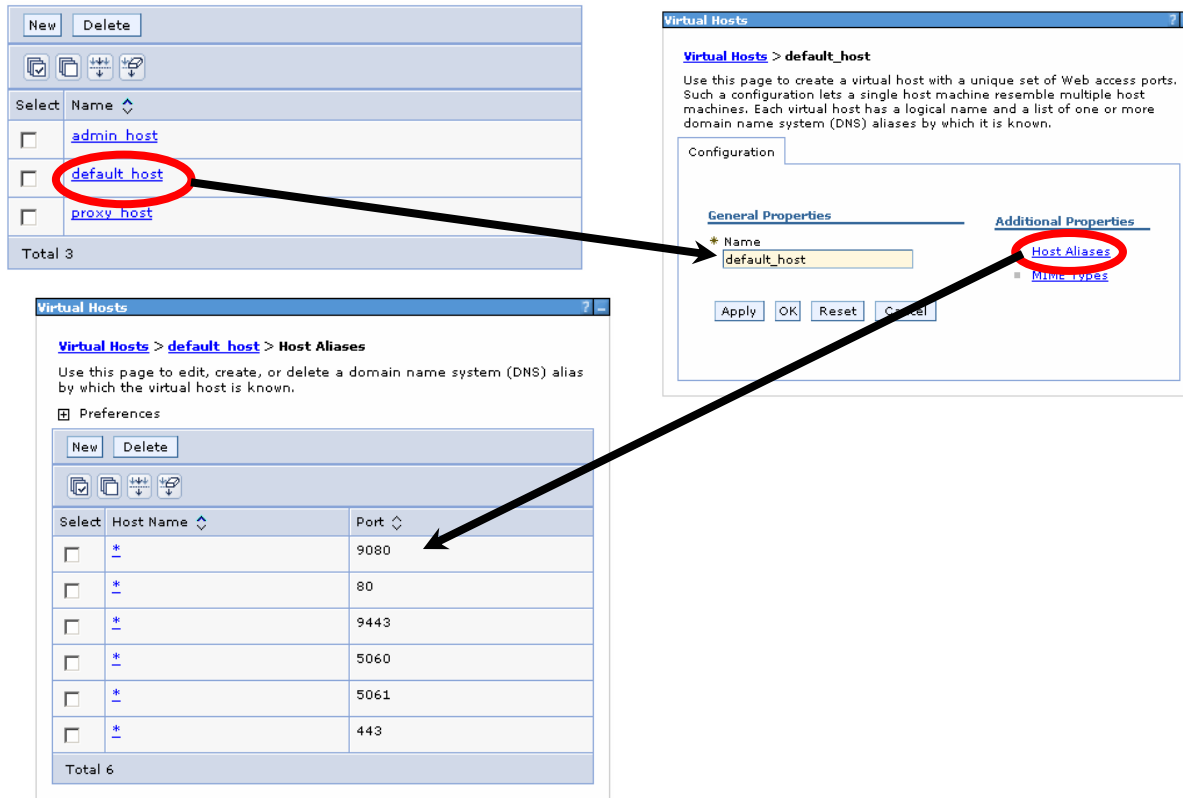
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Defining virtual hosts



© Copyright IBM Corporation 2007

Figure 4-26. Defining virtual hosts

WA3616.3

### Notes:

This is the definition for “default\_host”. Notice that the host aliases specified are “\*”. This means that a request for any host name on ports 9080, 80, and 9443 will get forwarded to this host. When applications are installed Web modules within those applications need to be mapped to a virtual host.

***Instructor notes:***

**Purpose —**

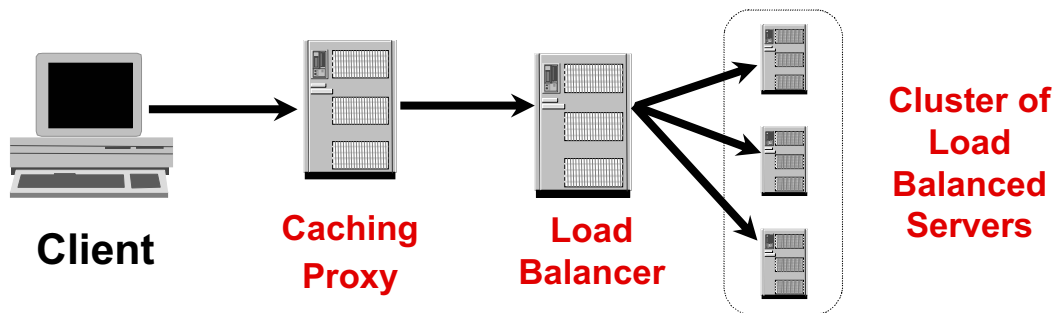
**Details —**

**Additional information —**

**Transition statement —**

## Edge Components

- WebSphere Application Server Network Deployment package contains the following Edge Component functionality:
  - Load balancer
  - Caching proxy
- Edge Components install separately from WebSphere Application Server.
- Load balancer is responsible for balancing the load across multiple servers that can be within either local area networks or wide area networks.
- Caching proxy's purpose is to reduce network congestion within an enterprise by offloading security and content delivery from Web servers and application servers.



© Copyright IBM Corporation 2007

Figure 4-27. Edge Components

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Checkpoint questions

---

1. A process that handles communications with the resources within the node is?
2. During what process does the node agent check for changes to the master configuration?
3. What is a configuration that enables a host machine to resemble multiple host machines?
4. What defines the runtime environment for either the deployment manager or the application server?

© Copyright IBM Corporation 2007

Figure 4-28. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.
- 4.

***Instructor notes:***

Purpose —

Details —

## Checkpoint solutions

---

1. Node agent
2. File synchronization
3. Virtual host
4. Profiles

© Copyright IBM Corporation 2007

**Additional information —**

**Transition statement —**

## Unit summary

---

Having completed this unit, you should be able to explain:

- Describe Network Deployment runtime flow
- Describe Network Deployment concepts and terminology such as cell, node, node agent, and deployment manager
- Describe Network Deployment administration flow
- Explain how to manage Web servers with WebSphere
- Describe high availability concepts and functions
- Describe data replication service concepts and functions
- Describe name service concepts and functions
- Describe platform messaging

© Copyright IBM Corporation 2007

Figure 4-29. Unit summary

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**



---

# Unit 5. WebSphere Application Server installation

## Estimated time

00:30

## What this unit is about

This unit describes the tasks involved with the planning, installation and post-installation of WebSphere Application Server V6.1.

## What you should be able to do

After completing this unit, you should be able to:

- Describe the installation process for WebSphere Application Server
- Verify the installation of WebSphere Application Server
- Describe WebSphere profiles
- Describe the directories and configuration files for WebSphere Application Server
- Describe how to use configuration commands
- Describe how to use backup and restore configuration utilities

## How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center



Figure 5-1. WebSphere Application Server V6.1 installation

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe the installation process for WebSphere Application Server
- Verify the installation of WebSphere Application Server
- Describe WebSphere profiles
- Describe the directories and configuration files for WebSphere Application Server
- Describe how to use configuration commands
- Describe how to use backup and restore configuration utilities

© Copyright IBM Corporation 2007

Figure 5-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

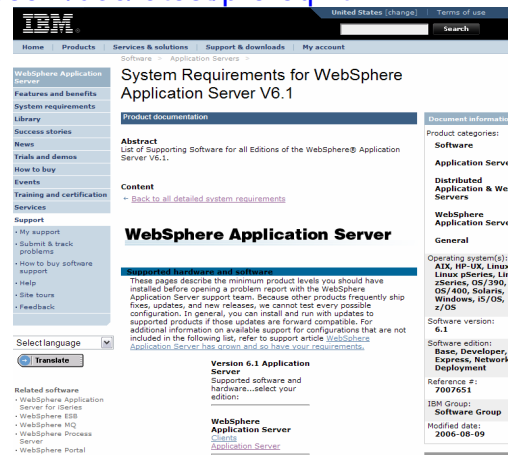
**Details —**

**Additional information —**

**Transition statement —** Next: Hardware requirements

# Hardware requirements

- Requirements vary by platform and your intended workload
- Minimum requirements are:
  - Minimum 512 MB memory; 1 GB recommended
  - CD-ROM drive for installation
- For the latest details, check:
  - <http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html>



© Copyright IBM Corporation 2007

Figure 5-3. Hardware requirements

WA3616.3

## Notes:

Note that these are the minimum requirements and platforms documented when this course was created, for the latest requirements check:

<http://www-3.ibm.com/software/webservers/appserv/docs/prereqs/latest.html>

Most production system will probably need more memory and a faster CPU depending on the number and types of applications deployed, the configuration and the load on the servers.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Software requirements – Operating systems

## Software requirements – Operating systems

---

- Supported operating systems include:
  - Windows
  - AIX
  - Sun Solaris
  - HP-UX
  - Linux/Intel, Linux/PPC, Linux for zSeries
  - I5/OS and OS/400
  - z/OS
  - And others
  
- For the latest specifics on versions and patch levels, check:
  - <http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html>

© Copyright IBM Corporation 2007

Figure 5-4. Software requirements – Operating systems

WA3616.3

### **Notes:**

Note that these are the minimum requirements and platforms documented when this course was created, for the latest requirements check:

<http://www-3.ibm.com/software/webservers/appserv/docs/prereqs/latest.html>

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Software requirements — Supported Web servers

## Software requirements – Supported Web servers

---

- Supported Web servers include:
  - Apache
  - IBM HTTP Server
  - Internet Information Server
  - Sun ONE Web Server
  - Lotus Domino
  - And others
  
- For the latest specifics on versions and patch levels, check:
  - <http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html>

© Copyright IBM Corporation 2007

Figure 5-5. Software requirements – Supported Web servers

WA3616.3

### **Notes:**

Note that these are the minimum requirements and platforms documented when this course was created, for the latest requirements check:

<http://www-3.ibm.com/software/webservers/appserv/docs/prereqs/latest.html>

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Software requirements - Database

## Software requirements – Database

---

- Supported database servers include:
  - IBM DB2
  - Oracle
  - Sybase
  - Microsoft SQL
  - Informix
  - And others
  
- For the latest specifics on versions and patch levels, check:
  - <http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html>

© Copyright IBM Corporation 2007

Figure 5-6. Software requirements – Database

WA3616.3

### **Notes:**

Note that these are the minimum requirements and platforms documented when this course was created, for the latest requirements check:

<http://www-3.ibm.com/software/webservers/appserv/docs/prereqs/latest.html>

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Installation overview

## Installation overview

---

- The WebSphere installation separates the core product binary data from the configuration data that is created as the application server runtime environment is customized
  - Can have multiple sets of customized runtime environments
  - Each environment is known as a profile
  - Profiles share product binaries
- Need to create profiles after installation of product binaries
  - Profile Management Tool is launched as part of the install
- Easier than multiple installations
  - Less disk space
  - Product update is simplified
- Each component is a separate installation

© Copyright IBM Corporation 2007

Figure 5-7. Installation overview

WA3616.3

### **Notes:**

The launchpad application is available on the product CD and on downloaded installation images. The launchpad is the recommended method of installing components that are on the product CD.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Installation environments

## Installation environments

Product	Default environments	Application Server instances created during installation
<b>WebSphere Application Server – Express/”Base”</b>	One stand-alone application server profile and the Profile Management Tool for creating other application server profiles on the same machine.	server1
<b>WebSphere Application Server – Network Deployment</b>	A default runtime environment is not created during installation of the core product files.	From the last installation panel, you can launch the Profile Management Tool to create one of the four available profiles: <ul style="list-style-type: none"> <li>* Deployment manager</li> <li>* Application server</li> <li>* Custom</li> <li>* Cell</li> </ul>

© Copyright IBM Corporation 2007

Figure 5-8. Installation environments

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

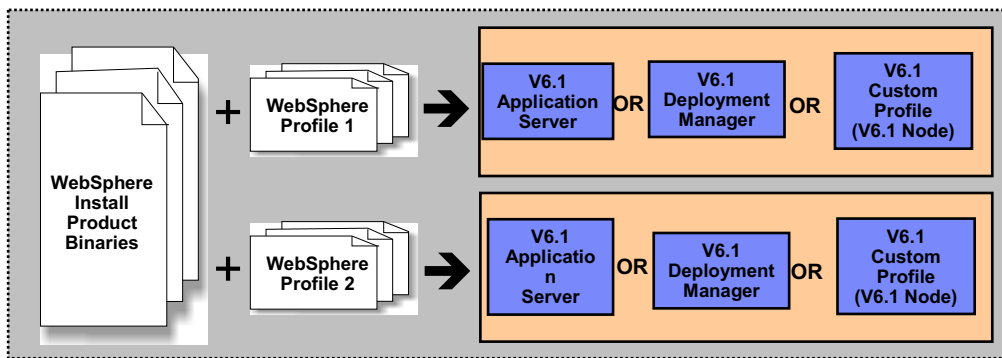
**Additional information —**

**Transition statement —** Next: WebSphere Profiles: Types

# WebSphere Profiles: Types

- **Profile types:** define application server configuration

Profile Types	V6.1 packages	Functions
Application server (default)	All	Create different instances of a standalone node – each stand-alone node has 1 application server
Deployment manager	Network Deployment	Create different instances of DMgr – each DMgr is its own cell
Custom (managed)	Network Deployment	Creates and federates a node containing <b>no pre-defined application server</b> definitions
Cell	Network Deployment	Creates a combination of a DMgr and a federated node.



© Copyright IBM Corporation 2007

Figure 5-9. WebSphere Profiles: Types

WA3616.3

## Notes:

The Profile Management Tool lets you create a deployment manager profile, an application server profile, or a custom profile. A *profile* consists of files that define the runtime environment for the deployment manager or the application server. Each environment has its own administrative interface. A custom profile is an exception. A custom profile is an empty node that you can federate into a deployment manager cell and customize. No default server processes or applications are created for a custom profile. Each deployment manager or application server profile has its own First steps console. The location of the command to launch the First steps console is within the set of files in the profile. A prompt to launch the First steps console that is associated with a profile displays on the last panel of the Profile Management Tool.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Creating profiles

## Creating profiles

---

- WebSphere provide multiple ways of creating profiles
  
- Profile Management Tool
  - Start menu (Windows only)
  - Launched after installation from install wizard
  - Launch via command line tool `pctWindows`
    - `<was_root>\bin\ProfileManagement\pmt.bat`
    - Similar command exists for UNIX platform (`pmt.sh`)
  - Wizard in first steps console
  
- Manually via the `manageprofiles` command line tool
  - Create profiles in silent mode using `-silent` option

© Copyright IBM Corporation 2007

Figure 5-10. Creating profiles

WA3616.3

### **Notes:**

Profiles are created based on templates supplied with the product. Each template consists of a set of files that provide the initial settings for the profile and a list of actions to perform after the profile is created. Currently there is no provision for modifying these templates for your use, or for creating templates based on existing applications servers. When creating a profile using `manageprofiles`, you need to specify one of the following templates:

- > default (for application server profiles)
- > dmgr (for deployment manager profiles)
- > managed (for custom nodes)
- > cell

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Pre-Installation tasks

## Pre-Installation tasks

---

- Confirm hardware and software requirements
  - Obtain and install necessary products and maintenance
  - Check WebSphere Web site for any maintenance
- Make sure TCP/IP networking is correctly configured
  - Host name of node should be in DNS or local hosts file
  - Host name of node should remain fixed
  - DHCP not supported
- Read ***Platform-specific tips for installing and migrating*** article in Information Center for installation platform tips
- Create/verify user ID for starting services
- Verify user ID for disk read/write access
- Review install readme file

© Copyright IBM Corporation 2007

Figure 5-11. Pre-Installation tasks

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Launchpad (1 of 2)

## Launchpad (1 of 2)

---

- Single point of reference for installing the entire application server environment
  - [Installs WebSphere product binaries](#)
- Provides links to solution installation diagrams and installation documentation
- Welcome page contains fastpath links to launch installation wizard for installable components
  - [WebSphere Application Server Network Deployment](#)
  - [IBM HTTP Server](#)
  - [Web server plug-ins](#)
  - [WebSphere Application Server Clients](#)
  - [Application Server Toolkit](#)

© Copyright IBM Corporation 2007

Figure 5-12. Launchpad (1 of 2)

WA3616.3

### **Notes:**

Application Server Toolkit only available on Linux and Windows. WebSphere Application Server is an integrated platform that contains an application server, a set of Web development tools, a Web server, and additional supporting software and documentation. The launchpad is the single point of reference for installing the entire Application Server environment.

The launchpad opens in the language of the machine locale setting. The launchpad identifies components on the product disc that you can launch.

The launchpad can install the tools component in the primary packet of discs. The launchpad on the separate Application Server Toolkit disc can launch the installation of the tool on Windows 2000 and Linux (Intel) systems.

The launchpad has diagrams that show the possible application server topologies that you can create with all of the components in the primary packet.

A link exists for each installable component on the product compact disc and for the installable tools component that is on a separate disc in the primary packet of compact discs in the product package.

***Instructor notes:***

**Purpose —**

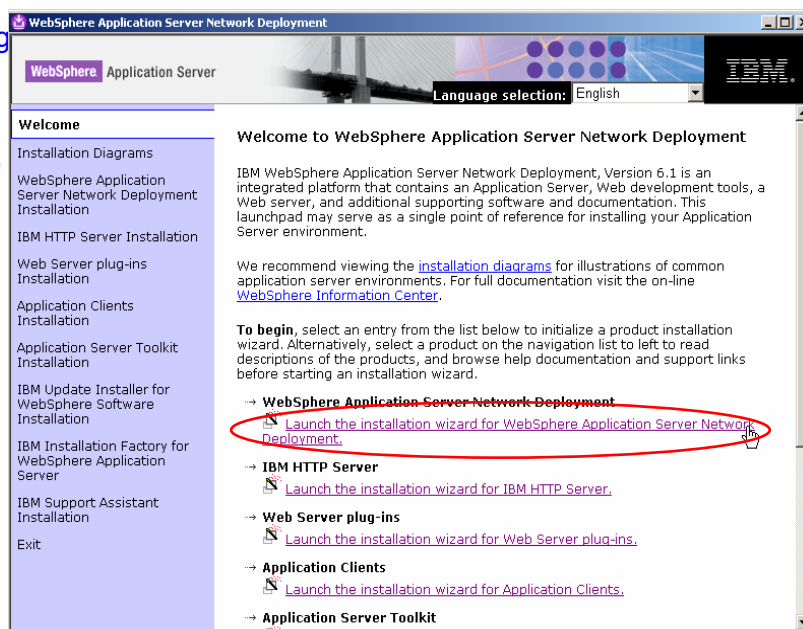
**Details —**

**Additional information —**

**Transition statement —** Next: Launchpad (2 of 2)

## Launchpad (2 of 2)

- Launch installation wizard
  - Checks prerequisites
  - Looks for a previous WebSphere Application Server installation for adding features or installing new binaries
  - Looks for a previous V6.x installation to upgrade
  - Prompts you to start profile creation wizard



© Copyright IBM Corporation 2007

Figure 5-13. Launchpad (2 of 2)

WA3616.3

### Notes:

***Instructor notes:***

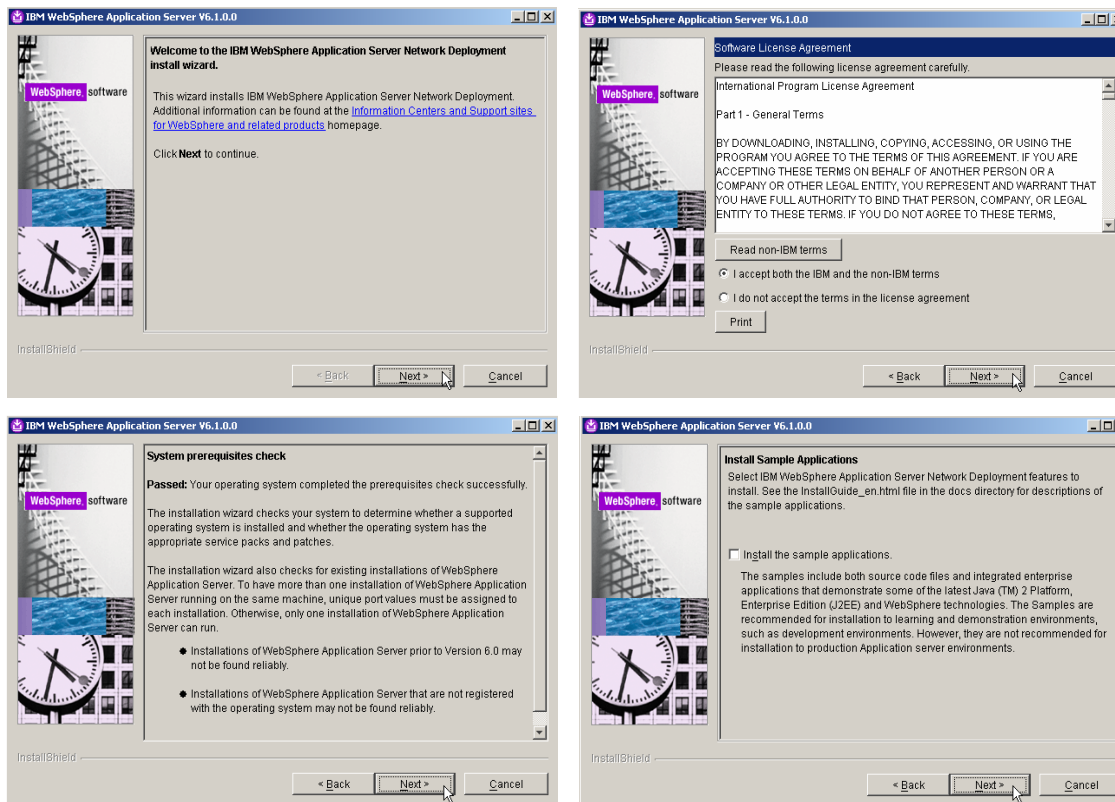
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Installation – Startup (1 of 2)

# Installation – Startup (1 of 2)



© Copyright IBM Corporation 2007

Figure 5-14. Installation – Startup (1 of 2)

WA3616.3

## Notes:

***Instructor notes:***

**Purpose** — Here is the initial sequence of screens.

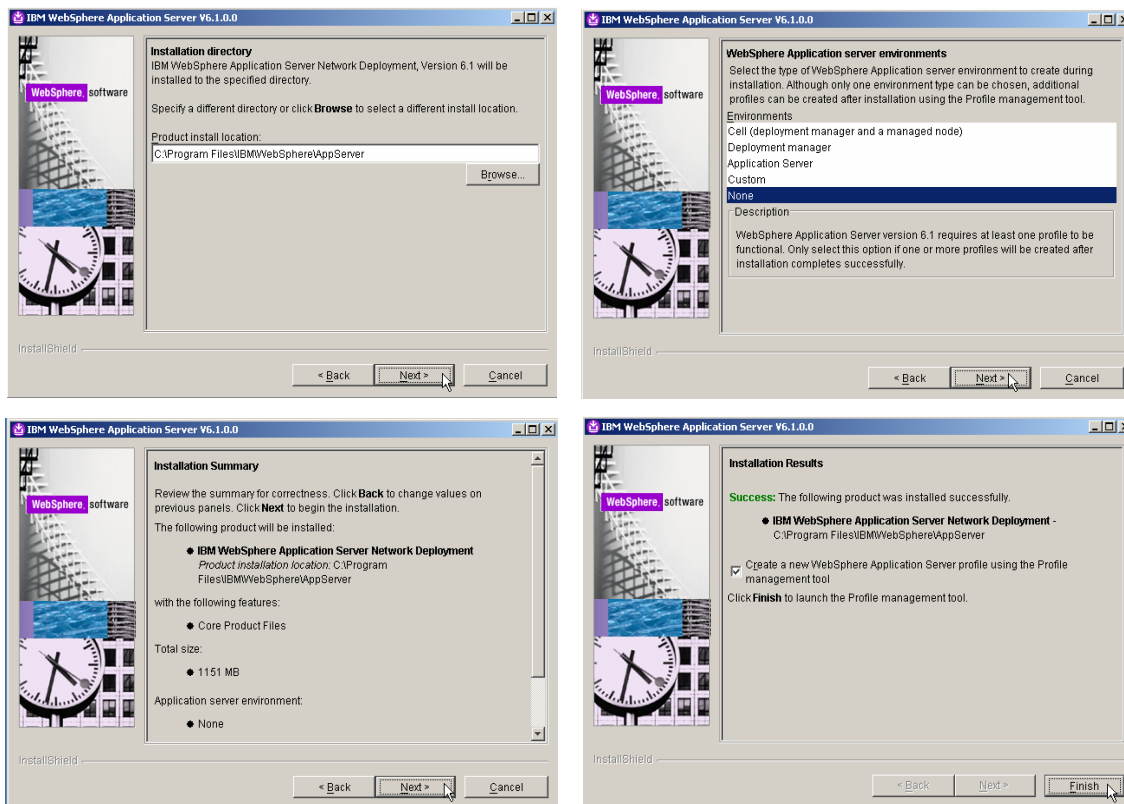
**Details** —

**Additional information** —

**Transition statement** — Next: Installation – Startup (2 of 2)



# Installation – Startup (2 of 2)



© Copyright IBM Corporation 2007

Figure 5-15. Installation – Startup (2 of 2)

WA3616.3

## Notes:

***Instructor notes:***

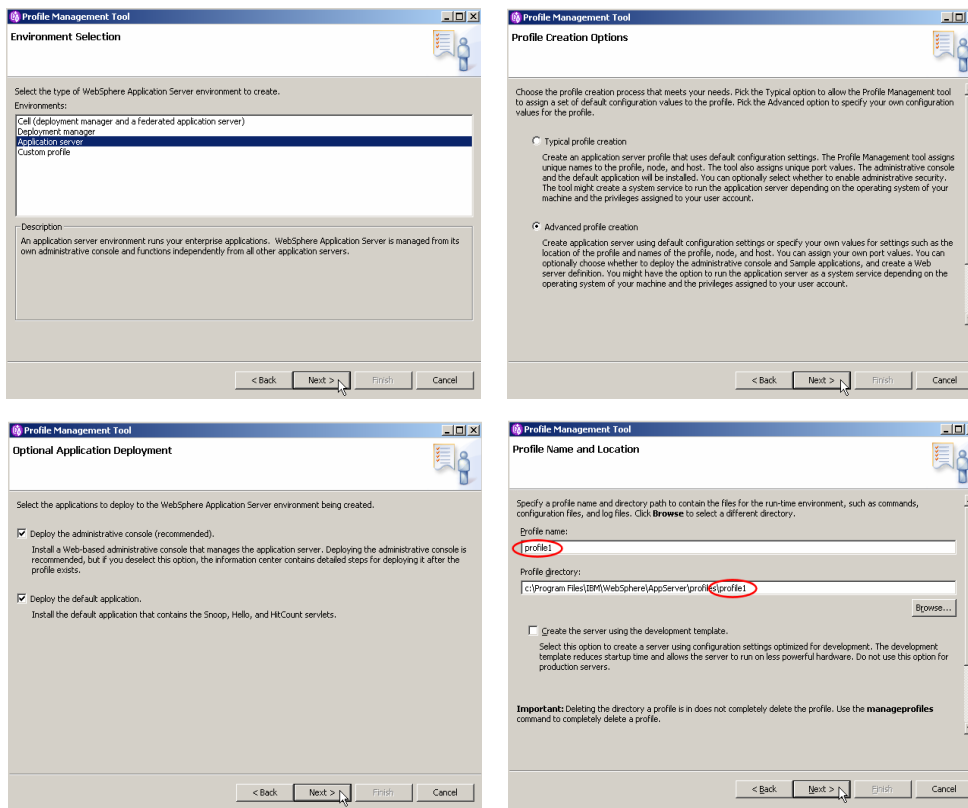
**Purpose** — Here is the sequence of screens.

**Details** —

**Additional information** —

**Transition statement** — Next: Profile Management Tool (1 of 3)

# Profile Management Tool (1 of 3)



© Copyright IBM Corporation 2007

Figure 5-16. Profile Management Tool (1 of 3)

WA3616.3

## Notes:

***Instructor notes:***

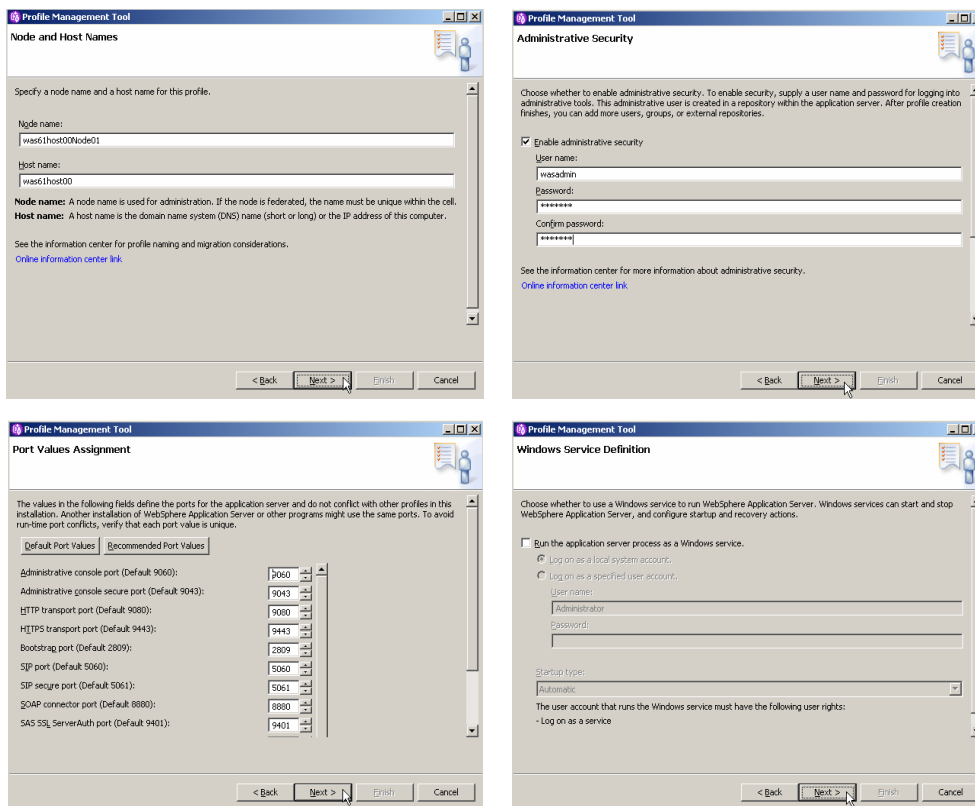
**Purpose** — Here is the initial sequence of screens for the Profile Management Tool.

**Details** —

**Additional information** —

**Transition statement** — Next: Profile Management Tool (2 of 3)

# Profile Management Tool (2 of 3)



© Copyright IBM Corporation 2007

Figure 5-17. Profile Management Tool (2 of 3)

WA3616.3

## Notes:

***Instructor notes:***

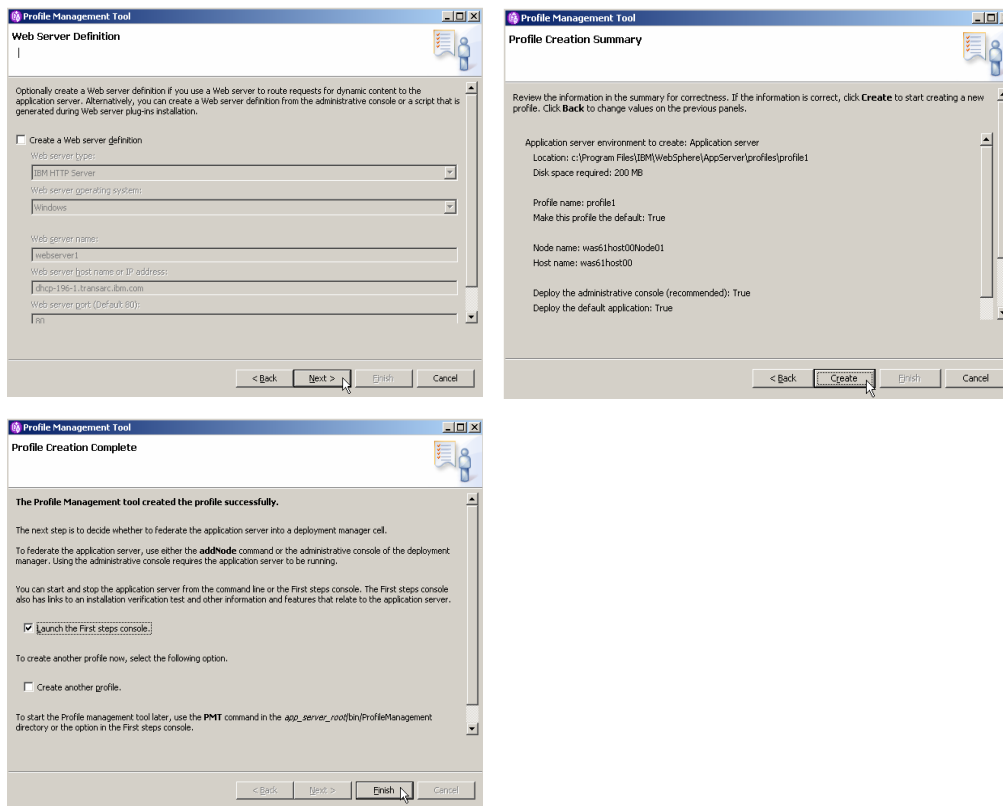
**Purpose** — Here is the initial of screens for Profile Management Tool.

**Details** —

**Additional information** —

**Transition statement** — Next: Profile Management Tool (3 of 3)

# Profile Management Tool (3 of 3)



© Copyright IBM Corporation 2007

Figure 5-18. Profile Management Tool (3 of 3)

WA3616.3

## Notes:

***Instructor notes:***

**Purpose** — Here is the initial of screens for Profile Management Tool.

**Details** —

**Additional information** —

**Transition statement** — Next: Profile creation – Command line tool



---

## Profile creation – Command line tool

---

- The manageprofiles script can
  - Create a new stand-alone application server profile

```
manageprofiles -create
-profileName
-profilePath
-templatePath
-nodeName
-cellName
-hostname
```
  - List all profiles

```
manageprofiles -listProfiles
```
  - Delete a profile

```
manageprofiles -delete -profileName
```

    - If desired, delete the left over files in the profile directory

© Copyright IBM Corporation 2007

Figure 5-19. Profile creation – Command line tool

WA3616.3

### Notes:

Deleting a profile leaves a number of files behind, including the logs directory. If desired, these will need to be deleted manually.

The list of profiles and their properties can be found in `<was_root>\properties\profileregistry.xml`.

Additional properties, such as log levels, can be found in `<was_root>\properties\wasprofile.properties`.

Be careful when using the Profile Management Tool. It is possible that it will preload a default host name by adding the default DNS suffix to the short machine name. This can cause problems if other profiles used only the short host name. It does not matter which form is used (shortname or fully qualified name), as long as the name is used consistently.

***Instructor notes:***

**Purpose —**

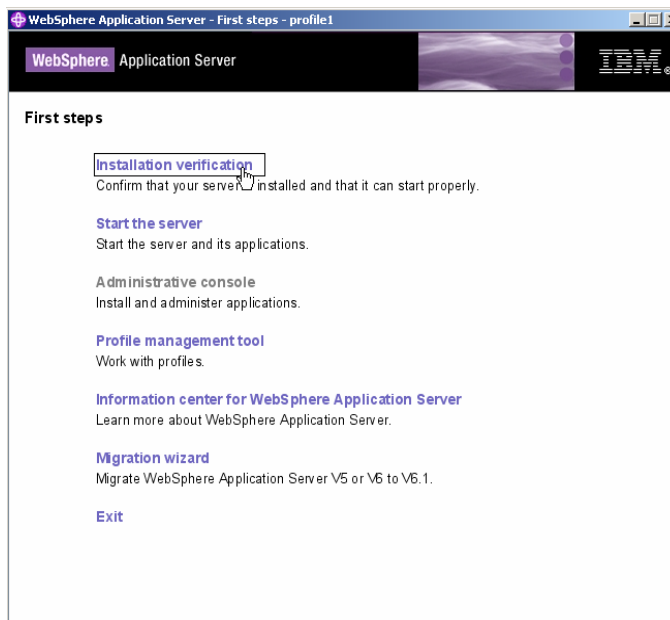
**Details —**

**Additional information —**

**Transition statement —** Next: First steps

## First steps

- Is a post-installation ease-of-use tool
- By default, starts automatically at the end of the installation
- Can be started via Windows Start menu or command line
- Can be used to verify installation
- One per profile



© Copyright IBM Corporation 2007

Figure 5-20. First steps

WA3616.3

### Notes:

First steps is a post-installation ease-of-use tool for directing WebSphere Application Server elements from one place. Options dynamically appear on the First steps panel, depending on features you install.

***Instructor notes:***

**Purpose** —

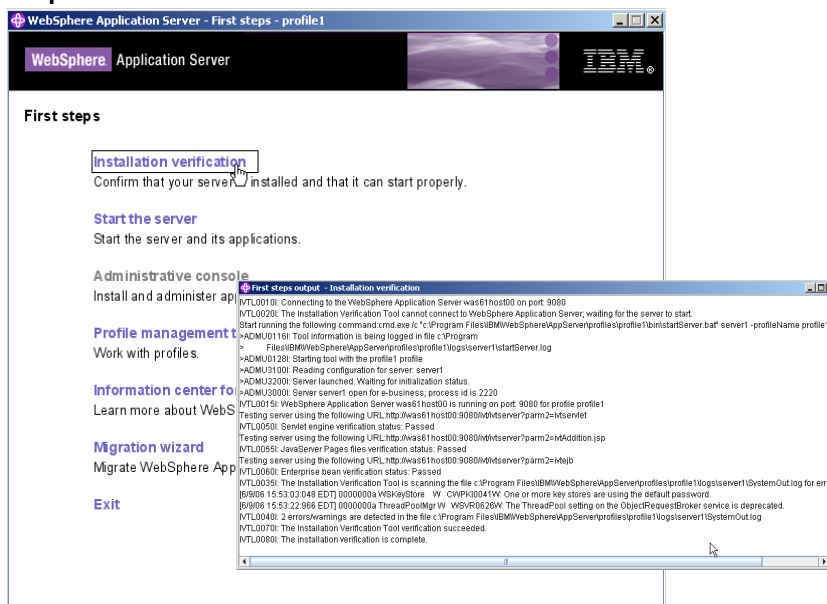
**Details** —

**Additional information** —

**Transition statement** — Next: Installation verification

## Installation verification

- After installing the product, use the installation verification test (IVT) tool
- The IVT program scans product log files for errors and verifies core functionality of the product installation



© Copyright IBM Corporation 2007

Figure 5-21. Installation verification

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Silent installation

---

## Silent installation

---

- Tailor the response file to specify the information required by the installation wizard
- Sample response files are provided
  - For Express installation: `responsefile.express.txt`
  - For Base installation: `responsefile.base.txt`
  - For Network Deployment installation: `responsefile.nd.txt`
- Tailor the response file exactly for the installation to be successful
- Issue one of these commands to use the custom response file:
  - `install.exe -options "myresponsefile.txt" -silent`

© Copyright IBM Corporation 2007

Figure 5-22. Silent installation

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Uninstall



## Uninstall

---

- Uninstall is dependent on the InstallShield MultiPlatform (ISMP) uninstaller
- Located in the WebSphere directory tree under the \_uninst directory
  - Command : `Uninstall.exe` (Windows)
- Always use the uninstaller to remove WebSphere components
  - Do not use other means like Add/Remove program to uninstall WebSphere
- Cannot custom uninstall parts of WebSphere installation (like Samples)
  - All the components are removed
- Silent uninstallation is supported
  - `Uninstall.exe –silent` (Windows)

© Copyright IBM Corporation 2007

Figure 5-23. Uninstall

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Applications installed during installation

## Applications installed during installation

---

- Occurs after WebSphere Application Server is installed and an application server profile is configured
- Multiple EAR files are installed
  - Default application
  - Installation verification test
  - Query
  - Samples (if selected)
- Optional samples include
  - PlantsByWebSphere
  - SamplesGallery
  - SchedulerCalenders
- System applications
  - adminconsole
  - filetransfer

© Copyright IBM Corporation 2007

Figure 5-24. Applications installed during installation

WA3616.3

### Notes:

A *system application* is a J2EE enterprise application that is central to a WebSphere Application Server product.

Examples of system applications include *adminconsole* and *filetransfer*.

Because a system application is an important part of a WebSphere Application Server product, a system application is deployed when the product is installed and is updated only through a product fix or upgrade. Users cannot change the metadata for a system application such as its J2EE bindings or J2EE extensions, unless the metadata assigns users and groups for security purposes. Non-security related metadata requiring a change must be updated through a product fix or upgrade.

System applications are not shown in the list of installed applications on the console Enterprise Applications page, or through wsadmin and Java application programming interfaces, to prevent users from accidentally stopping, updating or removing the system applications.

Note that J2EE Samples are not system applications even though they are provided as part of a WebSphere Application Server product. Similarly, applications that support changes to their metadata are not system applications. The samples gallery includes the following materials:

### **Plants by WebSphere application**

This application demonstrates several Java 2 Platform, Enterprise Edition (J2EE) functions, using an online store that specializes in plant and garden tool sales. The Greenhouse by WebSphere application can be used with this application to process its backorders.

### **WebSphere Bank application**

This application demonstrates multiple J2EE 1.4 technologies in a banking scenario.

### **Greenhouse by WebSphere application**

This application demonstrates multiple J2EE 1.4 technologies in a business-to-business (B2B) supplier scenario. It can process backorders from the Plants by WebSphere application.

### **Technology Samples**

These Samples demonstrate enterprise beans, servlets, JavaServer Pages technology, J2EE application client, security, and message-driven beans.

### **Web Services Samples**

These Samples demonstrate J2EE beans and JavaBeans components that are available as Web services.

### **Service Data Objects (SDO) Sample**

This Sample demonstrates data access to a relational database through Service Data Objects (SDO) and Java DataBase Connectivity (JDBC) Mediator technologies.

### **Been There Workload Management Sample**

This Sample demonstrates the workload management and clustering capabilities of the product, and is the only Sample that is supported in a multi-server, clustered environment.

### **Java Adventure Builder application**

This application demonstrates J2EE 1.4 technology using an online travel shopping store.

### **JACL scripts**

These scripts enable you to configure resources and install the Sample applications.

### **Programming model extensions Samples in the Samples Gallery**

These Samples demonstrate WebSphere programming model extension features such as dynamic query service, work area service, internationalization service,

ActivitySessions service, application profiling, Java Transaction API (JTA) extensions, asynchronous beans, and scheduler.

***Instructor notes:***

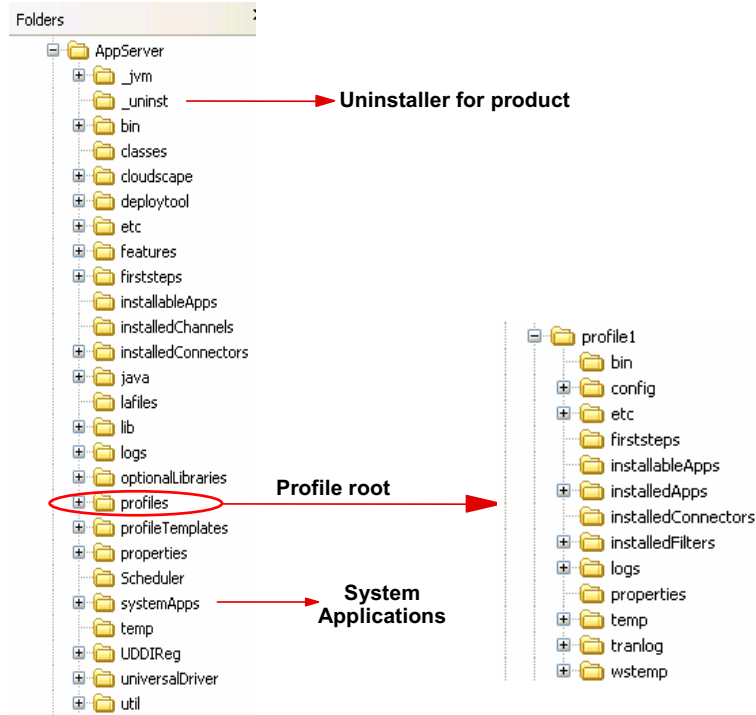
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Directory structure

# Directory structure



© Copyright IBM Corporation 2007

Figure 5-25. Directory structure

WA3616.3

## Notes:

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** — Students will browse the directory structure in the labs.

**Transition statement** — Next: Common command line tools



---

## Common command line tools

---

- Tools in bin directory
  - `versionInfo` – displays installed product versions
  - `startServer` – starts a server
  - `stopServer` – stops a server
  - `serverStatus` – displays server status

© Copyright IBM Corporation 2007

Figure 5-26. Common command line tools

WA3616.3

### **Notes:**

These are tools that are commonly used in a cell environment.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Server commands

---

## Server commands

---

- WebSphere commands are profile aware
  - There is a `-profileName` option on many WebSphere V6.x commands
  - Or issue the commands from the appropriate `<profile_root>\<profile>\bin` directory
  
- If no profile is used, the default profile is assumed
  - Can be only one default profile
  
- Examples:
  - `startServer server1 -profileName Node1`
  - `startManager -profileName DMgr`
  - `stopServer server1`
    - Assumes default profile

© Copyright IBM Corporation 2007

Figure 5-27. Server commands

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

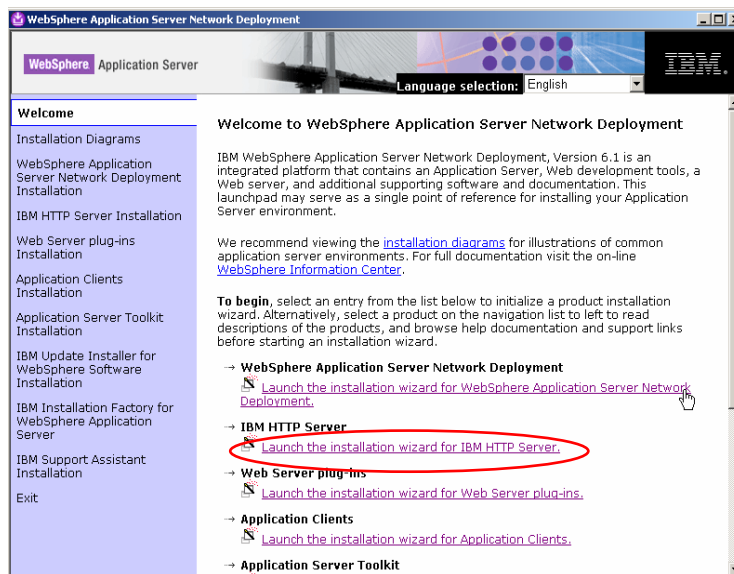
**Details** —

**Additional information** — This will also be covered in more details in a later lecture.

**Transition statement** — Next: IBM HTTP Server

# IBM HTTP Server

- Installation started using Launchpad
- IBM HTTP Server V6.1 is based on Apache 2.0.47
- Some of the administrative functionality is integrated into the WebSphere V6.1 administrative console



© Copyright IBM Corporation 2007

Figure 5-28. IBM HTTP Server

WA3616.3

## Notes:

The following are the supported Web servers for WebSphere Application Server V6.1:

Apache HTTP Server

Domino Web Server

IBM HTTP Server

Microsoft Internet Information Services

Sun Java System Web Server (formerly Sun ONE and iPlanet)

***Instructor notes:***

**Purpose —**

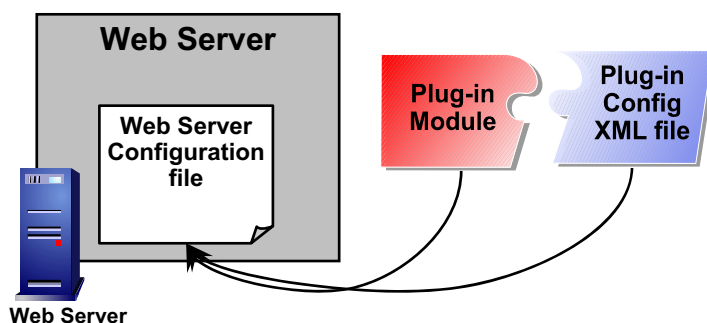
**Details —**

**Additional information —**

**Transition statement —** Next: Web server plug-in Installer

## Web server plug-in Installer

- Plug-in is a separate installation
- Typical plug-in installation action
  - Installs the plug-in binaries for the Web server
  - Updates the Web server configuration file
    - Specifies the loading of the plug-in module
    - Specifies the location of plug-in configuration file (plugin-cfg.xml)



© Copyright IBM Corporation 2007

Figure 5-29. Web server plug-in Installer

WA3616.3

### Notes:

To use a Web server other than IBM HTTP Server Version 6.1, install and configure the Web server before or after installing the WebSphere Application Server product, but before installing the Web server plug-ins for WebSphere Application Server. After installing the Web server and the Application Server, install the appropriate Web server plug-in for a supported, installed Web server. No further configuration is required for most Web servers.

***Instructor notes:***

**Purpose —**

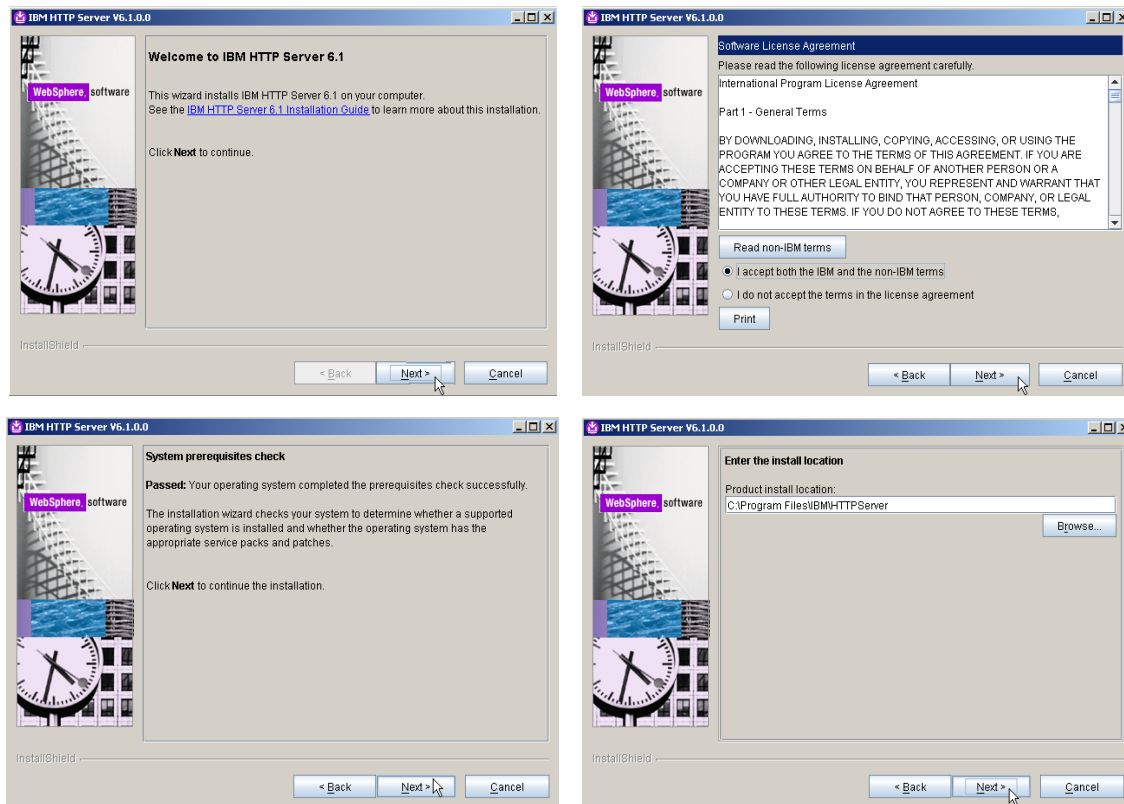
**Details —**

**Additional information —**

**Transition statement —** Next: IBM HTTP Server and plug-in installation (1 of 3)



# IBM HTTP Server and plug-in installation (1 of 3)



© Copyright IBM Corporation 2007

Figure 5-30. IBM HTTP Server and plug-in installation (1 of 3)

WA3616.3

## Notes:

***Instructor notes:***

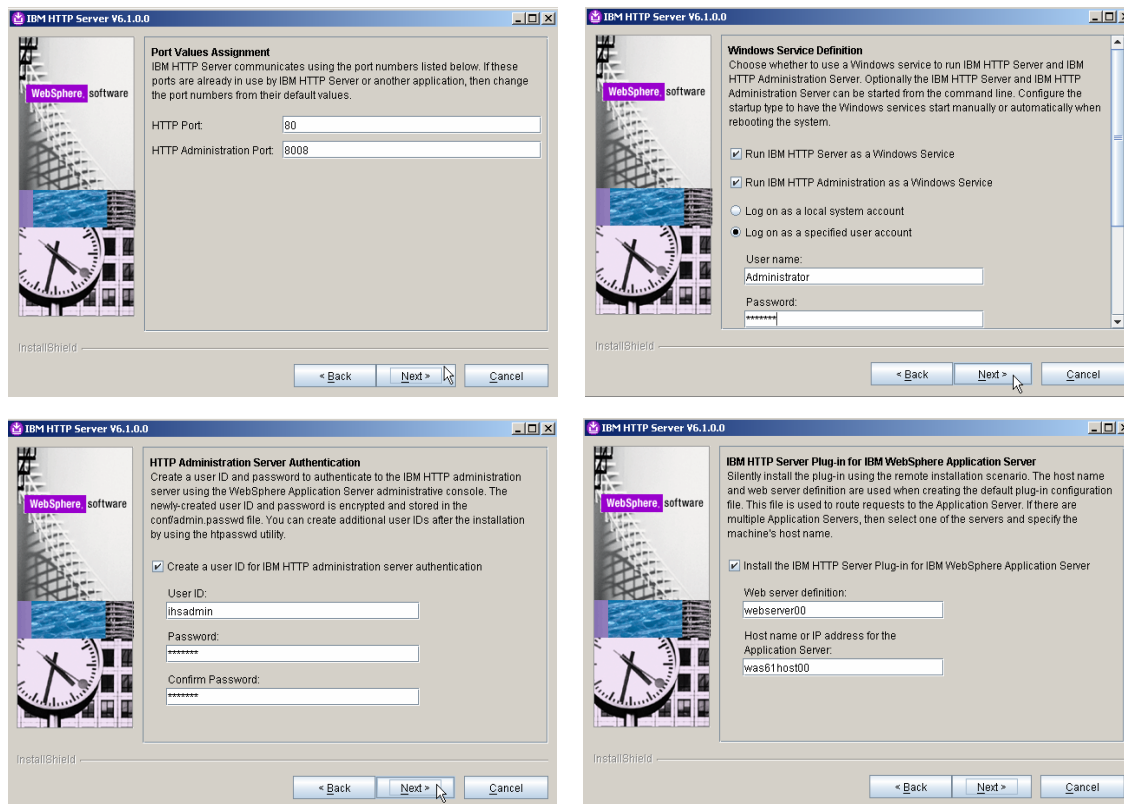
**Purpose** — Here is the initial sequence of screens.

**Details** —

**Additional information** —

**Transition statement** — Next: IBM HTTP Server and plug-in installation (2 of 3)

# IBM HTTP Server and plug-in installation (2 of 3)



© Copyright IBM Corporation 2007

Figure 5-31. IBM HTTP Server and plug-in installation (2 of 3)

WA3616.3

## Notes:

***Instructor notes:***

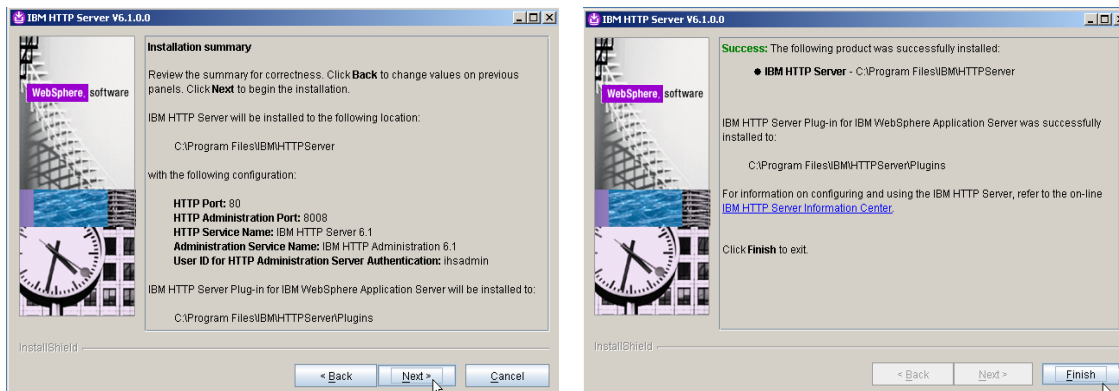
**Purpose** — Here is the initial sequence of screens.

**Details** —

**Additional information** —

**Transition statement** — Next: IBM HTTP Server and plug-in installation (3 of 3)

# IBM HTTP Server and plug-in installation (3 of 3)



© Copyright IBM Corporation 2007

Figure 5-32. IBM HTTP Server and plug-in installation (3 of 3)

WA3616.3

## Notes:

***Instructor notes:***

**Purpose** — Here is the initial sequence of screens.

**Details** —

**Additional information** —

**Transition statement** — Next: IBM HTTP Server: GUI and silent installation

---

## IBM HTTP Server: GUI and silent installation

---

- GUI install
  - Started from the launchpad
  - From the install executable from the IBM HTTP Server directory
- Silent install
  - Uses a response file as an input
  - `install -options "responsefile.txt" -silent`
- Option to change the install location
  - Support for local and remote Web server installs
- User ID and password with appropriate authority is required for Windows platform

© Copyright IBM Corporation 2007

Figure 5-33. IBM HTTP Server: GUI and silent installation

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**


**Transition statement —** Next: Installation Factory - the problem



# Installation Factory - the problem

- Installing and configuring WebSphere usually requires many steps:

- Full install of WebSphere: Select features and provide other install-time input  V6.0

- Install refresh pack (if needed)  V6.0.2

- Install fix packs (if needed)  V6.0.2.5  JDK 1.4.2 SR3

- Install one or more interim fixes (if needed)   iFix "A" and "B"

- Create and configure application servers and other artifacts

- Deploy applications  J2EE app  wsadmin script

*Typically need to iterate over these steps to achieve desired end result; this is not a pure linear process.*



© Copyright IBM Corporation 2007

Figure 5-34. Installation Factory - the problem

WA3616.3

## Notes:

***Instructor notes:***

**Purpose** —

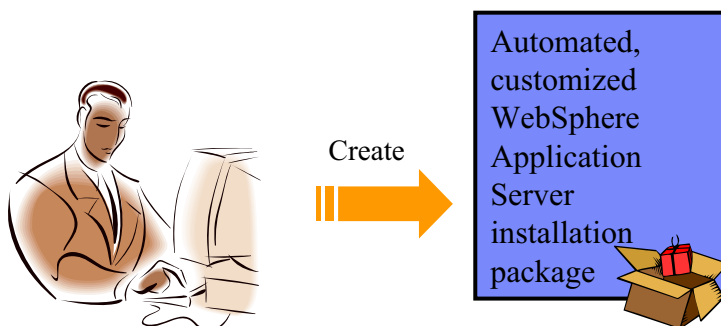
**Details** —

**Additional information** —

**Transition statement** — Next: Installation Factory: custom solutions

## Installation Factory: custom solutions

- Users need a simpler approach
  - Independent Software Vendors (ISVs)
  - Customers
  - IBM Products
- Different groups need to create fast, reliable and repeatable WebSphere Application Server installs that are tailored to their specific needs



© Copyright IBM Corporation 2007

Figure 5-35. Installation Factory: custom solutions

WA3616.3

### Notes:

Many types of WebSphere users need a solution to simplify this installation and configuration process. Independent software vendors, customers, and even other IBM products can all benefit from having an easier way to quickly install and update WebSphere Application Server. For example, another IBM product, such as Portal, may need to first have a specific version of WebSphere Application Server installed before the product can be installed. A simpler method for installing and updating WebSphere Application Server can actually improve the installation for these other products. For users of these products this type of solution can result in increased productivity.

***Instructor notes:***

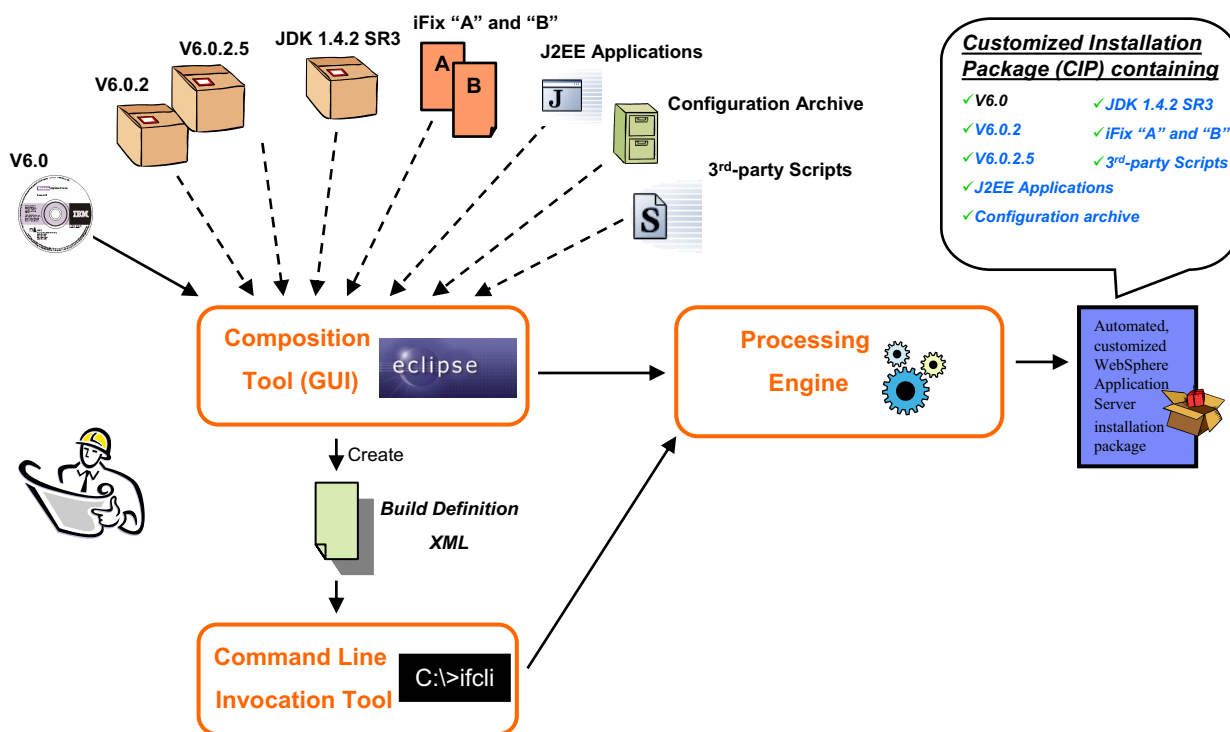
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Installation Factory creation-time capabilities

# Installation Factory creation-time capabilities



© Copyright IBM Corporation 2007

Figure 5-36. Installation Factory creation-time capabilities

WA3616.3

## Notes:

***Instructor notes:***

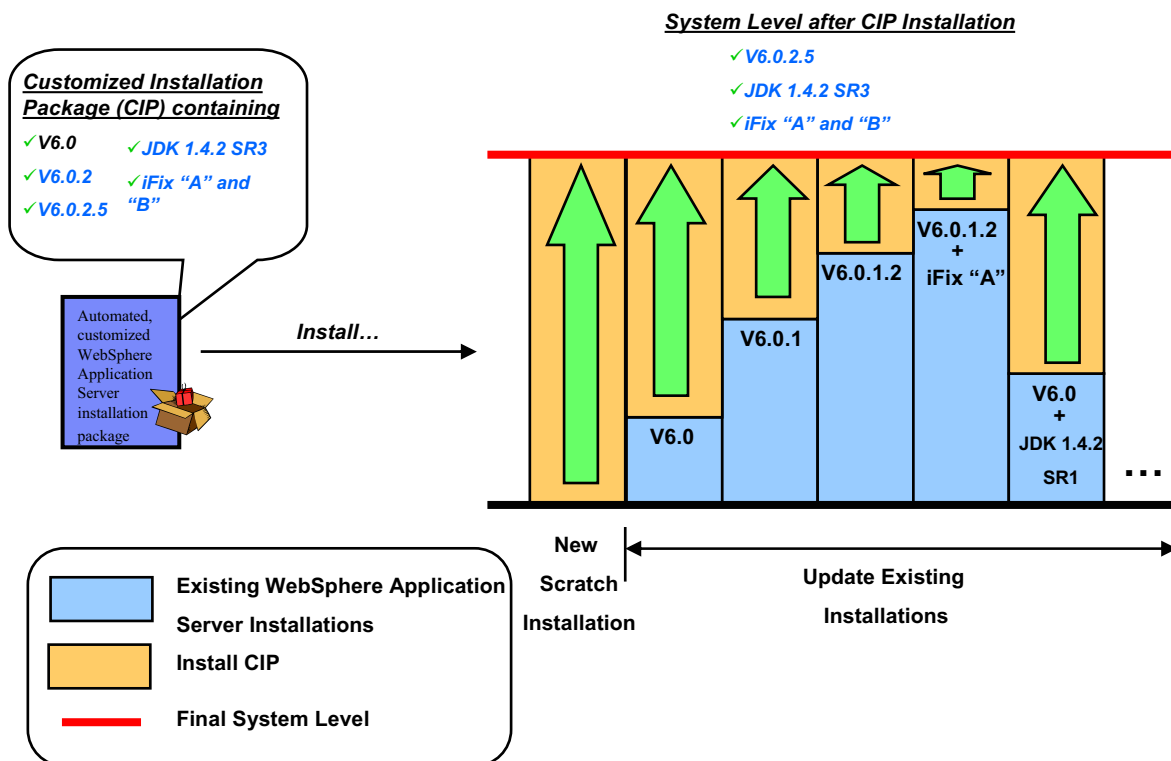
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Installation Factory example

# Installation Factory example



© Copyright IBM Corporation 2007

Figure 5-37. Installation Factory example

WA3616.3

## Notes:

A Custom Install Package or CIP can be used to upgrade an existing installation of WebSphere Application Server. In the example illustrated on this slide a CIP has been built that contains WebSphere Application Server V6.0, the V6.0.2 maintenance package, the fix pack for V6.0.2.5, an upgrade for the JDK, and two I-Fixes. When this custom installation package is installed on the various systems shown on the right, they will be upgraded to the same maintenance level as the custom installation package. The chart illustrates that the custom installer will only install what is needed to upgrade the installations.

***Instructor notes:***

**Purpose** —

**Details** —

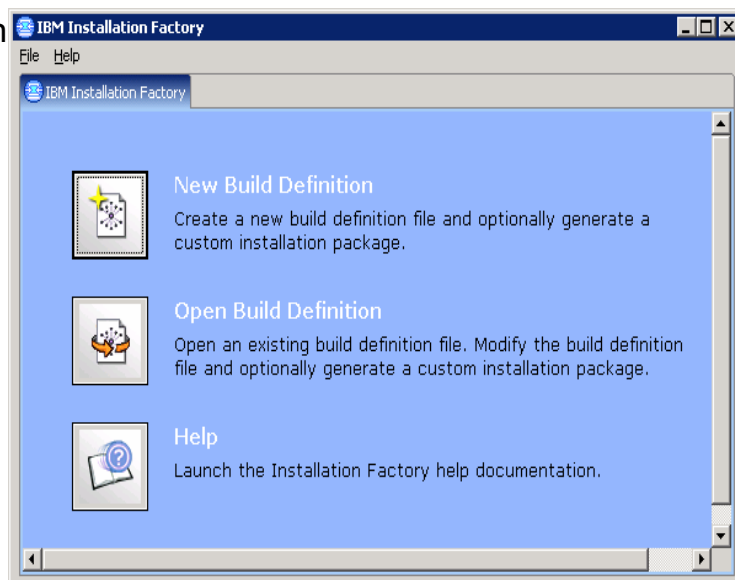
**Additional information** —

**Transition statement** — Next: Installation Factory opening panel



## Installation Factory opening panel

- Create a new build definition or work with an existing one
  - XML file that describes the CIP
- A number of panels are used to select the type of software package for which to make a CIP



© Copyright IBM Corporation 2007

Figure 5-38. Installation Factory opening panel

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Checkpoint questions

## Checkpoint questions

---

1. What are the types of profiles that can be configured?
2. Which log file is used to verify the installation was successful?
3. What is IVT and how can it be accessed?

© Copyright IBM Corporation 2007

Figure 5-39. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.

## **Instructor notes:**

**Purpose** —

**Details** —

## **Checkpoint solutions**

---

1. For the application server, an application server profile. For Network Deployment, you may also configure deployment manager, custom, and cell profiles.
2. Examine the log file *install\_root\logs\install\log.txt*
3. IVT is an installation verification tool that will scan log files to look for errors that may have occurred during installation. It can be launched from the command line or from the First steps console of each profile.

© Copyright IBM Corporation 2007

**Additional information** —

**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe the installation process for WebSphere Application Server
- Verify the installation of WebSphere Application Server
- Describe WebSphere profiles
- Describe the directories and configuration files for WebSphere Application Server
- Describe how to use configuration commands
- Describe how to use backup and restore configuration utilities

© Copyright IBM Corporation 2007

Figure 5-40. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

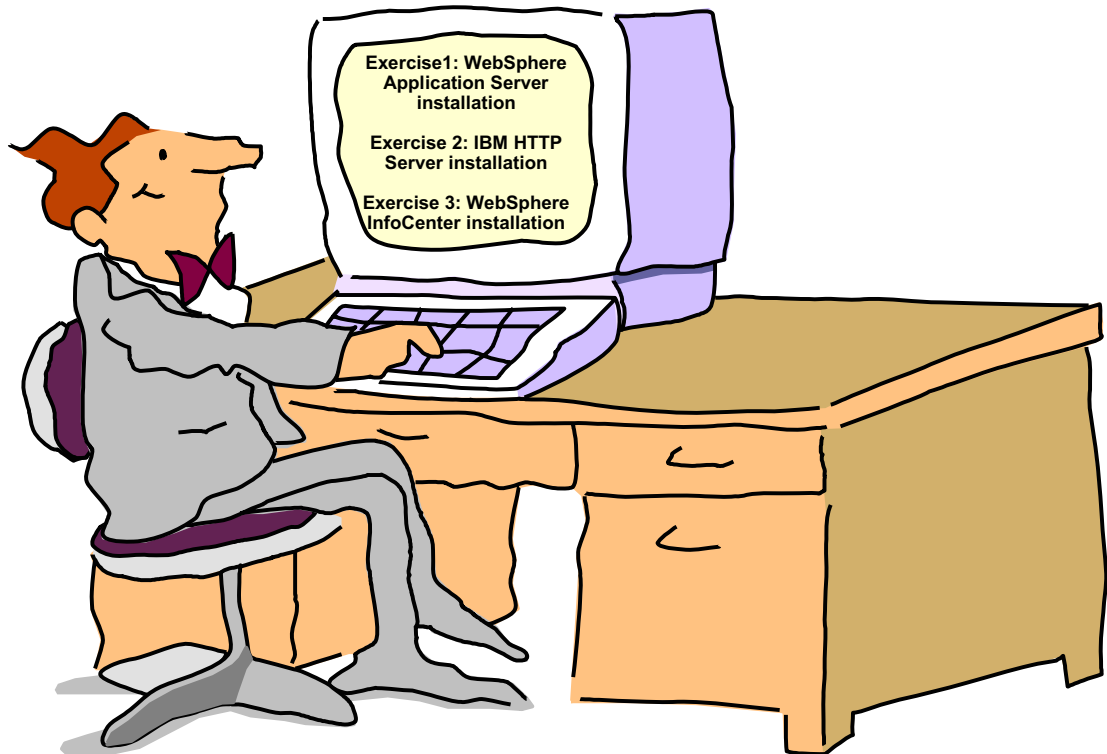
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Lab exercise

## Lab exercise



© Copyright IBM Corporation 2007

Figure 5-41. Lab exercise

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**



---

# Unit 6. WebSphere Application Server administrative console

## Estimated time

00:30

## What this unit is about

This unit describes how to interact with the WebSphere Application Server administrative console.

## What you should be able to do

After completing this unit, you should be able to:

- Describe how to access the administrative console
- Describe the administrative console interface
- Define JDBC providers and data sources
- Use IHS as a managed server
- Describe the properties of applications, Web modules, and EJB modules
- Describe the security properties of administrative security
- Describe the use of other administrative tools
- Describe the Integrated Performance Monitor
- Describe System Integration Bus, Platform Messaging, and JMS

## How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center



## WebSphere Application Server administrative console

© Copyright IBM Corporation 2007

Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

4.0.2

Figure 6-1. WebSphere Application Server administrative console

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe how to access the administrative console
- Describe the administrative console interface
- Define JDBC providers and data sources
- Use IHS as a managed server
- Describe the properties of applications, Web modules, and EJB modules
- Describe the security properties of administrative security
- Describe the use of other administrative tools
- Describe the Integrated Performance Monitor
- Describe System Integration Bus, Platform Messaging, and JMS

© Copyright IBM Corporation 2007

Figure 6-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose** —

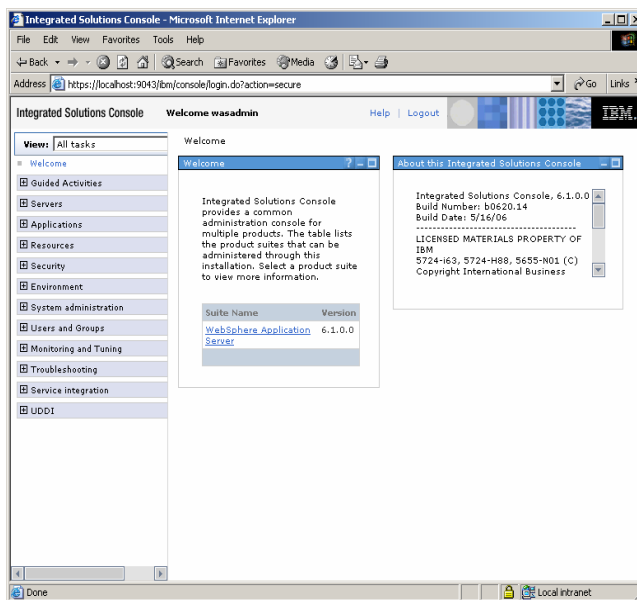
**Details** —

**Additional information** —

**Transition statement** — Next: Administrative console

## Administrative console

- Web browser-based tool that is used to manage IBM WebSphere Application Server
  - Supports a full range of product administrative activities



© Copyright IBM Corporation 2007

Figure 6-3. Administrative console

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

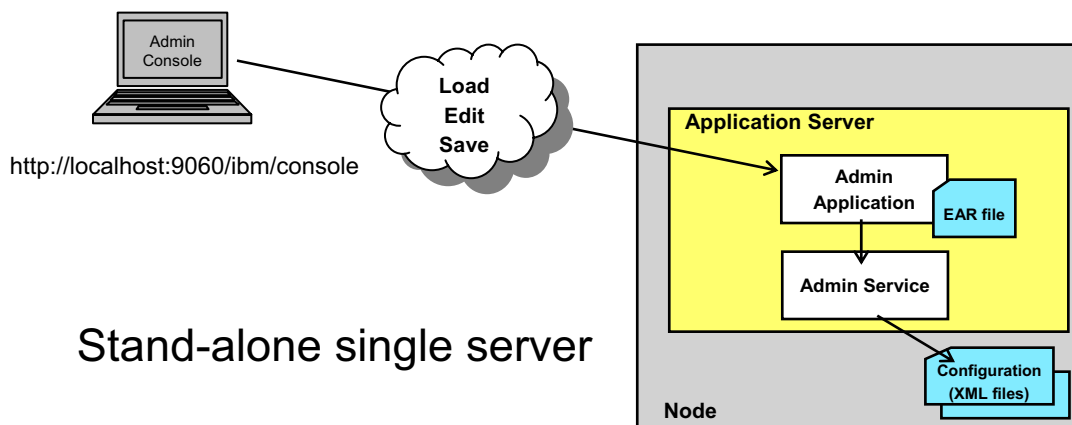
**Additional information —**

**Transition statement —** Next: Starting the administrative console



## Starting the administrative console

- The AdminApplication
  - Runs within an application server profile on a node
  - Is installed as a system application
    - Cannot be managed through console
    - Not shown in listing of installed applications
  - Is protected by WebSphere security
- Is accessed through **http://localhost:9060/ibm/console**



© Copyright IBM Corporation 2007

Figure 6-4. Starting the administrative console

WA3616.3

### Notes:

***Instructor notes:***

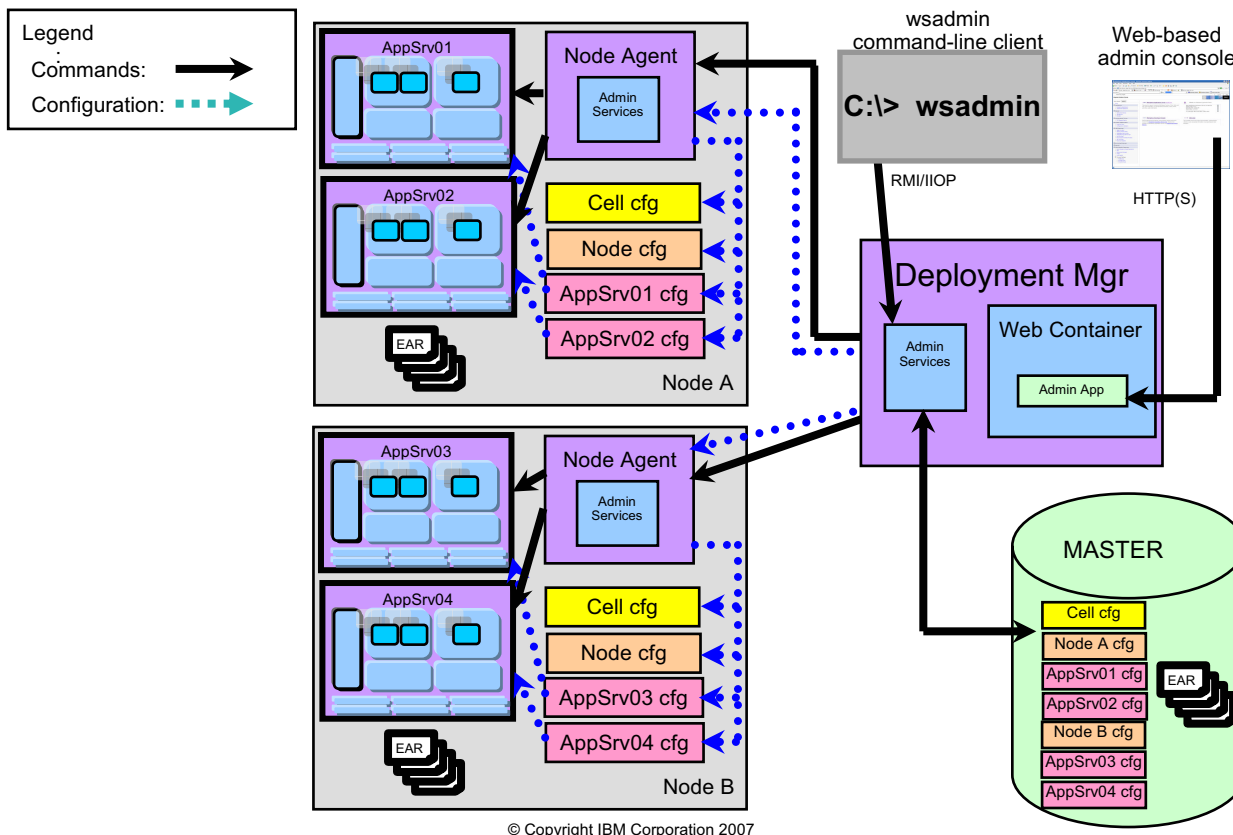
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Administrative console in a cell topology

# Administrative console in a cell topology



© Copyright IBM Corporation 2007

Figure 6-5. Administrative console in a cell topology

WA3616.3

## Notes:

The administrative console and wsadmin are still the two ways that the environment is administered. However, take note that these tools now talk to the Deployment Manager and **NOT** to the application servers directly. The communication of these commands flows from the tools to the Deployment Manager to the node agents, to the application servers. This allows administration of multiple nodes (each possibly containing multiple application servers) from a single focal point (the Deployment Manager).

There is **ONE** main repository for the configuration files within a cell, and those are associated with the Deployment Manager. All updates to the configuration files should go through the Deployment Manager. You will see in a moment how this process works. You should be very careful in connecting to an application server directly with wsadmin or the administrative console as any changes that are made to the configuration files are only temporary, they will be overwritten with the configuration files from the MASTER files.

***Instructor notes:***

**Purpose —**

**Details —** This slide is not animated. However, you may wish to walk through the administrative process like was done with the animated slide in the last architecture unit.

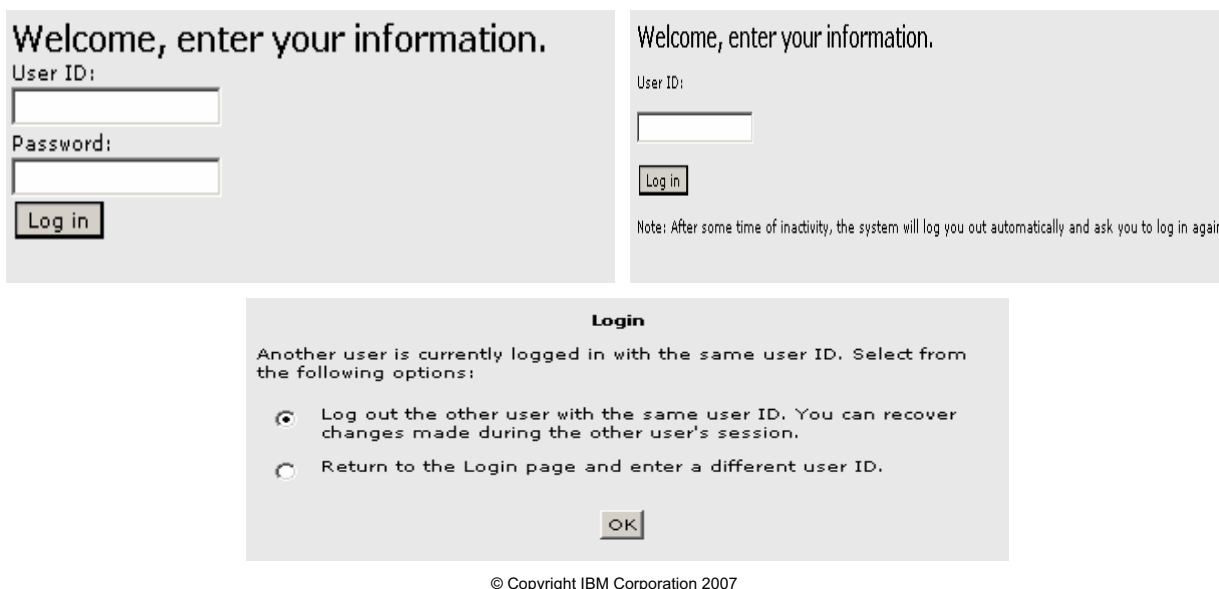
Make sure to point out all the information in the student notes.

**Additional information —**

**Transition statement —** Next: Console login

## Console login

- **User ID**
  - Specifies a string identifying the user
  - Used to track changes made by user
  - User ID must be **unique**
- **Password**
  - If security is enabled, specify a password



© Copyright IBM Corporation 2007

Figure 6-6. Console login

WA3616.3

### Notes:

Use this page to specify the user for the WebSphere Application Server administrative console. If you are using global security, then you must also specify a password.

When you specify a user, you can resume work done previously with the product. After you type in a user ID, and password if you are using global security, click **OK** to proceed to the next page and access the administrative console. **Resolving conflicts during login**

Conflicts can result if you log into the administrative console with a user ID that is already in use.

#### Another user is currently logged in with the same user name

Specifies whether to log out the user and to continue work with the user ID that is specified, or to return to the Login page and specify a different user ID, or wait for the user to log out.

This field is displayed if:

The user closed a Web browser while browsing the administrative console and did not first log out, then opened a new browser and tried to access the administrative console with the same user ID.

The user opened a Web browser to access the administrative console while accessing the administrative console in another open Web browser with the same user ID.

The user opens a Web browser and attempts to log into the console with the same user ID that is already in use by another user who logged into the console from another Web browser on another computer.

## **Recovering prior changes**

You can either recover changes that you made to the configuration from a prior session or use the master configuration. The default is to recover changes from a prior session.

### **Recover changes made in a prior session**

When enabled, this setting specifies that you want to use the same administrative configuration used for the last user's session. This option recovers changes made by the user since the last saving of the administrative configuration for the user's session.

This field is displayed only if the user changed the administrative configuration and then logged out without saving the changes.

### **Work with the master configuration**

When enabled, this setting specifies to use the default administrative configuration instead of the configuration that was last used for the user's session. Changes that are made to the user's session since the last saving of the administrative configuration are lost.

This field is displayed only if the user changed the administrative configuration and then logged out without saving the changes

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Administrative console areas

# Administrative console areas

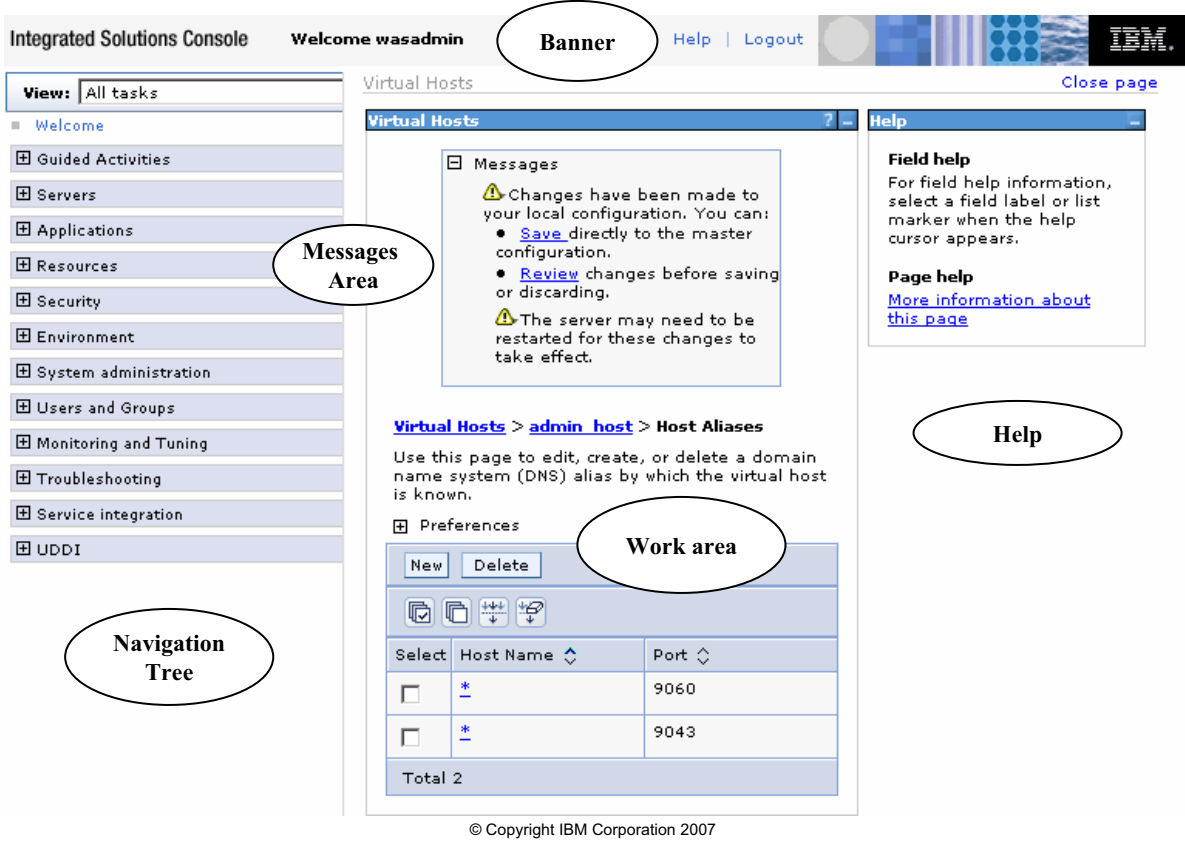


Figure 6-7. Administrative console areas

WA3616.3

## Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Administrative console navigation tree (1 of 2)

## Administrative console navigation tree (1 of 2)

⊕ Guided Activities
⊕ Servers
⊕ Applications
⊕ Resources
⊕ Security
⊕ Environment
⊕ System administration
⊕ Users and Groups
⊕ Monitoring and Tuning
⊕ Troubleshooting
⊕ Service integration
⊕ UDDI

Item	Descriptions
Guided Activities	Provides information and details for certain configuration options
Servers	Enables configuration of application servers and Web servers
Applications	Enables installation and management of applications
Resources	Enables configuration of resources and management of those resources
Security	Enables configuration and management of WebSphere security, SSL and Web services security
Environment	Enables configuration of hosts, replication domains, environment variables, naming and others

© Copyright IBM Corporation 2007

Figure 6-8. Administrative console navigation tree (1 of 2)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Administrative console navigation tree (2 of 2)

## Administrative console navigation tree (2 of 2)

+ Guided Activities	Item	Descriptions
+ Servers	System administration	Enables configuration and management of components, users and preferences
+ Applications	Users and Groups	Enables configuration of users and groups
+ Resources	Monitoring and Tuning	Enables configuration of the Performance Monitoring Infrastructure and Tivoli Performance Viewer
+ Security	Troubleshooting	Enables tracking and verification of configuration errors and problems
+ Environment	Service integration	Enables configuration with the service integration bus
+ System administration	UDDI	Enables configuration of UDDI nodes
+ Users and Groups		
+ Monitoring and Tuning		
+ Troubleshooting		
+ Service integration		
+ UDDI		

© Copyright IBM Corporation 2007

Figure 6-9. Administrative console navigation tree (2 of 2)

WA3616.3

### Notes:

***Instructor notes:***

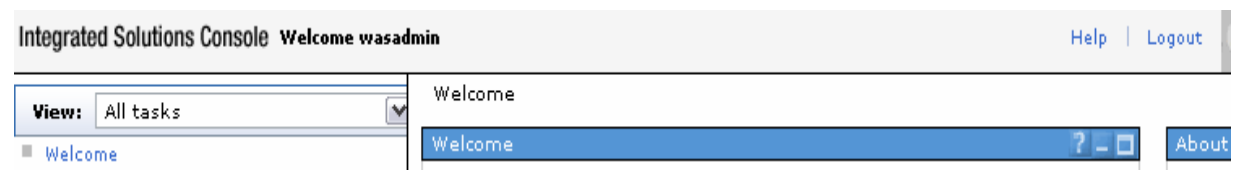
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Administrative console banner

# Administrative console banner



Item	Descriptions
<b>Welcome</b>	Displays the administrative console home page Contains links to information sources
<b>Logout</b>	Logs you out of the administrative console session Displays the Login page If changes were made and not saved the <b>Save</b> page will display
<b>Help</b>	Opens a new Web browser with detailed online help for the administrative console <b>Note:</b> This is not the Information Center

© Copyright IBM Corporation 2007

Figure 6-10. Administrative console banner

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

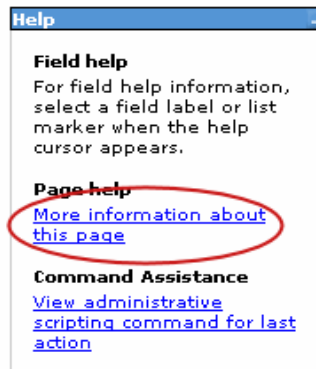
**Additional information —**

**Transition statement —** Next: Administrative console Help (1 of 2)

## Administrative console Help (1 of 2)

---

- Click **Help** on the administrative console banner and select from the list of reference topics
- Click **More Information about this page** in admin console workspace



© Copyright IBM Corporation 2007

Figure 6-11. Administrative console Help (1 of 2)

WA3616.3

### Notes:



***Instructor notes:***

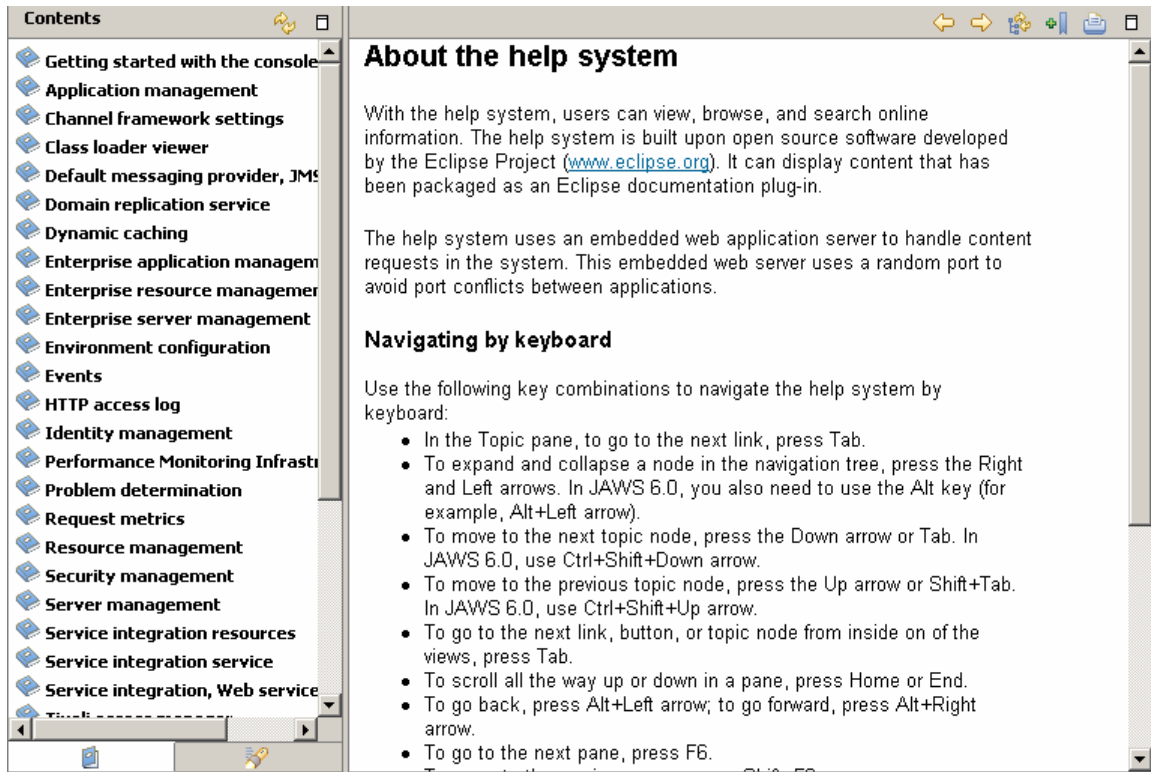
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Administrative console Help (2 of 2)

## Administrative console Help (2 of 2)



© Copyright IBM Corporation 2007

Figure 6-12. Administrative console Help (2 of 2)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

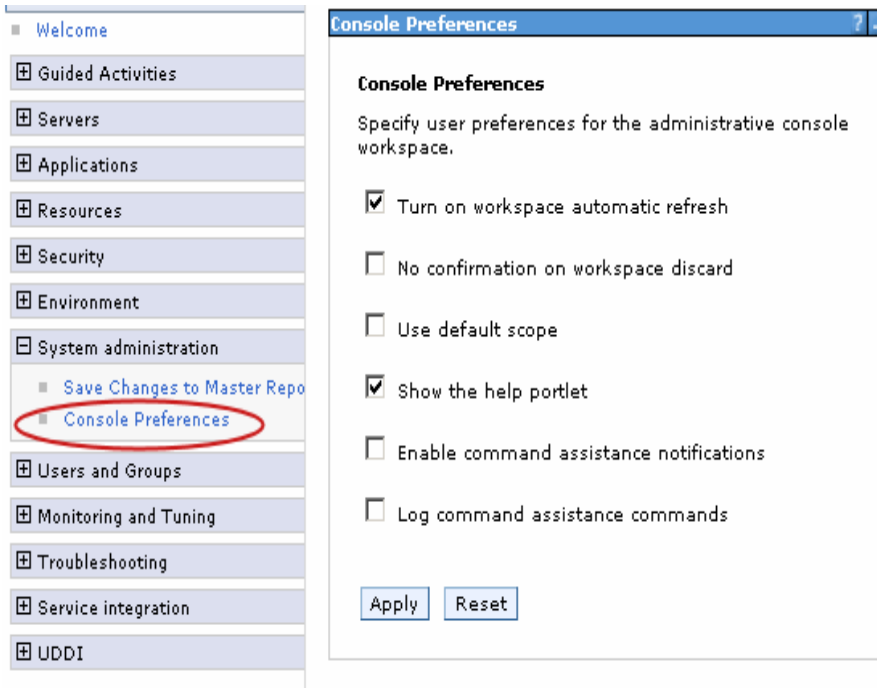
**Details —**

**Additional information —**

**Transition statement —** Next: Administrative console Preferences

# Administrative console Preferences

- Alters the look and feel of the administrative console



© Copyright IBM Corporation 2007

Figure 6-13. Administrative console Preferences

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

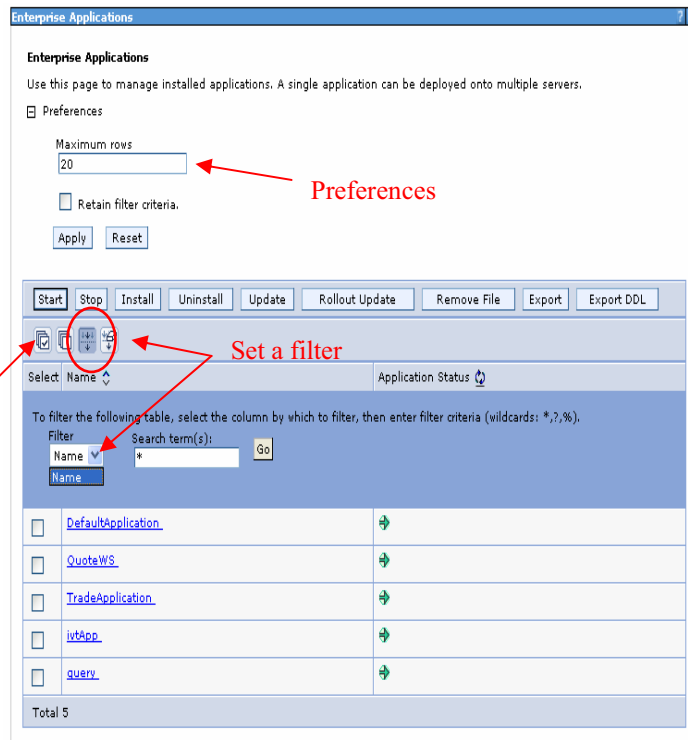
**Details —**

**Additional information —**

**Transition statement —** Next: Administrative console filters

# Administrative console filters

- Throughout the administrative console there are pages that have **Preferences**, **Filter**, and **Scope** fields
- To customize how much data is shown, select options in **Preferences**, **Filter** and **Scope**



© Copyright IBM Corporation 2007

Figure 6-14. Administrative console filters

WA3616.3

## Notes:

**Check all.** Selects each resource that is listed on the administrative console panel, in preparation for performing an action against the selected resources.

**Uncheck all.** Removes all the listed resources from each selection so that no action is performed against any of the resources.

**Filter the view.** Produces a dialog box for specifying the resources to view in the table on this administrative console page.

**Hide the filter view.** Hides the dialog box for specifying the resources to view in the table on this administrative console page.

When you produce the dialog box, select the column to filter and enter the filter criteria.

