WebSphere MQ for z/OS v7+ Proof-Of-Technology (POT) workshop

Welcome

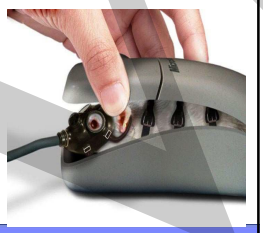
IBM EMEA panIMT Team zWebSphere team
(with thanks to IBM US ATS team for the original “Wildfire” material)

Your instructors today…

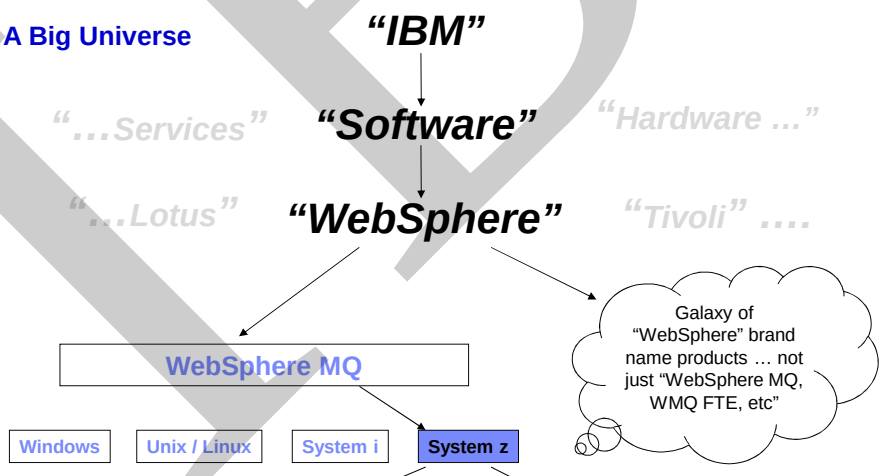
- Agnes ten Brink
  agnes.ten.brink@nl.ibm.com
- Carl Farkas
  farkas@fr.ibm.com
- Alberto González Dueñas
  alberto-gonzalez@es.ibm.com

MQ v7+ Proof-Of-Technology (POT) Workshop objectives

- Introduction to the new features of WebSphere MQ v7 and v7.1 (& v7.5)
  - Familiarise you with the new functions introduced in these (relatively) recent versions of MQ, especially those features that are meaningful on z/OS
    - Note: Most new MQ "Distributed" features will also be covered, but the focus is MQ on z/OS
  - Give you a chance to actually touch these features!
- Warning: *a POT does not replace MQ education*. It is *not* expected that you'll have enough knowledge at the end of this POT, for example, to master these new features. IBM and partners teach very good MQ classes – you're strongly encouraged to sign up for that education.
- Prerequisites
  - Be able to read some English
  - Some experience manipulating a Windows platform
  - Some experience in the z/OS environment
  - These exercises do assume that you are familiar with MQ on z/OS, its installation, configuration and administration



A Big Universe



Agenda – day 1

Start	Title
09:30	Lecture: Welcome
10:15	Lecture: What's new MQ v7.0
11:30	Lab: Connectivity and environment Lab: MQ Explorer Enhancements
12:15	Lecture: Pub/Sub
13:00	Lunch: Buen apetito
14:15	Lab: Pub/Sub
15:45	Lecture: What's new in WMQ V7.1
16:30	Lecture: Channels and Security
17:15	Done

Agenda – day 2

Start	Title
09:00	Lecture: Beyond MQ: WebSphere Message Broker
10:00	Lab: Channel authentication
11:30	Lecture: Introduction WMQ Advanced Message Security (AMS)
12:15	Lecture: Shared Queue Changes in WMQ 7.1
13:00	Lunch: Buen apetito
14:00	Lab: IP13 Performance test tool Lab: Shared Queues in v7.1 and SMDS
15:15	Lecture: Introduction WMQ File Transfer Edition (FTE)
16:00	Lab: Configuration FTE
17:15	Done

Introductions

Name ?

Company ?

