WebSphere Event Broker

Troubleshooting

Version 6 Release 0

WebSphere Event Broker

Troubleshooting

Version 6 Release 0

Note
Before you use this information and the product that it supports, read the information in the Notices appendix.

This edition applies to version 6, release 0, modification 0, fix pack 9 of IBM WebSphere Event Broker and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright International Business Machines Corporation 2000, 2009.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this topic collection v

Part 1. Troubleshooting 1

Troubleshooting and support 3
Making initial checks 3
Dealing with problems 10
Using logs 87
Using trace 90
Using dumps and abend files 104
Contacting your IBM Support Center 107
IBM Support Assistant (ISA) 108
Recovering after failure 111

Part 2. Reference 119

Troubleshooting 121
Logs 121
Trace 125
Dumps 128
Abend files 130
Abend in a user-defined extension 131
WebSphere Event Broker event reports 133
WebSphere MQ facilities 146
Database facilities 146
Other sources of diagnostic information on z/OS 147
Solutions to similar problems 148

Part 3. Appendixes 149

Appendix. Notices for WebSphere Event Broker 151
Trademarks in the WebSphere Event Broker information center 153

Index 155

© Copyright IBM Corp. 2000, 2009

About this topic collection

This PDF file has been created from the WebSphere Event Broker Version 6.0 (March 2009) information center topics. Always refer to the WebSphere Event Broker online information center to access the most current information. The information center is periodically updated on the document update site and this PDF and others that you can download from that Web site might not contain the most current information.

The topic content included in the PDF does not include the ″Related Links″ sections provided in the online topics. Links within the topic content itself are included, but are active only if they link to another topic in the same PDF collection. Links to topics outside this topic collection are also shown, but result in a "file not found "error message. Use the online information to navigate freely between topics.

Feedback: do not provide feedback on this PDF. Refer to the online information to ensure that you have access to the most current information, and use the Feedback link that appears at the end of each topic to report any errors or suggestions for improvement. Using the Feedback link provides precise information about the location of your comment.

The content of these topics is created for viewing online; you might find that the formatting and presentation of some figures, tables, examples, and so on are not optimized for the printed page. Text highlighting might also have a different appearance.

© Copyright IBM Corp. 2000, 2009

Part 1. Troubleshooting

Troubleshooting and support 3
Making initial checks 3
Has WebSphere Event Broker never run successfully before? 4
Did you log off Windows while WebSphere Event Broker components were active? 4
Are the Linux and UNIX environment variables set correctly? 4
Are there any error messages or return codes that explain the problem? 5
Can you see all of your files and folders? 5
Can you reproduce the problem? 5
Has the message flow run successfully before? . . 6
Have you made any changes since the last successful run? 7
Is there a problem with descriptive text for a command? 7
Is there a problem with a database? 7
Is there a problem with the network? 8
Does the problem affect all users? 8
Have you recently changed a password? 8
Have you applied any service updates? 8
Do you have a component that is running slowly? 9
Additional checks for z/OS users 9
Dealing with problems 10
Resolving problems when running commands . . 11
Resolving problems when creating resources . . 11
Resolving problems that occur when you start resources 17
Resolving problems that occur when migrating or importing resources. 32
Resolving problems when stopping resources . . 37
Resolving problems when deleting resources . . 38
Resolving problems when developing message flows 42
Resolving problems when deploying message flows 50
Resolving problems when you use the workbench 59
Resolving problems when using databases . . . 66
Resolving problems when using publish/subscribe 74
Resolving problems with performance 80
Resolving problems when developing Configuration Manager Proxy applications . . . 83
Resolving problems with user-defined extensions 84
Resolving problems when uninstalling 85
Using logs 87
Windows: Viewing the local error log. 87
Linux and UNIX systems: Configuring the syslog daemon 88
Viewing the Eclipse error log 89
Using trace 90
Starting user trace 90
Checking user trace options 91
Changing user trace options 92
Stopping user trace 93
Retrieving user trace 93
Starting service trace 94
Checking service trace options 94
Changing service trace options 95
Stopping service trace 95
Retrieving service trace 96
Formatting trace 96
Interpreting trace 98
Clearing old information from trace files . . . 98
Changing trace settings from the workbench . . 99
ODBC trace 100
WebSphere MQ Java Client trace 103
Configuration Manager Proxy trace 103
Using dumps and abend files 104
Checking for dumps 104
Using the DUMP command on z/OS 105
Checking for abend files. 107
Contacting your IBM Support Center 107
IBM Support Assistant (ISA) 108
Installing the IBM Support Assistant (ISA). . . 110
Updating the IBM Support Assistant 110
Recovering after failure 111
Recovering after the Configuration Manager fails. 111
Recovering after the broker fails 112
Recovering after the User Name Server fails . . 113
Recovering after an execution group fails . . . 113
Recovering after the broker's queue manager fails 114
Recovering after the Configuration Manager's queue manager fails 114
Recovering after the User Name Server's queue manager fails 116
Recovering after a shared queue manager fails 116
Recovering after the configuration repository fails 117
Recovering after the broker database fails . . . 117

© Copyright IBM Corp. 2000, 2009

Troubleshooting and support

If you are having problems with your WebSphere® Event Broker applications, there are a number of techniques you can use to help you diagnose and solve the problems.

This section contains information about the various techniques that you can use to diagnose problems with WebSphere Event Broker.

- "Making initial checks"
- "Dealing with problems" on page 10
- "Using logs" on page 87
- "Using trace" on page 90
- "Using dumps and abend files" on page 104
- "Recovering after failure" on page 111
- "Contacting your IBM Support Center" on page 107

You can also read the general troubleshooting guidance in the following topics:

- Troubleshooting overview
- Searching knowledge bases
- Getting fixes
- IBM Software Support
- "IBM Support Assistant (ISA)" on page 108

Making initial checks

Before you start problem determination in detail, consider whether there is an obvious cause of the problem, or an area of investigation that is likely to give useful results. This approach to diagnosis can often save a lot of work by highlighting a simple error, or by narrowing down the range of possibilities.

This section contains a list of questions to consider. As you go through the list, make a note of anything that might be relevant to the problem. Even if your observations do not suggest a cause straight away, they might be useful later if you have to carry out a systematic problem determination exercise.

- "Has WebSphere Event Broker never run successfully before?" on page 4
- "Did you log off Windows while WebSphere Event Broker components were active?" on page 4
- "Are the Linux and UNIX environment variables set correctly?" on page 4
- "Are there any error messages or return codes that explain the problem?" on page 5
- "Can you reproduce the problem?" on page 5
- "Has the message flow run successfully before?" on page 6
- "Have you made any changes since the last successful run?" on page 7
- "Is there a problem with descriptive text for a command?" on page 7
- "Is there a problem with a database?" on page 7
- "Is there a problem with the network?" on page 8
- "Does the problem affect all users?" on page 8

© Copyright IBM Corp. 2000, 2009

- "Have you recently changed a password?" on page 8
- "Have you applied any service updates?" on page 8
- "Do you have a component that is running slowly?" on page 9
- "Additional checks for z/OS users" on page 9

Has WebSphere Event Broker never run successfully before?

If you have successfully run a WebSphere Event Broker component, determine whether your problem is caused by incorrect installation and setup, or a different cause. If it has *not* run successfully before, work through the information to determine what is preventing successful operation.

Complete the following steps:

1. **Check your setup**

   You might have failed to set up WebSphere Event Broker correctly.

   Linux UNIX On Linux® and UNIX® operating systems, check that you have set up the command environment correctly; see Setting up a command environment for more information.

2. **Verify the installation**

   Check the Installation Guide for information about how you can verify a basic configuration on your system. On Linux on x86 and Windows®, you can use the Default Configuration wizard and the Samples Preparation wizard to help you with basic configuration and verification.

   For more detailed configuration information, refer to Configuring WebSphere Event Broker.

Did you log off Windows while WebSphere Event Broker components were active?

Windows On Windows, logging off when WebSphere Event Broker components (a broker, the Configuration Manager, or the User Name Server) are active can cause a problem. You might see messages, including BIP2070, BIP2642, BIP1102, and BIP1103 in the Windows Event log if this is the case.

When you log off, any queue managers that support these WebSphere Event Broker components are stopped unless they are defined to run as Windows services. The components all run as Windows services and remain active, but they find that the queue manager and queue manager objects that are associated with them are no longer available.

1. If you log off, restart the queue manager that supports the Configuration Manager or the User Name Server, and the component re-establishes the connection to the queue manager.
2. To avoid this problem, set the queue managers that are used by the broker, Configuration Manager and User Name Server to run as Windows services, so that logging off Windows does not cause this problem.

Are the Linux and UNIX environment variables set correctly?

UNIX On Linux and UNIX systems, the basic settings are made by the mqsiprofile command, which is located in the following directory:

install_dir/bin

See Setting up a command environment for information about setting up the command environment; see Creating a broker on Linux and UNIX systems for instructions about creating a broker on your operating system.

Are there any error messages or return codes that explain the problem?

You can find details of error messages and return codes in several places.

- **BIP messages and reason codes**

  WebSphere Event Broker error messages have the prefix BIP. If you receive any messages with this prefix (for example, in the UNIX, Linux, or z/OS® syslog), see Diagnostic messages for an explanation.

  Windows In the Windows Event log, references to BIP messages are identified by the source name "WebSphere Broker v6000", where v6000 can be replaced by a number representing the exact service level of your installation, for example 6001.

  WebSphere Event Broker messages that have a mixture of identifiers such as BIPv600, BIPv500, WMQIv210, MQSIv202, and MQSIv201 indicate a mixed installation, which does not work properly.

- **Other messages**

  For messages with a different prefix, such as AMQ or CSQ for WebSphere MQ, or SQL for DB2®, see the appropriate messages and codes documentation for a suggested course of action to help resolve the problem.

  Messages that are associated with the startup of WebSphere Event Broker, or were issued while the system was running before the error occurred, might indicate a system problem that prevented your application from running successfully.

  A large or complex WebSphere Event Broker broker domain might require some additional configuration of the environment beyond what is recommended in Installing complementary products. The need for such configuration changes is typically indicated by warning or error messages that are logged by the various components, including WebSphere MQ, the databases, and the operating system. These messages are normally accompanied by a suggested user response.

Can you see all of your files and folders?

If you are using Windows Explorer to view your files and you cannot see all of your files and folders, such as the broker workpath directory, this is because Windows Explorer, by default, hides some files and folders.

To show all files in Windows Explorer:

1. Click **Tools → Folder options**. The Folder Options dialog box opens.
2. Click the **View** tab and select **Show hidden files and folders**.

Can you reproduce the problem?

If you can reproduce the problem, consider the conditions under which you can do so:

- **Is the problem caused by a particular message flow?** If so, use user tracing to identify the problem.
- **Is the problem caused by a command?** On distributed operating systems, you issue commands at the system command line.

z/OS On z/OS, you can issue commands from the console, the syslog, or by submitting a batch job. You enter customization commands from an OMVS session. Console commands that you enter from the console or syslog might be converted to uppercase, depending on the system configuration. This can cause some commands, such as **mqsichangetrace**, to fail, especially if these commands contain parameters that must be lowercase. An error message indicating that the execution group is not available might be caused by the execution group name being in the wrong case. The same thing can happen on message flows.

- **Does a problem command work if it is entered by another user ID?**

  If the command works when it is entered by another user ID, check the environment of each user. Paths, especially shared library paths, might be different. On Windows, UNIX systems, and Linux verify that all users have set up their command environment correctly; refer to sample profile for more information.

  Windows On Windows, the environment for the broker is determined by the system settings, not by a particular user's variables. However, the user's variables affect non-broker commands.

  UNIX On LinuxUNIX systems, only the service ID that is specified when the broker was created can start a broker.

  Windows On Windows, any authorized user can start a broker.

Has the message flow run successfully before?

Sometimes a problem appears in a message flow that has previously run successfully. To identify the cause of the problem, answer the following questions:

- **Have you made any changes to the message flow since it last ran successfully?**

  If so, it is likely that the error exists somewhere in the new or modified part of the flow. Examine the changes and see if you can find an obvious reason for the problem.

- **Have you used all the functions of the message flow before?**

  Did the problem occur when you used part of the message flow that had never been invoked before? If so, it is likely that the error exists in that part. Try to find out what the message flow was doing when it failed by using user tracing.

  If you have run a message flow successfully on many previous occasions, check the current queue status and the files that were being processed when the error occurred. It is possible that they contain some unusual data value that invokes a rarely-used path in the message flow.

- **Does the message flow check all return codes?**

  Has your system been changed, perhaps in a minor way, but your message flow does not check the return codes it receives as a result of the change? For example:

  – Does your message flow assume that the queues that it accesses can be shared? If a queue has been redefined as exclusive, can your message flow deal with return codes indicating that it can no longer access that queue?

  – Have any security profiles been changed? A message flow could fail because of a security violation.

- **Does the message flow expect particular message formats?**

  If a message with an unexpected message format has been put onto a queue (for example, a message from a queue manager on a different operating system) it might require data conversion or a different form of processing. Also, check whether you have changed any of the message formats that are used.

- **Does the message flow run on other WebSphere Event Broker systems?**

  Is there something different about the way that your system is set up that is causing the problem? For example, have the queues been defined with the same maximum message length or priority? Are there differences in the databases used, or their setup?
- **Can you see errors from WebSphere Event Broker or external resources, such as databases?**

  Your message flow might be losing errors because of incorrect use of the failure terminals on built-in nodes. If you use the failure terminals, make sure that you handle errors adequately. See Handling errors in message flows for more information about failure terminals.

Have you made any changes since the last successful run?

When you are considering changes that might have been made recently, think not only about WebSphere Event Broker, but also about the other programs with which it interfaces, the hardware, device drivers, and any new applications.

- **Have you changed your initialization procedure?**

  On z/OS, have you changed any data sets or library definitions? Has the operating system been initialized with different parameters? Check for error messages generated during initialization.
- **Has the profile of the user who is running the commands on Linux or UNIX systems been changed?**

  If so, this might mean that the user no longer has access to the required objects and commands.
- **Has any of the software on your system been upgraded to a later release?**

  Check that the upgrade completed successfully, and whether the new software is compatible with WebSphere Event Broker (check the product readme.html file).
- **Do your message flows deal with the errors and return codes that they might get as a result of any changes that you have made?**

  Check that your message flow can handle all possible error situations.

Is there a problem with descriptive text for a command?

On Linux and UNIX systems, be careful when including special characters (back slash (\) and double quote (″) characters) in descriptive text for some commands. If you use either of these characters in descriptive text, precede them with a back slash,\; that is, enter \\ or \″ if you want \ or ″ in your text.

Is there a problem with a database?

If you have database problems, make the following checks:

- Check that the database is started.
- Check your ODBC.ini file on Linux or UNIX systems (if you are using the default ODBC.ini file, this is located at *install_dir*/odbc/.odbc.ini).

  For a broker database, the basic configuration is correct if the broker can be created. If the broker can be created, but not started, use ODBC tracing to investigate the cause.
- Check the number of database connections: on DB2 for AIX®, using local mode connections, the maximum is 10.

- For messages indicating that imbdfdb2v6.lil (the 32-bit XA support LIL) failed to load, ensure that a supported database is installed.

Is there a problem with the network?

WebSphere Event Broker uses WebSphere MQ for inter-component communication. If components are on separate queue managers, they are connected by message channels, although communications between the workbench and the Configuration Manager use MQI channels.

There can be communication problems between any of these:

- Brokers
- The User Name Server
- The Configuration Manager
- The workbench

If any two components are on different queue managers, make sure that the channels between them are working. Use the WebSphere MQ **display chstatus** command to see if messages are flowing.

Use the **ping** command to check that the remote computers are connected to the network, or if you suspect that the problem might be with the network itself. For example, use the command **ping *brokername***, where *brokername* is a computer name. If you get a reply, the computer is connected. If you don't get a reply, ask your network administrator to investigate the problem. Further evidence of network problems might be messages building up on the transmission queues.

Does the problem affect all users?

On distributed systems, problems can be caused by different users having different environments. Check whether there is a different user ID in an incoming message, or a different user ID issuing a command (or acting as the broker's service ID).

Have you recently changed a password?

Check that passwords are updated correctly.

If you have changed the password at the operating system level for the user ID that is used either as the ServiceUserId or the DataSourceUserId for any of the WebSphere Event Broker components, you might find these components have access problem. If so, perform the following steps:

- Change the properties of the affected component to reflect the password change. Use the mqsichangebroker command, mqsichangeconfigmgr command, and mqsichangeusernameserver command to change the properties of the WebSphere Event Broker components.

  If you are using a Derby database on Windows and you change a password, use the mqsichangedbimgr command to update the Windows service that runs the Derby Network Server.
- Ensure that your passwords do not contain reserved keywords. For example, "IBM", which is a reserved keyword in DB2.

Have you applied any service updates?

Check what service updates you have applied to your software.

If you have applied an APAR or a PTF to WebSphere MQ for z/OS, or a fix pack or interim fix has been applied to a distributed operating system, check that no error message was produced. If the installation was successful, check with the IBM Support Center for any known error with the service update.

- If a service update has been applied to any other product, consider the effect that it might have on WebSphere Event Broker.
- Ensure that you have followed any instructions in the service update that affect your system. For example, you might need to redefine a resource, or stop and restart a component.
- If you are not sure whether a service update has been applied to your system, search for and view the release notes stored in the product installation directory. These notes include the service level and details of the maintenance that you have applied.

Do you have a component that is running slowly?

If a particular component, or the system in general, is running slowly, complete the following steps:

- Check whether tracing is on. This could be WebSphere Event Broker user tracing or service tracing, ODBC tracing, WebSphere MQ tracing, or native database tracing. If it is, turn the tracing off.
- Clear out any old abend files from your errors directory. If you do not clear the directory of unwanted files, you might find that your system performance degrades due to significant space being used up.
- On Windows, use the workpath **-w** parameter of the **mqsicreatebroker** command to create the errors directory in a hard drive partition that does not contain WebSphere Event Broker or Windows.
- Increase your system memory.

Additional checks for z/OS users

On WebSphere Event Broker for z/OS, you might find that:

- The console has unexpected messages; look for an explanation in Diagnostic messages or other relevant messages and codes documentation.
- RRS is not started; issue the command **D A,RRS**.
- The queue manager is not available; issue the WebSphere MQ command **DISPLAY THREAD(*)**.
- DB2 is not working; issue the DB2 command **DISPLAY THREAD**.
- The User Name Server is not working.

The cause of your problem on z/OS could be in any of the following areas. Check each of these, starting with whichever seems the most likely, based on the nature of your problem.

- The queue manager address space
- A queue manager in your queue-sharing group
- The channel initiator address space
- Batch or TSO address space
- The z/OS system (including ARM, RRS, or the Coupling Facility)
- The network (including APPC or TCP/IP)
- Another system, for example a queue manager on another operating system or a WebSphere MQ client

- The external security manager product, for example RACF or ACF2
- DB2

Understanding interactions between the runtime components

There are three runtime components: the broker, the Configuration Manager and the execution group. These components communicate with each other by exchanging command requests inside WebSphere MQ messages to perform actions, such as deploying a message flow. After completion of the command request, responses are sent back to the originating component, indicating whether the request was successful.

On WebSphere Event Broker for z/OS, you see extra information messages issued by z/OS runtime components that allow you and IBM Service personnel to determine the interactions between the various runtime components, including the Configuration Manager, broker and execution group.

When the broker receives command requests from the Configuration Manager, the broker issues a message that identifies the command request. When this request completes, a message is issued that indicates whether the command is successful. Each command request that the broker receives results in at least one matching command request being sent to an execution group, and a corresponding message being issued. Every response from an execution group that results from these command requests will result in a message being issued. These messages can be turned on and off and are to be used by the IBM Service team.

When an execution group receives command requests from the broker, the execution group issues information messages that identify the command request. For actions that are contained within a command request, an information message is issued that identifies the action to be performed and the resource upon which the action is to take effect. When the request completes, the execution group issues a message that indicates that the message has been processed.

When a Configuration Manager receives a command request and when this request completes, information messages are issued.

Loading IBM and user-defined nodes and parsers

When an execution group starts, it loads all available IBM and user-defined nodes and parsers. For each library that is loaded, two messages are issued. One is issued before the library is loaded and one is issued after the library has been loaded.

Dealing with problems

Learn how to resolve some of the typical problems that can occur.

Before you start:

Make initial checks and check any relevant logs, trace and dumps to see if you can rectify the problem.

This section contains the following topics:

- "Resolving problems when running commands" on page 11
- "Resolving problems when creating resources" on page 11
- "Resolving problems that occur when you start resources" on page 17

- "Resolving problems that occur when migrating or importing resources" on page 32
- "Resolving problems when stopping resources" on page 37
- "Resolving problems when deleting resources" on page 38
- "Resolving problems when developing message flows" on page 42
- "Resolving problems when deploying message flows" on page 50
- Resolving problems when using messages
- "Resolving problems when you use the workbench" on page 59
- "Resolving problems when using databases" on page 66
- "Resolving problems when using publish/subscribe" on page 74
- "Resolving problems with performance" on page 80
- "Resolving problems when developing Configuration Manager Proxy applications" on page 83
- "Resolving problems when uninstalling" on page 85

Resolving problems when running commands

Advice for dealing with some common problems that can arise when you run commands.

ReportEvent() error message is issued on Windows when you attempt to run a command

- **Scenario:** A ReportEvent () message is generated whenever you attempt to run any `mqsi*` command, The ReportEvent () message is followed by the result of the command itself.
- **Explanation:** The Windows Application Log has become full.
- **Solution:** Clear the Windows Application Log from the Event Viewer.

Resolving problems when creating resources

Advice for dealing with some common problems that can occur when you create resources.

Look for the problem that you have encountered, and follow the guidance provided to recover from the error.

Problems when creating a broker:

- "Checklist for configuring databases" on page 12
- "Message BIP8146E is issued when creating a broker" on page 13
- "Message BIP1511 is issued when creating a broker" on page 13
- "Message BIP0874 is issued when saving the topology editor after creating or deleting a broker" on page 13
- "Messages BIP2321, BIP2322, BIP8040 are issued when creating a broker" on page 14
- "Message BIP8081 is issued when creating a broker" on page 14
- "You cannot create files when creating a broker on AIX" on page 14
- "The Configuration Manager does not remove references to a deleted broker's queue manager" on page 14
- "The JCL BIPGEN fails when you create a component on z/OS" on page 15
- "The mqsicreatebroker command hangs if the DB2 profile has not been run" on page 15

- "The mqsicreatebroker command core dumps when it is run using an Oracle data source" on page 15
- "Your DataFlowEngine terminates with an abend when you create a broker on HP-UX using Oracle" on page 16

Problems when creating other resources:

- "Error message BIP8075 is issued when creating a Configuration Manager" on page 16
- "Error message BIP2624 is issued when creating an execution group" on page 16
- "You do not know what authorities are set as part of the mqsicreateaclentry command" on page 16
- "The Default Configuration wizard fails with invalid argument specified" on page 17
- "The Default Configuration wizard fails after installing DB2" on page 17

Checklist for configuring databases

1. Ensure that the ODBC database definitions have been fully configured by following the documented instructions in Configuring broker databases. Remember the following points:
   a. Edit the final stanza in the ODBC files; the *[ODBC]* stanza.
   b. You must configure the 32-bit odbc.ini settings even if all your execution groups are 64-bit. The exception is HP-UX on Itanium, where all broker processes are 64-bit, and you must configure only the 64-bit ODBC definition file.
   c. Ensure that the ODBC drivers used are as specified in this information center and in the sample odbc.ini files. This check is particularly important when you migrate from an earlier release, when different driver names might have been required. You might need to update the odbc.ini file used with the earlier release to reflect new ODBC driver names, and to include new settings.
2. Check the environment settings, referring to Configuring broker databases for details. Remember the following points:
   a. Run the mqsiprofile command.
   b. Set the required database environment variables (for example: run db2profile for DB2).
   c. If you are using a 64-bit DB2 database instance, add `<DB2 instance directory>/sqllib/lib32` to the start of the library search path environment variable (for all 32-bit brokers; that is, all broker platforms except HP-UX on Itanium).
   d. If the broker needs to access an Informix® user database, add `$INFORMIX/lib:$INFORMIX/lib/esql:$INFORMIX/lib/cli` to the library search path environment variable (see the Informix product documentation for more information). You must ensure that any WebSphere Event Broker libraries appear *before* any Informix libraries in the library search path.
   e. If you are using a 64-bit Oracle instance, add `$ORACLE_HOME/lib32` to the start of the library search path environment variable (for all 32-bit brokers; that is, all broker platforms except HP-UX on Itanium).
3. Check that the user ID you use to access the database has sufficient permissions on the broker database to connect and to create tables and indices as described in Authorizing access to broker databases.

4. Ensure that the broker's service user ID has read permissions on the database client lib directories and files.
5. If you are using Oracle, you must also run the mqsi_setupdatabase command (for details see the mqsi_setupdatabase command).

Message BIP8146E is issued when creating a broker

- **Scenario:** The mqsicreatebroker command fails with the following message:

  ```
  BIP8146E: Unable to retrieve the SQL primitives supported by the Database.
  ```

- **Explanation:** The command has failed to successfully connect and retrieve database properties information.
- **Solution:** If the command has failed with this error:
  - Check that the user ID that you specified to access the database has sufficient permissions to connect to the database, and to create tables.
  - Check that the ODBC configuration has been performed in accordance with Enabling ODBC connections to the databases

Message BIP1511 is issued when creating a broker

- **Scenario:** Error message BIP1511E is displayed when you create a broker in the Topology editor; for example:

  ```
  BIP1511E: Queue manager 'QM1' cannot be associated with broker
  'BR1'; it is already associated with broker 'BR1'
  ```

- **Explanation:** A queue manager cannot be associated with the specified broker because it is already associated with that broker name. The problem occurs because you performed an incorrect sequence of actions when you originally deleted the broker, and you are now trying to re-create a broker of the same name.
- **Solution:** Delete the broker before attempting to re-create it:
  1. Stop the broker by using the mqsistop command.
  2. Delete the broker from the topology in the workbench.
  3. Check that the deletion was successful, and that the broker has disappeared from the list in the workbench.
  4. If you are using publish/subscribe, and the network of connected brokers has been modified as a result of the deletion, issue a delta deploy of the topology.
  5. Delete the broker by using the mqsideletebroker command.

Do not try to re-create the broker until the deleted topology has been deployed successfully. If the broker is no longer shown in the workbench, you can then re-create the broker:

1. Create the broker by using the mqsicreatebroker command.
2. Create the broker in the topology in the workbench.
3. Deploy the topology.

Message BIP0874 is issued when saving the topology editor after creating or deleting a broker

- **Scenario:** Message BIP0874E is displayed when you are saving the topology editor after creating or deleting a broker, and the broker is not created or deleted.
- **Explanation:** This problem occurs because you are no longer authorized to modify the topology.
- **Solution:** Ask your WebSphere Event Broker administrator to give your user ID full access to the broker topology.

If you close the editor and save the topology changes, you do not receive any error messages.

Messages BIP2321, BIP2322, BIP8040 are issued when creating a broker

- **Scenario:** The mqsicreatebroker command fails with the above set of messages.
- **Explanation:** The command has failed to successfully connect and retrieve database properties information.
- **Solution:** Make the following checks:
  1. The environment has been setup correctly for database access. In particular, if you are using a 64-bit DB2 or Oracle instance, ensure that the 32-bit database client libraries directory is at the start of the library search path.
  2. The broker's service user ID has read access to the database client libraries.
  3. For all broker platforms except HP-UX on Itanium, that the 32-bit ODBC file is configured, because the broker is a 32-bit application.

Message BIP8081 is issued when creating a broker

- **Scenario:** Message BIP8081E is displayed when you are creating a broker, the inserted message does not format correctly, and the broker is not created.
- **Explanation:** This problem occurs because you are not a member of the correct group.
- **Solution:** Read the explanation of message BIP8081, and ask your WebSphere Event Broker administrator to give your user ID access to the mqbrkrs group.

You cannot create files when creating a broker on AIX

- **Scenario:** When you run the mqsicreatebroker command on WebSphere Event Broker for AIX, the following message is displayed:

  ```
  BIP8135E Unable to create files. Operating System return code 1
  ```

- **Explanation:** The user ID that you create for WebSphere Event Broker testing must have a primary group of mqm, and have mqbrkrs as one of the group set. The following example shows an AIX SMIT panel listing the **Change / Show Characteristics of a User**:

```
                Change / Show Characteristics of a User

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
[TOP]                                [Entry Fields]
* User NAME                          peterc
  User ID                            [202]                    #
  ADMINISTRATIVE USER?               false                    +
  Primary GROUP                      [mqm]                    +
  Group SET                          [mqm,system,mqbrkrs,sys> +
```

The Configuration Manager does not remove references to a deleted broker's queue manager

- **Scenario:** The broker definition is incorrect, so you cannot add a new broker.
- **Explanation:** This problem can occur if you delete a broker component without removing it from the Message Brokers Toolkit first. If your broker definition is incorrect, you cannot instruct the Configuration Manager to add a new broker with the same queue manager to the broker domain because the Configuration Manager reports that the queue manager is already associated with the broker that you deleted.

If your broker definition is incorrect, you cannot redeploy. If you delete a broker component then try to deploy to it, any subsequent deploy to that broker fails because the Configuration Manager reports that it is already deploying to the broker that you have deleted.

- **Solution:** Delete the queue manager for the deleted broker, then deploy. When you deploy, the Configuration Manager recognizes that the queue manager does not exist and cleans up the broker's definition. You can now associate the queue manager with another broker or redeploy the broker.

The JCL BIPGEN fails when you create a component on z/OS

- **Scenario:** The BIPGEN job fails when you are copying the component profile (BIPBPROF, BIPCPROF, or BIPUPROF) from the PDSE to the file system.
- **Explanation:** There might not be enough space in the file system, the component profile might not exist, or you might not have the appropriate authority.
- **Solution:** Make the following checks:
  - The file system has sufficient space. You can check how much space is used and how much is free in a file system using the OMVS command df -P /pathname. 100 MB is 3 276 800 512 byte sectors.
  - The profile file exists in the PDSE.
  - Your user ID has the appropriate authority to write to the file system.

The mqsicreatebroker command hangs if the DB2 profile has not been run

- **Scenario:** The mqsicreatebroker command hangs if the DB2 profile has not been run. If you stop the command, this could result in a partial broker setup. This partially-created broker might be shown by the mqsilist command and the registry but none of the broker tables will have been created.
- **Explanation:** This error can occur if the environment has not been set up correctly.
- **Solution:** Remove any partial setup and ensure that the environment is set up correctly by following the instructions in Configuring WebSphere Event Broker. Run the DB2 profile before reissuing the mqsicreatebroker command.

The mqsicreatebroker command core dumps when it is run using an Oracle data source

- **Scenario:** The mqsicreatebroker command core dumps when it is run using an Oracle data source.
- **Explanation:** This problem occurs when the Oracle libraries fail to load.
- **Solution:** If you can run the Oracle sqlplus utility, your basic Oracle environment is correct. However, the default Oracle libraries are 64-bit versions and these are not compatible with the broker.
  - Check your **LD_LIBRARY_PATH** setting in the environment when you create the broker.

    The broker requires the 32-bit Oracle libraries; these are typically found in <ORACLE_HOME>/lib32 for a 64-bit Oracle installation. On some operating systems, you can install a 32-bit only version of Oracle; if you have this variant of Oracle installed, the libraries are in <ORACLE_HOME>/lib
  - Check that the broker's service user ID has read access to the Oracle client libraries.
  - Check that the 32-bit ODBC definition file has been configured for the broker on all platforms except HP-UX on Itanium.

Your DataFlowEngine terminates with an abend when you create a broker on HP-UX using Oracle

- **Scenario:** Your message flow (DataFlowEngine or DFE) terminates with an abend when you create a broker on HP-UX using Oracle.
- **Explanation:** This problem occurs when DB2 is installed on the same computer as Oracle.
- **Solution:** Remove the DB2 LIL files that are used by WebSphere Event Broker. For example, issue the following commands:

```
mv install dir/lib/imbdfdb2v6.lil install dir/lib/imbdfdb2v6.blank
mv install dir/lib/imbdfdb2.lil install dir/lib/imbdfdb2.blank
```

Error message BIP8075 is issued when creating a Configuration Manager

- **Scenario:** On a Windows system, the mqsicreateconfigmgr command fails with message BIP8075 and a Java exception unsatisfiedLinkException.
- **Explanation:** The Configuration Manager cannot find the JAR files that it needs to connect to the configuration database. If you have installed additional software since installing the broker, or have made manual updates to your CLASSPATH, these might cause the mqsicreateconfigmgr command to fail.
- **Solution:** Ensure that the DB2 JAR files, especially db2java.zip, are in your CLASSPATH.

Error message BIP2624 is issued when creating an execution group

- **Scenario:** When you create an execution group, you get several BIP2624 messages (MQRC=2012 (MQRC_ENVIRONMENT_ERROR)), and no WebSphere MQ messages are processed.
- **Explanation:** You have created the broker to run as a WebSphere MQ trusted application (that is, the broker runs in the same process as the WebSphere MQ queue manager), but the user ID that you specified does not have the required authority.
- **Solution:** If you request the trusted application option on the mqsicreatebroker command by specifying the **-t** parameter, perform the appropriate steps for your operating system:

  Windows **Windows**

  Using the **-i** parameter on the mqsicreatebroker command, specify a service user ID that is a member of WebSphere MQ group mqm.

  Linux UNIX **Linux and UNIX systems**

  Specify the user ID mqm on the **-i** parameter on the mqsicreatebroker command.

You do not know what authorities are set as part of the mqsicreateaclentry command

- **Scenario:** You do not know what authorities are set on the queue manager as part of the mqsicreateaclentry command.
- **Solution:** Use the WebSphere MQ dspmqaut command to check which authorities have been set on a queue manager by the mqsicreateaclentry command. Ensure that the following authorities are set:
  - inq
  - set
  - connect

- altusr
- chg
- dsp
- setall

The Default Configuration wizard fails with invalid argument specified

- **Scenario:** On a Windows system, the Default Configuration wizard fails. The Default Configuration wizard log contains the following error message:

  ```
  Invalid argument 'WBRK61_DEFAULT_CONFIGURATION_MANAGER' specified. Argument specified
  should be well formed. Correct and reissue the command.
  ```

- **Explanation:** You have entered a user name that contains one or more spaces. The Default Configuration does not support the use of user names containing spaces, because using the space character can cause problems in communications with other operating systems.
- **Solution:** Use an alternative user name that does not contain any spaces in the Default Configuration wizard.

The Default Configuration wizard fails after installing DB2

- **Scenario:** You have run the Default Configuration before installing DB2 on your system, and removed the components created by the Default Configuration wizard. You have then installed DB2 on your system, and attempted to run the Default Configuration again.
- **Explanation:** The broker database for the Default Configuration, `DEFBKDB6`, exists on the system as a Derby database. With DB2 installed, the Default Configuration wizard searches for DB2 databases and not Derby databases.
- **Solution:** To check if this is the cause of your problem, run the following command:

  ```
  mqsilist DatabaseInstanceMgr6
  ```

  The following message is displayed if the default broker database exists as a Derby database on the system: `BIP9814I: DEFBKDB6- Derby`. If the default broker database exists as a Derby database, you must manually delete the database, and then rerun the Default Configuration wizard. Use the following instructions to solve this problem:

  1. Run the following command:

     ```
     mqsideletedb -d DeleteAllDatabases
     ```

  2. Run the Default Configuration wizard.

Resolving problems that occur when you start resources

This topic contains advice for dealing with some common problems that can arise when you start resources.

- "Resolving problems when starting a broker" on page 18
  - "The broker and execution group are not running after you use the Default Configuration wizard" on page 18
  - "The broker or Configuration Manager fails to start because there is not enough space in the Java TMPDIR directory" on page 19
  - "The broker fails to start when it is installed on HP-UX with Oracle and DB2" on page 19
  - "Error message BIP2048 is issued on z/OS when your broker fails to start" on page 19

  - "Diagnostic message ICH408I is issued on z/OS when your broker fails to start" on page 20
  - "Abend code 047 is issued with a diagnostic message" on page 21
  - "Abend code 047 is issued in bin/bipimain" on page 21
  - "Database error message BIP2322 is issued when you start or stop a broker on Linux or UNIX" on page 22
  - "Error message BIP2228 is issued when you try to start a second broker on Linux or UNIX" on page 22
  - "MQIsdp client connection is refused by the broker" on page 22
  - "Error messages BIP2604 and BIP2624 are issued when you start a broker or a new message flow" on page 22
  - "When you start the broker through the DataFlowEngine, it cycles continually" on page 23
  - "You have changed your logon password and cannot start your Configuration Manager or broker" on page 23
  - "Broker on Windows platform fails to start with message BIP2818 and an abend" on page 24
- "Resolving problems when starting a Configuration Manager" on page 24
  - "The broker or Configuration Manager fails to start because there is not enough space in the Java TMPDIR directory" on page 19
  - "You have changed your logon password and cannot start your Configuration Manager or broker" on page 23
  - "Error messages BIP1708 and BIP8283 are issued when you start a Configuration Manager on Windows" on page 25
- "Resolving problems when starting other resources" on page 26
  - "Resources terminate during startup" on page 26
  - "Resources hang at startup on Windows" on page 26
  - "Error message BIP8048 is issued when you start a component" on page 26
  - "You experience problems with the default configuration" on page 27
  - "A ″Not Found″ error is issued when you click a link to the Samples Gallery or a specific sample" on page 28
  - "The Quick Tour displays as a blank window" on page 28
  - "Error message BIP0832 is issued on startup" on page 28
  - "Your execution groups restart repeatedly" on page 29
  - "You cannot tell whether startup is complete on z/OS" on page 29
  - "Abend code 0C1 is issued when you try to start the DataFlowEngine on z/OS" on page 30
  - "Error message BIP2604 with return code `MQRC_CONNTAG_IN_USE` is issued during the start of a message flow on z/OS" on page 30
  - "You cannot start the User Name Server" on page 31
  - "A device allocation error is issued" on page 31

Resolving problems when starting a broker

This topic contains advice for dealing with some common problems that can arise when you start a broker.

The broker and execution group are not running after you use the Default Configuration wizard:

- **Scenario:** You have used the Default Configuration wizard to create a default configuration, and the alerts in the Broker Administration perspective show that your broker and execution group are not running.
- **Explanation:** At this stage, nothing has been deployed to the broker or the execution group. therefore they are displayed as not running. After you have deployed to the broker, the correct state of the broker and execution group is displayed.

The broker or Configuration Manager fails to start because there is not enough space in the Java TMPDIR directory:

- **Scenario:** The broker or Configuration Manager fails to start and an error message indicates that insufficient space is available.
- **Explanation:** The broker and Configuration Manager use Java™ JAR files. When the broker or Configuration Manager starts, the Java runtime environment extracts the JAR files into a temporary directory, Java TMPDIR. On Linux, UNIX, and z/OS systems, the TMPDIR directory is typically /tmp; on Windows systems, it is c:\temp. If this directory is not large enough to hold the JAR files, the broker or Configuration Manager does not start.
- **Solution:** Use one of the following methods to specify the location of this temporary JAR directory:
  - Use the environment variable TMPDIR.
  - Set the system property java.io.tmpdir.

  Allow at least 50 MB of space in this directory for broker components. and 10 MB of space for Configuration Manager components. You might need more space if you deploy large user-defined nodes or other JARs to the broker component.

The broker fails to start when it is installed on HP-UX with Oracle and DB2:

- **Scenario:** Oracle 9i, or 10g, and DB2 V8 or V9 are installed on the same HP-UX system and the broker is using an Oracle broker database. The broker fails to start, and an abend file shows that a segmentation violation has occurred because symbols in the Oracle libraries are resolved in a DB2 library.
- **Explanation:** This problem is due to a coexistence issue between Oracle and DB2 on HP-UX.
- **Solution:** To use an Oracle broker database and have DB2 installed on the same HP-UX system, disable the *install_dir*/lib/imbdfdb2.lil file. Rename the imbdfdb2.lil file to solve this problem.

  When you disable this file, it stops the broker invoking DB2 libraries that the HP-UX operating system then uses to resolve the Oracle symbols.