***Instructor notes:***

**Purpose —**

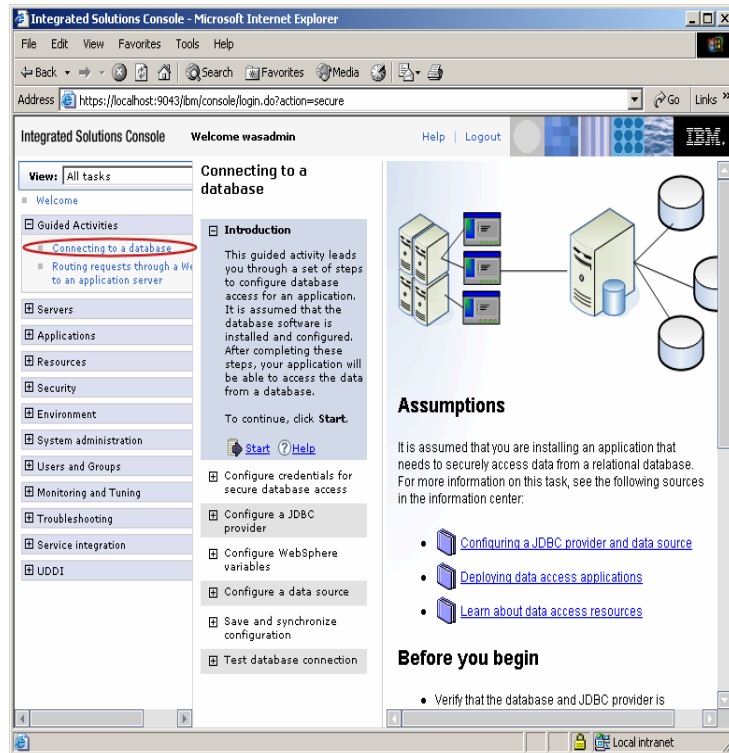
**Details —**

**Additional information —**

**Transition statement —** Next: Guided activities

## Guided activities

- Leads you through common administrative tasks that require you to visit multiple administrative console pages



© Copyright IBM Corporation 2007

Figure 6-15. Guided activities

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

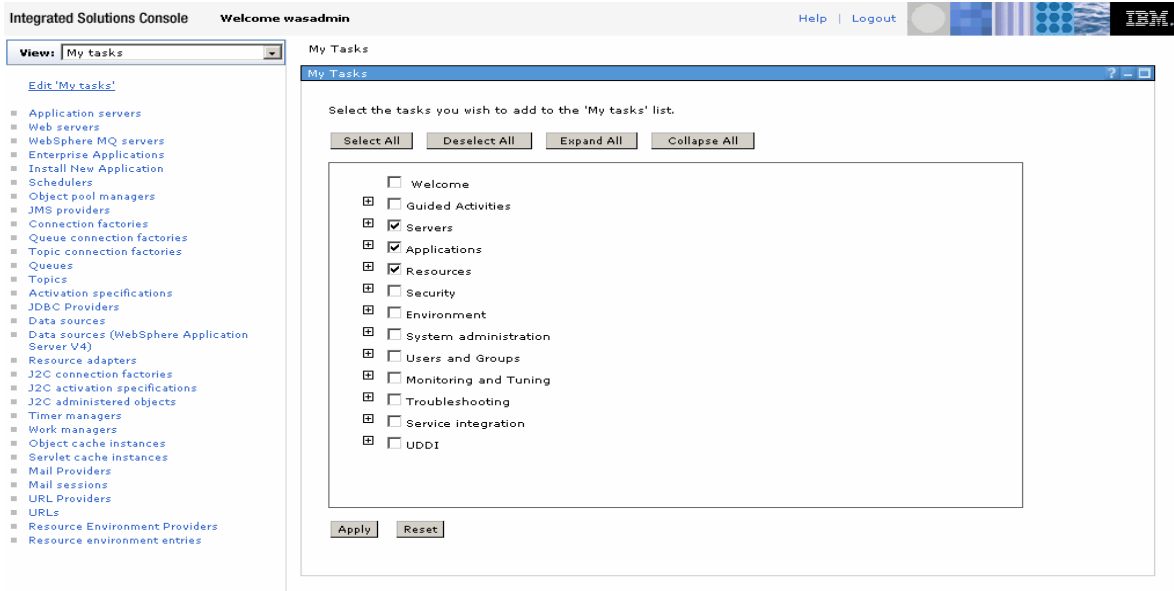
**Details —**

**Additional information —**

**Transition statement —** Next: My tasks

# My tasks

- Create and edit a list of tasks to view in the navigation tree
- Initially displayed each time you log into the administrative console after customization



© Copyright IBM Corporation 2007

Figure 6-16. My tasks

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

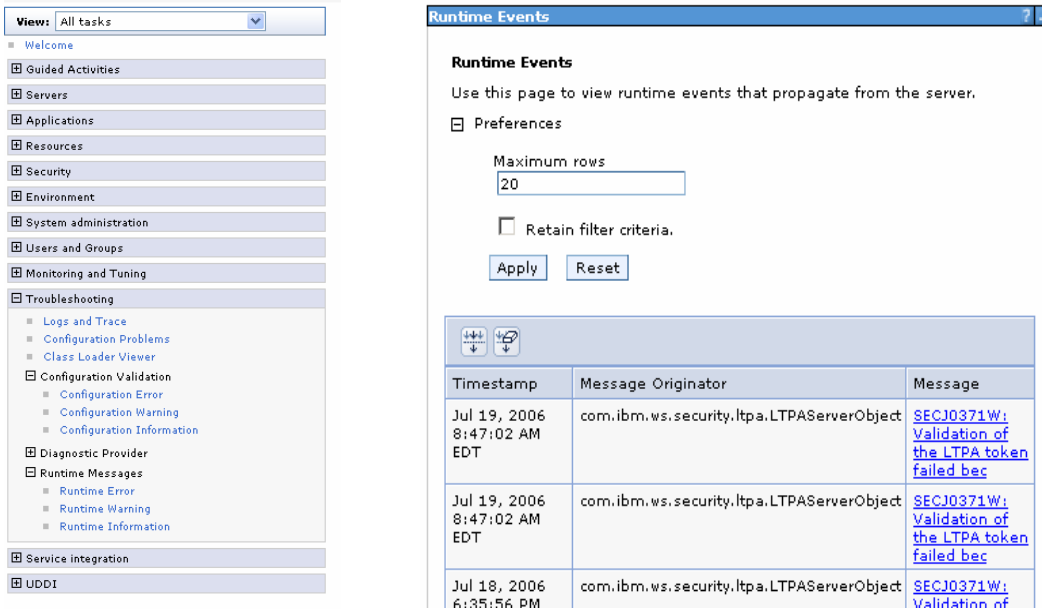
**Details —**

**Additional information —**

**Transition statement —** Next: Troubleshooting information

# Troubleshooting information

- Configuration problems and runtime messages now available under Troubleshooting



© Copyright IBM Corporation 2007

Figure 6-17. Troubleshooting information

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

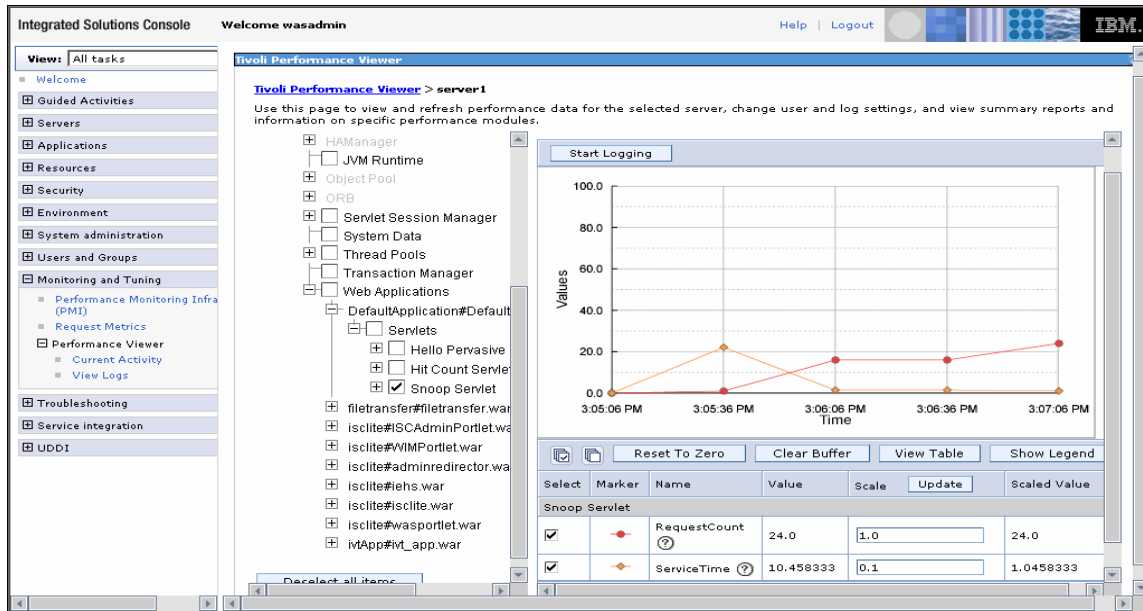
**Details —**

**Additional information —**

**Transition statement —** Next: Tivoli Performance Viewer

# Tivoli Performance Viewer

- Addition of integrated browser-based performance viewer and advisor



© Copyright IBM Corporation 2007

Figure 6-18. Tivoli Performance Viewer

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Checkpoint questions

## Checkpoint questions

---

1. The AdminApp application can be configured using the administrative console?
  - A. True
  - B. False
  
2. Enterprise applications can be installed using which of the following?
  - A. Scripts
  - B. wsadmin
  - C. Administrative console
  - D. WebSphere Rapid Deployment-admin (wrd-admin)

© Copyright IBM Corporation 2007

Figure 6-19. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.



**Instructor notes:**

Purpose —

Details —

## Checkpoint solutions

---

1. The AdminApp application can be configured using the administrative console?  
B. False
2. Enterprise applications can be installed using the following?  
A. Scripts  
B. wsadmin  
C. Administrative console

© Copyright IBM Corporation 2007

**Additional information** —**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe how to access the administrative console
- Describe the administrative console interface
- Define JDBC providers and data sources
- Use IHS as a managed server
- Describe the properties of applications, Web modules, and EJB modules
- Describe the security properties of administrative security
- Describe the use of other administrative tools
- Describe the Integrated Performance Monitor
- Describe System Integration Bus, Platform Messaging, and JMS

© Copyright IBM Corporation 2007

Figure 6-20. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Lab exercise

## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 6-21. Lab exercise

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**



---

## Unit 7. J2EE overview

### Estimated time

00:30

### What this unit is about

This unit provides an overview of J2EE terminology and concepts.

### What you should be able to do

After completing this unit, you should be able to:

- Describe Java basics
- Describe basic J2EE terminology
- Describe J2EE component technologies
- Describe J2EE services
- Describe J2EE messaging and communication
- Describe J2EE packaging and how it is implemented in WebSphere Application Server
- Describe how J2EE implements security
- Describe the roles defined by J2EE and the tools associated with those roles
- Describe the new enhanced EAR functionality, its implications, and how it complements the J2EE EAR file

### How you will check your progress

Accountability:

- Checkpoint

### References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center



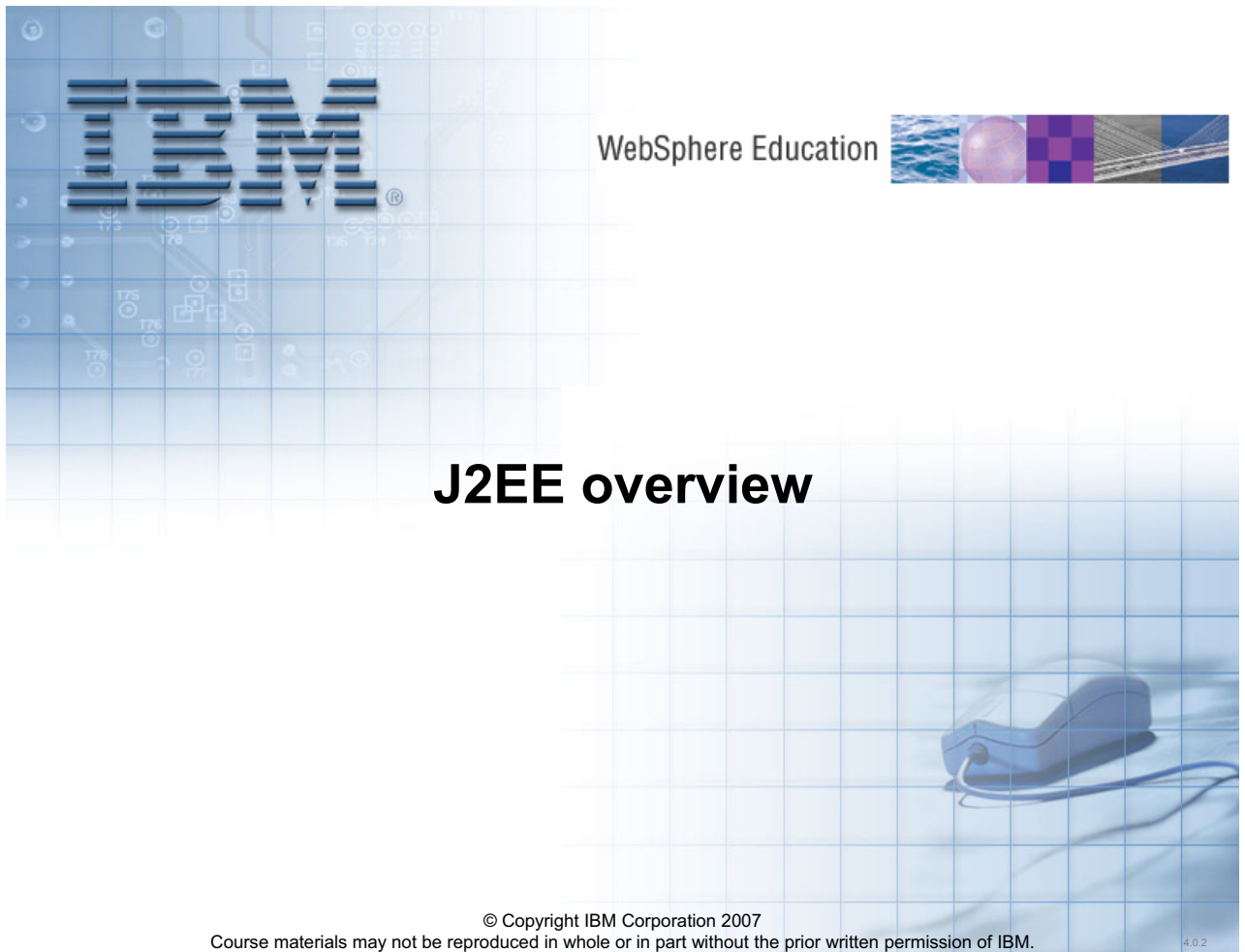


Figure 7-1. J2EE overview

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose** —

**Details** — This unit is **just** an overview. It is meant to get the soon to be WebSphere administrators in this course familiar with some of the terminology of J2EE. You should not spend an enormous amount of time on this section discussing all of the details of J2EE.

**Additional information** —

**Transition statement** — Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe Java basics
- Describe basic J2EE terminology
- Describe J2EE component technologies
- Describe J2EE services
- Describe J2EE messaging and communication
- Describe J2EE packaging and how it is implemented in WebSphere Application Server
- Describe how J2EE implements security
- Describe the roles defined by J2EE and the tools associated with those roles
- Describe the new enhanced EAR functionality, its implications, and how it complements the J2EE EAR file

© Copyright IBM Corporation 2007

Figure 7-2. Unit objectives

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Application programming model support

---

## Application programming model support

---

- Supported J2EE application versions in WebSphere Application Server V6.1
  - J2EE 1.4
  - J2EE 1.3
  - J2EE 1.2
- Makes a much better migration story by allowing your existing J2EE 1.2 or 1.3 applications to run in WebSphere Application Server V6.1
- Application running exclusively in V6.1 can use features of J2SE 5.

© Copyright IBM Corporation 2007

Figure 7-3. Application programming model support

WA3616.3

### **Notes:**

If application running on WebSphere Application Server V6.1 are not intended to also be run in lower versions, then J2SE 5 (1.5) features, such as bracketing, generics, and so on can be exploited.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: J2EE benefits

---

## J2EE benefits

---

- Standards
  - A wide range of standard services, components, clients and tools are supported.
  - Applications are portable across J2EE platforms.
- Distribution
  - Provides scalability for performance and availability.
- Common services
  - Most commonly required services provided by J2EE servers include transaction, security and persistence.
- Component model
  - Good separation of development responsibilities.
  - Good reuse of code and opportunities to share logic between applications.
- Interoperability
  - Integration with other systems using standard protocols.

© Copyright IBM Corporation 2007

Figure 7-4. J2EE benefits

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Standards supported



## Standards supported

Standards	Level	WebSphere Application Server V6.1
J2EE	1.4	✓ Fully certified and part of Sun's JCEE list
EJB	2.1	✓ EJB 2.1 and EJB 2.0 support
J2SE	1.5	✓ J2SE V5.0
Servlet	2.4	✓ Servlet 2.4
JSP	2.0	✓ JSP 2.0
JTS/JTA	1.0.1	✓ With distributed transactions
JMS	1.1	✓ With Native Provider, and MQ plug-in
JDBC	3.0	✓ 2PC across heterogeneous databases
JNDI	1.2.1	✓ JNDI 1.2 for EJB lookup and CosNaming
RMI/IIOP	1.0	✓ Fully supported
JavaMail/JAF	1.3/1.0.2	✓ Plus Domino support
SSL Security	2.0	✓ JSSE and JCE
XML JAXP	1.2	✓ XML in EJBs
J-IDL/CORBA		✓ IIOP 1.2
J2C	1.5	✓ Bean and container managed
LDAP		✓ IBM Directory Server, iPlanet, ActiveDirectory
HTTP	1.1	✓ Yes, plus across multiple Web servers
SOAP SOAP-SEC	2.2.2 1.0	✓ SOAP support for Web Services.
COM/ASP Support		✓ With Java wrapping and proxy
JMX	1.0	✓ JMX
XML4J	4.0	✓ XML support
XSL	2.3	✓ XSL parser

© Copyright IBM Corporation 2007

Figure 7-5. Standards supported

WA3616.3

### Notes:

#### Web services and XML support

**Standards and Portability** - XML Schema definitions for all deployment descriptors

**JAX-P 1.2** - New properties for XML parsers

**JAX-R** - XML registry API

**JAX-RPC** - APIs for representing WSDL-based services as RPCs in Java (and vice-versa)

**JSR 109** - Web services programming and deployment model

**SAAJ 1.1** - SOAP Attachments API for Java

#### Pluggable Messaging

##### EJB 2.1

Typed message beans (used for any inbound JCA including pluggable JMS provider)

Timer service Web service end-point support

##### JMS 1.1

Unification of point-to-point and pub-sub interfaces

### **J2CA 1.5**

In-bound connections (supporting pluggable JMS provider, generalized for other types)

RA life cycle support

Work manager (threads for resource adapters)

### **ISV Enablement**

#### **JMX 1.2 / JSR-077 (J2EE Management)**

Notification emitters, and standard patterns

Information model representing J2EE application server concepts

#### **JSR-088 (J2EE Deployment)**

XML-based deployment interfaces for J2EE

#### **JACC 1.0**

Java Authorization Contract with Containers

APIs for registering J2EE component authorization policies

### **Other**

#### **Servlet 2.4**

Extensible deployment descriptors

Request/response listeners

#### **JSP 2.0**

Expression Language

Simple Tag Extension

#### **JDBC 3.0**

Metadata and cursor support

#### **JavaMail 1.3 updates**

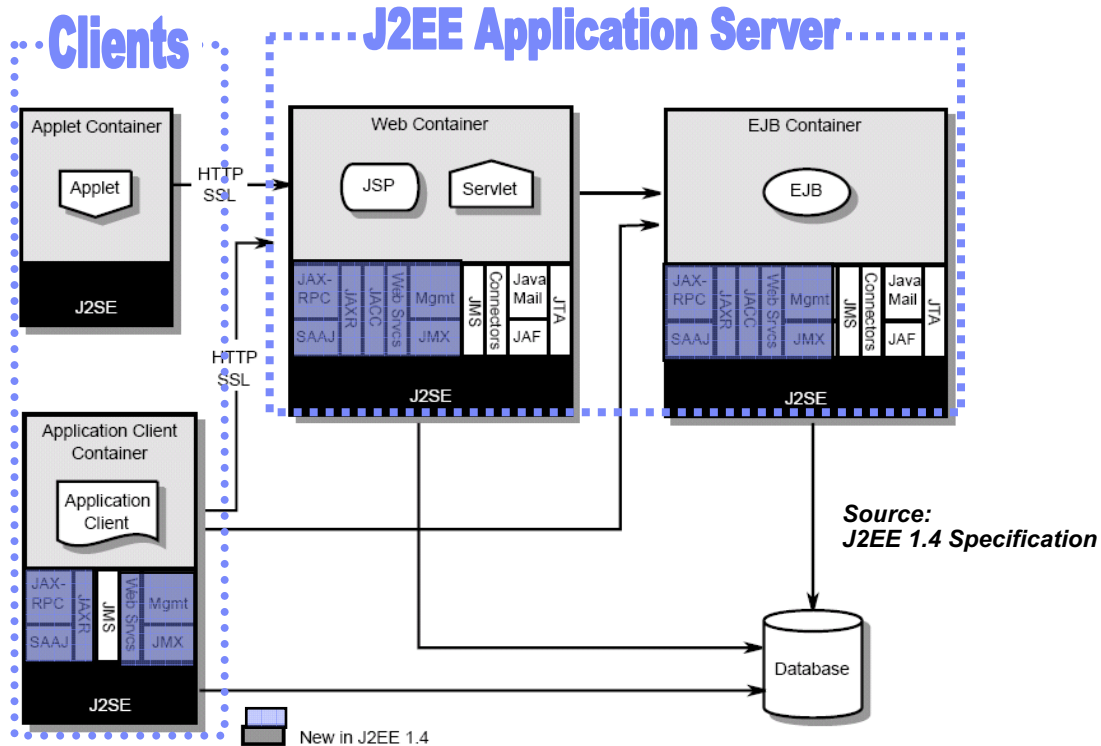
**Instructor notes:****Purpose —**

**Details —** This slide is not meant to be read to the students. This is the second time that the students have seen this list. The first time in the product overview it was just to mention that all of these are supported. Now, you can drill down into them (but just a little; do not dwell on it). Point out the major ones that will be covered and used within this course: J2EE, EJB, Server, JSP, JMS, JDBC, and so forth.

**Additional information —**

**Transition statement —** Next: J2EE 1.4 platform architecture

# J2EE 1.4 platform architecture



© Copyright IBM Corporation 2007

Figure 7-6. J2EE 1.4 platform architecture

WA3616.3

## Notes:

The diagram shown on this slide appears in the J2EE 1.4 specification document found at the following address: <http://www.jcp.org/en/jsr/all>.

The J2EE platform offers a component-based model for building distributed, multi-tiered enterprise applications. Each application is typically made up of one or more J2EE components that work together in a distributed environment. The types of components that can make up a J2EE based application are:

Client tier components (Application Clients and Applets) – Run on a client machine

Web Components (Servlets and JSPs) – Run in the Web Container within the application server

Business Components (EJBs) – Run in the EJB Container within the Application Server

Each container shown in this diagram shows the type of application component that it supports, along with services that are available to the components by the container to support various application functions.

**Instructor notes:****Purpose —**

**Details —** Do not spend a lot of time discussing how Web applications work on this slide as it is coming up in the architecture topics.

Also mention that in WebSphere Application Server V6.1 there are other containers which are not necessarily part of J2EE 1.4, such as the portlet and SIP containers.

**Additional information —**

**Transition statement —** Next: J2EE platform role overview

## J2EE platform role overview

---

- J2EE specification defines a set of roles that participate in creating, producing, and supporting information systems built on the J2EE platform.
- Roles that affect administration:
  - Application Component Provider
  - Application Assembler
  - Deployer
  - System Administrator

© Copyright IBM Corporation 2007

Figure 7-7. J2EE platform role overview

WA3616.3

### **Notes:**

There are additional roles that the J2EE specification define that are not covered within this section.

***Instructor notes:***

**Purpose —**

**Details —**

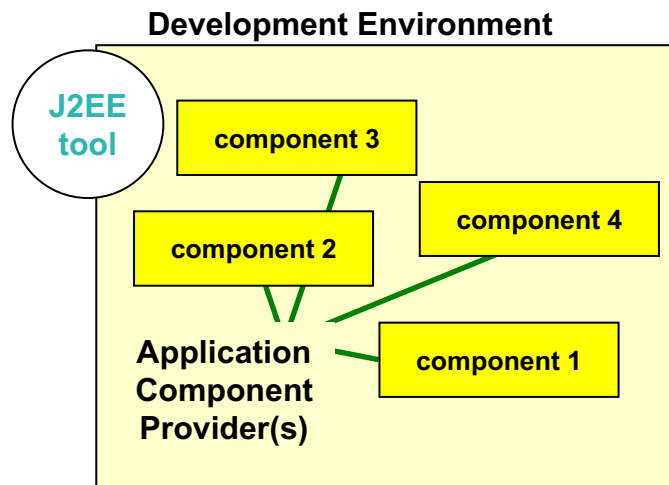
**Additional information —**

**Transition statement —** Next: J2EE platform roles (1 of 4)

## J2EE platform roles (1 of 4)

### –Application component provider

- Produces application building blocks (HTML documents, EJBs, JSPs, and so forth).
- In many cases this role will be further subdivided.



© Copyright IBM Corporation 2007

Figure 7-8. J2EE platform roles (1 of 4)

WA3616.3

### Notes:

J2EE also specifies:

- J2EE product provider
  - Implements a J2EE product that provides containers, APIs and other features
- Tool provider
  - Provides application component development and packaging tool



***Instructor notes:***

**Purpose —**

**Details —**

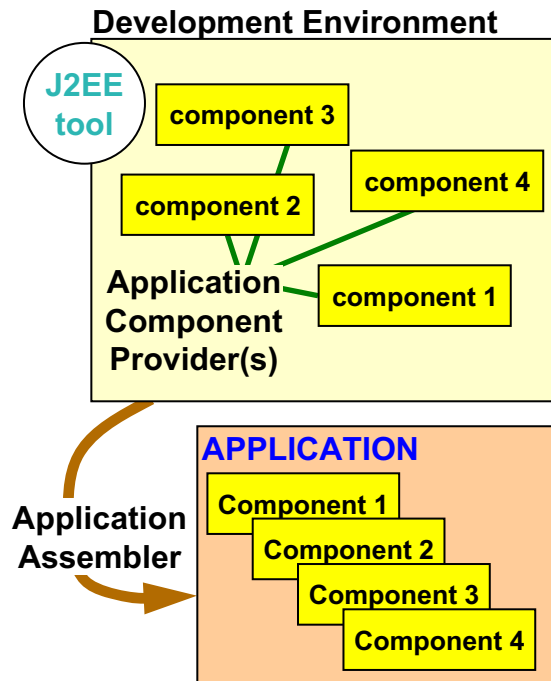
**Additional information —**

**Transition statement —** Next: J2EE platform roles (2 of 4)

## J2EE platform roles (2 of 4)

### –Application assembler

- Takes components developed by component providers and assembles them into a complete J2EE application



© Copyright IBM Corporation 2007

Figure 7-9. J2EE platform roles (2 of 4)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

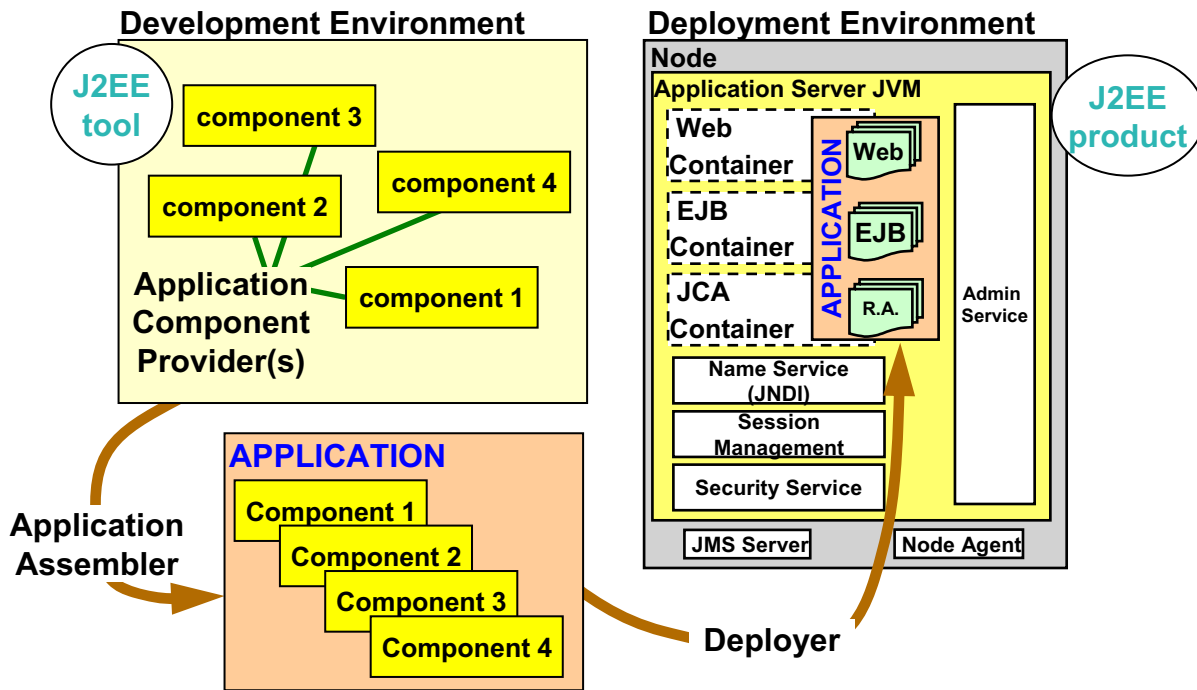
**Additional information —**

**Transition statement —** Next: J2EE platform roles (3 of 4)

## J2EE platform roles (3 of 4)

-Deployer

- Deploys, configures and runs EJBs and web applications



© Copyright IBM Corporation 2007

Figure 7-10. J2EE platform roles (3 of 4)

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: J2EE platform roles (4 of 4)

## J2EE platform roles (4 of 4)

–System administrator

- Configures and administers the infrastructure

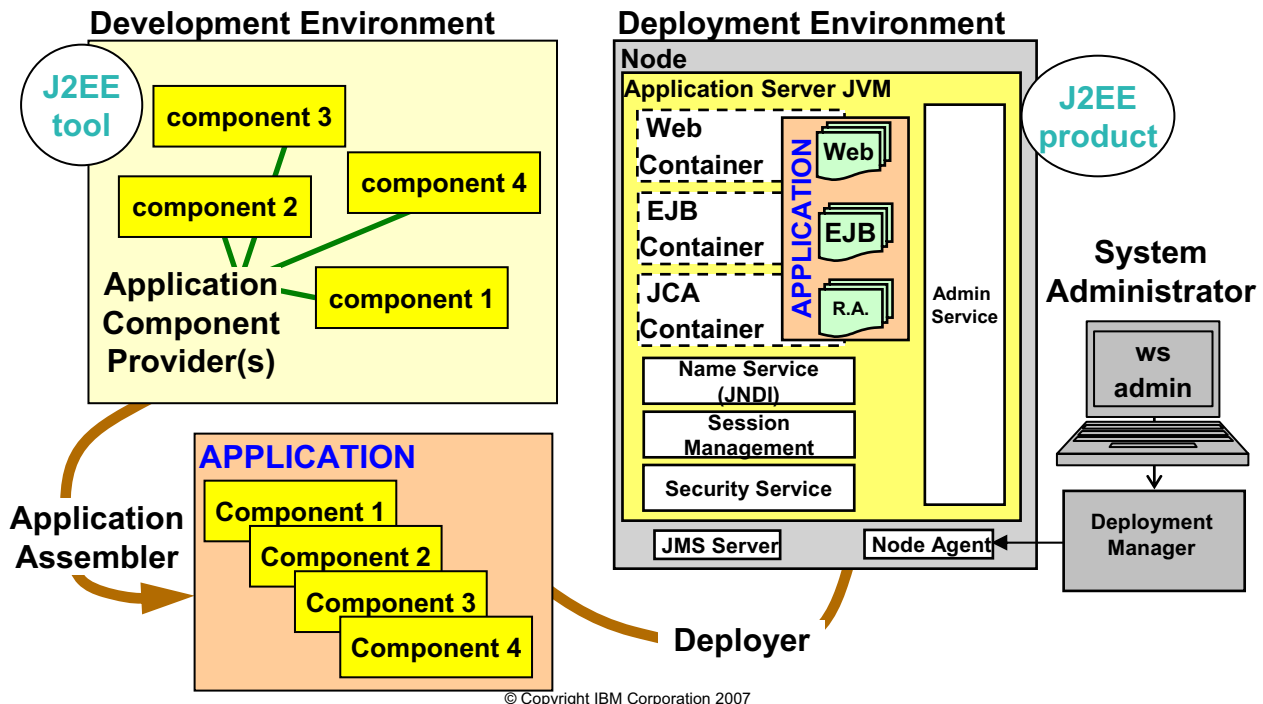


Figure 7-11. J2EE platform roles (4 of 4)

WA3616.3

**Notes:**

***Instructor notes:***

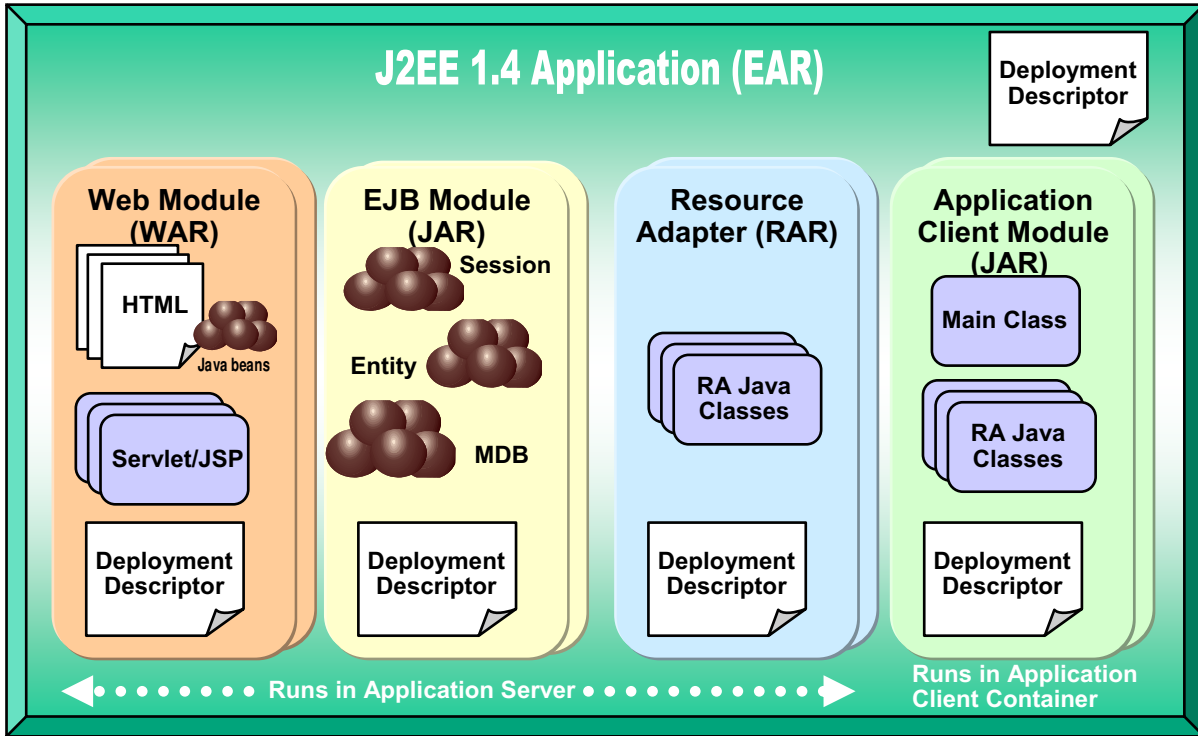
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: J2EE 1.4 application components

# J2EE 1.4 application components



© Copyright IBM Corporation 2007

Figure 7-12. J2EE 1.4 application components

WA3616.3

## Notes:

J2EE applications are packaged as Enterprise Archive files that are typically referred to as EAR files (these files have a .ear extension). Each ear file contains a deployment descriptor along with one or more J2EE modules. Each module groups one or more application components that are of the same type, along with a deployment descriptor for that module.

For example, a Web module is packaged in a Web Archive file (WAR file), and contains Web components such as JSP and Servlets along with a deployment descriptor for that module.



***Instructor notes:***

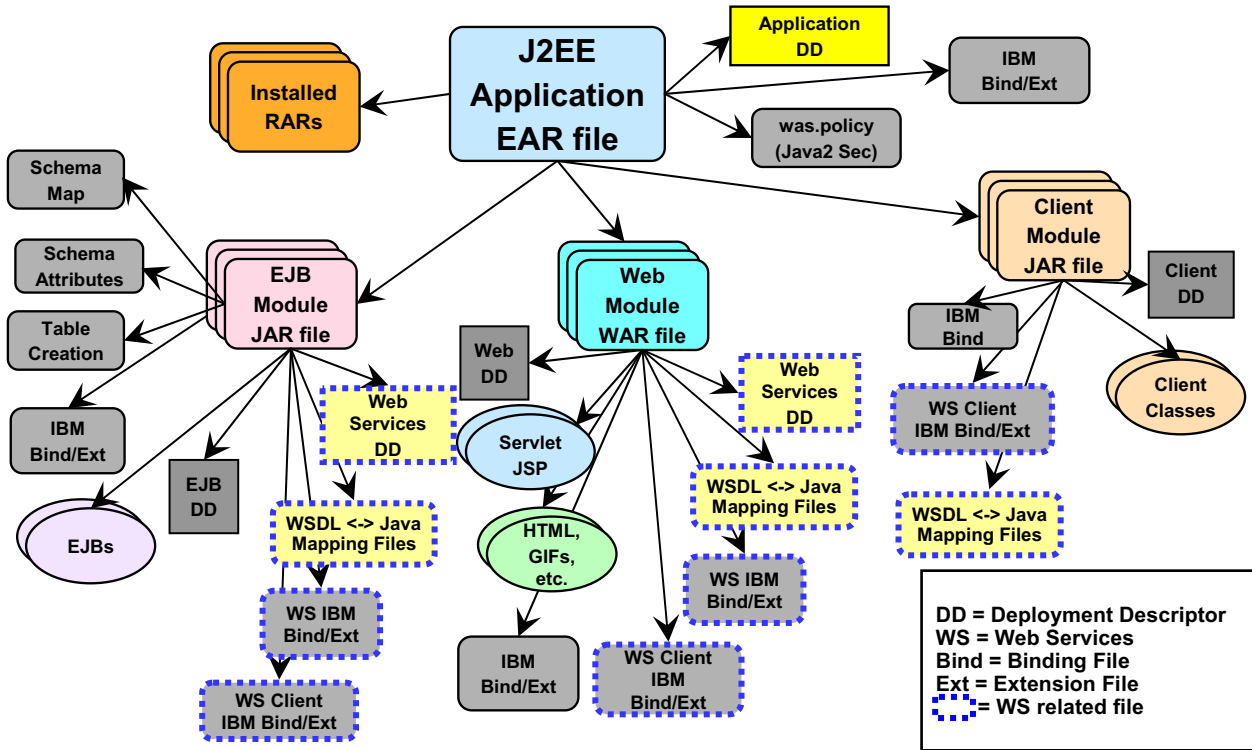
**Purpose** —

**Details** — You might wish to spend a little time here pointing out the major J2EE concepts that students will see within this class: JavaBeans, Servlets, JSPs, EJBs (session and entity).

**Additional information** —

**Transition statement** — Next: J2EE packaging files

# J2EE packaging files



© Copyright IBM Corporation 2007

Figure 7-13. J2EE packaging files

WA3616.3

## Notes:

***Instructor notes:***

**Purpose** —

**Details** — Point out the major “divisions” within the EAR file: EJB jar, WAR, and Client Module jar.

**Additional information** —

**Transition statement** — Next: Checkpoint questions

## Checkpoint questions

---

1. Which of the following J2EE application modules could contain Message Driven beans?
  - A. Client module
  - B. Web module
  - C. EJB module
  
2. The J2EE platform role of Application Assembler is assigned to a person who does what?
  - A. Installs and runs EJBs and Web applications
  - B. Configures and administers the infrastructure
  - C. Configures components into a complete J2EE application
  
3. Which of the following are possible components of Web modules?
  - A. Servlets, JSPs, HTMLs
  - B. Session beans, entity beans, MDBs
  - C. Servlets, JAR files, RAR files

© Copyright IBM Corporation 2007

Figure 7-14. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.

**Instructor notes:**

Purpose —

Details —

## Checkpoint solutions

---

1. Which of the following J2EE application modules could contain Message Driven beans?  
C. EJB module
2. The J2EE platform role of Application Assembler is assigned to a person who does what?  
C. Configures components into a complete J2EE application
3. Which of the following are possible components of Web modules?  
A. Servlets, JSPs, HTMLs

© Copyright IBM Corporation 2007

**Additional information** —**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe Java basics
- Describe basic J2EE terminology
- Describe J2EE component technologies
- Describe J2EE services
- Describe J2EE messaging and communication
- Describe J2EE packaging and how it is implemented in WebSphere Application Server
- Describe how J2EE implements security
- Describe the roles defined by J2EE and the tools associated with those roles
- Describe the new enhanced EAR functionality, its implications, and how it complements the J2EE EAR file

© Copyright IBM Corporation 2007

Figure 7-15. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**





---

## Unit 8. Application assembly

### Estimated time

01:00

### What this unit is about

This unit describes the application assembly process with a particular focus on the use of the Application Assembly Tool (AST).

### What you should be able to do

After completing this unit, you should be able to:

- Describe the application assembly process
- Describe the use of the Application Server Toolkit (AST)
- Describe how AST relates to IBM Rational Web/Application Developer
- Use AST to define an enterprise application
- Use AST to define an EJB module
- Use AST to define a Web module
- Use AST to define an application client module
- Use AST to define application scope resources
- Export an enhanced EAR file

### How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

### References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

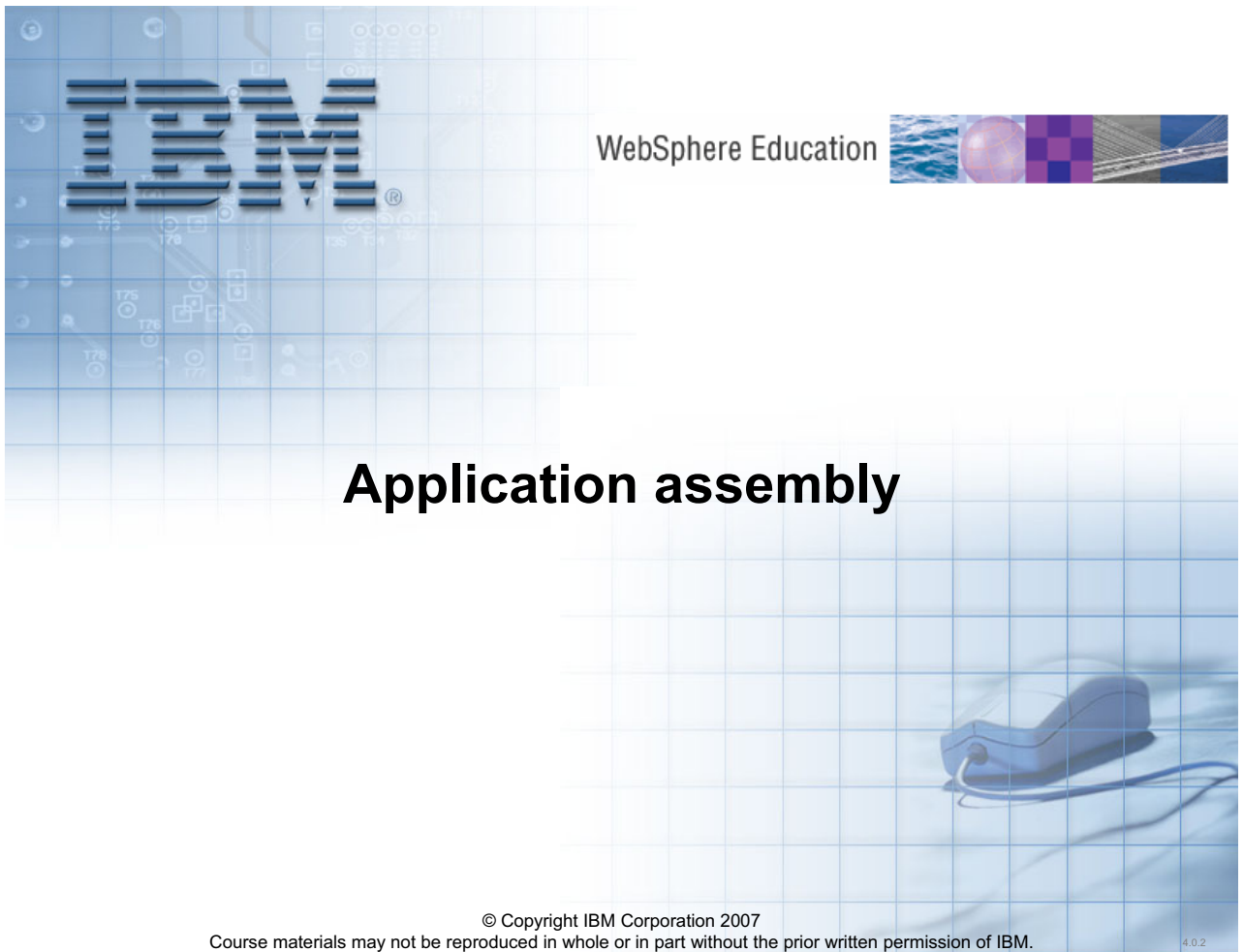


Figure 8-1. Application assembly

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe the application assembly process
- Describe the use of the Application Server Toolkit (AST)
- Describe how AST relates to IBM Rational Web/Application Developer
- Use AST to define an enterprise application
- Use AST to define an EJB module
- Use AST to define a Web module
- Use AST to define an application client module
- Use AST to define application scope resources
- Export an enhanced EAR file

© Copyright IBM Corporation 2007

Figure 8-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

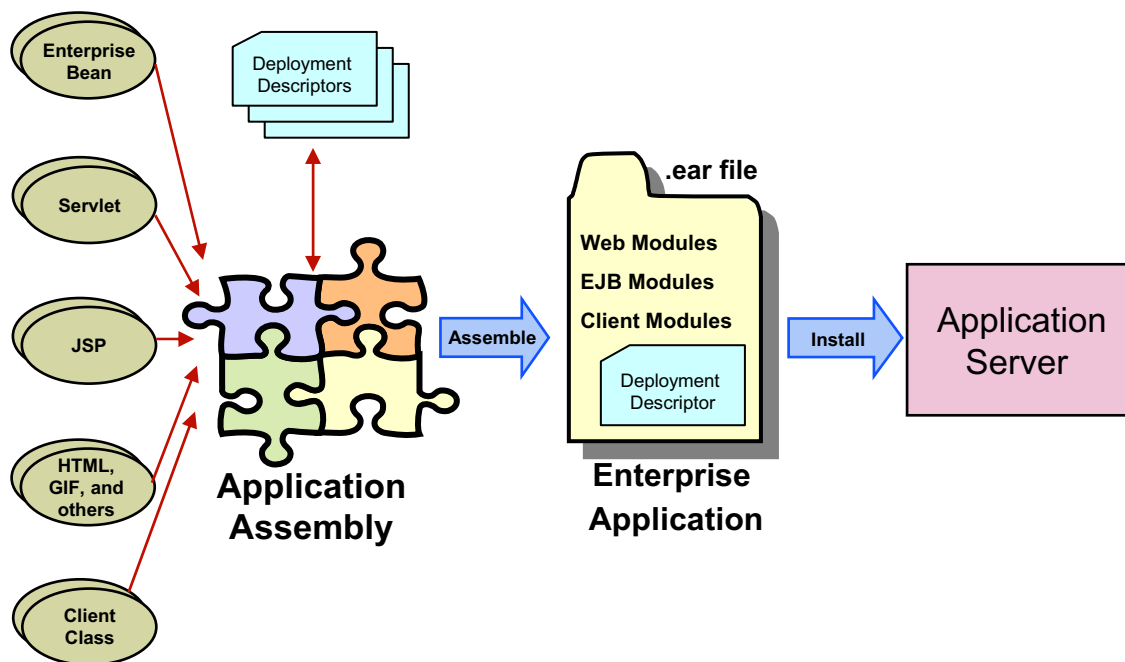
**Purpose** — The purpose of this visual is to present unit objectives.

**Details** —

**Additional information** —

**Transition statement** — Though it is the role of the application assembler to give the administrator a completely ready to install application EAR file the administrator will still need knowledge of the AST tool for problem determination and in depth application knowledge. Look at the big picture first.

## Overview of application assembly and installation



© Copyright IBM Corporation 2007

Figure 8-3. Overview of application assembly and installation

WA3616.3

### Notes:

Application assembly is a process of building and configuring an enterprise application archive (EAR) file for deployment onto a J2EE application server.

***Instructor notes:***

**Purpose** — The purpose of this visual is to present an overview of the total deployment process.

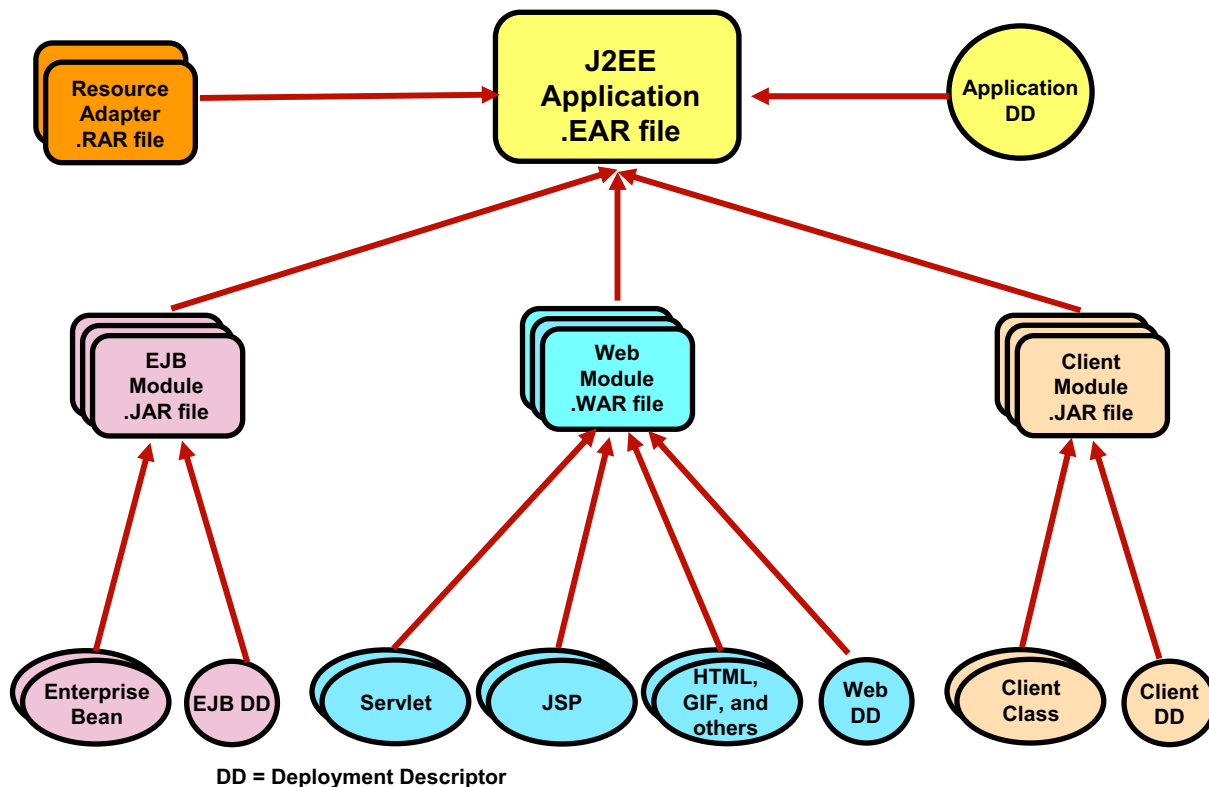
**Details** —

**Additional information** —

**Transition statement** — One of the potential jobs of the administrator is to install applications. The file that the administrator installs is an Enterprise Archive file usually referred to as an EAR file. The EAR file contains several components.



## J2EE packaging



© Copyright IBM Corporation 2007

Figure 8-4. J2EE packaging

WA3616.3

### Notes:

A J2EE application is packaged in an Enterprise Archive, a file with a .EAR extension. The application has a deployment descriptor, shown here as DD, allowing configuration to a specific container's environment when deployed. The application can include one or more modules.

J2EE components are grouped in modules, and each module has its own deployment descriptor.

EJB modules group related EJBs in a single module, and are packaged in Java Archive (JAR) files. Note that there is only one deployment descriptor for all of the EJBs in the module.

Web modules group servlet class files, JSPs, HTML files and images. They are packaged in Web Application Archive (WAR) files.

Application client modules are packaged in Java Archive (JAR) files.

Resource Adapters may be packaged to the application server or within an application .EAR file.

***Instructor notes:***

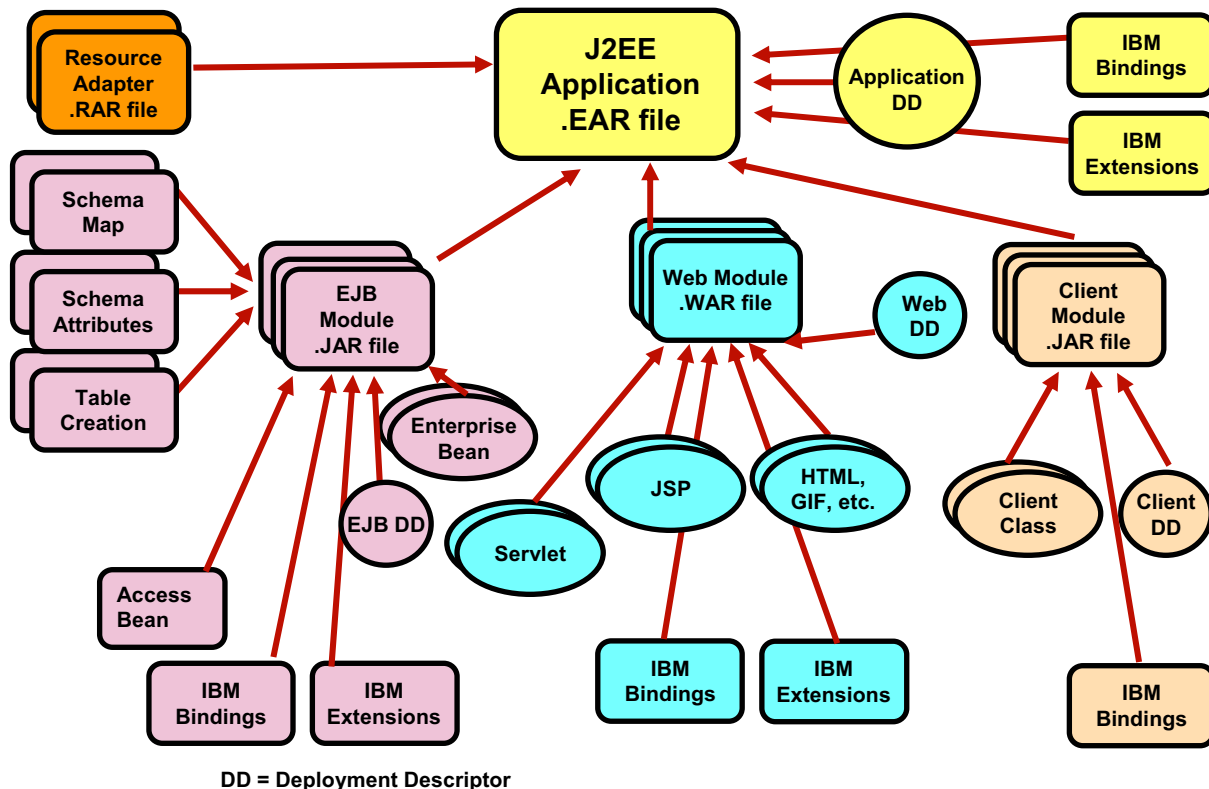
**Purpose** — The purpose of this visual is to introduce the contents of the EAR file and to present the Deployment descriptors.

**Details** —

**Additional information** —

**Transition statement** — There are actually more components in the EAR file.

# WebSphere: Application packaging



© Copyright IBM Corporation 2007

Figure 8-5. WebSphere: Application packaging

WA3616.3

## Notes:

In WebSphere Application Server, the J2EE Application EAR file is enhanced with IBM Bindings and Extensions. These adapt the generic J2EE application to the IBM WebSphere Application Server environment. Most J2EE server vendors have their own proprietary extensions to the specification.

The schema map, attributes, and table creation code provide the setup for entity Enterprise beans to store their values in a specific data store. There can be a set of these files for each data store the entity beans are mapped to. These are for entity beans that use Container Managed Persistence, or CMP. These are IBM specific.

All of these objects in the Application EAR file are packaged by the Application Server Toolkit.

***Instructor notes:***

**Purpose** — The purpose of this visual is to introduce the extensions to the EAR file.

**Details** — Keep the discussion here at a high level. The presentation will be going into the detail in following visuals.

**Additional information** —

**Transition statement** — Besides the actual application code the EAR contains other information also.

## WebSphere application contents

---

- Deployment Descriptors - J2EE standard
  - XML-based text files describing the application environment
  
- IBM Bindings
  - Bind application names to deployment platform specific resources
  
- IBM Extensions
  - Supports additional options, beyond J2EE specs, such as:
    - Access intent attributes
    - Web application reloading
    - File serving and servlet invoker (by class name)
  
- Database-related files (per mapped data store)
  - Map.mapxmi - Maps the EJB fields to the schema fields
  - xxx.schxmi - Specifies the database schema field attributes
  - Table.ddl - Commands to create a database table

© Copyright IBM Corporation 2007

Figure 8-6. WebSphere application contents

WA3616.3

### Notes:

WebSphere applications contain J2EE standard deployment descriptors, which are XML files that describe the application environment. Features that can be configured at deployment time include security checks and transaction control.

WebSphere applications contain additional information beyond the J2EE requirements.

IBM Bindings associate local names to deployment platform specific resources, for example security roles mapped to specific authentication credentials.

IBM Extensions support additional options such as access intent (caching and transaction isolation) and Web application reloading.

You can specify whether to use WebSphere internal Servlets like the File Serving Enabler or the Invoker, which serves up servlets by class name.

For EJB modules, a set of files map Entity EJBs to a specific data store “schema”. There are a set of files for each back-end system being mapped to.

***Instructor notes:***

**Purpose** — The purpose of this visual is to further define the format of deployment descriptors and introduced the other extensions.

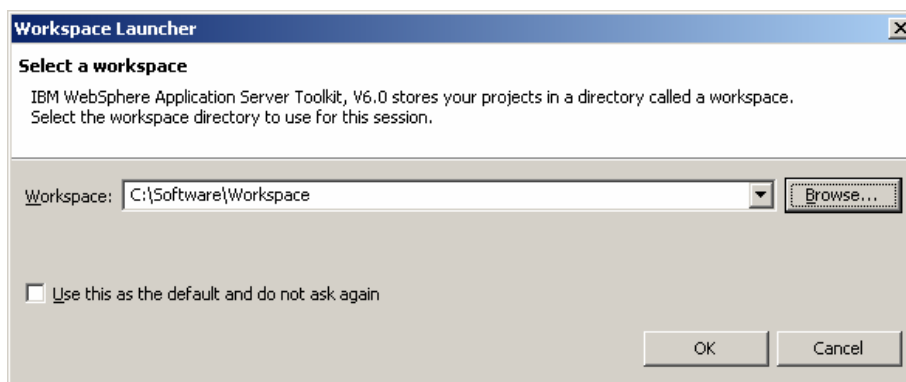
**Details** —

**Additional information** —

**Transition statement** — Look at the IBM Binding extension to the J2EE specifications a little closer.

## Application Server Toolkit (AST)

- Tool to perform assembly and configuration of Enterprise Applications
  - Based on Eclipse tooling platform
  - Similar look and feel as IBM Rational family of products
  - Complete development environment for Enterprise applications, including EJBs
  - Source level debugging and profiling applications
  - Jython editor and source debugger. Run wsadmin scripts inside AST
- Opening AST requires pointing to a workspace folder
  - Files and metadata are kept in workspace
  - Create a new workspace by pointing to an empty folder
  - Thereafter open workspace folder
  - Guideline: One workspace per Enterprise Application



© Copyright IBM Corporation 2007

Figure 8-7. Application Server Toolkit (AST)

WA3616.3

### Notes:

The IBM WebSphere Application Server Toolkit V6 (AST) is only available in the Windows and LINUX operating systems.

When starting the AST you can either select an empty workspace folder, or an existing workspace folder. Use the empty workspace folder if you will be creating an ear file from its individual components (WARs, JARs, and so forth). After importing the components into the new workspace, in subsequent sessions, open the existing workspace folder. Even if no modules are added to an empty workspace, AST will generate a “.metadata” folder where it will keep its own internal data representing the workspace. As modules are imported/added to the workspace other folders are created to hold their contents.

***Instructor notes:***

**Purpose** — The purpose of this visual is to introduce AST.

**Details** —

**Additional information** —

**Transition statement** — Though there are other tools to do the same function, for example the IBM Rational developer tools, one of the major tools administrators would use is the AST. It can do several things.



## Application Server Toolkit functionality

---

- Things you can do with the AST
  - Create/configure J2EE Enterprise Applications (EARs):
    - Build from scratch
    - J2EE modules
  - Modify deployment descriptor information
  - Modify binding information attributes
  - Modify the IBM extension attributes
  - Deploy applications to a remote server
  - Perform remote debugging and profiling sessions
    - IBM Agent Controller needs to be installed for remote profiling
  - Create, debug and run Jython scripts
    - Jython editor with command completion and color syntax
    - Ability to run and debug wsadmin scripts from AST
  - Import command assistance logs from console into Jython scripts
  - View, analyze, and correlate log files

© Copyright IBM Corporation 2007

Figure 8-8. Application Server Toolkit functionality

WA3616.3

### **Notes:**

The AST should mainly be used to assemble and configure J2EE Enterprise Applications. After these steps an EAR file can be exported out of the tool for deployment to an Enterprise Application Server, such as IBM WebSphere Application Server.

The AST can also be used to debug and profile applications. These applications need to be deployed to a remote server before they can be debugged or profiled.

Remote profiling requires that the companion product, the IBM Agent Controller, also be installed on the machine where the server resides.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Assembling an enterprise application

## Assembling an enterprise application

---

- When working with a workspace handed over by development, no assembly is required (already done automatically by tool).
  - [Configure modules](#)
  - [Export EAR file](#)
- Project Interchange files can be imported into a new or existing workspace.
- When assembling individual modules:
  - [Create new workspace \(first time\)](#)
  - [Import modules into workspace](#)
  - [Assign modules to Enterprise application](#)
  - [Configure module's deployment descriptors](#)
  - [Export EAR file](#)
- After assembly:
  - [Optional test within tool](#)
  - [Deploy to local or remote server](#)
  - [Debug code](#)

© Copyright IBM Corporation 2007

Figure 8-9. Assembling an enterprise application

WA3616.3

### Notes:

If your developers use IBM tools you may receive an existing, working workspace folder for final configuration and deployment. In this case the individual WAR and JAR files are not required as they already exist as part of the workspace. When working with workspaces all you need to do when starting AST, is point to the root directory of the workspace.

If you receive the individual WAR and JAR files, which are the modules for the application, you will need to point AST to an empty workspace which will hold the Enterprise application's workspace. You only do this the first time, thereafter you just point AST to this newly created workspace directory.

In this last scenario, assembly is just the action of importing the files containing the modules and associating them with the Enterprise Application. The end result is an EAR file, which contains all the modules and their deployment descriptors. The EAR file can then be installed (or deployed) to an application server.

***Instructor notes:***

**Purpose —**

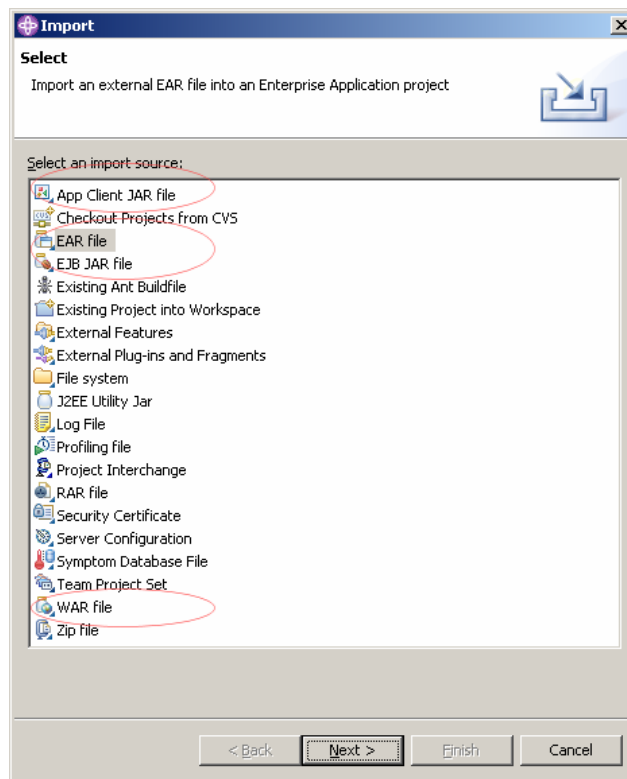
**Details —**

**Additional information —**

**Transition statement —** Next: Import modules

## Import modules

- Wizard driven
  - Import modules
    - EAR files
    - EJB JAR
    - Application Client JAR
    - Web module WAR
  - Imported into new or existing Enterprise Application



© Copyright IBM Corporation 2007

Figure 8-10. Import modules

WA3616.3

### Notes:

The **Import** wizard is used to import all types of files into the workspace. The circled files are the types of files which make up the modules of an Enterprise Application.

The Project Interchange file format can also be used to share entire or partial projects. These are special zip files, which when imported into an empty workspace folder, will recreate the tool metadata for the machine it is being imported into. This is an ideal format to share workspaces because no workspace metadata is stored within it, it is rather recreated when the file is imported.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: J2EE Perspective

# J2EE Perspective

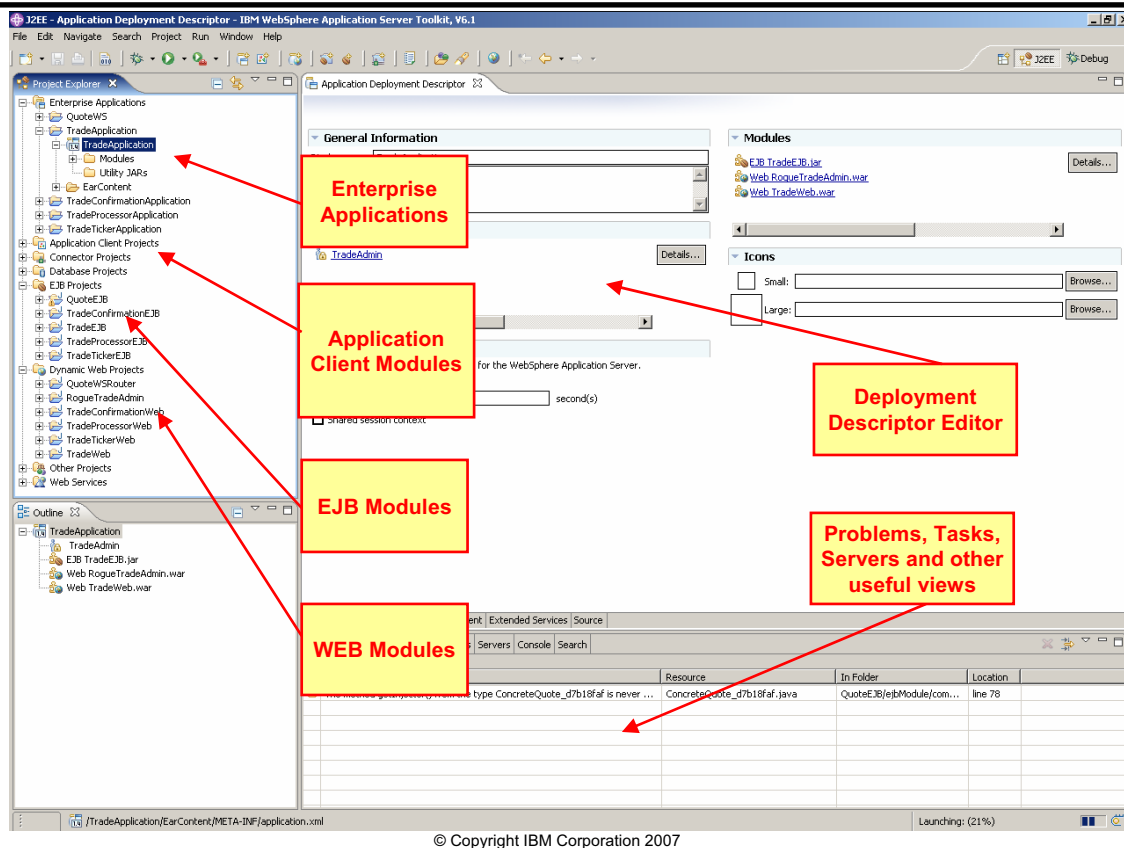


Figure 8-11. J2EE Perspective

WA3616.3

## Notes:

AST is organized into perspectives, panes, views and editors.

Perspectives show a predetermined number of views to facilitate a role, like deployment. Perspectives can be customized by the user and saved. Other useful perspectives are: Server, Debug, Java, Profiler.

The J2EE perspective is used during the Assembly/Configure/Deploy process.

Panes hold views. Views can be stacked in panes. If stacked, views are accessible through tabs on the pane.

Editors present data in a way that makes viewing and changing that data the easiest, based on the type of data. There are specialized editors for each of the deployment descriptor types, for each of the file types that make up J2EE applications (Java, XML, and so forth).

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Application deployment descriptor (1 of 2)



## Application deployment descriptor (1 of 2)

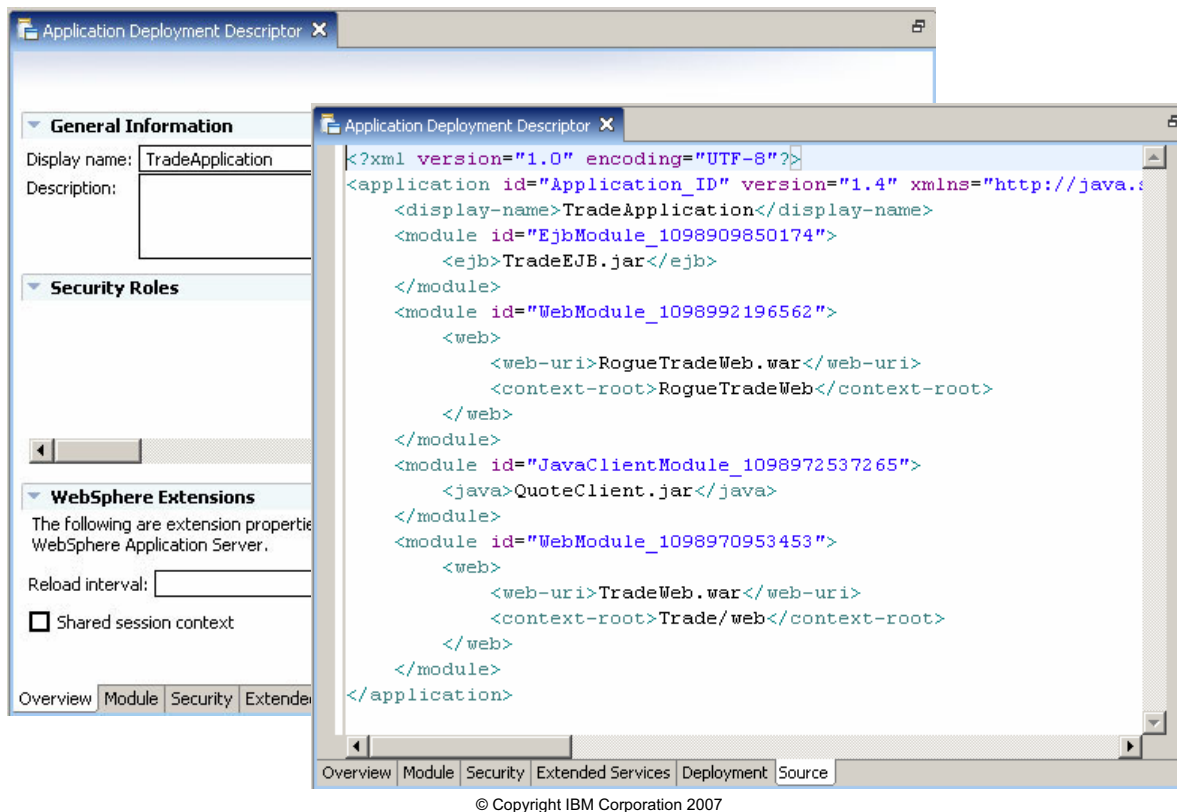


Figure 8-12. Application deployment descriptor (1 of 2)

WA3616.3

### Notes:

The Application's deployment descriptor is shown in the customized editor and in the source view of the editor.

Recommendation: Use customized editors when working with deployment descriptors, they will produce the correct XML without the risk of manual editing.

Use the editor's tabs to access different sections of the file.

The application.xml file defines all modules in the EAR file:

EJB jars

WARs

Resource Adapters

Application Client jars

Dependent Files

Bind Application Security Roles

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Application deployment descriptor (2 of 2)

---

## Application deployment descriptor (2 of 2)

---

- Simplest of deployment descriptors
  
- Editor can be used to:
  - Edit application's display name and description
  - Add and remove modules
    - Web
    - EJB
    - Application client
    - Resource adapter (connector)
  - Inspect URI and context root of modules
  - Work with security roles of the application
  - Define application scope resources (enhanced EAR files)

© Copyright IBM Corporation 2007

Figure 8-13. Application deployment descriptor (2 of 2)

WA3616.3

### **Notes:**

The customized editors in AST are specially designed to enable quick and easy access to all parts of the file being edited. The views (or pages) of the editor organize the data into logical groupings. The deployment descriptor editors produce the XML files required, ensuring that they are valid and conforming.

Note that the Context root of Web modules can not be edited directly here but is shown for completeness. Refer to the Web Modules slides to find out how to change the Context root after a Web module as been imported into the workspace.

***Instructor notes:***

**Purpose —**

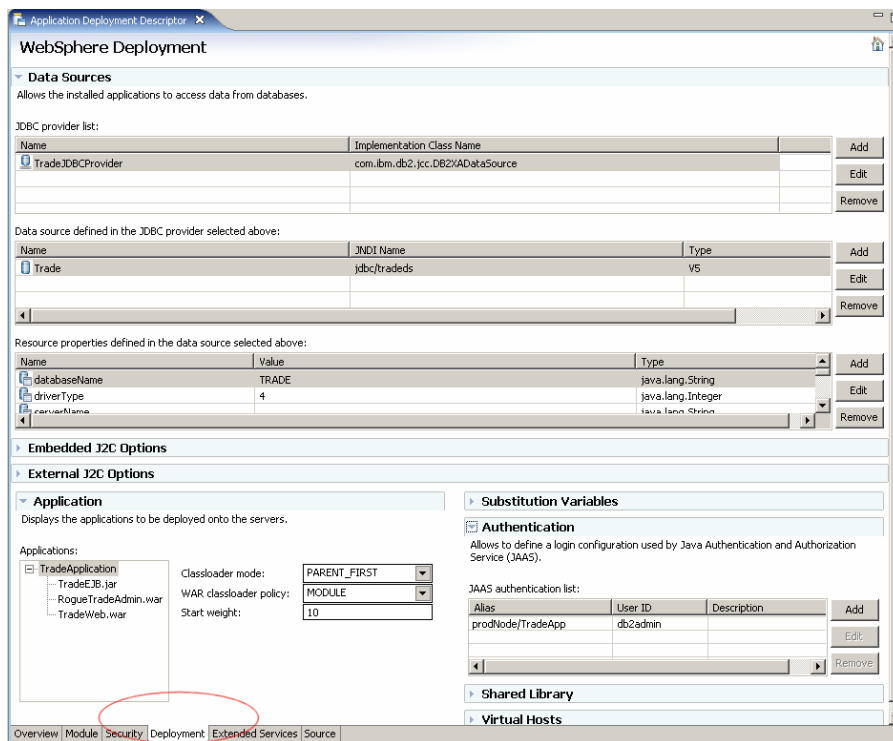
**Details —**

**Additional information —**

**Transition statement —** Next: Application scope resources

## Application scope resources

- Define resources to be included in Enhanced EAR file



© Copyright IBM Corporation 2007

Figure 8-14. Application scope resources

WA3616.3

### Notes:

On the Deployment tab of the deployment descriptor editor for the application you can define certain resources that are included within the EAR file.

At this time Data Sources, JAAS Authentication Aliases, Classloader policy, Shared Libraries and Virtual host information can be defined on this page.

Any resources on this page are defined at the new Application scope. Any changes made in the administrative console will not be saved to the EAR file. These changes are only saved to the runtime configuration file.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Enhanced EAR

## Enhanced EAR

---

- Enhancements:
  - Will not be recognized by application servers prior to WebSphere Application Server V6
  - Not part of the J2EE specification
- Includes resources for the enterprise applications
  - Facilitates installation
    - In many cases all configuration artifacts can be included
      - Data sources
      - JAAS authentication aliases
      - Environment variables
      - Other artifacts
  - Used mainly during development and test stages
  - Not intended to be used in production
- Resources are defined at Application scope
  - Are visible in the administrative console
    - Under Application's resources, not with other scoped resources
    - Can be tested in the console (for example, test connection for data sources)
  - Take precedence over all other scopes

© Copyright IBM Corporation 2007

Figure 8-15. Enhanced EAR

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Un-enhancing an EAR file



## Un-enhancing an EAR file

---

- Before deployment to production, resources defined in enhanced EAR file should be removed.
- Resources should be re-configured at the proper scope.
  - Preferably through scripting
- Resources can be removed:
  - Using AST (recommended)
  - Other ZIP file manipulating tool to remove the META-INF folder
  - In UNIX systems use the EARExpander script:
    - Expand the ear file
    - Delete contents of the META-INF folder
    - Compress the ear file
    - Deploy

© Copyright IBM Corporation 2007

Figure 8-16. Un-enhancing an EAR file

WA3616.3

### Notes:

When un-enhancing an EAR file, make sure you do not open the Deployment page of the application's deployment descriptor. Doing so will automatically create an ibmconfig folder in the EAR file, even if nothing is ever put in there. The application server will think that the EAR file is enhanced.

To un-enhance an EAR file, remove the META-INF/ibmconfig folder and the contents of the META-INF/ibmconfig folder. Be careful to not remove the META-INF root folder. Only remove META-INF/ibmconfig. This process can easily be done using the AST.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Dealing with enhancements at deploy time

## Dealing with enhancements at deploy time

- To ignore application scoped resources at install time

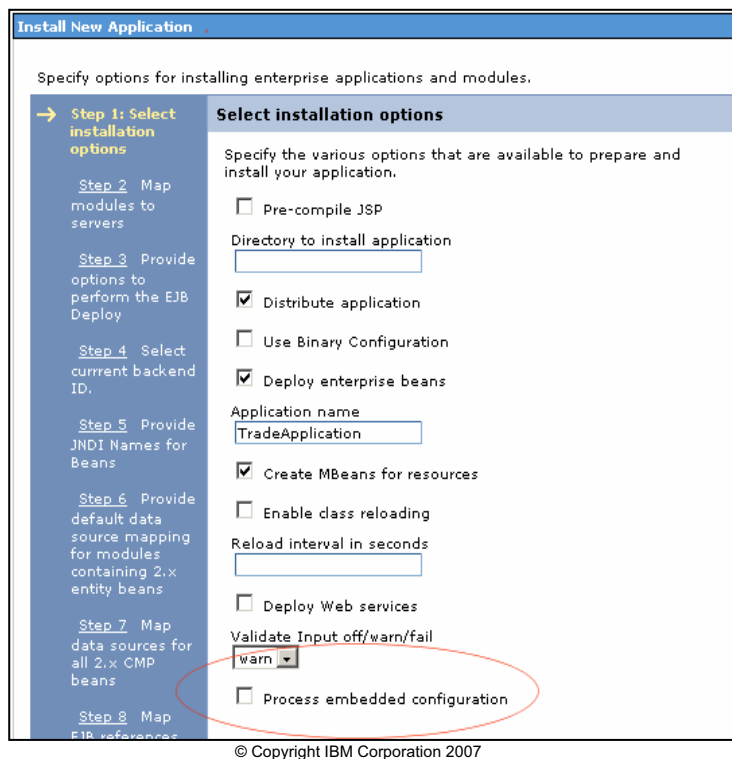


Figure 8-17. Dealing with enhancements at deploy time

WA3616.3

### Notes:

Process embedded configuration specifies whether the embedded configuration should be processed. An embedded configuration consists of files such as resource.xml and variables.xml. When selected or true, the embedded configuration is loaded to the application scope from the .ear file. If the .ear file does not contain an embedded configuration, the default is false. If the .ear file contains an embedded configuration, the default is true.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: EJB module assembly

---

## EJB module assembly

---

- Import wizard
  - Imports into new or existing enterprise application project
  - Adds Enterprise JavaBeans in module
    - Session bean
    - BMP entity
    - CMP entity
    - MDB
  
- Configure using deployment descriptor editor
  - EJB references
  - EJB relationships
  - EJBQL queries
  - Security roles
  - Method permissions
  - Access intent for entity beans

© Copyright IBM Corporation 2007

Figure 8-18. EJB module assembly

WA3616.3

### **Notes:**

EJB deployment descriptors can become quite complex. The custom EJB deployment descriptor editor gives you access to all the information in a clear and well organized set of forms.

All aspects of EJB configuration, for all three types of EJBs are available.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: EJB module expanded

## EJB module expanded

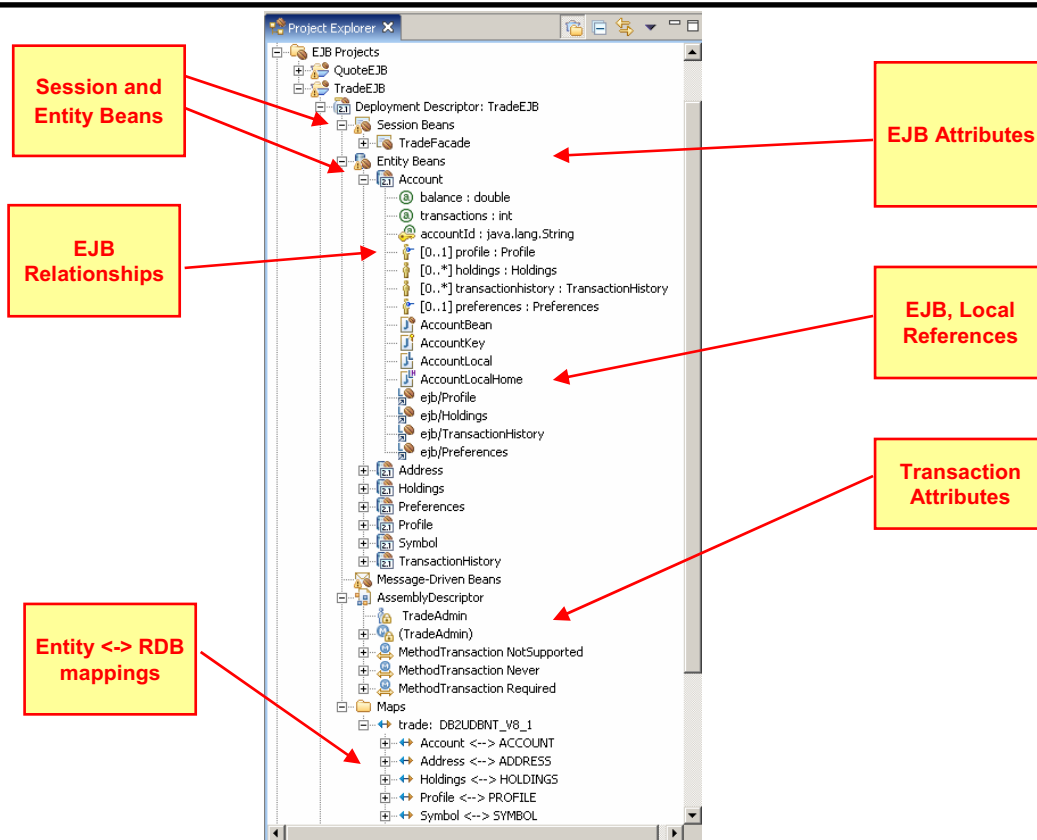


Figure 8-19. EJB module expanded

WA3616.3

### Notes:

The J2EE Hierarchy view can be expanded to see the overview of the EJB components of a module.

Double clicking any entry on the expanded view of an EJB module causes the deployment descriptor of the EJB project to open in the editor at the right place to view or modify the value.

When using J2EE security, roles and method permissions can also be seen on this view.

***Instructor notes:***

**Purpose** —

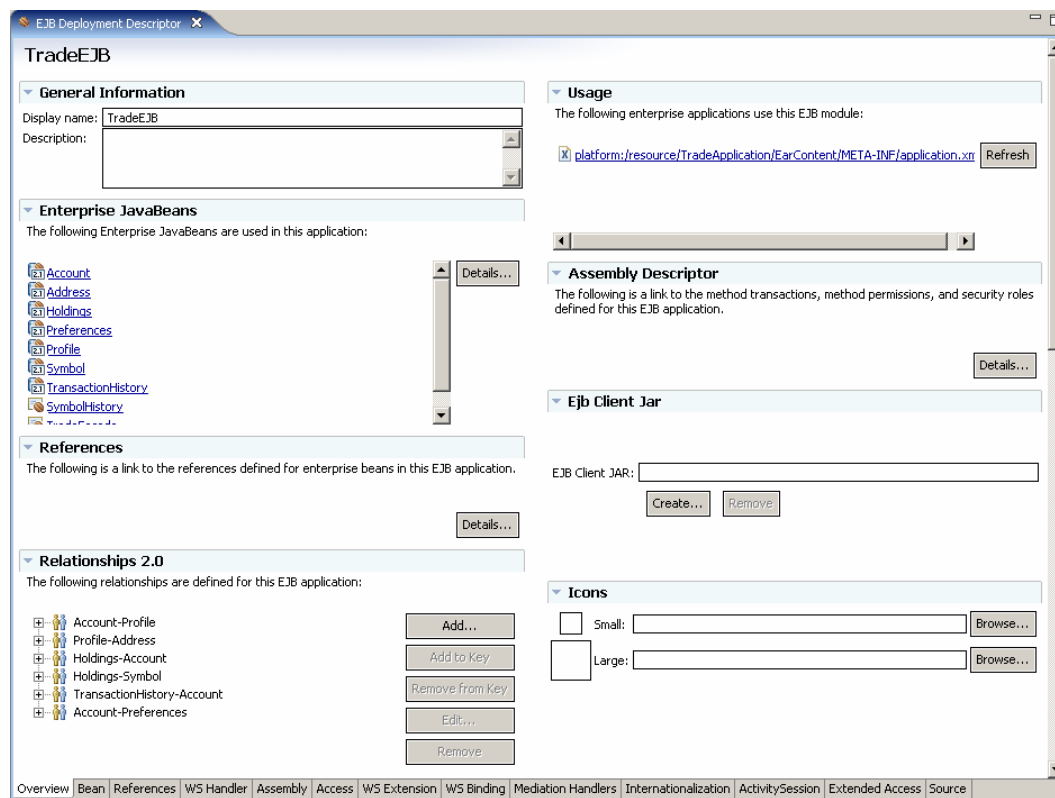
**Details** —

**Additional information** —

**Transition statement** — Next: EJB deployment descriptor editor



# EJB deployment descriptor editor



© Copyright IBM Corporation 2007

Figure 8-20. EJB deployment descriptor editor

WA3616.3

## Notes:

All parts of the deployment descriptor can be accessed through the editor. The Overview view can be used as the gateway to more detailed views. The editor tabs can also be used to reach a specific area of the deployment descriptor. As usual you can always see the XML source by clicking the Source tab. Resist the temptation to edit the deployment descriptor in the source view. Using the other editor view ensures that properly formatted, valid XML will be generated.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Web module assembly

## Web module assembly

---

- Import wizard
  - Imports into existing Enterprise Application project, or creates new
  - Adds Web components from WAR module
    - Servlets
    - JSPs
    - Static HTML
    - Supporting Java code
  
- Configure using deployment descriptor editor
  - Named servlets and JSPs (URL mappings)
  - Initialization parameters
  - Resource references
  - Security roles and constraints
  - Authentication mechanism
  - Welcome and error pages
  - Filters and listeners
  
- Context root defined under Properties -> J2EE, not the module's deployment descriptor

© Copyright IBM Corporation 2007

Figure 8-21. Web module assembly

WA3616.3

### Notes:

Web modules can be imported into new or existing Web projects. The Web projects can be part of new or existing Enterprise Applications.

When importing a WAR file containing a Web module into a Web project, you are given a chance to change the existing Context root for the module. After importing the module, you cannot change the Context root using the deployment descriptor editor. Context roots are kept in the Application's deployment descriptor. To change the Context root of a Web project, open its properties and display the Web page, here you can define it.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Web module

# Web module

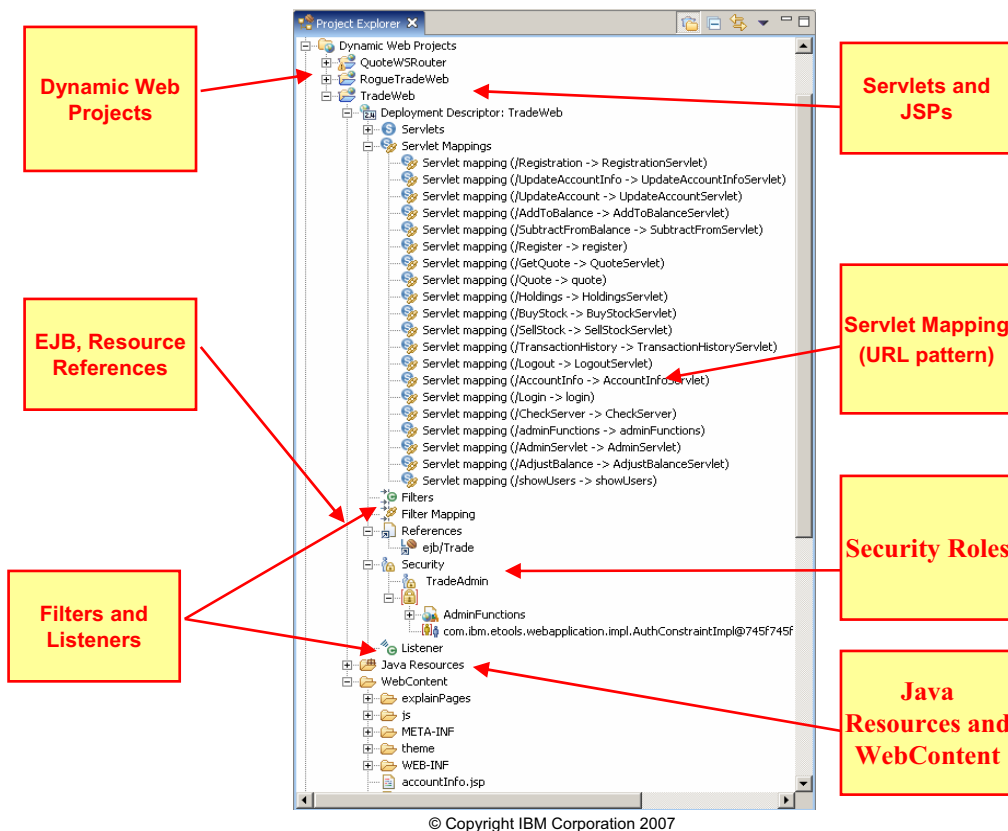


Figure 8-22. Web module

WA3616.3

## Notes:

In the J2EE Hierarchy view you can expand the Web module to see all the major components it contains. You can double-click any of these components and that action opens the Web deployment descriptor in the custom editor.

***Instructor notes:***

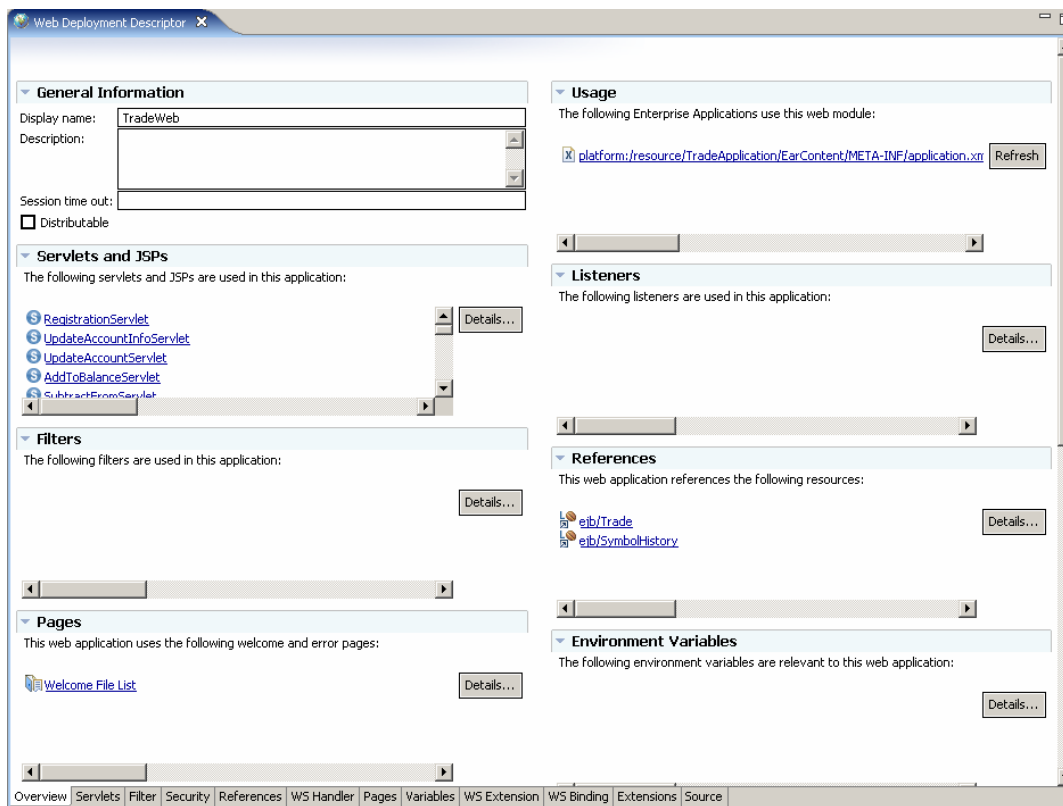
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Web deployment descriptor editor

# Web deployment descriptor editor



© Copyright IBM Corporation 2007

Figure 8-23. Web deployment descriptor editor

WA3616.3

## Notes:

All parts of the deployment descriptor can be accessed through the editor. The Overview view can be used as the gateway to more detailed views. The editor tabs can also be used to reach a specific area of the deployment descriptor. As usual you can always see the XML source by clicking the Source tab. Resist the temptation to edit the deployment descriptor in the source view. Using the other editor view ensures that properly formatted, valid XML will be generated.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Application client assembly



## Application client assembly

---

- Import wizard
  - Imports into existing Enterprise Application project, or creates new
  - Adds Application client components from JAR module
  - Specify the class path and the class containing the main() method used by the application client
  
- Add any icons
  
- Add EJB references
  
- Add resources references

© Copyright IBM Corporation 2007

Figure 8-24. Application client assembly

WA3616.3

### **Notes:**

Importing a J2EE application client module is very similar to importing the other types of modules. You pick the source file and the destination Enterprise Application and client project. The wizard does the rest.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Application Client module

# Application Client module

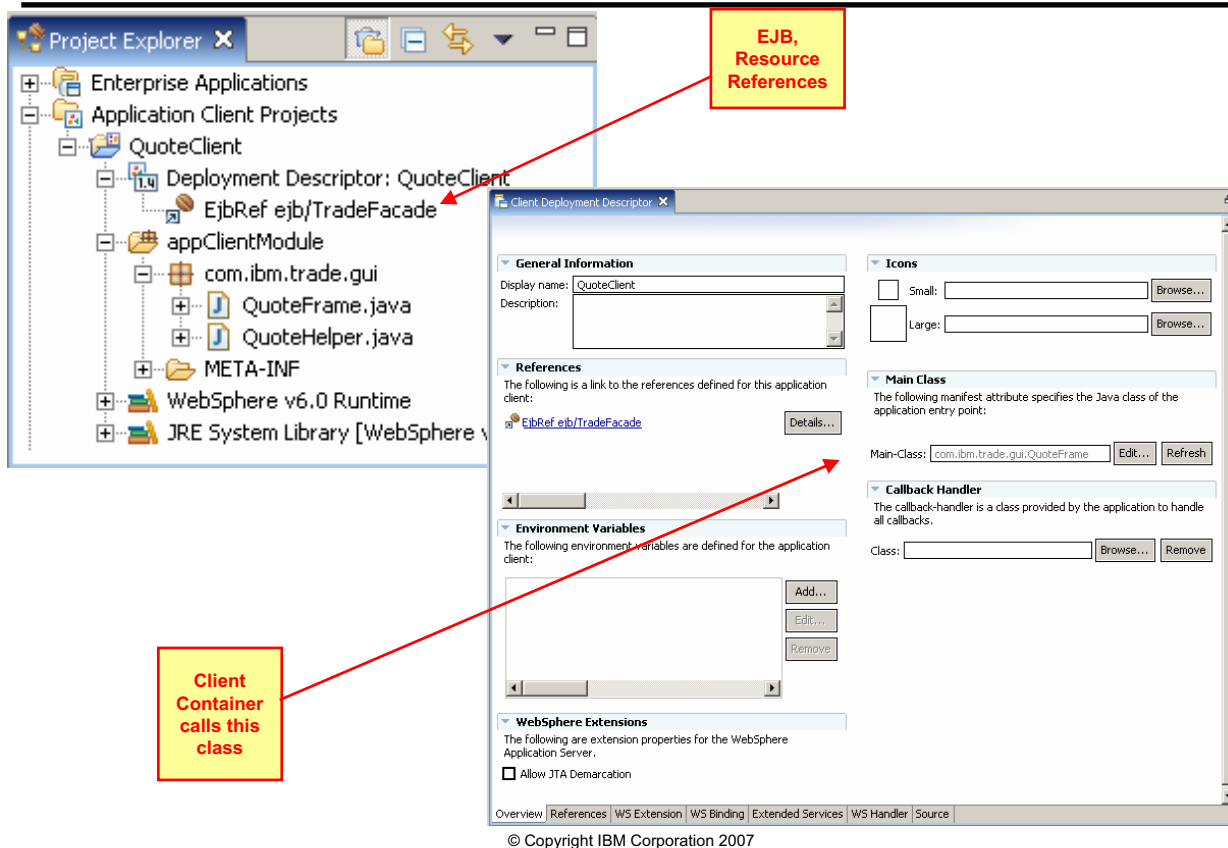


Figure 8-25. Application Client module

WA3616.3

## Notes:

An Application Client, being a stand-alone Java application, needs to have the class containing the main() method defined. This is the class that is loaded and starts the program by calling main().

