# IBM

@server

WebSphere Application Server for iSeries

# Troubleshooting

*Version 5.0.2*

# IBM

# @server

WebSphere Application Server for iSeries

# Troubleshooting

*Version 5.0.2*

> **Note**
>
> Before using this information and the product it supports, be sure to read the information in
> "Notices," on page 39.

# Contents

# Troubleshooting

WebSphere Application Server offers several methods you can use to troubleshoot problems. Which method you use depends on the nature of the problem. Generally, you use a combination of these methods to determine the cause of a problem and then decide on an appropriate method for its resolution. Whether you are a beginner or experienced user, this problem determination section leads you to resources and techniques to help you identify and respond to problems.

**"Troubleshoot by topic"**
This topic describes several general areas where problems may occur, and what resources you should use to determine the cause of the problem.

**"Resources for monitoring the WebSphere Application Server environment" on page 15**
This topic describes the resources you can use to monitor the WebSphere Application Server environment.

**"WebSphere Application Server log files" on page 23**
This topic describes the log files created by the WebSphere Application Server components. The topic also includes information on how to configure the log files.

**"Use the WebSphere Application Server trace service" on page 31**
This topic describes how to enable and configure the WebSphere Application Server trace service. The topic also includes information on interpreting the contents of a trace file.

**Add logging and tracing to your application**
This topic describes how to add logging and tracing to your applications using the JRas framework.

**Use the debugger tools**
This topic describes the debugger tools available for debugging your code running in WebSphere Application Server for iSeries.

**"WebSphere Application Server Message Reference" on page 34**
This topic describes the messages issued by WebSphere Application Server. The topic includes message identifiers, descriptions, and the user action which may be taken for a particular message.

**WebSphere Application Server system topology**
This topic describes the topology of the WebSphere Application Server environment and how your application fits into this topology.

## Troubleshoot by topic

In order to navigate through a problem, you must first identify the area in which the problem is occurring and, from there, use the appropriate resources to determine what is causing the problem.

Generally, problems occurring within WebSphere Application Server for iSeries fall within one of the categories shown below. Select the link that most closely matches the area of the problem to see the appropriate resources to consult for determining what is causing the problem.

All of the resource lists in the topics below include a link to the WebSphere Application Server Version

5.0 for iSeries Frequently Asked Questions (FAQs) database  . This database contains FAQs for many of the common problems encountered in the areas listed below. Each FAQ contains a short description of the problem, the symptom associated with the problem, and the solution to the problem.

**"Troubleshoot: Install" on page 3**

This topic describes the resources available to determine what may be causing a problem when you install WebSphere Application Server.

**"Troubleshoot: WebSphere Application Server startup" on page 4**

This topic describes the resources available to determine what may be causing a problem when you attempt to start the WebSphere application server or administrative console.

**"Troubleshoot: Install and configure resources" on page 4**

This topic describes the resources available to determine what may be causing a problem when you attempt to create, manage, or configure resources from the WebSphere administrative console.

**"Troubleshoot: Application Assembly Tool" on page 4**

This topic describes the resources available to determine what may be causing a problem when you attempt to use the Application Assembly Tool to configure an enterprise application.

**"Troubleshoot: WebSphere Application Server Administration and administrative console" on page 5**

This topic describes the resources available to determine what may be causing a problem when you attempt to use the administrative console or the WebSphere administrative scripting (wsadmin) tool.

**"Troubleshoot: Security" on page 6**

This topic describes the resources available to determine what may be causing a problem when using WebSphere Application Server security.

**"Troubleshoot: Naming service" on page 8**

This topic describes the resources available to determine what may be causing a problem when you attempt to use the naming service to lookup WebSphere resources such as data sources or enterprise beans.

**"Troubleshoot: Access Web resources" on page 8**

This topic describes the resources available to determine what may be causing a problem when a WebSphere application server fails to serve Web resources such as servlets, JavaServer Pages[TM] (JSP[TM]), and HTML files.

**"Embedded JMS provider" on page 10**

This topic describes the resources available to determine what may be causing a problem when you are attempting to use the Embedded JMS Provider included with WebSphere Application Server.

**"Troubleshoot: Enterprise applications" on page 10**

This topic describes the resources available to determine what may be causing a problem when you attempt to run your enterprise application within the WebSphere Application Server environment.

**Run a J2EE client application**

This topic describes the resources available to determine what may be causing a problem when you attempt to run your J2EE client application.

**"Troubleshoot: Universal Description, Discovery, and Integration (UDDI), Web Service, and SOAP" on page 14**

This topic describes the resources available to determine what may be causing a problem when you attempt to deploy and run applications that use WebSphere Application Server Web Services, UDDI, or SOAP.

# Troubleshoot: Install

If a problem occurs when you attempt to install WebSphere Application Server or WebSphere Application Server Network Deployment, the first thing you should do is consult the the installation documentation for the product. For WebSphere Application Server, refer to the Installation of WebSphere Application Server for iSeries documentation. For WebSphere Application Server Network Deployment, refer to the Installation of WebSphere Application Server Network Deployment for iSeries documentation. Step 2 of the install documentation takes you through the installation process in detail.

For the resources available for determining what the installation problem might be, see the list below.

- The WebSphere Application Server for iSeries installation code outputs error messages when something goes wrong with the install. When you install locally from Qshell or use the RUNJVA command, the output is displayed to the screen. When you install remotely from a workstation, the error messages are displayed to the screen from which you launched the install.

  If an OS/400 message is associated with the error, the message identifier is displayed in the error message. To get more information about the message, use the Display Message Description (DSPMSGD) command from an OS/400 command line.

- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html)  .

- Check the WebSphere Application Server FAQ database  .

- Refer to the WebSphere Application Server for iSeries newsgroup  .
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.

- Contact IBM support. See "Get support for WebSphere Application Server for iSeries" for more information.

## Get support for WebSphere Application Server for iSeries

Support services for IBM WebSphere Application Server for iSeries are provided under the usual terms and conditions for iSeries software products. Support services include program services, voice support, and consulting services. For more information, use the online information provided at

http://www.ibm.com/eserver/iseries/support/  or contact your local IBM representative. To refer to IBM WebSphere Application Server V5.0 for iSeries, specify program number 5733-WS5. To refer to IBM WebSphere Application Server V5.0, Network Deployment for iSeries., specify program number 5733-WS5, Option 5.

Resolving defects of WebSphere Application Server is supported under program services or voice support. Resolving application programming or debugging issues is supported under consulting services.

All programming assistance is supported under consulting services. This includes the program samples that are provided in the WebSphere Application Server for iSeries licensed program offering (LPO). Additional samples may be available on the Internet on an unsupported basis.

For the latest information about WebSphere Application Server, including PTF information, frequently asked questions (FAQ), and the WebSphere Application Server for iSeries newsgroup, check the IBM WebSphere Application Server for iSeries product Web site at

http://www.ibm.com/eserver/iseries/software/websphere/wsappserver/ 

**Getting fixes for IBM WebSphere Application Server for iSeries**

Program fixes for 5733-WS5 (IBM WebSphere Application Server Version 5.0 for iSeries) are available through iSeries Program Temporary Fixes (PTFs). Use the online information that is provided at

http://www.ibm.com/eserver/iseries/support/  for more information.

For the latest information about fixes, see the IBM WebSphere Application Server for iSeries product Web site at http://www.ibm.com/eserver/iseries/software/websphere/wsappserver/  .

## Troubleshoot: WebSphere Application Server startup

If a problem occurs when you attempt to start the WebSphere Application Server for iSeries product, the first thing you should do is consult the installation of WebSphere Application Server for iSeries documentation. For WebSphere Application Server, refer to the Installation of WebSphere Application Server for iSeries documentation. For WebSphere Application Server Network Deployment, refer to the Installation of WebSphere Application Server Network Deployment for iSeries documentation. Steps 3 and 4 guide you through the system verification and startup process.

These resources are available to help you determine what the startup problem might be.

- Check the joblog of the failing WebSphere Application Server job for errors. For information on viewing joblogs, see "Monitor WebSphere Application Server jobs on the iSeries server" on page 15.
- Check the WebSphere Application Server log files for errors. For information on the application server log files and where they are located, see "WebSphere Application Server log files" on page 23.
- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html)  .
- Check the WebSphere Application Server FAQ database  .
- Refer to the WebSphere Application Server for iSeries newsgroup  .
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.
- Contact IBM support. For more information, see "Get support for WebSphere Application Server for iSeries" on page 3.

## Troubleshoot: Install and configure resources

The resources listed below are available to determine the cause of a problem.

- If a problem occurs when you are installing or configuring a resource, typically an error page or message containing error information is displayed. Click the Details button to view more information on the possible cause of the problem.
- Check the application server standard output and standard error log files. For information on the log files and where they are located, see "WebSphere Application Server log files" on page 23.
- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html)  .
- Check the WebSphere Application Server FAQ database  .
- Refer to the WebSphere Application Server for iSeries newsgroup 
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.
- Contact IBM support. See "Get support for WebSphere Application Server for iSeries" on page 3 for more information.

## Troubleshoot: Application Assembly Tool

If you are having problems starting or using the WebSphere Application Server Application Assembly Tool (AAT), perform these steps:

- Enable the printing of messages and exceptions to the screen.
  - Modify the `assembly.bat` or `assembly.sh` file located in the bin directory of the product installation. Change the statement `"start javaw"` to just `"java"`.

- Restart the AAT and the command window is displayed through the lifetime of the Java process and displays messages and exceptions.
  - Look up any error or warning messages you see in the message reference link to message reference section.
- With a problem application open in the AAT, use the Verify menu command. This command goes through all components of the application and validates them for any XML errors or invalid entries such as missing fields, or invalid bean or class references.
- To verify the integrity of an EAR (Enterprise Archive) file, expand it manually (outside of the AAT) by running the EARExpander script and supplying the name of the EAR file as a parameter. For more information on the EARExpander script, see the The EARExpander script topic in *Administration*. Browse the directory structure of the expanded EAR file to see if it contains all the expected files.
- Ensure that the enterprise application complies with J2EE specification level 1.3 and that any enterprise beans it contains conform to the EJB 1.1 or 2.0 Specification level.
- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) .

- Check the WebSphere Application Server FAQ database .

