

Performing health checks for WebSphere Message Broker and WebSphere MQ

Regularly scheduled health checks are important in an SOA in which an advanced ESB provides connectivity and universal data transformation for both standards-based and non-standards-based applications and services. This article shows you how to perform a health check on WebSphere Message Broker and WebSphere MQ, including log check, message queue check, flow check, and database check.

- [Table of contents](#)

Introduction

IBM® WebSphere® Message Broker delivers an advanced ESB that provides connectivity and universal data transformation for both standard and non-standards-based applications and services to power your SOA. Therefore, it is critical to perform periodic health checks for WebSphere Message Broker, WebSphere MQ (which often serves as the basic message container for an SOA), and DB2® (which is often used to store WebSphere Message Broker configuration data). This article shows you how to perform these health checks, including log check, message queue check, flow check, and database check. The article has five sections:

- Overview of health checks
- Performing health checks on DB2
- Performing health checks on WebSphere MQ
- Performing health checks on WebSphere Message Broker
- Introduction to common problems and their solutions

The article is intended for experienced developers and architects with some knowledge of WebSphere Message Broker, WebSphere MQ, and DB2.

Required product versions

1. WebSphere Message Broker V6 on Linux
2. WebSphere MQ V6 on Linux
3. DB2 V8.2 on Linux

Overview of health checks

A health check includes an operation system and file system check, log check, queue check, and flow check. Here is a brief description of these types of health checks:

Operation system and file system check

Examines the performance and capacity of the operation system and file system. For example, CPU and memory performance are checked to ensure that the system has enough resources to run the applications. The file system is checked to ensure that there is enough free space to store the temporary data and persistent files.

Log check

Checks the logs of the system, applications, and middleware. If there are exceptions in a log file, you need to determine whether they will affect system performance.

Queue check

For WebSphere MQ or similar product, verifies queue configuration, examines the queue length, and performs a connectivity test on the queue.

Flow check

Checks process flow on ESB products such as WebSphere Message Broker and WebSphere Process Server. Verifies flow usability and checks exceptions.

Performing health checks on DB2

This section shows you how to do a file system check and log check on DB2.

DB2 file system check

The DB2 file system check verifies that there is enough free space for database files. If there is not enough free space, database operations such as insert and delete will fail. It is a good idea to put the database log file on a separate disk and make sure this disk has enough space. Use the following command to check the file system space:

```
df -k database_log_disk
```

If the percentage of used space exceeds 90%, watch for possible transaction errors in the database log. To resolve the problem, delete unneeded content from this disk or move the database log to another disk.

DB2 status check

WebSphere Message Broker requires a database such as DB2 to be running. Use the following commands to check the DB2 running status:

```
su - db2inst1
db2start
```

In addition, check the database log file to ensure that there are no exception messages that might affect DB2 functions.

To use the DB2 health check utility to do the DB2 health check, use the following command:

```
su - db2inst1
db2hc
```

Follow the suggestions and documentation of the health check utility to tune DB2 parameters.

Performing health checks on WebSphere MQ

This section shows you how to do a file system check, log check, and queue check on WebSphere MQ.

WebSphere MQ file system check

The file system check ensures that there is enough space available for the WebSphere MQ log. If the disk where it resides fills up, you will get WebSphere MQ transaction errors. Assuming that the log file is at the default location of `/var/mqm`, use the following command to check file system space:

```
df -k /var/mqm
```

If the percentage of used space exceeds 90%, delete unneeded content or add another disk to this logical disk.

WebSphere MQ log check

After the file system check, examine the log files to look for exceptions. The default locations for the WebSphere MQ log files are `/var/mqm/errors` and `/var/mqm/qmgrs/<queue manager name>/errors`.

1. Check `/var/mqm/errors` for any global exceptions related to WebSphere MQ.
2. Check `/var/mqm/qmgrs/<queue manager name>/errors`. If WebSphere Message Broker uses a queue manager, check the file `/var/mqm/qmgrs/WRK6_DEFAULT_QUEUE_MANAGER/errors` for queue manager exceptions.

WebSphere MQ queue check

The queue check examines queue manager status and queue manager components such as listener and channel.

WebSphere MQ queue manager status check

The queue manager contains queue components such as the queue, listener, and channel. In the WebSphere Message Broker runtime, the queue manager that message flows depend on must be running. To check queue manager status check, use the following command:

```
su - mqm
dspmq
```

A list of queue managers with status will be shown. Check whether the queue managers used by WebSphere Message Broker are all running. You can also use WebSphere MQ Explorer to view queue manager status. If a queue manager is not running, use the command `strmqm <queue_manager_name>` to start it.

WebSphere MQ listener status check

A listener in the queue manager is the bridge between the queue manager and the application. The message flow on WebSphere Message Broker connects to the queue manager through the listener, so the listener must be running. To view the status of the listener, use the command `ps -ef|grep runmq1sr | grep <listener_port>`, where `<listener_port>` is the listening port of this listener. For example, `ps -ef|grep runmq1sr | grep 141` gets the status of listeners whose listening ports are 141*, including 1410, 14101, and so on. To view the status of single listener, use the following command:

```
su - mqm
runmqsc <queue_manager_name>
display lsstatus(listener_name)
```

You can also use WebSphere MQ Explorer to view the listener status. If the listener is not running, use the following command to start it:

```
su - mqm
runmqsc <queue_manager_name>
start listener(listener_name)
```

The `runmqsc` command enters the WebSphere MQ command window, and the configuration command to WebSphere MQ can be run through this command window.