***Instructor notes:***

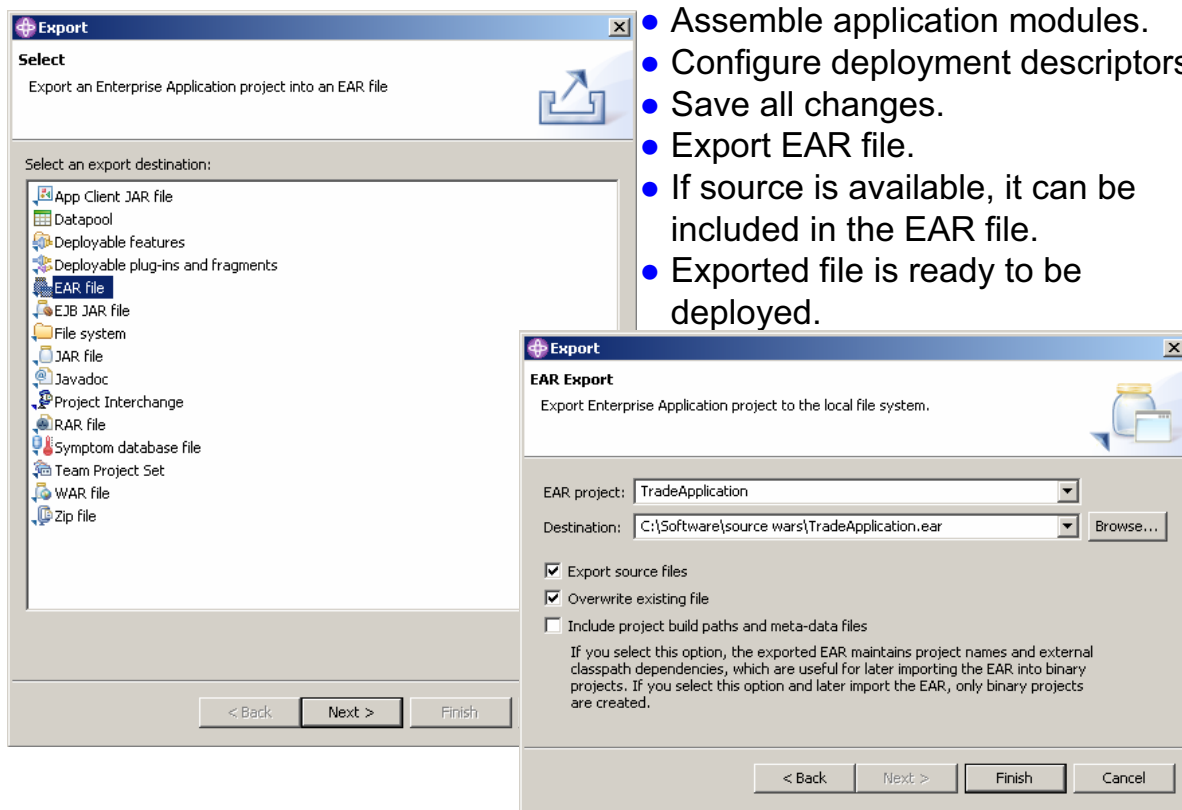
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Generating an EAR File for deployment

## Generating an EAR File for deployment



- Assemble application modules.
- Configure deployment descriptors.
- Save all changes.
- Export EAR file.
- If source is available, it can be included in the EAR file.
- Exported file is ready to be deployed.

© Copyright IBM Corporation 2007

Figure 8-26. Generating an EAR File for deployment

WA3616.3

### Notes:

After all modules have been imported and their deployment descriptors have been configured, all that is left to do is to export an EAR file which combines all the modules and their deployment descriptors.

Exporting an EAR file is very easy and only requires that you select the Enterprise Application and the location where to write out the new EAR file.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Checkpoint questions

## Checkpoint questions

---

1. The result of packaging an enterprise application is:?
  - A. An EAR file
  - B. A WAR file
  - C. An EJB JAR file
  - D. A RAR file
  
2. Which one of the following is not part of an enterprise application?
  - A. Application code modules
  - B. J2EE Deployment Descriptors
  - C. IBM Deployment Descriptor Extensions
  - D. WebSphere Variable Definitions
  
3. Enhanced Enterprise Applications are the recommended way to deploy applications in a production environment.
  - A. True
  - B. False

© Copyright IBM Corporation 2007

Figure 8-27. Checkpoint questions

WA3616.3

### Notes:

Write your answers here:

- 1.
- 2.
- 3.

**Instructor notes:**

Purpose —

Details —

## Checkpoint solutions

---

1. The result of packaging an enterprise application is:?  
A. An EAR file
2. Which one of the following is not part of an enterprise application?  
D. WebSphere Variable Definitions
3. Enhanced Enterprise Applications are the recommended way to deploy applications in a production environment?  
B. False

© Copyright IBM Corporation 2007

**Additional information** —

**Transition statement** — Next: Unit summary



## Unit summary

---

Having completed this unit, you should be able to:

- Describe the application assembly process
- Describe the use of the Application Server Toolkit (AST)
- Describe how AST relates to IBM Rational Web/Application Developer
- Use AST to define an enterprise application
- Use AST to define an EJB module
- Use AST to define a Web module
- Use AST to define an application client module
- Use AST to define application scope resources
- Export an enhanced EAR file

© Copyright IBM Corporation 2007

Figure 8-28. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

---

## Unit 9. Introduction to the Trade family of applications

### Estimated time

00:30

### What this unit is about

This unit describes the features and makeup of the Trade application.

### What you should be able to do

After completing this unit, you should be able to:

- Describe the use of the Trade application case study
- Describe the architecture of the Trade application case study
- Describe some of the Web views of the Trade application case study
- Describe the three EAR files that make up the complete application

### How you will check your progress

Accountability:

- Machine exercises

### References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

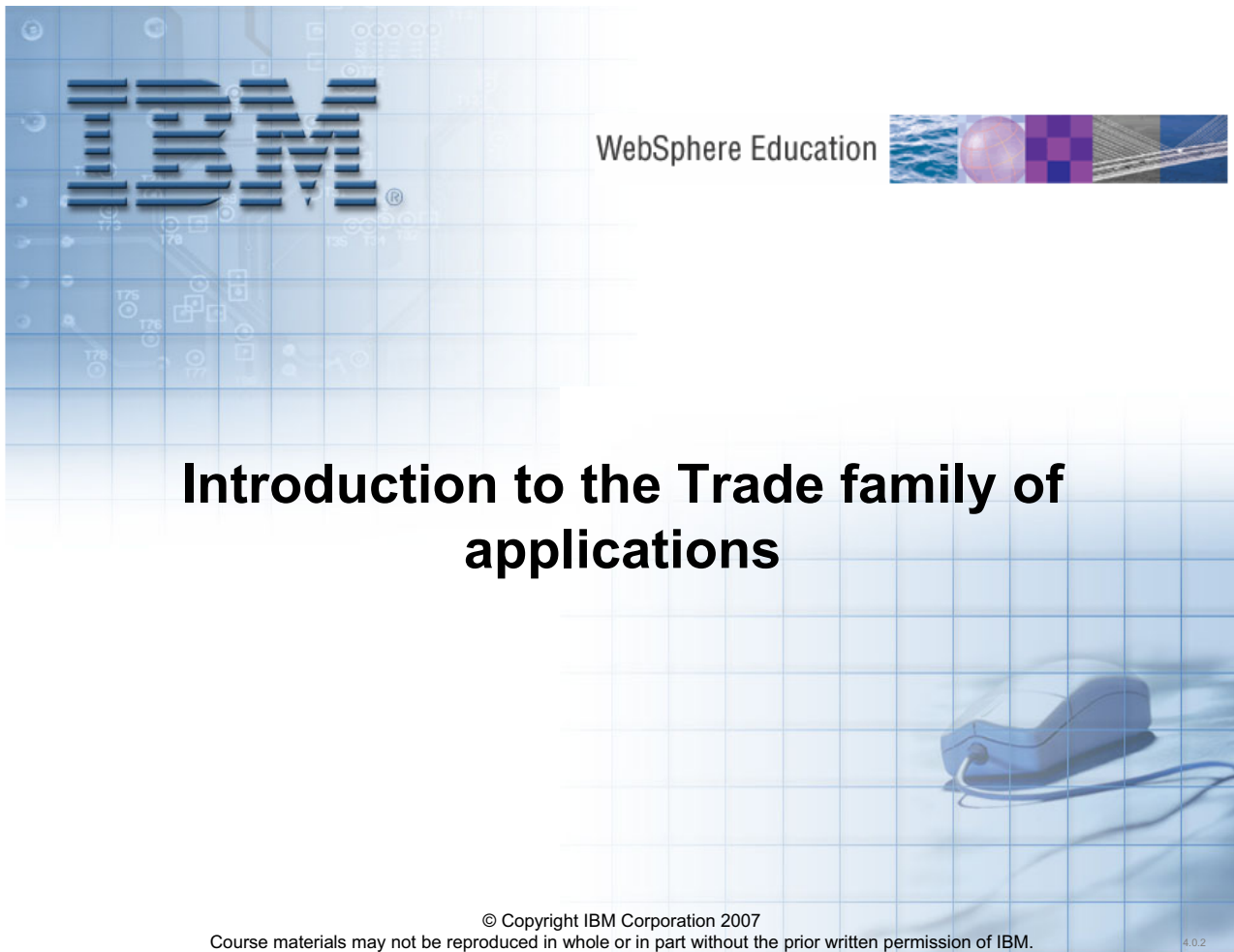


Figure 9-1. Introduction to the Trade family of applications

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe the use of the Trade application case study
- Describe the architecture of the Trade application case study
- Describe some of the Web views of the Trade application case study
- Describe the three EAR files that make up the complete application

© Copyright IBM Corporation 2007

Figure 9-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

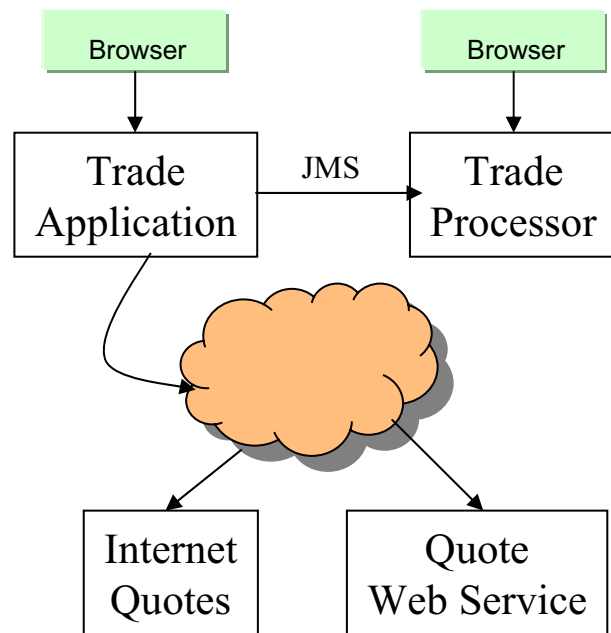
**Details —**

**Additional information —**

**Transition statement —** Next: Trade application family - Overview

## Trade application family overview

- Trade application provides simple stock quote and trading system
- Illustrates J2EE 1.4 functionality:
  - Three EAR files:
    - Trade Application (TradeApplication.ear)
    - Quote Web Service provider (QuoteWS.ear)
    - Trade Processor (TradeProcessor.ear)
  - EAR files contain:
    - One or more Web modules, EJB modules
    - HTML files, servlets and JSPs
    - EJBs: Session beans, Entity beans, Message Driven Beans
    - Web services client and provider



© Copyright IBM Corporation 2007

Figure 9-3. Trade application family overview

WA3616.3

### Notes:

The Trade family of applications is composed of three individual applications working in concert.

The TradeApplication.ear file represents the main part of the application and can run stand-alone, without the other module when connected to the network. A network connection is required because requesting local quotes requires the QUoteWS.ear application to be installed. Internet quotes are supplied by a public Web services provided, these quotes are 20 minutes delayed.

The QuoteWS.ear file provides a Web service that supplies local database quotes. This application can co-exist on the same application server as the Trade application, or in another application server on the internet. The URL for the Web service must be edited if the application is installed on another server.

The TradeProcessorApplication.ear file provides a ticker service displaying all buy and cell activity in the Trade application. Every time stocks are sold or bought an message (if Broadcast messages is enabled) is sent to the listener using JMS. This application can run on any server that is part of the message bus containing the queue destination.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Trade application - General functionality

## Trade application general functionality

---

- Additional capabilities of the Trade application
  - Register
  - Get quotes
  - View and update account information
  - View Holdings
  - Buy and Sell stock
  - View transaction history
  - Administration functions (adjust balances)
  - Logout
- URL: `http://<hostname>:<port>/Trade/web`

© Copyright IBM Corporation 2007

Figure 9-4. Trade application general functionality

WA3616.3

### **Notes:**

Other capabilities of the Trade application include:

Getting a list of the current user's holdings, view past transactions, buy and sell stocks and other administrative functions.

***Instructor notes:***

**Purpose —**

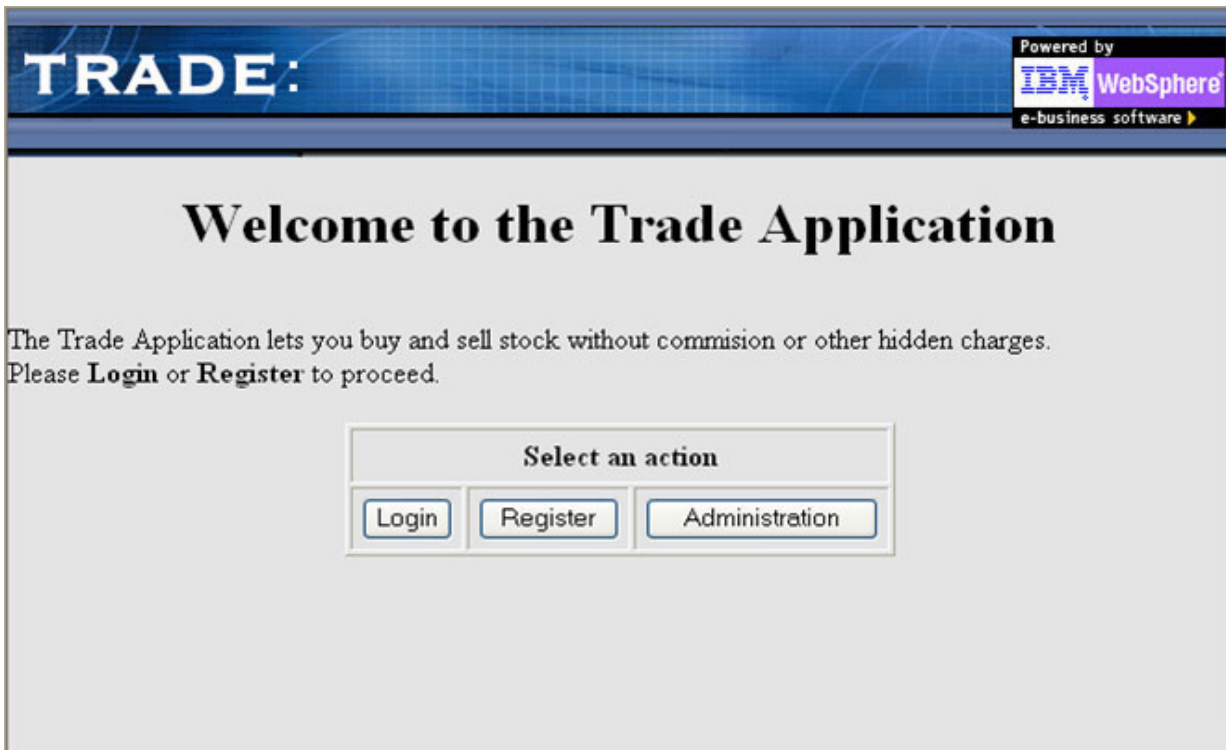
**Details —**

**Additional information —**

**Transition statement —** Next: Welcome page

## Welcome page

---



© Copyright IBM Corporation 2007

Figure 9-5. Welcome page

WA3616.3

### Notes:

From the welcome page a user can log in (assuming the user has previously registered), The Register option is for the user who needs to create a new account. The Administration option leads to the administrative functions of the application.

The address to get to this page is: <http://localhost/Trade/web/welcome.html>. Use port 9080 until the point in the exercises where the Web server is available.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Registration page

## Registration page

Account Information		Preferences	
Full Name:	<input type="text"/>	Use Internet quotes:	<input checked="" type="checkbox"/>
User Id:	<input type="text"/>	Broadcast transaction data:	<input type="checkbox"/>
Social Security:	<input type="text"/>		
Email:	<input type="text"/>		
Credit Account Number:	<input type="text"/>		
Starting Balance:	<input type="text"/>		
Home Address		Business Address	
Address Line 1:	<input type="text"/>	Address Line 1:	<input type="text"/>
Address Line 2:	<input type="text"/>	Address Line 2:	<input type="text"/>
City:	<input type="text"/>	City:	<input type="text"/>
State:	<input type="text"/>	State:	<input type="text"/>
Postal Code:	<input type="text"/>	Postal Code:	<input type="text"/>
Country:	<input type="text"/>	Country:	<input type="text"/>

Register Welcome

© Copyright IBM Corporation 2007

Figure 9-6. Registration page

WA3616.3

### Notes:

Selecting “Make same as Home Address will copy ALL the fields from the Home Address to the Business Address, anything entered in Business Address will be disregarded.

After registering, the user is automatically logged in and taken to the Account information page.

***Instructor notes:***

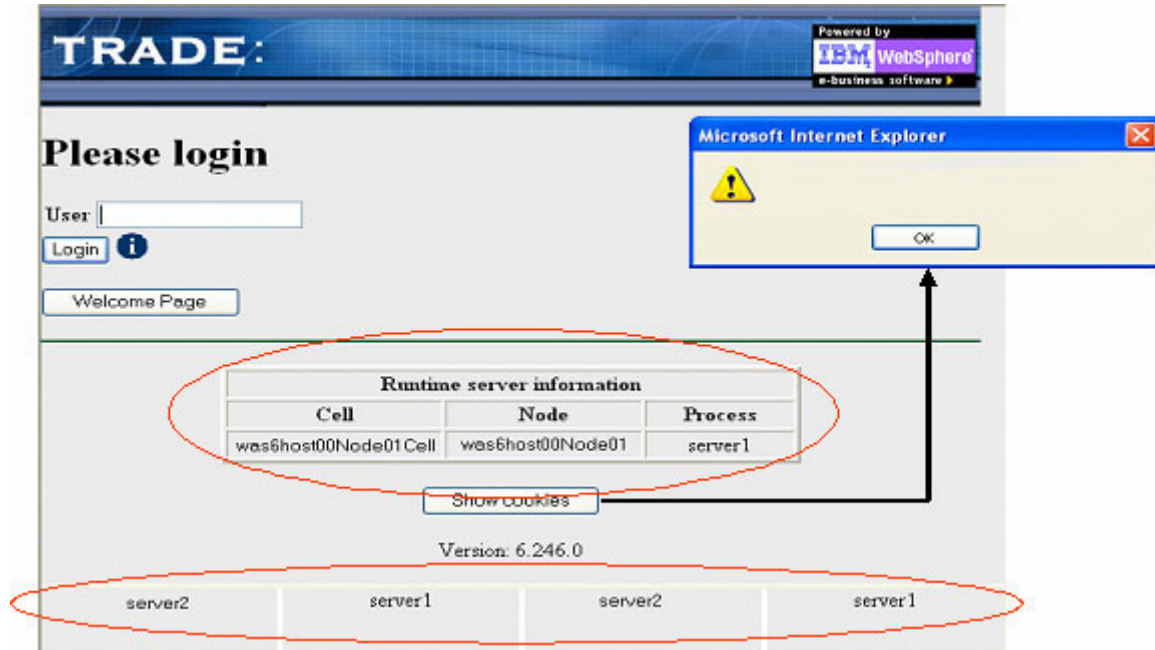
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Login page and runtime server information

## Login page and runtime server information



© Copyright IBM Corporation 2007

Figure 9-7. Login page and runtime server information

WA3616.3

### Notes:

In addition to allowing the user to log in to the application, this page provides additional information. The picture above shows that different parts of the pages are coming from different servers in a clustered environment. This will happen until affinity has been established.

Clicking "Show cookies" displays any cookies in the request, including JSESSIONID which is the cookie used to store session information. However at the login page no session ID cookies have been set. After login clicking "Show cookies" will display the session ID cookie's content.



***Instructor notes:***

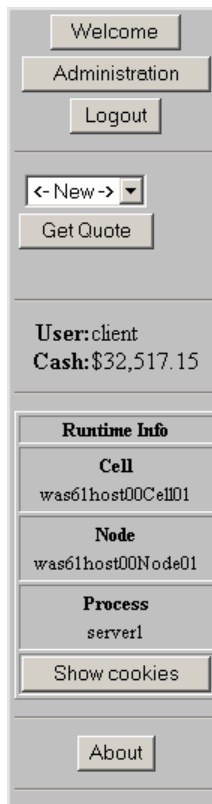
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Other information on pages

## Other information on pages



- This information appears to the left on all pages after logging in:
  - [Navigation buttons](#)
    - Welcome
    - Administration
    - Logout
  - [Get quotes](#)
    - Select from drop down list
    - Click <- New -> and type the symbol
    - Ten most recently used symbols kept in list box
  - [User information](#)
    - ID of currently logged in user
    - Current balance
  - [Runtime server information](#)
    - Cell name
    - Node name
    - Server name
  - [Show cookies button](#)
  - [About button](#)

© Copyright IBM Corporation 2007

Figure 9-8. Other information on pages

WA3616.3

### Notes:

Use the Login page to gain access to the application.

Note additional information at the bottom of the page. It represents runtime information to provide Cell, Cluster, Server and Cookie information.

The blue circle “i” icons throughout the application provide information as to what function the program performs when the action besides the icon is selected.

***Instructor notes:***


**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Account Information page

## Account Information page

<b>Account Information</b>		<b>Holdings</b>	<b>Buy and Sell</b>	<b>Transaction History</b>
<b>Account Information</b> Name: Jessie James AccountId: 87652289 Email: jjames@us.ibm.com Credit Card Number: 555444907 Balance: \$32,517.15		<b>Preferences</b> Use Internet quotes: <input checked="" type="checkbox"/> Broadcast transaction data: <input checked="" type="checkbox"/>		
<b>Home Address</b> 97 Milton Rd. Pittsburgh, PA 15222 USA		<b>Business Address</b> 1050 Chum Ave. Markham, ON L5G 1K9 Canada		
<input type="button" value="Update Account Page"/>				

© Copyright IBM Corporation 2007

Figure 9-9. Account Information page

WA3616.3

### Notes:

The account information show the particulars of the currently logged-in user. From this page the user can update account information.

***Instructor notes:***

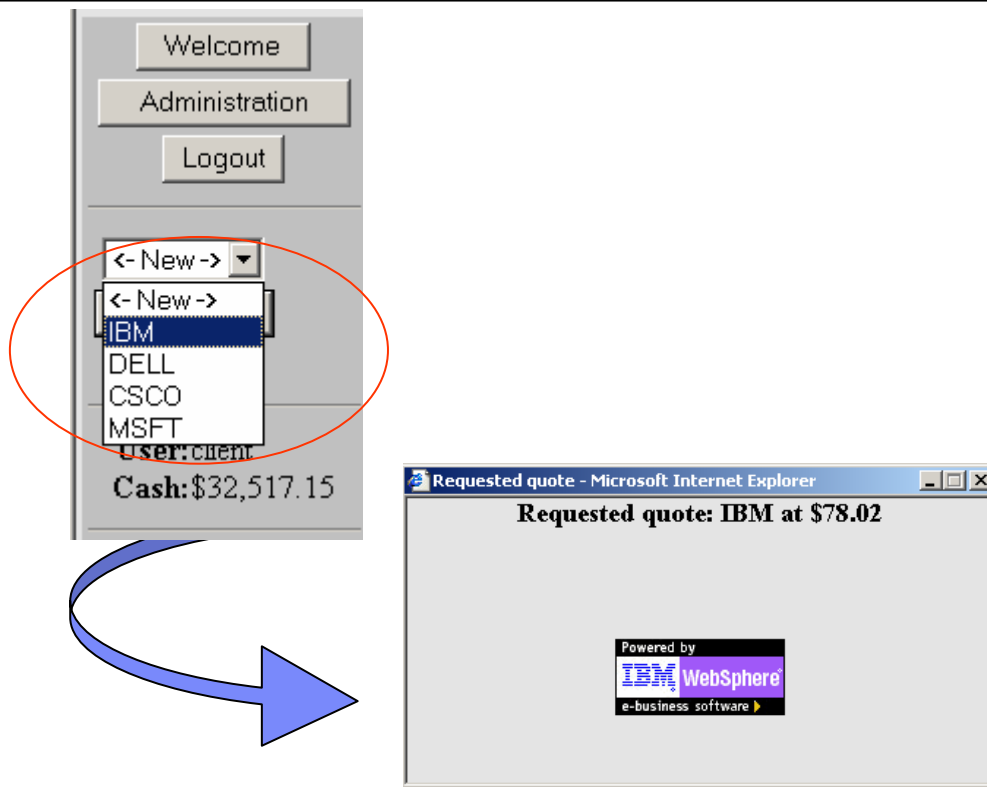
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Get Quote pop-up

## Get Quote pop-up



© Copyright IBM Corporation 2007

Figure 9-10. Get Quote pop-up

WA3616.3

### Notes:

Entering a valid stock symbol and clicking Get Quotes displays a pop-up window showing the most current price.

If Internet quotes are disabled, then the quotation comes from the database and is always the same value (QuoteWS.ear must be already installed).

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Update Account Information page

## Update Account Information page

<p><b>Account Information</b></p> <p>Name: Jessie James</p> <p>AccountId: 87652289</p> <p>Email: <input type="text" value="jjames@us.ibm.com"/></p> <p>Credit Account Number: <input type="text" value="555444907"/></p> <p>Balance: \$ <input type="text" value="32,517.15"/></p>	<p><b>Preferences</b></p> <p>Use Internet quotes: <input checked="" type="checkbox"/></p> <p>Broadcast transaction data: <input checked="" type="checkbox"/></p>
<p><b>Home Address</b></p> <p>Address Line 1: <input type="text" value="97 Milton Rd."/></p> <p>Address Line 2: <input type="text"/></p> <p>City: <input type="text" value="Pittsburgh"/></p> <p>State: <input type="text" value="PA"/></p> <p>Postal Code: <input type="text" value="15222"/></p> <p>Country: <input type="text" value="USA"/></p>	<p><b>Business Address</b> <input type="checkbox"/> <i>Make same as Home Address</i></p> <p>Address Line 1: <input type="text" value="1050 Chum Ave."/></p> <p>Address Line 2: <input type="text"/></p> <p>City: <input type="text" value="Markham"/></p> <p>State: <input type="text" value="ON"/></p> <p>Postal Code: <input type="text" value="L5G 1K9"/></p> <p>Country: <input type="text" value="Canada"/></p>

Save Updates

© Copyright IBM Corporation 2007

Figure 9-11. Update Account Information page

WA3616.3

### Notes:

On this page the user can change any personal information, decide whether Internet quotes are desired and also whether any buy and sell transaction will be broadcast to the Trade Processor application. This last option should only be chosen after the TradeProcessorApplication.ear has been installed, otherwise the user will get errors when buying or selling stock.

The user can, for testing purposes, adjust the account balance. For example a user can try to buy more stock than the account balance allows and see what happens.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Holdings page

## Holdings page

Account Information	Holdings	Buy and Sell	Transaction History
Holding	Shares	Latest Price Per Share	
DELL	5.0	\$24.59	
GOOD	20.0	\$19.00	
IBM	13.0	\$78.02	
YHOO	10.0	\$33.30	
<b>Total Equity: \$1,850.16</b>			

© Copyright IBM Corporation 2007

Figure 9-12. Holdings page

WA3616.3

**Notes:**

The holdings pages the current portfolio of the user. From here transactions can be made.

***Instructor notes:***

**Purpose —**

**Details —**


**Additional information —**

**Transition statement —** Next: Buy and Sell page


# Buy and Sell page

<b>Account Information</b>	<b>Holdings</b>	<b>Buy and Sell</b>	<b>Transaction History</b>
----------------------------	-----------------	---------------------	----------------------------

<b>Your current balance is:</b>	<b>Symbol</b>	<b>Quantity</b>	<b>BUY</b> 
\$32,517.15	<input type="text" value="TD"/>	<input type="text" value="12"/>	<input type="button" value="BUY"/>

<b>Holding</b>	<b>Shares</b>	<b>Latest Price Per Share</b>	<b>Shares to Sell</b> 
DELL	5.0	\$24.59	<input type="text"/> <input type="button" value="SELL"/>
GOOD	20.0	\$19.00	<input type="text"/> <input type="button" value="SELL"/>
IBM	13.0	\$78.02	<input type="text"/> <input type="button" value="SELL"/>
YHOO	10.0	\$33.30	<input type="text"/> <input type="button" value="SELL"/>

**Total Equity:** \$1,850.16

© Copyright IBM Corporation 2007

Figure 9-13. Buy and Sell page

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Transaction History page

## Transaction History page

Account Information	Holdings	Buy and Sell	Transaction History	
Trade Date	Holding	Price	Shares	Type
06/29/06 11:36 AM	IBM	\$76.99	1.0	SELL
06/29/06 11:34 AM	IBM	\$76.90	1.0	SELL
06/29/06 11:31 AM	IBM	\$76.99	1.0	SELL
06/29/06 11:30 AM	IBM	\$77.00	1.0	SELL
06/29/06 11:29 AM	IBM	\$76.91	12.0	BUY
06/28/06 01:51 PM	DELL	\$23.73	1.0	BUY
06/28/06 01:49 PM	IBM	\$76.26	2.0	BUY
06/28/06 01:47 PM	YHOO	\$31.97	2.0	SELL
06/28/06 01:41 PM	GOOD	\$18.53	1.0	SELL
06/28/06 01:36 PM	GOOD	\$18.53	2.0	SELL
06/28/06 01:30 PM	YHOO	\$31.96	12.0	BUY
06/28/06 01:29 PM	GOOD	\$18.59	23.0	BUY
02/22/05 11:32 AM	DELL	\$40.45	1.0	SELL
02/22/05 11:32 AM	DELL	\$40.44	5.0	BUY
02/22/05 11:32 AM	IBM	\$93.36	3.0	BUY

© Copyright IBM Corporation 2007

Figure 9-14. Transaction History page

WA3616.3

### Notes:

The transaction history page shows ALL transactions the user has ever made.

***Instructor notes:***

**Purpose —**

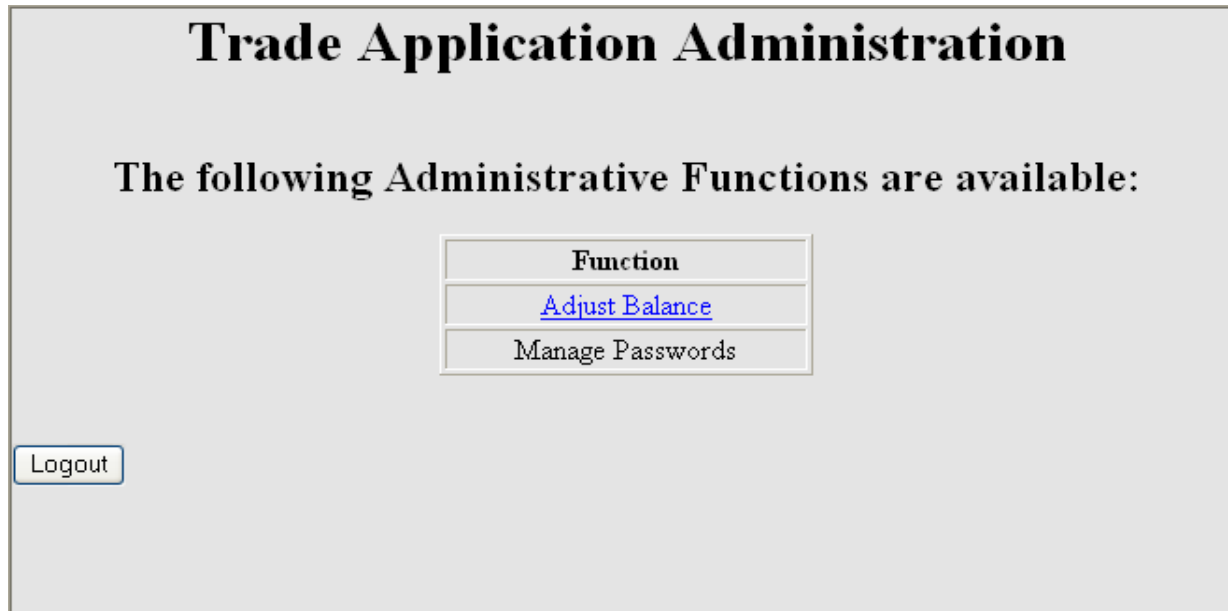
**Details —**

**Additional information —**

**Transition statement —** Next: Administrative Functions menu

## Administrative Functions menu

---



© Copyright IBM Corporation 2007

Figure 9-15. Administrative Functions menu

WA3616.3

### Notes:

The Administration part of the application is still under development and the only function available is to adjust a user's balance. As an administrator you can adjust any valid user's balance.

Once security is enable on the application server, the administrator will get challenged before getting to this page. A valid TradeAdmin userID and password must be supplied to log in.

Logging out of the Administrative part of the application will result in an error when security is not enabled. This is normal as you cannot log out if you have not logged in. And you cannot log in when security is disabled.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Administration – Adjust user balance

## Administration – Adjust user balance

### All user balances

Below are the current users of the Trade Application.

To adjust a user's balance select the user and click **Set balance**.

User Id	Full Name	Account Number	Current Balance	New Balance
ckid	Clyde Kid	147789999	\$12,759.77	<input type="text"/>
lranger	Larry Ranger	148443822	\$1,231,345.00	123145.89
jbond	John Bond	343335454	\$77,000.07	<input type="text"/>
mshark	Mako Shark	123444321	\$78,987.77	<input type="text"/>
bbird	Bob Bird	123456789	\$25,000.00	<input type="text"/>
userid	Mr. Student	987654321	\$321,321.00	<input type="text"/>
client	Jesse James	87652289	\$32,517.15	35000

© Copyright IBM Corporation 2007

Figure 9-16. Administration – Adjust user balance

WA3616.3

### Notes:

On this page the administrator selects the account and enters a new balance. Clicking Set balance makes the change effective.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Trade application - Modules

## Trade application modules

---

- There are two modules:
  - [TradeWeb.war](#) – Trade subsystem Web application
    - Registration, Login, Holdings, Buy and Sell, Transaction History with supporting servlets, JSPs and other code
  - [TradeEJB.jar](#) – Trade subsystem EJBs
    - Session - TradeFacade
    - Entity - Account, Address, Holdings, Profile, Symbol, TransactionHistory

© Copyright IBM Corporation 2007

Figure 9-17. Trade application modules

WA3616.3

### **Notes:**

If a network connection is available the Internet is used to obtain 20 minute delayed quotes from a Web service. If the Web service is unavailable the application will failover to the local database to obtain “fake” quotes. If there is no internet connection, or the internet is too slow the user can configure his account to obtain quotes from the local database.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: QuoteWS - General functionality

## QuoteWS general functionality

---

- The Trade application can get quotes from either
  - An Internet quote service
  - The Quote Web Service (QuoteWS)
  - Selectable through Account Preferences page
    - If Internet quotes are configured, but not available, failover is automatic to this service
  
- QuoteWS implements a Web services provider
  - Provides quotes based on values store in the QUOTE database
  - Used when an internet connection is not available or configured
  - Provides a very large, but limited number of symbols
    - Other symbols added as user buys symbols available on the Internet

© Copyright IBM Corporation 2007

Figure 9-18. QuoteWS general functionality

WA3616.3

### **Notes:**

If both Internet quotes and QuoteWS are NOT available, trying to obtain a quote results in a 404 error. Note that, upon initial installation of the Trade Application, only Internet quotes will work (assuming an Internet connection). Once the QuoteWS application has also been installed, then the Trade Application will be able to use either, depending on user choice.

QuoteWS does not need to be installed on the same machine as the Trade Application. Its location is configurable through a properties file.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: QuoteWS - Modules

## QuoteWS modules

---

- This EAR file contains two modules
  - QuoteEJB.jar
    - Provides service to back-end QUOTE database and table
    - EJBs:
      - Session – QuoteFaçade
      - Entity - Quote
  - QuoteWSRouter
    - Provides routing for Web service

© Copyright IBM Corporation 2007

Figure 9-19. QuoteWS modules

WA3616.3

### **Notes:**

This entire application was generated by IBM Rational Application Developer using the “Create Web Service from an EJB” wizard.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Trade Processor - General functionality

## Trade Processor general functionality

- Provides a table showing the transactions on the system
- The processor purges transactions after 30 seconds
- Uses MDB to listen to a queue where the trade application sends messages
- Formats and displays messages
- URL: `http://<hostname>:<port>/Trade/processor/monitor.html`

TRADE PROCESSOR

Trade Requests To Be Processed On Server: Server1

Refresh every:  seconds

Account	Buy/Sell	Symbol	Qty	Total cost	Transaction
87652289 (Jessie James)	SELL	DELL	2.0	\$49.18	server1 - 35
87652289 (Jessie James)	SELL	GOOD	1.0	\$19.00	server1 - 36
87652289 (Jessie James)	SELL	YHOO	5.0	\$166.50	server1 - 37

© Copyright IBM Corporation 2007

Figure 9-20. Trade Processor general functionality

WA3616.3

### Notes:

The Trade listener page displays in-process transactions.

Refreshes every 10 seconds by default, this is configurable on the GUI.

Thirty seconds after being received, transactions are considered processed and are removed from the list.

TradeListener does not need to be installed on the same machine as the Trade Application. The Trade Application sends a message to a queue. That queue will be defined administratively. The Trade application sends messages to the queue and the TradeListener removes messages from the queue.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Trade Ticker

## Trade Ticker

---

- Displays a scrolling ticker tape of the last few transactions
- Drops off oldest transaction when scrolling limit is reached
- URL:  
`http://<hostname>:<port>/Trade/processor/ticker`



© Copyright IBM Corporation 2007

Figure 9-21. Trade Ticker

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Combined Ticker and Processor page

## Combined Ticker and Processor page

- Combines Ticker and Processor on to single page with two frames
- More convenient to watch trading activity
- URL:  
<http://<hostname>:<port>/Trade/processor/monitor.html>

The screenshot shows a web application interface. At the top, a green ticker displays market data: "PG +5.0 @ \$56.30 TD -1.0 @ \$51.52 IBM +3.0 @ \$78.02". Below this is a blue header bar with the text "TRADE PROCESSOR" on the left and "Powered by IBM WebSphere e-business software Version: 6.1.246.0" on the right. The main content area has a yellow background and is titled "Trade Requests To Be Processed On Server: Server1". Below the title is a refresh control: "Refresh every: 10 seconds". A table lists trade requests with columns for Account, Buy/Sell, Symbol, Qty, Total cost, and Transaction.

Account	Buy/Sell	Symbol	Qty	Total cost	Transaction
87652289 (Jessie James)	SELL	DELL	2.0	\$49.18	server1 - 35
87652289 (Jessie James)	SELL	GOOD	1.0	\$19.00	server1 - 36
87652289 (Jessie James)	SELL	YHOO	5.0	\$166.50	server1 - 37

© Copyright IBM Corporation 2007

Figure 9-22. Combined Ticker and Processor page

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Trade Processor - Modules

## Trade Processor modules

---

- TradeProcessorEJB.jar
  - Message-driven - TradeProcessorEJB
- TradeProcessorWeb.war
  - One servlet
  - One JSP
  - Utility classes for parsing XML message

© Copyright IBM Corporation 2007

Figure 9-23. Trade Processor modules

WA3616.3

### **Notes:**

This is an extremely simple application with one message driven bean to receive the transaction messages, and one servlet to display the transactions in waiting. JavaScript on the page refreshes at a set interval to get new information from the servlet.



***Instructor notes:***

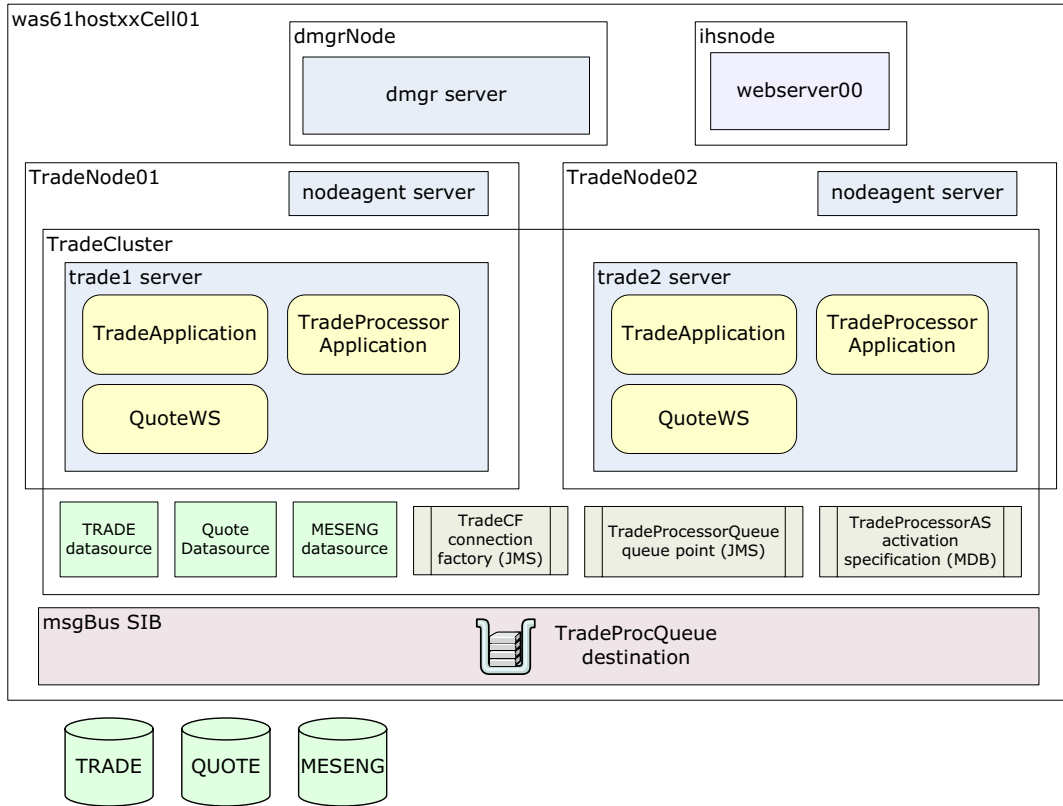
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Trade deployment architecture

# Trade deployment architecture



© Copyright IBM Corporation 2007

Figure 9-24. Trade deployment architecture

WA3616.3

## Notes:

This diagram shows the runtime deployment architecture of the Trade application that is constructed in the exercises.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe the use of the Trade application case study
- Describe the architecture of the Trade application case study
- Describe some of the Web views of the Trade application case study
- Describe the three EAR files that make up the complete application

© Copyright IBM Corporation 2007

Figure 9-25. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

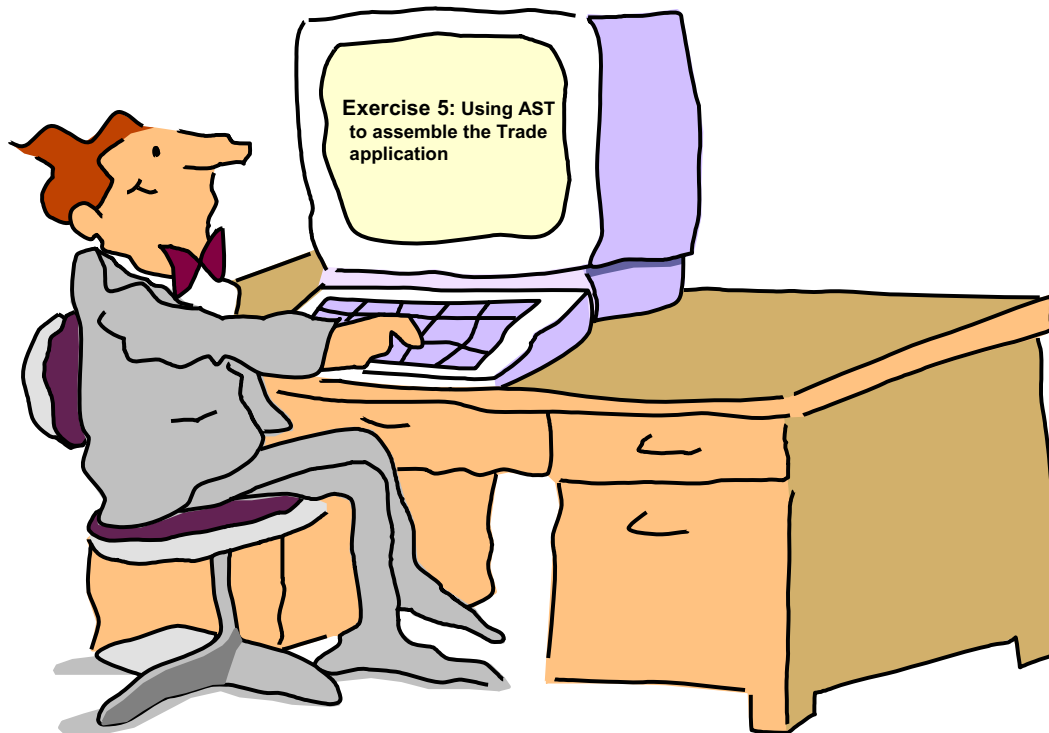
**Details —**

**Additional information —**

**Transition statement —** Next: Lab exercise

## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 9-26. Lab exercise

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**





---

# Unit 10. Application installation

## Estimated time

01:00

## What this unit is about

This unit describes how to install enterprise applications into WebSphere Application Server.

## What you should be able to do

After completing this unit, you should be able to:

- Describe methods of installing enterprise applications in WebSphere Application Server
- Describe methods of installing applications using the administrative console
- Describe how to install applications using WebSphere Rapid Deployment
- Describe fine grain application update
- Describe enterprise application properties

## How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center



Figure 10-1. Application installation

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe methods of installing enterprise applications in WebSphere Application Server
- Describe methods of installing applications using the administrative console
- Describe how to install applications using WebSphere Rapid Deployment
- Describe fine grain application update
- Describe enterprise application properties

© Copyright IBM Corporation 2007

Figure 10-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

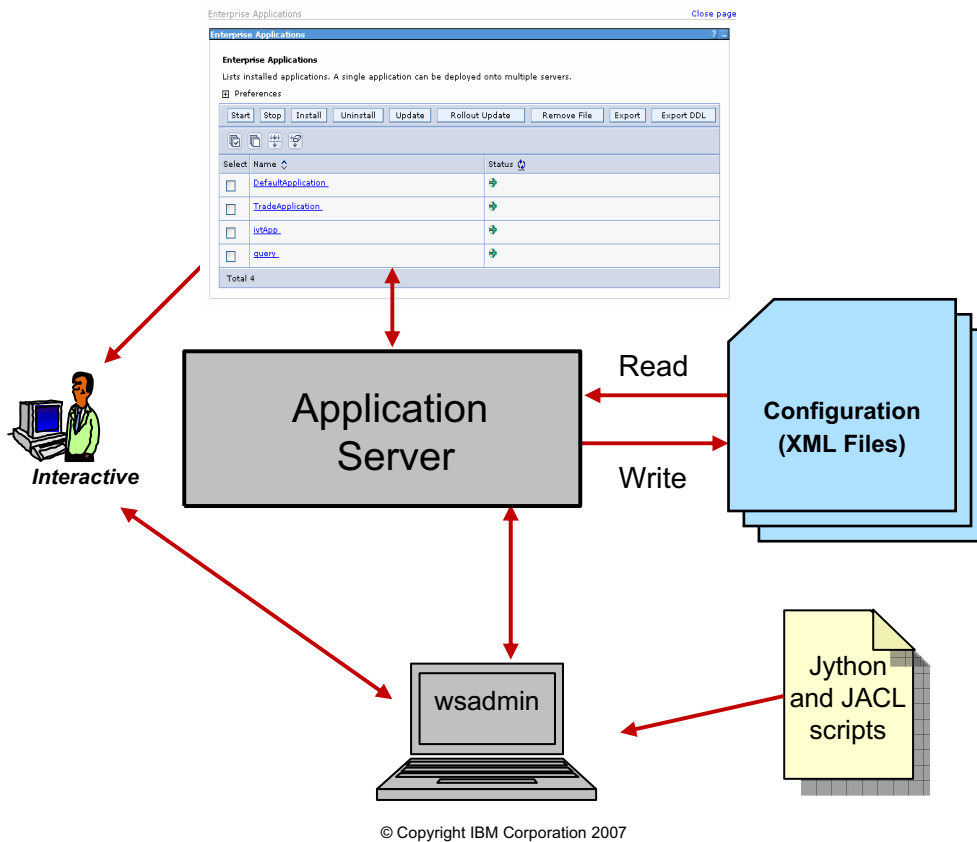
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Installing enterprise applications

# Installing enterprise applications



© Copyright IBM Corporation 2007

Figure 10-3. Installing enterprise applications

WA3616.3

## Notes:

The two most common ways to install applications on the server is to use the WebSphere administrative console, or a Jython script through wsadmin. JACL scripts are also supported at the moment, but the current statement of direction is that JACL has been deprecated.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Installation tasks

## Installation tasks

---

- Configure the application environment as required
  - Variables, virtual hosts, class path, security, and so forth
- Configure application resources
  - JDBC provider, data sources, JMS resources, SIBus
- Install application
- Manage static content
  - Files served by Web server, not in EAR file
  - Leave static content in EAR file

© Copyright IBM Corporation 2007

Figure 10-4. Installation tasks

WA3616.3

### **Notes:**

In general it is a good practice to leave the static content used by the application in the ear file and let the infrastructure take care of serving and caching the static content. In the end leaving the static in the ear file can be as effective as moving it to the Web server.



***Instructor notes:***

**Purpose —**

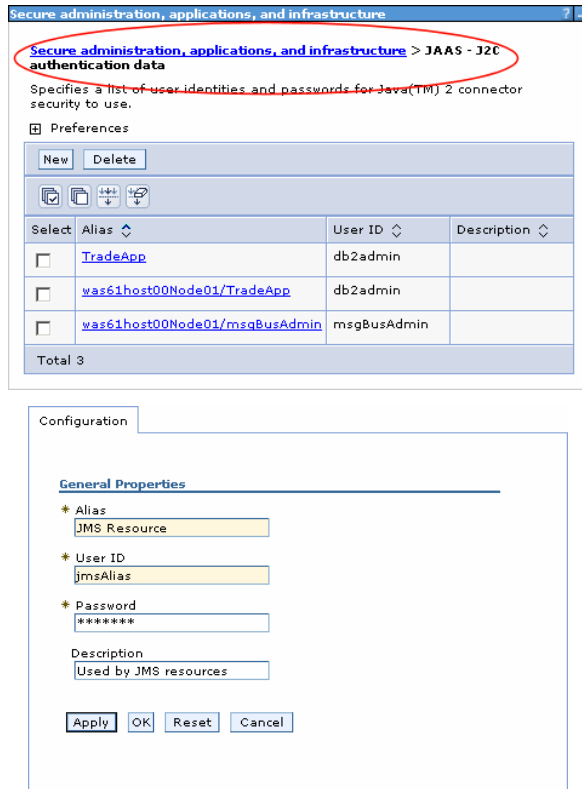
**Details —**

**Additional information —**

**Transition statement —** Next: Creating a J2C authentication alias

# Creating a J2C authentication alias

- Wizard can be reached from many pages in console
- Provide
  - Alias name
    - Console prefixes name with node name
  - User ID and corresponding password
  - Optional description
  - Used by EJBs, data sources, JMS resources, SIBus resources, and so on



© Copyright IBM Corporation 2007

Figure 10-5. Creating a J2C authentication alias

WA3616.3

## Notes:

You can access the page to create new J2C aliases from several places in the console, basically from most pages where an authentication alias is required.

J2C aliases can also be created through scripting.

Note that J2C aliases are not scoped resources. So when installing an enhanced EAR file which has a J2C alias definition within it the alias will be created and be available to any resource in the server.

Note that the WebSphere administration console prepends the node name to the alias name you provide. Also note that this does not happen when creating aliases from a wsadmin script.

***Instructor notes:***

**Purpose —**

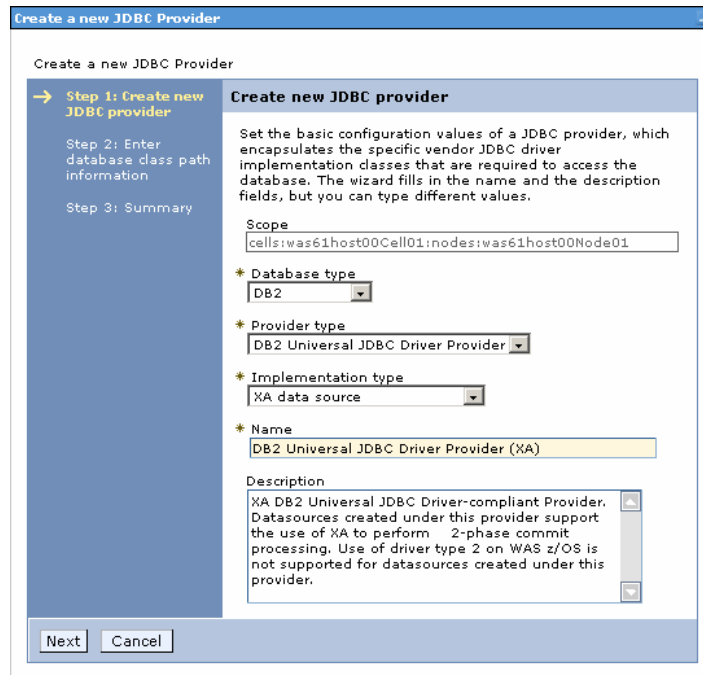
**Details —**

**Additional information —**

**Transition statement —** Next: Creating JDBC provider resources (1 of 2)

## Creating JDBC provider resources (1 of 2)

- Create JDBC provider before or while defining data sources.
- One JDBC provider needed for each database driver type.
- JDBC providers can be defined at cell, node, server or application scope (in an enhanced EAR file).



© Copyright IBM Corporation 2007

Figure 10-6. Creating JDBC provider resources (1 of 2)

WA3616.3

### Notes:

Starting in WebSphere Application Server V6.1 JDBC providers can be defined in the wizard that creates the data source.

***Instructor notes:***

**Purpose —**

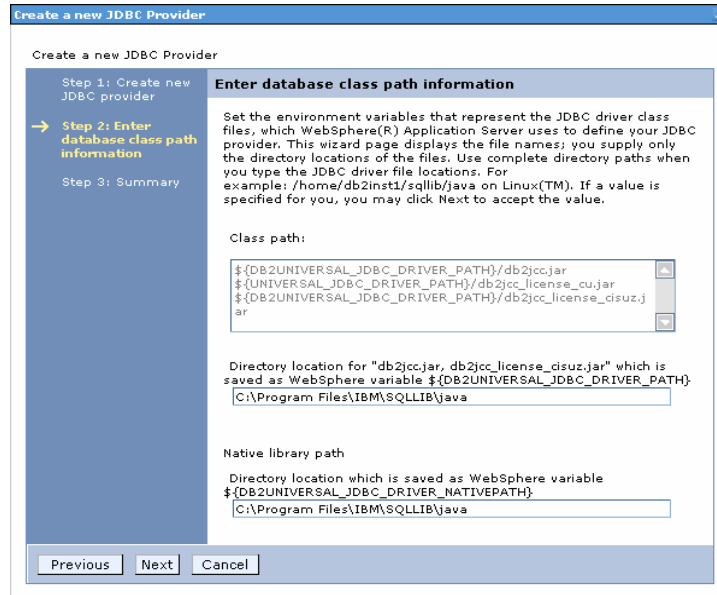
**Details —**

**Additional information —**

**Transition statement —** Next: Creating JDBC provider resources (2 of 2)

## Creating JDBC provider resources (2 of 2)

- Starting in WebSphere Application Server 6.1 the JDBC Driver paths can be defined in the wizard.
  - Paths entered here create corresponding WebSphere environment entries.
  - If environment entries already exist they are used to pre-fill the path entry fields.
  - If modified the environment variable is modified.



© Copyright IBM Corporation 2007

Figure 10-7. Creating JDBC provider resources (2 of 2)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

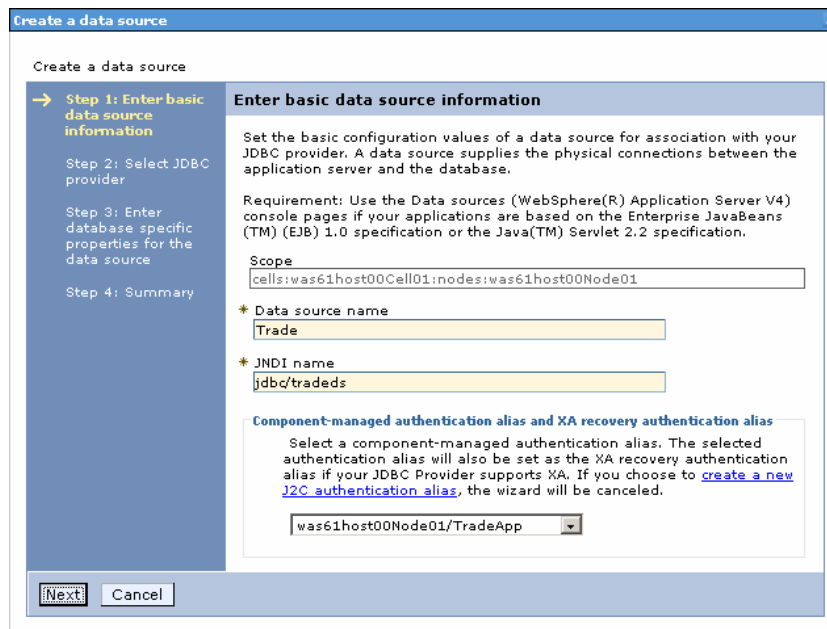
**Details —**

**Additional information —**

**Transition statement —** Next: Creating a data source (1 of 3)

## Creating a data source (1 of 3)

- Wizard driven process.
- Additional properties are available after creating data source.



© Copyright IBM Corporation 2007

Figure 10-8. Creating a data source (1 of 3)

WA3616.3

### Notes:

Installed applications that must interact with relational databases use JDBC providers for data access. Together, the JDBC provider and data source objects are functionally equivalent to the J2EE Connector architecture (JCA) connection factory (which provides access to non-relational databases). Installed applications use a *data source* to access the data from the database.

A data source is associated with a JDBC provider that supplies the specific JDBC driver implementation class. The data source represents the J2EE Connector Architecture (JCA) connection factory for the relational resource adapter. Application components use the data source to access connection instances to a specific database; a connection pool is associated with each data source.

You can create multiple data sources with different settings, and associate them with the same JDBC provider. (One reason to do this is to provide access to different databases.) JDBC providers that are supported by WebSphere Application Server are required to implement one or both of the following data source interfaces, which are defined by Sun



Microsystems. These interfaces enable the application to run in a single-phase or two-phase transaction protocol.

***Instructor notes:***

**Purpose —**

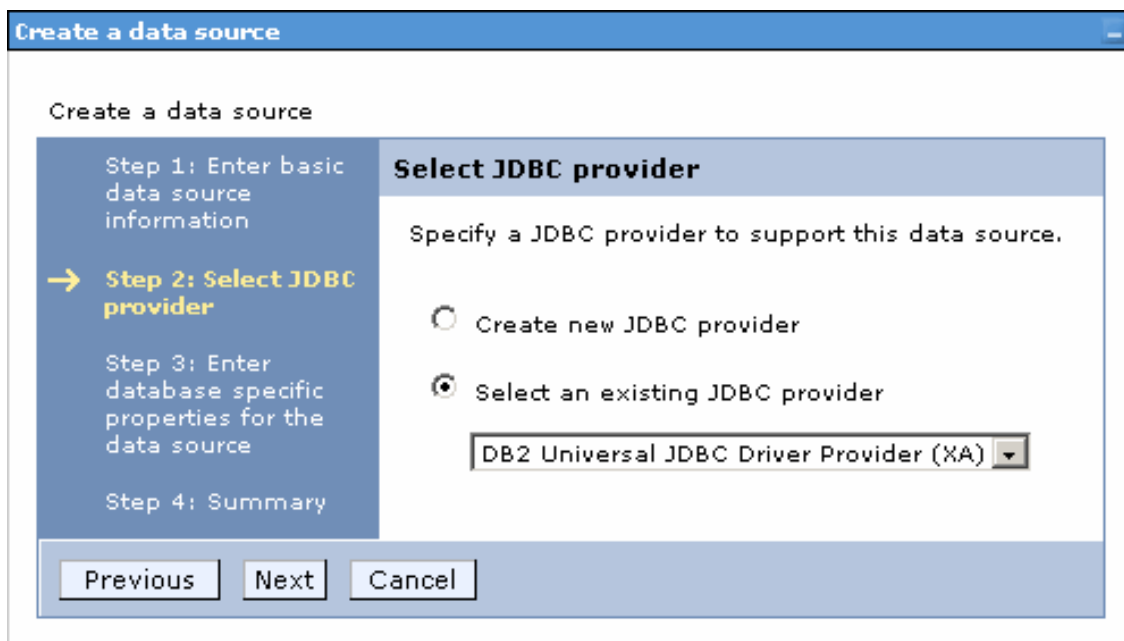
**Details —**

**Additional information —**

**Transition statement —** Next: Creating a data source (2 of 3)

## Creating a data source (2 of 3)

- On this page you get a chance to create a new JDBC provider, or to use a previously created provider.



© Copyright IBM Corporation 2007

Figure 10-9. Creating a data source (2 of 3)

WA3616.3

### Notes:

Selecting to create a new JDBC provider causes the wizard to be extended as the Create JDBC provider executes.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Creating a data source (3 of 3)

## Creating a data source (3 of 3)

- Provide the database specific parameters:
  - Database name
  - Driver type
  - Database server URL or host name and communication port number
- Type 4 drivers are recommended if available.
  - Pure Java drivers which translate JDBC calls to native database protocol
- Choose whether data source will be used with CMP beans.

© Copyright IBM Corporation 2007

Figure 10-10. Creating a data source (3 of 3)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Make application files available to server

---

## Make application files available to server

---

- The default directory assigned to hold the .ear file before it is installed is:  
`<profile_root>\<profile>\installableApps`
- Most often, the application file you receive is an Enterprise Application Archive (.ear) file.
- The .ear file is composed of one or more of the following objects and their deployment descriptors:
  - Enterprise JavaBeans (EJB) module, consisting of Java archive (.jar) files
  - Web Archive (.war) files, consisting of Web content, JSPs, and servlets
  - Application Client Archive (.jar) file
  - Resource Archive files (.rar)
  - Supporting utility (.jar) files

© Copyright IBM Corporation 2007

Figure 10-11. Make application files available to server

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

**Details** —

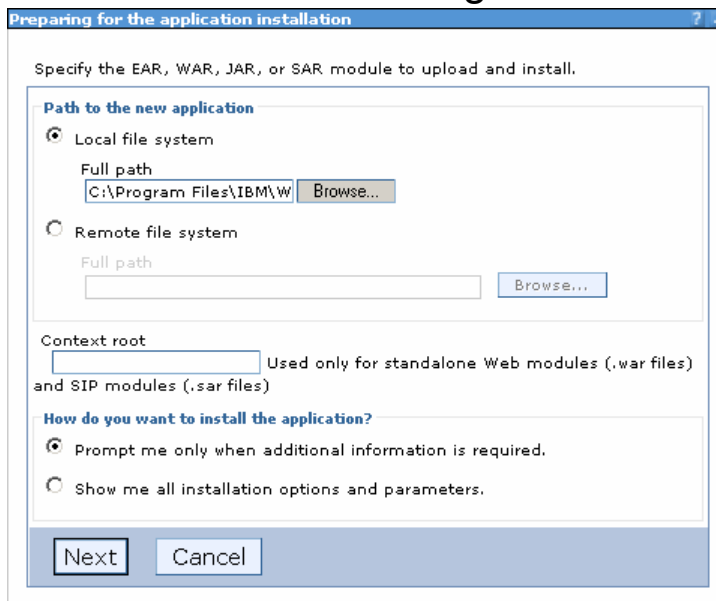
**Additional information** —

**Transition statement** — Next: Enterprise application installation (1 of 7)



## Enterprise application installation (1 of 7)

- Select **Applications -> Enterprise Applications -> Install New Application** or **Applications -> Install New Application**
- Click **Browse** and locate the EAR file containing the application's modules
- Click **Next**



© Copyright IBM Corporation 2007

Figure 10-12. Enterprise application installation (1 of 7)

WA3616.3

### Notes:

New with WebSphere Application Server V6.1 is the choice of how many configuration pages to go through while installing the application. If you only want to see the pages which require additional information choose the first radio button; if you want to see all of the pages of the installation wizard, select the second radio button.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Enterprise application installation (2 of 7)

## Enterprise application installation (2 of 7)

- Generate or override bindings
- Virtual host for Web modules
- In most cases this page can be skipped without changes
- Click **Next**

Preparing for the application installation

Choose to generate default bindings and mappings.

Generate Default Bindings

**Override:**

Do not override existing bindings

Override existing bindings

**Virtual Host**

Do not use default virtual host name for Web or SIP modules

Use default virtual host name for Web and SIP modules:

Host name  
default\_host

Specific bindings file  
 Browse...

Previous Next Cancel

© Copyright IBM Corporation 2007

Figure 10-13. Enterprise application installation (2 of 7)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

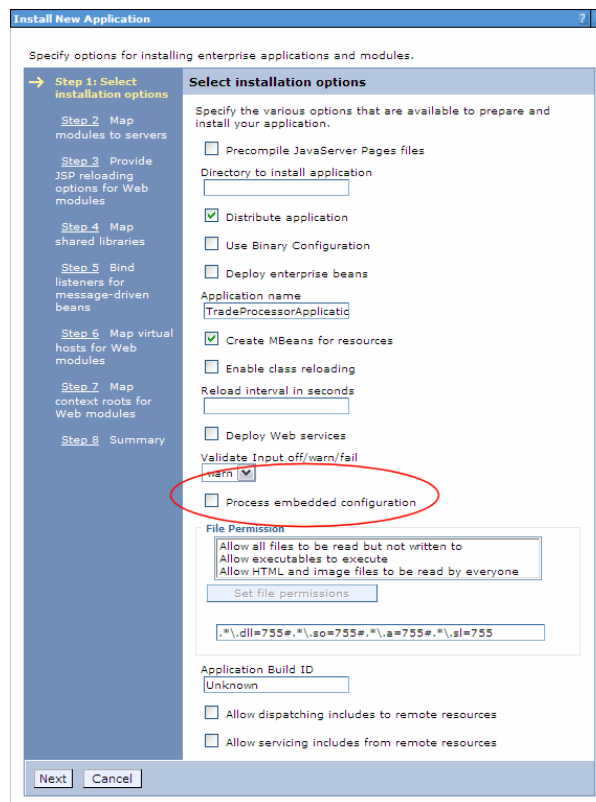
**Details —**

**Additional information —**

**Transition statement —** Next: Enterprise application installation (3 of 7)

## Enterprise application installation (3 of 7)

- Depending on choice on first page of wizard, number of steps will vary
- Pre-compile JSPs
- Change installed application directory
- Deploy EJBs
- Change the application name
- Use or discard application scoped resources



© Copyright IBM Corporation 2007

Figure 10-14. Enterprise application installation (3 of 7)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Enterprise application installation (4 of 7)

## Enterprise application installation (4 of 7)

- Installation wizard allows customizing the configuration of the application prior to installation.
- Depending on the contents of the EAR file, and selection on first page, not all steps are available.

Step	Function
2	Map modules to servers
3	Select current back-end ID
4	Provide JNDI names for beans
5	Provide default data source mapping for modules containing 2.0 entity beans
6	Map datasources for all 2.0 CMP beans
7	Map EJB references to beans
8	Map resource references to references
9	Map virtual hosts for Web modules
10	Map security roles to users/groups
11	Ensure all unprotected 2.0 methods have the correct level of protection
12	Summary

© Copyright IBM Corporation 2007

Figure 10-15. Enterprise application installation (4 of 7)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Enterprise application installation (5 of 7)



## Enterprise application installation (5 of 7)

- This page determines which modules run on which application server.
  - Select the application server
  - Check modules for the selected servers
  - Click **Apply**
- When using managed Web servers it is also necessary to map the modules to Web servers.



Figure 10-16. Enterprise application installation (5 of 7)

WA3616.3

### Notes:

Each module of a deployed application must be mapped to one or more target servers. The target server can be an application server, cluster of application servers, or Web server.

***Instructor notes:***

**Purpose —**

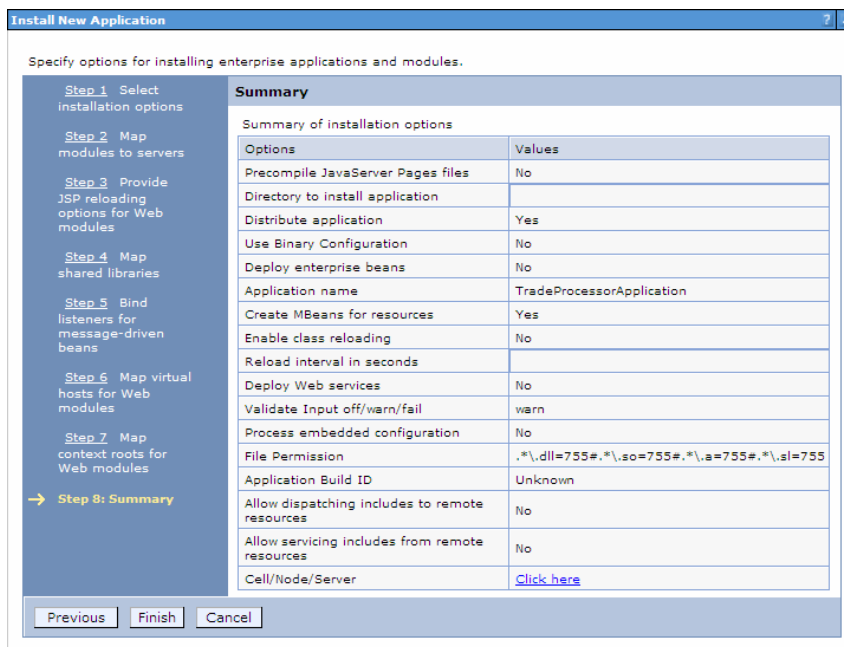
**Details —**

**Additional information —**

**Transition statement —** Next: Enterprise application installation (6 of 7)

## Enterprise application installation (6 of 7)

- Once all configuration parameters have been set to your requirements, click **Finish** to install the enterprise application
- Save the configurations after installation.



© Copyright IBM Corporation 2007

Figure 10-17. Enterprise application installation (6 of 7)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

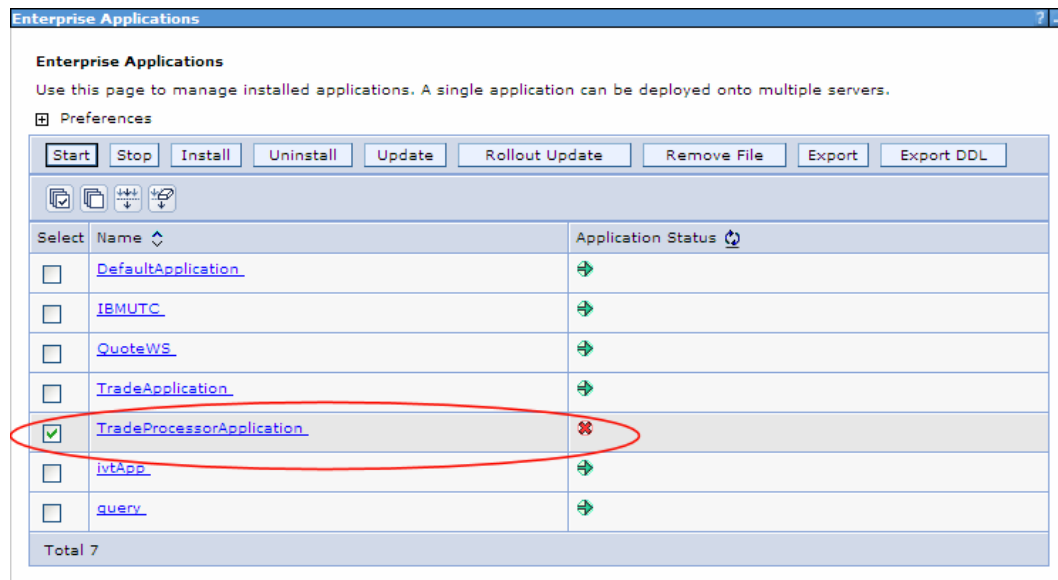
**Details —**

**Additional information —**

**Transition statement —** Next: Enterprise application installation (7 of 7)

## Enterprise application installation (7 of 7)

- After the application is installed you can:
  - Start and Stop
  - Uninstall
  - Update
  - Rollout Update
  - Remove File
  - Export or Export DDL



© Copyright IBM Corporation 2007

Figure 10-18. Enterprise application installation (7 of 7)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Managing applications

---

## Managing applications

---

- WebSphere administrative console
  - Easy way to install or update an application.
  - Wizards take you through the process and provide help information at each step.
  
- wsadmin scripts
  - Scripts to manage applications
  - More complicated than other methods
    - Requires skills in at least of one the supported scripting languages
    - Requires complete understanding of configuration
  
- WebSphere Rapid Deployment
  - Shortcut to installing, uninstalling and updating applications
  - Place J2EE application modules or artifacts into a configurable location on file system
  - Automatically detects added or changed parts of J2EE artifacts
  - Good for development and test phases

© Copyright IBM Corporation 2007

Figure 10-19. Managing applications

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

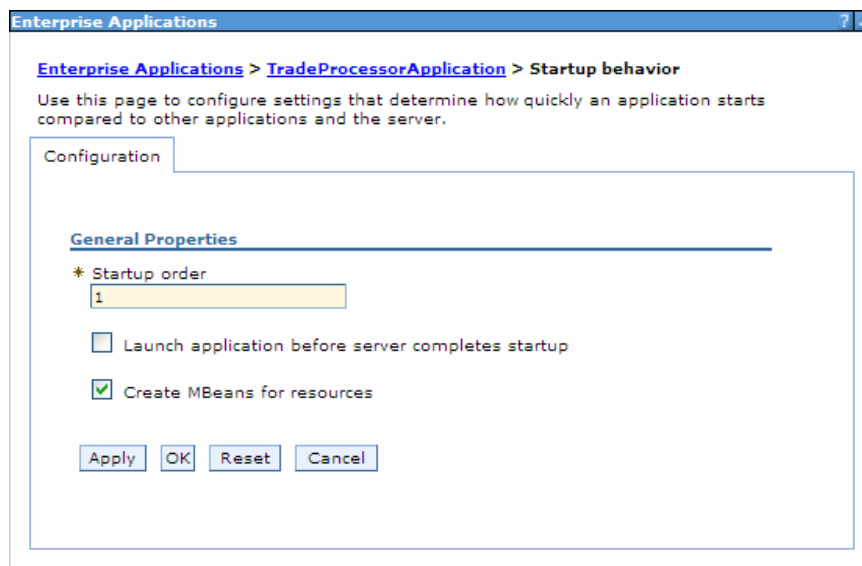
**Additional information —**

**Transition statement —** Next: Starting an application



## Starting an application

- Starting an application server starts applications mapped to the server.
- Order in which applications are started depends on the startup order.
  - Applications with lowest value are started first.
  - Applications with same value are started in no particular order.
- Can enable background application startup.



© Copyright IBM Corporation 2007

Figure 10-20. Starting an application

WA3616.3

### Notes:

Launch application before server completes startup.

- Specifies whether the application must initialize fully before the server starts.
- The default setting of false indicates that server startup will not complete until the application starts.
- A setting of true informs the server that the application might start on a background thread and thus server startup might continue without waiting for the application to fully start. Thus, the application might not be ready for use when the application server starts.

***Instructor notes:***

**Purpose** —

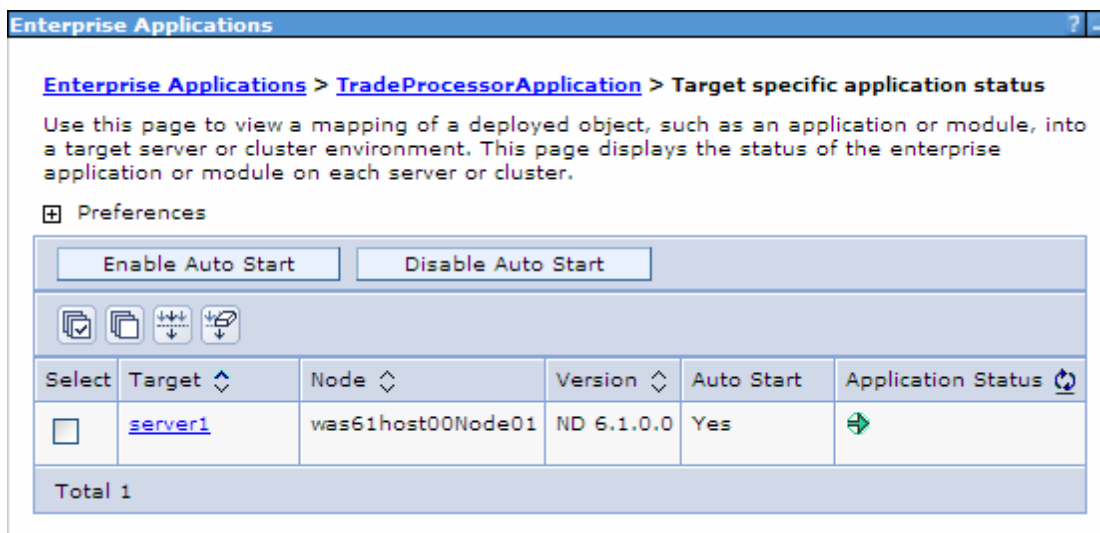
**Details** —

**Additional information** —

**Transition statement** — Next: Application auto start

## Application auto start

- Determines whether application starts automatically when the server starts.
- Default is to keep application started state as of last shutdown.



© Copyright IBM Corporation 2007

Figure 10-21. Application auto start

WA3616.3

### Notes:

***Instructor notes:***

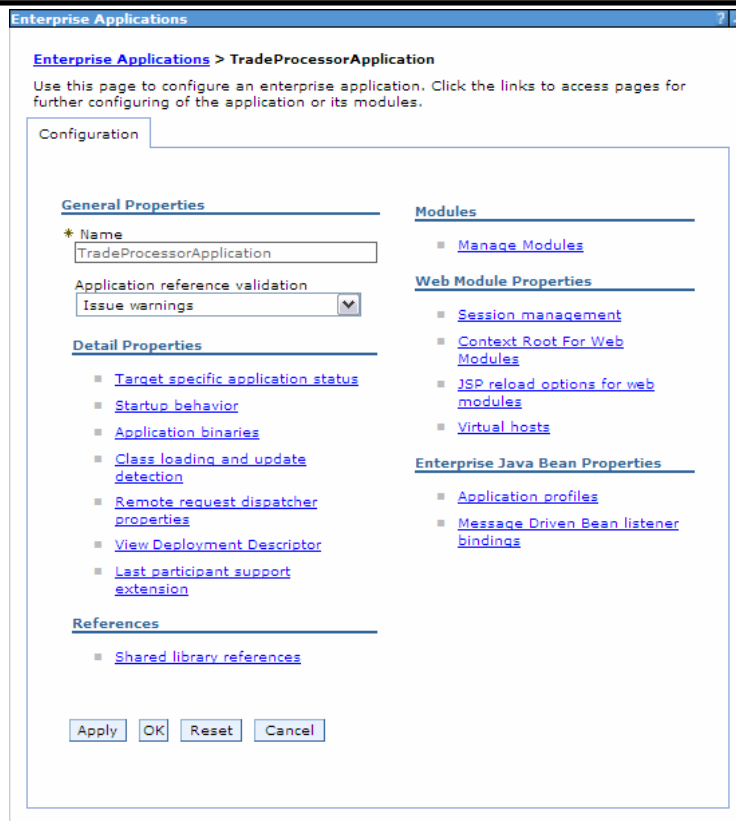
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Other application configuration options

## Other application configuration options



© Copyright IBM Corporation 2007

Figure 10-22. Other application configuration options

WA3616.3

### Notes:

All application configuration options can be tweaked after installation.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Checkpoint questions

## Checkpoint questions

---

1. Which of the following installation methods should only be used during Development and Test?
  - A. wsant
  - B. WebSphere Rapid Deployment (WRD)
  - C. wsadmin command line
  - D. wsadmin Jython script
  
2. Assuming equivalent functionality, which database driver type is generally recommended if available?
  - A. Type 1: JDBC-ODBC Bridge
  - B. Type 2: Native-API/partly Java driver
  - C. Type 3: Net-protocol/all-Java driver
  - D. Type 4: Native-protocol/all-Java driver
  
3. During an application server startup, the order in which enterprise applications start cannot be determined.
  - A. True
  - B. False

© Copyright IBM Corporation 2007

Figure 10-23. Checkpoint questions

WA3616.3

### Notes:

Write your answers here:

- 1.
- 2.
- 3.

**Instructor notes:**

Purpose —

Details —

## Checkpoint solutions

---

1. Which of the following installation methods should only be used during Development and Test?  
B. WebSphere Rapid Deployment (WRD)
2. Which database driver type is recommended if available?  
D. Type 4: Native-protocol/all-Java driver
3. During an application server startup, the order in which enterprise applications start cannot be determined.  
B. False

© Copyright IBM Corporation 2007

**Additional information** —

**Transition statement** — Next: Unit summary



## Unit summary

---

After completing this unit, you should be able to:

- Describe methods of installing enterprise applications in WebSphere Application Server
- Describe methods of installing applications using the administrative console
- Describe how to install applications using WebSphere Rapid Deployment
- Describe fine grain application update
- Describe enterprise application properties

© Copyright IBM Corporation 2007

Figure 10-24. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

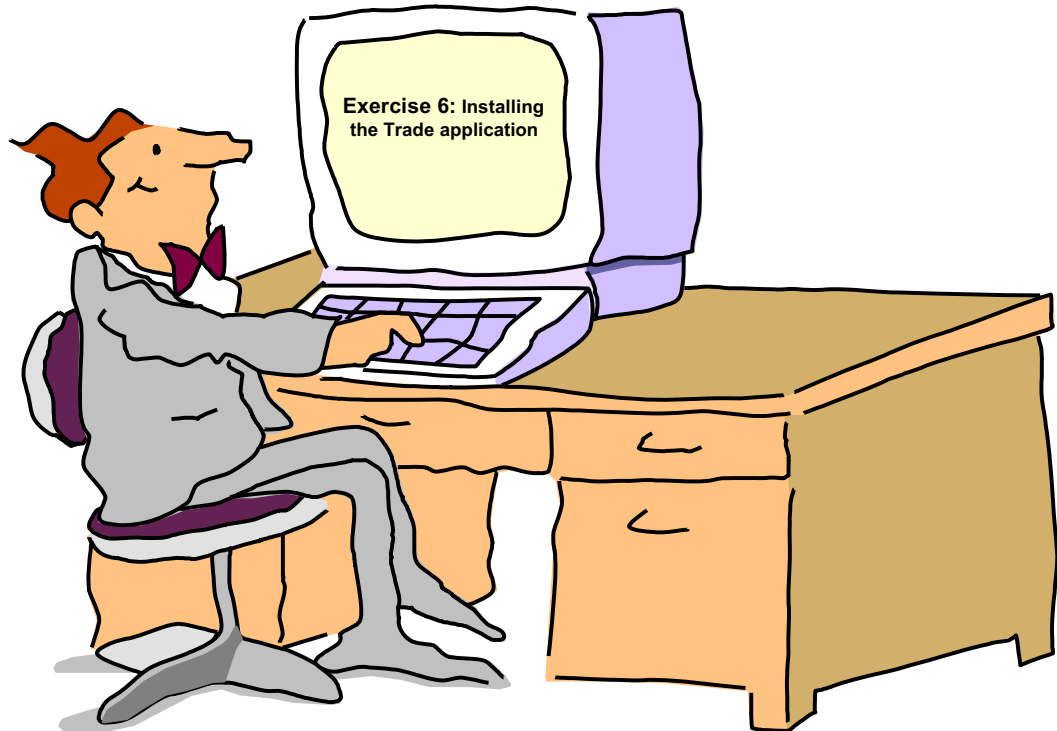
**Details —**

**Additional information —**

**Transition statement —** Next: Lab exercise

## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 10-25. Lab exercise

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

---

# Unit 11. Problem determination

## Estimated time

01:00

## What this unit is about

This unit describes a methodology for conducting problem determination. This unit also discusses the tools available to the administrator for investigating problems.

## What you should be able to do

After completing this unit, you should be able to:

- Describe problem determination issues
- Identify resources for problem determination
- Describe tools for troubleshooting
- Describe the location of various log files
- Use AST and Log Analyzer to examine the activity log
- Describe FFDC and the Collector tool
- Use HTTP logs for problem determination

## How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

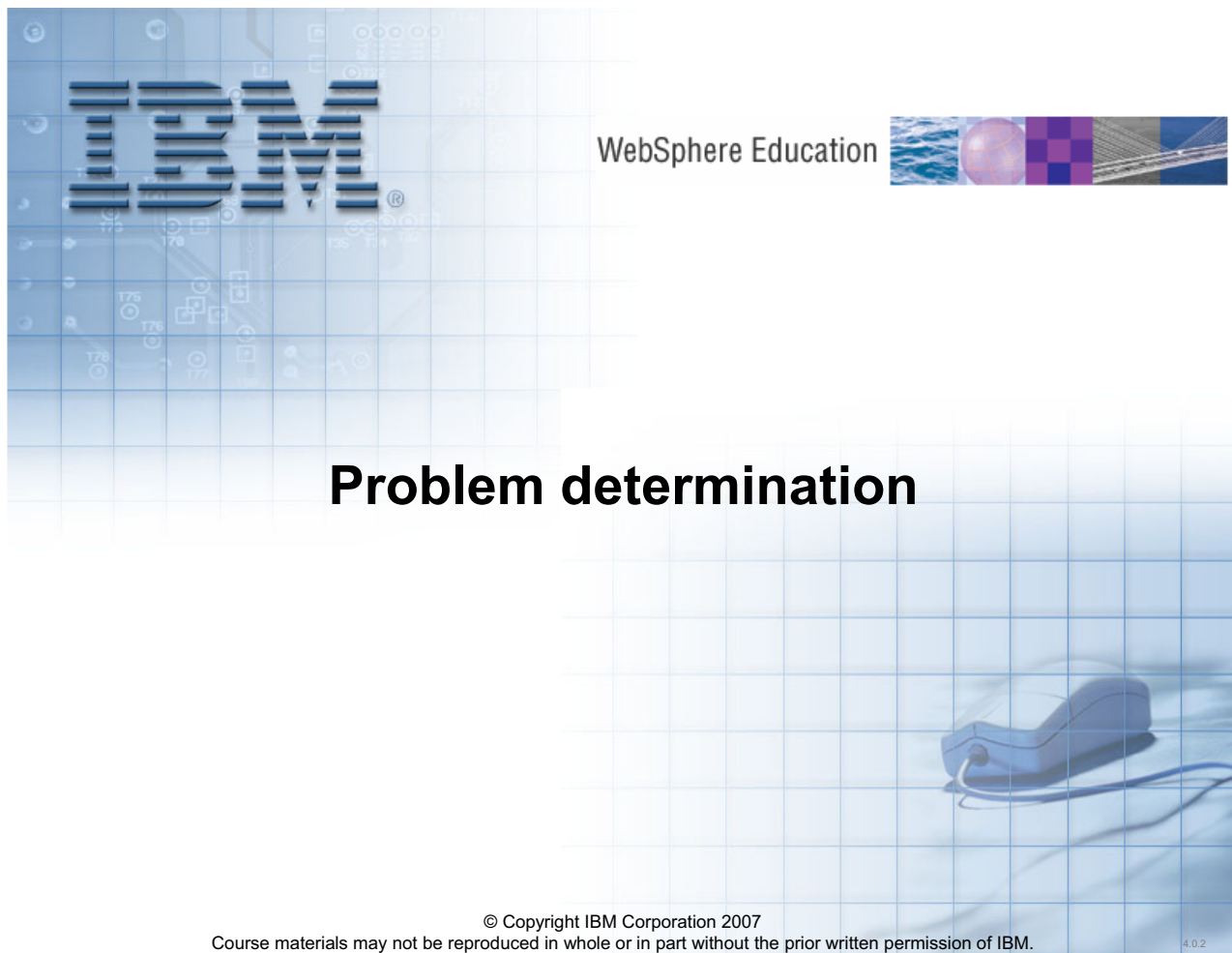


Figure 11-1. Problem determination

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe problem determination issues
- Identify resources for problem determination
- Describe tools for troubleshooting
- Describe the location of various log files
- Use AST and Log Analyzer to examine the activity log
- Describe FFDC and the Collector tool
- Use HTTP logs for problem determination

© Copyright IBM Corporation 2007

Figure 11-2. Unit objectives

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: What is the IBM Support Assistant?

## What is the IBM Support Assistant?

---

- The IBM Support Assistant (ISA) is a *free, stand-alone application* that you can install on any workstation.
- It can be enhanced by installing plug-in modules for the IBM products you use.
- Saves time searching product, support, and educational resources.
- If a Problem Management Report (PMR) needs to be opened, ISA helps with:
  - Gathering support information based on problem type
  - Creating and update the problem report
  - Tracking your electronic problem report
- Includes a support tool framework allowing for easy location and installation of useful product support tools.
- ISA can be downloaded from <http://www.ibm.com/software/support/isa/>

© Copyright IBM Corporation 2007

Figure 11-3. What is the IBM Support Assistant?

WA3616.3

### **Notes:**

Improves your ability to locate IBM support, development and educational information through a federated search interface (one search - multiple resources).

Provides quick access to the IBM Education Assistant and key product education road maps.

Simplifies access to IBM product home pages, product support pages, as well as product forums or newsgroups, through convenient links.

Saves you time submitting problems to IBM Support by collecting key information, then electronically creating a Problem Management Report (PMR) from within IBM Support Assistant.

Includes a support tool framework allowing for the easy installation of support tools associated with different IBM products.

The IBM Support Assistant provides a framework for IBM software products to deliver customized self-help information into the different tools within it. Customize your IBM

Support Assistant client by using the built-in Update capability to find and install new product features or support tools.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: IBM Support Assistant console

# IBM Support Assistant console

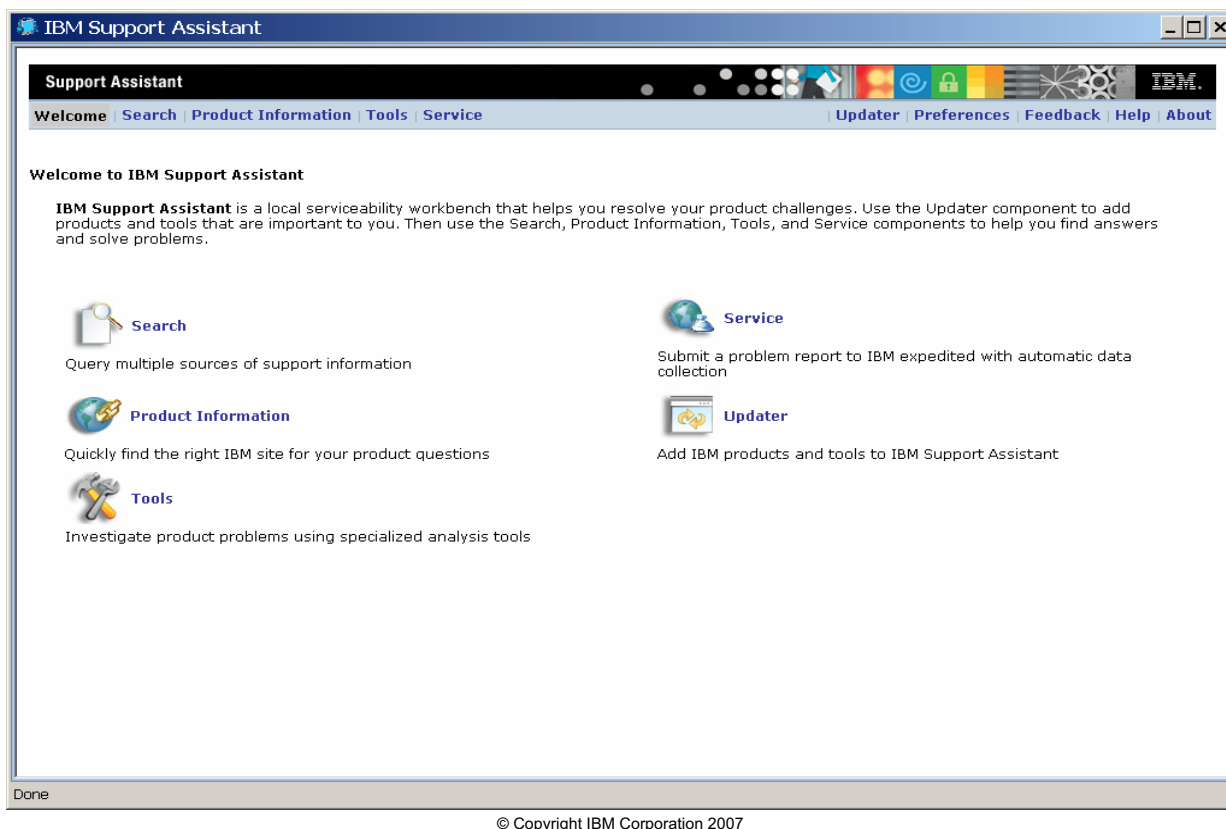


Figure 11-4. IBM Support Assistant console

WA3616.3

## Notes:

### Follow these steps to download and install ISA:

- Go to the <http://www.ibm.com/software/support/isa/>, choose IBM Support Assistant V3, and download the installation package for any platform where you will install IBM Support Assistant.
- You will need to log in using your IBM Web identity (same id as used for MySupport, developerWorks, and so on).
  - If you do not already have an IBM Web identity, you may complete the free registration process to obtain one.
- Download the compressed archive file(s).
- Uncompress the archive file in a temporary directory using a tool such as WinZip.
- The archive contains an installer program and the Installation and Troubleshooting Guide. Follow the directions in the Installation and Troubleshooting Guide to install IBM Support Assistant.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: IBM Support Assistant: Search component

---

## IBM Support Assistant: Search component

---

- Search component
  - Enables searching multiple information locations concurrently including:
    - Technical notes from the IBM software support Web site
    - Articles from IBM developerWorks
    - Postings from IBM newsgroups and forums
    - Google Web search
    - On-line InfoCenters
  - You can also narrow your search by selecting specific IBM software products and versions.

© Copyright IBM Corporation 2007

Figure 11-5. IBM Support Assistant: Search component

WA3616.3

### Notes:

#### Using Search Options

Search options are provided to help you narrow the scope of your search. The search location you choose determines what search options are available. For example, if you select Google Web Search, search options do not apply. The IBM developerWorks and IBM Newsgroups and Forums search locations let you narrow your search to specific product areas.

The IBM Software Support Documents search location offers the most options. You can select specific products and versions, and you can select specific types of documents.

Not all products have an online information center to search.

***Instructor notes:***

**Purpose** —

**Details** —

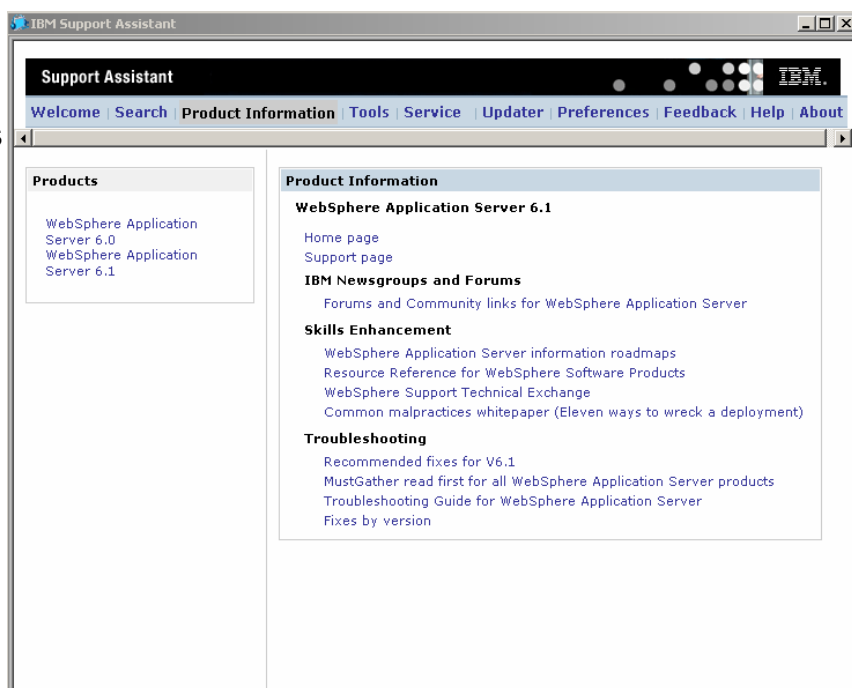
**Additional information** —

**Transition statement** — Next: IBM Support Assistant: Product information component



## IBM Support Assistant: Product information component

- Select a product to see product marketing, support, and forum resources in addition to skills enhancement (educational) and troubleshooting resources



© Copyright IBM Corporation 2007

Figure 11-6. IBM Support Assistant: Product information component

WA3616.3

### Notes:

#### Using Product Information

Helps you quickly find helpful IBM Web sites.

With product information links you can access a variety of support, troubleshooting, and educational options including traditional classes, Web-based training, CD-ROM training, certification, and so forth.

#### Selecting a Link

Select a link depending on what type of product information you are looking for.

Select the Product Home Page to find introductory information, with descriptions of features, benefits, and requirements. You can also find links to product literature, events, news, and success stories.

Select the Product Support Page to search IBM software support documents that pertain to your product. This page also provides links to support resources such as My Support, Passport Advantage, and support news flashes.

Select the Product Forums to see and contribute to product discussions.

Select the Skills Enhancements links to see educational related documents for your product.

Select the Troubleshooting links to see product specific troubleshooting information.

***Instructor notes:***

**Purpose —**

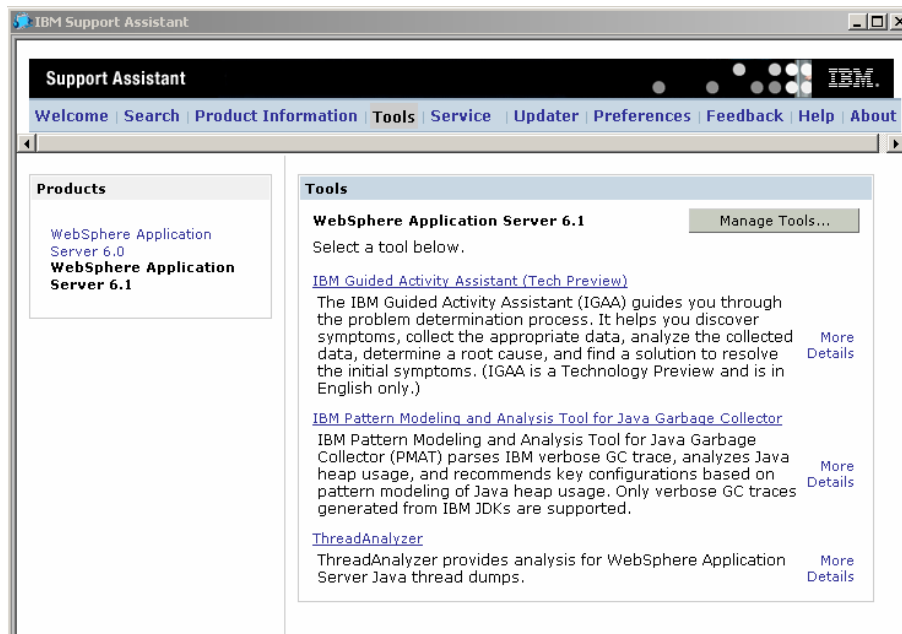
**Details —**

**Additional information —**

**Transition statement —** Next: IBM Support Assistant: Tools component

# IBM Support Assistant: Tools component

- Tools component
  - Provides support tools to identify and prevent problems



© Copyright IBM Corporation 2007

Figure 11-7. IBM Support Assistant: Tools component

WA3616.3

## Notes:

The Tools component is a way to provide support tools to you. These tools can be used diagnostically to identify problems you are having. Some tools may also be used preventively to stop a problem before it occurs.

Provides a way to use support related tools to identify problems or provide health checking tools to prevent problems. Tools are installed via the Updater component of IBM Support Assistant. Tools are launched in a new window by clicking their name.