- Refer to the WebSphere Application Server for iSeries newsgroup
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.
- Contact IBM support. For more information, see "Get support for WebSphere Application Server for iSeries" on page 3.

## Troubleshoot: WebSphere Application Server Administration and administrative console

Use the following resources to determine the cause of the problem when attempting to use the administrative console or the WebSphere admin scripting tool (wsadmin):

- Administrative functions are supported by the application server (such as ″server1″) for a WebSphere Application Server instance, or the Deployment Manager (such as ″dmgr″) for a Network Deployment instance.

  The server process must be running in order to use the administrative console. For wsadmin, many functions require a connection to the server as well. Verify that the server is running and ready to receive administrative requests. For more information on how to verify that the server is running in WebSphere Application Server, see Verify that the WebSphere Application Server environment has started in the *Installation* topic. For more information on how to verify that the server is running in Network Deployment, see Verify that the Network Deployment environment has started in the *Installation* topic.

  If the server is not started and you are using wsadmin for an administrative function that does not require the server to be started, ensure that you have specified -conntype NONE when invoking wsadmin.

- Use the TCP/IP `ping` command to test that the hostname where the application server or Deployment Manager is running is reachable from the system where the browser or wsadmin program are being used. If you are able to ping this hostname, this indicates that there are no firewall or connectivity issues.

- If the host where the application server or Deployment Manager is running is remote to the machine from which the client browser or wsadmin command is running, ensure that the hostname in the browser URL for the console is correct, or the `-host hostname` option of the wsadmin command is being used to direct wsadmin to the right server.

- If you are using the wsadmin tool, read the The wsadmin administrative tool topic in *Administration*. Also check the wsadmin log files located in the logs directory of your instance (by default) for errors.

- Check the logs files for the server for errors. For information on the log files and where they are located, see "WebSphere Application Server log files" on page 23.
- Read the release notes.WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) 🔜 .

- Check the WebSphere Application Server FAQ database 🔜 .

- Refer to the WebSphere Application Server for iSeries newsgroup 🔜 .
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.

- Contact IBM support. See "Get support for WebSphere Application Server for iSeries" on page 3 for more information.

## Troubleshoot: Security

Use these resources to determine the cause of problems that occur when using WebSphere Application Server security.

- Check the application server standard output and standard error log files. For information on the log files and where they are located, see "WebSphere Application Server log files" on page 23.

- When troubleshooting security-related problems, these questions are very helpful and should be considered:

  **Does the problem occur when security is disabled?**
  This is a good litmus test to determine that a problem is security related. However, just because a problem only occurs when security is enabled does not always make it a security problem. More troubleshooting is necessary to ensure the problem is really security-related.

  **Did security appear to initialize properly?**
  A lot of security code is visited during initialization. So you likely see problems there first if the problem is configuration related. The following sequence of messages generated in the SystemOut.log indicate normal code initialization of an application server. This varies based on the configuration, but the messages are similar:

```
SASRas        A JSAS0001I: Security configuration initialized.
SASRas        A JSAS0002I: Authentication protocol: CSIV2/IBM
SASRas        A JSAS0003I: Authentication mechanism: SWAM
SASRas        A JSAS0004I: Principal name: BIRKT20/pbirk
SASRas        A JSAS0005I: SecurityCurrent registered.
SASRas        A JSAS0006I: Security connection interceptor initialized.
SASRas        A JSAS0007I: Client request interceptor registered.
SASRas        A JSAS0008I: Server request interceptor registered.
SASRas        A JSAS0009I: IOR interceptor registered.
NameServerImp I NMSV0720I: Do Security service listener registration.
SecurityCompo A SECJ0242A: Security service is starting
UserRegistryI A SECJ0136I:
 Custom Registry:com.ibm.ws.security.registry.nt.NTLocalDomainRegistryImpl
 has been initialized
SecurityCompo A SECJ0202A: Admin application initialized successfully
SecurityCompo A SECJ0203A: Naming application initialized successfully
SecurityCompo A SECJ0204A: Rolebased authorizer initialized successfully
SecurityCompo A SECJ0205A: Security Admin mBean registered successfully
SecurityCompo A SECJ0243A: Security service started successfully
SecurityCompo A SECJ0210A: Security enabled true
```

  **Is there a stack trace or exception printed in the SystemOut.log?**
  A single stack trace can tell a lot about the problem. What code initiated the code that failed? What is the failing component? Which class did the failure actually come from?

  **Is this a distributed security problem or a local security problem?**

– If the problem is local, that is the code involved does not make a remote method invocation, then troubleshooting is isolated to a single process. It is important to know when a problem is local versus distributed since the behavior of the Object Request Broker (ORB), among other components, is different between the two.

– Once a remote method invocation takes place, an entirely different security code path is entered. When you know that the problem involves two or more servers, the techniques of troubleshooting change. You need to check the log files of all servers involved. If possible, make sure the timestamps on all machines match as closely as possible so that you can find the request and reply pair from two different processes.

**Is the problem related to authentication or authorization?**
Most security problems fall under one of these two categories. Authentication is the process of determining who the caller is. Authorization is the process of validating that the caller has the proper authority to invoke the requested method. When authentication fails, typically this is related to either the authentication protocol, authentication mechanism or user registry. When authorization fails, this is usually related to the application bindings from assembly and/or deployment and to the caller's identity who is accessing the method and the roles required by the method.

**Is this a Web or EJB request?**
Web requests have a completely different code path than EJB requests. Also, there are different security features for Web requests than for EJB requests, requiring a completely different body of knowledge to resolve. For example, when using the LTPA authentication mechanism, the Single SignOn feature is available for Web requests but not for EJB requests. Web requests involve HTTP header information not required by EJB requests due to the protocol differences. Also, the Web container (or servlet engine) is involved in the entire process. Any of these components could be involved in the problem and all should be considered during troubleshooting, based on the type of request and where the failure occurs.
Secure EJB requests heavily involve the ORB and Naming components since they flow over the RMI/IIOP protocol. In addition, when work flow management (WLM) is enabled, other behavior changes in the code can be observed. All of these components interact closely for security to work properly in this environment.

**Does the problem seem to be related to SSL?**
The Secure Socket Layer (SSL) is just that, a totally distinct, separate layer of security. Troubleshooting SSL problems are usually separate from troubleshooting authentication and/or authorization problems. There are many things to consider. Usually, SSL problems are first time setup problems because the configuration can be difficult.
Each client must contain the server's signer certificate. During mutual authentication, each server must contain the client's signer certificate. Also, there can be protocol differences (SSLv3 vs. TLS), and listener port problems related to stale IORs (for example, IORs from a server reflecting the port prior to the server restarting).

**Is the problem related to Java 2 Security?**
Java 2 Security is a new feature in WebSphere Application Server V5.0. It is a new programming model that is very pervasive and has a huge impact on application development. It is disabled by default, but is enabled automatically when Global Security is enabled. However, you can disable or enable it independently of Global Security.
There are implications if Java 2 Security is enabled, deployers or administrators are required to make sure that all the applications are granted the required permissions, otherwise, applications may fail to run. For example, you may see a message similar to this in you SystemOut.log:
java.security.AccessControlException: access denied (java.security.SecurityPermission printIdentity) For information on using and trouble shooting Java 2 Security, see Java 2 security.

- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) .

- Check the WebSphere Application Server FAQ database .

- Refer to the WebSphere Application Server for iSeries newsgroup
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.
- Contact IBM support. For more information, see "Get support for WebSphere Application Server for iSeries" on page 3.

## Troubleshoot: Naming service

The name service is a J2EE service which publishes and provides access to resources such as connection pools, enterprise beans, message listeners, etc, to client processes. If you have problems in accessing a resource which otherwise appears to be healthy, the naming service might be involved. Use these resources to determine the cause of the problem.

- Check the log files of the server for errors. Messages starting with NMSV are related to the Naming Service. For information on the log files and where they are located, see "WebSphere Application Server log files" on page 23.
- Use the Log Analyzer link to view the IBM service log of the server which is hosting the resource you are trying to access and check for error and warning messages. For more information on the Log Analyzer tool, see "Monitor the WebSphere Application Server environment by using the Log Analyzer" on page 20.
- With WebSphere Application Server running, use the `dumpNameSpace` script to view the resources bound into the name space for the application server. For more information on the dumpNameSpace script, see the The dumpNameSpace script topic in *Application Development*. This command results in a display of the objects in the name space, including the path and object name. Note that in a Network Deployment environment, there is a distributed name space. This means there are several name services, and each name service has a different name space. Be sure you connect to the same name service (by using the same host and port in the initial context) that the client experiencing the problem is. For more information, see the Java Naming and Directory Interface (JNDI) topic in *Programming*.

  If the object a client needs to access does not appear in the name space, use the administrative console to verify that:
  - The server hosting the target resource is started.
  - The Web module or EJB module, if applicable, hosting the target resource is running.
  - The jndi name of the target resource is correct and updated.
- If you see an exception that appears to be CORBA related ("CORBA" appears as part of the exception name) look for a naming-services-specific CORBA minor code, further down in the exception stack, for information on the real cause of the problem.
- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) .
- 

  Check the WebSphere Application Server FAQ database .
- Refer to the WebSphere Application Server for iSeries newsgroup .
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.
- Contact IBM support. See "Get support for WebSphere Application Server for iSeries" on page 3 for more information.

## Troubleshoot: Access Web resources

When you are unable to access a Web resource from your browser, you use different resources to determine what the problem is, depending on the nature of the problem occurring. For information about which resources to use, see the list of resources below.

**Unable to serve any Web resources**

- See the Installation of WebSphere Application Server for iSeries topic. Steps 3, 4, and 5 of that document guide you through the system verification and serving of a web resource. To ensure that your application server and Web server are configured correctly and have been started, read the information in steps 3, 4, and 5.

- If you are using IBM HTTP Server (powered by Apache) for your Web server, generate a trace of your HTTP server instance and look for errors. For information on tracing an IBM HTTP Server instance, see the "Check HTTP server logs for troubleshooting information" on page 24 topic in.

- Web server plugin trace. For information on tracing the WebSphere Application Server Web server plugin, see "Check HTTP server logs for troubleshooting information" on page 24.

- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) .

- Check the WebSphere Application Server FAQ database .

