#SHAREorg

SHARE

What's New in WebSphere MQ
WMQ V7.1

Mark Taylor
IBM Hursley

March 2012
Session 10695

How WebSphere MQ meets your Connectivity needs

*Dynamic network that delivers the **data** you require from wherever it resides to wherever you want it in whatever way you want it at whatever time you want it*

Universal Messaging

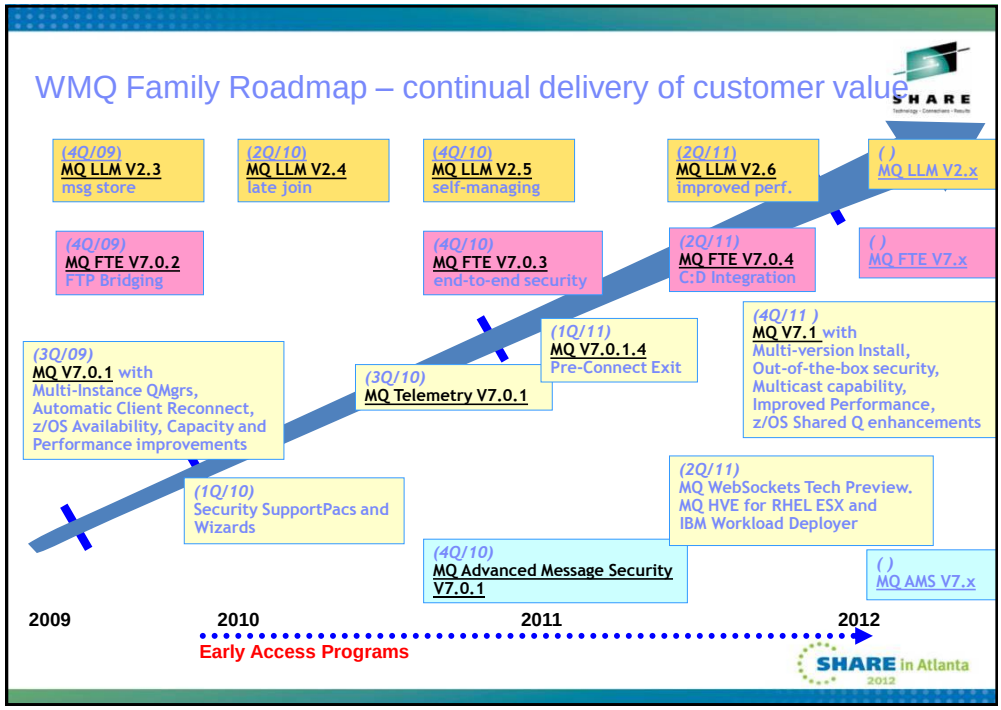
1. Anything Anywhere
- Any skills
- Any traffic
- Any language
- Any environment
- Any platform

2. Best Delivery
- Choice of service
- Resilience, Integrity, Security
- Throughput, Latency
- High availability

3. Scale Dynamically
- Start small
- Grow incrementally
- Stretch elastically
- Scale admin

CSS: F S

SHARE in Atlanta 2012



WebSphere MQ V7.1

WebSphere MQ V7.1: Feature Summary

WebSphere MQ V7.1
Announced: 4 October 2011
Availability: 11 November 2011

Simplification

New Feature	*Benefits*	
Multi-Version Install capability on Distributed platforms	Makes it easier to deploy and upgrade systems and stage version to version migration	Unix and Windows support for multiple versions of MQ V7.x (AND one copy of MQ V7.0.1) down to fixpack levels. Relocatable installation support. Applications can connect to any Qmgr
Enhanced Security	Simplified Configuration Enhanced Authorisation and Auditing	IP address Authorisation capability Additional crypto algorithms More granular authorisation for non-local queues Application Activity Reports
Cloud Support	Simplifies and support Cloud deployments	Additional HVE images
Enhanced Clustering	Improves ease-of-use	Authorisation on Cluster Q rather than XMIT Q on Dist. Platforms Bind-on-Group Support
Multicast capability	New messaging QoS provides low latency with high fan-out capability	MQ Pub/Sub Topic space can now map to multicast Group Addresses Provides direct interoperability with MQ LLM
Improved scalability and availability on z/OS	Further exploitation of z196 Customer control over CF storage use CF Connectivity Loss improvements	Code contention reduced to improve multiprocessor linear scaling Use of MQ Datasets rather than DB2 significantly improves "large" message capability Structure rebuild capability for CF Connectivity Loss scenarios
Improved Performance on Dist platforms	Improved multiprocessor exploitation	Various code improvements

SHARE in Atlanta 2012

Multi-Version Installation

- MQ on Unix and Windows can install multiple levels on a system
  - Relocatable to user-chosen directories
  - Can have multiple copies even at the same fixpack level
- Simplifies migration
  - Can move applications as needed, not all at once
  - No need for parallel hardware
- Easier for ISVs to imbed MQ in solutions
  - Can install in "private" locations without worrying about other copies
  - Reduces support concerns
- Permits a single copy of V7.0.1 to remain on system
  - So existing systems can be migrated
  - Must be 7.0.1.6 or later