***Instructor notes:***

**Purpose —**

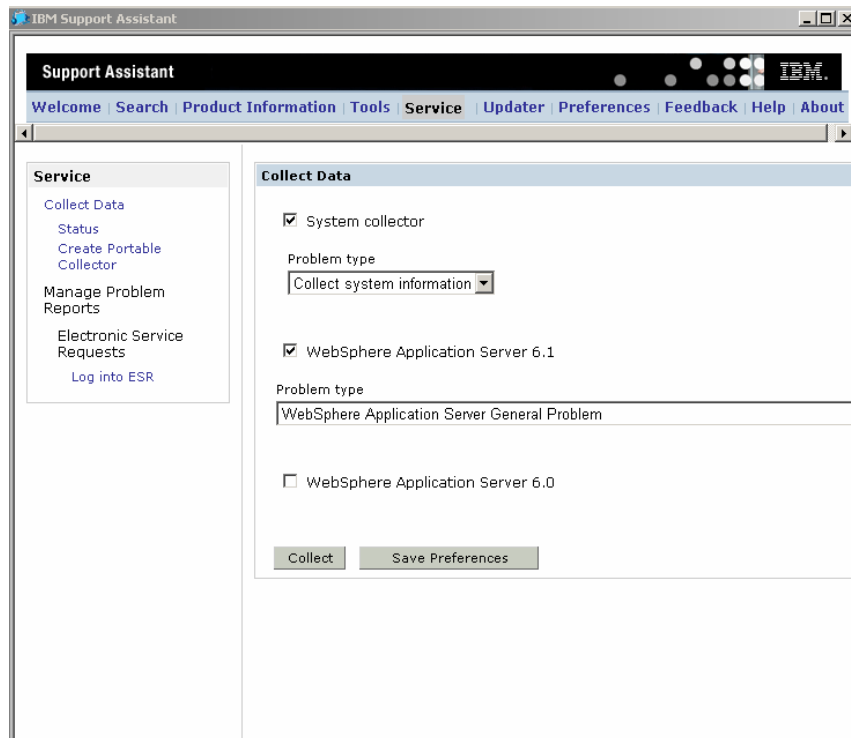
**Details —**

**Additional information —**

**Transition statement —** Next: IBM Support Assistant: Service component

# IBM Support Assistant: Service component

- Collect Data on
  - System information
  - Product configuration
- Log in and submit
  - PMRs
  - Collected data jar file
  - Other documents



© Copyright IBM Corporation 2007

Figure 11-8. IBM Support Assistant: Service component

WA3616.3

## Notes:

### Collect Data

Collecting data is an important part of submitting a problem report because it enables IBM to resolve your problem report more quickly.

When you click **Collect Data**, you will be asked to select the product you would like to collect data for. You will be prompted to select a problem type as well.

Depending on the product you select, you may be asked for other information. When you have selected a product and provided the required information, press **Collect**.

For most problem types, IBM Support Assistant will collect data for the product you selected and store it in a JAR file. For some problem types, the information collected may result in a report file or simple message to the user. You can collect data for more than one product at a time if desired.

### Manage Problem Reports

In order to manage problem reports, you need to log in. Once you are logged in, you can:

- Open a new problem report
- View existing problem reports
- Attach a file to a new or existing problem report.

To log in, you need to provide your IBM ID, password, IBM customer number, and country. The login process also verifies that you have an IBM service contract and that your ID is on the Authorized Callers list.

After you log in, the Service component shows you any problem reports you have already submitted. You can see these problem reports at any time by clicking **List Problem Reports**.

### Submit Problem Reports

Click **Submit Problem Report** if you would like to create a new problem report. To create a new report, you will need to answer a few basic questions:

- What product are you having a problem with?
- How severe is the problem?
- What kind of problem is it?
- What else can you tell IBM?
  - Were there any recent changes?
  - Did you make any attempts to fix the problem?
  - What is your operating system?
  - Any other helpful information?

You can attach a file to a new problem report. If you recently ran the Collector, the Collector file name will be visible next to a check box allowing you to send this Collector file to IBM automatically when you open the problem report. In addition, you can also attach a file from your file system by using the Attach File field.

Click **Submit** to send your problem report to IBM along with associated files.

You can also attach a file to an existing problem report. Click **List Problem Reports** and then click **Attach File**.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: IBM Support Assistant: Updater component



---

## IBM Support Assistant: Updater component

---

- Updater component
  - Lets you add IBM products to IBM Support Assistant
  - Adding a product provides more information to IBM Support Assistant
  - Here are examples of information that can be provided when you add a product:
    - **Search component:** The product name can be used as a search option. The product's on-line Information Center may also be used as a search target.
    - **Product information component:** Product support links and educational links become available.
    - **Service component:** Data collection for the product is enabled.
    - **Service component:** Problem reporting for the product is enabled.
    - **Tools component:** Support tools may be associated with the product.

© Copyright IBM Corporation 2007

Figure 11-9. IBM Support Assistant: Updater component

WA3616.3

### Notes:

#### When should you use the Updater Component?

The Updater component can be used to locate a more up-to-date version of the product information you currently have installed. If you would like to see more products, you can add them by using the updater to find, download, and install the required product information from the IBM Web site.

IBM Support Assistant product information is packaged as product plug-ins. You can add a product by installing a new product plug-in.

IBM Support Assistant pluggable tools are also installed by using the updater.

After you install new product information or a pluggable tool, you will need to restart IBM Support Assistant to make that information or tool available.

***Instructor notes:***

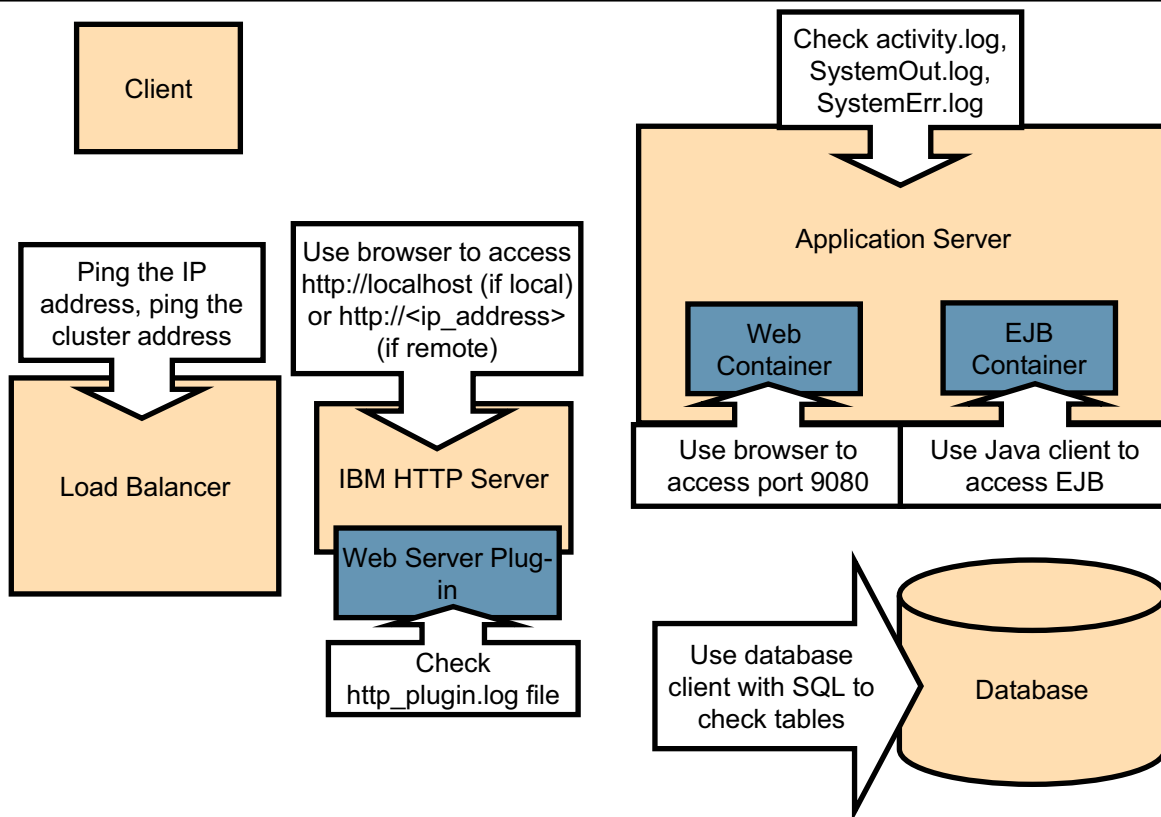
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Problem determination – Divide and conquer

## Problem determination – Divide and conquer



© Copyright IBM Corporation 2007

Figure 11-10. Problem determination – Divide and conquer

WA3616.3

### Notes:

To help divide and conquer, try to isolate the location of the problem. Test individual components and see which things succeed and which fail. That should help determine where the failure is isolated to.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Searching on the Support page

## Searching on the Support page

- Try using ISA
- Search on the exception or error message you are seeing
- Search on APAR or Fixpack
- Search for problem determination tools
- Search for *MustGather* documents

The screenshot shows the IBM WebSphere Application Server support page. At the top, there's a navigation bar with 'Home', 'Products', 'Services & industry solutions', 'Support & downloads', and 'My IBM'. Below this is a search bar and a 'Search' button. The main content area is titled 'WebSphere Application Server' and includes a 'Search Support (this product)' section with a search input field and a 'Search' button. There are also sections for 'Solve a problem', 'Download', and 'Learn'. The page also features a 'Flash' section with two messages, a 'Site Availability' section, and a 'Support feedback' section.

© Copyright IBM Corporation 2007

Figure 11-11. Searching on the Support page

WA3616.3

### Notes:

Choose keywords to research the problem in available knowledge sources:

- Error codes, exception names, and so on from the most promising symptoms
- If no explicit error, use high-level problem description
  - See list of problem categories from the IBM Support Web site

Research in IBM Support Assistant or on the WebSphere Application Server Support Web page

- Will automatically search product InfoCenter for error codes, and so on
- Support page address:  
<http://www.ibm.com/software/webservers/appserv/was/support/>

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## WebSphere Support page

---

- All the problem determination data is organized into a set of predefined problem categories or “components”
  - Currently forty-four categories for WebSphere Application Server: “100% CPU Usage”, “Admin Console”, “Classloader”, “Crash”, and so on
  - Also used to manage most support processes and other problem determination assets
- Library of technotes and other articles
  - Maintained by a specialized knowledge engineering team
  - Includes known problems, APARs, common questions, troubleshooting instructions for many specific problems, problem determination tools, and so on
- One *MustGather* document for each problem category (few exceptions)
  - Provides instructions on how to start troubleshooting that problem, and what information to provide to support if opening a PMR
- Extensive search facility

© Copyright IBM Corporation 2007

Figure 11-12. WebSphere Support page

WA3616.3

### Notes:

WebSphere Application Server support page address:

<http://www.ibm.com/software/webservers/appserv/was/support>

***Instructor notes:***

**Purpose —**

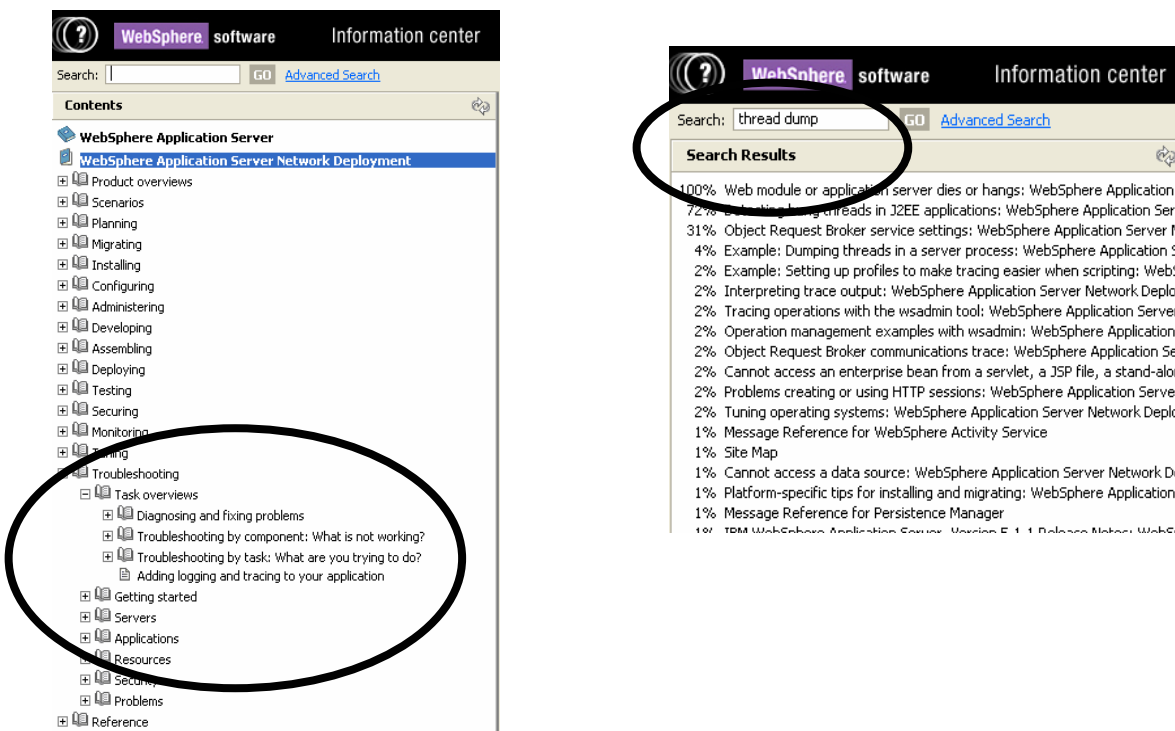
**Details —**

**Additional information —**

**Transition statement — -**



# Searching the Information center



© Copyright IBM Corporation 2007

Figure 11-13. Searching the Information center

WA3616.3

## Notes:

The WebSphere Application Server Information center includes a troubleshooting section and searching capabilities.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

# Product components and MustGather documents

- Use the collector tools to capture general information for each host machine

Main MustGather document page:  
<http://www.ibm.com/support/docview.wss?rs=180&uid=swg21145599>

Components	Collector Tool		
	Upgrade Policies		
	IBM Education Assistant		
100% CPU Usage		read	
Administrative Console (all non-scripting)	learn		read
Administrative Scripting Tools (for example: wsadmin or ANT)	learn		read
Application Client		read	
Application Server Toolkit (AST)			
Classloader	learn		read

For additional description of components you can go to the IBM support page (as shown above) and search for information about MustGather documents:

Component	Description
<b>100% CPU Usage</b>	Includes multiple product categories. 100% CPU usage due to a WebSphere Application Server process.
<b>Administrative Console (all non-scripting)</b>	Any problems associated with using the Administrative Console, not including scripting (wsadmin). Does not include problems that occur after changing a setting that causes a change in behavior.
<b>Administrative Scripting Tools (for example, wsadmin or ANT)</b>	Problems experienced performing Administration tasks using command line tools

© Copyright IBM Corporation 2007

Figure 11-14. Product components and MustGather documents

WA3616.3

## Notes:

Main MustGather document page

- <http://www.ibm.com/support/docview.wss?rs=180&uid=swg21145599>
- You can access specific documents or view all MustGather documents from this page.
- Use operating system-specific MustGather documents, when appropriate. Product components include symptoms like “Crash” that span product components.
- Updated frequently to help resolve problems more quickly.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## WebSphere Application Server logs

---

- JVM logs: created by redirecting the System.out and System.err streams of the JVM to independent log files.
  - One set of JVM logs for each application server and all of its applications located by default in the following directory:  
`<profile_root>/<profile_name>/logs/<server_name>`
- Process logs: contain two output streams (stdout and stderr) that are accessible to native code running in the process.
  - One set for each application server
- IBM service log: contains both the WebSphere Application Server messages that are written to the System.out stream and some special messages that contain extended service information that is normally not of interest, but can be important when analyzing problems.
- The HTTP server plug-in maintains a special log.

© Copyright IBM Corporation 2007

Figure 11-15. WebSphere Application Server logs

WA3616.3

### Notes:

#### Java virtual machine (JVM) Logs

The JVM logs are created by redirecting the System.out and System.err streams of the JVM to independent log files. WebSphere Application Server writes formatted messages to the System.out stream. In addition, applications and other code can write to these streams using the print() and println() methods defined by the streams.

In the case of a WebSphere Application Server Network Deployment configuration, JVM logs are also created for the deployment manager and each node agent because they also represent JVMs.

#### Process Logs

WebSphere Application Server processes contain two output streams that are accessible to native code running in the process. These streams are the stdout and stderr streams. Native code, including Java virtual machines (JVM), might write data to these process streams. In addition, JVM provided System.out and System.err streams can be configured to write their data to these streams also.

As with JVM logs, there is a set of process logs for each application server, since each JVM is an operating system process, and in the case of a WebSphere Application Server Network Deployment configuration, a set of process logs for the deployment manager and each node agent.

## **IBM Service Logs**

The IBM service log contains both the WebSphere Application Server messages that are written to the System.out stream and some special messages that contain extended service information that is normally not of interest, but can be important when analyzing problems. There is one service log for all WebSphere Application Server JVMs on a node, including all application servers. The IBM Service log is maintained in a binary format and requires a special tool to view. This viewer, the AST Log and Trace Analyzer, provides additional diagnostic capabilities. In addition, the binary format provides capabilities that are utilized by IBM support organizations.

The HTTP server plug-in log will be covered later in this presentation.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Log files: Types and locations

- The destination and names for the log files are configurable. The default location is:

```
<was_root>\profiles\<profile_name>\logs\<server_name>
```

- Log files:
  - **SystemOut.log** and **SystemErr.log** - Standard JVM output and error log
  - **startServer.log** and **stopServer.log**
    - Startup and shutdown of the Application Servers
  - **activity.log** - Events that show a history of activities
    - Use AST Log and Trace Analyzer to read output from this file
  - **trace.log** – Output from diagnostic trace
    - Destination and name are configurable
  - **http\_plugin.log** – Not in <was\_root>
    - Location: <plugin\_root>\logs\<webserver\_name>
  - **native\_stdout.log** and **native\_stderr.log**

© Copyright IBM Corporation 2007

Figure 11-16. Log files: Types and locations

WA3616.3

### Notes:

All WebSphere Application Server log files are under the <was\_root>\profiles\<profile\_name>\logs directory.

SystemOut.log and SystemErr.log are the default names for the JVM logs. They contain server as well as user program information (sent by: System.out.xxx code in the program).

The startServer.log and stopServer.log files can also be found under the <was\_root>\logs\<servername> directory. They contain information logged by the server as it starts-up and shuts-down

The activity.log file size can be set also using the admin console as IBM Service Logs. You can also disable activity.log.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Viewing logs

---

- JVM logs:
  - Click **Troubleshooting > Logs and Trace** in the administrative console navigation tree, or
  - Navigate to the `<profile_root>\<profile_name>\logs\<server_name>` directory on the machine where logs are stored, and open SystemOut.log or SystemErr.log in a text editor
  
- Process logs:
  - Open `<profile_root>\<profile_name>\logs\<server_name>\native_stderr.log`  
or  
`<profile_root>\<profile_name>\logs\native_stdout.log`  
in a text editor
  
- IBM service log:
  - Use the **AST Log and Trace Analyzer**

© Copyright IBM Corporation 2007

Figure 11-17. Viewing logs

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Configuring JVM logs

# Configuring JVM logs

- Troubleshooting → Logs and Trace
- > <server\_name> →
- JVM Logs
- Alternative:
- Servers -> Application servers
- <server\_name> →
- Logging and Tracing →
- JVM Logs
- System.out and System.err logs can be configured from this page
- Logs are self-managing
  - Can roll over based on time or file size
  - Number of historical log files is configurable
- To view logs through the console, use the Runtime tab

The screenshot shows the Configuration Console with the 'Runtime' tab selected. The 'General Properties' section is expanded to show the configuration for 'System.out'. The 'File Name' field is set to `${SERVER_LOG_ROOT}/SystemOut`. The 'File Formatting' dropdown is set to 'Basic (Compatible)'. The 'Log File Rotation' section has 'File Size' selected, with a 'Maximum Size' of 3 MB. The 'Time' option is unselected, with a 'Start Time' of 24 hours and a 'Repeat Time' of 24 hours. The 'Maximum Number of Historical Log Files' is set to 2. The 'Installed Application Output' section has both 'Show application print statements' and 'Format print statements' checked.

© Copyright IBM Corporation 2007

Figure 11-18. Configuring JVM logs

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

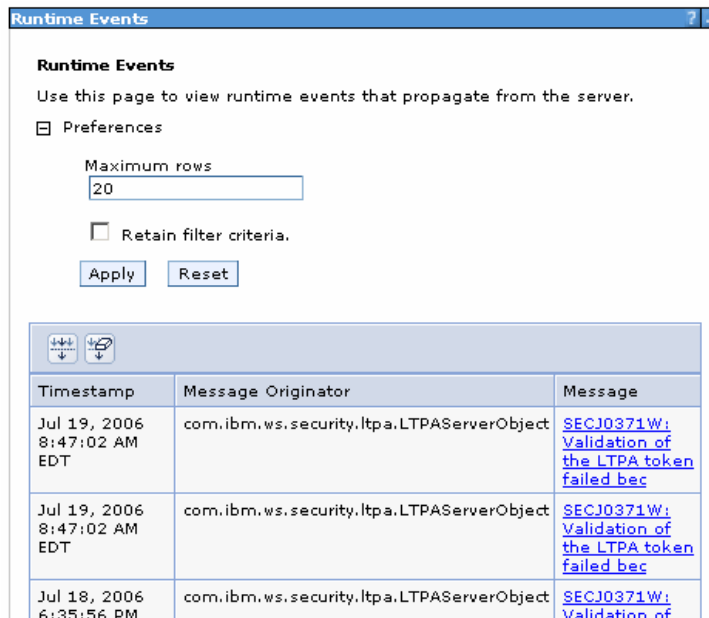
**Details —**

**Additional information —**

**Transition statement — -**

# Viewing runtime messages in the console (1 of 2)

- Runtime events are grouped by severity:  
Error, Warning, Information
- View: **Troubleshooting -> Runtime Messages -> Click:**
  - [Error](#)
  - [Warning](#)
  - [Information](#)



© Copyright IBM Corporation 2007

Figure 11-19. Viewing runtime messages in the console (1 of 2)

WA3616.3

## Notes:

While viewing runtime messages, first select the Error, Warning or Information category links (a count of zero means nothing is available). Then the details for the selected category are shown. Selecting one of these links gives you detail information (see next slide).

Note that you may have multiple pages of messages, the button on the bottom of the page will allow you to see them all.

***Instructor notes:***

**Purpose —**

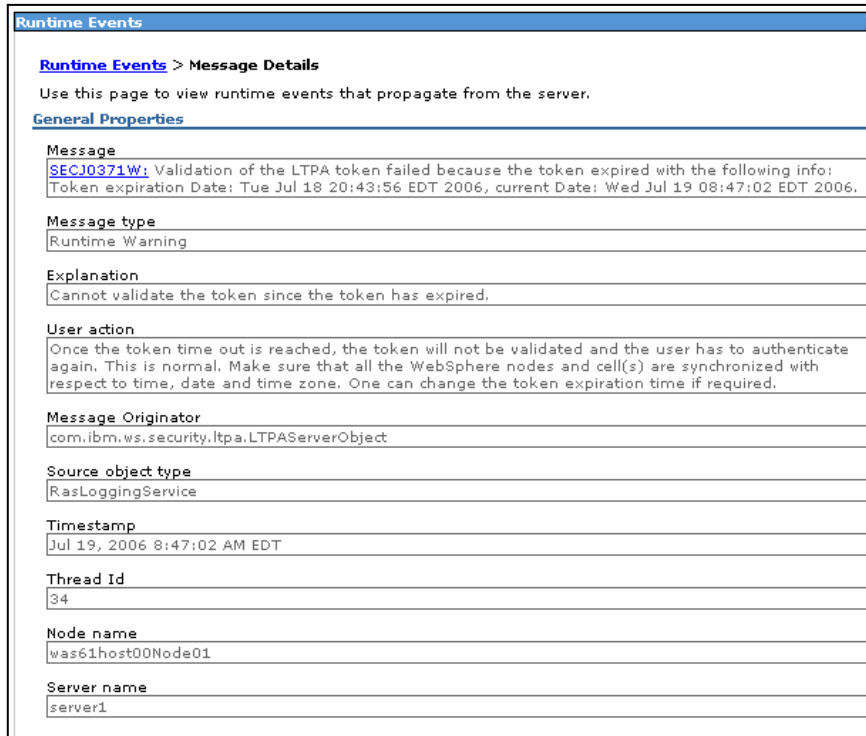
**Details —**

**Additional information —**

**Transition statement — -**

## Viewing runtime messages in the console (2 of 2)

- Runtime events details



© Copyright IBM Corporation 2007

Figure 11-20. Viewing runtime messages in the console (2 of 2)

WA3616.3

### Notes:

In WebSphere Application Server, most runtime messages are designed with improved message text. Additional components have messages now.

Information is displayed on the detail screen for the event for you to resolve the problem with user action.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## HTTP plug-in logs and tracing

---

- Click **Servers > Web Servers > *web\_server\_name* Plug-in Properties > Configuration tab > Plug-in logging** to edit fields
- Default location:  
`<plugin_root>/logs/<web_server_name>/http_plugin.log`
- Set the Log level to Trace to trace all the steps in the request process (caution: this produces a lot of output)

© Copyright IBM Corporation 2007

Figure 11-21. HTTP plug-in logs and tracing

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Embedded HTTP Server logs

- Administrative console panels for configuring embedded HTTP server logs (access and error)
- From main application server panel, click HTTP Error and NCSA Access Logging
- Access and error logs can be controlled separately
- When maximum file size is reached, oldest entries are pruned

The screenshot shows the Configuration tab for Embedded HTTP Server logs. It is divided into two sections: NCSA Access log and Error log.

**General Properties**

- Enable service at server startup

**NCSA Access log**

- Enable access logging
- \* Access log file path:
- \* Access log maximum size:  MB
- \* NCSA access log format:

**Error log**

- Enable error logging
- \* Error log file path:
- \* Error log maximum size:  MB
- \* Error log level:

© Copyright IBM Corporation 2007

Figure 11-22. Embedded HTTP Server logs

WA3616.3

### Notes:

To enable the access or error log, you must check the “Enable service at server startup” check box, and also the check box for the specific log you want to enable. Notice there is no runtime tab for these logs. Logging will only begin once you have saved these changes to your configuration and restarted the application server.

Click **Servers > Web Servers > web\_server\_name Plug-in Properties > Configuration tab > Plug-in logging** to edit fields

Default location:

*plugins\_root/logs/web\_server\_name/http\_plugin.log*

Set the Log level to Trace to trace all the steps in the request process (caution: this produces a lot of output)

***Instructor notes:***

**Purpose —**

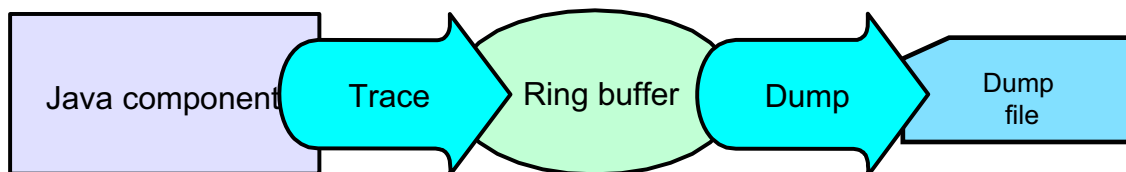
**Details —**

**Additional information —**

**Transition statement — -**

## Traces

- Trace files show the time and sequence of methods called by WebSphere Application Server base classes, and you can use these files to pinpoint the failure
- Trace can be started
  - While server is running using Runtime Diagnostic Trace
  - When server is started using Configuration Diagnostic Trace
- Trace output can be directed to:
  - Memory ring buffer - dumped after trace stops
  - File
- Trace has a significant impact on performance
  - Enable temporarily for problem determination
  - Trace to file is slower than trace to memory ring buffer Runtime tab



© Copyright IBM Corporation 2007

Figure 11-23. Traces

WA3616.3

### Notes:

Trace output allows administrators to examine processes in the application server and diagnose various issues.

On an application server, trace output can be directed either to a file or to an in-memory circular buffer. If trace output is directed to the in-memory circular buffer, it must be dumped to a file before it can be viewed.

On an application client or stand-alone process, trace output can be directed either to a file or to the process console window.

In all cases, trace output is generated as plain text in either basic, advanced or log analyzer format as specified by the user. The basic and advanced formats for trace output are similar to the basic and advanced formats that are available for the JVM message logs.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Enabling trace

- Enabled by default (no trace output)
- **Troubleshooting → Logs and Trace → <server\_name> → Diagnostic Trace**
- Enable Log check box enables tracing
- Configurable output
  - Memory buffer or file
- The location for entering the trace string is on the Log Detail Level page
- **Note:** Change Log Detail Level to get trace output

Configuration **Runtime**

**General Properties**

Enable Log

**Trace Output**

Memory Buffer

\* Maximum Buffer Size  
8 thousand entries

File

\* Maximum File Size  
20 MB

\* Maximum Number of Historical Files  
1

\* File Name  
{\$SERVER\_LOG\_ROOT}/trace.log

Trace Output Format  
Basic (Compatible)

Apply OK Reset Cancel

© Copyright IBM Corporation 2007

Figure 11-24. Enabling trace

WA3616.3

### Notes:

The Diagnostic Trace Service box looks mostly the same as it did in previous versions. The **Configuration** and **Runtime** tabs behave as they always have, with **Configuration** affecting the configuration repository and taking effect at the next startup, while **Runtime** takes effect immediately but is only optionally persisted to the server configuration.

The **Enable Log** check box is enabled by default. Because the Log Detail Level is set to `*=info` there is no trace output.

The major change on this panel is the absence of a space to enter the trace string. Trace strings have been moved to a separate panel (Log Detail Level).

View and dump available in the **Runtime** tab of diagnostic trace. Application Server Toolkit (AST) can be used to analyze trace output, but you may prefer to use your favorite editor; advanced users may want to use a tool like ProTrace. Before you can view or dump trace you need to specify log detail level



***Instructor notes:***

**Purpose —**

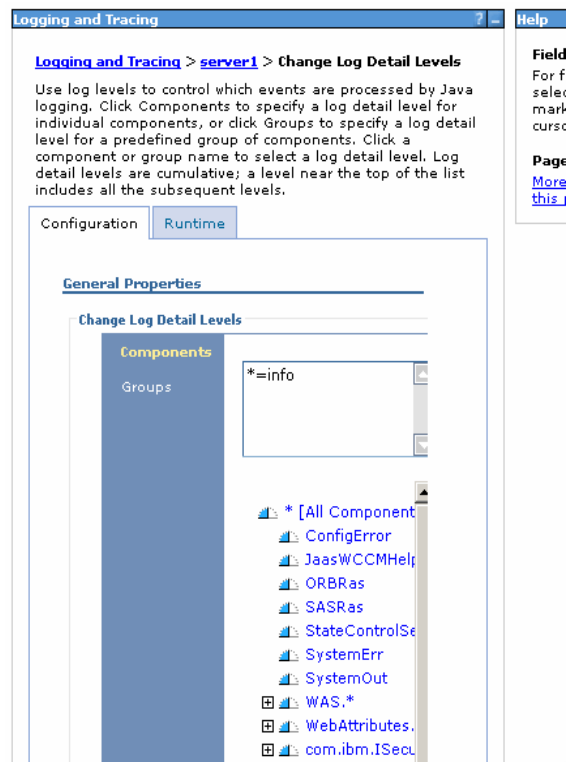
**Details —**

**Additional information —**

**Transition statement — -**

## Setting the log detail level

- **Logging and Tracing** → **<server>** → **Change Log Detail Level**
- Log detail level affects tracing and regular logging
  - **Setting levels below info reduces the amount of data in logs**
  - **\*=off disables logging altogether**
- Trace levels (**fine, finer, finest**) do not appear in the trace file unless logging is enabled
- Log string can be typed in or set using the graphical menu
- Default is \*=info
- User-created applications can be instrumented too, and be included in the trace output



© Copyright IBM Corporation 2007

Figure 11-25. Setting the log detail level

WA3616.3

### Notes:

Log Levels control which events are processed by Java logging.

WebSphere Application Server controls the levels of all loggers in the system. The level value is set from configuration data when the logger is created and can be changed at run time from the administrative console.

Note: Trace information, which are events at levels Fine, Finer and Finest, can only be written to the trace log. Therefore, if you do not enable diagnostic trace, setting the log detail level to Fine, Finer or Finest will not have an effect on the data that is logged.

Log String Syntax: <component / group> = <log level>

### Examples

```
com.ibm.ws.classloader.ClassGraph=finest
```

Enables finest trace level for com.ibm.ws.classloader.ClassGraph

```
EJBContainer=fine
```

Enables least verbose trace level for all components in the EJBContainer group  
com.ibm.ws.classloader.\*=finer

Enables detailed trace for all classes in the com.ibm.ws.classloader package

\*=info

Sets the log level for all components to info (default – no trace output)

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Reading a log or trace file

- Timestamps give good clues:
  - Time stamps are real machine time values
  - Good when comparing traces from different processes
- Look for exceptions (search for exception from top)
  - Events prior to exception are probable causes
  - Events after exception are recovery attempts
- Often useful to follow a single thread
- Log and trace file format:

<	Entry to a method (debug)
<	Exit a method (debug)
A	Audit
W	Warning
X	Error
E	Event (debug)
D	Detail (debug)
T	Terminate (exits process)
F	Fatal (exits process)
I	Information
O	Program output
C	Configuration

[01/30/05 9:51:15:081 GMT] 3c07adad PMImpl A PMON0001A:PMI enabled

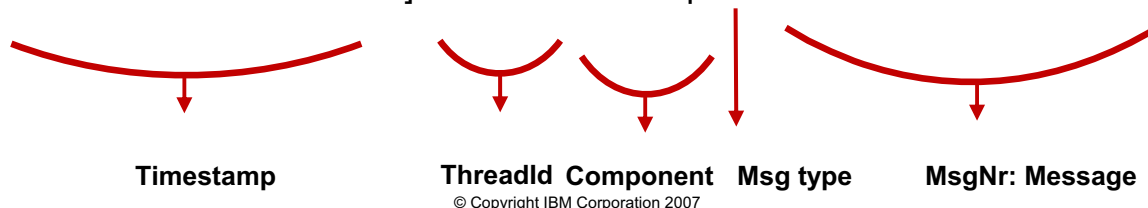


Figure 11-26. Reading a log or trace file

WA3616.3

### Notes:

The following list defines the sections of a trace entry:

- TimeStamp
 

The timestamp is formatted using the locale of the process where it is formatted. It includes a fully qualified date (YYMMDD), 24 hour time with millisecond precision and the time zone.
- ThreadId
 

An 8 character hexadecimal value generated from the hash code of the thread that issued the trace event.
- ThreadName
 

The name of the Java thread that issued the message or trace event.
- ShortName

The abbreviated name of the logging component that issued the trace event. This is typically the class name for WebSphere Application Server internal components, but may be some other identifier for user applications.

- LongName

The full name of the logging component that issued the trace event. This is typically the fully qualified class name for WebSphere Application Server internal components, but may be some other identifier for user applications.

- EventType

A one character field that indicates the type of the trace event. Trace types are in lower case

- ClassName

The class that issued the message or trace event

- MethodName

The method that issued the message or trace event

- Organization

The organization that owns the application that issued the message or trace event

- Product

The product that issued the message or trace event

- Component

The component within the product that issued the message or trace event

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Generating an IBM Heap Dump

---

- IBM Heap Dump is created in `<profile_root>` directory when an `OutOfMemoryError` exception is thrown
- Can be disabled with an environment variable
  - `IBM:HEAPDUMP_OUTOFMEMORY=false`
- Enable Heap Dump:
  - Environment variable: `IBM_HEAPDUMP=true`
  - Enables heap dump
  - Heap dump is in `.phd` (Portable Heap Dump) format
    - `.phd` is a more compact format used by some of the newer tools
- `IBM_JAVA_HEAPDUMP_TEXT=true`
  - Formats heap dump as text (classic heap dump format)
- `SIGQUIT` (kill -3) generates Heap Dump
  - Can also be triggered using `wsadmin`

© Copyright IBM Corporation 2007

Figure 11-27. Generating an IBM Heap Dump

WA3616.3

### Notes:

IBM Support Assistant includes a tool for inspecting Java memory dumps, called the Memory Dump Diagnostic Tool for Java. You can launch it, and any diagnostic plug-ins that you might download through the “Updater” component from the **Tools** tab. This tool can analyze several Java heap dump formats, including the standard dump formats from both the IBM and Sun Java runtime environments, and also z/OS SVC dumps. It displays and analyzes the data structures in the heap and their relationships, helping you identify the structures that are most likely responsible for memory leakage.

Using `wsadmin` to generate the dump:

- `set jvm [$AdminControl completeObjectName type=JVM,process=<server_name>,*]`
- `$AdminControl invoke $jvm dumpThreads`



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Hung thread detection

---

- Hung threads can be hard to diagnose. They are often not noticed until enough threads have hung to cause a user problem.
- The ThreadMonitor architecture monitors Web container, ORB, and async bean thread pools.
  - Enabled by default
- No action taken to kill the thread. Only a notification mechanism.
- When a thread is suspected to be hung, notification is sent three ways:
  - JMX notification for JMX listeners
  - ThreadPool metric for PMI clients
  - Message written to SystemOut.log:

```
[7/15/04 15:03:11:502 EDT] 3c3b4e37 ThreadMonitor W WSVR0605W: Thread
"Servlet.Engine.Transports : 0" (37c18e37) has been active for 68,839
milliseconds and may be hung. There are 1 threads in total in the server
that may be hung.
```

© Copyright IBM Corporation 2007

---

Figure 11-28. Hung thread detection

WA3616.3

### Notes:

Application threads can hang for a number of reasons, including infinite loops or deadlocks. There is a component known as the ThreadMonitor that monitors the Web Container, ORB, and Async Bean thread pools for hung threads.

The thread monitor doesn't try to deal with the hung threads, it just issues notifications, so that the administrator or developer can deal with the issues.

When a hung thread is detected, three notifications are sent: a JMX notification for JMX listeners, PMI Thread Pool data is updated for tools like the Tivoli Performance Viewer, and a message is written to the SystemOut log.

When the thread pool gives work to a thread, it notifies the thread monitor. The thread monitor notes thread ID and timestamp

Thread monitor compares active threads to timestamps. Threads that are active longer than the time limit are marked "potentially hung." The performance impact of this is minimal (< 1%)

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Connection leak diagnostics

---

- Poorly written applications often do not properly release database connections.
  - Forget to call `connection.close()`.
  - Most often in the exception case.
  - Connections should be closed in a `finally{} block`.
- Orphaned connections only return to the pool after timeout.
  - Can cause a backup of new connections waiting for old connections to time out.
  - New connections that have waited too long throw a `connectionWaitTimeoutException`.
- Connection manager is instrumented to print stack traces when a `connectionWaitTimeoutException` occurs.

© Copyright IBM Corporation 2007

Figure 11-29. Connection leak diagnostics

WA3616.3

### **Notes:**

Applications can suffer from performance problems and even appear to “hang” if they do not close their connections properly. This is most often caused by developers not properly using the `connection.close()` method. To ensure that connections will be closed properly, they should be closed in a `finally{} block`.

WebSphere is smart enough to eventually time out orphaned connections and return them to the pool, but for an application that makes frequent use of database connections, this might not be enough. New connections can get queued up waiting for the database while old connections are waiting to be timed out. This can bring the application grinding to a halt, and you can see `connectionWaitExceptions`.

Connection manager has lower performance impact than connection manager tracing. When activated, it enables a connection manager wrapper that holds the stack trace of all `getConnection()` calls in a throwable object. When an exception is thrown due to waiting on a full connection pool, print stack traces of all open connections. This enables you to significantly narrow your search area when you look at the application's source code to try and find the responsible code.

Connection leaks have traditionally been hard to diagnose because the error messages do not usually provide specific enough information about the source of the problem. Usually a source code review is needed to find points in the code where connections are not properly closed. The connection manager makes this task much easier.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

---

## Checking version levels and applying APARs

---

- The **versionInfo** command generates a report from data extracted from XML files in the properties/version folder. The report includes a list of changed components and installed or uninstalled maintenance packages.
- APAR: Authorized Program Analysis Report; tracks software defects reported by customers.
- Download recommended fixes, and fixes by version from the WebSphere Support page.
- **Fixpack installer**: standard tool for installing fixes:
  - [You can find more information on this in the Information Center.](#)

© Copyright IBM Corporation 2007

Figure 11-30. Checking version levels and applying APARs

WA3616.3

### Notes:

You will use the versionInfo command in the lab exercises; the versionInfo command is also documented in the WebSphere Application Server Information Centers.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**



## Install problems

---

- Log files for installation and uninstallation for all packages can be found under the `<install_root>\logs\` directory
- If the logs directory can not be accessed during installation, the logs are saved under `<user home>\<productID>logs`
  - Application server: `<user home>\waslogs`
  - IHS: `<user home>\ihlogs`
  - Plugin: `<user home>\plglogs`
  - Client: `<user home>\cltlogs`
- If there is a JDK related problem and if you can not find the logs look under `<OS temp directory>\niflogs`
  - `/tmp/niflogs` on UNIX and `%TEMP%/niflogs` on Windows
- If you see “*A suitable JVM could not be found*”, the following command will help diagnose the problem by catching the ISMP launcher log:
  - `install.exe -is:log catchinstallerlogs.txt`

© Copyright IBM Corporation 2007

Figure 11-31. Install problems

WA3616.3

### Notes:

Log files contain information about installing, configuring and updating WebSphere Application Server V6.1 components. The log file will initially be created in the user's current “temp” directory and then copied into the `<Install Home>/logs/install` directory. If the install is terminated before the file is copied, then it will remain in the user's “temp” directory.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Install exit codes

---

- All V6.1 installers, including IHS and plug-in installers use a common logging strategy, including exit codes.
- Linux and UNIX Platforms echo “\$?” Command after install completion will display.
  - 0 – Success, 1 – Failed, 2 – Partial Success
- Exit codes are not set on Windows platforms due to an Install Shield limitation.
  - Consult [log.txt](#) for information on success or failure
    - INSTCONFSUCCESS implies successful install and profile creation.
    - INSTCONFFAILED implies a failed install.
    - INSTCONFPARTIALSUCCESS usually implies a successful install, but some configuration action failed.

© Copyright IBM Corporation 2007

Figure 11-32. Install exit codes

WA3616.3

### Notes:

All of the installation packages in V6.1 will follow a common logging and tracing strategy, meaning they use a consistent location for log and trace files, and a consistent format for those files.

The installers set exit codes to indicate success or failure on all Linux and UNIX platforms. A user or program can check the exit code after the install has completed. “Partial success” means that some portion of the install failed but the installation is still at least partially usable and the failure may be recoverable to make the installation completely usable. You will need to consult the log, to determine the exact failure and what, if any, corrective actions can be taken.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Installation log files (1 of 2)

## Installation log files (1 of 2)

Group	Log Files (default)	Description
Application Server installation logs	<Install_root>/logs/install/ <b>log.txt</b>	This is the primary ISMP install log. Look for one of the following messages: <ul style="list-style-type: none"> <li>● <b>INSTCONFSUCCESS</b> – All configuration actions executed successfully</li> <li>● <b>INSTCONFPARTIALSUCCESS</b> – Implies some non-fatal configuration actions failed</li> <li>● <b>INSTCONFFAILED</b> – Indicates total failure,</li> </ul>
	<install_root>/logs/install/ <b>installconfig.log.gz</b>	XML-based file contain Logs the activities of ANT configuration scripts that run at the end of the installation procedure

© Copyright IBM Corporation 2007

Figure 11-33. Installation log files (1 of 2)

WA3616.3

### Notes:

This slide can be used reference; it lists the different log files that can contain information obtained from the WebSphere Application Server installation process. These logs should be used during problem determination of an unsuccessful install.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Installation log files (2 of 2)

## Installation log files (2 of 2)

Group	Log Files (default)	Description
Profile Creation	<i>&lt;Install_root&gt;/logs/manageprofiles/ pmt.log, wasprofile_create_&lt;profile&gt;.log ProfileName/component.logs</i>	Problems when launching the PMT and creating profiles – used by the installer and also when creating profiles outside the installation process
Application Deployment	<i>&lt;Install_root&gt;/logs/wasprofile/&lt;profile&gt; &lt;name&gt;_deploy.log, &lt;name&gt;_config.log</i>	Deployment and configuration logs for each of the enterprise applications installed by the WebSphere installer (Admin console, samples, etc)
Default Server	<i>&lt;Install_root&gt;/logs/wasprofile dir, logs for servers</i>	Logs for creating default server (server1), Service integration bus, and others.

© Copyright IBM Corporation 2007

Figure 11-34. Installation log files (2 of 2)

WA3616.3

### Notes:

This slide is also for reference, listing the different log files that can contain information obtained during profile creation process. In V6.1 all the profile creation logs are also saved under the *<install\_root>/logs* directory, which will help resolve profile creation problems, and remain in the *install\_root* even after profile has been deleted.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Problem determination tool availability



---

## Problem determination tool availability

---

- Some tools are integrated into the WebSphere administrative console.
  - [Tivoli Performance Viewer](#), [Configuration validator](#), [Classloader viewer](#), and so on.
- Some tools are separate programs shipped with WebSphere Application Server itself, or ship as separate products in their own right.
  - [Collector](#), [Application Server Toolkit \(AST\)](#), and so on.
- IBM Support Assistant (ISA):
  - Serves as a central point from which many tools can be found and even executed directly inside ISA.
  - The collection of tools released through ISA will grow steadily over the next year.
- In the meantime, until most tools are available through ISA, many tools can be found and downloaded individually from the Web.
  - [Search the Technotes on the IBM Support Web site \(or in ISA\)](#), which contains references to individual tools.

© Copyright IBM Corporation 2007

Figure 11-35. Problem determination tool availability

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Problem determination tools (1 of 2)

---

## Problem determination tools (1 of 2)

---

- IBM Guided Activity Assistant (IGAA)
  - Provides step-by-step guidance to perform various problem determination tasks
  - Available as a tool in ISA
- Collector
  - Facilitates gathering the log files and other artifacts needed for analysis of a problem, and sending them to IBM Support
  - Available as a standalone program shipped with WebSphere Application Server, or through ISA
- Tivoli Performance Viewer and Performance Advisor
  - Captures and analyzes statistical performance data from a running WebSphere Application Server system
  - Integrated into the WebSphere administrative console
- Configuration validator
  - Checks for errors in a WebSphere Application Server configuration
  - Integrated into the WebSphere administrative console
- Application Server Toolkit (AST) Log/Trace Analyzer
  - Analyzes and correlates logs from multiple servers
  - Available as a standalone program distributed with WebSphere Application Server, or download from the IBM Support Web site

© Copyright IBM Corporation 2007

Figure 11-36. Problem determination tools (1 of 2)

WA3616.3

### **Notes:**

### **hNotes:**

The Tivoli Performance Viewer will be described in detail later in the course.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Problem determination tools (2 of 2)

---

## Problem determination tools (2 of 2)

---

- Classloader Viewer
  - Diagnoses problems related to loading of classes in applications
  - Integrated into the WebSphere administrative console (V6.1), or available as a standalone download/install (V6)
- Memory Dump Diagnostic Tool for Java (MDD4J)
  - Analyzes heap dumps to diagnose out-of-memory conditions and leaks
  - Available in ISA
- Pattern Modeling and Analysis Tool for Java Garbage Collector (PMAT)
  - Analyzes a JVM verboseGC log to diagnose out-of-memory conditions
  - Available in ISA or standalone download
- Thread Analyzer
  - Analyzes a JVM thread dump to diagnose hangs, bottlenecks, etc.
  - Available as a standalone download
- DumpNameSpace
  - Dumps the contents of the JNDI name space to diagnose application configuration and startup problems
  - Available as a standalone download
- And more

© Copyright IBM Corporation 2007

Figure 11-37. Problem determination tools (2 of 2)

WA3616.3

### **Notes:**

This topic is not intended to be an exhaustive discussion about problem determination tools, but only introduces some of the common ones.

High-end tools, such as IBM Tivoli Composite Application Manager for WebSphere or other 3rd party products, are also useful.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

---

## IBM Guided Activity Assistant

---

- Provides step-by-step guidance to perform various problem determination tasks
  - Instructs user at each step, then asks a question and takes the user to the appropriate next step
  - Can invoke other tools hosted in ISA semi-automatically
- Hosted on IBM Support Assistant
- Currently, only a few problem topics are available, but the set of topics will grow over time; keep checking back

© Copyright IBM Corporation 2007

Figure 11-38. IBM Guided Activity Assistant

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Collector tools



## Collector tools

---

- Collect information about the WebSphere Application Server installation and configuration.
- Two versions are now available:
  - The “default” standalone collector collects almost all the logs and configuration files found on the system, without distinction.
  - A new flexible collector tool is integrated into ISA.
- In either case, the result is a JAR file that contains a lot of information needed by IBM Support to diagnose the problem.
  - JAR file is sent to IBM Support.
  - Only needs to be run upon IBM Support’s request.

© Copyright IBM Corporation 2007

Figure 11-39. Collector tools

WA3616.3

### Notes:

Default collector tool:

- Implemented as a shell script in <profile\_root>\bin\collector.bat or collector.sh
- You must run the collector from inside this <profile\_root>\bin directory in order to properly identify the target server

Collector in ISA:

- Choose from a menu of problem types in ISA, which controls which files to collect for each problem
- Can include an interactive session: enable trace, reproduce the problem, collect trace, and so on, all under the control of the collector tool
- Provides for local collection (on the ISA machine itself) and creating a script for remote collection on a remote machine

***Instructor notes:***

**Purpose —**

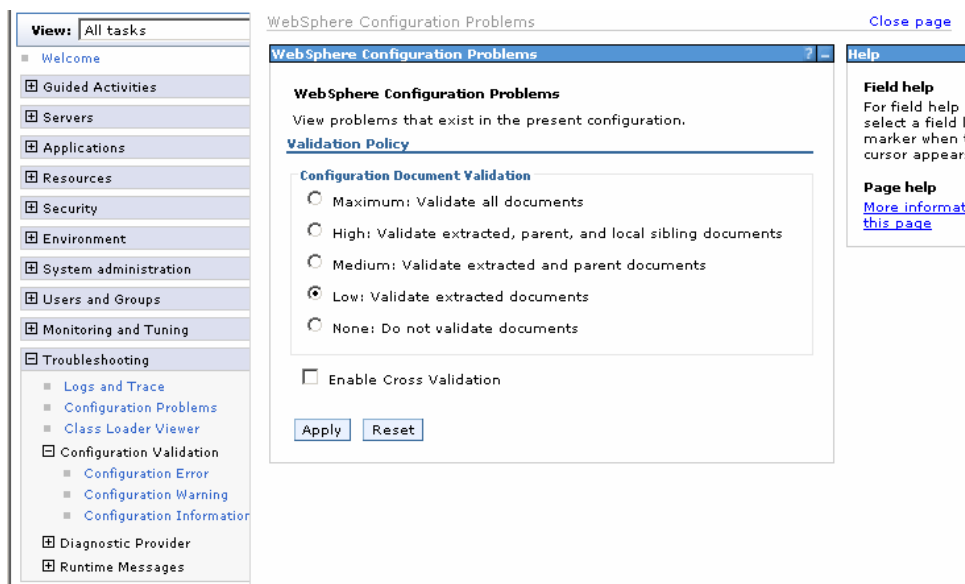
**Details —**

**Additional information —**

**Transition statement —** Next: Configuration validator

## Configuration validator

- Change Validation Policy: **Troubleshooting -> Configuration Problems**
- View: select **Troubleshooting**, expand **Configuration Validation**, and then select appropriate level of messages



© Copyright IBM Corporation 2007

Figure 11-40. Configuration validator

WA3616.3

### Notes:

The Auto-Refresh option can be enabled or disabled from the Preferences of the console settings.

Select: System administration -> Console setting -> Preferences.

***Instructor notes:***

**Purpose —**

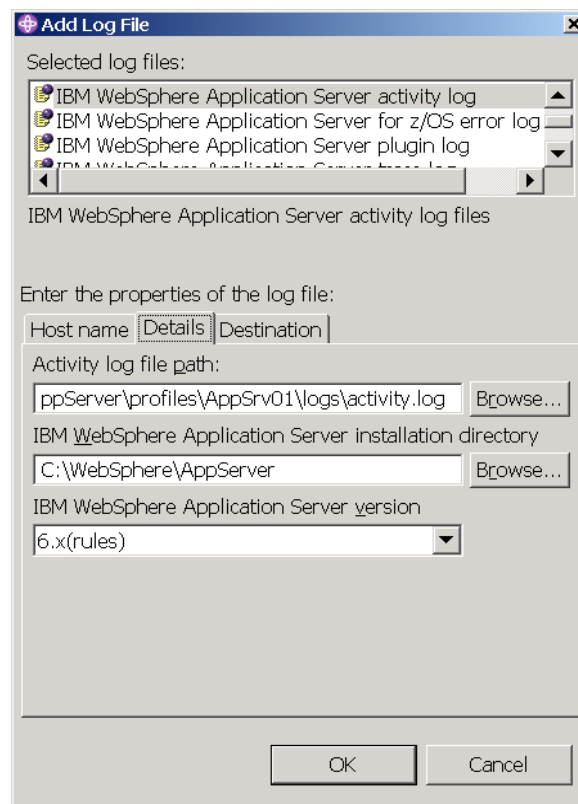
**Details —**

**Additional information —**

**Transition statement —** Next: Application Server Toolkit (AST)

## Application Server Toolkit (AST)

- Based on Eclipse tooling
- Similar to Rational Studio Application Developer, limited function
  - Remote debugger
  - Profiling and logging
  - J2EE packaging
  - Java perspectives
- Installed as separate program
- To work with log files:
  - File → Import
  - Select **Log File**, then type of log file
- For remote connection ensure that the **IBM Agent Controller** is running on remote host.



© Copyright IBM Corporation 2007

Figure 11-41. Application Server Toolkit (AST)

WA3616.3

### Notes:

The Log and Trace Analyzer in AST provides stand-alone and plug-in support for new and existing users of the JSR-047 Java Logging API and a lot of common loggings (DB2, AIX, all trace and log files of WebSphere Application Server, Portal Server, and so on).

***Instructor notes:***

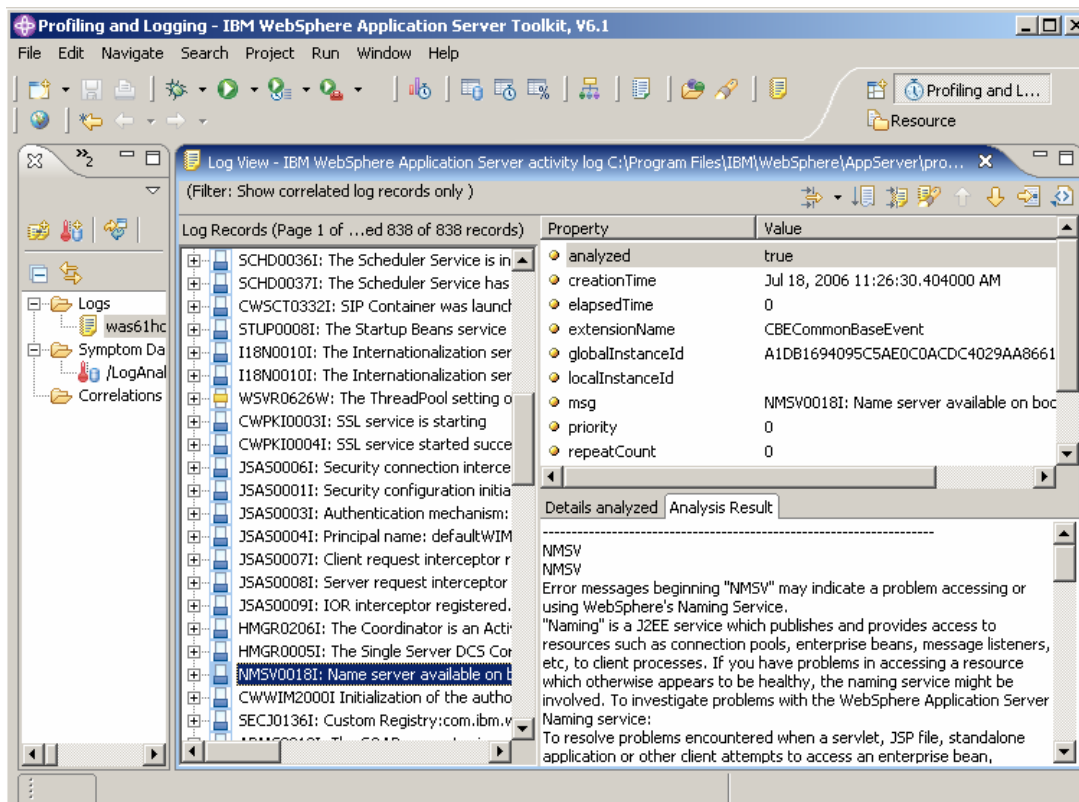
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: AST – Profiling and logging perspective

## AST – Profiling and logging perspective



© Copyright IBM Corporation 2007

Figure 11-42. AST – Profiling and logging perspective

WA3616.3

### Notes:

Analyzing log records means to compare log records using specified symptom databases that are loaded in memory. The solution is reported in the Analysis Result pane, giving the customer advice on resolving the reported problem.

1. To analyze a particular log record, in Log view, select it from the list of Log Records.
2. Right-click, then select **Analyze > Default Log Analyzer**.
3. To analyze multiple log records, use the Ctrl key to select the records. Right-click and select **Analyze > Default Log Analyzer**.
4. To analyze all log records, right-click, then select **Analyze All > Default Log Analyzer**.
5. If a match is found in the symptom database, the record is highlighted in blue and the solution information appears in the Analysis Result tab.

The stand-alone Log Analyzer is no longer available so the AST Log and Trace Analyzer is the only tool that can be used to view the IBM Service Log file.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**



---

## Dumping the JNDI name space

---

- DumpNameSpace utility shows JNDI directory content
- Useful to ensure correct association of named objects:
  - Data sources
  - EJBs
  - JMS resources
  - Other resources
- Syntax and some of the options:

```
<was_root>\bin\DumpNameSpace  
  [-host bootstrap_host_name (defaults to localhost)]  
  [-port bootstrap_port_number (defaults to 2809)]  
  [-startAt subcontext/in/the/tree]
```

- Output can be redirected to a file and inspected

© Copyright IBM Corporation 2007

Figure 11-43. Dumping the JNDI namespace

WA3616.3

### Notes:

Usage: DumpNameSpace [-keyword value]

If a keyword occurs more than once with different values, the value associated with the last occurrence will be used.

The keywords and associated values are:

-host myhost.austin.ibm.com

Bootstrap host, that is, the WebSphere host whose namespace you want to dump. Defaults to "localhost".

-port nnn

Bootstrap port. Defaults to 2809.

-factory com.ibm.websphere.naming.WsnInitialContextFactory

The initial context factory to be used to get the JNDI initial context. Defaults as shown and normally does not need to be changed.

-root [ cell | server | node | host | legacy | tree | default ]

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement — -**

## Dumping the JNDI name space example

```

.DistributedObjectCache
33 <top>/nodes/was61host01Node01/servers/server1/services/extensionregistry
33                                     javax.naming.Context
34 <top>/nodes/was61host01Node01/servers/server1/services/extensionregistry/g
lobal
34                                     com.ibm.wkplc.extension
registry.ExtensionRegistryProxy
35 <top>/nodes/was61host01Node01/servers/server1/ejb javax.naming.Context
36 <top>/nodes/was61host01Node01/servers/server1/ejb/QuoteFacadeHome
36                                     com.ibm.trade.QuoteFaca
deHome
37 <top>/nodes/was61host01Node01/servers/server1/ejb/TradeFacade
37                                     com.ibm.trade.TradeFaca
deHome
38 <top>/nodes/was61host01Node01/servers/server1/ejb/ivtEJBObject
38                                     com.ibm.websphere.ivt.i
vtEJB.ivtEJBHome
39 <top>/nodes/was61host01Node01/servers/server1/ejb/mgmt
39                                     javax.naming.Context
40 <top>/nodes/was61host01Node01/servers/server1/ejb/mgmt/MEJB
40                                     javax.management.j2ee.M
anagementHome
41 <top>/nodes/was61host01Node01/servers/server1/jta javax.naming.Context
42 <top>/nodes/was61host01Node01/servers/server1/jta/usertransaction
42                                     java.lang.Object
43 <top>/nodes/was61host01Node01/servers/server1/wm javax.naming.Context
44 <top>/nodes/was61host01Node01/servers/server1/wm/default
44                                     com.ibm.websphere.async
hbeans.WorkManager
45 <top>/nodes/was61host01Node01/servers/server1/eis javax.naming.Context
46 <top>/nodes/was61host01Node01/servers/server1/eis/DefaultDatasource_CMP

```

© Copyright IBM Corporation 2007

Figure 11-44. Dumping the JNDI namespace example

WA3616.3

### Notes:

The figure shows an example of the JNDI namespace dump. Note: the whole dump is not shown above because of space limitations.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Checkpoint questions

## Checkpoint questions

---

1. Name some resources that you can use to gather information on how to troubleshoot a problem.
2. What is a MustGather document?
3. What is the default location for the WebSphere Application Server logs?
4. What is the default location for the HTTP plug-in logs?
5. True or false: tracing cannot be started while the server is running.
6. Trace output can be directed to \_\_\_\_\_ or \_\_\_\_\_.

© Copyright IBM Corporation 2007

Figure 11-45. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.

**Instructor notes:****Purpose** —

**Details** — There are several right answers for Question 1. The Checkpoint Solutions graphic lists a few of the main ones.

## Checkpoint solutions

---

1. Resources for information gathering:
  - a. [IBM Support Assistant \(ISA\)](#)
  - b. [The WebSphere Support Page](#)
  - c. [The Information Center](#)
  - d. [Troubleshooting guide](#)
  - e. [And more](#)
2. A MustGather document tells you the specific information that you need to collect for a particular type of problem
3. `<was_root>\profiles\<profile_name>\logs`
4. `<plugins_root>/logs/<web_server_name>/http_p  
login.log`
5. False.
6. Trace output can be directed to a file or ring buffer.

© Copyright IBM Corporation 2007

**Additional information** —

**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe problem determination issues
- Identify resources for problem determination
- Describe tools for troubleshooting
- Describe the location of various log files
- Use AST and Log Analyzer to examine the activity log
- Describe FFDC and the Collector tool
- Use HTTP logs for problem determination

© Copyright IBM Corporation 2007

Figure 11-46. Unit summary

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

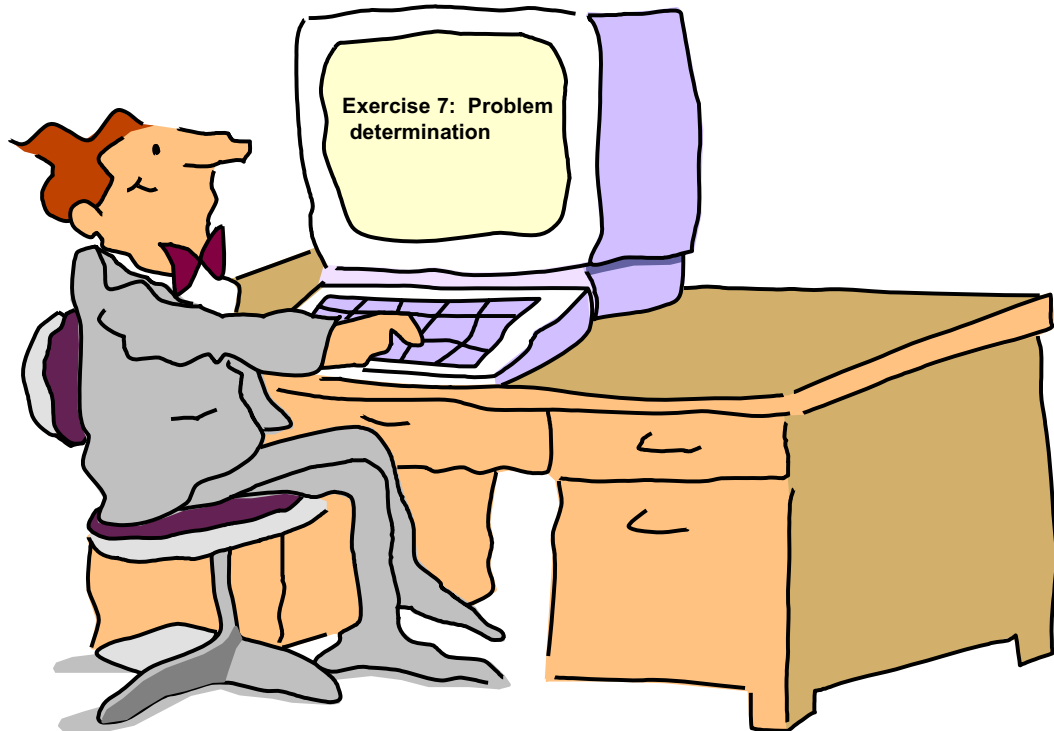
**Details —**

**Additional information —**

**Transition statement —** Next: Lab exercise

## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 11-47. Lab exercise

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**



---

# Unit 12. Introduction to wsadmin and scripting

## Estimated time

01:00

## What this unit is about

This unit describes the capabilities of wsadmin as an administrative tool for WebSphere Application Server.

## What you should be able to do

After completing this unit, you should be able to:

- Use wsadmin to issue administrative commands
- Create scripts in both JACL and Jython to execute wsadmin commands
- Use AdminApp to install an application
- Create various administrative artifacts
- Develop, test, and debug Jython scripts using AST

## How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

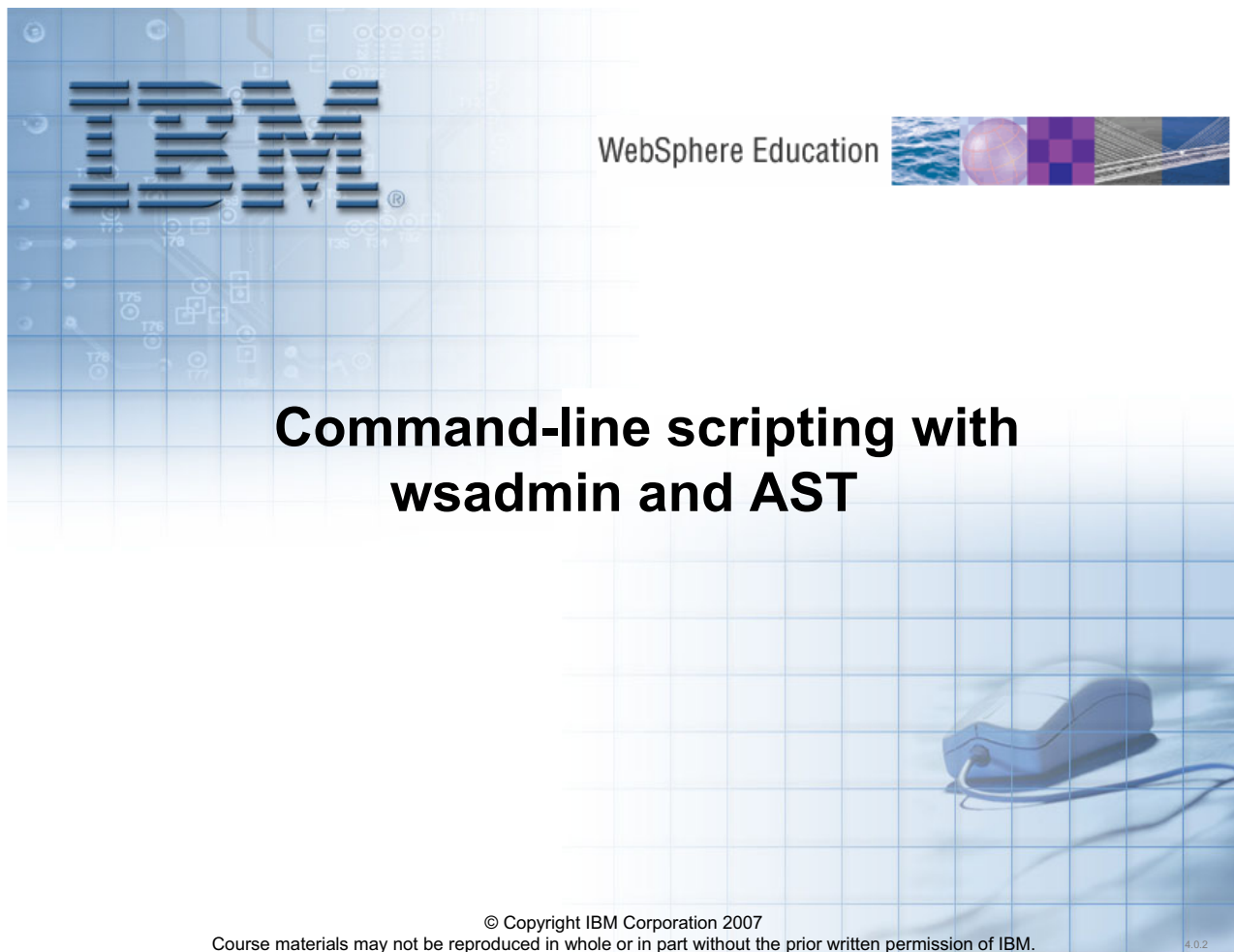


Figure 12-1. Introduction to wsadmin and scripting

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Unit objectives

---

After completing this unit, you should be able to:

- Use wsadmin to issue administrative commands
- Create scripts in both JACL and Jython to execute wsadmin commands
- Use AdminApp to install an application
- Create various administrative artifacts
- Develop, test, and debug Jython scripts using AST

© Copyright IBM Corporation 2007

Figure 12-2. Unit objectives

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

# Comparison of wsadmin and administration console

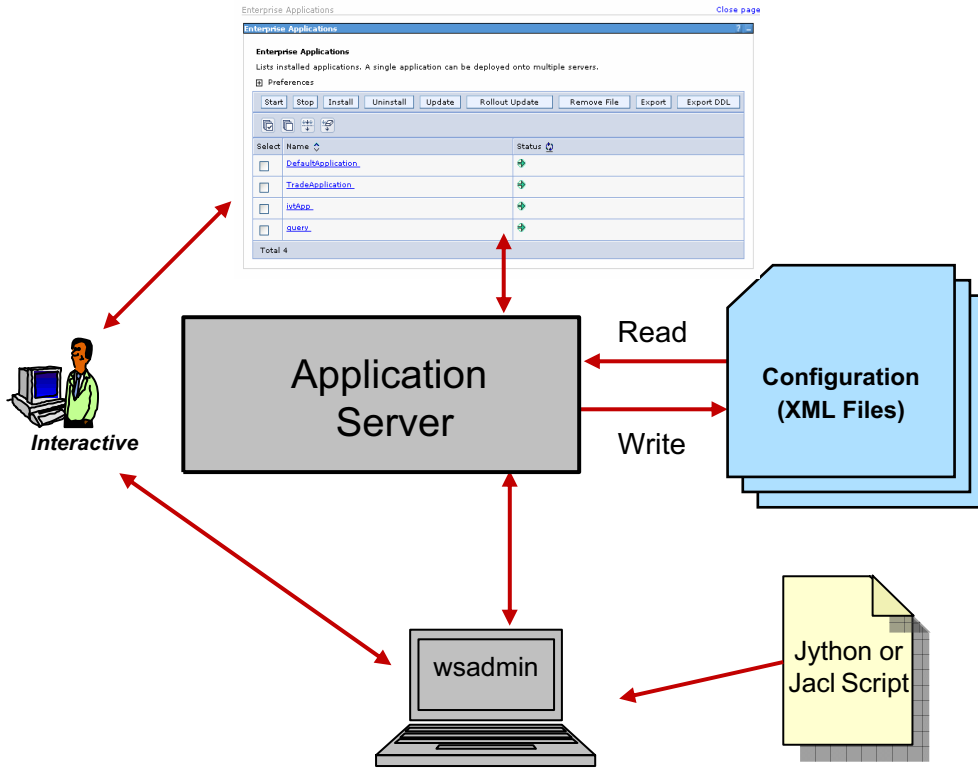


Figure 12-3. Comparison of wsadmin and administration console

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Why use scripting?

---

- Making configuration changes using the WebSphere administrative console is error prone and dangerous in production environments.
- Scripting is beneficial for production environments, especially with multiple nodes.
- Allows multiple configuration changes/updates to be applied to all targeted resources in an efficient and consistent manner.
- Allows for automating of routine administration tasks without having to rely on an operator running the administrative console.
- Scripts can be scheduled to run at times when it might be inconvenient to have an operator at the administrative console.

© Copyright IBM Corporation 2007

Figure 12-4. Why use scripting?

WA3616.3

### **Notes:**

Executing scripts is the primary purpose for wsadmin. While wsadmin can be used in interactive mode, its true power is the ability to run scripts.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Introduction to wsadmin

---

- Provides:
  - Scripting capabilities
  - Command-line administration
- Common operational and configuration tasks can be performed from scripts and the command line instead of through the administrative console.
- Examples:
  - Start and stop deployment manager, nodes, application servers, enterprise applications and clusters
  - Configure virtual hosts, JDBC providers, JMS resources
  - Create application servers
  - Create clusters and add members to a cluster
  - Almost all tasks available through administrative console

© Copyright IBM Corporation 2007

Figure 12-5. Introduction to wsadmin

WA3616.3

### **Notes:**

The WebSphere Application Server wsadmin tool provides the ability to execute scripts. You can use the wsadmin tool to manage a WebSphere Application Server V6.1 installation. This tool uses the Bean Scripting Framework (BSF), which supports a variety of scripting languages to configure and control your WebSphere Application Server installation.

The wsadmin launcher makes administrative objects available through language specific interfaces. Scripts use these objects for application management, configuration, operational control, and for communication with MBeans running in WebSphere server processes.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Overview of wsadmin

---

- wsadmin modes:
  - [Command line](#)
  - [Interactive](#)
  - [Script file](#)
  
- Support for:
  - [Python commands through Jython](#)
  - [Tcl commands through Java Command Language \(Jacl\)](#)
  
- Jacl versus Jython:
  - Use comes down to a personal choice
  - Jython syntax seems more natural to programmers used to Java or C
  - Jacl syntax can be more familiar to administrators familiar with Tcl
  - Each language has its own style and syntax, but they end up being able to do the same things
  - Jacl to be deprecated in V6.1
    - Still supported for the next two versions
    - Conversion assistant tool available to convert from Jacl to Jython (Jacl2Jython)
    - Much better support in tooling
      - [AST \(Jython editor, command completion, debugger\)](#)
      - [WebSphere administrative console \(Console Command Assistance \)](#)

© Copyright IBM Corporation 2007

Figure 12-6. Overview of wsadmin

WA3616.3

### **Notes:**

While wsadmin can be used in an interactive mode, the main purpose for wsadmin is to provide the ability to run scripts. While the contents of this unit and the lab try to make this very evident, you should make sure to emphasize this point at all times.



***Instructor notes:***

**Purpose —**

**Details —**

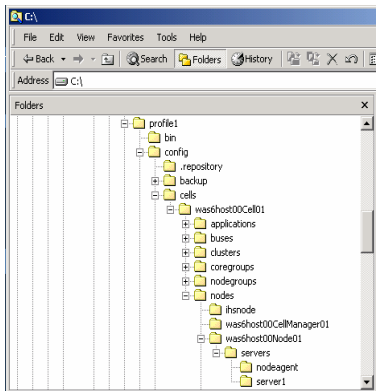
**Additional information —**

**Transition statement —**

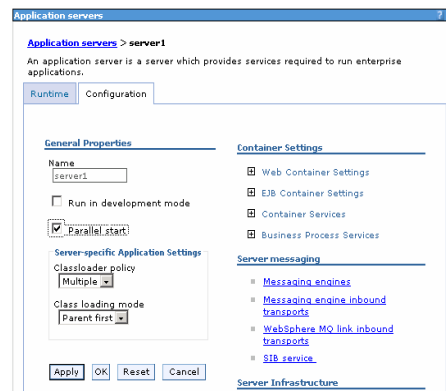
# Administrative functions using wsadmin

- WebSphere Application Server system management separates administrative functions into two categories:
  - Configuration of WebSphere Application Server installations (repository)
  - Running objects in WebSphere Application Server installations

## Repository



## Running Application Server



© Copyright IBM Corporation 2007

Figure 12-7. Administrative folders using wsadmin

WA3616.3

## Notes:

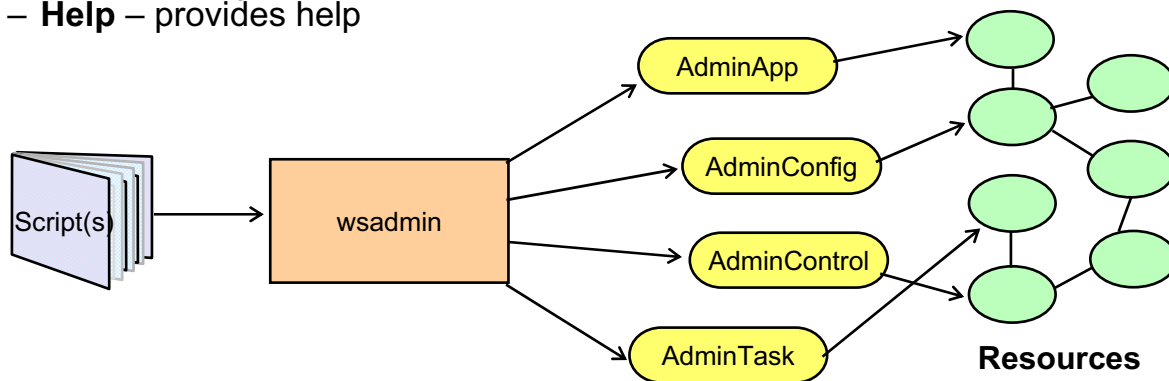
***Instructor notes:*****Purpose —**

**Details —** The screen captures serve two purposes. One is to show the repository (or at least its location) for the application server. You might want to point out all the directories that are important here. With the screen capture from the admin console, make sure to point out the two tabs, **Runtime** and **Configuration** and what their purposes are.

**Additional information —****Transition statement —**

## Administrative objects in wsadmin

- wsadmin acts as an interface to Java objects for access by scripts
- wsadmin uses the same interface (through JMX) as the administrative console to make configuration changes and control servers
- There are five administrative objects that perform different operations
  - **AdminControl** - work with "live" running objects, perform traces and data type conversion
  - **AdminConfig** - create/change WebSphere Application Server "static" configuration
  - **AdminApp** - install, modify or administer applications
  - **AdminTask** - access administrative commands to provide alternate way to access configuration commands
  - **Help** – provides help



© Copyright IBM Corporation 2007

Figure 12-8. Administrative objects in wsadmin

WA3616.3

### Notes:

Note the addition of the AdminTask administrative object to this picture. The "Help" administrative object has been removed from this picture but it still plays a vital role.

***Instructor notes:***

**Purpose —**

**Details —**

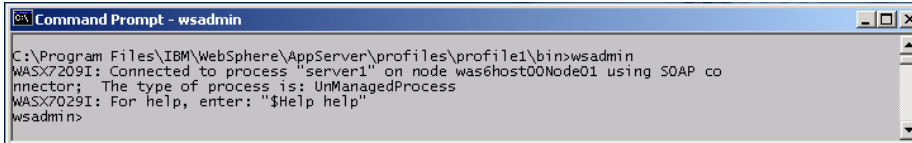
**Additional information —**

**Transition statement —**

## Invoking wsadmin

There are several ways to invoke wsadmin:

- Interactively `wsadmin`

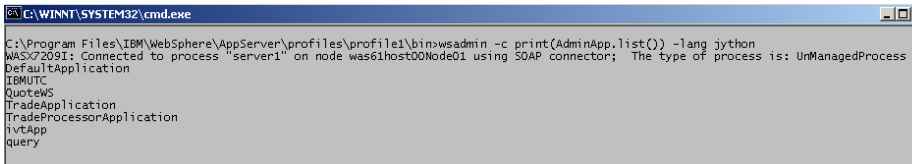


```

C:\Program Files\IBM\WebSphere\AppServer\profiles\profile1\bin>wsadmin
WASX7029I: Connected to process "server1" on node was6host00Node01 using SOAP co
nnecto; The type of process is: UnManagedProcess
WASX7029I: For help, enter: "$Help help"
wsadmin>

```

- Command option `wsadmin -c command`

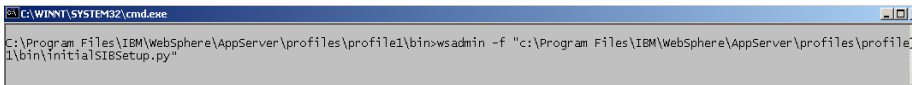


```

C:\Program Files\IBM\WebSphere\AppServer\profiles\profile1\bin>wsadmin -c print(AdminApp.list()) -lang jython
WASX7029I: Connected to process "server1" on node was61host00Node01 using SOAP connector; The type of process is: UnManagedProcess
DefaultApplication
IBMUTC
QuoteWS
TradeApplication
TradeProcessorApplication
vvtApp
query

```

- Script file `wsadmin -f scriptfile`



```

C:\Program Files\IBM\WebSphere\AppServer\profiles\profile1\bin>wsadmin -f "c:\Program Files\IBM\WebSphere\AppServer\profiles\profile1\bin\initialISESetup.py"

```

© Copyright IBM Corporation 2007

Figure 12-9. Invoking wsadmin

WA3616.3

### Notes:

When executing `wsadmin` with `-f` you do not need to specify the script's language with the `-lang` command-line option. `Wsadmin` can recognize the script's language by looking at the extension of the file name (`.py` for Jython and `.jacl` for Jacl).

***Instructor notes:***

**Purpose —**

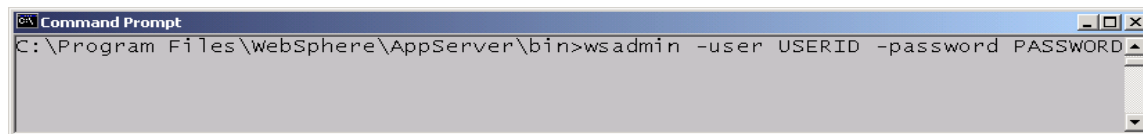
**Details —**

**Additional information —**

**Transition statement —**

## Invoking wsadmin with security enabled

- If security is enabled, authentication information must be supplied for server communications.
- User name and password can be supplied in one of several ways:
  - [Command-line parameters](#)
  - [sas.client.props](#) file for connecting with RMI
  - [soap.client.props](#) file for connecting with SOAP
    - [Encode password with PropFilePasswordEncoder script](#)
  - [Prompted](#)
- Security related wsadmin commands:
  - [securityoff](#)
  - [securityon user password](#)



© Copyright IBM Corporation 2007

Figure 12-10. Invoking wsadmin with security enabled

WA3616.3

### Notes:

In WebSphere V6.1 console security is enabled by default. That means that wsadmin needs to authenticate as well before it will execute.

Both `-user` and `-username` can be used interchangeably.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Connection type in wsadmin

---

- `wsadmin -conntype [SOAP | RMI | NONE ]`
  - The `-conntype` specifies the protocol type
  - Default is SOAP
  - NONE allows wsadmin to read and write directly to configuration files without going through an application server
- `wsadmin -profile <profile_script_name>`
  - Allows wsadmin to preload a profile script

© Copyright IBM Corporation 2007

Figure 12-11. Connection type in wsadmin

WA3616.3

### **Notes:**

An example of when NONE could be used is when the application server configuration files are so corrupted that the application server can not be started in order to access the administrative tasks of the running server.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Profile scripts

- Profile scripts can be used to preload wsadmin with predefined settings and functions
  - Run during wsadmin startup
  - Called by either:
    - Using the `-profile` parameter on the command line
    - Defined in `wsadmin.properties`  
`com.ibm.ws.scripting.profiles=`
  - Could be used to create a standard wsadmin environment for all WebSphere administrators
  - Profile scripts can be written in Jython or Jacl

```

Command Prompt - wsadmin -lang jython
C:\Program Files\IBM\WebSphere\AppServer\profiles\profile1\bin\wsadmin
WASX7209I: Connected to process "server1" on node was6host00Node01 using SOAP co
nnectior; The type of process is: UnManagedProcess

Hello, and welcome to wsadmin using jython

Running global_profile.py. Global definitions and settings could be added
here. It would also be possible to extend wsadmin by defining new
customized commands and procedures.

Cell: was6host00Node01Cell
Node: was6host00Node01

WASX7031I: For help, enter: "print Help.help()"
wsadmin>
  
```

```

-----
# Print whereAmI
#-----
def whereAmI():
    #Print cell and node names
    print "Cell: " + AdminConfig.showAttribute(AdminConfig.list("Cell"), "name")
    print "Node: " + AdminConfig.showAttribute(AdminConfig.list("Node"), "name")
    return

#-----
# Start of main
#-----
print ""
print "Hello, and welcome to wsadmin using jython"
print ""
print "Running global_profile.py. Global definitions and settings could be added"
print "here. It would also be possible to extend wsadmin by defining new"
print "customized commands and procedures"
print ""

whereAmI()

print ""
  
```

© Copyright IBM Corporation 2007

Figure 12-12. Profile scripts

WA3616.3

### Notes:

The example above automatically called the profile by having the profile defined in the `wsadmin.properties` file. It could have also been called using the `-profile` flag.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Properties of wsadmin

---

- Certain default behaviors for wsadmin can be changed by editing `<profile_root>\<profile_name>\properties\wsadmin.properties`
- Properties include:
  - `com.ibm.ws.scripting.connectionType=SOAP`
  - `com.ibm.ws.scripting.port=8880`
  - `com.ibm.ws.scripting.host=localhost`
  - `com.ibm.ws.scripting.defaultLang=jython`
  - `com.ibm.ws.scripting.traceFile=`
  - `com.ibm.ws.scripting.validationOutput=`
  - `com.ibm.ws.scripting.traceString=com.ibm.*=all=enabled`
  - `com.ibm.ws.scripting.profiles=`
  - `com.ibm.ws.scripting.emitWarningForCustomSecurityPolicy=true`
  - `com.ibm.ws.scripting.tempdir=`
  - `com.ibm.ws.scripting.validationLevel=`
  - `com.ibm.ws.scripting.crossDocumentValidationEnabled=`
  - `com.ibm.ws.scripting.classpath=`

© Copyright IBM Corporation 2007

Figure 12-13. wsadmin properties

WA3616.3

### Notes:

Useful and commonly used properties are provided by `connectionType`, where you can define whether to use NONE, SOAP or RMI connection types. The `port` can be used to define the connection port number to be used if different from the default.

The `defaultLang` can be used to change the default language. For example, it could be changed to Jython. Be aware that if you change this, some WebSphere Application Server scripts which depend on the default language being Jacl will fail unless rewritten to call wsadmin with `-lang jac1` command-line switch.

***Instructor notes:***

**Purpose —**

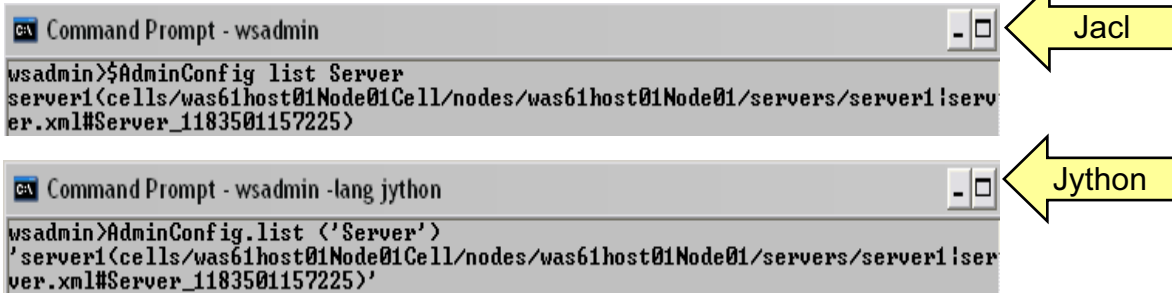
**Details —**

**Additional information —**

**Transition statement —**

## AdminConfig – Managing configurations

- Configuration management scripts use the AdminConfig object to access the repository where configuration information is stored



```
Command Prompt - wsadmin
wsadmin>$AdminConfig list Server
server1(cells/was61host01Node01Cell/nodes/was61host01Node01/servers/server1|server.xml#Server_1183501157225)

Command Prompt - wsadmin -lang jython
wsadmin>AdminConfig.list ('Server')
'server1(cells/was61host01Node01Cell/nodes/was61host01Node01/servers/server1|server.xml#Server_1183501157225)'
```

- Use the AdminConfig object to:
  - List configuration objects and their attributes
  - Create configuration objects
  - Modify configuration objects
  - Remove configuration objects
  - Obtain help

© Copyright IBM Corporation 2007

Figure 12-14. AdminConfig – Managing configurations

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

**Details —**

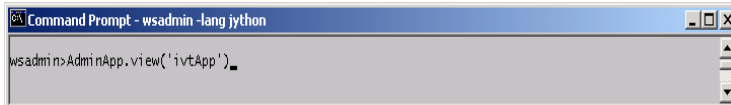
**Additional information —**

**Transition statement —**

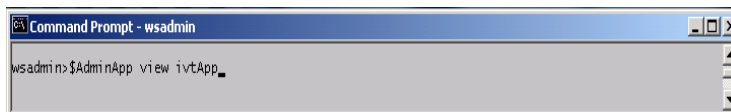
## AdminApp – Managing applications

---

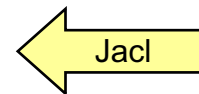
- Application management scripts use the AdminApp object to manage applications in the application server configuration
- Example:



```
Command Prompt - wsadmin -lang jython
wsadmin>AdminApp.view('ivtApp')
```



```
Command Prompt - wsadmin
wsadmin>$AdminApp view ivtApp
```



- Use the AdminApp object to:
  - Install and uninstall applications
  - List installed applications
  - Edit application configurations
  - Obtain help

© Copyright IBM Corporation 2007

Figure 12-15. AdminApp – Managing applications

WA3616.3

### Notes:

When running the above commands it will produce a lot of output specific to the application. In the Jython command output is easier to read when using the print option.

In Jython: `print AdminApp.view ('ivtApp')`

***Instructor notes:***

**Purpose —**

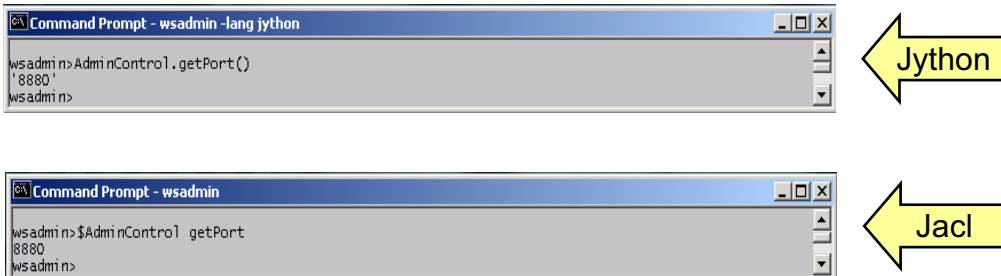
**Details —**

**Additional information —**

**Transition statement —**

## AdminControl – Managing running objects

- Operation management scripts use the AdminControl object to communicate with the MBeans that represent running objects.
- Example:



- Use the AdminControl object to:
  - List running objects and their attributes
  - Invoke actions on running objects
  - Obtain dynamic information about Mbeans that represent running objects
  - Obtain help

© Copyright IBM Corporation 2007

Figure 12-16. AdminControl – Managing running objects

WA3616.3

### Notes:

The number and type of MBeans available to the scripting client depends on the server to which the client is connected. If the client is connected to a deployment manager, then all the MBeans running in the deployment manager are visible, as are all the MBeans running in the node agents connected to this deployment manager, and all the MBeans running in the application servers on those nodes.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## AdminTask – Accessing administrative functions

- Used to access set of administrative commands to provide alternate way to access configuration commands
- Runs simple and complex commands
- Grouped based on their function
  - Example: Commands related to security are grouped into a security management command group.



- Can be run in batch or interactive mode
- Can be run in connected or local mode
  - If run in local mode not all commands available

© Copyright IBM Corporation 2007

Figure 12-17. AdminTask – Accessing administrative functions

WA3616.3

### Notes:

Use the AdminTask object to access a set of administrative commands that provide an alternative way to access the configuration commands and the running object management commands. The administrative commands run simple and complex commands. They provide more user friendly and task-oriented commands. The administrative commands are discovered dynamically when you start a scripting client. The set of available administrative commands depends on the edition of WebSphere Application Server you install. You can use the AdminTask object commands to access these commands.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Help within wsadmin

---

- Jython
  - `print Help.help()`
  - `print Help.AdminConfig()`
  - `print Help.AdminTask()`
  - `print Help.AdminControl()`
  - `print Help.AdminApp()`
  
- Jacl
  - `$Help help`
  - `$AdminConfig help`
  - `$AdminTask help`
  - `$AdminControl help`
  - `$AdminApp help`

© Copyright IBM Corporation 2007

Figure 12-18. Help within wsadmin

WA3616.3

### **Notes:**

For more complete documentation, see the Information Center.

The Information Center also includes a number of examples that demonstrates some useful wsadmin functionality (with both Jacl and Jython).

`Help.<object> == <object>.help()`



***Instructor notes:***

**Purpose —**

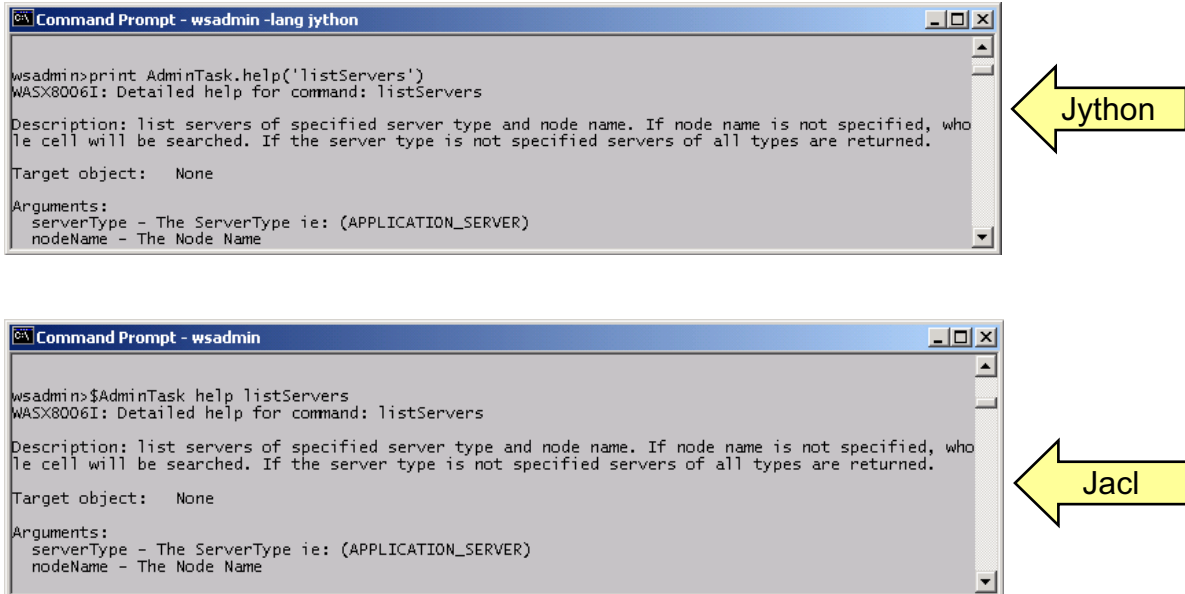
**Details —**

**Additional information —**

**Transition statement —**

## Help on specific commands

- Help can be obtained on specific commands within the administration objects.