Error message BIP2048 is issued on z/OS when your broker fails to start:

- **Scenario:** The following message is written to the SDSF SYSLOG on z/OS when your broker fails to start:

```
+(MA00BRK) 0 BIP2048E: An Exception was caught while issuing database SQL command
connect.
+(MA00BRK) 0 BIP2321E: Database error: ODBC return code '-1'.
+(MA00BRK) 0 BIP2322E: Database error: SQL State '58004'; Native Error Code '-99999'
```

- **Explanation:** The SQL State value of 58004 reveals that a system error occurred. This might be due to a DB2 authorization problem.
- **Solution:** The ID under which the broker runs (the started task ID) needs the following DB2 authorizations to be granted:
  - DELETE, INSERT, SELECT and UPDATE authorization on the broker database tables

– EXECUTE authorization on plan DSNACLI
– SELECT authorization on table SYSIBM.SYSSYNONYMS
– SELECT authorization on table SYSIBM.SYSTABLES
– SELECT authorization on table SYSIBM.SYSDATABASE

Ensure that the broker user ID (or the group of which it is a member) has been granted the necessary authorizations.

If the started task ID does not have execute authorization on the DSNACLI plan, you get the errors listed in the scenario. If ODBC tracing is turned on, you also see the following entries in the traceodbc file for the broker:

```
SQLAllocConnect( hEnv=1, phDbc=&1c6bacec )
SQLAllocConnect( )
    ---> SQL_ERROR
SQLError( hEnv=1, hDbc=0, hStmt=0, pszSqlState=&19ca445c, pfNativeError=&19ca4458,
pszErrorMsg=&1c3d9e50, cbErrorMsgMax=1024, pcbErrorMsg=&19ca4444 )
SQLError( pszSqlState="58004", pfNativeError=-99999, pszErrorMsg="{DB2 for OS/390}
{ODBC Driver}
SQLSTATE=58004
ERRLOC=2:170:6
RRS "CREATE THREAD" failed using DB2 system:DFL0 and Plan:DSNACLI RC=08 and REASON=00f30034",
pcbErrorMsg=159)
    ---> SQL_SUCCESS
SQLError( hEnv=1, hDbc=0, hStmt=0, pszSqlState=&19ca445c, pfNativeError=&19ca4458,
pszErrorMsg=&1c3d9e50, cbErrorMsgMax=1024, pcbErrorMsg=&19ca4444 )
SQLError( pszSqlState="42505", pfNativeError=-922,  pszErrorMsg="{DB2 for OS/390}
{ODBC Driver}
DSNT408I
SQLCODE = -922, ERROR:  AUTHORIZATION FAILURE: PLAN ACCESS ERROR.
REASON 00F30034
DSNT418I
SQLSTATE = 42505 SQLSTATE RETURN CODE   DSNT415I
SQLERRP = DSNAET03 SQL PROCEDURE DETECTING ERROR    ERRLOC=2:170:8", pcbErrorMsg=371 )
    ---> SQL_SUCCESS
```

If the started task ID does not have SELECT authorization on tables SYSIBM.SYSSYNONYMS, SYSIBM.SYSDATABASE and SYSIBM.SYSTABLES, and if ODBC tracing is turned on, you see the following type of entry in the traceodbc file:

```
SQLTables( hStmt=1, szTableQualifier=Null Pointer, cbTableQualifier=0,
szTableOwner="MA15USR", cbTableOwner=-3, szTableName=Null Pointer,
cbTableName=0, szTableType=Null Pointer, cbTableType=0 )
SQLTables( )
    ---> SQL_ERROR
```

Diagnostic message ICH408I is issued on z/OS when your broker fails to start:

- **Scenario:** The following diagnostic message is written to the SDSF SYSLOG on z/OS when your broker fails to start:

```
 ICH408I USER(MA10USR ) GROUP(TSOUSER ) NAME(OTHER, A N (ANO) 484
   /argo/MA10BRK/ENVFILE
 -                                          --TIMINGS (MINS.)--
  ----PAGING COUNTS---
 -JOBNAME  STEPNAME PROCSTEP    RC   EXCP    CPU    SRB  CLOCK   SERV  PG
    PAGE   SWAP    VIO SWAPS
   CL(DIRSRCH ) FID(01D7D3E2E3F1F9002D08000000000003)
   INSUFFICIENT AUTHORITY TO LOOKUP
   ACCESS INTENT(--X)  ACCESS ALLOWED(OTHER ---)
```

- **Explanation:** The started task ID, under which the broker runs, needs to be in an RACF® or USS group that has rwx permissions on the broker directory. As an example, consider a broker that is created under directory /argo/MA00BRK. It

runs under started task ID MA00USR. Issuing the ls -al command from the root directory / to find the permission bit settings on /argo reveals:

```
drwxrwx---  5 BPXROOT  ARGOUSR      8192 Jul 30  13:57  argo
```

If you issue the id MA00USR command to find the group membership of started task ID MA00USR you see:

```
uid=14938(MA00USR) gid=5(TSOUSER) groups=229(ARGOUSR)
```

These results show that the started task ID MA00USR potentially has rwx permissions on subdirectories to /argo, because these are set for both the user and the group that is associated with MA00USR. If the permissions are not set correctly, you see the type of diagnostic message shown in the scenario.

- **Solution:** Make sure that the started task ID, under which the broker runs, is in an RACF or USS group that has rwx permissions on the broker directory.

Abend code 047 is issued with a diagnostic message:

- **Scenario:** On z/OS, your broker generates abend code 047 when you try to start it, and the following diagnostic message is written to the SDSF SYSLOG:

```
IEA995I SYMPTOM DUMP OUTPUT 463
SYSTEM COMPLETION CODE=047
 TIME=10.53.47  SEQ=00419  CPU=0000  ASID=008E
 PSW AT TIME OF ERROR  078D0000   98D09E52  ILC 2  INTC 6B
   ACTIVE LOAD MODULE           ADDRESS=18D08828  OFFSET=0000162A
   NAME=SPECIALNAME
        61819987 968995A2 A3618499 89A58599 */argoinst/driver*
        F1F46D82 96858261 A4A29961 93979761 *14_boeb/usr/lpp/*
        A6949889 61828995 61828997 89948189 *wmqi/bin/bipimai*
        95                                  *n               *
   DATA AT PSW  18D09E4C - 58109948  0A6B5820  B8E95020
   GPR  0-3  00000000  0000003C  00000000  00000000
   GPR  4-7  18D10300  18D115F0  00000013  00000004
   GPR  8-11 18D111CF  18D101D0  18D0BBBE  18D0ABBF
   GPR 12-15 98D09BC0  18D101D0  98D09E22  00000000
 END OF SYMPTOM DUMP
```

- **Explanation:** System completion code 047 means that an unauthorized program issued a restricted Supervisor Call (SVC) instruction. The diagnostic message also indicates that the program in error was bipimain.
- **Solution:** When you install WebSphere Event Broker, issue the command extattr +a bipimain from the bin directory of the installation path to give APF authorization to program bipimain.

Abend code 047 is issued in bin/bipimain:

- **Scenario:** You get abend code 047 in bin/bipimain.
- **Explanation:** If you see this abend code, check the APF authorization.

  If your broker fails to start, submit a job to run the broker procedure. For example:

```
//MQP1BRK  JOB 1,MSGCLASS=H
//S1  EXEC PROC=MQP1BRK
```

  If this works, the started task user ID is probably unable to access resources.
- **Solution:** To help identify the cause of the problem, carry out the following procedure:
  1. Submit the CVP job to run with the user ID of the broker started task, and resolve any problems that arise.
  2. Submit the CVP job using the started task user ID, by putting `user=xxxxx,password=yyyy` on the job card temporarily or by using a

surrogate facility of the External Security Manager. To set up the surrogate facility with RACF, create a SURROGAT class profile and give access to either the user only or a group only. For example, use the following RACF commands:

```
RDEFINE SURROGAT MQP1BRK.SUBMIT OWNER(mqp1BRK) AUDIT(ALL(READ))
PE MQP1BRK.SUBMIT CLASS(SURROGAT) ID(myid) ACC(READ)
```

When you issue these commands, it allows `myid` to put `USER=MQP1BRK` in the job card of a job, which then runs under the MQP1BRK user ID.

Database error message BIP2322 is issued when you start or stop a broker on Linux or UNIX:

- **Scenario:** The following error message is displayed when you start or stop a broker on a Linux or UNIX system:

  ```
  Database Error. DataDirect ODBC driver License file not found.
  ```

- **Explanation:** When you start or stop a broker from a user ID other than the one that created the broker (that is, not the service user ID), a problem might occur where WebSphere Event Broker cannot find the necessary DataDirect license file (*install_dir*/wmqi/IVUK.LIC).
- **Solution:** Insert a link to the DataDirect license file into the /usr/local/lib directory.

Error message BIP2228 is issued when you try to start a second broker on Linux or UNIX:

- **Scenario:** Error message BIP2228, which mentions `semctl` in the syslog, is displayed when you try to start a second broker on Linux or UNIX.
- **Explanation:** This error typically indicates a permissions problem with a semaphore used by WebSphere Event Broker. A semaphore is created when the first broker starts after a reboot (or after an initial installation), and only members of the semaphore creator's primary group can access this semaphore. This problem is a consequence of the UNIX System V IPC primitives that are used by WebSphere Event Broker.

  The BIP2228 message is logged by any broker that is started by a user who is not a member of the semaphore creator's primary group. The broker tries to access the semaphore, but fails with a permissions-related error. The broker then terminates with the BIP2228 message.
- **Solution:** Avoid this problem by ensuring that all user IDs used to start WebSphere Event Broker have the same primary group. If this is impractical, ensure that all user IDs are members of each others' primary groups. Contact your IBM® Support Center for further assistance.

MQIsdp client connection is refused by the broker:

- **Scenario:** When a new MQIsdp client tries to connect to the broker, its connection is refused.
- **Explanation:** MQIsdp Client ID fields must be unique. If a client sends a CONN packet that contains the same Client ID as a currently-connected client, the behavior is undefined.
- **Solution:** Ensure that Client IDs are unique.

Error messages BIP2604 and BIP2624 are issued when you start a broker or a new message flow:

- **Scenario:** The following messages are written to the USS syslog on z/OS when your execution group, or a newly deployed or started message flow, fails to start:

```
(PMQ1BRK.default)[8]BIP2624E: Unable to connect to queue manager 'PMQ5':
        MQCC=2; MQRC=2025; message flow node 'ComIbmMQConnectionManager'
(PMQ1BRK.default)[8]BIP2604E: Node failed to open WebsphereMQ queue
       'INPUT1' owned by queue manager 'PMQ5': completion code 2; reason code 2025
```

- **Explanation:** The WebSphere MQ return code of 2025 indicates that the maximum number of concurrent connections has been exceeded. On z/OS, a typical cause of this problem is the setting for IDBACK in the WebSphere MQ CSQ6SYSP macro.
- **Solution:** See the *z/OS System Setup Guide* section of the WebSphere MQ Version 6 information center online or the WebSphere MQ Version 5.3 book on the WebSphere MQ library Web page for information about setting the IDBACK variable.

When you start the broker through the DataFlowEngine, it cycles continually:

- **Scenario:** When you start the broker through the DataFlowEngine, it continually cycles, starts and stops, and errors BIP2801 and BIP2110 appear in the log:

```
Unable to load implementation file '/opt/IBM/DistHu b/v2/lib/libdhbNBIO.so',
rc=The file access permissions do not allow the specified action.
Message broker internal program error.
```

- **Explanation:** The permissions on /opt/IBM have a value of 700, meaning that the broker ID cannot read the disthub files.
- **Solution:** Set the permissions on /opt/IBM to 755, which is rwxr-xr-x.

The broker terminates with an abend on startup on AIX:

- **Scenario:** The broker terminates with an abend on startup. Error message BIP2228 is issued, stating that the /usr/java131 directory cannot be found. In addition, a number of abend files are generated.
- **Explanation:** The Java Runtime Environment (JRE) is a prerequisite on AIX and must be installed in a directory named /usr/java131 in order to start a broker.
- **Solution:** Make sure that you install the JRE in the /usr/java131 directory before you start a broker. Alternatively, create a symbolic link between the /usr/java131 directory and the directory where the JRE was installed.

You have changed your logon password and cannot start your Configuration Manager or broker:

- **Scenario:** You have changed your logon password and when you start the Configuration Manager or broker, you see the following error message:

```
Unable to start the component.
The component could not be started using the service userid that was supplied when
the component was created. Ensure that the service userid and password are still
valid. Ensure that the service userid has permission to access all of the products
directories, specifically the 'bin' and 'log' directories.  Check for system
messages (on Windows NT this would be the application event log).
```

- **Solution:** Change properties of your broker, Configuration Manager, and possibly DB2 services by performing the following steps:
  1. Change your broker using the command:

```
mqsichangebroker brokername -i ServiceUserID -a ServicePassword -p DataSourcePassword
```

     For example, to change your logon password to `user1pwd` for user ID `user1` on the broker called `WBRK_BROKER`, use the command:

```
mqsichangebroker WBRK_BROKER -i user1 -a user1pwd -p user1pwd
```

  2. Change your Configuration Manager using the command:

```
mqsichangeconfigmgr  -i ServiceUserID -a ServicePassword -p DataBasePassword
```

For example, to change your logon password to `user1pwd` for user ID `user1` on the Configuration Manager, use the command:

```
mqsichangeconfigmgr -i user1 -a user1pwd -p user1pwd
```

3. Change the logon properties on your DB2 services:
   a. Click **Start** → **Settings** → **Control Panel** → **Administrative Tools** → **Services**
   b. Right-click **DB2 services**, click **Properties**, and click the **Log On** tab.
   c. Change your account and password to reflect the database passwords used in the commands above.
   d. Stop and start your services.
   e. Restart your computer.
4. Restart your components.

Broker on Windows platform fails to start with message BIP2818 and an abend:

- **Scenario:** You have more than one broker defined on a single Windows system, and your brokers use different service IDs. You can start brokers that share a common service ID, but if you try to run a brokers with a different service ID at the same time you see BIP2818 message in Event Viewer saying "Failed to create semaphore". An abend file is also created (message BIP2110). The abend shows function CreateMutexW and errno 17.
- **Explanation:** On Windows systems, brokers that run on the same machine must all use the same service ID.
- **Solution:** Revise your broker configuration so that all brokers on a single Windows system run with the same service ID.

Resolving problems when starting a Configuration Manager

This topic contains advice for dealing with some common problems that can arise when you start a Configuration Manager.

The broker or Configuration Manager fails to start because there is not enough space in the Java TMPDIR directory:

- **Scenario:** The broker or Configuration Manager fails to start and an error message indicates that insufficient space is available.
- **Explanation:** The broker and Configuration Manager use Java JAR files. When the broker or Configuration Manager starts, the Java runtime environment extracts the JAR files into a temporary directory, Java TMPDIR. On Linux, UNIX, and z/OS systems, the TMPDIR directory is typically /tmp; on Windows systems, it is c:\temp. If this directory is not large enough to hold the JAR files, the broker or Configuration Manager does not start.
- **Solution:** Use one of the following methods to specify the location of this temporary JAR directory:
  - Use the environment variable TMPDIR.
  - Set the system property java.io.tmpdir.

  Allow at least 50 MB of space in this directory for broker components. and 10 MB of space for Configuration Manager components. You might need more space if you deploy large user-defined nodes or other JARs to the broker component.

You have changed your logon password and cannot start your Configuration Manager or broker:

- **Scenario:** You have changed your logon password and when you start the Configuration Manager or broker, you see the following error message:

```
Unable to start the component.
The component could not be started using the service userid that was supplied when
the component was created. Ensure that the service userid and password are still
valid. Ensure that the service userid has permission to access all of the products
directories, specifically the 'bin' and 'log' directories.  Check for system
messages (on Windows NT this would be the application event log).
```

- **Solution:** Change properties of your broker, Configuration Manager, and possibly DB2 services by performing the following steps:
  1. Change your broker using the command:

```
mqsichangebroker brokername -i ServiceUserID -a ServicePassword -p DataSourcePassword
```

     For example, to change your logon password to `user1pwd` for user ID `user1` on the broker called `WBRK_BROKER`, use the command:

     ```
     mqsichangebroker WBRK_BROKER -i user1 -a user1pwd -p user1pwd
     ```

  2. Change your Configuration Manager using the command:

     ```
     mqsichangeconfigmgr  -i ServiceUserID -a ServicePassword -p DataBasePassword
     ```

     For example, to change your logon password to `user1pwd` for user ID `user1` on the Configuration Manager, use the command:

     ```
     mqsichangeconfigmgr -i user1 -a user1pwd -p user1pwd
     ```

  3. Change the logon properties on your DB2 services:
     a. Click **Start** → **Settings** → **Control Panel** → **Administrative Tools** → **Services**
     b. Right-click **DB2 services**, click **Properties**, and click the **Log On** tab.
     c. Change your account and password to reflect the database passwords used in the commands above.
     d. Stop and start your services.
     e. Restart your computer.
  4. Restart your components.

Error messages BIP1708 and BIP8283 are issued when you start a Configuration Manager on Windows:

- **Scenario:** Error messages BIP1708 and BIP8283 are issued when you start a Configuration Manager on Windows:

```
BIP8283E:  Configuration_Manager_name ) User Name Services are disabled.

The Configuration Manager communicates with the User Name Server in order to procure
user and group information for configuring topic ACLs.
No action unless you want to manage topic ACLs. If you want to manage topic ACLs
you  must reconfigure the Configuration Manager by providing a queue manager name
for the User Name Server.

BIP1708E:  Configuration_Manager_name ) User Name Services are disabled.
The Configuration Manager communicates with the User Name Server in order to procure user
and group information for configuring topic ACLs.
No action unless you want to manage topic ACLs. If you want to manage topic ACLs
you must reconfigure the Configuration Manager by providing a queue manager name
for the User Name Server.
```

- **Explanation:** These two errors are issued if the Windows component File and Printer Sharing for Microsoft® Networks is not installed on this system, or was uninstalled after you installed WebSphere Event Broker. The Configuration Manager uses APIs that are provided by the File and Printer Sharing component to implement user access authorizations.
- **Solution:** Install the Windows File and Printer Sharing component. If you do not want to use this component to allow users to share printers or files on this

system, disable the component. The Configuration Manager requires the component to be installed, but does not need the component to be enabled to complete its access checks.

Resolving problems when starting other resources

Advice for dealing with some common problems that can arise when you start resources other then a broker or the Configuration Manager.

Resources terminate during startup:

- Windows **Scenario:** The following error message is displayed when you start a broker, the Configuration Manager, or the User Name Server on Windows:

  ```
  ServiceName - DLL initialization failure Initialization of the
  dynamic link library c:\windows\system32\user32.dll failed.
  The process is terminating abnormally.
  ```

- **Explanation:** This error is issued by Windows when it fails to start a service because it has insufficient storage.
- **Solution:** This is an operating system problem. Information on how to recover from this problem is available in the Microsoft Developer Network (MSDN). You can access MSDN on the Web at `http://msdn.microsoft.com`.

Resources hang at startup on Windows:

- Windows **Scenario:** You try to start the Configuration Manager or a broker on Windows, but nothing happens in the Event log to show that a connection has started.
- **Explanation:** This problem is usually caused by processes having only one thread. To see if this is the case, check the Windows Task Manager. If either of the processes bipconfigmgr.exe or dataflowengine.exe has started, check the number of threads owned by the process. If the process has only one thread, you are likely to encounter this problem.
- **Solution:**
  1. Shut down the broker using the mqsistop command and end the process from within the Task Manager.
  2. From the Windows **Start** button, click **Settings** → **Control Panel**
  3. Double-click **Administrative Tools**
  4. Double-click **Services** to open the Services window. From the list of available services, locate and right-click the broker or Configuration Manager resource you are attempting to start (the service name begins with IBM WebSphere Message Broker component). Click on **Properties** from the menu.
  5. Make a note of the **This Account** setting. Contact the system administrator to obtain the password associated with **This Account**, as these settings will be lost as you make changes.
  6. Select **System Account** as the **Log On As** option, and select **Allow Service to Interact with Desktop.** These selections allow you to see any hidden dialog messages. Click **OK** to accept the changes.
  7. Restart the resource that is failing and report any subsequent error messages and dialog box messages to your IBM Service Representative.
  8. When your IBM Service Representative has resolved this problem for you, make sure that you restore the **This Account**, **Password**, and **Confirm Password** entries to those that you used when you created the broker.

Error message BIP8048 is issued when you start a component:

- **Scenario:** Error message BIP8048 is issued when you start a component.

- **Explanation:** This message indicates that WebSphere MQ is not responding as expected when it tries to start the queue manager. This could be because the strmqm executable file is not present on UNIX or Linux systems, or the amqmdain executable file is not present on Windows, or permissions are incorrect.
- **Solution:** Check that your WebSphere MQ installation is fully functional:
  - On Windows, start the ″IBM MQSeries®″ service.
  - On UNIX or Linux, issue the **strmqm** command to start the queue manager that is associated with this component.

  If the check fails, your WebSphere MQ installation is incomplete. This error occurs typically because you have previously installed WebSphere Application Server, which installs an embedded WebSphere MQ component that does not support WebSphere Event Broker.

  Uninstall WebSphere Application Server, then install the full WebSphere MQ product that is provided with WebSphere Event Broker.

You experience problems with the default configuration:

- **Scenario:** You have run the Default Configuration wizard but there are problems with the default configuration.
- **Solution:** Use the Default Configuration wizard to remove the default configuration.

  If the Default Configuration wizard does not remove the default configuration completely, a Wizard failure window opens with instructions on where to find the wizard's log. Take the following steps:

  1. Follow any advice that is given in the Default Configuration wizard log and retry each step.
  2. If this fails, restart the computer and run the Default Configuration wizard again.
  3. If the wizard still does not remove the default configuration, remove each component manually by performing the following actions in the order shown:
     a. Issue the following commands to find out which components are installed:
        - mqsilist (lists the broker components: broker and Configuration Manager)
        - dspmq (lists WebSphere MQ components: the queue manager)
        - db2 list db directory (lists DB2 databases)

     b. In the workbench, delete the connection file LocalDomain.configmgr from the project LocalProject.
     c. In the workbench, delete the project LocalProject.
     d. Stop the default Configuration Manager by issuing the following command in the Command Console:

        ```
        mqsistop WBRK6_DEFAULT_CONFIGURATION_MANAGER
        ```

     e. Stop the default broker by issuing the command:

        ```
        mqsistop WBRK6_DEFAULT_BROKER
        ```

     f. Delete the default broker by issuing the command:

        ```
        mqsideletebroker WBRK6_DEFAULT_BROKER -w
        ```

        The **-w** parameter deletes all files related to this broker from the associated workpath.

     g. Delete the default Configuration Manager by issuing the command:

```
mqsideleteconfigmgr WBRK6_DEFAULT_CONFIGURATION_MANAGER -w -q -n
```

The **-q** parameter deletes the queue manager, so do not use this parameter if you have not already deleted the broker. The **-w** parameter deletes all files related to the Configuration Manager from the associated workpath and the **-n** parameter deletes the configuration repository.

h. Remove the database by issuing the following commands:

- On Linux:

```
db2 disconnect DEFBKDB6
db2 drop DEFBKDB6
```

- On Windows:

```
mqsideletedb DEFBKDB6
```

Do not use these commands unless you have already deleted the broker. The database must never be deleted before the broker.

i. If you need to remove the queue manager manually, issue the following commands:

```
endmqlsr -w -m WBRK6_DEFAULT_QUEUE_MANAGER
endmqm -i WBRK6_DEFAULT_QUEUE_MANAGER
dltmqm WBRK6_DEFAULT_QUEUE_MANAGER
```

4. If you still experience problems after removing the default configuration manually, contact your IBM Support Center.

A ″Not Found″ error is issued when you click a link to the Samples Gallery or a specific sample:

- **Scenario:** You see a ″Not Found″ error when you click a link to the Samples Gallery or to a specific sample, indicating that a URL is invalid.
- **Explanation:** The Samples Gallery and the sample applications are available only if you are viewing the information center from within the Message Brokers Toolkit. If you are viewing a standalone information center, you cannot access these resources.
- **Solution:** If you want to access the Samples Gallery, ensure that you are viewing the information center from within the Message Brokers Toolkit.

The Quick Tour displays as a blank window:

- **Scenario:** The Quick Tour displays as a blank window.
- **Explanation:** The Quick Tour requires Adobe® Flash Player (was Macromedia Flash Player) Version 7, and is displayed as a blank window if the Adobe Flash Player plug-in is not installed or is at the wrong level on the default browser.
- **Solution:** Download the Adobe Flash Player using one of the following methods:
  - Follow the link at the bottom of the Quick Tour page.
  - Select the specific Adobe Flash Player for your operating system and browser from the Adobe Flash Player download page.

  If the Quick Tour is not displayed correctly through the information center, you can view it by opening this file in a browser that has the Adobe Flash Player installed: evtoolkit\eclipse\plugins\com.ibm.etools.mft.quicktour.doc_6.0.0\quick_tour\quick_tour.html.

Error message BIP0832 is issued on startup:

- **Scenario:** The following error message is displayed on startup:

```
BIP0832E: A class java.io.FileNotFoundException exception occurred which
reported the following message: [filepath] (The process cannot access the
file because it is being used by another process). Resolve the reason of
error and try again.
```

- **Explanation:** An invalid WebSphere MQ Java Client trace output file has been specified on the Enqueue preferences screen.
- **Solution:**
  1. Open the Enqueue preferences screen by clicking **Windows** → **Preferences**, then clicking **Enqueue** on the left.
  2. In the **To file** field, specify a valid output file (one that is not read-only or already in use).

Your execution groups restart repeatedly:

- **Scenario:** Your execution groups restart repeatedly. The syslog might show an error, such as BIP2060.
- **Explanation:** The problem might be caused by:
  - The broker environment variables incorrectly defined
  - Incorrect Loadable Implementation Library directory permissions
  - Incorrect database permissions
  - Invalid user-written LILs
- **Solution:** Check:
  - Environment variables as described in Environment variables after installation
  - File and directory permissions as described in Checking the permission of the installation directory
  - Group memberships as described in Execution groups and Adding an execution group to a broker in the workbench

You cannot tell whether startup is complete on z/OS:

- **Scenario:** You cannot tell whether startup has completed on your z/OS system.
- **Solution:** To determine if startup is complete:
  1. Check the messages in the syslog. The following example shows a syslog entry for a startup of a broker with one execution group:

```
S STU3053
$HASP100 STU3053  ON STCINRDR
IEF695I START STU3053  WITH JOBNAME STU3053  IS ASSIGNED TO USER STU3
 , GROUP STCGROUP
$HASP373 STU3053  STARTED
+(broker53) 0 BIP9141W: The component was started.
+(broker53) 0 BIP2001I: The WebSphere Business Integration Message Broker
service has started
 process ID 33554919.
+(broker53.default) 0 BIP2201I: Execution Group started: process '67109
442
 196'; thread '0'; additional information 'broker53', '76eb7f2d-e800-00
 00-0080-974c271866d2', 'default', 'true', 'Q4A3', 'false', 'ARG5D651',
  'ARGO53', '********', 'false', 'f9c27f2d-e800-0000-0080-974c271866d2'
 , '/local/argo/driver/drv3', '/local/argo/tgrp53/broker53'.
+(broker53.default) 0 BIP9137I: A work manager has been registered by R
443
 RMS registration services, work manager name is BIP.STU30532.006710919
 6.IBM.UA  .
```

  2. Display the address spaces. The following example shows the display of a broker with one execution group:

```
D OMVS,U=STU3
BPXO040I 18.49.59 DISPLAY OMVS 446
OMVS     000E ACTIVE          OMVS=(68,05)
USER     JOBNAME  ASID        PID       PPID STATE   START     CT_SECS
STU3     STU30531 0069   33554696   33554919 HR    18.49.15      2.217
  LATCHWAITPID=         0 CMD=bipbroker broker53
STU3     STU30532 03FD   67109196   67109222 HR    18.49.23     19.816
  LATCHWAITPID=         0 CMD=DataFlowEngine broker53 76eb7f2d-e800-00
STU3     STU3053  0036   33554768   83886483 HRI   18.49.08       .653
  LATCHWAITPID=         0 CMD=bipservice Q4A3BRK AUTO
STU3     STU30532 03FD   67109222   33554696 1W    18.49.23     19.816
  LATCHWAITPID=         0 CMD=bipimain DataFlowEngine broker53 76eb7f2
STU3     STU3053  0036   83886483          1 1WI   18.49.08       .653
  LATCHWAITPID=         0 CMD=/local/argo/driver/drv3/bin/bipimain bip
STU3     STU30531 0069   33554919   33554768 1W    18.49.15      2.217
  LATCHWAITPID=         0 CMD=bipimain bipbroker broker53
```

The infrastructure main program bipimain is the first process in every address space. It starts bipservice, bipbroker, DataFlowEngine, or bipuns as the second process in the same address space. For each execution group, an additional address space is started. In this example, only one execution group is available.

Abend code 0C1 is issued when you try to start the DataFlowEngine on z/OS:

- **Scenario:** The first two WebSphere Event Broker address spaces start successfully, but the third address space (the DataFlowEngine) fails to start. The result is an 0C1 abend.
- **Explanation:** The DataFlowEngine address space is generated by the admin agent. If the region size is too small, either because an insufficient region size was specified in the procedure, or because the region size was overridden by the z/OS IEFUSI exit, the DataFlowEngine address space might fail to start, and fails with an 0C1 abend.
- **Solution:**
  1. Use the IPCS command on the dump (move the dump from the file system to a traditional MVS™ data set if required):

     ```
     verbx vsmdata,'noglobal,jobname(vcp0brk2)'
     ```

     where *vcp0brk2* is the name of the failing job.
  2. Find the string `'VSM LOCAL DATA AREA AT ADDRESS '`. The field ELIM gives the available region size and must be greater than 0C800000. If the field SMFEL is not ffffffff, the IEFUSI exit has changed the allowable region size. This value must also be greater than 0C80000.
  3. If you have an IEFUSI exit, check that the exit does not limit the broker address spaces. For example, a commonly-used field is OUCBSUBN. This can be STC or OMVS for the broker and indicates how the address space was started.

Error message BIP2604 with return code `MQRC_CONNTAG_IN_USE` is issued during the start of a message flow on z/OS:

- **Scenario:** Error message BIP2604 is issued with return code `MQRC_CONNTAG_IN_USE` during the start of a message flow on z/OS:

  ```
  BIP2604E: Node failed to open WebSphere MQ queue [queue name]
  owned by queue manager [queue manager name]
  ```

  This message is output every 30 minutes.

- **Explanation:** On z/OS, WebSphere MQ supports serialized access to shared resources, such as shared queues, through the use of a connection tag (serialization token) when an application connects to a queue manager that participates in a queue sharing group.

  In this case, a message flow node fails to connect to the indicated WebSphere Event Broker queue manager that is associated with the input queue, because the serialization token that it passed is already in use within the queue sharing group.

  This message is for information only. It indicates that serialization is occurring when two or more message flow input nodes try to connect to a queue manager to get messages from a shared queue.
- **Solution:** Check whether another instance of the message flow, or a flow using the same serialization token, is already running. If this is the case, no further action is needed. Otherwise contact your IBM Support Center.

You cannot start the User Name Server:

- Windows **Scenario:** You have created the User Name Server, specified the local system name for the Windows security domain, and used the same identifier for the service user ID. When you start the User Name Server, the following error message is issued:

```
BIP8026E: Unable to start the component.
The component could not be started using the service userid that was supplied when
the component was created. Ensure that the service userid and password are still
valid. Ensure that the service userid has permission to access all of the products
directories, specifically the 'bin' and 'log' directories.  Check for system
messages (on Windows NT this would be the application event log).
```

- **Explanation:** The configuration that you have specified is not supported by Windows. When you start the component that is created with this combination of parameters, a Windows system call incorrectly retrieves the SID (Security Identifier) of the computer rather than that of the account.
- **Solution:** To use your local account domain as the domain from which users and groups are drawn, specify a different user ID for the service user ID.

  For example, if you specify `NTSecurityDomainName` (on the **-d** parameter on the mqsicreatebroker command) as `NYBROKER` on local system `\\NYBROKER`, the service user ID on the **-i** parameter on the mqsicreatebroker command cannot be `nybroker`.

  Delete your User Name Server and re-create it using another user ID that has the correct authorizations.

A device allocation error is issued:

- **Scenario:** A device allocation error is issued.
- **Explanation:** A likely cause of this problem is that you do not have the correct permissions set on the component file system for the started task ID.
- **Solution:** Check the syslog; if the problem is caused by having incorrect permissions set for the started task ID, you often see an RACF authorization failure message, as shown in the following example.

```
ICH408I USER(TASKID1 ) GROUP(TSOUSER ) NAME(FRED (FRED) 959
 /argo/MA11BRK/ENVFILE
 CL(DIRSRCH ) FID(01D7C7E2E3F0F8000F16000000000003)
 INSUFFICIENT AUTHORITY TO LOOKUP
 ACCESS INTENT(--X)  ACCESS ALLOWED(OTHER ---)
IEE132I START COMMAND DEVICE ALLOCATION ERROR
IEA989I SLIP TRAP ID=X33E MATCHED.  JOBNAME=*UNAVAIL, ASID=00A8.
D J,BPXAS
IEE115I 11.13.04 2001.212 ACTIVITY 601
```

In this example, the started task ID does not have access to the file system component. The ICH408I message shows:

- The file that the task is trying to access
- The user ID that is trying to access the file
- The permissions that the ID is expecting to have (`INTENT` in the message)
- The permissions that the ID actually has (`ALLOWED` in the message)

You can use this information to correct the permissions and then reissue, in this example, the start broker request. This type of message is produced if the user who is issuing the command (which might be to start the broker, or to submit JCL to start one of the utility jobs) does not have the correct file system permissions for the file system component. Use the ICH408I information to rectify the problem.

Another possible reason for authorization failures is inconsistencies in the RACF definitions for a user ID in the MVS image and the OMVS segment. You should also check with your system administrator that the RACF ID that is used on MVS has a corresponding OMVS image created.

Error creating database when creating the default configuration on Windows:

- **Scenario:** The error occurs when you are using a domain user ID to create the default configuration, and you are using DB2.
- **Explanation:** You need an equivalent local account (that is, a local account with the same name and password as the domain account.) You only need this to use the default configuration; when creating components manually you can specify a database ID to use when accessing the database.
- **Solution:** Create a local account and try again.

Resolving problems that occur when migrating or importing resources

This topic contains advice for dealing with some common problems that can occur when you import or migrate resources.

Migration information is regularly updated on the WebSphere Event Broker support Web page with the latest details available. Look under the heading ″Solve a problem″ for the document ″Problems and solutions when migrating″.

- "Resolving problems when migrating or importing message flows" on page 33
  - "The **mqsimigratemsgflows** command gives unexpected results" on page 33
  - "Message flows that have been migrated from Version 2.1 cannot resolve subflows" on page 33
  - "A broker has been migrated from Version 2.1 to Version 6.0 and some message flows are now stopped" on page 33
  - "Execution groups retain many open queue handles" on page 33
- "Resolving problems when migrating or importing other resources" on page 34
  - "Error messages are issued after you migrate a Version 2.1 broker" on page 34
  - "The mqsimigratecomponents command fails with database error BIP2322" on page 34
  - "The **File** → **Import** menu provides only the option to import a compressed file inside an existing project" on page 35
  - "The source directory name contains spaces" on page 35

– "Migration to the Version 6.0 Configuration Manager repository is unsuccessful" on page 35
– "A migrated Configuration Manager does not start" on page 35
– "You redefined a broker in a migrated Configuration Manager and used the Configuration Manager Proxy to modify the broker reference's UUID" on page 36
– "You encounter problems when you try to connect to a migrated Configuration Manager from Message Brokers ToolkitVersion 6.0" on page 37

Resolving problems when migrating or importing message flows

This topic contains advice for dealing with some common problems that can occur when you import or migrate message flows.

The mqsimigratemsgflows command gives unexpected results:

- **Scenario:** You have imported your Version 2.1 or Version 5.0 message flows into the workbench using the **mqsimigratemsgflows** command, but the results are not what you were expecting.
- **Explanation:** The **mqsimigratemsgflows** command creates a report called mqsimigratemsgflows.report.txt in the directory from which you ran the command (typically the /eclipse directory). Read the report for details of the actions taken by the command, and for actions that you need to take to correct errors or inconsistencies.

Message flows that have been migrated from Version 2.1 cannot resolve subflows:

- **Scenario:** You have migrated a set of message flows from Version 2.1 to Version 6.0 and an error message is issued indicating that: `Subflow` *`message flow name`* `cannot be located`.
- **Explanation:** The subflow in the message flow has not been migrated correctly. The message flow was exported from Version 2.1 without the subflow, or it was exported into a different file to the subflow.
- **Solution:** Export everything from Version 2.1 into one large export file, then migrate the export file to Version 6.0. Alternatively, if the subflow has been imported but the message flow still shows the error, right-click the node that is causing the error and click **Locate subflow**.

A broker has been migrated from Version 2.1 to Version 6.0 and some message flows are now stopped:

- **Scenario:** You have migrated a broker from Version 2.1 to Version 6.0 and some message flows are now stopped.
- **Explanation:** This error can be caused by invalid ESQL in the deployed message flows that was not detected on the Version 2.1 broker.
- **Solution:** Remove the affected message flows from the execution group, then redeploy them. This provides more information about the error to help you to resolve it.

Execution groups retain many open queue handles:

- **Scenario:** You have migrated message flows that access WebSphere MQ queues to Version 6.0 from Version 2.1. The execution groups to which you have deployed the message flows maintain far more open handles than they did in Version 2.1
- **Explanation:** When a message flow opens a WebSphere MQ queue, for example in the MQInput or MQOutput node, the queue is held open until the message

flow is redeployed or removed from the execution group, or until the broker is stopped. In Version 2.1, the execution group queue cache is shared between threads in the message flow, even though queue handles are unique to a thread. When the number of queue handles reaches a threshold, older queue handles are considered for closing. The default threshold in Version 2.1 is 30.

In Version 6.0, each thread maintains its own queue cache, and the new default threshold per thread is 240. Therefore an execution group maintains far more open queue handles than in the previous version, particularly if it has started a large number of threads. (The same behavior is also true for Version 5.0, therefore message flows migrated from Version 5.0 are unaffected in this way when you migrate them.)

- **Solution:** To release the queue handles that are no longer in use, stop and restart the message flow. If you choose, you can reduce the default value with the following command:

```
mqsichangeproperties broker -e execgroup -o ComIbmMQConnectionManager -n queueCacheMaxSize -v new_value
```

You cannot set a value less than 10; if you do, the value reverts back to the default value 240. The new value is used for each thread in the execution group (not for the execution group itself).

If you change the value with the mqsichangeproperties command, you must stop and restart the broker for the new value to take effect.

Resolving problems when migrating or importing other resources

This topic contains advice for dealing with some common problems that can arise when you import or migrate resources other than message flows.

Error messages are issued after you migrate a Version 2.1 broker:

- **Scenario:** When you migrate a broker from Version 2.1 to Version 6.0, the following error messages are issued: BIP2238, BIP2239, BIP2280, BIP2281, and BIP2308.
- **Explanation:** These error messages are issued when there are clashes between Version 6.0 and Version 2.1 LIL files.
- **Solution:** These error messages do not indicate a problem, therefore you can ignore them. To suppress them, use the **-l** parameter of the mqsichangebroker command to change the LILPATH used by the broker. If all the Version 2.1 LILs are removed from the LILPATH, the error messages do not appear.

The mqsimigratecomponents command fails with database error BIP2322:

- **Scenario:** The mqsimigratecomponents command fails with database error BIP2322: `The 'CREATE TABLE' command is not allowed within a multi-statement transaction in the 'BROKER1' database.`
- **Explanation:** If you are using the mqsimigratecomponents command to migrate a broker that uses a Sybase database, you must modify the database to enable the Data Definition Language (DDL) that the command uses.
- **Solution:** Take the following steps:
  1. Log on to ISQL using a system administrator account.
  2. Issue the following series of commands:

```
1> use master
2> go
1> sp_dboption "BROKER1","ddl in tran",TRUE
2> go
Database option 'ddl in tran' turned ON for database 'BROKER1'.
Run the CHECKPOINT command in the database that was changed.
```

```
(return status = 0)
1> use BROKER1
2> go
1> checkpoint
2> go
```

where *BROKER1* is the name of the Sybase broker database.

