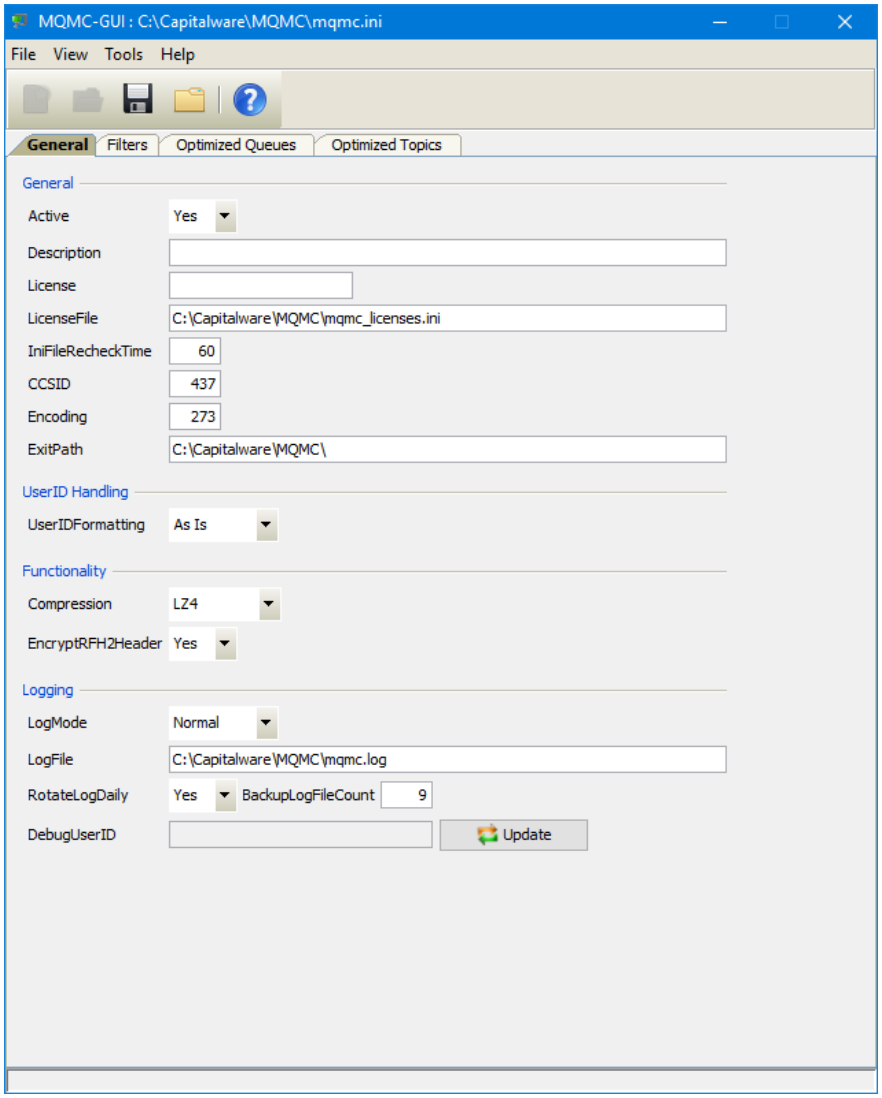
MQMC-GUI User Guide



Capitalware Inc.
Unit 11, 1673 Richmond Street, PMB524
London, Ontario N6G2N3
Canada
sales@capitalware.com
https://www.capitalware.com

Last Updated: November 2020.
© Copyright Capitalware Inc. 2020.

Table of Contents

1 INTRODUCTION....1

1.1 Overview....1

2 INSTALLATION....5

2.1 Prerequisites....5
2.2 Software....5
2.3 Hardware / OS....5
2.4 Setting Up and Running the Application....5
2.4.1 For Windows Users:....5
2.4.2 For Linux and Max OS X (requires JRE/JDK):....6

3 CREATING / UPDATING INIFILES....7

3.1 New IniFile....7
3.2 Open an IniFile....8
3.3 Close an IniFile....9
3.4 Save IniFile....10
3.5 Save As IniFile....10

4 TABS....11

4.1 General Tab....11
4.1.1 General....12
4.1.2 UserId Handling....12
4.1.3 Functionality....13
4.1.4 Logging....14
4.2 Filters Tab....15
4.2.1 Exclude Filters....16
4.3 Optimized Queues Tab....17
4.3.1 Managing Optimized Queues....18
4.4 Optimized Topics Tab....20
4.4.1 Managing Optimized Topics....21

5 OTHER TOOLS....23

5.1 Hex Converter....23
5.2 Hex Editor....24
5.3 Tail File....25
5.4 Text Editor....26
5.5 Visual Difference....28
5.6 XML Editor....30

6 MISCELLANEOUS....31

6.1 Language....31
6.2 Reset View....32
6.2.1 Windows Command Line....32
6.2.2 Linux and macOS Command Line....32

7 APPENDIX A – FREQUENTLY ASKED QUESTIONS (FAQ)....33

8 APPENDIX B – MQMC-GUI UPGRADE PROCEDURES.....34

8.1 Windows Upgrade.....34
8.2 Linux or macOS Upgrade.....34

9 APPENDIX C – SUPPORT.....35

10 APPENDIX D – SUMMARY OF CHANGES.....36

11 APPENDIX E – LICENSE AGREEMENT.....37

12 APPENDIX F – NOTICES.....39

1 Introduction

1.1 Overview

MQMC-GUI application allows the user to create or update MQMC IniFiles. MQMC-GUI can run on any platform that supports Java v1.6 (or higher): Linux, Mac OS X and Windows Vista/7/8/8.1/10.

MQ Message Compression (MQMC) provides compression for MQ message data while it resides in a queue or topic and in the MQ logs (i.e. all data at rest). Data compression is the process of modifying and/or converting the bits structure of data in such a way that it consumes less space in memory and/or on disk.

Question: Would you trade a little CPU time to drastically reduce the disk I/O time?

MQMC has implemented 7 lossless compression algorithms:

- ***LZ4*** is a lossless data compression algorithm that is focused on compression and decompression speed. It belongs to the LZ77 family of byte-oriented compression schemes. LZ4 algorithm is incredibly fast. MQMC implemented LZ4 project.

- ***LZW*** encodes sequences of 8-bit data as fixed-length 12-bit codes. I used Michael Dipperstein's implementation of LZW (Lempel-Ziv-Welch). MQMC implemented LZW from Michael Dipperstein's lzw project.

- LZMA uses a dictionary compression algorithm, whose output is then encoded with a range encoder, using a complex model to make a probability prediction of each bit (Lempel–Ziv–Markov). MQMC implemented LZMA SDK from 7-Zip.

    - ***LZMA_FAST*** uses LZMA SDK with a Level set to 4.
    - ***LZMA_BEST*** uses LZMA SDK with a Level set to 5.

- ***RLE*** (Run Length Encoding) encodes sequences of the same data value occurring over many consecutive data elements are stored as a single data value and count.

- zlib only supports one algorithm, called DEFLATE, which uses a combination of a variation of LZ77 (Lempel–Ziv 1977) and Huffman coding. MQMC implemented zlib from Rich Geldreich's miniz project.

    - ***ZLIB_FAST*** uses zlib with a Level of *Z_BEST_SPEED*.
    - ***ZLIB_BEST*** uses zlib with a Level of *Z_BEST_COMPRESSION*.

Each queue in a queue manager is assigned two buffers to hold messages (one for persistent messages and one for non-persistent messages). The persistent queue buffer size is specified using the tuning parameter ***DefaultPQBufferSize***. The non-persistent queue buffer size is specified using the tuning parameter ***DefaultQBufferSize***.

- DefaultPQBufferSize has a default value of 128KB for 32-bit Queue Managers and 256KB for 64-bit Queue Managers.
- DefaultQBufferSize has a default value of 64KB for 32-bit Queue Managers and 128KB for 64-bit Queue Managers.

Here's the process of the queue manager handling an application putting a message to a queue:

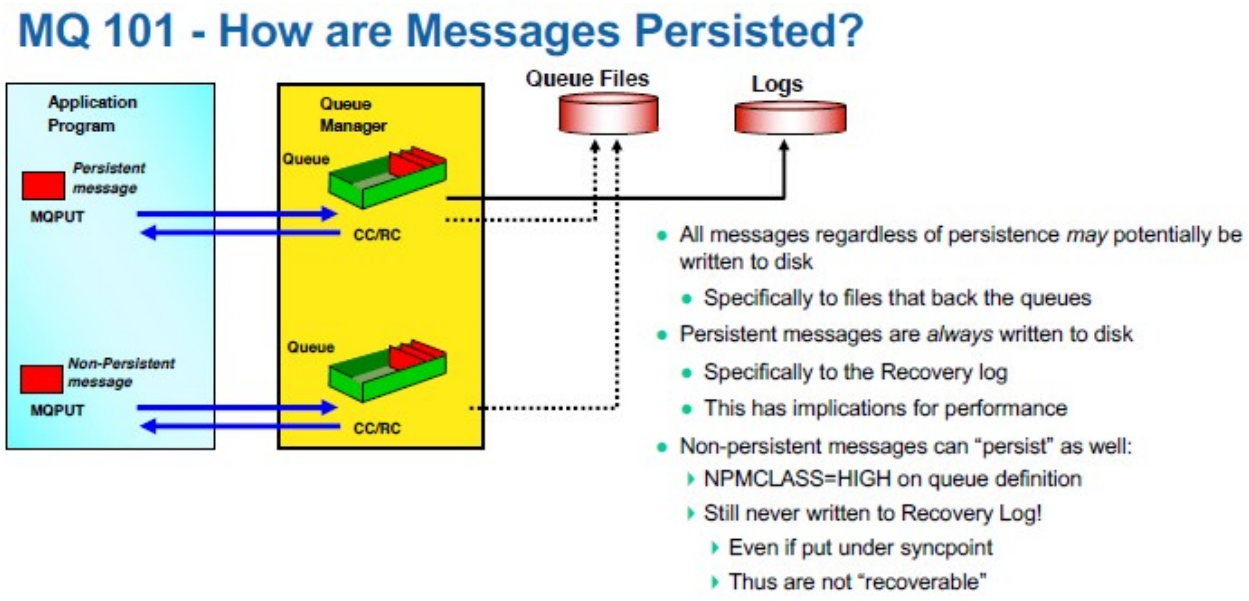
- The message will be put into the buffer of the waiting application if it can fit.
- If that fails, the queue manager tries to write the message to the queue buffer, if it can fit.
- Otherwise, it is written to the queue file.

