

What's New for MQ

John Tarby

October 14, 2002



(c) Copyright IBM Corporation 2002

Any references to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.



TPF WebSphere MQ

- Previously known as TPF MQSeries... same product, just a new name.
- This is purely a product name change from MQSeries to WebSphere MQ.

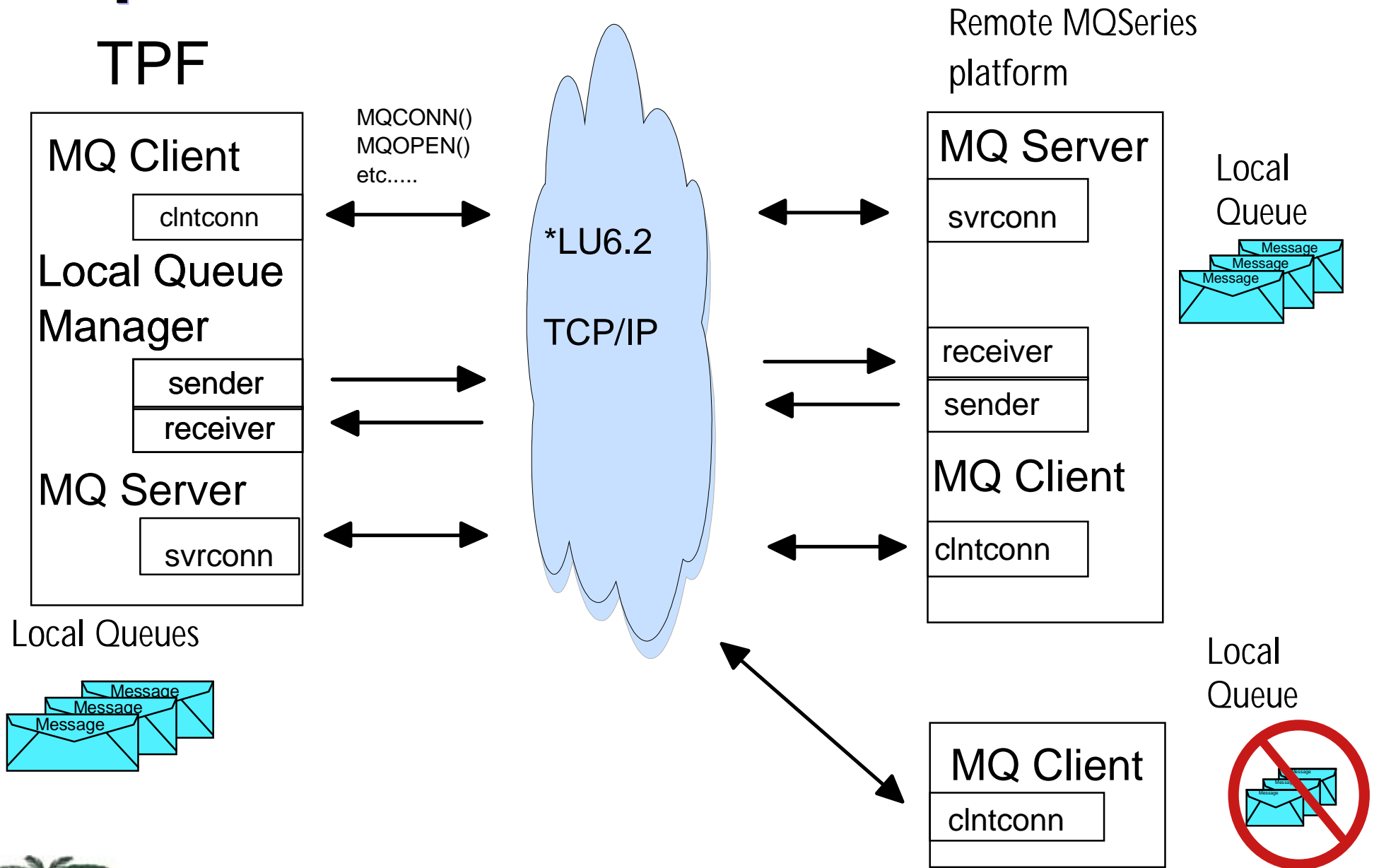


On to more important things..

- TPF now has all the fundamental elements in place
 - ▶ Client
 - ▶ Server
 - ▶ Local Queue manager
- Focus now will be adding functionality, both from the API level and through administrative functions.



A picture is worth a thousand words



* LU6.2 is for TPF MQ Client support only



Newly released changes and updates

- New administrative functions:
 - ▶ PJ28758
 - Channel Retry for sender channels
 - Improvements to the sweeper
 - Utility to capture MQ definitions.
 - ▶ PJ28759
 - Restore of MQ definitions.
- No changes to the API this time.