The File → Import menu provides only the option to import a compressed file inside an existing project:

- **Scenario:** You have a compressed file that contains message flow projects. When you click **File → Import**, you have only the option to import the compressed file inside an existing project, but you want to re-create the message flow projects.
- **Solution:** When you export and import files, do not export or import the root directory, which is created for you because of the project file. When you export your message flow projects:
  1. Click **Create only selected directories**.
  2. Clear the project root folder.
  3. Select the files and subdirectories as required. The project root folder is selected, but is displayed as gray.

  Then, when you import the compressed file:
  1. Deselect the root (/) folder.
  2. Select the files and subfolders as required. The project root folder is selected, but is displayed as gray.

The source directory name contains spaces:

- **Scenario:** You want to use the mqsimigratemsgflows command, but your source directory contains embedded spaces.
- **Solution:** Put the directory name in double quotes. For example:

```
mqsimigratemsgflows -p MYFLOW01 -d "E:\Program Files\My message flows"
```

Migration to the Version 6.0 Configuration Manager repository is unsuccessful:

- **Scenario:** When you migrate a Version 2.1 or Version 5.0 Configuration Manager, migration of its database to the Version 6.0 Configuration Manager repository is unsuccessful.
- **Explanation:** If migration to the internal repository cannot be completed successfully, the internal repository remains empty, but it is primed with an empty topology. The Configuration Manager tries to start up after a failed migration attempt and reports any problems in the system log.
- **Solution:** To migrate configuration data that is contained within a DB2 database, take the following steps:
  1. Stop the Version 6.0 Configuration Manager.
  2. Delete the Version 6.0 Configuration Manager using the mqsideleteconfigmgr command, specifying the **-n** parameter, which causes the configuration repository to be deleted.
  3. Re-create the Version 6.0 Configuration Manager using the mqsicreateconfigmgr command, specifying the DB2 database that is required for migration.

A migrated Configuration Manager does not start:

- **Scenario:** You have migrated a Version 2.1 or Version 5.0 Configuration Manager to Version 6.0. You cannot start the Version 6.0 Configuration Manager for the

first time and the following event messages appear repeatedly in the Windows Application Log: 2001, 1223, 1224, 1228, 1205, 1007.

- **Explanation:** This error can occur if a large domain is being migrated; for example, the Configuration Manager manages a number of brokers, each of which has one thousand persistent subscriptions. The error is due to the DB2 **APP_CTL_HEAP_SZ** parameter being too small for the size of the domain that is being migrated.
- **Solution:** Increase the size of the **APP_CTL_HEAP_SZ** parameter by taking these steps:
  1. Stop the migrated Version 6.0 Configuration Manager by issuing the mqsistop command.

     When the Configuration Manager has stopped, a message appears in the Windows Event Viewer's Application Log with a source identifier of WebSphere Broker v6000 and event identifier 2002.
  2. For the database of the Version 2.1 or Version 5.0 Configuration Manager, complete the following steps at a DB2 command prompt:
     a. Connect to the database by issuing the command:

        ```
        db2 connect to cm_db user db_userid using db_password
        ```

     b. Determine the current value of the **APP_CTL_HEAP_SZ** parameter by issuing the command:

        ```
        db2 get db dfg for cm_db | find "APP_CTL_HEAP_SZ"
        ```

        The output looks like this:

```
Max appl. control heap size (4KB) (APP_CTL_HEAP_SIZE) = 128
```

     c. Increase the value of the **APP_CTL_HEAP_SZ** parameter by issuing the command:

```
db2 update db cfg for cm_db using APP_CTL_HEAP_SZ higher_value
```

     d. Issue the following command to force all applications to disconnect from the database to allow the changes to become effective:

        ```
        db2 force application all
        ```

     e. To force the change to be effective immediately, stop and restart the DB2 database by issuing these commands:

        ```
        db2stop
        db2start
        ```

  3. Start the migrated Version 6.0 Configuration Manager by issuing the mqsistart command.

When the Version 6.0 Configuration Manager has started successfully, the following event messages appear in the system's local error log with the source identifer WebSphere Broker v6000: 2001, 1223, 1224, 1228, 8255, 8280, and 1003.

If the error log still contains the original six event identifiers repeatedly (2001, 1223, 1224, 1228, 1205, and 1007), repeat steps 1 to 3, increasing the value that is assigned to the **APP_CTL_HEAP_SZ** parameter.

You redefined a broker in a migrated Configuration Manager and used the Configuration Manager Proxy to modify the broker reference's UUID:

- **Scenario:** During the migration of a Version 2.1 or Version 5.0 Configuration Manager from one computer to a second computer, you tried to redefine a broker in the new Configuration Manager and use the Configuration Manager Proxy to modify the broker reference's unique identifier (UUID).

- **Explanation:** This procedure is not valid for migration. To migrate a Version 2.1 or Version 5.0 Configuration Manager from one computer to another, use one of the following methods:
  - Use the mqsibackupconfigmgr command to back up the Configuration Manager on the source computer, and use the mqsirestoreconfigmgr command on the target computer to move the Configuration Manager.
  - Specify the source computer's DB2 data source on the mqsicreateconfigmgr command.
- **Solution:** Delete the Configuration Manager on the target computer and follow the instructions given in the appropriate topic:
  - Migrating from WebSphere MQ Event Broker Version 2.1 to WebSphere Event Broker Version 6.0
  - Migrating a Configuration Manager from Version 5.0

You encounter problems when you try to connect to a migrated Configuration Manager from Message Brokers ToolkitVersion 6.0:

- **Scenario:** You encounter problems when you try to connect to a migrated Configuration Manager from Message Brokers Toolkit Version 6.0.
- **Explanation:** This problem can occur if you have configured MCA users on the WebSphere MQ channels to the Configuration Manager.
- **Solution:** Configure access control lists (ACLs) for users who are running Message Brokers Toolkit Version 6.0. Follow the instructions in Considering security for the workbench.

Resolving problems when stopping resources

This topic contains advice for dealing with some common problems that can arise when stopping resources:

- "You cannot stop the broker queue manager"
- "The execution group terminates abnormally"

You cannot stop the broker queue manager

- **Scenario:** You are trying to use the WebSphere MQ **endmqm** command to stop a broker queue manager on a distributed system, but it does not stop.
- **Explanation:** In certain circumstances, attempting to stop a broker queue manager does not cause the queue manager to stop. This can happen if you have configured any message flows with multiple threads (you have set the message flow property Additional Instances to a number greater than zero).
- **Solution:** If you want to stop the broker's queue manager, stop the broker using the **mqsistop** command specifying the **-q** parameter. (The **-q** parameter is not available on z/OS.) This issues the WebSphere MQ **endmqm** command on your behalf in a controlled fashion that shuts down the broker and the queue manager cleanly.

The execution group terminates abnormally

- **Scenario:** Your execution group processes terminate abnormally.
- **Explanation:** When execution group processes terminate abnormally, they are restarted automatically by the bipbroker process. If an execution group process fails, it is restarted three times during each five-minute interval. The first five-minute interval begins when the execution group is first started. The broker database table BROKERAAEG contains a column labeled RetryInterval. This

column defines the restart time in minutes. Each row in this table represents an execution group. The row is populated on the first deployment of an execution group and RetryInterval defaults to 5.

If you remove the execution group from the broker configuration, deploy the broker configuration, and then later add the execution group and redeploy the broker configuration. The row is re-created and RetryInterval is set to its default value of 5.

- **Solution:** To change the default value:
  1. Stop the broker.
  2. Change the RetryInterval's value in the database table.
  3. Restart the broker.

Resolving problems when deleting resources

This topic contains advice for dealing with some common problems that can arise when deleting resources:

Problems when deleting a broker:

- "You cannot delete your broker after deleting your broker database"
- "The workbench is deploying to a deleted broker"
- "You cannot delete a broker following a disk failure, or the Configuration Manager contains references to a deleted broker" on page 39
- "The mqsideletebroker command hangs if the DB2 profile has not been run" on page 40
- "You cannot delete a broker after you change a user ID password" on page 40

Problems when deleting other resources:

- "Error message BIP8052 is issued when you delete the Configuration Manager" on page 41
- "You cannot delete a project from your workspace" on page 41
- "You are unable to remove a domain from the Domains navigator" on page 41
- "You cannot delete a Configuration Manager after you change a user ID password" on page 41

You cannot delete your broker after deleting your broker database

- **Scenario:** You deleted your broker database, and now you cannot delete your broker, and get warning message BIP8040.
- **Explanation:** The mqsideletebroker command checks for the broker database tables and issues warning message BIP8040 because the database is not there.
- **Solution:** You can work around this problem by creating a dummy database with the same name as the database that you deleted. You must also re-create the ODBC connection. Reissue the mqsideletebroker command and delete the dummy database.

The workbench is deploying to a deleted broker

- **Scenario:** You used the mqsideletebroker command to delete a broker. You deleted it from the topology, but the workbench is still deploying to the deleted broker.

- **Explanation:** This problem occurs because you have either performed an incorrect sequence of actions when you deleted the broker, or you have re-created a broker of the same name before the previous deletion has completed.
- **Solution:** These steps explain how to delete a broker correctly. If a step completed successfully when you tried to delete your broker previously, you do not need to repeat it.
  1. Delete the broker from the topology.
  2. In the Event Log editor, look in a response message from the Configuration Manager that states that the broker has updated its configuration and all references to the broker have been removed from the configuration repository. If you see an error, correct the problem that is reported and perform a delta deploy of the topology, which asks the broker to tidy up its resources.
  3. After the broker resources have been tidied and the broker has been removed from the Configuration Manager repository, stop the broker using the mqsistop command.
  4. Delete the broker by using the mqsideletebroker command.

If the deployment step fails because you have already deleted the broker by using the mqsideletebroker command, or the broker is otherwise unable to respond, remove the broker from the topology in the workbench. Ensure that the broker's SYSTEM.BROKER.ADMIN.QUEUE queue has been deleted, then start another topology deploy.

Alternatively, you can use the Configuration Manager Proxy API TopologyProxy.removeDeletedBroker to programmatically purge all references to the broker:

1. Start the Configuration Manager Proxy API Exerciser sample (which is available on the **Start** menu in Windows).
2. Connect to the Configuration Manager.
3. Right-click the topology object, and click **Remove references to a previously deleted broker**.

The broker should now have disappeared from the workbench.

Do not try to re-create a broker by using the mqsicreatebroker command until you are confident that it has been removed from the workbench. Similarly, do not reuse broker or queue manager names until you are confident that they are not in use elsewhere.

You cannot delete a broker following a disk failure, or the Configuration Manager contains references to a deleted broker

- **Scenario:** You have had a complete disk failure on your broker system. You have decided that you no longer need the broker that was on that system, therefore you now want to delete this broker from the topology.
- **Explanation:** Under normal circumstances, the broker is deleted from the topology, and the topology is redeployed before the broker is deleted when you run the mqsideletebroker command. Even if you delete the broker by using the mqsideletebroker command before the topology was updated, a redeploy of the topology detects that the broker has been deleted. This situation happens because the broker's control queues no longer exist on the broker's queue manager, and the redeploy can detect the missing queues.

However, when a system crashes, you cannot start the remote queue manager, nor the channels that connect to it. All deployments to the crashed system result in messages being left on the transmission queue for the remote queue manager. Therefore the workbench identifies that the failing deployment is caused by a remote queue manager that is not responding.

- **Solution:** To force the deletion of the lost broker from the topology:
  1. Create a new queue manager with the same name as that used by the broker that you have lost.
  2. Start the new queue manager.
  3. Connect the new queue manager to the queue manager associated with Configuration Manager: create its transmission queue and its sender and receiver channels.
  4. Start the channels.
  5. Remove the broker from the Topology editor, and save the topology. The Configuration Manager detects that the broker's queues no longer exist and it removes the broker from the configuration.
- **Alternative solution:** You can remove references to deleted brokers from the configuration by using the Configuration Manager Proxy API call TopologyProxy.removeDeletedBroker. You can also do this by using the Configuration Manager Proxy API Exerciser sample:
  1. Start the Configuration Manager Proxy API Exerciser sample (which is available on the **Start** menu in Windows).
  2. Connect to the Configuration Manager.
  3. Select the PubSubTopology object, then right-click and select **Remove references to a previously deleted broker**.
  4. Enter the name of the broker that you want to delete.

  CAUTION:
  Do not remove a working broker using this facility. By removing references to a broker from the configuration, you will make it unmanageable.

The mqsideletebroker command hangs if the DB2 profile has not been run

- **Scenario:** The mqsideletebroker command hangs if the DB2 profile has not been run.
- **Solution:** Run the DB2 profile, then re-issue the mqsideletebroker command.

You cannot delete a broker after you change a user ID password

- **Scenario:** You have changed the password that you specified for the broker user ID when you ran the mqsicreatebroker command to create a broker on a Windows system. You now want to delete the broker, but the mqsideletebroker command fails.
- **Solution:** You must also change the password that is associated with the Configuration Manager before you can delete it:
  1. Windows If you created the broker on a Windows system, change the password for the associated Windows service:
     a. Click **Start** → **Control Panel** and open **Administrative Tools**.
     b. Double-click **Services**. The services are listed in the **Services** window.
     c. Find the service for the broker that you want to change; its name is in the format `IBM WebSphere Message Broker component` *`broker_name`*. Double-click the service to open the service properties.

d. Select the **Log on** tab, enter and confirm the new password, and click **OK**. The password is updated.

This step is not required on Linux or UNIX systems.

2. Run the following command to change the password associated with the broker component:

   ```
   mqsichangebroker broker_name -a user_ID -p updated_password
   ```

3. Run the command to delete the broker:

   ```
   mqsideletebroker broker_name
   ```

   The broker is successfully deleted.

Error message BIP8052 is issued when you delete the Configuration Manager

- **Scenario:** Error message BIP8052 is displayed when you delete the Configuration Manager by using the mqsideleteconfigmgr command.
- **Explanation:** Error message BIP8052 indicates that the queue SYSTEM.BROKER.SECURITY.REPLY, one of the WebSphere MQ queues that are created specifically for the Configuration Manager cannot be deleted, and the action fails. This problem typically occurs because the queue is currently in use.
- **Solution:** Make sure that your Configuration Manager is stopped, then retry the mqsideleteconfigmgr command after a short interval.

You cannot delete a project from your workspace

- **Scenario:** You cannot delete a project from your workspace. You get error messages indicating that the containing directory cannot be deleted, or the project file is missing.
- **Explanation:** If you attempt to delete a project, and the directory that contains the project is in use, or you have any files that are contained within the project that have been opened by programs other than the workbench, some of the resources in the project are not deleted, but others, including the project file, might be deleted.
- **Solution:** Before you delete a project, make sure that other applications do not have the files open, and that you do not have an open command prompt located in the directory. To recover from this problem, manually delete any remaining files and directories from your workspace directory, then click **Delete** from the project in the workbench.

You are unable to remove a domain from the Domains navigator

- **Scenario:** You are unable to remove a domain from the Domains navigator.
- **Solution:** Ensure that you have disconnected from the domain, then delete the associated domain connection from the Broker Administration perspective. The domain targeted by this domain connection is then removed from the Domains navigator.

You cannot delete a Configuration Manager after you change a user ID password

- **Scenario:** You have changed the password that you specified for the Configuration Manager user ID when you ran the mqsicreateconfigmgr command to create a Configuration Manager. You now want to delete the Configuration Manager, but the mqsideleteconfigmgr command fails.
- **Solution:** You must also change the password that is associated with the Configuration Manager before you can delete it:

1. Windows If you created the Configuration Manager on a Windows system, change the password for the associated Windows service:
   a. Click **Start** → **Control Panel** and open **Administrative Tools**.
   b. Double-click **Services**. The services are listed in the **Services** window.
   c. Find the service for the Configuration Manager that you want to change; its name is in the format `IBM WebSphere Message Broker component` *`configmgr_name`*. Double-click the service to open the service properties.
   d. Select the **Log on** tab, enter and confirm the new password, and click **OK**. The password is updated.

   This step is not required on Linux or UNIX systems.
2. Run the following command to change the password associated with the Configuration Manager component:

   ```
   mqsichangeconfigmgr configmgr_name -a user_ID -p updated_password
   ```
3. Run the command to delete the Configuration Manager:

   ```
   mqsideleteconfigmgr configmgr_name
   ```

   The Configuration Manager is successfully deleted.

Resolving problems when developing message flows

Advice for dealing with some common problems that can arise when developing message flows.

"Resolving migration problems when developing message flows" on page 43

- "You do not know whether you need to migrate the Version 2.1 IBM primitives" on page 43
- "An error message is issued when you save a message flow that contains MQe nodes" on page 43

"Resolving implementation problems when developing message flows" on page 43

- "Messages are directed to the Failure terminal of an MQInput node" on page 43
- "Error message BIP2211 is issued in z/OS by the MQInput node" on page 44
- "Messages enter the message flow but do not exit" on page 44
- "Your execution group is not reading messages from the input queues" on page 46
- "The DataFlowEngine terminates while processing messages" on page 46
- "Your execution group hangs, or terminates with a core dump" on page 47
- "You experience problems when sending a message to an HTTP node's URL" on page 47
- "QoS1 and QoS2 SCADA messages are not being processed" on page 48

"Resolving trace problems when developing message flows" on page 48

- "You cannot determine which node is being referenced in your trace file" on page 48
- "You cannot see any alerts when you change user trace" on page 48

"Resolving appearance problems when developing message flows" on page 49

- "You rename a flow that contains errors, but the task list entries remain" on page 49

- "Terminals on a subflow get out of sync as changes are made" on page 49

"Resolving other problems when developing message flows" on page 49

- "The values of your promoted properties are lost after editing" on page 49
- "The Message Flow editor experiences problems when opening a message flow, and opens in error mode" on page 49
- "You get an exception when saving changes to a .tblxmi file" on page 50
- "You do not know when to use the MQeMbMsgObject object" on page 50

Resolving migration problems when developing message flows

This topic contains advice for dealing with some common migration problems that can arise during message flow development.

You do not know whether you need to migrate the Version 2.1 IBM primitives:

- **Scenario:** You do not know whether you need to migrate the message flow IBM primitives.
- **Explanation:** You do not need to migrate these primitives. The Version 2.1 primitives are supplied with WebSphere Event Broker Version 6.0. The Version 2.1 export file does not contain IBM primitives.

An error message is issued when you save a message flow that contains MQe nodes:

- **Scenario:** An error message is issued when you save a message flow that contains WebSphere MQ Everyplace® nodes.
- **Explanation:** Do not use message flows that contain MQeInput and MQeOutput nodes in WebSphere Event Broker Version 6.0. You can continue to use them to deploy from Version 6.0 to previous versions.
- **Solution:** Override the error and save the message flow. When you deploy the message flow to a previous version of the broker, it works as before. When you deploy the message flow to a Version 6.0 broker, create a suitable configuration to deal with it, as described in Configuring after MQe message flow deployment.

Resolving implementation problems when developing message flows

This topic contains advice for dealing with some common problems that can arise when running message flows.

Messages are directed to the Failure terminal of an MQInput node:

- **Scenario:** Messages that are received at a message flow are directed immediately to the Failure terminal on the MQInput node (if it is connected) or are rolled back.
- **Explanation:** When a message is received by WebSphere MQ, an error is signalled if the following conditions are all true:
  - The MQInput node requests that the message content is converted (the Convert property is set to `yes` on the node).
  - The message consists solely of an MQMD followed by the body of the message.
  - The message format, as specified in the MQMD, is set to MQFMT_NONE.

  This error causes the message to be directed to the Failure terminal.

- **Solution:** In general, there is no need to request WebSphere MQ to convert the message content, because WebSphere Event Broker processes messages in any code page and encoding that is supported by WebSphere MQ. Set the Convert property to `no` to ensure that messages flow from the MQInput node to successive nodes in the message flow.

Error message BIP2211 is issued in z/OS by the MQInput node:

- **Scenario:** The following error message is issued by the MQInput node, indicating an invalid attribute:

```
BIP2211: (Invalid configuration message containing attribute value [attribute value]
which is not valid for target attribute [target attribute name],
object [object name]; valid values are [valid values])
```

- **Explanation:** On z/OS, WebSphere MQ supports serialized access to shared resources, such as shared queues, through the use of a connection tag (serialization token) when an application connects to the queue manager that participates in a queue sharing group. In this case, an invalid attribute has been specified for the z/OS serialization token.
- **Solution:** Check that the value that is provided for the z/OS serialization token conforms to the rules as described in the *Application Programming Reference* section of the WebSphere MQ Version 6 information center online or the WebSphere MQ Version 5.3 book on the WebSphere MQ library Web page.

Messages enter the message flow but do not exit:

- **Scenario:** You have sent messages into your message flow, and they have been removed from the input queue, but nothing appears at the other end of the message flow.
- **Explanation:** There are several possible causes for this. Consider the following scenarios to try to identify the situation that is causing your failure:
  1. Check your message flow in the workbench.

     You might have connected the MQInput node Failure terminal to a successive node instead of the Out terminal. The Out terminal is the *middle* terminal of the three. Messages directed to an unconnected Out terminal are discarded.
  2. If the Out terminal of the MQInput node is connected correctly to a successive node, check the broker's local error log for an indication that message processing has been terminated due to problems. Additional messages give more detailed information.

     If the Failure terminal of the MQInput node has been connected (for example, to an MQOutput node), these messages do not appear.

     Connecting a node to a Failure terminal of any node indicates that you have designed the message flow to deal with all error processing. If you connect a Failure terminal to an MQOutput node, your message flow ignores any errors that occur.
  3. If the Out terminal of the MQInput node is connected correctly to a successive node, and the local error log does not contain any error messages, turn user tracing on for the message flow:
     a. Switch to the Broker Administration perspective.
     b. In the Domain view, select the message flow, and right-click. (You might have to refresh the view by clicking the **Refresh** icon.)
     c. Click **User Trace** → **Normal**.

     This action produces a user trace entry from only the nodes that the message visits.

On distributed systems, you can retrieve the trace entries using the mqsireadlog command, format them using the mqsiformatlog command, and view the formatted records to check the path of the message through the message flow.

z/OS For z/OS, edit and submit the BIPRELG job in COMPONENTPDS to execute the mqsireadlog and mqsiformatlog commands to process traces.

4. If the user trace shows that the message is not taking the expected path through the message flow, increase the user trace level to Debug by selecting the message flow, right-clicking it, and clicking **User Trace → Debug**.

   Resend your message into the message flow. Debug-level trace produces much more detail about why the message is taking a particular route, and you can then determine the reasons for the actions taken by the message flow.

   Do not forget to turn tracing off when you have solved the problem, otherwise performance will be adversely affected.

5. If the MQPUT command to the output queue that is defined on the MQOutput node is not successful (for example, the queue is full or **put** is disabled), the final destination of a message depends on:
   - Whether the Failure terminal of the MQOutput node is connected.
   - Whether the message is being processed transactionally (which in turn depends on the transaction mode setting of the MQInput node, the MQOutput node, and the input and output queues).
   - Whether the message is persistent or nonpersistent. When **transaction mode** is set to the default value of Automatic, message transactionality is derived from the way that it was specified at the input node). All messages are treated as persistent if **transaction mode**=yes, and as nonpersistent if **transaction mode**=no.

   In general, if a path is not defined for a failure (that is, neither the Catch terminal nor the Failure terminal of the MQInput node is connected):
   - Non-transactional messages are discarded.
   - Transactional messages are rolled back to the input queue for retry:
     - If the backout count of the message is less than the backout threshold (BOTHRESH) of the input queue, the message is retried and re-sent to the Out terminal.
     - When the backout count equals or exceeds the backout threshold, one of the following might happen:
       - The message is placed on the backout queue, if one is specified (using the BOQNAME attribute of the input queue.)
       - The message is placed on the dead-letter queue, if there is no backout queue defined or if the MQPUT to the backout queue fails.
       - If the MQPUT to the dead-letter queue fails, or if there is no dead-letter queue defined, then the message flow loops continuously trying to put the message to the dead-letter queue.
   - If a path is defined for the failure, then that path defines the destination of the message. If both the Catch terminal and the Failure terminal are connected, the message is propagated through the Catch terminal.

6. If your message flow uses **transaction mode**=yes on the MQInput node properties, and the messages are not appearing on an output queue, check the path of the message flow.
   - If the message flow has paths that are not failures (but that do not end in an output queue), then either:

- The message flow has not failed and the message is not backed out.
- The message flow is put to an alternative destination (for example, the Catch terminal, the dead-letter queue, or the queue's backout queue).

Your execution group is not reading messages from the input queues:

- **Scenario:** Your execution group has started, but is not reading messages from the specified input queues.
- **Explanation:** A started execution group might not read messages from the input queues of the message flows because previous errors might have left the queue manager in an inconsistent state.
- **Solution:** Complete the following steps:
  1. Stop the broker.
  2. Stop the WebSphere MQ listener.
  3. Stop the WebSphere MQ channel initiator.
  4. Stop the WebSphere MQ queue manager.
  5. Restart the WebSphere MQ queue manager.
  6. Restart the WebSphere MQ channel initiator.
  7. Restart the WebSphere MQ listener.
  8. Restart the broker.

The DataFlowEngine terminates while processing messages:

- **Scenario:** While processing a series of messages, the DataFlowEngine process size grows steadily without levelling off. This might cause the DataFlowEngine process to terminate if it cannot allocate any more memory and restart. The error message BIP2106 might be logged indicating the out of memory condition.

  In addition, if you are using DB2 on distributed systems, you might get the message:

  ```
  SQL0954C  Not enough storage is  available in the application heap to process the
  statement.
  ```

  z/OS On z/OS, an SQLSTATE of HY014 might be returned with an SQL code of -99999, indicating that the DataFlowEngine process has reached the DB2 z/OS process limit of 254 prepared SQL statement handles.
- **Explanation:** When a database call is made from within a message flow node, the flow constructs the appropriate SQL, and this is sent using ODBC to the database manager. As part of this process, the SQL statement is prepared using the SQLPrepare function, and a statement handle is acquired so that the SQL statement can be executed.

  For performance reasons, after the statement is prepared, the statement and handle are saved in a cache to reduce the number of calls to the SQLPrepare function. If the statement is already in the cache, the statement handle is returned so that it can be re-executed with newly bound parameters.

  The statement string is used to perform the cache lookup. By using hardcoded SQL strings that differ slightly for each message, the statement is not found in the cache, and an SQLPrepare function is always performed (and a new ODBC cursor is opened). When using PASSTHRU statements, use parameter markers so that the same SQL prepared statement can be used for each message processed, with the parameters being bound at runtime. This approach is more efficient in terms of database resources and, for statements that are executed repeatedly, it is faster.

  However, it is not always possible to use parameter markers, or you might want to dynamically build the SQL statement strings at runtime. This potentially leads

to many unique SQL statements being cached. The cache itself does not grow that large because these statements themselves are generally not big, but many small memory allocations can lead to memory fragmentation.

- **Solution:** If you encounter these types of situations, disable the caching of prepared statements by setting the MQSI_EMPTY_DB_CACHE environment variable to any value. When this environment variable has been created, the prepared statements for that message flow are emptied at the end of processing for each message. There might be a slight performance degradation because every SQL statement is prepared.

Your execution group hangs, or terminates with a core dump:

- **Scenario:** While processing a message, an execution group either hangs with high CPU usage, or terminates with a core dump. The stack trace from the core dump or abend file is large, showing many calls on the stack. Messages written to the system log might indicate "out of memory" or "bad allocation" conditions. The characteristics of the message flow in this scenario often include a hard-wired loop around some of the nodes.
- **Explanation:** When a message flow thread executes, it requires storage to perform the instructions that are defined by the logic of its connected nodes. This storage comes from the execution group's heap and stack storage. The execution of a message flow is constrained by the stack size, the default value of which differs depending on the operating system.
- **Solution:** If a message flow that is larger than the stack size is required, you can increase the stack size limit and then restart the brokers that are running on the system so that they use the new value. For information on setting the stack size for your operating system, see System considerations for message flow development.

You experience problems when sending a message to an HTTP node's URL:

- **Scenario:** Sending a message to an HTTP node's URL causes a timeout, or the message is not sent to the correct message flow.
- **Explanation:** The following rules are true when URL matching is performed:
  - There is one-to-one matching of HTTP requests to HTTPInput nodes. For each HTTP request, only one message flow receives the message. This is true even if two message flows are listening on the same URL. Similarly, you cannot predict which MQInput node that is listening on a particular queue will receive a message.
  - Messages are sent to wildcard URLs only if no other URL is matched. Therefore a URL of /* receives all messages that do not match any other URL.
  - Changing a URL in an HTTPInput node does not automatically remove the entry from the HTTP listener. For example, if a URL /A is used first, then changed to a URL of /B, the URL of /A is still used to listen on, even though there is no message flow to process the message. This incorrect URL does get removed after the broker has been stopped and restarted twice.
- **Solution:** To find out which URL the broker is currently listening on, look at the file wsplugin6.conf in the following location:
  - Linux UNIX On Linux and UNIX: /var/mqsi/components/*broker_name*/config
  - Windows On Windows: %ALLUSERSPROFILE%\Application Data\IBM\MQSI\components/*broker_name*/config where `%ALLUSERSPROFILE%` is the environment variable that defines the system working directory. The default directory is C:\Documents and Settings\All Users\Application Data\IBM\MQSI\components/*broker_name*/config.

If problems persist, empty wsplugin6.conf, restart the broker, and redeploy the message flows.

QoS1 and QoS2 SCADA messages are not being processed:

- **Scenario:** The message flow that handles SCADA messages is not processing QoS1 and QoS2 messages.
- **Explanation:** The broker database, in which these messages are stored, is causing deadlock.
- **Solution:** If you are using DB2 as your broker database, turn off DB2 next key locking to avoid these deadlock problems. Issue the following command in a DB2 command window to make this change:

  ```
  db2set DB2_RR_TO_RS=YES_OVERRIDE_RI
  ```

  Restart the DB2 database manager for this change to take effect.

Resolving trace problems when developing message flows

Follow this advice to deal with some common trace problems that can arise when you develop message flows.

- "You cannot determine which node is being referenced in your trace file"
- "You cannot see any alerts when you change user trace"
- "Data that the Trace node sends to the syslog on UNIX is truncated" on page 49

You cannot determine which node is being referenced in your trace file:

- **Scenario:** You have generated a service trace file for your message flow, to trace the path of a message. However, you cannot determine which node is being referenced in the trace file.
- **Explanation:** In the trace file you might see text such as:

  ```
  Video_Test#FCMComposite_1_1 ComIbmMQInputNode , Video_Test.VIDEO_XML_IN
  ```

  The elements in the text have the following meanings:

  Video_Test
  : is the name of the message flow

  FCMComposite_1_1
  : is the internal name for the node

  ComIbmMQInputNode
  : is the type of node

  VIDEO_XML_IN
  : is the node label, and is the name you see in your flow

  The number at the end of the internal name is incremented; for example, FCMComposite_1_4 would be the fourth node you added to your flow. In the example, this section of the trace is referring to the first node in the message flow.

You cannot see any alerts when you change user trace:

- **Scenario:** You cannot see any alerts for an execution group or message flow in the Alerts viewer when you use the mqsichangetrace command to change the user trace setting.
- **Explanation:** No alert is generated when an execution group runs user trace at normal or debug level. Alerts are generated only for message flow trace. For message flow trace, the workbench is not notified of trace changes that are initiated by the mqsichangetrace command.

- **Solution:** To see the alert, refresh the message flow, or disconnect and then reconnect to the domain.

Data that the Trace node sends to the syslog on UNIX is truncated:

- **Scenario:** You are using a Trace node on UNIX and have set the Destination property to Local Error Log. You send a message to the Trace node that consists of multiple lines, but the message that appears in the syslog is truncated at the end of the first line.
- **Explanation:** On UNIX, syslog entries are restricted in length and messages that are sent to the syslog are truncated by the new line character.
- **Solution:** To record a large amount of data in a log on UNIX, set the Destination property on the Trace node to File or User Trace instead of Local Error Log.

Resolving appearance problems when developing message flows

This topic contains advice for dealing with some common appearance problems that can arise when developing message flows:

You rename a flow that contains errors, but the task list entries remain:

- **Scenario:** When you rename a message flow in the workbench for which there are error icons (red crosses) displayed on nodes and connections, those error icons are removed when changes are made. However, the task list entries remain.
- **Solution:** Refresh the Message Flow editor by closing and reopening it.

Terminals on a subflow get out of sync as changes are made:

- **Scenario:** You have a message flow that contains subflow nodes. The name or number of terminals on the subflow gets out of sync when changes are made on the subflow itself. The same problem can happen with promoted properties.
- **Solution:** Refresh the Message Flow editor by closing and reopening it. Close the Message Flow editor that contains subflows while the subflows are being changed.

Resolving other problems when developing message flows

Follow these instructions for how to deal with problems that can arise when developing message flows, and that are not covered in the specific categories listed in ″Resolving problems when developing message flows″

The values of your promoted properties are lost after editing:

- **Scenario:** You edited a message flow using the Message Flow editor, and the values of your promoted properties are lost.
- **Explanation:** The values of promoted properties for nodes with more than a single subflow definition (that is, two identically named subflows in the same project reference path) are lost if the flow is edited and saved.
- **Solution:** To avoid this problem, ensure that each subflow in your project has a different name.

The Message Flow editor experiences problems when opening a message flow, and opens in error mode:

- **Scenario:** You attempt to open an existing message flow in the Message Flow editor and it opens in read-only error mode, displaying a list of parsing or loading errors. The message flow is not open and a message is displayed indicating that the message flow file is not valid.

- **Explanation:** The message flow file is unreadable or is corrupted, and the Message Flow editor cannot render the model graphically.
- **Solution:** Contact IBM Customer Support for assistance with the corrupted file.

You get an exception when saving changes to a .tblxmi file:

- **Scenario:** You are using the Database Table editor to make changes to a `.tblxmi` file, but you repeatedly get exceptions when you try to save the changes.
- **Explanation:** When you make changes to a `.tblxmi` file using the Database Table editor, the editor successfully saves the changes for the first save only. Subsequent saves after additional changes fail with an exception.
- **Solution:** To stop this from happening, close the Database Table editor after each save, and reopen it when you want to continue making changes.

You do not know when to use the MQeMbMsgObject object:

- **Scenario:** You do not know when to use the MQeMbMsgObject object rather than the base MQeMsgObject object.
- **Explanation:** Any messages from existing WebSphere MQ Everyplace applications (which are of type MQeMsgObject, or a subclass) can pass through WebSphere Event Broker unchanged. WebSphere Event Broker cannot parse the contents of these messages. However, the full representation of the data that has been dumped is available within the broker. Do not operate on this data.
- **Solution:** If you need to parse or modify the data contained within a WebSphere MQ Everyplace message, use an MQeMbMsgObject object. This is similar to standard WebSphere MQ messages: you can set fields such as correlation ID, and there is a field that can be parsed using any WebSphere Event Broker parser.

Resolving problems when deploying message flows

Advice for dealing with some common problems that can arise when message flows are deployed.

1. To debug problems when deploying, check the following logs:
   - The broker domain Event log
   - The local error log (the Windows Event log or the syslog)
   - The WebSphere MQ logs

   These logs might be on separate computers, and must be used in conjunction with the workbench output to ensure that the deployment was successful.

   Use the mqsilist command to check that the deployment was successful, or look in the Windows Event or broker domain Event log.
2. Use this checklist when you have deployment problems:
   - Make sure that the remote queue manager is running.
   - Make sure that channels are running.
   - Display the channel status to see if the number of system messages sent increases.
   - Check the channel from the remote end.
   - Check the queue manager name.
   - Determine whether the channel is a cluster channel.

Resolving common problems when deploying message flows

Advice for dealing with some common problems that can arise when message flows are deployed.

"Resolving problems that occur when preparing to deploy message flows"

- "Migrated message flows are not available to add to a broker archive file"
- "You cannot drag a broker archive file to a broker"
- "The compiled message flow file (.cmf) has not been generated" on page 52

"Resolving problems that occur during deployment of message flows" on page 52

- "The message flow deploys on the test system, but not elsewhere" on page 52
- "Your deployment indicates that the broker does not exist" on page 53
- "The Configuration Manager is trying to deploy to a broker that does not exist" on page 53
- "The Configuration Manager reports that it is out of memory" on page 53
- "Error messages are issued when you deploy to z/OS" on page 54
- "Expected serialization of input is not occurring for a shared queue that serves multiple instances of a message flow on z/OS" on page 54
- "MQe nodes do not work as expected after deployment to a WebSphere Event Broker Version 6.0 broker" on page 55
- "Error messages are issued when you deploy" on page 55

"Resolving problems that occur when canceling deployment of message flows" on page 57

- "When canceling a deployment, mqsilist shows a message flow in an execution group but the tooling does not" on page 57

"Resolving problems that occur after deployment of message flows" on page 57

- "You are not notified of the result of a deploy" on page 57
- "You do not receive confirmation that the deployment was successful" on page 58
- "You cannot see any deployed message flows" on page 58
- "A deleted broker remains in the domains navigator" on page 59

Resolving problems that occur when preparing to deploy message flows

This topic contains advice for dealing with some common problems that can arise during preparations to deploy message flows or message sets.

Migrated message flows are not available to add to a broker archive file:

- **Scenario:** You have migrated your Version 2.1 message flows into the workbench using mqsimigratemsgflows, but the message flows are not available to add to a broker archive (BAR) file.
- **Solution:** Migrated message flows are displayed in the workbench as a closed project. To make the message flows available to be added to a BAR file:
  1. Open the project by right-clicking it and clicking **Project → Open Project**
  2. Clean the project by right-clicking it and clicking **Clean**. This operation shows any errors, and makes the project resources available to be added to a BAR file.

You cannot drag a broker archive file to a broker:

- **Scenario:** You cannot drag a broker archive (BAR) file to a broker.
- **Explanation:** BAR files can be deployed only on an execution group. When you use drag and drop, you cannot drop the BAR file onto a broker in the Broker Topology editor or in the Domains navigator.
- **Solution:** Drop the BAR file onto an execution group in the Domains navigator, or select an execution group in the deploy dialog.

The compiled message flow file (.cmf) has not been generated:

- **Scenario:** The compiled message flow file (.cmf) has not been generated. Therefore, it is not added to the broker archive file and cannot be deployed.
- **Explanation:** When you create files, the overall file path length of those files must not exceed 256 characters, due to a Windows file system limitation. If you have a message flow that includes files that have a path length that exceeds 256 characters, the message flow cannot be compiled when you try to add it to a BAR file, and therefore cannot be deployed.
- **Solution:** To ensure that the path length does not exceed 256 characters, use names that are as short as possible for the following resources:
  - The installation path
  - Project names and broker schema names

Resolving problems that occur during deployment of message flows

Advice for dealing with some common problems that can arise during deployment of message flows.

- "The message flow deploys on the test system, but not elsewhere"
- "Your deployment indicates that the broker does not exist" on page 53
- "The Configuration Manager is trying to deploy to a broker that does not exist" on page 53
- "The Configuration Manager reports that it is out of memory" on page 53
- "Error messages are issued when you deploy to z/OS" on page 54
- "Expected serialization of input is not occurring for a shared queue that serves multiple instances of a message flow on z/OS" on page 54
- "MQe nodes do not work as expected after deployment to a WebSphere Event Broker Version 6.0 broker" on page 55
- "Error messages are issued when you deploy" on page 55

The message flow deploys on the test system, but not elsewhere:

- **Scenario:** The message flow that you have developed deploys on the test system, but not elsewhere.
- **Solution:** Carry out the following checks:
  - Ensure that you have verified the installation on the target system by creating and starting a broker, and deploying a single execution group. These actions show that WebSphere Event Broker and the broker database are in place.
  - Ensure that the broker archive (BAR) file's broker.xml file contains references to the correct resources for the new system.
  - If database resources are not accessible or authorized from the target system, the deploy fails. On distributed systems, ensure that your databases are defined as ODBC sources so that they can be accessed from WebSphere Event Broker. Also, set the broker's environment to allow access to the databases. (You might need to run a profile on Linux or UNIX systems.)

Your deployment indicates that the broker does not exist:

- **Scenario:** You have initiated deployment, but get an error indicating that the broker does not exist.
- **Explanation:** During the first stage of deployment, the Configuration Manager sends a configuration data stream to the SYSTEM.BROKER.ADMIN.QUEUE queue of each target broker. When the configuration data has been sent to all relevant brokers, control is returned to you. If an error is detected during this first stage of deployment, the deployment is abandoned; no configuration data is sent to any broker and an error message is written to the log.
- **Solution:** Check that you have specified the correct queue manager to be associated with the broker. If the queue manager is correct, it is possible that the deployment message cannot be delivered because the queue to which it was sent (SYSTEM.BROKER.ADMIN.QUEUE) cannot be found on the target queue manager. This problem returns a WebSphere MQ reason code of 2085.

