MS0S – WebSphere MQ Explorer MQSC Editing Plug-in

Version 1.7.0

Jeff Lowrey

IBM

jlowrey@us.ibm.com

1 Introduction

This SupportPac provides a mechanism in the WebSphere MQ Explorer to work with script files written using the MQSC language. Existing MQSC files are automatically discovered in the Eclipse workspace and appear in the MQExplorer Navigator view under a new folder called "MQSC Script Files".

This SupportPac allows you to manage your WebSphere MQ environment with the MQ Script Command language (MQSC). All MQSC commands are supported, and any queue manager that the MQ Explorer connects to can be managed with this Support Pac, regardless of platform.

Smart editing features allow for syntax checking, code assistance, and rapid identification and navigation to code problems.

MQSC files can be created, imported, deleted, opened, and run against any active queue manager known to MQ Explorer.

The standard Eclipse Team support can be used to store and retrieve MQSC files from Version Control Systems.

1.1 Changes/enhancements/updates for the 1.7 release

The 1.7 release of MS0S is an upgrade release that provides support for MQ version 8.0 changes to the MQSC scripting language. It also is modified to run in the newer level of Eclipse required for MQ Explorer v8.0.

1.2 Changes/enhancements/updates for the 1.6 release

The 1.6 release of MS0S is an upgrade release that provides support for MQ version 7.5 changes to the MQSC scripting language. No major new functionality to MS0S is introduced at the 1.6 level.

1.3 Changes/enhancements/updates for the 1.5 release

The 1.5 release of MS0S is an upgrade release that provides support for MQ version 7.1, both the MQ Explorer and changes to the MQSC scripting language. MS0S 1.5 provides functions and preferences to view all MQSC Script files in a single folder in the MQExplorer tree, or broken out into subfolders based on the hierarchy of Eclipse projects and folders in the workspace being used. MS0S 1.5 introduces new functions to work with Eclipse projects and folders without changing out of the MQExplorer perspective, includes additional packaging options for simplified installation and bug fixes against the 1.2 release.

1.4 Changes/enhancements/updates for the 1.2 release

The 1.2 release of MS0S is an upgrade release that provides support for MQ Version 7 Explorer. It also includes enhanced functionality for running scripts against multiple queue managers and for capturing script results into text files. There are bug fixes from the 1.1 release, and the 1.2 release no longer forces the user into the Resource Perspective in the Eclipse workbench.

1.5 Changes/enhancements/updates for the 1.1 release

The 1.1 release of MS0S is a maintenance release, consisting of performance enhancements, bug fixes, and very minor usability tweaks. The performance enhancements and bug fixes are significant, and all users of 1.0 should update to this version.

In addition, the 1.1 release will allow a base install in any MQ Explorer, regardless of what national language it is running under – although no national language features or translations are included with this package.

2 Installation

This SupportPac ships in multiple package formats to provide a simplified install process. You can unpack the Dropin zip file into the dropins folder in the MQExplorer install, or you can download the Eclipse Update Site and use that to install into into any Eclipse runtime that has the WebSphere MQ Explorer installed.

2.1 IMPORTANT NOTE

MQ Explorer has significantly reduced the size of its install in newer versions, in part by significantly reducing the number and amount of Eclipse packages that it includes in its default install.

You may wish to follow the instructions given for Installing into Eclipse Environments at http://publib.boulder.ibm.com/infocenter/wmqv7/v7r1/topic/com.ibm.mq.explorer.doc/e_install_in_eclipse.htm to install MQ Explorer into a standalone full Eclipse release. You would then follow the instructions in this documentation to install MS0S into that Eclipse package rather than into the MQExplorer default install.

2.2 Updating to 1.7

MS0S 1.7 requires MQExplorer version 8.0, which uses a newer level of Eclipse than comes with MQExplorer v7.1 or 7.5. You cannot run MS0S 1.6 or earlier in the level of Eclipse that comes with MQExplorer v8.0. You cannot run MS0S 1.7 in earlier levels of Eclipse or MQExplorer.

If you coexist MQ 8.0 with MQ 7.1 or 7.5, you will have to install MS0S separately as a new install into MQ Explorer 8.0.

After installing MS0S 1.7, you may need to run strmqmcfg -i followed by strmqcfg -c.
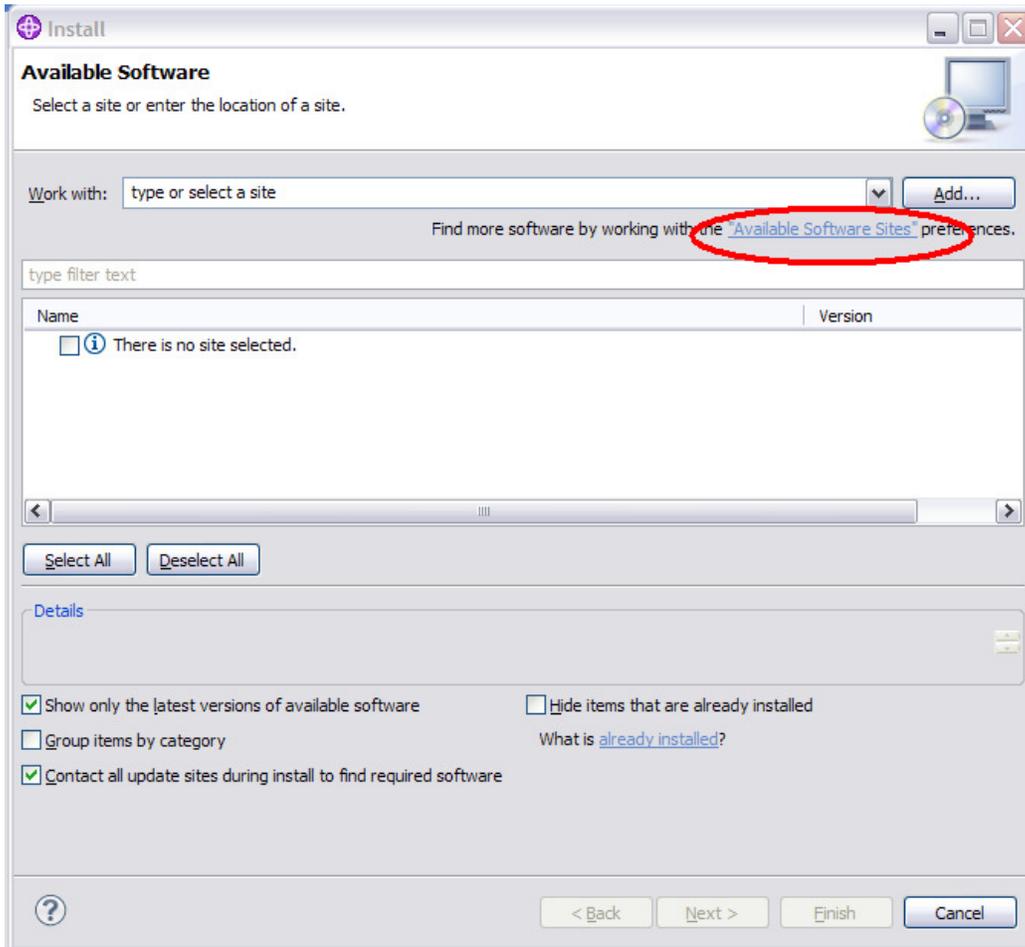
2.3 Installing the dropin version

Download the zip file labeled 'ms0s1.7.0.0-DropInSite.zip'.
Unzip this file into a new folder in <MQ Install>\MQExplorer\eclipse\dropins. Restart MQExplorer.
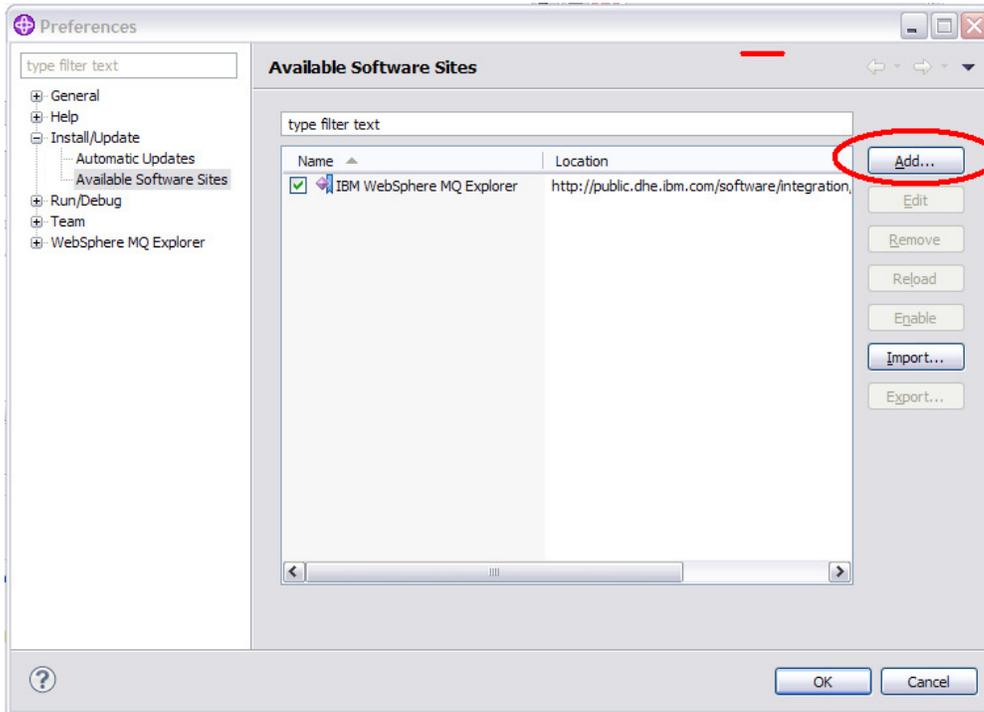
2.4 Installing the Update Site version

