

IBM® DB2® Information Integrator

Q Analyzer Guide

How to gather information about Q replication environment

Version 8.2



Table of contents

1 Overview of the Q Analyzer	3
2 Installing the Q Analyzer	4
2.1 Prerequisites	4
2.2 Installing the Q Analyzer	5
3 Gathering data with the Q Analyzer	7
3.1 Before you run the Q Analyzer	7
3.2 Running the Q Analyzer	7
3.3 The two files that are outputted	14

1 Overview of the Q Analyzer

The Q Analyzer is a program that gathers data about your Q replication configuration to help technical support representatives determine the cause of possible problems. This document explains how to set up the program and use it to gather data about your Q replication configuration to give to technical support representatives.

The Q Analyzer gathers information from the following sources:

- Q Capture control tables
- Q Apply control tables
- Q replication log files
- WebSphere® MQ queue managers

The program generates an XML document with the information that it collected.

You can run the Q Analyzer on any Linux™, UNIX™, or Windows™ platform. It can gather the information from the z/OS® platform.

2 Installing the Q Analyzer

This section describes the prerequisites for the Q Analyzer and how to install it. You must install the following programs and libraries before you start the Q Analyzer.

2.1 Prerequisites

The Q Analyzer shares several Java libraries with the administration tools for Q replication.

1. Add Q replication Java™ libraries to the CLASSPATH environment variable

Ensure that the following files are in the CLASSPATH:

`<INSTALL DIRECTORY>\sqllib\java\Common.jar`

`<INSTALL DIRECTORY>\sqllib\tools\db2cmn.jar`

`<INSTALL DIRECTORY>\sqllib\tools\db2qreplapis.jar`

`<INSTALL DIRECTORY>\sqllib\java\db2java.zip`

2. Ensure you have the correct Java Runtime Environment

The Q Analyzer requires a Java runtime environment of Java 1.3.1 or later. You can download Java 1.3.1 from the Sun Microsystems™ Web site by using the URL below.

Java 1.3.1

<http://java.sun.com/j2se/1.3/download.html>

3. Optional: Install the WebSphere MQ Java Application Programming Interface

If the Q Analyzer is running on a machine other than where the queue manager is, ensure that you have the WebSphere MQ Java Application Programming Interface library installed where you run `gather`. This WebSphere MQ Java library allows the Q Analyzer to connect to a WebSphere MQ queue manager and gather information about the queues.

This WebSphere MQ Java library is optional, but, without it, the Q Analyzer cannot connect remotely to a queue manager and therefore cannot retrieve information from about its queues if the database is cataloged from a remote system. If the database is local and the WebSphere MQ Java library is not present, then the Q Analyzer attempts to connect with the `runmqsc` command interface to query the queue data. This library is to be installed in the system where `gather` is run. You can optionally install this WebSphere MQ Java Messaging component with WebSphere MQ 5.3.

After you install the Websphere MQ Java library, add the following files to the CLASSPATH:

```
<INSTALL DIRECTORY>\com.ibm.mq.jar
```

```
<INSTALL DIRECTORY>\connector.jar
```

2.2 Installing the Q Analyzer

This section describes how to download and install the Q Analyzer.

1. Download and install the Q Analyzer

You can download the Q Analyzer in several archived formats. Choose the package for the platform that you want the Q Analyzer to run on. After you have downloaded the Q Analyzer, un-archive the files to your desired location. The files are in a directory that is named “qanalyzer” in the archive. The following files will be in the directory:

qanalyzer/qanalyzer.jar

qanalyzer/README.PDF (the Q Analyzer Guide that you are reading)

Windows: qanalyzer/gather.exe

Linux and UNIX: qanalyzer/gather

2. Modify the CLASSPATH environment variable

Modify the CLASSPATH environment variable. Add the file “qanalyzer.jar” to the CLASSPATH environment variable by issuing the following commands:

Windows:

```
SET CLASSPATH=%CLASSPATH%;<INSTALL  
DIRECTORY>\qanalyzer\qanalyzer.jar
```

Linux and UNIX:

```
EXPORT CLASSPATH=$CLASSPATH:<INSTALL  
DIRECTORY>/qanalyzer/qanalyzer.jar
```

3. Optional: Modify the PATH environment variable

By modifying the PATH environment variable, you can run the Q Analyzer from any current working directory. Modify the PATH environment variable by issuing the following commands

Windows:

```
SET PATH=%PATH%;<INSTALL  
DIRECTORY>\qanalyzer\gather.exe;
```

Linux and UNIX:

```
EXPORT PATH=$PATH:<INSTALL DIRECTORY>/qanalyzer/gather:
```

3 Gathering data with the Q Analyzer

This section describes how to gather data about your Q replication configuring by using the Q Analyzer. You will need to gather data from each database and schema pair in your configuration.