  If that queue has been deleted, the broker environment has been damaged, therefore you must repair it.

The Configuration Manager is trying to deploy to a broker that does not exist:

- **Scenario:** The Configuration Manager is trying to deploy to a broker that does not exist.
- **Explanation:** This problem can occur for a number of reasons:
  - A broker component was deleted without first being removed from the workbench.
  - The name of the broker was mistyped in the workbench, creating a reference to a broker that does not physically exist.
  - The correct procedure for deleting a broker was not followed.
  - The broker does not seem to exist because of connectivity problems.
- **Solution:**
  1. Verify that the broker does not exist by using the **mqsilist** command to list resources on the computer where the broker should exist.
  2. Check your configuration to ensure that the broker does not exist on a different computer in the domain.
  3. If you are confident that the broker does not exist, and you want to remove the reference that is still in the workbench, follow the steps in Deleting a broker and Removing a broker from a broker domain.
  4. Remove completely all references to the broker from the Configuration Manager repository by removing the broker from the topology using the workbench. Ensure that the broker's SYSTEM.BROKER.ADMIN.QUEUE queue has been deleted, and initiate a further topology deployment.

     Alternatively, use the Configuration Manager Proxy API TopologyProxy.removeDeletedBroker(String) to programmatically purge all references to the broker:

     a. Start the Configuration Manager Proxy API Exerciser sample (which is available on the **Start** menu in Windows).
     b. Connect to the Configuration Manager.
     c. Right-click the topology object and click **Remove references to a previously deleted broker**.

     See Viewing and managing a broker domain using the Configuration Manager Proxy API Exerciser for more information.

The Configuration Manager reports that it is out of memory:

- **Scenario:** The broker archive file is not deployed because the Configuration Manager claims to be out of memory, as indicated by error message BIP1111.
- **Explanation:** The typical cause of this problem is that the Java Virtual Machine (JVM), which the Configuration Manager uses to perform the deployment logic, has run out of temporary storage space.
- **Solution:** Change the **maxJVMHeapSize** parameter on the Configuration Manager to increase the amount of memory that is allocated to its JVM:

  ```
  mqsistop ConfigMgr
  mqsichangeconfigmgr -j [newJVMHeapSizeMb]
  mqsistart ConfigMgr
  ```

  where *ConfigMgr* is the name of your Configuration Manager and *newJVMHeapSizeMb* is the size in megabytes to allocate; the minimum value is 64.

  Try to deploy the operation again. If the problem still occurs, further increase the value of the **maxJVMHeapSize** parameter.

Error messages are issued when you deploy to z/OS:

- **Scenario:** The following messages are written to the log when you deploy to z/OS:

  ```
  +(MQ05BRK) 0 BIP2070E: A problem was detected with WebSphere MQ while issuing
  MQPUT for WebSphere MQ queue SYSTEM.BROKER.ADMIN.REPLY, WebSphere MQ queue
  manager QM_01.  MQCC=2, MQRC=2030.
  +(MQ05BRK) 0 BIP2068E: The broker was unable to put an internal configuration
  message to message queue SYSTEM.BROKER.ADMIN.REPLY.
  ```

- **Explanation:** The transmission queue is not large enough for the messages that are issued by WebSphere Event Broker.
- **Solution:** See the WebSphere MQ documentation for details on how to increase the size of the transmission queue.

Expected serialization of input is not occurring for a shared queue that serves multiple instances of a message flow on z/OS:

- **Scenario:** Expected serialization of input is not occurring for a shared queue that serves multiple instances of a message flow on z/OS.
- **Explanation:** On z/OS, WebSphere MQ supports serialized access to shared resources, such as shared queues, through the use of a connection tag (serialization token) when an application connects to the queue manager that participates in a queue sharing group.

  This problem can occur for a number of reasons:

  – You have assigned different serialization tokens to the input nodes of the flows that get messages from the shared queue.
  – The message flows are running in the same execution group. Serialization can only be effected between flows that are running in different execution groups, either on the same broker, or on different brokers whose queue managers participate in the same queue sharing group.
  – The queue managers on which the brokers are running are not participating in a queue sharing group, or not participating in the same queue sharing group.
- **Solution:** Carry out the following checks:

  – Check that the same z/OS serialization token has been configured for the MQInput nodes in each flow that use the shared queue.
  – Check that the message flows are running in different execution groups.

– Check that the broker queue managers are part of the same queue sharing group, and that no errors are reported from the queue managers in the z/OS system log (SDSF log).

MQe nodes do not work as expected after deployment to a WebSphere Event Broker Version 6.0 broker:

- **Scenario:** A message flow that contains WebSphere MQ Everyplace nodes works on a broker at a previous version, but does not work as expected when you deploy the message flow to a Version 6.0 broker.
- **Explanation:** When you deploy to a Version 6.0 broker, the MQe nodes are converted into MQ nodes. You can use MQeInput and MQeOutput nodes to deploy from WebSphere Event Broker Version 6.0 to previous versions.
- **Solution:** Do not use message flows that contain MQeInput and MQeOutput nodes in WebSphere Event Broker Version 6.0.
  1. Redesign your message flow to use MQ nodes instead of MQe nodes.
  2. Configure an external MQe gateway.

     As a route to migration, you can deploy the message flow, and create a suitable configuration to deal with it, as described in Configuring after MQe message flow deployment. See full migration details in Migrating a message flow that contains WebSphere MQ Everyplace nodes.

Error messages are issued when you deploy:

Additional error messages that might be generated during a deployment are explained in this section.

Message BIP1536

- **Scenario:** You have defined a Configuration Manager to run with one user ID, and you have defined a broker to run on a different computer with a different user ID. Deployment is successful but error message BIP1536 is issued when you deploy message flows to the broker.
- **Explanation:** The Configuration Manager is unable to register for internal subscriptions with the broker because the broker is running under one ID and the Configuration Manager is running under another ID. The broker and the Configuration Manager relay internal messages back and forth via publish/subscribe. These messages are carried through WebSphere MQ, which requires certain authorizations.
- **Solution:**
  – Ensure that the broker's user ID is a member of the mqm and mqbrkrs groups.
  – Define the broker's user ID on the computer where the Configuration Manager is running.
  – Define the user ID used by the Configuration Manager on the computer where the broker is running.
  – Ensure that all IDs are in lowercase so that they are compatible between computers.

Messages BIP1536 and BIP7017

- **Scenario:** Error messages BIP1536 and BIP7017 are displayed.
- **Explanation:** The Configuration Manager has a problem registering its internal subscriptions on topics that relate to broker status change. The Configuration Manager tries to register its internal subscriptions each time that you deploy a

complete configuration. The cause of the problem is given by the message BIP7017, which indicates that you are running with a User Name Server configured, but that the broker to which you are deploying does not have the Configuration Manager service user ID in its user cache.

- **Solution:** Make the following checks. When you identify and correct the problem, the subscriptions will be registered correctly the next time you deploy:
  - Ensure that the User Name Server is started.
  - Ensure that the WebSphere MQ channels between the User Name Server and the broker are started.
  - Ensure that the Configuration Manager service user ID is present in the User Name Server's domain.

Message BIP2045

- **Scenario:** Error message BIP2045 is displayed in the workbench after deployment.
- **Explanation:** Each broker is identified by a universally unique identifier (UUID), which is stored in the Configuration Manager when the broker is defined. The UUID is also stored in the broker when it receives its first deployment message. If the broker receives a deployment message that contains a different UUID, it rejects the deployment message and issues error message BIP2045. This problem can occur in the following circumstances:
  - You try to deploy from a second Configuration Manager; this scenario is not supported in WebSphere Event Broker.
  - You re-create the Configuration Manager but not the broker.
  - You experience problems with the WebSphere MQ channel while you are deleting and re-creating the broker.
- **Solution:** Ensure that you are not using more than one Configuration Manager to control a single broker. If your configuration is correct, follow the instructions in the BIP2045 error message to regain control of the broker by rebuilding the broker, and redeploying your message flow applications.

Message BIP2066

- **Scenario:** You have initiated a deployment request; for example, you have deployed a BAR file to an execution group. Error message BIP2066 is returned one or more times.
- **Explanation:** The deployment request was not acknowledged by the execution group before the broker timeout **ConfigurationChangeTimeout** plus the **InternalConfigurationTimeout** (default 60 seconds) expired.
- **Solution:** Increase these timeout values using the **-g** and **-k** parameters of the mqsicreatebroker or mqsichangebroker command. See Setting configuration timeouts for information about factors that affect timeouts and how to set appropriate values.

Message BIP2080

- **Scenario:** The broker has started an execution group; for example, if you have issued mqsistart for the broker, or an error has occurred and the execution group is being recovered. Error message BIP2080 is displayed one or more times.
- **Explanation:** The internal configuration request was not acknowledged by the execution group before the **InternalConfigurationTimeout** (default 60 seconds) expired.

- **Solution:** Change the configuration timeout using the **-k** parameter of the mqsicreatebroker or mqsichangebroker command. See Setting configuration timeouts for information about factors that affect timeouts and how to set appropriate values.

Message BIP2242

- **Scenario:** Error message BIP2242 is displayed.
- **Explanation:** The deployment request (configuration change) was not accepted before the broker timeout **ConfigurationChangeTimeout** expired. This configuration timeout must be long enough for the message flow to complete processing its current message and then accept the deploy request; the default is 300 seconds.
- **Solution:** Set the configuration timeouts using the **-g** and **-k** parameters of the mqsicreatebroker or mqsichangebroker command.

Message BIP7053S

- **Scenario:** When you deploy to a broker, error message BIP7053S is displayed.
- **Explanation:** This error occurs in a multi TCP/IP stack environment, and indicates that the UNIX System Services (USS) TCP/IP environment has not been set up correctly.

  WebSphere Event Broker uses USS functions to obtain the hostname for a particular system. The following error message is displayed if the default hostname is not set up correctly in the USS environment:

  ```
  BIP7053S: Broker $SYS_mqsi 0 unexpected Java exception java.lang.Error:
  -2103399272!java.net.UnknownHostException :
  Hostname: Hostname
  ```

  The hostname that is reported in the error message is the one that has been returned to the broker as a result of the `gethostname` call.
- **Solution:** Ensure that the TCP/IP environment is configured correctly in USS.

Resolving problems that occur when canceling deployment of message flows

This topic contains advice for dealing with some common problems that can arise when canceling the deployment of message flows or message sets.

When canceling a deployment, mqsilist shows a message flow in an execution group but the tooling does not:

- **Scenario:** When canceling a deployment, mqsilist shows a message flow in an execution group but the tooling does not show it. You have canceled deployment to an execution group, but the deployment takes place and is shown by mqsilist.
- **Solution:** Delete the execution group. mqsilist still shows the old execution group if it is the only one that has been deployed. If you re-create the execution group in the tooling, it is listed twice by mqsilist until the broker is recycled.

Resolving problems that occur after deployment of message flows

Advice for dealing with some common problems that can arise after deploying message flows or message sets.

You are not notified of the result of a deploy:

- **Scenario:** After you have closed the Broker Topology editor, saved your changes, and requested a deploy, you are not notified of the result of a configuration deploy.
- **Explanation:** When you close the editor, the notification link with the Configuration Manager is broken, so that the workbench is not notified of the deploy result. This is the same for any eventual failure notifications.
- **Solution:** To ensure that you receive any notification, save the editor without closing it and wait for the alert `Waiting for answer from Configuration Manager` to disappear. This alert is removed automatically when the editor receives all expected responses from the Configuration Manager.

You do not receive confirmation that the deployment was successful:

- **Scenario:** You have initiated deployment but have not received confirmation that the deployment was successful.
- **Solution:** Assume that communications from the workbench to the Configuration Manager are working, because the workbench gives an error immediately if this link fails. Follow these steps to find out if there is a communication problem that stops the receipt of deployment confirmation:
  1. Stop the broker using the mqsistop command.
  2. On the broker queue manager, check the SYSTEM.BROKER.ADMIN.QUEUE queue. If there are messages on this queue, there might be a broker error; look in the local error log on the broker's system. If the queue is empty, continue with the next step.
  3. Redeploy your message flow.
  4. Recheck the SYSTEM.BROKER.ADMIN.QUEUE queue on the broker queue manager. If the queue depth has not increased, there is a problem with the channel between the Configuration Manager and the broker; check the WebSphere MQ logs. If the queue depth has increased to one, continue with the next step.
  5. Stop the Configuration Manager, then start the broker.

     This allows the broker to process the configuration change and send a response. Wait for the broker to process the message; this might take some time. (If the message takes more than five minutes to process, there is a problem with the deployed message and failure messages are displayed in the local error log on the broker's system indicating the reason for the failure.)
  6. Check the SYSTEM.BROKER.ADMIN.REPLY queue on the Configuration Manager's queue manager. This queue should contain a response. If it does not, check that the channel from the broker to the Configuration Manager is running.
  7. Restart the Configuration Manager.
  8. Check that the messages are read from the SYSTEM.BROKER.ADMIN.REPLY queue on the Configuration Manager's queue manager. If they are not, check the local error log on the Configuration Manager's system for errors.
  9. Refresh the Event Log editor in the workbench. You should now get a reply.

You cannot see any deployed message flows:

- **Scenario:** After you deploy a broker archive (BAR) file, you cannot see any deployed message flows under the execution group.
- **Explanation:** After deployment, an information window should tell you whether deployment has been started successfully on the Configuration Manager. When the deploy has started, the time it takes to complete depends on your

configuration size and the workload on your computers. However, if, after a while, no deployment-related messages are displayed in the Event Log editor, it means that the broker has not responded to the deployment request.

- **Solution:** Check that the broker is running and that all WebSphere MQ queue managers and channels between the Configuration Manager and the broker are running. If the messages are present in the Event Log editor, see if any errors have occurred that mean that your changes were not activated.

A deleted broker remains in the domains navigator:

- **Scenario:** After you have deleted a broker and deployed changes, the broker remains in the domains navigator.
- **Explanation:** If you did not receive any failure response indicating that the deletion failed, and the Event Log editor has been automatically refreshed with a set of messages about the deployment result, the deletion succeeded but the workbench has not been notified of this deletion.
- **Solution:** For the broker topology to reflect the actual Configuration Manager content, refresh the broker topology. If the broker topology is open, click **Revert**.

Resolving problems when you use the workbench

This topic contains advice for resolving problems that might occur you use the workbench.

Resolving problems that occur when connecting:

- "Your broker is not recognized by the workbench" on page 60
- "You cannot connect to the Configuration Manager from the workbench or command line" on page 60
- "A Version 5.0 or Version 5.1 Message Brokers Toolkit is unable to connect to a Version 6.0 Configuration Manager using a domain user ID" on page 61
- "Connection to the Configuration Manager is slow" on page 62

Resolving error messages that occur when using the workbench:

- "Errors occur while you are using the workbench" on page 62
- "An error is issued when you access the help system" on page 62
- "The workbench displays an error message after the error has been fixed" on page 62
- "Error message BIP0800 is issued when you rename a domain object" on page 63
- "Error message BIP0935 is issued when you edit domain connection parameters" on page 63
- "Warning message BIP0876 is issued when you make changes in the Broker Topology editor" on page 63
- "The message `Unable to create part: filename.extension` is issued" on page 63

Resolving problems relating to the appearance of the workbench:

- "You cannot close a message dialog box" on page 64
- "The Domains navigator content is updated but you have not made any changes" on page 64
- "A server project containing a valid domain connection file is not displayed" on page 64
- "Some objects are not visible in the Domains Navigator" on page 64

- "You want to filter entries in the Tasks view" on page 64
- "Your message flow projects have changed their appearance" on page 65

Resolving non-specific problems when using the workbench:

- "Editors do not update automatically when the same file is open in multiple windows" on page 65
- "Deleting or closing a project takes a long time" on page 65
- "You are experiencing poor performance when working with large or complex projects" on page 65
- "You are considering moving database resources within the workbench" on page 65
- "You do not know how to return to the welcome page" on page 66

Resolving problems that occur when connecting to the workbench and the Configuration Manager

This topic contains advice for dealing with some common problems that can arise when you connect to the workbench.

Your broker is not recognized by the workbench:

- **Scenario:** Your broker is not recognized by the workbench.
- **Explanation:** Broker names in the workbench are case sensitive.
- **Solution:** When you identify a broker in the workbench, make sure that you use the same case that was used when it was created. On some operating systems, the broker's name might be folded to uppercase when it was created, even though you entered its name in lowercase.

You cannot connect to the Configuration Manager from the workbench or command line:

- **Scenario:** Error messages are issued when you try to connect to the Configuration Manager from the workbench or the command line.
- **Solution:** The following table shows the checks to carry out when an error message is issued in the workbench, or returned to the command line, when you try to connect to the Configuration Manager:

Error message		Actions
Message Brokers Toolkit	**Command line**	
BIP0914E	BIP1046E with MQ reason code 2059	Check that the Configuration Manager's queue manager and listener are running, and that the correct port is specified for the queue manager. (The WebSphere MQ documentation describes the WebSphere MQ return codes.)
BIP0889E	BIP1047E	Check that the Configuration Manager is running. Check the system event log to ensure that the Configuration Manager is available for use, and correct any errors that are shown.

Error message		Actions
Message Brokers Toolkit	**Command line**	
BIP0915E	BIP1046E with MQ reason code 2035	Check that the workbench local user ID is created on the Configuration Manager system. If it is not, create it. Check if the workbench local user ID that is created on the Configuration Manager system is authorized to connect to the Configuration Manager's queue manager. • Use the WebSphere MQ dspmqaut command to determine what WebSphere MQ authorities the user has. • If necessary, add the Message Brokers Toolkit local user ID to the **mqm** group on the Configuration Manager system.

BIP0991E with BIP1711W specifying that the user cannot view the Configuration Manager Proxy	BIP1047E	Add an access control list (ACL) entry to the Configuration Manager for the user with the workbench local user ID. For example, if *USERA* connects to the Message Brokers Toolkit on *COMPUTERA*, which belongs to *DOMAINA*, and is trying to connect to Configuration Manager *CMA*:
		• *USERA* must exist as a user on the Configuration Manager system and belong to the **mqm** group
		• *USERA* could have one of the following ACL entries:
		– To create an ACL entry for *USERA* to connect with full Configuration Manager Proxy authority from any computer or domain, issue the following command:
		`mqsicreateaclentry CMA -u USERA -a -x F -p`
		– To create an ACL entry for *USERA* to connect with full Configuration Manager Proxy authority from *COMPUTERA* only, issue the following command:
		`mqsicreateaclentry CMA -u USERA -m COMPUTERA -x F -p`
		– To create an ACL entry for *USERA* to connect with full Configuration Manager Proxy authority from *DOMAINA* only, issue the following command:
		`mqsicreateaclentry CMA -u USERA -m DOMAINA -x F -p`
		If a domain name and computer name match, and an ACL entry specified the **-m** parameter, connection from either the domain or computer is acceptable because they are not considered to be different users.
		
		ACLs are necessary for all connections to the Configuration Manager, so the user who issues a command such as mqsideploy from the command line must also have appropriate ACL entries.

A Version 5.0 or Version 5.1 Message Brokers Toolkit is unable to connect to a Version 6.0 Configuration Manager using a domain user ID:

- **Scenario:** You cannot use a Version 5.0 or Version 5.1 Message Brokers Toolkit to connect to a Version 6.0 Configuration Manager using a domain user ID with access that is controlled by a domain user ID access control list (ACL) entry.
- **Solution:** If you are logged on to the Version 5.0 or Version 5.1 Message Brokers Toolkit with a domain user ID, you can connect to the Version 6.0 Configuration Manager by creating an ACL on the Configuration Manager for the local user ID. Specify the **-a** parameter to allow that user ID to be used from any computer or domain. For example:

  ```
  mqsicreateaclentry CMGR01 -u USERA -a -x F -p
  ```

To use a domain user ID to connect to the Version 6.0 Configuration Manager and specify a user ID to access the Configuration Manager, as controlled by an ACL, you must upgrade the Message Brokers Toolkit to Version 6.0.

Connection to the Configuration Manager is slow:

- **Scenario:** Connection to the Configuration Manager is slow and you are unable to perform other actions in the workbench when you perform the following operations:
  - Connecting to a domain
  - Creating a new broker
  - Creating a new execution group
  - Deploying a broker archive file, using the Deploy a BAR File wizard
  - Opening the Broker Topology editor
  - Opening the Topics Hierarchy editor
  - Opening the Subscriptions Query editor
  - Opening the Event Log editor
- **Solution:** Click **Cancel** on the progress monitor dialog box to cancel the in-progress connection. Connection to the Configuration Manager is cancelled, allowing you to perform other actions in the workbench. You can then resubmit the cancelled operation.

Resolving error messages that occur when using the workbench

Work through the advice provided to help you to resolve some common problems that can occur when you use the workbench.

Errors occur while you are using the workbench:

- **Scenario:** An error has occurred while you are using the workbench, and you want information to help you to diagnose the problem.
- **Solution:** Some errors that occur in the workbench are logged in the .log file in your workspace\.metadata directory. You can view this log by switching to the Plug-in Development perspective and clicking the Error Log tab in the bottom right pane. The log shows in which plug-in the error occurred, and gives further information.

  You can also use trace to try and determine the cause of your problem. See “Using trace” on page 90 for more information about tracing.

An error is issued when you access the help system:

- **Scenario:** You are accessing the help system through the workbench, and an error message is issued stating that the web page that you requested is not available offline.
- **Explanation:** This error might occur if you previously had a connection to a Web-based version of the help system and lost it, or if you have **Work Offline** selected in your Internet Explorer options.
- **Solution:** Click **Connect** to load the help system.

The workbench displays an error message after the error has been fixed:

- **Scenario:** The workbench is in an inconsistent state, or displays an error message after the error has been fixed
- **Solution:** Clean the workspace:
  1. From the workbench, click **Project** → **Clean**.
  2. Click **Clean all projects** and **Finish**.

     This action cleans the whole workspace of any internal files, which are then re-created so that none of your data is lost.

Error message BIP0800 is issued when you rename a domain object:

- **Scenario:** Error message BIP0800 is issued when you click **OK** in the rename dialog box on a domain object.
- **Explanation:** This error occurs when the Domains Navigator receives a notification about object creation or deletion and changes its selection while you are entering the new name in the dialog box.
- **Solution:** Refresh the parent of the object that you are trying to rename to see whether the action was sent to the Configuration Manager before the error occurred. If the name of the object has not changed, wait for an automatic refresh in the Domains Navigator and try to rename the object again. An automatic refresh typically occurs when you or another user have saved many changes in the Topics hierarchy.

Error message BIP0935 is issued when you edit domain connection parameters:

- **Scenario:** You want to edit the connection parameters for the domain, and see only one connection file. However, error message BIP0935 is issued, stating that several domain connection files exist for the domain.
- **Explanation:** This error can occur when you delete the port value in the Domain Connection editor and save this file. The corresponding domain is displayed in the Domains view with a label like `yourQMGR@yourMachine:`. Error BIP0935 is issued when you try to edit parameters for this domain.
- **Solution:** In the Broker Administration perspective, edit the corresponding .configmgr file to set the port to 1414, and save the file. Even if the domain label is not updated in the Domains view, the port parameter has been taken into account in the domain connection file.

  If you want to see the correct label and open the Domain Connection editor from the Domains view, close and restart the workbench.

Warning message BIP0876 is issued when you make changes in the Broker Topology editor:

- **Scenario:** While you are making some changes in the Broker Topology editor, warning message BIP0876 is issued and you are invited to close the editor without saving.
- **Explanation:** The editor has been notified that Configuration Manager content has changed, and the changes that you have initiated conflict with those recent changes. This conflict is typically because another user has made some changes on the topology. To avoid potential conflicts between other users' changes and yours, you are invited to close the editor without saving. You can then reopen the editor, which is populated with the actual Configuration Manager content.
- **Solution:** If you are sure that your changes will not conflict with those made by another user and you want to save your changes, you can save them. The Configuration Manager tries to apply your changes on top of its actual content. You are notified if any of the requests fail.

  The same message might be raised in the Topics editor for the same reason.

The message `Unable to create part: filename.extension` is issued:

- **Scenario:** You open an editor in the workbench and an error message is raised stating that a file cannot be created. Two possible explanations for this problem are given below:
- **Explanation 1:** The given file is not associated with any editor.
  - **Solution 1:** Right-click the file and choose the default editor to open it, or choose another editor if the default editor cannot open it.

- **Explanation 2:** The given file contains syntax errors and cannot be loaded into the chosen editor. However, if you then try to open a valid file, you get the same error message repeatedly.
  - **Solution 2:** Restart the workbench; do not load the file into an editor again until you have fixed the syntax errors.

Resolving problems relating to the appearance of the workbench

This topic contains advice for dealing with some common problems that relate to the appearance of the workbench.

You cannot close a message dialog box:

- **Scenario:** You have opened a pop-up menu, and a message dialog box tells you that previous changes made have been processed successfully by the Configuration Manager. You cannot close the dialog.
- **Solution:** If you cannot close the dialog box by clicking **OK**, press Esc.

The Domains navigator content is updated but you have not made any changes:

- **Scenario:** You are connected to a domain and have not made any changes. However, the Domains Navigator and editor's content are updated.
- **Explanation:** Another user has made changes on the Configuration Manager. The Domains Navigator and editors are notified of these changes and are refreshed automatically.

A server project containing a valid domain connection file is not displayed:

- **Scenario:** You have a valid domain connection file in one of your workspace server projects, but you cannot see this project in the Broker Administration perspective, or an associated domain in the Domains Navigator.
- **Explanation:** The project is closed. Closed projects are not displayed in the Broker Administration perspective, and their content is not analyzed.
- **Solution:** Open the project from the Navigator view in the Resource perspective. The project appears in the Broker Administration perspective, and its content is analyzed and displayed. The domain connection and domain then appear in the appropriate navigators.

Some objects are not visible in the Domains Navigator:

- **Scenario:** Some objects, for example broker topologies, topics, brokers or execution groups, are not visible in the Domains Navigator.
- **Explanation:** ACL-restricted objects might not be visible in the Domains Navigator if the corresponding **Show ACL restricted objects in Navigators** preference was previously not selected.
- **Solution:**
  1. Click **Window** → **Preferences** → **Broker Administration preference page**.
  2. Check whether the **Show ACL restricted objects in Navigators** preference is selected. If it is not, select it and click **OK**.

  Any restricted objects are displayed in the Domains Navigator.

You want to filter entries in the Tasks view:

- **Scenario:** The Tasks view has such a large number of entries that it is difficult to find the entry that you want.

- **Solution:** At the top of the Tasks View, click the icon with the arrows pointing to the right. This opens the Filter Tasks dialog box, in which you can tailor your selections to display only a subset of the task list entries, such as only those entries for the currently-selected project.

Your message flow projects have changed their appearance:

- **Scenario:** You have created a new simple project in the workbench and now all your message flow projects look different.
- **Explanation:** When you create a new simple project, the workbench switches automatically to the Broker Application Development perspective.
- **Solution:** To return to the previous view, switch to the Broker Administration perspective.

Resolving non-specific problems when using the workbench

Work through the advice provided to help you to resolve some common problems that can occur when you use the workbench that are not dealt with in previous categories.

Editors do not update automatically when the same file is open in multiple windows:

- **Scenario:** You are working in the Broker Application Development perspective, and are using the associated editor to work with one or more resources; for example, you are editing a message flow in the message flow editor. You have clicked **Window → New Window** to create a second Eclipse view, and have opened the same resource in the second window. Changes that you make to the resource in the first editor window are not reflected in the second editor window.
- **Explanation:** The workbench editors do not automatically update multiple windows in which you have opened the same resource.
- **Solution:** Save the contents of the resource file in the first editor window, then close and reopen additional windows. The reopened windows reflect the updated content.

Deleting or closing a project takes a long time:

- **Scenario:** Deleting or closing a project to save memory takes a long time.
- **Explanation:** If a project is referenced by other projects, removing that project requires all the other projects, and the projects that refer to them recursively, to be built fully. This process occurs to keep the content-assist and validation models current.
- **Solution:** To keep a project open in the workspace requires very little memory, therefore you do not need to close or delete projects.

You are experiencing poor performance when working with large or complex projects:

- **Scenario:** You are experiencing poor performance in the workbench when working with large or complex projects.
- **Explanation:** Frequent project changes, such as adding and removing projects, or using **Project → Clean**, use large amounts of memory due to the size and number of files and the connections between them.
- **Solution:** Increase your system memory.

You are considering moving database resources within the workbench:

- **Scenario:** You are considering moving database resources within the workbench, such as the XMI files for connection, database, schema, and tables.
- **Solution:** You cannot move database resources after they are imported. The only move that you can perform without affecting the integrity of the files is between two projects while keeping the project-relative path of the files unchanged. To place the database resources in a different folder, import them again from the database.

You do not know how to return to the welcome page:

- **Scenario:** You do not know how to return to the welcome page that was displayed in the workbench when you first started using it.
- **Solution:** To open the welcome page:
  1. From the **Help** menu, select **Welcome**. If only one welcome page is available, it is displayed. If more than one is available, a list is displayed.
  2. Select the welcome page that you want, for example, WebSphere Event Broker Message Brokers Toolkit.
  3. Click **OK**.

Resolving problems when using databases

Use the advice given here to help you to resolve problems that can arise when using databases.

Before you start:

- Read "Is there a problem with a database?" on page 7

This topic contains advice for dealing with some common problems that can arise when using databases:

- "DB2 error message SQL0443N is issued" on page 67
- "DB2 error message SQL0805N is issued" on page 67
- "DB2 error message SQL0998N is issued" on page 67
- "DB2 error message SQL1040N is issued" on page 68
- "DB2 error message SQL1224N is issued when you connect to DB2" on page 68
- "DB2 error message SQLSTATE=58005 is issued" on page 70
- "DB2 or ODBC error messages are issued on z/OS" on page 70
- "Error message BIP1780 is issued on AIX" on page 71
- "You do not know how many database connections a broker requires" on page 71
- "You want to use XA with DB2" on page 71
- "XA coordination fails with DB2 V8 Fix Pack 2" on page 71
- "XA coordination fails if the database restarts while the broker is running" on page 72
- "On Oracle, a database operation fails to return any rows, even though the rows exist" on page 72
- "There is a memory leak from the Oracle Client Interface on HP-UX 11" on page 72
- "Oracle error message ORA-12500 is issued on Solaris 8 with Oracle 9" on page 72
- "Error messages BIP2731, BIP2321, and BIP2322 are issued when you use retained publications with a Sybase database" on page 73

- "Error message BIP2275 is issued when a large message flow is deployed to a Sybase broker database" on page 73
- "The DataDirect trace system is unable to open the ODBC trace file" on page 73
- "Abend when deploying a message flow" on page 74

DB2 error message SQL0443N is issued

- **Scenario:** After you upgrade the server to DB2 Universal Database (DB2 UDB) Version 8.1 Fix Pack 10 (also known as Version 8.2 Fix Pack 3), a DB2 error message SQL0443N is issued if you invoke a DB2 Call Level Interface (CLI) catalog function, such as SQLTables(), SQLColumns(), or SQLStatistics(). An example of the error message is:

  ```
  SQL0443N Routine "SYSIBM.SQLTABLES" (specific name "TABLES") has returned
  an error SQLSTATE with diagnostic text SYSIBM:CLI:-805". SQLSTATE=38553.
  ```

- **Solution:** Bind the db2schema.bnd file against each database by entering the following commands at a command prompt:

  ```
  db2 terminate
  db2 connect to database-name
  db2 bind path\db2schema.bnd blocking all grant public sqlerror continue
  db2 terminate
  ```

  where *database-name* is the name of the database to which the utilities must be bound, and *path* is the full path name of the directory where the bind files are located. For example, the default location on Windows is C:\Program Files\IBM\SQLLIB\bnd\.

  To list all the names of databases for a particular DB2 instance, run the DB2 CLI command db2 list database directory. For further information, see the DB2 documentation.

DB2 error message SQL0805N is issued

- **Scenario:** When you run the mqsicreatebroker command, SQL error `SQL0805N NULLID.SQLLF000` is issued.
- **Solution:** Open a DB2 Command Line Processor window and issue a bind command to the database.

  Linux UNIX On Linux and UNIX systems, enter the commands:

  ```
  connect to db
  bind ~/sqllib/bnd/@db2cli.lst grant public CLIPKG 5
  connect reset
  ```

  where *db* is the database name.

  Windows On Windows systems, enter the commands:

  ```
  connect to db
  bind x:\sqllib\bnd\@db2cli.lst blocking all grant public
  connect reset
  ```

  where *x:* identifies the drive onto which you installed DB2, and *db* is the database name.

DB2 error message SQL0998N is issued

- **Scenario:** When you use a globally coordinated message flow with a DB2 8.1 database, error SQL0998N is issued with Reason Code 09 and Subcode 02.
- **Solution:** Use the following procedure to set the DB2 database manager configuration parameter **TP_MON_NAME** to *MQ*, and alter the XAOpenString in the qm.ini file for your queue manager:

1. To set the **TP_MON_NAME** parameter, use the following command from a login that has administrative authority for the DB2 instance:

   ```
   db2 update dbm cfg using TP_MON_NAME MQ
   ```

2. Stop and restart the DB2 instance so that the value is recognized.
3. To set the AXLIB value in the qm.ini file, alter the XAOpenString to match the following example:

   ```
   XAOpenString=DB=databasename,UID=userid,PWD=password,AXLIB=library
   ```

4. Set the AXLIB library path to one of the following values, depending on your operating system:
   - On Windows:

     ```
     AXLIB=mqmax
     ```

   - On AIX:

     ```
     AXLIB=usr/mqm/lib/libmqmax_r.a
     ```

   - On HP-UX:

     ```
     AXLIB=/opt/mqm/lib/libmqmax_r.sl
     ```

   - On Linux:

     ```
     AXLIB=/opt/mqm/lib/libmqmax_r.so
     ```

   - On Solaris:

     ```
     AXLIB=/opt/mqm/lib/libmqmax.so
     ```

5. Stop and restart your broker and queue manager for these changes to take effect.

DB2 error message SQL1040N is issued

- **Scenario:** You are using DB2, and error message BIP2322 is issued with error SQL1040N.
- **Explanation:** The following DB2 message indicates that the value of the DB2 database configuration parameter **maxappls** has been reached:

```
"SQL1040N The maximum number of applications is already connected to the database.
SQLSTATE=57030"
```

  DB2 has rejected the attempt to connect.

  If this database is one of the defined broker databases, implying that a broker thread connection request has failed, the broker is probably not functioning correctly.
- **Solution:**
  1. Stop all brokers that connect to the affected database.
  2. Increase the value of the **maxappls** configuration parameter. Also, check the value of the associated parameter **maxagents** and increase it in line with **maxappls**.
  3. Restart the DB2 database.

DB2 error message SQL1224N is issued when you connect to DB2

- **Scenario:** DB2 error message SQL1224N is issued when you connect to DB2. This error indicates that a database agent could not be started, or was terminated as a result of a database shutdown or force command.
- **Solution:** On AIX, use TCP/IP to connect to DB2 to avoid the shared memory limit of 10 connections. To set up AIX and DB2 loop-back to use a TCP/IP connection:
  1. Use the mqsideletebroker command to delete your broker.

2. Configure DB2 to use TCP/IP, and to start the TCP/IP listener. On the server machine, log in as the DB2 instance owner, typically `db2inst1`, and issue the following commands:

   ```
   db2set DB2COMM=tcpip
   db2stop
   db2start
   ```

3. If the DB2 connection port is not defined in /etc/services, edit the services file to add the DB2 connection and interrupt ports. You must use unique names, and port numbers that are not already defined in the services file; for example:

   ```
   db2svc1     3700/tcp          # DB2 Connection Service
   db2isvc1    3701/tcp          # DB2 Interrupt Service
   ```

4. Update the DB2 configuration; for example:

   ```
   db2 update dbm cfg using svcename db2svc1
   ```

   where *db2svc1* is the name of the DB2 Connection port service in /etc/services.

   Alternatively, you can specify a port number directly.

5. Stop and restart the database by using the following commands:

   ```
   db2stop
   db2start
   ```

6. Catalog a new TCP/IP connection node:

   ```
   db2 catalog tcpip node NODENAME remote HOSTNAME server db2svc1
   ```

   where:

   NODENAME
   : is the name of the new TCP/IP connection node. You can use `local` as your node name, provided that it is a unique identifier.

   HOSTNAME
   : is the name of your computer.

   db2svc1
   : is the name of the DB2 connection port service in `/etc/services`.

   Message DB20000I is displayed when the command completes successfully.

7. Catalog the database with a new alias name; for example:

   ```
   db2 catalog database DATABASE as DBALIAS at node NODENAME
   ```

   where:

   DATABASE
   : is the physical name of the database.

   DBALIAS
   : is the database alias name that you want to use.

   Specify the new alias name in all subsequent references to the local database; for example, when you run the mqsicreatebroker command.

8. Stop and start DB2:

   ```
   db2 terminate
   db2stop
   db2start
   ```

9. Log on with the broker service user ID.
10. Update the ODBC configuration file for each broker to add definitions for the database (the default file odbc.ini file is in directory /var/wmqi/odbc):

a. At the top of the file, add a definition for the database alias name:

```
DBALIAS=IBM DB2 ODBC Driver
```

b. Add a new stanza for the database alias:

```
 [DBALIAS]
Driver=INSTHOME/sqllib/lib/libdb2.a
Description=Broker Database Alias
Database=DBALIAS
```

where *INSTHOME* is the path to your DB2 Instance directory.

11. Create a new broker using the DBALIAS on the **-n** parameter; for example:

```
mqsicreatebroker BrokerName -i mqsiuid -a mqsipw
   -q BrokerQueueManager -n DBALIAS -n
```

12. Start the broker, and redefine it in the workbench.
13. Deploy the broker and test the flows.

DB2 error message SQLSTATE=58005 is issued

- **Scenario:** DB2 error SQLSTATE=58005 is issued when you use WebSphere Event Broker on Linux with DB2 Version 8.1 and Fix Pack 9.
- **Explanation:** This error is issued when the values of the kernel parameters (msgmni, sem) are too low.
- **Solution:** Increase the kernel parameters (msgmni, sem).

  These kernel parameters should be significantly above their minimum values and at least the highest of the recommended values for DB2, WebSphere MQ, and WebSphere Event Broker. The following example shows possible values for a heavy workload environment where the broker has two execution groups with 200 message flows deployed, and approximately 45 applications that are using these message flows:

```
/proc/sys/kernel/msgmni  2048
/proc/sys/kernel/sem     250   256000   32   2048
```

DB2 or ODBC error messages are issued on z/OS

- **Scenario:** DB2 or ODBC messages are issued on z/OS indicating that:
  - An exception was caught while issuing the database SQL connect command.
  - There was a database error with an ODBC return code of -1, an SQL state of 58004 and a native error code of -99999.
  - The broker made an unsuccessful attempt to access its database using a particular user ID.
- **Solution:** If an ODBC message is displayed:
  1. Turn ODBC application tracing on to produce the traceodbc file.
  2. Locate the traceodbc file. This is in the /output subdirectory. For example /u/argo/VCP0BRK/output/traceodbc.
  3. Go to the bottom of this file and search for previous instances of SQLerror.

  Common DB2 problems include:

  - ODBC return code -1, SQL state 58004, Native error code -99999

    This might be caused by:

    - No SQL code. The DB2 subsystem is not started
    - RRS is not started.
  - SQLCODE 922.

    The user ID of the started task is not authorized to use plan DSNACLI.
  - ODBC return code -1, SLQ state 42503, Native error code -553

This might be caused by the user ID of the started task not being authorized to use the current SQL ID. Re-configure the broker and specify `DB2_TABLE_NAME` as a valid name, or create an RACF group and connect the started task user ID to this group.

Error message BIP1780 is issued on AIX

- **Scenario:** Error message BIP1780 is issued on AIX, indicating that the user ID could not be validated.
- **Explanation:** If you change your AIX operating system password after you create the broker, when you perform an action such as deploying a bar file, it fails because the password change causes the connection to the DB2 database to fail.
- **Solution:** Run the mqsichangebroker command for the user ID that is used to connect to the database:

  ```
  mqsichangebroker broker -p password
  ```

  This allows DB2 to authenticate the user ID correctly.

