

MQSeries - Queue depth monitoring tool

Version 2.3

14 November 2001

Francois van der Merwe
PO Box 4034
Cape Town
8001
South Africa

francoisvdm@za.ibm.com

Property of IBM

Take Note!

Before using this report be sure to read the general information under "Notices".

Second Edition, November 2001

This edition applies to Version 2.3 of *MQSeries - Queue depth monitoring tool* and to all subsequent releases and modifications unless otherwise indicated in new editions.

© **Copyright International Business Machines Corporation 2001**. All rights reserved. Note to US Government Users -- Documentation related to restricted rights -- Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule contract with IBM Corp.

Table of Contents

Trademarks and service marks.....	iv
Chapter 1. Usage.....	1
Basic use	1
Install	1
Chapter 2. Description.....	2
Small Screen layout.....	2
Big Screen layout.....	3
Startup file	4
Wish list	4

Notices

The following paragraph does not apply in any country where such provisions are inconsistent with local law.

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any reference to an IBM licensed program or other IBM product in this publication is not intended to state or imply that only IBM's program or other product may be used. Any functionally equivalent program that does not infringe any of the intellectual property rights may be used instead of the IBM product.

Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, New York 10594, USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS-IS. The use of the information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item has been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Trademarks and service marks

The following terms, used in this publication, are trademarks of the IBM Corporation in the United States or other countries or both:

IBM

MQSeries

Acknowledgments

Thank you to Peter Potkay who made some very good suggestions.

Summary of Amendments

Date	Changes
8 November 2001	V2.3 - Initial release as an MQSeries SupportPac
14 November 2001	Add details of how queue manager and queues are defined to WatchQ command

Preface

This tool (WatchQ) was born out of a need to get some basic information about what is going on in queues while programming and debugging an MQSeries application. The pain of "clicking and refreshing" windows was just too big for me in the end, so I created this helper utility.

When active, the application usually only occupies a small portion of the screen, so you can make it sit in the corner of your screen without obstructing what you do.

The basic functionality is just to show and regularly refresh a few selected queue depths, but please read on, I'm sure there are a few surprises in it as well.

Chapter 1. Usage

Basic use

The MQSeries - Queue depth monitoring tool (WatchQ) is a small utility that sits silently in the corner of a computer screen and reports on queue depths. Yes, it got some other functions as well, but more on that later. By default, it floats on top of all other general windows and it refreshes every 1 second.

This makes it an ideal tool for programmers or debuggers who just want to watch MQSeries queue depths as they work. No more need to flip back and forth between your development environment and your "queue watching" utility or tool just to see if your latest changes actually posted some messages.

It can also be used by system debuggers and monitors to silently watch some queues while they carry on with their day-to-day tasks, and to raise an alert when something goes wrong.

But, the utility does a little bit more than just watching queues. It is also very easy to clear a queue with just one click of the mouse and to generate test messages from memory, files or other queues. It can also show you the information in a queue and even delete the displayed message.

Install

Just unzip the exe file into a directory and *presto* you are done. You could also unzip the .ini file into your working directory; it is a good way of starting WatchQ with some default parameters.

To run WatchQ, make sure the MQSeries client environment is installed and working. Either MQServer OR the MQCHLTAB/MQCHLLIB environment variables must be set up.

Then run the application as follows from the command line:

```
watchq qmgr queue1 queue2 queue3 queue4 queue5
```

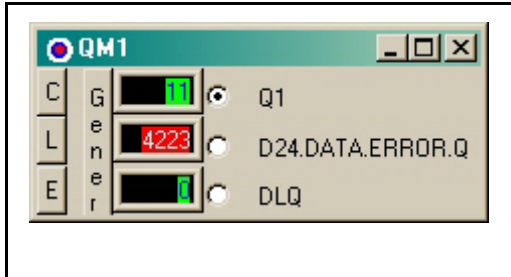
where:

qmgr	is the queue manager where the environment variable(s) are pointing, and
queuex	are queues on that queue manager

Chapter 2. Description

Small Screen layout

When the application starts, it pops up as a small window on the screen.