The Q Analyzer gathers information from one database at a time. After you input a database and a schema, the Q Analyzer checks whether both Q Capture and Q Apply control tables are present. If either or both sets of control tables are present, it collects the appropriate information from those control tables.

3.1 Before you run the Q Analyzer

1. Ensure that you have completed all prerequisites

Make sure that all the PATH and CLASSPATH variables are set and that all other prerequisites have been completed. See section 2.1 Prerequisites.

2. Ensure that the databases are set up correctly

Before you start the Q Analyzer, ensure that the database that you want to collect information from is either local or cataloged to the system where the Q Analyzer runs. If the database is on OS/390, then that database must be cataloged to a Linux, Windows, or UNIX system where the Q Analyzer runs.

Ensure that the DB2 Universal Database is started.

3. Ensure that the logs are set up correctly

Ensure that the Q Capture and Q Apply logs from remotely cataloged databases are transferred over to the current working directory where you will run the Q Analyzer. Make sure that the log files are transferred using ASCII mode if it is being sent via FTP. If the Q Capture and Q Apply programs are running locally, change to the current working directory where the Q Capture and Q Apply log files are stored before you start the Q Analyzer.

4. Ensure that the queue managers are active

Before you start the Q Analyzer, ensure that the queue managers are active. If the queue managers are not active, the Q Analyzer cannot connect to the queue managers to gather information about the queues.

3.2 Running the Q Analyzer

To start gathering information with the Q Analyzer, use the `gather` command. You must run the `gather` command once for each schema that is involved in the Q replication configuration.

To run the `gather` command, type `gather` followed by the desired parameter command from the list of available parameters. You can use these input parameters to filter what information you want to collect. You might want to filter information because the Q replication control tables might contain a large amount of information, which can take a while for the Q Analyzer to gather.

Note: The parameter name is not case sensitive but the values for each parameter is case sensitive.

You can invoke the list of parameters by typing the following command:

```
> gather -help or > gather ?
```

The following list shows the parameters that you can use with the `gather` command .

Mandatory and optional parameters: Only the DATABASE parameter is mandatory; the rest are optional.

database

The name of the database that you want to gather information from.

instance

Default: DB2

This is the name of the DB2 instance where the database resides on LUOW. This parameter is required to resolve the name of the log file that Q Capture and Q Apply create. The naming convention of the log file is a 4 part name consisting of:

<INSTANCE>.<DATABASE>.<SCHEMA>.<COMPONENT>.log

subsystem

Default: DSN1

This is the name of the DB2 subsystem on z/OS where the database resides. This parameter is needed only if the database where information is being collected from is cataloged from a z/OS machine. This parameter is required to resolve the name of the log file that Q Capture and Q Apply create. The naming convention of the log file is a 4 part name consisting of:

<SUBSYSTEM>.<DATABASE>.<SCHEMA>.<COMPONENT>.log

userid

The ID of an account that has access to the database or subsystem. An implicit connect using the current logged in ID is used if you do not provide an explicit user ID. If you do provide an explicit user ID, you must also provide a password file. If you do not, you will be prompted to type the password.

passfile

The name of the file that stores the password for the user ID. You can either provide a password file or you can enter the password when prompted. To create a password file, create an ASCII file that contains only the password. Save this file in the directory where you are running the `gather` command. The password cannot be followed by any trailing blanks, special characters, or carriage returns. Q Analyzer cannot use ASNPWD password files because the encryption algorithm cannot be encrypted in Java. There are also U.S. Government regulated restrictions placed upon the export of cryptographic materials that need review and approval before it can be made available to the general public.

Tip: You might want to use a password file because commands that are issued at the prompt might be logged in the history, and, thus, passwords would be logged as well. A password file also saves you from having to type your password each time you run the Q Analyzer.

schema

Default: ASN

The name of the schema for the Q Capture or Q Apply control tables that you want to collect information about.

If the database or subsystem contains Q Capture control tables with the schema that you provide, the information will be gathered from those control tables. If the database or subsystem contains Q Apply control tables with the schema that you provide, the information will be gathered from those control tables. If a schema is not explicitly provided, the default of ASN will be used.

caplogdir

Default: Current working directory where you are running the Q Analyzer

The name of the directory where the Q Capture program writes the log file.