You do not know how many database connections a broker requires

- **Scenario:** You do not know how many database connections to set up for your broker.
- **Solution:** Determine the number of database connections required by a broker for capacity and resource planning. On DB2, the default action taken is to limit the number of concurrent connections to a database to the value of the **maxappls** configuration parameter; the default for **maxappls** is 40. The associated parameter **maxagents** also impacts the current connections.

  The connection requirements for a single message broker are:

  – Five are required by internal broker threads.

  – One is required for each publish/subscribe neighbor, if the topology has been deployed.

  – One is required for each message flow thread that parses MRM messages.

You want to use XA with DB2

- **Scenario:** You want to use XA with DB2.
- **Solution:** If you want to use XA coordination with DB2 V8.1, ensure that your queue manager is configured to use **ThreadOfControl=THREAD**. On Linux or UNIX you can configure this parameter in the XAResourceManager stanza in qm.ini . On Windows you can configure this parameter using the WebSphere MQ Explorer or WebSphere MQ Services snap-in depending on what version of WebSphere MQ you are using.

XA coordination fails with DB2 V8 Fix Pack 2

- **Scenario:** XA coordination fails with DB2 V8 Fix Pack 2 on AIX, HP-UX, Linux, Solaris, or Windows.
- **Explanation:** If a non-XA coordinated message flow tries to access a database that was previously being used by an XA coordinated message flow, failure can occur because DB2 believes that the later flow is still coordinated from the previous flow.
- **Solution:** This is fixed in DB2 V8 Fix Pack 3 by APAR IY44711.

XA coordination fails if the database restarts while the broker is running

- **Scenario:** XA global coordination fails and you get an error similar to the following example, which is from a DB2 user database:

  ```
  Database error: SQL State '40003'; Native Error Code '-900';  Error Text '[IBM]
  [CLI Driver] SQL0900N  The application state  is in error. A database connection
  does not exist.SQLSTATE=08003'.
  ```

- **Explanation:** A globally coordinated message flow cannot automatically reconnect to a user database if the user database is restarted while the broker is still running.
- **Solution:** Stop and restart the broker if the user database goes down or is brought down for a scheduled maintenance.

On Oracle, a database operation fails to return any rows, even though the rows exist

- **Scenario:** You are using Oracle databases in your message flows and ESQL binds against columns, which are declared as data type CHAR, and those parameter markers are referenced in a WHERE clause. The database operation fails to return any rows, even though the rows exist.
- **Explanation:** Such fixed length character strings need to be padded with blank characters on Oracle for this type of comparison to succeed.
- **Solution:** Define the CHAR columns as VARCHAR2 columns, or pad the ESQL variable with blank characters to the required column length, so that the comparison locates the desired rows from the table.

There is a memory leak from the Oracle Client Interface on HP-UX 11

- **Scenario:** A memory leak has been observed from the Oracle Client Interface (OCI) when using Oracle 9i Release 2 (9.2.0.1) on HP-UX 11.
- **Solution:** This problem has been fixed by Oracle and the fix is available by upgrading to Oracle 9i Release 2 Database Server Patch Set 2 for HP9000 Series HP-UX (64-bit). This has the Oracle patch number of 2761332. To install the above Oracle patch, you might need to upgrade your Oracle installer (OUI) to version 2.2.0.18.0 first. The Oracle patch number for this upgrade is 2878462.

Oracle error message ORA-12500 is issued on Solaris 8 with Oracle 9

- **Scenario:** Oracle error message ORA-12500 is issued on Solaris 8 with Oracle 9 when you run workload applications that are performing database insert, update, or delete operations.
- **Explanation:** This error is issued when the values of the tuning parameters are too low.
- **Solution:** Increase the following Oracle 9i Release 2 tuning SGA parameters in the initSID.ora file:
  - JAVA_POOL_SIZE
  - SHARED_POOL_SIZE
  - SORT_AREA_SIZE

  The following example shows possible increased values for a workload setup with a single broker that has two execution groups, each with 92 message flows:

  ```
  .java_pool_size=125829120
  .shared_pool_size=218862284
  .sort_area_size=65536
  ```

Error messages BIP2731, BIP2321, and BIP2322 are issued when you use retained publications with a Sybase database

- **Scenario:** The following error messages are issued when you use retained publications and a Sybase broker database:

  ```
  BIP2731 Database statement 'INSERT INTO dbo.BRETAINEDPUBS
  VALUES(?, ?, ?, ?, ?, ?, ?, ?, ?, ?)' could not be executed.
  BIP2321 Database error: ODBC return code '-1'.
  BIP2322 Database error: SQL State '40001'; Native Error Code '1205'.
  Text '[SYBASE][ODBC Sybase Driver][SQL Server]Your server command
  (family id #0, process id #234) encountered a deadlock situation.
  Please re-run your command.'
  ```

- **Explanation:** These errors are likely to occur only when you use retained publications with multiple topics, with a substantial workload.
- **Solution:** Apply row level locking to one of the broker's database tables:
  1. At a Sybase command prompt, enter the command:

     ```
     isql -Uusername -Ppassword
     ```

  2. Connect to the broker's database:

     ```
     use broker DSN
     ```

     where *broker DSN* is the ODBC Data Source Name (DSN) for the broker database.
  3. Enter the command:

     ```
     alter table dbo.BRETAINEDPUBS lock datarows
     ```

     where *dbo* is the schema name.
  4. Enter the command:

     ```
     go
     ```

Error message BIP2275 is issued when a large message flow is deployed to a Sybase broker database

- **Scenario:** You are using Sybase as the broker database and error messages BIP2275, BIP5009, and BIP5004 are issued during broker startup.
- **Explanation:** Error messages BIP2275, BIP5009, and BIP5004 indicate that a message flow XML definition retrieved from the broker database at startup is invalid XML. This can be caused by a message flow definition which is greater than 1 MB in length being truncated when it is retrieved from the BROKERRESOURCES table at broker startup. The reason that it is truncated is that the default maximum length of data, for Sybase, that can be retrieved at broker startup from the RESOURCEDATA column (where message flow definitions are stored) of the BROKERRESOURCES table is 1 MB.
- **Solution:**
  1. Stop all brokers that connect to the affected database.
  2. Add `DefaultLongDataBuffLen=2048` to the DSN definition in the ODBC configuration file(s) (either $ODBCINI or $ODBCINI64, or both).
  3. Restart the broker.

The DataDirect trace system is unable to open the ODBC trace file

- **Scenario:** Commands, such as mqsichangetrace and mqsicreatebroker, fail and a message is displayed stating that the ODBC trace file cannot be opened.
- **Explanation:** The DataDirect ODBC file is inaccessible or its size has exceeded 2 GB and tracing has been turned on.
- **Solution:** To resolve this problem:

1. Turn off tracing in the file to which the ODBCINI environment variable points.
2. Locate the ODBC trace file to which 'TraceFile' in the ODBC stanza points. If you need to keep the trace, move the ODBC trace file to another location. If you do not need to keep the trace, delete the ODBC trace file.
3. If you still need to gather ODBC trace, reset `TRACE=1` in the odbc.ini file.

Abend when deploying a message flow

- **Scenario:** You are running DB2 version 9 on HP-UX on PA-RISC. You try to deploy a message flow, but the deployment fails.
- **Solution:** Export environment variable MQSI_SIGNAL_EXCLUSIONS in the broker environment:

  ```
  export MQSI_SIGNAL_EXCLUSIONS=10
  ```

Resolving problems when using publish/subscribe

Get solutions to common problems with publish/subscribe.

- "Your broker is not registering with the User Name Server"
- "Application responses are not received" on page 75
- "Your application is not receiving publications" on page 75
- "Access control appears not to work" on page 76
- "You cannot access the broker topology after you change the access control list (ACL)" on page 76
- "Publishing a message causes a filter error" on page 77
- "Symbols in subscription filters cause problems" on page 77
- "There are problems with multiple brokers and communications" on page 77
- "The workbench does not display non-Windows users" on page 78
- "There are performance problems on AIX when the JIT compiler is not loaded" on page 78
- "Error message BIP2332 is issued with SQLSTATE 57011 when you access the publish/subscribe database on z/OS" on page 78
- "Your broker does not have the correct authority to run PGM/IP" on page 79
- "Your client does not have the correct authority to run PGM/IP" on page 79
- "Your client does not have sufficient memory to run PGM/IP" on page 79

Your broker is not registering with the User Name Server

- **Scenario:** Your broker has been enabled to use topic-based security by setting the **-s** parameter of the **mqsicreatebroker** command to the User Name Server queue manager, but many occurrences of warning message BIP8303 are displayed on the console, indicating that the broker's user or group cache is empty. When you try to deploy ACLs, more occurrences of BIP8303 are displayed, as well as the error message BIP7017:
- **Explanation:** It is typical to see a few occurrences of BIP8303, but if you continue to see this message after you deploy ACLs and try to use topic-based security, it is likely that your broker has not been registered with the User Name Server.
- **Solution:**
  - Check that the broker has the correct name for the User Name Server queue manager by going to the broker's file system and looking at the value of the `UserNameServerQueueManagerName` field in the registry entry for the broker. If

the value is not correct, use the **mqsichangebroker** command, setting the **-s** parameter to the correct User Name Server queue manager.

– Check that the User Name Server process and the control process are both running. WebSphere MQ intercommunication must also be running correctly if the User Name Server is on a different queue manager.

In each of these cases, restart the broker, deploy any topics data, and look for the message:

```
+() 0 BIP8204I: User Name Server is registering a client with UUID
15db2a8e-869e-11d5-8000-091465ac0000, and cache version 0.
```

Application responses are not received

- **Scenario:** Application responses are not received.
- **Explanation:** Depending on the application code, a publisher or subscriber might request confirmation that its message was processed successfully. Responses can make debugging client problems much easier, because a response code is given in the event of a problem occurring.

  This response should always be possible for a subscriber, but for the publishing side, a message might be published without the knowledge of the originating application (for example, by using the default topic property on the input node). The results of processing that message are still logged in user trace, which might give clues as to what is happening.

  If a response is not received, there are several possible causes:

  – The system is busy. Messages might build up on the input queue, and the client might not be waiting long enough for its response.

  – WebSphere MQ expiry is being used. There are cases where this is what is required, but the expiry of the input message is copied to the response. As a result, the message might expire on the broker's input queue, or it might expire on the way back to the client. This is not an error, even with a persistent message.

  – The input message or response might have been put to the dead-letter queue, if one is configured. Look on this queue to see if any new messages are there. This situation is typically accompanied by error messages in the broker's log that describe the problem. For example, the reply-to queue might have been specified incorrectly in the input message, therefore the reply message has been put to the dead-letter queue.
- **Solution:** If your application is not asking for responses (that is, not using messages of type MQMT_REQUEST) consider doing so, particularly when developing applications.

Your application is not receiving publications

- **Scenario:** Your application is not receiving publications.
- **Explanation:** If an application has subscribed successfully (that is, has received an OK response to a RegSub message), it should receive publications that match its subscription. Details of its registration can be viewed in the Subscriptions pane of the Broker Administration perspective. This can be useful to check that the details are correct.

  Subscribers are sent messages only if they match the topic, the subscription point, *and* the filter. Because the subscription point is specified in the message flow, not in the publication message, an incorrect message flow setting can cause unexpected failures.
- **Solution:** If a filter is being used, a user trace shows you whether this is being evaluated as expected.

The case with multiple execution groups, or multiple brokers, is more complex. A response is sent to a subscriber after the message has been processed by the target execution group. Other execution groups (and brokers) are updated asynchronously. As a result, there might be a delay before publications made elsewhere are received. If the broker is busy, there can be a delay before messages are processed fully. In a multi-broker setup, if communications have been suspended, subscription changes are propagated through the network of brokers. Check the channels.

With multiple execution groups or brokers, it might be possible to fill intermediate WebSphere MQ queues if the load is very high. This might be reported in the syslog (if a broker cannot put to a queue because it is full) or in the WebSphere MQ log (if a message coming across a channel cannot be put to the target queue because it is full). If you see messages of this type, display the queue depths on all your queue managers to see if any are almost full.

Access control appears not to work

- **Scenario:** Access control appears not to work.
- **Solution:** Ensure that every broker is configured with the User Name Server. Use a single User Name Server for the whole broker domain. In addition, all the brokers and the Configuration Manager must be able to access the User Name Server's queue manager.

  If BIP8303 errors are recorded in any broker's log after startup, it indicates a communication problem with the User Name Server. Check your WebSphere MQ channels to and from the User Name Server and try again. You should see event message BIP8204I for both the Configuration Manager and any brokers, showing that they have successfully registered with the User Name Server.

  There might be problems with users and group memberships if your broker domain spans a number of systems with different lists of users. For each computer holding a broker, there must be a group called **mqbrkrs**. This must contain the following principals, or authorization failures are reported:

  – Local broker's service user ID

  – All neighboring brokers' service user IDs

  – The Configuration Manager's service user ID

  For client applications, if the user ID in the message is a member of the **mqbrkrs** group, security is bypassed (even if a User Name Server is being used).

  If your publish/subscribe request fails, and event message BIP7017 is issued, check that the client user ID is known to the system on which the User Name Server is running. Also, if you are operating in a Windows domain environment, ensure that the User Name Server was created with the **-d** parameter on the **mqsicreateusernameserver** command set to the appropriate domain, and that all client application user IDs are members of this domain.

  Message persistence in publish/subscribe is typically preserved. However, a subscriber might not get the expected persistence if ACLs do not allow it.

You cannot access the broker topology after you change the access control list (ACL)

- **Scenario:** After connecting to the Configuration Manager, the broker topology is displayed in the Domains Navigator with restricted access, because your user does not belong to suitable Windows security groups, or an ACL entry has not been defined for this user to access the broker topology. After you have created a suitable ACL entry using the **mqsicreateaclentry** command, and disconnected from, and reconnected to, the domain, the broker topology is still shown with restricted access.

- **Solution:** The ACL information has not yet been refreshed. To force a refresh, stop and restart the User Name Server and the Configuration Manager.

Publishing a message causes a filter error

- **Scenario:** When you publish a message, you receive an error response message with reason text MQRCCF_FILTER_ERROR.
- **Explanation:** A broker returns this message to a publication when subscriptions have been registered that specify filter expressions (for Content Based Routing) and an error has been encountered when the broker attempts to filter the published message. This can occur, for example, if a message is published that includes unsupported data types, or if the message body is corrupted.

Symbols in subscription filters cause problems

- **Scenario:** If you specify certain symbols when you use filters in a subscription, the filter does not work. Sometimes your subscription messages are put to the dead-letter queue, and a number of error messages are written to the local error log indicating MQRFH2 parsing errors.
- **Explanation:** The MQRFH2 header employs standard XML encoding, so that its parser interprets some symbols in a special way.
- **Solution:** If you want to include these symbols in your filters, use the appropriate escape character to ensure that they are parsed correctly:

Symbol	Escape character
<	<
>	>
"	"
'	'
&	&

For example, if you want to use:

```
<Filter>Body.e_ALERT_BODY.eqnum<6</Filter>
```

specify:

```
<Filter>Body.e_ALERT_BODY.eqnum&lt;6</Filter>
```

There are problems with multiple brokers and communications

- **Scenario:** You are experiencing problems with multiple brokers and communications.
- **Explanation:** Channels and transmission queues are needed between each connected broker in the topology. These are configured in the same way as for communication between a single broker and the Configuration Manager. Within a collective, every broker is connected to every other broker, therefore many channels are required.
- **Solution:** Using WebSphere MQ clustering might make the administration job easier. Otherwise, ensure that every channel is working.

  When the configuration is deployed, a response is received from each broker; these responses are displayed in the workbench. However, also check each broker's local log. For each connection, a BIP7113 message is issued. To test communications, the following sample might be useful:

  – Soccer Results sample

  You can run each part on a separate broker to test communications between them. You can view samples only when you use the information center that is integrated with the Message Brokers Toolkit.

The workbench does not display non-Windows users

- **Scenario:** The workbench and Configuration Manager are running on Windows, and the User Name Server and broker are running on z/OS, or a Linux or UNIX system. After starting the workbench, you find that the expected list of users and groups from z/OS, Linux, or UNIX is not displayed in the Topics view of the Broker Administration perspective.
- **Explanation:** The Configuration Manager has not registered with the User Name Server.
- **Solution:**
  1. Stop the Configuration Manager, using the **mqsistop** command.
  2. Issue the **mqsichangeconfigmgr** command, setting the **-s** parameter to the correct User Name Server queue manager name.
  3. Restart the Configuration Manager using the **mqsistart** command.
  4. Check that the User Name Server is running.

  You need to have both sender and receiver channels and transmission queues established to and from both the Configuration Manager queue manager and the User Name Server queue manager. Check that these channels have been created and are running.

  If you make any changes, restart the Configuration Manager and check for Windows event message BIP8258, which shows that the Configuration Manager has successfully registered with the User Name Server. If you make any changes, for example to channels, you also need to restart the workbench to refresh the displayed list of users.

  On z/OS, if the User Name Server is running, and if you have the previous options set correctly, but you are still not seeing z/OS users listed in the Topics view of the Broker Administration perspective, check that the z/OS users all have OMVS segments defined. If they do not, they are not listed by the User Name Server.

There are performance problems on AIX when the JIT compiler is not loaded

- **Scenario:** There are performance problems on AIX when the JIT compiler is not loaded.
- **Explanation:** The environment variable LIBPATH can affect the loading of the Java JIT (just-in-time) compiler on AIX. WebSphere Event Broker for AIX runs correctly without the JIT compiler, but publish/subscribe performance is adversely affected.
- **Solution:** Ensure that the JIT compiler runs.

  The JIT compiler loads and runs correctly if LIBPATH is not set, therefore do not set LIBPATH. You can make libraries available by linking them into /var/wmqi/lib (for all WebSphere Event Broker for AIX processes) or /usr/lib (for all processes on the system). WebSphere Event Broker for AIX configuration does this action for DB2 libraries.

  If it is necessary to set LIBPATH, update it to include the directory /usr/java130/bin. For example, you can use the following command to start the broker:

  ```
  LIBPATH=/usr/local/lib:/usr/java130/bin mqsistart mybroker
  ```

Error message BIP2332 is issued with SQLSTATE 57011 when you access the publish/subscribe database on z/OS

- **Scenario:** You are using publish/subscribe on z/OS with the MQ transport and large numbers of retained publications or subscriptions are being changed. Error

message BIP2332 is issued with SQLSTATE 57011 when you access the publish/subscribe database. You might also see DB2 errors in your SDF log; for example:

```
#DFL0 DSNILMCL RESOURCE UNAVAILABLE 558
  CORRELATION-ID=ST03BRK
  CONNECTION-ID=RRSAF
  LUW-ID=*
  REASON 00C90092
  TYPE 00000905
  NAME IRLM
*DXR175E IFL0001 IRLM IS UNABLE TO OBTAIN STORAGE - ECSA
 DSNT501I  #DFL0 DSNILMCL RESOURCE UNAVAILABLE 560
  CORRELATION-ID=ST04BRK
  CONNECTION-ID=RRSAF
  LUW-ID=*
  REASON 00C90092
  TYPE 00000905
  NAME IRLM
```

- **Explanation:** This problem indicates a DB2 tuning issue with the IRLM (lock manager) subcomponent.
- **Solution:** Make sure that you run IRLM with the lock information being held in local storage; use the startup parameter **PC=YES**. For more information, see the IBM Redbooks® publication *DB2 UDB for OS/390® Version 6 Performance Topics, SG24-5351*.

Your broker does not have the correct authority to run PGM/IP

- **Scenario:** The broker does not have the correct authority to run and has reverted back to PGM/UDP.
- **Explanation:** The broker is configured to use PGM/IP but does not have the correct user authority. The broker has reverted back to PGM/UDP.
- **Solution:** Specify a valid user authority for the broker, or select a different protocol using the multicast properties panel or the **mqsichangeproperties** command.

Your client does not have the correct authority to run PGM/IP

- **Scenario:** An exception occurred when the client attempted to connect with the broker.
- **Explanation:** The client is configured to use PGM/IP but does not have the correct user authority. An exception occurred when the client tried to connect with the broker.
- **Solution:** Obtain the correct user authority for your client. Also obtain root access and ensure the broker is running as root (as Administrator or as a regular user who is a member of the Administrators group on Windows), in a secure environment. Be aware of the security implications of doing this.

Your client does not have sufficient memory to run PGM/IP

- **Scenario:** Your client does not have sufficient memory to run PGM/IP
- **Explanation:** Clients running PGM/IP require the multicast property Maximum Memory Size to be set to a suitable value.
- **Solution:** Change the value of Maximum Client Memory Size parameter using the multicast properties panel or the **mqsichangeproperties** command. The Maximum Client Memory Size parameter is applicable only to PGM protocols, and is greyed out unless Protocol Type is set to one of the PGM protocols. This parameter is available in the Advanced properties panel.

Resolving problems with performance

- **Scenario:** You are experiencing problems with performance, such as:
  - Poor response times in the workbench when developing message flows
  - Poor response time at deployment
  - Individual messages taking a long time to process
  - Poor overall performance, or performance that does not scale well
- **Solution:** Possible solutions are:
  - Speed up WebSphere MQ persistent messaging by optimizing the I/O (Input/Output)
  - Increase system memory
  - Use additional instances or multiple execution groups

A good source of information about performance is the set of reports in WebSphere MQ Family Category 2 (freeware) SupportPacs, available for download from the WebSphere MQ SupportPacs Web page.

For more information, read "Do you have a component that is running slowly?" on page 9.

This topic contains advice for dealing with some common performance problems that can arise:

- "The system is getting increasingly slower"
- "You experience configuration problems with a large number of components"
- "Remote waitForMessages calls with WebSphere MQ Everyplace are slow" on page 82
- "You are experiencing poor performance in the workbench when working with large projects" on page 82
- "The PutTime reported by WebSphere MQ on z/OS, and other times or timestamps are inconsistent" on page 82

The system is getting increasingly slower

- **Scenario:** The system is getting increasingly slower.
- **Explanation:** When a process ends abnormally, an entry is made in the local error log, and a dump data file might be written to the errors or z/OS log directory. This directory is unbounded, and if you do not clear it of unwanted files, you might find that your system performance degrades due to significant space being used by old abend files.
- **Solution:** Periodically clear the errors or z/OS log directory of unwanted files.

  On Windows, use the **-w** parameter of the **mqsicreatebroker** command to create the errors directory in a hard drive partition that does not contain WebSphere Event Broker or Windows itself.

You experience configuration problems with a large number of components

- **Scenario:** You are experiencing configuration problems with a large number of components.
- **Explanation:** If you are running WebSphere Event Broker with a large number of configured components, the memory footprint of the broker processes (particularly the DataFlowEngine) might exceed their memory limits. In

particular, the user process limit might be exceeded, or the address space limit might be reached. You might encounter problems, such as the BIP2106E error message, when running a broker with:

– A large number of message flows

– Very large input or output messages

- **Solution:** Use tools that are specific to your operating system to check the maximum size of the failing process, then check for any user limits (if applicable) or computer limits on process size.

  1. Use the **ulimit** command to check user limits on Linux or UNIX systems and increase these limits if needed.

     A hard limit is set on each operating system for the maximum process size, beyond which failure is inevitable:

Operating system	Maximum process size
Windows Windows	2 GB
Solaris Solaris (32 bit system)	About 3.75 GB
HP-UX HP-UX (32 bit system)	About 3 GB, depending on kernel settings
AIX AIX (32 bit system)	2 GB
z/OS z/OS	2 GB

     Increasing user limits beyond these values does not make any difference. If your broker processes regularly reach these sizes, consider spreading your message flows across more execution groups to reduce the size of each one below these limits.

  2. On AIX, WebSphere Event Broker limits 32-bit DataFlowEngines to four 256 MB memory segments; that is, 1 GB. Normally DataFlowEngine processes do not require more than 1 GB of the AIX address space, therefore the DataFlowEngine executable file is linked to have a default address space of four segments. However, you can extend the size of the AIX address space available to the DataFlowEngine process by increasing the number of 256 MB memory segments available to it. You can patch the DataFlowEngine to use more memory segments using the following shell command:

     ```
     /usr/bin/echo '\0140\0\0\0'|dd of=executable_file_name bs=4 count=1 seek=19
     conv=notrunc
     ```

     This command creates a version of the DataFlowEngine that uses six segments; that is, 1.5 GB.

     Alternatively, on AIX 5.2 and above, you can achieve the same result by using the **ldedit** command:

     ```
     ldedit -b maxdata:0x60000000 DataFlowEngine
     ```

     The application of fix pack maintenance replaces the existing DataFlowEngine, so if you have used the above process, repeat it after every fix pack installation.

     Another way of enabling the AIX "large address space" model is to set the environment variable `LDR_CNTRL`:

```
export LDR_CNTRL=MAXDATA=0x10000000
start_process (for example, mqsistart brokername)
unset LDR_CNTRL
```

or

```
LDR_CNTRL=MAXDATA=0x40000000 mqsistart brokername
```

Each increment in the value of MAXDATA and, therefore, the number of segments, increases the process memory limit by 256 MB.

The technique described above overrides only the soft limit and not the hard limit. If your broker processes regularly reach these sizes, consider spreading your message flows across more execution groups to reduce the size of each one to be below these limits.

Remote waitForMessages calls with WebSphere MQ Everyplace are slow

- **Scenario:** Remote waitForMessages calls with WebSphere MQ Everyplace are slow.
- **Explanation:** A remote waitForMessages() call necessarily results in a polling action: the client queue manager attempts to get a message repeatedly until one is available on the remote queue or a timeout is reached.
- **Solution:** Define a store-and-forward queue on the broker's queue manager and a home-server queue on the client. These queues provide a level of de-coupling that allows the arrangement to work in the event of the client queue manager becoming unavailable. Alternatively, specify a remote queue in the WebSphere MQ Everyplace output node, so that a local GET call is sufficient on the client side.

You are experiencing poor performance in the workbench when working with large projects

- **Scenario:** You are experiencing poor performance in the workbench when working with large or complex projects.
- **Explanation:** Performance is reduced because of frequent project changes, such as adding and removing projects, or using **Project → Clean**. Complete project updates use large amounts of memory due to the size, number, and connections between files.
- **Solution:** Increase your system memory.

The PutTime reported by WebSphere MQ on z/OS, and other times or timestamps are inconsistent

- **Scenario:** The PutTime reported by WebSphere MQ on z/OS, and other times or timestamps are inconsistent. A difference of approximately 20 seconds is detected in:
  - Traces (including those obtained from the Trace node)
  - The MQPUTTIME timestamp in the message MQMD header
  - Timestamps obtained from ESQL (for example, in a Compute node)
- **Explanation:** WebSphere Event Broker reports the time using Coordinated Universal Time (UTC), which does not account for leap seconds. However, on z/OS, the message putTime that is reported by WebSphere MQ in the MQMD header of a message *does* account for leap seconds, using the value specified for the number of leap seconds in the CVT field.

  This inconsistency can cause:
  - Problems when debugging

  - Problems with message flows if you use timestamps to control the flow of messages
  - Misinformation
- **Solution:** Set the CVT field so that it agrees with the UTC leap seconds. Alternatively, add an offset to adjust a z/OS timestamp reading. For example, add 20 seconds when getting the CURRENT_TIME in ESQL.

Resolving problems when developing Configuration Manager Proxy applications

This topic contains advice for dealing with some common problems that can arise when developing Configuration Manager Proxy (CMP) applications:

- "Your Configuration Manager Proxy application hangs if the Configuration Manager is not available"
- "You set a property of an object and query its value, but the value has not changed"
- "You cannot connect to a Configuration Manager using .CONFIGMGR files" on page 84
- "You tried to move a broker between Configuration Managers and now you cannot see the broker's resources" on page 84

Your Configuration Manager Proxy application hangs if the Configuration Manager is not available

- **Scenario:** When the Configuration Manager is unavailable, the Configuration Manager Proxy application hangs.
- **Explanation:** Communication between the Configuration Manager Proxy and the Configuration Manager is asynchronous, so the Configuration Manager Proxy hangs because it is waiting for a message from the Configuration Manager.
- **Solution:** Configure the maximum amount of time that the Configuration Manager Proxy waits by using the following method:

  ```
  // Wait for a maximum of 10 seconds
  ConfigManagerProxy.setRetryCharacteristics(10000);
  ```

  This represents the time in milliseconds that the Configuration Manager Proxy will wait for information before throwing the *ConfigManagerProxyPropertyNotInitializedException* exception.

  If you set this timeout value too low, an exception will be thrown even if the Configuration Manager is available.

You set a property of an object and query its value, but the value has not changed

- **Scenario:** You have set a property of an object, then queried its value; the value has not changed.
- **Explanation:** Methods that change properties of domain objects are not processed immediately. If you call a property change method on a Configuration Manager Proxy object, this causes the Configuration Manager Proxy to send a message to the Configuration Manager asking it to make the requested change. The Configuration Manager processes the request asynchronously and notifies any *AdministeredObjectListeners* of the affected object when the change has been attempted.
- **Solution:** Methods that change state usually return as soon as the request has been put to the Configuration Manager's queue manager, or following a call to *ConfigManagerProxy.beginUpdates()*, as soon as the request has been added to the

current batch. If the property has still not been updated after the action's response to the request has been returned to the Configuration Manager Proxy, consult the response message for more details.

You cannot connect to a Configuration Manager using .CONFIGMGR files

- **Scenario:** You cannot connect to a Configuration Manager using .CONFIGMGR files.
- **Explanation:** The *MQPropertyFileConfigManagerConnectionParameters* class allows Configuration Manager Proxy applications to connect to Configuration Managers using files with a .configmgr extension, although it requires an XML parser in order to do this.
- **Solution:** Ensure that a supported parser is available on the CLASSPATH. A supported parser is shipped with the product.

  Alternatively, use the *MQConfigManagerConnectionParameters* class instead of the *MQPropertyFileConfigManagerConnectionParameters* class. This allows you to connect to Configuration Managers by specifying the host name, queue manager name and port of the target Configuration Manager directly. This method does not require an XML parser.

You tried to move a broker between Configuration Managers and now you cannot see the broker's resources

- **Scenario:** You used the BrokerProxy.setUUID() method to attempt to move a broker from one Configuration Manager to another and you can no longer see the broker's execution groups, message flows, and subscriptions.
- **Explanation:** This is a limitation of the setUUID() method.
- **Solution:** Remove all deployed resources and redeploy them to the broker.

Resolving problems with user-defined extensions

Advice for dealing with some common problems that can arise when you work with user-defined extensions

- "You cannot deploy one of your user-defined nodes, despite having a plug-in LIL in the correct directory."
- "You cannot deploy a flow with one of your user-defined nodes in it." on page 85
- "An error is issued when you deploy a user-defined extension on z/OS" on page 85

You cannot deploy one of your user-defined nodes, despite having a plug-in LIL in the correct directory.

- **Scenario:** You cannot deploy one of your user-defined nodes, despite having a plug-in LIL in the correct directory.
- **Explanation:** You have `memset()` the data area to zero and have not initialized the CNI_VFT structure with the initialization constant `{CNI_VFT_DEFAULT}`.
- **Solution:** Initialize by copying a predefined initialization structure over the function table area, as follows:

  ```
  static CNI_VFT virtualFunctionTable = {CNI_VFT_DEFAULT};
  ```

  In addition, implement logging from your user-defined node so that you can see if the plug-in API is producing error codes; the broker does not log these to its own log, unless you take a service trace.

You cannot deploy a flow with one of your user-defined nodes in it.

- **Scenario:** You cannot deploy a flow with one of your user-defined nodes in it.
- **Explanation:** Your LIL file has failed to load.
- **Solution:** Check system log (syslog or Eventviewer) of broker startup; did you see a BIP2308 message saying a LIL file failed to load? If there are any problems loading a LIL file, a BIP2308 message appears in the system log.

An error is issued when you deploy a user-defined extension on z/OS

- **Scenario:** When you deploy a user-defined extension on z/OS, Linux, or UNIX, an error is displayed in the log of each execution group, stating that there is insufficient authority to open the LIL file.
- **Explanation:** On Linux and UNIX, the user-defined extension must have group read permission. On z/OS, the user-defined extension must have group execute permission.
- **Solution:**
  - On Linux and UNIX, set the file permissions of the user-defined extension to group read by issuing the command **chmod a+r**.
  - On z/OS, set the file permissions of the user-defined extension to group read and execute by issuing the command **chmod a+rx**.

Resolving problems when uninstalling

Work through the advice provided to help you to deal with problems that can arise when the product is uninstalled.

- Uninstalling the Message Brokers Toolkit:
  - "RPU shortcut is removed during uninstall"
  - "Files are left behind after uninstall completes"
  - "Uninstall wizard reports an error" on page 86
- Uninstalling runtime components:
  - "The uninstallation process is interrupted" on page 87

RPU shortcut is removed during uninstall

- **Scenario:** When you uninstall the Message Brokers Toolkit on Linux on x86, the IBM Rational® Product Updater (RPU) menu shortcut is removed.
- **Explanation:** A menu entry is created for Rational Product Updater (RPU) when you install any Rational product, including the Message Brokers Toolkit. You can use this entry to start the RPU to update Rational products. When you uninstall the Message Brokers Toolkit, this entry is removed even if other Rational products are installed on the system. If you uninstall the last Rational product, but retain the Message Brokers Toolkit, the same problem occurs.
- **Solution:** Select **Help → Software Updates → IBM Rational Product Updater** to access the RPU from within the Message Brokers Toolkit or other installed Rational products. You cannot recreate the menu entry.

Files are left behind after uninstall completes

- **Scenario:** When you uninstall the Message Brokers Toolkit on Linux on x86 or Windows, the wizard completes successfully but some files and directories are left behind in the installation directory.
- **Solution:** Follow the instructions below for the scenario that reflects your system:

– **The Message Brokers Toolkit was the only Rational product installed on this system**

  When the uninstallation wizard for the Message Brokers Toolkit has completed, delete all remaining files and directories in the installation directory.

– **Other Rational products are installed on this system**

  When the uninstallation wizard for the Message Brokers Toolkit has completed, do not remove any files or directories in the installation directory.

  You can check whether other RAD products are still installed on the system:

  1. Navigate to the following directory:
     - Linux On Linux on x86: /etc/IBM/RAT60/.sdpinst
     - Windows On Windows: *install_dir*\IBM\RAT60\.sdpinst
  2. Look for the properties file cdi_ref.properties:
     - If the file exists, and its size is greater than zero bytes, RAD products are installed on the system. Do not delete any files or directories.
     - If the file does not exist, or it is present but has a size of zero bytes, no RAD products are installed. Delete the files and directories that have been left behind when the Message Brokers Toolkit was uninstalled.

     Do not manipulate this properties file in any way; check only its size and existence.

Uninstall wizard reports an error

- **Scenario:** When you uninstall the Message Brokers Toolkit on Linux on x86 or Windows, the wizard reports an error and indicates that you should look in the log file.
- **Explanation:** During uninstallation, the Rational product code records the following error in the log file *home_directory*/webt_uninstall.log. (To improve readability, lines in the record have been split but content is unchanged.)

```
14-Nov-2005 1:12:18 AM), Uninstall, com.installshield.wizard.platform.win32.Win32ProductServiceImpl,
msg1, uninstalling CDI Delete Directory (b_updater_start_cdi_refcount_info)
(14-Nov-2005 1:12:18 AM), Uninstall, com.installshield.wizard.platform.win32.Win32ProductServiceImpl,
msg1, uninstalling CDI Delete Directory (b_updater_start_updater_workspace)
(14-Nov-2005 1:12:19 AM), Uninstall, com.installshield.wizard.platform.win32.Win32ProductServiceImpl,
err, An error occurred and product uninstallation failed.
Look at the log file C:\Documents and Settings\Administrator/wmbt_uninstall.txt for details.
(14-Nov-2005 1:12:20 AM), Uninstall, com.ibm.rational.cdi.wrapper.CDIDeleteDirectory,
err, ProductException: (error code = 601; message="err";
additional data = [Unable to delete C:\Program Files\IBM\MessageBrokersToolkit\6.0\.\updater\workspace :
ServiceException: (error code = -30020; message = "The system cannot find the path specified.
 (3)"; severity = 0)])
STACK_TRACE: 13
ProductException: (error code = 601; message="err";
additional data = [Unable to delete C:\Program Files\IBM\MessageBrokersToolkit\6.0\.\updater\workspace :
ServiceException: (error code = -30020; message = "The system cannot find the path specified.
 (3)"; severity = 0)])
at com.installshield.product.actions.DeleteDirectory.deleteDirectory(DeleteDirectory.java:82)
at com.installshield.product.actions.DeleteDirectory.uninstall(DeleteDirectory.java:39)
at com.ibm.rational.cdi.wrapper.CDIDeleteDirectory.uninstall(CDIDeleteDirectory.java:124)
at com.installshield.product.service.product.PureJavaProductServiceImpl.
 uninstallProductAction(PureJavaProductServiceImpl.java:1947)
at com.installshield.product.service.product.PureJavaProductServiceImpl.
 uninstallProductComponent(PureJavaProductServiceImpl.java:2521)
at com.installshield.product.service.product.PureJavaProductServiceImpl.
 uninstallSoftwareObject(PureJavaProductServiceImpl.java:2406)
at com.installshield.product.service.product.PureJavaProductServiceImpl.
 access$8(PureJavaProductServiceImpl.java:2346)
at com.installshield.product.service.product.PureJavaProductServiceImpl$Uninstaller.
 execute(PureJavaProductServiceImpl.java:3870)
```

```
at com.installshield.wizard.service.AsynchronousOperation.run(AsynchronousOperation.java:41)
at java.lang.Thread.run(Thread.java:566)

(14-Nov-2005 1:12:20 AM), Uninstall, com.installshield.wizard.platform.win32.Win32ProductServiceImpl,
msg1, uninstalling CDI Delete File (b_updater_start_delete_updater_flag)
(14-Nov-2005 1:12:20 AM), Uninstall, com.installshield.wizard.platform.win32.Win32ProductServiceImpl,
msg1, uninstalling CDI Reference Install (b_updater_start_install)
```

- **Solution:** This error is temporary, and you can safely ignore it. The uninstallation has completed.

The uninstallation process is interrupted

- **Scenario:** When you uninstall the runtime components on any distributed system, the process is interrupted, for example by a power failure.
- **Solution:** Delete the *install_dir* and all its contents. You can now reinstall product components if you choose.

Using logs

There are a variety of logs that you can use to help with problem determination and troubleshooting.

This section describes how to view the various logs available to you with WebSphere Event Broker, and how to interpret the information in those logs. It contains the following topic areas:

Local error log

- "Windows: Viewing the local error log"
- "Linux and UNIX systems: Configuring the syslog daemon" on page 88

Eclipse log

- "Viewing the Eclipse error log" on page 89

There is also a section of reference topics about the various types of log.

Windows: Viewing the local error log

The Windows Event Viewer is where WebSphere Event Broker writes records to the local system. Use Windows system facilities to view this log.

Viewing the system log

The system log contains events logged by the Windows system components. For example, the failure of a driver or other system component to load during startup is recorded in the system log. To view the system log:

1. Open a command prompt.
2. At the prompt, type `eventvwr`. This opens the Windows Event Viewer.
3. In the left pane of this window, click the **System Log** (this is often selected as default). All the events that have been written to the local system are displayed.

Viewing the application log

The application log contains events that are logged by applications or programs. For example, a database program might record a file error in the application log. To view the application log:

1. Open a command prompt.
2. At the prompt, type `eventvwr`. This opens the Windows Event Viewer.

3. In the left pane of this window, click **Application Log**. This log is where all the events that have been logged by applications or programs are displayed.

Interpreting log information

In both logs, each event is displayed on a separate row, in date and time order (most recent first), with the following information:

- **Type:** The event type, which can be information, a warning, or an error.
- **Date and time:** The date and time when the event was written to the log.
- **Source:** What action has caused the event.
- **Category:** The category of the event. The default category is `none`.
- **Event:** The event number.
- **User:** The name of the user at the time of the event.
- **Computer:** The name of the local machine.

To view an individual log entry:

1. Within the system or application log, find the log entry.
2. Right-click the entry and click **Properties**. This option opens the Information Properties window, where there is a description of the event. There is also a Data section that details bytes or words that were parsed when the record was written to the log.
3. In the Information Properties window, use the up and down arrows to move through the events of the log.
4. To close the Information Properties window, click **OK** to return to the system or application log.