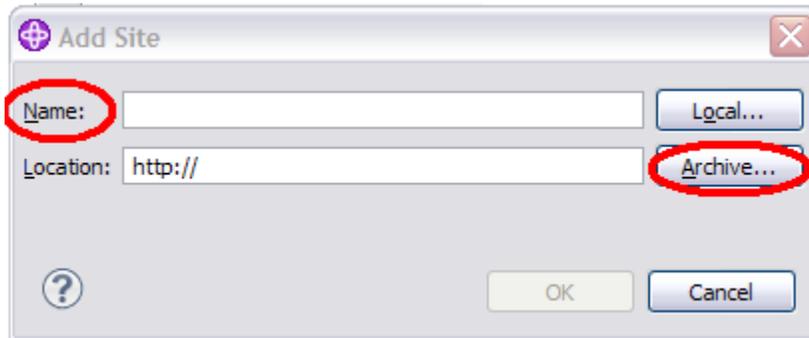Download the zip file named ms0s1.7.0.0-UpdateSite.zip.

Launch MQExplorer.
1) From the Help Menu, choose "Install New Software…"
2) Select "Available Software sites…"
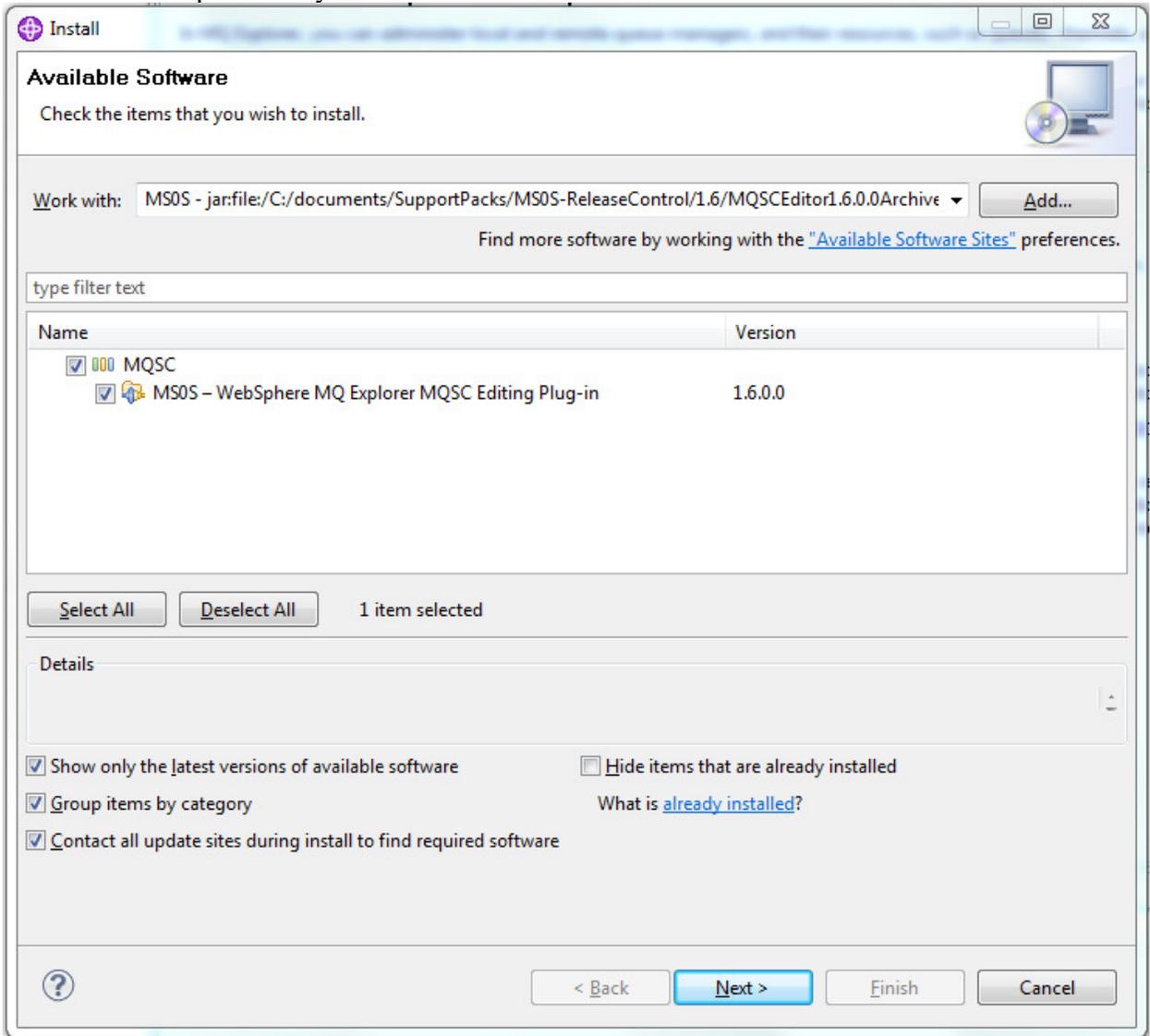
3) Chose the 'Add" button to add a new site

4) Enter a Name and then choose 'Archive..'

5) Navigate to the ms0s1.7.0.0-UpdateSite.zip you downloaded and give the update site a name of your choosing.

6) Select the new update site you created and select MS0S.

7) Click Next… and Next… and accept the license agreements.
8) Restart MQExplorer when prompted.