- Refer to the WebSphere Application Server for iSeries newsgroup .
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.

- Contact IBM support. For more information, see "Get support for WebSphere Application Server for iSeries" on page 3.

**Unable to serve a particular web resource**

- Check the application server standard output and standard error log files for errors. For information on the application server log files and where they are located, see "WebSphere Application Server log files" on page 23.

- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) .

- Check the WebSphere Application Server FAQ database .

- Refer to the WebSphere Application Server for iSeries newsgroup .
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.

- IBM support services. For more information, see "Get support for WebSphere Application Server for iSeries" on page 3.

**Servlet or JSP displays an error or Java exception instead of the expected output**

- Check the application server standard output and standard error log files for errors. For information on the application server log files and where they are located, see "WebSphere Application Server log files" on page 23.

- Add logging and tracing to your application using the JRas framework. For more information, see the Add logging and tracing to your application topic in *Programming*.

- Add output to your servlet, JSP, or Java code in the form of System.out.println() statements to aid in debugging. The output from any System.out.println() statements is written to the standard output log file for the application server in which the code is running. For information on the application server log files and where they are located, see "WebSphere Application Server log files" on page 23.

- Use one of the debugger tools available for debugging WebSphere Application Server for iSeries to step into the servlet or Java class in which the error is occurring.

- If the failure is occurring within a JSP, use the keepgenerated attribute to keep the generated servlet source file to inspect for problems.

- Web server plugin trace. For information on tracing the WebSphere Application Server Web server plugin, see "Check HTTP server logs for troubleshooting information" on page 24.

- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) .

- Check the WebSphere Application Server FAQ database .

- Refer to the WebSphere Application Server for iSeries newsgroup .
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for
  iSeries.

- IBM support services. For more information, see "Get support for WebSphere Application Server for
  iSeries" on page 3.

## Embedded JMS provider

Use the resources listed below to determine the cause of problems that occur when using the embedded
JMS provider within WebSphere Application Server.

- Ensure that you do not have a WebSphere Application Server Version 5.1 instance running with the
  same instance name as your Version 5.0.x instance. If you do have a Version 5 instance with the same
  name as a Version 5.1 instance, the following error appears in the SystemOut.log file when you try to
  start the second instance:

```
Starting the Queue Manager
Start Queue Manager command failed with exit code: 24
```

  See Use embedded JMS with your applications in the *Administration* topic for more information.

- Check the application server standard output and standard error log files for errors. For information on
  the application server log files and where they are located, see "WebSphere Application Server log
  files" on page 23.

- If you are using publish/subscribe messaging model, check the WebSphere Embedded Messaging
  publish and Subscribe (WEMPS) event log. The file name of the log is wemps_event.log and the file is
  located in Integrated File System directory /QIBM/UserData/WebAS5/wemps/log

- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) .

- Check the WebSphere Application Server FAQ database .

- Refer to the WebSphere Application Server for iSeries newsgroup .
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for
  iSeries.

- Contact IBM support. For more information, see "Get support for WebSphere Application Server for
  iSeries" on page 3.

## Troubleshoot: Enterprise applications

Use the resources listed below to determine the cause of problems that occur when running your
application code within WebSphere Application Server.

- Check "Troubleshooting guide for the J2EE application client" on page 11
  This topic describes debugging tips for resolving several J2EE application client problems.

- Check the application server standard output and standard error log files for errors. For information on
  the application server log files and where they are located, see "WebSphere Application Server log
  files" on page 23.

- Add logging and tracing to your application using the JRas framework. See Add logging and tracing in
  the *Application Development* topic for more information.

- Add output to your enterprise bean or other Java code in the form of System.out.println() statements
  to aid in debugging. The output from any System.out.println() statements is written to the standard

output log file for the application server in which the code is running. For information on the application server log files and where they are located, see "WebSphere Application Server log files" on page 23.
- Use one of the debugger tools available for debugging WebSphere Application Server for iSeries to step into the Java code in which the error is occurring. For more information, see the Debugger tools topic in *Programming*.
- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) .
- Check the WebSphere Application Server FAQ database .
- Refer to the WebSphere Application Server for iSeries newsgroup .
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.
- Contact IBM support. For more information, see "Get support for WebSphere Application Server for iSeries" on page 3.

## Troubleshooting guide for the J2EE application client

This section provides debugging tips for resolving several J2EE application client problems. To use this troubleshooting guide, review the trace entries for one of the J2EE application client exceptions, and then locate the exception in the guide.

Specify this command-line arguments for the launchClient script to generate a trace and to redirect the trace output to a file:

```
launchClient -CCtrace=true -CCtracefile=appclient.trc
```

In the example above, the resulting trace output is written to a file called appclient.trc in the current working directory.

Here are some possible errors:

**Error:** `java.lang.NoClassDefFoundError`

**Explanation:** This exception is thrown when Java cannot load the specified class.

**Possible causes:**
- Invalid or non-existent class
- Classpath problem
- Manifest problem

**User response:**

First check to determine if the specified class exists in a jar file within your ear file. If it does, make sure the package for the class is correct. For example, if you get the exception:

```
java.lang.NoClassDefFoundError: WebSphereSamples.HelloEJB.HelloHome
```

ensure the class HelloHome exists in one of the jar files in your ear file. If it exists, ensure the package for the class is WebSphereSamples.HelloEJB.

If both the class and package are correct, then it is most likely a classpath issue. If this is the case, you do not have the failing jar file of the class specified in the client jar file's manifest. To verify this, open your Enterprise Archive (EAR) file with the Application Assembly Tool and click on the Application Client. Add the names of the other jar files in the ear file to the Classpath field. This exception is generally caused by a missing EJB module name from the Classpath field.

**Note:**If you have multiple jars to enter in the Classpath field, be sure to separate the jar names with spaces.

If you still have the problem, you have a situation where a class is being loaded outside of the ear file. This is a very difficult situation to debug because the offending class is not the one specified in the exception. Instead, another class is loaded outside of the EAR file before the one specified in the exception. To correct this, review the classpaths specified with the `-CCclasspath` option and the classpaths configured with the Application Client Resource Configuration Tool. Look for classes that also exist in the EAR file. You must resolve the situation where one of the classes is found outside of the EAR file instead of the version contained in the EAR file. You do this by removing entries from the classpaths or by including the `.jar` files and classes in the EAR file instead of referencing them from outside of the EAR file.

If you are using the `-CCclasspath` parameter or resource classpaths in the Application Client Resource Configuration Tool, and you have configured multiple jars or classes, verify they are separated with the correct character for your operating system. Unlike the classpath field in the Application Assembly Tool, these classpath fields use platform-specific separator characters, usually a colon (on iSeries and UNIX platforms) or a semi-colon (on Windows 32-bit platforms).

**Note:** The system classpath is not used by the Application Client runtime if you use the launchClient script. In this case, the system classpath would not cause this problem. However, if you load the launchClient class directly, you do have to search through the system classpath as well.

**Error:** `com.ibm.websphere.naming.CannotInstantiateObjectException: Exception occurred while attempting to get an instance of the object for the specified reference object. [Root exception is javax.naming.NameNotFoundException: xxxxxxxxxx]`

**Explanation:** This exception occurs when you perform a lookup on an object that is not installed on the server. Your program looks up the name in the local client Java Naming and Directory Interface (JNDI) name space but receives a NameNotFoundException exception because the object is not located on the server. Looking up an enterprise bean that is not installed on the server is a typical example. This exception might also occur if the JNDI name you configured in your Application Client module does not match the actual JNDI name of the server-side resource.

**Possible causes:**
- Incorrect host server invoked
- Resource is not defined
- Resource is not installed
- Application server is not started
- Invalid JNDI configuration

**User response:** If you are accessing the wrong server, run the launchClient script again with the `-CCBootstrapHost` and optionally the `-CCBootstrapPort` parameters specifying the correct server name and port number. If you are accessing the correct server and port number, use the dumpName Space utility to see a listing of the host-server's JNDI name space. For more information on the dumpName Space utility, see The dumpNameSpace script. If you do not see the failing object's name, either the resource is not installed on the server or the appropriate application server is not started. If you determine the resource is already installed and started, your JNDI name in your client application does not match the global JNDI name on the server. Use the Application Assembly Tool to compare the JNDI bindings of the client-side object's name to the JNDI bindings of the object in the server application. The bindings must match.

**Error:** `javax.naming.ServiceUnavailableException: Caught exception when resolving initial reference=NameService. Root exception is org.omg.CORBA.INTERNAL: JORB00105: In Profile.getIPAddress(), InetAddress.getByName( invalidhostname ) threw an UnknownHostException minor code: 0 completed: No`

**Explanation:** This exception occurs when you specify an invalid host server name or port number.

**Possible causes:**
- Incorrect host server invoked
- Invalid host server name
- Invalid port number

**User response:** Run the launchClient script again and specify the correct name of your host server with `-CCBootstrapHost` and, optionally, `-CCBootstrapPort`.

**Error:**

`javax.naming.CommunicationException: Caught CORBA.COMM_FAILURE when resolving initial reference=WsnNameService. Root exception is org.omg.CORBA.COMM_FAILURE: minor code: 3 completed: No`

**Explanation:** This exception occurs when you run the launchClient script to a server that does not have the Application Server started. You also receive this exception when you specify an invalid server name or invalid port number. This might happen if you do not specify a server name when you run the launchClient script. The default behavior is for the launchClient script to run to `localhost`, because WebSphere Application Server does not know the name of your server.

**Possible causes:**
- Incorrect host server invoked
- Invalid host server name
- Invalid reference to `localhost`
- Invalid port number
- Application server is not started

**User Response:** If you are not running to the correct server, run the launchClient script again and specify the server name with `-CCBootstrapHost`. Also, ensure that you have specified the `-CCBootstrapPort` parameter if neccessary.

Otherwise, start the Application Server on the server and run the launchClient script again.

**Error:** `javax.naming.NameNotFoundException: Name comp/env/ejb not found in context "java:"`

**Explanation:** This exception is thrown when Java cannot locate the specified name in the local JNDI name space.

**Possible causes:**
- No binding information for the specified name
- Binding information for the specified name is incorrect
- Wrong class loader was used to load one of the program's classes

**Explanation:** Open the ear file with the Application Assembly Tool and check the bindings for the failing name. Ensure this information is correct. If it is correct, you could have a class loader problem.