SHARE in Atlanta 2012

Multi-Version Installation: Concepts

- Main concept is an installation
  - Refers to the directory containing the binaries from a particular version of MQ
  - Can have a descriptive name
- One installation can be designated as primary
  - Required on Windows where some OS-specific elements have to be registered
  - Optional on Unix, creates symlinks to commands and libraries in /usr
    - Not created by default so your PATH will not always find MQ commands
- Queue Managers are owned by a specific installation
  - Governs the level of function available when the queue manager is running
  - Ownership can be changed to a newer installation for migration

Multi-Version Installation: Application Impacts

- Existing applications “know” where the MQ libraries are
  - Embedded path or PATH/LIBPATH/LD_LIBRARY_PATH
  - Has always been a fixed location on Unix
- When MQ libraries move, apps will need to know where the new location is
  - /usr cannot be assumed
- New application libraries able to connect to any version of queue manager
  - Libraries such as libmqm, libmqic etc redesigned
  - Dynamically loading dependent libraries associated with the corresponding qmgr
  - If your app can find one V7.1 libmqm, it can connect to any qmgr, including future versions
- **MIGRATION NOTE:** Exits that invoke the MQI will need to be updated
  - Such as API Exits
  - Do not want exits to pull in different libraries than main application
  - Extended interface provides pointers instead for invoking MQI

Administration Examples

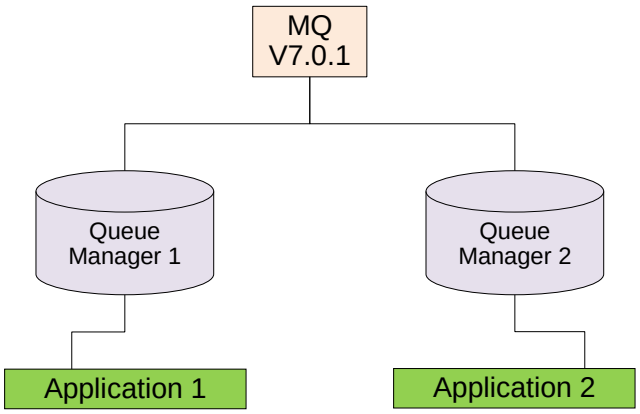
```
$ /usr/mqm/bin/dspmqver -i
Name:        WebSphere MQ
Version:     7.1.0.0
Level:       p000-L110915
BuildType:   IKAP - (Production)
Platform:    WebSphere MQ for AIX
Mode:        64-bit
O/S:         AIX 6.1
InstName:    Installation1
InstPath:    /usr/mqm
InstDesc: My default installation
DataPath:    /var/mqm
Primary:     Yes
MaxCmdLevel: 710

Name:        WebSphere MQ
Version:     7.1.0.0
InstName:    Installation2
InstPath:    /usr/mqm2/usr/mqm
InstDesc:    A second installation
Primary:     No
```

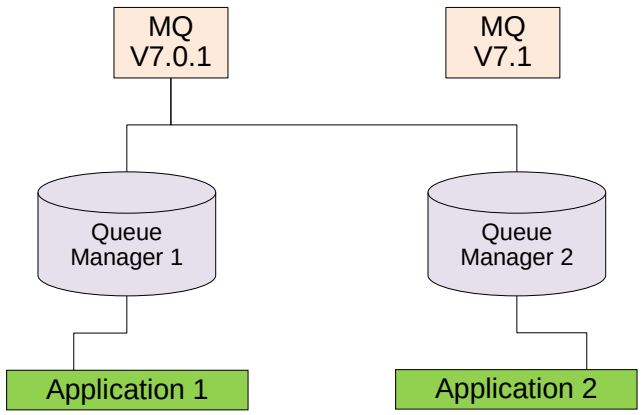
```
$ dspmq -o installation
QMNAME(V71A)
        INSTNAME(Installation1)
        INSTPATH(/usr/mqm)
        INSTVER(7.1.0.0)
QMNAME(V71B)
        INSTNAME(Installation1)
        INSTPATH(/usr/mqm)
        INSTVER(7.1.0.0)
QMNAME(INST2QM)
        INSTNAME(Installation2)
        INSTPATH(/usr/mqm2/usr/mqm)
        INSTVER(7.1.0.0)
```

```
$ /usr/mqm/bin/endmqm INST2QM
AMQ5691: Queue manager 'INST2QM' is
associated with a different installation.
```

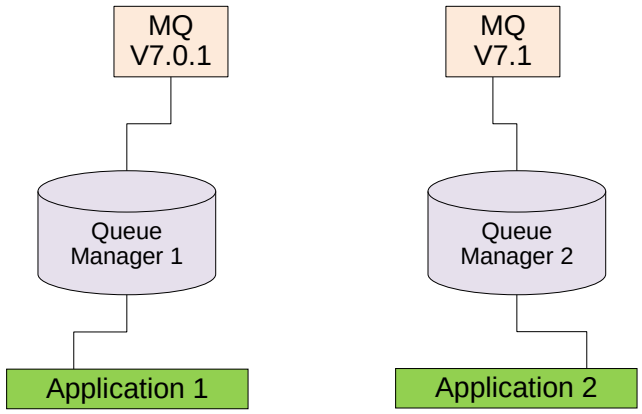
Application Migration: (1) Identify Applications to Migrate