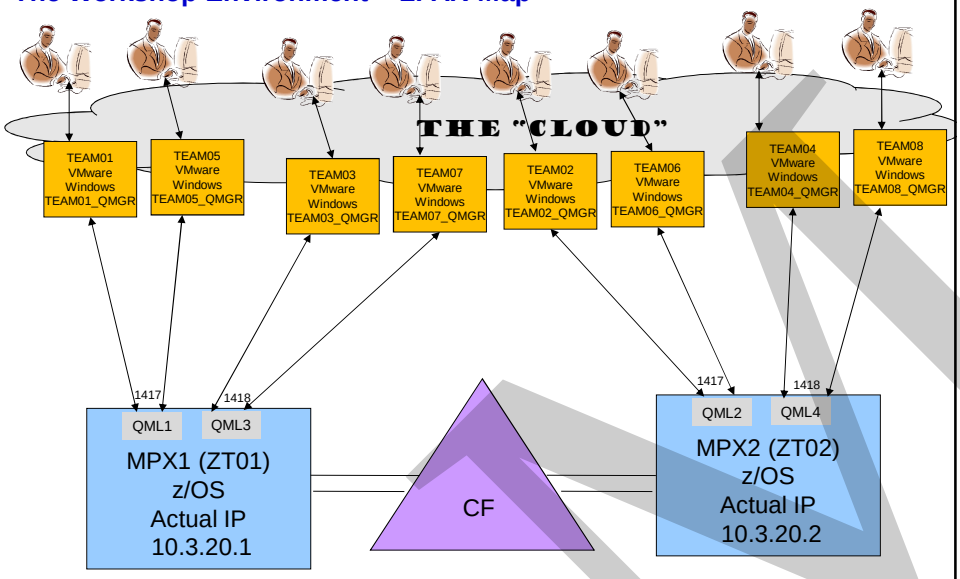
Background ?

Plans for WMQ V7 or any WMQ "stack" products ?

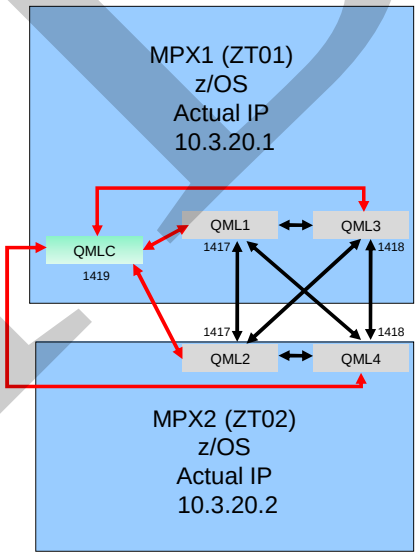
Administrivia

- Start & Stop times
- Lunch & Breaks
- Restrooms
- Cell Phones & Pagers
- In Case of Emergency