**Error:** `java.rmi.RemoteException: CORBA UNKNOWN 0 Maybe Nested exception is:`
`org.omg.CORBA.UNKNOWN: minor code: 0 completed: Maybe`

**Explanation:**This exception occurs when your server-side application component has a problem. For example, this exception displays when your J2EE Application client invokes an EJB method that has thrown an exception.

**Possible causes:**

- Exception is occuring in your server-side application component

**User Response:** Check the stdout.log and stderr.log files for your application server. For the default application server, these files are Default_Server_stdout.log and Default_Server_stderr.log and are located in the /QIBM/UserData/WebAS5/default/logs/ directory on your iSeries server. The log files should contain messages describing the cause of the error.

**Error:** `java.lang.ClassCastException: Unable to load class:`
`org.omg.stub.WebSphereSamples.HelloEJB._HelloHome_Stub at`
`com.ibm.rmi.javax.rmi.PortableRemoteObject.narrow(portableRemoteObject.java:269)`

**Explanation:** This exception occurs when the application program attempts to narrow to the EJB's home class and the classloaders cannot find the EJB's client side bindings.

**Possible causes:**

- The files, `*_Stub.class` and `_Tie.class`, are not in the EJB .jar file
- Classloader could not find the classes

**User Response:** Look at the EJB `.jar` file located in the `.ear` and verify the class contains the EJB client side bindings. These are class files whose names end in `_Stub` and `_Tie`. If these files are not present, then use the Application Assembly Tool to generate the binding classes. If the binding classes are in the EJB `.jar` file, then you might have a classloader issue.

## Troubleshoot: Universal Description, Discovery, and Integration (UDDI), Web Service, and SOAP

This section provides debugging tips for resolving several Web services problems. If you do not see your problem, use the resources below to help you determine what might be causing problems when you deploy or run applications which use Web Services, UDDI or SOAP:

- A Java API XML-based remote procedure call Simple Object Access Protocol handler is unable to access EJB container resources.
  **Symptom:**
  A Java API XML-based remote procedure call (JAXRPC) Simple Object Access Protocol (SOAP) handler deployed on the server-side to an EJB-based Web service is unable to access the resources of the EJB container. A JAXRPC SOAP handler does not execute in the EJB container.
  **Problem:**
  A JAXRPC SOAP handler executes in the Java 2 Platform, Enterprise Edition (J2EE) 1.3 Web container of a router-module. The router module directs the SOAP request to the target enterprise bean. The environment of the router module container is visible to the handler, while the environment of the EJB container is not visible. This lack of visibility might cause unexpected and possibly misleading results to handlers accessing the container environment.
  **Response:**
  J2EE 1.3 does not require the execution of Web services JAXRPC SOAP handlers in an EJB container. J2EE 1.4 requires handlers to execute in the context of the target EJB container.
- Check the WebSphere Application Server log files for errors. For information on the application server log files and where they are located, see "WebSphere Application Server log files" on page 23.

- Check the HTTP server log files for errors. For more information, see "Check HTTP server logs for troubleshooting information" on page 24.
- Read the release notes. WebSphere Application Server Release Notes

  (http://publib.boulder.ibm.com/was400/docs/relnotes502.html) .
- Check the WebSphere Application Server FAQ database .
- Refer to the WebSphere Application Server for iSeries newsgroup
  This iSeries Technical Support Web-based forum is dedicated to WebSphere Application Server for iSeries.
- Contact IBM support. See "Get support for WebSphere Application Server for iSeries" on page 3 for more information.

## Resources for monitoring the WebSphere Application Server environment

WebSphere Application Server has several resources for monitoring your WebSphere Application Server environment. These topics provide information on the different methods you can use to monitor your WebSphere Application Server environment.

**"Monitor WebSphere Application Server jobs on the iSeries server"**
This topic describes the jobs that make up the WebSphere Application Server runtime and how you can use iSeries commands to monitor them.

**"Monitor WebSphere Application Server jobs with a message queue" on page 19**
This topic describes how you can use a message queue to monitor the jobs that make up the WebSphere Application Server runtime. The same messages that are sent to to the joblog for the WebSphere Application Server jobs can also be sent to a user-specified message queue.

**"Monitor the WebSphere Application Server environment by using the Log Analyzer" on page 20**
This topic describes the Log Analyzer tool and how you can use it to monitor your WebSphere Application Server environment.

**Monitor application server resources with the Tivoli Performance Viewer**
This topic describes how to get and use the Tivoli Performance Viewer. The Tivoli Performance Viewer lets you track resources, including servlets, enterprise beans, sessions, database pools, Java$^{(TM)}$ virtual machine memory, and thread pools.

**"Verify product prerequisites" on page 23**
This topic describes how you can use the prerequisite validator to verify whether you have all the required product prerequisites.

**"Check for port conflicts" on page 23**
This topic describes how you can use the port validator to check for port conflicts.

## Monitor WebSphere Application Server jobs on the iSeries server

The WebSphere Application Server for iSeries product is shipped configured to use a single WebSphere instance, the default instance. However, multiple instances are also supported. For more information, see the Create a new instance topic in *Administration*. This page describes the jobs used by a WebSphere instance and how you can monitor the jobs by using iSeries command language (CL) commands.

**WebSphere Application Server jobs**

Each WebSphere Application Server instance consists of one or more jobs. These jobs run in the QEJBAS5 subsystem:

- A single application server job
  The job name is the first 10 characters of the application server name . If the first 10 characters do not provide a valid iSeries job name, the WebSphere Application Server runtime creates a valid job name. If the runtime cannot create a valid job name for the application server, the application server does not start. The default instance is configured with a single application server, server1, whose job name is *SERVER1.*

  The WebSphere Application Server provides the run-time environment for server-side Java(TM) components (such as servlets, JavaServer Pages(TM) (JSP(TM)), and enterprise beans). The application server plug-in interfaces with the Web server to handle client requests for server-side resources and route them to the application server for processing.

  If the instance is not part of a Network Deployment domain, and the instance is not using embedded JMS, this is the only job started for an instance.

- An MQ listener job
  If the instance is configured to use the embedded JMS server, this job is used by the embedded JMS provider to establish connections to the embedded JMS server. 17 The job name is *QEJBMQLSR.* If you are using multiple instances of WebSphere Application Server with embedded JMS enabled, you see a *QEJBMQLSR* job for each instance.

  To determine which QEJBMQLSR job is in use by your application server, you can display the joblog for your application server job to see the job information for the QEJBMQLSR job the application server is using. Use the **F10** key when displaying the joblog to view all messages. There are several messages indicating the WebSphere MQ jobs that are started and ended before the QEJBMQLSR job is started. The following message is displayed when the WebSphere MQ listener job is started for your application server:

  ```
  Job 053579/QEJBSVR/QEJBMQLSR started.
  ```

  Alternatively, you can display the joblog for each QEJBMQLSR job. The joblog contains message identifier AMQ7163 which indicates the queue manager name for the listener. The queue manager name contains the name of the application server for which it is listening. For example, the QEJBMQLSR job for the default instance on iSeries server MYISERIES would contain message AMQ7163 with the following information:

  ```
  WebSphere MQ job 7127 started for WAS_MYISERIES_server1.
  ```

  The last string in the message indicates the node name and server name of the application server.

- A node agent job
  The node agent job exists only if the instance is managed by a Network Deployment Deployment Manager (in other words, it is part of a Network Deployment domain). The job name for the node agent process is *NODEAGENT.*If you are using multiple instances of WebSphere Application Server and the instances have been added to a Network Deployment cell, you see one *NODAGENT* job for each instance.

  If you are using multiple WebSphere Application Server instances which have been added to a Network Deployment cell, you can determine which NODEAGENT job is for your instance as follows:

  1. Display the joblog for the NODEAGENT job
  2. The first message in the joblog contains the instance directory path and the name of the server.

     ```
     >> CALL PGM(QEJBAS5/QEJBSTRSVR) PARM('-instance' '/QIBM/UserData/WebAS5/Base/
     myinstance' '-server' 'nodeagent')
     ```

     The last part of the value specified for the `-instance` parameter indicates the instance for which the nodeagent is running.

Each WebSphere Application Server Network Deployment instance consists of one job. This job runs in the QEJBASND5 subsystem:

- A Deployment Manager server job
  If the instance is a Network Deployment instance, there is a single job for the deployment manager.

The job name is the first 10 characters of the application server name for the Deployment Manager. If the first 10 characters do not provide a valid iSeries job name, the WebSphere Application Server runtime creates a valid job name. If the runtime cannot create a valid job name for the application server, it uses the default job name QEJBSVR. The Network Deployment product is shipped configured to use the default instance. The job name for the specialized application server for the default Deployment Manager instance is *DMGR*.

**Other jobs**

In addition to the jobs that run in the QEJBAS5 or QEJBASND5 subsystems, WebSphere instances use other jobs:

- If the instance is configured to use the embedded JMS server, these jobs are started in the QMQM subsystem when your application server or node agent server is started:

      AMQALMPX
      AMQPCSEA
      AMQRMPPA
      AMQRRMFA
      AMQZDMAA
      AMQZLAA0
      AMQZXMA0
      RUNMQCHI

  The joblog for each of these jobs contains a message indicating the queue manager name for the listener. The queue manager name contains the name of the application server for which it is listening. For example, the queue manager name for the default instance on iSeries server MYISERIES is:

  WAS_MYISERIES_server1.

  The last two parts of the queue manager name indicate the node name and server name of the application serverrespectivelyy.

  These jobs represent the embedded JMS server and are used to process JMS messages. All of these jobs run in the QMQM subsystem. When the application server job is ended, these jobs are ended also.

- If you are using the IBM Developer Kit for Java JDBC driver for database access, your instance uses one or more QSQSRVR jobs that run in the QSYSWRK subsystem. To determine what QSQSRVR job your application server job is using, view the joblog for the server job you are interested in. For each JDBC connection obtained, you see message SQL7908 with message text similar to "Job 163707/QUSER/QSQSRVR used for SQL server mode processing.".

- If you are using the IBM Toolbox for Java JDBC driver for database access, your instance uses one or more QZDASOINIT jobs that run in subsystem QUSRWRK.