Tip: If this log file resides on a remote system where capture and apply are running, the log files need to be transferred (via FTP or some other file transport mechanism) to the system where the Q Analyzer program is run. This is especially important for z/OS users because Q Analyzer does not run natively on z/OS and the database is always cataloged to a workstation. If no log files were transferred, and no log files exist in the specified directory, then the `IBMQREP_CAPTRACE` and `IBMQREP_APPLYTRACE` will be read for logging information, however this data is not as complete as the log files.

applogdir

Default: Current working directory where you are running the Q Analyzer

The name of the directory where the Q Apply program writes the log file.

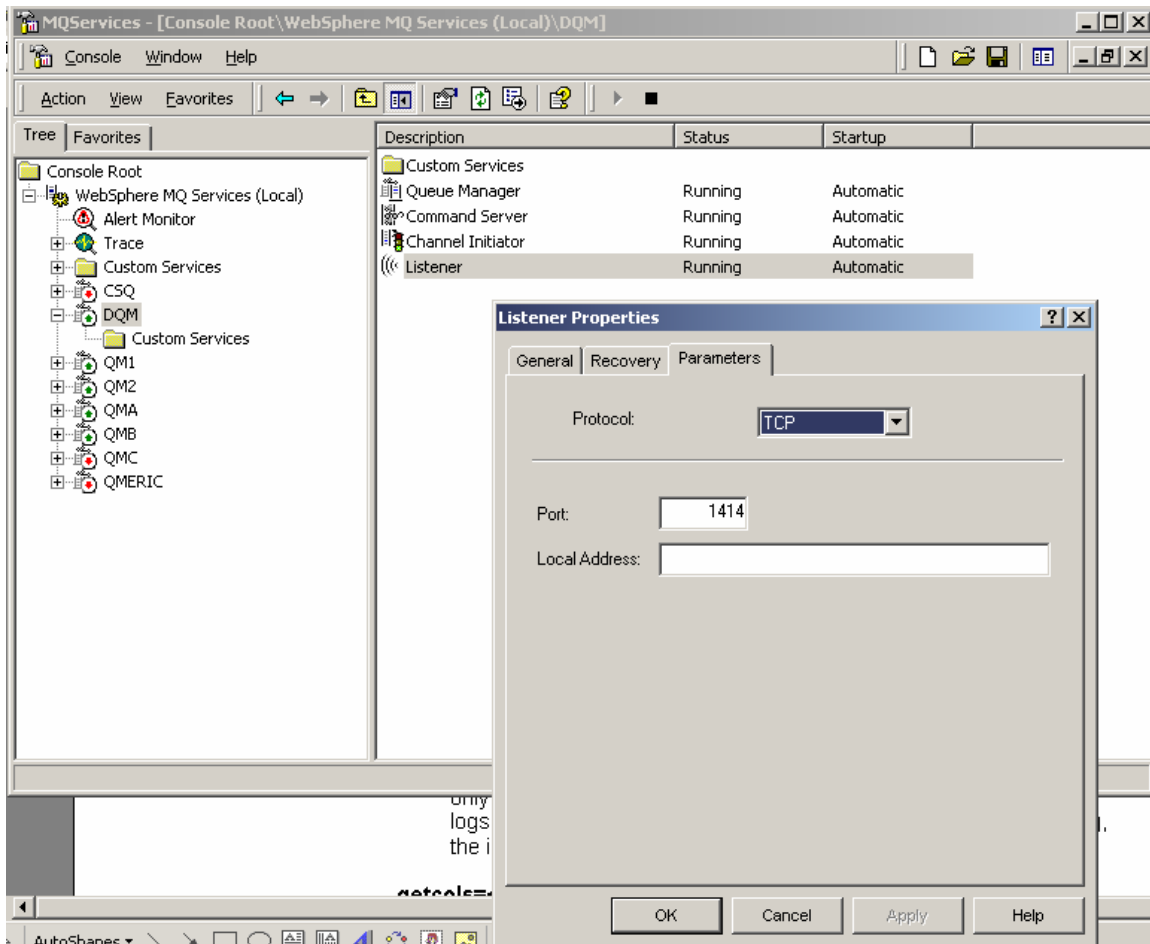
Tip: If this log file resides on a remote system where capture and apply are running, the log files need to be transferred (via FTP or some other file transport mechanism) to the system where the Q Analyzer program is run. This is especially important for z/OS users because Q Analyzer does not run natively on z/OS and the database is always cataloged to a workstation. If no log files were transferred, and no log files exist in the specified directory, then the IBMQREP_CAPTRACE and IBMQREP_APPLYTRACE will be read for logging information, however this data is not as complete as the log files.

port

Default: 1414

The port number that the queue manager listens to for connections and commands. If the database and the queue manager are remote from where the data is being gathered, then you must specify a port.