Here's the process of the queue manager handling an application getting a message from a queue:

- When the consumer (non-waiting) gets a message from a queue, the queue manager will retrieve it from the queue buffer, if available, otherwise from the queue file.
- If the consumer was waiting for a message then the queue manager will attempt to write it directly to the applications buffer.

Persistent messages are always written to the recovery log files.

Here is a screen-shot from Chris Frank's MQ Technical Conference 2016 session called: More Mysteries of the MQ Logger (page 9) that provides a high-level view of disk I/O.



In the picture, the solid line shows the queue manager writing the messages to the recovery log files. The dotted lines means that the message may or may not be written to the queue file. See the above for the scenarios of when/why the queue manager would write a message to the queue file.

The goal of MQMC is to improve the disk I/O (Input/Output) speed by compressing the message data when the queue manager is writing to the queue buffers, queue backing files and the recovery log files.

MQMC is an MQ API Exit that operates with IBM MQ v7.1, v7.5, v8.0, v9.0, v9.1 and v9.2 in Windows, Unix, IBM i (OS/400) and Linux platforms.

On AIX, HP-UX, Linux, Solaris and Windows, MQMC can be configured and used with a non-default installation of MQ in a multi-install MQ environment.

MQMC includes an auxiliary program called: testcmprsn. The testcmprsn program allows the end-user to test various types of messages to see which types of messages would benefit from message compression.

Here's an example of testcmprsn program being run against a 9.17MB XML file (huge file):

```
~/test> ./testcmprsn very_lrg_msg.xml
testcmprsn version 1.0.0.0 (Linux64) {Oct 3 2020}

very_lrg_msg.xml size is 9614354 (9.17MB)
Time taken to perform memcpy() is 4.8770ms

Algorithm               Compressed      Compression     Compression     Decompression
                           Size         Time in ms        Ratio          Time in ms
LZ4                  112253 (109.62KB)       3.4830      85.65 to 1        2.9540
LZMA Fast             32872 (32.10KB)      108.4230     292.48 to 1       11.0730
LZMA Best             27675 (27.03KB)     1152.6960     347.40 to 1       10.6730
LZW                  287184 (280.45KB)     203.0840      33.48 to 1       80.8820
RLE                13213500 (12.60MB)       28.1200       0.73 to 1       26.2680
ZLIB Fast            240612 (234.97KB)      28.3140      39.96 to 1       11.2530
ZLIB Best             83375 (81.42KB)       88.5010     115.31 to 1        8.4590
testcmprsn is ending.
```

2 Installation

This chapter will describe the installation and configuration (optional) of MQMC-GUI. For Windows users, the installation of MQMC-GUI is very simple and straightforward. Other Operating Systems will need to unzip the downloaded software package.

2.1 Prerequisites

This section lists the required hardware and software components needed to run the MQMC-GUI application.

2.2 Software

The following third party software components must be installed in order to properly run MQMC-GUI:

- Java Runtime Environment (JRE) 1.6 (or higher) or Java 2 SDK, Standard Edition (J2SE) v1.6 (or higher)

2.3 Hardware / OS

MQMC-GUI is capable of running on the following hardware platforms that support Java v1.6 (or higher):

- Linux
- Mac OS X
- Windows Vista, 7, 8, 8.1 & 10

2.4 Setting Up and Running the Application

The following section provides instructions on installing and running the MQMC-GUI application.

2.4.1 For Windows Users:

- Run the install program called: **mqmcgui-setup-withjre.exe**
- The installer should follow the standard Windows install procedures and provide default values for the users.
- When the install program has installed MQMC-GUI, there will be a newly created folder under **Start** -> All **Programs** called **MQMC-GUI**.
- To run **MQMC-GUI**, select the **MQMC-GUI** icon from within the **MQMC-GUI** folder.

2.4.2 For Linux and Max OS X (requires JRE/JDK):

- MQMC-GUI is provided as a zip file: **mqmcgui.zip**
- To unzip the file, open a shell prompt and execute the following commands:

  unzip mqmcgui.zip

- When the unzip command has finished, there should be a directory tree as follows:

  Capitalware/MQMC-GUI

- Change directory by typing the following:

  cd Capitalware/MQMC-GUI

- Before running MQMC-GUI for the first time, you need to add the execute attribute to the mqmcgui.sh shell scripts. Type the following:

  chmod +x mqmcgui.sh

- To run **MQMC-GUI** from the **Capitalware/MQMC-GUI** directory, then type the following:

  ./mqmcgui.sh

3 Creating / Updating IniFiles

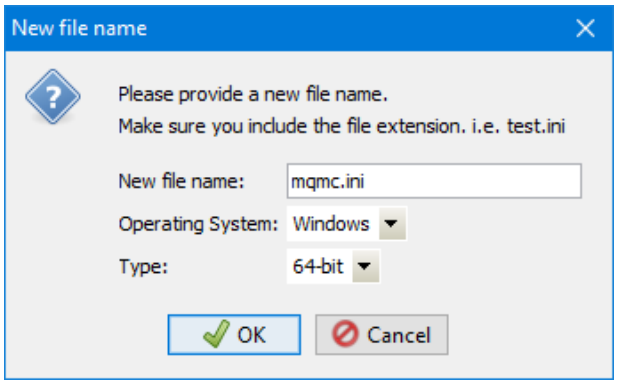
This chapter will describe the how to create and/or update IniFiles.

3.1 New IniFile

This section will describe how to create a new MQMC IniFile. The IniFile can be for a local or remote MQMC implementation.

From the main menu of MQMC-GUI, select **File** then **New File** or click the **New File** button.

The following is the New IniFile panel:



Fill in the following information:

- Input a file name including the file extension of '.ini'
- Select the Operating System
- Choose between a 32-bit or 64-bit environment

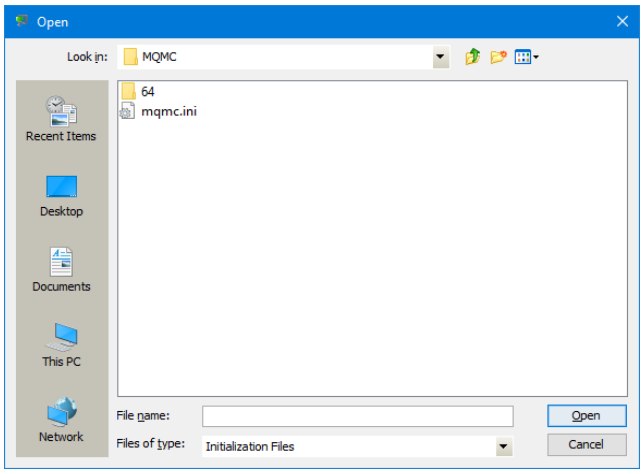
Finally, click the **Ok** button to create and open the new MQMC IniFile.

3.2 Open an IniFile

This section will describe how to open a MQMC IniFile to be viewed or edited. The IniFile can be for a local or remote MQMC implementation.

From the main menu of MQMC-GUI, select **File** then **Open File** or click the **Open File** button.

The following is the **Open** dialog box:



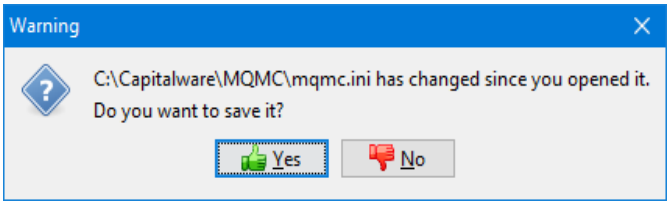
Select the IniFile that you wish to view / edit and then press the **Open** button.

3.3 Close an IniFile

This section will describe how to close a MQMC IniFile.

From the main menu of MQMC-GUI, select **File** then **Close** or click the **Close** button.

If the user has altered the IniFile then the user will be presented with the following dialog box:



Select **Yes** to save the changes otherwise select **No**.

The MQMC-GUI program will make a backup of the original file in the same directory and call it: ***{inifilename}.backup***

3.4 Save IniFile

This section will describe how to save an open MQMC IniFile.

From the main menu of MQMC-GUI, select **File** then **Save** or click the **Save** button.

The parameter information will be saved to the IniFile.

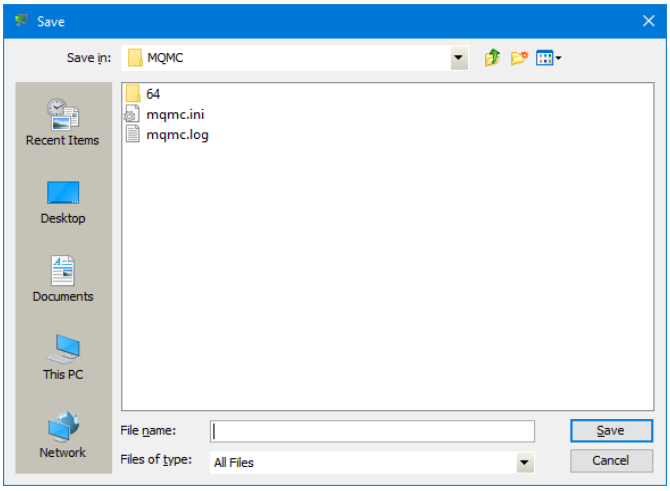
The MQMC-GUI program will make a backup of the original file in the same directory and call it: ***{inifilename}.backup***

3.5 Save As IniFile

This section will describe how to save an open MQMC IniFile.

From the main menu of MQMC-GUI, select **File** then **Save As**

Select or Input the IniFile name that you wish use then press the **Save** button.