- Depending on which Web server you are using to serve Web resources, your instance uses jobs from the Web server instance:

  – IBM HTTP Server (powered by Apache)
    Each IBM HTTP Server (powered by Apache) instance consists of two or more jobs that run in the QHTTPSVR subsystem. The name of each job is the same as the name of your HTTP server instance. The WebSphere Application Server Web server plugin code runs in the second job listed (via WRKACTJOB SBS(QHTTPSVR)) for your HTTP server instance.

  – Domino Web Server
    Each Domino Web Server instance has a corresponding subsystem in which the jobs for the instance run. The subsystem for the first Web server instance created is DOMINO001, the subsystem for the second web server instance created is DOMINO002, and so on. The WebSphere Application Server Web server plugin code runs in the job named HTTP in the subsystem in which your Web server instance is running.

**OS/400 command language (CL) commands for monitoring jobs**

Use iSeries command language (CL) commands to monitor jobs that run in your WebSphere Application Server environment. You can view job logs, display message details, and view spooled files associated with the jobs. The job logs and associated spooled files for the jobs can contain valuable information for determining the root of a problem.

**Note:** For more information on iSeries work management CL commands, see OS/400 Work Management V5R1 (SC41-5306-03) or iSeries Work Management V5R2.

This table lists some iSeries commands that you can use to monitor WebSphere Application Server jobs:

| Task | iSeries command and description |
|---|---|
| Work with active WebSphere Server jobs | `WRKACTJOB SBS(QEJBAS5)`<br><br>For WebSphere Application Server V5.0 jobs that run in the QEJBAS5 subsystem.<br>`WRKACTJOB SBS(QEJBASND5)`<br><br>For WebSphere Application Server V5.0 jobs that run in the QEJBASND5 subsystem. |
| Work with all jobs with a specific name | `WRKJOB JOB(`*job_name*`)`<br><br>This command lists all jobs, active or not, named *job_name*. The jobs are listed by date, most recent first.<br><br>This command is very useful when a WebSphere Application Server job ends abnormally or fails to start successfully, and you wish to view the joblog. For instance, if the application server job is starting and then ending immediately, you could do this to view its joblog:<br>• Run the command WRKJOB JOB(SERVER1).<br>• Specify option **1** on the option line next to the first job listed that has a status of OUTQ.<br>• Specify option **4** (Work with spooled files) on the command line of the Work with Job screen.<br>• Specify option **5** next to the QPJOBLOG spooled file to view the contents. |
| Work with a specific job | `WRKJOB JOB(`*job_number/job_user/job_name*`)`<br><br>This command displays the Work with Job screen for the specified job.<br><br>This command is very useful when you know the fully qualified job information for the job. |

| Task | iSeries command and description |
|---|---|
| View the job log for an active WebSphere server job | WRKACTJOB SBS(QEJBAS5)<br><br>or<br><br>WRKACTJOB SBS(QEJBASND5)<br><br>1. Run one of the commands above.<br>2. Enter option **5** (Work with) in the option line next to the active job whose job log you want to view.<br>3. In the Work with Job display, enter option **10** (Display job log, if active or on job queue).<br>4. Press **F10** to display all messages.<br>5. In the job log, position the cursor on a message for which you want to display extended message information.<br>6. Press **F1** (Help).<br><br>Similarly, you can view ended jobs by using the Work with User Jobs (WRKUSRJOB) or Work with Job (WRKJOB) command. |
| Work with WebSphere server jobs that run under the QEJBSVR user profile | WRKUSRJOB USER(QEJBSVR) |
| Delete spooled files for QEJBSVR user profile | DLTSPLF FILE(*SELECT) SELECT(QEJBSVR) |
| View the spooled files for a job | WRKACTJOB SBS(QEJBAS5)<br><br>or<br><br>WRKACTJOB SBS(QEJBASND5)<br><br>1. Run one of the commands above.<br>2. Enter option **8** (Work with Spooled Files) in the option line next to the active job whose spooled files you want to view.<br>3. Enter option **5** (Display) in the option line next to the spooled file you want to view.<br><br>Similarly, you can view ended jobs by using the Work with User Jobs (WRKUSRJOB) or Work with Job (WRKJOB) command. |

In addition to sending product messages to the joblogs of WebSphere Application Server jobs, the WebSphere Application Server code can also send the messages to a specified message queue. For more information, see "Monitor WebSphere Application Server jobs with a message queue."

## Monitor WebSphere Application Server jobs with a message queue

WebSphere Application Server for iSeries allows you to specify an iSeries message queue object to which product messages are sent. The product messages are the same WebSphere Application Server messages that are sent to the joblog of the application server job.

To enable the message queue support for an application server, use the administrative console to specify the os400.websphere.message.queue system property for your server. The message queue is specified by using the Integrated File System pathname of the object. For message queues in library QSYS, the format is /QSYS.LIB/*messageQueue*.MSGQ. For message queues in libraries other than QSYS, the format is /QSYS.LIB/*yourLib*.LIB/*messageQueue*.MSGQ. The message queue must exist and the QEJBSVR user profile must have *CHANGE authority to the message queue.

Follow the steps below to specify the os400.websphere.message.queue system property for your application server:

1. Start the administrative console.
2. Expand **Servers** and click **Application Servers**.
3. Click on the link for the application server you wish to change.
4. Scroll down in the right-hand frame and click **Process Definition**.
5. Under the *Additional Properties* section, click **Java Virtual Machine**.
6. Scroll down and click **Custom Properties** under the *Additional Properties* section in the right-hand frame.
7. Click **New**.
8. Enter os400.websphere.message.queue in the **Name** field and enter the integrated file system pathname for the message queue in the Value field. For example, you would specify /QSYS.LIB/QSYSOPR.MSGQ to use the QSYSOPR message queue.
9. Click **Ok**.
10. Click **Save** at the top of the right-hand frame to save the configuration change.
11. On the *Save* page, click **Save** to save the changes.

**Note:** You can enable the message queue support for any WebSphere Application Server server such as a Network Deployment Deployment Manager, a node agent, or a JMS server. Simply use the administrative console to locate the server and then follow steps 3 through 11 above.

## Monitor the WebSphere Application Server environment by using the Log Analyzer

The Log Analyzer merges all the data from one or more IBM Service logs and displays the entries. For more information on the IBM Service logs, see "IBM Service log file" on page 30. Based on its symptom database, it analyzes and interprets the error conditions in the log entries to help you debug problems. The Log Analyzer can download the latest symptom database from the IBM Web site. For more information on obtaining and using the Log Analyzer to monitor your WebSphere Application Server environment, see these topics:

- Obtain the Log Analyzer (page 20)
- Use the Log Analyzer (page 20)
- Update the symptom database (page 21)
- View the IBM Service log file without the Log Analyzer (page 21)

**Obtaining the Log Analyzer**

The Log Analyzer is installed on your workstation as part of the WebSphere Application Server workstation components. For details on how to install the workstation components, see the Install the workstation tools for WebSphere Application Server topic in *Installation*.

**Use the Log Analyzer**

You can start the Log Analyzer on your client workstation by using any of these methods:

- From the Windows NT/2000 task bar, select **Start** -> **Programs** -> **IBM WebSphere** -> **Application Server Version 5.0** -> **Log Analyzer**.
- From the workstation command prompt:
  - On Windows NT or Windows 2000, enter this:
    `[WAS_INSTALL_ROOT]\bin\waslogbr.bat`
  - On Unix platforms, enter this:
    `[WAS_INSTALL_ROOT]/bin/waslogbr.sh`

Once the Log Analyzer window is displayed, perform these steps to open an activity log file:

1. Do either of these steps:
   - Use ftp to send the activity.log file in binary format to your log analyzer workstation.
   - On Windows NT or Windows 2000, map a drive to the iSeries server. Alternatively, on Unix platforms, mount a drive to the iSeries server.
2. Select **File** -> **Open**.
3. Navigate to the directory containing the activity.log file.
4. Select the activity.log file.
5. Select **Open**.
6. Expand the tree in the left-hand pane of the Log Analyzer to view messages.

Uncolored records are "normal", yellow are warnings, and pink are errors. If you select a record, you see its contents, including the basic error or warning message, the date, the time, which WebSphere component logged the record, and which process (for example: application server, node agent) it came from, in the right-hand pane.

The activity log does not analyze any other log files, such as the SystemOut.log or native_stdout.log file. To analyze the records, right click on a record in the tree on the left (click on the **UnitOfWorkView** at the top to get them all), and select **Analyze**. Any records with a green check mark next to them match a record in the symptom database. When you select a check-marked record, you see an explanation of the problem in the lower right-hand pane.

Use the Log Analyzer **Help** to get more detailed information on using Log Analyzer.

**Update the symptom database**

The database of known problems and resolutions used when you click **Analyze** on the menu is periodically enhanced as new problems come to light and new versions of the product are introduced. To ensure that you have the latest version of the database, use the **File —> Update Database —> WebSphere Application Server Symptom Database** menu item from within the log analyzer tool at least once a month. Users who have just installed the product and have never run the update should do so immediately, since updates may have occurred since the version was first released.

The knowledge base used for analysis is built gradually from problems reported by users. For a recently released version of the product, you might not find any analysis hits. However, the Log Analyzer tool still provides a way to see all error messages and warnings from all components in a convenient, user-friendly way.

**View the IBM Service log without the Log Analyzer.**

If using the Log Analyzer to view the IBM Service log file is impractical or inconvenient, you can use an alternative tool named showlog. To use the showlog tool, perform these steps:

1. On the iSeries command line, run the Start Qshell Interpreter (STRQSH) command.
2. Change directory to the bin directory for the product:

   ```
   cd was_install_root/bin
   ```

   where *was_install_root* is /QIBM/ProdData/WebAS5/Base or /QIBM/ProdData/WebAS5/ND for WebSphere Application Server and WebSphere Application Server Network Deployment respectively.
3. Run the showlog script. The syntax is:

   ```
   showlog [-instance instance_name] activity_log_file
           [output_file]
   ```

   where the parameters are:

**-instance**

Optional. The value *instance_name* specifies the name of the WebSphere Application Server instance. The default value for instance_name is default.

