Customers are moving to IBM MQ V9 at the moment, and IBM Support are seeing the same two questions coming up again and again…so here is a blog post that provides the answers.

**Do I need to upgrade WebSphere Application Server in order to connect to IBM MQ V9 queue managers?**

The answer to this question is “No”. Any supported version of WebSphere Application Server can communicate with queue managers that are running on a supported version of IBM MQ.

Embedded into the WebSphere Application Server installation is a component called the WebSphere MQ resource adapter, which handles all of the communication between the application server and a queue manager. The resource adapter contains logic to detect which version of IBM MQ it is connected to, and then use the appropriate functions and features provided by that version.

For example, WebSphere Application Server V8.5.5 includes the WebSphere MQ V7.1 resource adapter. Using this version of the application server to connect to an IBM MQ V9 queue manager, using either the BINDINGS or CLIENT transport, is fully supported.
Figure 1: Applications running inside of WebSphere Application Server V8.5.5 or earlier can connect to IBM MQ V9 queue managers.

For connections using the BINDINGS transport, the WebSphere MQ messaging provider’s native library path needs to be configured to load the IBM MQ native libraries from the IBM MQ V9 installation. Information on how to do this can be found in the Configuring the WebSphere MQ messaging provider with native libraries information topic in the WebSphere Application Server sections of IBM Knowledge Center.
Figure 2: The Native library path for the WebSphere MQ messaging provider needs to be set to point to the directory containing the IBM MQ native libraries.

IBM MQ V9 comes with both a 32-bit and 64-bit native library. If your WebSphere Application Server instance is using a 32-bit Java Runtime Environment (JRE), then the native library path must be set to the location of the 32-bit native library, which is `<MQV9_Installation_Path>\java\lib`. This will cause the application server to load the 32-bit native library. If the application server is using a 64-bit JRE, then the native library path needs to be set to `<MQV9_Installation_Path>\java\lib64` so that the application server will load the 64-bit version of the native library.

Do I need to install the IBM MQ V9 resource adapter into WebSphere Application Server V8.5.5 or earlier to connect to IBM MQ V9 queue managers?

The answer to this question is also “No”. The WebSphere MQ resource adapter component within WebSphere Application Server V8.5.5 and earlier does not need to be manually updated to the IBM MQ V9 level to connect to IBM MQ V9 queue managers.

The IBM MQ V9 resource adapter is compliant with the JMS 2.0 specification, which is part of Java Enterprise Edition V7, and so can only be deployed into Java EE 7 compliant application servers. WebSphere Application Server V9 Traditional is Java EE 7 compliant, and includes the IBM MQ V9 resource adapter. However, earlier versions of WebSphere Application Server are compliant with earlier versions of Java EE and so do not support JMS 2.0. This means that the IBM MQ V9 resource adapter will not work if it is deployed into WebSphere Application Server V8.5.5 or earlier.
As mentioned above, WebSphere Application Server V8.5.5 and earlier can connect to IBM MQ V9 queue managers. They will not be able to make use of any new JMS 2.0 functionality introduced as part of IBM MQ V9, such as shared subscriptions, though. At the time of writing, the only way to use JMS 2.0 functionality when connecting to IBM MQ V9 queue managers is to use WebSphere Application Server V9.

As always, I hope this helps! If you have any additional questions, feel free to ask and I’ll be happy to answer them.

Tags ibm mq, WAS, websphere

by Paul_Titheridge