© Copyright IBM Corporation 2007

Figure 12-19. Help on specific commands

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Important points to remember when using wsadmin

---

- Commands are case-sensitive.
- wsadmin -f "scriptfile" faster than wsadmin -c "command"
  - Better to run multiple commands in a script file than individual commands with wsadmin -c
- Saving configuration changes is a two step process:
  - First part validates the changes.
  - Second part performs the save but will throw an exception if changes conflict (example: creating two servers with the same name).
- Save periodically in the script file or interactive mode to persist configuration updates to existing objects.
  - AdminConfig.save() (Jython)
  - \$AdminConfig save (Jacl)

© Copyright IBM Corporation 2007

Figure 12-20. Important points to remember when using wsadmin

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Scripting – Simple script

```
print "Simple wsadmin scripting example."

cell = AdminConfig.list("Cell")
node = AdminConfig.list("Node")

cellName = AdminConfig.showAttribute(cell, "name")
nodeName = AdminConfig.showAttribute(node, "name")

print ""
print "Cell name is: " + cellName
print "Node name is: " + nodeName
```

**Jython**

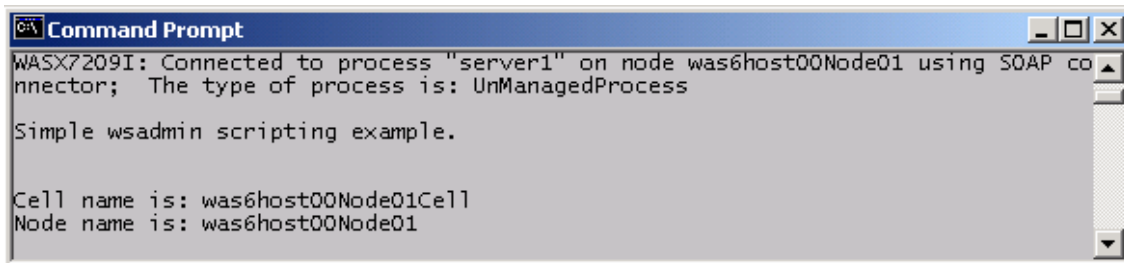
```
puts "Simple wsadmin scripting example."

set cell [${AdminConfig list Cell}]
set node [${AdminConfig list Node}]

set cellName [${AdminConfig showAttribute $cell name}]
set nodeName [${AdminConfig showAttribute $node name}]

puts ""
puts "Cell name is: $cellName"
puts "Node name is: $nodeName"
```

**Jacl**



```
Command Prompt
WASX7209I: Connected to process "server1" on node was6host00Node01 using SOAP co
nnecter; The type of process is: UnManagedProcess

Simple wsadmin scripting example.

Cell name is: was6host00Node01Cell
Node name is: was6host00Node01
```

© Copyright IBM Corporation 2007

Figure 12-21. Scripting – Simple script

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Scripting – Looping

```
print "Simple loop script"
print ""

appNames = AdminApp.list()
appNamesArray = appNames.split('\r\n')

for appName in appNamesArray:
    print "App Name: " + appName
```

### Jython

```
puts "Simple loop script"
puts ""

set applicationNames [$AdminApp list]

foreach appName $applicationNames {
    puts "App Name: $appName"
}
```

### Jacl

```
Command Prompt
WASX7209I: Connected to process "server1" on node was6host00Node01 using SOAP connector; The type of process is: UnManagedProcess
Simple loop script

App Name: DefaultApplication
App Name: QuoteWS
App Name: Trade Order Processor
App Name: TradeApplication
App Name: ivtApp
App Name: query

C:\Program Files\IBM\WebSphere\AppServer\profiles\profile1\bin>
```

© Copyright IBM Corporation 2007

Figure 12-22. Scripting – Looping

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

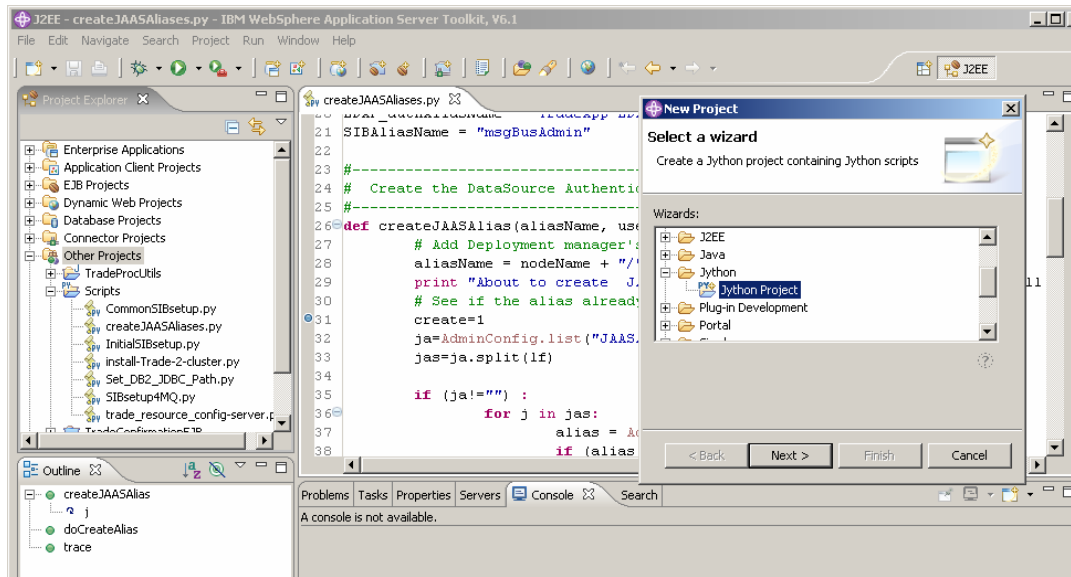
**Details —**

**Additional information —**

**Transition statement —**

## Create, test, and debug scripts with AST

- Jython projects and source files fully supported in IDE
  - Create Jython project and files
  - Syntax colored editor, command completion
  - Source Outline view
  - Source level debugging



© Copyright IBM Corporation 2007

Figure 12-23. Create, test, and debug scripts with AST

WA3616.3

### Notes:

Jython projects and scripts treated as first class objects. Jython editor provides same facilities as all other specialized editors, for example syntax coloring, command completion, automatic formatting, and so on. Jython Projects should be created under the Other Projects folder in AST.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

# Launch Jython debugger

- To launch the Jython debugger
  - Select the script's context menu
  - Select **Debug As -> Administration script**
- First time requires setup of:
  - Provide any runtime parameters
  - Security information for server
  - Click **Debug**

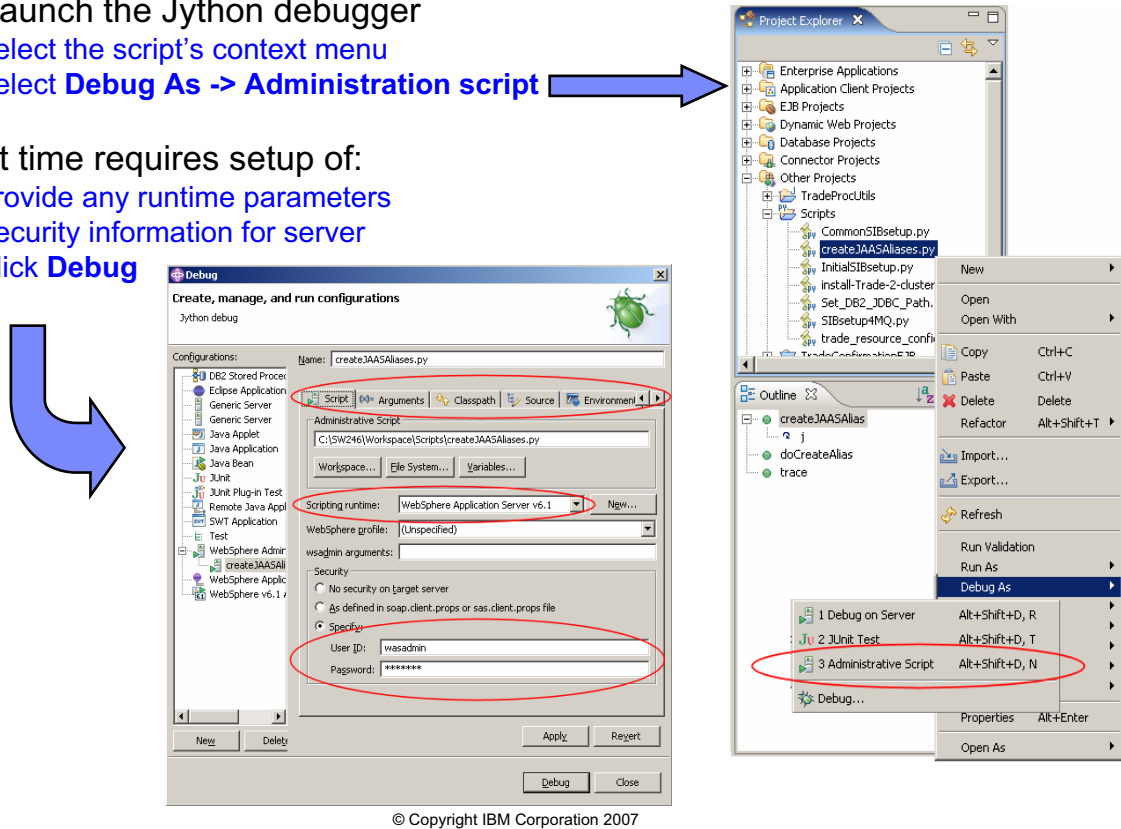


Figure 12-24. Launch Jython debugger

WA3616.3

## Notes:

***Instructor notes:***

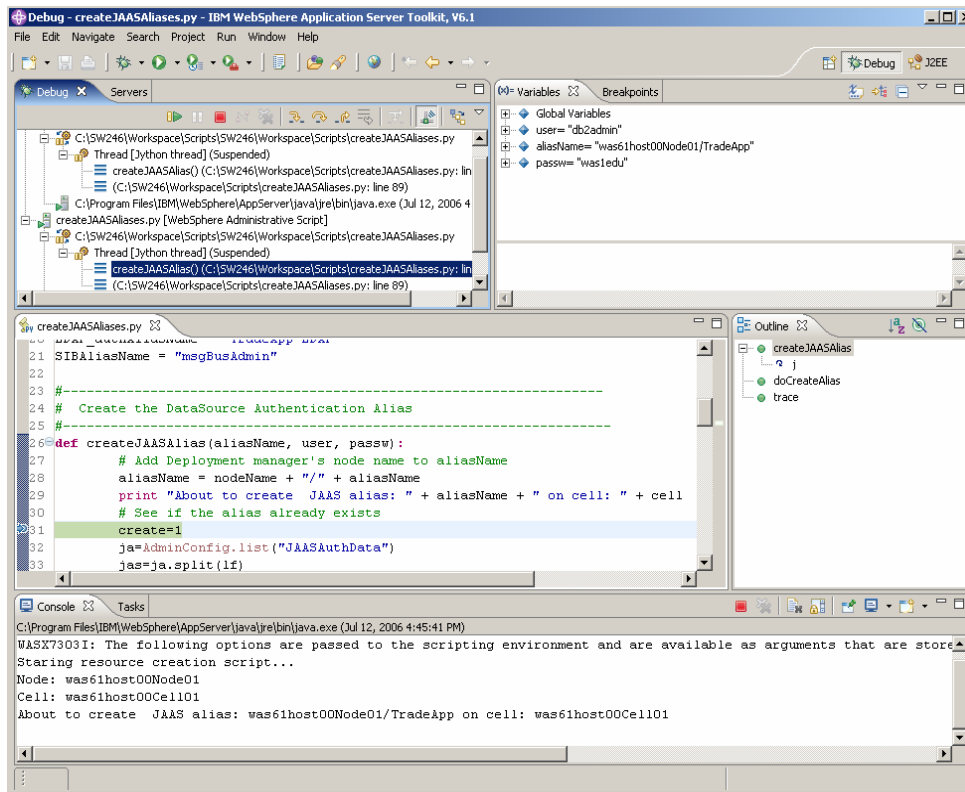
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

# Debugger perspective



© Copyright IBM Corporation 2007

Figure 12-25. Debugger perspective

WA3616.3

## Notes:

Be aware that the Jython debugger alters the order of command-line parameters. In the debugger `argv[0]` is the name of the script, while running outside the debugger `argv[0]` is the first command-line argument passed to the script. This needs to be compensated for in the script if command-line arguments are used.

***Instructor notes:***

**Purpose —**

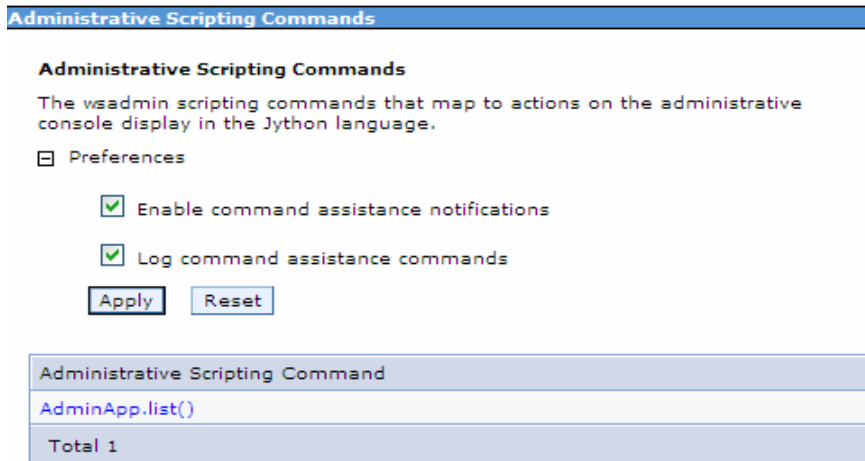
**Details —**

**Additional information —**

**Transition statement —**

## WebSphere Administration Command assist

- Works in concert with administrative console
  - Makes last executed commands available in AST
  - Commands can be pasted directly to Jython script
- Server side setup
  - Select **Applications -> Enterprise Applications**
  - Under **Help -> Command Assistance**, click **View administrative scripting** command for last action
  - Expand **Preferences** and enable both options.



© Copyright IBM Corporation 2007

Figure 12-26. WebSphere Administration Command assist

WA3616.3

### Notes:

When you perform server operations in the WebSphere Administrative Console, the WebSphere Administration Command assist tool captures and displays the wsadmin commands issued. You can transfer the output from the WebSphere Administration Command view directly to a text editor, such as the Jython editor, enabling you to develop Jython scripts based on actual console actions.



***Instructor notes:***

**Purpose —**

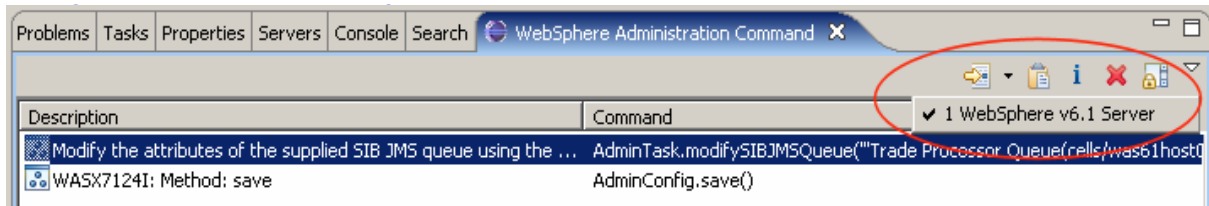
**Details —**

**Additional information —**

**Transition statement —**

# WebSphere Administration Command assist

- AST setup
  - Add WebSphere Administration Command view
    - Select **Window -> Show View -> Other -> Server -> WebSphere Administration Command**
  - Open the WebSphere Administration Command view



- Select desired command
- Copy into open Jython script file using second toolbar button

```

*CommonSIBsetup.py
7 #
8 # This example runs on WebSphere Application Server Version 6.0 and beyond.
9 #
10 # The purpose of this script is to create and configure a System Integration E
11 # and other bus and JMS resources for the Trade family of applications on the
12 #-----
13 #
14 import sys
15
16 AdminTask.modifySIBJMSQueue (
17     "Trade Processor Queue(cells/was61host00Cell101/nodes/was61host00Node01|resc
18     [-name "Trade Processor Queue" -jndiName jms/tradeProcJMSQueue -descriptor
19
    
```

© Copyright IBM Corporation 2007

Figure 12-27. WebSphere Administration Command assist

WA3616.3

## Notes:

The server must be started and with its preferences set to enable command assistance as in the previous slide.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Build tool: Apache Ant

---

- Another Neat Tool written and extended in Java
- Cross platform
  - Unless you choose to invoke shell commands
- Write scripts by writing Ant targets which
  - Invoke Ant tasks
  - Invoke WebSphere tasks
  - Depend on other targets
  - May have the side effect of creating an artifact
- Ant tasks implemented in Java
  - Use existing
  - Write your own
- WebSphere ships own version: ws\_ant

© Copyright IBM Corporation 2007

Figure 12-28. Build tool: Apache Ant

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## WebSphere Ant tasks

---

- WebSphere Application Server includes Ant tasks to:
  - Install and uninstall applications
  - Run EJB deployment and JSP pre-compilation tools
  - Start and stop servers
  - Run administrative scripts or commands
- To run Ant use the ws\_ant command
- For complete set of WebSphere Ant tasks and how to use them, see InfoCenter under:
  - [Reference -> Developer -> API Documentation -> Application programming interfaces -> Package com.ibm.websphere.ant.tasks](#)

© Copyright IBM Corporation 2007

Figure 12-29. WebSphere Ant tasks

WA3616.3

### **Notes:**

- For details of tasks see the Javadoc for package com.ibm.websphere.ant.tasks.
- For information about Ant see <http://jakarta.apache.org/ant/index.html>

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Some WebSphere Ant tasks

---

- wsStartServer, wsStopServer, wsServerStatus
- wsInstallApp, wsUninstallApp, wsUpdateApp, wsListApps
- wsejbdeploy, wsdeploy, JspC
- WSDL2Java, Java2WSDL
- wsadmin

© Copyright IBM Corporation 2007

Figure 12-30. Some WebSphere Ant tasks

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Checkpoint questions

---

1. Which of the following is not one of the five Java objects that perform different operations?
  - A. AdminConfig
  - B. AdminControl
  - C. AdminTask
  - D. Help
  - E. AdminStart
  
2. The default protocol type for wsadmin is?
  - A. SOAP
  - B. RMI
  - C. None
  
3. The default behaviors for wsadmin can be changed by editing which file?

© Copyright IBM Corporation 2007

Figure 12-31. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.

***Instructor notes:***

Purpose —

## **Checkpoint solutions**

---

1. E AdminStart
2. A. SOAP
3. wsadmin.properties

© Copyright IBM Corporation 2007

Transition statement —

## Unit summary

---

Having completed this unit, you should be able to:

- Use wsadmin to issue administrative commands
- Create scripts in both JACL and Jython to execute wsadmin commands
- Use AdminApp to install an application
- Create various administrative artifacts
- Develop, test, and debug Jython scripts using AST

© Copyright IBM Corporation 2007

Figure 12-32. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 12-33. Lab exercise

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**





---

# Unit 13. Federating a cell

## Estimated time

01:00

## What this unit is about

This unit describes the major concepts involved when administering cells in WebSphere Application Server. This unit also discusses the process for creating profiles.

## What you should be able to do

After completing this unit, you should be able to:

- Describe WebSphere Application Server cell concepts
- Describe the installation process for creating profiles
- Verify the installation of profiles
- Describe the directories and configuration files for WebSphere profiles
- Add a WebSphere Application Server node to a cell
- Explain the difference between managing a stand-alone server and a cell

## How you will check your progress

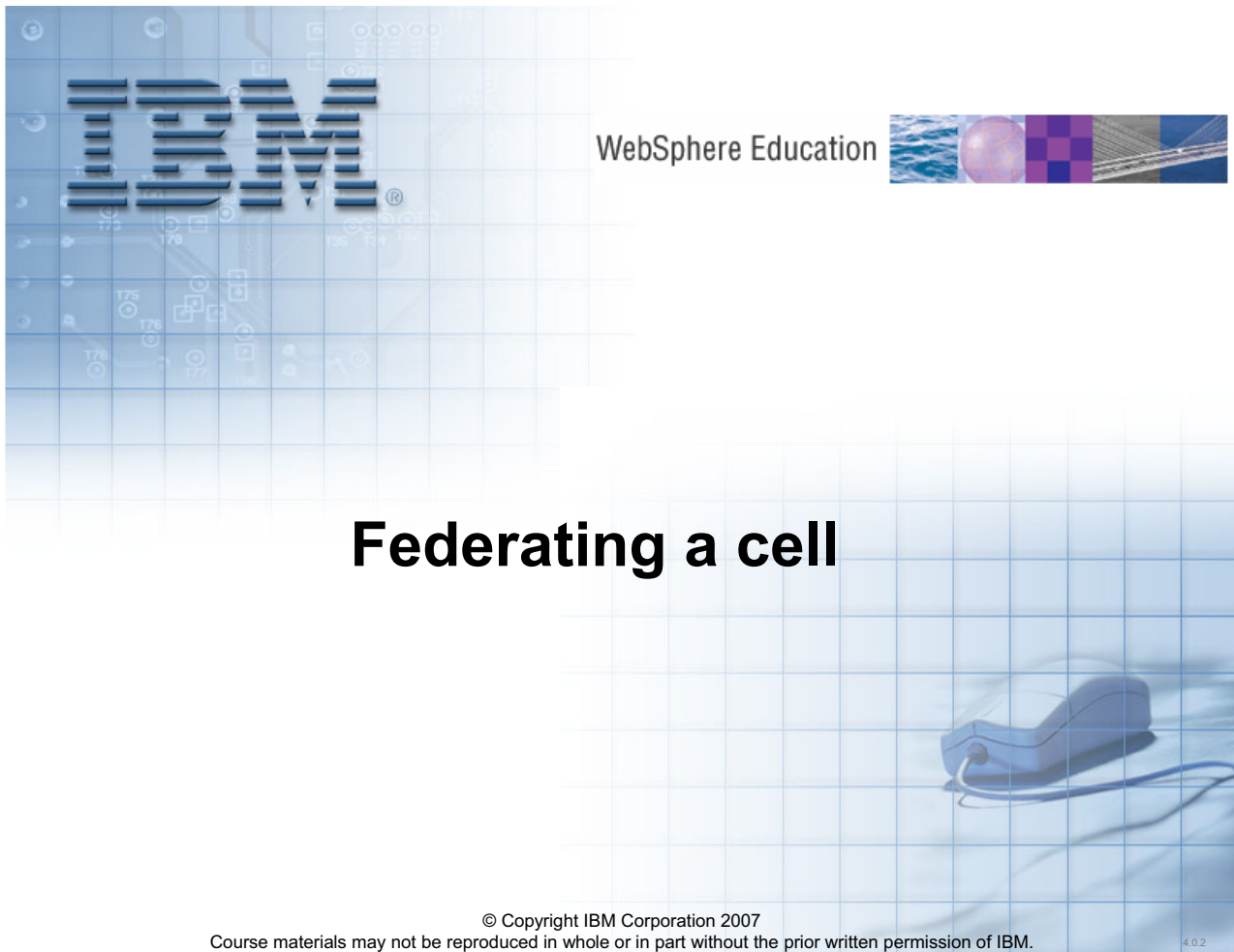
Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center



## Federating a cell

Figure 13-1. Federating cells

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe WebSphere Application Server cell concepts
- Describe the installation process for creating profiles
- Verify the installation of profiles
- Describe the directories and configuration files for WebSphere profiles
- Add a WebSphere Application Server node to a cell
- Explain the difference between managing a stand-alone server and a cell

© Copyright IBM Corporation 2007

Figure 13-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere cells

## WebSphere cells

- A WebSphere cell defines administrative domain.
  - Available in WebSphere Application Server Network Deployment.
  - Deployment manager provides centralized administration of all resources in the cell.
    - Created as a profile.
  - Nodes run application components in application servers.
    - WebSphere Application Server base nodes can be added to a cell.

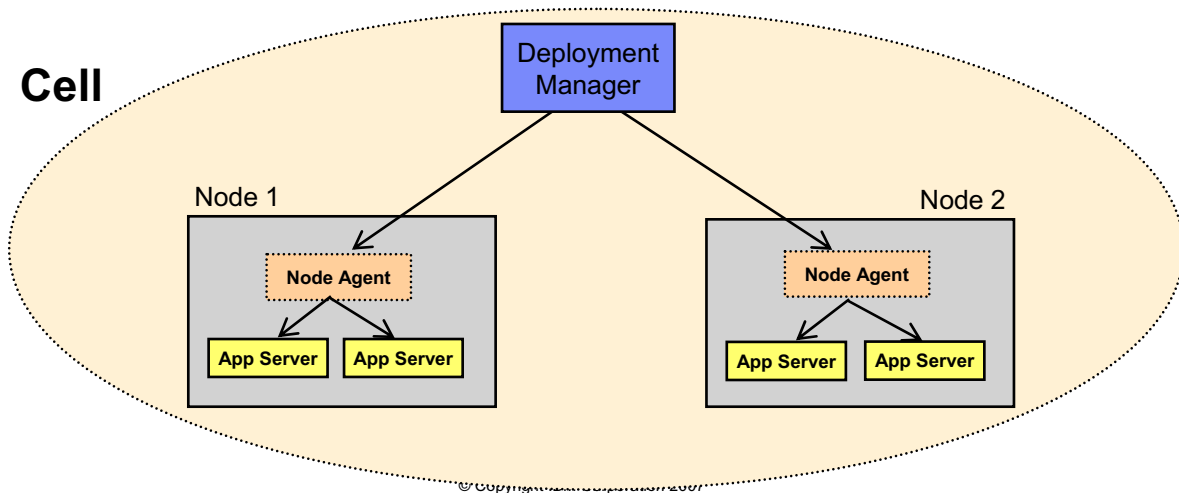


Figure 13-3. WebSphere cells

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere Application Server types

## WebSphere Application Server types

---

- **Application server**
  - Provides the functions that are required to support and host user applications
  - Runs on only one node, but one node can support many application servers
  
- **Node agent**
  - Created and installed when a node is federated
  - Works with the deployment manager to perform administrative activities on the node
  
- **Deployment manager**
  - Administers multiple application servers from one centralized manager
  - Works with the node agents on each node to manage all the servers in a distributed topology

© Copyright IBM Corporation 2007

Figure 13-4. WebSphere Application Server types

WA3616.3

### **Notes:**

There are three main types of WebSphere managed processes that make a cell.



***Instructor notes:***

**Purpose** — This slide is mostly here for reference. It would probably be more effective to move on to the next slide and talk to the picture.

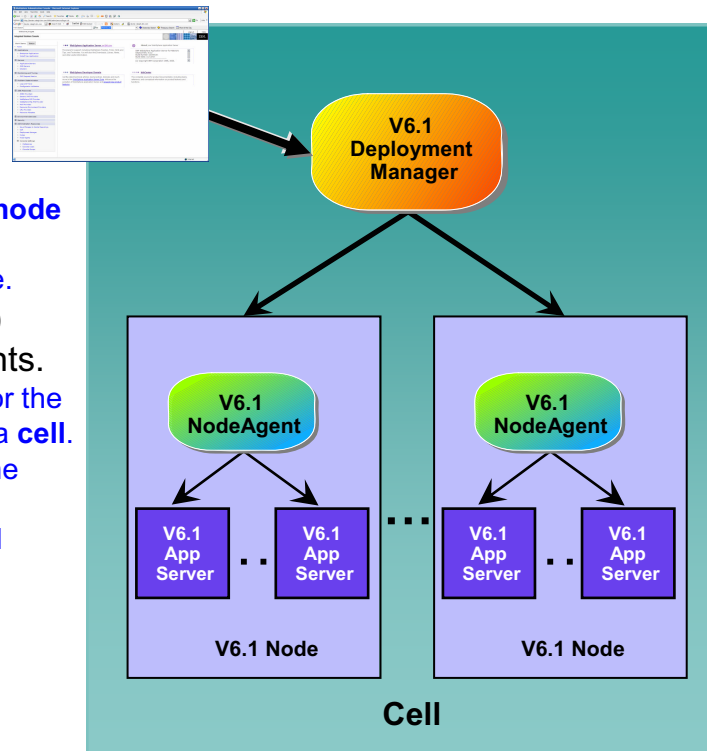
**Details** —

**Additional information** —

**Transition statement** — Next: Network deployment concepts

# Network deployment concepts

- A **node** is a logical grouping of servers.
  - Each node is managed by a single **node agent** process.
  - Each node is defined within a profile.
- A **deployment manager (DMgr)** process manages the node agents.
  - Holds the configuration repository for the entire management domain, called a **cell**.
  - Administrative service runs inside the DMgr.
  - The deployment manager is defined within a profile.



© Copyright IBM Corporation 2007

Figure 13-5. Network deployment concepts

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere profiles overview

## WebSphere profiles overview

---

- WebSphere files are split into two components:
  - **Product files**
    - Set of shared read-only static files or product binaries shared by any functional instance of the WebSphere Application Server product
  - **Configuration files (profiles)**
    - Set of user-customizable data files, called a profile
    - Files include: WebSphere configuration, installed applications, resource adapters, properties, log files, and so forth



© Copyright IBM Corporation 2007

Figure 13-6. WebSphere profiles overview

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere profiles benefits

## WebSphere profiles benefits

---

- Benefits of profiles in network deployment:
  - Think of profiles as representing a node
    - Can install multiple profiles on a single machine
    - Can contain a single stand-alone application server
    - Can be federated
  - Each profile uses the same product files
    - Application server profile (stand-alone)
      - Equivalent to Base or Express application server
      - Has a node name and a cell name property, and corresponding directories.
      - Cell directory is overwritten upon federation
    - Deployment manager profile
      - Creates a deployment manager
    - Custom profile (managed)
      - Creates a managed node which, by default, is federated into a cell.
      - Creates a node agent, but no application servers.
    - Cell profile
      - Creates both a deployment manager and a federated node

© Copyright IBM Corporation 2007

Figure 13-7. WebSphere profiles benefits

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose** —

**Details** —

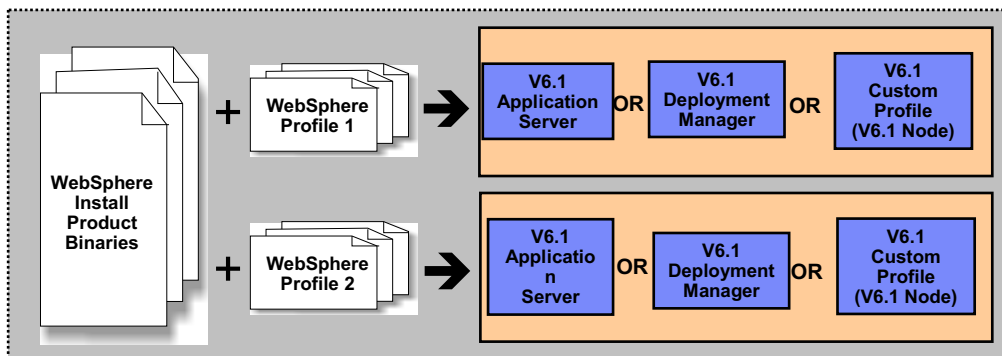
**Additional information** — This was effectively possible to do in V5 by either installing WebSphere Application Server multiple times on the same machine or using wsinstance.

**Transition statement** — Next: WebSphere profiles types

# WebSphere profiles types

- **Profile types:** define application server configuration

Profile Types	V6.1 packages	Functions
Application server (default)	All	Create different instances of a standalone node – each stand-alone node has 1 application server
Deployment manager	Network Deployment	Create different instances of DMgr – each DMgr is its own cell
Custom (managed)	Network Deployment	Creates and federates a node containing <b>no pre-defined application server</b> definitions
Cell	Network Deployment	Creates a combination of a DMgr and a federated node.



© Copyright IBM Corporation 2007

Figure 13-8. WebSphere profiles types

WA3616.3

## Notes:

The Profile Management Tool lets you create a deployment manager profile, an application server profile, or a custom profile. A *profile* consists of files that define the runtime environment for the deployment manager or the application server. Each environment has its own administrative interface. A custom profile is an exception. A custom profile is an empty node that you can federate into a deployment manager cell and customize. No default server processes or applications are created for a custom profile. Each deployment manager or application server profile has its own First steps console. The location of the command to launch the First steps console is within the set of files in the profile. A prompt to launch the First steps console that is associated with a profile displays on the last panel of the Profile creation wizard.

The PMT refers to these as *environments* because selecting *cell* results in creating two profiles: a Deployment Manager profile and a federated Application server profile. There is really no such thing as a *cell profile*. It is just a fast way to bootstrap a cell.



***Instructor notes:***

**Purpose —**

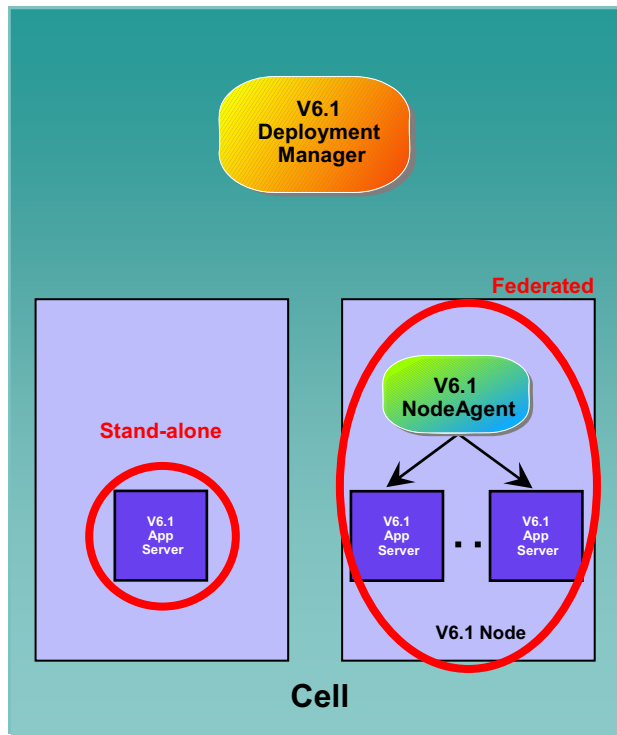
**Details —**

**Additional information —**

**Transition statement —** Next: Application server profile

## Application server profile

- **Application server** profiles provide a base installation.
  - Application server in the network deployment product can run in a deployment manager cell as a **managed node** or on its own as a **stand-alone** application server.
  - Multiple application server profiles can be created on a single machine.
  - Each application server profile can be federated into a cell.
    - If there are multiple base profiles on a single machine, they can be federated into the same cell, different cells, or remain stand-alone.



© Copyright IBM Corporation 2007

Figure 13-9. Application server profile

WA3616.3

### Notes:

When an application server profile is created, that application server can remain stand-alone. Or, for each profile, it can be federated into a cell. Once in a cell, it becomes its own node which activates a node agent.

***Instructor notes:***

**Purpose —**

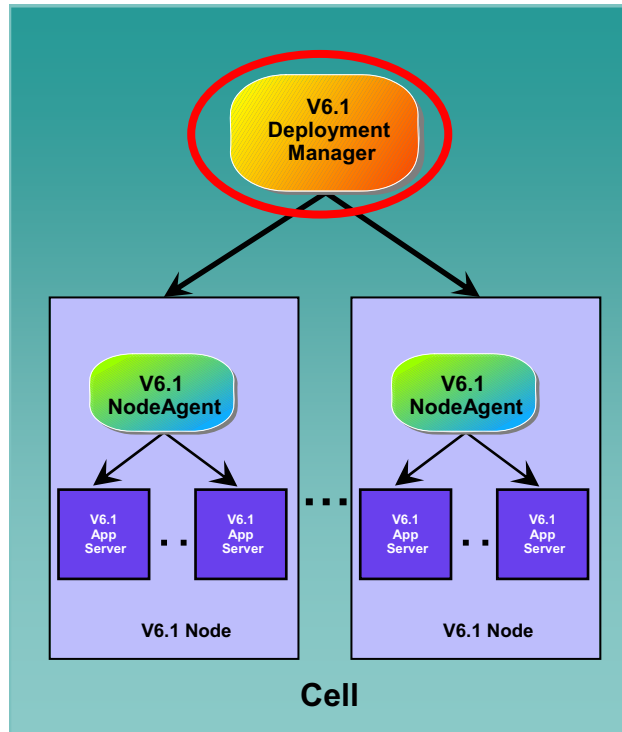
**Details —**

**Additional information —**

**Transition statement —** Next: Deployment manager profile

## Deployment manager profile

- Used to create a **deployment manager** process (Dmgr)
  - Can exist on independent machine
  - Can exist on machine with other profiles
  - Provides centralized administration of managed application server nodes and custom nodes as a single cell



© Copyright IBM Corporation 2007

Figure 13-10. Deployment manager profile

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

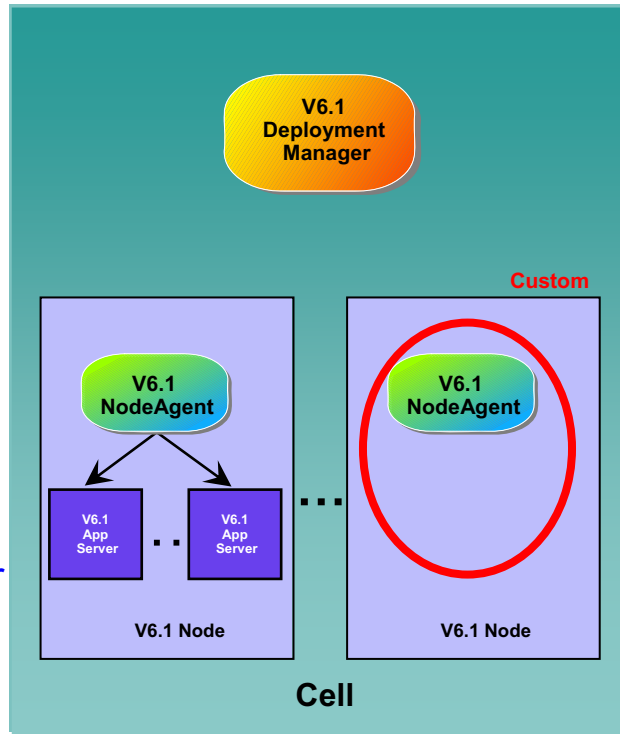
**Details —**

**Additional information —**

**Transition statement —** Next: Custom profile

## Custom profile

- A **custom profile** creates WebSphere node without any application
  - By default is automatically federated into a cell during profile creation
  - No application servers are created during profile creation
  - Use the deployment manager's administrative console to create servers and clusters on the federated node
  - Consider a custom profile as a production-ready shell, ready for customization to contain your servers and applications



© Copyright IBM Corporation 2007

Figure 13-11. Custom profile

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

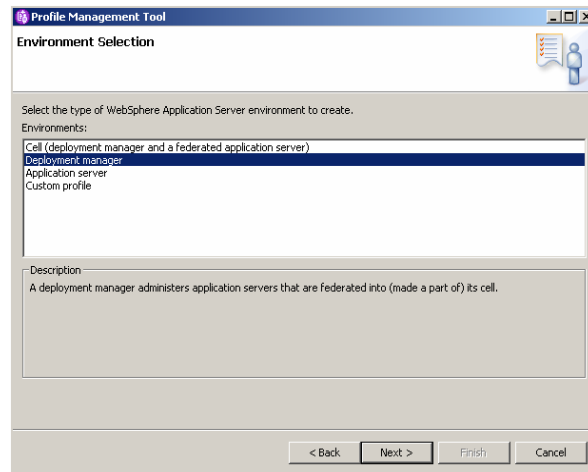
**Details —**

**Additional information —**

**Transition statement —** Next: Creating profiles

## Creating profiles

- WebSphere provides multiple ways of creating profiles
  - **Profile Management Tool**
    - Start menu (Windows only)
    - Launched after installation from install wizard
    - Launch via command-line tool pmt.bat
      - In `<was_root>\bin\ProfileManagement\`
      - Similar command exists for UNIX platform
    - Wizard in First steps console
  - Manually via the **manageprofiles** command-line tool
    - Create profiles in silent mode using `manageprofiles -silent` option
    - Other `manageprofiles` options include: `-listProfiles` `-delete`



```

C:\Program Files\IBM\WebSphere\AppServer\bin>manageprofiles.bat -create -profile
Name profile3 -profilePath "C:\Program Files\IBM\WebSphere\AppServer\profiles\pr
ofiles3" -templatePath "C:\Program Files\IBM\WebSphere\AppServer\profileTemplates
\default" -nodeName was61host88Node03 -cellName was61host88Cell103 -hostName was6
1host88
INSTCONFSUCCESS: Success: Profile profile3 now exists. Please consult C:\Program
Files\IBM\WebSphere\AppServer\profiles\profile3/logs/AboutThisProfile.txt for m
ore information about this profile.
C:\Program Files\IBM\WebSphere\AppServer\bin>
  
```

© Copyright IBM Corporation 2007

Figure 13-12. Creating profiles

WA3616.3

### Notes:

Profiles can be silently created:

- as part of a silent WebSphere installation process
- manually using the `-silent` option with the `pctWindow` utility

Silent creation of profiles requires a response file which should be customized:

- Dmgr: **responsefile.pct.NDdmgrProfile.txt**
- Application Server: **responsefile.pct.NDStand-aloneProfile.txt**
- Custom: **responsefile.pct.NDmanagedProfile.txt**



***Instructor notes:***

**Purpose —**

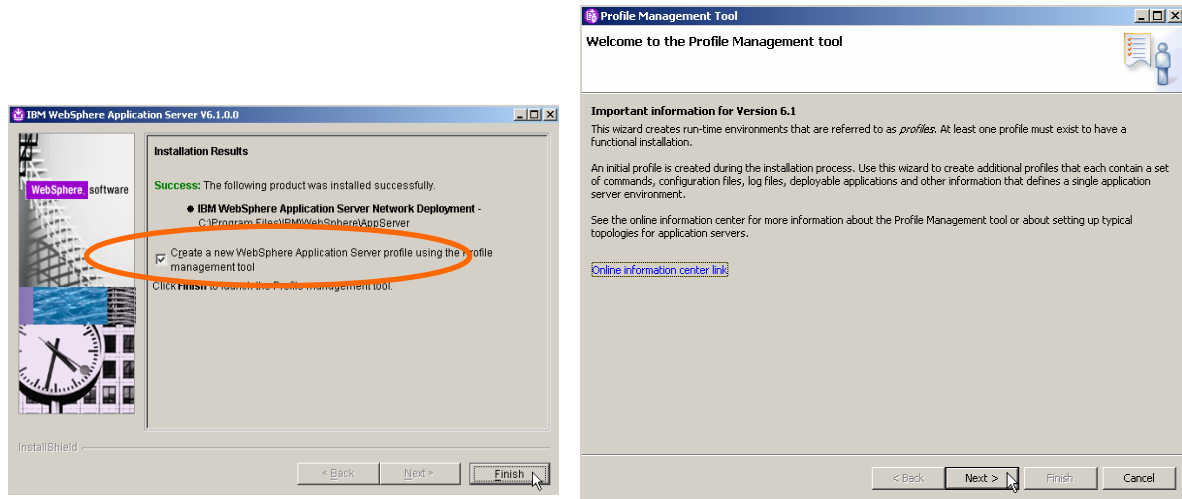
**Details —**

**Additional information —**

**Transition statement —** Next: Profile Management Tool (1 of 4)

## Profile Management Tool (1 of 4)

- Profile Management Tool can be started a number of ways:
  - Through the WebSphere installation wizard
  - Through the Windows Start menu
  - Through the command-line tools in the `<was_root>\bin\ProfileManagement` directory



© Copyright IBM Corporation 2007

Figure 13-13. Profile Management Tool (1 of 4)

WA3616.3

### Notes:

***Instructor notes:***

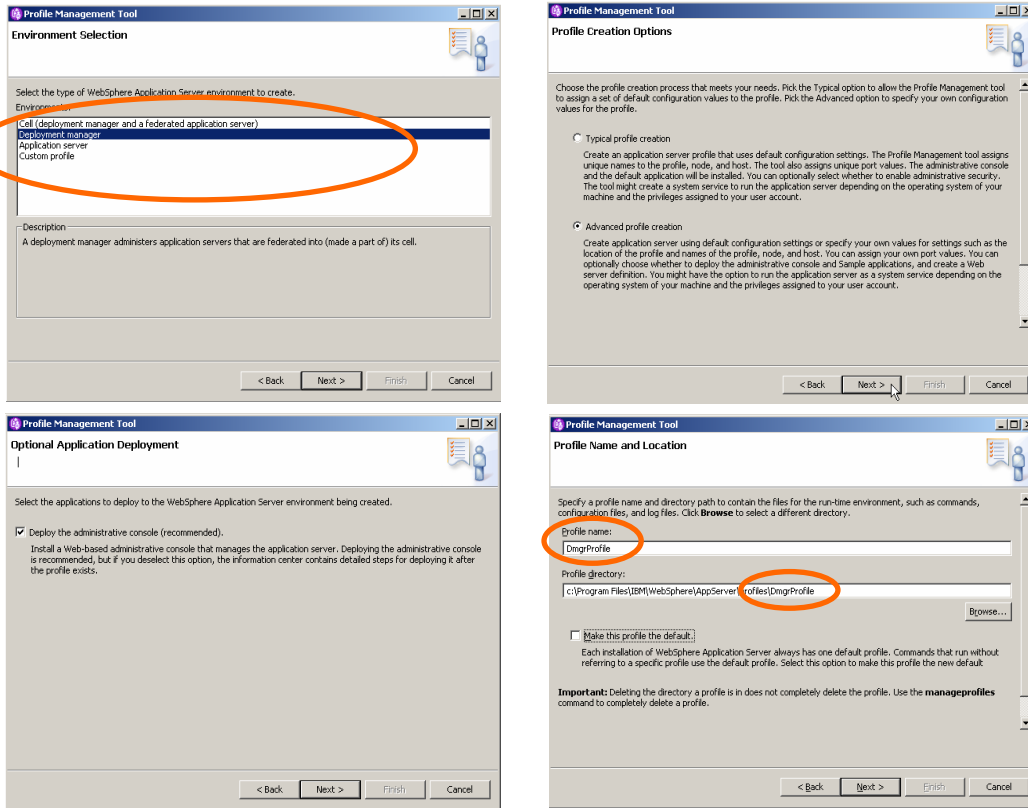
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Profile Management Tool (2 of 4)

# Profile Management Tool (2 of 4)



© Copyright IBM Corporation 2007

Figure 13-14. Profile Management Tool (2 of 4)

WA3616.3

## Notes:

***Instructor notes:***

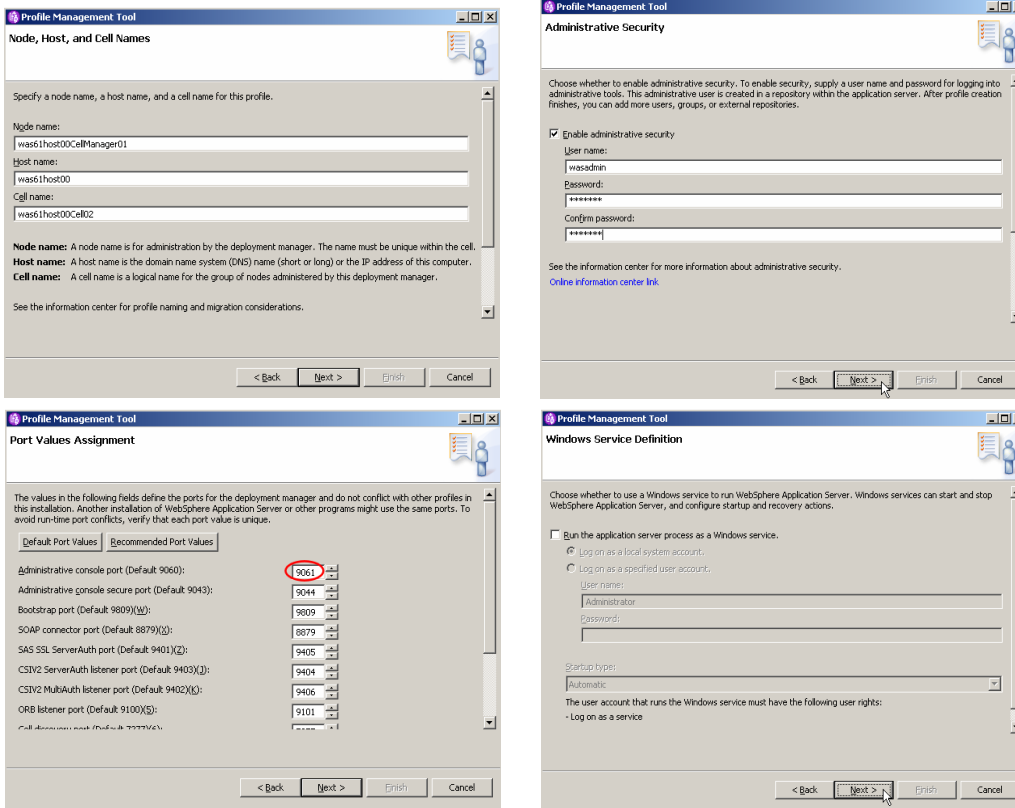
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Profile Management Tool (3 of 4)

# Profile Management Tool (3 of 4)



© Copyright IBM Corporation 2007

Figure 13-15. Profile Management Tool (3 of 4)

WA3616.3

## Notes:

***Instructor notes:***

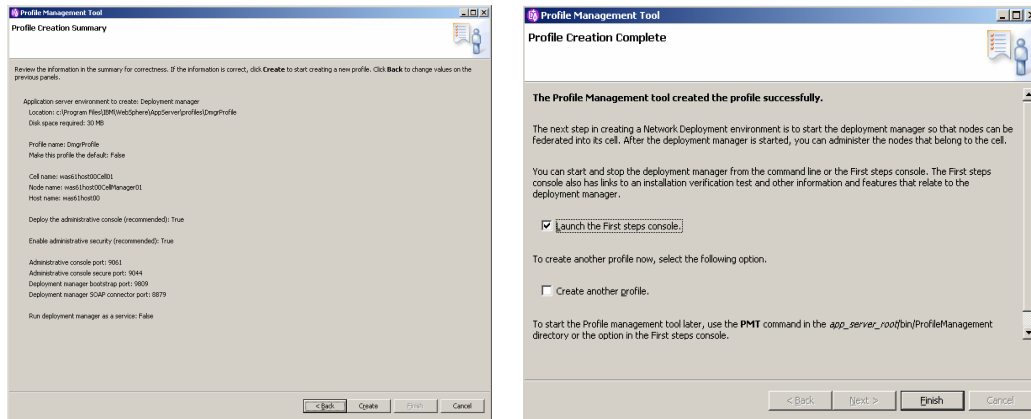
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Profile Management Tool (4 of 4)

# Profile Management Tool (4 of 4)



© Copyright IBM Corporation 2007

Figure 13-16. Profile Management Tool (4 of 4)

WA3616.3

## Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Profile Creation – Command-line tool

## Profile Creation – Command-line tool

- The **manageprofiles** script supports a number of functions including:
  - Creating a new stand-alone application server profile
    - manageprofiles –create
    - profileName
    - profilePath
    - templatePath
    - nodeName
    - cellName
    - hostname
  - Listing all profiles
    - manageprofiles –listProfiles
  - Deleting a profile
    - manageprofiles –delete –profileName
- If desired, delete the left over files in the profile directory

```

C:\Program Files\IBM\WebSphere\AppServer\bin>manageprofiles.bat -create -profileName profile3 -profilePath "C:\Program Files\IBM\WebSphere\AppServer\profiles\profile3" -templatePath "C:\Program Files\IBM\WebSphere\AppServer\profileTemplates\default" -nodeName was61host88Node03 -cellName was61host88Cell03 -hostname was61host88
INSTCONFSUCCESS: Success: Profile profile3 now exists. Please consult C:\Program Files\IBM\WebSphere\AppServer\profiles\profile3/logs/AboutThisProfile.txt for more information about this profile.
C:\Program Files\IBM\WebSphere\AppServer\bin>
  
```

© Copyright IBM Corporation 2007

Figure 13-17. Profile Creation – Command-line tool

WA3616.3

### Notes:

Deleting a profile leaves a number of files behind, including the logs directory. If desired, these will need to be deleted manually.

The list of profiles and their properties can be found in `<was_root>\properties\profileregistry.xml`

Additional properties, such as log levels, can be found in `<was_root>\properties\wasprofile.properties`

***Instructor notes:***

**Purpose —**

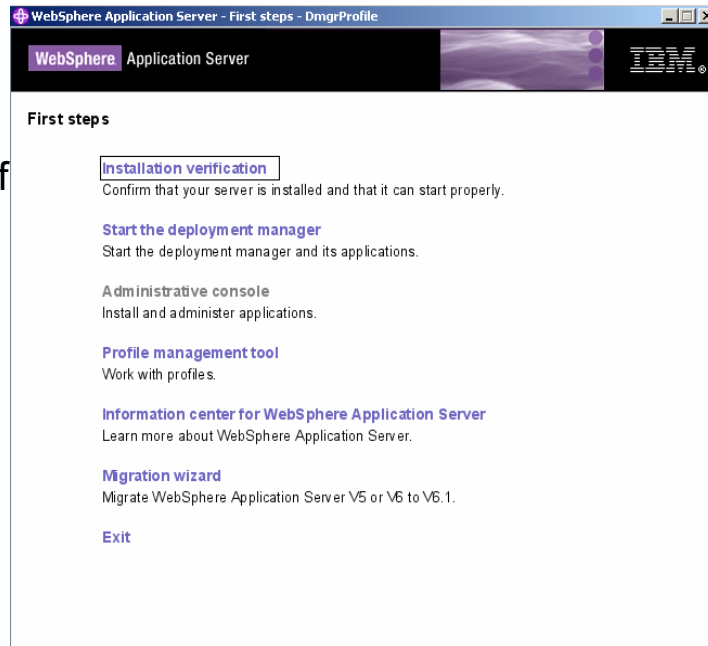
**Details —**

**Additional information —**

**Transition statement —** Next: First steps

## First steps

- Is a post-installation ease-of-use tool
- By default, starts automatically at the end of the wizard based installation
- Can be started via Windows Start menu or command line
- Can be used to verify installation
- One per profile



© Copyright IBM Corporation 2007

Figure 13-18. First steps

WA3616.3

### Notes:

*First steps* is a post-installation ease-of-use tool for directing WebSphere Application Server elements from one place. Options dynamically appear on the First steps panel, depending on features you install.

***Instructor notes:***

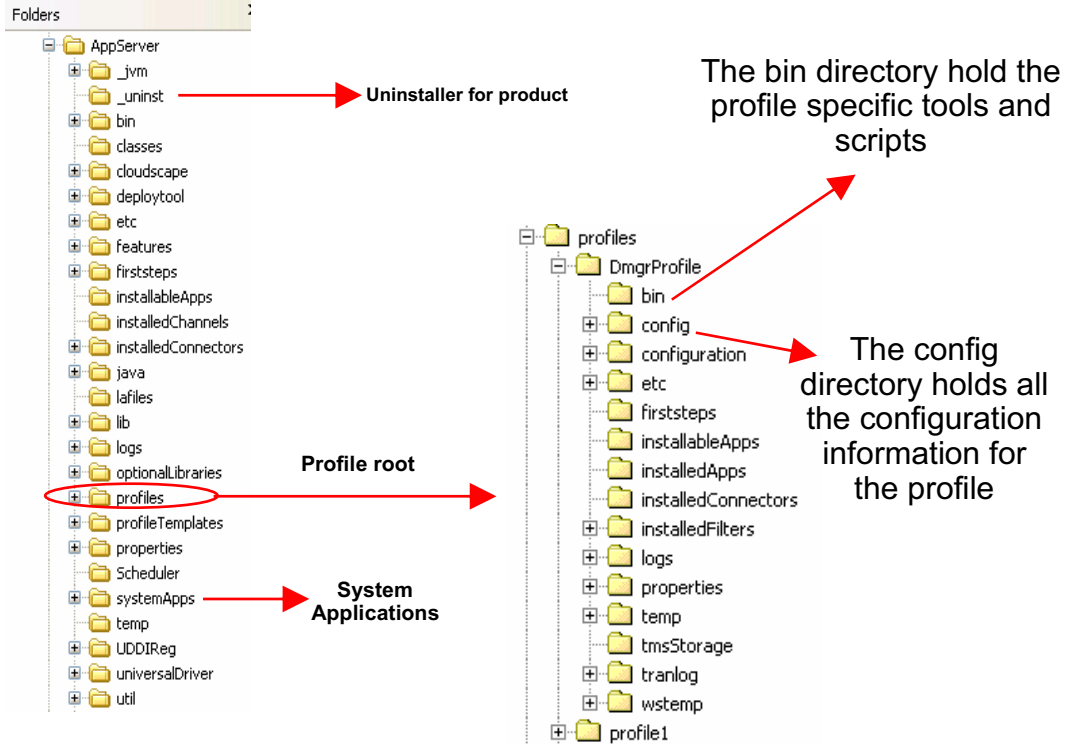
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Directory structure

# Directory structure



© Copyright IBM Corporation 2007

Figure 13-19. Directory structure

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Server commands

## Server commands

---

- WebSphere commands are profile aware
  - There is a `-profile` option on many WebSphere commands
  - Or issue the commands from the appropriate `<profile_root>\<profile_name>\bin` directory
- If no profile is used, the default profile is assumed
  - There can only be one default profile
  - Unless otherwise manually set, the first profile created will be the default
- Examples (from `<was_root>\bin`):

```
startServer server1 -profileName profile1
startManager -profileName DMgr
stopServer server1 ← assumes default profile
```

© Copyright IBM Corporation 2007

Figure 13-20. Server commands

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Common command-line tools

## Common command-line tools

---

- Tools in <was\_root>\bin directory (as well as the profile's bin directory)
  - addNode – Add a node to a cell
  - syncNode – Synchronize a node with the cell configuration
  - removeNode – Remove a node from a cell
  - startNode – Start the node agent
  - stopNode – Stop the node agent
  - startManager – Start the deployment manager
  - stopManager – Stop the deployment manager

© Copyright IBM Corporation 2007

Figure 13-21. Common command-line tools

WA3616.3

### **Notes:**

These are tools that are commonly used in a cell environment.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere profiles Issues

# WebSphere profiles Issues

- When multiple profiles are created on a single machine, be careful
  - Use correct profile bin directory to
    - startServer
    - stopServer
    - serverStatus
  - Be aware of possible port conflicts for node agents and application servers
    - Make sure all of the servers have unique ports
  - Be aware that there may be multiple server1 instances on a single machine (as part of different profiles)
  - Ensure that consistent hostnames within a machine are used
    - was61host00 and was61host00.ibm.com are different

Communications

Ports

Port Name	Port	Details
BOOTSTRAP_ADDRESS	9810	
SOAP_CONNECTOR_ADDRESS	8880	
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS	9401	
CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS	9403	
CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS	9402	
WC_adminhost	9060	
WC_defaulthost	9080	
DCS_UNICAST_ADDRESS	9353	
WC_adminhost_secure	9043	
WC_defaulthost_secure	9443	
SIP_DEFAULTHOST	5060	
SIP_DEFAULTHOST_SECURE	5061	
SIB_ENDPOINT_ADDRESS	7276	
SIB_ENDPOINT_SECURE_ADDRESS	7286	
SIB_MQ_ENDPOINT_ADDRESS	5558	
SIB_MQ_ENDPOINT_SECURE_ADDRESS	5578	
ORB_LISTENER_ADDRESS	9100	

© Copyright IBM Corporation 2007

Figure 13-22. WebSphere profiles Issues

WA3616.3

## Notes:

Be careful when using the Profile Management Tool. It is possible that it will preload a default host name by adding the default DNS suffix to the short machine name. This can cause problems if other profiles used only the short host name. It doesn't matter which form is used (shortname or fully qualified name), as long as the name is used consistently.

Example: was61host00 and was61host00.ibm.com are different

***Instructor notes:***

**Purpose —**

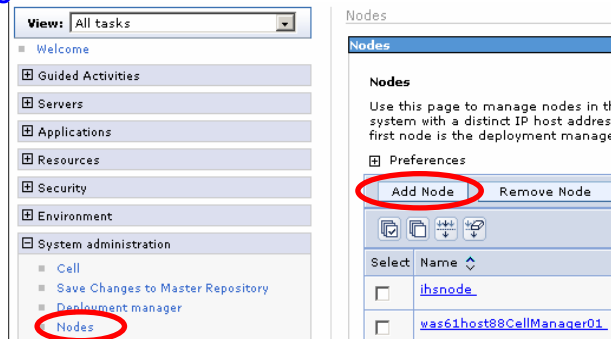
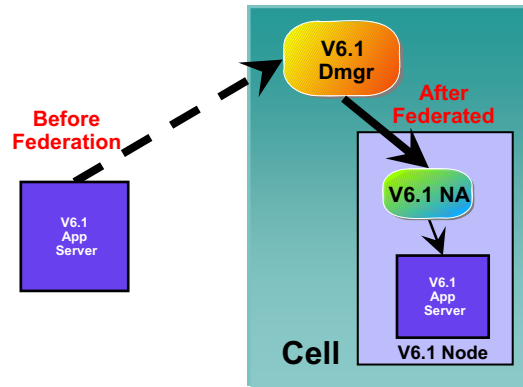
**Details —**

**Additional information —**

**Transition statement —** Next: Adding a WebSphere node to a cell

## Adding a WebSphere node to a cell

- Adding a node to a cell can be done using:
  - **Administrative console**
    - System Administration → Nodes → Add Nodes
  - **addNode** command-line tool
- The process of adding a node to the cell:
  - Creates backup of current configuration
  - Connects to deployment manager
  - Configures node agent
  - Adds node's applications to cell configuration if the **-includeapps** option is used
    - For example,
      - **addNode mydmgr 8879 -includeapps**
- After the node has been added
  - Use **startNode** to start the node agent
  - Use **syncNode** to synchronize a node with the cell configuration



© Copyright IBM Corporation 2007

Figure 13-23. Adding a WebSphere node to a cell

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Cell topology

## Cell topology

- Cell topology can be viewed through the administrative console
  - From **System Administration** → **Cell** → **Local Topology**



© Copyright IBM Corporation 2007

Figure 13-24. Cell topology

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

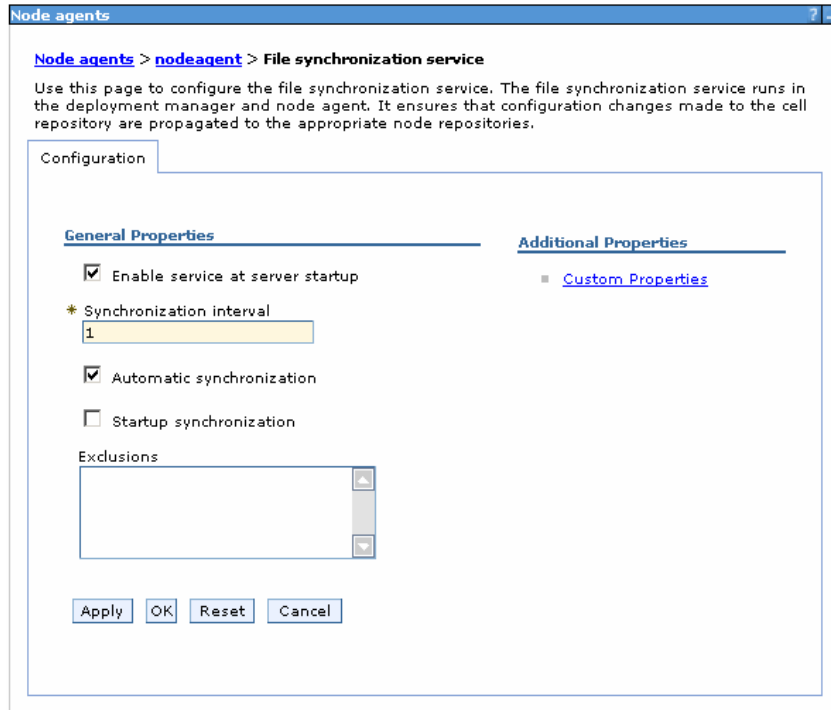
**Details —**

**Additional information —**

**Transition statement —** Next: Configuring synchronization

## Configuring synchronization

- From a node agent's detail page select **File Synchronization Service**



© Copyright IBM Corporation 2007

Figure 13-25. Configuring synchronization

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Remove WebSphere node from a cell

## Remove WebSphere node from a cell

---

- Use **removeNode** (at node) to remove a node from a cell
  - Restores backup of node's stand-alone configuration
  - Can be run through the administrative console, **Nodes** page
    - Remotely executes removeNode
- Use **cleanupNode** to clean up node configuration from cell repository
  - Only use this command to clean up a node if you have a node defined in the cell configuration, but the node no longer exists
  - Can be run through the administrative console, **Nodes** page with the **Force Delete** button
    - Force Delete action is equivalent to running the cleanupNode command at the deployment manager.

© Copyright IBM Corporation 2007

Figure 13-26. Remove WebSphere node from a cell

WA3616.3

### **Notes:**

When a node is removed from a cell, the profile reverts back to the configuration it had when it was initially federated into a cell. Any applications or configuration changes that were made while it was part of a cell are lost.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Checkpoint questions

## Checkpoint questions

---

1. Which managed processes can be part of a cell?
  - A. Deployment Manager
  - B. Node Agent
  - C. Load Balancer
  - D. Application Server
  
2. Which profiles can be created using the Profile Management Tool?
  - A. Load Balancer profile
  - B. Custom profile
  - C. Plug-in profile
  - D. IBM HTTP Server profile
  
3. All application servers have a corresponding node agent?
  - A. True
  - B. False

© Copyright IBM Corporation 2007

Figure 13-27. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.

**Instructor notes:**

Purpose —

Details —

## Checkpoint solutions

---

1. Which managed processes can be part of a cell?
  - A. Deployment Manager
  - B. Node Agent
  - D. Application Server
2. Which profiles can be created using the Profile Management Tool?
  - B. Custom profile
3. All application servers have a corresponding node agent?
  - B. False

© Copyright IBM Corporation 2007

**Additional information** —**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe WebSphere Application Server cell concepts
- Describe the installation process for creating profiles
- Verify the installation of profiles
- Describe the directories and configuration files for WebSphere profiles
- Add a WebSphere Application Server node to a cell
- Explain the difference between managing a stand-alone server and a cell

© Copyright IBM Corporation 2007

Figure 13-28. Unit summary

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Lab exercise

## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 13-29. Lab exercise

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**



---

# Unit 14. Workload management

## Estimated time

01:00

## What this unit is about

This unit describes workload management principles and how WebSphere Application Server can be configured to participate in workload management.

## What you should be able to do

After completing this unit, you should be able to:

- Define workload management
- Create clusters and cluster members
- Compare clustered configurations
- Explain how weights are used in workload management
- Discuss failover scenarios
- Describe the role of the HTTP plug-in in workload management
- Explain session management
- Configure distributed session management

## How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

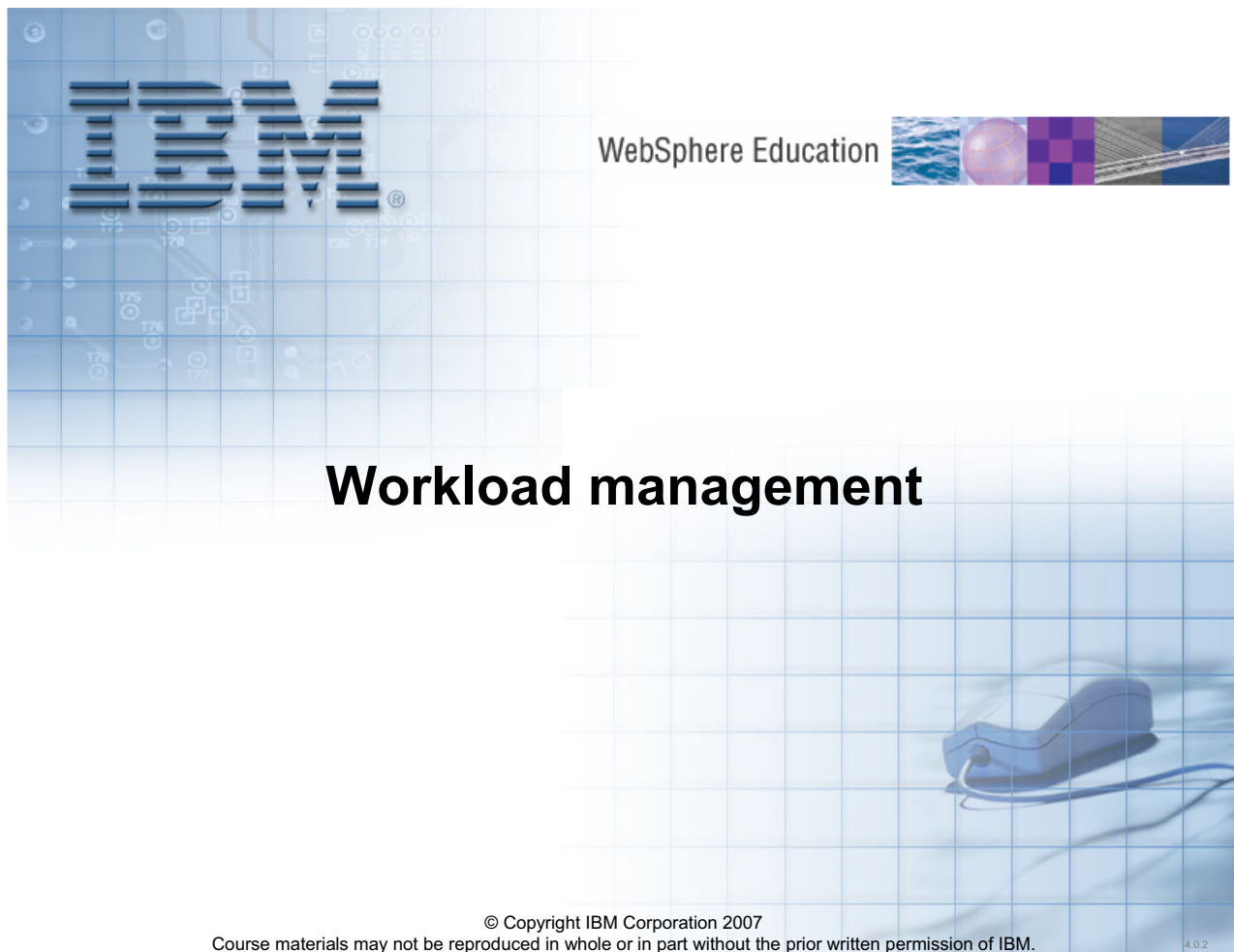


Figure 14-1. Workload management

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Define workload management
- Create clusters and cluster members
- Compare clustered configurations
- Explain how weights are used in workload management
- Discuss failover scenarios
- Describe the role of the HTTP plug-in in workload management
- Explain session management
- Configure distributed session management

© Copyright IBM Corporation 2007

Figure 14-2. Unit objectives

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: What is workload management (WLM)?

## What is workload management (WLM)?

---

- Sharing requests across multiple application servers
- Configuration options that improve:
  - Performance – improve response time for requests
  - Scalability – grow capacity as the number of users increases
  - Load balancing – allocate workload proportionately among available resources
  - Availability – applications are still available if a server fails

© Copyright IBM Corporation 2007

Figure 14-3. What is workload management (WLM)?

WA3616.3

### **Notes:**

***Instructor notes:***

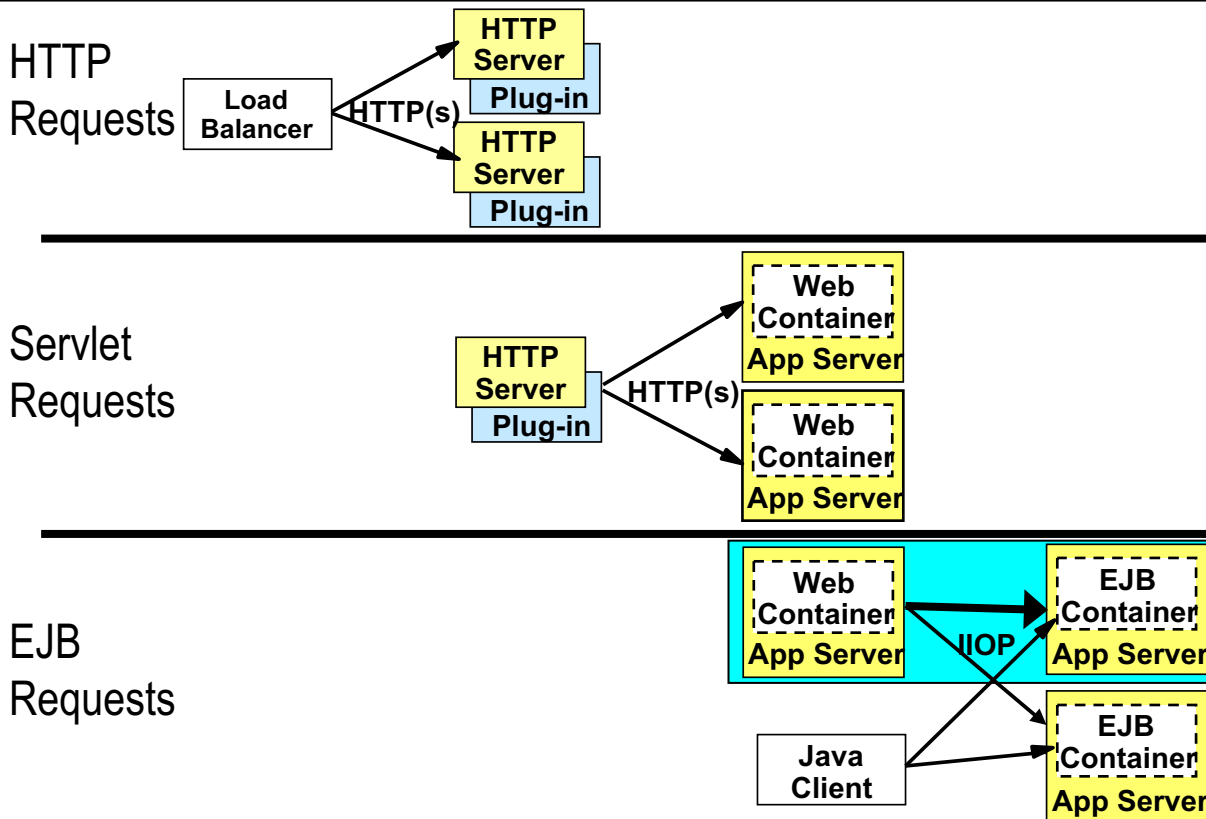
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: What can be workload managed?

# What can be workload managed?



© Copyright IBM Corporation 2007

Figure 14-4. What can be workload managed?

WA3616.3

## Notes:

***Instructor notes:***

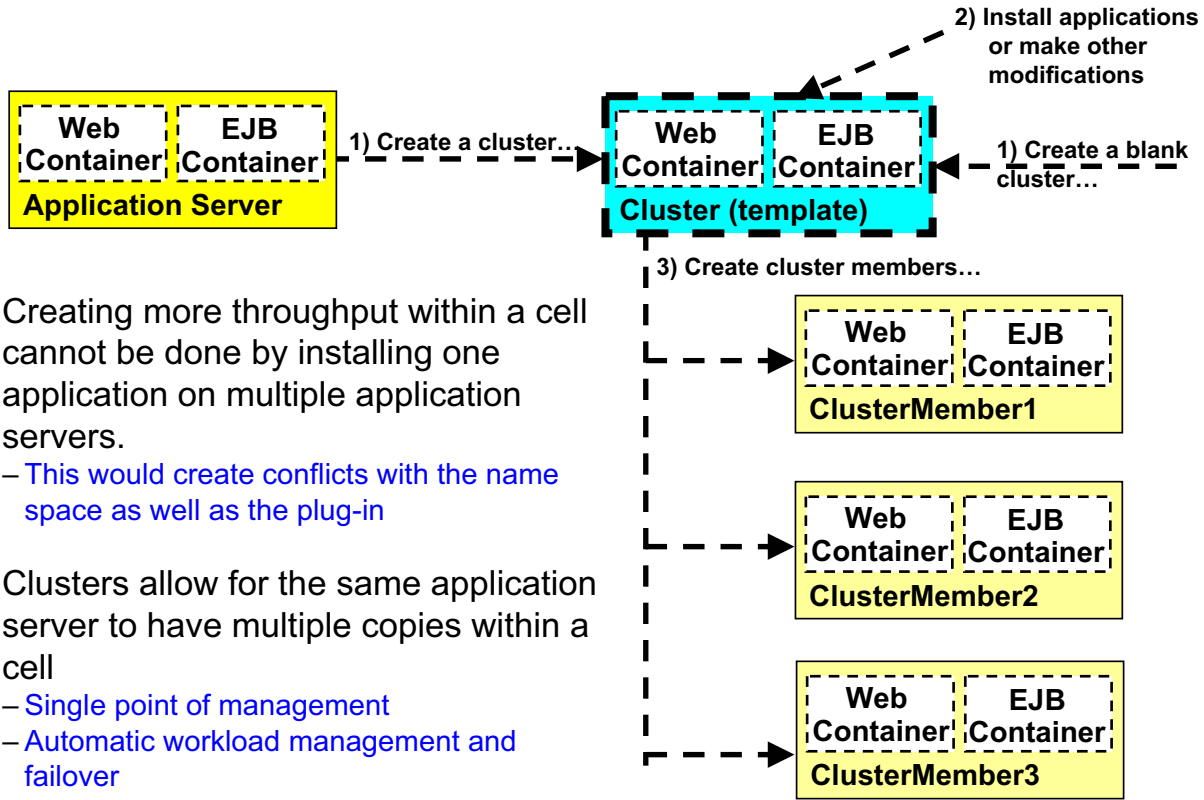
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Clusters

# Clusters



- Creating more throughput within a cell cannot be done by installing one application on multiple application servers.
  - This would create conflicts with the name space as well as the plug-in
- Clusters allow for the same application server to have multiple copies within a cell
  - Single point of management
  - Automatic workload management and failover

© Copyright IBM Corporation 2007

Figure 14-5. Clusters

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Clusters and cluster members

## Clusters and cluster members

---

- Clusters are a set of application servers having the same applications installed, and grouped logically for workload management
  - Applications installed to the cluster are automatically distributed to the cluster members
  
- Creation of a cluster
  - Can use an existing server to become the first cluster member
  - Additional cluster members are created from templates
  
- Cluster members are similar to 'clones' in previous WebSphere Application Server versions in that they:
  - Run the same applications
  - Share workload
  - Can be centrally administered

© Copyright IBM Corporation 2007

Figure 14-6. Clusters and cluster members

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

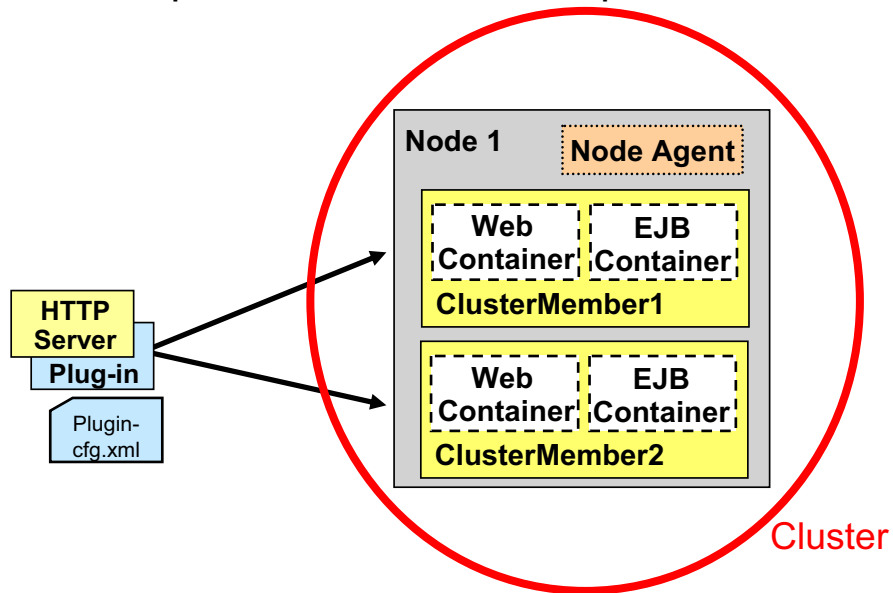
**Details —**

**Additional information —**

**Transition statement —** Next: Configurations - Vertical scaling

## Configurations: Vertical scaling

- May provide better performance with multiple CPUs



© Copyright IBM Corporation 2007

Figure 14-7. Configurations: Vertical scaling

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

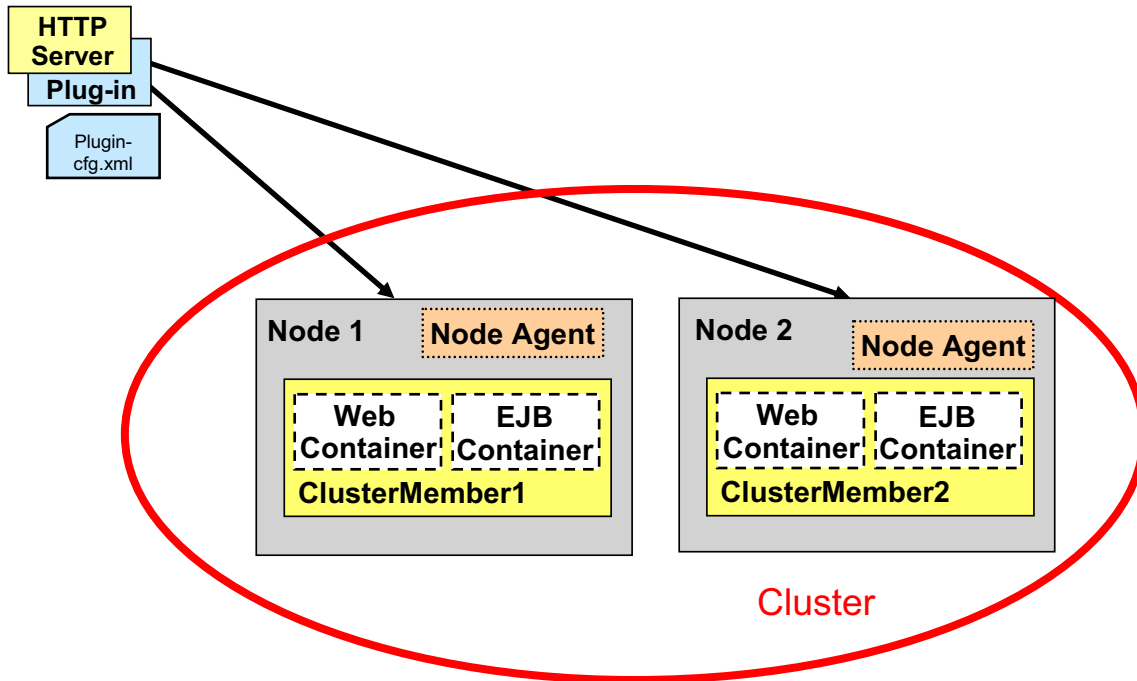
**Details —**

**Additional information —**

**Transition statement —** Next: Configurations - Horizontal scaling

## Configurations: Horizontal scaling

- Supports machine failover



© Copyright IBM Corporation 2007

Figure 14-8. Configurations: Horizontal scaling

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

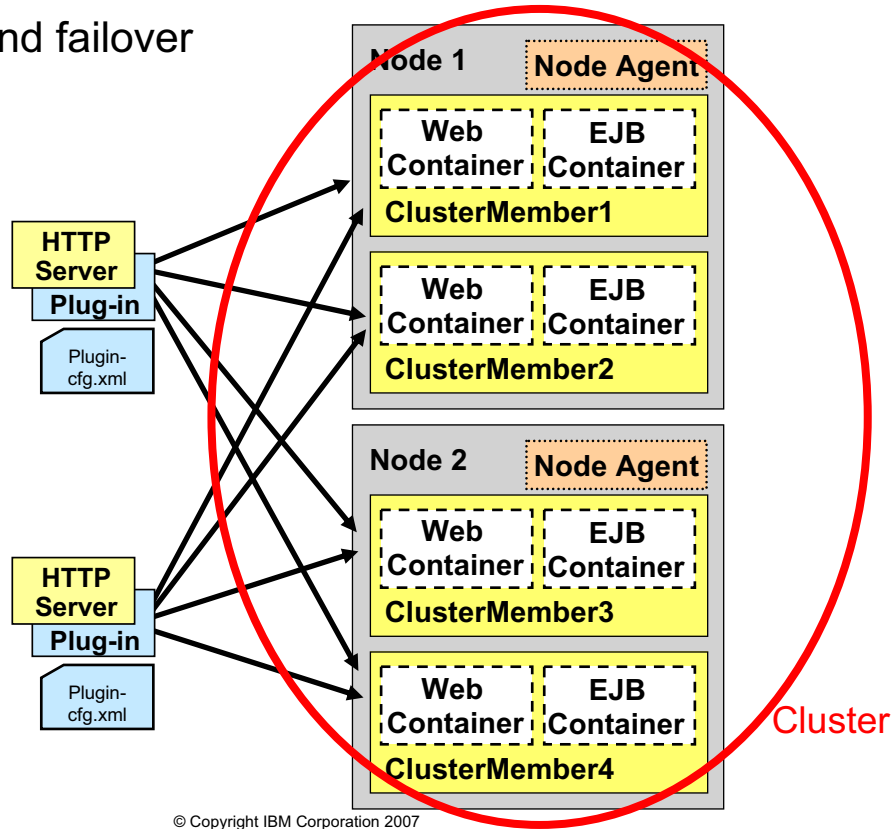
**Details —**

**Additional information —**

**Transition statement —** Next: Configurations: Vertical and horizontal scaling

# Configurations: Vertical and horizontal scaling

- Performance and failover



© Copyright IBM Corporation 2007

Figure 14-9. Configurations: Vertical and horizontal scaling

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Creating a cluster (1 of 3)

## Creating a cluster (1 of 3)

- In console, select **Servers** → **Clusters** and click **New**
- Enter Cluster name
- Check options
  - Prefer local
  - Configure HTTP session memory to memory replication
- Click **Next**

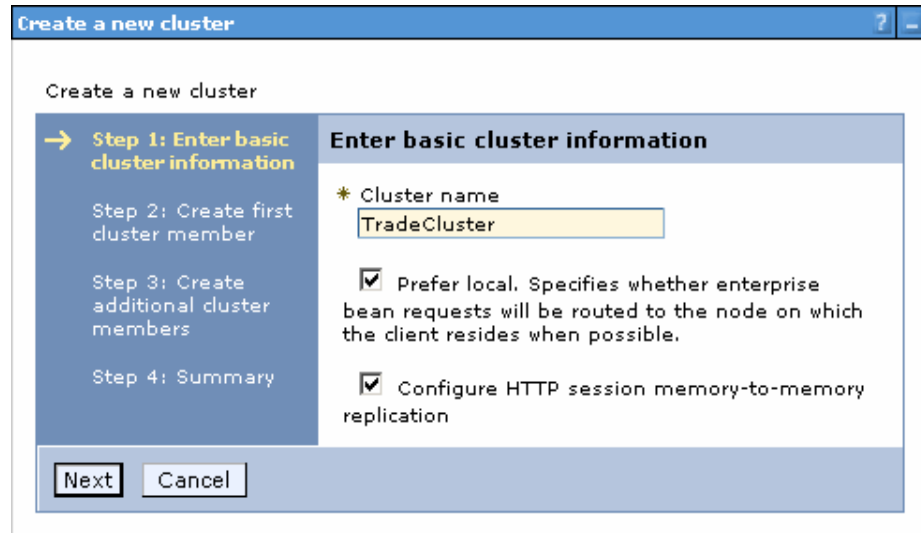


Figure 14-10. Creating a cluster (1 of 3)

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Creating a cluster (2 of 3)

## Creating a cluster (2 of 3)

- Create the first cluster member based on:

- A server template
- An existing server
- Converting an existing server
- An empty cluster

- Enter Member name

- Select node

- Enter weight

- Click **Next**

Create a new cluster

Step 1: Enter basic cluster information

→ Step 2: Create first cluster member

Step 3: Create additional cluster members

Step 4: Summary

### Create first cluster member

The first cluster member determines the server settings for the cluster members. A server configuration template is created from the first member and stored as part of the cluster data. Additional cluster members are copied from this template.

\* Member name  
server1

Select node  
was61host00Node01(ND 6.1.0.0)

\* Weight  
2 (0..20)

Generate unique HTTP ports

Select basis for first cluster member:

Create the member using an application server template.  
default

Create the member using an existing application server as a template.  
was61host00Cell01/was61host00Node01(ND 6.1.0.0)/server1

Create the member by converting an existing application server.  
was61host00Cell01/was61host00Node01(ND 6.1.0.0)/server1

None. Create an empty cluster.

Previous Next Cancel

© Copyright IBM Corporation 2007

Figure 14-11. Creating a cluster (2 of 3)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Creating a cluster (3 of 3)

## Creating a cluster (3 of 3)

- Enter name
- Select node
- Set weight
- Check options:
  - Generate Unique HTTP ports (default)
- Click **Add Member**
- Repeat to create other cluster members
- Click **Next**
- Click **Finish** on Summary screen

Create a new cluster

Step 1: Enter basic cluster information

Step 2: Create first cluster member

→ Step 3: Create additional cluster members

Step 4: Summary

### Create additional cluster members

Enter information about this new cluster member, and click Add Member to add this cluster member to the member list. A server configuration template is created from the first member and stored as part of the cluster data. Additional cluster members are copied from this template.

\* Member name

Select node

\* Weight  
 (0..20)

Generate unique HTTP ports

Use the Edit function to edit the properties of a cluster member that is already included in this list. Use the Delete function to remove a cluster member from this list. You are not allowed to edit or remove the first cluster member or an already existing cluster member.

Select	Member name	Nodes	Version	Weight
<input checked="" type="checkbox"/>	server1	was61host00Node01	ND 6.1.0.0	2

© Copyright IBM Corporation 2007

Figure 14-12. Creating a cluster (3 of 3)

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

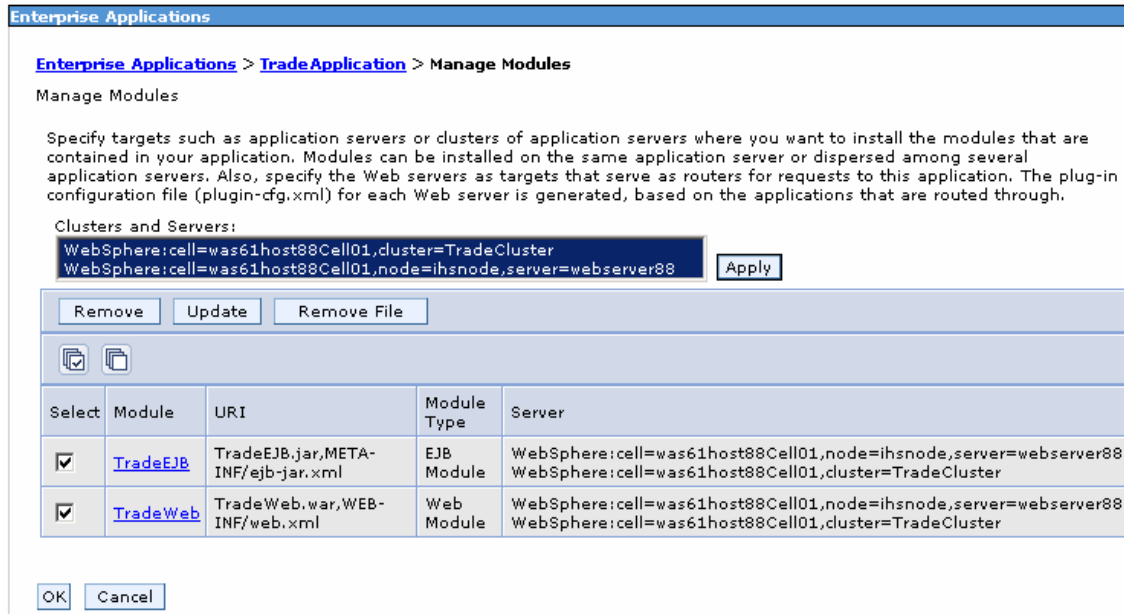
**Details** —

**Additional information** —

**Transition statement** — Next: Installing enterprise applications to a cluster

# Installing enterprise applications to a cluster

- Same steps as installing to base server except:
  - Select a cluster as the target, rather than a server
  - And possibly map to Web servers as well



© Copyright IBM Corporation 2007

Figure 14-13. Installing enterprise applications to a cluster

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Controlling a cluster

## Controlling a cluster

- Start the cluster
  - Start starts all servers together
  - Ripplestart starts servers one at a time
  
- Status
  - Started – all servers are started
  - Partial Start – some servers are started
  - Stopped – all the servers are stopped
  - Partial Stop – some servers are stopped



© Copyright IBM Corporation 2007

Figure 14-14. Controlling a cluster

WA3616.3

### Notes:

If ripplestart is used on a running cluster, each application server in the cluster is stopped one at a time, and restarted.

The screen capture shows a partially stopped cluster.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Modification of clusters

## Modification of clusters

---

- Use the administrative console or wsadmin
  
- What can be changed in a cluster?
  - Cluster member settings:
    - Weights
    - Prefer local
  - Install or update applications
  - Application server settings
    - Cluster members are application servers and the normal application server settings can be modified
  
- After making changes:
  - Save the configuration and resynchronize
  - Regenerate the HTTP Server plug-in and redistribute it if necessary

© Copyright IBM Corporation 2007

Figure 14-15. Modification of clusters

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

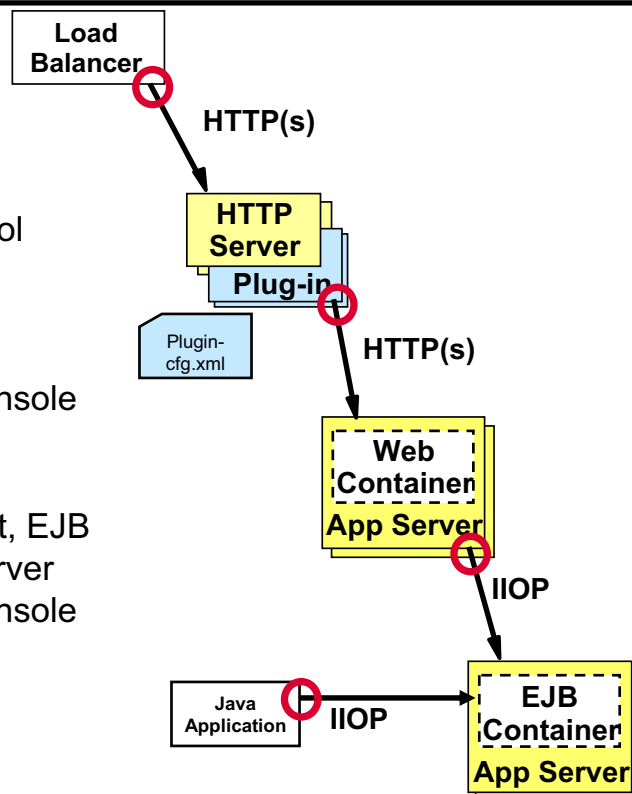
**Details —**

**Additional information —**

**Transition statement —** Next: Basic WLM request routing

## Basic WLM request routing

- Routing decision points
  - Load Balancer
    - Routing decision table stored internally
    - Configured using Load Balancer tool
    - Multiple intelligent routing options
  - HTTP Server plug-in
    - Routing table part of plugin-cfg.xml
    - Configured using administrative console or wsadmin scripting tool
  - WLM-aware client
    - Includes Web container, Java client, EJB
    - Routing table supplied by name server
    - Configured using administrative console or wsadmin scripting tool
    - Options:
      - Prefer local - yes or no



© Copyright IBM Corporation 2007

Figure 14-16. Basic WLM request routing

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

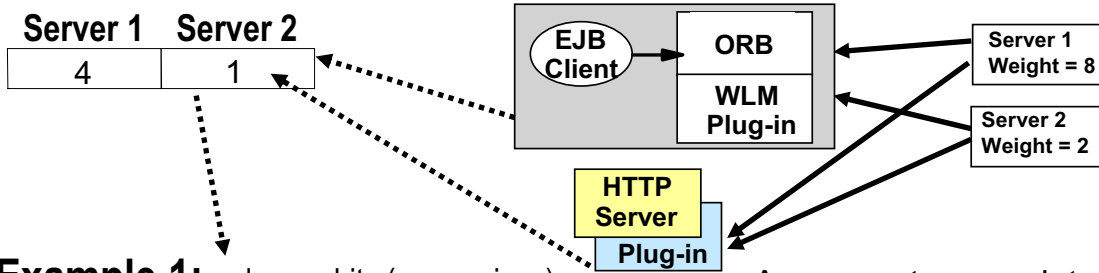
**Additional information —**

**Transition statement —** Next: Weighted routing example – No affinity

# Weighted routing example with no affinity

## Routing Table Initial Values\*

\* Plug-in reduces to lowest common denominator. EJB WLM uses two tables for prefer local and remote



### Example 1: only new hits (no sessions)

After:	Server 1	Server 2
request 0	4	1
request 1	3	1
request 2	3	0
request 3	2	0
request 4	1	0
request 5	0	0
Reset:	4	1

Once all values =0 a reset is done

Reset adds original values of 4 and 1

- As requests come into the plug-in, round-robin is used to distribute the hits
- Each request decrements the value in the internal table
- Once a server has a count of zero, no new hits are sent to that server
- Once all servers have a value of zero, the values are all reset

© Copyright IBM Corporation 2007

Figure 14-17. Weighted routing example with no affinity

WA3616.3

### Notes:

This algorithm is used for both the plug-in and ORB WLM.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

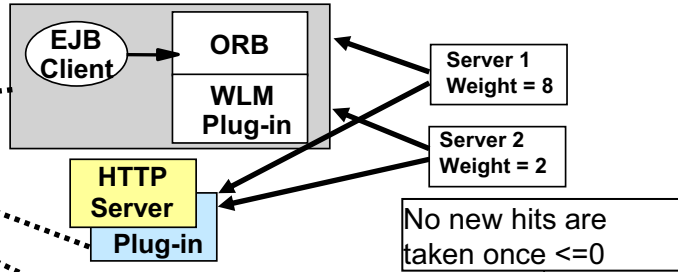
**Transition statement —** Next: Weighted routing example – With affinity

# Weighted routing example with affinity

## Routing Table Initial Values\*

Server 1	Server 2
4	1

\* Plug-in reduces to lowest common denominator. EJB WLM uses two tables for prefer local and remote



- Just like the previous example except:
  - Requests with sessions go to the servers that have affinity
  - If a server has a value of zero or less and it receives a request that has affinity, the value is still decremented and goes into negative number
  - Once all servers have a value of zero or less, the initial values are added to the current values.

## Example 2: old hits with sessions\*\*

Forced hits due to sticky sessions\*\*

Once all values <=0 a reset is done

After:	Server 1	Server 2
request 0	4	1
request 1	3	1
request 2**	2	1
request 3	2	0
request 4**	2	-1
request 5	1	-1
request 6**	1	-2
request 7	0	-2
Reset:	4	-1

Reset adds original values of 4 and 1

© Copyright IBM Corporation 2007

Figure 14-18. Weighted routing example with affinity

WA3616.3

### Notes:

If it is possible for the distribution of hits to become unbalanced. This could happen, for example, if one server happens to get a bunch of requests that require affinity. In this case, the server in question could be doing more work than intended. This algorithm will correct this unbalance by preventing that server from taking any new hits until such a time that it has rebalanced itself relative to the work done by the other servers. So, even through over a short period of time an imbalance can occur, the system will automatically correct itself while still ensuring that requests with affinity go to the right server.



***Instructor notes:***

**Purpose —**

**Details —**

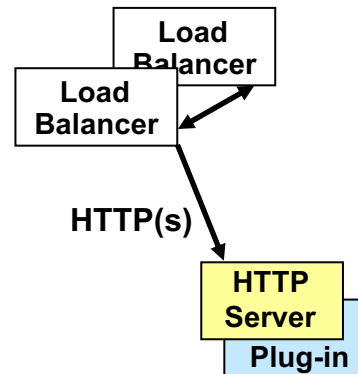
**Additional information —**

**Transition statement —** Next: Edge components failover

## Edge components failover

---

- Load Balancer can be paired with a backup machine
- Topology is active/standby
  - One machine does all the work
  - The other waits for a failure to begin handling routing



© Copyright IBM Corporation 2007

Figure 14-19. Edge components failover

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

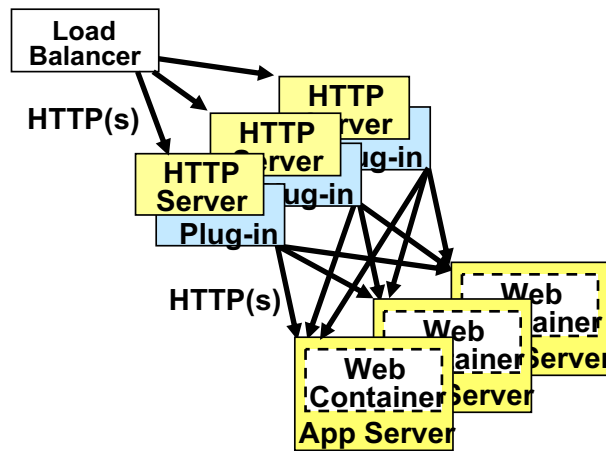
**Details —**

**Additional information —**

**Transition statement —** Next: HTTP Server failover

# HTTP Server failover

- Multiple HTTP servers provide coverage
- Load Balancer can route around a failed HTTP server
- HTTP plug-in
  - Every plug-in knows about all Web containers
  - Session key contains address of server
  - Sessions get properly routed
- All HTTP servers handle load before failover



© Copyright IBM Corporation 2007

Figure 14-20. HTTP Server failover

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

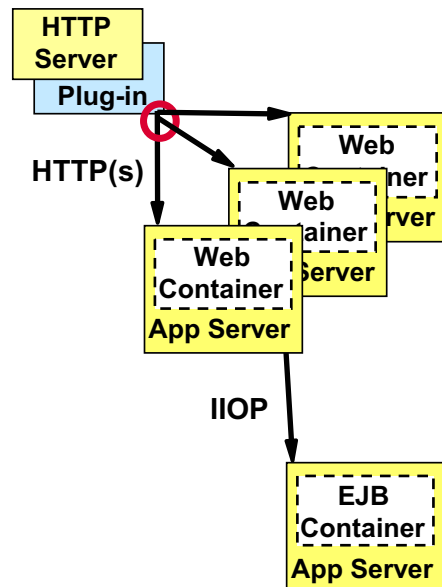
**Details —**

**Additional information —**

**Transition statement —** Next: Web container failover

## Web container failover

- HTTP server plug-in
  - Detects failure of Web containers
  - Marks container as unavailable
  - Tries next cluster member in the cluster
  
- What about in-flight sessions?
  - Sessions may be persisted to database
  - Sessions may be replicated in memory



© Copyright IBM Corporation 2007

Figure 14-21. Web container failover

WA3616.3

### Notes:

If a plug-in detects that a Web container is unavailable, it will mark that container as down and will not make any additional attempts to communicate with it for two minutes. Once the two minutes is over, it will again attempt to communicate with that server. If it is still unavailable, it will again be marked down for two minutes and continue the process.

This algorithm allows each plug-in to dynamically deal with Web containers that either become unexpectedly unavailable or taken offline. At the same time, when new cluster members are added, and the plugin-cfg.xml is generated and propagated, the plug-in will automatically be aware of the new containers and dynamically add them to the pool.

