

Welcome to **QueueZee v2.x** - *QueueZee* is a powerful interactive application designed for making message management, manipulation and analysis using IBM's Websphere MQ as easy and efficient as possible.

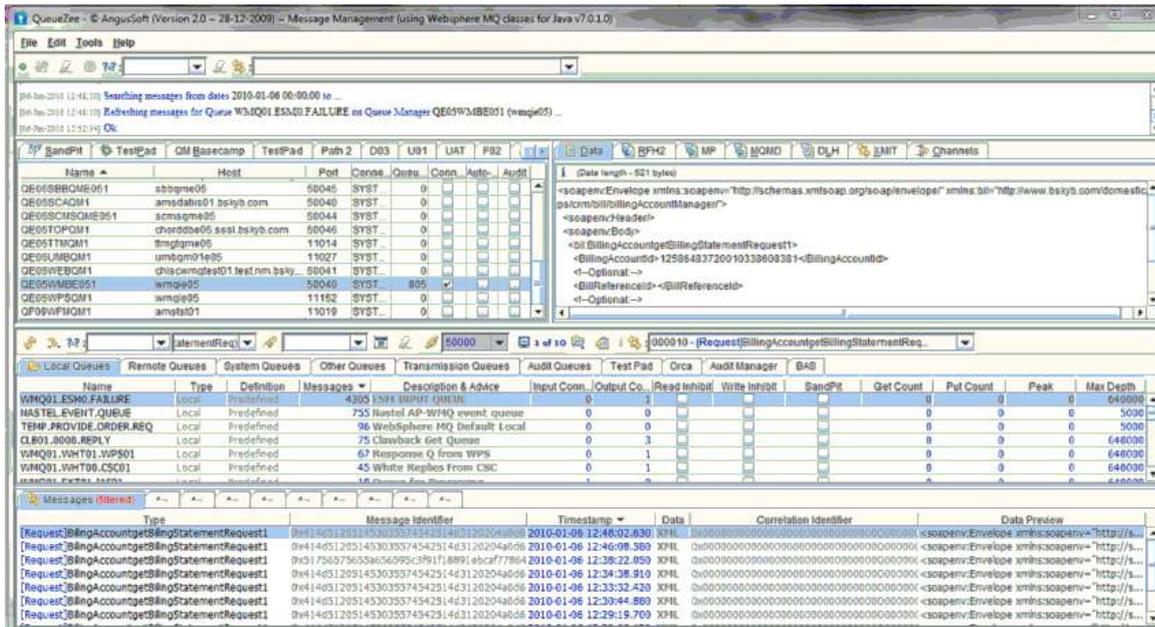
The primary user group *QueueZee* is aimed at developers and testers working with IBM WMQ version 6 & 7 as the main infrastructure for data exchange in anything from a relatively small environment to large Enterprise Systems. *QueueZee* itself uses the IBM WMQ classes for Java v7.0.1.0.

*QueueZee* has been tested on Windows 7, Vista & XP (32 & 64-bit) plus Linux (Fedora) and comes packaged in a single compressed file, (**qzv2.x.zip**). This must be extracted to a directory of your choice (for example C:\QueueZee). Next you should read the *ReadMe.txt* file before executing (double clicking) the file **qz.bat** found in the top level directory - you have probably done this already if you're reading this from the *QueueZee* help menu option!

Once *QueueZee* is up and running you must close it cleanly when you've finished to enable the configuration file to update. This is especially important after the first invocation as this is when the file is created and configuration details are saved for future reference. If *QueueZee* ends abnormally and fails to update the configuration file cleanly it will attempt to recover configuration from a backup copy it maintains automatically for you on the next invocation. **We also recommend you periodically manually save a copy of your backup configuration file somewhere safe.**

Once *QueueZee* is started for the first time it will look for any *IBM Websphere MQ Explorer* Queue Manager definitions it can find and ask if you would like them loaded. If you answered **yes** these will all be loaded into the Queue Manager *Basecamp* group (see [Queue Managers](#)).

Once *QueueZee* has successfully started with various Queue Managers listed and connected, the application will look something similar to this:



All the various window panes can be pulled and stretched to suit your personal preference and with the exception of column re-ordering *QueueZee* will remember all your settings and how you left things the next time it's invoked. Columns can be sorted in ascending and descending order by clicking on the column header.

Each pane contains the following information:

- **Top pane** ~ this contains a continuous log of your current activity. If something doesn't appear to work always check this panel for any error messages first, (eg. a Queue Manager not responding etc.). Often the error message will be an expected response so certain MQ exceptions don't always mean something is wrong or needs attention.
- **Middle left tabbed pane** ~ this contains all your Queue Manager groups, the *TestPad* (not available for v2.x) and the *SandPit*. All these are explained later in this document.
- **Middle right tabbed pane** ~ this contains body and header information for a message if one is selected. The message data for the body, RFH2 & Message Properties can be freely edited and modified. If the Queue selected is a transmission Queue with an associated channel this will also be shown in the *Channels* tab.
- **Second from bottom tabbed pane** ~ this contains all the viewable Queues from the selected Queue Manager (if one is selected). Queues are split into folders that can be customised to suit your needs.
- **Bottom tabbed pane** ~ this contains all the messages refreshed from the selected Queue and relevant Queue folder (if one is selected). Refreshed messages can be filtered by specifying a variety of different search criteria ([explained here](#)).

To discover the full power of *QueueZee* explore the rest of this document!

Access Keys

[Previous Top Next](#)

Although *QueueZee v2.x* is available as Freeware you can use it in browse mode indefinitely you will require an access key to unlock the fully featured product.

To request an access key a donation must be made from our website ([AngusSoft](#)) then the access key information displayed in the *Access Key* dialog which appears on startup must be copied (Ctrl-v) and sent to [admin@AngusSoft.co.uk](mailto:admin@AngusSoft.co.uk).

You will receive an access key by reply and there is no minimum limit for donations. You can also request up to 5 access keys from one donation.

**IMPORTANT: Please provide your name and location (i.e. Edinburgh, UK) when you request an access key.** Your access key will be returned and will work on a per-host basis so only one need be requested per computer. **Please keep your key in a safe place as you may need it again if upgrades to your computer remove the original key store.** There is no limit to the amount of key requests per person or company, (as per the donation rule).

If you would like to evaluate *QueueZee* immediately without waiting for an access key you can tick (click on) the **browse only** check-box when the access key dialog first appears. This will allow *QueueZee* to work in browse only mode indefinitely.

A separate configuration file is maintained per user on the same computer so several logins can use *QueueZee* on the same installation.

Support

[Previous Top Next](#)

- ❖ **QueueZee v2.x is Freeware** (browse only mode).
- ❖ Support for v2.x is also provided **free of charge** and although no guarantees are made about timeframes for upgrades & patches every effort will be made to meet user demands. Visit [AngusSoft](http://AngusSoft) for forum support or email [support@AngusSoft.co.uk](mailto:support@AngusSoft.co.uk). Users with access keys will be given priority.
- ❖ A lot of time and effort has gone into the development of **QueueZee** and as such any donations are very welcome and help to ensure quality of service and future maintenance!! (visit [AngusSoft](http://AngusSoft) for details).

Contacting AngusSoft

[Previous](#) [Top](#) [Next](#)

- ✓ Please contact [support@AngusSoft.co.uk](mailto:support@AngusSoft.co.uk) or visit us at [AngusSoft](http://AngusSoft) for anything concerning **QueueZee**.

Copyrights & Disclaimers

[Previous](#) [Top](#) [Next](#)

➤ **AngusSoft provides QueueZee v2.x under the following conditions:**

Copyright © 2009-2011, Angus Cooke  
Copyright © 2009-2011, AngusSoft  
All rights reserved.

Redistribution of the QueueZee v2.x package, without modification, is permitted provided that the following conditions are met:

Neither the name of QueueZee nor the name of AngusSoft may be used to endorse or promote products developed or tested using this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

➤ **QueueZee message export formats are supported by XStream 1.1.3:**

Copyright (c) 2003-2006, Joe Walnes  
Copyright (c) 2006-2007, XStream Committers  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of XStream nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

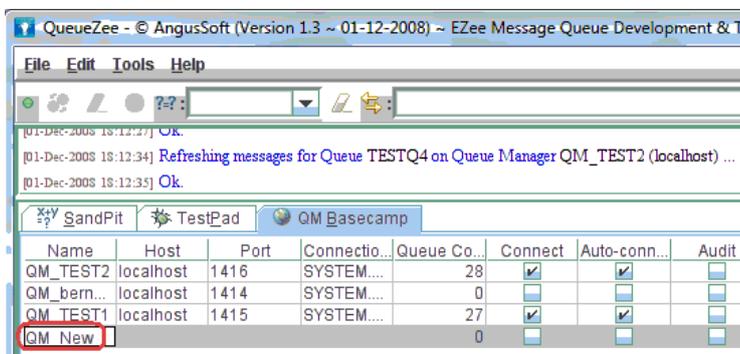
Queue Managers

[Previous](#) [Top](#) [Next](#)

QueueZee organises and references Queue Managers within Queue Manager Groups which are shown in the top left split pane (see [Getting Started](#)).

An initial Group called **QM Basecamp** is created for you by default and this is where all newly imported Queue Managers will appear unless you have a custom Queue Manager Group selected.

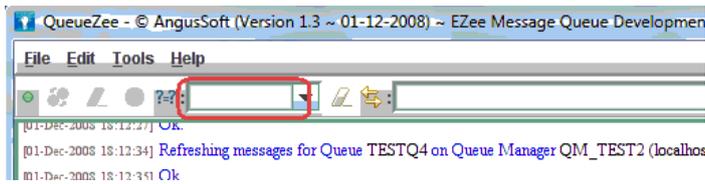
To manually add a Queue Manager to any Group you simply click in the relevant empty fields and enter the details. A field can be edited as soon as you click on it (double click to get full focus):



QueueZee will automatically fill the *Host*, *Port* & *Connection Channel* fields with the defined defaults (see [Preferences](#)) once a Queue Manager name has been entered and either *[Enter]* is pressed or a previous field is completed.

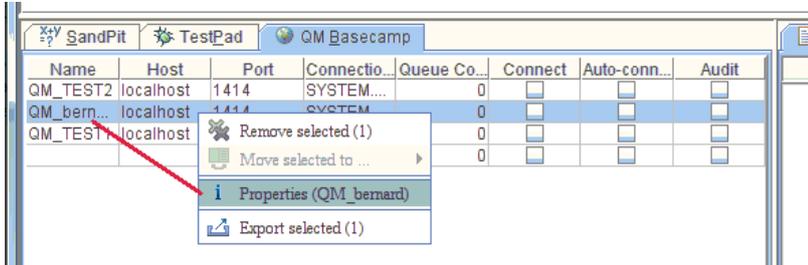
As soon as all four definition fields are complete the new Queue Manager definition will be added to the Group. Once this has happened you cannot edit the *Name* & *Host* as these form the key for internal reference. You must remove and re-add a Queue Manager to change key details.

If you have a large list of Queue Managers in a single Group you can apply a filter using the Queue Manager Quick Search drop-down box:



The Quick Search is used in exactly the same way as the [Queue & Message](#) Quick Searches.

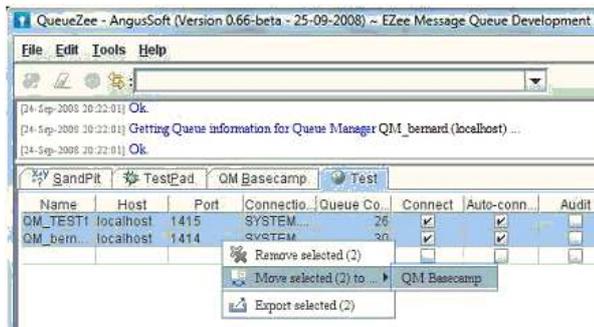
If you need to specify a user id and password for Queue Manager connection you must have **QueueZee v1.1** or above. (download available from [our website](#)). If you are at or above this version you will see a *Properties* option on the Queue Manager pop-up menu when a single Queue Manager is selected:



Selecting this option will let you specify a user id and password for all connections to the specified Queue Manager.