2.4.1 Verify the Installation

After WebSphere MQ Explorer restarts, if the installation is successful, there will be only one visible change. There will now be an additional folder in the WebSphere MQ Explorer Navigator view, called **MQSC Script Files**
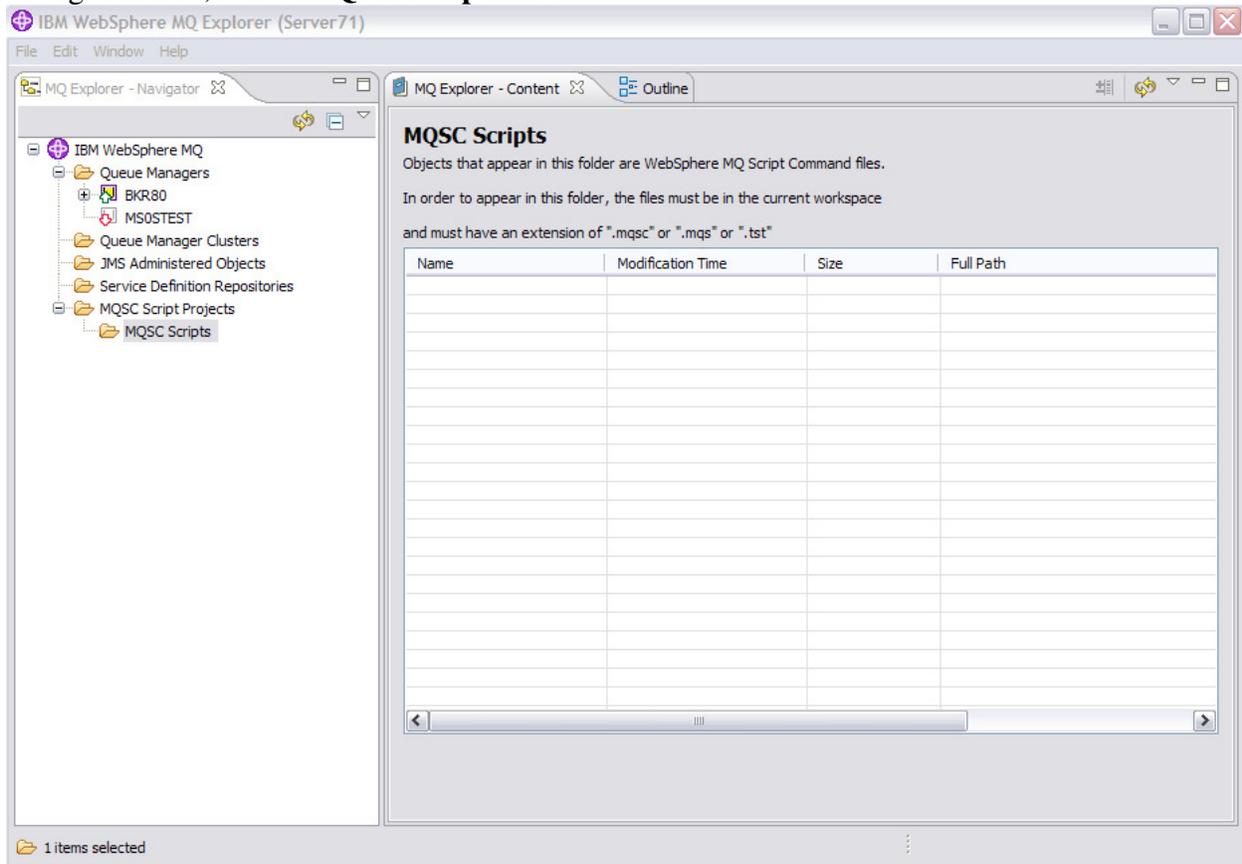
A second method of verification is to confirm that there is an entry in Preferences for **WebSphere MQ MQSC Script Editor.**
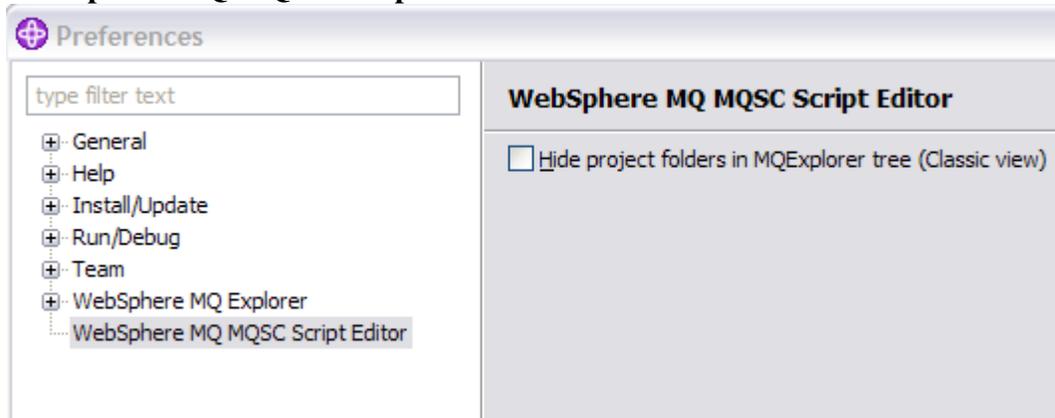
3 Working with Script Files

The five basic options that this SupportPac provides are to create, import, delete, edit, and run MQSC Script files. MQSC Script files are text files that contain commands in the MQSC language.   As of version 1.5 and later, this SupportPac also provides access to basic functions to create and delete Eclipse Projects and folders, to allow for a more granular level of organization of script files.

MQSC Script Files are identified in the WebSphere MQ Explorer and Eclipse using file extensions of ".mqsc", ".mqs" or ".tst". The ".tst" file extension is for historical purposes only, and should not be used for new script files. This SupportPac will not recognize or be able to work with any files, regardless of content or location, that do not have one of those three extensions.

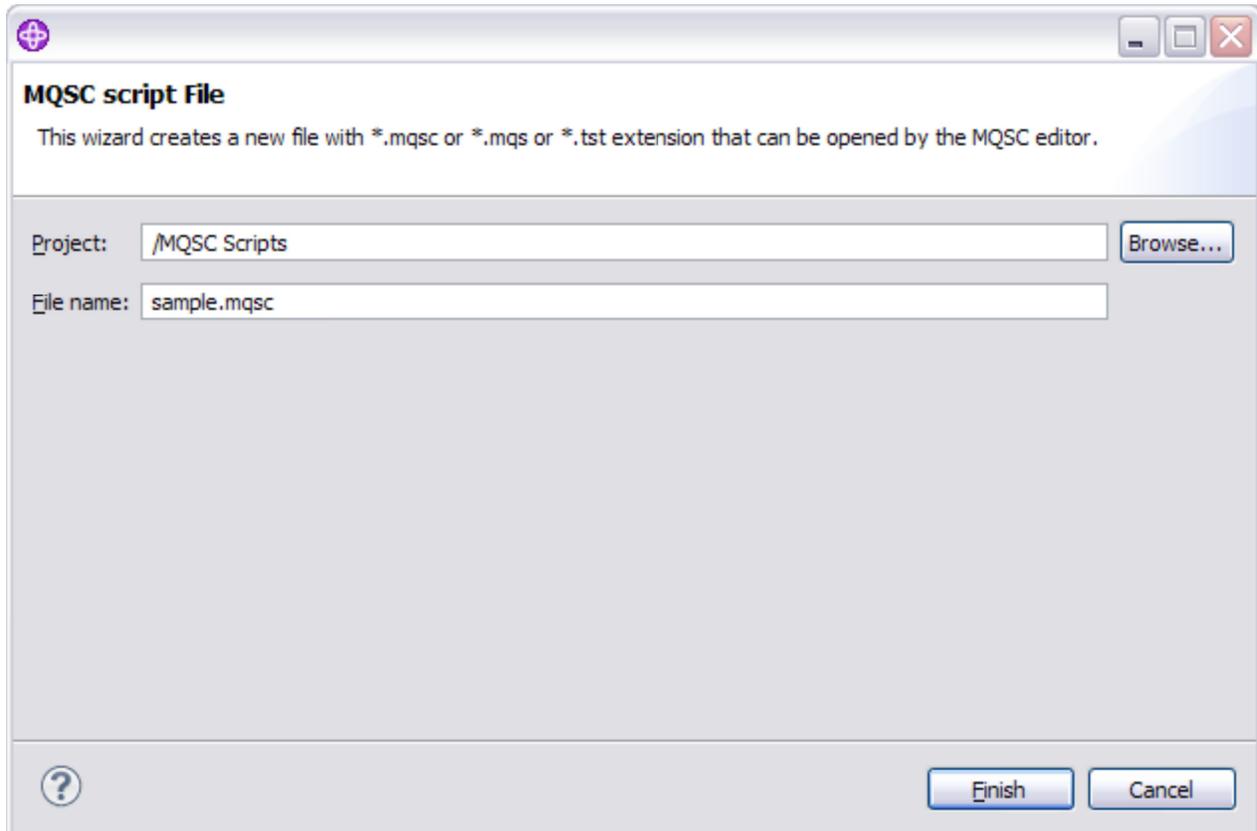## 3.1 Creating a New MQSC Script File

To create a new MQSC Script file, first switch to the WebSphere MQ Explorer perspective.

Right-click on the **MQSC Script Files** tree item in the Explorer Navigator perspective.
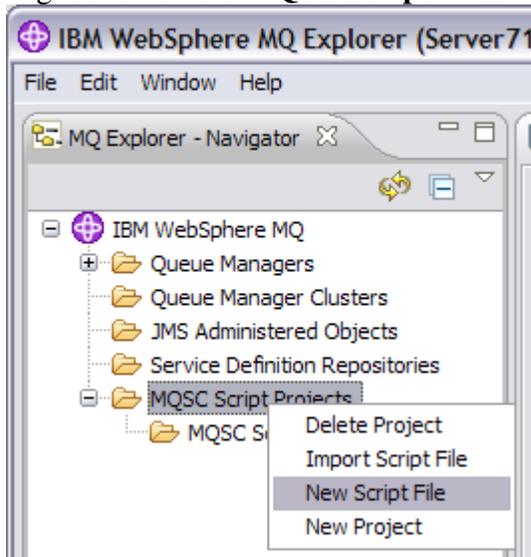
Enter a file name, like **sample.mqsc** into the New File dialog

In this dialog box, you can also choose which Eclipse project will receive the new file. You can not create a new Eclipse Project from this dialog box, you must do that separately. You can only create new MQSC Script Files in existing Eclipse Projects. The project "MQSC Scripts" is created for you, as a convenience. It is a good idea to organize your MQSC scripts into separate projects for each queue manager or application or type of change or etc, depending on the nature of the files. Each file will show up in the Navigator view, regardless of which project they belong to in the workspace.

Click **Finish** to create the file, and open it. Note that the Resource Perspective is *not* opened as of the 1.2 release of MS0S. You can still manually switch to it if you prefer.