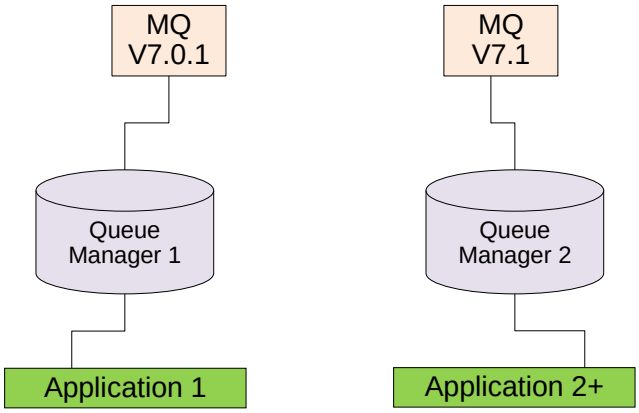
Application Migration: (2) Install V7.1 alongside V7.0.1



Application Migration: (3) Associate V7.1 code with a QMgr



Application Migration: (4) Modify App to Exploit New Features



Security: Channel Access Control

- Simplifying configuration for channel access
  - Clients and queue managers
- Rules are based on
  - Partner IP address
  - Partner Queue Manager name
  - SSL Distinguished Name mapping
  - Asserted identity (including *MQADMIN option)
- Easy to test rules that you define
  - DISPLAY CHLAUTH can “execute” rules
- Rules can be applied in WARNING mode
  - Not actually blocked, but errors generated

- **MIGRATION NOTE**: Standard rules block clients on new queue managers
  - “Secure by default”
  - Migrated queue managers behave as before until you enable the rules
  - Queue manager attribute CHLAUTH(ENABLED|DISABLED) provides overall control

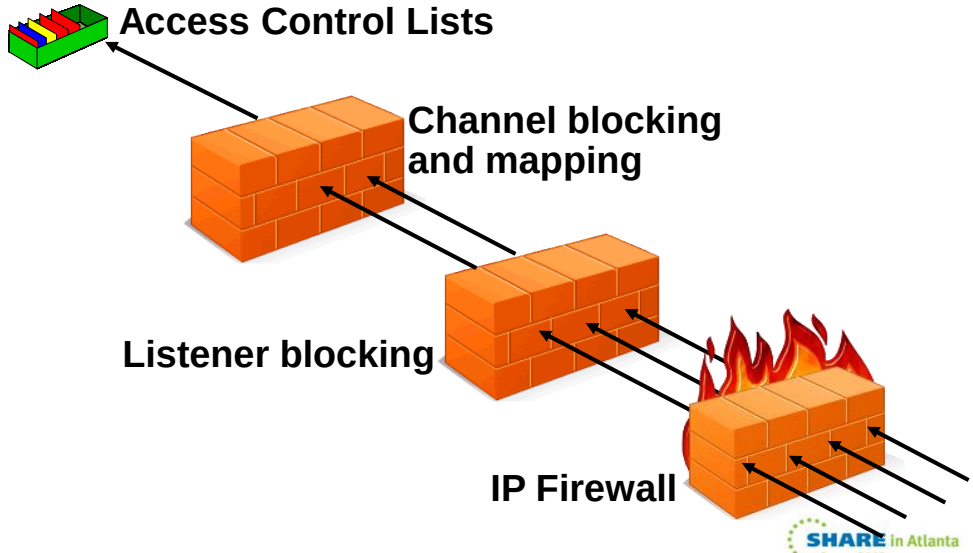
Security: Channel Access Control – example uses

- Block connections from specific IP addresses
- Block connections from specific Userids

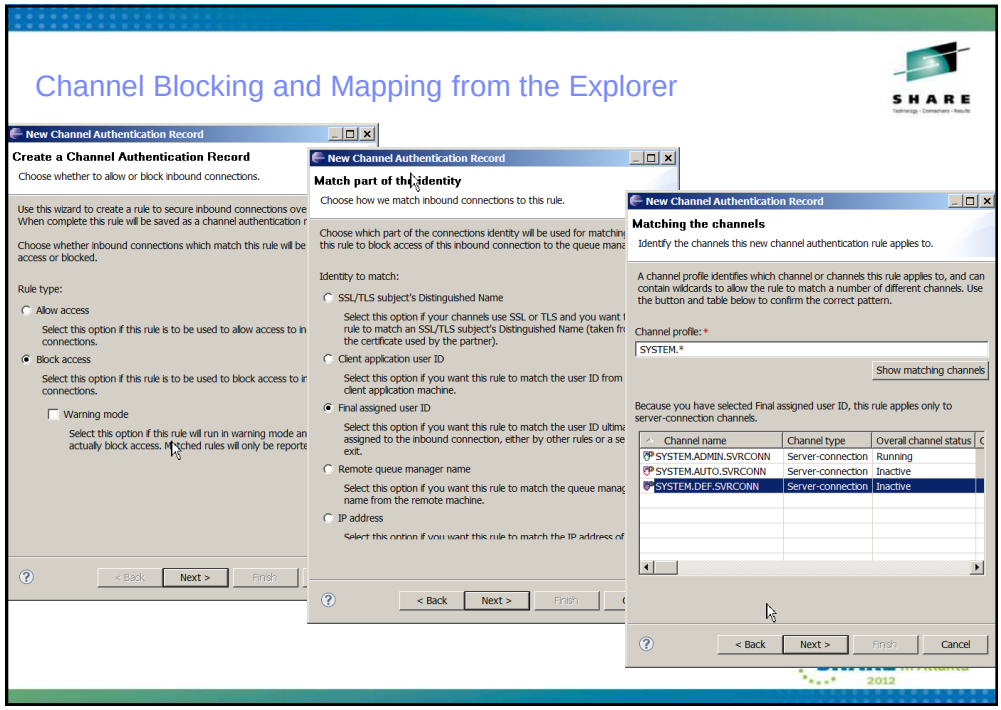
- Set MCAUSER value used for any channel coming from a specific IP address
- Set MCAUSER value used for any channel having a specific SSL or TLS DN
- Set MCAUSER value used for any channel connecting from a specific Qmgr