*activity_log_file*

Required. The qualified integrated file system path name of the activity log file you wish to view. For example, /QIBM/UserData/WebAS5/Base/default/logs/activity.log

*output_file*

Optional. The qualified integrated file system path name of the file to contain the formatted, readable contents of the activity log file. If this parameter is not specified, the output is written to standard out (the screen).

Examples:

```
showlog /QIBM/UserData/WebAS5/Base/default/logs/activity.log
showlog -instance myinst /QIBM/UserData/WebAS5/ND/myinst/logs/activity.log
/home/myuserid/myinst_activity.txt
```

The output of the showlog script file consists of message event entries separated by horizontal lines.

Here is a description of the fields that make up the entries:

| | |
|---|---|
| **ComponentId** | Currently not used. |
| **ProcessId** | Job information for the job from which the message event was issued. |
| **ThreadId** | Java $^{(TM)}$ thread ID of the thread from which the message event was issued. |
| **FunctionName** | Currently not used. |
| **ProbeId** | Currently not used. |
| **SourceId** | The class name of the object from which the message event was issued. |
| **Manufacturer** | IBM |
| **Product** | Product whose code issued the message event. Currently this is WebSphere Application Server. |
| **Version** | Product version or build level |
| **SOMProcessType** | Currently not used |
| **ServerName** | |
| **ClientHostName** | Currently not used. |
| **ClientUserId** | Currently not used. |
| **TimeStamp** | Timestamp indicating when the message event was issued. The default time zone is Greenwich Mean Time (GMT). See Set the time zone for information on how to change the default timezone for WebSphere Application Server. |
| **UnitOfWork** | Currently not used. |
| **Severity** | Indicates the severity of the message. Audit message events are severity 3, warning message events are severity 2, and fatal/terminate/error message events are severity 1. |
| **Category** | The type of message event. Valid values are AUDIT, WARNING, ERROR, FATAL, and TERMINATE |
| **FormatWarning** | Currently not used. |
| **PrimaryMessage** | Currently not used. |
| **ExtendedMessage** | Message Identifier: Message text |
| **RawDataLen** | The length of any raw data that follows the message. Typically this value is zero. |

## Verify product prerequisites

WebSphere Application Server may not run properly if all of the required software prerequisites are not installed. If the server does not start, run the prerequisite validator. It verifies your installation, ensures that you have the appropriate prerequisite software and PTFs installed, and verifies configuration settings required by WebSphere Application Server.

To run the prerequisite validator, run the checkprereqs script from the Qshell command line.

1. On the OS/400 command line, enter the STRQSH (Start Qshell) command.

2. On the Qshell command line, use the cd command to change to the directory that contains the script:

   `cd /QIBM/ProdData/WebAS5/`*`edition`*`/bin`

   where *edition* is Base for WebSphere Application Server and ND for WebSphere Application Server Network Deployment.

3. Run the checkprereqs script:

   `checkprereqs`

   **Note:** The prerequisite validator can also be run as a parameter of the servicetools script. For more information and options for this script, see The checkprereqs script in the *Administration* topic.

## Check for port conflicts

WebSphere Application Server may not run properly if you have port conflicts. For example, your server may not start. The port validator verifies your configuration to ensure that you do not have port conflicts across WebSphere Application Server instances and products. It detects port conflicts between:

- Instances in a Version 4 product or a Version 5 product.

- Servers in a Version 4 product or a Version 5 product.

- Instances in different Version 4 and Version 5 products (Version 4 Advanced Edition, Version 4 Advanced Edition Single Server, Version 5 Base, Version 5 Network Deployment, and Version 5 Express).

To run the port validator, run the servicetools script from the Qshell command line and specify the -portconflict parameter.

1. On the OS/400 command line, enter the STRQSH (Start Qshell) command.

2. On the Qshell command line, use the cd command to change to the directory that contains the script:

   `cd /QIBM/ProdData/WebAS5/`*`edition`*`/bin`

   where *edition* is Base for WebSphere Application Server and ND for WebSphere Application Server Network Deployment.

3. Run the servicetools script to check for port conflicts:

   `servicetools -portconflict -products all`

   For more information, see The port validator in the *Administration* topic.

## WebSphere Application Server log files

WebSphere Application Server has a variety of logs to which messages are written. For example, system messages, which can be written by any application server component or application are written to general-purposes logs such as the JVM logs and the IBM service, or activity, log. Other logs are very specific in nature and are scoped to a particular component or activity. For example, the HTTP server plugin maintains a component-specific plugin log.

In general, the general purpose logs such as the JVM logs and the IBM service, or activity, log are used to monitor the health of the application server and are used in most problem determination procedures. However, the problem determination procedure for a specific component may direct you to examine the contents of a component or product specific log.

The topics below describe the log files available for WebSphere Application Server, and how you can configure and view the files.

**"JVM log files" on page 27**
This topic describes the *JVM* log files. All WebSphere Application Server message events go to these files as well as System.out and System.err print statements for your applications. This topic also provides information on how to configure and view the JVM log files.

**"Process log files" on page 25**
This topic describes the *process* log files. Any output for the native code supporting the java virtual machine or native code for an application is written to these files. This topic also provides information on how to configure and view the process log files.

**"IBM Service log file" on page 30**
This topic describes the *IBM service*, or *activity.log*, file. The IBM Service log file is a binary file to which WebSphere Application Server writes the message events for any servers running under an instance (node). This topic also provides information on how to configure and view the IBM Service log file.

**"Check HTTP server logs for troubleshooting information"**
This topic describes the plug-in log. The plug-in log records HTTP server-side processing and servlet request routing between the HTTP server and WebSphere Application Server.

# Check HTTP server logs for troubleshooting information

For help in troubleshooting a servlet problem, there are several log and trace files available for you to inspect. For a description of the files, see these topics.

**IBM HTTP Server for iSeries (powered by Apache) trace log**

The IBM HTTP Server for iSeries (powered by Apache) generates a trace file. You must start the HTTP server in a mode that outputs trace information, using option -vv. For example:
```
STRTCPSVR SERVER(*HTTP) HTTPSVR(serverInstanceName '-vv')
```

where *serverInstanceName* is the name of your HTTP server instance.

The spooled file for the HTTP server job is named QPZHBTRC.

To gather trace data from an IBM HTTP Server (powered by Apache) use the DMPUSRTRC command. See http://publib.boulder.ibm.com/iseries/v5r1/ic2924/index.htm?info/rzaie/rzaietrace.htm for information.

You can find additional log information by enabling the IBM HTTP Server for iSeries (powered by Apache) error log. This log records client request errors, such as timing out or access denial. To enable logging, perform these steps:
1. Make sure the HTTP Server ADMIN instance is running.
   a. In the WRKACTJOB display, look for the ADMIN job under the QHTTPSVR subsystem.
   b. If the job is not displayed, start the ADMIN server with this command:
      ```
      STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)
      ```
2. Enter this URL in a JavaScript-enabled browser:

```
http://your.host.name:2001
```

where *your.host.name* is the domain name of your HTTP server.

3. Click on **IBM HTTP Server for iSeries (AS/400)**.

4. Select the **Manage** tab.

5. In the expanded Server menu, select your HTTP server instance's configuration name from the pull-down menu.

6. In the left-hand pane, ensure **Server Properties** is expanded.

7. Under **Server Properties**, click on **Logging**. You may have to scroll down to find the link.

8. Select the **Error Logs** tab.

9. In the main window, set up the error log. (The help text explains the meaning of the values in the page. To view the help text, click the **?** icon.) Click **Apply**.

10. Stop your HTTP server instance and restart it.

The HTTP server changes the name you specify for the log file. For example, if you name the file myerror.log, the HTTP server generates the file with the name myerror.log.Q*yyymmdd*, where *yyy* is the last three digits of the year, *mm* is the month, and *dd* is the day of the month (for example, myerror.log.Q0991005). To view the log file, you can use the iSeries Edit File Utility (EDTF).

**Web server plug-in log**

The plug-in log records HTTP server-side processing and servlet request routing between the HTTP server and WebSphere Application Server. (The plug-in connects the HTTP server and WebSphere Application Server together.)

Manage this log by editing the application server's plugin-cfg.xml file. (For the default application server, you can find this file in the /QIBM/UserData/WebAS5/Base/default/config/cells directory.) The plugin-cfg.xml file contains a tag near the beginning named Log. Log has two attributes: LogLevel and Name.

* LogLevel specifies the amount and type of information that is logged to a file. Valid values are Trace, Warn, and Error.
* Name specifies the location and name of the file where logging information is written.

**Standard out and standard error**

For information about standard out and standard error, see "JVM log files" on page 27.

**HTTP Server joblogs**

You should inspect the joblog for your HTTP server instance for unexpected messages or exceptions. You can locate the IBM HTTP Server (powered by Apache) job within the QHTTPSVR subsystem, listed with a job name matching that of its instance. The Domino HTTP server job is named **HTTP** within a subsystem named DOMINO*xx*.

# Process log files

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 the JVM, may write data to these process streams.

By default, the stdout and stderr streams are redirected to log files at application server startup, which contain text written to the stdout and stderr streams by native modules (for example, *SRVPGM objects). The WebSphere Application Server does no special processing or formatting of the output that is written to the process logs.

This is a change from previous versions of WebSphere, which by default had one log file for both JVM standard output and native standard output, and one log file for both JVM standard error and native error output.

The granted authorities for the files are:

| | |
|---|---|
| *PUBLIC | *EXCLUDE |
| QEJBSVR | *RW |

If your application server is running under a user profile other than the default (QEJBSVR) and that user profile does not have QEJBSVR specified as a group profile, you must explicitly grant *RW authority to the user profile for the activity.log file.

Refer to the topics below for more information on configuring and viewing the process log files.

**"Configure the process log files"**
This topic describes how you can use the administrative console to change the name and location for the process logs.

**"View the process log files"**
This topic describes the methods available for viewing the contents of the process logs.

## Configure the process log files

Use the administrative console to configure the process log files for an application server. The file name for a process log is the only attribute that can be changed. Unlike the JVM logs, the process logs are not self-managing so the size and number of historical files cannot be configured. Generally, these file are empty or contain very small amounts of information, so self-management is not required.

Configuration changes for the process logs that are made to a running application server are not applied until the next restart of the application server.