See the section on how to **Edit an MQSC Script** for more information on this view and how the MQSC Editor works.

3.2   Importing an existing MQSC Script File

To import a MQSC Script file, first switch to the WebSphere MQ Explorer perspective.

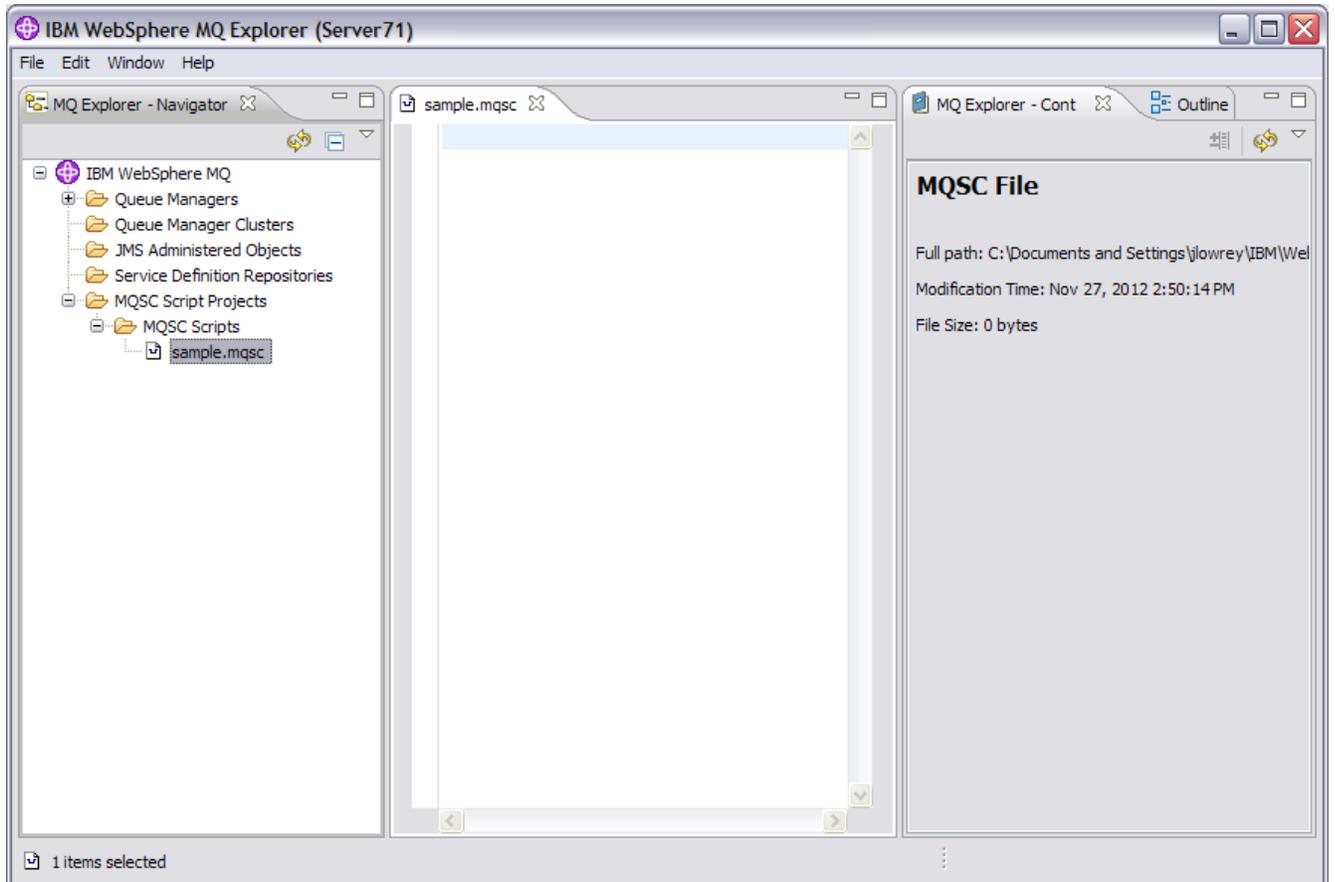Right-click on the **MQSC Script Files** tree item in the Explorer Navigator perspective, and choose **Import Script File**
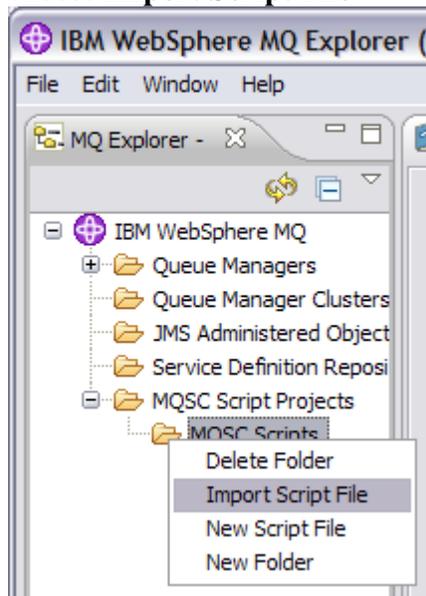
The Import dialog box opens
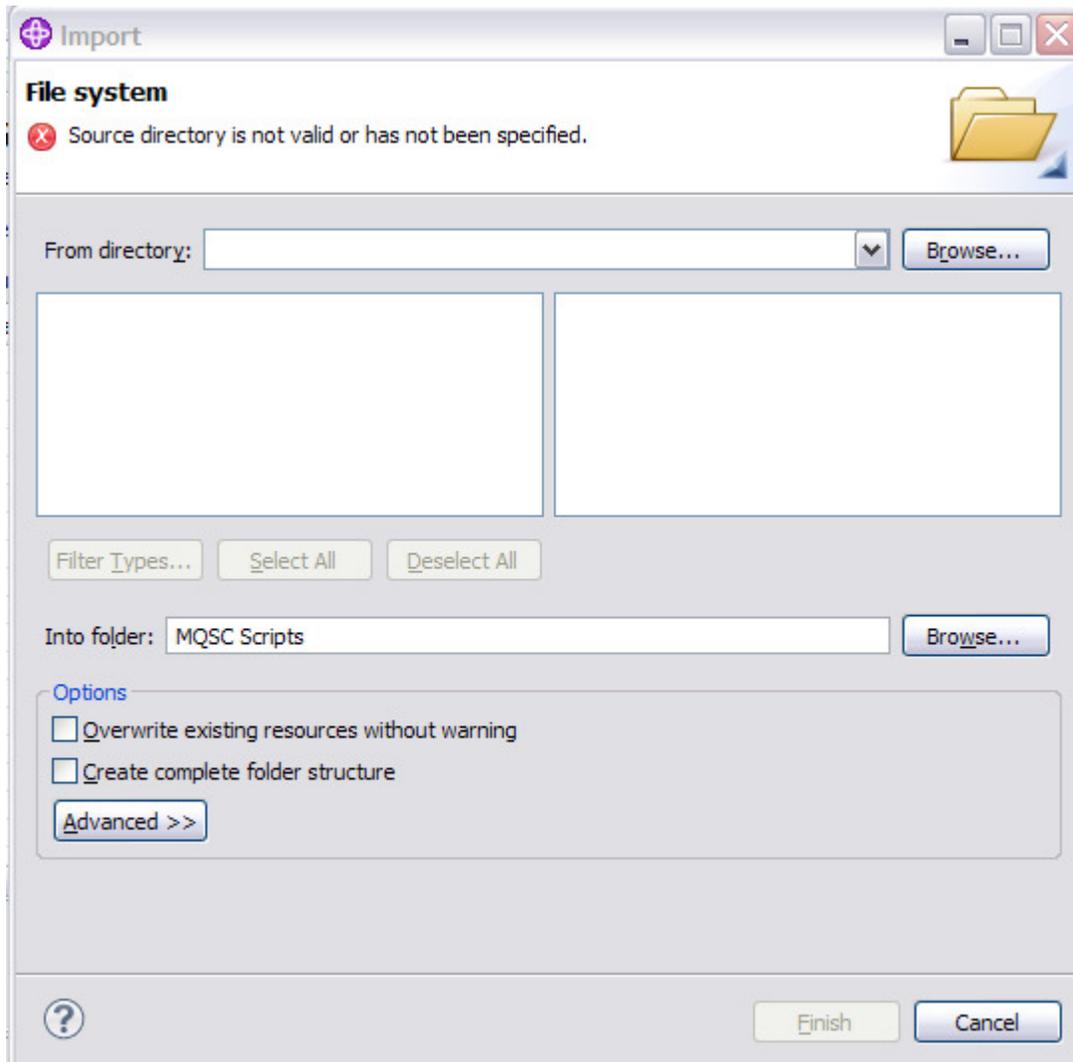
Use the **Browse** button next to **From Directory** to locate the folder that holds the file you want to import.