- Block connections claiming to be from a particular Qmgr unless the connection is from a specific IP address
- Block connections claiming to be from a particular Client Userid unless the connection is from a specific IP address
- Block connections presenting a particular SSL or TLS certificate unless the connection is from a specific IP address

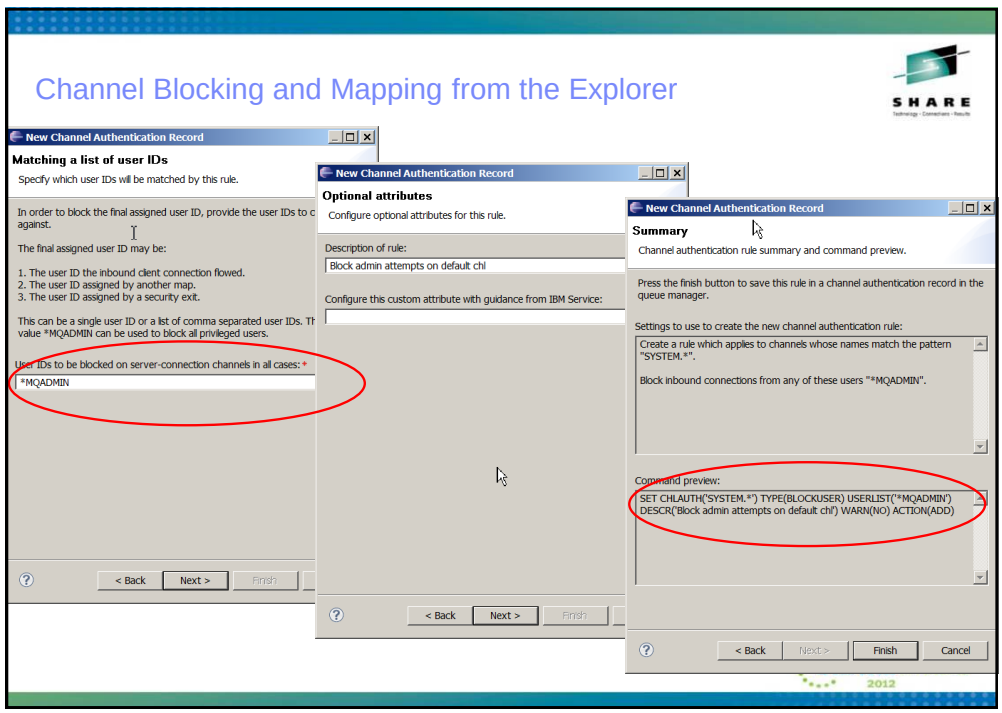
Channel Access Blocking Points



Channel Blocking and Mapping from the Explorer



Channel Blocking and Mapping from the Explorer



Channel Access Scenario (1)

SET CHLAUTH(*) TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)

"We must make sure our system is completely locked down"

Channel Access Scenario (2)

SET CHLAUTH(*) TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)

SET CHLAUTH(BPCHL.*) TYPE(SSLPEERMAP) SSLPEER('O=Bank of Shetland') MCAUSER(BANK123)

SET CHLAUTH(BPCHL.*) TYPE(SSLPEERMAP) SSLPEER('O=Bank of Orkney') MCAUSER(BANK456)

"Our Business Partners must all connect using SSL, so we will map their access from the certificate DNs"

Channel Access Scenario (3)

SET CHLAUTH(*) TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)

SET CHLAUTH(BPCHL.*) TYPE(SSLPEERMAP) SSLPEER('O=Bank of Shetland') MCAUSER(BANK123)

SET CHLAUTH(BPCHL.*) TYPE(SSLPEERMAP) SSLPEER('O=Bank of Orkney') MCAUSER(BANK456)

SET CHLAUTH(SYSTEM.ADMIN.SVRCONN) TYPE(ADDRESSMAP) ADDRESS('9.20.1-30.*') MCAUSER(ADMUSER)

"Our Administrators connect in using MQ Explorer, but don't use SSL. We will map their access by IP Address"

Channel Access Scenario (4)

SET CHLAUTH(*) TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)

SET CHLAUTH(BPCHL.*) TYPE(SSLPEERMAP) SSLPEER('O=Bank of Shetland') MCAUSER(BANK123)

SET CHLAUTH(BPCHL.*) TYPE(SSLPEERMAP) SSLPEER('O=Bank of Orkney') MCAUSER(BANK456)

SET CHLAUTH(SYSTEM.ADMIN.SVRCONN) TYPE(ADDRESSMAP) ADDRESS('9.20.1-30.*') MCAUSER(ADMUSER)

SET CHLAUTH(TO.CLUS.*) TYPE(QMGRMAP) QMNAME(CLUSQM*) MCAUSER(CLUSUSR) ADDRESS('9.30.*')

"Our internal cluster doesn't use SSL, but we must ensure only the correct queue managers can connect into the cluster"