One or more Queue Managers may be selected in a Group and moved to another Group, removed from QueueZee or be exported so the definitions may be imported by another instance of QueueZee.

Select the Queue Managers you want to action then right click to bring up a pop-up menu:



Once a Queue Manager definition has been defined you can connect to it by simply checking the *Connect* check-box. If you would like QueueZee to automatically connect to a Queue Manager when you start it up you must also check the *Auto-connect* check-box.

If *Auto-connect* is checked QueueZee automatically attempts a re-connection whenever you change the *Port* or *Connection Channel* values.

QueueZee will always try to stay connected once you are connected to a Queue Manager. Most of time you won't even notice if the connection has been refreshed but occasionally there may be an unusual delay in operation if QueueZee is struggling to maintain your connection.

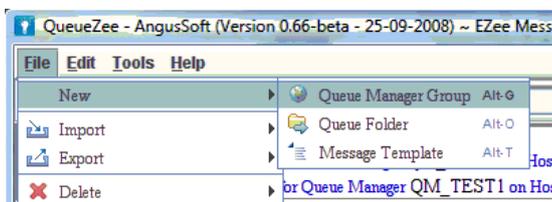
If the Queue Manager becomes unavailable for any length of time QueueZee will give up and you will see a connection error in the log pane.

If you would like to audit any state changes to messages (eg. updates, deletions etc.) then check the *Audit* check-box against the relevant Queue Manager.

Auditing can be turned on and off while connected and QueueZee will create a new audit file each time auditing is turned on. These can be found in QueueZee's home directory, for example:

**C:\Program files\QueueZee\QM\_TEST1\_[localhost]\_AuditLog\_2008-08-10 14-19-42.qz**

A new Queue Manager Group can be created from the **File** menu:



Queue Manager groups can also be deleted from the **File->Delete->Queue Manager Group** menu option. The Queue Manager Group you currently have selected will be deleted and any Queue Manager definitions it contains will automatically be moved back to the **QM Basecamp** Group.

Queue Manager definitions may also be imported from the **File->Import->Queue Manager Definitions** menu option. QueueZee will read Queue Manager definitions from MQ Explorer (v6 & 7) XML files or previously exported QueueZee Queue Manager definition files. There are no guarantees QueueZee will read all MQ Explorer v6 & 7 XML definition files as this is purely provided as a "nice to have" feature. If a file format isn't recognised it will be ignored.

For MQ Explorer v6 on Windows XP the XML definition file is generally found a location similar to this:

**C:\Documents and Settings\AngusSoft\Application Data\IBM\MQ Explorer\metadata\plugins\com.ibm.mq.explorer.ui\WMQ\_Handles.xml**

All newly imported Queue Manager definitions will be loaded into the current Group you have selected where they can be moved to other Groups if required. If no Queue Manager Group is selected they will be imported to the **QM Basecamp** group by default.

**Note:** QueueZee will attempt to read any file it is given and make sense of the contents so specifying files with unrecognised or garbage content can result in strange definitions being loaded!

General navigation

[Previous](#) [Top](#) [Next](#)

Once a Queue Manager is connected and selected all its Queues will be displayed in the Queue pane:

Name	Host	Port	Connectio...	Queue Co...	Connect	Auto-conn...	Audit
QM_TEST2	localhost	1416	SYSTEM...	28	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
QM_berm...	localhost	1414	SYSTEM...	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
QM_TEST1	localhost	1415	SYSTEM...	26	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
				0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Name	Type	Definition	Messages	Description &...	Input Connec...	Out
TESTQ3	Local	Predefined	4050			0
AMQ_48E6832E	Local	Temporary ...	0 (MQJ001: ...			0
TESTQ4	Local	Predefined	3984			0

Queues are organised into folders which by default fall into one of four categories; *Local Queues*, *Remote Queues*, *System Queues* & *Other Queues*. These default folders are fixed and cannot be removed or renamed.

To quickly search for a specific Queue you can use the *Quick Search* text box. This parses search text as a regular expression (as do the other search boxes on the Queue toolbar).

Previous searches are stored in the text box drop-down to allow easy access on subsequent searches. For example, to search for Queues **TESTQ3** or **TESTQ4** you might enter "(TESTQ3)|(TESTQ4)" as follows:

Name	Type	Definition	Messages	Description &...	Input Connec...	Out
TESTQ3	Local	Predefined	4050			0
TESTQ4	Local	Predefined	3984			0

While any search criteria is displayed in the Queue *Quick Search* text box this will be applied to any folder you have selected - it's worth remembering this if you think all your Queues have disappeared when you switch folders!

As with any other table in QueueZee Queues shown may be sorted by clicking a column header, (columns may also be re-ordered by dragging and dropping the headers). To clear all the search boxes on Queue toolbar and display all Queues and Messages again you can click the *Filter Clear* button as shown:

Name	Type	Definition	Messages	Description &...	Input Connec...	Out
TESTQ3	Local	Predefined	4050			0
AMQ_48E6832E	Local	Temporary ...	0 (MQJ001: ...			0
TESTQ4	Local	Predefined	3984			0

If you have a lot of Queues or specific Queues you may want to quickly access frequently for one particular project you might want to consider creating custom folders.

QueueZee lets you create as many custom folders as you like which allows you to move Queues from the default folders into folders of your choice.

For example, you might create a folder called "Test" from the main **File->New->Queue Folder** menu option and move the Queues **TESTQ3** & **TESTQ4** into it, as shown:

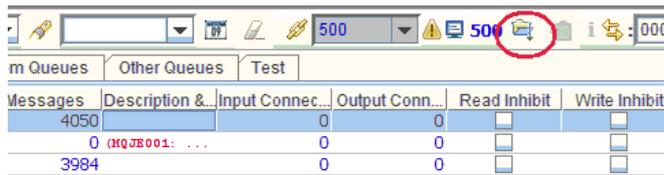
Name	Type	Definition	Messages	Description &...	Input Connec...	Output Conn...
TESTQ3	Local	Predefined	4050			0
AMQ_48E6832E	Local	Temporary ...	0			0
TESTQ4	Local	Predefined	3984			0

Queue Folders are managed on a per Queue Manager basis so if you have Queue **TESTQ3** in the "Test" folder for one Queue Manager a Queue of the same name on another Queue Manager will not also automatically be in the "Test" folder unless you also explicitly move it there.

A custom Queue Folder may be deleted when it is selected from the main **File->Delete->Queue Folder** menu option. If you remove Queues from a custom folder they will dynamically move back to their default folders.

When you delete a folder any Queues it contained will also automatically move back to their default folder locations although you will need to refresh the Queue list for the Queue Manager to see them again.

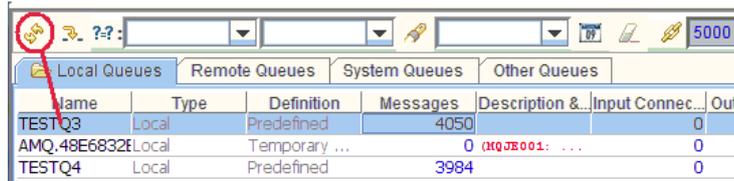
A quick way to set column widths for new custom folders the same as all the other folders is to select an existing folder and click on the *Make all folders like this* button as shown:



This will apply the current folder column width settings to all the other folders for the Queue & Message panes.  
Refreshing & searching

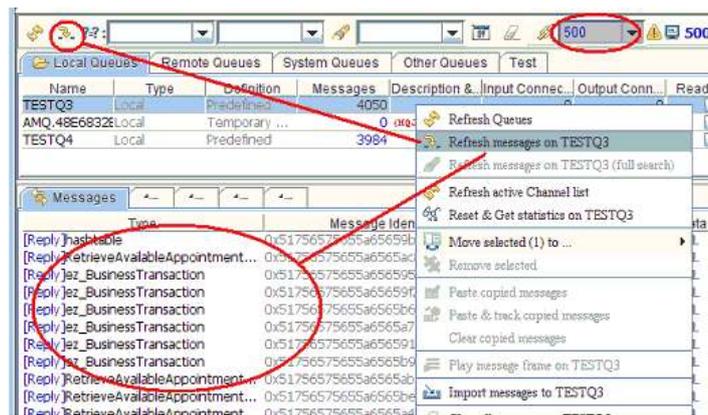
[Previous](#) [Top](#) [Next](#)

If a new Queue has been created you want to display or want to refresh the Queue list on a connected Queue Manager at any time this can be done via the *Queue Refresh* button on the far left of the Queue toolbar or via the *Queue* pop-up menu (visible by right clicking the mouse when a Queue is selected and in focus).

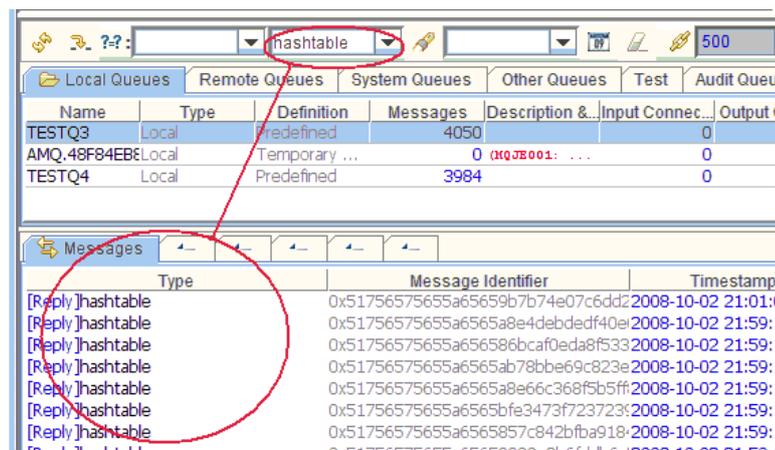


Every time a Queue is selected the *Messages*, *Input Connections* & *Output Connections* are updated if there is still a valid connection to the Queue Manager. If the connection has been lost QueueZee will not attempt reconnection until the next refresh/search operation.

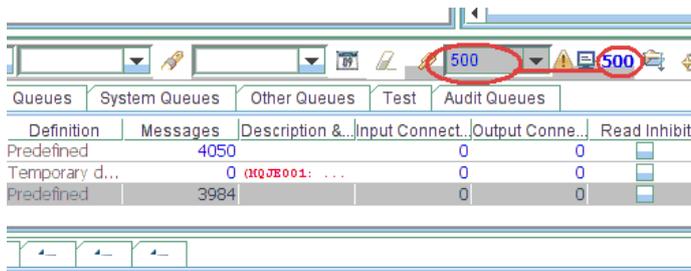
To show messages on a Queue you can either click the *Message Refresh* button next to the *Queue Refresh* button or select the *Message Refresh* option from the Queue pop-up menu. Messages will be shown in the message pane below the Queue pane up to a limit determined by the *Maximum Messages* drop-down as shown:



Searching for a specific message in the list of recently refreshed messages can be achieved by entering a regular expression in the *Message Quick Search* in the same way you would if you were searching for a specific Queue ([General navigation](#)). The *Message Quick Search* text box is located as shown:



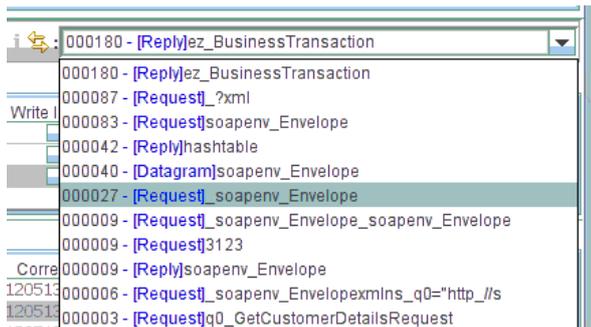
The maximum number of messages QueueZee will show in the *Messages* pane is determined by what's currently selected in the *Maximum Messages* drop-down found on the Queue toolbar as previously stated and shown:



The number to the right of this drop-down will show how many messages are actually shown.

If there are more messages available than specified by the current maximum messages setting a yellow warning triangle will appear indicating the message limit was reached, (as shown above).