Linux and UNIX systems: Configuring the syslog daemon

UNIX On Linux and UNIX systems, all WebSphere Event Broker messages (other than those generated by the command line utilities) are sent to the syslog, so it is useful to redirect user messages to a separate file.

On UNIX, syslog entries are restricted in length and messages that are sent to the syslog are truncated by the new line character. To record a large amount of data in a log on UNIX, set the Destination property on the Trace node to File or User Trace instead of Local Error Log.

Before you create a broker on Linux or UNIX systems, configure the syslog daemon to redirect user messages to a file called user.log:

1. Log on as root.
2. Enter the following commands to create a file called user.log.

   On UNIX systems, enter the command:

   ```
   touch /var/adm/user.log
   chown root:mqbrkrs /var/adm/user.log
   chmod 640 /var/adm/user.log
   ```

   On Linux, enter the command:

   ```
   touch /var/log/user.log
   chown root:mqbrkrs /var/log/user.log
   chmod 640 /var/log/user.log
   ```

3. Add the following line to the /etc/syslog.conf file (on later versions of SUSE Linux, this is /etc/syslog-ng.conf) to redirect debug level messages to the file user.log:

On UNIX systems, enter the command:

```
user.info /var/adm/user.log
```

On Linux, enter the command:

```
user.info /var/log/user.log
```

You can use `user.* -` instead of `user.info` in the preceding examples.

- `*` means that information, notice, warning, and debug messages are caught
- `-` means that `syslog` does not synchronize the file after writing to it.

You might experience a performance gain, but you can lose some data if the computer fails immediately after it has written to the file.Add a line to ensure that messages at the required level from the user facility are recorded. If you specify a level of `info`, all operational messages are recorded; these messages provide important information about the operation of the broker, and can be useful in diagnosing problems.

4. Restart the syslog daemon.

   On AIX, enter the command:

   ```
   refresh -s syslogd
   ```

   On HP-UX and Solaris, enter the command:

   ```
   kill -HUP 'cat /etc/syslog.pid'
   ```

   On Linux, enter the command:

   ```
   /etc/init.d/syslogd restart
   ```

   or

   ```
   /etc/rc.d/init.d/syslogd restart
   ```

   for systems where `rc.d` is not a soft link

For other syslog options, see the documentation for your operating system.

Viewing the Eclipse error log

The Eclipse error log captures internal errors that are caused by the operating system or your code.

To view the Eclipse error log:

1. Switch to the Plug-in Development perspective.
2. In the Tasks view, in the bottom right pane, click the **Error Log** tab.

   The error log is displayed, showing the following information for each error:

   - The status of the error (for example, error or warning)
   - A brief description of the error
   - From which plug-in the error derived
   - The date and time that the error was produced
3. If an error has a plus sign (+) at the start of it, it is a complex problem, and there are a number of errors contributing to it. Click the plus sign to view the individual errors.
4. To see the details of a particular problem, double-click the entry in the Tasks view. A separate window is displayed, showing more details of the error.

Using trace

You can use different types of trace to help you with problem determination and troubleshooting.

How to use the optional trace.

For user trace:

- "Starting user trace"
- "Checking user trace options" on page 91
- "Changing user trace options" on page 92
- "Retrieving user trace" on page 93
- "Stopping user trace" on page 93

For service trace:

- "Starting service trace" on page 94
- "Checking service trace options" on page 94
- "Changing service trace options" on page 95
- "Retrieving service trace" on page 96
- "Stopping service trace" on page 95

For both types of trace:

- "Formatting trace" on page 96
- "Interpreting trace" on page 98
- "Clearing old information from trace files" on page 98
- "Changing trace settings from the workbench" on page 99

Other types of trace:

- "ODBC trace" on page 100
- "WebSphere MQ Java Client trace" on page 103
- "Configuration Manager Proxy trace" on page 103

You can also use the "IBM Support Assistant (ISA)" on page 108 to help with data collection.

Starting user trace

Before you start:

Before you start to trace a broker, or any of its execution groups or message flows, the broker must be running, and you must have deployed the message flows using the workbench. Follow the instructions in the Deploying topic.

To start a user trace:

1. Start WebSphere Event Broker user trace facilities by using the **mqsichangetrace** command, or, for execution groups and assigned message flows, from the workbench. You can select only one broker on each invocation of the command, but you can activate concurrent traces for more than one broker, by invoking the command more than once.
2. Specify an individual execution group or message flow within the specified broker to limit the scope of a trace. The events that are recorded when you select the message flow option include:
   - Sending a message from one Message Processing node to the next

3. Start your trace. You can start trace at two levels:

   normal This tracks events that affect objects that you create and delete, such as nodes.

   debug This tracks the beginning and end of a process, as well as monitoring objects that are affected by that process.

Example: starting user trace for the default execution group

To start normal level user tracing for the default execution group on a broker that you have created with the name WBRK_BROKER, on distributed systems, enter the command

```
mqsichangetrace WBRK_BROKER -u -e default -l normal
```

where:

-u specifies user trace
-e specifies the execution group (in this case, the default execution group)
-l specifies the level of trace (in this case, normal)

z/OS On z/OS, enter the command

```
F MQP1BRK,ct u=yes, e='default', l=normal
```

Example: starting user trace for the default execution group from the workbench

To start normal level user tracing for the default execution group from the workbench:

1. Switch to the Broker Administration perspective.
2. In the Domains view, right-click the default execution group.
3. Click **User trace** → **Normal**.

Example: starting user trace for a message flow from the workbench

To start normal level user tracing for one of your message flows from the workbench:

1. Switch to the Broker Administration perspective.
2. In the Domains view, right-click the message flow for which you want to start trace.
3. Click **User trace** → **Normal**.

An alert saying `Message Flow is tracing at level 'normal'` is displayed in the Alert Viewer.

Checking user trace options

Use the **mqsireporttrace** command to check what tracing options are currently active for your brokers.

Specify the component for which the check is required, for example, the execution group or message flow. The command responds with the current trace status for the component that you have specified.

Example: checking user trace options for a broker

To check what options are currently set for the broker WBRK_BROKER and its execution group test, on distributed systems, enter the command

```
mqsireporttrace WBRK_BROKER -u -e test
```

where:

-u specifies user trace
-e specifies the execution group (in this case, test)

z/OS On z/OS, enter the command

```
F MQP1BRK,reporttrace u=yes, e='test'
```

If you have started tracing by following the example in "Starting user trace" on page 90, the response to the **mqsireporttrace** command is:

```
BIP8098I: Trace level: normal, mode: safe, size: 1024 KB
BIP8071I: Successful command completion
```

Changing user trace options

Use the **mqsichangetrace** command to change the trace options that you have set. You can also use the workbench to change the trace options for execution groups and assigned message flows.

Example: changing user trace from normal to debug

To change from a normal level of user trace to a debug level on the default execution group of a broker called WBRK_BROKER, on distributed systems, enter the command

```
mqsichangetrace WBRK_BROKER -u -e default -l debug
```

where:

-u specifies user trace
-e specifies the execution group (in this case, the default execution group)
-l specifies the level of trace (in this case, changing it to debug)

z/OS On z/OS, enter the command

```
F MQP1BRK,ct u=yes, e='default', l=debug
```

Example: changing user trace from normal to debug from the workbench

To change from a normal level of user trace to a debug level for one of your message flows from the workbench:

1. Switch to the Broker Administration perspective.
2. In the Domains view, right-click the message flow for which you want to change trace.
3. Click **User trace** → **Debug**.

An alert saying `Message Flow is tracing at level 'debug'` is displayed in the Alert Viewer.

Stopping user trace

Use the **mqsichangetrace** command with a trace level of `none` to stop an active trace. This stops the trace activity for the component that you specify on the command. It does not affect active traces on other components. For example, if you stop tracing on the execution group `test`, an active trace on another execution group continues.

You can also stop an active trace for execution groups or assigned message flows using the workbench. If you redeploy a component from the workbench, trace for that component is returned to its default setting of `none`.

Example: stopping user trace on the default execution group

To stop the trace started by the command shown in "Starting user trace" on page 90, on distributed systems, enter the command

```
mqsichangetrace WBRK_BROKER -u -e default -l none
```

where:

-u specifies user trace
-e specifies the execution group (in this case, the default execution group)
-l specifies the level of trace (in this case, none)

z/OS On z/OS, enter the command

```
F MQP1BRK,ct u=yes, e='default', l=none
```

Example: stopping user trace on a message flow from the workbench

To stop trace for one of your message flows from the workbench:

1. Switch to the Broker Administration perspective.
2. In the Domains view, right-click the message flow for which you want to stop trace.
3. Click **User trace** → **None**.

Retrieving user trace

Use the **mqsireadlog** command to access the trace information that is recorded by the user trace facilities. This command retrieves the trace details according to parameters that you specify on the command, and writes the requested records to a file, or to the command line window, in XML format.

Example: retrieving user trace information in XML format

To retrieve information for the user trace activated with the **mqsichangetrace** command and write it to an output file, on distributed systems, enter the command

```
mqsireadlog WBRK_BROKER -u -e default -o trace.xml
```

where:

-u specifies user trace
-e specifies the execution group (in this case, the default execution group)
-o specifies the output file (in this case, trace.xml)

This sends a log request to the broker to retrieve the user trace log, and stores the responses in the trace.xml file. You can view this file using a plain text editor.

Starting service trace

Activate service traces only when you receive an error message that instructs you to start service trace, or when directed to do so by your IBM Support Center.

Use the **mqsichangetrace** command to start WebSphere Event Broker service trace facilities.

You can select only one broker on each invocation of the command, but you can activate concurrent traces for more than one broker by invoking the command more than once.

You must specify the individual component that you want to trace; for example the broker, Configuration Manager, or User Name Server, to limit the scope of a trace.

If the trace cannot be associated with a specific component, the component name part of the file name is set to `utility`, for example, when tracing a command such as **mqsilist**, when no arguments are used.

In order to trace the **mqsideploy**, **mqsistartmsgflow**, **mqsistopmsgflow**, **mqsicreateexecutiongroup**, and **mqsideleteexecutiongroup** commands, use the **-v** parameter. This takes an argument that is the name of the file to which trace will be written.

If you want to trace the command executable files themselves, set the environment variables MQSI_UTILITY_TRACE and MQSI_UTILITY_TRACESIZE before you initiate trace. Ensure that you reset these variables when the command that you are tracing has completed. If you do not do so, all subsequent commands are also traced, and their performance is therefore degraded.

Example: starting service trace for the broker

To start debug level service tracing for the broker `A` on distributed systems, enter the command

```
mqsichangetrace BrokerA -t -b -l debug
```

where:

-t specifies service trace
-b specifies that trace for the `agent` subcomponent of the specified component is to be started
-l specifies the level of trace (in this case, debug)

z/OS On z/OS, enter the command

```
F MQPIBRK,ct t=yes, b=yes, l=debug
```

Checking service trace options

Use the **mqsireporttrace** command to check what tracing options are currently active for your brokers.

Specify the component for which the check is required. The command responds with the current trace status for the component that you have specified.

Example: checking service trace options for a broker

To check what options are currently set for the User Name Server, on distributed systems, enter the command

```
mqsireporttrace UserNameServer -t
```

where **-t** specifies service trace.

z/OS On z/OS, enter the command

```
F MQP1UNS,reporttrace t=yes
```

If you have started tracing by following the example in "Starting service trace" on page 94, the response to the **mqsireporttrace** command is:

```
BIP8098I: Trace level: debug, mode: safe, size: 1024 KB
BIP8071I: Successful command completion
```

Changing service trace options

Use the **mqsichangetrace** command to change the service trace options that you have set.

Example: changing service trace from debug to normal

To change from a debug level of trace to a normal level on the User Name Server, on distributed systems, enter the command

```
mqsichangetrace UserNameServer -t -b -l normal
```

where:

-t specifies service trace
-b specifies that tracing for the `agent` subcomponent of the specified component is to be changed
-l specifies the level of trace (in this case, changing it to normal)

z/OS On z/OS, enter the command

```
F MQP1UNS,ct t=yes, b=yes, l=normal
```

Stopping service trace

Use the **mqsichangetrace** command with a trace level of `none` to stop an active trace. This stops the trace activity for the component that you specify on the command. It does not affect active traces on other components. For example, if you stop tracing on the execution group `test`, an active trace on another execution group continues.

If you redeploy a component from the workbench, trace for that component is returned to its default setting of `none`.

Example: stopping service trace on the User Name Server

To stop the trace started by the command shown in "Starting service trace" on page 94, on distributed systems, enter the command

```
mqsichangetrace UserNameServer -t -b -l none
```

where:

-t specifies service trace
-b specifies that trace for the `agent` subcomponent of the specified component is to be stopped
-l specifies the level of trace (in this case, none)

z/OS On z/OS, enter the command

```
F MQP1UNS,ct t=yes, b=yes, l=none
```

Retrieving service trace

Use the **mqsireadlog** command to access the trace information recorded by the service trace facilities.
This command retrieves the trace details according to parameters that you specify on the command, and writes the requested records to a file, or to the command line window, in XML format.

Example: retrieving service trace information in XML format

To retrieve information for the service trace activated with the **mqsichangetrace** command, and write it to an output file, on distributed systems, enter the command

```
mqsireadlog UserNameServer -t -b agent -f -o strace.xml
```

where:

-t specifies service trace
-b *agent* specifies that trace for the `agent` subcomponent of the specified component is to be retrieved
-f specifies that the log file is to be read directly from the file system (this flag is mandatory for service trace)
-o specifies the output file (in this case, strace.xml)

This sends a log request to the broker to retrieve the service trace log, and stores the responses in the file strace.xml. You can view this file using a plain text editor.

Formatting trace

The trace information that is generated by the **mqsireadlog** command is not easy to read unless you use an XML viewer (such as an Internet browser) or an XML editor that understands the document type descriptor (DTD) in the file.

WebSphere Event Broker provides the command **mqsiformatlog** to format the trace information to a flat file, so that you can view it using a text editor.

The **mqsiformatlog** command takes a file generated by the **mqsireadlog** command as input, and flattens the XML log into structured records. It also retrieves the

inserts for the XML message in your current locale. You can direct the formatted output to a file, or view it in the command line window.

Each trace entry contains a time stamp and a WebSphere Event Broker message that contains a number (for example, BIP2622) and a text string containing variable inserts.

Example: formatting user trace information on Windows

Windows On Windows, to format the trace file that is created in "Starting user trace" on page 90, enter the command

```
mqsiformatlog -i trace.xml -o formattrace.log
```

where:

-i specifies the input file (in this case, trace.xml)
-o specifies the output file (in this case, formattrace.log)

This command reads the trace information in the file trace.xml, formats it, and writes it to the file formattrace.log. The following example shows a portion of the output of the **mqsiformatlog** command for a normal level trace file.

```
Timestamps are formatted in local time, 330 minutes past GMT.

2003-06-19 11:30:29.795999     2852   UserTrace   BIP2632I: Message received and
 propagated  to 'out' terminal of MQ Input node 'Video_Test.VIDEO_XML_IN'.
2003-06-19 11:30:29.795999     2852   UserTrace   BIP6060I: Parser type 'Properties'
 created on behalf of node 'Video_Test.VIDEO_XML_IN' to handle portion of incoming
 message of length
 0 bytes beginning at offset '0'.
2003-06-19 11:30:29.795999     2852   UserTrace   BIP6061I: Parser type 'MQMD'
 created on behalf of node 'Video_Test.VIDEO_XML_IN' to handle portion of incoming
 message of length '364' bytes beginning at offset '0'. Parser type selected based
 on value 'MQHMD' from previous parser.
2003-06-19 11:30:29.795999     2852   UserTrace   BIP6061I: Parser type 'MRM'
 created on behalf of node 'Video_Test.VIDEO_XML_IN' to handle portion of incoming
 message of length '650' bytes beginning at offset '364'. Parser type selected based
 on value 'MRM' from previous parser.
2003-06-19 11:30:29.795999     2852   UserTrace   BIP2537I: Node 'Video_Test.Extract
 Fields':
 Executing statement 'BEGIN ... END;' at (.Video_Test_Compute.Main, 2.2).
2003-06-19 11:30:29.795999     2852   UserTrace   BIP2537I: Node 'Video_Test.Extract
 Fields':
 Executing statement 'SET OutputRoot = InputRoot;' at (.Video_Test_Compute.Main, 7.3).
2003-06-19 11:30:29.795999     2852   UserTrace   BIP2538I: Node 'Video_Test.Extract
 Fields':
 Evaluating expression 'InputRoot' at (.Video_Test_Compute.Main, 7.20).
2003-06-19 11:30:29.795999     2852   UserTrace   BIP2568I: Node 'Video_Test.Extract
 Fields':
 Performing tree copy of 'InputRoot' to 'OutputRoot'.

...

2003-06-19 11:30:29.827999     2852   UserTrace   BIP4124I: Message propagated to
 'out' terminal of Compute node 'Video_Test.Extract Fields'.
2003-06-19 11:30:29.827999     2852   UserTrace   BIP2638I: The MQ Output node
 'Video_Test.VIDEO_OUT' attempted to write a message to queue 'VIDEO_OUT' connected
 to queue manager ''. The MQCC was '0' and the MQRC was '0'.
2003-06-19 11:30:29.827999     2852   UserTrace   BIP2622I: Message successfully
```

```
 output by output node 'Video_Test.VIDEO_OUT' to queue 'VIDEO_OUT' on queue manager ''.

Threads encountered in this trace:
  2852
```

Interpreting trace

A formatted log file, like the one in “Formatting trace” on page 96, contains a sequence of WebSphere Event Broker messages that records the activity in a specific part of the system (the part that you identify when you start the trace). You can use this sequence to understand what is happening, and to check that the behavior that is recorded is what you are expecting.

For example, if you have activated message flow trace, you can see entries that record the path that a message takes through the message flow, and why decisions result in this path (where a choice is available).

If you see unexpected behavior in a message flow or execution group, use this trace information to check the actions that have been taken and identify the source of an error or other discrepancy.

The messages contain identifiers for the resources that are being traced, for example the execution groups and message flows. The identifier that is given is usually the label (the name) that you gave to the resource when you defined it.

Here is an extract from a user trace file. In the example, each column has been labeled:

```
Timestamp                Thread ID Trace type  Message
2005-07-12 16:17:18.242605   5344   UserTrace   BIP2537I: Node 'Reply.MapToRequestor':
                                                Executing statement   ''SET I = I + 1;''
                                                at ('.MapToRequestor.CopyMessageHeaders',
                                                '6.4').
2005-07-12 16:17:18.242605   5344   UserTrace   BIP2539I: Node 'Reply.MapToRequestor':
                                                Evaluating expression ''I'' at
                                                ('.MapToRequestor.CopyMessageHeaders',
                                                '6.12'). This resolved to ''I''. The
                                                result was ''1''.
2005-07-12 16:17:18.242605   5344   UserTrace   BIP2539I: Node 'Reply.MapToRequestor':
                                                Evaluating expression ''I + 1'' at
                                                ('.MapToRequestor.CopyMessageHeaders',
                                                '6.14'). This resolved to ''1 + 1''.
                                                The result was ''2''.
2005-07-12 16:17:18.242605   5344   UserTrace   BIP2566I: Node 'Reply.MapToRequestor':
                                                Assigning value ''2'' to field / variable
                                                ''I''.
```

References such as `'6.12'` apply to the row and column number within a function that specify the location of the command that is being executed; in this case, row 6, column 12.

Clearing old information from trace files

If the component that you are tracing has stopped, you can delete its trace files from the log subdirectory of the WebSphere Event Broker home directory.

If you are tracing an execution group, you can use the **-r** parameter of the **mqsichangetrace** command to reset (clear) the trace log (the **-r** parameter can be

specified only if you specify the **-e** parameter). You might do this, for example, when you start a new trace to ensure that all the records on the log are unique to the new trace.

Example: clearing the user trace log for the default execution group

To clear the user trace log for the default execution group, on distributed systems, enter the command: **mqsichangetrace WBRK_BROKER -u -e default -r** where:

WBRK_BROKER specifies the name of the broker
-u specifies user trace
-e specifies the execution group (in this case the default execution group)
-r clears the trace log

z/OS On z/OS, enter the command `F MQP1BRK,ct u=yes, e='default', r=yes`

Changing trace settings from the workbench

In addition to collecting user trace and service trace for various components, you can also collect the following additional types of trace from the workbench:

- Trace for all WebSphere Event Broker components, including the workbench itself. This uses the logger.
- Configuration Manager Proxy trace
- Broker Administration trace

The following sections tell you how to change the settings for these types of trace.

Changing the logger settings

Before you change the settings, take these steps to enable logger settings to be displayed on the preferences screen:

1. Click **Window** → **Preferences** → **Workbench** → **Capabilities**.
2. Select the **Tester** check box.
3. Click **Apply**, and then **OK**.

WebSphere Event Broker provides a logger that controls tracing for all the components of WebSphere Event Broker. This is the com.ibm.etools.mft.util plug-in. By default, the logger is set to produce trace for severe errors, but you can change this level:

1. Click **Window** → **Preferences**.
2. In the left menu, click **Logging**.
3. Click **com.ibm.etools.mft.util**.
4. Click the **General** tab.
5. From the **Default logging level** list, click the logging level that you want. For the most detailed report, select **FINEST**.
6. Click **OK**.

   Changes are applied, and the Preferences window closes.

The files from this trace are written to the directory *install_dir*\eclipse\workbench\.metadata. Logging utility output is written to the LoggingUtil.log file. Any workbench errors are written to the .log file.

Changing the Configuration Manager proxy trace settings

You control tracing of the Configuration Manager proxy from the workbench:

1. Click **Window** → **Preferences**.
2. In the left menu, expand **Broker Administration**.
3. Click **Configuration Manager proxy**.
4. In the Configuration Manager proxy tracing section of the menu, change the level of tracing that is performed. The default is **Message Logging Utility**. If you select **Errors to file**, the default file to which trace is written is *install_dir*\eclipse\workspace\cmptraceerror.log.
5. Click **OK**.

   Changes are applied, and the Preferences window closes.

Changing the Broker Administration trace settings

You control tracing of the Broker Administration from the workbench:

1. Click **Window** → **Preferences**.
2. In the left menu, expand **Broker Administration**.
3. Click **Trace**.
4. Change the type of tracing. The default is **None**. If you select **Stored in a file**, the default file to which trace is written is *install_dir*/eclipse/workspace/.metadata/.plugins/com.ibm.etools.mft.admin/mbda.log.
5. Click **OK**.

   Changes are applied, and the Preferences window closes.

ODBC trace

There are various methods for initiating trace for ODBC activity, depending on the operating system that you are using.

Windows For Windows, use the **Tracing** tab of the ODBC function:

1. Click **Start** → **Settings** → **Control Panel** → **Administrative Tools**.
2. Double-click **Data Sources**.
3. Click the **Tracing** tab.
4. Click **Start Tracing Now**.
5. Click **OK**.

To stop ODBC tracing, on the **Tracing** tab, click **Stop Tracing Now**, then **OK**.

UNIX For Linux and UNIX operating systems, edit the ODBC.ini file to initiate trace for ODBC activity (if you are using the default ODBC.ini file, located at /var/wmqi/odbc/.odbc.ini):

1. Under the stanza entry [ODBC], change `Trace=0` to `Trace=1`
2. Optionally, modify the TraceFile file path value to a preferred value. All trace records go to one file, which can be confusing if you have multiple flows or execution groups.

Trace information is output to the location that is specified in the .odbc.ini file, or in the ODBC configuration on Windows. By default on non-Windows systems, this is /var/wmqi/odbc/odbctrace.out.

z/OS For z/OS, to initiate application trace for ODBC activity:

1. Edit the BIPDSNAO file in the component dataset and under the stanza entry [COMMON], change APPLTRACE=0 to APPLTRACE=1
2. Remove the comment from the COMPDIR variable declaration and the APPLTRACE DD in the WebSphere Event Broker started task JCL.
3. Stop and restart the broker after you have made the changes to the BIPDSNAO file and the started task JCL.

   By default, the trace output file is written to <component_HFS>/output/, into a file called db2appltrace.. Each address space has a unique number appended to the end of db2appltrace..

   This unique number, appended to the ODBC file, is the SE number in the execution group address space JOBLOG.

   You should look for the value of SE in the JOBLOG for which you want to view the ODBC trace, and find the file that is named db2appltrace.<SE number>.

DB2 on WebSphere Event Broker for z/OS

The following sample ODBC trace files show the layout of a trace file, together with some examples of successful and error returns. The general layout of each group in an ODBC file is that:

- Each line is preceded by a time stamp.
- The first line displays what the call does.
- The second line displays the return.
- The third line displays the result.

The first trace file shows a trace where a call fails because an object does not have the correct authority to perform an action:

```
[2008-09-24 15:49:20.544123] SQLAllocStmt( hDbc=2, phStmt=&1c7f9554 )
[2008-09-24 15:49:20.544156] SQLAllocStmt( phStmt=1 )
[2008-09-24 15:49:20.544163]     ---> SQL_SUCCESS

[2008-09-24 15:49:20.544179] SQLFreeStmt( hStmt=1, fOption=SQL_CLOSE )
[2008-09-24 15:49:20.544189] SQLFreeStmt( )
[2008-09-24 15:49:20.544194]     ---> SQL_SUCCESS

[2008-09-24 15:49:20.544205] SQLPrepare( hStmt=1 )
[2008-09-24 15:49:20.544212] ( pszSqlStr="SELECT TESTTABLE.ID FROM
WMQI77.TESTTABLE TESTTABLE", cbSqlStr=-3 )
[2008-09-24 15:49:20.587083] SQLPrepare( )
[2008-09-24 15:49:20.587101]     ---> SQL_ERROR

[2008-09-24 15:49:20.587157] SQLError( hEnv=0, hDbc=0, hStmt=1,
pszSqlState=&3902af28, pfNativeError=&3902af24, pszErrorMsg=&1b88b0b0,
cbErrorMsgMax=1024, pcbErrorMsg=&3902aefc )
[2008-09-24 15:49:20.587190] SQLError( pszSqlState="42501", pfNativeError=-551,
pszErrorMsg="{DB2 FOR OS/390}{ODBC DRIVER}{DSN09015}
 DSNT408I SQLCODE = -551, ERROR:  WMQI83 DOES NOT HAVE THE PRIVILEGE TO PERFORM
          OPERATION SELECT ON OBJECT WMQI77.TESTTABLE
 DSNT418I SQLSTATE   = 42501 SQLSTATE RETURN CODE
 DSNT415I SQLERRP    = DSNXOSC SQL PROCEDURE DETECTING ERROR
 DSNT416I SQLERRD    = -100  0  0  -1  0  0 SQL DIAGNOSTIC INFORMATION
 DSNT416I SQLERRD    = X'FFFFFF9C'  X'00000000'  X'00000000'  X'FFFFFFFF'
          X'00000000'  X'00000000' SQL DIAGNOSTIC INFORMATION
  ERRLOC=1:13:2", pcbErrorMsg=623 )
[2008-09-24 15:49:20.587666]     ---> SQL_SUCCESS

[2008-09-24 15:49:20.587725] SQLError( hEnv=0, hDbc=0, hStmt=1,
```

```
pszSqlState=&3902af28, pfNativeError=&3902af24, pszErrorMsg=&1b88b0b0,
cbErrorMsgMax=1024, pcbErrorMsg=&3902aefc )
[2008-09-24 15:49:20.587752] SQLError( )
[2008-09-24 15:49:20.587757]     ---> SQL_NO_DATA_FOUND

[2008-09-24 15:49:20.588049] SQLFreeStmt( hStmt=1, fOption=SQL_DROP )
[2008-09-24 15:49:20.588075] SQLFreeStmt( )
[2008-09-24 15:49:20.588080]     ---> SQL_SUCCESS

[2008-09-24 15:49:20.593800] SQLTransact( hEnv=1, hDbc=0, fType=SQL_COMMIT )
[2008-09-24 15:49:20.593887] SQLTransact( )
[2008-09-24 15:49:20.593893]     ---> SQL_SUCCESS
```

The second trace file shows the same trace file with the operation working:

```
[2008-09-24 16:00:25.287052] SQLAllocStmt( hDbc=1, phStmt=&1c7f8e54 )
[2008-09-24 16:00:25.287068] SQLAllocStmt( phStmt=1 )
[2008-09-24 16:00:25.287075]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.287088] SQLFreeStmt( hStmt=1, fOption=SQL_CLOSE )
[2008-09-24 16:00:25.287098] SQLFreeStmt( )
[2008-09-24 16:00:25.287104]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.287114] SQLPrepare( hStmt=1 )
[2008-09-24 16:00:25.287121] ( pszSqlStr="SELECT TESTTABLE.ID FROM
WMQI77.TESTTABLE TESTTABLE", cbSqlStr=-3 )
[2008-09-24 16:00:25.302484] SQLPrepare( )
[2008-09-24 16:00:25.302510]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.302539] SQLFreeStmt( hStmt=1,
fOption=SQL_CLOSE )
[2008-09-24 16:00:25.302555] SQLFreeStmt( )
[2008-09-24 16:00:25.302560]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.302573] SQLExecute( hStmt=1 )
[2008-09-24 16:00:25.302622] SQLExecute( )
[2008-09-24 16:00:25.302628]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.302660] SQLNumResultCols( hStmt=1,
pcCol=&3902c7fa )
[2008-09-24 16:00:25.302672] SQLNumResultCols( pcCol=1 )
[2008-09-24 16:00:25.302679]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.302697] SQLDescribeCol( hStmt=1, iCol=1,
pszColName=&3902cb10, cbColNameMax=200, pcbColName=&3902c804,
pfSQLType=&3902c802, pcbColDef=&3902c858, pibScale=&3902c800,
pfNullable=&3902c7fe )
[2008-09-24 16:00:25.302733] SQLDescribeCol( pszColName="ID",
pcbColName=2, pfSQLType=SQL_CHAR, pcbColDef=10, pibScale=0,
pfNullable=SQL_NULLABLE )
[2008-09-24 16:00:25.302819]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.302826] SQLColAttribute( hStmt=1, iCol=1,
fDescType=SQL_DESC_OCTET_LENGTH, rgbDesc=NULL, cbDescMax=0,
pcbDesc=NULL, pfDesc=&3902c864 )
[2008-09-24 16:00:25.302850] SQLColAttribute( pfDesc=10 )
[2008-09-24 16:00:25.302857]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.302866] SQLBindCol( hStmt=1, iCol=1,
fCType=SQL_C_CHAR, rgbValue=&1b48829c, cbValueMax=12,
pcbValue=&1b488298 )
[2008-09-24 16:00:25.302888] SQLBindCol( )
[2008-09-24 16:00:25.302894]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.302901] SQLSetStmtAttr( hStmt=1,
fAttribute=SQL_ATTR_ROW_BIND_TYPE, pvParam=&10, iStrLen=0 )
[2008-09-24 16:00:25.302917] SQLSetStmtAttr( )
```

```
[2008-09-24 16:00:25.302922]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.302928] SQLSetStmtAttr( hStmt=1,
fAttribute=Unknown value 9, pvParam=&20, iStrLen=0 )
[2008-09-24 16:00:25.302943] SQLSetStmtAttr( )
[2008-09-24 16:00:25.302949]     ---> SQL_SUCCESS

[2008-09-24 16:00:25.302956] SQLExtendedFetch( hStmt=1,
fFetchType=SQL_FETCH_NEXT,iRow=0, pcRow=&1c7f6894,
rgfRowStatus=&1bca17d0 )
[2008-09-24 16:00:25.317947] ( Row=1, iCol=1, fCType=SQL_C_CHAR,
rgbValue="TABLG     ", pcbValue=10 )
[2008-09-24 16:00:25.317980] ( Row=2, iCol=1, fCType=SQL_C_CHAR,
rgbValue="TABLF     ", pcbValue=10 )
[2008-09-24 16:00:25.318001] ( Row=3, iCol=1, fCType=SQL_C_CHAR, r
gbValue="TABLE     ", pcbValue=10 )
[2008-09-24 16:00:25.318022] ( Row=4, iCol=1, fCType=SQL_C_CHAR,
rgbValue="TABLD     ", pcbValue=10 )
[2008-09-24 16:00:25.318044] ( Row=5, iCol=1, fCType=SQL_C_CHAR,
rgbValue="TABLC     ", pcbValue=10 )
[2008-09-24 16:00:25.318065] ( Row=6, iCol=1, fCType=SQL_C_CHAR,
rgbValue="TABLB     ", pcbValue=10 )
[2008-09-24 16:00:25.318087] ( Row=7, iCol=1, fCType=SQL_C_CHAR,
rgbValue="TABLA     ", pcbValue=10 )
[2008-09-24 16:00:25.318109] SQLExtendedFetch( pcRow=7 )
```

WebSphere MQ Java Client trace

You can activate WebSphere MQ Java Client trace while putting a message on a WebSphere MQ queue using the Enqueue editor.

1. Click **Window** → **Preferences**.
2. On the left, click **Enqueue**. The Enqueue preferences screen opens.
3. To activate WebSphere MQ Java Client tracing, click the **To file** radio button. The default output file is specified. You can specify a different output file by typing in the filepath or using the **Browse** button.

   If the output file that you specify is invalid (if it is read-only or already in use), error message BIP0832 is displayed:

   ```
   A class java.io.FileNotFoundException exception occurred which reported the
   following message: [filepath] (The process cannot access the file because it is being
   used by another process). Resolve the reason of error and try again.
   ```

   If you do not resolve this error, this BIP0832 error message will be displayed next time you start up if the output file is still invalid.

You can also enable and disable WebSphere MQ Java Client service trace from the **File** menu of the Configuration Manager Proxy API Exerciser.

If trace is activated, any trace messages that are written using the Enqueue editor are written to the specified output file.

Configuration Manager Proxy trace

You can enable and disable service trace for the Configuration Manager Proxy (CMP).

Enabling Configuration Manager Proxy trace

To enable tracing for the Configuration Manager Proxy for your application, use the following command in your code:

```
// Enable Config Manager Proxy service trace
   ConfigManagerProxy.enableTracing("outputfile.txt");
```

This logs all calls to the Configuration Manager Proxy to the outputfile.txt file in the current directory. All Configuration Manager Proxy activity in the entire Java Virtual Machine is logged.

You can also enable Configuration Manager Proxy service trace from the **File** menu of the Configuration Manager Proxy API Exerciser.

In addition, because the Configuration Manager Proxy uses the MQ Java client, you can enable MQ Java client tracing.

Disabling Configuration Manager Proxy trace

To disable tracing for the Configuration Manager Proxy for your application, use the following command in your code:

```
// Disable Config Manager Proxy service trace
   ConfigManagerProxy.disableTracing();
```

You can also disable Configuration Manager Proxy service trace from the **File** menu of the Configuration Manager Proxy API Exerciser.

Using dumps and abend files

A dump or an abend file might be produced when a problem occurs. Dumps and abend files can be used by IBM to resolve the problem.

This section contains the following topics:

- "Checking for dumps"
- "Using the DUMP command on z/OS" on page 105
- "Checking for abend files" on page 107

You can also use the "IBM Support Assistant (ISA)" on page 108 to help with data collection.

Checking for dumps

If a dump occurs on your system, an error message is produced.

- **On Windows**

  BIP2111 error message (message broker internal error). The error message contains the path to the MiniDump file in your errors directory.

- **On UNIX**

  BIP2060 error message (execution group terminated unexpectedly). Look in the directory where the broker was started, or in the service user ID's home directory, to find the core dump file.

- **On z/OS**

  – BIP2060 error message (execution group ended unexpectedly) from the main Broker Address Space. This message should be accompanied by one of the following messages and dump.

  – IEF450I message in the syslog, or component's joblog, showing an abend X'2C1' followed by a reason code, for example:

  ```
  IEF450I MQ83BRK DEFAULT - ABEND=S2C1 U0000 REASON=000000C4
  ```

Look in the system's dump dataset `hlq` for the dump dataset, or search the syslog for the appropriate IEA611I message to find out the dump dataset name.

– IEA993I message in the syslog for a SYSMDUMP. Look in the started task user's directory for the coredump.pid file, as specified in the syslog:

```
IEA993I SYSMDUMP TAKEN TO coredump.00500319
```

– An error message for an SVC dump; see "Dumps on WebSphere Event Broker for z/OS" on page 128 for further information on SVC dumps.

Using the DUMP command on z/OS

z/OS You might be asked to dump any or several of the following address spaces for IBM to resolve the problem:

- Control address space
- DataFlowEngine address space
- UserNameServer address space
- OMVS kernel address space

The following procedure demonstrates how to dump the DataFlowEngine address space. This procedure is the same for any of the address spaces.

1. Find the address space ID of the address space that you want to dump using the display command on the z/OS syslog:

   ```
   D OMVS,U=your started task user ID
   ```

   This command displays the address spaces of all the processes that are running from your started task user ID, for example:

   ```
   D OMVS,U=MQ01BRK
   BPXO040I 16.14.30 DISPLAY OMVS 237
   OMVS     000D ACTIVE          OMVS=(14)
   USER     JOBNAME  ASID        PID       PPID STATE   START     CT_SECS

   MQ01BRK  MQ01BRK  009D   67306064   84083282 HRI--- 15.41.55     48.37
     LATCHWAITPID=         0 CMD=bipservice MQ01BRK AUTO
   MQ01BRK  MQ01BRK  009D   84083282          1 1WI--- 15.41.55     48.37
     LATCHWAITPID=         0 CMD=/argoinst/S000_L30307_P/usr/lpp/mqsi/bin
   MQ01BRK  MQ01BRK  009D   16974444   67306064 HRI--- 15.42.01     48.37
     LATCHWAITPID=         0 CMD=bipbroker MQ01BRK
   MQ01BRK  MQ01BRK  009F   16974445          1 1W---- 15.42.05   2914.22
     LATCHWAITPID=         0 CMD=/argoinst/S000_L30307_P/usr/lpp/mqsi/bin
   MQ01BRK  MQ01BRK  009F   33751662   16974445 HR---- 15.42.05   2914.22
     LATCHWAITPID=         0 CMD=DataFlowEngine MQ01BRK ca614eec-f300-000
   ```

   The infrastructure main program bipimain is the first process in every address space. For a control address space, bipimain starts the bipservice process, which starts the bipbroker process, which might also start the biphttplistener process, depending on the configuration. For a DataFlowEngine address space, bipimain starts the DataFlowEngine process. For a UserNameServer address space, bipimain starts the bipuns process. For each execution group, an additional DataFlowEngine address space is started. In this example, only one execution group is deployed.

2. Use the z/OS DUMP command to dump the DataFlowEngine address space, which is shown in the above example as `9F`.

   a. Enter the following command:

      ```
      DUMP TITLE=(DFE)
      ```

The console returns:

```
*`15 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
```

b. Enter:

```
R 15,ASID=9F,CONT
```

The console returns:

```
*16 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
```

c. Enter:

```
R 16,SDATA=(CSA,RGN,PSA,SQA,LSQA,LPA,TRT,GRSQ,SUM),END
```

The console returns:

```
IEE600I REPLY TO 16 IS;SDATA=(CSA,RGN,PSA,SQA,LSQA,LPA,TRT,GRSQ,SUM),END
IEA794I SVC DUMP HAS CAPTURED: 356
DUMPID=014 REQUESTED BY JOB (*MASTER*)
DUMP TITLE=DFE
IEF196I IGD101I SMS ALLOCATED TO DDNAME (SYS00018)
IEF196I         DSN (SYS3.DUMP.ARG5.#MASTER#.T142958.S00014      )
IEF196I         STORCLAS (SMS) MGMTCLAS (DUMP) DATACLAS (        )
IEF196I         VOL SER NOS= ARGSMR
```

The dump is stored in either a pre-allocated dump data set called SYS1.DUMPxx, or an automatically allocated dump data set named according to an installation-specified pattern.