This will cause a list of the files in that directory to appear in the middle panes.

Put a check mark next to the file or files you wish to Import and click Finish to complete the Import. The file or files will be loaded into the specified project in the workspace and opened.

3.3 Delete a MQSC Script File

To delete a given file from the workspace, you can either delete it from the project in the Resources view, or you can delete the file in the WebSphere MQ Explorer Navigator.
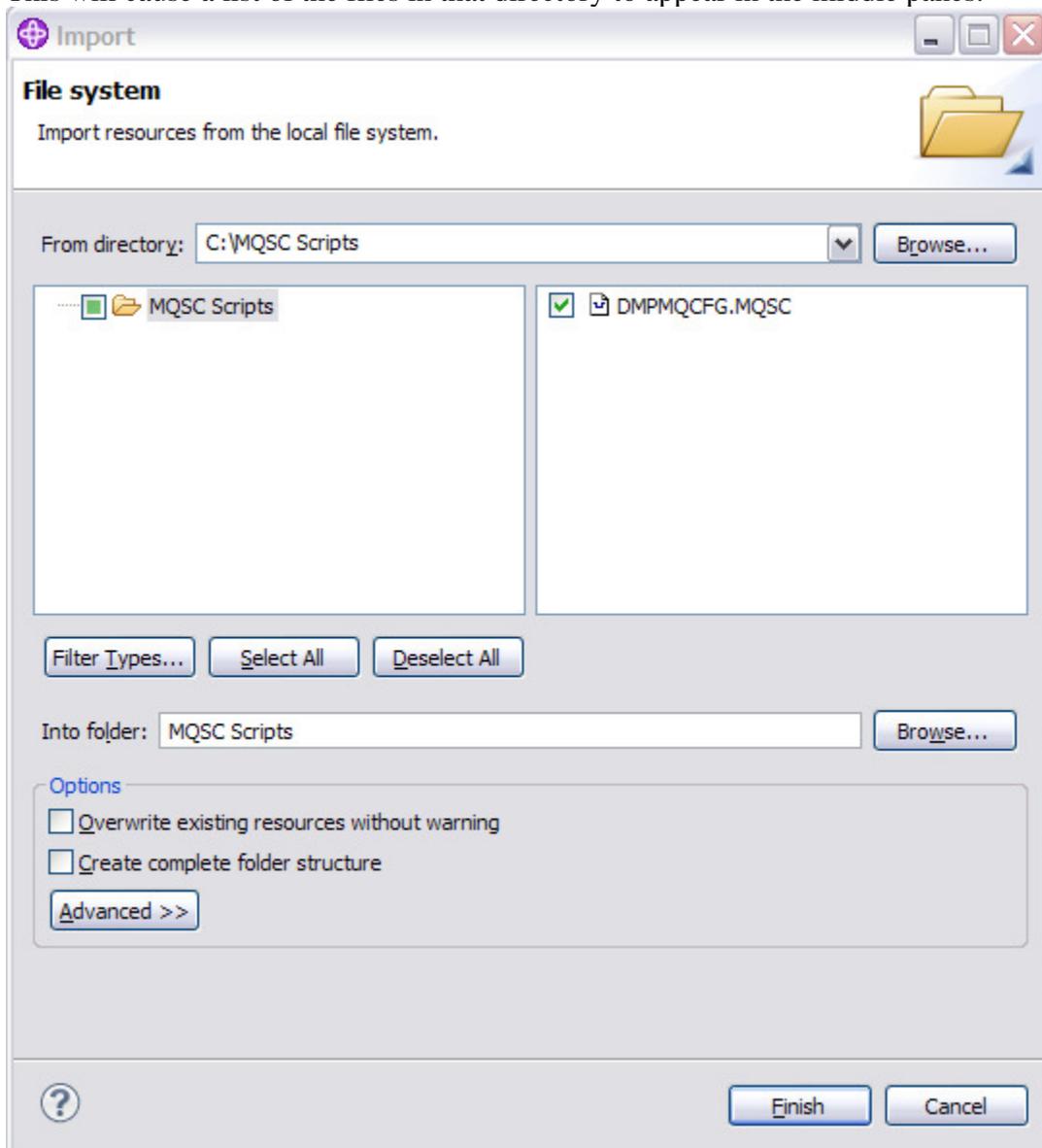
To delete the file using the WebSphere MQ Explorer Navigator right-click on the file you want to delete under the MQSC Scripts folder and choose **Delete Script**
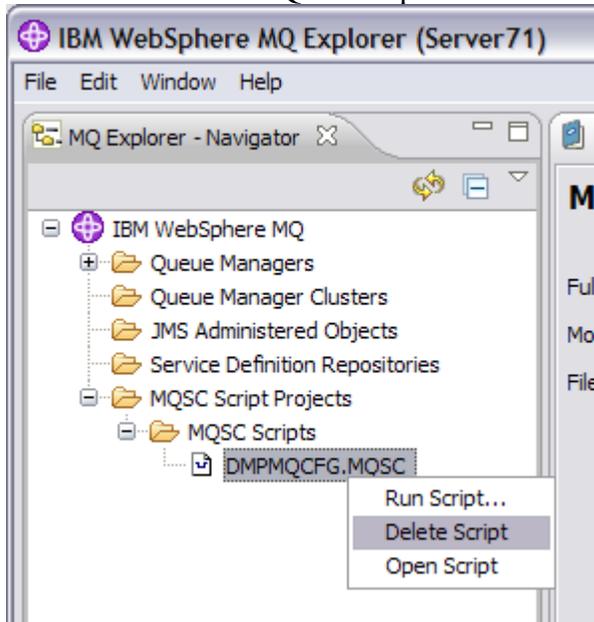
You will then be asked to confirm your delete
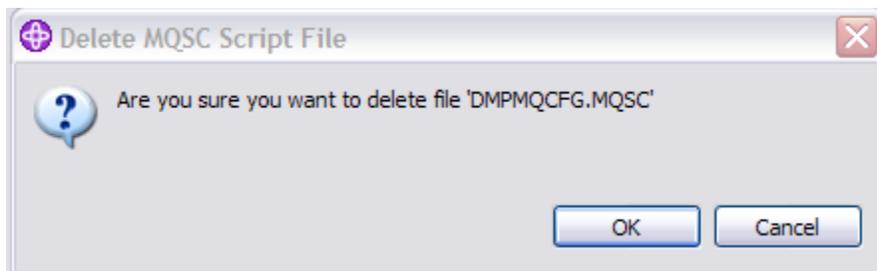
Once you say OK, the file is deleted from the workspace and the file system and deleted from the WebSphere MQ Explorer Navigator View

3.4 Edit a MQSC Script File

3.4.1 Open the File

To edit an MQSC Script file, you can either open it from the Resource Perspective, or you can open the file in the WebSphere MQ Explorer Navigator.

To open the file using the WebSphere MQ Explorer Navigator right-click on the file you want to open under the MQSC Scripts folder and choose **Open Script**
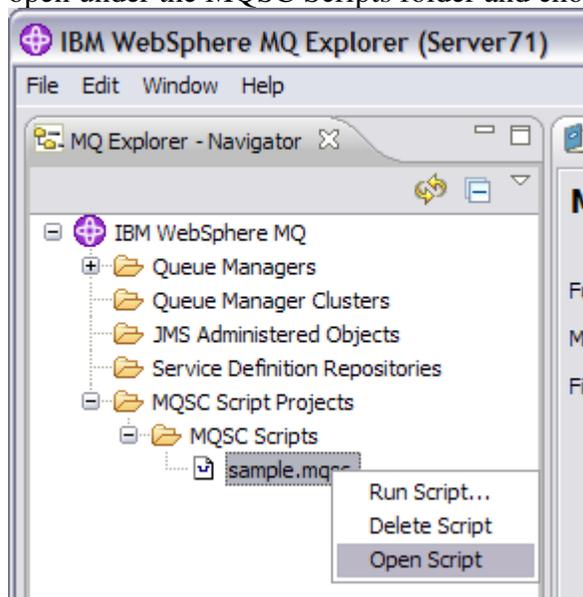
Or you can double-click on the file in the view, and it will open.

The file will be opened and is now ready to be edited.