When messages are loaded in the message pane you can quickly see the distribution of all the various message types, (as identified by QueueZee), from the drop-down list at the far right-hand side of the Queue toolbar, as shown:

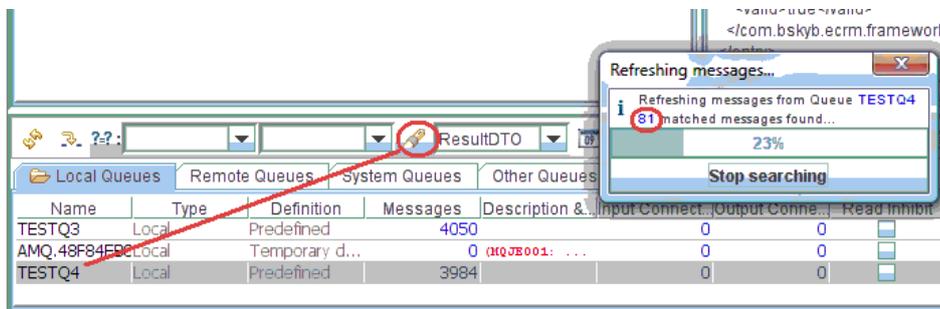


The number on the left indicates how many of each message type is loaded and is always sorted in descending order.

This list is purely for visual purposes and provides no other function. To see large message distributions across multiple Queues including different Queue Managers you must use a similar feature available from the [SandPit](#).

Often there is a requirement to search for a particular message or messages based on key data inside the message.

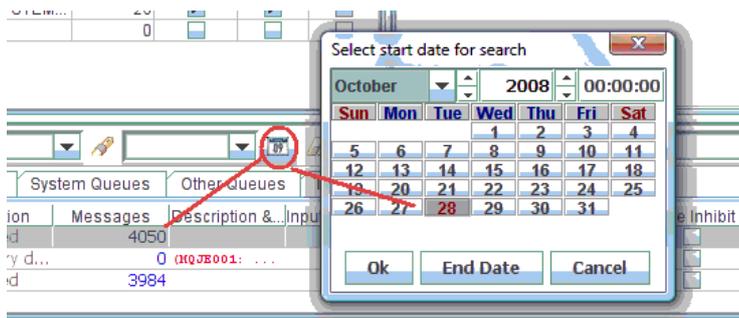
QueueZee lets you do this on a per Queue basis allowing a regular expression search as used in the other *Quick Search* text boxes. To do this select the Queue you want to search, enter your regular expression in the *Full Search* text box and hit the search button as shown:



In the example above we are searching for all messages containing the text "ResultDTO" on the Queue TESTQ4. All the message data except the MQMD will be searched, (this includes the RFH2 and any Dead Letter Headers etc.).

As you can see in the *Refreshing messages* dialog the number of messages found that match the search criteria are dynamically shown as the search progresses. This allows you to terminate the search, by closing the dialog or hitting the **Stop searching** button, at any time and view the messages found so far. If you were searching for only a few messages on a Queue containing hundreds of thousands of messages you can imagine what a time saving feature this could be!

QueueZee also provides a quick and easy way to search for messages using a date & time criteria by clicking on the calendar button as shown:



The calendar will always pop up with the current date and midnight selected.

If you wish to see all the messages from today just click **Ok**. If you wish to see messages between specific dates and times select the start date & time then click **End Date** to select the upper date & time band.

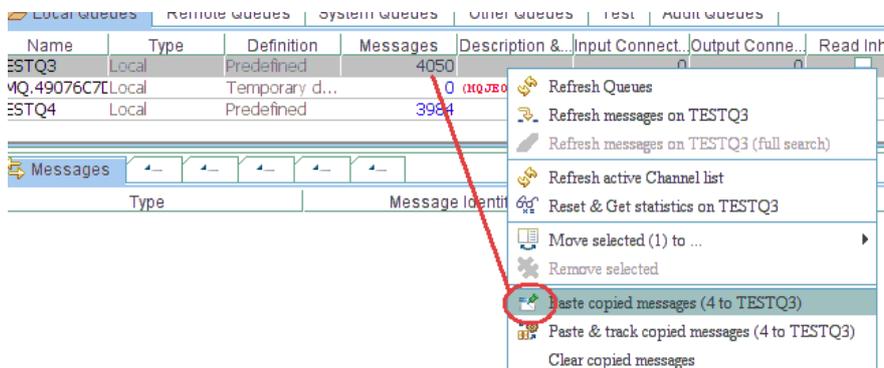
**NOTE:** All date searches will combine the date search criteria with any data in the full search text box, (described above), so make sure it is empty if you simply want just a date & time search.

Pasting messages

[Previous](#) [Top](#) [Next](#)

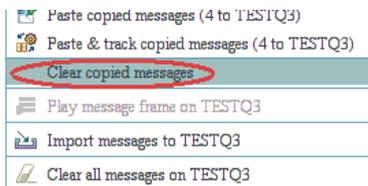
Any messages that have been copied to QueueZee's cache, (see [Messages](#)), can be pasted (put) on to any other Queue, (assuming the Queue Manager will allow it!).

Simply select the Queue you wish to paste the messages on, right click the mouse to bring up the Queue pop-up menu and select the paste copied messages option as shown:



Always check the log pane at the top of the GUI to make sure your actions were successful and that no errors occurred.

If you wish to clear the copied message cache completely just select the clear copied messages option from the Queue pop-up menu as shown:



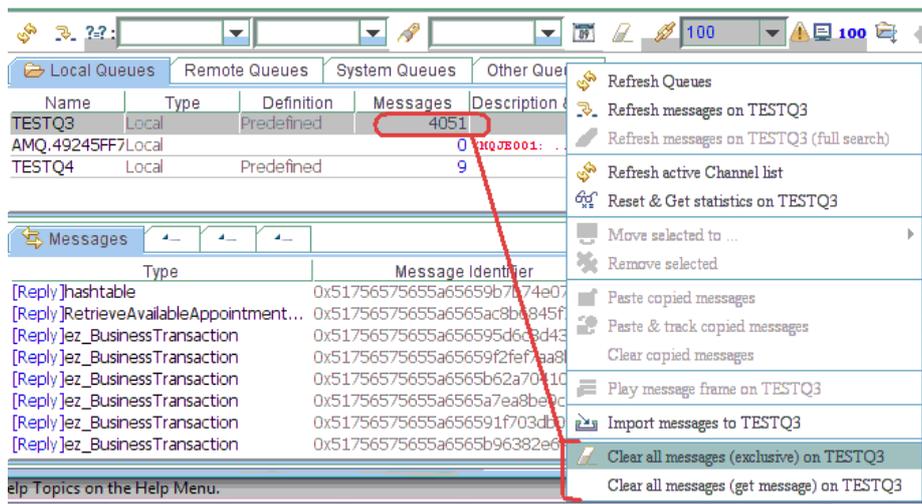
Messages may also be pasted and tracked which is a feature of the [SandPit](#).

Deleting, Importing & Remaining Features

[Previous](#) [Top](#) [Next](#)

Messages can be deleted selectively from Queues from the message pane and message pop-up menu, (see [Messages](#)).

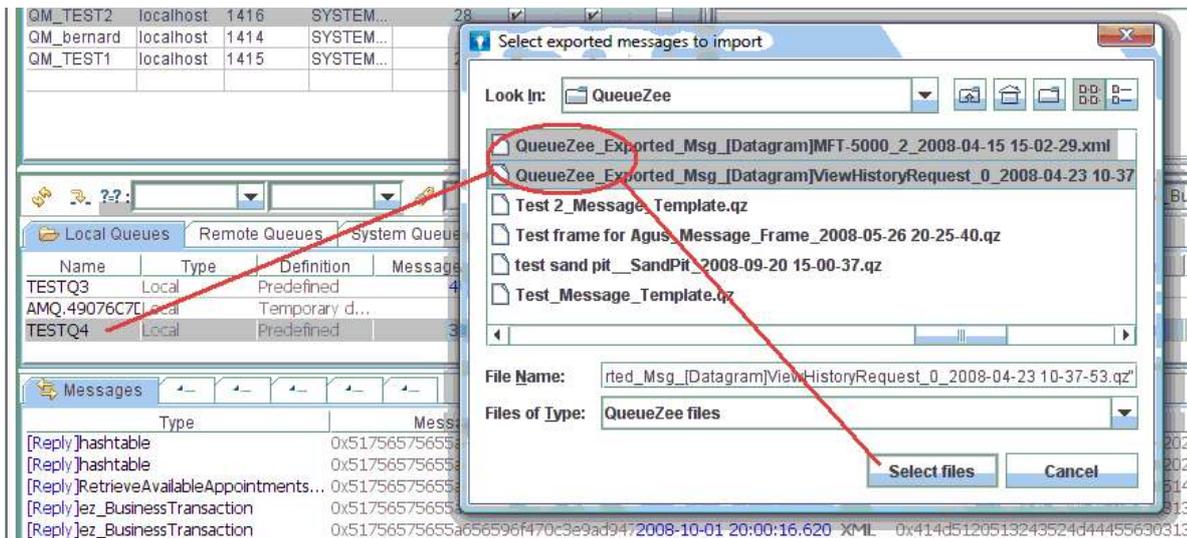
If, however, there are no open connections on a Queue and you wish to clear it of all messages this can be done easily and quickly from the Queue pop-up menu using the *exclusive* version of the *Clear all messages* option as shown:



Any errors will appear in the log pane if this operation fails and this **will** fail if QueueZee cannot get an exclusive connection to the Queue. To clear all the messages from a Queue with existing connections you should use the *get message* version of the *Clear all messages* option.

Any previously exported messages, (either in **XStream** or **QueueZee** message format) can be imported to a selected Queue.

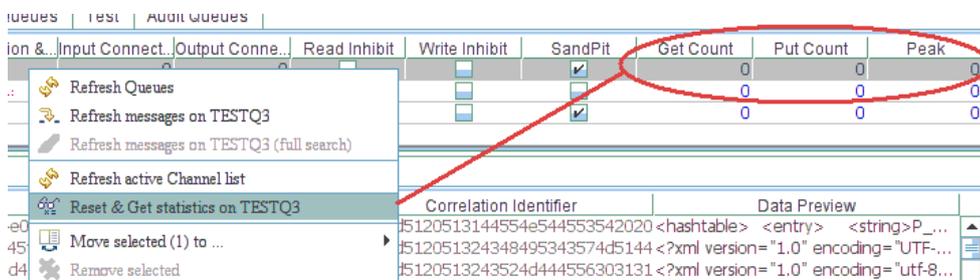
QueueZee will prompt you to select the exported message files which must contain only one message per file. Again this is achieved via the Queue pop-up menu as shown:



Don't forget to make sure the maximum message limit for the Queue you are importing to is sufficient when importing large amounts of messages.

Sometimes it is useful to see statistics for a particular Queue in terms of the peak message count plus number of messages added and removed, (put & get).

QueueZee provides a menu option from the Queue pop-up menu to update the statistic figures for a selected Queue by resetting the statistics thus providing figures accumulated from the last time the statistics were reset:



QueueZee will show all the Queues it can find on a Queue Manager some of which may be local transmission Queues.

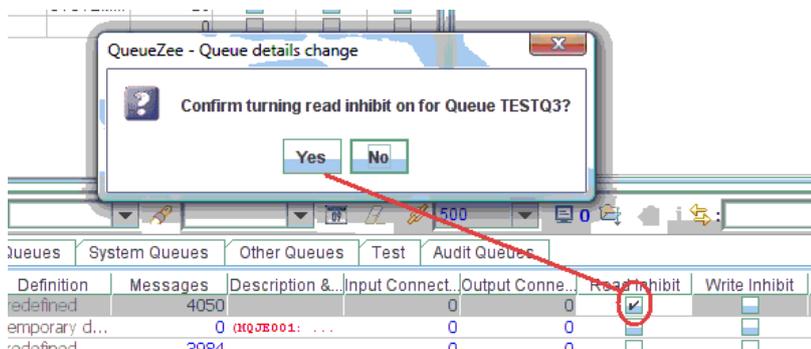
These Queues may have associated channels, (see [Messages](#)), and QueueZee will show any of these that are currently active, (ie. running, stopped, binding etc.). If a channel associated with a transmission Queue is not active QueueZee will not see or show it.