3. In some scenarios, all address spaces for a given broker, that is, all those listed in the example above, plus the OMVS address space and dataspaces, are needed by IBM to resolve the problem. Use the z/OS DUMP command to dump all MQ01BRK address spaces.

a. Enter the command:

```
DUMP TITLE=(ALL)
```

The console returns:

```
`15 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
```

b. Enter:

```
R 15,JOBNAME=(OMVS,MQ01BRK),DSPNAME=('OMVS'.*),SDATA=(PSA,SQA,LSQA,RGN,TRT,
  LPA,CSA,GRSQ,SUM,NUC)
```

The console returns:

```
IEE600I REPLY TO 15 IS;JOBNAME=(OMVS,MQ01BRK),DSPNAME=('OMVS'.*),S
IEA794I SVC DUMP HAS CAPTURED: 303
DUMPID=040 REQUESTED BY JOB (*MASTER*)
DUMP TITLE=ALL
IEE853I 13.40.40 SYS1.DUMP TITLES 306
SYS1.DUMP DATA SETS AVAILABLE=000 AND FULL=000
CAPTURED DUMPS=0001, SPACE USED=00000447M, SPACE FREE=00001053M
    DUMP.MVK4.#MASTER#.D030415.T134007.S00039 DATA UNAVAILABLE WHILE
        BEING DUMPED TO
IEA611I COMPLETE DUMP ON DUMP.MVK4.#MASTER#.D030415.T134007.S00039 309
DUMPID=040 REQUESTED BY JOB (*MASTER*)
FOR ASIDS(000D,009D,009F)
```

You can also find information on the individual thread by issuing the DISPLAY z/OS console command, as in the example:

```
D OMVS,PID=83886535
```

Checking for abend files

Abend files are produced when a process ends abnormally. The information contained in an abend file helps the IBM Support Center to diagnose and fix the problem.

The following list contains examples of what might cause the broker to produce an abend file:

- The broker runs out of memory.
- A user-defined extension causes an instruction in the broker process that is not valid.
- An unrecoverable error occurs in the broker.

Abend files are never produced during normal operation. If an abend file is produced, contact the IBM Support Center for assistance.

Contacting your IBM Support Center

If you cannot resolve problems that you find when you use WebSphere Event Broker, or if you are directed to do so by an error message generated by WebSphere Event Broker, you can request assistance from your IBM Support Center.

Before you contact your Support Center, use the checklist below to gather important information. Some items might not be relevant in every situation, but you should provide as much information as possible to enable the IBM Support Center to re-create your problem. You can also use the "IBM Support Assistant (ISA)" on page 108 to help with data collection.

For WebSphere Event Broker:

- The product version.
- Any fix packs applied.
- Any interim fixes applied.
- All current trace and error logs, including relevant Windows Event log or LinuxUNIX operating system syslog entries, and any abend or dump files from the *install_dir*\errors directory on Windows, or the /var/mqsi/errors directory on LinuxUNIX. Obtain user trace log files at debug level for all relevant message flows and preferably format them. Also include any requested service trace files.

  To send files from distributed systems, create a zipped file using any zip utility.

  To send a file from the file system to IBM, use **tar** to compress the file. For example `tar -cx -f coredump.0002009E coredump.toibm`. To send MVS data sets to IBM, terse them using `TRSMAIN`, which you can download from z/OS tools download.
- A list of the components installed. Include details of the number of computers and their operating systems, the number of brokers and the computers on which they are running, and the existence and details of any User Name Servers.
- The zipped file obtained by exporting your workspace and appropriate message flows. This action is performed from the workbench.
- Details of the operation that you were performing, the results that occurred, and the results that you were expecting.
- A sample of the messages that were being used when the problem arose

- If relevant, the report file from the C or COBOL importer. This file is located in the directory from which the file import was attempted.

For WebSphere MQ:

- The product version.
- Any fix packs applied.
- Any interim fixes applied.
- All current trace and error logs, including relevant Windows Event log or Linux and UNIX operating system syslog entries and First Failure Support Technology™ (FFST™) output files. You can find these files, which have the extension .fdc, in the errors subdirectory within the WebSphere MQ home directory.
- Details of WebSphere MQ client software, if appropriate.

For each database that you are using:

- The product and release level (for example, DB2 7.1).
- Any fix packs applied.
- Any interim fixes applied.
- All current trace and error logs, including relevant Windows Event log or Linux and UNIX operating system syslog entries, for example the db2dialog.log file on DB2. Check the database product documentation for details of where to find these files.
- Definitions of any database tables.
- Any ODBC traces.

Windows **For Windows:**

- The version.
- The Service Pack level.
- The version of the system files msvcrt.dll, msvcp60.dll, msvcirt.dll, and mfc42.dll. You can find these files in the Windows\SYSTEM32 directory. Use the Windows Explorer file properties to display the versions.
- The environment settings.

UNIX **For Linux and UNIX operating systems:**

- The product version. You can find the version installed by using the **uname -a** command.
- Any service level and patches that have been applied.
- The environment settings.

z/OS **For z/OS:**

- The product version
- The list of PTFs that have been applied
- The environment settings
- The joblogs from all address spaces

IBM Support Assistant (ISA)

The IBM Support Assistant (ISA) helps you to resolve questions and problems with IBM software products by providing access to support-related information and troubleshooting tools.

ISA is available at no charge to install on your computer; you then install the relevant product plug-ins. ISA has a built-in user guide, and the ISA download package includes an installation and troubleshooting guide. This topic contains a brief overview of the features of ISA Version 3; you can find more detailed information on the ISA Web page.

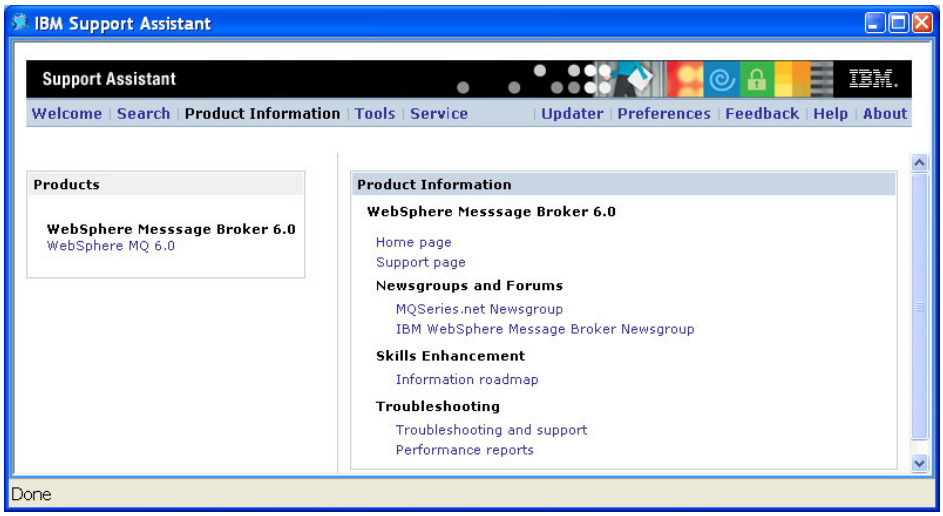
Search

ISA provides a tool that searches multiple sources of product information, and returns the results in the following categories:

- IBM software support documents
- IBM developerWorks
- IBM news groups and forums
- Google Web search
- Product information center

Product Information

The product information component directs you to the appropriate IBM Web sites for WebSphere Event Broker or another product that you have selected, as shown in the following example (the WebSphere Message Broker plug-in provides support information for WebSphere Event Broker):



Tools

If diagnostic tools are available for WebSphere Event Broker, or any other product plug-in that you have installed on ISA, you can access them through the Tools component. More tools might be made available, so check for updates periodically.

Service

Use the Service component to perform the following tasks:

- Collect diagnostic data automatically
- Send files to IBM Support for problem determination
- Create and submit a new problem report
- View or update an existing problem report

Two data collectors are supplied for the WebSphere Message Broker plug-in:

- A general runtime data collector (to run for run time problems)
- A general toolkit data collector (to run for toolkit problems)

Run the appropriate data collector before you open a PMR, so that you can send the collected data to IBM with the PMR.

You can use the problem submission tool to enter your entitlement information once, and save it for future sessions. You can then create a problem report and attach the collector file at the same time.

Updater

Use the Updater component to perform the following tasks:

- Install and uninstall product plug-ins and tools
- Upgrade the ISA when an updated version is available

As well as WebSphere, product plug-ins are available for other products, such as DB2, Lotus, Rational, and Tivoli. You can also choose from a number of diagnostic tools that are common to all products.

Installing the IBM Support Assistant (ISA)

Before you start:

Read the concept topic about the "IBM Support Assistant (ISA)" on page 108.

Follow these steps to install the IBM Support Assistant (ISA) on your computer:

1. Go to the ISA downloads Web page to download the installation package.
2. Log in using your IBM ID and password. If you do not have an IBM ID, click **register now** to create one.
3. Select the version of the ISA that you want and click **continue**.
4. Click **View license** to read the license agreement in a separate window, then select **I agree** and click **I confirm**.
5. Click **Download now** for the relevant operating system and save the compressed file to a temporary directory.
6. Extract the files from the compressed file to a temporary directory. The files that you extract include an installation and troubleshooting guide and an executable file.
7. Follow the instructions in the guide to install ISA.

When you have installed ISA successfully, you can use the desktop icon to open it, or you can click **Programs → IBM Support Assistant → IBM Support Assistant**.

Now that you have installed ISA, install the WebSphere Event Broker plug-in, as described in "Updating the IBM Support Assistant."

Updating the IBM Support Assistant

Before you start:

1. Read the concept topic about the "IBM Support Assistant (ISA)" on page 108.
2. Install the IBM Support Assistant.

You can update the ISA in two ways: installing product plug-ins, and upgrading the ISA itself. Use the Updater component to perform both tasks:

1. Open the IBM Support Assistant by clicking **Programs → IBM Support Assistant → IBM Support Assistant**.
2. Click **Updater**.
   - To install a product plug-in, click **New Products and Tools**.
   - To install an updated version of ISA, click **Upgrades**.
3. Select the item or items to install. Plug-in features are categorized by product family, so expand **WebSphere** and select WebSphere Event Broker.
4. Click **Install**.
5. Click the feature to read its licence agreement, then accept the license agreement.
6. Click **OK** to restart ISA.

When you have installed the product plug-in successfully, click **Product Information**, **Search**, **Tools**, or **Service** to find out what resources are available for WebSphere Event Broker.

Recovering after failure

You can follow a set of procedures to recover after a serious problem.

The types of recovery that are described in this section are really a last resort. You must try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center.

This section contains the following topics:

- "Recovering after the Configuration Manager fails"
- "Recovering after the broker fails" on page 112
- "Recovering after the User Name Server fails" on page 113
- "Recovering after an execution group fails" on page 113
- "Recovering after the broker's queue manager fails" on page 114
- "Recovering after the Configuration Manager's queue manager fails" on page 114
- "Recovering after the User Name Server's queue manager fails" on page 116
- "Recovering after a shared queue manager fails" on page 116
- "Recovering after the configuration repository fails" on page 117
- "Recovering after the broker database fails" on page 117

Recovering after the Configuration Manager fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center. Use the procedure in this section only as a last resort.

If the Configuration Manager environment is damaged, or the configuration repository is corrupted, and the problem cannot be corrected using problem

determination, or by the IBM Support Center, perform the following sequence of operations to re-create the Configuration Manager.

The following sequence of operations applies to Windows, Linux, and UNIX systems. For z/OS, use the JCL equivalents of the commands.

1. Ensure that all workbench sessions are stopped.
2. Stop the Configuration Manager using the **mqsistop** command.
3. Stop the Configuration Manager's queue manager using the **endmqm** command.
4. Delete the Configuration Manager using the **mqsideleteconfigmgr** command:
   a. If you are recovering the Configuration Manager because the configuration repository is damaged, delete it when you delete the Configuration Manager. This destroys all information pertinent to the broker domain (not just data internal to the Configuration Manager). You must export everything you can from the domain before you issue the **mqsideleteconfigmgr** command specifying the **-n** parameter.
   b. If the configuration repository is undamaged, *omit* the **-n** parameter. This preserves your configuration data in the repository.

   The **mqsideleteconfigmgr** command:
   - Stops the service that runs the Configuration Manager.
   - Stops and deletes the WebSphere MQ queue manager for the Configuration Manager, if requested.
   - Removes the tables from the configuration repository, if requested.
   - Removes the record for the component in the broker registry.
5. If you are recovering the Configuration Manager because the configuration repository is damaged, restore the repository from a previously successful backup version using the **mqsirestoreconfigmgr** command.
6. Recreate the Configuration Manager using the **mqsicreateconfigmgr** command.
7. Start the Configuration Manager using the **mqsistart** command.
8. Restart the workbench if it is not currently running, and switch to the Broker Administration perspective.
9. If you have completed step 4a and step 5, you must also redeploy the domain configuration to ensure that the configuration across the broker domain is consistent.

Recovering after the broker fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center. Use the procedure in this section only as a last resort.

If a broker's database becomes corrupted, follow the instructions described in "Recovering after the broker database fails" on page 117. If the database is not corrupted, but the problem cannot be corrected using problem determination, perform the following sequence of operations to re-create the broker:

1. Ensure that no workbench users are deploying to brokers. You must wait until these actions have completed.
2. Stop the broker using the **mqsistop** command.
3. Stop the broker's queue manager using the **endmqm** command.

4. Take a backup of the broker database tables. These tables are interdependent and must all be in a consistent state when restored. You cannot backup or restore individual tables.
5. Delete the broker using the **mqsideletebroker** command.
6. Recreate the broker using the **mqsicreatebroker** command.
7. Restore the broker database tables, either from the backup you have just taken, or from a previously successful backup version.
8. Start the broker using the **mqsistart** command.
9. Redeploy the domain configuration to ensure that the configuration across the broker domain is consistent.

Recovering after the User Name Server fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center. Use the procedure in this section only as a last resort.

If the User Name Server environment becomes damaged, and the problem cannot be corrected using problem determination, or by the IBM Support Center, perform the following sequence of operations to re-create the User Name Server:

1. Stop the User Name Server using the **mqsistop** command.
2. Stop the User Name Server's queue manager using the **endmqm** command.
3. Delete the User Name Server using the **mqsideleteusernameserver** command.
4. Re-create the User Name Server using the **mqsicreateusernameserver** command.
5. Start the User Name Server using the **mqsistart** command.

Recovering after an execution group fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center. Use the procedure in this section only as a last resort.

If a single execution group fails, and the problem cannot be corrected using problem determination, or by the IBM Support Center, perform the following sequence of operations to re-create the execution group:

1. Ensure that no workbench users are deploying to the broker to which the execution group belongs. You must wait until these actions have completed.
2. Stop the broker using the **mqsistop** command.
3. Delete the execution group.
4. Start the broker using the **mqsistart** command.
5. Clean the execution group.
6. Redeploy the configuration.

If more than one execution group fails, you might need to re-create the broker. See "Recovering after the broker fails" on page 112 for information on how to do this.

Recovering after the broker's queue manager fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center. Use the procedure in this section only as a last resort.

If the failed queue manager is shared between a broker, User Name Server and Configuration Manager, re-create all three components, because each component creates different queues. For information on how to recover a shared queue manager, see "Recovering after a shared queue manager fails" on page 116.

If the broker's queue manager fails and cannot be corrected using problem determination, or by the IBM Support Center, perform the following sequence of operations to re-create the queue manager:

1. Ensure that no workbench users are deploying to the broker. You must wait until any such actions have completed.
2. Stop the broker using the **mqsistop** command.
3. If there is no damage to any of the broker database tables, take a backup of these tables. These tables are interdependent and must all be in a consistent state when restored. You cannot back up or restore individual tables.
4. Delete the broker using the **mqsideletebroker** command, with the **-q** parameter to remove the queue manager.
5. Re-create the broker using the **mqsicreatebroker** command. The **mqsicreatebroker** command creates the queue manager and default queues automatically.
6. Re-create any specific queues that are needed for your message flows.
7. Restore the broker database tables, either from the backup that you have just taken, or from a previously-successful backup version.
8. Start your brokers using the **mqsistart** command.
9. Redeploy the domain configuration.

Recovering after the Configuration Manager's queue manager fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center.

Follow the link below for your operating system:

- Windows, Linux, and UNIX systems
- z/OS

If you need to transfer the Configuration Manager onto another queue manager, follow the steps described in Moving the Configuration Manager to a new queue manager.

Recovering after the Configuration Manager's queue manager fails on Windows, Linux, and UNIX systems

Use the procedure in this section only as a last resort.

If the failed queue manager was shared between a broker, User Name Server and Configuration Manager, re-create all three components, because each component creates different queues. For information on how to recover a shared queue manager, see "Recovering after a shared queue manager fails" on page 116.

If the Configuration Manager's queue manager fails and cannot be corrected using problem determination, or by the IBM Support Center, perform the following sequence of operations to re-create the queue manager:

1. Ensure that no workbench users are deploying to brokers. You must wait until any such actions have completed.
2. Stop all brokers that are using the Configuration Manager using the **mqsistop** command.
3. Stop the Configuration Manager using the **mqsistop** command.
4. Delete the Configuration Manager using the **mqsideleteconfigmgr** command, with the **-q** parameter to remove the queue manager.
5. Re-create the Configuration Manager using the **mqsicreateconfigmgr** command. The **mqsicreateconfigmgr** command creates the queue manager and default queues automatically.
6. Start the Configuration Manager using the **mqsistart** command.
7. Start your brokers using the **mqsistart** command.
8. Redeploy the domain configuration.

Recovering after the Configuration Manager's queue manager fails on z/OS

Use the procedure in this section only as a last resort.

If the failed queue manager was shared between a broker, User Name Server and Configuration Manager, re-create all three components, because each component creates different queues. For information on how to recover a shared queue manager, see "Recovering after a shared queue manager fails" on page 116.

If the Configuration Manager's queue manager fails and cannot be corrected using problem determination, or by the IBM Support Center, perform the following sequence of operations to re-create the queue manager:

1. Ensure that no workbench users are deploying to brokers. You must wait until any such actions have completed.
2. Stop all brokers that are using the Configuration Manager.
3. Stop the Configuration Manager.
4. Correct any queue manager problems.
5. Re-create the Configuration Manager queues using the BIPCRCM JCL; see the Sample BIPCRCM file. Include the **mqsicreateconfigmgr** command **-2** parameter.
6. Start the Configuration Manager.
7. Start your brokers.
8. Redeploy the domain configuration.

Recovering after the User Name Server's queue manager fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center. Use the procedure in this section only as a last resort.

If the failed queue manager was shared between a broker, User Name Server, and Configuration Manager, re-create all three components, because each component creates different queues. For information on how to recover a shared queue manager, see "Recovering after a shared queue manager fails."

If the User Name Server's queue manager fails and cannot be corrected using problem determination, or by the IBM Support Center, perform the following sequence of operations to re-create the queue manager:

1. Ensure that no workbench users are deploying to brokers. You must wait until any such actions have completed.
2. Stop the User Name Server using the **mqsistop** command.
3. Delete the User Name Server using the **mqsideleteusernameserver** command, with the **-q** parameter to remove the queue manager.
4. Re-create the User Name Server using the **mqsicreateusernameserver** command. The **mqsicreateusernameserver** command creates the queue manager and default queues automatically.
5. Start the User Name Server using the **mqsistart** command.
6. Redeploy the domain configuration.

Recovering after a shared queue manager fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center. Use the procedure in this section only as a last resort.

If a queue manager that is shared between a broker, User Name Server, and Configuration Manager fails and cannot be corrected using problem determination, or by the IBM Support Center, re-create all three components, because each component creates different queues. Perform the following sequence of operations:

1. Ensure that no workbench users are deploying to brokers. You must wait until any such actions have completed.
2. Stop each component using the **mqsistop** command.
3. Delete the User Name Server using the **mqsideleteusernameserver** command.
4. Delete the broker using the **mqsideletebroker** command.
5. Delete the Configuration Manager using the **mqsideleteconfigmgr** command, with the **-q** parameter to remove the queue manager.

   You must delete the queue manager last. It cannot be deleted while other components are still sharing it.
6. Re-create each component using the relevant mqsi command.
7. Start each component using the **mqsistart** command.
8. Redeploy the domain configuration.

If you need to transfer the Configuration Manager onto another queue manager, follow the steps described in Moving the Configuration Manager to a new queue manager.

Recovering after the configuration repository fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center. Use the procedure in this section only as a last resort.

If the configuration repository fails and cannot be corrected using problem determination, or by the IBM Support Center, perform the following sequence of operations to re-create the configuration repository:

1. Ensure that no workbench users are deploying to brokers. You must wait until any such actions have completed.
2. Stop all brokers that are using the Configuration Manager using the **mqsistop** command.
3. Stop the Configuration Manager using the **mqsistop** command.
4. Re-create the configuration repository from a backup version using the **mqsirestoreconfigmgr** command on the same machine as the Configuration Manager.

   If you do not have a backup of your configuration repository, re-create the Configuration Manager at this stage. This re-creates the configuration repository. However, any of the data that you previously had in the repository will be lost. To re-create the Configuration Manager:

   a. Delete the Configuration Manager using the **mqsideleteconfigmgr** command, specifying the **-n** parameter.
   b. Re-create the Configuration Manager using the **mqsicreateconfigmgr** command. The **mqsicreateconfigmgr** command creates the configuration repository automatically.
5. Start the Configuration Manager using the **mqsistart** command.
6. Start your brokers using the **mqsistart** command.

Recovering after the broker database fails

Try to get to the root of the problem first, using the diagnosis techniques described throughout the "Troubleshooting and support" on page 3 section of the information center. If your problem cannot be resolved using these techniques, contact your IBM Support Center. Use the procedure in this section only as a last resort.

If the broker database fails and cannot be corrected using problem determination, or by the IBM Support Center, perform the following sequence of operations to re-create the broker:

1. Ensure that no workbench users are deploying to the broker. You must wait until any such actions have completed.
2. Stop the broker using the **mqsistop** command.
3. If possible, re-create the broker database from a backup version. If this is not possible, perform the following actions in the order shown:

   a. Delete the broker using the **mqsideletebroker** command.

b. Remove completely all references to the broker from the Configuration Manager repository. You can do this by removing the broker from the topology using the Message Brokers Toolkit. Ensure that the broker's SYSTEM.BROKER.ADMIN.QUEUE queue has been deleted and initiate a topology deploy.

   Alternatively, you can use the Configuration Manager Proxy API "TopologyProxy.removeDeletedBroker(String)" to programmatically purge all references to the broker:

   1) Start the Configuration Manager Proxy API Exerciser sample (which is available on the **Start** menu on Windows).
   2) Connect to the Configuration Manager.
   3) Right-click the topology object, then click **Remove references to a previously deleted broker**.

c. Re-create the broker using the **mqsicreatebroker** command.

d. Add the broker to the Configuration Manager's domain topology.

4. Start the broker using the **mqsistart** command.
5. If you were unable to restore the broker's database from a backup, redeploy the broker's configuration.

Part 2. Reference

Troubleshooting 121
Logs 121
Local error logs. 122
WebSphere Event Broker logs 122
z/OS broker joblogs 123
WebSphere MQ logs 124
Database logs 125
Trace 125
User trace 126
Service trace. 127
Dumps 128
Dumps on Windows 128
Dumps on UNIX systems 128
Dumps on WebSphere Event Broker for z/OS 128
WebSphere Event Broker for z/OS abends. . . 130
Abend files 130
Abend in a user-defined extension 131
WebSphere Event Broker event reports 133
WebSphere Event Broker event reports: general architecture 133
WebSphere Event Broker event reports: configuration changes 134
WebSphere Event Broker event reports: operational information 139
WebSphere Event Broker event reports: operational warnings 142
WebSphere Event Broker event reports: notification message schema 143
WebSphere MQ facilities 146
WebSphere MQ logs 146
FFST files. 146
WebSphere MQ events 146
Database facilities 146
Database logs 147
ODBC trace 147
Other sources of diagnostic information on z/OS 147
Files from a broker, Configuration Manager, or User Name Server 147
Files in the home directory 147
Problems accessing files in the file system . . . 148
Solutions to similar problems 148
Useful Web sites 148
RETAIN 148

© Copyright IBM Corp. 2000, 2009

Troubleshooting

Use the reference information in this section to help you diagnose errors in WebSphere Event Broker.

Select the appropriate topic from the following list to diagnose problems and errors in WebSphere Event Broker:

- "Logs"
- "Trace" on page 125
- "Dumps" on page 128
- "Abend files" on page 130
- "WebSphere Event Broker event reports" on page 133
- "WebSphere MQ facilities" on page 146
- "Database facilities" on page 146
- "Other sources of diagnostic information on z/OS" on page 147
- "Solutions to similar problems" on page 148
- Diagnostic messages

Logs

If an error is reported by a WebSphere Event Broker component, start your investigations into its causes by looking at the product and systems logs to which information is written during component operation.

The information that is recorded in a log typically consists of a time stamp, which indicates when the error occurred, and a brief description of the error, which can be expanded to provide more details of the error. In some situations, a general error is written and is followed by a group of more specific errors that provide details about the general error.

When an error occurs, check the local error log first. These logs record information about major activities within the system. All components of WebSphere Event Broker provide diagnostic information whenever error or warning conditions affect broker operation. These conditions include:

- Unsuccessful attempts to write a message to a WebSphere MQ output queue
- Errors interacting with databases
- The inability to parse an input message

Additional logs that are specific to WebSphere Event Broker are written to record runtime errors, internal errors that are produced by the operating system or your code, or errors related to the work that you are doing in a particular perspective, all of which you can view using the workbench.

Data is written to the following logs:

- "Local error logs" on page 122
- "WebSphere Event Broker logs" on page 122
- "z/OS broker joblogs" on page 123
- "WebSphere MQ logs" on page 124

© Copyright IBM Corp. 2000, 2009

- "Database logs" on page 125

Local error logs

WebSphere Event Broker components use the local error log (also known as the system log) to record information about major activities within the system. When an error occurs, check the local error log first.

Windows On Windows, the local error log is the Windows Event log (Application view).

UNIX On UNIX and Linux systems, the local error log is the *syslog*. Where syslog messages are sent depends on how you configure your UNIX or Linux system.

z/OS On z/OS, the local error log is the operator console.

Entries in the local error log that are generated by WebSphere Event Broker are identified as follows:

- Windows On Windows, by events from the source `WebSphere Broker v****` and a message number of the form `nnnn`, where `****` is the current four-digit product version number.
- UNIX On UNIX, by a message number of the form `BIPnnnn`.
- z/OS On z/OS, by a started task ID, and a message number of the form `BIPnnnn`.

When a broker encounters an error, more than one message might be written to the local error log. Typically these start with a general message (for example, `Could not process a message`), and further messages give more details about the cause of the error (for example, `An error was detected while processing the following SQL statement.`).

Windows On Windows the event log can fill up; ensure that you have a log size that is sufficient, or that you have enabled circular logging.

WebSphere Event Broker logs

WebSphere Event Broker writes information to a number of product-specific logs to report the results of actions that you take.

The following logs are used for reporting errors and events:

- "Eclipse error log"
- "Broker domain Event Log" on page 123
- "The Problems or Alerts view" on page 123

Eclipse error log

The Eclipse error log captures internal errors from Eclipse and the code associated with your user-defined extensions. Check this log at run time for exceptions that you might have caused during development. When errors are triggered, they are added to the Error Log view of the Plug-in Development perspective, and you can double-click each one to examine details. Entries are sorted in reverse chronological

order; that is, most recent first. You can open the Properties view to see entry details. You can also clear the view when you have fixed all the errors, although you cannot clear individual errors.

Some errors might be hierarchical; that is, a general entry for a complex problem might be followed by a number of child entries that list the individual problems in detail.

The same information that is shown in the Error Log view is stored in the .log file in the workspace directory. The default location for the workspace directory depends on your environment:

- Linux *user_home_dir*/IBM/wmbt60/workspace/
- Windows C:\Documents and Settings*user_ID*\IBM\wmbt6.0\workspace\

The Eclipse error log shows errors that are generated only by the user ID that is working with that particular workbench session.

Broker domain Event Log

Broker domain log information is written to the broker domain Event Log. This log is stored and managed by the Configuration Manager. The Event Log displays messages about events that occur within the broker domain, such as deploying topology or topics hierarchy configurations. The messages can be information, errors, or warnings. You view the messages in the editor area of the Broker Administration perspective.

The broker domain Event Log shows errors that are generated by all workbench users in that particular domain.

The Problems or Alerts view

Either the Problems or the Alerts view is displayed, depending on the perspective being used at the time.

- In the Broker Application Development perspective, the Problems view shows information messages, warnings, and errors for message flow applications and associated resources, such as message flows.
- In the Broker Administration perspective, the Alerts view shows information about stopped resources (such as message flows, execution groups, and brokers).

z/OS broker joblogs

Understanding the broker address spaces

On z/OS, the broker writes messages to the appropriate z/OS system log and joblogs. These might include information, warning, error, and severe messages to indicate various situations and events.

The broker run time consists of multiple address spaces. There is a single control address space that is always running when the broker is active and is responsible for communicating with the Configuration Manager and workbench. Each execution group within the broker is mapped to its own address space and as these execution groups start and stop, the corresponding address spaces are started and stopped. The control address space is assigned a JOBNAME and STEPNAME, which is the same as the broker name. The execution groups have a JOBNAME that is also the same as the broker name, and a STEPNAME that matches the last

seven characters of the execution group name.

Viewing the z/OS system console log

The broker writes all of its messages to the z/OS system console log. You can see messages from all address spaces running on the z/OS system in this log. It is easy to identify jobs associated with the broker in the console log because of the naming of broker address spaces. Using the console log, you can see the order of event reporting for different products. This is helpful for cross-product problem determination.

Viewing the broker joblogs

The broker control address space and each of the execution group address spaces has its own joblog. Selecting the joblog for the appropriate broker address space allows you to see all messages relating to that address space. This is helpful in a busy system where the system console log might have many messages from different products obscuring the information in which you are interested.

Interpreting log information

In both logs, each message is displayed on its own, in date and time order; it might span multiple lines, if necessary. For each message, the following information is available:

- **Date and time**: The exact date and time when the event was written to the log.
- **JOBID**: The started task job identifier of the address space.
- **Message Number**: The message number that identifies whether the event is information, a warning, or an error, with diagnostic text.
- **JOBNAME**: The JOBNAME of the address space issuing the message. This is always the same as the broker name.
- **STEPNAME**: The STEPNAME of the address space that is issuing the message. For the control address space, this is the same as the broker name; otherwise, it is the same as the last seven characters of the execution group that is issuing the message.

WebSphere MQ logs

WebSphere MQ messages are written to the local error log in the same way as WebSphere Event Broker messages. WebSphere MQ messages start with the prefix AMQ followed by a number between 4000 and 9999. The number specifies from which part of WebSphere MQ the message originates:

4000 to 4999
Windows user interface messages

5000 to 5999
Installable services messages

6000 to 6999
Common services messages

7000 to 7999
WebSphere MQ product messages

8000 to 8999
WebSphere MQ administration messages

9000 to 9999
Remote messages

For a full explanation of each message, see the WebSphere MQ library Web page.

Database logs

Databases write severe error and warning conditions to the local error log (syslog). Typically, databases also write errors to a database log file, such as the db2diag.log file on DB2.

You can open the db2diag.log file in a text editor.

The example below shows a typical entry from the db2diag.log file:

```
Jun 18 15:02:53 bluj DB2[46827]: DB2(db2inst1.000(1))oper_system_services sqlobeep(2)
 reports:(3)
Jun 18 15:02:53 (4) bluj (5) DB2[46827(6)]: extra symptom string provided:(7) RIDS/sqlesysc_
Jun 18 15:02:53 bluj DB2[46827]:     data: (8)  54686973  20697320  616e2065  78616d70
Jun 18 15:02:53 bluj DB2[46827]:     data:   6c65206f  66206c6f  67676564  20646174
Jun 18 15:02:53 bluj DB2[46827]:     data:   61
Jun 18 15:02:53 bluj DB2[46827]: 2 piece(s) of dump data provided... to file(9) /u/db2inst1/
Jun 18 15:02:53 bluj DB2[46827]:         1. 'DUMP EXAMPLE #1' has been dumped (10)
Jun 18 15:02:53 bluj DB2[46827]:         2. 'DUMP EXAMPLE #2' has been dumped
```

The bold numbers in the example show various items in the log file:

(1) The instance name and node number

(2) The reporting component and function

(3) The probe ID and error and alert numbers

(4) A time stamp for when the event occurred

(5) The host name

(6) The process ID of the reporting process. Use the **ps** command to view information about the process ID of the reporting process. For example, enter the following command to get information about the reporting process:

```
ps -fu 46827
```

(7) A symptom string that contains additional information about where and why the problem occurred

(8) A hexadecimal dump of data that includes return codes and other information that can be interpreted by your IBM Support Center

(9) Information about additional dump files. Larger structures and other binary data might be dumped to additional files. The name of the file is identified in the syslog file

(10) An entry to identify a piece of dump data

Trace

If you cannot get enough information about a particular problem from the entries that are available in the various logs described in “WebSphere Event Broker logs” on page 122 and “Local error logs” on page 122, the next troubleshooting method to consider is using trace. Trace provides more details about what is happening

while code executes. The information produced from trace is sent to a specified trace record, so that you or IBM support personnel can analyze it to discover the cause of your problem.

Trace is inactive by default, and must be explicitly activated by a command, or by the workbench.

There are two main types of trace available in WebSphere Event Broker: user trace and service trace.Typically, you utilize user trace for debugging your applications; you can trace brokers, execution groups, and deployed message flows. With service trace, you can activate more comprehensive broker tracing, and start tracing for the workbench, Configuration Manager, and User Name Server. You can also trace the execution of all the commands described in Commands.

When you start user tracing, you cause additional processing for every activity in the component that you are tracing. Large quantities of data are generated by the components. Expect to see some impact on performance while trace is active. You can limit this additional processing by being selective about what you trace, and by restricting the time during which trace is active.

User trace

User trace is one of two types of optional trace that are available in WebSphere Event Broker. User trace is inactive by default; you must activate it explicitly by using a command, or by selecting options in the workbench, when you need more information than that provided by the entries that are written to the Event Log.

For more information about logs, see “Logs” on page 121.

Typically, you use user trace for debugging your applications, as it can trace brokers, execution groups, and deployed message flows.

You can use the workbench to control most of the trace activity that you need. The workbench also enables you to start and stop tracing on remote systems.

When you activate user trace, you cause additional processing for every activity in the component that you are tracing. Large quantities of data are generated by the components. Expect to see some impact on performance while user trace is active. You can limit this additional processing by being selective about what you trace, and by restricting the time during which trace is active.

The user trace log files

When trace is active for any component, information is recorded in binary form.

The location of the trace logs depends on your environment:

Windows **Windows**

If you set the workpath by using the **-w** parameter of the mqsicreatebroker command, the location is workpath\log.

If you have not specified the broker workpath, the default location is %ALLUSERSPROFILE%\Application Data\IBM\MQSI\common\log where the default setting for the environment variable %ALLUSERSPROFILE% is C:\Documents and Settings\All Users. The actual value might be different on your computer.

Linux UNIX **Linux and UNIX**
/var/mqsi/common/log

z/OS **z/OS**
/component_filesystem/log

The file names reflect the component and subcomponent for which the trace is active. For example, the broker name and unique execution group identifier form part of the file name when you are tracing activity within that execution group.

For example, if you have created a broker called WBRK_BROKER, you might see the following files in the log subdirectory:

```
WBRK_BROKER.682ec116-dc00-0000-0080-ce28a236e03d.userTrace.bin.1
WBRK_BROKER.682ec116-dc00-0000-0080-ce28a236e03d.userTrace.bin.2
```

You cannot view these files directly; use the commands that are provided to access the trace information and convert it to a viewable format. Use the mqsireadlog command to retrieve the trace log for the specified component. Issue the command on the computer on which the log was generated. You can specify the output to be directed to a file, which is independent of operating system, and can be transferred to other systems for browsing or formatting by using the mqsiformatlog command.

Service trace

Service trace is one of two types of optional trace that are available in WebSphere Event Broker. Service trace is inactive by default; you must activate it explicitly by using a command when you need more information than that provided by the entries that are written to the Event Log or to user trace.

With service trace, you can activate more comprehensive broker tracing, and start tracing for the workbench, Configuration Manager, and User Name Server. You can also trace the execution of all the commands described in Commands, including the trace commands themselves. Use the mqsichangetrace command to work with service trace; you cannot use the workbench.

Activate service trace only when you receive an error message that instructs you to, or when directed to do so by your IBM Support Center.

When you activate service trace, you cause additional processing for every activity in the component that you are tracing. Large quantities of data are generated by the components. Expect to see some impact in performance while service trace is active. You can limit this additional processing by being selective about what you trace, and by restricting the time during which trace is active.

The location of the trace logs depends on your environment:

Windows **Windows**
If you have set the workpath using the **-w** parameter of the mqsicreatebroker command, the location is workpath\log.

If you have not specified the broker workpath, the default location is %ALLUSERSPROFILE%\Application Data\IBM\MQSI\common\log where the default setting for the environment variable %ALLUSERSPROFILE% is C:\Documents and Settings\All Users. The actual value might be different on your computer.

Linux UNIX **Linux and UNIX**
/var/mqsi/common/log

z/OS **z/OS**
/component_filesystem/log

The directory to which the service trace logs are written must be able to hold all the logs for that computer. You might want to place it on a separate file system, if allowed by your system operator.

Dumps

Under exceptional circumstances, Windows MiniDumps, UNIX core dumps, or z/OS SVC or core dumps might be produced.

For example, if errors occur in the broker, Configuration Manager, or User Name Server executable files, or in the infrastructure main program. The amount and complexity of data produced in these cases normally requires the assistance of your IBM Support Center.

Dumps on Windows

Windows MiniDumps might be produced by broker processes in extreme cases. They are never produced during normal operation.

They are accompanied by a BIP2111 error message (message broker internal error) that contains the path to the MiniDump file in your workpath/errors directory. MiniDump files have the extension `.dmp`. You can analyze these MiniDumps using a suitable debugger; contact IBM for assistance.

Dumps on UNIX systems

UNIX core dumps are produced by broker processes in extreme cases. They are never produced during normal operation.

A BIP2060 error message (execution group terminated unexpectedly) might be produced. Look in the directory where the broker was started to find the core dump file. If this directory is not writable by the service ID, the core dumps are produced in the service user ID's home directory.

You can analyze these core dumps using a suitable debugger; contact IBM for assistance.

Dumps on WebSphere Event Broker for z/OS

A broker, Configuration Manager, or User Name Server, produces an SVC dump that is written to a dataset named using the system defined naming convention.

The format of this dataset name can be displayed using the dump options command `/D dump`.

Normally, the name definition has the following format:

```
DUMP.&SYSNAME..&JOBNAME..D&DATE..T&LTIME..S&SEQ
```

and, for example, this format displays a resolved name of:

```
DUMP.MVS1.MQ83BRK.D080924.T171755.S00005
```

An SVC dump that is caused by z/OS or WebSphere Event Broker for z/OS is written to the system's dump directory, and can be formatted with IPCS. The name of the dump is listed on the z/OS syslog.

The following example shows the format of the output you receive:

```
IEA794I SVC DUMP HAS CAPTURED:
        DUMPID=006 REQUESTED BY JOB (MQ83BRK )
        DUMP TITLE=MQ83BRK ,ABN=S0C4-00000004,C=M7500.600.BRKR
IEA611I COMPLETE DUMP ON DUMP.MVS1.MQ83BRK.D080924.T171851.S00006
        DUMPID=006 REQUESTED BY JOB (MQ83BRK )
        FOR ASID (00B8)
        INCIDENT TOKEN: PLEXS    MVS1     09/24/2008 16:19:11
        ID = MQ83BRK ,ABN=S0C4-00000004
```

Under some circumstances, SVC dumps are not produced. Generally, dumps are suppressed because of time or space problems, or security violations.

In addition, you can suppress SVC dumps that duplicate previous dumps using z/OS DAE, for example, use the command SET DAE=xx. (DAE is Dump Analysis and Elimination: a z/OS service that enables an installation to suppress SVC dumps and ABEND SYSDUMP dumps that are not needed because they duplicate previously written dumps.)