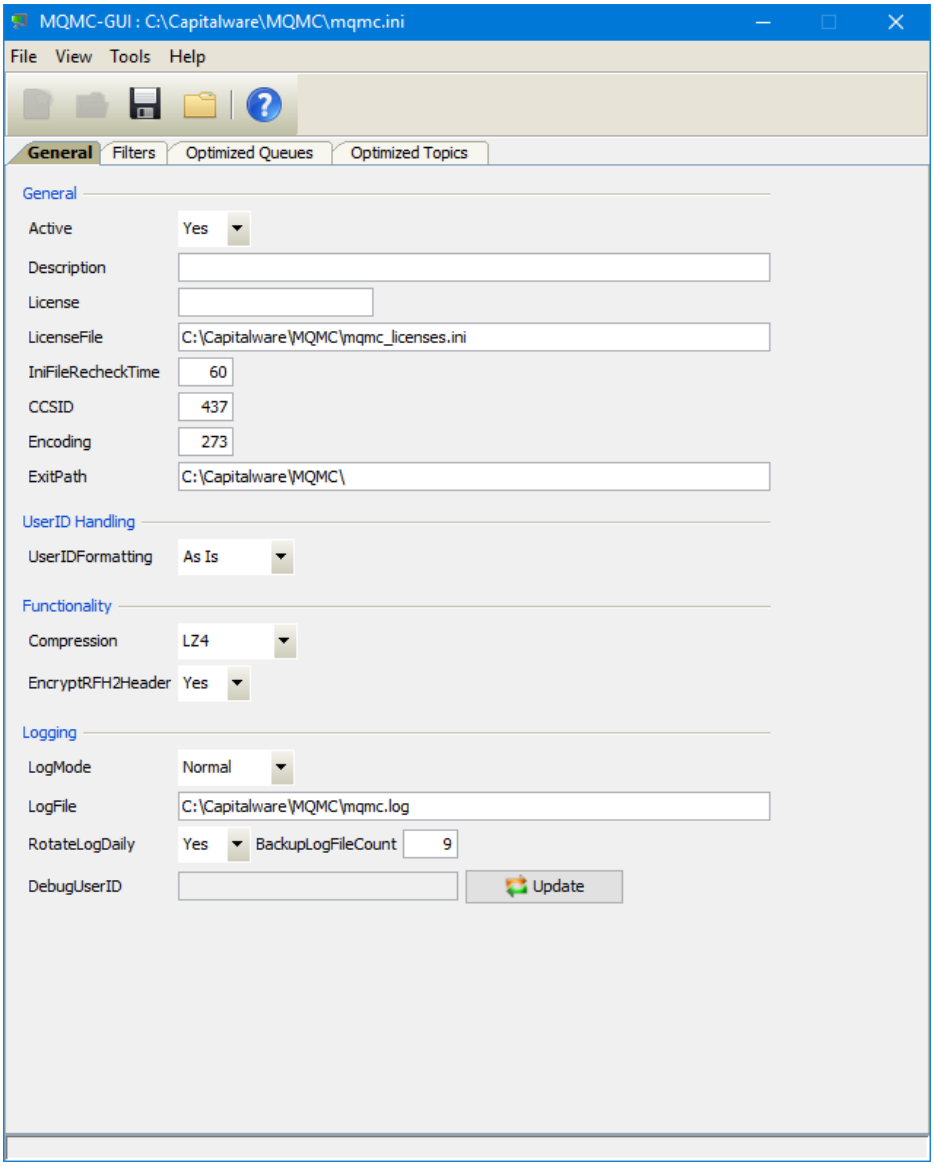


4 Tabs

This chapter will describe the various parameters on each tab.

4.1 General Tab

This section will describe various parameters on the General tab of a MQMC IniFile.
The following is a General Tab with default values:



The IniFile keywords and their values on the General Tab are discussed below.

4.1.1 General

This section describes the general keywords and their values.

➢ **Active** controls enabling or disabling MQMC.

➢ **Description** parameter is optional and is not used by MQMC. It can be used to provide a brief description / purpose of the IniFile.

➢ **License key** is provided by Capitalware Inc. The license is for a particular queue manager. Your license will look something like: 1000-AAAA-BBBBBBBB (Note: This is a sample license only and will NOT work).

➢ **LicenseFile** specifies the location of License file that contains all of the customer's license keys. The following are the default values for LicenseFile:

- For Windows: **C:\Capitalware\MQMC\mqmc_licenses.ini**
- For IBM MQ 32-bit on Unix & Linux: **/var/mqm/exits/mqmc_licenses.ini**
- For IBM MQ 64-bit on Unix and Linux: **/var/mqm/exits64/mqmc_licenses.ini**
- For IBM i: **/QIBM/UserData/mqm/mqmc/mqmc_licenses.ini**

➢ **CCSID** enables monitoring of internal MQ applications issuing MQ API calls.

➢ **Encoding** specifies what type of monitoring the user wishes to have.

➢ **IniFileRecheckTime** specifies the amount, in seconds, before the IniFile is checked whether it has changed or not. The default value is 60.

➢ **ExitPath** explicitly sets the API exit path.

4.1.2 UserId Handling

This section describes the necessary entries on how to handle the incoming UserId.

➢ **UserIDFormatting** supports 3 values [A / U / L]. ('As Is, Uppercase and Lowercase). The default value is A.

4.1.3 Functionality

This section describes the keywords related to functionality.

➢ **Compression** indicates which functionality that MQMC will perform. Compression supports 7 values (LZ4, LZW, LZMA_FAST, LZMA_BEST, RLE, ZLIB_FAST & ZLIB_BEST). The default value is LZ4.

- ***LZ4*** is a lossless data compression algorithm that is focused on compression and decompression speed. It belongs to the LZ77 family of byte-oriented compression schemes. LZ4 algorithm is incredibly fast.
- ***LZW*** (Lempel-Ziv-Welch) encodes sequences of 8-bit data as fixed-length 12-bit codes. I used Michael Dipperstein's implementation of LZW (Lempel-Ziv-Welch).
- ***LZMA_FAST*** uses the LZMA SDK from 7-Zip with a Level set to 4.
- ***LZMA_BEST*** uses the LZMA SDK from 7-Zip with a Level set to 5.
- ***RLE*** (Run Length Encoding) encodes sequences of the same data value occurring over many consecutive data elements are stored as a single data value and count.
- ***ZLIB_FAST*** uses zlib with a Level of *Z_BEST_SPEED*.
- ***ZLIB_BEST*** uses zlib with a Level of *Z_BEST_COMPRESSION*.

Note: LZMA_Fast and LZMA_Best are currently not available for IBM i.

➢ **CompressRFH2Header** specifies that the MQRFH2 header fields and folders are to be compressed when the message payload is compressed. CompressRFH2Header supports 2 values [Y / N]. The default value is Y.

4.1.4 Logging

This section describes the keywords related to logging and their values.

➢ **LogMode** specifies what type of logging the user wishes to have. LogMode supports 4 values: Quiet, Normal, Verbose and Debug. The default value is Normal.

➢ **LogFile** specifies the location of the log file. The default is as follows:

- For Windows: C:\Capitalware\MQMC\mqmc.log
- For IBM MQ 32-bit on Unix and Linux: /var/mqm/exits/mqmc.log
- For IBM MQ 64-bit on Unix and Linux: /var/mqm/exits64/mqmc.log
- For IBM I: /QIBM/UserData/mqm/mqmc/mqmc.log

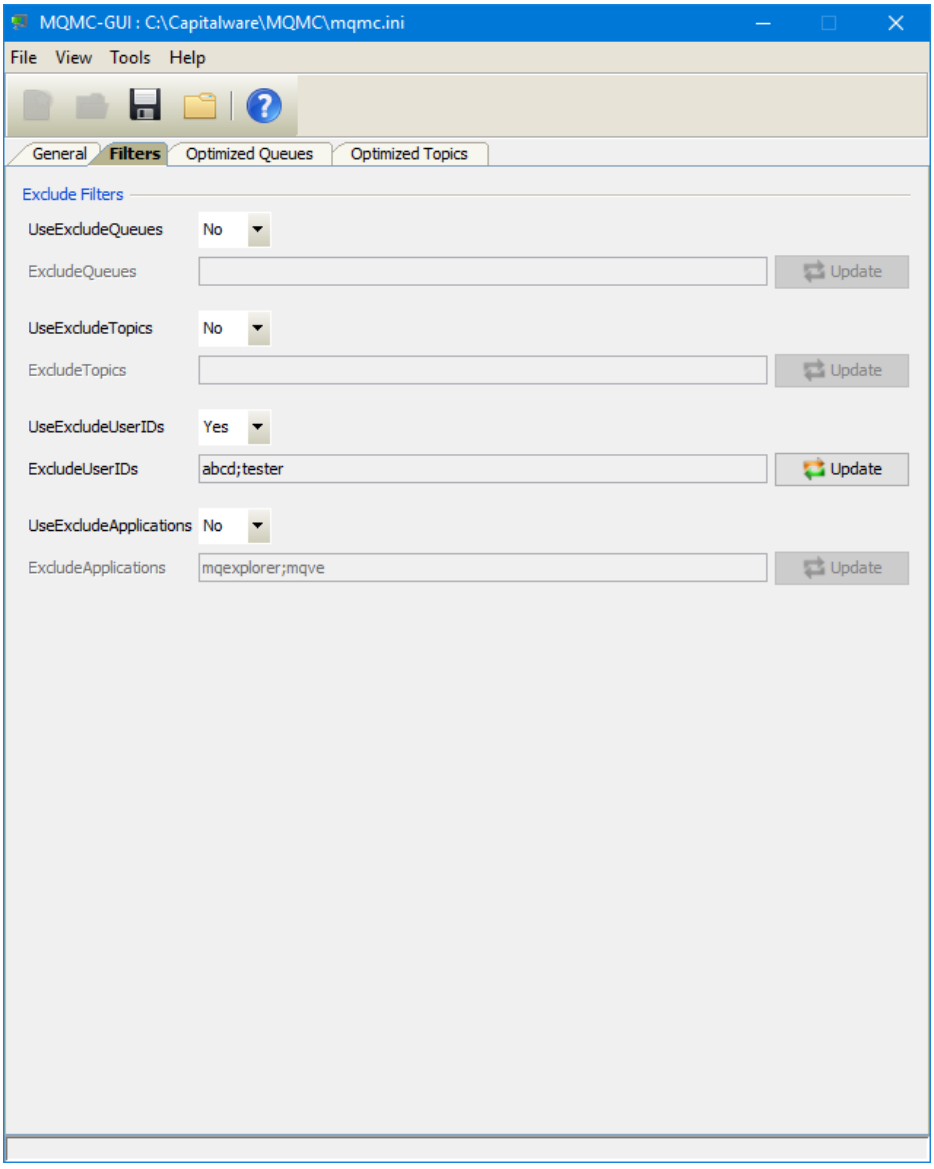
➢ **RotateLogDaily** specifies whether or not MQMC when auto-rotate the log files. The default value is Yes.