***Instructor notes:***

**Purpose —**

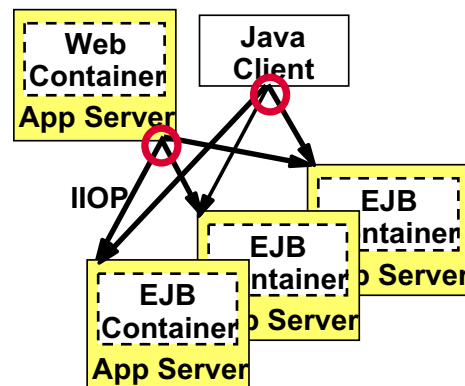
**Details —**

**Additional information —**

**Transition statement —** Next: EJB container failover

## EJB container failover

- Client code and ORB plug-in can route to next available cluster member
- Failure occurs before any work was initiated on the cluster member:
  - ORB automatically reroutes EJB request
  - If no other cluster member available, throws `NO_IMPLEMENT` exception
- Failure occurs after EJB method call initiated work:
  - System exceptions are thrown
  - Client needs to determine appropriate recovery action
    - Re-issue request, rollback transaction, and so forth
  - If `NO_IMPLEMENT` exception thrown, no recovery is possible



© Copyright IBM Corporation 2007

Figure 14-22. EJB container failover

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

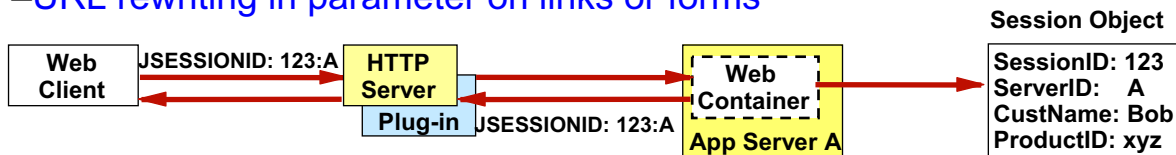
**Details —**

**Additional information —**

**Transition statement —** Next: HTTP session management

## HTTP session management

- HTTP is a stateless protocol
- Sessions allow you to maintain state information across a series of HTTP requests from the same client
  - For example, maintain shopping cart until checkout
- Java servlet specification defines session management process for Web applications
- Session manager stores session information, and sends client a unique session ID through:
  - Cookie in HTTP header, or
  - URL rewriting in parameter on links or forms



© Copyright IBM Corporation 2007

Figure 14-23. HTTP session management

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

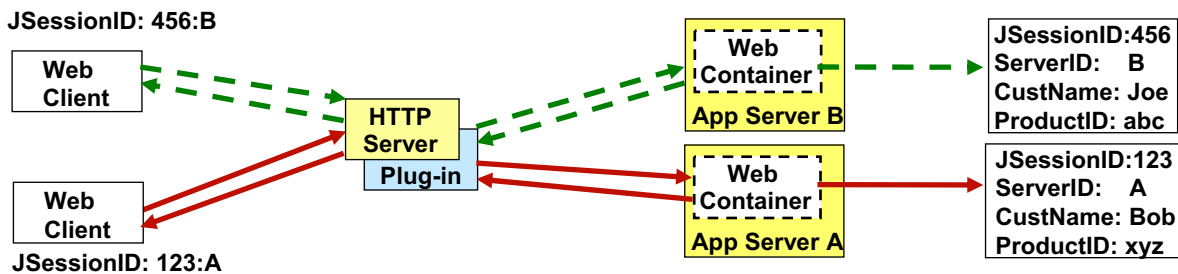
**Details —**

**Additional information —**

**Transition statement —** Next: Session affinity

## Session affinity

- An application may retain state information for a user's session in memory
  - Other servers in the cluster might not have this information
- HTTP server plug-in routes subsequent servlet requests consistently to the same application server after the session created
- Using server ID passed with session ID in cookie or URL



© Copyright IBM Corporation 2007

Figure 14-24. Session affinity

WA3616.3

### Notes:

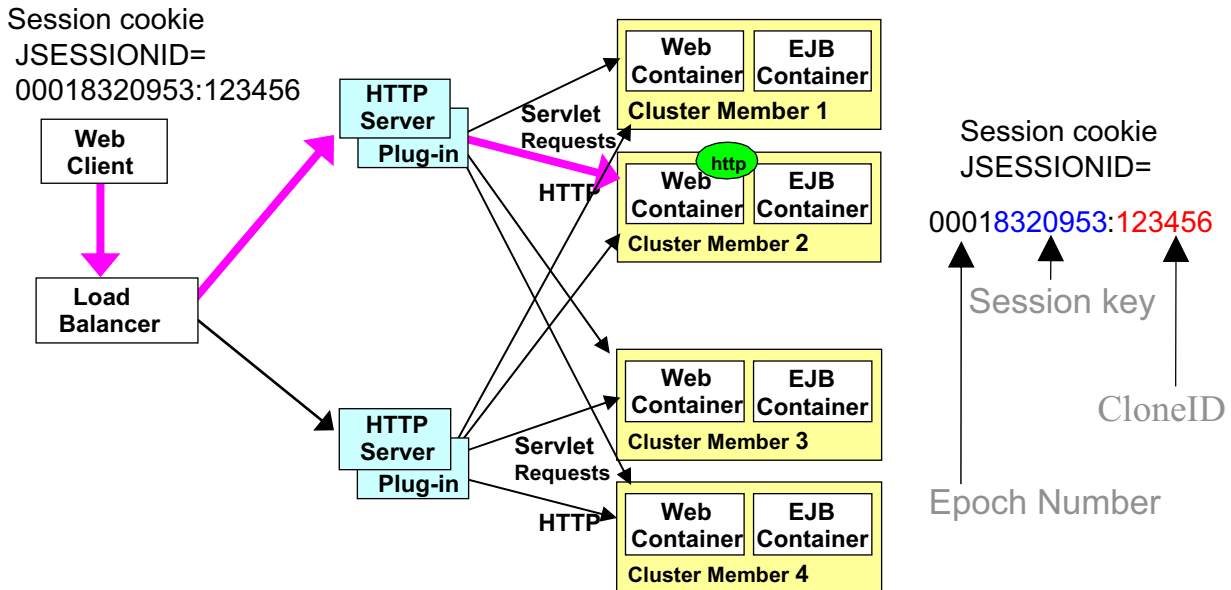
**Instructor notes:****Purpose —**

**Details —** When no session persistence has been configured, the other cluster members will not have the session information available. It is then necessary to rely on session affinity to make sure that the customers subsequent requests go to the same application server that holds their state information.

**Additional information —**

**Transition statement —** Next: WebSphere session affinity

# WebSphere session affinity



- Session cookie is generated
- Allows affinity on first hit by using clone ID
- Plug-ins know how to route to correct clone ID
- Can be turned off by removing clone IDs from plug-in file

© Copyright IBM Corporation 2007

Figure 14-25. WebSphere session affinity

WA3616.3

## Notes:

The format for the JSESSIONID cookie changes when using memory-to-memory replication.

***Instructor notes:***

**Purpose —**

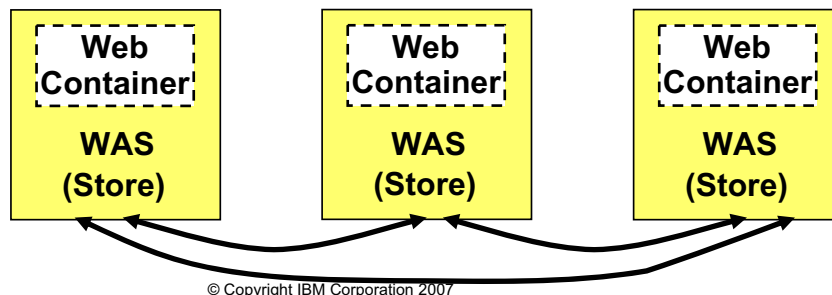
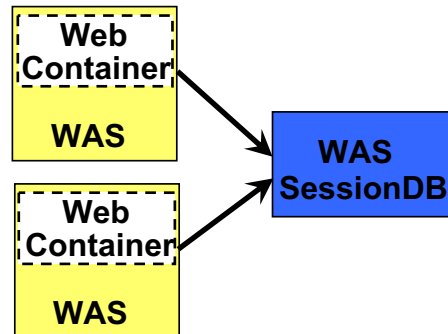
**Details —**

**Additional information —**

**Transition statement —** Next: Session persistence

## Session persistence

- Session objects are cached in server memory by default, and therefore are lost if server fails
- Two methods to enable session persistence
  - Database
    - Session objects persisted using JDBC data source
    - DB2 included in package for session persistence
  - Memory-to-memory replication
    - Sessions copied to another server using WebSphere Platform Messaging



© Copyright IBM Corporation 2007

Figure 14-26. Session persistence

WA3616.3

### Notes:

If your application can not tolerate loss of session information, session persistence can be enabled. The two choices are database and memory-to-memory. The database approach has greater overhead, but would provide persistence to a database. Memory-to-memory could still fail if all copies of the session are lost.

There are a number of settings that can be used to affect the balance between failover and performance. This will be discussed in upcoming slides.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Session persistence tuning

## Session persistence tuning

- Session persistence can be tuned to favor performance or failover
- Accesses through **Servers** → **<servername>** → **Session management** → **Distributed environment settings** → **Tuning parameter**

Configuration

**General Properties**

**Tuning level:**

Very high (optimize for performance)  
Write frequency Time based: 300 seconds  
Write contents Only updated attributes  
Schedule sessions cleanup: true

High  
Write frequency Time based: 300 seconds  
Write contents All session attributes  
Schedule sessions cleanup: false

Medium  
Write frequency End of servlet service  
Write contents Only updated attributes  
Schedule sessions cleanup: false

Low (optimize for failover)  
Write frequency End of servlet service  
Write contents All session attributes  
Schedule sessions cleanup: false

[Custom settings](#)  
Write frequency Time based: 10 seconds  
Write contents Only updated attributes  
Schedule sessions cleanup: false

Apply OK Reset Cancel

© Copyright IBM Corporation 2007

Figure 14-27. Session persistence tuning

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

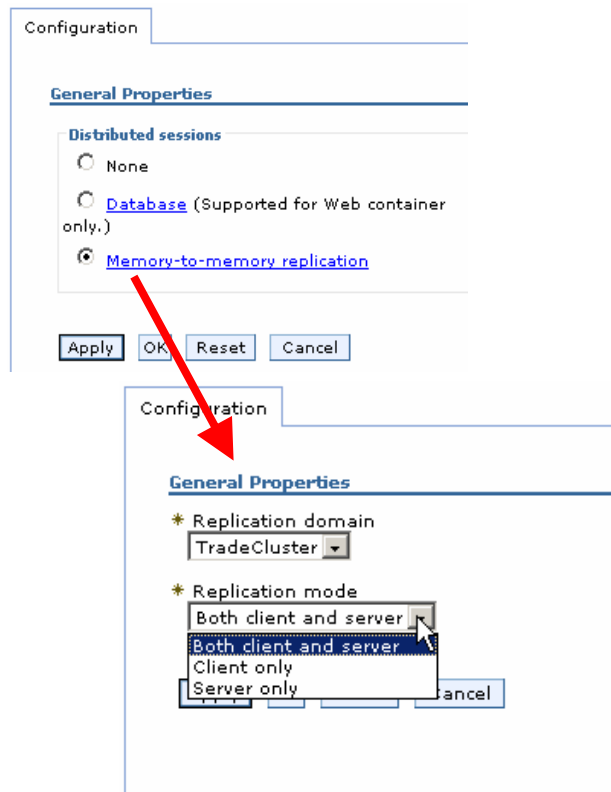
**Details —**

**Additional information —**

**Transition statement —** Next: Session configuration: Memory-to-memory

## Session configuration: Memory-to-memory

- Configuring Memory-to-memory session replication
  - Found under **Application servers** → **<servername>** → **Session management**
  - Click **Distributed environment settings**
  - Select **Memory-to-memory**
  - Configure the replication domain



© Copyright IBM Corporation 2007

Figure 14-28. Session configuration: Memory-to-memory

WA3616.3

### Notes:

Replication modes:

- **Server mode:** Only store backup copies of other WebSphere Application Server sessions and not to send out copies of any session created in that particular server
- **Client mode:** Only broadcast or send out copies of the sessions it owns and not to receive backup copies of sessions from other servers
- **Both mode:** Simultaneously broadcast or send out copies of the sessions it owns and act as a backup table for sessions owned by other WebSphere Application Server instances

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Session configuration: Replication domains

## Session configuration: Replication domains

- Creating replication domains
  - Can be created when creating a cluster
  - Or use **Environment** → **Replication Domains** → **New**
- To configure replication domains after creation
  - Through the administrative console, select the Replication domain under **Environment** → **Replication domains**
  - Select the Replication domain
  - Select the Number of replicas
    - Single replica
    - Entire domain
    - Specified

The screenshot shows a web-based configuration interface for 'Replication domains' under the 'TradeCluster' environment. The 'Configuration' tab is active, displaying 'General Properties'. The 'Name' field is set to 'TradeCluster'. The 'Request timeout' is set to 5 seconds. The 'Encryption' section shows 'Encryption type' set to 'none' with a 'Regenerate encryption key' button. The 'Number of replicas' section has three radio buttons: 'Single replica' (selected), 'Entire Domain', and 'Specify'. Below the 'Specify' option is a text input field for 'Number of replicas'. At the bottom are 'Apply', 'OK', 'Reset', and 'Cancel' buttons.

© Copyright IBM Corporation 2007

Figure 14-29. Session configuration: Replication domains

WA3616.3

### Notes:

Single replica means that for every session object, there will be one backup copy stored on a different server for failover. It is also possible to have the sessions backed up on every member of the domain. This would tend to not scale as well since every member has to store every other members session objects. But, it would mean that the chances for a successful failover is higher.

Depending on the needs of the application and the tolerance for failure, it is possible to tune the number of members that backup every session object.

***Instructor notes:***

**Purpose** —

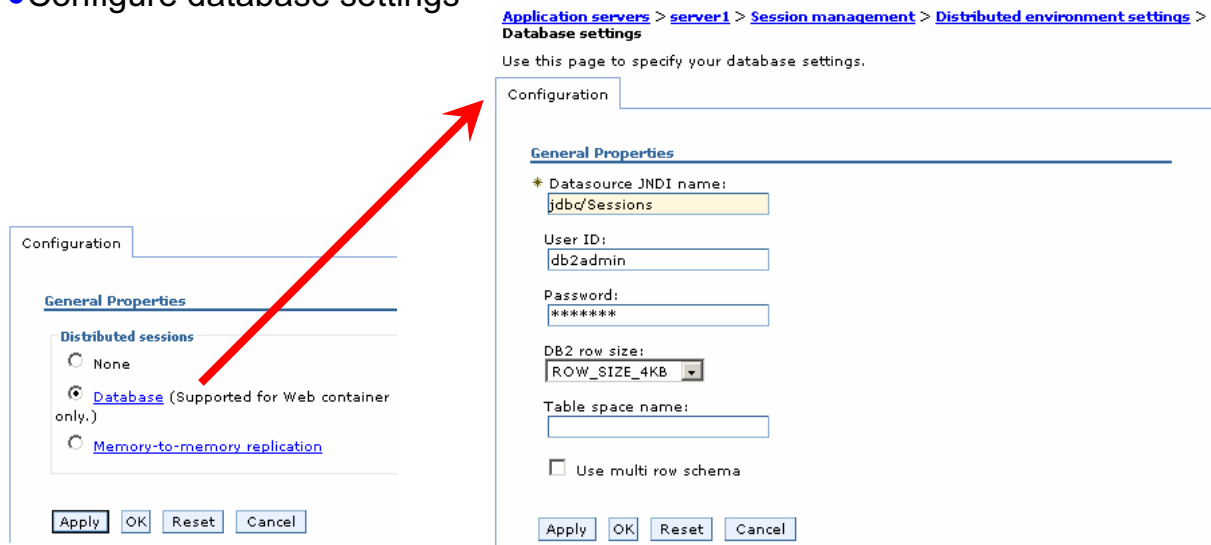
**Details** —

**Additional information** —

**Transition statement** — Next: Database persistence configuration

# Database persistence configuration

- Create database in DBMS
- Create data source: **Resources** → **Data sources**
- Select **Database** in Distributed environment settings page
  - Found under **Application servers** → **<servername>** - → **Session management**
- Configure database settings



© Copyright IBM Corporation 2007

Figure 14-30. Database persistence configuration

WA3616.3

## Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Checkpoint questions

## Checkpoint questions

---

1. A WebSphere cluster member is what type of process?
  - A. An application server
  - B. A Web server
  - C. An edge server
  - D. A proxy server
  
2. The creation of a cluster can be based on which of the following?
  - A. An application
  - B. An application server
  - C. An enterprise application
  - D. An application manager
  
3. Having session affinity means that session information is not lost during failover .
  - A. True
  - B. False

© Copyright IBM Corporation 2007

---

Figure 14-31. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.

**Instructor notes:**

Purpose —

## Checkpoint solutions

---

1. A WebSphere cluster member is what type of process?  
A. An application server
  
2. The creation of a cluster can be based on which of the following?  
B. An application server
  
3. Having session affinity means that session information is not lost during failover .  
B. False

© Copyright IBM Corporation 2007

**Additional information** —**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Define workload management
- Create clusters and cluster members
- Compare clustered configurations
- Explain how weights are used in workload management
- Discuss failover scenarios
- Describe the role of the HTTP plug-in in workload management
- Explain session management
- Configure distributed session management

© Copyright IBM Corporation 2007

Figure 14-32. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

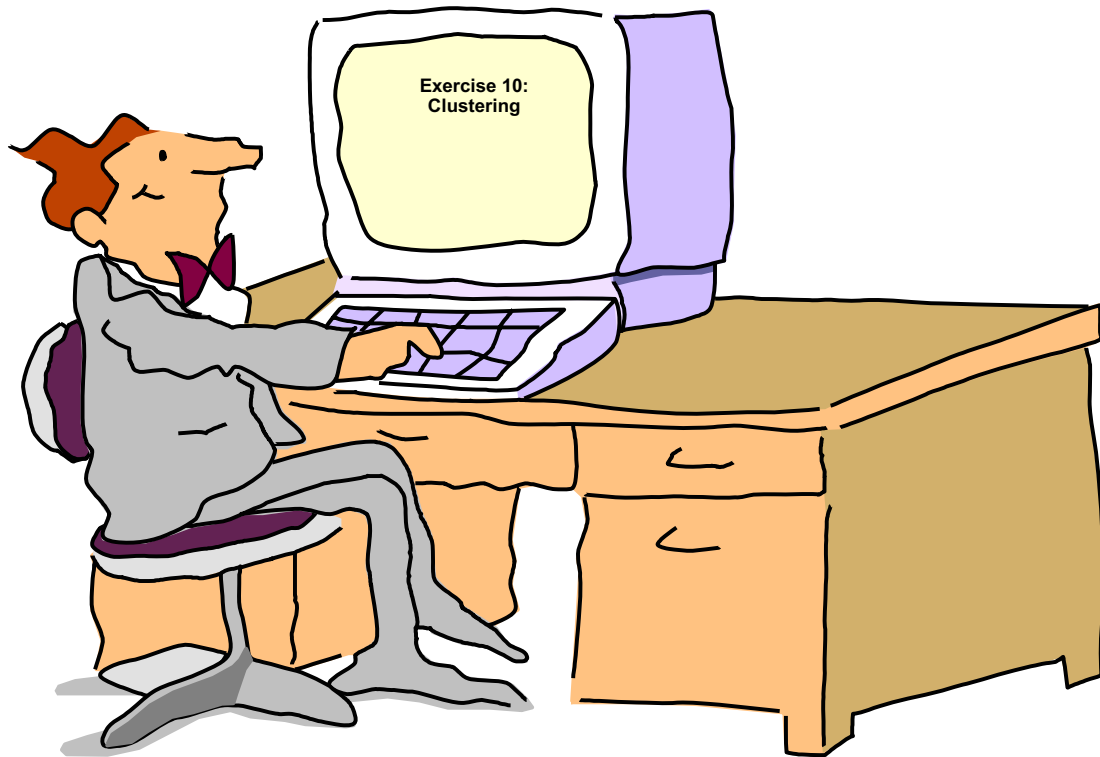
**Details —**

**Additional information —**

**Transition statement —** Next: Lab exercise

## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 14-33. Lab exercise

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**





---

# Unit 15. Introduction to WebSphere Platform Messaging

## Estimated time

01:00

## What this unit is about

This unit describes an overview of WebSphere Platform Messaging (WPM) with several sample topologies explained.

## What you should be able to do

After completing this unit, you should be able to:

- Describe what WebSphere Platform Messaging is and how it is used
- Explain which WebSphere Application Server components use WebSphere Platform Messaging
- Describe how JMS and MQ use WPM to provide application messaging services

## How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

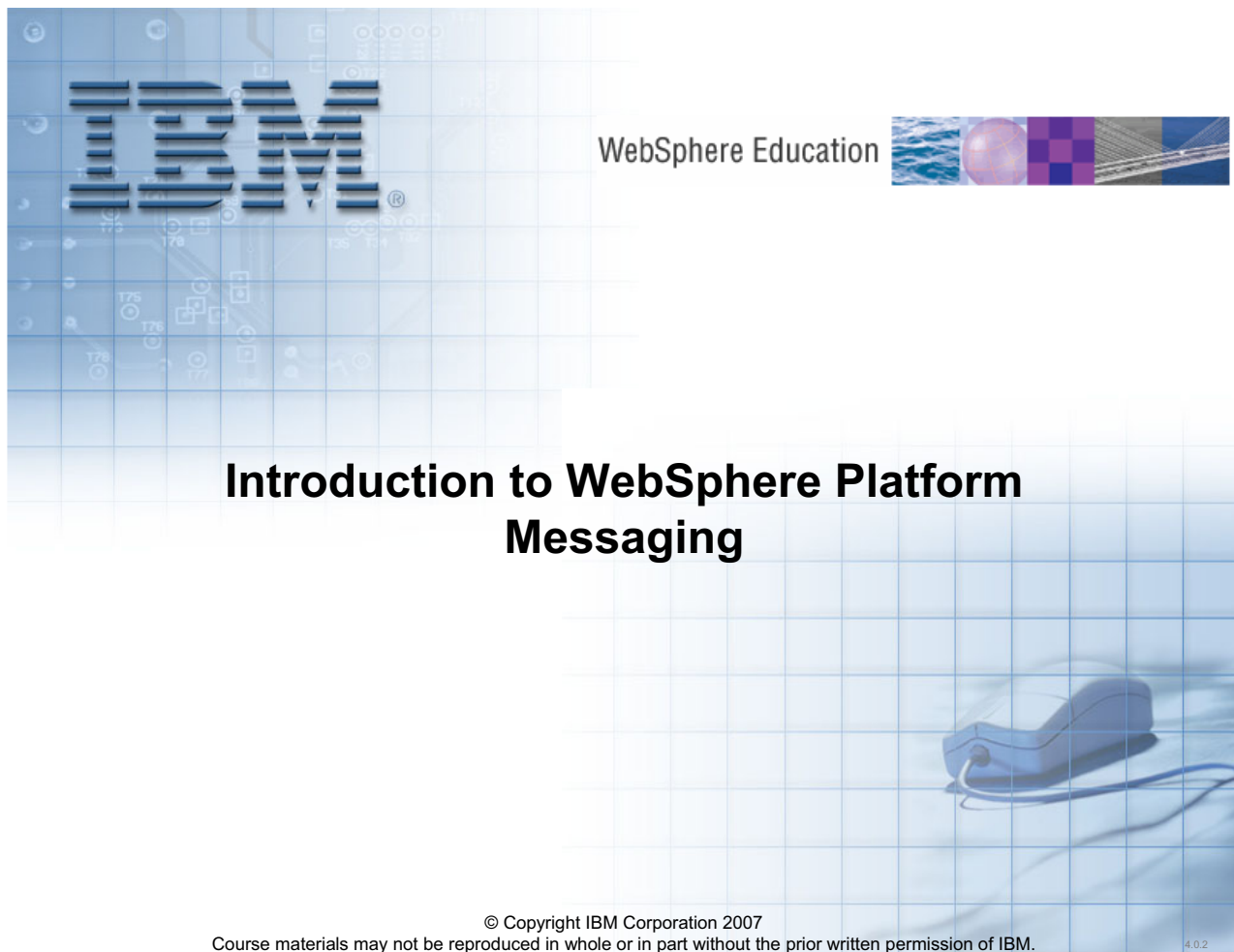


Figure 15-1. Introduction to WebSphere Platform Messaging

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe what WebSphere Platform Messaging is and how it is used
- Explain which WebSphere Application Server components use WebSphere Platform Messaging
- Describe how JMS and MQ use WebSphere Platform Messaging to provide application messaging services

© Copyright IBM Corporation 2007

Figure 15-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: What is JMS

## What is JMS

---

- The Java Message Service (JMS) API is an API for accessing enterprise messaging systems. It is part of the Java 2 Platform, Enterprise Edition (J2EE).
- The Java Message Service allows business applications to asynchronously send and receive business data and events.
- The Java Message Service supports both message queuing and publish/subscribe styles of messaging.
- A **JMS Provider** provides the implementation of the JMS API

© Copyright IBM Corporation 2007

Figure 15-3. What is JMS

WA3616.3

### **Notes:**

JMS defines an API, it is not an implementation.

JMS just defines some interfaces for applications to use.

It is up to someone else to provide the implementation under those interfaces.

The implementation is delivered as a **JMS Provider**

***Instructor notes:***

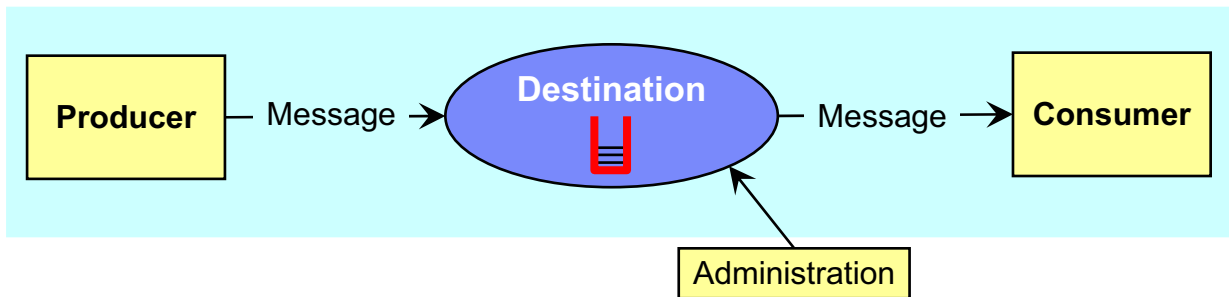
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Messaging – Basic flow

## Messaging – Basic flow



- Producers send/put messages to destinations
- Consumers receive/get messages from destinations
- Destinations are Platform Messaging managed points of communication rendezvous
  - JMS queues
  - JMS topics (publish / subscribe)
  - Web service endpoints
- While the messaging flow is inherently asynchronous, it can be used to provide synchronous communication

© Copyright IBM Corporation 2007

Figure 15-4. Messaging – Basic flow

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere Platform Messaging

## WebSphere Platform Messaging

---

- WPM is the JMS Provider delivered with WebSphere Application Server
- Messaging capabilities are fully integrated in the WebSphere platform
  - Fully integrated with application server management including security and high availability
  - Fully compliant JMS 1.1 provider
- Complement and extend WebSphere MQ and WebSphere Application Server
  - Share and extend messaging family capabilities
  - Interoperable with WebSphere MQ
- Other higher level implementations use WPM
  - WebSphere Enterprise Service Bus (WESB)
  - WebSphere Process Server (WPS)
  - Supported in tooling by WebSphere Integration Developer (WID)

© Copyright IBM Corporation 2007

Figure 15-5. WebSphere Platform Messaging

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Service Integration Bus (SIBus)

## Service Integration Bus (SIBus)

---

- The Service Integration Bus (SIBus) is used by WebSphere's implementation of the JMS Provider
  - [Intelligent infrastructure for service-oriented integration](#)
  - [Unifies Service-Oriented Architecture \(SOA\), messaging, message brokering and publish/subscribe](#)
- An SIBus Contains **Bus Members** and **Destinations**
- SIBus Bus Members can host Producers and Consumers
  - [Producers and Consumers can also be external to WebSphere](#)
- The SIBus manages communication with Destinations
  - [Asynchronous or Synchronous messaging](#)
- When SIBus is used for JMS applications, it is referred to as a messaging bus

© Copyright IBM Corporation 2007

Figure 15-6. Service Integration Bus (SIBus)

WA3616.3

### **Notes:**

Service Integration Bus (SIBus) also provides an underpinning for WebSphere Business Integration.

WS Gateway uses the SIBus

The following capabilities are provided by the SIBus:

- Any application can exchange messages with any other application by using a *destination* to which one application sends, and from which the other application receives.
- A message-producing application, that is, a *producer*, can produce messages for a destination regardless of which messaging engine the producer uses to connect to the SIBus.
- A message-consuming application, that is, a *consumer*, can consume messages from a destination (whenever that destination is available) regardless of which messaging engine the consumer uses to connect to the SIBus.

The SIBus supports the following types of messaging:

- Sending messages synchronously (this requires the consuming application to be running and reachable)
- Sending messages asynchronously (possible whether the consuming application is running or not and whether or not the destination is reachable). Both point-to-point and publish/subscribe messaging are supported
- Publishing events or other notifications. Notification messages can also be generated by the SIBus itself.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Bus member

## Bus member

---

- Bus members of a SIBus are application servers or clusters
- When a new bus member is defined, a **Messaging Engine (ME)** is automatically created on the corresponding application server or cluster.
  - One per cluster member on a cluster
- Can add or remove bus members; this effectively adds or removes Messaging Engines

© Copyright IBM Corporation 2007

Figure 15-7. Bus member

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Messaging engine (ME)



## Messaging engine (ME)

---

- MEs run inside the application server or cluster, and manage messaging resources.
  - A common pattern is one ME per server
- MEs provide a connection point for clients to put or get messages.

© Copyright IBM Corporation 2007

Figure 15-8. Messaging engine (ME)

WA3616.3

### **Notes:**

One ME is automatically created for the application server or the cluster when defining a new SIBus member (application server or cluster). Can have multiple MEs running within the same cluster.

Within a SIBus, each ME has a unique identity - made up of SIBus name and server name.

***Instructor notes:***

**Purpose —**

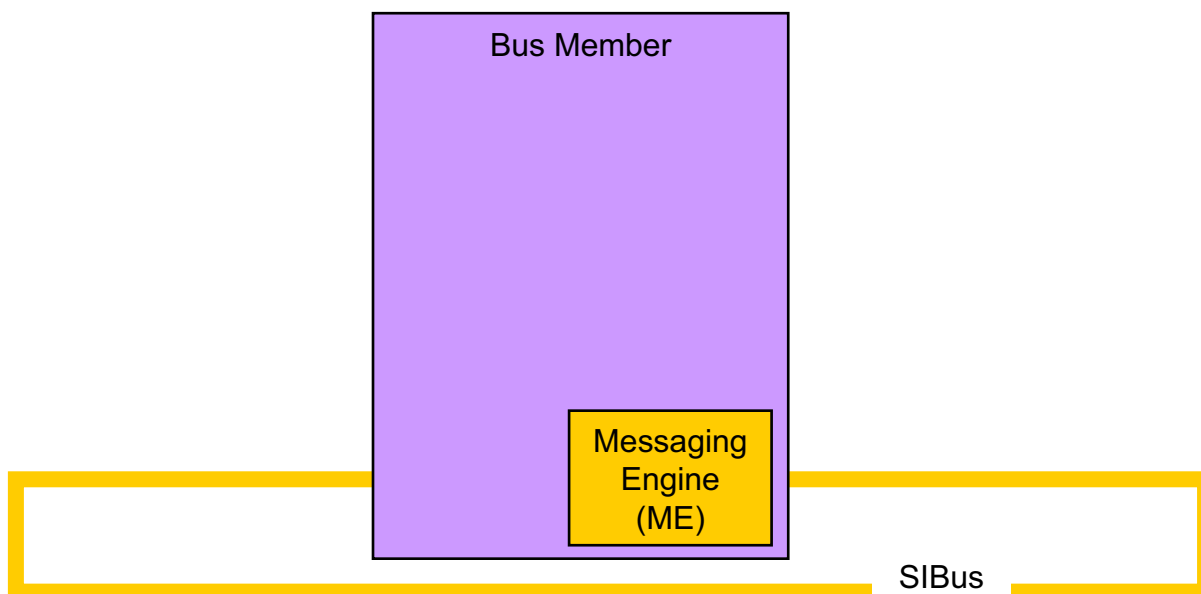
**Details —**

**Additional information —**

**Transition statement —** Next: Bus member and messaging engine

## Bus member and messaging engine

- An SIBus bus member can be a server or a cluster.
- Each bus member contains a Messaging Engine (ME).



© Copyright IBM Corporation 2007

Figure 15-9. Bus member and messaging engine

WA3616.3

### Notes:

One ME is automatically created for the application server or the cluster when defining a new SIBus member (application server or cluster). It is possible to have multiple MEs running within the same cluster.

Within a SIBus, each ME has a unique identity - made up of SIBus name and server name.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Message store

## Message store

---

- An ME requires a persistent backing data store (even for non-persistent messages).
  - JDBC database
  - File system
- Multiple MEs may share a database, but each ME has its own schema within the database (which results in different tables).
- Derby database is used as the default in Base. In a federated environment, a distributable database, such as DB2, is required.

© Copyright IBM Corporation 2007

Figure 15-10. Message store

WA3616.3

### Notes:

The message store is a sub-component of the messaging engine. This is used to buffer in-flight messages and hold a number of other pieces of information (for example records of message delivery when delivering multiple copies of a single message).

A message store can be either **Persistent** or **Non-Persistent**. A persistent message store is capable of holding both persistent objects (state which survives after an engine stops for whatever reason) and volatile objects (state which does not survive an ME failure, and may or may not survive an orderly shutdown of the messaging engine). In contrast a Non-Persistent message store can only hold volatile objects, it cannot hold persistent objects.

ME requires a persistent back-end data store, even for non-persistent messages (example, for spilling).

Out of the box support for persistence using Derby. Support for DB2, Oracle, Sybase etc. to enable customers to use an RDBMS of choice. These databases should also offer higher performance than Derby. Starting with version 6.1 a File System Data Store is also available.

The data store preserves messages, subscriptions, and so on, so that they survive if the server or messaging engine is stopped and restarted. It is also used for the overflow of the non-persistent messages in some Quality of Service (QOS) options.

***Instructor notes:***

**Purpose —**

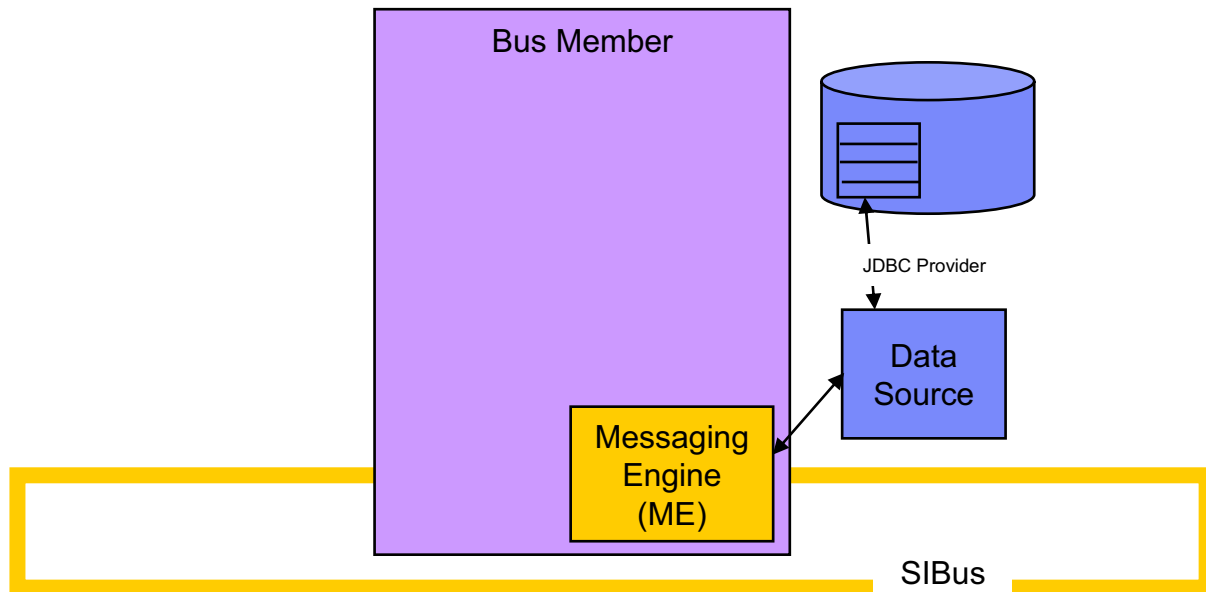
**Details —**

**Additional information —**

**Transition statement —** Next: Messaging engine Data stores

## Messaging engine Data stores

- Messaging Engines have an associated data store.



© Copyright IBM Corporation 2007

Figure 15-11. Messaging engine Data stores

WA3616.3

### Notes:

Persistent storage can be provided by a file store, or a data store.

File store provides an easier to manage and better performing environment.

In a clustered environment a shared file system must be used so all instances of messaging engines can have access for failover purposes.

The data store preserves messages, subscriptions, and so on, so that they survive if the server or messaging engine is stopped and restarted. It is also used for the overflow of the non-persistent messages in some Quality of Service (QOS) options.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Bus destinations

## Bus destinations

---

- A bus destination is a virtual place within an SIBus, to which applications (producers, consumers, or both) attach to exchange messages.
- Bus destinations can be permanent or temporary.
  - **Temporary - Created and deleted automatically for API specific destinations**
    - Created programmatically, usually to specify a JMSReplyTo destination within a message
  - **Permanent – Created by administrator**
    - Deleted only when administrator deletes it
- Types of destinations.
  - **Queue - For point-to-point messaging**
  - **Topicspace - For publish/subscribe messaging**
  - **Alias - Destination that is an alias for another target destination**
  - **Foreign - Destination that identifies a destination on another bus**
  - **Exception – Destination that is used to handle messages that cannot be sent to intended bus destination**

© Copyright IBM Corporation 2007

Figure 15-12. Bus destinations

WA3616.3

### **Notes:**

Temporary destinations are bus destinations that are created and deleted automatically for API-specific temporary destinations. For example, if an application creates a temporary JMS queue for use with the default messaging provider, which is based on service integration technologies, the SIBus service automatically creates a temporary destination on the bus. Temporary destinations appear on the list of destinations for a service integration bus, but normally need no administration.

A topicspace is a hierarchy of topics used for publish/subscribe messaging. Topics with the same name can exist in multiple topicspaces.

Alias destinations provide a level of abstraction between applications and the underlying target bus destinations that hold messages. Applications interact with the alias destination, so the target bus destination can be changed without changing the application. Each alias destination identifies a target bus destination and target service integration bus.

Applications can use an alias destination to route messages to a target destination in the same bus or to another (foreign) bus (including across an MQLink to a queue provided by WebSphere MQ).

Each *messaging engine* has a default exception destination, `SYSTEM.DEFAULT.EXCEPTION.DESTINATION.me_name`, that can be used to handle undeliverable messages for all bus destinations that are localized to the messaging engine. The foreign destination identifies a destination on another bus, and enables applications on one bus to access directly the destination on another bus.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Linking destinations to bus members

## Linking destinations to bus members

---

- Bus destinations are associated with one or more bus members, thereby associating it with the corresponding MEs.
  - Allows administrator to control which database is used for persistence.
  - In most cases, a destination is associated with one ME.
  - Multiple MEs provide scalability and failover.
- Queue for point-to-point messaging.
  - Administrator defines a queue destination on one assigned bus member.
  - Each ME in that assigned bus member has a queue point where messages are held.
- Topicspace for publish/subscribe messaging.
  - Every ME in the SIBus is a publication point where messages are held.

© Copyright IBM Corporation 2007

Figure 15-13. Linking destinations to bus members

WA3616.3

### Notes:

Messages are transported through a 'bus' (e.g. an SIBus). A bus is a virtual messaging environment that spans servers. If you want to send or receive messages through a bus you have to connect to it. Servers (e.g. Server1) can be added to a bus, and in doing so you will start/create a messaging engine (ME) in that server.

SIBus destinations are defined on the bus (rather than on an ME). Queues will be localized to a particular ME (e.g. messages sent to that queue will be stored in a database on one particular ME within the bus), where as TopicSpaces (groups of topics) are localized on every ME.

A message point is the general term for the location on a messaging engine where messages are held for a bus destination. For point-to-point messaging, the administrator selects one bus member, an application server or cluster, as the assigned bus member that is to implement the runtime state of a queue destination. This action automatically defines a queue point for each messaging engine in the assigned bus member. For a queue destination assigned to an application server, all messages sent to that destination are

handled by the messaging engine of that server, and message order is preserved.

For publish/subscribe messaging, the administrator configures the destination as a topic space (a hierarchy of topics), but does not need to select any assigned bus member for the topicspace. A topic space has a publication point defined automatically for each messaging engine in the bus.

***Instructor notes:***

**Purpose —**

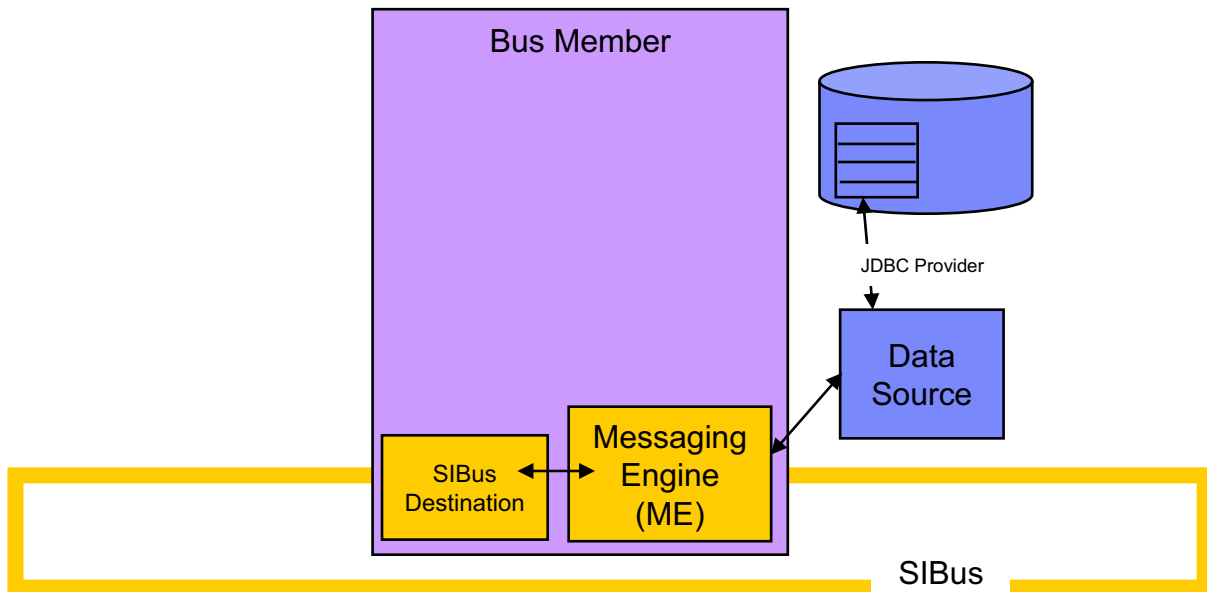
**Details —**

**Additional information —**

**Transition statement —** Next: SIBus destination

## SIBus destination

- SIBus destinations are associated with one or more MEs.



© Copyright IBM Corporation 2007

Figure 15-14. SIBus destination

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: J2EE access to bus members

## J2EE access to bus members

---

- J2EE applications (producers, consumers) access the SIBus and the bus members through the JMS API.
- JMS defines Queues and Topic Space destinations.
- JMS defines interfaces for accessing destinations.
  - [ConnectionFactory](#)
  - [ActivationSpec](#)

© Copyright IBM Corporation 2007

Figure 15-15. J2EE access to bus members

WA3616.3

### **Notes:**

In WebSphere JMS uses J2C Activation Specs to activate message-driven beans (MDB) that listen to JMS destinations.

***Instructor notes:***

**Purpose —**

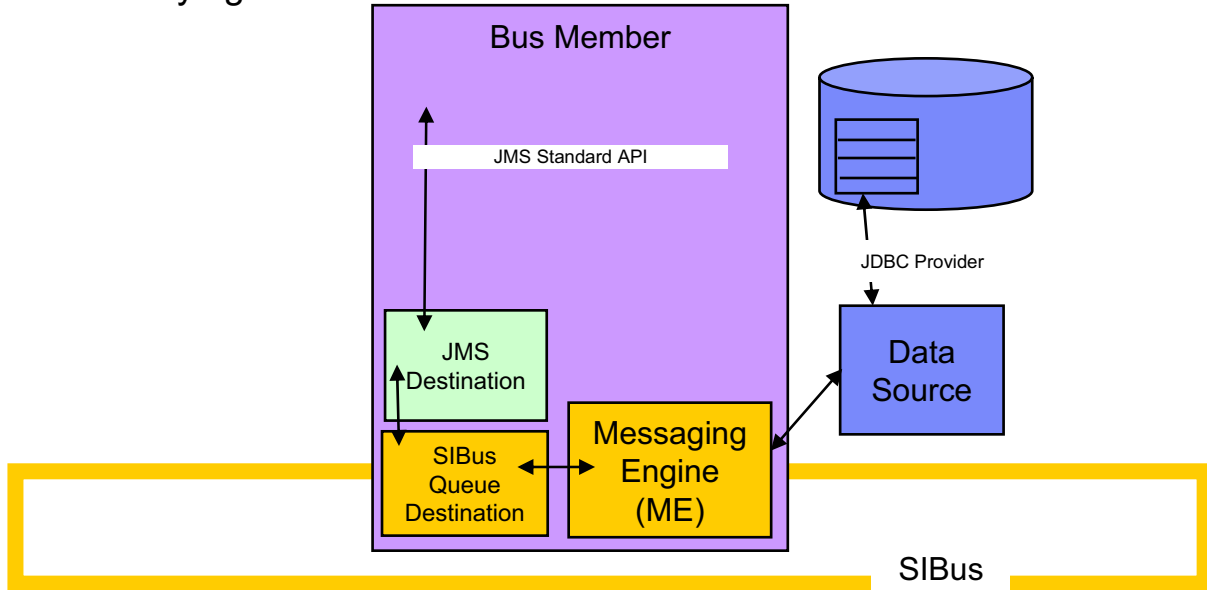
**Details —**

**Additional information —**

**Transition statement —** Next: JMS Destinations

## JMS Destinations

- JMS Destinations are associated with SIBus Destinations.
- The SIBus Destination implements JMS Destination function.
- Producers and consumers use the JMS API to work with the underlying Destination.



© Copyright IBM Corporation 2007

Figure 15-16. JMS Destinations

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

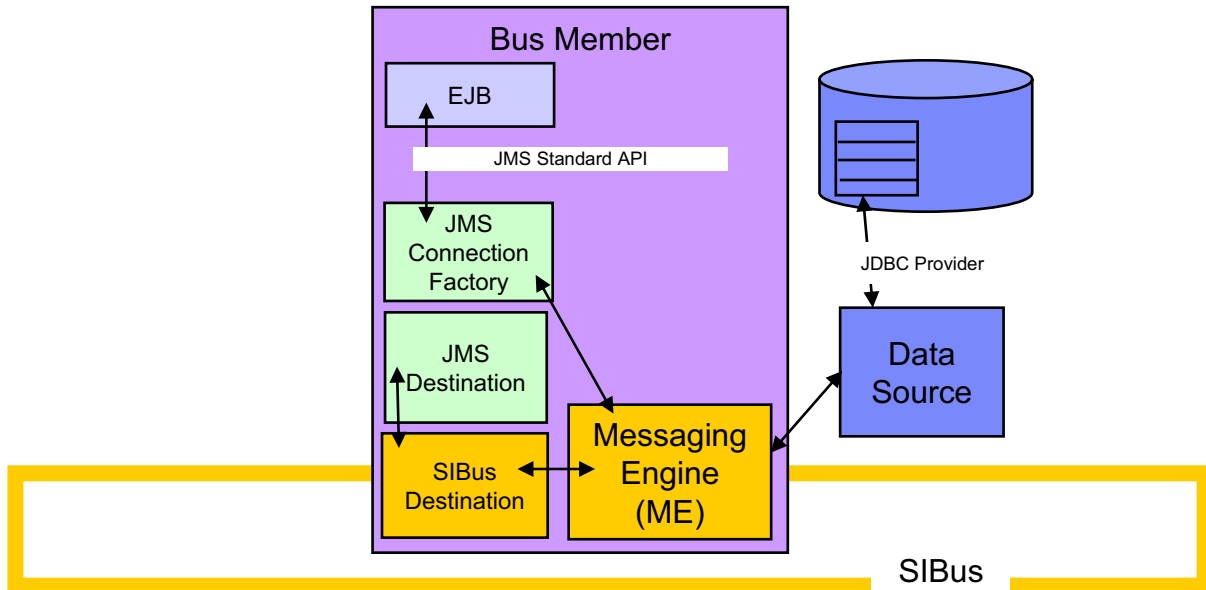
**Details —**

**Additional information —**

**Transition statement —** Next: JMS Connection Factory

# JMS Connection Factory

- Session EJBs use a JMS ConnectionFactory to connect to the JMS Provider (WPM).
  - Session EJBs can be producers or consumers.



© Copyright IBM Corporation 2007

Figure 15-17. JMS Connection Factory

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

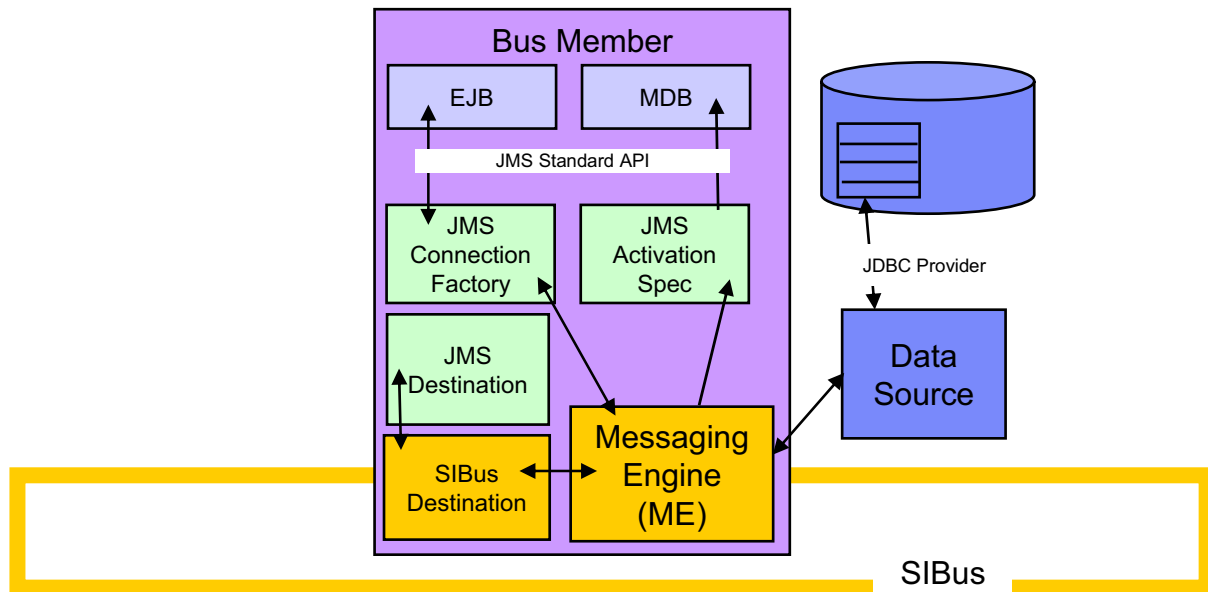
**Details —**

**Additional information —**

**Transition statement —** Next: JMS ActivationSpec

## JMS ActivationSpec

- Message Driven Beans (MDBs) use a JMS ActivationSpec to connect to the JMS Provider (WPM).



© Copyright IBM Corporation 2007

Figure 15-18. JMS ActivationSpec

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Clustering for high availability

## Clustering for high availability

---

- Add cluster as a bus member – MEs are automatically created.
- By default, only one ME is active at any given time.
  - HA Mgr decides which server the ME runs on.
- In case of active ME failure, the HA Mgr fails over the ME to another cluster member.
- No additional configuration required.

© Copyright IBM Corporation 2007

Figure 15-19. Clustering for high availability

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

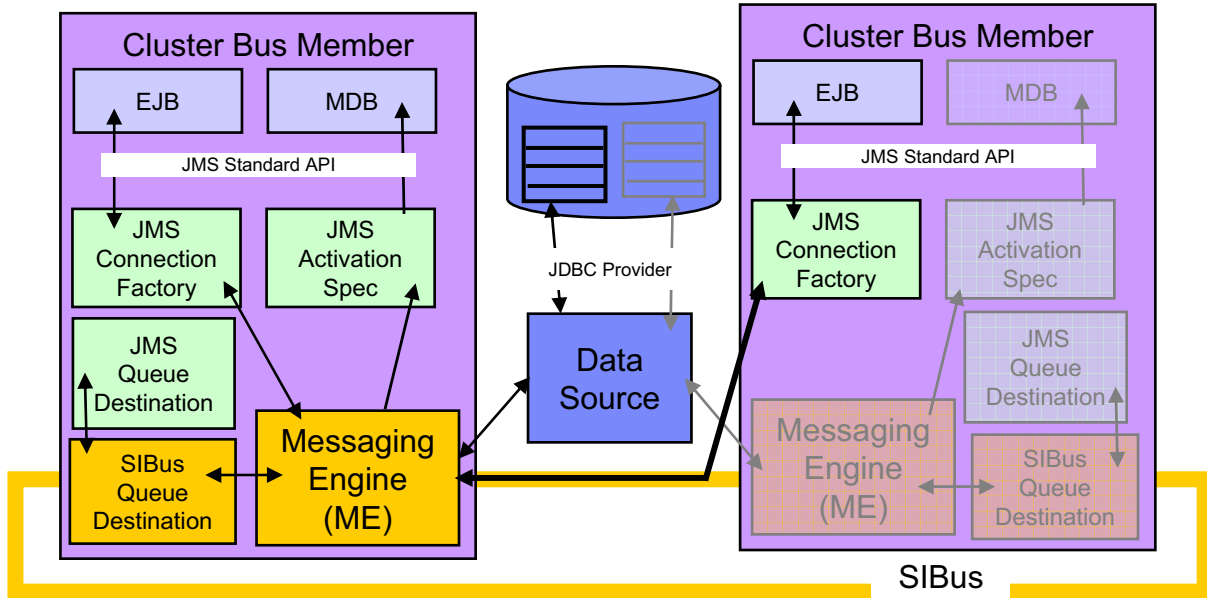
**Details —**

**Additional information —**

**Transition statement —** Next: Clustering for High Availability: Queue example

# Clustering for High Availability: Queue example

- All producers and consumers communicate with the active ME.
- Only one data source instance is used as the queue for all messages.
- Only the MDB with the active ME is given messages from the queue.



© Copyright IBM Corporation 2007

Figure 15-20. Clustering for High Availability: Queue example

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Clustering for scalability: Queue example

## Clustering for scalability: Queue example

---

- A single logical queue can be partitioned across the cluster
  - With each of  $n$  partitions perhaps holding an  $n$ th of the messages
- All MEs are active all the time
- Messaging ordering not preserved
- To work properly additional configuration is required
  - Must modify the HA Manager policy of the Bus Member HA Group

© Copyright IBM Corporation 2007

Figure 15-21. Clustering for scalability: Queue example

WA3616.3

### **Notes:**

This case provides scalability. The destination has been localized to a cluster ME. It is therefore partitioned across the MEs within the cluster. With a partitioned destination, the recoverable objects associated with the destination are split between separate message stores and hence separate data stores. This configuration has the disadvantage that message order cannot be preserved, but has advantages. One is that multiple consumers (or producers) can be deployed across the same cluster to provide high messaging bandwidth. Messaging operations would always be locally fulfilled.

Scalability can be increased by introducing additional MEs into a cluster. A set of MEs localizing a partitioned destination must all be in the same cluster. Likewise, an availability group of instances of the same ME must all be in the same cluster.

***Instructor notes:***

**Purpose —**

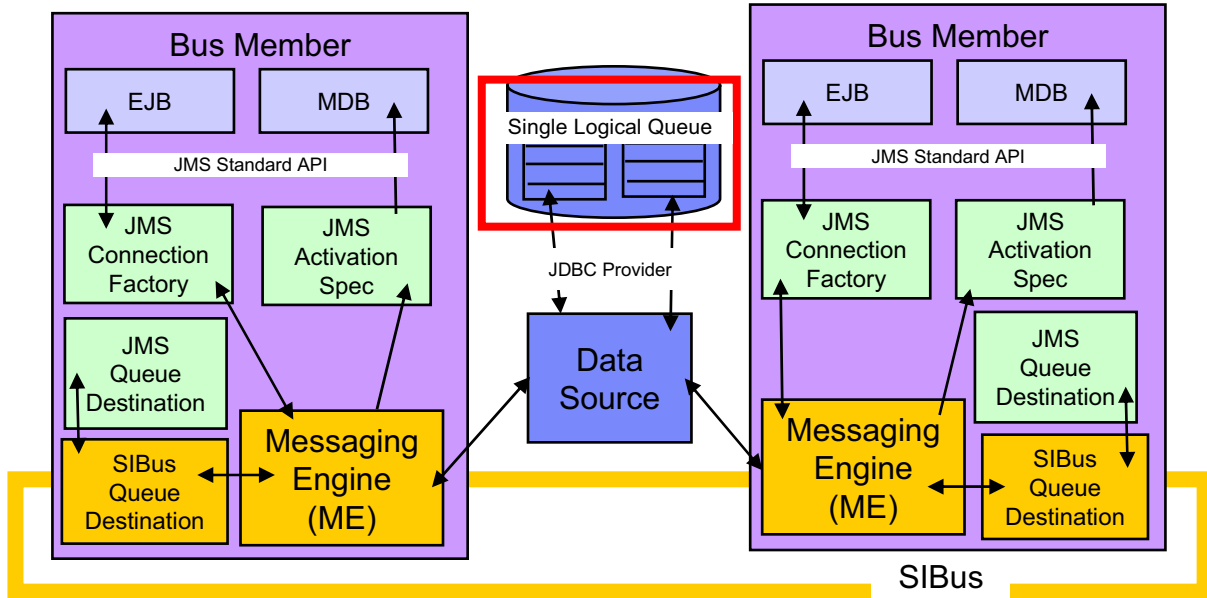
**Details —**

**Additional information —**

**Transition statement —** Next: Clustering for scalability: Queue example

# Clustering for scalability: Queue example

- Producers / Consumers communicate with the local ME
- Messages are partitioned across all data source instances
- All MDBs can process messages from their partition of queue



© Copyright IBM Corporation 2007

Figure 15-22. Clustering for scalability: Queue example

WA3616.3

## Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Messaging engine topology

## Messaging engine topology

---

- Can have multiple interconnected buses in a cell or stand-alone node (single server)
  - A common pattern is to have one SIBus in a stand-alone single server
- The default topology consisting of just one ME in a bus is adequate for many applications
- Advantages in deploying more than one ME, and linking them together are:
  - Spreading messaging workload across multiple servers
  - Placing message processing close to the applications that are using it
  - Improving availability in the face of system or link failure
  - Accommodating firewalls or other network restrictions that limit the ability of network hosts all to connect to a single ME

© Copyright IBM Corporation 2007

Figure 15-23. Messaging engine topology

WA3616.3

### **Notes:**

***Instructor notes:***

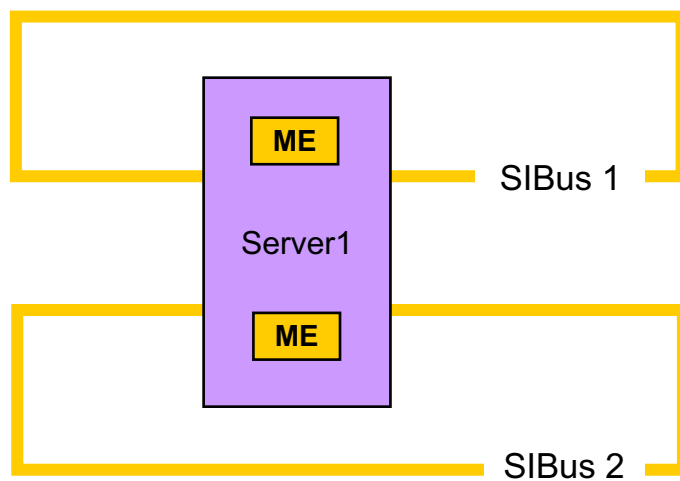
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: SIBus and MEs in a stand-alone server

## SIBus and MEs in a stand-alone server



- A stand-alone server can have multiple **buses**
- When the server is made a bus member, a **messaging engine** is created

© Copyright IBM Corporation 2007

Figure 15-24. SIBus and MEs in a stand-alone server

WA3616.3

### Notes:

A typical WebSphere Application Server configuration is to define a single bus for each WebSphere cell and to run a default ME in each application server that is a member of this bus. This serves as the embedded JMS provider, and provides SIBus capabilities for Web Services applications.

Messaging Engines run inside an application server, alongside J2EE applications and their containers. You can have multiple MEs running in the same application server. This is to allow the processing related to different sets of messaging applications running within the process to be separated, and for the different applications to be using different buses, each with their own topology and set of resources. There is no architectural requirement for MEs within the same process to be members of different buses, it is possible to have two MEs that are part of the same bus both running in the same process. However this is not a recommended configuration.

***Instructor notes:***

**Purpose —**

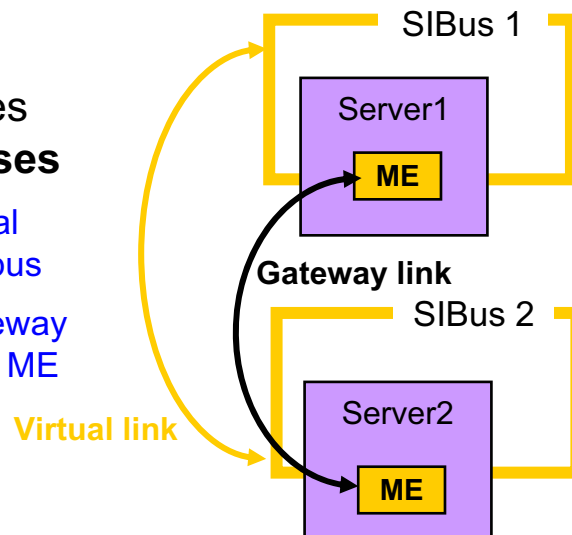
**Details —**

**Additional information —**

**Transition statement —** Next: Bus topology – Stand-alone servers

## Bus topology – Stand-alone servers

- An enterprise might deploy multiple interconnected messaging buses for organizational reasons
  - For example, separately administered buses for each department
- A bus can connect to other buses which are known as **foreign buses**
  - The administrator first creates a virtual link from the local bus to the foreign bus
  - The administrator then creates a gateway link from an ME in the local bus to an ME in the foreign bus



© Copyright IBM Corporation 2007

Figure 15-25. Bus topology – Stand-alone servers

WA3616.3

### Notes:

A bus can connect to other buses, which are referred to as foreign buses. If messaging engines are on different buses, applications can use those different buses, each with its own topology and set of resources. The inter-bus links might reflect the distribution of buses across organizations, across departments within organizations, or perhaps the separation of test and production facilities. To create a link to a foreign bus, the administrator first creates a virtual link from the local bus to the foreign bus, then creates a physical gateway link from a messaging engine in the local bus to the foreign bus.

***Instructor notes:***

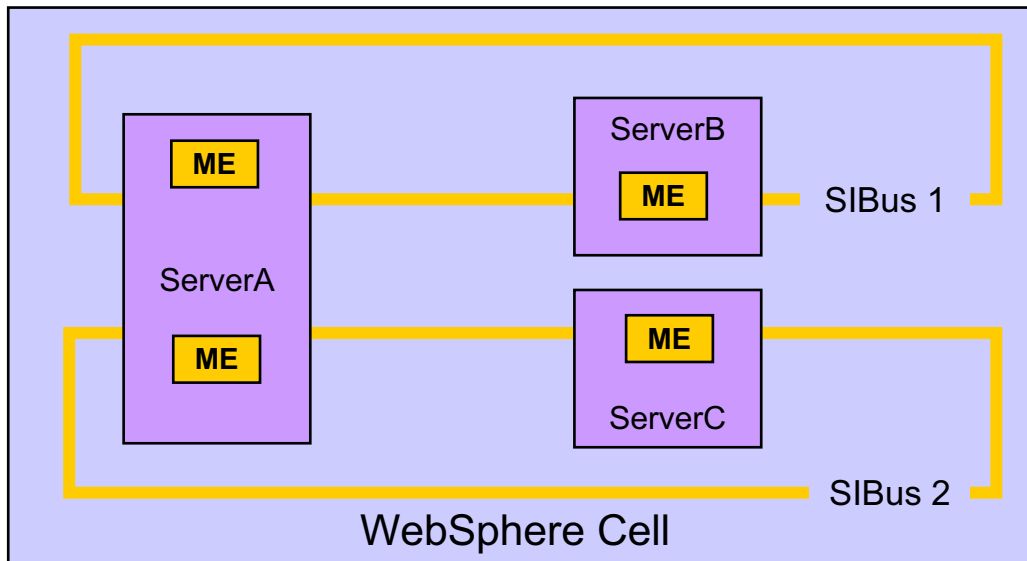
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: SIBus and MEs in a Network Deployment Cell

## SIBus and MEs in a Network Deployment Cell



- A WebSphere Application Server Cell can have multiple **buses**
- Each bus can have servers and clusters as **bus members**
- When a server, or cluster, is made a bus member, a **messaging engine** is created

© Copyright IBM Corporation 2007

Figure 15-26. SIBus and MEs in a Network Deployment Cell

WA3616.3

### Notes:

A typical WebSphere Application Server configuration is to define a single bus for each WebSphere cell and to run a default ME in each application server that is a member of this bus. This serves as the embedded JMS provider, and provides SIBus capabilities for Web Services applications.

Messaging Engines run inside an application server, alongside J2EE applications and their containers. You can have multiple MEs running in the same application server. This is to allow the processing related to different sets of messaging applications running within the process to be separated, and for the different applications to be using different buses, each with their own topology and set of resources. There is no architectural requirement for MEs within the same process to be members of different buses, it is possible to have two MEs that are part of the same bus both running in the same process. However this is not a recommended configuration.



***Instructor notes:***

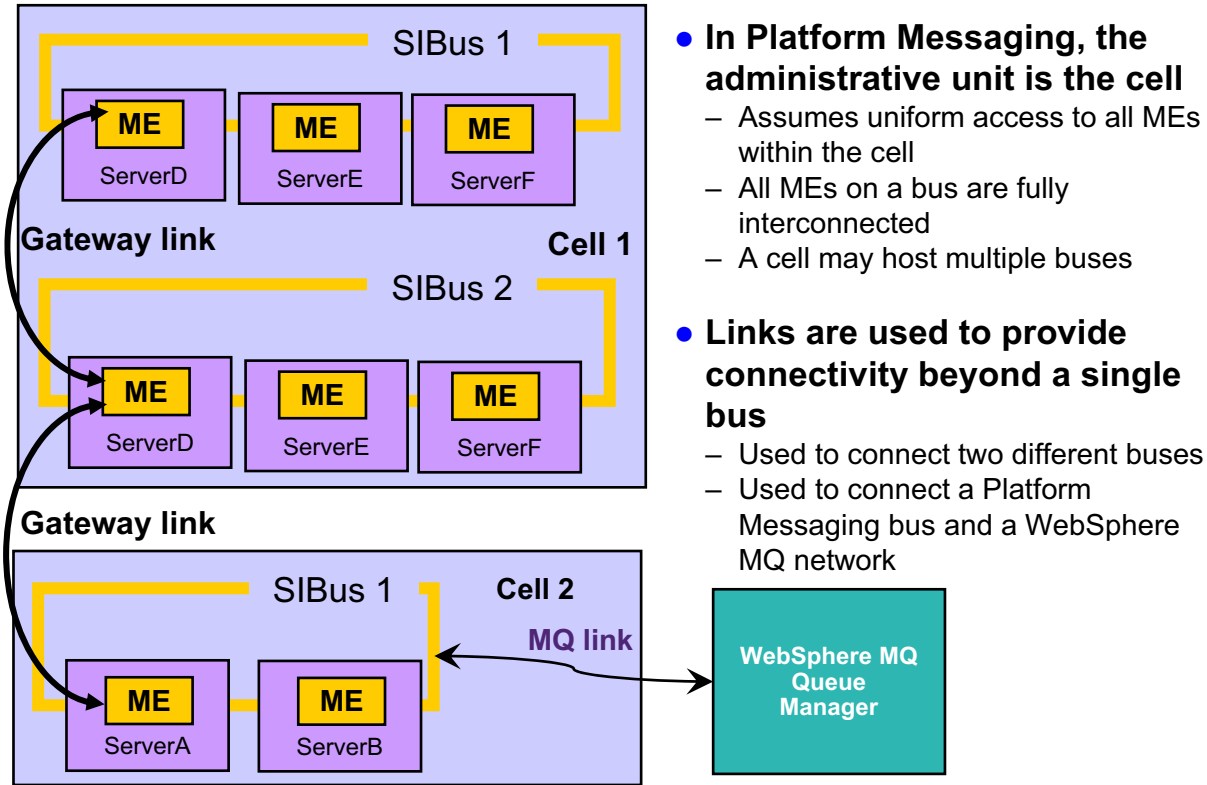
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Network topologies – Network Deployment cell

# Network topologies – Network Deployment cell



© Copyright IBM Corporation 2007

Figure 15-27. Network topologies – Network Deployment cell

WA3616.3

## Notes:

Protocol attach is the only way WebSphere MQ applications can interact with Platform Messaging applications. Platform Messaging will not directly support any WebSphere MQ programming interface, such as the MQI or the AMI. Connectivity between a message engine and an MQ Queue Manager is established by defining an MQLink. One of the primary functions of the MQLink is to convert between the formats and protocols used by WebSphere MQ and Platform Messaging, this is handled by component called the JS/MQ protocol adapter.

When you have a Bus, every destination on that bus must have a unique name - so to a certain extent the bus is a 'namespace'. If you have a second bus, you could have another destination with the same name as one in the first bus. When you go to the level of the JMS Proxies in the JNDI namespace, a JMS destination specifies only the destination Name, not the bus name. The Connection Factory Specifies the name of the bus to connect to (so it would

be possible to use the same JMS destination with 2 different ConnectionFactories (specifying different bus names) to connect to 2 different destinations).

***Instructor notes:***

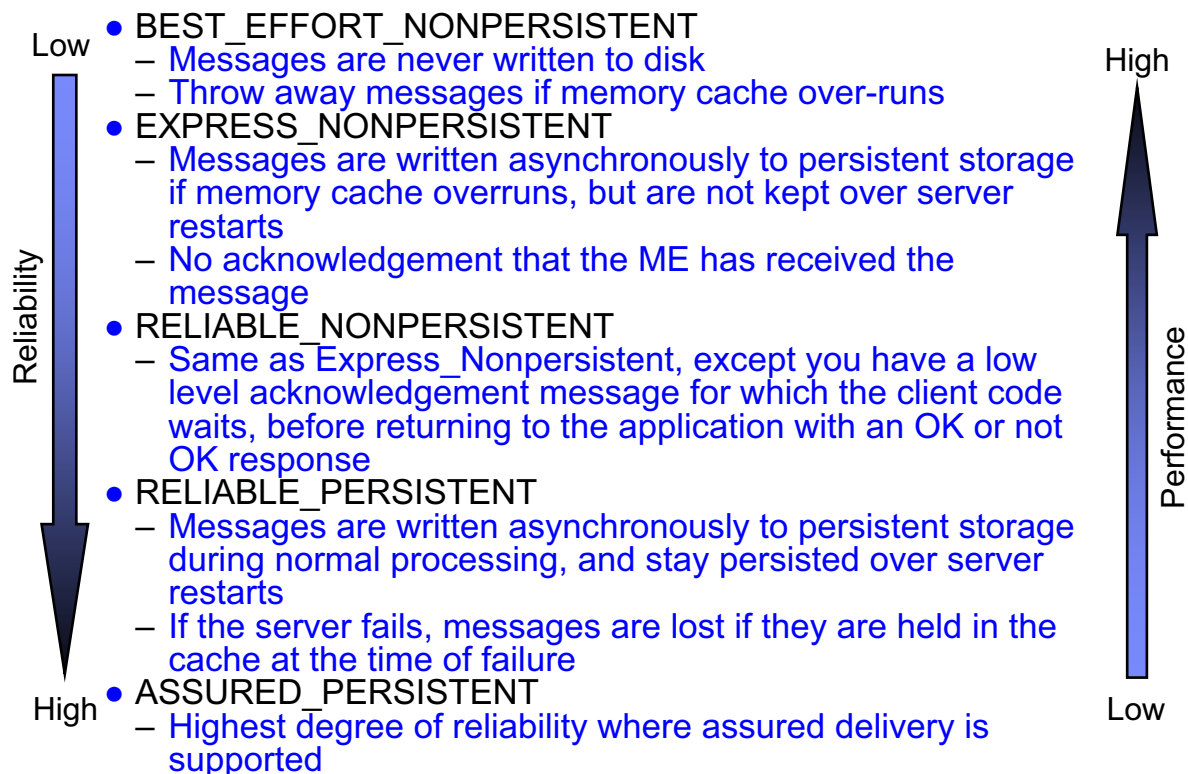
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Destination quality of service for reliability

## Destination quality of service for reliability



© Copyright IBM Corporation 2007

Figure 15-28. Destination quality of service for reliability

WA3616.3

### Notes:

Express non-persistent and reliable-persistent are defaults for non-persistent and persistent. In express non-persistent, messages are sent from a producer to the ME, but there is never an acknowledgement flow (at the low level communications layer) to indicate that the ME has the message. You return to the application immediately and assume that all is well. In the reliable non-persistent you have a low level acknowledgement message that the client code waits for before returning to the application with an OK, Not OK response. So - express runs faster, but with a slightly lower level of reliability.

Quality of service is configured through the administrative console in the Service Integration -> Buses -> Destinations page.

***Instructor notes:***

**Purpose —**

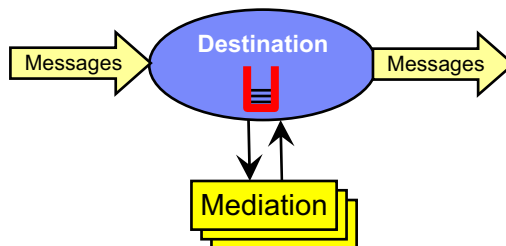
**Details —**

**Additional information —**

**Transition statement —** Next: Mediation

## Mediation

- Mediation - The ability to manipulate a message as it traverses the messaging bus (destination)
  - Transform the message
  - Copy or reroute the message to a different destination, or sequence of destinations
  - Allow interaction with non-messaging resource managers (for example, databases)
- Mediation attached administratively to a destination
- Mediation construction scenarios:
  - Built from supplied mediation subcomponents (mediation beans)
    - Subcomponent implementations shipped with WPM
  - Mediation beans supplied by IBM or third-party
    - IBM-supplied mediation beans come in the future – not in V6.1



© Copyright IBM Corporation 2007

Figure 15-29. Mediation

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Platform Messaging- Interoperability



---

## Platform Messaging- Interoperability

---

- Full interoperability with other SIBus in the same or different cell
- WebSphere V5 embedded JMS server interoperation
  - Existing WebSphere V5 embedded JMS clients can connect to V6.1 destinations
  - V6 JMS application to connect to an embedded JMS provider hosted in a V5 server
  - Note that it is not possible to connect a V5 embedded JMS server into a V6.1 SIBus
- **MQ Client Link** can be created to support any WebSphere V5 clients to talk to WebSphere V6.1 ME

© Copyright IBM Corporation 2007

Figure 15-30. Platform Messaging- Interoperability

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Relationship to WebSphere MQ

---

## Relationship to WebSphere MQ

---

- WebSphere MQ queue manager or a WebSphere MQ Integrator or Event Broker can coexist on the same machine as a ME
  - WebSphere MQ and Platform Messaging are separate products and do not share any modules or configuration data
- Connectivity between ME and MQ Queue Manager is established by defining a WebSphere MQLink
  - WebSphere MQLink converts between the formats and protocols used by WebSphere MQ and Platform Messaging
- Functions not supported in WebSphere V6.1
  - An MQ queue manager cannot attach to the bus using any communications protocol other than TCP/IP
  - A messaging engine cannot participate in a WebSphere MQ cluster

© Copyright IBM Corporation 2007

Figure 15-31. Relationship to WebSphere MQ

WA3616.3

### Notes:

***Instructor notes:***

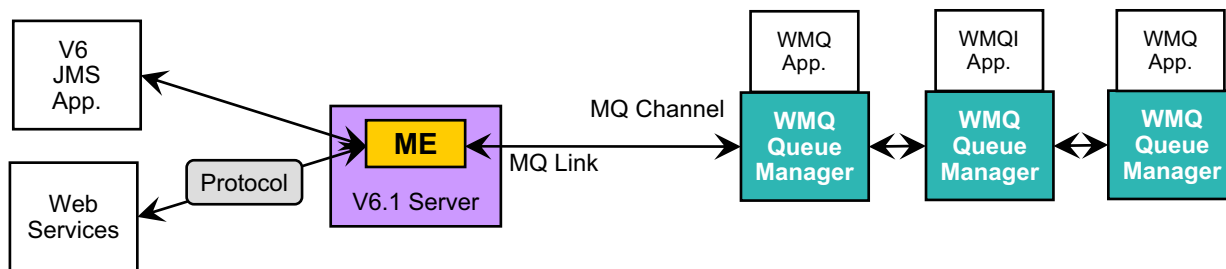
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Interoperability with MQ

## Interoperability with MQ



- Tight integration with WebSphere Platform Messaging and WebSphere MQ
- WebSphere MQ thinks that the V6.1 ME is another queue manager
- WebSphere MQ applications can send messages to queues hosted on V6.1 messaging
- WebSphere V6.1 messaging applications can send messages to WebSphere MQ queues

© Copyright IBM Corporation 2007

Figure 15-32. Interoperability with MQ

WA3616.3

### Notes:

You can have multiple MQLinks out of a bus, but each link goes to a different queue manager, and further these queue managers should not be interconnected. The link engine can be part of a cluster, but the issue is in handling failover. The ME hosting the MQLink must keep a fixed host/port address because that is what MQ expects, and so you have to marry the new WebSphere Application Server HA support with more traditional HACMP like HA solutions.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Usage scenarios for Platform Messaging

---

## Usage scenarios for Platform Messaging

---

- Use Platform Messaging:
  - Customers and J2EE developers currently using WebSphere Application Server V5 embedded JMS provider for intra-WAS messaging
  - Messaging between WebSphere Application Server and existing MQ backbone and its applications
  
- Use WebSphere MQ:
  - Customers currently using WebSphere MQ may continue to use it
  - Access is required to heterogeneous non-JMS applications, WebSphere MQ clustering, or other WebSphere MQ functions

© Copyright IBM Corporation 2007

Figure 15-33. Usage scenarios for Platform Messaging

WA3616.3

### **Notes:**

The messaging support of the default messaging provider is only accessible from WebSphere Application Server Web, EJB and client containers, and is interoperable with WebSphere MQ. If access is required to heterogeneous non-JMS applications, WebSphere MQ clustering, or other WebSphere MQ functions, you should install WebSphere MQ as a messaging provider.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: V6 support for external JMS providers



---

## V6 support for external JMS providers

---

- WebSphere Application Server V6.1 supports external JMS 1.1 providers
- V6.x supports the following JMS providers:
  - [Default Messaging Provider \(Platform Messaging\)](#)
  - [WebSphere MQ V5.4](#)
  - [Generic JMS Providers](#)

© Copyright IBM Corporation 2007

Figure 15-34. V6 support for external JMS providers

WA3616.3

### **Notes:**

You can install and use one or more of the following messaging providers:

The default messaging provider. This is installed as part of WebSphere Application Server, administered through the administrative console, and managed as part of the WebSphere Application Server runtime.

WebSphere MQ version 5.4 or later

Other “generic” messaging providers

***Instructor notes:***

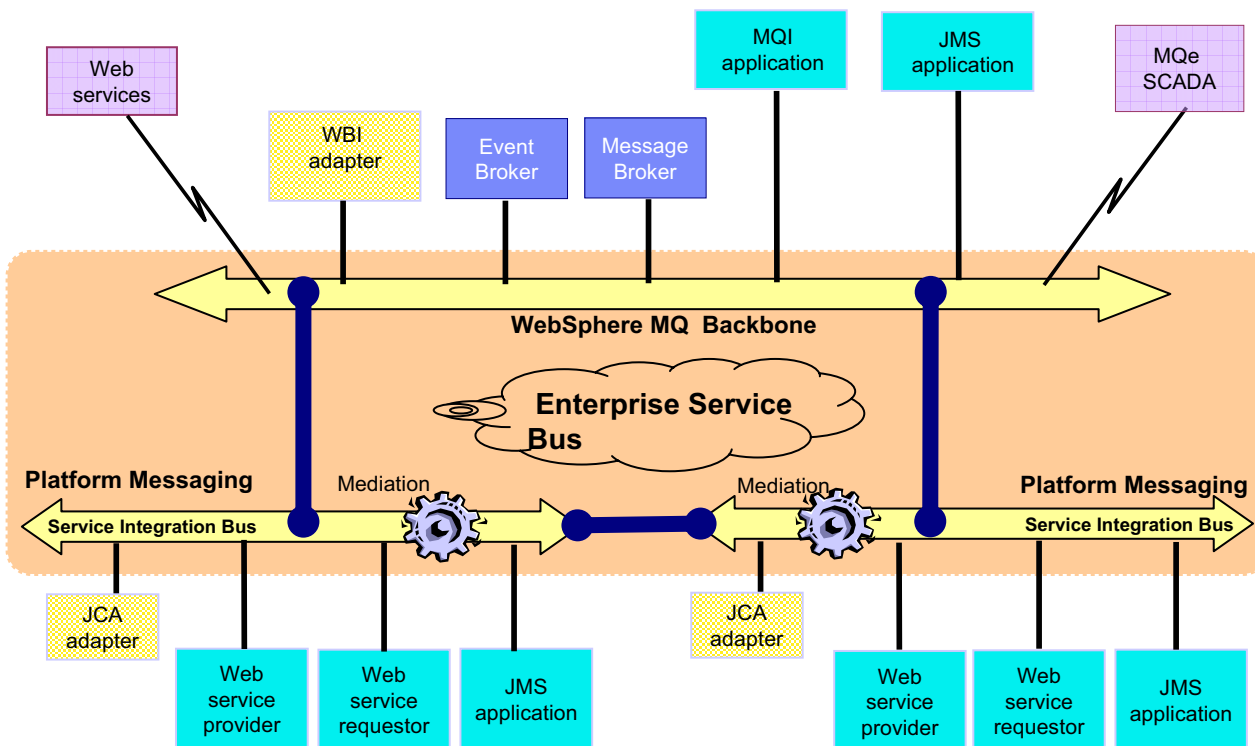
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Service Integration as part of an ESB

## Service Integration as part of an ESB



© Copyright IBM Corporation 2007

Figure 15-35. Service Integration as part of an ESB

WA3616.3

### Notes:

The existing WebSphere messaging products provide a comprehensive range of functions which enable customers to build an ESB today.

This will be augmented by the new Platform Messaging which provides greater support for J2EE and Web Services standards.

Platform Messaging provides a more flexible and integrated messaging solution for the Application Server with connectivity onto the ESB.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** — Supervisor Control and Data Acquisition (SCADA) IBM implementation is MQe middleware.

**Transition statement** — Next: Checkpoint questions

---

## Checkpoint questions

---

1. What is the virtual place within an SIBus to which applications can attach to exchange messages?
2. What is a service that runs inside a server and manages messaging resources?
3. What supports applications using message-based and service-oriented architectures?
4. \_\_\_\_\_ is the ability to manipulate a message as it traverses the SIBus.

© Copyright IBM Corporation 2007

Figure 15-36. Checkpoint questions

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.
- 4.

***Instructor notes:***

Purpose —

Details —

## **Checkpoint solutions**

---

1. Destination
2. Messaging engine
3. SIBus
4. Mediation

© Copyright IBM Corporation 2007

**Additional information** —

**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe what WebSphere Platform Messaging is and how it is used
- Explain which WebSphere Application Server components use WebSphere Platform Messaging
- Describe how JMS and MQ use WebSphere Platform Messaging to provide application messaging services

© Copyright IBM Corporation 2007

Figure 15-37. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

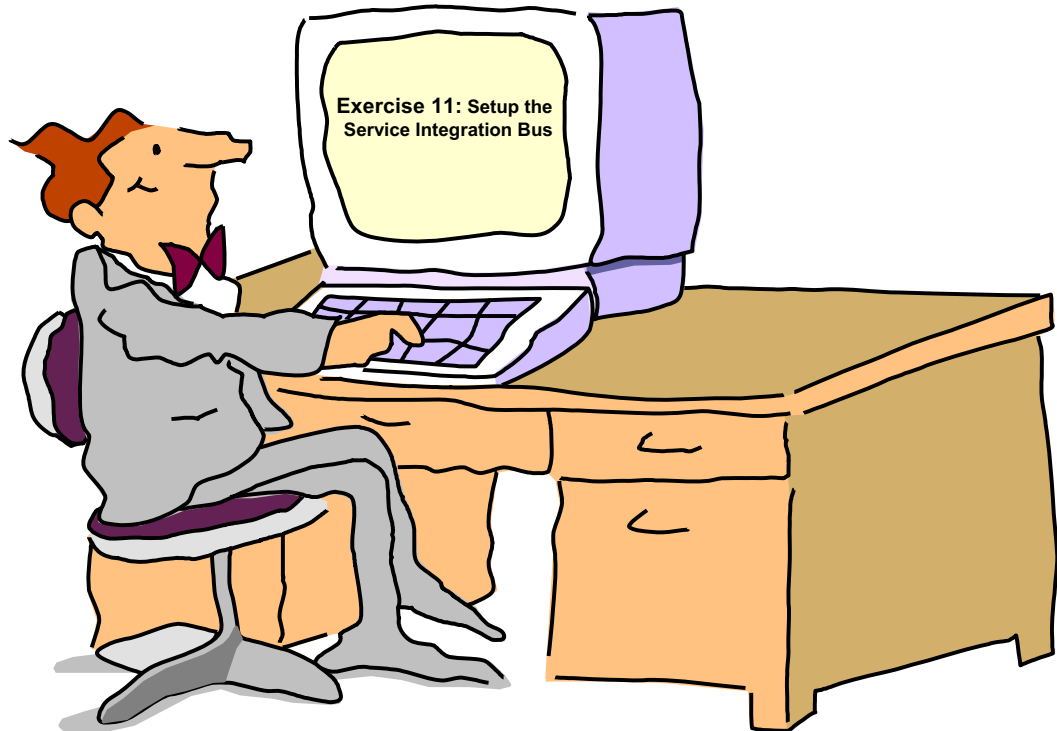
**Additional information —**

**Transition statement —** Next: Lab exercise



## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 15-38. Lab exercise

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

---

# Unit 16. WebSphere security

## Estimated time

01:30

## What this unit is about

This unit describes the WebSphere Security Service that runs on WebSphere Application Server.

## What you should be able to do

After completing this unit, you should be able to:

- Describe basic security concepts
- Describe WebSphere security architecture and concepts
- Configure security settings for console access
- Configure security settings for an application
- Describe SSL and how to configure it

## How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

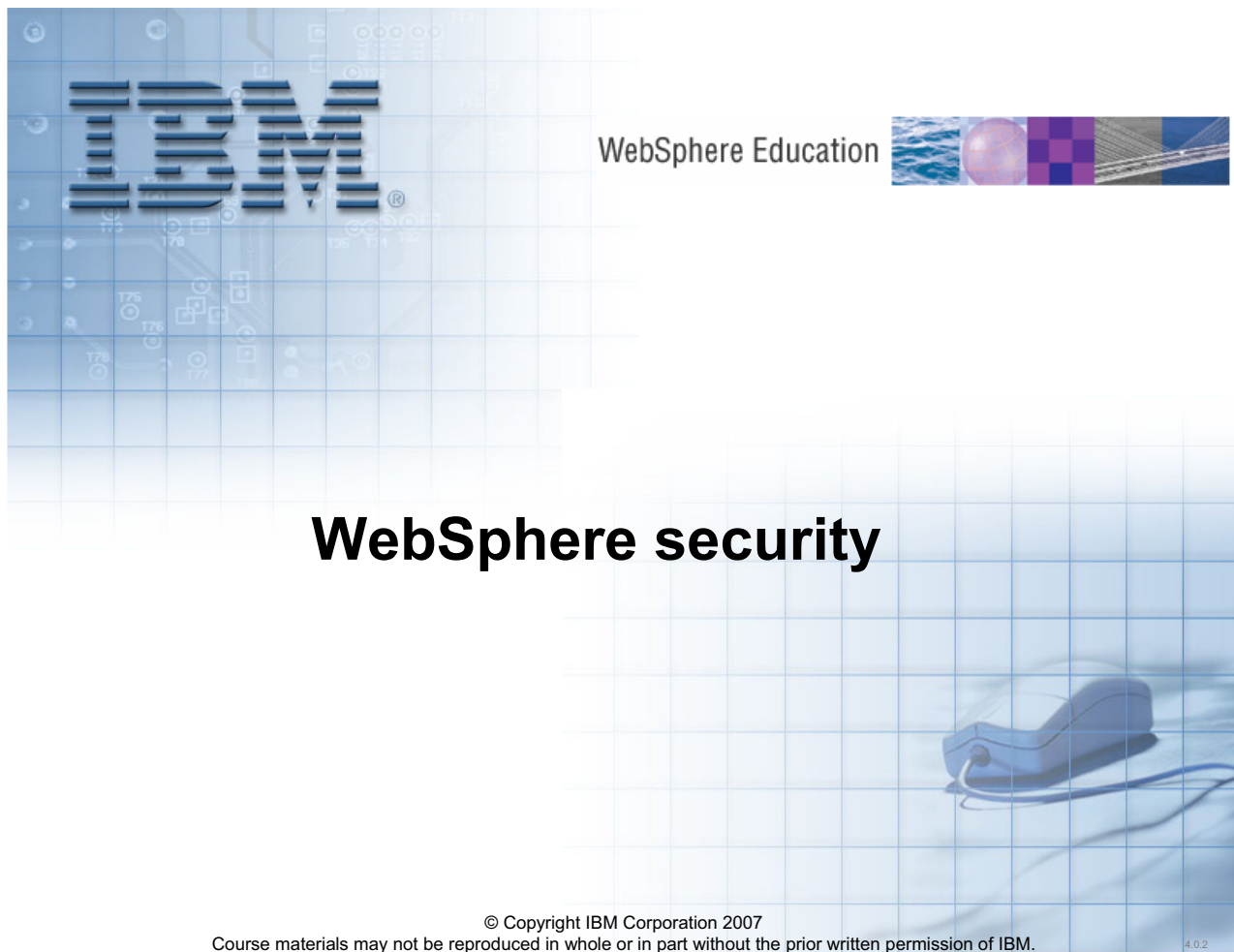


Figure 16-1. WebSphere security

WA3616.3

**Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe basic security concepts
- Describe WebSphere security architecture and concepts
- Configure security settings for console access
- Configure security settings for an application
- Describe SSL and how to configure it

© Copyright IBM Corporation 2007

Figure 16-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

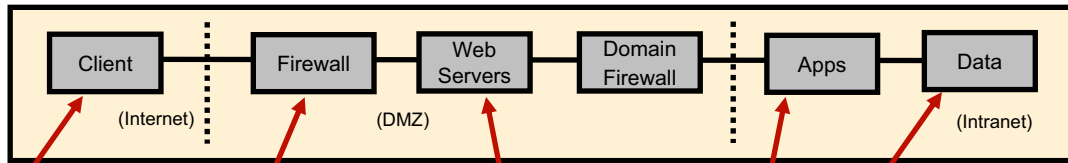
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Basic security end-to-end model

## Basic security end-to-end model



This is the customer. The client can be either a single user PC or a whole company. The devices are specified when necessary.

This is the router to the Internet. It typically provided by an ISP (Internet Service Provider). This model, however, can apply to a corporate Intranet. This router separates the internal network from the outside network.

This is the company's Web server. It provides the universal access to the company's business logic. It allows browser access to the company's corporation. Above the Web server is the organization's company router/access device.

This is the business logic. These applications have access to the databases and provide the company business rules.

This is the company's data. Ultimately this is the information the company wishes to share with the customer. This data can be centralized or distributed. For this example, the data is accessed through business logic.

© Copyright IBM Corporation 2007

Figure 16-3. Basic security end-to-end model

WA3616.3

### Notes:

With an e-business application, a general topology needs to be secured from end-to-end. This involves securing many different parts.



***Instructor notes:***

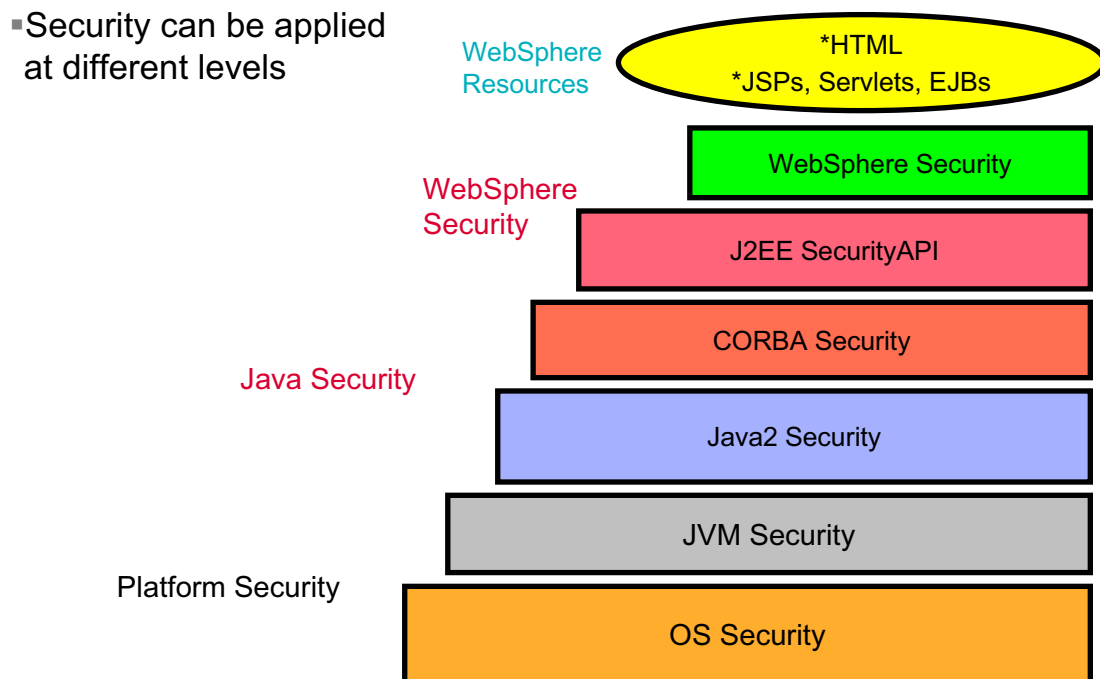
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere Application Server security overview

# WebSphere Application Server security overview



© Copyright IBM Corporation 2007

Figure 16-4. WebSphere Application Server security overview

WA3616.3

## Notes:

There are many levels that are involved in security in an environment. WebSphere only provides part of the total security that needs to be applied. Things like file system security still need to be taken into account to protect things like your configuration files and keyrings.

- **Operating System Security** - The security infrastructure of the underlying operating system provides certain security services to the WebSphere Security Application. This includes the file system security support to secure sensitive files in WebSphere product installation. The WebSphere system administrator can configure the product to obtain authentication information directly from the operating system user registry, for example the NT Security Access Manager - SAM.
- **JVM 1.4** - The JVM security model provides a layer of security above the operating system layer.
- **CORBA Security** - Any calls made among secure ORBs are invoked over a Secure Association Service (SAS) or CSIv2 layer that sets up the security context and the

necessary quality of protection. After the session is established, the call is passed up to the enterprise bean layer. This layer is for DISTRIBUTED platform only

- J2EE Security - The security collaborator enforces J2EE based security policies and support J2EE security APIs.
- WebSphere Security - WebSphere security enforces security policies and services in a unified manner on access to Web resources and enterprise beans. It consists of WebSphere security technologies and features to support the needs of a secure enterprise environment.

***Instructor notes:***

**Purpose —**

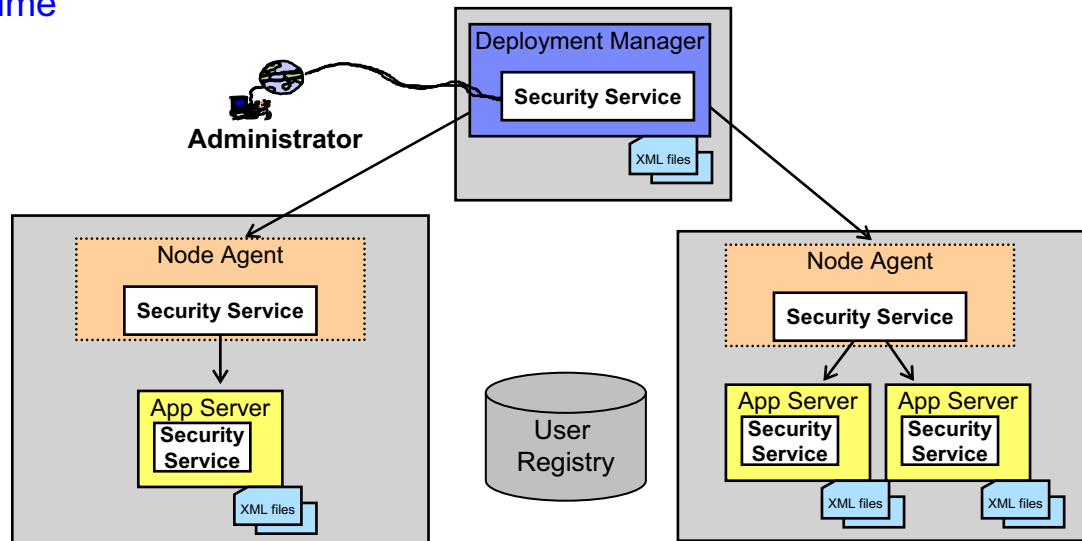
**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere security service – Big picture

## WebSphere security service – Big picture

- Security service runs locally in each process (Deployment Manager, Node Agent, and Application Server)
  - Security workload not bottlenecked to a single process
  - Security service failure only affects a single process
- Separation of authentication mechanism and user registry
  - Only one authentication mechanism and registry can be enabled at a time



© Copyright IBM Corporation 2007

Figure 16-5. WebSphere security service – Big picture

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

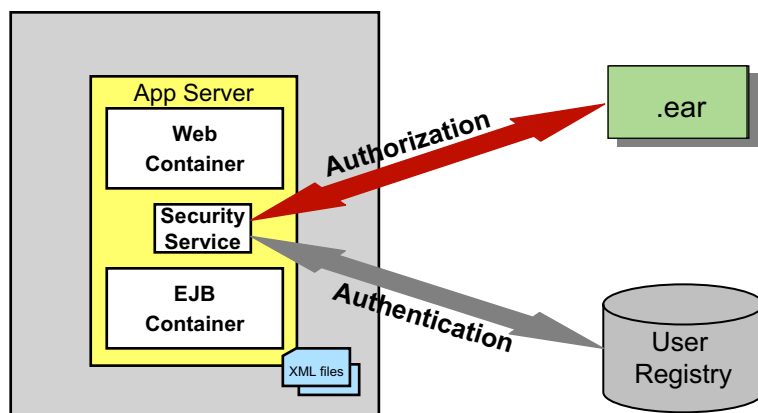
**Details —**

**Additional information —**

**Transition statement —** Next: Authentication and access control

## Authentication and access control

- Authentication
  - Who are you?
- Authorization
  - What are you allowed to do?



© Copyright IBM Corporation 2007

Figure 16-6. Authentication and access control

WA3616.3

### Notes:

Authentication information can be found in a user registry. Authorization information can be found within the ear file. The WebSphere security service is responsible for making sure that protected resources are only accessible by authenticated and correctly authorized users.