If you wish to see channels that may have become active since connection to the Queue Manager without re-connecting this can be done easily by selecting the refresh active channel option from the Queue pop-up menu as shown:



This will refresh all the active channels for the selected Queue Manager, not just the ones for the selected Queues.

You can also turn read and write inhibits on & off for any selected Queue. To do this simply check or uncheck the required check-box. QueueZee will always prompt you for confirmation as shown:

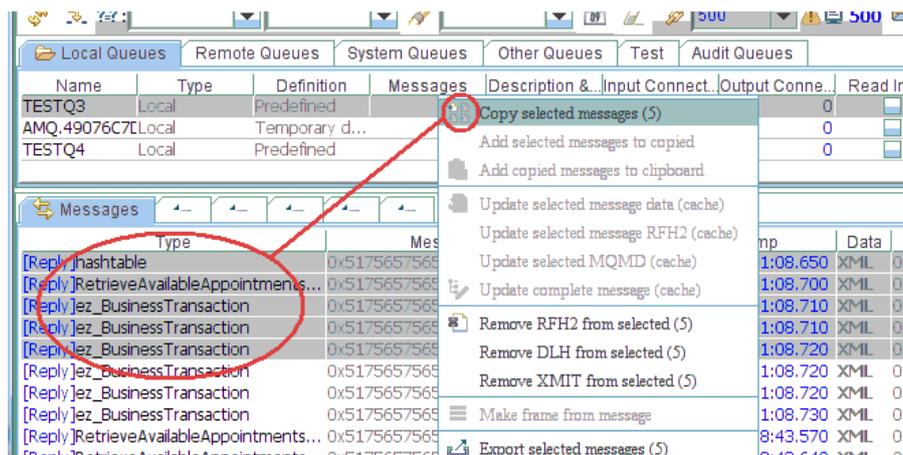


**NOTE:** Use this feature with caution as you may be stopping someone else reading their messages!  
General

[Previous](#) [Top](#) [Next](#)

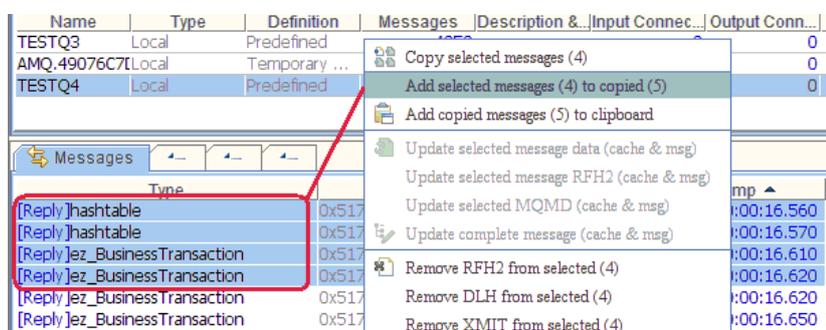
Messages are refreshed (loaded) and displayed as explained in the [Queues](#) section of this document.

To copy a set of messages to QueueZee's message cache simply select the messages you want then right click to bring up the message pop-up menu and select the *Copy selected messages* option as shown:



Messages can be selected using the standard Ctrl keys, (ie. Ctrl-a to select all of them).

If you wish to copy messages from several different Queues you can just keep adding messages to the ones already copied, as shown:



Once you have messages copied to the cache you can copy them to another Queue, add more messages to the copied ones from another Queue or copy them to the clipboard for pasting into an external document etc.

All copied messages can also be copied to the clipboard by selecting the *Add copied messages to clipboard* option as shown above. These can then be pasted into any external document or application. The detail of what's copied from the messages and how it's formatted can be set via the [Preferences](#).

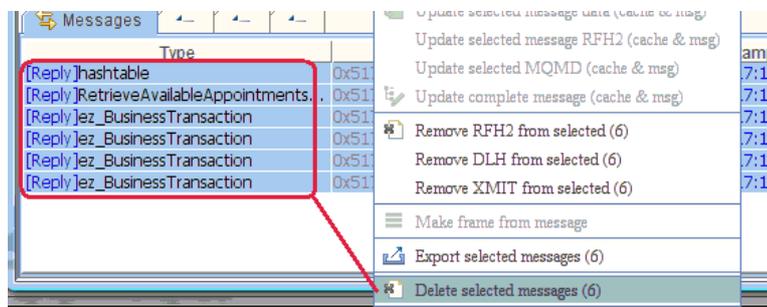
You might want to export messages so you can archive them or move them to another computer for more testing or analysis. QueueZee supports two types of message export format; it's own and an XML representation of the MQ message class via the XStream Java classes. Both can be edited in a normal editor once exported, (if you know what you're changing!). We recommend that generally you don't attempt this though and also recommend the XStream XML format for exports as they are a faster, more efficient way of exporting/importing. (XStream is the default format).

To export selected messages simply select the *Export selected messages* option from the Messages pop-up menu as shown in the first diagram in this section.

QueueZee also lets you delete selected messages (non-browse read) from a Queue without interfering with any messages copied in the message cache. The obvious advantage of this is you can copy a set of messages you wish to paste multiple times for testing while freely deleting other sets of selected messages.

You could even delete the real messages you have copied then paste them back again from the message cache, (why you would want to do this deliberately though I don't know!).

To delete selected messages select the *Delete selected messages* option from the Messages pop-up menu as shown:



Message Data

[Previous](#) [Top](#) [Next](#)

Whenever a message is selected in the messages pane all its associated headers and data are shown in the tabbed pane to the right of the Queue Manager pane.

The first tab shows the message data. If the message contains headers QueueZee doesn't recognise they will also be shown in the data pane. For example, a message containing some simple XML data might look like this:

```

(Data length - 376 bytes)
<?xml version="1.0" encoding="utf-8"?>
<ez:BusinessTransaction xmlns:ez="http://www.easynet.net">
  <ez:BusinessTransactionHeader operatorId="001" operatorTransactionId="12345"/>
  <ez:BusinessTransactionBody>
    <ez:ValidateMac directoryNumber="012345678" code="876543210"/>
  </ez:BusinessTransactionBody>
</ez:BusinessTransaction>

```

500 [500] : 000166 - [Reply]ez\_BusinessTransaction

One powerful feature of QueueZee is it allows you pretty much free editing of the message data, RFH2 and MQMD.

For example, if you wanted to change the **operatorId** and add a **ez:CustomerID** to the XML shown you can just go ahead and edit it as shown:

```

(Data length - 376 bytes) - (90 bytes of Text data selected)
<?xml version="1.0" encoding="utf-8"?>
<ez:BusinessTransaction xmlns:ez="http://www.easynet.net">
  <ez:BusinessTransactionHeader operatorId="999" operatorTransactionId="12345"/>
  <ez:CustomerID>112233</ez:CustomerID>
  <ez:BusinessTransactionBody>
    <ez:ValidateMac directoryNumber="012345678" code="876543210"/>
  </ez:BusinessTransactionBody>
</ez:BusinessTransaction>

```

500 [500] : 000166 - [Reply]ez\_BusinessTransaction

As you can see the number of bytes of the selected text plus the type of data selected is also shown on the header bar of the data pane. If the text selected was valid XML QueueZee would tell you this by showing "90 bytes of XML data selected" instead. This is an easy way to check if a section of a message, (or indeed the whole message), is valid XML or not.

Now the data is changed QueueZee recognises this and allows you to change the message data in the cache and (optionally) also on the Queue itself, (depending on your [Preferences](#)). This can be done via the message pop-up menu as shown:

Name	Type	Definition	Messages	Description & ...	Input Connect.	Output Conne..	Read Inhibi
STQ3	Local	Predefined	4050				
IQ.49076C7E	Local	Temporary d...	0	(MQMD)001: ...			
STQ4	Local	Predefined	3984				

Type	Message Identifier	Read Inhibi
aplyJhashtable	0x51756575655a65659b7b74e07...	x414
aplyJRetrieveAvailableAppointments	0x51756575655a6565ac8b6845f...	x414
aplyJez_BusinessTransaction	0x51756575655a656595d6c8d44...	x414
aplyJez_BusinessTransaction	0x51756575655a65659f2fef7aa8...	x414
aplyJez_BusinessTransaction	0x51756575655a6565b62a70410...	x414
aplyJez_BusinessTransaction	0x51756575655a6565a7ea8be9c...	x414

After the message has been updated in the message cache it can be copied & pasted to another Queue, (or indeed the same Queue if that's what is desired). If you have your [Preferences](#) set to update messages in the cache and on the Queue you may have also been prompted for confirmation before the message was updated.

When QueueZee updates a message on the Queue it actually updates it in the cache first (this is always performed), writes the new edited message to the Queue then deletes the old one, (by reading it off the Queue by its message identifier).

If the deletion fails for some reason like the message has already disappeared from the Queue you will see an error in the log pane indicating this. In this particular scenario it isn't a problem as the new, updated message is still written to the Queue.

You should always carefully check the relevance of any errors in the log. Often they are merely indicating failures that are expected or not serious, (such as trying to browse a remote Queue!).

You can view, (not edit), any message in a hexadecimal format in addition to the normal text mode. This can be useful for viewing binary messages or pinpointing rogue non-ASCII characters in XML documents etc. To do this right click in the message Data pane to bring up the Data pane pop-up menu and select the *Show as hexadecimal* option:

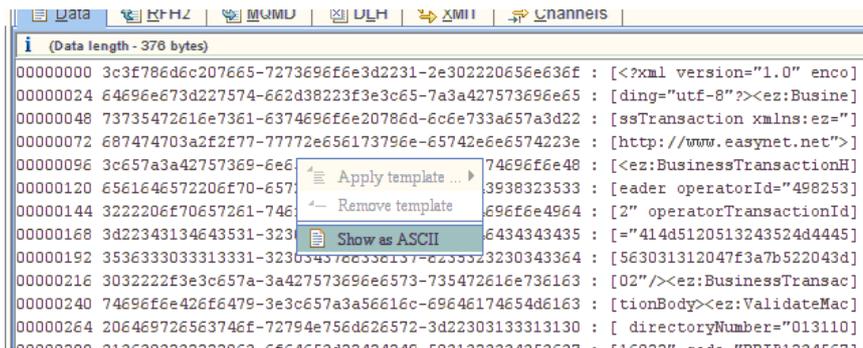
```

(Data length - 376 bytes)
<?xml version="1.0" encoding="utf-8"?>
<ez:BusinessTransaction xmlns:ez="http://www.easynet.net">
  <ez:BusinessTransactionHeader operatorId="4982532" operatorTransactionId="414d5120513243524d4445563031312047f7b59943490u">
  <ez:BusinessTransactionBody>
    <ez:ValidateMac directoryNumber="BBIP12345678/12JH1"/>
  </ez:BusinessTransactionBody>
</ez:BusinessTransaction>

```

1010 0101 Show as hexadecimal

The message pane will then switch to look something like this:



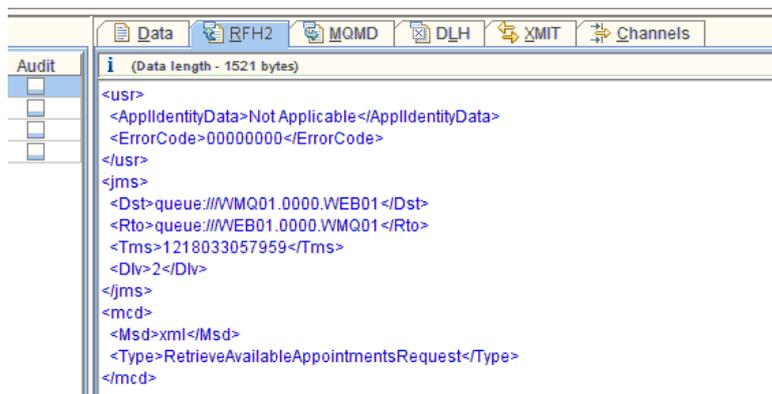
To switch back to the textual view where you can edit the data select the (changed) menu option of *Show as ASCII*.

Templates can also be applied to the message Data pane where the data can be edited. This is explained in the [Message Templates](#) section of the help.