➢ **BackupLogFileCount** specifies the number of backup log files to be kept / retained. The default value is 9.

4.2 Filters Tab

This section will describe various parameters on the Filters tab of a MQMC IniFile.

The following is the Filters Tab with sample values:



The following are the IniFile parameters on the Filters Tab:

4.2.1 Exclude Filters

This section describes the Exclude Filter keywords and their values.

- **UseExcludesApplications** allows the user to specifically exclude applications from the compression / decompression processing.

- **ExcludesApplications** explicitly specifies which applications should not be optimized by MQMC.

- **UseExcludesQueues** allows the user to specifically exclude queues from the compression / decompression processing.

- **ExcludesQueues** explicitly specifies which queues should not be optimized by MQMC.

- **UseExcludesTopics** allows the user to specifically exclude topics from the compression / decompression processing.

- **ExcludesTopics** explicitly specifies which topics should not be optimized by MQMC.

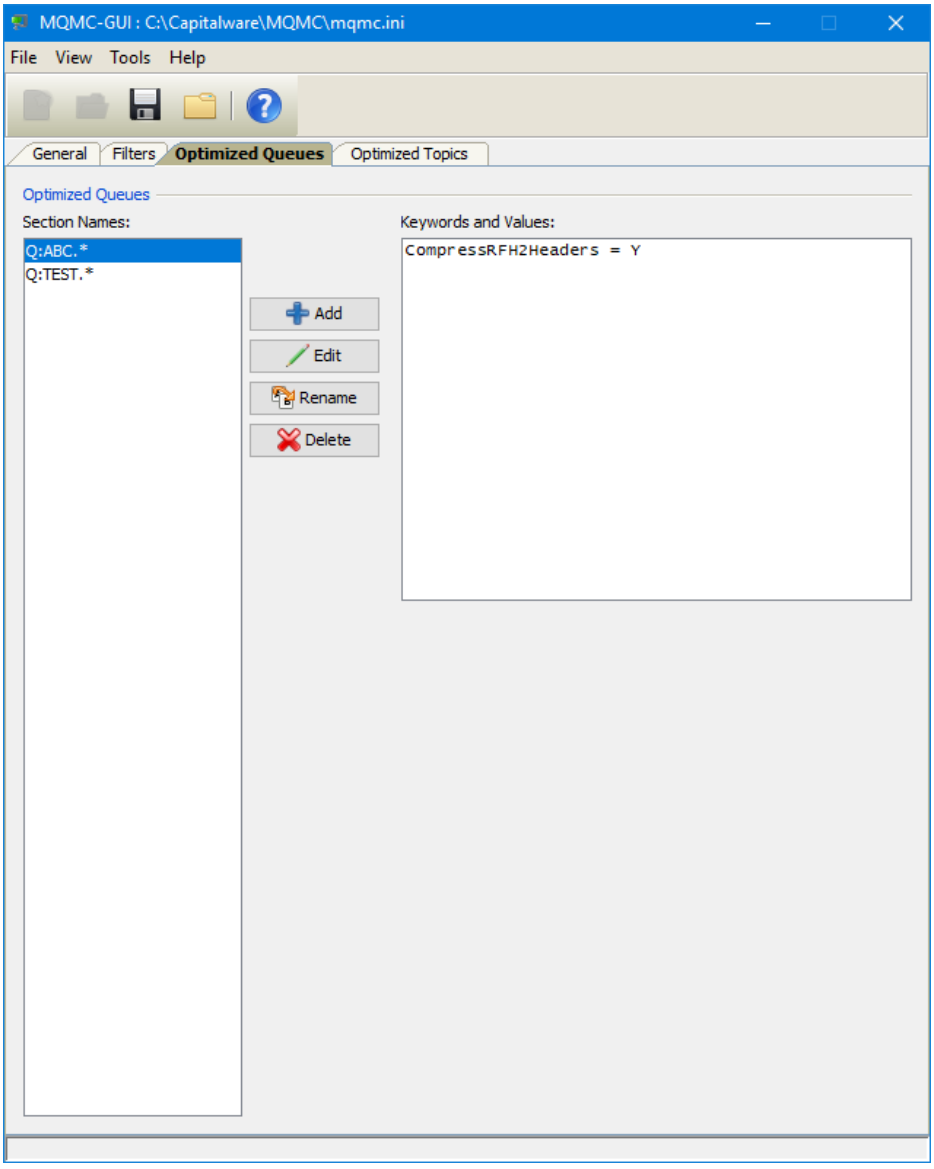
- **UseExcludesUserIDs** allows the user to specifically exclude UserIds from the compression / decompression processing.

- **ExcludesUserIDs** explicitly specifies which UserIds should not be optimized by MQMC.

4.3 Optimized Queues Tab

This section will describe various parameters on the Optimized Queues tab of a MQMC IniFile.

The following is the Optimized Queues Tab with sample values:

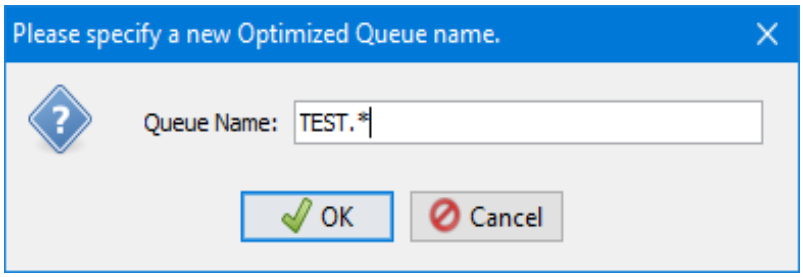


This section describes IniFile keywords that the user will use so that MQMC will perform compression / decompression processing against.

4.3.1 Managing Optimized Queues

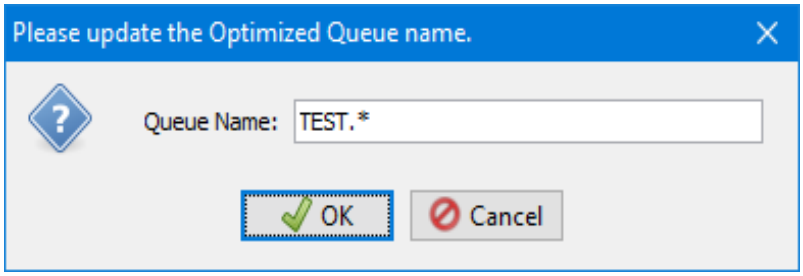
This section describes the necessary steps to manage the optimized queue:

Click the **Add** button to add an entry to the list.



Input the Optimize Queue name including wildcards and then click the **Ok** button.

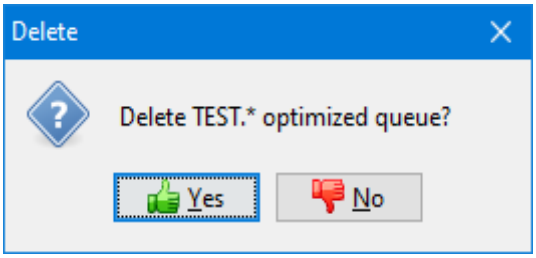
Click the **Rename** button to delete an entry in the list.



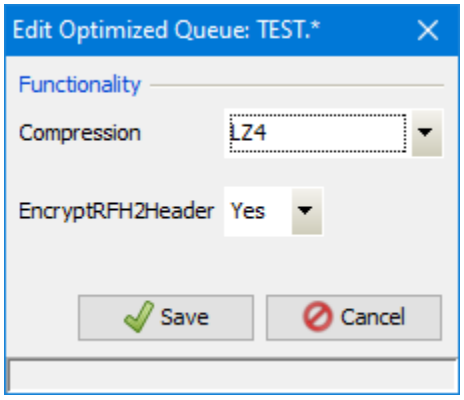
Update the override type and / or name then click the **Ok** button.

Click the **Delete** button to delete an entry in the list.

Click the **Yes** button to permanently remove the entry from the list.



Click the **Edit** button to update an entry in the list.



➢ **Compression** indicates which functionality that MQMC will perform. Compression supports 7 values (LZ4, LZW, LZMA_FAST, LZMA_BEST, RLE, ZLIB_FAST & ZLIB_BEST). The default value is LZ4.

- ***LZ4*** is a lossless data compression algorithm that is focused on compression and decompression speed. It belongs to the LZ77 family of byte-oriented compression schemes. LZ4 algorithm is incredibly fast.
- ***LZW*** (Lempel-Ziv-Welch) encodes sequences of 8-bit data as fixed-length 12-bit codes. I used Michael Dipperstein's implementation of LZW (Lempel-Ziv-Welch).
- ***LZMA_FAST*** uses the LZMA SDK from 7-Zip with a Level set to 4.
- ***LZMA_BEST*** uses the LZMA SDK from 7-Zip with a Level set to 5.
- ***RLE*** (Run Length Encoding) encodes sequences of the same data value occurring over many consecutive data elements are stored as a single data value and count.
- ***ZLIB_FAST*** uses zlib with a Level of *Z_BEST_SPEED*.
- ***ZLIB_BEST*** uses zlib with a Level of *Z_BEST_COMPRESSION*.