WebSphere MQ channel status check

The channel defines a the message path. If user-defined channels are used in the WebSphere Message Broker environment, these channels must be running. To get the detailed information, use the following command:

```
su - mqm
runmqsc <queue_manager_name>
display channel(channel_name)
```

To get the channel status, use the command `display chstatus<channel_name>` from the WebSphere MQ command window. You can also use WebSphere MQ Explorer to view the

status of the channels. If the channel is not running, start it by entering the following command in the WebSphere MQ command window:

```
start channel<channel_name>
```

WebSphere MQ queue depth check

Queue depth remains low if the environment is in normal status. If exceptions occur or message flow is blocked, queue depth will increase in some queues. So you can check overall system health by checking the queue depth. If many messages remain in one queue, exceptions may be blocking the message flow. To determine queue depth, use the following command:

```
su - mqm  
runmqsc <queue_manager_name>  
display ql(*) CURDEPTH
```

* means that this command will show the depth of all queues. WebSphere MQ Explorer can give you an overall picture of queue depth. If there are many messages in one queue, check the message flow and find the pain point. After that, increase the maximum depth of the queue to avoid queue overflows using the command `alter ql(<queue_manager_name>) MAXDEPTH(depth)` from the WebSphere MQ command window. Investigate whether the messages are useful in the production environment. If the messages are not helpful, use the command `clear ql(<queue_manager_name>)` from the WebSphere MQ command window to clear the messages from the queue.

WebSphere MQ queue input/output count check

Queue input/output count is the number of applications that write messages to or read messages from the queue. The input/output count should remain low. A high input/output count may be caused by an exception in the message flow, such as a dead lock loop. To view the queue input/output count, enter the following command in the WebSphere MQ command window:

```
display ql(<queue_manager_name>) IPPROCS OPPROCS
```

Performing health checks on WebSphere Message Broker

This section shows you how to do file system, log, and flow checks on WebSphere Message Broker.

WebSphere Message Broker file system check

As with WebSphere MQ, the WebSphere Message Broker health check includes a file system check and log check. The default log file location is `/var/mqsi`. Use the command `df -k`

/var/mqsi to check the available space in the log file location. If the percentage of used space exceeds 90%, delete unneeded content or add another disk to this logical disk.

WebSphere Message Broker log check

Most WebSphere Message Broker logs are stored in the operation system log. On Windows, use the Event Viewer to get the log information for WebSphere Message Broker. On Linux, the system log syslog is used to store log information, and it must be redirected to a file before you can view it. Open the file /etc/syslog.conf and check for a line similar to user.* - /var/log/user.log to redirect the log file. If it is not present, use the following command to redirect the log information:

```
su - root
cd /var/log
touch user.log
chown root:mqbrkrs user.log
chmod 640 user.log
```

The file /var/log/user.log will be created. Modify the file /etc/syslog.conf to add the following line to the end: user.* - /var/log/user.log

Then restart the demon process syslog using the command /etc/init.d/syslogd restart. The system log is re-directed to the file user.log, and you can open this file to see if there are any exceptions.

WebSphere Message Broker flow check

The flow check includes execution group check and message flow check.

WebSphere Message Broker execution group check

Examines the execution group number and consumed resource. Use the command ps -ef |grep DataFlowEngine to get the statistics of the execution group. Use the command ps aux |grep DataFlowEngine to get the memory usage of the execution group. If the memory usage of an execution group is excessive, there may be a memory leak in the message flow.

WebSphere Message Broker message flow check

Examines message flow status. Message flows and execution groups used by the system should be in running status. To view the status of the execution group, use the following command:

```
mqsiprofile
mqsilist broker_name
mqsilist broker_name -e executiongroup_name
```

If some execution group or message flow is not running, go to the WebSphere Message Broker log to determine the reason -- usually a problem with the application or the message flow.

Common problems and best practices

Input/output count too high

First check if there are any exceptions in the flow using the procedures described above. If system performance is okay, increase the maximum connection parameter of the queue manager: edit the file `/var/mqm/qmgrs/(<queue_manager_name>)/qm.ini` and modify the following lines to change `<channel_no>` to the desired number:

```
CHANNELS:  
    MaxChannels=<channel_no>  
    MaxActiveChannels=<channel_no>
```

Broker can't be started

Sometimes, the broker can't be started using the command `strmqbrk -m <queue_manager_name>`. If the error code is AMQ5855 and the error message is 5081 resource problem, then check the system queue `SYSTEM.BROKER.CONTROL.QUEUE` in this queue manager. This problem is usually caused by a full `SYSTEM.BROKER.CONTROL.QUEUE`. To resolve the problem, clear the messages or increase the queue depth parameter.

System resource check

Before doing the WebSphere Message Broker health check, do a system resource check to ensure that no applications or services are consuming excessive system resources. To view the CPU and memory utilization of every process, use the command `top`. Check processes that are using a lot of CPU and/or memory resources. Also, list all disks to check whether there is enough space.

Conclusion

This article has described health checks for DB2, WebSphere MQ, and WebSphere Message Broker. Perform these health checks regularly help to avoid problems and keep your system stable.