3.4.2 Parts of the MQSC Editor in the MQ Explorer Perspective
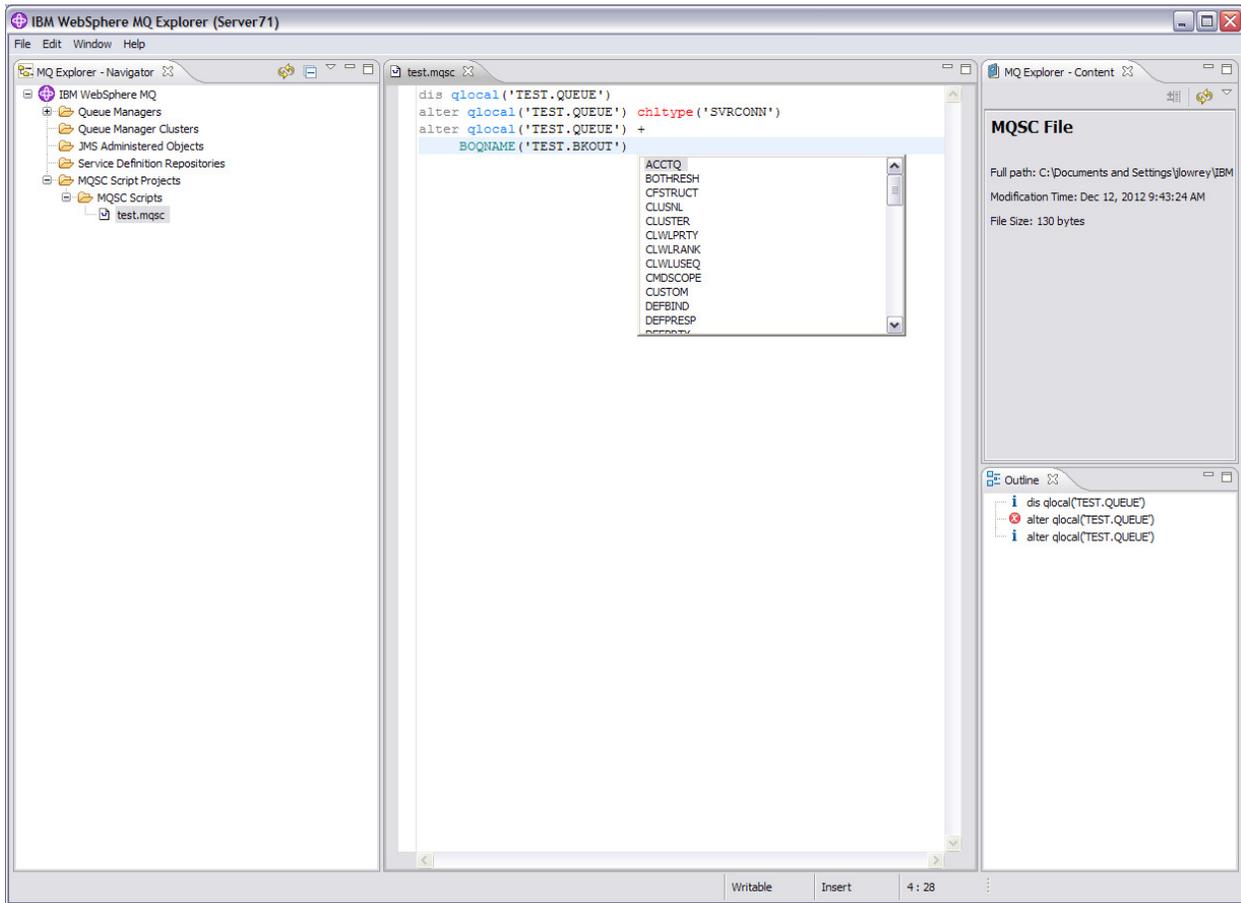
This picture shows the major parts of the MQ Explorer Perspective clockwise from top-left: the MQExplorer Navigator, the Editor window, the MQ Explorer Content view and the Outline View.

The Editor window shows the active content of an MQSC file, with syntax coloring in full effect, and marked problems. It also shows the code completion window active for the third command. Edit the script here as if it was any other Eclipse code.

The Outline View provides a quick overview of all the commands in the file, and allows you to click on an item and navigate to the command in the file.

3.4.3 Editing a Script using Eclipse

The MQSC Editor is built using the Eclipse standard mechanisms for code editing. Commands are typed into the editor window as text, and syntax coloring is applied as you type to provide assistance.

Code assistance is also available through the standard code-completion mechanism. To activate code-completion at the current insertion point, hit Ctrl-Space. This will provide a pop-up menu of suggestions for completing the current word or adding an additional term or parameter or starting an entirely new command. These features allow for rapid development of new MQSC scripts, with minimal typing.

Syntax problems are quickly apparent from syntax coloring. In addition, syntax problems are logged in the Problems view and commands with bad syntax are marked with red X's in the Outline view. The items in both the Problem View and the Outline View can be used to navigate to a particular problem or command by simple clicks of the mouse.

Script files can be imported or exported from Eclipse in the Resource view, using standard Eclipse functions. Files can easily be shared between different WebSphere MQ Explorer installations or workspaces, and MQSC files created by other tools can be edited and executed using this SupportPac.

In addition, files and projects can be managed using Source Control or Version Control Systems using the Eclipse standard Team Sharing mechanisms. The WebSphere MQ Explorer ships with the base Eclipse plug-ins for working with CVS repositories. In addition, any Eclipse plug-ins that are compatible with the 3.3 release of the Eclipse SDK can be installed into the WebSphere MQ Explorer Eclipse, and used.

3.5 Run a MQSC Script File

To run a MQSC Script file, right-click on the file you want to execute and choose **Run Script..**
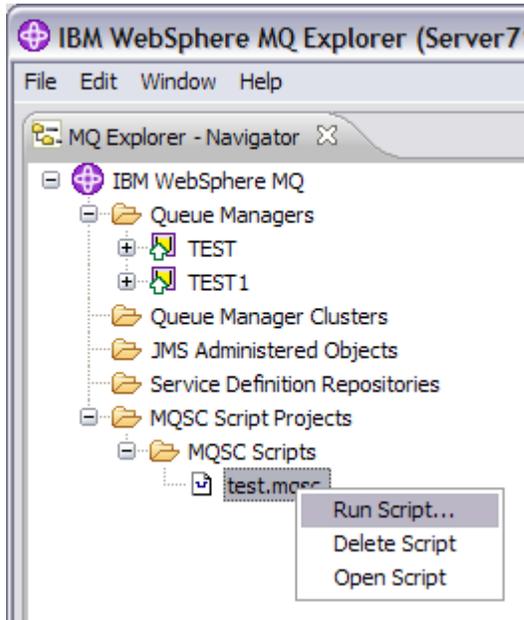
You will be asked to choose the target Queue Manager for the script.
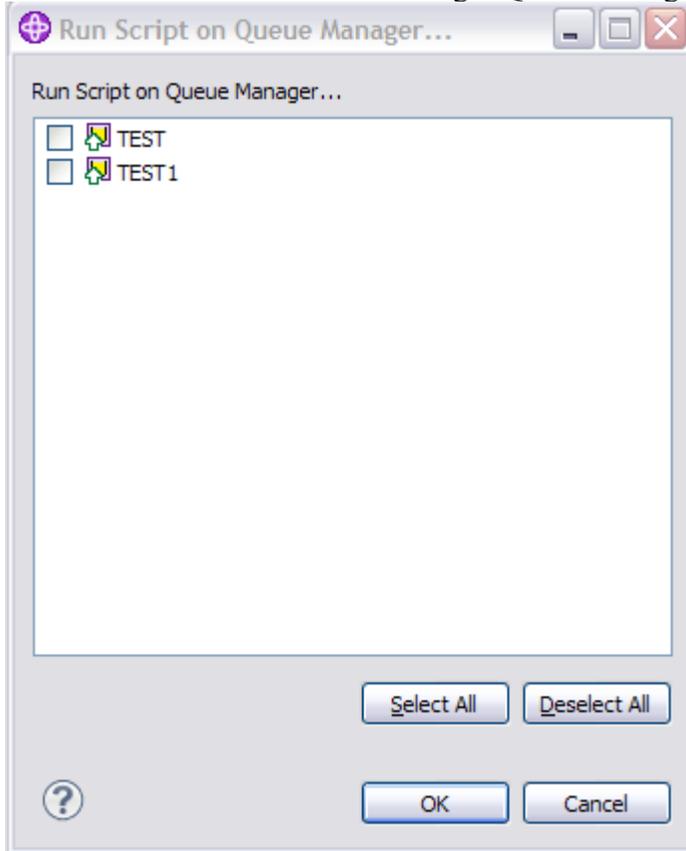
This will only show queue managers that are currently connected in MQ Explorer. All queue managers that MQ Explorer can connect to are supported, including zOS platform queue managers.

Note that in version 1.2 and later of MS0S, you can now run the same script on multiple queue managers with a single step. The script will be run on each queue manager independently in sequence.