Security: SSL

- More crypto algorithms supported for SSL
  - Stronger algorithms are now available and recommended
  - MQ V7.0.1 added some SHA-2
  - MQ V7.1 adds more, with support for the NSA “Suite B” standard which includes Elliptic Curve cryptography
- Some older algorithms (eg SHA-1) should be considered deprecated
  - No plans to withdraw older algorithms immediately
  - But expect them to be removed in a future version of MQ
- Newer algorithms supported by gskit8 on Distributed platforms
  - Waiting for z/OS and iSeries SSL implementations before MQ can support them there
- The gskit toolkit is now provided inside the MQ installation
  - Will not clash with alternative levels from other MQ installations or other products

Security: Authorisations for Non-Local (Clustered) Queues

- Distributed platforms now have authorisations for non-local queues
  - Including clustered queues
  - Making it consistent with z/OS
  - Also consistent with Topic authorisations
- So there is no longer a need to authorise access to the cluster transmit queue
- Grant authorisation to the remote queue manager instead
  - A new pseudo-object known to the OAM

```
setmqaut -m QM1 -t queue -n SYSTEM.CLUSTER.TRANSMIT.QUEUE -p mquser +put
                                BECOMES
setmqaut -m QM1 -t rqmname -n QM2 -p mquser +put
```

Application Activity Reports

- New set of events to report on MQI operations by applications
  - One PCF event may contain multiple MQI operations
- Configurable in granularity
  - Amount of data
  - Which applications
- Enables scenarios such as
  - Application audit trail
  - Message duplication
  - Resource usage: which queues or topics are actually being used
  - Problem Determination: most recent MQI calls by applications
  - Application Coding Standards: does everyone use the MQI in the recommended way
  - And more …
- On all Distributed platforms

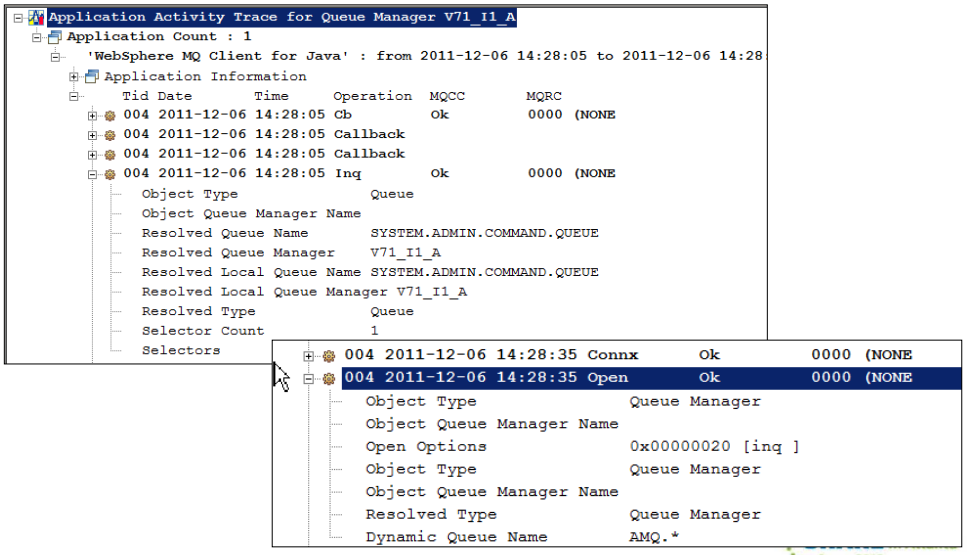
Extract from Report

```
MonitoringType: MQI Activity Trace
QueueManager: 'V71'
Host Name: 'rockall.hursley.ibm.com'
CommandLevel: 710
ApplicationName: 'WebSphere MQ Client for Java'
ApplicationPid: 18612354
UserId: 'mquser'
ConnName: '9.20.95.106'
Channel Type: MQCHT_SVRCONN
Platform: MQPL_UNIX
==========================================================================
Time     Operation  CompCode MQRC  HObj  (ObjName)

10:04:09 MQXF_INQ   MQCC_OK  0000  2
10:04:09 MQXF_CLOSE MQCC_OK  0000  2
10:04:09 MQXF_OPEN  MQCC_OK  0000  4  ()
10:04:09 MQXF_INQ   MQCC_OK  0000  4
10:04:09 MQXF_CLOSE MQCC_OK  0000  4
10:04:09 MQXF_OPEN  MQCC_OK  0000  4  (SYSTEM.DEFAULT.LOCAL.QUEUE)
10:04:09 MQXF_INQ   MQCC_OK  0000  4
```

SHARE in Atlanta 2012

Forthcoming release of MS0P



Clustering

- "Bind on group"
  - All messages within a logical group are routed to the same queue manager
  - Workload balancing is done for each group
  - Simpler for applications that use message groups
  - Previously would have had to close and reopen the queue
- New option in the MQI and DEFBIND attribute for queues
- Once a group has started its path to a selected queue manager, messages in that group will not be reallocated in the event of a failure
- New sample **amqsclm** to monitor queues and redistribute delivered messages
  - If a queue has no getters, block further deliveries and redistribute existing messages
  - Includes source code, so easy to modify

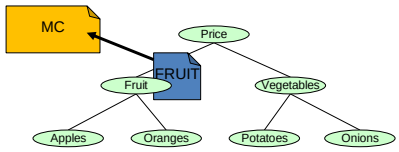
MQ Clients

- A client is now available on System i enabling connectivity from C and RPG programs without needing a local queue manager
  - Platform already had a Java client
- MQI libraries like libmqm connect to local and remote queue managers
  - Smart switching for clients, as well as handling multi-version systems
- API Exits available in C clients
  - Same interface as available for local binding applications