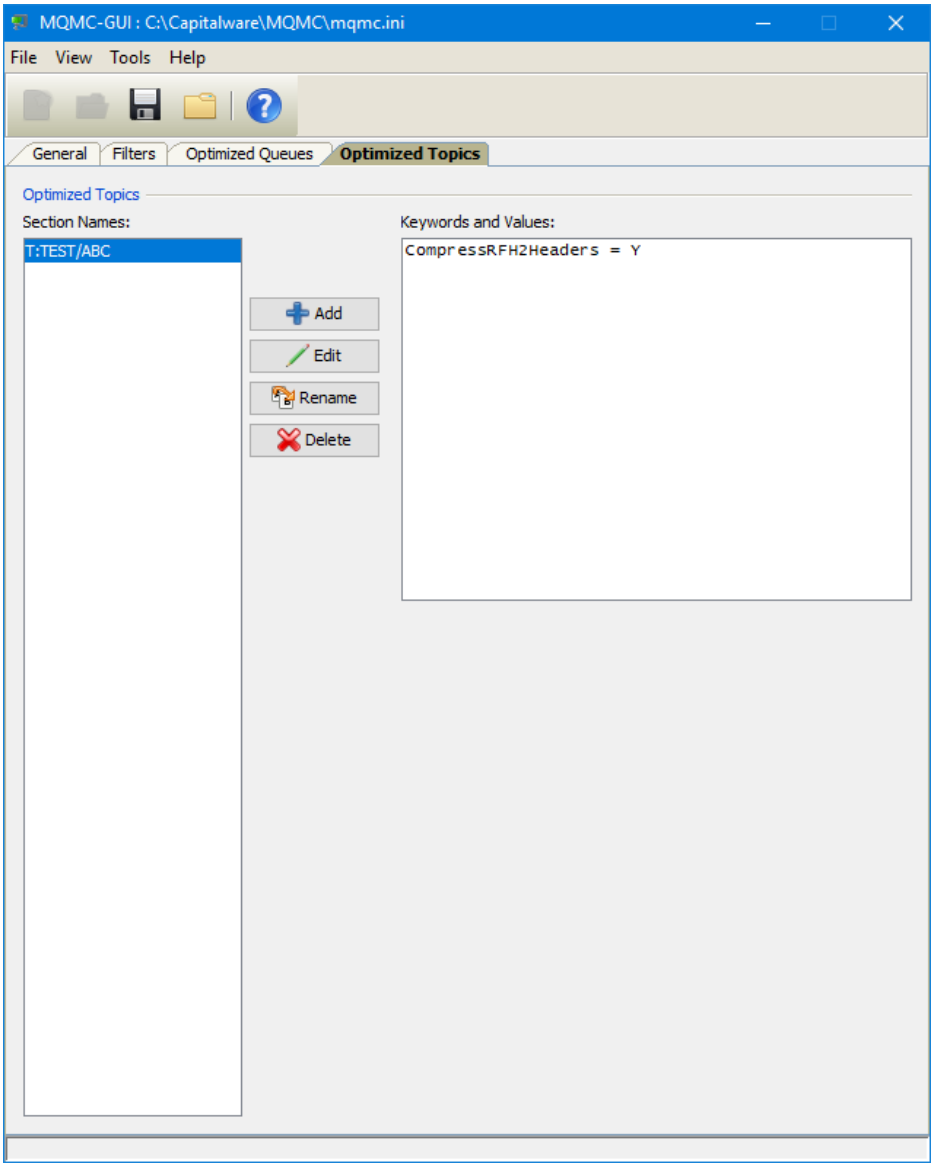
Note: LZMA_Fast and LZMA_Best are currently not available for IBM i.

➢ **CompressRFH2Header** specifies that the MQRFH2 header fields and folders are to be compressed when the message payload is compressed. CompressRFH2Header supports 2 values [Y / N]. The default value is Y.

4.4 Optimized Topics Tab

This section will describe various parameters on the Optimized Topics tab of a MQMC IniFile.

The following is the Optimized Topics Tab with sample values:

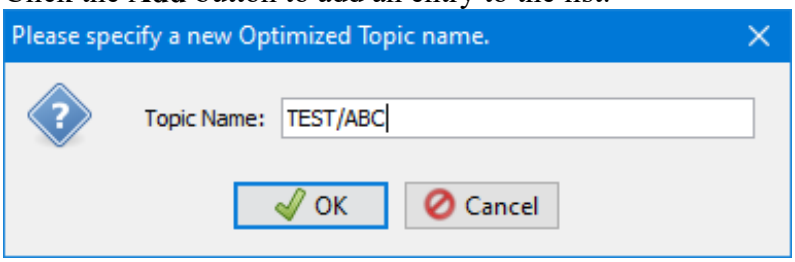


This section describes IniFile keywords that the user will use so that MQMC will perform compression / decompression processing against.

4.4.1 Managing Optimized Topics

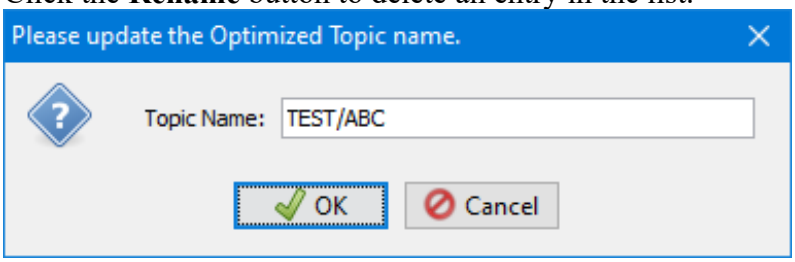
This section describes the necessary steps to manage the optimized topic:

Click the **Add** button to add an entry to the list.



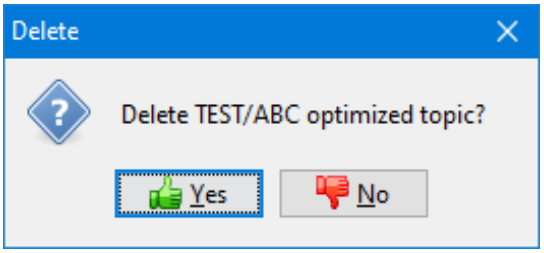
Input the Optimize Topic name including wildcards and then click the **Ok** button.

Click the **Rename** button to delete an entry in the list.



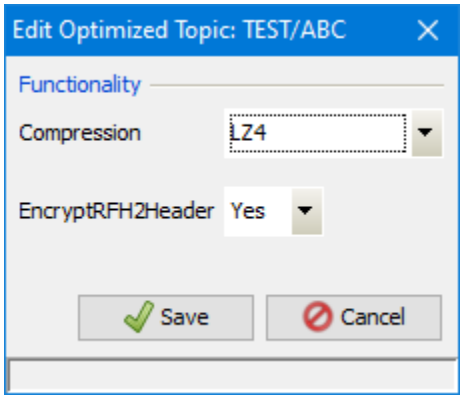
Update the override type and / or name then click the **Ok** button.

Click the **Delete** button to delete an entry in the list.



Click the **Yes** button to permanently remove the entry from the list.

Click the **Edit** button to update an entry in the list.



➢ **Compression** indicates which functionality that MQMC will perform. Compression supports 7 values (LZ4, LZW, LZMA_FAST, LZMA_BEST, RLE, ZLIB_FAST & ZLIB_BEST). The default value is LZ4.

- ***LZ4*** is a lossless data compression algorithm that is focused on compression and decompression speed. It belongs to the LZ77 family of byte-oriented compression schemes. LZ4 algorithm is incredibly fast.
- ***LZW*** (Lempel-Ziv-Welch) encodes sequences of 8-bit data as fixed-length 12-bit codes. I used Michael Dipperstein's implementation of LZW (Lempel-Ziv-Welch).
- ***LZMA_FAST*** uses the LZMA SDK from 7-Zip with a Level set to 4.
- ***LZMA_BEST*** uses the LZMA SDK from 7-Zip with a Level set to 5.
- ***RLE*** (Run Length Encoding) encodes sequences of the same data value occurring over many consecutive data elements are stored as a single data value and count.
- ***ZLIB_FAST*** uses zlib with a Level of *Z_BEST_SPEED*.
- ***ZLIB_BEST*** uses zlib with a Level of *Z_BEST_COMPRESSION*.

Note: LZMA_Fast and LZMA_Best are currently not available for IBM i.

➢ **CompressRFH2Header** specifies that the MQRFH2 header fields and folders are to be compressed when the message payload is compressed. CompressRFH2Header supports 2 values [Y / N]. The default value is Y.

5 Other Tools

This chapter will describes how the other tools included with MQMC-GUI.

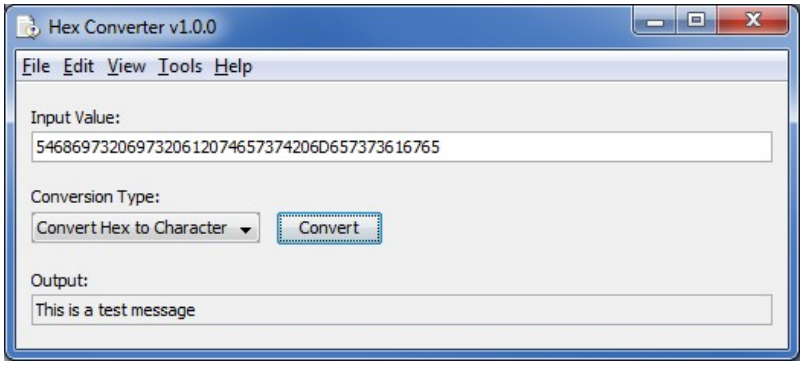
5.1 Hex Converter

This section will describe how to invoke and use the Hex Converter tool. Hex Converter tool can convert a hexadecimal number to its corresponding string representation and vice versa. i.e. 0x'3132333041424344' to '1230ABCD'.

For the Other Tools, you can have unlimited number of them running (up to what your box can support). Also, while any of the Other Tools are running, the user can still use MQMC-GUI.

From the main panel of MQMC-GUI, select **Tools**, then **Hex Converter**.

This is the main panel of the Hex Converter.