***Instructor notes:***

**Purpose —**

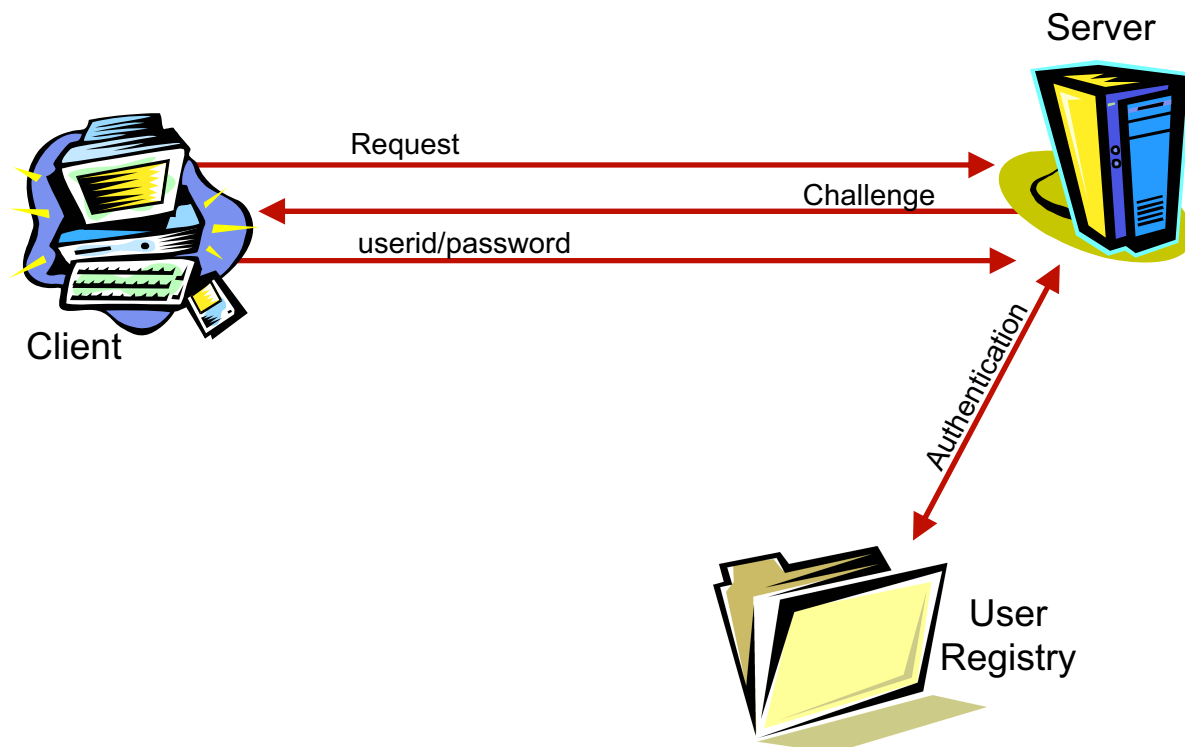
**Details —**

**Additional information —**

**Transition statement —** Next: Authentication basic steps



## Authentication basic steps



© Copyright IBM Corporation 2007

Figure 16-7. Authentication basic steps

WA3616.3

### Notes:

Authentication in its simplest form is rather straight forward.

- A request is made
- A challenge is returned
- A userid/password is sent
- The server checks a user registry to see if the information is valid

***Instructor notes:***

**Purpose —**

**Details —**

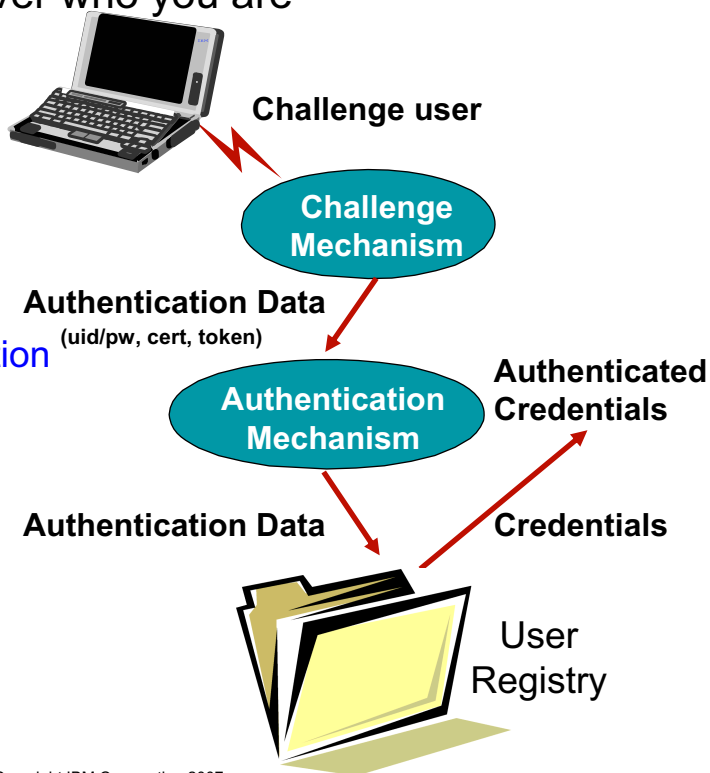
**Additional information —**

**Transition statement —** Next: Authentication

## Authentication

- Authentication: Tell a server who you are

- Challenge mechanism (how to obtain authentication data: basic authentication, certificate-based, form-based)
- Authentication mechanism (how authentication data is passed to a user registry: LTPA)
- User registry (associates credentials with principal: local OS, LDAP, custom, federated)



© Copyright IBM Corporation 2007

Figure 16-8. Authentication

WA3616.3

### Notes:

Within WebSphere, authentication needs to be configured. The configuration includes defining a challenge mechanism (such as a 401 – Basic Authentication), an authentication mechanism (such as LTPA) and a registry (such as LDAP). In order to turn security on within WebSphere, all of these need to be defined.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Basic authentication challenge types

## Basic authentication challenge types

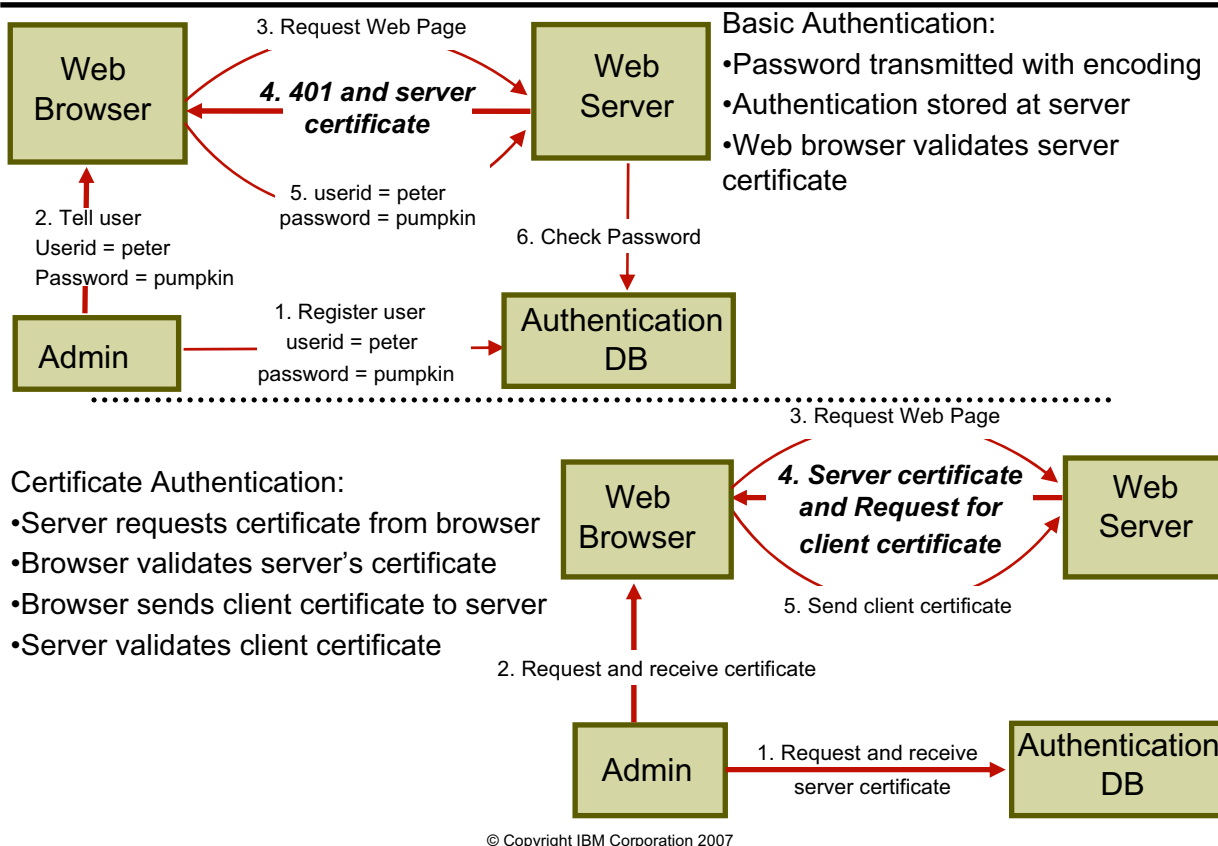


Figure 16-9. Basic authentication challenge types

WA3616.3

### Notes:

“Basic Authentication” through the browser allows the server to initiate a userid/password challenge by responding with a 401. The userid/password is then encoded and sent to the server. This is not terribly secure since the encoding is not very powerful. It is generally recommended that basic authentication be done after an https session has been established.

Other challenge types, such as form based, are also supported. Those will be application specific.

***Instructor notes:***

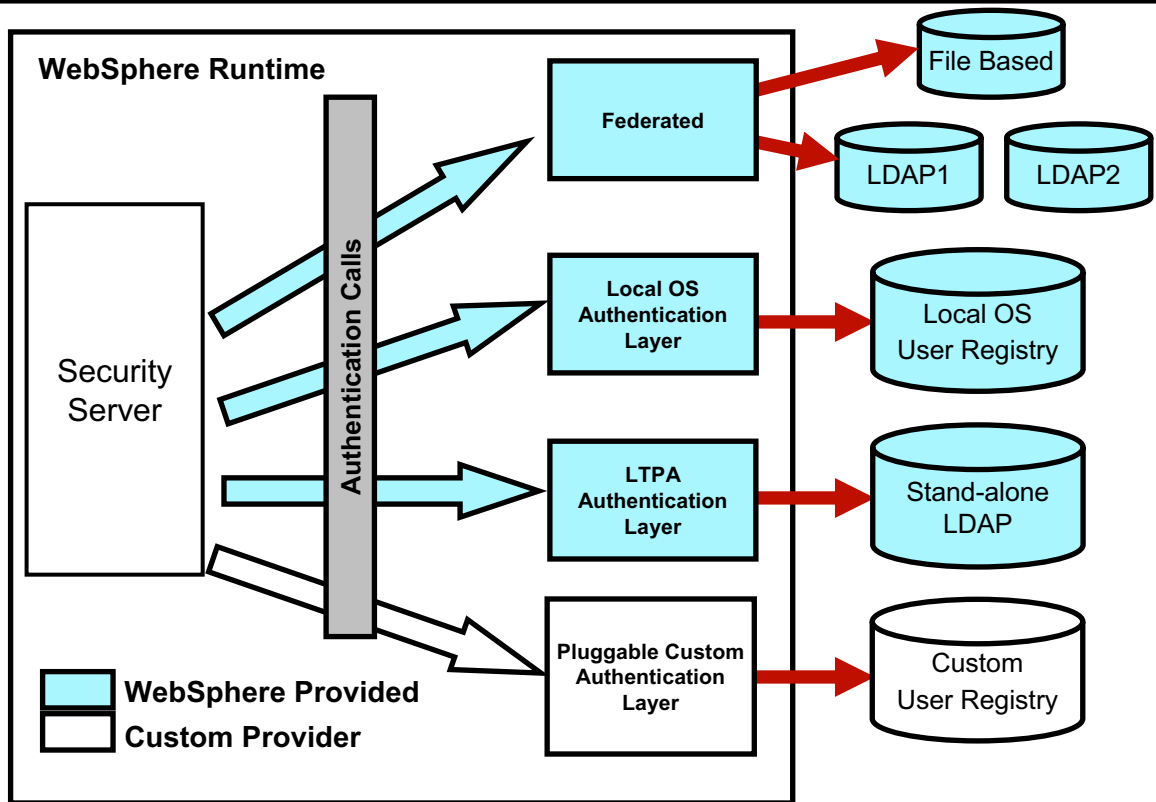
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Registries and authentication mechanisms

# Registries and authentication mechanisms



© Copyright IBM Corporation 2007

Figure 16-10. Registries and authentication mechanisms

WA3616.3

## Notes:

## ***Instructor notes:***

### **Purpose —**

### **Details — Authentication mechanisms**

An *authentication mechanism* defines rules about security information, such as whether a credential is forwardable to another Java process, and the format of how security information is stored in both credentials and tokens.

*Authentication* is the process of establishing whether a client is who or what it claims to be in a particular context. A client can be either a user, a machine, or an application. An authentication mechanism in WebSphere Application Server typically collaborates closely with a *user registry*. The user registry is the user and groups account repository that the authentication mechanism consults with when performing authentication. The authentication mechanism is responsible for creating a *credential*, which is an internal product representation of a successfully authenticated client user. Not all credentials are created equally. The abilities of the credential are determined by the configured authentication mechanism.

LTPA is the only authentication mechanism available in a Network Deployment environment.

### **User Registries**

In WebSphere Application Server, a user registry or repository authenticates a user and retrieves information about users and groups to perform security-related functions, including authentication and authorization. The information about users and groups reside within a registry or repository. WebSphere Application Server makes access control decisions using the user registry or repository.

WebSphere Application Server provides implementations that support multiple types of registries and repositories including the local operating system registry, a stand-alone Lightweight Directory Access Protocol (LDAP) registry, a stand-alone custom registry, and federated repositories.

These authorization mechanism choices are valid for all user registries and repositories, such as virtual member manager, with the exception of Tivoli Access Manager, which is supported for Stand-alone LDAP registry configuration only.

With WebSphere Application Server, a user registry or a repository, such as virtual member manager, authenticates a user and retrieves information about users and groups to perform security-related functions including authentication and authorization.

With WebSphere Application Server, a user registry or repository is used for:

Authenticating a user using basic authentication, identity assertion, or client certificates

Retrieving information about users and groups to perform security-related administrative functions, such as mapping users and groups to security roles



Although WebSphere Application Server supports different types of user registries, only one user registry can be active. This active registry is shared by all of the product server processes.

### **Stand-alone Custom User Registries**

A *stand-alone custom registry* is a customer-implemented registry that implements the UserRegistry Java interface, as provided by the product. A custom-implemented registry can support virtually any type of an account repository from a relational database, flat file, and so on. The custom user registry provides considerable flexibility in adapting product security to various environments where some form of a registry or repository other than federated repositories, stand-alone Lightweight Directory Access Protocol (LDAP) registry or local operating system registry already exists in the operational environment.

WebSphere Application Server security provides an implementation that uses various local operating system-based registries and various stand-alone Lightweight Directory Access Protocol (LDAP)-based registries. However, situations can exist where your user and group data resides in other repositories or custom user registries, such as a database, and moving this information to either a local operating system registry or a stand-alone LDAP registry implementation might not be feasible. For these situations, WebSphere Application Server security provides a service provider interface (SPI) that you can implement to interact with your current registry. The custom registry feature supports any user registry that is not implemented by WebSphere Application Server. A *stand-alone custom registry* is a customer-implemented registry that implements the UserRegistry Java interface, as provided by the product. A custom-implemented registry can support virtually any type of an account repository from a relational database, flat file, and so on. The custom user registry provides considerable flexibility in adapting product security to various environments where some form of a registry or repository other than federated repositories, stand-alone Lightweight Directory Access Protocol (LDAP) registry or local operating system registry already exists in the operational environment.

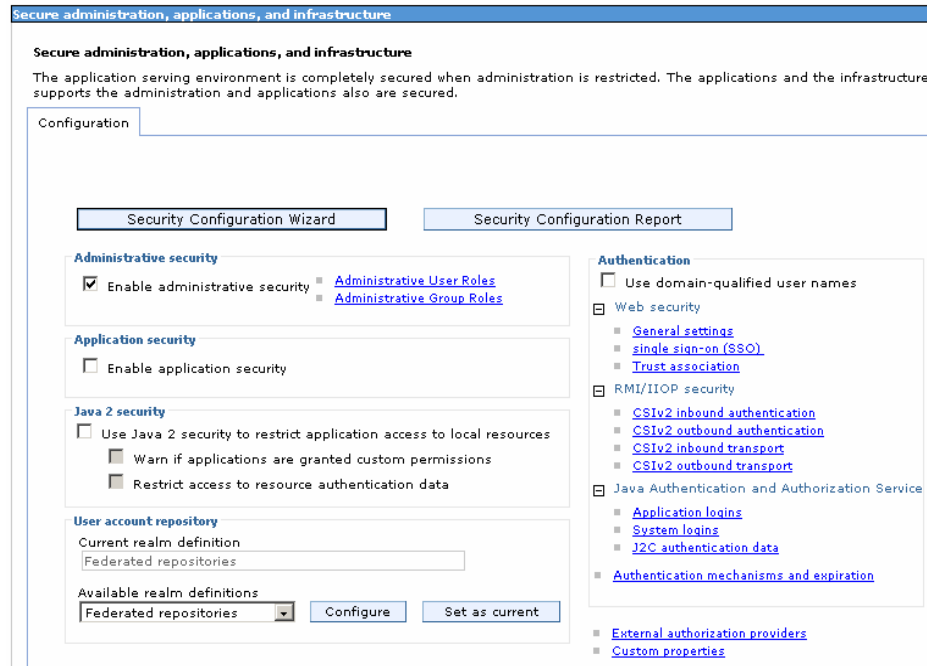
WebSphere Application Server security provides an implementation that uses various local operating system-based registries and various stand-alone Lightweight Directory Access Protocol (LDAP)-based registries. However, situations can exist where your user and group data resides in other repositories or custom user registries, such as a database, and moving this information to either a local operating system registry or a stand-alone LDAP registry implementation might not be feasible. For these situations, WebSphere Application Server security provides a service provider interface (SPI) that you can implement to interact with your current registry. The custom registry feature supports any user registry that is not implemented by WebSphere Application Server.

### **Additional information —**

**Transition statement —** Next: WebSphere administrative and application security

# WebSphere administrative and application security

- WebSphere administrative security is enabled by default during profile creation.
- Application and Java 2 security are disabled by default.



© Copyright IBM Corporation 2007

Figure 16-11. WebSphere administrative and application security

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Federated repositories

## Federated repositories

---

- The installation wizard and profile management tool have a default of enabling administrative security.
  - The default repository type is a file based federated repository.
- Federated repositories enables the use of multiple repositories with WebSphere Application Server.
  - Can be file-base, LDAP, multiple LDAPs or a sub-tree of an LDAP.
  - Defined and theoretically combined under a single realm.
  - All of the user repositories that are configured under the federated repository functionality are invisible to WebSphere Application Server.
- Federation capabilities provided by the VMM (Virtual Member Manager).

© Copyright IBM Corporation 2007

Figure 16-12. Federated repositories

WA3616.3

### **Notes:**

***Instructor notes:***

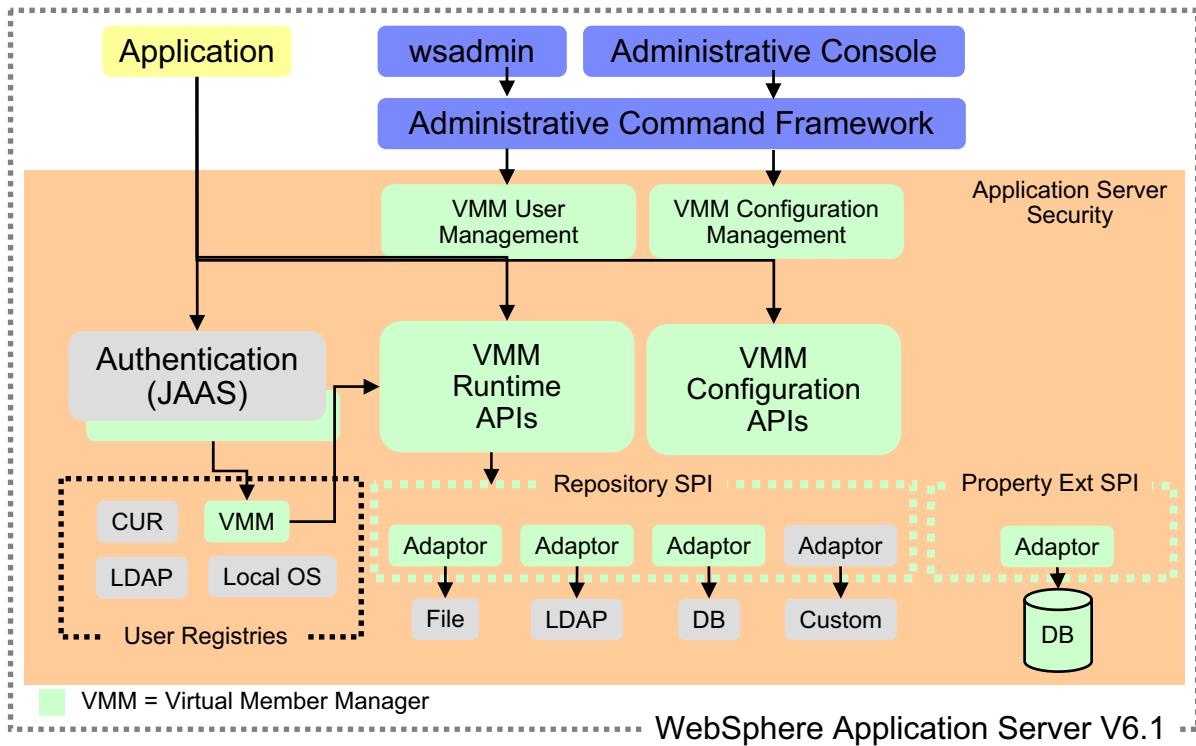
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Federated repositories: architecture overview

# Federated repositories: architecture overview



© Copyright IBM Corporation 2007

Figure 16-13. Federated repositories: architecture overview

WA3616.3

## Notes:

The diagram on this slide illustrates how the VMM fits into the overall application server security architecture.

**Instructor notes:****Purpose —**

**Details —** *Federated repositories* enable you to use multiple repositories with WebSphere Application Server. These repositories, which can be file-based repositories, LDAP repositories, or a sub-tree of an LDAP repository, are defined and theoretically combined under a single realm. All of the user repositories that are configured under the federated repository functionality are invisible to WebSphere Application Server.

When you use the federated repositories functionality, all of the configured repositories, which you specify as part of the federated repository configuration, become active. It is required that the user ID, and the distinguished name (DN) for an LDAP repository, be unique in multiple user repositories that are configured under the same federated repository configuration. For example, there might be three different repositories that are configured for the federated repositories configuration: Repository A, Repository B, and Repository C. When user1 logs in, the federated repository adapter searches each of the repositories for all of the occurrences of that user. If multiple instances of that user are found in the combined repositories, an error message displays.

In addition, the federated repositories functionality in WebSphere Application Server supports the logical joining of entries across multiple user repositories when the Application Server searches and retrieves entries from the repositories. For example, when an application calls for a sorted list of people whose age is greater than twenty, WebSphere Application searches all of the repositories in the federated repositories configuration. The results are combined and sorted before the Application Server returns the results to the application.

**Additional information —**

**Transition statement —** Next: User registry support

# User registry support

## User Registries Supported by WebSphere Application Server

Local OS	LDAP
NT Domain, NT WorkGroup, Windows 2000	IBM Tivoli Directory Server
AIX	IBM SecureWay Directory
Solaris	Sun ONE
HP-UX	IBM Lotus Domino
Linux	Microsoft Active Directory
OS/400	Novell eDirectory
	Others (using custom config) not "supported"

© Copyright IBM Corporation 2007

Figure 16-14. User registry support

WA3616.3

### Notes:

Other LDAP servers can also be used by defining the appropriate schema mappings through the Advanced LDAP properties.

Local OS should be avoided in distributed, non domain, environments.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Custom registry

## Custom registry

---

- Allows custom implementation of user registry
  
- Some possible implementations:
  - Database
  - Flat file
  - OS-based, with additional custom logic
  - Use other, not directly supported, registries
  
- WebSphere provides:
  - Base types
    - Implementing classes extend the `com.ibm.websphere.security.UserRegistry` class
  - Working sample implementation
    - `com.ibm.websphere.security.FileRegistrySample`

© Copyright IBM Corporation 2007

Figure 16-15. Custom registry

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Custom registry - Configuration

## Custom registry: Configuration

- Configured from administrative console:
  - Security → Secure administration, application, and infrastructure
  - Select Standalone custom registry from the Available realm definitions
  - Click Configure
    - User ID and password must exist
    - Classname must be implemented and in class path

### [Secure administration, applications, and infrastructure](#) > Standalone custom registry

Specifies a custom registry that implements the UserRegistry interface in the com.ibm.websphere.security package. For backward compatibility, the application server also supports a custom registry that implements the CustomRegistry interface in the com.ibm.websphere.security package. When security is enabled and any of the properties on this panel are changed, go to the Security > Secure administration, applications, and infrastructure panel. Click Apply or OK to validate the changes.

© Copyright IBM Corporation 2007

Figure 16-16. Custom registry - Configuration

WA3616.3

### Notes:

At the time the server is configured the custom registry implementation classes must be coded and available on the server's class path. Also the user and password defined here must be defined and available to the implementing class because when enabling security with a custom registry (or any other registry) WebSphere Application Server will check and make sure the defined user can be authenticated, if it cannot be authenticated you will not be able to enable security. This is a safety feature which prevents you from enabling security and not being able to log back in the next time the server is restarted.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Authentication mechanisms

# Authentication mechanisms

Authentication Mechanism	Intended Use and Supported Package
<p><b>Lightweight Third Party Authentication (LTPA) mechanism</b> Available on all platforms and packages</p>	<ul style="list-style-type: none"> <li>• For distributed, multiple application server environments</li> <li>• Support <i>forwardable</i> credentials or single sign-on (SSO) through cryptography</li> <li>• Requires all the servers authentication registry to be a centrally shared registry like LDAP</li> <li>• Uses cryptographic keys (LTPA keys) to encrypt and decrypt user data that passes between the servers                             <ul style="list-style-type: none"> <li>- If Servers are in different cell, the LTPA keys need to be shared                                     <ul style="list-style-type: none"> <li>• Generate, Export and Import LTPA keys in the admin console</li> </ul> </li> <li>- All servers in the domain must be synchronized</li> </ul> </li> </ul>
<p><b>Simple WebSphere Authentication Mechanism (SWAM)</b> - Not available and not needed in WebSphere Application Server V6 Network Deployment and higher packages - Deprecated in V6.1</p>	<ul style="list-style-type: none"> <li>• For simple, non-distributed, single application server environments</li> <li>• Does not support <i>forwardable</i> credentials or single sign-on (SSO)</li> <li>• Caller identity is not forwarded from client on one server to EJB on another server - What gets forwarded in unauthenticated credential which may fail on the receiving server</li> </ul>

© Copyright IBM Corporation 2007

Figure 16-17. Authentication mechanisms

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

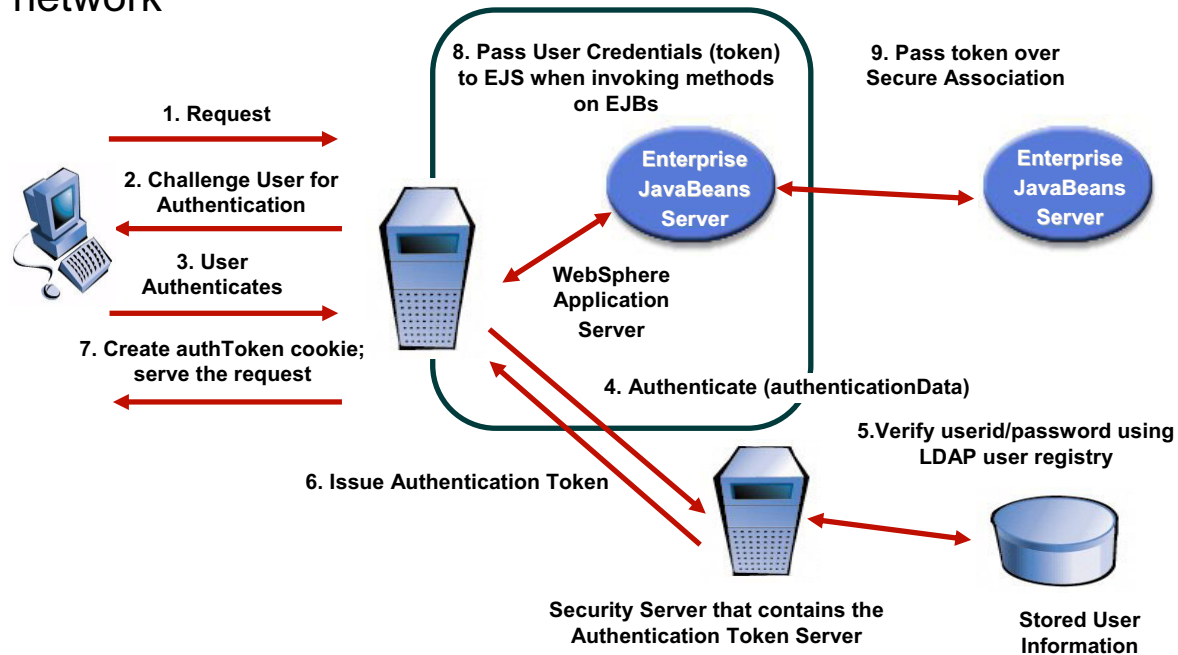
**Details —**

**Additional information —**

**Transition statement —** Next: LTPA and LDAP

# LTPA and LDAP

- Allows a user's identity to be passed around the distributed network



© Copyright IBM Corporation 2007

Figure 16-18. LTPA and LDAP

WA3616.3

## Notes:

Regardless of what registry type is used, either an LTPA or SWAM token will be generated.



***Instructor notes:***

**Purpose —**

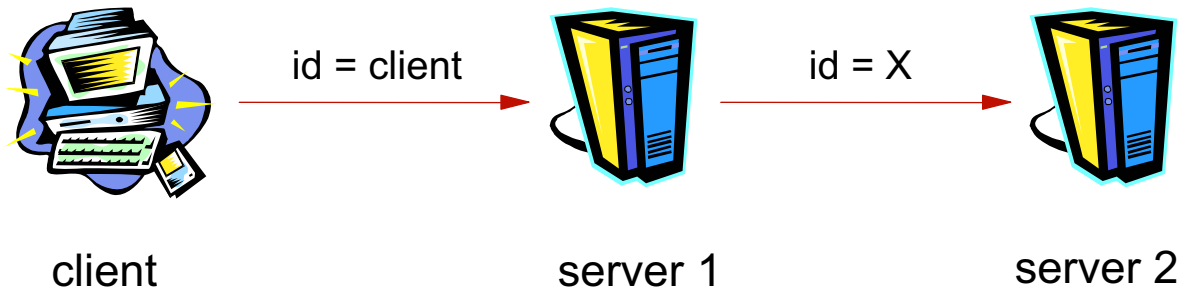
**Details —**

**Additional information —**

**Transition statement —** Next: Delegation

## Delegation

---



**X can run as:**

**Option 1. Client**

**Option 2. Server 1**

**Option 3. Specified identity**

- Using a LTPA token supports Delegation

© Copyright IBM Corporation 2007

Figure 16-19. Delegation

WA3616.3

### **Notes:**

A request for work can pass its security context containing its credentials. As the call proceeds through the servers to its final destination, credentials can be changed. The options are to keep the clients credentials, switch to the server's credentials, or some other specified identity.

***Instructor notes:***

**Purpose —**

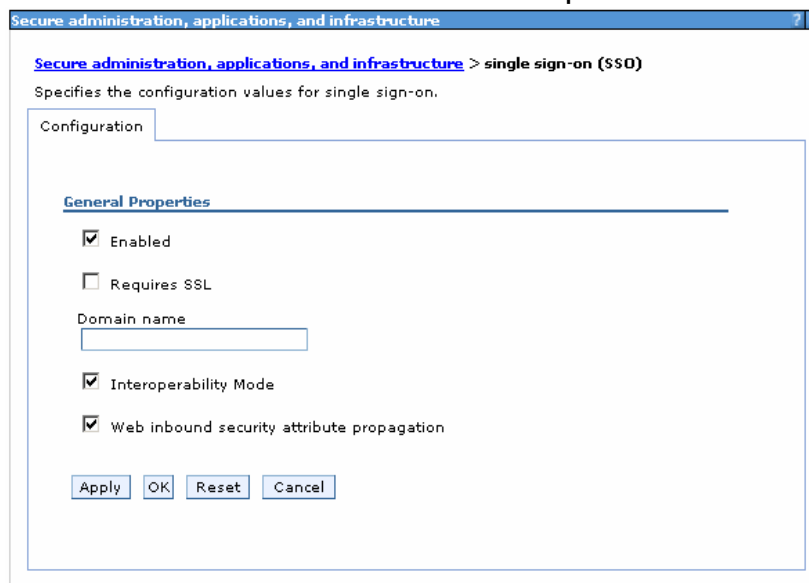
**Details —**

**Additional information —**

**Transition statement —** Next: Single sign-on (SSO)

## Single sign-on (SSO)

- Once a client has a valid LTPA token, they do not need to reauthenticate within a cell
- SSO is on by default
- Issues cookies to Web browser to track user authentication information
- Provides for SSO within or even between WebSphere cells



© Copyright IBM Corporation 2007

Figure 16-20. Single sign-on (SSO)

WA3616.3

### Notes:

User authenticates only once in a DNS domain and can access resources in other WebSphere Application Server cells without getting prompted again.

Requires LTPA across the cells within the domain participating in SSO

Same realm names on each system in the SSO domain.

For local OS:

On the Windows platform, the realm name is the domain name, if a domain is in use, or the machine name

On the UNIX platform, the realm name is the same as the host name.

For LDAP: the realm name is the host:port of the LDAP server.

***Instructor notes:***

**Purpose —**

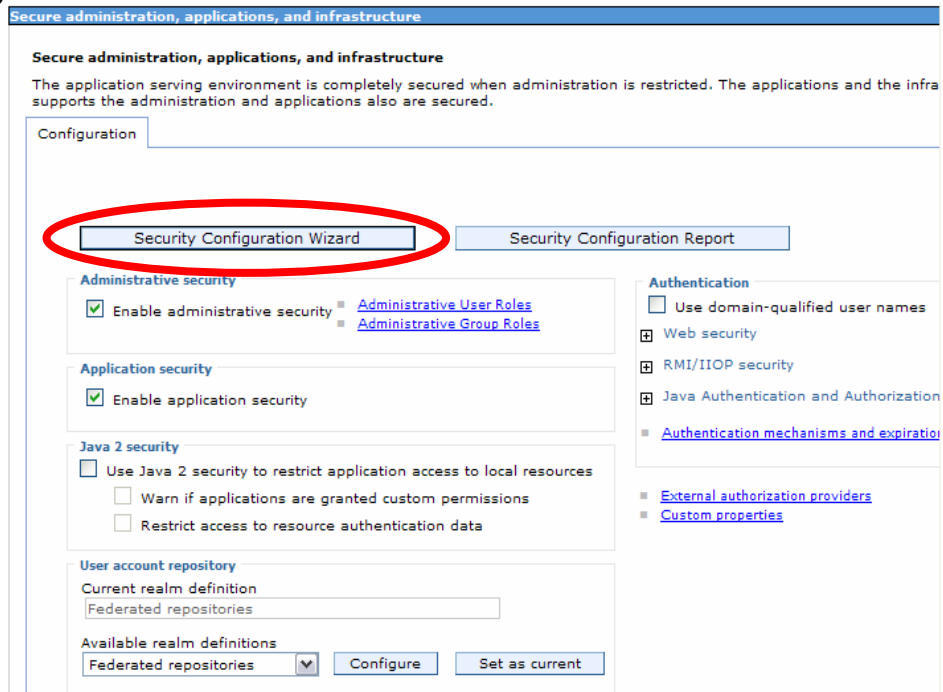
**Details —**

**Additional information —**

**Transition statement —** Next: Security wizard

# Security wizard

A security wizard exists to help step through the process of setting up initial security



© Copyright IBM Corporation 2007

Figure 16-21. Security wizard

WA3616.3

## Notes:

***Instructor notes:***

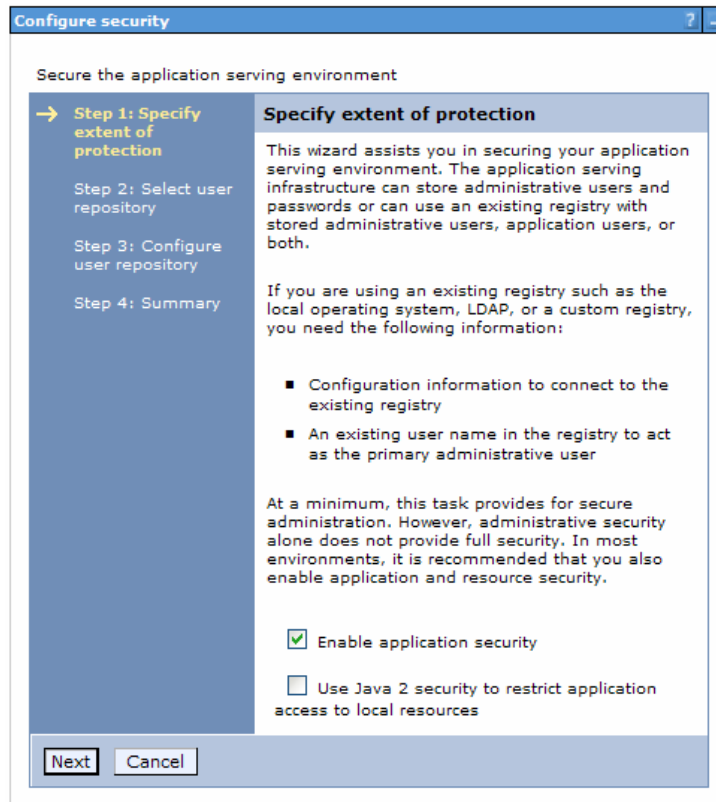
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Security wizard – step 1

# Security wizard – step 1



© Copyright IBM Corporation 2007

Figure 16-22. Security wizard – step 1

WA3616.3

## Notes:



***Instructor notes:***

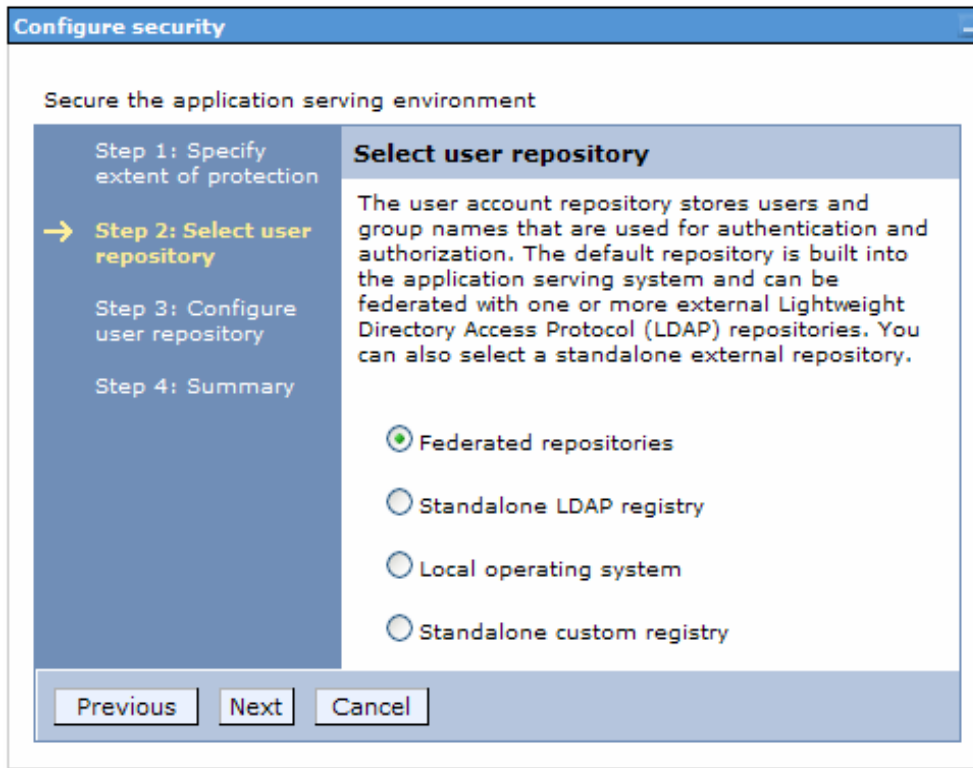
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Security wizard – step 2

## Security wizard – step 2



© Copyright IBM Corporation 2007

Figure 16-23. Security wizard – step 2

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Security wizard – step 3

## Security wizard – step 3

The screenshot shows a window titled "Configure security" with a help icon and a close button. The main content area is titled "Secure the application serving environment". On the left, a vertical sidebar lists four steps: "Step 1: Specify extent of protection", "Step 2: Select user repository", "Step 3: Configure user repository" (highlighted with a yellow arrow), and "Step 4: Summary". The main panel is titled "Configure user repository" and contains the following text: "A secure, file-based user repository is built into the system for storing administrative users or environments with a small number of users. The file-based user repository can be federated with one or more external LDAP repositories. If this is the first time security has been enabled using this repository, provide a new user name and password to act as an administrator. If security was previously enabled using this repository, provide the name of a user with administrator privileges that is in the built-in repository." Below this is a note: "Note: Use this panel to configure a federated repository with a built-in, file-based repository in the realm. To configure a federated repository with a non file-based repository in the realm, you must use the User accounts repository section on the Secure administration, applications, and infrastructure panel." At the bottom of the main panel are three input fields: "\* Primary administrative user name" with the value "wasadmin", "Password" with masked characters "•••••", and "Confirm password" with masked characters "•••••". At the bottom of the window are three buttons: "Previous", "Next", and "Cancel".

© Copyright IBM Corporation 2007

Figure 16-24. Security wizard – step 3

WA3616.3

### Notes:

***Instructor notes:***

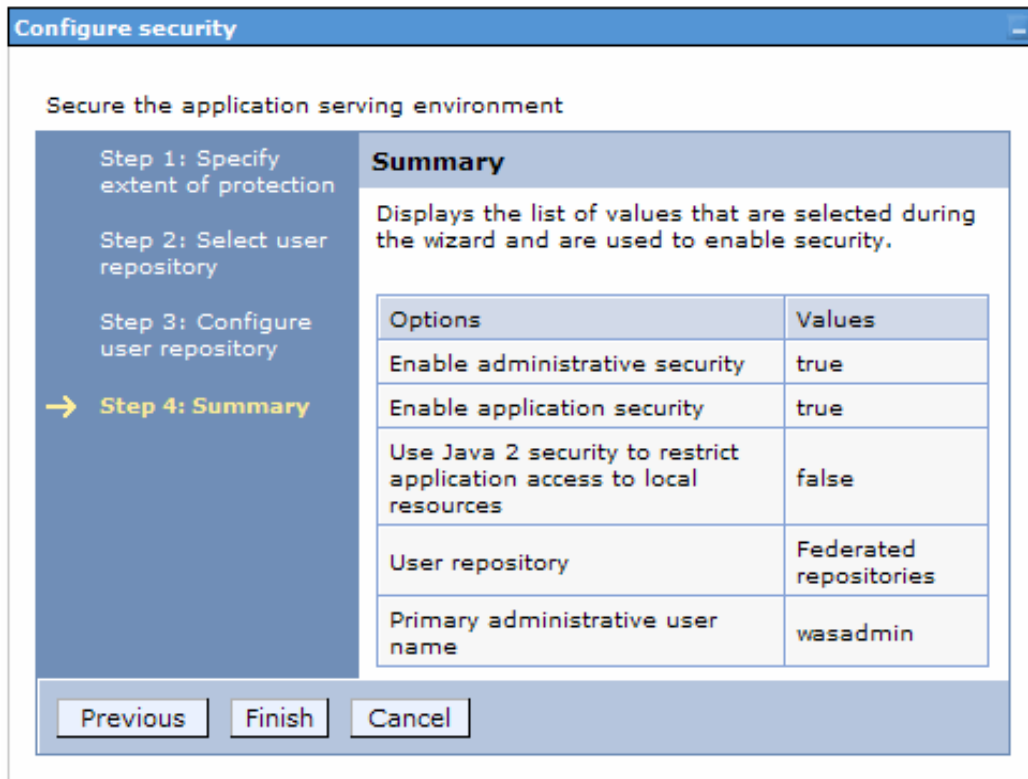
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Security wizard – step 4

## Security wizard – step 4



© Copyright IBM Corporation 2007

Figure 16-25. Security wizard – step 4

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

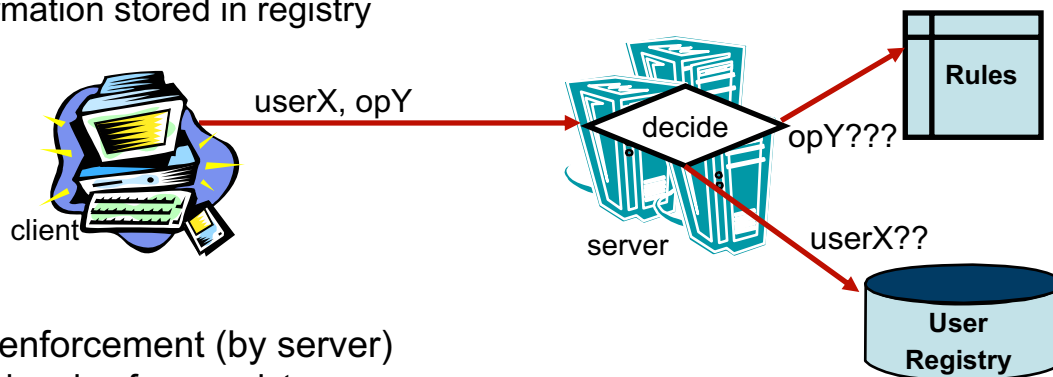
**Details —**

**Additional information —**

**Transition statement —** Next: Authorization

## Authorization

- Authorization involves granting trusted principals permission to perform actions on resources (Web pages, servlets, JSPs and EJBs)
- Control access to resources
  - Security lookup (by server)
    - Determine security privileges for principal
    - Information stored in registry



- Rule enforcement (by server)
  - Obtain rules from registry
  - Given privileges of principal and rules, determine access

© Copyright IBM Corporation 2007

Figure 16-26. Authorization

WA3616.3

### Notes:

The rules engine can be provided by J2EE security through the WebSphere Application Server. Alternatively, a third party authorization service (such as Tivoli Access Manager) can be used.



***Instructor notes:***

**Purpose —**

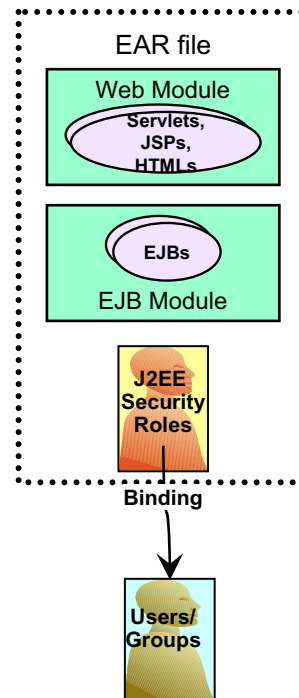
**Details —**

**Additional information —**

**Transition statement —** Next: J2EE security roles: Application authorization

## J2EE security roles: Application authorization

- Authorization is performed using the J2EE security roles
  - Specify security at an abstract level without knowledge of actual users and groups
- Security roles are then applied to the Web and EJB application components
  - Web URIs or EJB methods
- Binding of the users and groups to the J2EE security roles is usually done at the application install time



© Copyright IBM Corporation 2007

Figure 16-27. J2EE security roles: Application authorization

WA3616.3

### Notes:

J2EE security is concerned with controlling access to application resources, not system resources.

***Instructor notes:***

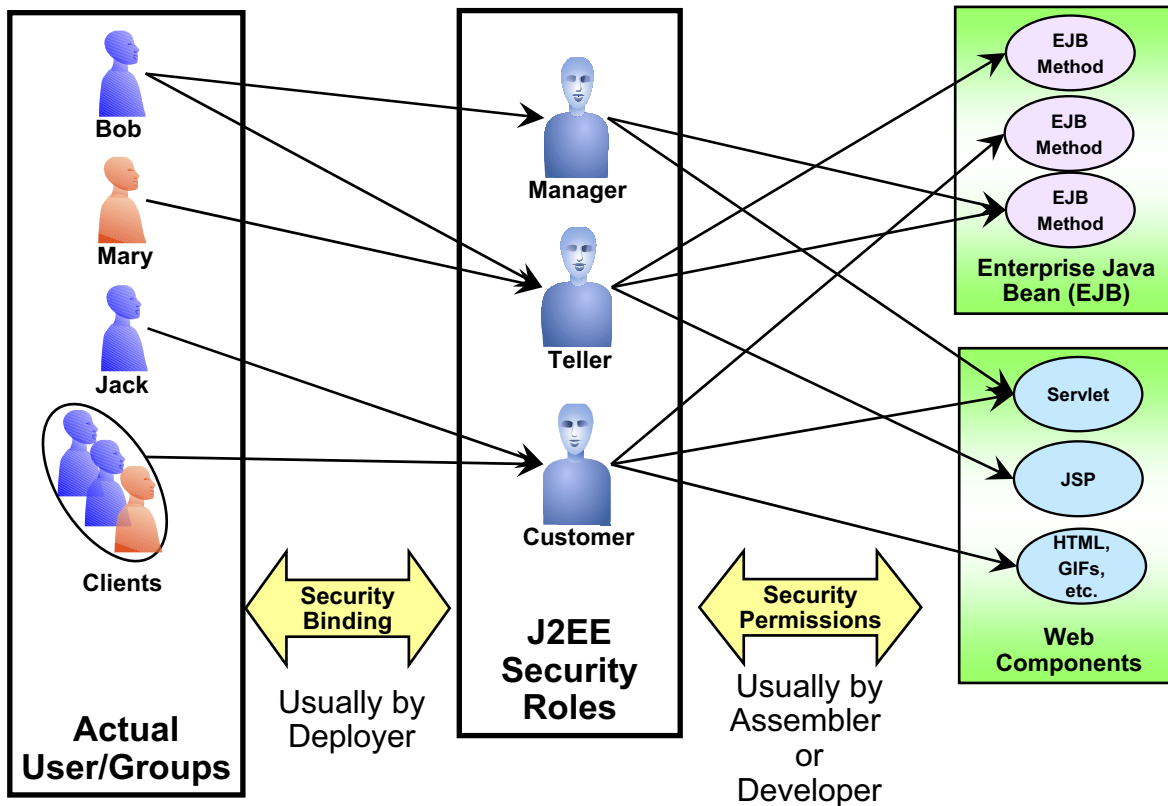
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Securing J2EE application artifacts

# Securing J2EE application artifacts



© Copyright IBM Corporation 2007

Figure 16-28. Securing J2EE application artifacts

WA3616.3

## Notes:

J2EE security is fairly granular, usually to the method level of a resource.

***Instructor notes:***

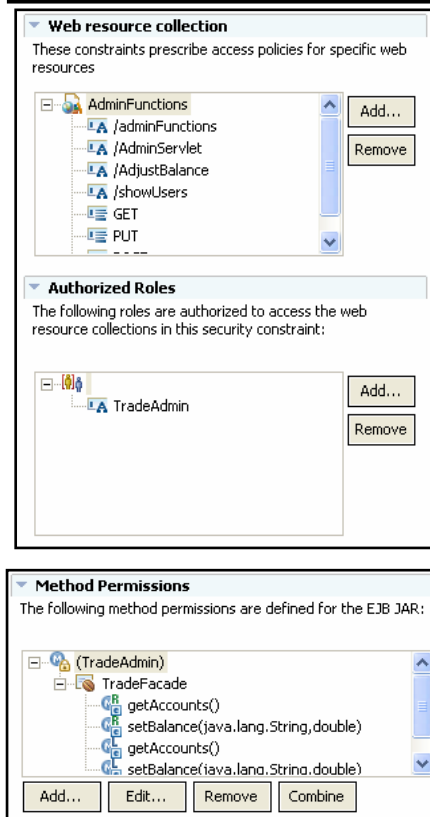
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Applying J2EE security with AST

## Applying J2EE security with AST



- J2EE security can be applied to resources within an application
  - J2EE security roles are defined in the application deployment descriptor
  - Servlets and JSPs are protected with security constraints, which are mapped to the security roles
  - EJBs are protected with method permissions, which are mapped to the security roles
- The J2EE security roles are then mapped to actual users and groups during installation of the application

© Copyright IBM Corporation 2007

Figure 16-29. Applying J2EE security with AST

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

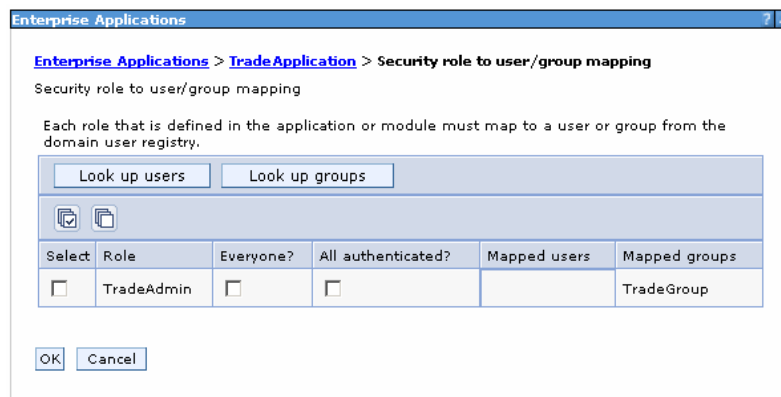
**Details** —

**Additional information** —

**Transition statement** — Next: Applying J2EE security roles using the console

## Applying J2EE security roles using the console

- The mapping of users and groups to J2EE security roles can take place during or after application installation
  - After installation, using the administrative console, go to the application and under Detailed Properties, select **Security role to user/group mapping**
  - Select the role and either
    - Lookup and apply users or groups
    - Map to Everyone
    - Map to All authenticated



© Copyright IBM Corporation 2007

Figure 16-30. Applying J2EE security roles using the console

WA3616.3

### Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Console security

## Console security

- Defines which roles have access to the administrative tools
  - **Monitor:** Least privileged; allows a user to view the WebSphere configuration and current application server state
  - **Configurator:** Monitor privilege plus the ability to change the WebSphere configuration.
  - **Operator:** Monitor privilege plus the ability to change runtime state, such as starting or stopping servers
  - **Administrator:** Operator, configurator, and iscadmins privilege, plus additional privileges granted solely to the administrator role, such as:
    - Modifying the primary administrative user and password
    - Mapping users and groups to the administrator role
    - Enabling or disabling administrative and Java 2 security

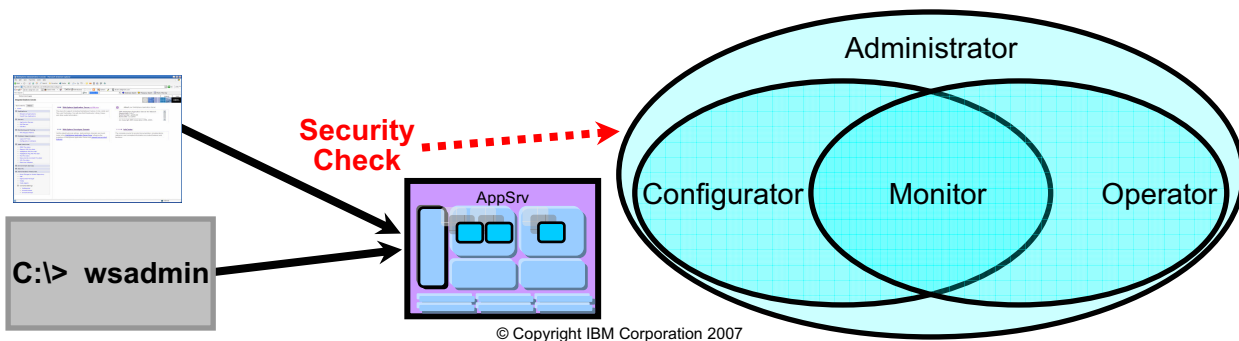


Figure 16-31. Console security

WA3616.3

### Notes:

Once global security is enabled, there is a security check when the console application is accessed. The security check makes sure that the accessing user is authenticated and has been mapped to one of the four console security roles. Depending on the console role to which the user is mapped, different functions will be available.

Whichever userid is used to run the application server process has implicit access as a console administrator user.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Additional console security roles

## Additional console security roles

---

- **iscadmins** (Integrated Solutions Console)
  - *Only available for administration console users*
  - allows a user to manage users and groups in the federated repositories
- **Deployer**
  - *Only available for wsadmin users (not administration console)*
  - allows a user to change configuration and runtime state on applications using wsadmin
- **Admin Security Manager**
  - *Only available for wsadmin users*
  - allows a user to map users to administrative roles using wsadmin
  - when restricted access to resource authentication data is in effect, users can also manage authorization groups

© Copyright IBM Corporation 2007

Figure 16-32. Additional console security roles

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

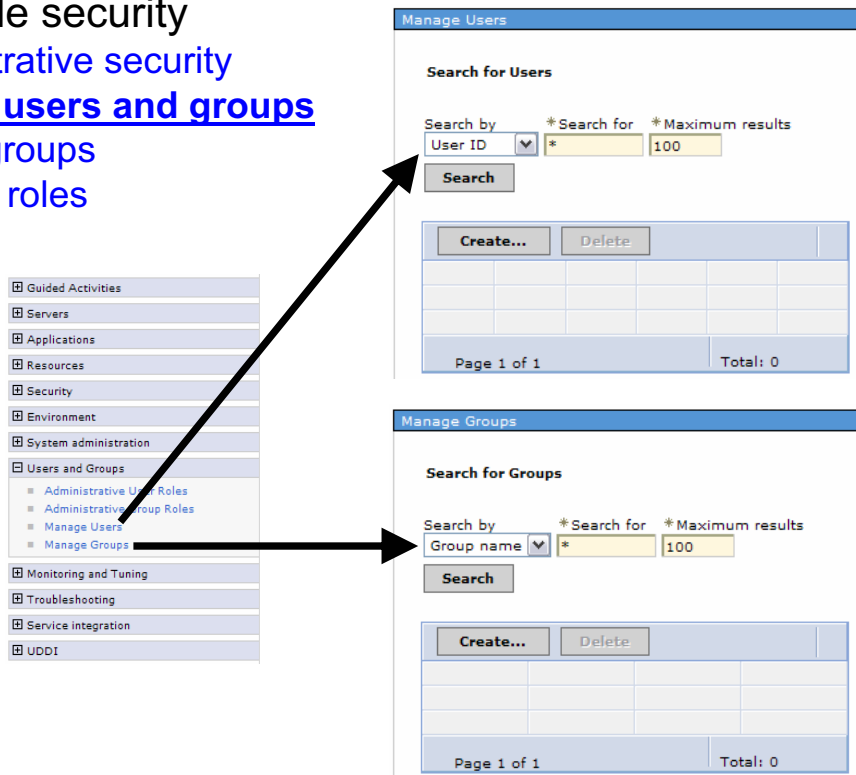
**Details —**

**Additional information —**

**Transition statement —** Next: Console security – creating users and groups

# Console security – creating users and groups

- To set up console security
  - Turn on administrative security
  - **Create console users and groups**
  - Map users and groups to administrative roles



© Copyright IBM Corporation 2007

Figure 16-33. Console security – creating users and groups

WA3616.3

## Notes:

***Instructor notes:***

**Purpose** —

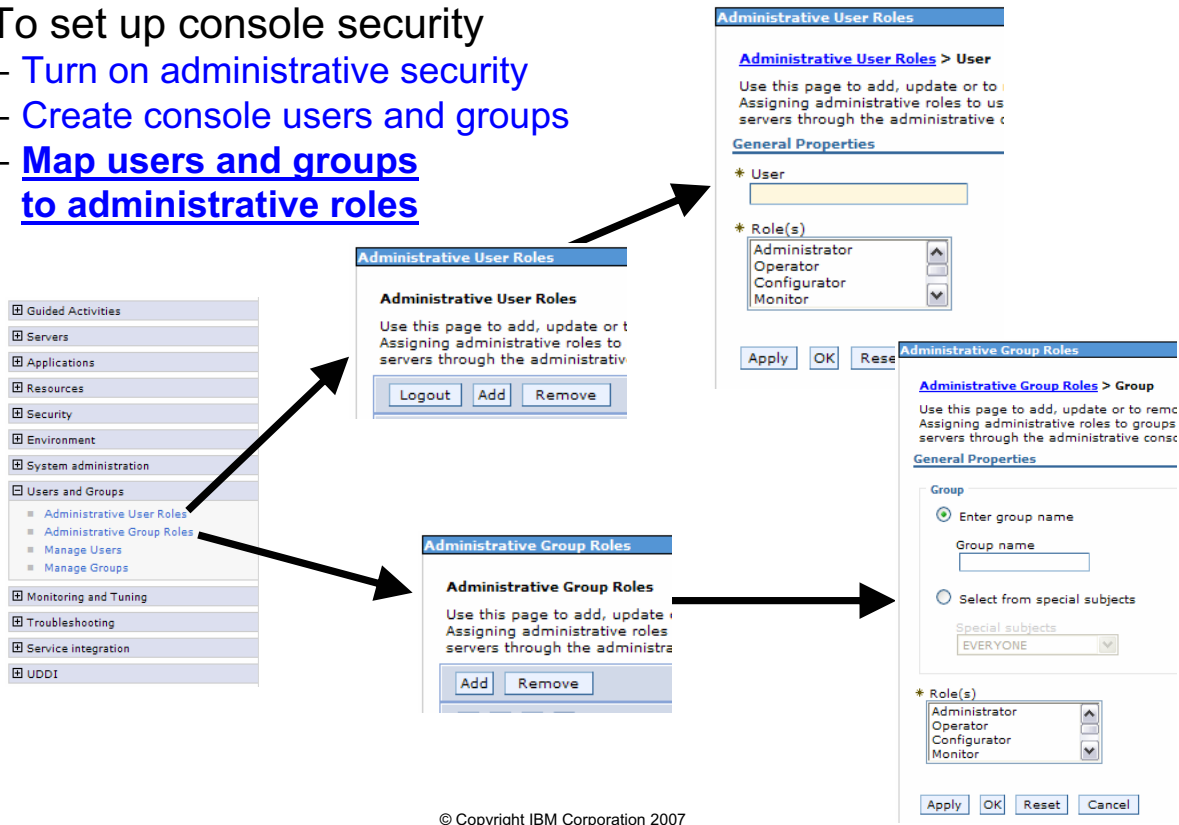
**Details** —

**Additional information** —

**Transition statement** — Next: Console security – mapping users and groups

# Console security – mapping users and groups

- To set up console security
  - Turn on administrative security
  - Create console users and groups
  - **Map users and groups to administrative roles**



© Copyright IBM Corporation 2007

Figure 16-34. Console security – mapping users and groups

WA3616.3

## Notes:



***Instructor notes:***

**Purpose —**

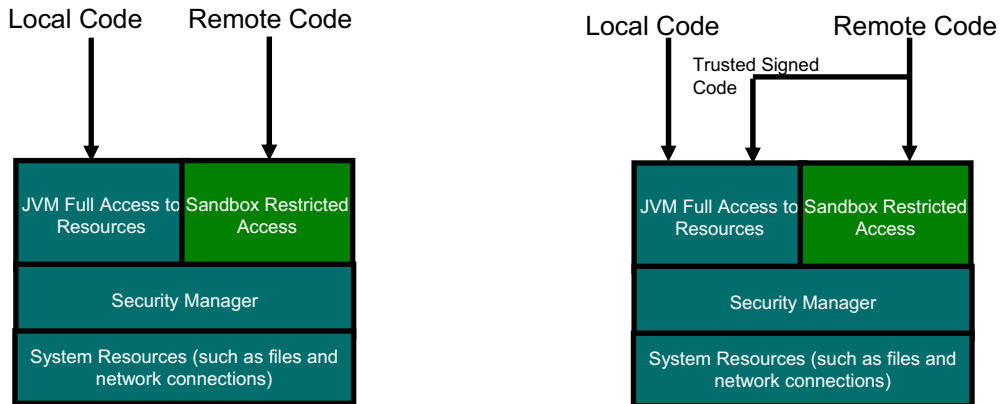
**Details —**

**Additional information —**

**Transition statement —** Next: Java security model

# Java security model

- Java 1.0 (sandbox model):
  - Downloaded code (untrusted) runs in a Sandbox (restricted environment)
  - Application code (local Java classes) have full access to resources (trusted and no protection)
- Java 1.1 (signed code):
  - Extends 1.0 Sandbox Model, introduce signed code, digitally signed remote code is treated like local code if public key used to verify the signature is trusted.



© Copyright IBM Corporation 2007

Figure 16-35. Java security model

WA3616.3

## Notes:

***Instructor notes:***

**Purpose —**

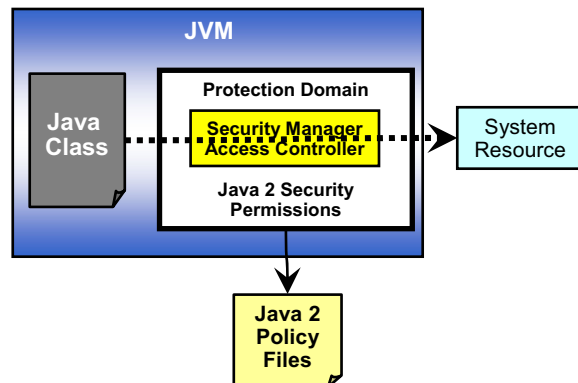
**Details —**

**Additional information —**

**Transition statement —** Next: Java 2 security overview

## Java 2 security overview

- Protects the system from the applications
- Provides an access control mechanism to manage the application's access to system level resources
  - File I/O, network connections (sockets), property files, and so forth
  - Policy-based
- Policies define a set of permissions available from various signers and code locations
  - Stored in policy files
- All Java code runs under a security policy
  - Grants access to certain resources
- Can be turned on or off independently of administrative security



- Java code needs access to certain system resources
- Java code needs to get the permission from Java 2 access control
- Access control looks at the Java 2 policy files to determine if the requesting Java code has the appropriate permission

© Copyright IBM Corporation 2007

Figure 16-36. Java 2 security overview

WA3616.3

### Notes:

Orthogonal to J2EE role based security. Java 2 security is about protecting system resources.

Policy based (controlled by several .policy files)

Fine-Grained Access Control to system resources

- File I/O
- Sockets
- Properties

Applications must be made ready or problems will occur

- Security **java.security.AccessControlExceptions** in the SystemOut.log, SystemError.log file)

***Instructor notes:***

**Purpose —**

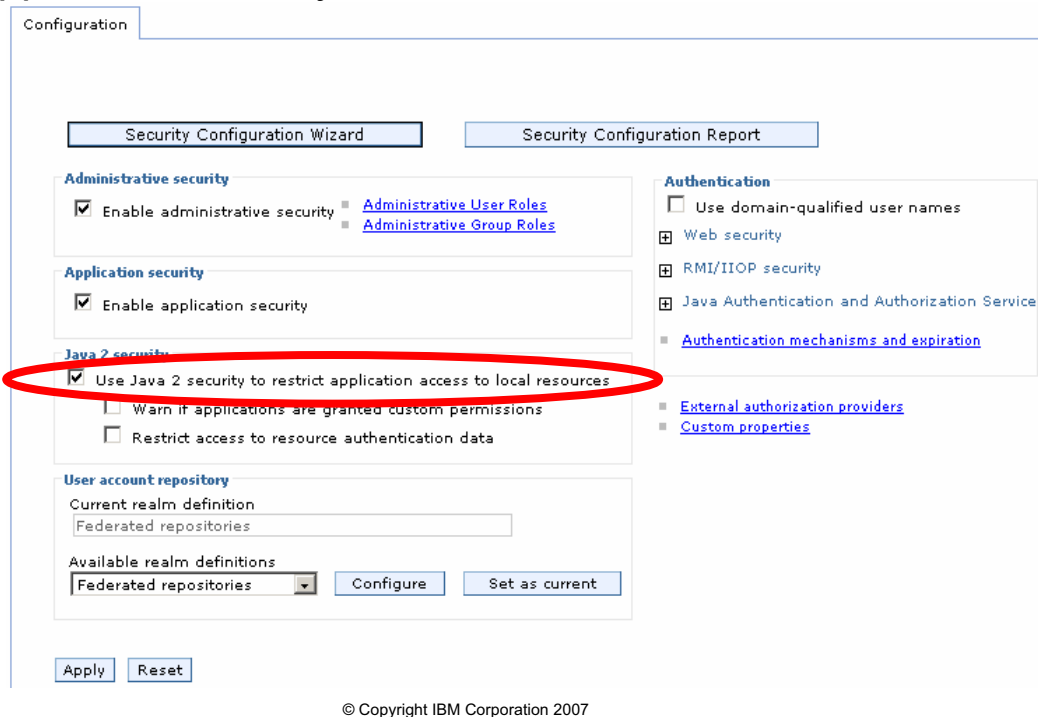
**Details —**

**Additional information —**

**Transition statement —** Next: Enabling Java 2 security

## Enabling Java 2 security

Can be enabled and disabled independently of administrative and application security



© Copyright IBM Corporation 2007

Figure 16-37. Enabling Java 2 security

WA3616.3

### Notes:

When Java 2 Security is enabled in WebSphere, the Security Manager component will by default throw an `java.security.AccessControl` exception when a permission violation occurs. This exception, if not handled, will often cause a runtime failure. This exception is also logged in `SystemOut.log`.

However, when the JVM `com.ibm.websphere.java2secman.norethrow` property is set and has a value of `true`, the Security Manager will not throw the `AccessControl` exception -- it will only be logged.

Note: This property is intended for a sandbox or debug environment only since it instructs the Security Manager not to throw the `AccessControl` exception. By not rethrowing the exception, Java 2 security is not actually enforced. This property should not be used in a production environment where a relaxed Java 2 security environment will weaken the very integrity Java 2 security is intended to produce.

The JVM parameter entered on the command line starts the server, usually in the script `startServer`. Enter as `-Dcom.ibm.websphere.java2secman.norethrow=true`

Look in the log for the line below to verify that the above is in place:

```
SecurityManag W SECJ0381I: Warning, the  
com.ibm.websphere.java2secman.norethrow property is true. The WebSphere  
Java 2 Security Manager is not rethrowing AccessControl exceptions. This debug  
setting should not be used in a production environment. See the InfoCenter for Java 2  
Security debugging features.
```

***Instructor notes:***

**Purpose —**

**Details —**

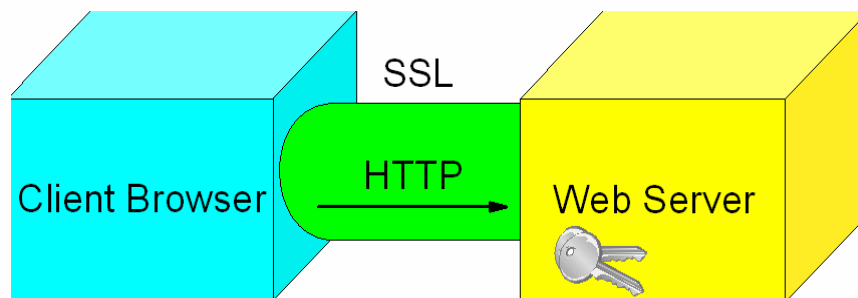
**Additional information —**

**Transition statement —** Next: What is SSL?



## What is SSL?

- SSL stands for Secure Sockets Layer
- SSL provides connection security through:
  - Communication privacy - the data on the connection is encrypted
  - Communication integrity - the protocol includes a built-in integrity check
  - Authentication - the client knows who the server is
- SSL creates a VPN, securing the data using a combination of symmetric and asymmetric encryption



© Copyright IBM Corporation 2007

Figure 16-38. What is SSL?

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

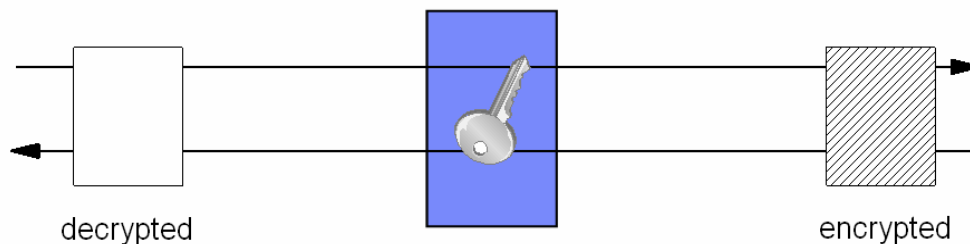
**Details —**

**Additional information —**

**Transition statement —** Next: Symmetric key encryption

## Symmetric key encryption

- Symmetric or secret key technology is a model in which two parties have a shared secret
- The same key is used for both encryption and decryption.



© Copyright IBM Corporation 2007

Figure 16-39. Symmetric key encryption

WA3616.3

### Notes:

Symmetric key encryption allows two parties to send each other secured messages, but required a shared secret. The trouble with this approach is that at some point that secret needs to be shared. In an e-Business application, this would be rather difficult.

***Instructor notes:***

**Purpose —**

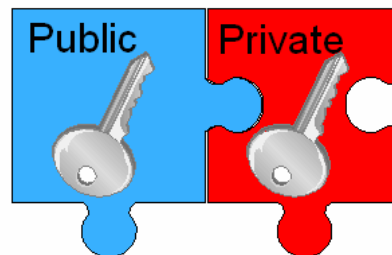
**Details —**

**Additional information —**

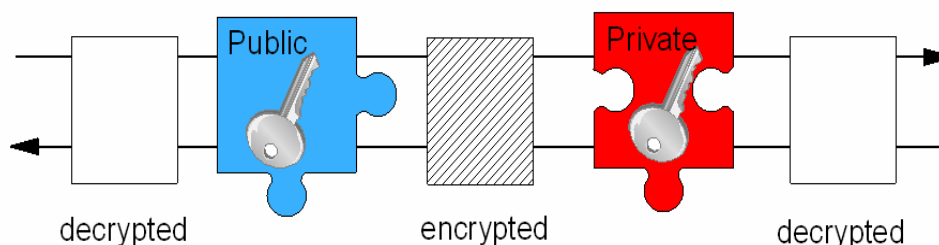
**Transition statement —** Next: Asymmetric key encryption

## Asymmetric key encryption

- Public key cryptography
  - Two keys that are cryptographically related
    - Public key (Can be shared with everyone)
    - Private key (Must never be shared; possession is proof)



- Keys are asymmetric
  - Given message is encrypted with one key and decrypted with the other



© Copyright IBM Corporation 2007

Figure 16-40. Asymmetric key encryption

WA3616.3

### Notes:

If a server has a public-private set, it can send out its public key (through a signing certificate – also known simply as a certificate) to client machines. Those client machines can then use that public key to encrypt messages destined for the server which then only the server can decrypt. Unlike symmetric key encryption, this process does not require the client and server to have a shared secret.

Since the client can validate the server's certificate, there is one way authentication. But the server has no way (at this point) to authenticate the client. Nor can the server send the client secured messages.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: How does SSL work?

## How does SSL work?

- SSL uses a combination of asymmetric and symmetric encryption to create a session between the client and server.
  - Asymmetric encryption is used to negotiate a session key (shared secret)
    - Asymmetric encryption is slow but does not require a shared secret
  - Symmetric encryption is used to transfer data between the client and server
    - Symmetric encryption is fast but requires a shared secret

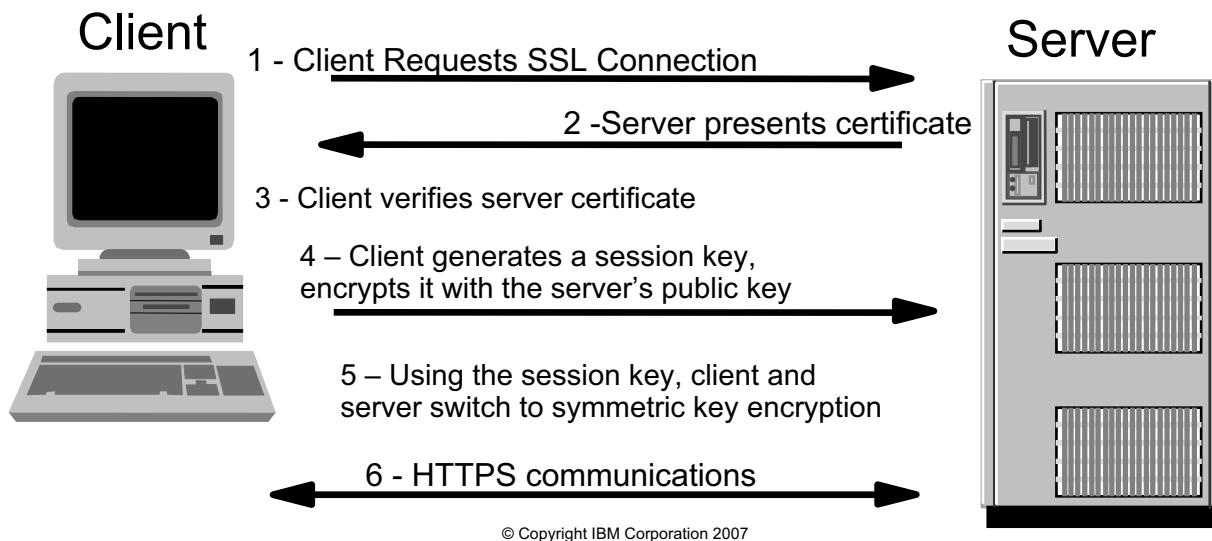


Figure 16-41. How does SSL work?

WA3616.3

### Notes:

Because the client chooses its own session key, nobody else knows it. It can securely send that session key to the server using the server's public key. Now, nobody but the client and server know the session key. The session key is then used as a “shared secret” to switch to the much more efficient symmetric key encryption.

A certificate (or signing certificate) contains information about the server, including the server's public key, and is digitally signed by the certificate authority.

***Instructor notes:***

**Purpose —**

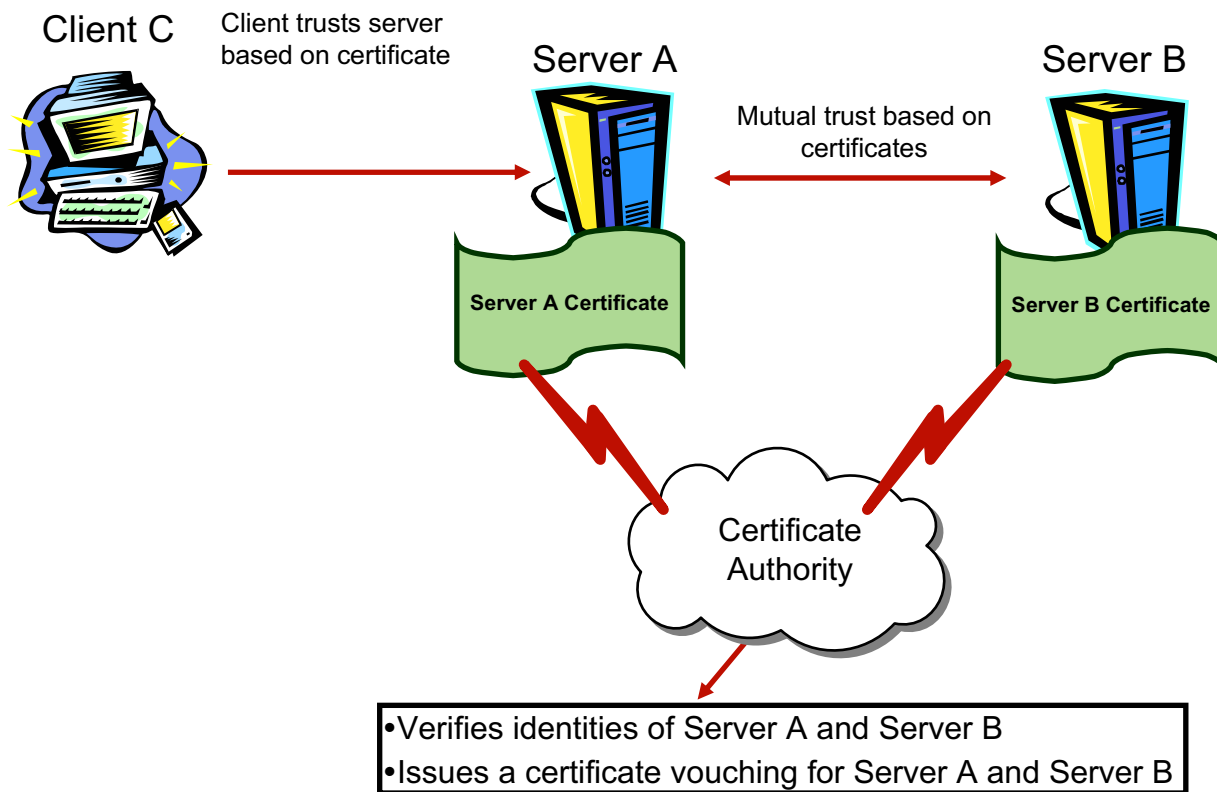
**Details —**

**Additional information —**

**Transition statement —** Next: Certificates and certificate authority (CA)



## Certificates and certificate authority (CA)



© Copyright IBM Corporation 2007

Figure 16-42. Certificates and certificate authority (CA)

WA3616.3

### Notes:

If a client is going to trust a certificate from a server, the client must be able to validate that certificate. This is possible because certificates are generated and signed by a Certificate Authority with the certificate authority's private key. That means that the browser can verify the digital signature using the CA's public key. The public keys from the standard CAs are built into browsers. If the browser does not have a copy of the CA's public key, the user will be prompted.

### Public Key Infrastructure (PKI)

- Based on public/private key cryptography
- The whole infrastructure that makes public-key security work
  - Certificate Authority (CA)
  - Registration Authority (RA)
  - PKI enabled applications
  - Directory (optional)

***Instructor notes:***

**Purpose —**

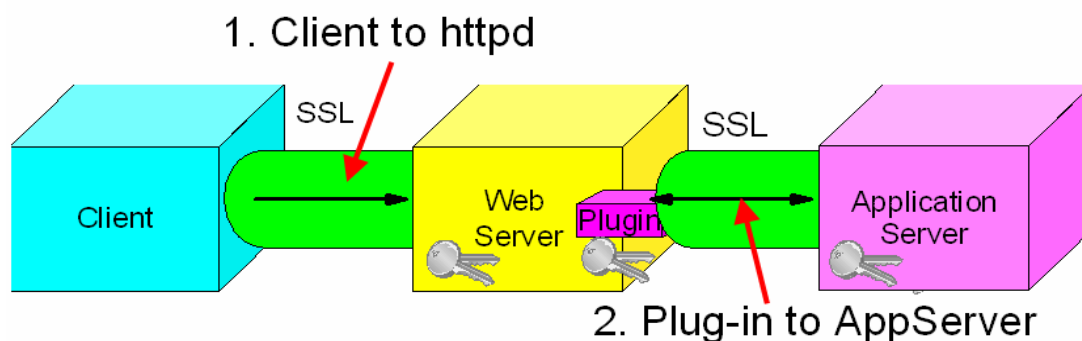
**Details —**

**Additional information —**

**Transition statement —** Next: SSL within WebSphere Application Server

## SSL within WebSphere Application Server

- SSL can be used to secure network traffic for a number of links
  - From the client to the Web server
  - From the plug-in to the application server
  - Other network links can also be secured (LDAP and others)
- **ikeyman** can be used to create and manage the necessary keys and keyrings
  - Keyrings contain digital certificates needed for SSL to establish secure communication between points A and B.



© Copyright IBM Corporation 2007

Figure 16-43. SSL within WebSphere Application Server

WA3616.3

### Notes:

In order to configure SSL for the Web server, a Keyring and Stash file need to be created. The Web server then needs to have SSL configured and point at the Keyring and stash file.

***Instructor notes:***

**Purpose** —

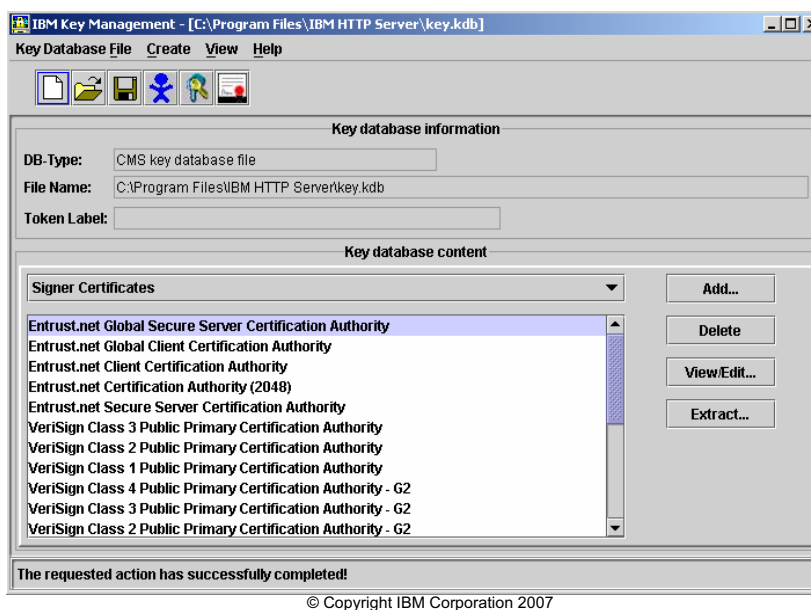
**Details** —

**Additional information** —

**Transition statement** — Next: Creating key rings and certificates

## Creating key rings and certificates

- The tool for manipulating the key rings is called **iKeyman**
  - Creates keyrings
  - Requests and imports CA certificates
  - Creates self-signed certificates
- **Beware:** There are multiple versions of iKeyman shipping with different products. They are not always compatible.



© Copyright IBM Corporation 2007

Figure 16-44. Creating key rings and certificates

WA3616.3

### Notes:

There are copies of ikeyman that come with IBM HTTP Server, the plug-in and with the Application Server. The versions that come with IBM HTTP Server and the plug-in are compatible, but the version that comes with the Application server is different, even though the interfaces look identical.

***Instructor notes:***

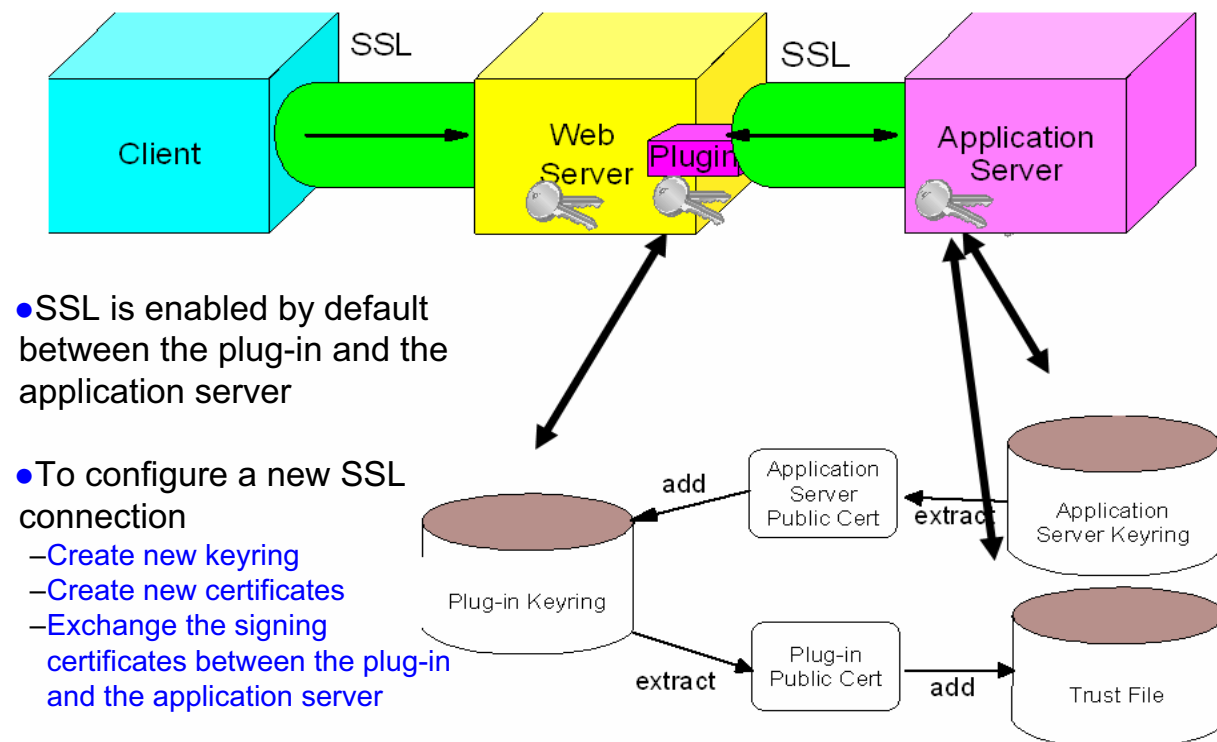
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: SSL between plug-in and the application server

## SSL between plug-in and the application server



© Copyright IBM Corporation 2007

Figure 16-45. SSL between plug-in and the application server

WA3616.3

### Notes:

The dummy keyring is not secure since it is readily available to anyone who has a copy of WebSphere.

The SSL implementation used by the WebSphere Application Server stores personal certificates in an SSL *key file* and signer certificates in a *trust file*. A key file contains a collection of certificates, each one of which may be presented during an SSL connection initiation in order to prove identity. A trust file contains a collection of certificates that are considered trustworthy and against which the presented certificate will be matched during an SSL connection initiation in order to assure identity.

Self-signed keys can actually be more secure than CA certificates for internal communications like those between the plug-in and the application server.

To configure SSL between the plug-in and the application server with a new keyring:

- Create a new keyring for the plug-in
- Create a self-signed key in the plug-in's keyring
- Export the plug-in's signing certificate (public cert)

- Create a new keyring for the application server
- Create a self-signed key for the application server
- Export the application server's signing certificate (public cert)
- Create a trust file for the application server
- Import the application server's signing certificate into the plug-in's keyring
- Import the plug-in's signing certificate into the application server's trust file
- Configure the plugin-cfg.xml to point at the new plug-in keyring and stash file
- Configure the application server to use the new application server keyring and trust file



***Instructor notes:***

**Purpose —**

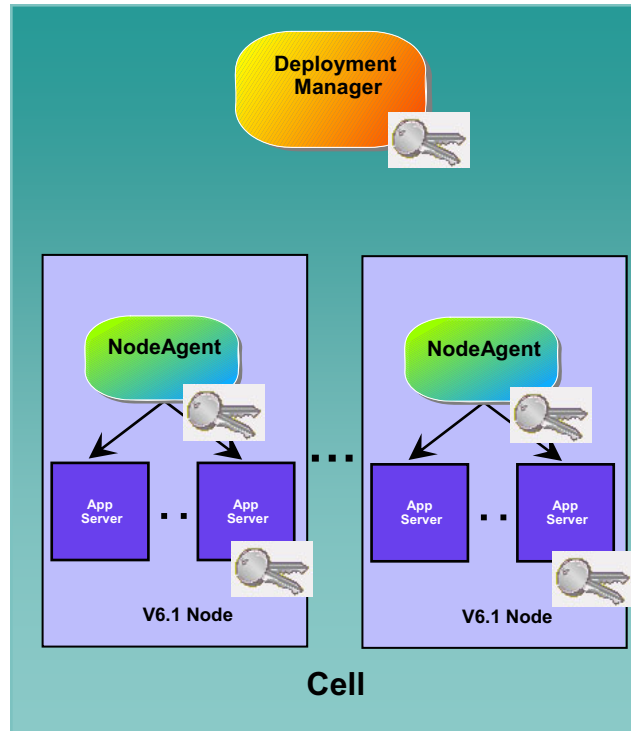
**Details —**

**Additional information —**

**Transition statement —** Next: Cell key rings and truststores

## Cell key rings and truststores

- Each managed process will have a self-signed certificate
  - The default lifetime of the certificates is one year
  - The default is for the certificates to be self managed
  - As the certificates approach their expiration (60 days prior), a warning is generated and the certificate is automatically replaced
- All signing certificates for the cell are put into a cell wide truststore
  - By sharing a single trust store, the any member of the cell can securely communicate with any other member of the cell



© Copyright IBM Corporation 2007

Figure 16-46. Cell key rings and truststores

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

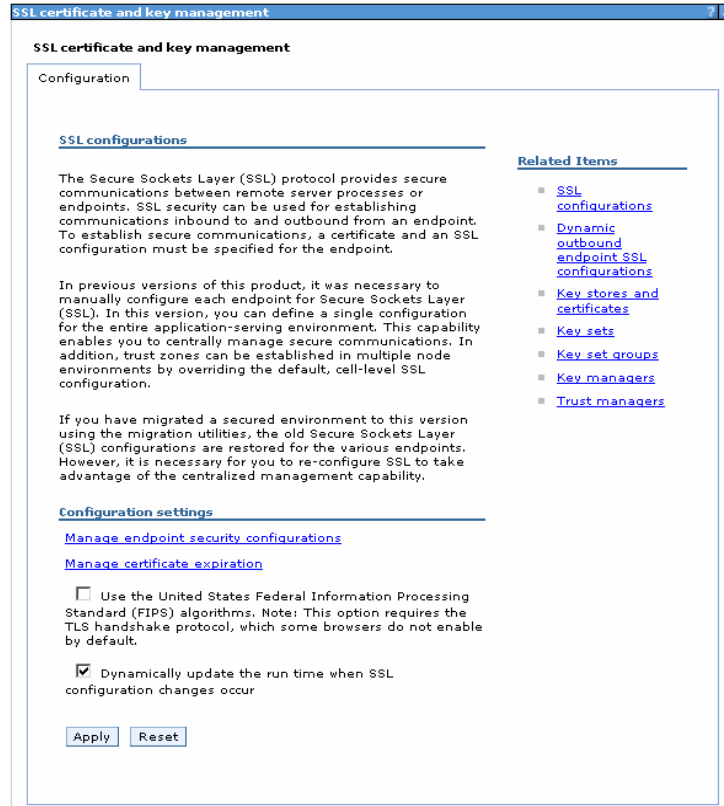
**Details —**

**Additional information —**

**Transition statement —** Next: Managing WebSphere key rings

# Managing WebSphere key rings

Key rings and certificates for the cell can be managed directly from the console



© Copyright IBM Corporation 2007

Figure 16-47. Managing WebSphere key rings

WA3616.3

## Notes:

***Instructor notes:***

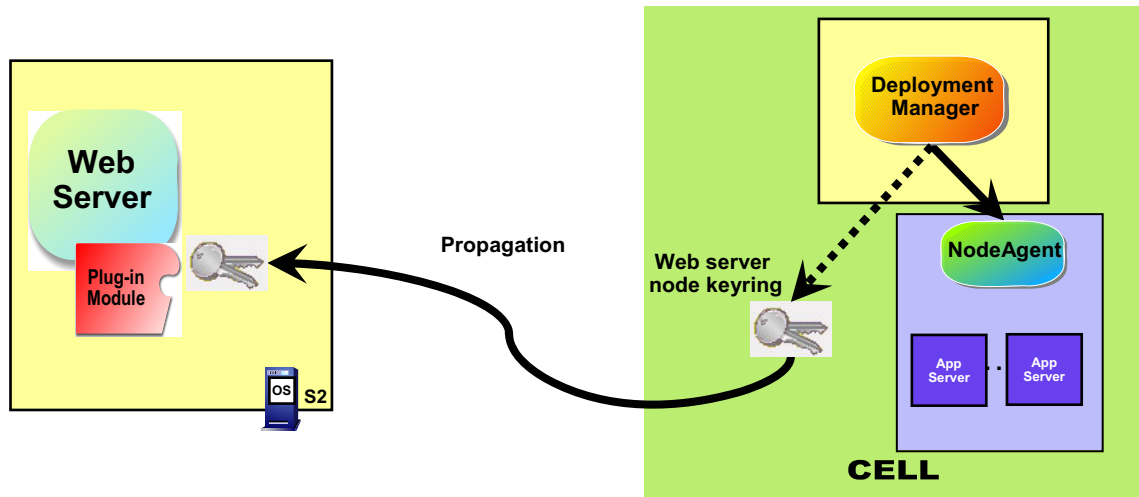
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Plug-in key rings

## Plug-in key rings



- A self-signed certificate is generated by WebSphere for each defined Web server node
  - A customer generated certificate can also be used
- The current certificate needs to be propagated to the Web server
  - During the plug-in install a temporary certificate/keyring is made available for the Web Server
  - This temporary keyring needs to be replaced in order to be able to communicate securely with the cell
- **IMPORTANT:** The new key ring includes signing certificates for every node in the cell
  - If any nodes are subsequently added to your cell or any certificates for the nodes are updated, the Web server key rings will need to be updated

© Copyright IBM Corporation 2007

Figure 16-48. Plug-in key rings

WA3616.3

### Notes:

***Instructor notes:***

**Purpose** —

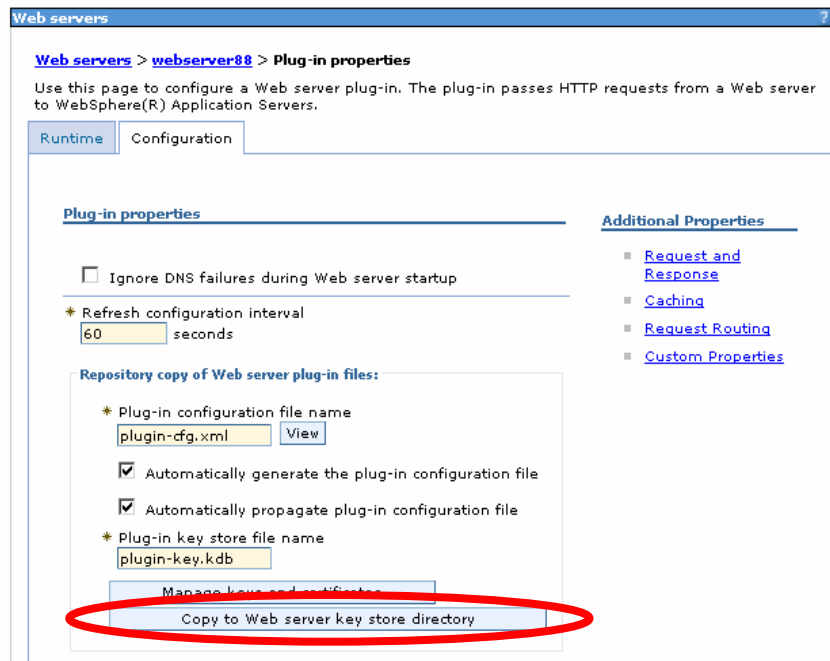
**Details** —

**Additional information** —

**Transition statement** — Next: IBM HTTP Server key ring propagation

# IBM HTTP Server key ring propagation

- The initial key ring for IBM HTTP Server servers can be remotely propagated
  - This does up automatically update the keyring in the future
  - The key ring is not updated after initial creation



© Copyright IBM Corporation 2007

Figure 16-49. IBM HTTP Server key ring propagation

WA3616.3

## Notes:



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Checkpoint questions

## Checkpoint questions

---

1. Which type of security restricts access to the application?
  - A. Administrative security
  - B. J2EE security
  - C. Java 2 security
  - D. File system security
  
2. Which type of security restricts access to the operating system?
  - A. Administrative security
  - B. J2EE security
  - C. Java 2 security
  - D. File system security
  
3. Which type of security restricts access to the console?
  - A. Administrative security
  - B. J2EE security
  - C. Java 2 security
  - D. File system security

© Copyright IBM Corporation 2007

---

Figure 16-50. Checkpoint questions

WA3616.3

### **Notes:**

Write down your answers here:

- 1.
- 2.
- 3.

**Instructor notes:**

Purpose —

Details —

---

## Checkpoint solutions

---

1. Which type of security restricts access to the application?  
B. J2EE security
2. Which type of security restricts access to the operating system?  
C. Java 2 security
3. Which type of security restricts access to the console?  
A. Administrative security

© Copyright IBM Corporation 2007

**Additional information** —**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe basic security concepts
- Describe WebSphere security architecture and concepts
- Configure security settings for console access
- Configure security settings for an application
- Describe SSL and how to configure it

© Copyright IBM Corporation 2007

Figure 16-51. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Lab exercise

## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 16-52. Lab exercise

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**





---

# Unit 17. Performance monitoring and tuning tools

## Estimated time

01:00

## What this unit is about

This unit describes the Performance Monitoring Infrastructure (PMI) and the use of Tivoli's Integrated Performance Viewer.