RFH2 Header

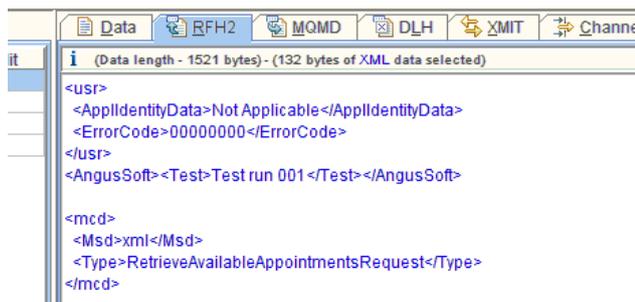
[Previous](#) [Top](#) [Next](#)

When a message is selected any RFH2 data will be displayed in the RFH2 pane to the right of the message data pane:

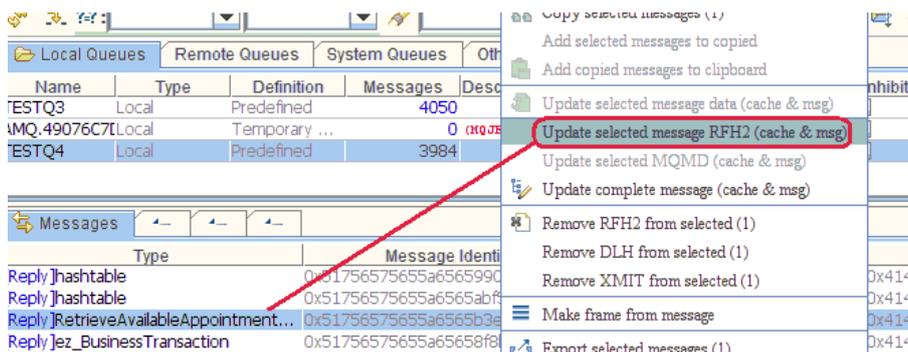


The RFH2 folders can be edited in the same way you can edit the message data pane. QueueZee will validate any folder edits for valid XML ignoring the complete edit for any invalid XML data.

For example, if you wanted to remove the `<jms>` folder and add an `<AngusSoft>` folder with a single element within it called `<Test>` your edit could look like this:



The format, (ie. spaces & new-lines), are not important as QueueZee will tidy that up for you when a new, edited message is written to the Queue. As long as it's valid XML that's all that matters. To change the edited message on the Queue make sure you have *Updates in cache & on Queue* selected in the [Preferences](#), right click off the message in the Messages pane and select the *Update selected message RFH2* option:

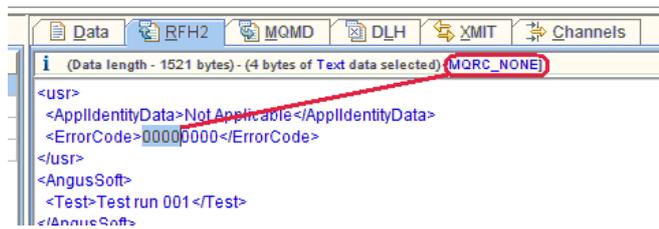


Click **Yes** when you are prompted for confirmation and the new message will be written to the Queue, the old one deleted and the message refreshed. Assuming you have your *Preferences* set to create a new message identifier and timestamp your new message should be the latest one on the Queue when sorted by

message timestamp, (your message will always be at the end of the Queue in terms of message ordering).

To remove the RFH2 header from a message simply delete all the folders or select the *Remove RFH2 from selected* option from the message pop-up menu. RFH2 headers can also be created by just adding new folders, (QueueZee will let you add whatever folders you want so that doesn't guarantee the RFH2 is valid for other systems and applications! - it will, however, always be a valid structure for an MQ message).

In the same way as is shown for the message Data pane the number of bytes and type of data selected is shown on the pane header bar. A neat feature available in this pane, the Data pane and the log pane is the ability to show MQ error codes if the selected text is a valid MQ error number, as shown:



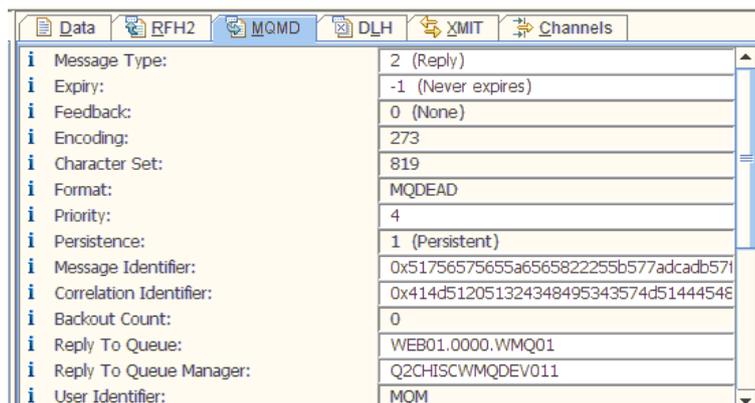
An option from the RFH2 pop-up menu is provided to automatically add a `<QueueZeeUuid>` folder which will contain just a UUID string. This may be useful for manually tracking or tagging messages when there are many similar or the same making them difficult to identify:



## MQMD Settings

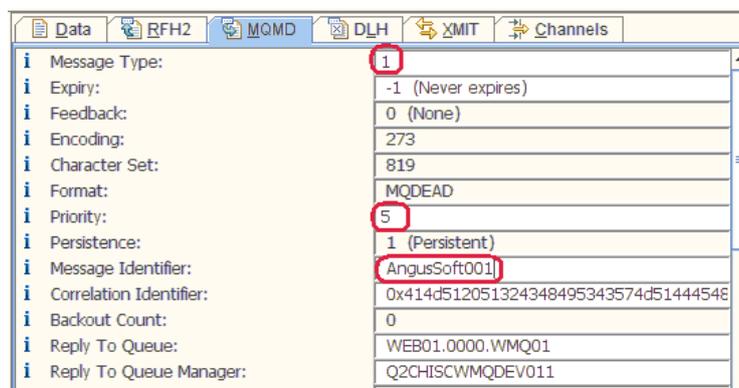
[Previous](#) [Top](#) [Next](#)

When a message is selected the MQMD (MQ Message Descriptor) values will be displayed in the MQMD pane to the right of the message RFH2 pane:



QueueZee allows you to edit certain values in the MQMD so you can update or copy and paste a new message to another Queue. The values you can change appear in the fields with a white background.

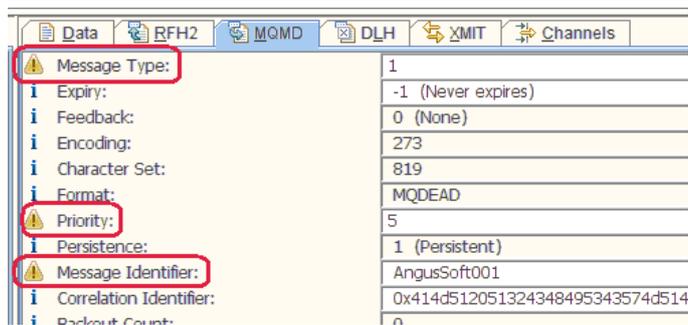
For example, to change the above example to a request message with a priority of 5 and a message identifier of "AngusSoft001" the edited fields would look like this:



Note QueueZee will let you edit any message identifier as either plain text or a hexadecimal string, (which must start "0x"), and will work out automatically which it is you've entered. It will also automatically pad out the remainder of the identifier with zeros if it is too short, or indeed truncate it if it's too long!

To change the message we've just edited in the message cache simply select *Update selected MQMD* from the Messages pop-up menu, (and select **No** if you have your [Preferences](#) set to update the message on the Queue as well).

The MQMD pane will now show the fields that have been changed by putting a yellow warning triangle next to the field name tag:



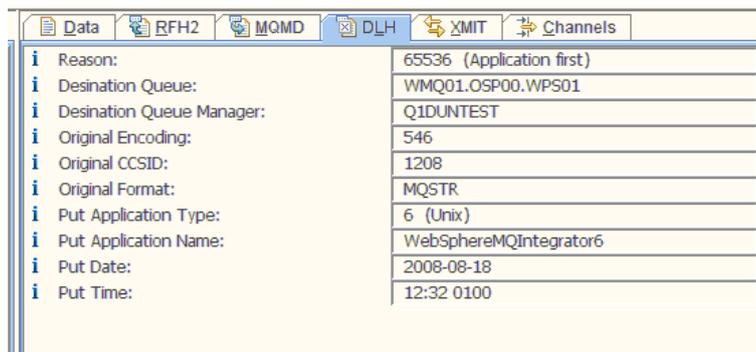
This message can then be copied so the a message with the new values can be pasted onto another Queue or copied to the clipboard. The message will stay changed in the message cache until either the Queues are refreshed or the messages are next refreshed on that Queue.

As the MQMD is an integral part of all MQ messages it obviously cannot be removed!

#### DLH - Dead Letter Header

[Previous](#) [Top](#) [Next](#)

If a selected message has an associated Dead Letter Header this will be shown in the DLH pane to the right of the MQMD pane in the tabbed message data window:

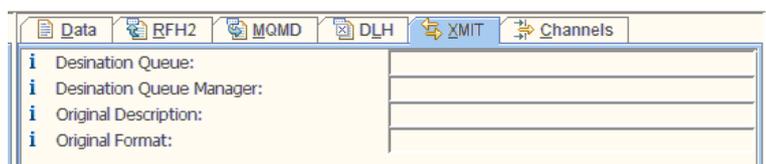


Dead Letter Headers can be removed from messages via the Messages pane pop-up menu, (this feature also supports multiple selection).

#### XMIT - Transmission Header

[Previous](#) [Top](#) [Next](#)

If a selected message has an associated Transmission Header this will be shown in the XMIT pane to the right of the DLH pane in the tabbed message data window. If the message does not contain a Transmission Header the pane will just show blank fields when selected:

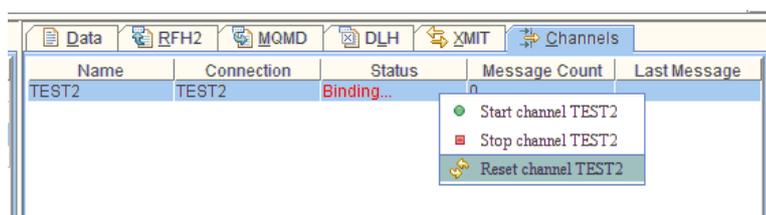


Transmission Headers can be removed from messages via the Messages pane pop-up menu, (this feature also supports multiple selection).

#### Channels

[Previous](#) [Top](#) [Next](#)

When a transmission Queue is selected in the Queue pane and it has an active, (any status other than *inactive*), channel associated with it, the channel details will be shown in the Channels pane to the right of the XMIT pane:

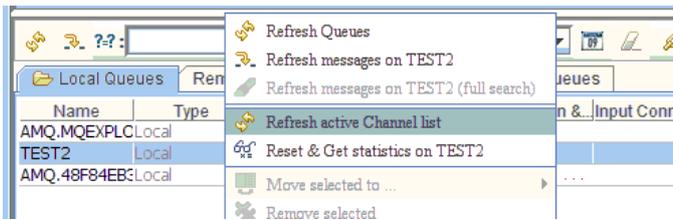


Each time the selected Queue is clicked on the channel status, message count and last message timestamp will be updated.

Basic administration of the channel can also be performed from QueueZee via the channel pop-up menu as shown above. This is especially useful when running tests where you want to stop and start a channel to see messages on a transmission Queue before they get transmitted.

Using QueueZee you can put both the Queue you're pasting test messages to and the channel you want to start and stop in the same folder as long as they're both on the same Queue Manager.

When you connect to a Queue Manager QueueZee will look for any active channels associated with transmission Queues. Sometimes a channel you're interested in may be inactive when you connect to the Queue Manager and become active during your testing. To make QueueZee aware of the new active channel without disconnecting and reconnecting to the Queue Manager you can simply select the *Reset active Channel list* option from the Queue pop-up menu from any selected Queue, as shown:



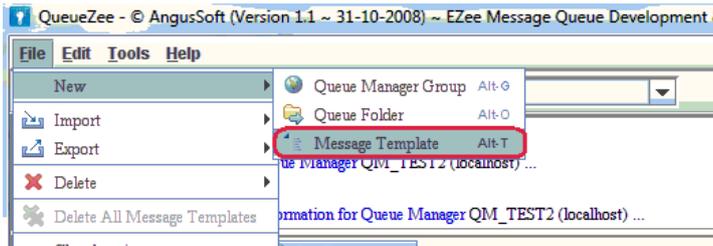
Message Templates

[Previous](#) [Top](#) [Next](#)

Many messages you view in QueueZee are likely to be XML or a similar readable format. Some, however, may be a continuous lump of data or fixed length text. This can obviously not always be the easiest message to read, especially if it's rather large.

