Running a .NET application compiled using a later version of IBM MQ than the primary installation

Alamelu |June 29 2016 Updated

**Scenario:**

IBM MQ 8.0.0.0 is installed and it is the primary installation. (Since IBM MQ 8.0.0.0 is the primary installation, amqmdnet.dll of Version 8.0.0.0 will be available in GAC). SimplePublish.exe is the MQ .NET application complied with IBM MQ 8.0.0.4. Running the application as such will report a System.IO.FileNotFoundException.



**How to fix this error:**

If there is a primary installation on the system, the MQ .NET assemblies and policy files of that installation are registered to the global assembly cache (GAC). By default, applications use the MQ .NET assemblies (amqmdnet.dll) from the primary installation. If there is no primary installation, or you do not want to use the primary installation assemblies, you must update the application configuration file, or the DEVPATH environment variable.

In this scenario, application can't use the amqmdnet.dll in GAC since the version in which it was compiled is higher than the one in GAC. .NET runtime will then try to locate the MQ .NET assembly through probing, if .NET runtime is not able to resolve through probing it fails with System.IO.FileNotFoundException.

In such a case, the application configuration file can be used to help to redirect to the assembly versions of our interest. To redirect one assembly version to another (i.e) from compile time binding reference of the IBM MQ.NET assembly to an earlier version of the assembly, add the <dependentAssembly> property to the application configuration file. The following example snippet in the app.config file redirects an application that was compiled using the MQ Version 8.0.0.4 of the MQ.NET assembly to earlier version of MQ .NET assembly.

A sample application configuration file NonPrimaryRedirect.config is shipped in the folder MQ\_INSTALLATION\_PATH\tools\dotnet\samples\base. This file can be modified or used as reference to redirect the application to use appropriate assembly.

<configuration>

<runtime>

<assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">

<!-- amqmdnet related binding redirect-->

<dependentAssembly>

<assemblyIdentity name="amqmdnet" publicKeyToken="dd3cb1c9aae9ec97" culture="neutral" />

<bindingRedirect oldVersion="1.0.0.3-8.0.0.4" newVersion="8.0.0.0"/>

<publisherPolicy apply="no" />

</dependentAssembly>

</assemblyBinding>

</runtime>

</configuration>

Here in this example,

1. <bindingRedirect> element, redirects the amqmdnet assembly compiled with MQ 8.0.0.4 version to MQ 8.0.0.0 version. Since oldVersion is given within a range, ideally, it means any version from 1.0.0.3 to 8.0.0.4 can be redirected to MQ 8.0.0.0.
2. <assemblyIdentity> element and the attribute, mentions the name of the assembly that that will be redirected.
3. <publisherPolicy> element, mentions whether the runtime applies publisher policy. If publisherPolicy is set to no, the policy files will not be applied.
4. This example redirects, only amqmdnet.dll. Similarly binding redirection can be done for other assemblies like amqmdxcs, amqmdnsp, IBM.XMS as per the requirement. Use one dependentAssembly element for each assembly.

Please note that by executing the application in this way, the new functionality introduced in a later version (eg. 8.0.0.4) will not work when executed using the earlier version of MQ .NET assembly (eg. 8.0.0.0).