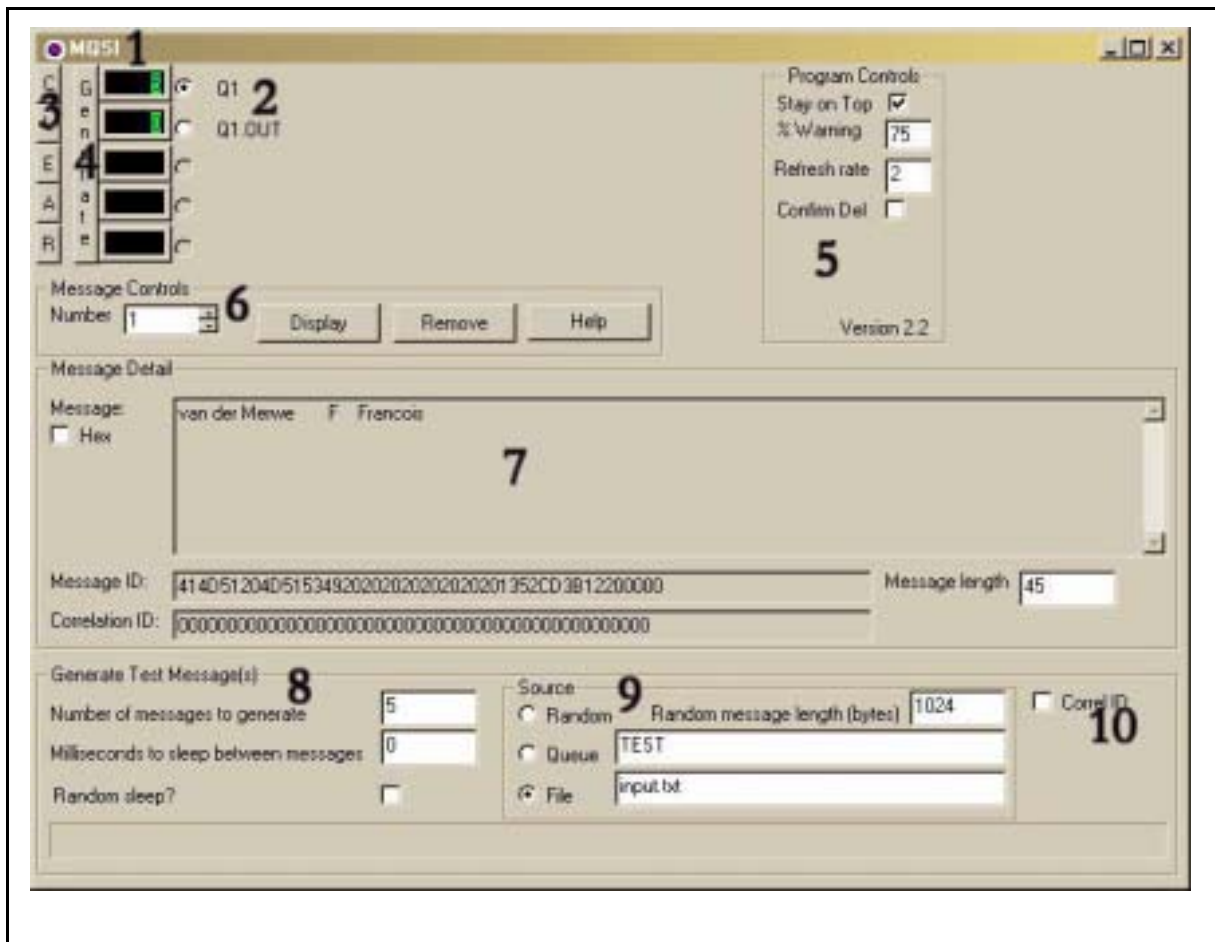
This window will “float” (stay on top) over most other windows on the screen. Drag the window to a convenient position on the screen.

There are five basic elements on the screen. They are, from the left:

- (i) Clear buttons. Click to clear queue.
- (ii) Generate button. Use to generate information to populate selected queue.
- (iii) Queue depth indicator. If this indicator is red, then the queue is more than x% full.
- (iv) Queue selection indicator.
- (v) Queue name.

Big Screen layout

When the application is maximized, the following window will display (without the black numbers).



1. Name of "queue manager" utility is watching.
2. Queues (maximum 5) in watch list. To the left of queue names are queue selection button. Most actions performed will go against the selected queue. To the left of the queue selection buttons are the queue depth indicators, showing the number of messages in the queue.
3. One button for every queue to clear all messages from the queue.
4. Generate test messages according to settings in 8 and 9 in the selected queue.
5. Some basic program controls. If "Stay on Top" is selected then the watchq window will stay on top of almost all other windows on the screen. The "%Warning" field will be used to turn the queue depth indicators red when queue reaches x% depth. The "Refresh rate" indicates every how many seconds the queue depth indicators will be refreshed. If the "Confirm Del" indicator selected then a window will pop up when you attempt to clear a queue asking you if you are sure.
6. Display or remove the active message in the selected queue.
7. Message Display area.
8. When the "Generate" button is pushed it will use 8 and 9 to direct behavior. When the number of messages to generate is more than what is available in the source, the source will wrap. The

sleep milliseconds field is the amount of time the generator will "sleep" before it sends the next message. When the "Random sleep" indicator is on, the generator will sleep a random amount of time with a maximum of the amount indicated in the sleep field.

9. Indicates the source for messages that must be generated in selected queue. When "Random" is selected then a random string of bytes will be generated with the length as specified in the "Random message length" field. When "Queue" is selected, then it will use the local queue specified in the queue field to generate messages from. This queue specified does not have to be in the "list of five" monitored by watchq. When "File" is selected then watchq will use the file specified as source, every "line" will be a separate message.
10. When CorrelID is selected then watchq will generate a "timestamp" CorrelID. Just something I used somewhere else and I plan to incorporate more function around this in future release.

Startup file

When the file "watchq.ini" is in the active directory, then it is possible to give some startup values to WatchQ. An example file is as follows:

```
1 75 2 0 5 0 0 f 1024 TEST input.txt 1
```

Description (other options after description in []) :

- 1 = Stay on top [0]
- 75 = % Warning
- 2 = Refresh rate
- 0 = Do not confirm delete [1]
- 5 = Number of messages to generate
- 0 = Number of milliseconds to wait between messages
- 0 = Do not wait random [1]
- f = Make file selection the default [r,q]
- 1024 = Initial size of random generated message
- TEST = Initial queue to generate from
- Input.txt= Initial file to generate from
- 1 = Generate CorrelID [0]

Wish list

1. Add startup position to ini file.
2. Add basic queue stats graph to window. Indicates queue depth over time as a graph.
3. Send warning/error messages under certain error conditions.
4. Add error code fields (completion code + reason code).
5. Queue Triggered indication

End of Document