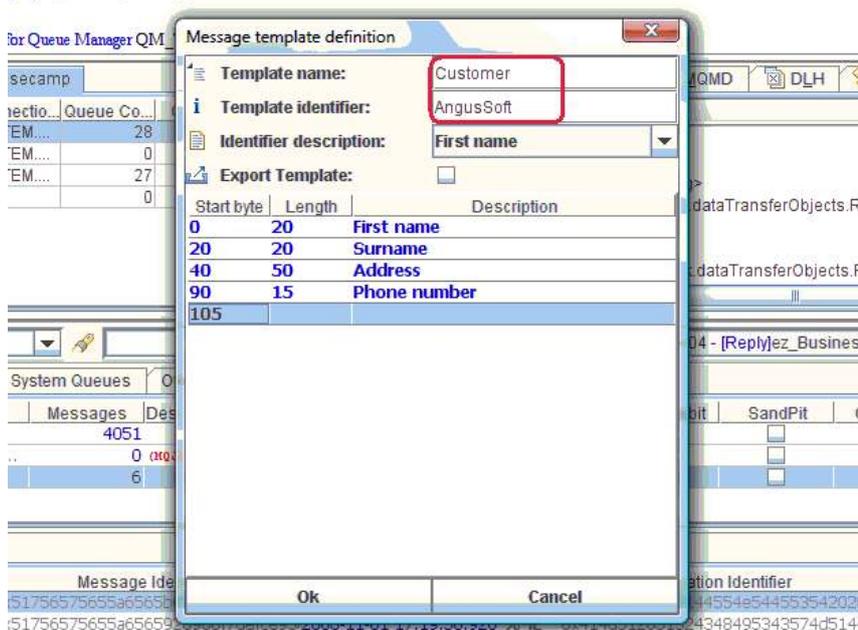
To help solve this and make things a lot easier you can create Message Templates in QueueZee which can be applied in the message Data pane so certain messages can be instantly read without trying to work out which are the individual fields in a big lump of text!

To create a Message Template first select **File->New->Message Template** from the main menu:



You will then be presented with the Message Template dialog.

For this example lets create a simple template that consists of only a **First name**, **Surname**, **Address** & **Phone number**. They will each have field length of 20, 20, 50 & 15 characters respectively. After entering this the dialog would look like this:

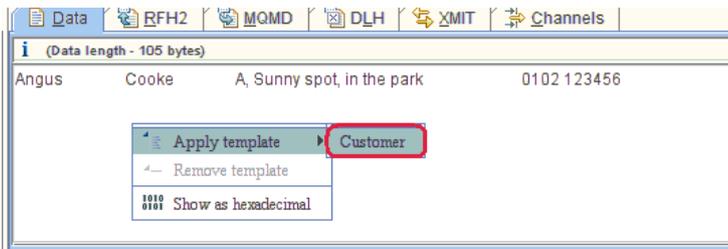


Notice I have called the template *Customer* and the template identifier *AngusSoft*. The template identifier and identifier description drop-down menu are not used in QueueZee v1.x but you **must** set them along with the template name - this is mandatory! They can be set to anything you like as long as they don't conflict with an existing template name.

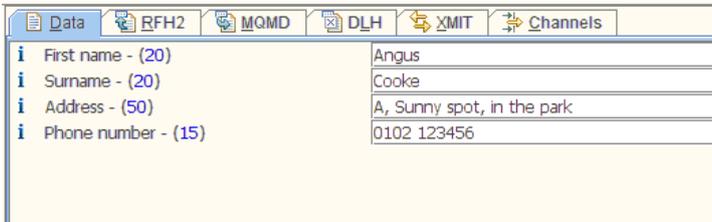
The identifier feature will be available in v2.x and will allow you to specify a field and identifier string so QueueZee can automatically recognise messages that fit this template and apply it when the message is viewed. If you wish to export the message when you create it then the *Export Template* box should also be checked.

Now we've created a template just click **Ok** to save it as part of your configuration, (if you exported it you can safely delete it and re-import it when needed). There are no Message Template editing features available in QueueZee v1.x so if you make a mistake creating a field or wish to change an existing template you must cancel or delete the current template first then create a new one. You could also edit an exported template in a normal editor if you know what you're doing.

To apply our new template to a message simply select a suitable message from the Messages pane and right click in the message Data pane to bring up the pop-up menu and select the apply template option:



Once selected the message will be shown like this:



Although a very simple example you can see how much easier this is to read.

To remove the template bring up the pop-up menu again and select the *Remove template* option. The message can be edited and updated in the template in the same way you can edit the [MQMD](#). Remember you will only be editing the portion of the message the template is showing and if you remove the template before updating the message in the cache the original message data will be reapplied when you remove the template, (this works both ways, ie. the original message will also be shown in the template if you have edited it in the normal view but not updated it!).

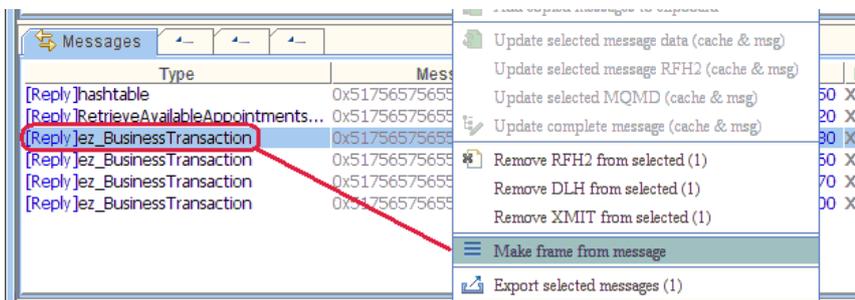
Message Templates can be exported, imported and deleted in the usual way from the main **File** menu. It may be useful to create templates for certain projects or environments, export them then only import the set you need for the tests or development you're working on.

[Previous](#) [Top](#) [Next](#)

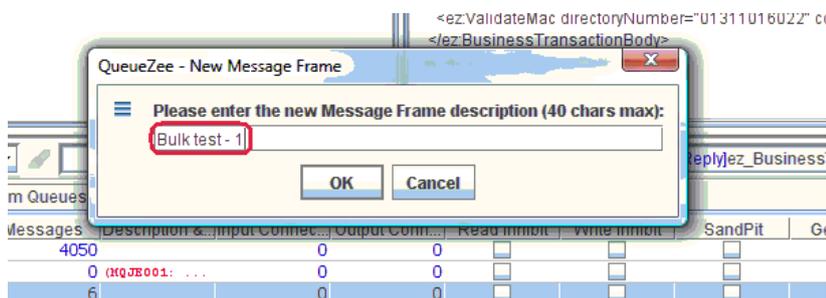
When a particular message is selected you can create a message frame from that message.

Message frames provide a way to store all message details except the message data (body). This can make it easy to put multiple messages with varying message data onto a single Queue keeping all the header information the same.

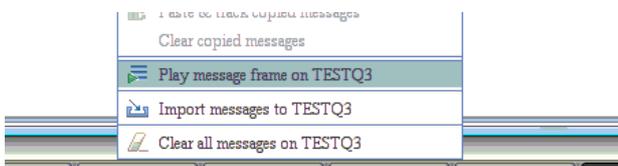
For example, you may want to performance test a certain environment with a bulk message drop onto one Queue using a sample set of message data but keeping the message headers the same. First you would select your initial sample message as shown:

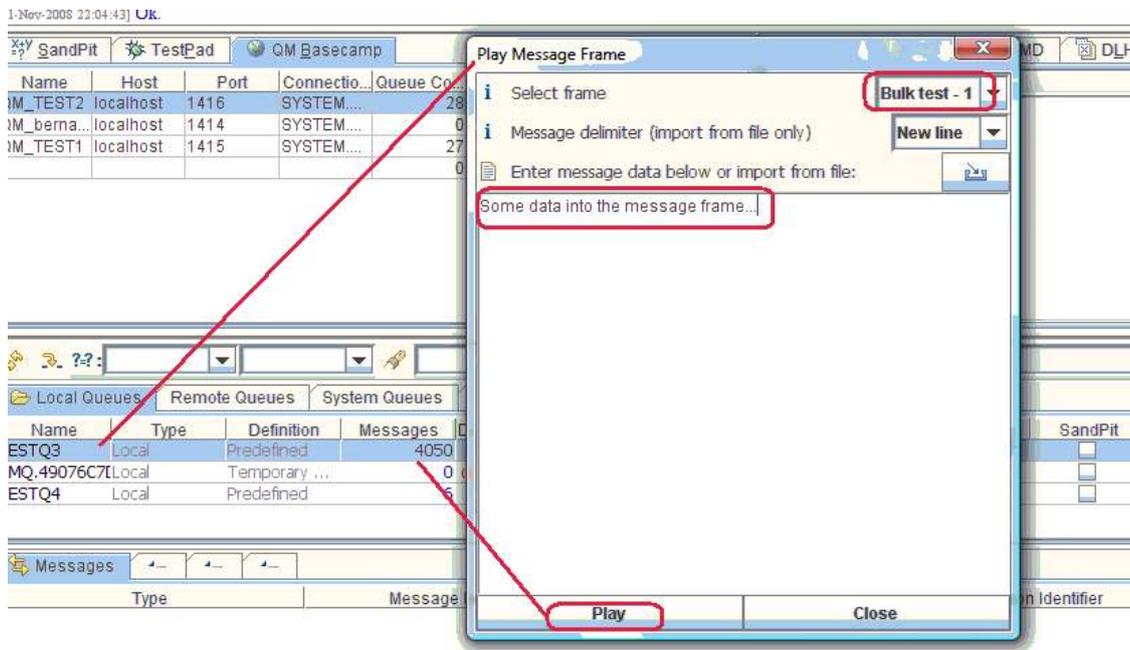


Lets call our message frame *Bulk test - 1*:



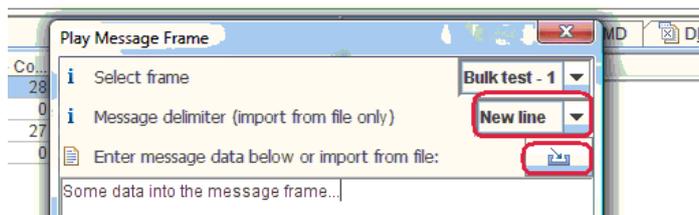
This will save all the header details of the selected message to your configuration as message frame "*Bulk test 1*". This message frame can then be used to play various different message data through the message frame (using the same saved header details) onto any Queue you choose, as shown:





When the message frame dialog is displayed you can either type or copy any data you like into the text input panel and click the **Play** button to put a new message on the Queue you have selected or take message data from a file, (shown in the following diagram). Each time you click the **Play** button QueueZee will put a new message on the Queue with the text data contained within the text input panel.

For our bulk performance test it would much more efficient to have the message date contained in a file or multiple files. You can do this by first selecting whether each file you specify should read message data as the whole file (**End of file**) or a new message for every line (**New line**) in the file, then click the file import button, select your file(s) and let QueueZee do the rest:



Message Frames can be exported, imported or deleted from the main **File** menu. This means you can distribute frames you've made to other QueueZee users and don't have to keep all your message frames in your main configuration at once. It may be useful to organise sets of frames per test scenario or environment.