Tip: In Windows, this can be found via the Websphere MQ Services program.



In Unix and Linux you can try to do a "ps -ef | grep runmqslr" but that doesn't always show the port #. In that case, you will have to check with your MQ Administrator. For z/OS, you must ask the MQ Administrator for the port number.

channel

Default: SYSTEM.DEF.SVRCONN

The name of the server connection channel that the queue manager received commands from.

hostname

Default: LOCALHOST

The IP address or the hostname of the system in which the remote queue manager is located.

warnerrsonly=<OFF, ON>

Default: OFF

This option specifies whether you want the Q Analyzer to collect only the warning and error entries from the Q Capture and Q Apply logs. If you choose the default of OFF, then all messages, including, the informational messages, are collected.

getcols=<ON, OFF>

Default: ON

This option specifies whether you want the Q Analyzer to collect information about only the columns that are being replicated. If you choose YES, then all columns are collected, whether they are being replicated or not.

getmonitor=<ON, OFF>

Default: ON

This option specifies whether you want the Q Analyzer to collect information from the Q Capture and Q Apply monitor information tables.

logdays

Default: 3 (days from the present)

The number of days, from the time Q Analyzer is executed, worth of log record entries the you want the Q Analyzer to collect from the log files, trace tables, signals, and monitor tables. The Q Analyzer will look at the timestamps associated with the records, and only gather those between the current time, and the number of days specified from the current time.

zip=<ON, OFF>

Default: ON

This option specifies what format you want the Q Analyzer to output the data that it collects. If this option is ON, then the output will be a zip file with the XML file archived inside. If this option is OFF, then

the output will be an XML file. Data can get corrupted if `gather` stops in the middle of execution. In this case, delete the zip files and rerun, or turn off the zip option and just have the raw XML outputted.

Example of `gather` command:

The following command collects all the Q replication information from the Q Capture and Q Apply control tables that are stored in the database named "SAMPLEDB" and have the default schema "ASN".

```
> gather DATABASE=SAMPLEDB
```

If you added the correct executable files to the PATH environment variable (see section 2.1 Prerequisites), you can run the `gather` command from any working directory.

Tip: Run the `gather` command from the working directory that the Q Capture and Q Apply programs write the output log files to. That way, if you do not specify a CAPLOGDIR or APPLOGDIR, then the Q Analyzer looks at the current working directory for the log files. If the database is on a remote system, transfer the log files to the system where you will run the `gather` command.

3.3 *The two files that are outputted*

The XML file

After the `gather` command completes successfully, two files are outputted: an XML file (or an XML file compressed in a zip archive) and a log file. (See The log file for more about the log file.) The data that the Q Analyzer collects is outputted to a structured XML file. This is the file that you will send to the technical support representative.

The XML file contains data for only a single schema, which includes both the Q Capture and Q Apply control tables sharing that single schema, on a single database. This data includes information that was collected from the Q replication control tables, the Q Capture and Q Apply log files, and the WebSphere MQ queue managers. The data only represents a snapshot of the Q replication configuration at one particular point in time.

The XML file will be named in the form `<DATABASE>.<SCHEMA>.xml` or `<DATABASE>.<SCHEMA>.zip` if you specified the ZIP parameter. For example, if you ran the `gather` command against a database that is named SAMPLE and the schema of the Q Capture and Q Apply control tables is ASN, the XML file will be named: SAMPLE.ASN.xml.

The log file

After the `gather` command completes successfully, two files are outputted: an XML file and a log file. (For more about the XML file, see section 3.3 The two files that are outputted.) The log file, which is named `qanalyzer.log`, shows the values of the parameters that the Q Analyzer used. The log file is in ASCII format.

Also, if any errors occur, the errors are logged in the `qanalyzer.log` file. For example, if the queue manager was down and the Q Analyzer was not able to connect and get information about the queue, the error is logged in the `qanalyzer.log` file.

Example of the qanalyzer.log file:

```
DATABASE: SAMPLE
INSTANCE: DB2
USERID: Eric
PASSWORD FILE: passfile.pwd
SCHEMA: ASN
Remote Port: 1414
Remote Channel: SYSTEM.ADMIN.SVRCONN
Remote Hostname: 127.0.0.1
Print only warning and error log messages: false
Get rows from tables: false
Current time: 2004-09-21 21:45:03.769
Log time: 2004-09-18 21:45:03.769
Capture Log File: DB2.SAMPLE.ASN.QCAP.log
Apply Log File: DB2.SAMPLE.ASN.QAPP.log
```