- The user first selects the Conversion type: "Hex to Character" or "Character to Hex"
- Input the value hex or character value into the "Input Value" field
- Click the Convert button to convert the data

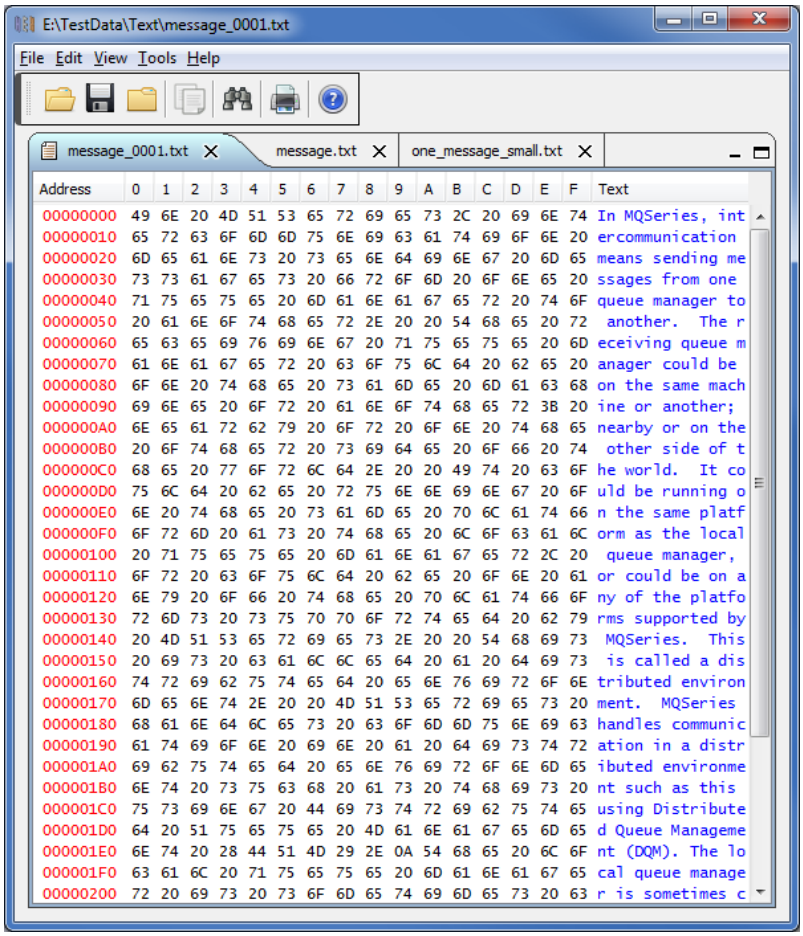
5.2 Hex Editor

This section will describe how to invoke and use the Hex Editor tool. The Hex Editor is a simple multipurpose hex editor. The Hex Editor supports multi-file editing.

For the Other Tools, you can have unlimited number of them running (up to what your box can support). Also, while any of the Other Tools are running, the user can still use MQMC-GUI.

From the main panel of MQMC-GUI, select **Tools**, then **Hex Editor**.

This is the main panel of the Hex Editor and it has 3 open files.



Supported features:

- Open / editing of multiple files
- Recent File List
- Copy, Find and Find Next
- Save, Save As and Print

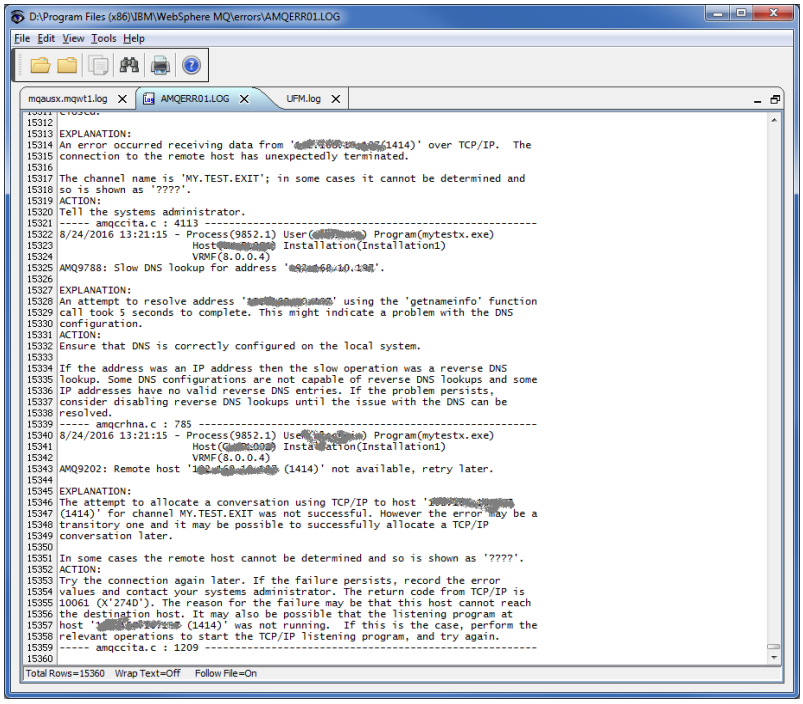
5.3 Tail File

This section will describe how to invoke and use the Tail File tool. The Tail File is a simple multipurpose file monitoring viewer. The Tail File supports monitoring of multiple files at the same time.

For the Other Tools, you can have unlimited number of them running (up to what your box can support). Also, while any of the Other Tools are running, the user can still use MQMC-GUI.

From the main panel of MQMC-GUI, select **Tools**, then **Tail File**.

This is the main panel of the Tail File and it has 3 open files.



Supported features:

- Opening of multiple files
- Wrap Text and Recent File List
- Copy, Find, Find Next and Print

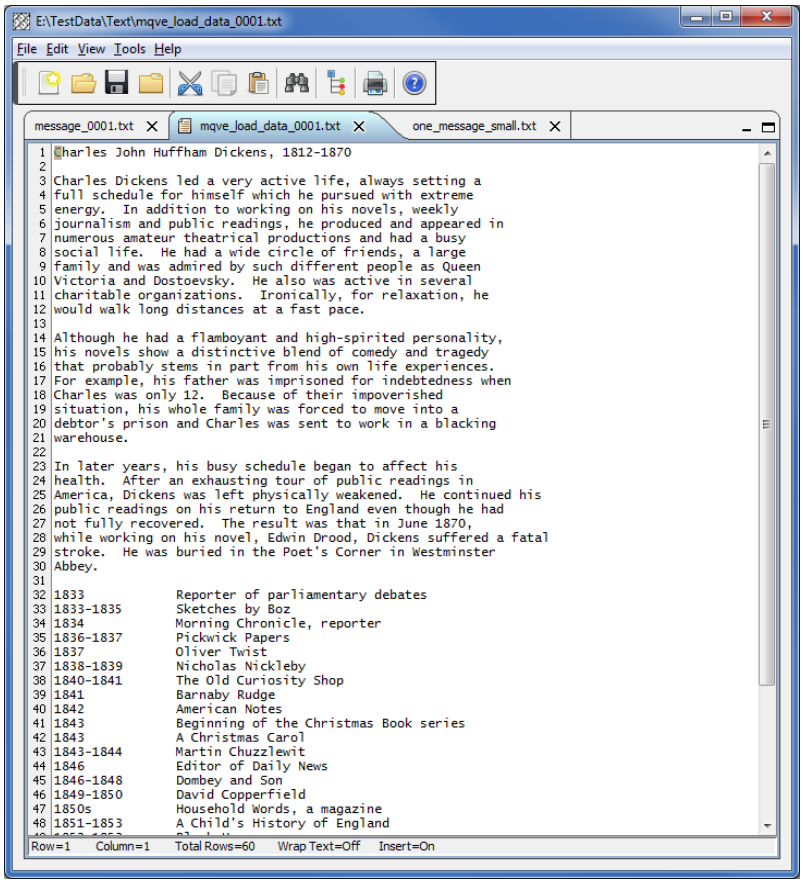
5.4 Text Editor

This section will describe how to invoke and use the Text Editor tool. The Text Editor is a simple multipurpose plain-text editor. The Text Editor supports multi-file editing.

For the Other Tools, you can have unlimited number of them running (up to what your box can support). Also, while any of the Other Tools are running, the user can still use MQMC-GUI.

From the main panel of MQMC-GUI, select **Tools**, then **Text Editor**.

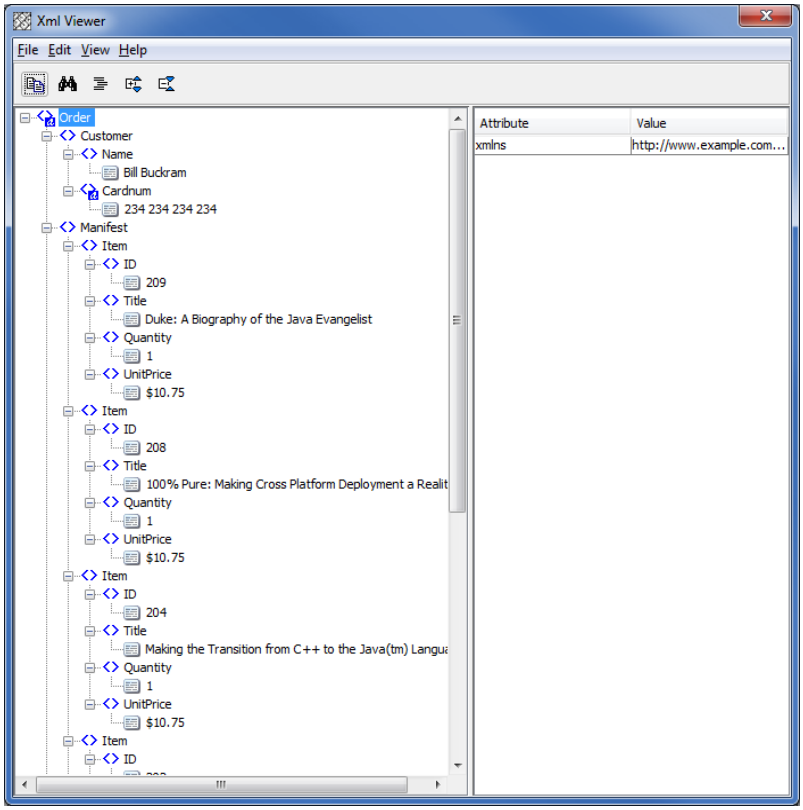
This is the main panel of the Text Editor and it has 3 open files.



Supported features:

- Open / editing of multiple files
- Builtin XML Viewer, Wrap Text
- Recent File List
- Cut, Copy, Paste, Find and Find Next
- Save, Save As and Print

The following shows the Text Editor's builtin XML Viewer.



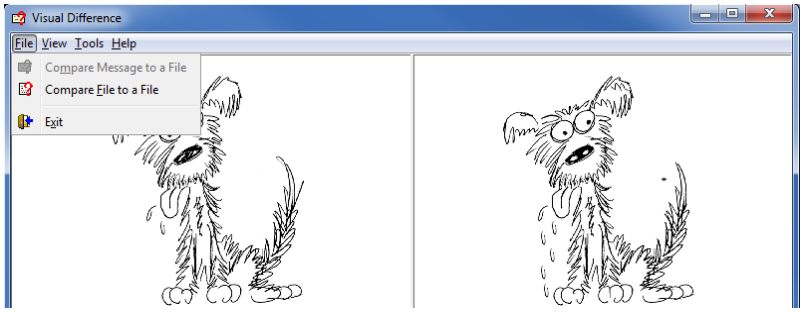
5.5 Visual Difference

This section will describe how to invoke and use the Visual Difference tool. The Visual Difference is a simple file to file comparison tool.