Choose OK to start the execution of the script.   If the MQSC Script has any errors in it, then you will get prompted on how to handle this:
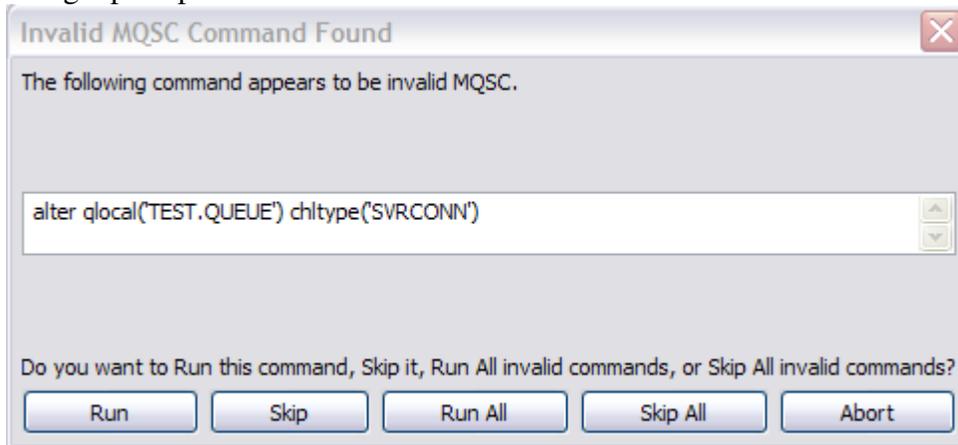
You can choose to run this command anyway, skip it, run all bad commands, skip all bad commands and only run good ones, or stop processing now. The answer to this question is valid only for the current queue manager – so if you are running the script across more than one queue manager at once, you will be asked at least once for each queue manager.

When the script is complete, the results will be presented.


```
Results of running Script test.mqsc on Queue Managers          [Save Results]

[TEST1] [TEST]

AMQ8409: Display Queue details.
   QUEUE(TEST.QUEUE)              TYPE(QLOCAL)
   ACCTQ(QMGR)                    ALTDATE(2012-12-28)
   ALTTIME(10.18.42)             BOQNAME()
   BOTHRESH(0)                    CLUSNL()
   CLUSTER()                     CLWLPRTY(0)
   CLWLRANK(0)                    CLWLUSEQ(QMGR)
   CRDATE(2012-12-28)             CRTIME(10.18.42)
   CURDEPTH(0)                    CUSTOM()
   DEFBIND(OPEN)                  DEFPRTY(0)
   DEFPSIST(NO)                   DEFPRESP(SYNC)
   DEFREADA(NO)                   DEFSOPT(SHARED)
   DEFTYPE(PREDEFINED)            DESCR()
   DISTL(NO)                     GET(ENABLED)
   HARDENBO                       INITQ()
   IPPROCS(0)                    MAXDEPTH(5000)
   MAXMSGL(4194304)              MONQ(QMGR)
   MSGDLVSQ(PRIORITY)             NOTRIGGER
   NPMCLASS(NORMAL)              OPPROCS(0)
   PROCESS()                     PUT(ENABLED)
   PROPCTL(COMPAT)               QDEPTHHI(80)
   QDEPTHLO(20)                  QDPHIEV(DISABLED)
   QDPLOEV(DISABLED)             QDPMAXEV(ENABLED)
   QSVCIEV(NONE)                 QSVCINT(999999999)
   RETINTVL(999999999)           SCOPE(QMGR)
   SHARE                        STATQ(QMGR)
   TRIGDATA()                    TRIGDPTH(1)
   TRIGMPRI(0)                   TRIGTYPE(FIRST)
   USAGE(NORMAL)
 From command dis qlocal('TEST.QUEUE')
AMQ8008: WebSphere MQ queue changed.
 From command alter qlocal('TEST.QUEUE')    BOQNAME('TEST.BKOUT')

                            [OK]
```

The format of the output may differ based on the platform the queue manager is running on. There are now two results shown, one for each queue manager the script was run on. Each of these results can be viewed individually by selecting the tab named after the queue manager.

There is also now a Save Results button at the top of the pane. This button will allow you to save the contents of the results for the selected queue manager into a file in the workspace.
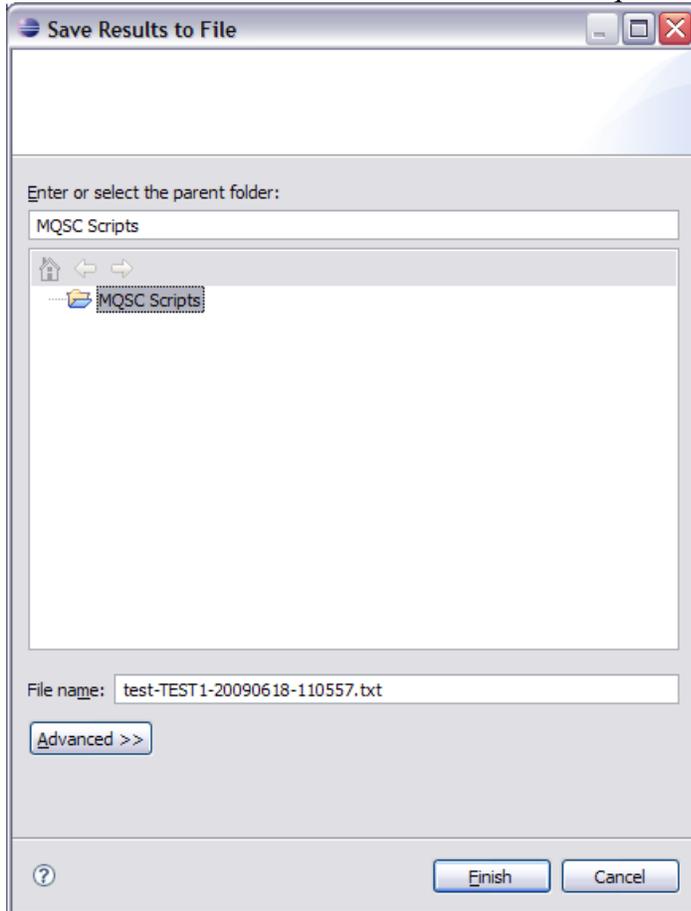
A name built from the name of the script file, the name of the queue manager, and a date and time stamp is selected, with a .txt extension. This name is merely a suggestion and can be changed.

4   Uninstalling

MS0S will be uninstalled using the same mechanism that was used to install it.

If you used the Dropin Site to install MS0S into MQExplorer, then simply remove the MS0S folder from the dropins folder, or move it to another location.

If you used the Archive Site, then you will need to use Eclipse tools to uninstall, as follows:

1. In WebSphere MQ Explorer, click **Help > About IBM WebSphere MQ Explorer**
2. Click on the "Installation Details" Button
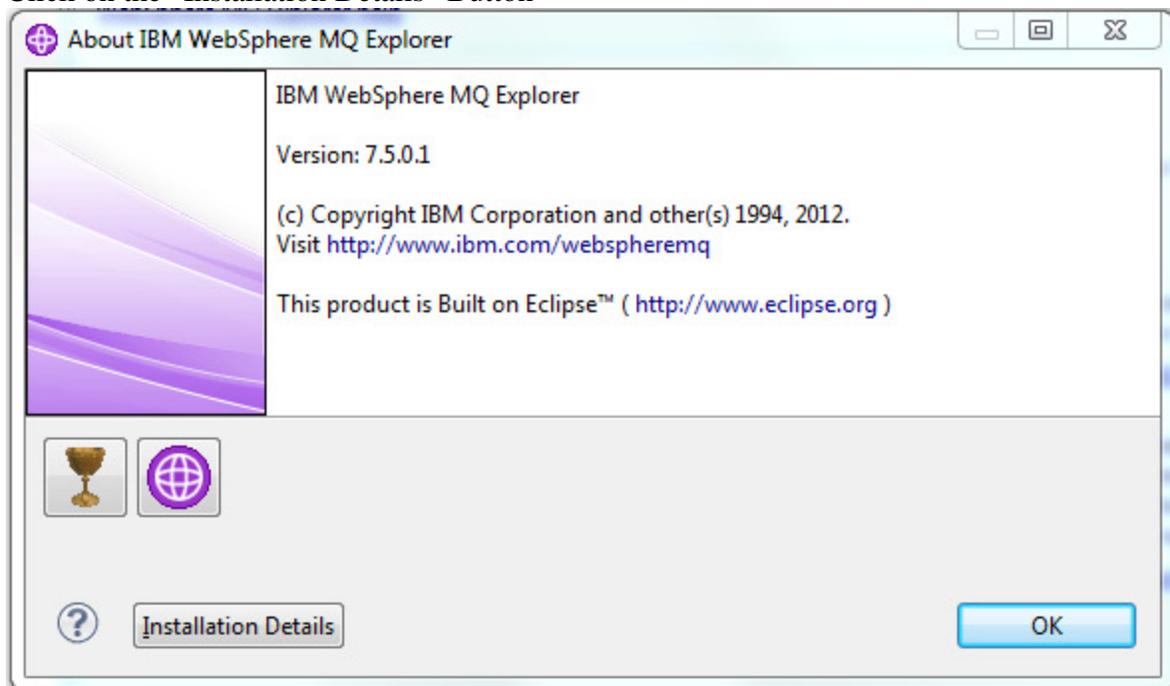
3. The "IBM WebSphere MQ Explorer Installation Details" window will appear