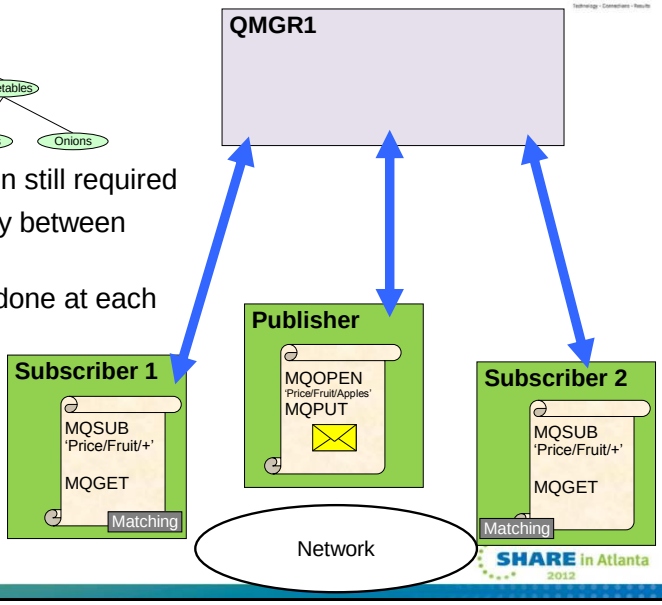
MQ Clients – Multicast

- Publish/Subscribe is enhanced to support multicast communication
  - Uses technology from the MQ Low Latency Messaging product
  - So it is interoperable with LLM
- Provides new Quality of Service
  - Low latency with high fan-out
  - Provides higher speeds for non-persistent messages
  - Provides higher availability as queue manager can be removed without affecting flow
  - Provides “fairness” as all recipients of a message get it at the same time
  - Higher scalability as additional subscribers cause no additional traffic
- Mapping MQ topic space to multicast group addresses
  - Can have mix of multicast and queue-based subscribers
  - Topic objects have associated COMMINFO objects to define addresses and other attributes
- Supports direct communication from publisher to subscriber, bypassing qmgr
- Queue manager maintains status and statistics for monitoring

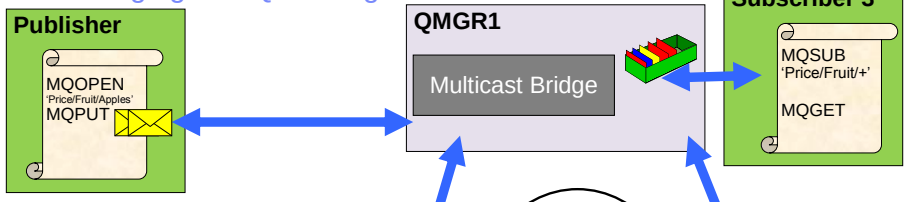
Multicast Publish/Subscribe



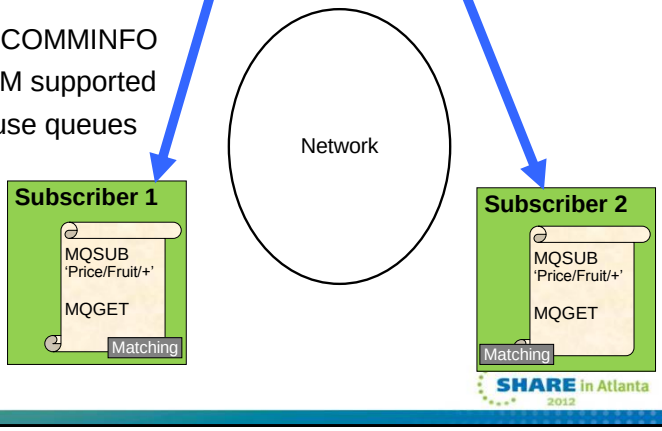
- Normal MQ connection still required
- Messages flow directly between clients
- Subscriber matching done at each client



Multicast Bridging and Queueing



- Bridging controlled by COMMINFO
- Only outbound from QM supported
- Subscribers can also use queues



Channels

- See the MQ version of connecting partner
  - Level of clients and queue managers available in channel status
  - For example a V7.0.0.1 client shows as RVERSION(07000001)
  - Can distinguish Java, C, .Net client programs
  - Helps administrator determine whether partner needs upgrading
- Distributed platforms now use DISCINT to disconnect idle clients
  - ClientIdle qm.ini parameter ignored
  - Consistent with z/OS
- Alternative channel batch control based on byte counts
  - BATCHLIM attribute
  - Useful when a transmission queue holds mix of large and small messages
  - Can make batch time (latency) more consistent
  - Batch is ended when first of either bytes or messages transferred reach configured limit
- Per-channel control of Dead Letter Queue
  - New channel attribute USEDLQ(YES|NO)
- DEFRECON added to client channels
- Any pending sequence number reset is shown on DIS CHL