The following example shows a message on the z/OS syslog, indicating whether duplicates of SYSMDUMPs are suppressed:

```
   IEA995I SYMPTOM DUMP OUTPUT 504
 SYSTEM COMPLETION CODE=0C4  REASON CODE=00000004
  TIME=11.02.24  SEQ=02327  CPU=0000  ASID=0060
  PSW AT TIME OF ERROR  078D2000   8D70A656  ILC 4  INTC 04
    ACTIVE LOAD MODULE           ADDRESS=0D708F68  OFFSET=000016EE
    NAME=SPECIALNAME
         61939683 81936199 85A2F1F0 61A48689 */local/res10/ufi*
         A7F5F161 82899561 82899789 94818995 *x51/bin/bipimain*
    DATA AT PSW  0D70A650 - 91345000  00001F22  41209140
    GPR  0-3  00000000  0D711B18  0D711B18  00000000
    GPR  4-7  0D711300  0D70BD04  00000000  0D711B18
    GPR  8-11 0D7121CF  0D7111D0  0D70C316  0D70B317
    GPR 12-15 8D70A318  0D7111D0  00000312  00000000
  END OF SYMPTOM DUMP
 IEA838I SYSMDUMP SUPPRESSED AS A DUPLICATE OF: 505
         ORIGINAL:DATE 01170 TIME 10:59:40:05 CPU 8A7645349672
         MOD/#PATHNAM CSECT/BIPIP PIDS/5655G9700 AB/S00C4
         REXN/IMBSIREC FI/9134500000001F2241209140 REGS/0C33E
         HRC1/00000004 SUB1/INFRASTRUCTURE#MAIN
```

The *MVS Diagnosis: Tools and Service Aids* manual provides details about using z/OS dump analysis and elimination (DAE).

In extreme cases, you might instead receive a core dump, which will be written to the started task user's directory. The maximum size of a core dump is defined through MAXCORESIZE in the BPXPRMxx parmlib member. The IBM-supplied default is 4 MB. To ensure completeness of a core dump of any WebSphere Event Broker for z/OS address space, change this to 2 GB. The started task user's directory must then have at least this size.

To make use of these core dumps, copy them to a partitioned data set using the TSO/E OGET command. Allocate the data set as a sequential data set with a logical record length (LRECL) of 4160 and a record format of FBS. Specify a

primary allocation of at least 500 cylinders and a secondary allocation of at least 250 cylinders. The data set must be transferred as binary. For example:

```
oget '/u/user_directory/coredump.pid' 'mvs_dataset_name.pid' bin
```

To ensure that all the diagnostic information is collected in these extreme cases, specify the following dump options in SYS1.PARMLIB:

- `Member IEADMP* SDATA=(LSQA,TRT,CB,ENQ,DM,IO,ERR,SUM)`
- `Member IEADMR* SDATA=(NUC,SQA,LSQA,SWA,TRT,RGN,LPA,CSA,SUM,GRSQ)`

WebSphere Event Broker for z/OS abends

Abend code
2C1

Explanation
The WebSphere Event Broker for z/OS infrastructure encountered a severe internal error. The reason code helps the IBM Support Center to identify the source of the error. It has the format `X'ppmmnnnn'`, where `pp` defines the part within the infrastructure where the error occurred, `mm` defines the specific location of the error, and `nnnn` defines a recursion index.

You can resolve the following reason codes but you should refer other codes to the IBM Support Center.

ABN=S2C1-0001xxxx Error opening a file system file
The file system component might be full or the broker user ID might not have the correct permissions to access files or directories in the file system component.

ABN=S2C1-0113xxxx Region size too small

Source
WebSphere Event Broker for z/OS

System Action
The system might issue a dump.

Programmer Response
None

System Programmer Response
Search the problem-reporting databases for a fix for the problem. If no fix exists, contact the IBM Support Center, providing the reason code and, if present, the dump.

Abend files

When a process does not end normally an abend file is generated.

When a process abends, an entry is made in the syslog, or the Windows Event log. If there is more data to be written than is appropriate for the log, a new file is created to contain it, and the log entry tells you the file name. You need to send the file to your IBM Support Center for analysis.

Abend files are never generated during normal operation; always involve your IBM Support Center when they do occur.

The new file is called:

Windows **Windows**

filename.abend; if you have set the workpath using the **-w** parameter of the **mqsicreatebroker** command, the location is workpath\common\errors. If you have not specified the broker workpath, the workpath can be resolved by issuing the command **echo %MQSI_WORKPATH%** from the installation's Command Console. If it is not possible to write to this directory, the file is put in the workpath\common\log directory, but in either case the message in the Event Log tells you where to find it.

UNIX **UNIX**

filename.abend in the /var/mqsi/common/errors directory.

z/OS **z/OS**

filename.abend in the /<component_HFS>/common/errors directory.

where *filename* is a unique dynamically-allocated name, as given in the syslog or Event Log message.

The abend file might provide a stack for the failing thread; this might help you to identify problems in the plug-in code when an abend occurs here. In which case, for z/OS, this stack will be available in a CEEDUMP file in the same directory as the abend file by default.

Each abend file contains a header that includes the following sections:

- Product details (the broker product that is being used)
- Operating System
- Environment (including the installation path and the process ID that contains the failing thread)
- Deployment (including the component and, where applicable, the execution group name and UUIDs)
- Build Information (for IBM internal use)
- Failure Location (including the time of failure and, where applicable, the message flow name)

The content and organization of the abend file and its header are subject to change without notice.

Periodically clear any unwanted abend files from the errors or z/OS log directory. You can do this by moving the files to an archive, or by deleting them when they are no longer needed. This ensures that the workpath does not become full; in extreme conditions, your system performance can be degraded if significant space is being used up with old abend files.

Abend in a user-defined extension

Information that applies only when you have written your own C language user-defined extension.

If the broker on WebSphere Event Broker for z/OS abends, and you have a user-defined extension written in the C programming language, you can use the traceback to locate the source of the problem.

The traceback is placed into a CEEDUMP file, which resides in the <component_HFS>/common/errors directory.

Each traceback is preceded by the date, time, and unique identifier; for example, `CEEDUMP file - CEEDUMP.20080924.171754.84017230`

Consider the following example trace in the CEEDUMP file:

```
Traceback:
  DSA Addr  Program Unit  PU Addr   PU Offset  Entry         E Addr    E  Offset   Statement  Load Mod  Service  Status
  38F9DBD0  CEEVRONU      0707D2B8  +00001004  CEEVRONU      0707D2B8  +00001004              CEEPLPKA  HLE7730  Call
  390253A0                1DF418F8  +000000DE  ImbAbend::printStackForCurrentThread(int,bool,const void*,vo
                                                             1DF418F8  +000000DE              *PATHNAM  FP2....  Call
  39025780                1E221258  +000003C2  ImbAbend::terminateProcessInternal(const void*,const bool,vo
                                                             1E221258  +000003C2              *PATHNAM  FP2....  Call
  39026080                1DF457F8  +000005BE  IMBCOND       1DF457F8  +000005BE              *PATHNAM  FP2....  Call
  39026120                0707B2E0  +00001252  CEEVROND      0707B338  +000011FA              CEEPLPKA           Call
  38F9A928  CEEHDSP       06F7C4D0  +000024BC  CEEHDSP       06F7C4D0  +000024BC              CEEPLPKA  HLE7730  Call
  38F99DA8  CEEHRNUH      06F8B010  +00000092  CEEHRNUH      06F8B010  +00000092              CEEPLPKA  HLE7730  Call
  390261E0                38F39BB0  +000000F2  _NumCompute_evaluate
                                                             38F39BB0  +000000F2              *PATHNAM           Exception
  39027B00                33EFF078  +000004E4  ImbCniNode::evaluate(const ImbMessageAssembly&,const ImbData
                                                             33EFF078  +000004E4              *PATHNAM  FP2....  Call
  39028840                201AE2B0  +00000208  ImbDataFlowTerminal::evaluate(const ImbMessageAssembly&)
                                                             201AE2B0  +00000208              *PATHNAM  FP2....  Call
  39028920                201AE078  +000000BE  ImbDataFlowTerminal::propagateInner(const ImbMessageAssembly
                                                             201AE078  +000000BE              *PATHNAM  FP2....  Call
  39029220                201ABD70  +00000552  ImbDataFlowTerminal::propagate(const ImbMessageAssembly&)
                                                             201ABD70  +00000552              *PATHNAM  FP2....  Call
  39029360                32AC4878  +00003C2E  ImbCommonInputNode::run(ImbOsThread*)
                                                             32AC4878  +00003C2E              *PATHNAM  FP2....  Call
  3902BA00                32AD3488  +00000046  ImbCommonInputNode::Parameters::run(ImbOsThread*)
                                                             32AD3488  +00000046              *PATHNAM  FP2....  Call
  3902BA80                1DE7FD98  +00000074  ImbThreadPoolThreadFunction::run(ImbOsThread*)
                                                             1DE7FD98  +00000074              *PATHNAM  FP2....  Call
  3902C400                1E10A2E8  +000000A8  ImbOsThread::innerThreadBootStrapWrapper(void*)
                                                             1E10A2E8  +000000A8              *PATHNAM  FP2....  Call
  3902CD20                1E109E80  +0000025A  ImbOsThread::threadBootStrap(void*)
                                                             1E109E80  +0000025A              *PATHNAM  FP2....  Call
  3902D6A0                1E109E38  +00000008  threadBootStrapWrapper
                                                             1E109E38  +00000008              *PATHNAM  FP2....  Call
  3902D720                0707B2E0  +00001252  CEEVROND      0707B338  +000011FA              CEEPLPKA           Call
  38FAAEE0  CEEOPCMM      00035438  +00000908  CEEOPCMM      00035438  +00000908              CEEBINIT  HLE7730  Call
```

The message you see in the execution group joblog is:

```
CEE0374C CONDITION=CEE3204S TOKEN=00030C84 59C3C5C5 00000000
         WHILE RUNNING PROGRAM _NumCompute_ev WHICH STARTS AT 38F39BB0
         AT THE TIME OF INTERRUPT
         PSW     078D1400 B8F39CA6
         GPR 0-3 00000008 1C097DA8 00000001 1C097D88
         GPR 4-7 390261E0 00000000 1B049080 B8F39C9A
         GPR 8-B 33F093E8 1F9E1808 38F3A1D8 00000000
         GPR C-F 38F98BD8 33FC1B20 39026C90 00000000
         FLT 0-2 0000000000000018  0000000000000000
         FLT 4-6 4018500208C00000  0000000000000000
```

By studying the joblog and the preceding trace output, you can see that the abend is happening in a function called `_NumCompute_evaluate`. The following statement in the trace calls `ImbCniNode::evaluate` which tells you that the error has occurred in a user-defined extension.

Next, you see the following in the joblog or syslog:

```
IEA794I SVC DUMP HAS CAPTURED: 577
DUMPID=006 REQUESTED BY JOB (MQ83BRK )
DUMP TITLE=MQ83BRK ,ABN=S0C4-00000004,C=M7500.600.BRKR
```

followed by:

```
IEF196I IEF285I   DUMP.MVS1.MQ83BRK.D080924.T171755.S00005     CATALOGED
IEF196I IEF285I   VOL SER NOS= PSSD01.
IEA611I COMPLETE DUMP ON DUMP.MVS1.MQ83BRK.D080924.T171755.S00005 486
DUMPID=005 REQUESTED BY JOB (MQ83BRK )
FOR ASID (00BF)
INCIDENT TOKEN: PLEXS    MVS1     09/24/2008 16:18:15
ID = MQ83BRK ,ABN=S0C4-00000004
```

This shows the location of the dump dataset.

If you are still unable to resolve the problem, send the CEEDUMP file and dump dataset with joblogs and syslogs to your IBM Support Center for analysis.

If you do not have a user-defined extension and the broker abends, you need to send the trace file to your IBM Support Center for analysis.

WebSphere Event Broker event reports

Event messages are published by a WebSphere Event Broker broker in response certain conditions that occur while the broker is active.

Reports are raised in response to:

- "WebSphere Event Broker event reports: configuration changes" on page 134
- "WebSphere Event Broker event reports: operational information" on page 139

Further information about events is provide in the following topics:

- "WebSphere Event Broker event reports: general architecture"
- "WebSphere Event Broker event reports: operational warnings" on page 142
- "WebSphere Event Broker event reports: notification message schema" on page 143

The events are published on a series of system-defined topics. The body of the message contains additional information in XML format. Every message is generated in code page 1208.

The following set of events can be reported:

Configuration changes

- An execution group has been created, changed, or deleted
- A message flow has been created, changed, or deleted
- A neighbor has been created, changed, or deleted
- An ACL for a topic has been created, changed, or deleted

Operational information

- A broker has been started or stopped
- A message flow has been started or stopped
- A subscription has been registered or de-registered

Operational warnings

- A retained publication has expired
- A subscription has expired

WebSphere Event Broker event reports: general architecture

Brokers publish messages on reserved topics after significant events within the broker. By subscribing to these topics, a client can be informed when these events occur.

For each topic, the type of event and message body are explained. The body of these messages is in XML format.

An event publication can contain more than one entry if the topic is the same (for example, if several message flows are created in the same operation).

The general form of the system topics on which events are published is:
`$SYS/Broker/broker_name/event_type/...`

where:

broker_name
is the name of the broker issuing or raising this event.

event_type
is the type of the event and is one of:

- Configuration
- Neighbor
- Subscription
- Topic
- Status
- Expiry

This specification of topics helps clients to filter events, based on the broker from which the event originated and the type of event. The clients register subscriptions for these topics to receive the reports.

For specific events, additional information is included in the topic to help filter on the specific object that raised the event. The inclusion of the string `Broker` at the second level of the topic hierarchy allows for extension should other subsystems (for example, the queue manager or Configuration Manager) that publish system management events through the broker be supported in the future.

WebSphere Event Broker event reports: configuration changes

Configuration changes include changes to the operational configuration of a single broker (for example, the addition or removal of a message flow) and changes to the topology for a multibroker network.

This information is described in the following topics:

- "Changes to the local configuration of the broker"
- "Neighbor changes" on page 135
- "ACL updates" on page 137

Changes to the local configuration of the broker

Notification of changes to the broker's configuration (create, change, or delete entities) is provided by publishing events on the following system topic:
`$SYS/Broker/broker_name/Configuration/ExecutionGroup/exec_grp_name`

where:

broker_name
is the name of the broker issuing this message

exec_grp_name
is the name of the execution group for which the configuration has changed

One such event is published for each configuration request message that is received and processed by an execution group within the broker and can thus contain information that reflects complex configuration changes to multiple entities within the broker.

The body of each publication is the part of the configuration request that causes the event to be triggered. If an execution group is renamed, subsequent publications that report the state of that execution group use the new name.

These events are published non-persistently as non-retained publications.

Only create, change, and delete actions on the message flow are reported.

Configuration change:

The following figure shows an example notification when a message flow is created. The number of attributes mentioned in the example can vary.

```
<Broker uuid="1234" label="Broker1" version="1">
 <ExecutionGroup uuid="2345">
  <Create>
   <MessageFlow uuid="3456" label="MessageFlow1">
    <!-- Create the Input and Output Nodes -->
    <ComIbmMqInputNode uuid="4567"
     queueName="InputQueue1" label="InputNode1" />
    <ComIbmMqOutputNode uuid="5678"
                    queueName="OutputQueue1"
                    label="OutputNode1"/>
    <ComIbmMqOutputNode uuid="6789"
     queueManagerName="QueueManager1"
     queueName="OutputQueue2"
                    label="OutputNode2"/>
    <!-- Create the filter -->
    <ComIbmFilterNode uuid="7890"
                      filterExpression="Company=IBM"
                      label="FilterNode1"/>
    <!-- Connect them together -->
    <Connection sourceNode="4567"
     sourceTerminal="out"
     targetNode="7890" targetTerminal="in"/>
    <Connection sourceNode="7890"
     sourceTerminal="true"
     targetNode="5678" targetTerminal="in"/>
    <Connection sourceNode="7890"
     sourceTerminal="false"
     targetNode="6789" targetTerminal="in"/>
   </MessageFlow>
  </Create>
 </ExecutionGroup>
</Broker>
```

Neighbor changes

A change in the set of neighbors (the topology) for a given broker publishes an event using the following system topic:

`$SYS/Broker/`*`broker_name`*`/Neighbor`

where *`broker_name`* is the name of the broker issuing this message.

The body of each publication is an XML message that describes the change made.

These events are nonpersistent, non-retained publications.

Certain operations delete all neighbor (topology) information at a broker. In this case, the deleted neighbors are not published in the event publication. Instead, the body of the event publication contains a single XML tag that indicates that all neighbors have been deleted.

Examples:

The following example event messages show a neighbor being created, changed, and deleted, together with an example of all neighbors being deleted.

Neighbor created:

```
Event publication topic = "$SYS/Broker/Broker1/Neighbor"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Create>
        <Neighbor name="5678" collectiveId="">
          <MQBrokerConnection queueManagerName="nbr_QM_Name"/>
        </Neighbor>
      </Create>    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

Neighbor changed:

```
Event publication topic = "$SYS/Broker/Broker1/Neighbor"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Change>
        <Neighbor name="5678"
          collectiveId="12345678-1234-1234-1234-123456789abc"/>
      </Change>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

Neighbor deleted:

```
Event publication topic = "$SYS/Broker/Broker1/Neighbor"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Delete>
        <Neighbor name="5678"/>
      </Delete>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

All neighbors deleted:

```
Event publication topic = "$SYS/Broker/Broker1/Neighbor"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Delete>
        <AllNeighbors/>
      </Delete>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

ACL updates

The creation, deletion, or modification of the ACL associated with a topic causes a publication using the following system topics:

`$SYS/Broker/`*`broker_name`*`/Topic/`*`topic`*

where:

broker_name
: is the name of the broker issuing this message

topic
: is the topic whose ACL is being modified

The body of each publication is an XML message that describes the ACL update.

These events are nonpersistent, non-retained publications.

Certain operations delete all ACL entries for a single topic or for all topics. In this case, the individual entries are not published in the event publication. Instead, the body of the event publication contains a single XML tag that indicates that all ACL entries have been deleted.

Examples:

Here are example event messages for an ACL being created, changed, and deleted, together with examples for all ACLs being deleted on a single topic and all topics.

ACL created:

```
Event publication topic = "$SYS/Broker/Broker1/Topic/stock/IBM"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Topic name="stock/IBM">
        <Create>
          <ACLEntry
            principalName="Fred"
            principalType="user"
            publish="false"
            subscribe="inherit"
            persistent="true"/>
        </Create>
      </Topic>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

ACL changed:

```
Event publication topic = "$SYS/Broker/Broker1/Topic/stock/IBM"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Topic name="stock/IBM">
        <Change>
          <ACLEntry
            principalName="Fred"
            principalType="user"
            publish="true"
            subscribe="false"
            persistent="inherit"/>
        </Change>
      </Topic>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

ACL deleted:

```
Event publication topic = "$SYS/Broker/Broker1/Topic/stock/IBM"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Topic name="stock/IBM">
        <Delete>
          <ACLEntry principalName="Fred"/>
        </Delete>
      </Topic>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

All ACLs deleted on a single topic:

```
Event publication topic = "$SYS/Broker/Broker1/Topic/stock/IBM"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Topic name="stock/IBM">
        <Delete>
          <AllACLEntries/>
        </Delete>
      </Topic>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

All ACLs deleted on all topics:

```
Event publication topic = "$SYS/Broker/Broker1/Topic"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
        <Delete>
          <AllACLEntries/>
        </Delete>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

WebSphere Event Broker event reports: operational information

Changes to the processing state of a broker or an individual message flow publish events with predefined system topics.

The following system topics are used:

`$SYS/Broker/`*`broker_name`*`/Status`

`$SYS/Broker/`*`broker_name`*`/Status/ExecutionGroup/`*`exec_grp_name`*

where:

broker_name
: is the name of the broker whose execution state has changed

exec_grp_name
: is the name of the execution group that contains the message flow whose execution state has changed

The body of each publication is an XML message that gives additional information concerning the state change that caused the event to be triggered, specifically indicating whether the entity has been started or stopped.

For example, starting a message flow generates the following message:

```
<Broker uuid="1234" label="Broker1" version="1">
<ExecutionGroup uuid="5678">
  <Start>
   <MessageFlow uuid="7812"/>
  </Start>
 </ExecutionGroup>
</Broker>
```

Stopping a broker generates the following message body:

```
<Broker uuid="1234" label="Broker1" version="1">
 <StatusChange state="Stopped"/>
</Broker>
```

Currently, the only states that are notified for both brokers and message flows are Started and Stopped.

These events are nonpersistent, retained publications.

Subscriptions and topics

Events are published to provide notification of changes to the subscription tables, changes in the list of defined topics or their access control lists.

Registration or de-registration of subscriptions causes events to be published using the following system topics:

`$SYS/Broker/`*`broker_name`*`/Subscription/`*`topic`*

where:

broker_name
: is the name of the broker issuing this message

topic
: is the original topic on which the subscription is being, or was, registered

The body of each publication is an XML message giving additional information concerning the registration or de-registration request.

These events are nonpersistent, non-retained publications.

Examples:

Here are example event messages for a subscription being created, changed, and deleted.

Subscription created:

```
Event publication topic = "$SYS/Broker/Broker1/Subscription/stock/IBM"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Topic name="stock/IBM">
        <Create>
          <Subscription
            clientId="mqrfh2:Broker1:client1queue"
            subscriptionPoint="poundsSterling"
            filter="currentPrice>100"
            user="Fred"
            persistent="true"
            localOnly="false"
            pubOnReqOnly="false"
            informIfRet="true"
            expiryTimeStamp="2000-12-31 23:59:59"
            createTimeStamp="2000-01-01 00:00:00"
            tempDynamicQueue="false"
            clientContext="hex digits"/>
        </Create>
      </Topic>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

Subscription changed:

```
Event publication topic = "$SYS/Broker/Broker1/Subscription/stock/IBM"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Topic name="stock/IBM">
        <Change>
          <Subscription
            clientId="mqrfh2:Broker1:client1queue"
            subscriptionPoint="poundsSterling"
            filter="currentPrice>100"
            user="Fred"
            persistent="false"
            localOnly="true"
            pubOnReqOnly="true"
            informIfRet="false"
            expiryTimeStamp="2005-12-31 23:59:59"
            createTimeStamp="2000-01-01 00:00:00"
            tempDynamicQueue="false"
            clientContext="hex digits"/>
        </Change>
      </Topic>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

Subscription deleted (deregistered):

```
Event publication topic = "$SYS/Broker/Broker1/Subscription/stock/IBM"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Topic name="stock/IBM">
        <Delete>
          <Subscription
            clientId="mqrfh2:Broker1:client1queue"
            subscriptionPoint="poundsSterling"
            filter="currentPrice>100"
            user="Fred"/>
        </Delete>
      </Topic>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

WebSphere Event Broker event reports: operational warnings

Operational warnings provide notification about expired subscriptions or retained publications.

Details about these warnings are provided n "Expired publications and subscriptions."

Expired publications and subscriptions

Notification of the expiry of retained publications and subscriptions is published on the following system topics:

`$SYS/Broker/`*`broker_name`*`/warning/expiry/Publication/`*`timestamp`*`/`*`topic`*

`$SYS/Broker/`*`broker_name`*`/warning/expiry/Subscription/`*`timestamp`*`/`*`topic`*

where:

broker_name
: is the name of the broker issuing this message

timestamp
: is the time stamp of the expiry of the subscription or retained publication (expressed as a GMT time)

topic
: is the topic of the subscription or retained publication

These events are nonpersistent, non-retained publications.

Examples:

The following example event messages show a retained publication and subscription having expired.

Publication expired:

```
Event publication topic =
"$SYS/Broker/Broker1/warning/expiry/Publication/2000-12-31
23:59:59/stock/IBM"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Topic name="stock/IBM">
        <Delete>
          <RetainedPublication
            subscriptionPoint="poundsSterling"/>
        </Delete>
      </Topic>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

Subscription expired:

```
Event publication topic =
"$SYS/Broker/Broker1/warning/expiry/Subscription/2000-12-31
23:59:59/stock/IBM"

<Broker uuid="1234" label="Broker1" version="1">
  <ControlGroup>
    <DynamicSubscriptionEngine>
      <Topic name="stock/IBM">
        <Delete>
          <Subscription
            clientId="mqrfh2:Broker1:client1queue"
            subscriptionPoint="poundsSterling"
            filter="currentPrice>100"
            user="Fred"/>
        </Delete>
      </Topic>
    </DynamicSubscriptionEngine>
  </ControlGroup>
</Broker>
```

WebSphere Event Broker event reports: notification message schema

The following specification describes the structure of all valid notification messages. This example describes the structure of the messages only. It does not define about how many elements are in the messages or the order in which they appear. The rules for the number of elements are:

- One broker element
- Other elements: zero, one, or more

There are no rules for the order of notification messages.

In this example, `<...>` denotes an XML element, and `???` indicates where individual class names are allowed:

```
<Broker identifier label>
. <ExecutionGroup identifier>
. . <Create>
```

```
. . . <MessageFlow message_flow_identifier message_flow_attributes>
. . . . <???Node node_identifier node_attributes>
. . . . <Connection connection_identifier>
. . <Change>
. . . <MessageFlow message_flow_identifier message_flow_attributes>
. . <Delete>
. . . <AllMessageFlows>
. . . <MessageFlow message_flow_identifier>
. . <Start>
. . . <AllMessageFlows>
. . . <MessageFlow message_flow_identifier>
. . <Stop>
. . . <AllMessageFlows>
. . . <MessageFlow message_flow_identifier>
. <ControlGroup>
. . <DynamicSubscriptionEngine>
. . . <Create>
. . . . <Neighbor neighbor_identifier neighbor_attributes>
. . . . . <MQBrokerConnection mqbrokerconnection_attributes>
. . . . <Neighbor neighbor_identifier neighbor_attributes>
. . . . <Topic topic_identifier>
. . . . . <ACLEntry>
. . . . . <Subscription>
. . . <Change>
. . . . <Neighbor neighbor_identifier neighbor_attributes>
. . . <Delete>
. . . . <AllNeighbors>
. . . . <Neighbor neighbor_identifier>
. . . . <Topic topic_identifier>
. . . . <AllACLEntries>
. . . . <AllSubscriptions>
. . . . <AllRetainedPublications>
. . . <Topic topic_identifier>
. . . . <Create>
. . . . . <ACLEntry acl_identifier acl_attributes>
. . . . . <Subscription subscription_identifier subscription_attributes>
. . . . <Change>
. . . . . <ACLEntry acl_identifier acl_attributes>
. . . . . <Subscription subscription_identifier subscription_attributes>
. . . . <Delete>
. . . . . <AllACLEntries>
. . . . . <ACLEntry acl_identifier>
. . . . . <AllSubscriptions>
. . . . . <Subscription subscription_identifier>
. . . . . <RetainedPublication retained_publication_identifier>
```

Identifiers and attributes used in the schema

Items shown as `[ .... ]` are optional attributes. Items shown as {xxx | yyy} mean that the value can be one of the alternatives given. Items shown in *italics* mean that the variable can have any value.

uuid The universally unique identifier for WebSphere Event Broker objects (for example, nodes in a message flow)

client identifier
The delivery destination for the subscriber. For example:

```
mqrfh2:WBRK_QM:subscriberqueue
mqrfh:WBRK_QM:subscriberqueue2:CorrelId
```

identifier
uuid="*uuid*"

label label="*label*"

message_flow_identifier

uuid="*message_flow_identifier*" //This is usually a uuid

message_flow_attributes

[label='*label*']
[additionalInstances="*number*"]
[commitCount="*number*"]
[commitInterval"*number*"]
[coordinatedTransaction={"yes" | "no" }]

node_identifier

uuid='*node_flow_identifier*' //This is usually a number of concatenated uuids

node_attributes

[label='*label*']

Others vary according to type. All are optional. See the node descriptions for details.

connection_identifier

sourceNode='*source_node identifier*'
sourceTerminal='*source_terminal_name*'
targetNode='*target_node_identifier*'
targetTerminal='*target_terminal_name*'

neighbor_identifier

name="*uuid*"

neighbor_attributes

collectiveId={"" | "*uuid*"}

CollectiveId is *""* if the neighbor is in the same collective as the broker; otherwise, it is the identifier for the neighbor's collective.

mqbrokerconnection_attributes

queueManagerName="*queueManagerName*"

topic_identifier

name="*topicName*"

acl_identifier

principalName="*userIdentifier*"

acl_attributes

principalType={"user" | "group"}
[publish={"yes" | "no" | "inherit"}]
[subscribe={"yes" | "no" | "inherit"}]
[persistent={"yes" | "no" | "inherit"}]

subscription_identifier

clientId="client identifier"
[subscriptionPoint="*subscriptionPointName*"]
[filter="*filterExpression*"]

subscription_attributes

userId="*userIdentifier*"
persistent={"true" | "false" | "asPublish" | "asQDef"}
localOnly={"true" | "false"}
pubOnReqOnly={"true" | "false"}
informIfRet={"true" | "false"}
expiryTimeStamp={"*GMTTimeStamp*" | "0" }
createTimeStamp="*GMTTimeStamp*"
tempDynamicQueue={"true" | "false"}
clientContext="*clientContext*"

retainedpublication_identifier
[subscriptionPoint="*subscriptionPointName*"]

WebSphere MQ facilities

WebSphere Event Broker components depend on WebSphere MQ resources in many ways. You can therefore gain valuable information from the WebSphere MQ logs and events.

WebSphere MQ logs

The WebSphere MQ product logs can be very useful in diagnosing errors that occur in your broker domain. For example, if the Configuration Manager cannot communicate with a broker, the channels that connect them might be wrongly configured, or experiencing network problems.

On distributed systems, operational messages in a user-readable format (such as `queue manager started`), are written to the error logs in the errors subdirectory of the queue manager's subdirectory.

FFST files

First Failure Support Technology (FFST) records are normally severe, unrecoverable errors, and indicate either a configuration problem with the system or a WebSphere MQ internal error. FFST files are named AMQnnnnn.mm.FDC, where nnnnn is the ID of the process that is reporting the error, and mm is a sequence number. On Windows, records are written to the *install_dir*\errors directory. Operational messages and FFST records are also written to the Event log. On UNIX and Linux systems, records are written to the /var/mqm/errors directory. WebSphere MQ writes one line for each FFST to the syslog containing the name of the FFST file, but no operational messages.

WebSphere MQ events

WebSphere MQ provides information about errors, warnings, and other significant occurrences in queue managers in the form of instrumentation event messages.

You can activate event activity using the MQSC or PCF interfaces in three areas:

- Queue manager events
- Performance events
- Channel events

When active, these event messages are sent to event queues that can be monitored or triggered. You might find it appropriate to activate WebSphere MQ events when you are investigating the performance, or unexpected behavior, of your WebSphere Event Broker broker domain.

Database facilities

The database products used by WebSphere Event Broker also record information that might be useful if you have any problems with their access. Refer to the database product documentation for details of logs and other problem determination options.

Database logs

DB2 has a number of facilities that assist you with problem diagnosis and recovery. For example, there are the error logs, db2diag.log on distributed systems, and db2alert.log on z/OS, that contain error and alert information recorded by various components of the DB2 product. Refer to the *DB2 Troubleshooting Guide* for comprehensive information on what options are available, how to use them, and how to interpret the information provided.

Other database products have similar logs; consult your database administrator or database documentation for more information.

ODBC trace

You can trace ODBC activity; follow the instructions in "ODBC trace" on page 100.

Trace information is output to the location specified in the odbc.ini file, or in the ODBC configuration on Windows. By default, this is /var/wmqi/odbc/odbctrace.out on non-Windows systems.

Other sources of diagnostic information on z/OS

This section describes some other files that you might find useful for problem determination on z/OS.

Files from a broker, Configuration Manager, or User Name Server

The broker, Configuration Manager, and User Name Server put files into the file system. If you do not know where the files are for the broker, look at the `MQSI_REGISTRY` value in the component profile (`BIPBPROF`, `BIPCPROF`, or `BIPUPROF`) in the component dataset. This is set to a value like `/u/argo/VCP0BRK`. Be careful to access the file system on the correct system; if the file system is not shared, there could be the same directory on each MVS image. To be sure, log on to the system where the broker is running and access the file system. You can use the **TSO ISHell** command, or go into OMVS itself. Go to the directory specified and select the output directory. The traceodbc file should appear in this directory.

When submitting JCL from the component PDSE, output files are written to the `++HOME++` location. The following files should appear in this directory:

- ENVFILE
- bipprof

Files in the home directory

Other files are stored in the home directory of the user ID of the started task. The MVS command **D A, jobname** displays the user ID of the job. The directory of the user ID is typically /u/userid but you should check this with your system programmer, or issue the TSO command **LU userid OMVS**. If you are authorized, this command displays text including the HOME statement.

There might be several different sorts of files in the home directory, such as core dump files.

Problems accessing files in the file system

If you cannot access a file system file or directory, it might be that you do not have permission to do so. This could mean that you are not able to issue certain commands. You need to ask the owner of the file or directory to give you permission to use the file.

Solutions to similar problems

There are a variety of resources that contain information about problems with IBM licensed programs and their use. Searching these resources might help you to find information that helps solve your problem.

Useful Web sites

There are a variety of Web sites available on the Internet, not all managed by IBM, that you might find useful aids to problem determination. Here are some URLs that you might try:

- WebSphere Event Broker support Web page
- WebSphere MQ Integrator newsgroup
- MQSeries.net

RETAIN

IBM keeps records of all known problems with its licensed programs on its software support database (RETAIN®). IBM Support Center staff continually update this database as new problems that are found, and they regularly search the database to see if problems that they are told about are already known.

If you have access to one of IBM's search tools, such as INFORMATION/ACCESS or INFORMATION/SYSTEM, you can look on the RETAIN database yourself. If not, you can contact the IBM Support Center to search for you.

You can search the database using a string of keywords to see if a similar problem already exists.

You can use the keyword string (also called the symptom string) that appears in a dump or SYS1.LOGREC record to search the database, or you can build your own keyword string.

If the search is successful, you find a similar problem description and, usually, a fix. If the search is unsuccessful, use these keywords when contacting IBM for additional assistance, or when documenting a possible authorized program analysis report (APAR).

Part 3. Appendixes

© Copyright IBM Corp. 2000, 2009

Appendix. Notices for WebSphere Event Broker

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 information 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 information. The furnishing of this information does not give you any license to these patents. You can send license inquiries, in writing, to:

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

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

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

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the information. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this information 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.

© Copyright IBM Corp. 2000, 2009

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

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

IBM United Kingdom Laboratories,
Mail Point 151,
Hursley Park,
Winchester,
Hampshire,
England
SO21 2JN

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

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

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

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

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

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

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not

been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

(C) (*your company name*) (*year*). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. *_enter the year or years_*. All rights reserved.

Trademarks in the WebSphere Event Broker information center

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at Copyright and trademark information at `www.ibm.com/legal/copytrade.shtml`.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel and Pentium are trademarks of Intel Corporation in the United States and other countries.

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

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

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

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

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

Index

Special characters

$SYS/Broker 133

A

abend files
  checking 107
  general 130
  user-defined extension 131
access control lists
  updates 137
application log 87

B

BIP messages 5
BIP2130 error message 43
broker
  problems
    starting 18
broker domains
  logs 122
brokers
  configuration
    changing 134
    local 134
  joblogs 123
  recovering after failure 112

C

commands
  problems
    running 11
    special characters in 7
configuration
  broker
    changing 134
    local 134
  syslog daemon 88
Configuration Manager
  problems
    starting 24
  recovering after failure 111
Configuration Manager Proxy
  problems 83
  trace 103
configuration repository, recovering after failure 117
core dumps 128

D

databases
  facilities 146
  problems
    initial checks 7
    resolving 66
debugging
  initial checks 3
deployment
  problems
    after deployment 57
    canceling deployment 57
    during deployment 52
    preparing to deploy 51
diagnosis 121
  techniques 3
  typical problems 10
dumps 128
  checking 104
  core dumps 128
  DUMP command, z/OS 105
  SVC dumps 128

E

Eclipse error log 122
  viewing 89
error diagnosis 121
  techniques 3
  typical problems 10
error logs 87
  application log 87
  broker domain logs 122
  database logs 146
    viewing 125
  Eclipse error log 122
    viewing 89
  local error log 87
  system log 87
  TDS log 122
  WebSphere Message Broker log 122
  WebSphere MQ log 146
    viewing 124
  z/OS joblogs 123
error messages 5
events
  access control lists (ACLs)
    updates 137
  configuration changes
    broker 134
    local broker 134
  defined topics list 140
  neighbor changes 135
  publications 133
  reports 133
  state changes, processing 139
  subscriptions
    deregistration 140
    registration 140
    tables 140
  topology changes 135
execution groups
  recovering after failure 113

F

FFST files 146
fix packs 9

H

hidden files 5

I

IBM Support Assistant 109
  installing 110
  updating 110
IBM Support Center 107
importing
  problems 32
installation
  checking 4
ISA 109
  installing 110
  updating 110

J

joblogs, z/OS 123

L

local error log 87
logs 87
  application log 87
  broker domain logs 122
  database logs 146
    viewing 125
  Eclipse error log 122
    viewing 89
  local error log 122
    viewing 87
  system log 87
  TDS log 122
  WebSphere Message Broker log 122
  WebSphere MQ log 146
    viewing 124
  z/OS joblogs 123

M

Message Brokers Toolkit
  problems uninstalling 85
message flows
  problems
    after deployment 57
    appearance 49
    canceling deployment 57
    deploying 50
    developing 42
    during deployment 52
    execution 43

© Copyright IBM Corp. 2000, 2009

message flows *(continued)*
problems *(continued)*
importing 33
migrating 33
preparing to deploy 51
trace 48
unsuccessful run 6
messages
BIP 5
notification
attributes and identifiers 144
schema 143
migration
problems 32
problems with message flows 43

N

neighbor changes, event message 135
network problems 8

O

ODBC
trace 100
operational warnings 142

P

performance
problems 80
problems
BIP messages 5
broker
starting 18
commands
running 11
special characters 7
Configuration Manager
starting 24
Configuration Manager Proxy 83
Configuration Manager Proxy
trace 103
databases, initial checks 7
databases, resolving 66
diagnosis 121
diagnostic techniques 3
error messages 5
fix packs 9
hidden files 5
initial checks 3
installation, checking 4
logging off Windows 4
maintenance updates 9
message flows
after deployment 57
appearance 49
canceling deployment 57
deploying 50
developing 42
during deployment 52
execution 43
preparing to deploy 51
running 6
trace 48
network 8
problems *(continued)*
ODBC trace 100
passwords 8
performance 80
publish/subscribe 74
reproducing 5
resources
creating 11
deleting 38
importing 32
migrating 32
starting 17
stopping 37
service updates 9
setup, checking 4
slow components 9
starting
broker 18
Configuration Manager 24
other resources 26
system changes 7
typical 10
uninstalling 85
UNIX environment variables 4
user IDs 8
user-defined extensions 84
WebSphere MQ Java Client trace 103
workbench 59
appearance 64
connecting 60
error messages 62
z/OS
checking 9
diagnostic information 147
publish/subscribe
problems 74

Q

queue managers
recovering after failure
broker's queue manager 114
Configuration Manager's queue
manager - z/OS 115
shared queue manager 116
User Name Server's queue
manager 116

R

recovery after failure 111
broker database failure 117
broker failure 112
broker's queue manager failure 114
Configuration Manager failure 111
Configuration Manager's queue
manager failure 114
Windows, Linux, UNIX
systems 115
z/OS 115
configuration repository failure 117
execution group failure 113
shared queue manager failure 116
User Name Server failure 113
User Name Server's queue manager
failure 116
resources
problems
creating 11
deleting 38
importing 32
migrating 32
starting 17
stopping 37
RETAIN database 148

S

service
problems after applying 9
updates 9
service trace 127
options
changing 95
checking 94
retrieving 96
starting 94
stopping 95
service updates 9
setup, checking 4
subscriptions
expired 142
Support Assistant, IBM 109
installing 110
updating 110
Support Center, IBM 107
SVC dumps 128
syslog daemon 88
system log 87

T

TDS format
log 122
TDS log 122
Toolkit
problems uninstalling 85
topology
changes 135
trace 125
changing settings 99
clearing files 98
Configuration Manager Proxy 103
formatting 96
how to use 90
interpreting 98
ODBC trace 100
service trace 127
user trace 126
WebSphere MQ Java Client trace 103
trademarks 153
troubleshooting 121
techniques 3
typical problems 10

U

uninstalling
problems 85
UNIX
environment variables 4

User Name Server
  recovering after failure 113
user trace 126
  options
    changing 92
    checking 91
  retrieving 93
  starting 90
  stopping 93
user-defined nodes
  problems 84

W

warnings, operational 142
WebSphere MQ
  facilities 146
  log 124
WebSphere MQ Java Client trace 103
workbench
  changing trace settings 99
  problems 59
    appearance 64
    connecting 60
    error messages 62

Z

z/OS
  DUMP command 105

Printed in USA