4. Select **MS0S – WebSphere MQ Explorer MQSC Editing Plug-in** (this will start with the name of the update site that you created during install, so it may have a different name). Click on the Uninstall button at the bottom of the page.
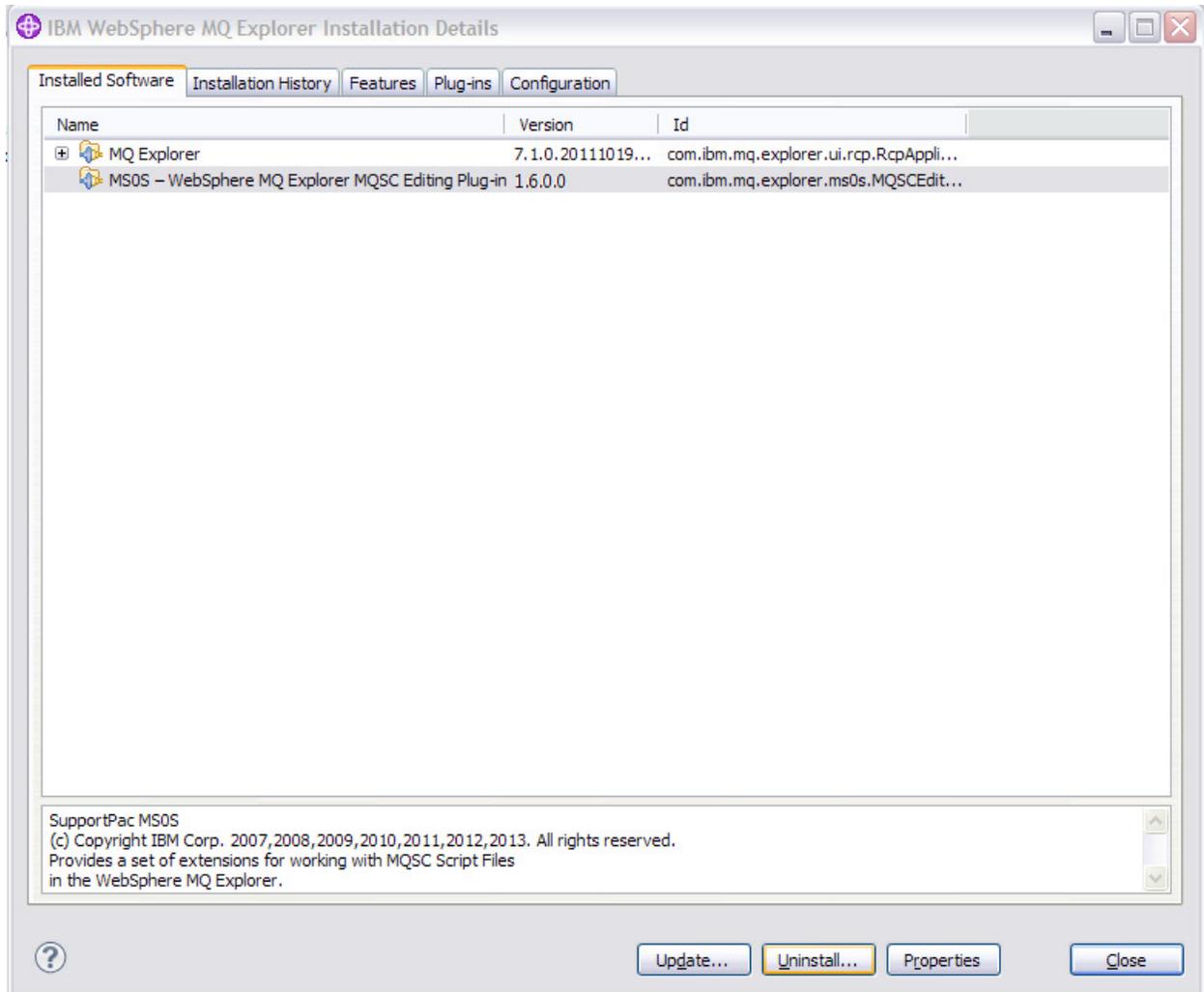
5. Confirm that you want to uninstall by selecting Finish
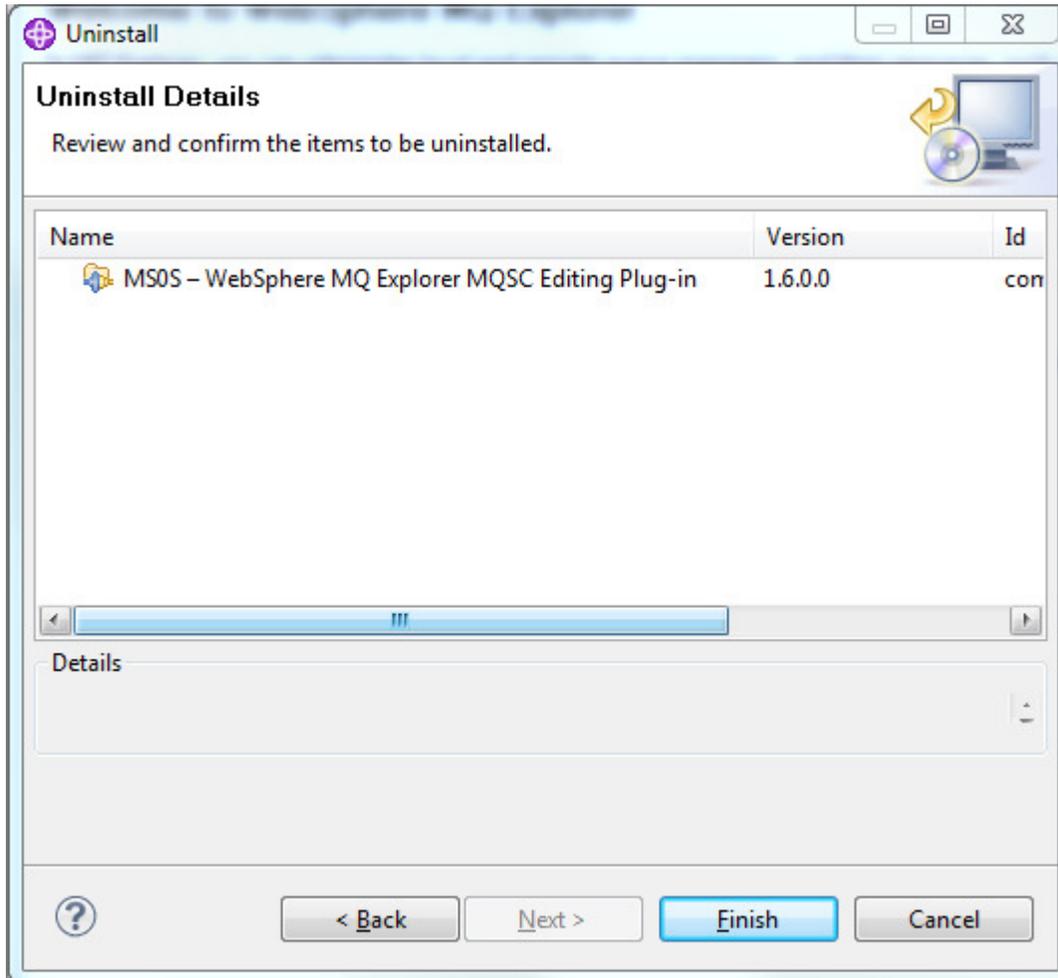
6. Choose "Restart Now" to complete the uninstallation.
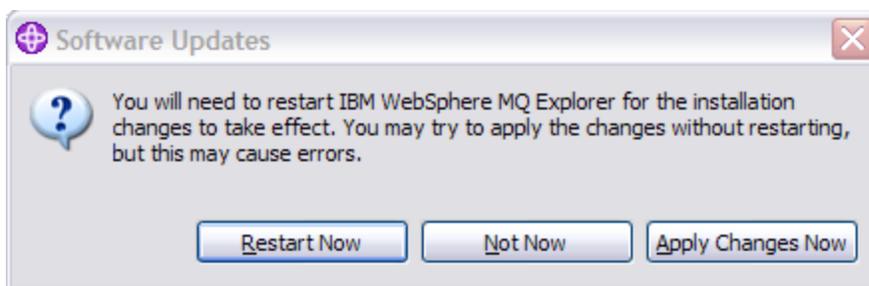
Once the workbench is restarted, MS0S should be uninstalled.