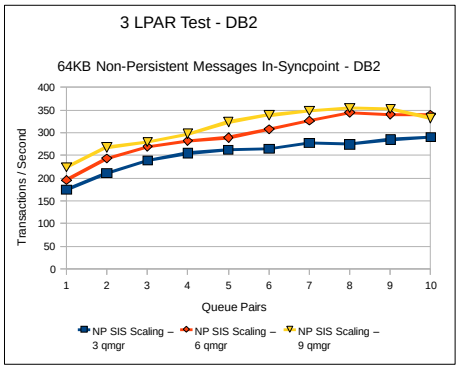
z/OS Performance and Availability

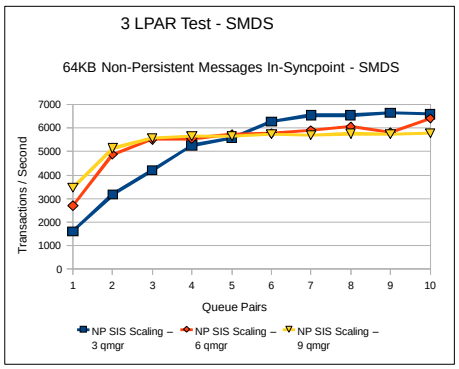
- Performance
  - z196 Scaling improvements for both non-shared and shared queues
    - Have successfully processed more than ONE MILLION non-shared messages/sec through a single queue manager
    - Have also successfully processed 150K shared msgs/sec with 3 queue managers
  - Improved performance by using SMDS for large messages on shared queues
- Availability
  - Structure rebuild when connectivity to CF is lost improves availability of Shared Queues
  - GroupUR function from MQ V7.0.1 for Distributed QSG connections available for CICS usage
    - CICS 4.2 can use this to enhance the MQ Group Attach originally provided in CICS 4.1

Large Shared Queue Messages: SMDS

- Using DB2 BLOBs to store large (>63KB) messages is expensive
  - Both CPU and pathlength
- Shared Message DataSets (SMDS) removes DB2 for large message storage
  - DB2 still needed for storing shared definitions
  - CF still holds small messages and pointers for offloaded messages
- Shared VSAM datasets increase shared queues capacity and performance
  - All queue managers in the QSG can access the datasets
- CF Structure message reference still controls locking, ordering, deletion etc.
  - So every message still has a “pointer” in the CF
- Rules control offload message size and % Structure-full offload trigger
  - Set per CF structure
  - Offloading messages at 63K gives 1.25M messages on a 100GB structure
  - Offloading all messages at 1KB gives 45M messages on same structure
- All QSG members must be at new level to access queues with this capability

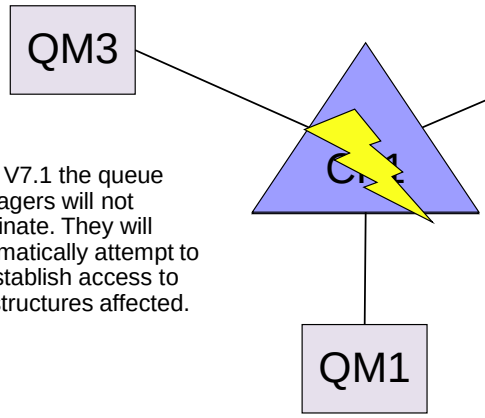
SMDS Performance Improvement





- Early Test Results on z196
- Tests show comparable CPU savings making SMDS a more usable feature for managing your CF storage
- SMDS per CF structure provides better scaling than DB2 BLOB storage

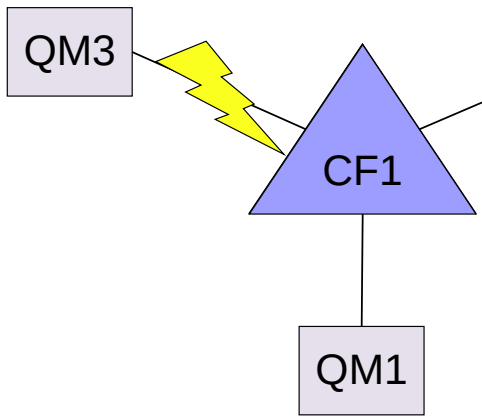
Total Loss of CF Connectivity: V7.1



With V7.1 the queue managers will not terminate. They will automatically attempt to re-establish access to the structures affected.

In the case of a total loss of connectivity the queue managers can automatically recover (RECOVER CFSTRUCT) the structures that were on the failed CF into an alternative CF (if available)

Partial Loss of CF Connectivity: V7.1



In the case of a partial loss of connectivity, a System Managed Rebuild will be automatically initiated by the queue managers to rebuild the structures into a more available CF. This will mean that both persistent and non-persistent messages will be retained.

Scalability & Performance – Distributed platforms

- Performance measured and improved for a range of scenarios
  - Hardware capabilities have evolved over years to have more CPUs, more memory etc
  - MQ topologies have evolved to have more clients and larger/fewer queue managers
- "Fastest MQ ever": better performance than V6 and V7
  - Improved scaling in multi-core environments
  - Improved throughput for typical message sizes (25% for 2KB messages)
  - Improved logging rates for typical message sizes (50% for 2KB messages)
- Multicast faster than traditional non-persistent
  - Over 5x for one-many publications
- Performance reports to be released on availability

Scalability & Performance – MQ Explorer

- Design changes to MQ Explorer reduce its footprint and improve performance
- Now does not include full Eclipse development workbench
  - But Explorer can be easily added to other Eclipse installations and products
- Many Explorer installs are supported within the overall multi-version support
  - But each Explorer only fully manages queue managers associated with its own installation
  - Use client connections for other installation queue managers on same machine