## What you should be able to do

After completing this unit, you should be able to:

- Describe the features of Integrated Performance Viewer
- Describe how performance data is collected in Integrated Performance Viewer
- Set data collection levels on started and stopped resources
- Describe the performance data collected by Integrated Performance Viewer
- Use the logging function for viewing and storing data
- Set up and view a graphical representation of collected data

## How you will check your progress

Accountability:

- Checkpoint
- Machine exercises

## References

SG24-7304-00 WebSphere Application Server V6.1 System Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

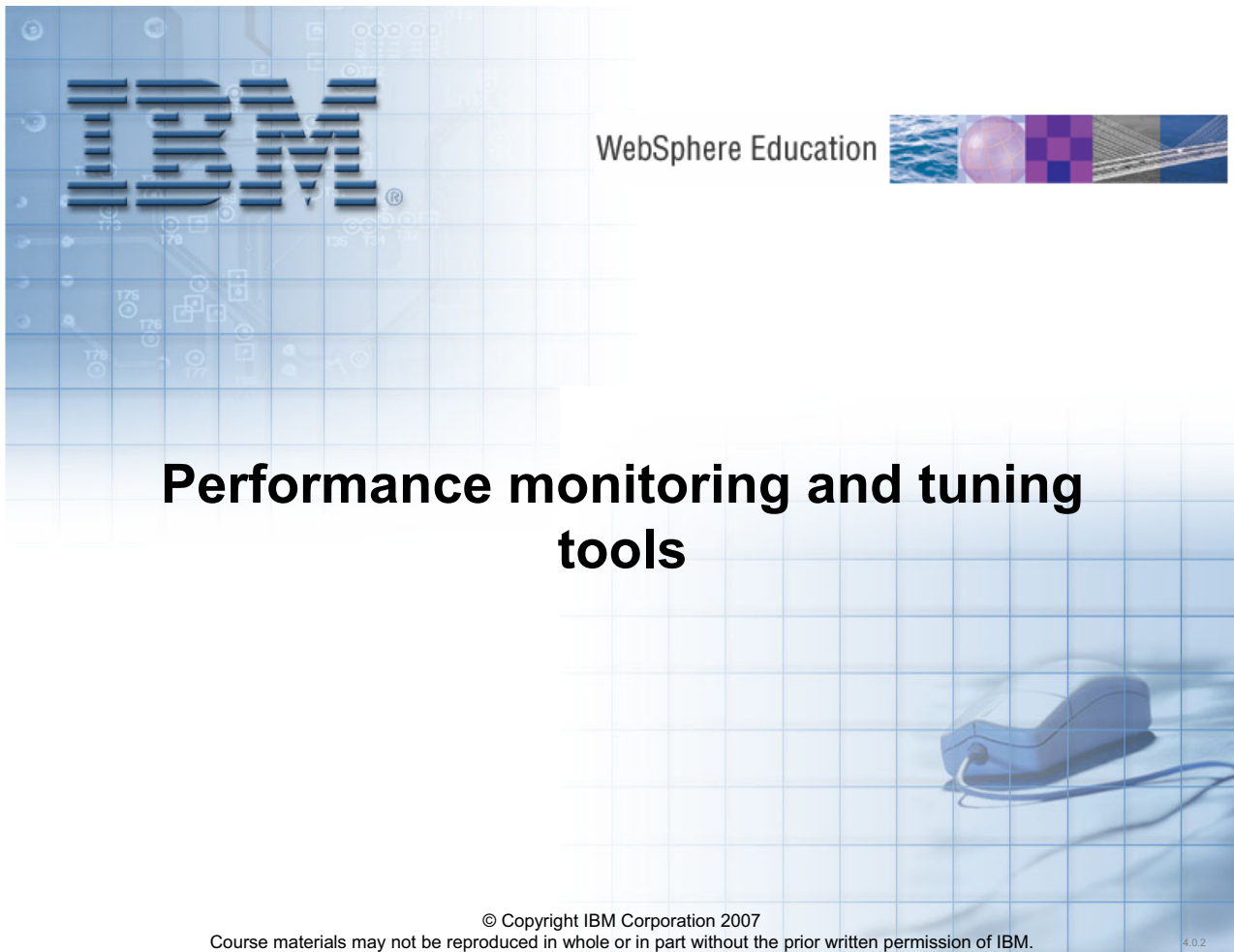


Figure 17-1. Performance monitoring and tuning tools

WA3616.3

## **Notes:**

### **What this unit is about**

This unit describes the performance monitoring and tuning tools available in WebSphere.

### **What you should be able to do**

After completing this unit, you should be able to:

- Describe the different performance monitoring technologies in WebSphere
- Identify the available performance monitoring and tuning tools
- Use Tivoli Performance Viewer to monitor overall system health
- Utilize Request Metrics to monitor application flow
- Use the Performance Advisors to tune WebSphere

### **How you will check your progress**

Accountability:

- Machine exercise

## References

- IBM WebSphere Application Server, Version 6.1 Information Center - *Monitoring*  
[http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/welcome\\_nd.html](http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/welcome_nd.html)Web page name
- IBM WebSphere Application Server, Version 6.1 Information Center – *Tuning Performance*  
[http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/welcome\\_nd.html](http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/welcome_nd.html)Web page name

***Instructor notes:***

**Purpose —**

**Details — Estimated time**

01:00

**Additional information —**

**Transition statement —** Next: Unit objectives

## Unit objectives

---

After completing this unit, you should be able to:

- Describe the features of Integrated Performance Viewer
- Describe how performance data is collected in Integrated Performance Viewer
- Set data collection levels on started and stopped resources
- Describe the performance data collected by Integrated Performance Viewer
- Use the logging function for viewing and storing data
- Set up and view a graphical representation of collected data

© Copyright IBM Corporation 2007

Figure 17-2. Unit objectives

WA3616.3

### **Notes:**

***Instructor notes:***

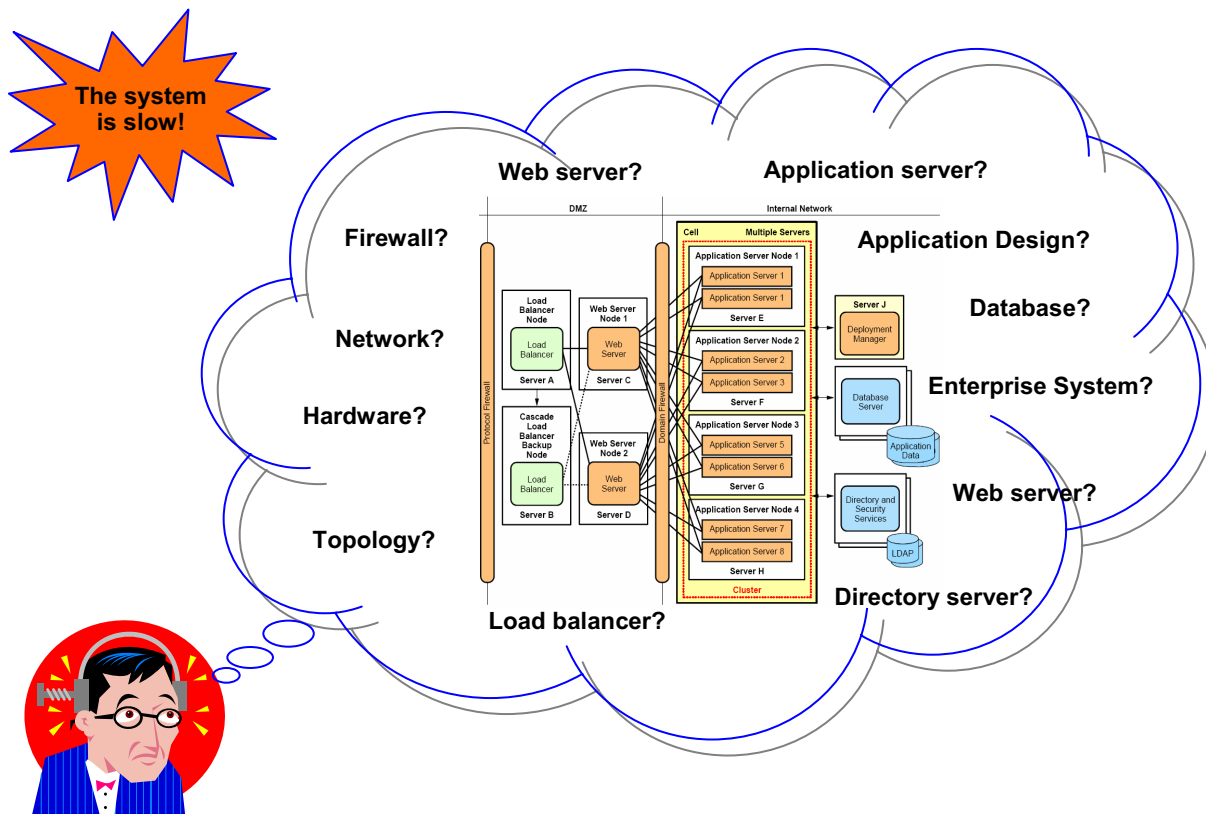
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: What is the cause of the performance problem?

# What is the cause of the performance problem?



© Copyright IBM Corporation 2007

Figure 17-3. What is the cause of the performance problem?

WA3616.3

## Notes:

Problems can be (and have been) found almost anywhere within an application. The problem can be related to the network, hardware or back-end system. It can be caused by actual product bugs, or, quite often, application design issues.

With today's applications becoming increasingly complex, it is getting harder and harder to track down these problems. Typical e-business applications now have one or more of the following attributes:

- Multiple logical or physical tiers
- Mixture of operating system platforms
- Mixture of hardware architectures
- Fuzzy application boundaries

Today's software is capable of producing massive quantities of numbers including CPU usage statistics, memory usage statistics, queuing statistics, transaction rate statistics and buffer pool statistics. The problem is that the large quantity of statistics can actually be

overwhelming and may hamper, rather than help, a person's ability to understand and analyze performance data.

You need effective tools to aid in preventing performance problems, and diagnosing and fixing them when they occur.



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: The need for performance monitoring and tuning

## The need for performance monitoring and tuning

---

- Poor performance results in:
  - Escalating application support costs
  - Loss of customer confidence
  - Loss of revenue
- Performance problems can be anywhere in the application serving environment
  - Monitoring ensures that applications are running as expected and, if not, determine why and where the problem lies
- WebSphere Application Server can perform well with default settings but:
  - Improving performance and reducing end-user response times requires additional tuning

© Copyright IBM Corporation 2007

Figure 17-4. The need for performance monitoring and tuning

WA3616.3

### **Notes:**

The goal of performance monitoring is to collect runtime statistics on your application and its environment in order to quantify their performance behavior. It allows you to determine whether or not your application meets its performance objectives and helps to identify any performance bottlenecks.

Your application and its runtime environment should also be tuned optimally. This process entails conducting many iterations of a Monitor – Tune – Test cycle. In short, monitoring, performance testing and tuning are essential tasks for ensuring a well-performing application serving environment.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere performance tools

## WebSphere performance tools

---

- WebSphere provides integrated tools to monitor and tune system and application performance:
  - **Tivoli Performance Viewer (TPV)**
    - Enables administrators to monitor the overall health of WebSphere Application Server
    - Accessed from within the administrative console
  - **Request Metrics (tool)**
    - Enables you to track individual transactions, recording the processing time in each of the major WebSphere Application Server components
    - Output viewed in standard logs or using an Application Response Measurement (ARM) based tool
  - **Performance advisors**
    - Analyze collected performance data and provide configuration recommendations to improve the application server performance
    - Output viewed in TPV or in administrative console runtime messages
  - **Performance servlet**
    - Provides simple retrieval of performance data in XML format
    - Accessed through a browser

© Copyright IBM Corporation 2007

Figure 17-5. WebSphere performance tools

WA3616.3

### **Notes:**

These tools will be discussed in greater detail in the subsequent topics.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere performance technologies

## WebSphere performance technologies

---

- Underlying performance data collection supported by the following technologies:
  - **Performance Monitoring Infrastructure (PMI)**
    - Core performance data collection technology for WebSphere Application Server
    - Supports the Java 2 Platform, Enterprise Edition (J2EE) Management Reference Implementation (JSR-077)
  - **Request Metrics**
    - Technology to trace each individual transaction and record its response time at different stages as it flows through the application server
    - Supports externalization of data to standard log files and/or an Application Response Measurement (ARM) agent

© Copyright IBM Corporation 2007

Figure 17-6. WebSphere performance technologies

WA3616.3

### **Notes:**

These technologies will be discussed in greater detail in the subsequent topics.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Performance tool usage and data collection technology

## Performance tool usage and data collection technology

	Tivoli Performance Viewer	Performance servlet	Request Metrics tool	Performance advisors
<b>Intended usage</b>	<ul style="list-style-type: none"> <li>• Monitor overall system health</li> </ul>	<ul style="list-style-type: none"> <li>• Simple retrieval of performance data in XML format</li> </ul>	<ul style="list-style-type: none"> <li>• Monitor application flow</li> </ul>	<ul style="list-style-type: none"> <li>• Performance tuning</li> </ul>
<b>Performance data source</b>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Request Metrics</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>

© Copyright IBM Corporation 2007

Figure 17-7. Performance tool usage and data collection technology

WA3616.3

### Notes:

This matrix allows you to differentiate the WebSphere performance tools by usage and data collection technology. It shows that WebSphere provides the following tools and technologies to support performance monitoring and tuning:

- Data collection: PMI and Request Metrics collect performance data for externalization.
- Data viewer:
  - The Performance servlet provides a simple means to view the raw performance data collected by the PMI and extract it into XML.
  - The Tivoli Performance Viewer that is integrated with the administrative console allows the administrator to view the collected PMI data.
- Advisors: the Performance Advisors provided in the administrative console or Tivoli Performance Viewer provide configuration advice based on the collected PMI data and WebSphere's current configuration.



***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Use this matrix as a guide for discussing WebSphere performance tools and technologies in the subsequent topics.

## Performance Monitoring Infrastructure (PMI) data collection technology

	Tivoli Performance Viewer	Performance servlet	Request Metrics tool	Performance advisors
<b>Intended usage</b>	<ul style="list-style-type: none"> <li>• Monitor overall system health</li> </ul>	<ul style="list-style-type: none"> <li>• Simple retrieval of performance data in XML format</li> </ul>	<ul style="list-style-type: none"> <li>• Monitor application flow</li> </ul>	<ul style="list-style-type: none"> <li>• Performance tuning</li> </ul>
<b>Performance data source</b>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Request Metrics</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>

© Copyright IBM Corporation 2007

Figure 17-8. Performance Monitoring Infrastructure (PMI) data collection technology

WA3616.3

### Notes:

The Performance Monitoring Infrastructure (PMI) is WebSphere's central data collection technology and enables the following tools to externalize performance data:

- Performance servlet
- Tivoli Performance Viewer
- Performance advisors

***Instructor notes:***

**Purpose —**

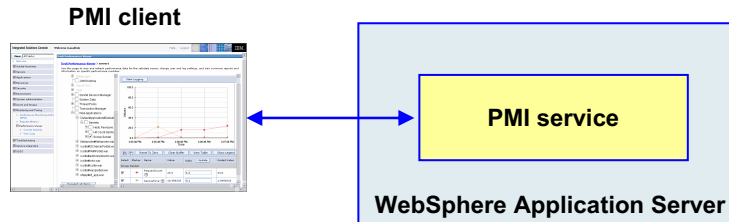
**Details —**

**Additional information —**

**Transition statement —** Next: Performance Monitoring Infrastructure overview

## Performance Monitoring Infrastructure overview

- WebSphere service that collects performance data inside the server and provides an interface to externalize it to clients



- Performance data collected on:
  - [Application resources](#)
    - Example: Custom PMI, EJBs, servlets/JSPs, Web services
  - [WebSphere runtime resources](#)
    - Example: JVM memory, thread pools, database connection pools
  - [System resources](#)
    - Example: CPU usage, total free memory
  - [Detailed list of all types and components can be found in Information Center](#)

© Copyright IBM Corporation 2007

Figure 17-9. Performance Monitoring Infrastructure overview

WA3616.3

### Notes:

The Performance Monitoring Infrastructure (PMI) collects performance data in memory and provides an interface that allows external clients to retrieve it for display or further processing.

The performance data that is generated falls into three general categories:

1. *Customer application performance data.* This includes information about your installed enterprise beans, servlets, Web services, or even custom defined PMI data. The custom PMI data will be discussed further in an upcoming slide.
2. *WebSphere runtime resources.* This includes information from WebSphere's JVM, active thread pools and database connection pools and most of the other resources that you have defined within WebSphere.
3. *Performance data collected from the system where WebSphere is currently running.* This includes performance data on the system's CPU usage or the total amount of free memory on the system. To see a complete list of all of the PMI data that is available from a WebSphere server, visit the WebSphere Information Center.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: PMI data modules and counters

# PMI data modules and counters

### Servlet session manager

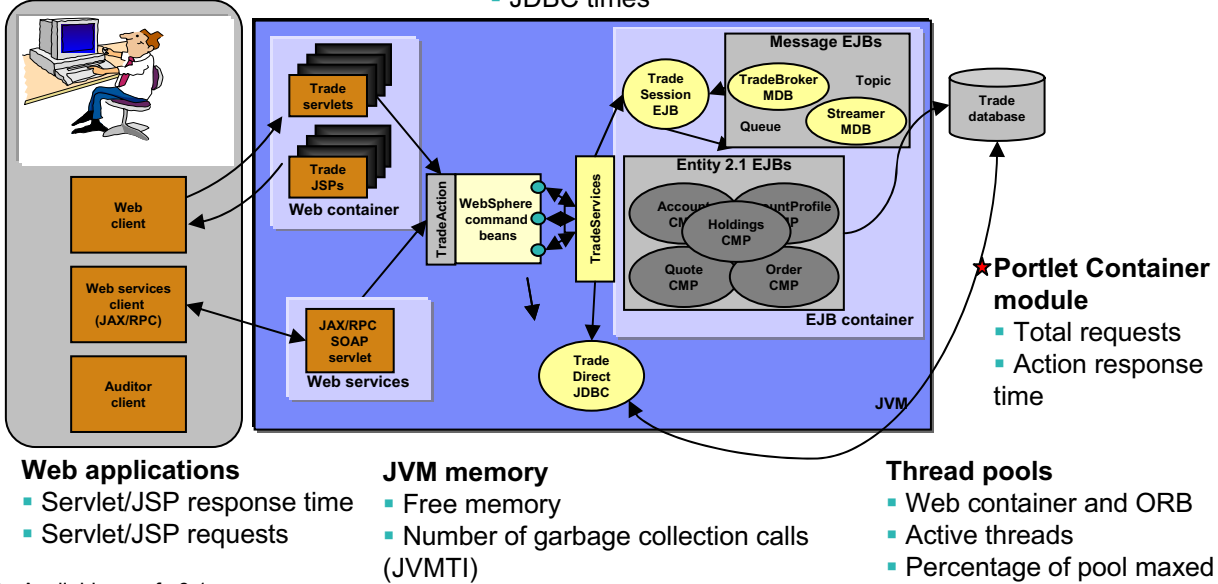
- Avg number of active HTTP sessions
- Session size

### JDBC connection pools

- Percentage of pool in use
- Average wait time
- JDBC times

### EJB module

- EJB response time
- EJB requests



### Web applications

- Servlet/JSP response time
- Servlet/JSP requests

### JVM memory

- Free memory
- Number of garbage collection calls (JVMTI)

### Thread pools

- Web container and ORB
- Active threads
- Percentage of pool maxed

★ Available as of v6.1

### And many components more...

- For example: System data (memory), J2C connections, Cache, JMS and SIB, Web services

© Copyright IBM Corporation 2007

Figure 17-10. PMI data modules and counters

WA3616.3

## Notes:

The graphic above gives only a partial view into the amount of PMI data that is collected by WebSphere. Modules and counters are continually being added to the PMI data set.

Version 6.1 provides new counters for per-process CPU data and PMI data at the Struts/Tiles level for greater granularity. It also includes new transactional data for J2C and naming, and a new Portlet Container module. Note also that as of version 6.1, the Java Virtual Machine Profiling Interface (JVMPPI) is deprecated and replaced with the Java Virtual Machine Toll Interface (JVMTI) which is new for the v1.5 JVM.

For a complete list of available PMI data, visit the WebSphere Information Center. The Information Center also provides information about the release in which a PMI metric was added.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Data collection settings

## Data collection settings

---

- PMI uses *statistics sets* to specify the type and amount of performance data to collect:
  - **None**
    - All statistics are disabled
  - **Basic**
    - Statistics specified in J2EE 1.4, as well as top statistics like CPU usage and live HTTP sessions, are enabled
    - This set is enabled by default and provides basic performance data about runtime and application components (overhead of up to 2%)
  - **Extended**
    - Basic set, plus key statistics from various WebSphere Application Server components like WLM and dynamic caching, are enabled
    - This set provides detailed performance data about various runtime and application components (overhead of up to 3%)
  - **All**
    - All statistics are enabled (overhead of up to 6%)
  - **Custom**
    - Statistics are enabled or disabled individually

© Copyright IBM Corporation 2007

Figure 17-11. Data collection settings

WA3616.3

### Notes:

The *Custom* set allows you to enable or disable individual modules or counters.

A performance impact is incurred on the application server when PMI data is enabled. The extent of the performance impact is determined by the amount and type of PMI counters that are collected. Counters that add up the amount of times a method is called will cause less of a performance impact than a counter that averages the amount of time spent executing a method call.

It is important to note that the PMI service is enabled by default and set to the *Basic* setting. This is a change from previous versions of WebSphere.

When collecting additional PMI data from WebSphere's JVM using the Java Virtual Machine Tool Interface (JVMTI) or Java Virtual Machine Profiler Interface (JVMPPI – deprecated as of version 6.1), you can expect additional overhead over the value stated above:

- Collecting just GC and thread data adds up to 2%
- Collecting all JVMTI data adds 20-25%



Also note of collecting data on HTTP Session size is a heavyweight function, since all session data must be serialized to calculate size.

***Instructor notes:***

**Purpose —**

**Details —**

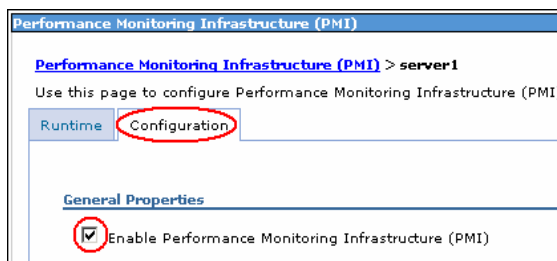
**Additional information —**

**Transition statement —** Next: Enabling PMI

## Enabling PMI

---

- PMI is enabled by default
- If necessary, use the administrative console to enable PMI:
  - Select **Monitoring and Tuning -> Performance Monitoring Infrastructure (PMI)**
  - Select the server to be monitored
  - On the **Configuration** tab, select the **Enable Performance Monitoring Infrastructure (PMI)** check box
  - Restart the server



© Copyright IBM Corporation 2007

Figure 17-12. Enabling PMI

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

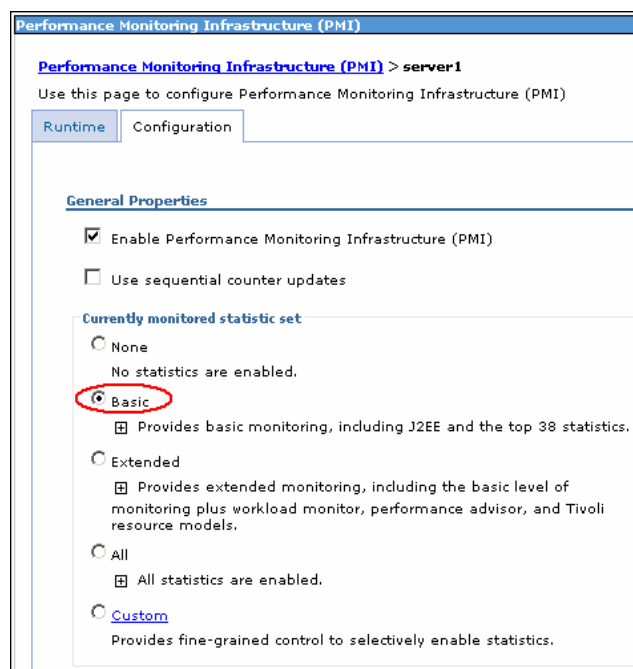
**Details —**

**Additional information —**

**Transition statement —** Next: Specifying the data collection setting

## Specifying the data collection setting

- Default statistics set is **Basic**
- In the administrative console:
  - Select **Monitoring and Tuning -> Performance Monitoring Infrastructure (PMI)**
  - Select the server to be monitored
  - Select the **Configuration** tab or **Runtime** tab
  - Select the radio button for the desired statistic set (None, Basic, Extended, All or Custom)
  - If **Configuration** tab was selected, restart the server
- PMI must be enabled and the collection level set in order to view data using Tivoli Performance Viewer or the Performance servlet



© Copyright IBM Corporation 2007

Figure 17-13. Specifying the data collection setting

WA3616.3

### Notes:

Two tabs are available when configuring PMI data collection:

- Configuration tab: Settings apply only after the server is restarted.
- Runtime tab: Settings apply immediately.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Monitoring system health with Tivoli Performance Viewer

## Monitoring system health with Tivoli Performance Viewer

	Tivoli Performance Viewer	Performance servlet	Request Metrics tool	Performance advisors
<b>Intended usage</b>	<ul style="list-style-type: none"> <li>• Monitor overall system health</li> </ul>	<ul style="list-style-type: none"> <li>• Simple retrieval of performance data in XML format</li> </ul>	<ul style="list-style-type: none"> <li>• Monitor application flow</li> </ul>	<ul style="list-style-type: none"> <li>• Performance tuning</li> </ul>
<b>Performance data source</b>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Request Metrics</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>

© Copyright IBM Corporation 2007

Figure 17-14. Monitoring system health with Tivoli Performance Viewer

WA3616.3

### Notes:

Tivoli Performance Viewer allows you to monitor overall system health by externalizing performance data collected by the Performance Monitoring Infrastructure.

***Instructor notes:***

**Purpose —**

**Details —**

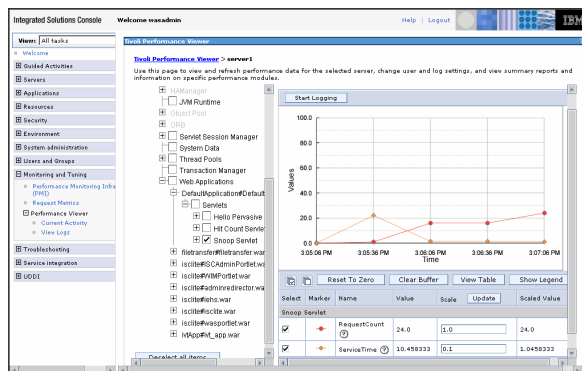
**Additional information —**

**Transition statement —** Next: Tivoli Performance Viewer overview



# Tivoli Performance Viewer overview

- WebSphere built-in monitoring tool used to view PMI data
  - [Integrated into WebSphere Administrative console](#)



- Displays PMI data collected from local and remote application servers
  - [Summary reports show key areas of contention](#)
  - [Graphical or tabular views show raw PMI data](#)
- Allows you to optionally save collected PMI data to logs
  - [Save performance data for later analysis or problem determination](#)
- Provides configuration advice via Advisor section
  - [Tuning advice formulated from gathered PMI and configuration data](#)

© Copyright IBM Corporation 2007

Figure 17-15. Tivoli Performance Viewer overview

WA3616.3

## Notes:

Tivoli Performance Viewer has had several different names throughout the life of the WebSphere product. In WebSphere Version 3.5 and Version 4, it was known as Resource Analyzer. In WebSphere v5, it was rebranded as Tivoli Performance Viewer.

Note that:

- Resource Analyzer only works with WebSphere Version 3.5 or Version 4.
- Tivoli Performance Viewer only works with WebSphere 5 and higher.

Tivoli Performance Viewer is capable of connecting directly to a base application server or to the Deployment Manager in a Network Deployment configuration, and is accessed through the Administrative Console.

By default, log files are stored in the `<WAS_ROOT>/profiles/<profile_name>/logs/tpv` directory on the node on which the server is running. To save space, TPV automatically compresses the log into a .zip file.

The Performance Advisors will be discussed at length later in the unit.

***Instructor notes:***

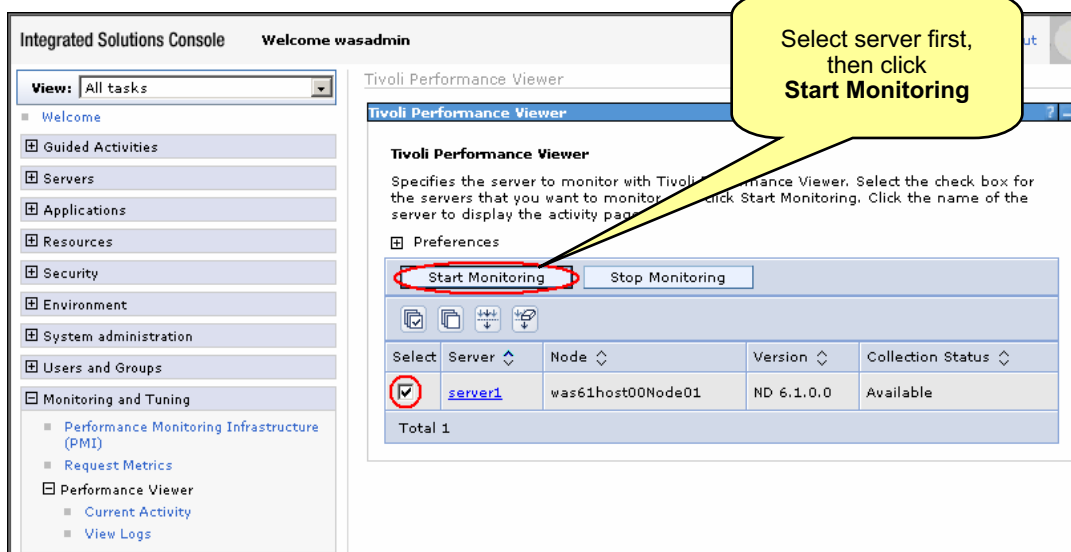
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: TPV monitoring: First step

## TPV monitoring: First step



- After enabling PMI, you need to instruct Tivoli Performance Viewer to start monitoring a selected server:
  - Enabling PMI causes server to collect performance data
  - Start Monitoring instructs to TPV retrieve the data for display

© Copyright IBM Corporation 2007

Figure 17-16. TPV monitoring: First step

WA3616.3

### Notes:

Scalable Vector Graphics (SVG) is a new graphics file format and Web development language based on XML. SVG enables Web developers and designers to create dynamically generated, high-quality graphics from real-time data with precise structural and visual control.

To learn more about SVG, visit the <http://www.w3.org/TR/svg> Web site.

To download a viewer, go to the <http://www.adobe.com/svg> Web site.

***Instructor notes:***

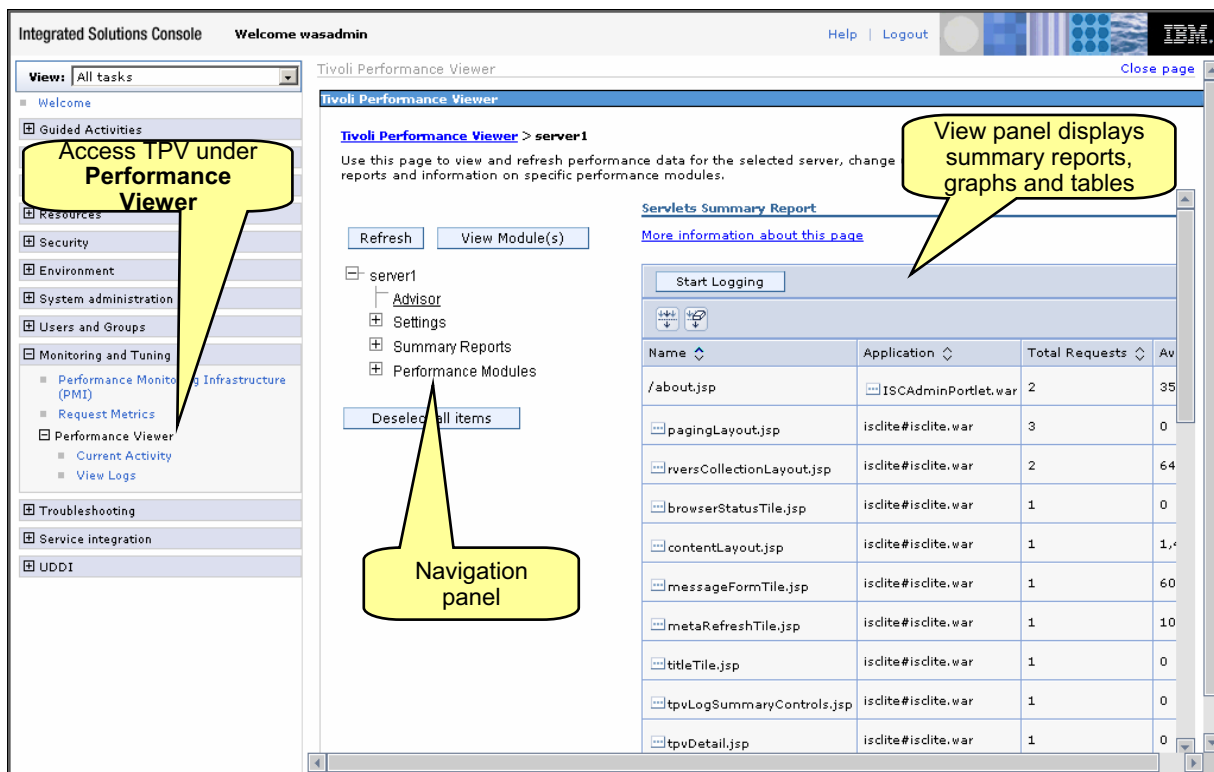
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: TPV user interface

# TPV user interface



© Copyright IBM Corporation 2007

Figure 17-17. TPV user interface

WA3616.3

## Notes:

This screen image shows the different panels comprising the TPV user interface in the administrative console.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: TPV summary reports

## TPV summary reports

---

- **Provide quick and easy access to important metrics for key resource categories:**
  - **Servlets:** Lists key metrics for all servlets that are running in the application server, including:
    - Total number of requests
    - Average response time
  - **EJBs:** Lists key metrics for all enterprise beans running in the server, including:
    - Amount of time spent in their methods
    - Number of EJB invocations
    - Total time spent in each enterprise bean
  - **EJBMethods:** Shows statistics for each EJB method
  - **Connection Pool:** Lists all data source connections that are defined in the application server, and shows their usage over time
  - **Thread Pool:** Shows the usage of all thread pools in the application server over time

© Copyright IBM Corporation 2007

Figure 17-18. TPV summary reports

WA3616.3

### Notes:

The default monitoring level (*Basic*) enables all reports except the report on EJB methods. To enable the EJB method summary report, configure PMI service to use a set of statistics that includes EJB method metrics (*All* or *Custom*).

For servlets, EJBs and EJBMethods:

Sort by *Avg Response Time* to find the slowest.

Sort by *Total Requests* or *Method Calls* to find what is used the most.

Sort by *Total Time* to find the most costly.

For Connection or Thread Pool, when the application is experiencing normal to heavy usage, the pools used by that application should be nearly fully utilized. Low utilization means that resources are being wasted by maintaining connections or threads that are never used. Consider the order in which work progresses through the various pools. If the resources near the end of the pipeline are underutilized, it might mean that resources near the front are constrained, or that more resources than necessary are allocated near the end of the pipeline.

***Instructor notes:***

**Purpose —**

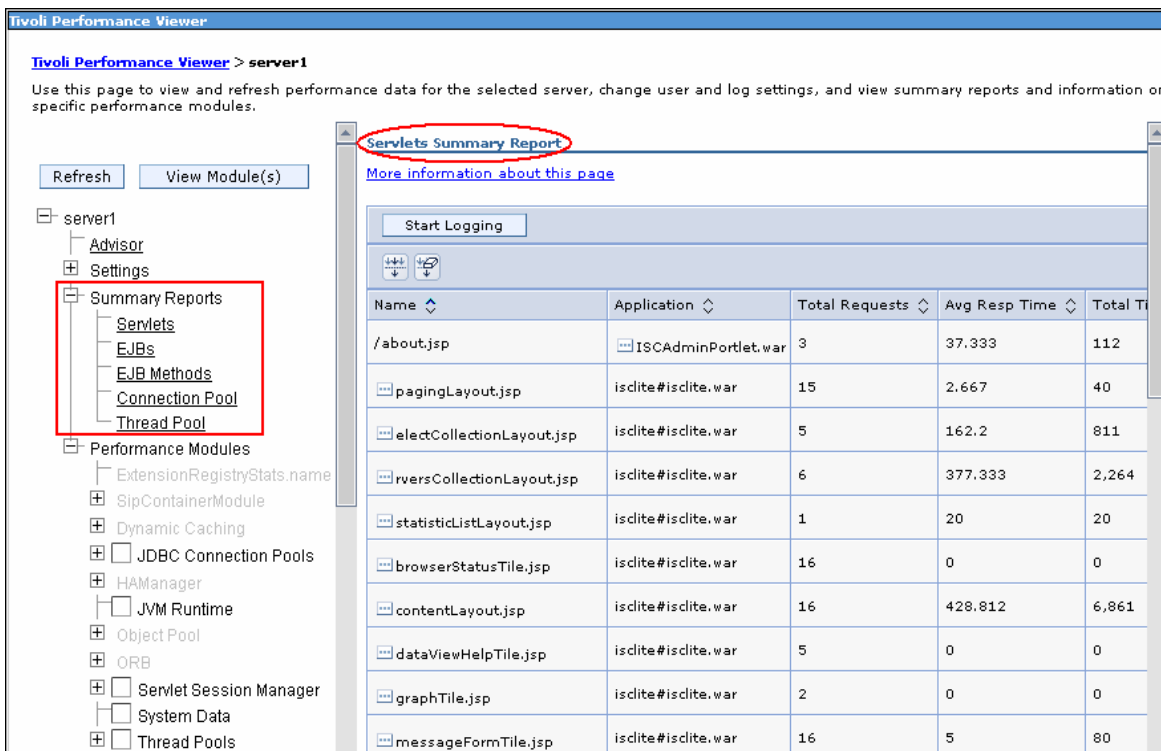
**Details —**

**Additional information —**

**Transition statement —** Next: Summary report example: Servlets



# Summary report example: Servlets



© Copyright IBM Corporation 2007

Figure 17-19. Summary report example: Servlets

WA3616.3

## Notes:

To access a summary report, expand *Summary Reports* in the TPV navigation panel and select from the list.

***Instructor notes:***

**Purpose —**

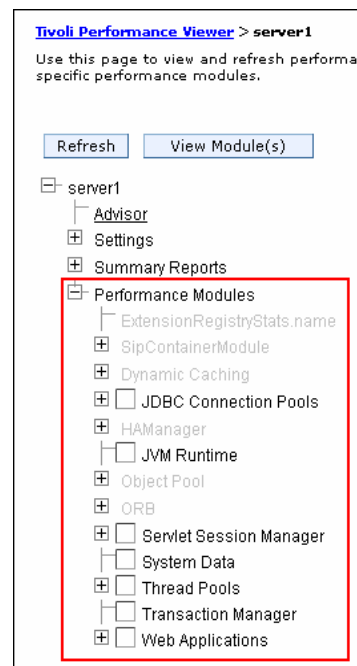
**Details —**

**Additional information —**

**Transition statement —** Next: TPV performance modules

## TPV performance modules

- A performance module represents a category of resources for which performance metrics can be displayed
- Available performance modules include:
  - **JDBC Connection Pools**
    - Usage information for JDBC connection pools
  - **JVM Runtime**
    - Memory usage and JVMTI data
  - **Servlet Session Manager**
    - Usage information for HTTP sessions
  - **Web Applications**
    - Response time and usage of individual servlets
- In the TPV navigator, check the modules to monitor
  - **Corresponding PMI modules need to be enabled before they can be checked for monitoring**



© Copyright IBM Corporation 2007

Figure 17-20. TPV performance modules

WA3616.3

### Notes:

To display performance data by module, extend **Performance Modules** and check the modules to be monitored.

For a complete list of all the available modules and associated counters, visit the WebSphere InfoCenter at:

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>

Select the chapter on *PMI data organization*.

***Instructor notes:***

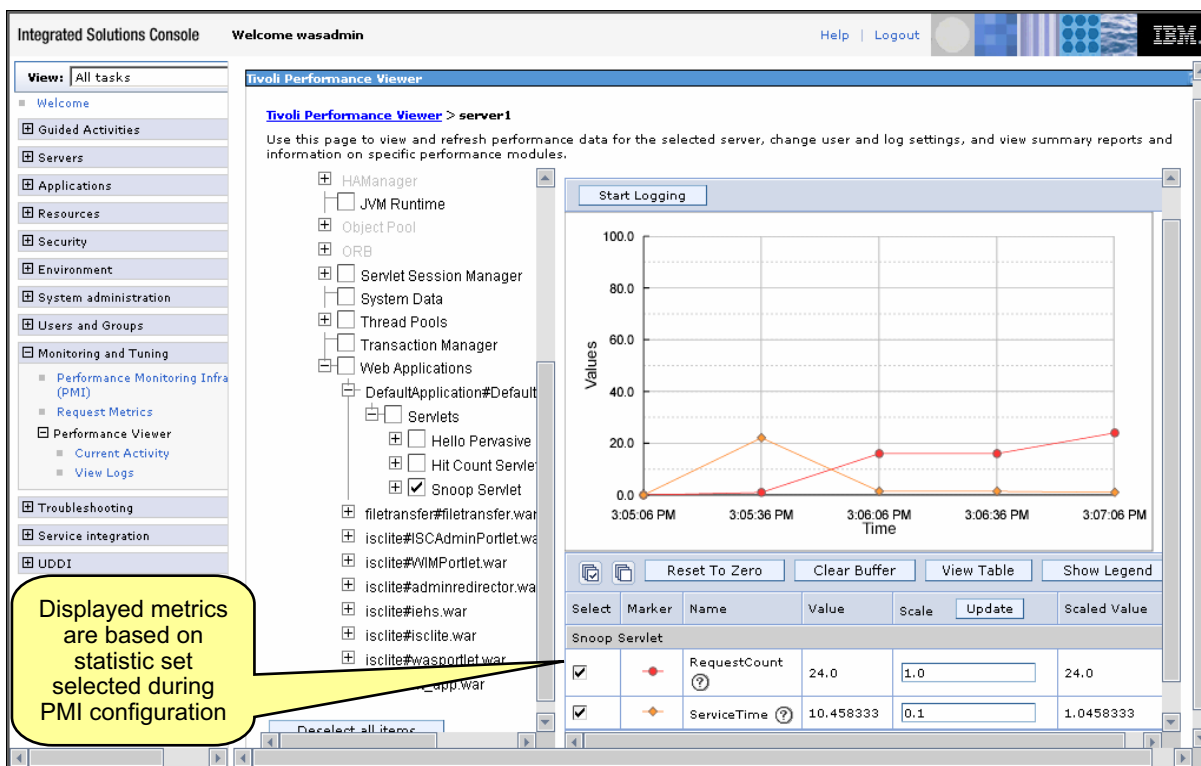
**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Performance module example: Web applications

## Performance module example: Web applications



© Copyright IBM Corporation 2007

Figure 17-21. Performance module example: Web applications

WA3616.3

### Notes:

This screen image shows the monitoring of the *Snoop* servlet. Two graphs are shown for the *RequestCount* and *ServiceTime* counters.

By default, the performance module's data is displayed in a graphic chart in the right pane. To view the data as a table, click **View Table**.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Retrieving PMI data with the Performance servlet

## Retrieving PMI data with the Performance servlet

	Tivoli Performance Viewer	Performance servlet	Request Metrics tool	Performance advisors
<b>Intended usage</b>	<ul style="list-style-type: none"> <li>• Monitor overall system health</li> </ul>	<ul style="list-style-type: none"> <li>• Simple retrieval of performance data in XML format</li> </ul>	<ul style="list-style-type: none"> <li>• Monitor application flow</li> </ul>	<ul style="list-style-type: none"> <li>• Performance tuning</li> </ul>
<b>Performance data source</b>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Request Metrics</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>

© Copyright IBM Corporation 2007

Figure 17-22. Retrieving PMI data with the Performance servlet

WA3616.3

### Notes:

The Performance servlet allows you to retrieve performance data collected by the PMI and view it in an XML format.

***Instructor notes:***

**Purpose —**

**Details —**

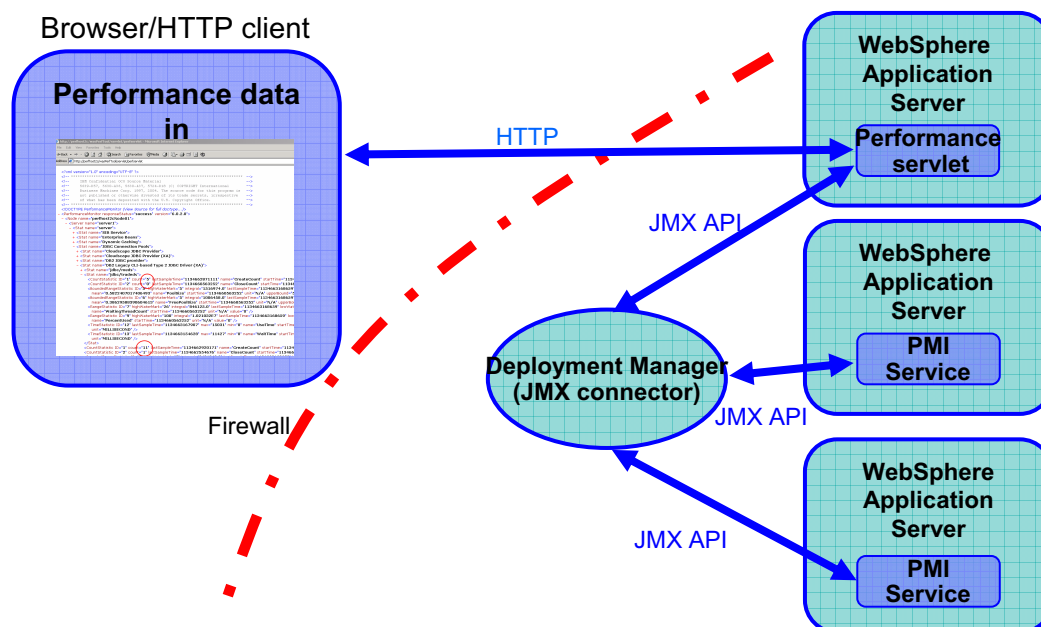
**Additional information —**

**Transition statement —** Next: Performance servlet overview



## Performance servlet overview

- Utility servlet that allows you to retrieve performance data collected by the PMI and view it in an XML format.



© Copyright IBM Corporation 2007

Figure 17-23. Performance servlet overview

WA3616.3

### Notes:

The Performance servlet, also known as *PerfServlet*, displays performance data as an XML document. PerfServlet Version 6.0 uses the JMX PerfMBean interface to retrieve the PMI data and outputs an XML document that uses the J2EE 1.4 Performance Data Framework to describe the statistics.

Note that PerfServlet Version 6.0 can still provide an output that is compatible PerfServlet Version 5.0 by using the deprecated PMI client interface.

The performance servlet is contained in the *PerfServletApp.ear* application file located in the *WAS\_HOME/installableApps* directory.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Performance servlet output example

## Performance servlet output example

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- ***** -->
<!-- IBM Confidential OCO Source Material -->
<!-- 5639-D57, 5630-A36, 5630-A37, 5724-D18 (C) COPYRIGHT International -->
<!-- Business Machines Corp. 1997, 2004. The source code for this program is -->
<!-- not published or otherwise divested of its trade secrets, irrespective -->
<!-- of what has been deposited with the U.S. Copyright Office. -->
<!-- ***** -->
<!DOCTYPE PerformanceMonitor (View Source for full doctype...)>
- <PerformanceMonitor responseStatus="success" version="6.0.2.0">
- <Node name="perfhost2cNode01">
- <Server name="server1">
- <Stat name="server">
+ <Stat name="SIB Service">
+ <Stat name="Enterprise Beans">
+ <Stat name="Dynamic Caching">
- <Stat name="JDBC Connection Pools">
+ <Stat name="Cloudscape JDBC Provider">
+ <Stat name="Cloudscape JDBC Provider (XA)">
+ <Stat name="DB2 JDBC provider">
- <Stat name="DB2 Legacy CLI-based Type 2 JDBC Driver (XA)">
+ <Stat name="jdbc/meds">
- <Stat name="jdbc/tradeds">
+ <CountStatistic ID="1" count="5" lastSampleTime="1134662871111" name="CreateCount" startTir
+ <CountStatistic ID="2" count="0" lastSampleTime="1134660563252" name="CloseCount" startTir
+ <BoundedRangeStatistic ID="3" highWaterMark="5" integral="1516974.0" lastSampleTime="1134663168659"
+ mean="0.5822407017406493" name="PoolSize" startTir="1134660563252" unit="N/A" upperBound="5
+ <BoundedRangeStatistic ID="6" highWaterMark="5" integral="1006458.0" lastSampleTime="1134663168659"
+ mean="0.38629588390604613" name="FreePoolSize" startTir="1134660563252" unit="N/A" upperBo
+ <RangeStatistic ID="7" highWaterMark="26" integral="846125.0" lastSampleTime="1134663168659" lowWat
+ name="WaitingThreadCount" startTir="1134660563252" unit="N/A" value="0" />
+ <RangeStatistic ID="9" highWaterMark="100" integral="1.021032E7" lastSampleTime="1134663168659" low
+ name="PercentUsed" startTir="1134660563252" unit="N/A" value="0" />
+ <TimeStatistic ID="12" lastSampleTime="1134663167907" max="15031" min="0" name="UseTime" startTir
+ unit="MILLISECOND" />
+ <TimeStatistic ID="13" lastSampleTime="1134663154628" max="11427" min="0" name="WaitTime" startTir
+ unit="MILLISECOND" />
+ </Stat>
+ <CountStatistic ID="1" count="11" lastSampleTime="1134662920171" name="CreateCount" startTir="1134
+ <CountStatistic ID="2" count="1" lastSampleTime="1134662554676" name="CloseCount" startTir="113466
```

© Copyright IBM Corporation 2007

Figure 17-24. Performance servlet output example

WA3616.3

### Notes:

The graphic above shows output from the performance servlet. By default, all of the modules are expanded. This collection was collapsed and the JDBC connection pool *create* and *close* values are circled.

The format of the XML document is described by the provided document type definition (DTD) file located inside *PerfServletApp.ear*. It has the following characteristics:

- Uses the J2EE 1.4 Performance Data Framework to describe the statistics
- Leaves of the structure provide the actual observations of performance data
- Paths provide the context to the leaves

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Performance servlet features

## Performance servlet features

- Provides a way to use simple HTTP requests to query performance metrics
  - Good for situations where firewall issues need to be addressed
- Metrics can be queried for a single server or an entire WebSphere Application Server administrative domain
  - Entire domain
    - Enter the default Web address:  
`http://hostname/wasPerfTool/servlet/perfservlet`
  - Node
    - To limit the data collection to the node 'rjones', enter the Web address:  
`http://hostname/wasPerfTool/servlet/perfservlet?node=rjones`
  - Server
    - To limit the data collection to the 'testserver' server located on the host 'rjones', enter the Web address:  
`http://hostname/wasPerfTool/servlet/perfservlet?node=rjones&server=testserver`
  - Module
    - To limit the data collection to the 'beanModule' on the server 'testserver' on the node 'rjones', invoke the Web address:  
`http://hostname/wasPerfTool/servlet/perfservlet?node=rjones&server=testserver&module=beanModule`
- Servlet is contained in the *PerfServletApp.ear* application file located in the *WAS\_HOME/installableApps* directory
  - EAR file must be installed prior to servlet use

© Copyright IBM Corporation 2007

Figure 17-25. Performance servlet features

WA3616.3

### Notes:

You can also use a subset of the above. For example, to limit the data collection to the *beanModule* on all servers and nodes, invoke the following URL:

```
http://hostname/wasPerfTool/servlet/perfservlet?module=beanModule
```

You can request multiple modules by using the following URL:

```
http://hostname/wasPerfTool/servlet/perfservlet?module=beanModule+jvmRuntimeModule
```

To find the list of the modules, invoke the PerfServlet help with the following URL:

```
http://hostname/wasPerfTool/servlet/perfservlet?action=help
```

When the performance servlet is first initialized, it retrieves the list of nodes and servers within the domain in which it is deployed. Because the collection of this data is expensive, the performance servlet holds this information as a cached list. If a new node is added to the domain or a new server is started, the performance servlet does not automatically retrieve the information about the newly created element.

To force the servlet to refresh its configuration, you must add the *refreshConfig* parameter to the invocation as follows:

`http://hostname/wasPerfTool/servlet/perfservlet?refreshConfig=true`

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Monitoring application flow with Request Metrics

## Monitoring application flow with Request Metrics

	Tivoli Performance Viewer	Performance servlet	Request Metrics tool	Performance advisors
<b>Intended usage</b>	<ul style="list-style-type: none"> <li>• Monitor overall system health</li> </ul>	<ul style="list-style-type: none"> <li>• Simple retrieval of performance data in XML format</li> </ul>	<ul style="list-style-type: none"> <li>• Monitor application flow</li> </ul>	<ul style="list-style-type: none"> <li>• Performance tuning</li> </ul>
<b>Performance data source</b>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Request Metrics</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>

© Copyright IBM Corporation 2007

Figure 17-26. Monitoring application flow with Request Metrics

WA3616.3

### Notes:

Request Metrics track the time spent by a transaction in the major system components that it traverses, allowing you to understand the application flow that satisfies the user's request. This view of performance is important for the in-depth internal understanding of who is using specific resources and which application components can be potential bottlenecks.



***Instructor notes:***

**Purpose —**

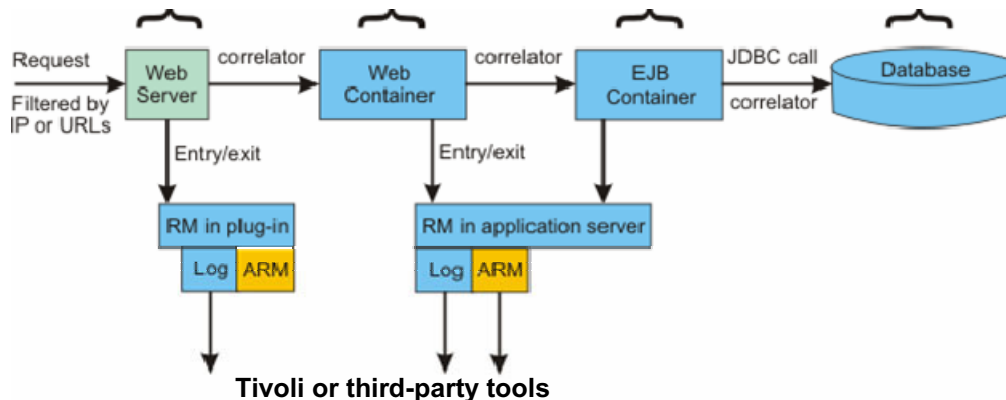
**Details —**

**Additional information —**

**Transition statement —** Next: Request Metrics overview

## Request Metrics overview

- Tracing facility that allows you to measure the amount of time a request spends in each component traversed during its execution



- Captured information include:
  - Elapsed time in the Web server
  - Response time of invoked components in the Web and EJB containers
  - Response time of related JDBC calls
- Writes trace records to *SystemOut.log* and/or sends metrics to an Application Response Measurement (ARM) agent

© Copyright IBM Corporation 2007

Figure 17-27. Request Metrics overview

WA3616.3

### Notes:

Request Metrics allow you to monitor the transaction flow and analyze the response time of the components involved in executing it. This analysis can help you target performance problem areas and debug resource constraint problems. For example, it can help determine if a transaction spends most of its time in the Web server plug-in, the Web container, the Enterprise JavaBeans (EJB) container or the back-end database. The response time that is collected for each level includes the time spent at that level and the time spent in the lower levels. For example, if the total response time for the servlet is 130 milliseconds, and it includes 38 milliseconds from the enterprise beans and Java Database Connectivity, then, 92 ms can be attributed to the servlet process.

Note that an ARM agent does not ship with WebSphere Application Server and is usually provided by third-party tools.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Request Metrics features

## Request Metrics features

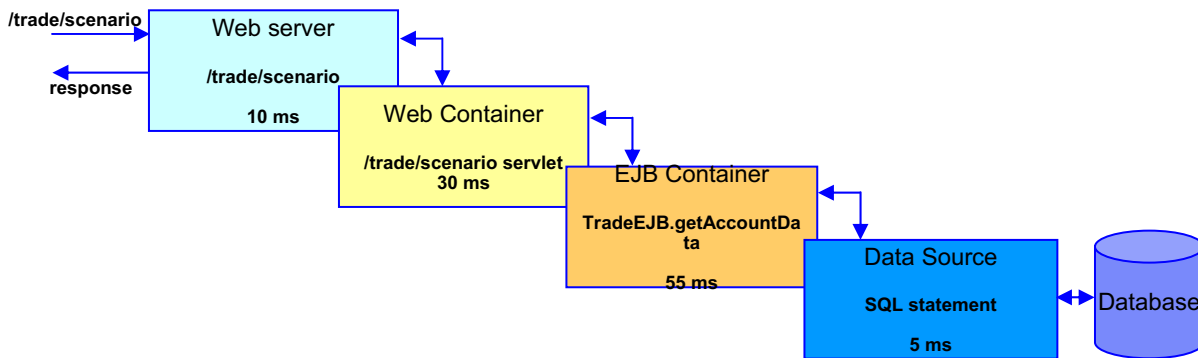
- Tracks HTTP requests and enterprise bean remote requests
  - Provides filtering mechanism to monitor only transactions of interest
- Provides measurements across multiple processes and services
  - Achieved by using [data correlation](#)
  - Displays [cumulative response times](#) and reflects a hierarchical view of data. For example:

The correlated Request Metrics measurements for the HTTP request below...

```

type=HTTP detail=/trade/scenario elapsed=100
  type=URI detail=/trade/scenario elapsed=90
type=EJB detail=TradeEJB.getAccountData elapsed=60
  Type=JDBC detail=executeQuery elapsed=5
  
```

... indicate the following transaction flow and individual component response times:



© Copyright IBM Corporation 2007

Figure 17-28. Request Metrics features

WA3616.3

### Notes:

In the example above, the individual component response times are calculated from the Request Metrics measurements as follows:

- Web server response time =  $100 - 90 = 10$  ms
- Servlet response time =  $90 - 60 = 30$  ms
- EJB response time =  $60 - 5 = 55$  ms
- JDBC response time = 5

Request Metrics are instrumented in the following components:

- Web server plug-in which is only available when using the Web server port.
- Proxy Server, instrumented as servlet and Web services requests.
- Web container, including servlet and servlet filters
- Enterprise JavaBeans (EJB) container

- Java DataBase Connectivity (JDBC) calls
- J2EE Connector Architecture (JCA)
- Web services, both on the server and the client side
- the Java Message Service (JMS) engine
- Service Integration Bus (SIB)
- Portlet container, including the portlet requests
- Asynchronous Beans

Comparing Request Metrics to PMI:

- PMI provides information about average **system resource** usage, with no correlation between the data across different WebSphere components
- Request Metrics provides data about each **individual transaction**, correlating this information across the various WebSphere components to provide an end-to-end picture of the transaction

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Using Request Metrics (1 of 2)

## Using Request Metrics (1 of 2)

- In the administrative console:

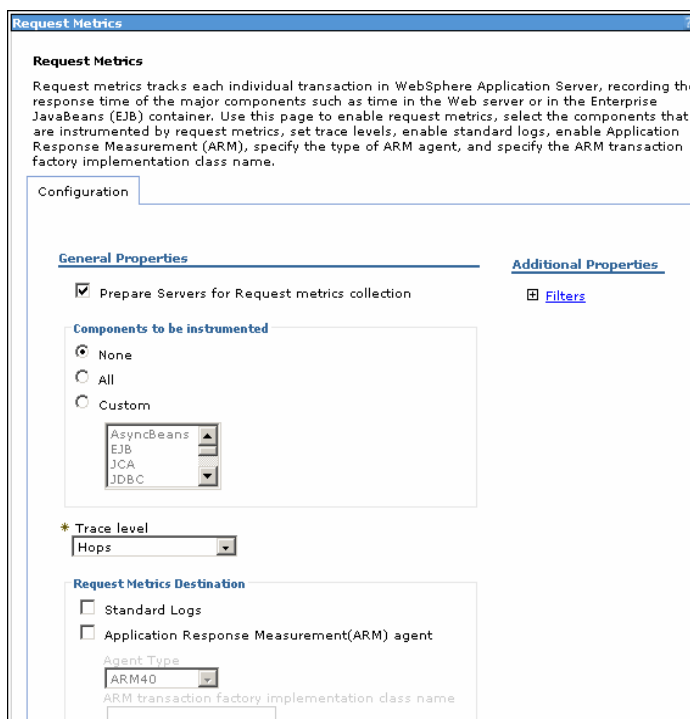
- Select Monitoring and Tuning → Request Metrics

- Enable Request Metrics
  - Check **Prepare servers for request metrics collection**

- Specify components to be instrumented
  - Select **All**, **None** or **Custom**

- Specify how much data to collect
  - **None**, **Hops**, **Performance\_debug** or **Debug**

- Specify destination
  - Standard Logs
  - ARM agent



© Copyright IBM Corporation 2007

Figure 17-29. Using Request Metrics (1 of 2)

WA3616.3

### Notes:

The components to be instrumented that you can specify are:

- Asynchronous Beans (AsyncBeans)
- Enterprise JavaBeans (EJB)
- J2EE Connector Architecture (JCA)
- Java Database Connectivity (JDBC)
- Java Message Service (JMS)
- Java Naming and Directory Interface (JNDI)
- Portlet
- Service Integration Bus (SIB)
- Servlet
- Servlet Filter
- Web Services

The available Trace levels consist of:

- Hops
  - Generates instrumentation information on process boundaries only (for example, a servlet request coming from a browser or a Web server and a JDBC request going to a database).
- Performance\_debug
  - Generates the data at Hops level and the first level of the intra-process servlet and Enterprise JavaBeans (EJB) call (for example, when an inbound servlet forwards to a servlet and an inbound EJB calls another EJB). Other intra-process calls like naming and service integration bus (SIB) are not enabled at this level.
- Debug
  - Provides detailed instrumentation data, including response times for all intra-process calls. Note that requests to servlet filters will only be instrumented at this level.

The trace records for Request Metrics data are output to two log files: the Web server plug-in log file and the application server log file. The default directory for these log files is *plugin\_install\_root/logs/web\_server\_name/http\_plugin.log* and *install\_root/profiles/profile\_name/logs/server\_name* and the default names are SystemOut.log and http\_plugin.log.

If you are enabling an Application Response Measurement (ARM) agent, you must also specify its type (ARM40 or TIVOLI\_ARM) and supply the name of the ARM transaction factory implementation class.



***Instructor notes:***

**Purpose —**

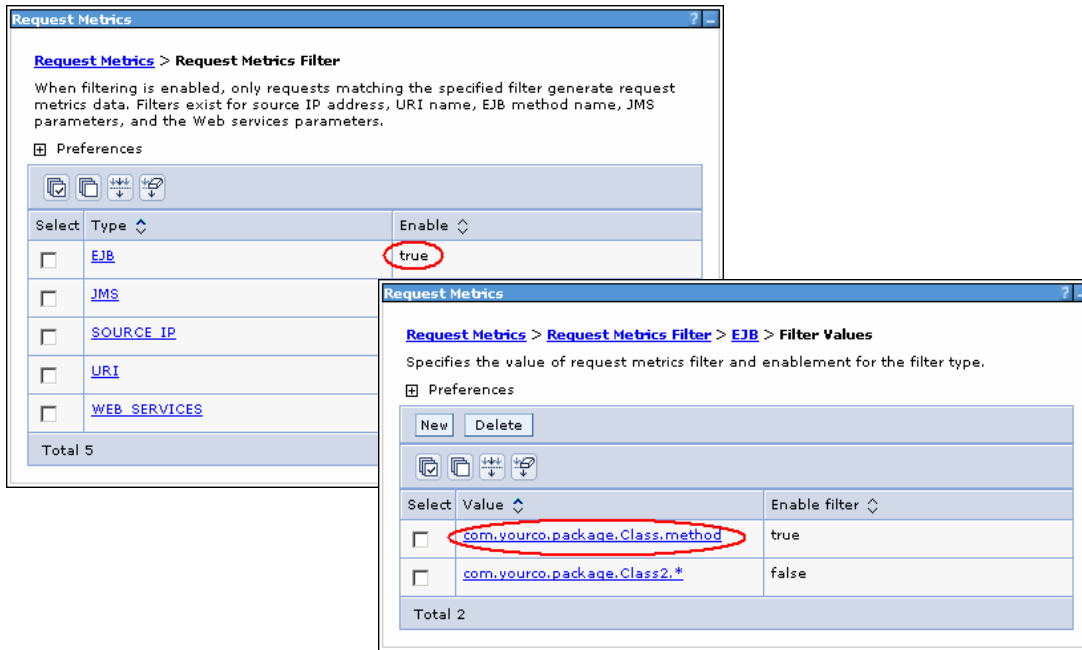
**Details —**

**Additional information —**

**Transition statement —** Next: Using Request Metrics (2 of 2)

## Using Request Metrics (2 of 2)

- Specify filters for the requests to be tracked
  - Incoming HTTP requests can be filtered by IP address, URI or both
  - Incoming enterprise bean requests can be filtered by method name



© Copyright IBM Corporation 2007

Figure 17-30. Using Request Metrics (2 of 2)

WA3616.3

### Notes:

When you have the isolated problem areas, use the Request Metrics filtering mechanism to focus specifically on those areas. For example, when you have an isolated problem in a particular servlet or EJB method, use the Uniform Resource Identifier (URI) algorithms or EJB filter to enable the instrumentation only for the servlet or EJB method. This filtering mechanism supports a more focused performance analysis.

Five types of filters are supported:

- Source IP filter
- URI filter
- EJB method name filter
- JMS parameters filter
- Web services parameters filter

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Using the Performance advisors to tune WebSphere

## Using the Performance advisors to tune WebSphere

	Tivoli Performance Viewer	Performance servlet	Request Metrics tool	Performance advisors
<b>Intended usage</b>	<ul style="list-style-type: none"> <li>• Monitor overall system health</li> </ul>	<ul style="list-style-type: none"> <li>• Simple retrieval of performance data in XML format</li> </ul>	<ul style="list-style-type: none"> <li>• Monitor application flow</li> </ul>	<ul style="list-style-type: none"> <li>• Performance tuning</li> </ul>
<b>Performance data source</b>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>	<ul style="list-style-type: none"> <li>• Request Metrics</li> </ul>	<ul style="list-style-type: none"> <li>• Performance Monitoring Infrastructure (PMI)</li> </ul>

© Copyright IBM Corporation 2007

Figure 17-31. Using the Performance advisors to tune WebSphere

WA3616.3

### Notes:

The Performance advisors provide configuration recommendations that can help you tune your WebSphere Application Server environment.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Performance advisors overview

## Performance advisors overview

---

- WebSphere provides two separate advisors:
  - Performance and Diagnostic Advisor
  - Tivoli Performance Viewer Advisor
- Both provide configuration advice based on collected PMI data on a per server basis
  - Advisors do not compare counters among different application servers
- Advice based on basic rules of thumb for tuning WebSphere Application Server
  - Rules are IBM-defined and non-configurable
- Do not automatically tune WebSphere based on advice
  - Administrator must manually apply recommendations
  - Suggested settings must be checked against baseline performance to verify improvement

© Copyright IBM Corporation 2007

Figure 17-32. Performance advisors overview

WA3616.3

### **Notes:**

There are two performance advisors that provide suggestions to help tune systems for optimal performance. Both advisors use Performance Monitoring Infrastructure (PMI) data to suggest configuration changes to the WebSphere thread pools, connection pools, prepared statement cache, session cache and heap size, among others.

***Instructor notes:***

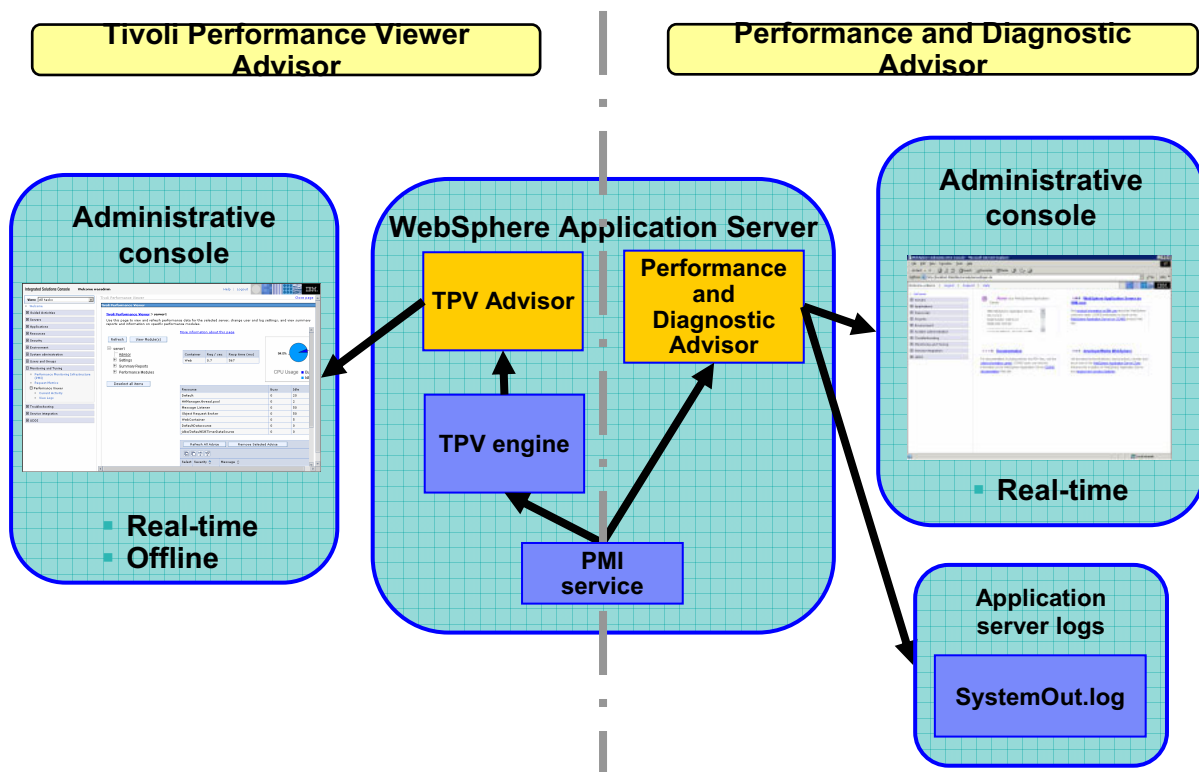
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Advisor architecture

# Advisor architecture



© Copyright IBM Corporation 2007

Figure 17-33. Advisor architecture

WA3616.3

## Notes:

This slide shows the architecture of the Performance advisors in a single server configuration: both run within the same application server JVM process.

In a Network Deployment environment:

- The Performance and Diagnostic Advisor still runs in the JVM of each application server in the cell.
- The TPV Advisor runs in the JVM of the Node Agent of each individual node in the cell.



***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: TPV Advisor and Performance and Diagnostic Advisor comparison

## TPV Advisor and Performance and Diagnostic Advisor comparison

	Runs when you ask for it <ul style="list-style-type: none"> <li>• Good for calculation intensive operations</li> </ul>	Runs in the background <ul style="list-style-type: none"> <li>• The first step toward auto tuning</li> </ul>
	Tivoli Performance Viewer Advisor	Performance and Diagnostic Advisor
<b>Location of execution</b>	Application server or Node Agent	Application server
<b>Invocation of tool</b>	Tivoli Performance Viewer	Administrative console
<b>Output</b>	Advisor section of Tivoli Performance Viewer	<ul style="list-style-type: none"> <li>• SystemOut.log file</li> <li>• Runtime Messages section of Administrative console</li> <li>• JMX notifications</li> </ul>
<b>Frequency of operation</b>	When you select <b>Refresh</b> in Tivoli Performance Viewer	Configurable
<b>Types of advice</b>	Performance advice: <ul style="list-style-type: none"> <li>• ORB service thread pools</li> <li>• Web container thread pools</li> <li>• Connection pool size</li> <li>• Persisted session size and time</li> <li>• Prepared statement cache size</li> <li>• Session cache size</li> <li>• Dynamic cache size</li> <li>• JVM heap size</li> <li>• DB2 Performance Configuration Wizard</li> </ul>	Performance advice: <ul style="list-style-type: none"> <li>• ORB service thread pools</li> <li>• Web container thread pools</li> <li>• Connection Pool size</li> <li>• Persisted session size and time</li> <li>• Prepared statement cache size</li> <li>• Session cache size</li> <li>• Memory leak detection</li> </ul> Diagnostic advice: <ul style="list-style-type: none"> <li>• Connection factory diagnostics</li> <li>• Data source diagnostic</li> </ul>

© Copyright IBM Corporation 2007

Figure 17-34. TPV Advisor and Performance and Diagnostic Advisor comparison

WA3616.3

### Notes:

Another important difference is that the Tivoli Performance Viewer Advisor requires that you enable individual PMI performance modules or counters, whereas the Performance and Diagnostic Advisor just requires the PMI service to be enabled. Also, the TPV Advisor can be run against the TPV logs.

The Performance and Diagnostic Advisor is integrated with the WebSphere runtime, and, when enabled, generates advice continuously based on PMI data collected for a configurable time interval.

Starting with WebSphere Application Server, Version 6.0.2, you can use the Performance and Diagnostic Advisor to enable the lightweight memory leak detection, which is designed to provide early detection of memory problems in test and production environments.

The diagnostic features of the Performance and Diagnostic Advisor are discussed in the next slide.

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Examples of advice

## Examples of advice

---

- Data Sources
  - Situation: The prepared statement discard rate is too high, and heap space is available
  - Advice: Increase statement cache size
  
- Thread pools (ORB, Web Container, Data Source (DS))
  - Situation: The number of connections is low (at the minimum)
  - Advice: Decrease pool size
  
  - Situation: All DS connections are heavily used, and heap space is available
  - Advice: Increase maximum pool size
  
  - Situation: The size of the pool is fluctuating a lot (high variance), possibly indicating batch processing and wasted resources
  - Advice: Decrease pool size
  
- JVM heap size
  - Situation: Heap size is too small (less than 256 MB)
  - Advice: Increase the heap size to a value greater than 256 MB

© Copyright IBM Corporation 2007

Figure 17-35. Examples of advice

WA3616.3

### **Notes:**

Additional examples of advice that the Performance advisors would give in certain situations include:

#### **Unbounded thread pools**

Situation: Threads added to an unbounded pool are not pooled

Advice: If the average number of threads is higher than the pool size, then the pool should be increased in order to allow better pooling

#### **Sessions**

Situation: Read/Write time or size is too large

Advice: Warn of application problem

Situation: The number of live sessions is greater than the session cache, and memory is available

Advice: Increase Session Cache

Situation: Requests are turned down because there is no room for new sessions

Advice: There are either too many active sessions, or the cache size is too small

### **DB2 Performance Configuration Wizard**

Situation: A DB2 database is detected in the configuration

Advice: Use the DB2 Performance Wizard to tune the indicated database

***Instructor notes:***

**Purpose —**

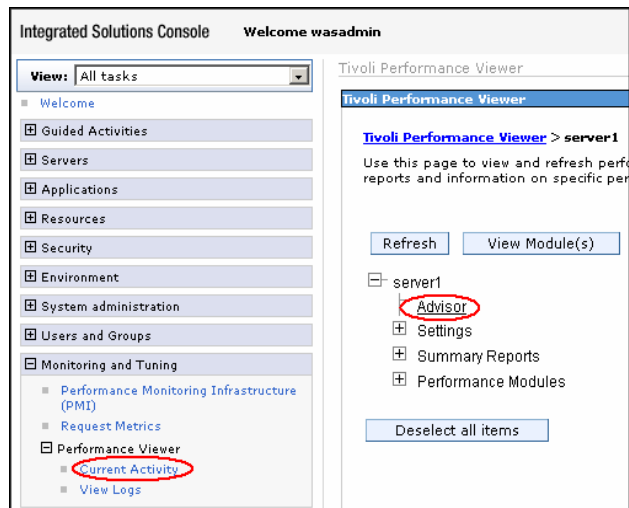
**Details —**

**Additional information —**

**Transition statement —** Next: Using the TPV Performance Advisor

## Using the TPV Performance Advisor

- Enable PMI and select **Extended** statistic set
- If desired, selectively enable data collection on the PMI metrics that you want to monitor
- Start Tivoli Performance Viewer, and navigate to the Performance Advisor section
  - In the administrative console, select **Monitoring and Tuning -> Performance Viewer -> Current Activity**.
  - Click **Advisor** in the Tivoli Performance Viewer navigation tree
- Run production level test
- Click **Refresh All Advice** on top of the advice table to recalculate the advice based on the current data in the buffer
- Advice is refreshed based on last minute (non-configurable) of PMI data
- Evaluate advice and make configuration changes



© Copyright IBM Corporation 2007

Figure 17-36. Using the TPV Performance Advisor

WA3616.3

### Notes:

The Tivoli Performance Viewer (TPV) Performance Advisor generates advice only when refreshed from within the Advisor section. The advice is for only a single point in time. In order to generate more advice you will need to refresh the tool again.

It is also important to note that the advice is generated over only the last minute of PMI data that is gathered. Since PMI data is taken over an interval of time and averaged to provide advice, details within the advice message appear as averages.

Enable PMI data collection and select the *Extended* statistic set. The statistic set (also called the monitoring level) determines which data counters are enabled. You can select the statistics set dynamically on the *Runtime* tab without restarting the server. The monitoring level directly influences the type of advice you obtain.

The steps described above describe using the TPV Performance Advisor against a running application server, but it can also be run against a log file. To do this, open the log file in TPV and start replaying it. Then, in the Advisor section of TPV, refresh to view any advice that may be generated against the data.

For more accurate advice, set TPV refresh rate to 10 seconds or less.



***Instructor notes:***

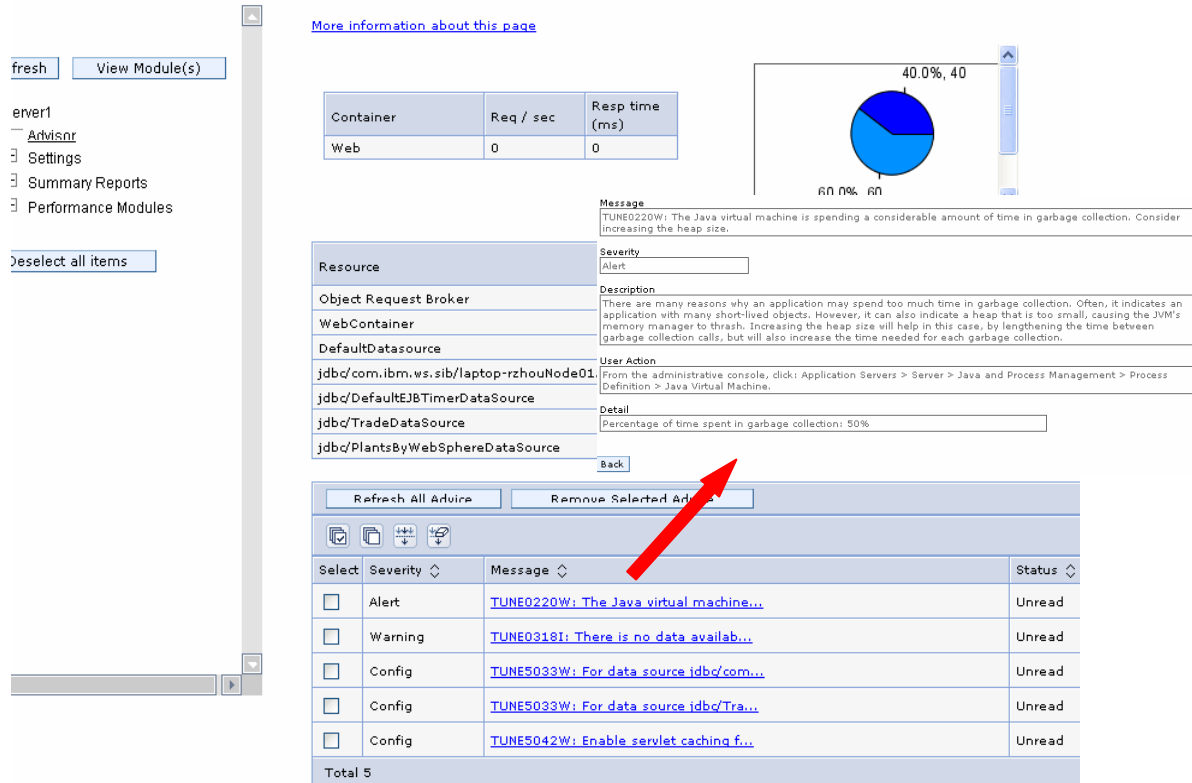
**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: TPV Performance Advisor output example

# TPV Performance Advisor output example



© Copyright IBM Corporation 2007

Figure 17-37. TPV Performance Advisor output example

WA3616.3

## Notes:

This TPV Performance Advisor output contains the following information:

- Table showing generated advice
- Table showing throughput and response time for Web and EJB Containers
- Resource usage table for Web, ORB, and Database connection pools
- CPU usage chart

Double-click one of the entries in the advice table to see the details of the message.

In example above, you can see that the CPU is currently busy 40% of the time. The advisor has noted that the JVM is spending a large amount of time performing garbage collections, and recommends that the heap size be increased. It describes where to make the configuration modification.

***Instructor notes:***

**Purpose —**

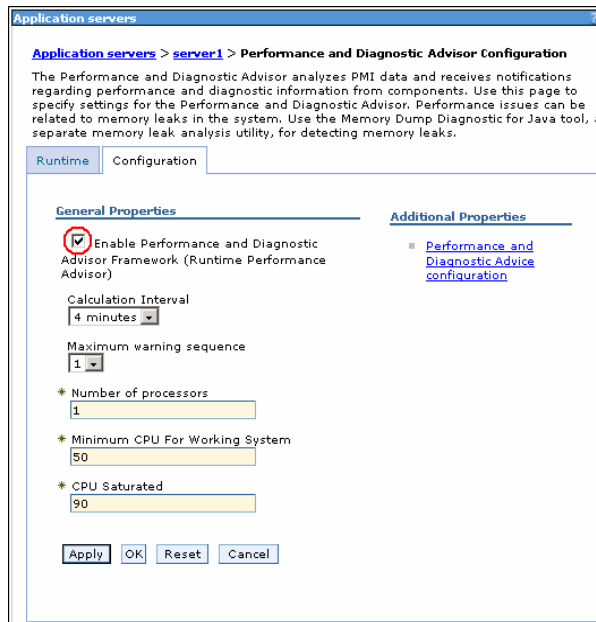
**Details —**

**Additional information —**

**Transition statement —** Next: Using the Performance and Diagnostic Advisor

# Using the Performance and Diagnostic Advisor

- Enable PMI
- Change configuration settings under **Servers -> Application Servers -> [SERVER\_NAME] -> Performance and Diagnostic Advisor Configuration**
  - [Number of processors](#)
  - [Calculation Interval](#)
  - [Maximum Warning Sequence](#)
- Optionally, specify the categories for recommendations
- Restart server in order for new settings to take effect
- Run production level test
- Enable the Performance and Diagnostic Advisor
- Monitor Warnings in WebSphere Runtime Messages log
  - [Advice generated every ten seconds \(non-configurable\)](#)



© Copyright IBM Corporation 2007

Figure 17-38. Using the Performance and Diagnostic Advisor

WA3616.3

## Notes:

The settings under *General properties* are:

- **Calculation Interval:** The period of time over which performance advice is averaged; details within the advice messages appear as averages over this interval.
- **Maximum warning sequence:** The number of consecutive warnings issued before the threshold is updated. For example, if the warning sequence is set to 3, the advisor will only send three warning messages to indicate that a certain threshold has been exceeded. After that, a new alert will only be issued if the new PMI data value exceeds the new threshold.
- **Number of processors:** The number of processors in the system where the application server is running.
- **Minimum CPU for Working System:** The CPU level that indicates an application server is under production load.
- **CPU Saturated:** The level at which the CPU is considered fully utilized.

In the optional step above, you can choose the advice categories for which recommendations should be collected. By default, WebSphere Application Server tries to collect information for all categories where data is available. To view this configuration page, click **Servers > Application Servers > *server\_name* > Performance and Diagnostic Advisor Configuration**, and under the Additional Properties section, select **Performance and Diagnostic Advice Configuration**.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Performance and Diagnostic Advisor output example (1 of 2)

## Performance and Diagnostic Advisor output example (1 of 2)

- Look for advice warnings in:
  - The WebSphere Application Server log file `SystemOut.log`
  - The **Runtime Messages** panel of the administrative console under **Troubleshooting -> Runtime Messages -> Runtime Warning**

**Runtime Events**  
Runtime events propagating from the server

Preferences

Timestamp	Message Originator	Message
Oct 19, 2004 9:28:35 AM EDT	com.ibm.ws.performance.tuning.serverAlert.TraceResponse	<a href="#">TUNE0208W: Data source jdbc/PlantsByWebSphereDataS</a>
Oct 19, 2004 9:28:35 AM EDT	com.ibm.ws.performance.tuning.serverAlert.TraceResponse	<a href="#">TUNE0206W: Decreasing the connection pool for data</a>
Oct 19, 2004 9:28:35 AM EDT	com.ibm.ws.performance.tuning.serverAlert.TraceResponse	<a href="#">TUNE0208W: Data source jdbc/DefaultEJBTimerDataSou</a>
Oct 19, 2004 9:28:35 AM EDT	com.ibm.ws.performance.tuning.serverAlert.TraceResponse	<a href="#">TUNE0208W: Data source jdbc/com.ibm.ws.sib/laptop-</a>
Oct 19, 2004 9:28:35 AM EDT	com.ibm.ws.performance.tuning.serverAlert.TraceResponse	<a href="#">TUNE0208W: Data source DefaultData:source does not</a>
Total 5		

© Copyright IBM Corporation 2007

Figure 17-39. Performance and Diagnostic Advisor output example (1 of 2)

WA3616.3

### Notes:

The advice given by the Performance and Diagnostic Advisor is all on the server level. The only difference when running in a Network Deployment environment is that you might receive contradictory advice on resources that are declared at the node or cell level and used at the server level.

For example, two sets of advice are given if a data source is declared at the node level to have a connection pool size of {10,50} and is used by two servers (server1 and server2). If server1 uses only two connections, and server2 uses all fifty connections during peak load, the optimal connection pool size is different for the two servers. Therefore, the Performance and Diagnostic Advisor gives two sets of advice, one for each server. The data source is declared at the node level, and you must make your decisions appropriately by setting one size that works for both, or by declaring two different data sources for each server with the appropriate level.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Performance and Diagnostic Advisor output example (2 of 2)



## Performance and Diagnostic Advisor output example (2 of 2)

[Runtime Events](#) > [Message Details](#)

Runtime events propagating from the server

**General Properties**

---

**Message**  
 TUNE0206W: Decreasing the connection pool for data source jdbc/TradeDataSource by setting the minimum size to 0 and the maximum size to 3 may improve performance. Additional explanatory data follows. Pool utilization: 1.2%. This alert has been issued 1 time(s) in a row. The threshold will be updated to reduce the overhead of the analysis.

**Message type**  
 Warning

**Explanation**  
 Decreasing the size supports better pooling and frees memory resources.

**User action**  
 From the administrative console, click: Resources > JDBC Providers > JDBC\_provider > Data Sources > data\_source > Connection pool properties.

**Message Originator**  
 com.ibm.ws.performance.tuning.serverAlert.TraceResponse

**Source object type**  
 RasLoggingService

**Timestamp**  
 Oct 19, 2004 9:28:35 AM EDT

**Thread Id**  
 15

**Node name**  
 laptop-rzhouNode01

**Server name**  
 server1

[Back](#)

© Copyright IBM Corporation 2007

Figure 17-40. Performance and Diagnostic Advisor output example (2 of 2)

WA3616.3

### Notes:

This slide shows the message details for a generated advice.

***Instructor notes:***

**Purpose** —

**Details** —

**Additional information** —

**Transition statement** — Next: Performance advisors best practices

## Performance advisors best practices

---

- Should only be used during stable production load tests
  - Application needs to remain stable during production tests
    - Any exceptions and deadlock issues need to be resolved prior to running
  - The test load should be consistent
    - Varied load might lead to contradictory advice
- Should be enabled once production load tests have reached peak load levels
  - Exclude ramp-up and ramp-down times from monitoring
    - Increasing or decreasing loads might lead to contradictory advice
  - Certain types of advice only generated when CPU is being stressed
- Important: tune your application before you tune WebSphere

© Copyright IBM Corporation 2007

Figure 17-41. Performance advisors best practices

WA3616.3

### Notes:

When using the Performance advisors, CPU utilization must rise above 50% before advice is generated. Typically when running your production level load, you would push the CPU usage to 80-100% before turning on one of the Performance advisors.

Consider the following when using a Performance advisor for tuning:

- If the load changes on the system under test, contradictory advice will be generated. This is because the collected PMI data will show a different type of environment, causing the advice to shift. To avoid this, always run the advisors while simulating the load WebSphere will experience during deployment (usually peak load).
- If the pool size's minimum and maximum values are the same, the Performance advisor rules are much more likely to give contradictory advice when load fluctuates.
- The amount of system activity is determined by the amount of CPU usage. The advisors do not consider disk activity, network activity, memory usage, or other factors to get a more realistic view of system load.
- Recommendations are only generated when CPU load reaches 50% and higher.

- Performance advisors from different application servers may give contradictory advice on the same node resources. This is because the application servers *only* take into account how they are individually employing the resource. In this situation, if the advice from the different advisors varies greatly, consider the generated advice and make the changes that make sense. However, if all advisors are giving the same recommendations, then you should seriously consider the suggested changes.
- If the Performance advisor suggests setting a pool size to  $X$ , you should set the minimum value to  $X/2$  and the maximum value to  $X$ .
- If the Performance advisor suggests setting the *Prepared Statement Cache* value to a certain setting, check the amount of memory that is available before setting. The advisors do not take into account the amount of actual physical memory available on the system before making the suggestions.

***Instructor notes:***

**Purpose —**

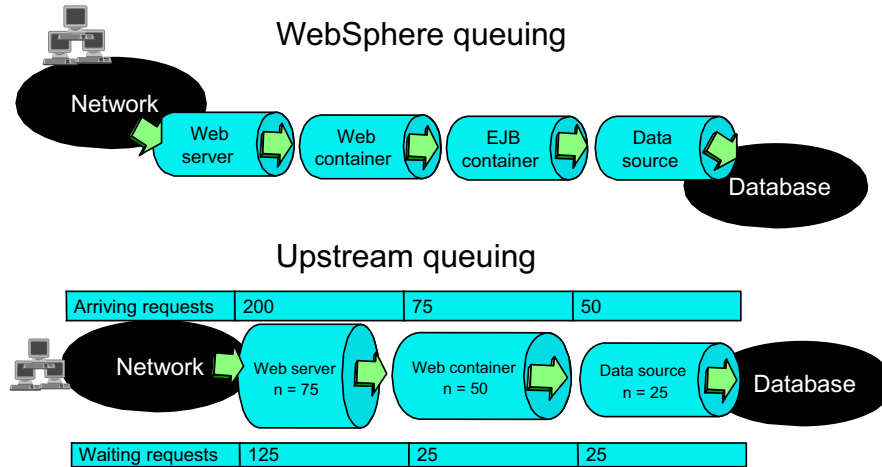
**Details —**

**Additional information —**

**Transition statement —** Next: WebSphere tuning best practice: Upstream queuing

## WebSphere tuning best practice: Upstream queuing

- Upstream queuing attempts to allow more work to be done by limiting the number of connections at each tier of the application
- Performance advisors rule set employs this best practice when generating advice



© Copyright IBM Corporation 2007

Figure 17-42. WebSphere tuning best practice: Upstream queuing

WA3616.3

### Notes:

Consider the different levels when setting-up a funneling effect on the work being done by WebSphere.

Design the pooling system to allow as much work on the database and application server as possible, without over-stressing each area. ORB Thread pool size in the EJB container is a soft limit, whereas every other connection can be implemented as a hard limit. The different pools can be used to funnel the work in order to allow the resources to be used by the proper processes.

Do not overload any one tier because it will result in poor performance due to waiting threads. This is one of the most difficult things to do during performance testing. For instance, if a database is being overworked and causing slow response time, you may need to lower the number of connections. This allows those active connections to get the resources they need to perform the requested queries.

Most of these settings can be monitored by the Tivoli Performance Viewer and changed through the Administrative console. The objective is to achieve maximum throughput within the environment.

***Instructor notes:***

**Purpose** — It is important to review the queuing mechanism within WebSphere as the advisors specifically target these modules.

**Details** —

**Additional information** —

**Transition statement** — Next: Checkpoint



## Checkpoint

---

1. What are the two performance data collection technologies in WebSphere?
2. Which WebSphere performance tool allows you to monitor overall system health?
3. T/F: The Performance Monitoring Infrastructure is enabled by default.
4. T/F: The Tivoli Performance Viewer Advisor tool generates tuning advices and automatically applies them to the environment.

© Copyright IBM Corporation 2007

Figure 17-43. Checkpoint

WA3616.3

### **Notes:**

Write your answers here:

- 1.
- 2.
- 3.
- 4.

**Instructor notes:**

Purpose —

Details —

## Checkpoint solutions

---

1. The Performance Monitoring Infrastructure (PMI) and Request Metrics provide the data collection mechanisms in WebSphere.
2. The Tivoli Performance Viewer allows you to monitor overall system health.
3. True. PMI is enabled by default.
4. False. The performance advisor tools do not automatically tune the environment. You have to do this manually and test the effect of the changes.

© Copyright IBM Corporation 2007

**Additional information** —

**Transition statement** — Next: Unit summary

## Unit summary

---

Having completed this unit, you should be able to:

- Describe the features of Integrated Performance Viewer
- Describe how performance data is collected in Integrated Performance Viewer
- Set data collection levels on started and stopped resources
- Describe the performance data collected by Integrated Performance Viewer
- Use the logging function for viewing and storing data
- Set up and view a graphical representation of collected data

© Copyright IBM Corporation 2007

Figure 17-44. Unit summary

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Lab exercise

## Lab exercise

---



© Copyright IBM Corporation 2007

Figure 17-45. Lab exercise

WA3616.3

### Notes:

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**

---

# Unit 18.Course summary

## Estimated time

00:20

## What this unit is about

This unit serves as the wrap up to the course.

## What you should be able to do

After completing this unit, you should be able to:

- Describe the objectives that were presented at the beginning of the course.
- List various resources available for further study.

## How you will check your progress

- There are no exercises associated with this unit.

## References

SG24-7304-00 WebSphere Application Server V6.1 System  
Management and Configuration

<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp>  
WebSphere Application Server Version 6.1  
Information Center

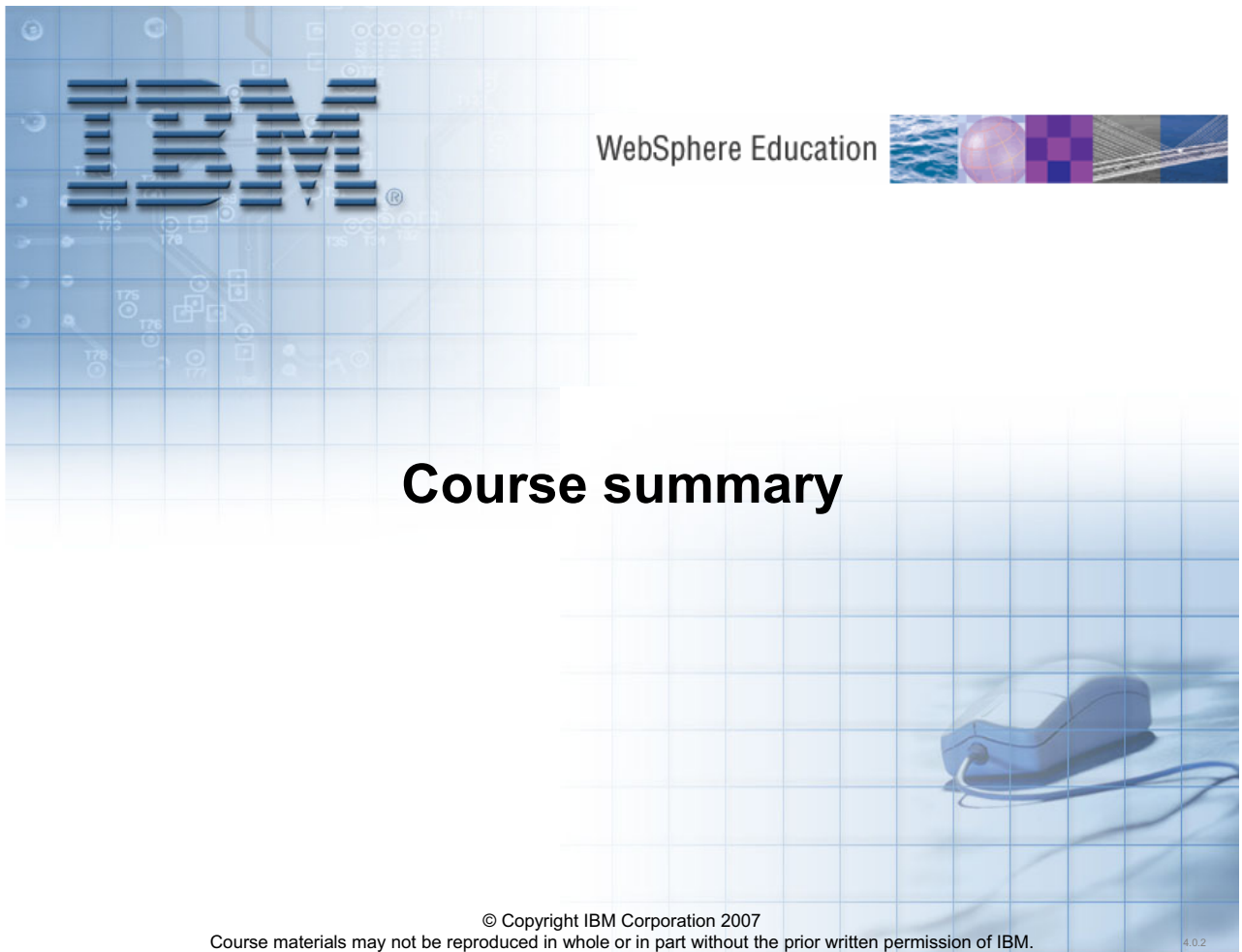


Figure 18-1. Course summary

WA3616.3

**Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Course objectives

## Course objectives

---

After completing this course, students should be able to :

- Relate WebSphere Application Server to the WebSphere family of products
- Describe the architectural upgrades and additions related to WebSphere Application Server V6.1
- Install and configure WebSphere Application Server and IBM HTTP Server
- Explain the basics of the Java 2 Platform, Enterprise Edition 1.4
- Assemble and configure server-side Java enterprise applications using WebSphere Application Server Toolkit (AST)
- Use WebSphere Application Server administrative tools to configure and manage enterprise applications
- Configure WebSphere Platform Messaging (WPM)
- Configure and deploy enterprise applications
- View performance information about server and application components
- Use problem determination tools and log files to troubleshoot problems

© Copyright IBM Corporation 2007

Figure 18-2. Course objectives

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Resources

## Resources

---

- Web pages
  - WebSphere: <http://www.ibm.com/websphere>
  - Redbooks: <http://www.ibm.com/redbooks>
  - Education: <http://www.ibm.com/services/learning>  
<http://ibm.com/developerworks/websphere/education/enablement/>
  
- Forums at: [ibmforums.ibm.com](http://ibmforums.ibm.com)
  - [software.websphere.appserver](http://software.websphere.appserver)
  - [software.\\*](#) (many other available)

© Copyright IBM Corporation 2007

Figure 18-3. Resources

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Other WebSphere courses

## Other WebSphere courses

---

- IBM WebSphere Application Server
- IBM Rational Application Developer
- IBM WebSphere Portal
- IBM WebSphere Commerce
- IBM WebSphere Process Server

© Copyright IBM Corporation 2007

Figure 18-4. Other WebSphere courses

WA3616.3

### **Notes:**

***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —** Next: Evaluations

## Evaluations

---

- Your impressions of this class are important
- You can provide feedback via the course evaluation forms



© Copyright IBM Corporation 2007

Figure 18-5. Evaluations

WA3616.3

### **Notes:**



***Instructor notes:***

**Purpose —**

**Details —**

**Additional information —**

**Transition statement —**