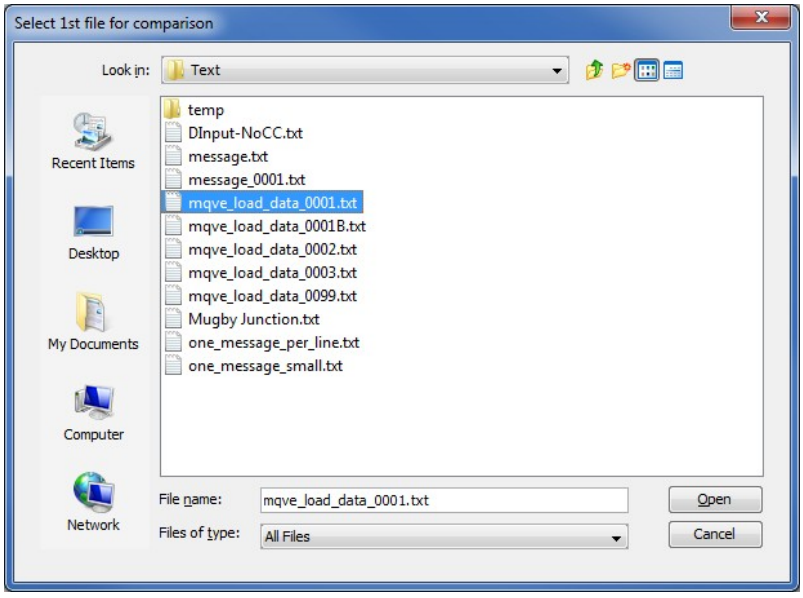
For the Other Tools, you can have unlimited number of them running (up to what your box can support). Also, while any of the Other Tools are running, the user can still use MQMC-GUI.

From the main panel of MQMC-GUI, select **Tools**, then **Visual Difference**.

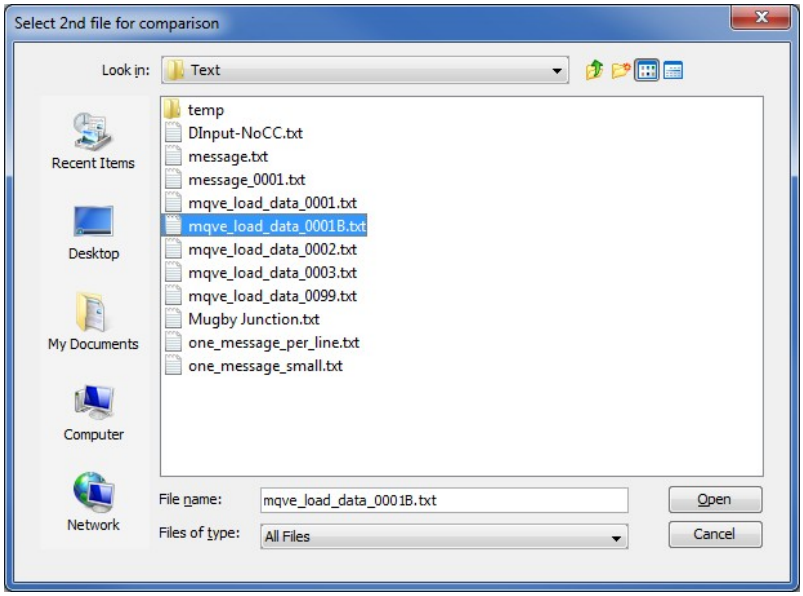
This is the main panel of the Visual Difference. To initiate the file comparison process, click File and then click Compare File to File



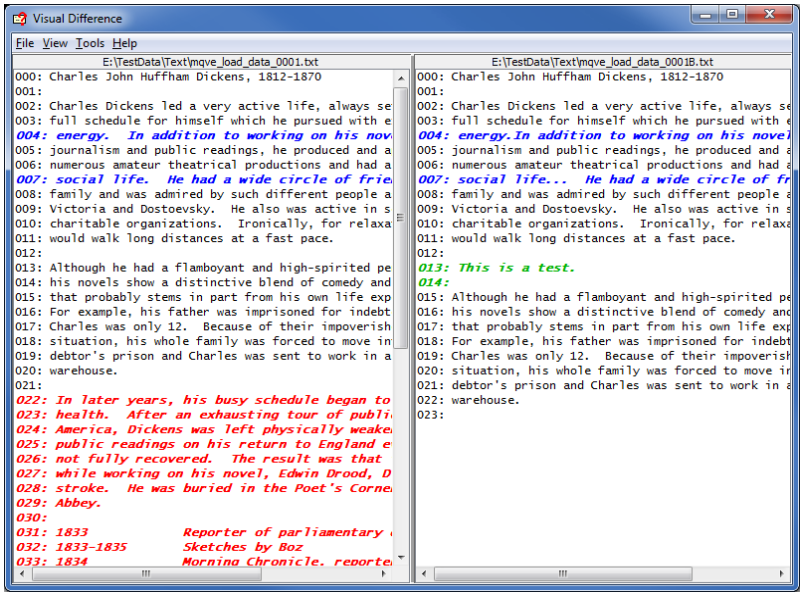
First the user needs to select the first file for the comparison.



Next, the user needs to select the second file for the comparison.



The following shows the difference between 2 files. Differences are highlighted: Green for inserts, Red for deletions and Blue for changes.



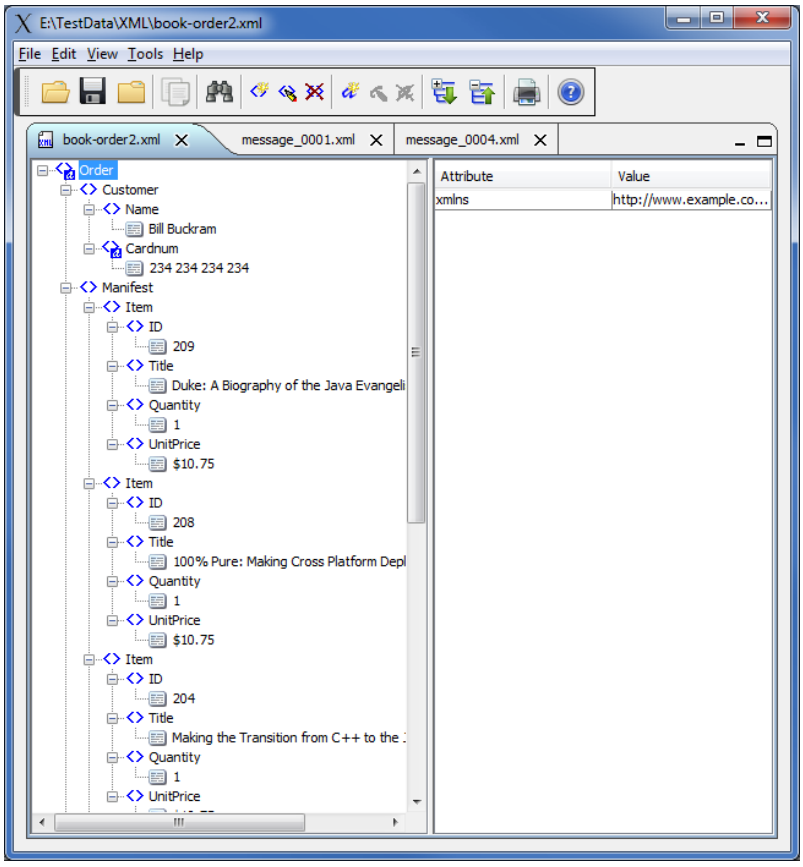
5.6 XML Editor

This section will describe how to invoke and use the XML Editor tool. The XML Editor is a simple multipurpose plain-text editor. The XML Editor supports multi-file editing.

For the Other Tools, you can have unlimited number of them running (up to what your box can support). Also, while any of the Other Tools are running, the user can still use MQMC-GUI.

From the main panel of MQMC-GUI, select **Tools**, then **XML Editor**.

This is the main panel of the XML Editor and it has 3 open files.



Supported features:

- Open / editing of multiple files
- Recent File List
- Copy, Find and Find Next
- Save, Save As and Print

6 Miscellaneous

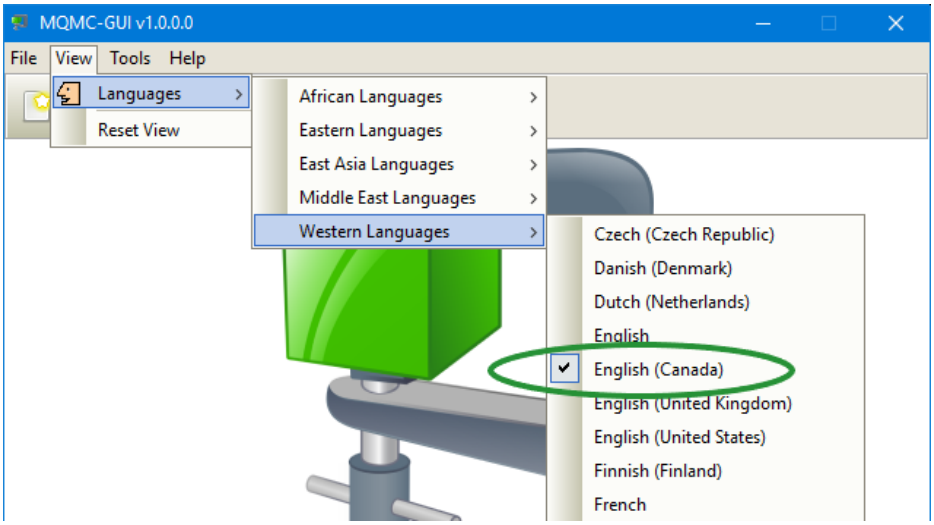
This chapter will describes how the miscellaneous features of MQMC-GUI.