	V7.0.1	**V7.1**
Time to install MS0T	203 seconds	92 seconds
Startup Time	6 seconds	4 seconds
Connect to 100 queue managers	At least 53 seconds	7 seconds
Enable and disable Sets for 100 queue managers	35 seconds	1 second

Management of Distributed platforms

- New integrated command (dmpmqcfg) to extract configuration
  - Fulfills the role that MS03 (saveqmgr) has done over many years
    - Backup your configuration, change control, rebuild systems etc
  - MAKEDEF already available on z/OS
  - Different syntax than MS03, but similar function
- MQSC commands equivalent to setmqaut/dspmqaut
  - So you don't need to drop out of the command interface to modify security
  - Can simplify scripting of configuration changes
  - No current plans to remove *mqaut commands
- Multi-instance Queue Managers on Windows
  - The need for domain controllers ("domainlets") has been removed
  - New option when creating queue managers to define ownership

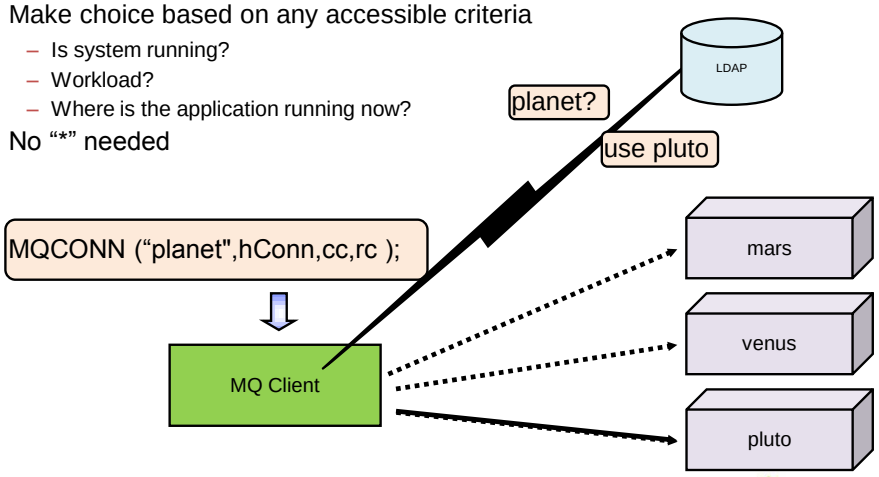
CSS: F_S

Extending the reach of MQ – MQ Telemetry Transport (MQTT)

- IBM developed a protocol for constrained systems like industrial controllers
  - Later renamed MQ Telemetry Transport (MQTT) due to broader telemetry adoption
  - Built for low bandwidth, high latency, unreliable, high cost networks
  - Tailored for resource-constrained client application environments
- Traditional messaging qualities of service provided where environment allows
- Feature available from MQ 7.0.1.4; server platform coverage extended in V7.1
  - Highly scalable
    - A single queue manager can handle up to 100K concurrently connected devices
  - Fully integrated / interoperable with WMQ
    - Publishers and subscribers can exchange messages with MQI and JMS applications
- In addition any 3rd party, open source or roll your own MQTT client can be used

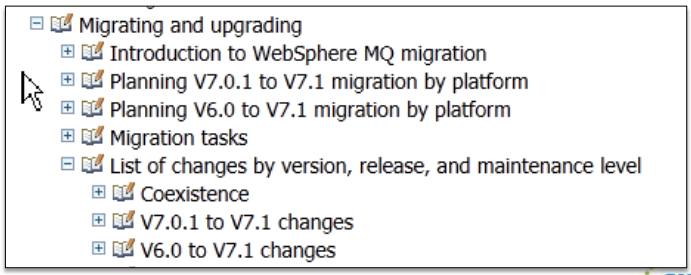
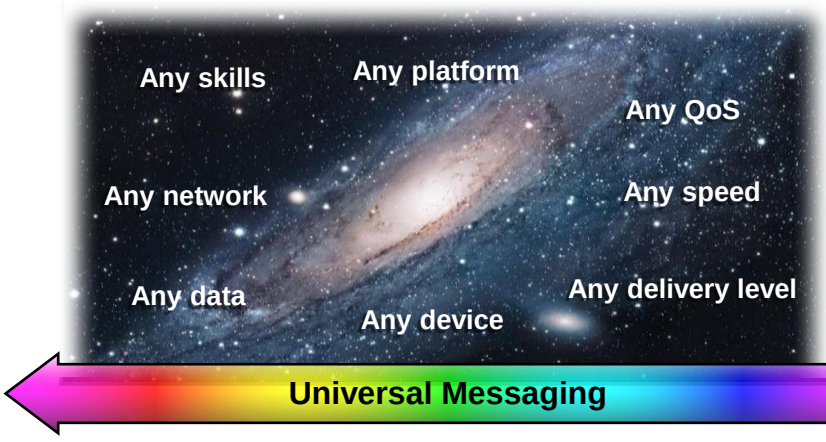
Dynamic Client Connection Using Pre-Connect Exit

- Look up in a directory such as LDAP
- Make choice based on any accessible criteria
  - Is system running?
  - Workload?
  - Where is the application running now?
- No "*" needed



No Longer Supported

- V7.1 removes a few older features including
  - Support for HP-UX on PA-RISC hardware
  - Windows Performance Monitor
  - Windows Active Directory Service Interface
- See Migration Guide in InfoCenter



Universal Messaging with WebSphere MQ