Perform these steps to configure the process logs:
1. Start the administrative console.
2. Expand **Troubleshooting** and click **Logs and Trace**.
3. Click the link for the server you wish to configure.
4. Click **Process Logs**.
5. Select the **Configuration** tab.
6. The file name for each of the process streams (stdout and stderr) is displayed.
7. Change the file names as appropriate.
8. Click **Apply**.
9. Click the **Save** link at the top of the page to save your configuration changes. Click **Save** on the resulting page.

## View the process log files

The process logs are written as plain ASCII text files. By default, the process logs are located in the logs/*servername* subdirectory of the WebSphere instance you are using, where *servername* is the name of the server. If you are using the default WebSphere Application Server instance, the path is /QIBM/UserData/WebAS5/Base/default/logs/server1. If you are using the default Network Deployment instance, the path is /QIBM/UserData/WebAS5/ND/default/logs/dmgr. For a WebSphere Application Server instance that has been added to a Network Deployment domain (cell), the log files for the node agent are located in subdirectory logs/nodeagent and the logs files for the JMS server are located in subdirectory logs/jmsserver.

You can view the process log files using one of these methods:

- View the process logs from the administrative console.
  Perform these steps to view the process logs using the administrative console:
  1. Start the administrative console.
  2. Expand **Troubleshooting**.
  3. Click **Logs and Trace**.
  4. Click the link for the server whose logs you wish to view.
  5. Click **Process Logs**.
  6. Select the **Runtime** tab.
  7. Click **View** for the log you want to view.
- View the process logs on the iSeries where they are stored.
  To use the Edit File (EDTF) CL command to view the process log files:
  From the OS/400 command line, invoke the EDTF CL command specifying the Integrated File System pathname for the file you wish to view. For example:

  ```
  EDTF FILE('/QIBM/UserData/WebAS5/Base/default/logs/server1/native_stdout.log')
  EDTF FILE('/QIBM/UserData/WebAS5/Base/default/logs/server1/native_stderr.log')
  ```
- From a non-iSeries workstation.
  Perform these steps to view the process logs from a mapped or mounted driver:
  1. Map (Windows 32-bit workstation) or mount (Unix workstation) a drive to the iSeries.
  2. Open the file in a text editor or drag and drop the file into a file editing/viewing program.

## JVM log files

The JVM log files are one of the first places to start when troubleshooting a problem. These log files contain the output for the System.out and System.err output streams for the application server process. There is one log file specified for the System.out output stream and one file specified for the System.err output stream. The JVM logs contain print data written by applications. The data may be written directly by the application in the form of System.out.print(), System.err.print(), etc, method calls. Data may also be written indirectly by the application calling a JVM function, such as Exception.printStackTrace(). In addition, the System.out JVM log contains system messages (also known as message events) written by the WebSphere application server.

The JVM log files are self-managing in that you can configure the files to not grow beyond a certain size, and you can configure the number of historical, or archived, files to retain. In addition, you can configure the log files to rollover (be archived) based on time as well as size.

The granted authorities for the files are:

| | |
|---|---|
| *PUBLIC | *EXCLUDE |
| QEJBSVR | *RW |

If your application server is running under a user profile other than the default (QEJBSVR) and that user profile does not have QEJBSVR specified as a group profile, you must explicitly grant *RW authority to the user profile for the activity.log file.

Depending on how the JVM log is configured, application print data may be formatted to look like WebSphere system messages, or may be displayed as plain text with no additional formatting. WebSphere system messages are always formatted.

Refer to these topics for more information on configuring and using the JVM log files.

**"Configure the JVM log files"**
This topic describes how you can use the administrative console to change the name, size, number of files kept, and location for the JVM logs.

**"View the JVM log files"**
This topic describes the methods available for viewing the contents of the JVM logs.

**"Interpret the JVM log files" on page 29**
This topic describes the format of the information that is written to the JVM logs.

## Configure the JVM log files

Use the administrative console to configure the JVM logs for an application server. Configuration changes for the JVM logs that are made to a running application server are not applied until the next restart of the application server.

Perform these steps to configure the JVM logs:

1. Start the administrative console.
2. Expand **Troubleshooting** and click **Logs and Trace**.
3. Click the link for the server you wish to configure.
4. Click **JVM Logs**.
5. Select the **Configuration** tab.
6. Scroll through the panel to display the attributes for the stream to be configured.
7. Change the appropriate configuration attributes. See JVM log settings ❓ for a description of the attributes for the JVM logs.
8. Click **Apply**.
9. Click the **Save** link at the top of the page to save your configuration changes. Click **Save** on the resulting page.

## View the JVM log files

The JVM logs are written as plain ASCII text files. By default, the JVM logs are located in the logs/*servername* subdirectory of the WebSphere instance you are using, where *servername* is the name of the server. If you are using the default WebSphere Application Server instance, the path is /QIBM/UserData/WebAS5/Base/default/logs/server1. If you are using the default Network Deployment instance, the path is /QIBM/UserData/WebAS5/ND/default/logs/dmgr. For a WebSphere Application Server instance that has been added to a Network Deployment domain (cell), the log files for the node agent are located in subdirectory logs/nodeagent and the logs files for the JMS server are located in subdirectory logs/jmsserver.

You can view the JVM log files using one of the methods listed below. For information on interpreting the contents of the logs, see "Interpret the JVM log files" on page 29.

- View the JVM logs from the administrative console.
  Perform these steps to view the JVM logs using the administrative console:
  1. Start the administrative console.
  2. Expand **Troubleshooting**.
  3. Click **Logs and Trace**.
  4. Click the link for the server whose logs you wish to view.
  5. Click **JVM Logs**.
  6. Select the **Runtime** tab.
  7. Click **View** for the log you want to view.
- View the JVM logs on the iSeries where they are stored.
  To use the Edit File (EDTF) CL command to view the JVM log files:

From the OS/400 command line, invoke the EDTF CL command specifying the Integrated File System pathname for the file you wish to view. For example:

```
EDTF FILE('/QIBM/UserData/WebAS5/Base/default/logs/server1/SystemOut.log')
EDTF FILE('/QIBM/UserData/WebAS5/Base/default/logs/server1/SystemErr.log')
```

- From a non-iSeries workstation.
  Perform these steps to view the JVM logs from a mapped or mounted driver:
  1. Map (Windows 32-bit workstation) or mount (Unix workstation) a drive to the iSeries.
  2. Open the file in a text editor or drag and drop the file into a file editing/viewing program.

## Interpret the JVM log files

The JVM logs contain print data written by applications. The data may be written directly by the application in the form of System.out.print(), System.err.print(), etc, method calls. Data may also be written indirectly by the application calling a JVM function, such as Exception.printStackTrace(). In addition, the System.out JVM log contains system messages written by the WebSphere application server.

Depending on how the JVM log is configured, application print data may be formatted to look like WebSphere system messages, or may be displayed as plain text with no additional formatting. WebSphere system messages are always formatted.

Formatted messages may be written to the JVM logs in either basic or advanced format, depending on how the JVM log is configured.

This information describes the two formats and the fields that make up the messages.

**Message formats**

Formatted messages may be written to the JVM logs in one of two formats:

**Basic Format**
> This is the format used in earlier versions of WebSphere application server.

**Advanced Format**
> Extends the basic format by adding information about an event, when possible.

**Basic and advanced format fields**

Basic and Advanced Formats use many of the same fields and formatting techniques. The various fields that may be found in these formats include:

**TimeStamp**
> The timestamp is formatted using the locale of the process where it is formatted. It includes a fully qualified date (for example YYMMDD), 24 hour time with millisecond precision and a time zone.

**ThreadId**
> An 8 character hexidecimal value generated from the hash code of the thread that issued the message.

**ShortName**
> The abbreviated name of the logging component that issued the message or trace event. This is typically the class name for WebSphere internal components, but may be some other identifier for user applications.

**LongName**
> The full name of the logging component that issued the message or trace event. This is typically the fully qualified class name for WebSphere internal components, but may be some other identifier for user applications.

**EventType**

A one character field that indicates the type of the message or trace event. Message types are in upper case. Possible values include:

**A** An Audit message.

**I** An Informational message.

**W** A Warning message.

**E** An Error message.

**F** A Fatal message.

**O** A message that was written directly to System.out by the user application or WebSphere internal components.

**R** A message that was written directly to System.err by the user application or WebSphere internal components.

**u** A special message type used by the message logging component of the WebSphere run time.

**Z** A placeholder to indicate the type was not recognized.

**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.

**UOW** The unit of work identifier for the event. This field is not currently used.

**Basic format**

Message events displayed in basic format use this format. The notation <name> indicates mandatory fields that are always displayed in the basic format message. The notation [name]indicates optional or conditional fields that is included if they can be determined.

*TimeStampThreadIdShortNameEventType*[ClassName][MethodName]*message*

**Advanced format**

Message events displayed in advanced format use this format. The notation <name> is used to indicate mandatory fields that are always displayed in the advanced format for message entries. The notation [name] is used to indicate optional or conditional fields that are included if they can be determined.

*TimeStampThreadIdEventTypeUOWsource=LongName*[ClassName]
[methodName]*OrganizationProductComponentmessage*

# IBM Service log file

The IBM Service, or activity, log file is a binary file to which WebSphere Application Server writes the message events for any servers running under an instance (node). The WebSphere Application Server runtime creates the file (named activity.log by default) in the logs subdirectory of your WebSphere

Application Server instance. For the default WebSphere Application Server instance, this subdirectory is /QIBM/UserData/WebAS5/Base/default/logs. For the default WebSphere Application Server Network Deployment instance, this subdirectory is /QIBM/UserData/WebAS5/ND/default/logs. The granted authorities for the files are:

*PUBLIC                                          *EXCLUDE
QEJBSVR                                          *RW


If your application server is running under a user profile other than the default (QEJBSVR) and that user profile does not have QEJBSVR specified as a group profile, you must explicitly grant *RW authority to the user profile for the activity.log file.

Refer to the topics below for more information on configuring and viewing the IBM Service log file.

**"Configure the IBM Service log file"**
This topic describes how you can use the administrative console to change the name, size and location for the IBM Service log.

**"Monitor the WebSphere Application Server environment by using the Log Analyzer" on page 20**
You cannot view the IBM Service log easily by using a text editor. This topic describes how to use the Log Analyzer tool to view the contents of the IBM Service log file. See Use showlog (page 21) for information on viewing the IBM Service log file on the iSeries when using the Log Analyzer tool is not convenient.