The SandPit

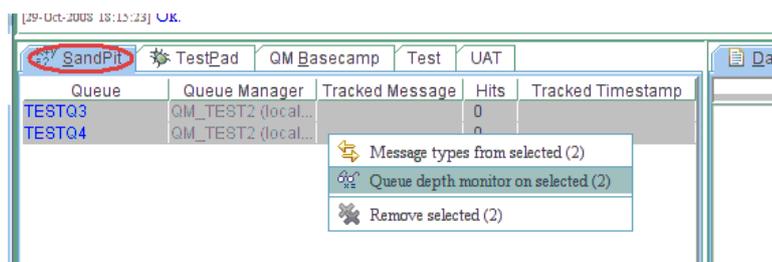
[Previous](#) [Top](#) [Next](#)

The SandPit is a place where you can put many Queues from different Queue Managers for various special features within QueueZee. Once a Queue is marked as in the SandPit and initially displayed it will always be shown, even if its Queue Manager connection has disappeared. If a Queue in the SandPit belongs to a Queue Manager that isn't connected it will be ignored for any SandPit operations. To re-activate it simply reconnect to the Queue Manager it resides on.

Queues are copied to the SandPit by simply checking their *SandPit* check-box as shown:

Description & Input Connect...	Output Conne...	Read Inhibit	Write Inhibit	SandPit	Get Co
50	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
0 (MQJE001: ...	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
34	0	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

To use the SandPit features with the Queues you have selected simply select the *SandPit* tab in the Queue Manager pane and select the Queues you want to use then right click the mouse to bring up the SandPit pop-up menu:



In the example above you can see we've selected the Queue depth monitor option. This is especially useful for monitoring Queue depth activity for many Queues across many different Queue Managers. The Queue monitor dialog looks like this:

Queue	Messages	Activity	Avg Get Count	Avg Put Count	Trough	Trough Time	Peak	Peak Time	Input Conn.	Output Conn.
TESTQ4 (QM_TEST2)	6 = Static		0	0	6	02-Nov-2008 10:23:05	6	02-Nov-2008 10:23:05	0	0
TESTQ3 (QM_TEST2)	4051 = Static		0	0	4051	02-Nov-2008 10:23:05	4051	02-Nov-2008 10:23:05	0	0

[Started: 02-Nov-2008 10:23:05] - [Refresh interval: 4 (seconds)] - Last refresh: 02-Nov-2008 10:23:13

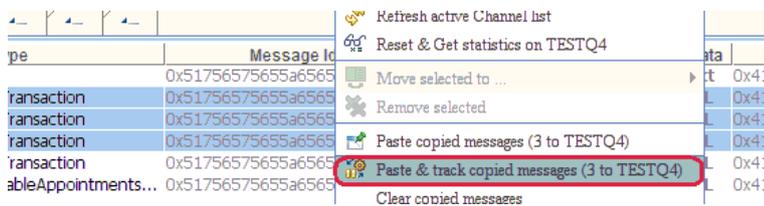
Refresh Close

By default QueueZee will monitor and update the shown statistics for each Queue every 4 seconds, (this can be changed in the [Preferences](#)). The *Activity* column will show whether the messages on the Queue are *Static*, *Dropping* or *Climbing*. Obviously if all the activity for all the Queues stays static there is probably not a lot happening! The *Average Get Count* & *Average Put Count* reflects the average activity between monitor intervals for messages added and removed from the Queue. If, for example, 10 messages were put on a the Queue and 10 removed in one interval the counts would not change as the put/get difference is zero.

This feature is particularly useful for monitoring Queues during long running tests where you want to get some idea of the message throughput and how busy the Queues got.

The *Message Types* option from the SandPit pop-up menu works in exactly the same way as the message distribution drop-down on the Queue pane toolbar but across all the selected Queues rather than a just one, (the drop-down for this is on the top toolbar below the main menubar).

Another very useful feature of the SandPit is tracking messages that have been pasted on a Queue from the Queue pane pop-up menu:



Once the copied messages have been pasted QueueZee will attempt to track them by monitoring all the Queues in the SandPit for a specified time period that's set in your [Preferences](#), (the default is 2 seconds).

If the message is found on any of the SandPit Queues during this time QueueZee will automatically switch to the SandPit at the end of the time interval and show where the hits were of all the messages it was tracking, (ie. which Queues they appeared on within the tracking time interval):

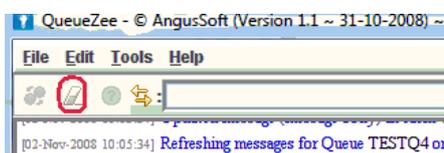
[02-Nov-2008 10:47:11] Pasted 3 message(s) to Queue TESTQ4 on Queue Manager QM\_TEST2 (localhost)

Queue	Queue Manager	Tracked Message	Hits	Tracked Timestamp
TESTQ3	QM_TEST2 (local...		0	
TESTQ4	QM_TEST2 (local...	[Reply]ez_BusinessTransaction	6	02-Nov-2008 10:47:12

As you can see in our example we got two hits per message on the target Queue during the 2 second time interval. This was a very simple example where we already knew where the message was going but you can imagine how valuable this feature could be where you have a message in a large enterprise system passing through many Queue Managers to a remote application and back - not the easiest thing to track manually when the message isn't going where you expect. If QueueZee doesn't switch to the SandPit after you've pasted your message(s) you at least know where the message isn't going!

Queues can be added and removed from the SandPit from the Queue pane, (or also removed from the SandPit pop-menu). SandPits can be also be exported and imported in the usual way from the main **File** menu if you have different sets you would like to use for different tests or environments.

You can quickly clear (remove) the current SandPit by clicking on the *Clear sandpit* button on the main toolbar:



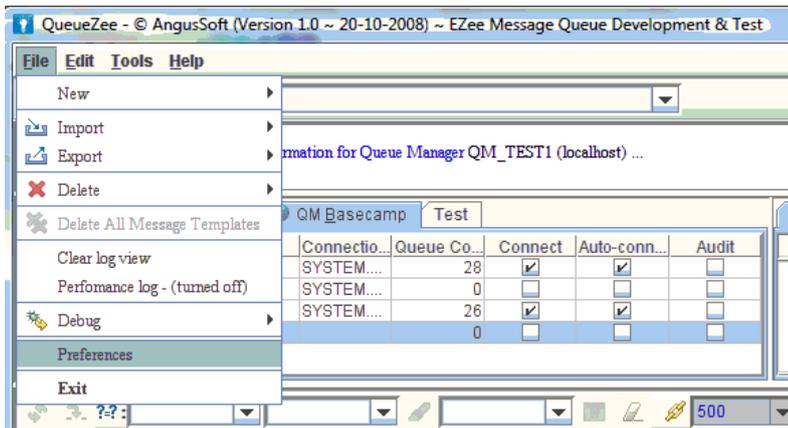
As already stated if you attempt any SandPit operation with Queues that belong to disconnected Queue Managers or Queues that cannot be browsed (remote or read inhibited), QueueZee will ignore them, (and possibly show an error in the log view).

General

[Previous](#) [Top](#) [Next](#)

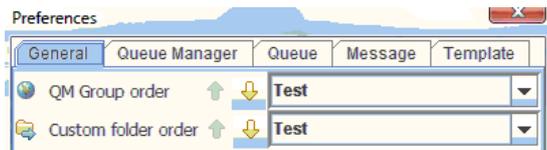
Many aspects of the way QueueZee behaves can be customised and modified from the Preferences dialog. Any preference changes made will only be applied if the **Ok** button is clicked.

The dialog is selected from the main **File** menu as shown:



The *General* tab allows you to change the following:

If you have many custom Queue Manager Groups and/or custom Queue Folders you can re-order them by moving them up or down the list as shown:



The new ordering will not take affect until QueueZee is restarted.

By default QueueZee will only save new configuration changes to the configuration file when the application is closed.

To force configuration changes to take place after each change you can check the last option in *General* preferences as shown:

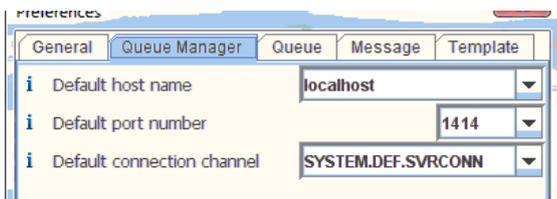


QueueZee will automatically maintain a backup copy of your configuration file in its home directory called **config.qz.bak**. It will revert back to this if it fails to update the main configuration file (**config.qz**) for any reason.

**Note:** Don't forget you must select the **OK** button to exit the Preferences dialog window for your changes to take affect.  
Queue Manager

[Previous](#) [Top](#) [Next](#)

When defining Queue Manager definitions QueueZee will automatically fill in defaults for missing information taken from the preferences. These can be changed if required as shown:



The default Queue Manager port number must be a valid integer, (QueueZee will default back to **1414** if it isn't!).

**Note:** Don't forget you must select the **OK** button to exit the Preferences dialog window for your changes to take affect.  
Queue

[Previous](#) [Top](#) [Next](#)

By default QueueZee will show all available Queues on a connected Queue Manager, including temporary Queues. Often it is useful to hide temporary Queues if they are not of interest and there are many of them. This can be achieved by checking the *Hide temporary Queues* option in the *Queue* tab:



By default QueueZee will not refresh a full search after all filters have been cleared.

This is to stop potentially unwanted long refresh times when performing many searches between filter clears, (even though you can stop a refresh at any point). To force a message refresh after a filter clear the following option can be checked:



The default refresh interval for Queue monitoring (available from the [SandPit](#)) is 4 seconds. This can be changed by selecting a new interval from the following drop-down:

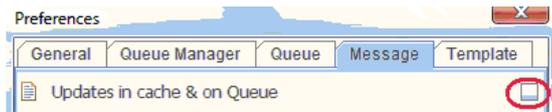


**Note:** Don't forget you must select the **OK** button to exit the Preferences dialog window for your changes to take affect.  
Message

[Previous](#) [Top](#) [Next](#)

The *Message* tab in the Preferences dialog allows you to set many different aspects of behaviour when QueueZee is performing operations related to messages. These are all shown and described below.

Update a message in the QueueZee message cache and on the Queue it resides on:



This option, when checked, will update an edited message in QueueZee's message cache and on the actual Queue by creating a new message containing the changed data then deleting the original. If this option is checked and you just want to update a message in the cache for a particular change you can just respond **No** to the confirmation question dialog asking you to confirm updating the message on the Queue, (this will simply update the message in the message cache).

**Note:** bear in mind this option can produce unpredictable results if selected and you are changing messages that have duplicate message identifiers on the same Queue!

Create a new "put message" timestamp for newly created messages:



This is selected by default and means QueueZee will create a new timestamp for every new message it creates, (typically from a message copy & paste). This can be unchecked if you wish to preserve the message timestamp for any reason.

Refresh messages on the selected Queue when QueueZee starts up:



Check this option if you want QueueZee to automatically refresh messages on the Queue you have selected when it starts up. This is useful if you maybe have a selection of test messages on a particular Queue you are always selecting from when running a series of tests. This option is not selected by default.

Time (in seconds) to spend tracking messages in the [SandPit](#) when a message is pasted with tracking:



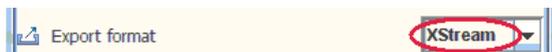
This option is set to 2 seconds by default and can be set to anything from 1 second to 240 seconds. This is the time QueueZee will take polling the Queues in the SandPit for a message you have pasted with tracking activated.

Which parts of a copied message are copied to the clipboard when requested, (including whether separators are also included):



These two options determine which parts of a message are copied to the clipboard when requested and whether separators are added or not. The message body & RFH2 plus separators are copied by default.

Format for exported messages:



This option selects which format exported messages are written in. For version 1.x QueueZee supports two formats, XStream & it's own format. By default the XStream format is selected which writes the exported messages out as a complete structure in a readable XML format. Although possible it is not recommended this be edited for future imports unless you know for certain what you're changing!

Which parts of the message RFH2 header should be checked to identify a message, (including in which order they are checked):

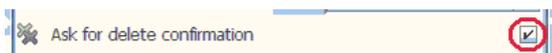


By default QueueZee will attempt to identify messages as different types by looking at the message data. This has no relevance other than making it easier to look at groups of messages.

You can further enhance this identification process for message with RFH2 headers by telling QueueZee to look at certain fields in a particular order contained in the RFH2. Up to 5 folder & tag combinations can be entered in the above option and they will be searched for in the order they are entered. You can clear & reset all these entries by clicking the clear button to the left of the identifier text box drop-down.