Channel Retry

- Sender channels only
- Attempts to restart the channel without operator intervention
- Retries the send operation in situations where the error may be transient and a retry is reasonable
- Parallels the implementation on other platforms.



Channel Retry (cont)

- Set and adjusted using ZMQSC DEFine and ZMQSC ALTer
- SHORTRTY SHORTTMR for short retry count and timer
 - ▶ TPF will attempt to restart the channel every 'short timer' seconds for 'shortrty' attempts.
- LONGRTY and LONGTMR for long retry count and timer.
 - ▶ When the short retries are exhausted, TPF will begin the long retry attempts.
 - ▶ When long retries are exhausted, the channel will be stopped.



Channel Retry defaults

- Default values for all channels, both newly defined and existing:
 - ▶ SHORTRTY-60
 - ▶ SHORTTMR-10
 - ▶ LONGRTY-999999999
 - ▶ LONGTMR-1200.
- Retries will be attempted every 10 seconds for 10 minutes.
- Retries will be attempted every 20 minutes for 38,000 years.
 - ▶ Unless Queue Manager is stopped or channel is manually stopped.



What is the sweeper?

- Messages in processor unique queues are contained in SWBs.
- If the queue depth is increasing, the sweeper will move some of those messages from memory into a TPFCS BLOB to free up SWBs.
- As more messages are added, additional BLOBs are created as needed.
- Messages are moved back into SWBs when required (aka unsweep).



A smarter sweeper

- It's not that the old sweeper wasn't smart, it was a bit overanxious to do work.
- Sweeper will only be activated when a user-specified percentage of SWBs are in use.
 - ▶ ZMQSC DEF/ALT QMGR SWEEPSWB-*nn*
 - ▶ Change takes effect immediately on all processors.
 - ▶ Previously, the sweeper would always be active regardless of the number of available SWBs in the system.



A smarter sweeper (*continued*)

- Previously the entire BLOB was unswept, potentially making the queue a candidate for sweeping again, possibly causing a thrashing condition.
- Now, when a queue is unswept, only enough messages are brought into memory to satisfy the application for the short term.
- Avoids re-sweeping and potential CTL-C conditions.



A smarter sweeper (*continued*)

- Queues with sporadic message arrival rates make it difficult for TPF to predict how an application will be able to service this queue, causing too few messages to be left in memory.
- A user-specified number of messages can be left on the queue before sweeping is started, regardless of the queue activity.
 - ▶ ZMQSC DEF/ALT QL-nnnnnn
SWEEPDEPTH-xxxxx



A faster sweeper

- Increased buffer sizes for sweeper to maximize TPFCS BLOB write rate
 - ▶ Block size was previously 32 K
 - ▶ Increased to 1MB.
- Allows sweeper to handle queues growing at faster rates or queues with large message sizes.



Saving MQ definitions

- ZMQSC CAPTURE FILE-nnnn
- Generates a file of all the ZMQSC messages that would be needed to create the MQ definitions in their current state.
- File resides in TPF File System.
- Can be FTP'd to other systems to build other MQ images, or used to restore the existing system in case of system failure.
- Does *not* capture message data, only MQ definitions.



Restoring MQ definitions

- ZEXEC FILE-nnnnnnn
 - ▶ Will read a TPF File System file containing commands and serially execute them
 - ▶ Created as an MQ restore utility, but will work for any TPF command, not just ZMQSC
 - ▶ Separate ECB is created for each line
 - ▶ Subsequent lines are not executed until the previous ECB has exited
 - ▶ Inherits the properties of the issuing terminal for security purposes
 - ▶ Responses are routed to the issuing terminal.




Extra stuff

- Support for MAXHANDLES
 - ▶ Prevents an errant application from issuing too many MQOPEN APIs
 - ▶ ZMQSC DEF/ALT QMGR MAXHANDLES-nnnn
- Consistent usage of QMGR keyword
 - ▶ ZMQSC DEFine/ALTer QMGR
 - ▶ Matches usage in ZMQSC DISPLAY.



Resources and contacts

- MQ Task Force
 - ▶ Discussion group on Web
 - Informally chat with other TPF MQSeries users.
- Sample drivers
 - ▶ www.ibm.com/tpf/download/mqdriver.htm
 - ▶ Both basic and advanced test cases.
- TPF Customer Service Web site
 - ▶ www.ibm.com/tpf/pages/support.htm
- TPF Information Center Web site
 - ▶ www.ibm.com/tpf/pubs/tpfpubs.htm



IBM, MQSeries and WebSphere are registered trademarks of International Business Machines Corporation in the United States, other countries, or both. Other company, product, and service names may be trademarks or service marks of others.