6.1 Language

This section will describe how to switch MQMC-GUI's Language Text to another Language Text. MQMC-GUI has been designed to support Java's multi-language environment. Currently, MQMC-GUI has language support (menu items, buttons and titles) for the following 55 languages:

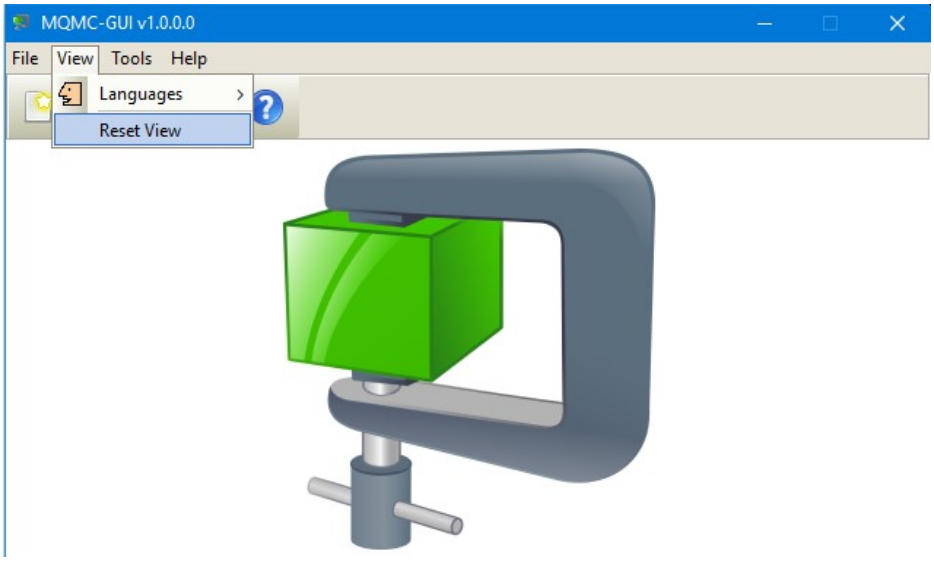
➢ Amharic (am)	➢ Hungarian (hu)	➢ Shona (sn)
➢ Arabic (ar)	➢ Igbo (ig)	➢ Sindhi (sd)
➢ Azerbaijani (az)	➢ Indonesian (in)	➢ Spanish (es)
➢ Bengali (bn)	➢ Italian (it)	➢ Sundanese (su)
➢ Cebuano (ceb)	➢ Japanese (ja)	➢ Swahili (sw)
➢ Chinese (China) (zh)	➢ Javanese jv)	➢ Swedish (sv)
➢ Chinese (Taiwan) (zh)	➢ Kannada (kn)	➢ Tamil (ta)
➢ Czech (cs)	➢ Korean (ko)	➢ Telugu (te)
➢ Danish (da)	➢ Malay (ms)	➢ Thai (th)
➢ Dutch (nl)	➢ Malayalam (ml)	➢ Turkish (tr)
➢ English (en)	➢ Marathi (mr)	➢ Ukrainian (uk)
➢ Finnish (fi)	➢ Norwegian (no)	➢ Urdu (ur)
➢ French (fr)	➢ Panjabi (pa)	➢ Uzbek (uz)
➢ German (de)	➢ Pashto (ps)	➢ Vietnamese (vi)
➢ Greek (el)	➢ Persian (fa)	➢ Xhosa (xh)
➢ Gujarati (gu)	➢ Polish (pl)	➢ Yoruba (yo)
➢ Hausa (ha)	➢ Portuguese (pt)	➢ Zulu (zu)
➢ Hebrew (iw)	➢ Romanian (ro)	
➢ Hindi (hi)	➢ Russian (ru)	

From the main panel of MQMC-GUI, select **View**, then **Language**.



6.2 Reset View

This section will describe how to reset the view of MQMC-GUI if the display is not correct.



The user can also reset the view from the command line by passing in the ***RG*** parameter.

6.2.1 Windows Command Line

On Windows, issue the following command to reset the view of MQMC-GUI:

mqmcgui.exe RG

6.2.2 Linux and macOS Command Line

On Linux and macOS, issue the following command to reset the view of MQMC-GUI:

./mqmcgui.sh RG

7 Appendix A – Frequently Asked Questions (FAQ)

Q. Can IniFiles be create on Windows for MQMC on Unix?
A. Yes. Once the IniFile has been saved, the user can copy or ftp (in ASCII mode) the file to the remote server.

8 Appendix B – MQMC-GUI Upgrade Procedures

To upgrade an existing installation of MQMC-GUI, please do the following in the appropriate section below.

8.1 Windows Upgrade

- Stop all instances of MQMC-GUI
- Backup any MQMC-GUI data files in the MQMC-GUI install directory
- Go to the Windows **Control Panel** and select **Program and Features**
- Select MQMC-GUI and then click the **Uninstall** button
- Run the install program called: **mqmcgui-setup-withjre.exe**
- The installer follows the standard Windows install procedures and provides default values for the user.
- When the install program has completed execution, there will be a newly created folder under **Start** -> **All Programs** called **MQMC-GUI**.
- Restore the MQMC-GUI data files if necessary
- To run MQMC-GUI, select the MQMC-GUI icon from within the MQMC-GUI folder.

8.2 Linux or macOS Upgrade

- Stop all instances of MQMC-GUI
- Backup any MQMC-GUI data files in the MQMC-GUI install directory
- Remove the installation i.e. rm -Rf Capitalware/MQMC-GUI/
- Unzip the file, open a shell prompt and execute the following commands:

  unzip mqmcgui.zip

- When the unzip command has finished, there should be a directory tree as follows:

  Capitalware/MQMC-GUI

- Change directory by typing the following:

  cd Capitalware/MQMC-GUI

- Before running MQMC-GUI for the first time, you need to add the execute attribute to the mqmcgui.sh shell scripts. Type the following:

  chmod +x mqmcgui.sh

- To run MQMC-GUI from the Capitalware/MQMC-GUI directory, then type the following:

  ./mqmcgui.sh

9 Appendix C – Support

The support for MQMC-GUI can be found at the following location:

By email at:
support@capitalware.com

By regular mail at:

Capitalware Inc.
Attn: MQMC-GUI Support
Unit 11, 1673 Richmond Street, PMB524
London, Ontario N6G2N3
Canada

10 Appendix D – Summary of Changes

- MQMC-GUI v1.0.0
  - Initial release.

11 Appendix E – License Agreement

This is a legal agreement between you (either an individual or an entity) and Capitalware Inc. By opening the sealed software packages (if appropriate) and/or by using the SOFTWARE, you agree to be bound by the terms of this Agreement. If you do not agree to the terms of this Agreement, promptly return the disk package and accompanying items for a full refund.
SOFTWARE LICENSE

1. GRANT OF LICENSE. This License Agreement (License) permits you to use one copy of the software product identified above, which may include user documentation provided in on-line or electronic form (SOFTWARE). The SOFTWARE is licensed as a single product, to an individual queue manager, or group of queue managers for an Enterprise License. This Agreement requires that each queue manager of the SOFTWARE be Licensed, either individually, or as part of a group. Each queue manager's use of this SOFTWARE must be covered either individually, or as part of an Enterprise License. The SOFTWARE is in use on a computer when it is loaded into the temporary memory (i.e. RAM) or installed into the permanent memory (e.g. hard disk) of that computer. This software may be installed on a network provided that appropriate restrictions are in place limiting the use to registered queue managers only. Each licensed queue manager will be provided with a perpetual license key and the licensee may continue to use the SOFTWARE, so long as the licensee is current on the Yearly Maintenance Fee. If the licensee stops paying the Yearly Maintenance Fee, then the SOFTWARE must be removed from all systems at the end of the current maintenance period.

2. COPYRIGHT. The SOFTWARE is owned by Capitalware Inc. and is optimized by United States Of America and Canada copyright laws and international treaty provisions. You may not copy the printed materials accompanying the SOFTWARE (if any), nor print copies of any user documentation provided in on-line or electronic form. You must not redistribute the registration codes provided, either on paper, electronically, or as stored in the files mqmc.ini, mqmc_licenses.ini or any other form.

3. OTHER RESTRICTIONS. The registration notification provided, showing your authorization code and this License is your proof of license to exercise the rights granted herein and must be retained by you. You may not rent or lease the SOFTWARE, but you may transfer your rights under this License on a permanent basis, provided you transfer this License, the SOFTWARE and all accompanying printed materials, retain no copies, and the recipient agrees to the terms of this License. You may not reverse engineer, decompile, or disassemble the SOFTWARE, except to the extent the foregoing restriction is expressly prohibited by applicable law.

LIMITED WARRANTY

LIMITED WARRANTY. Capitalware Inc. warrants that the SOFTWARE will perform substantially in accordance with the accompanying printed material (if any) and on-line documentation for a period of 365 days from the date of receipt.

CUSTOMER REMEDIES. Capitalware Inc. entire liability and your exclusive remedy shall be, at Capitalware Inc. option, either (a) return of the price paid or (b) repair or replacement of the SOFTWARE that does not meet this Limited Warranty and that is returned to Capitalware Inc. with a copy of your receipt. This Limited Warranty is void if failure of the SOFTWARE has resulted from

accident, abuse, or misapplication. Any replacement SOFTWARE will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer.

NO OTHER WARRANTIES. To the maximum extent permitted by applicable law, Capitalware Inc. disclaims all other warranties, either express or implied, including but not limited to implied warranties of merchantability and fitness for a particular purpose, with respect to the SOFTWARE and any accompanying written materials.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES. To the maximum extent permitted by applicable law, in no event shall Capitalware Inc. be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of the use or inability to use the SOFTWARE, even if Capitalware Inc. has been advised of the possibility of such damages.

12 Appendix F – Notices

Trademarks:

AIX, IBM, MQSeries, OS/2 Warp, OS/400, iSeries, MVS, OS/390, WebSphere, IBM MQ and z/OS are trademarks of International Business Machines Corporation.

HP-UX is a trademark of Hewlett-Packard Company.

Intel is a registered trademark of Intel Corporation.

Java, J2SE, J2EE, Sun and Solaris are trademarks of Sun Microsystems Inc.

Linux is a trademark of Linus Torvalds.

Mac OS X is a trademark of Apple Computer Inc.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation.

UNIX is a registered trademark of the Open Group.

WebLogic is a trademark of BEA Systems Inc.