For example, if you want QueueZee to look at the **Type** tag in the RFH2 **mcd** folder first to identify a message you can type **mcd%Type** followed by <Enter> in the text box above. When you then click the drop-down button you will see QueueZee has remembered this as the 1st place to take the message type from if it exists.

Ask for confirmation when deleting selected message from a Queue:



This option is checked by default. If you don't wish to be asked for confirmation when deleting messages from a Queue uncheck this option. It is recommended this option is left checked unless unselecting it is really necessary!

Ask for confirmation when updating a message on the Queue as well as in the message cache:



This should be checked when a confirmation dialog is required for updating a message on the Queue as well as in the message cache. By default this is checked and recommended it remains so! If the message update on the Queue is not confirmed it will still updated in the message cache.

Clear the message cache and refresh after deleting selected messages:



When selected QueueZee will clear the message cache for the selected Queue and refresh the message after messages have been deleted. Turn this off if you wish to leave deleted messages shown and in the cache after they have been deleted - only recommended if you have a habit of accidentally deleting messages you didn't mean to! This is selected by default.

Create a new message identifier for every new message created:



If selected QueueZee will create a new message identifier for every new message it creates, (typically as a result of a copy & paste). Turn this off if you wish to preserve the message identifier when creating new messages from copied ones. This may be useful if a particular test is sensitive to maintaining the same message identifier.

Display XML message data as nicely formatted text:



If selected QueueZee will always attempt to show XML message data, (in the message Data pane), as nicely formatted text. It will even do this if the XML data is invalid thus allowing the user to easily spot what is wrong with a badly formed XML document. Although QueueZee will not alter the XML text data when doing this it will remove and adjust any new-line characters to produce correct formatting. Therefore it is recommended you turn off this option if you want to analyse the raw message data as it is stored on the Queue. This option is selected by default.

Show message identifiers as a readable ASCII string:



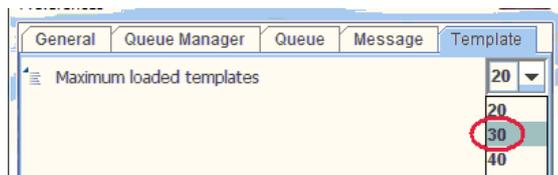
This option is off by default and QueueZee will show all message identifiers as hexadecimal strings, (ie. 0x01a245ff etc.). Select this option if you want to see message identifiers as pure ASCII strings - this is useful if you are looking at messages with text readable message identifiers.

**Note:** Don't forget you must select the **Ok** button to exit the Preferences dialog window for your changes to take affect.

## Template

[Previous](#) [Top](#) [Next](#)

One aspect of message templates can be changed in version 1.x. This is the amount of templates that can be loaded at any one time as shown:



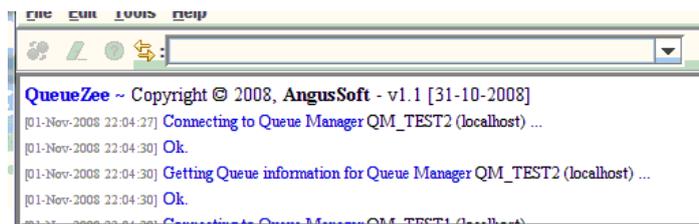
Message templates can be cleared and re-loaded at any time to allow different sets to be used for different test data, (see [Message Templates](#)).

**Note:** Don't forget you must select the **Ok** button to exit the Preferences dialog window for your changes to take affect.

## Logging, Auditing & Debug

[Previous](#) [Top](#) [Next](#)

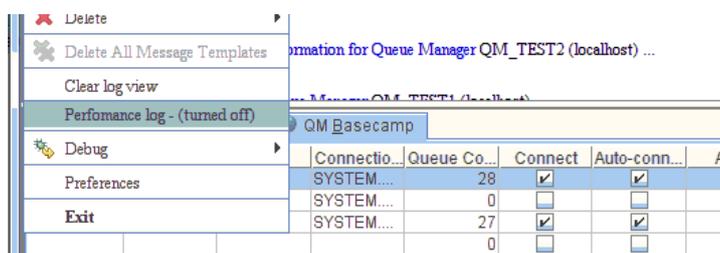
QueueZee v1.x only provides auditing at a [Queue Manager](#) level and logging is limited to what's provided in the log pane at the top of the application:



If you wish to save any of this information you should cut and paste it somewhere safe before you quit QueueZee! The logging is not circular so it will continue to grow until you either quit QueueZee or clear the log view, (an option available from the main **File** menu).

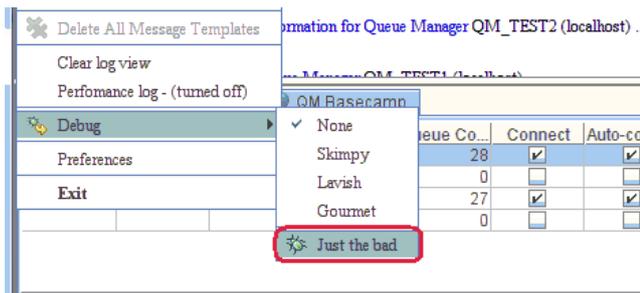
**NOTE:** QueueZee v2.x will provide much more comprehensive logging facilities.

Most significant actions produce a log message and of these a lot also produce two log messages so you can see from the timestamps how long an operation took. For actions that don't produce a log message or only produce one message you can turn on the performance log from the main **File** menu:

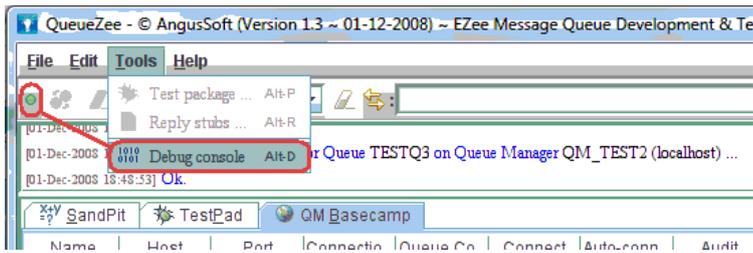


This will produce an entry and exit timestamp for every action you do that's shown in the log view.

You can also turn on various levels of debug from the main **File** menu but we recommend you leave this off unless you are instructed to turn it on by [AngusSoft](#) or you wish to track normal MQ exceptions like connection failures to dead Queue Managers etc. For this you should enable the *Just the bad* debug option as shown:



This will show the Java stack trace for every exception QueueZee encounters in the *Debug console*. You can view the debug console at any time from the **Tools** menu as shown:



When *Just the bad* debug is turned on QueueZee will indicate the debug console status from an icon on the far left of the main toolbar (shown above). This icon will remain green until an exception is encountered at which point it will turn into a yellow warning triangle. Once the debug console has been viewed the status indicator will return to a green light.

**Note: QueueZee handles all the exceptions it receives so virtually all the exceptions you see (if any) as a result of turning on debug won't mean anything is wrong in QueueZee itself!**

#### Command Line Options

[Previous](#) [Top](#) [Next](#)

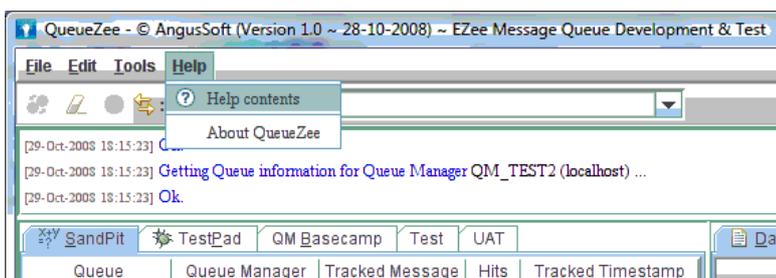
QueueZee v1.x only supports one command line option which is the `-x` parameter. If this is specified it tells QueueZee to ignore any auto-connects set for all Queue Managers. This is probably only useful if you have one or more Queue Managers that are causing the client connection request not to complete, (ie. QueueZee never starts up!).

QueueZee v2.x will support many more command line options allowing you to run it from scripts as part of a testing infrastructure. Please check our [website](#) regularly for development plans and beta-release details.

#### Help Menu

[Previous](#) [Top](#) [Next](#)

The help menu is selected from the main menu bar at the top of the GUI as shown:



Selecting it fires up this help module, so you have probably already done that if you're reading this!  
FAQ

[Previous](#) [Top](#) [Next](#)

#### Why are some port numbers wrong when I import Queue Managers?

[Previous](#) [Top](#) [Next](#)

QueueZee will always fill in any missing information it has available as defaults, (in this case the port is taken from the default defined in your [Preferences](#)). MQ Explorer Queue Manager definitions do not usually have listener port information saved for local Queue Managers as this generally isn't needed. QueueZee always connects to Queue Managers as a remote client so imported local definitions may need the port details edited and corrected.

#### Why are some features disabled in this version?

[Previous](#) [Top](#) [Next](#)

Some features not available in version 1.x will be available in version 2.x. Although they would do nothing if you could select them they have been disabled to prevent any confusion.

Please contact [AngusSoft](#) for version 2.x availability and release details.

Where does QueueZee store my configuration details?

[Previous](#) [Top](#) [Next](#)

QueueZee saves all configuration details to a file called **config.qz** in the QueueZee home directory every time it closes, (unless it has been asked to save after every change in the [Preferences](#)). This file is also automatically backed up to the same directory to minimise the risk of data loss. Despite this we also recommend you back this file up somewhere else periodically as general good practice. Also, if QueueZee is terminated forcibly (ie. killing it from the Windows task manager) it will not be able to update the configuration file.

Although it is definitely not recommended this file can be edited manually on occasions when needed or instructed to do so by [AngusSoft](#).

I have lots of messages loaded and QueueZee is behaving strangely, what's going on?!

[Previous](#) [Top](#) [Next](#)

---

**QueueZee** works very hard to provide uninterrupted operation for all its available features. If, however, things take a turn for the worse and available memory gets low or worse still, non-existent, certain operations may not work as expected!

**QueueZee** stays alive to enable you to save any unsaved configuration changes but you should close **QueueZee** or, at the very least, turn on debug to see what's happening when this occurs.

QueueZee won't connect to a Queue Manager and says it can't get the Queue list, what does this mean?

[Previous](#) [Top](#) [Next](#)

---

The first thing QueueZee does after establishing a remote connection to a Queue Manager is request the complete Queue list. Sometimes a Queue Manager may be alive and allowing client connections but refusing to allow PCF commands to complete. If this is the case QueueZee will consider this a failed connection as it can't operate without knowing about the Queues that reside on the Queue Manager it's trying to connect to.

From QueueZee v1.2 onwards a connection will be established even if individual Queue inquiries fail.

I made a change to my QueueZee configuration (adding a Queue folder perhaps) but this wasn't remembered when I next started QueueZee, [Previous](#) [Top](#) [Next](#) why is this?

---

QueueZee can only remember changes that get saved to the configuration file (**config.qz**). If QueueZee terminates unexpectedly or forcibly the configuration file does not get updated and hence any new changes will be lost.

To force QueueZee to save every configuration change after you make it you can configure it to do so from the [Preferences](#) dialog.

QueueZee says the message I select is Text but when I look at it in the Data pane and select it all it says it's XML, why is this?

[Previous](#) [Top](#) [Next](#)

---

You almost certainly have QueueZee configured to show XML nicely formatted in the Data pane, (check the *Message* tab in the [Preferences](#) dialog) and the unformatted message data contains some new-line characters in places that upset standard XML parsers!

As QueueZee uses its own parser to format XML it will quite happily format invalid XML which makes it useful for spotting missing angled brackets etc. However, in the case described above it will mask the rogue new-line character(s). To see the message data as it is in the actual message turn off the XML formatting in the *Message* tab in the [Preferences](#) dialog.

I've updated a message on the Queue but the old one is still there and/or a different one has been deleted, what's going on?

[Previous](#) [Top](#)

---

When QueueZee updates a message on the Queue it actually creates a new message and then deletes the old one, (as there is not really a concept of *updating* a message on a Queue!). This can, understandably, produce unpredictable results if there are messages with duplicate message identifiers on the Queue.

Therefore, it is not recommended you update messages on the Queue when you have many messages with duplicate message identifiers! (even though you are unlikely to lose any messages as QueueZee creates the new one before deleting the old one).