## Configure the IBM Service log file

Use the administrative console to configure the IBM Service log for an application server. Configuration changes for the activity log that are made to a running application server are not applied until the next restart of the application server.

Perform these steps to configure the IBM Service log:

1. Start the administrative console.
2. Expand **Troubleshooting** and click **Logs and Trace**.
3. Click the link for the server you wish to configure.
4. Click **IBM Service Logs**.
5. Select the **Configuration** tab.
6. Scroll through the panel to display the attributes for the stream to be configured.
7. Change the appropriate configuration attributes. See IBM Service log settings ❓ for a description of the attributes for the IBM Service log.
8. Click the **Apply** button.
9. Click the **Save** link at the top of the page to save your configuration changes. Click **Save** on the resulting page.

## Use the WebSphere Application Server trace service

Tracing is useful when you are experiencing a problem whose cause cannot be determined from the WebSphere Application Server log files or other normal problem determination channels. Trace allows you to obtain detailed information about the execution of WebSphere Application Server components, including application servers, clients, and other processes in the environment. Trace files show the time and sequence of methods called by WebSphere Application Server runtime classes, and can be used to pinpoint the failure. Generally, you should not require the use of trace files to determine what is causing an application problem; however, IBM Support personnel may request that you provide traces for a problem.

Refer to these topics for more information on enabling and using the WebSphere Application Server trace service.

**"Enable the trace service"**
This topic describes how you can use the administrative console to enable and disable the trace service for a server.

**"Interpret the Trace Service output" on page 33**
This topic describes the format of the information that is written to the trace file.

For more information on the Trace Service setting, see Trace service settings  .

# Enable the trace service

Use the administrative console to enable the trace service for an application server. The trace service can be enabled dynamically or statically for an application server.

When you enable trace dynamically for a running application server, the trace settings are in effect only for the lifetime of the server. The trace settings are not saved to the application server configuration. Use dynamic trace when the problem you are diagnosing occurs after the application server has started successfully.

When you enable trace statically for an application server, the trace settings are not enabled until the application server is started (or restarted). The trace settings are used every time you start the application server. Use static trace when the problem you are diagnosing occurs during application server startup.

**Note:** Depending on the amount of trace event data being collected, the trace service can negatively affect performance of the application server. Once you have gathered the appropriate trace, be sure to disable (page 32) the trace service.

**Enable the trace service**

Perform these steps to enable the trace service:
1. Start the administrative console.
2. Expand **Troubleshooting** and click **Logs and Trace**.
3. Click the link for the server you wish to configure.
4. Click **Diagnostic trace**.
5. To enable trace statically, select the **Configuration** tab.
   To enable trace dynamically, select the **Runtime** tab.
6. Scroll through the panel to display the current or default trace settings.
7. Change the appropriate configuration attributes. See Trace service settings for a description of the trace service settings. 
8. Click **Apply**.
9. If you are enabling static trace, click the **Save** link at the top of the page to save your configuration changes. Click **Save** on the resulting page.

**Disable the trace service**

Perform these steps to disable the trace service:
1. Start the administrative console.
2. Expand **Troubleshooting** and click **Logs and Trace**.
3. Click the link for the server you wish to configure.

4. Click **Diagnostic trace**.

5. To disable static trace, select the **Configuration** tab. Uncheck the **Enable Trace** checkbox.
   To disable dynamic trace, select the **Runtime** tab. Change the **Trace Specification** to `*=all=disabled`.

6. Click **Apply**.

7. If you are disabling **static** trace, click the **Save** link at the top of the page to save your configuration changes. Click **Save** on the resulting page.

## Interpret the Trace Service output

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 standalone 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 when enabling the trace service.

This information describes the three formats and the fields that make up the messages.

**Trace output formats**

Formatted trace events may be written to the trace file in one of three formats:

**Basic Format**
> This is the format used in earlier versions of WebSphere application server.

**Advanced Format**
> Extends the basic format by adding information about an event, when possible.

**Log Analyzer Format**
> This is the same binary format used for the IBM Service (activity.log) log file. This format allows you or IBM Service to use the "Monitor the WebSphere Application Server environment by using the Log Analyzer" on page 20 tool to open and interpret the trace output.

**Basic and advanced format fields**

Basic and Advanced Formats use many of the same fields and formatting techniques. The various fields that may be found in these formats include:

**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 a Time zone.

**ThreadId**
> An 8 character hexidecimal value generated from the hash code of the thread that issued the trace event.

**ShortName**
> The abbreviated name of the logging component that issued the trace event. This is typically the class name for WebSphere 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 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. Possible values include:

| | |
|---|---|
| > | a trace entry of type method entry. |
| < | a trace entry of type method exit. |
| e | a trace entry of type event. |
| d | a trace entry of type debug. |
| m | a trace entry of type dump. |
| u | a trace entry of type unconditional. |
| Z | a placeholder to indicate that the trace type was not recognized. |

**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.

**UOW** The unit of work identifier for the event. This field is not currently used.

**Basic format**

Trace events displayed in basic format use this format, where *name* indicates mandatory fields that are always displayed in the formatted message, and [name] indicates optional fields that are displayed if they can be determined.

```
TimeStampThreadIdShortNameEventType[ClassName][MethodName]
textmessage[parameter 1][parameter 2]
```

**Advanced format**

Trace events displayed in advanced format use this format, where *name* indicates mandatory fields that are always displayed in the formatted message, and [name] indicates optional fields that are displayed if they can be determined.

```
TimeStampThreadIdEventTypeUOWsource=LongName[ClassName][MethodName]
OrganizationProductComponenttextMessage[parameter 1=parameterValue]
[parameter 2=parameterValue]
```

---

# WebSphere Application Server Message Reference

The links below provide the message identifiers, explanation, and possible user action for messages issued by various WebSphere Application Server software components.

You can log WebSphere Application Server system messages from a variety of sources, including application server components and applications. Messages logged by application server components and

associated IBM products start with a unique message identifier that indicates the component or application that issued the message. The message identifier can be either 8 or 9 characters in length and has the form:

`CCCC1234X`

where *CCCC* is a four character alphabetic component or application identifier. *1234* is a four character numeric identifier used to identify the specific message for that component. *X* is an optional alphabetic severity indicator. (I=Informational, W=Warning, E=Error).

AATL Application Assembly Tool messages

ACIN WebSphere Access Intent messages

ADFS WebSphere Management File Service Subsystem messages

ADMC WebSphere Management Connector Subsystem messages

ADMD WebSphere Management Process Discovery messages

ADME WebSphere Management Event Subsystem messages

ADMG WebSphere Management Connector Subsystem messages

ADML WebSphere Management Process Launching Tool messages

ADMN WebSphere Activity Service messages

ADMR WebSphere Management Subsystem messages

ADMS WebSphere Management Subsystem messages

ADMU WebSphere Management Utilities messages

BNDE WebSphere Management Subsystem messages

CHKP IBM WebSphere PME Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKW IBM WebSphere Validation messages

CHKX IBM WebSphere XD Validation messages

CNTR EJB Container messages

CONM Connection Manager messages

DRSW WebSphere Data Replication Service messages

DYNA WebSphere Dynacache messages

INST Install messages

IVTL Installation Verification Tool messages

J2CA J2EE Connector messages

JSAS Security Association Service messages

JSSL ORB SSL Extensions messages

LTXT WebSphere Localizable Tex messages

MIGR WebSphere Release-to-Release Migration Tooling messages

MSGS WebSphere JMS Server messages
NMSV JNDI Name Services PLGN Webserver Plugins and Native Code SRVE Servlet Engine WTRN

WebSphere Transactions messages

NMSV WebSphere Naming Service messages

ORBX WebSphere ORB Extensions messages

PLPR PluginProcessor messages

PMGR Persistence Manager messages

PMON Tivoli Performance Viewer messages

PMRM WebSphere Performance Monitoring Request Metrics messages

SECJ WebSphere Security messages

SESN Session and User Profiles messages

SOAP Support messages

TRAS WebSphere Trace Facility messages

WACT WebSphere Activity Service messages

WKSP WorkSpace messages

WKSQ Workspace Query Utilties messages

WLTC WebSphere Transaction Monitor messages

WMSG WebSphere Messaging Service messages

WSCL WebSphere Client messages

WSCL WebSphere Client messages

WSCP WebSphere Management Scripting System WASX Non WSCP Scripting messages

WSEC WebServices Security (WS-Security) messages

WSIF (Web Services Invocation Framework) messages

WSVM IBM WebSphere Validation Manager Implementation messages

WSVM IBM WebSphere Validation Manager messages

WSVR WebSphere Runtime messages

WSVR WebSphere Server messages

WSVR WebSphere Server Runtime messages

WSWS WAS.webservices, Web Services messages

WUDU WebUI Deployment Descriptor Utilities. messages

WUPD WebSphere Update Installer messages

WUPD WebSphere Update Installer messages

WUPD WebSphere Update Installer messages

WVER WebSphere Application Server Product History Information messages

WVER WebSphere Application Server Product Information messages

WVER WebSphere Application Server Product Information messages

WWLM WebSphere WLM Client messages

WAS Plugin Configuration Generator

IBM WebSphere Validation

Entity Change Notification Service

WebSphere Web Services

WebSphere Web Services

WebSphere WebServices Engine

WebSphere Web Services

# Appendix. Notices

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

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

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

```
IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY  10594-1785
U.S.A.
```

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

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

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

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

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

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

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

## Trademarks

Lotus, Freelance, and WordPro are trademarks of International Business Machines Corporation and Lotus Development Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation 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.

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

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

## Terms and conditions for downloading and printing publications

Permissions for the use of the publications you have selected for download are granted subject to the following terms and conditions and your indication of acceptance thereof.

**Personal Use:** You may reproduce these Publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these Publications, or any portion thereof, without the express consent of IBM.

**Commercial Use:** You may reproduce, distribute and display these Publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these Publications, or reproduce, distribute or display these Publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the Publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the Publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations. IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

All material copyrighted by IBM Corporation.

By downloading or printing a publication from this site, you have indicated your agreement with these terms and conditions.

**IBM** ®

Printed in USA