The Workshop Environment - LPAR Map



Zoom on MQ

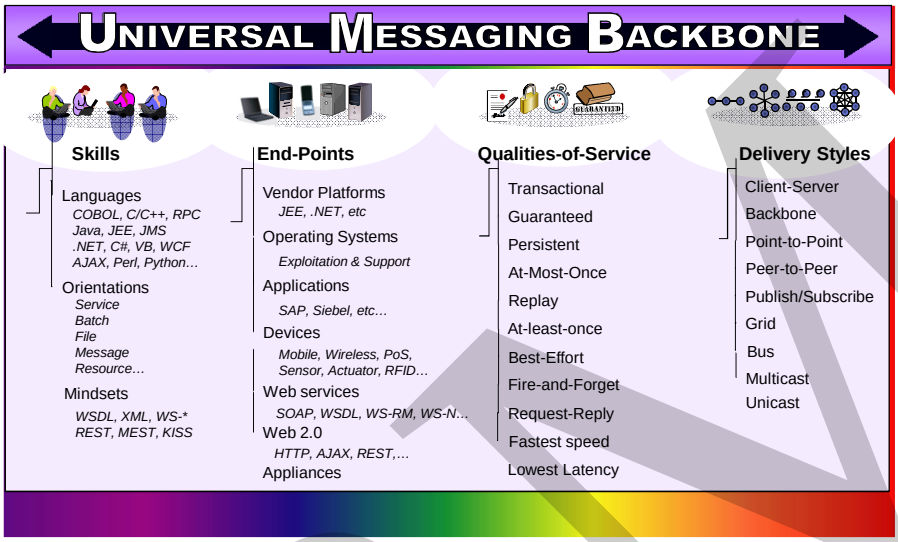


Rules of engagement

- Each lab includes detailed instructions.
- You're strongly advised to follow the instructions exactly.
- There are 8 "teams" (TSO userids): TEAM01, TEAM02….
  - –Your instructor will assign your team, and provide the password
- Given the nature each exercise, it's probably not a go beginning.
- If you're not sure structor. Better to ask too much, tha
- These LPARs are
  - –You are given enou exercises, see results, etc.
  - –Please try to resist ettings, change the z/OS userid "settings", eg classes/students use these ids!

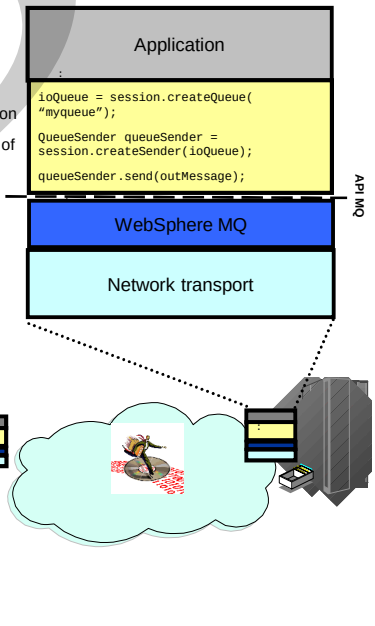
Let's talk about messaging….

IBM's Vision – Universal Messaging Backbone

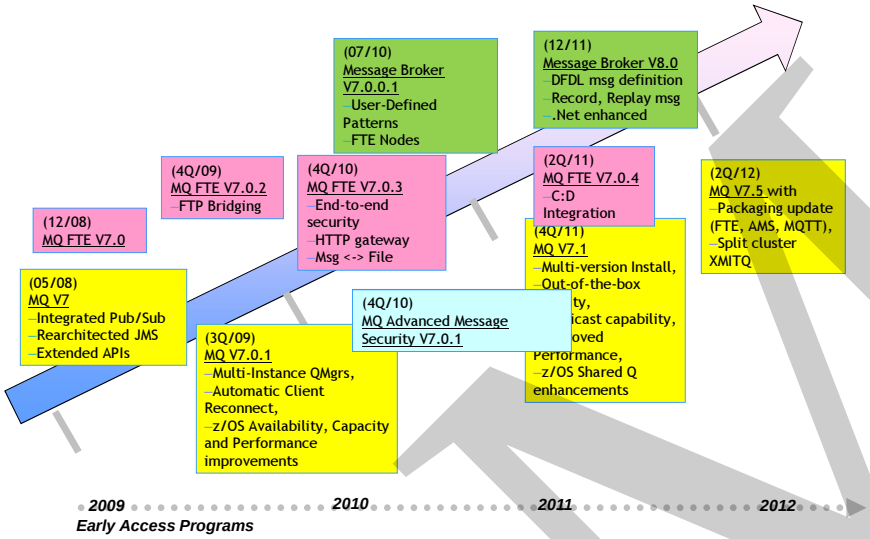


What's WebSphere MQ?

- A simple, efficient API for sending/receiving data messages
- Enables fast, reliable **asynchronous messaging** from application to application
- An industry standard for Message-Oriented-Middlewares (MOM) with 65-80% of the market.
- Can be used on over **45 different platforms** (Windows, Linux, AIX, Solaris, HP-UX, iSeries-AS/400, etc, etc., and of course…… System Z !)
- Can be used from **all major programming languages** (C, C++, COBOL, Fortran, BAL, PL/I, Java/JMS, VB, RPG, etc.)
- Includes support for the **MQ Client** – a remote API accessing the queue manager resources.
- Other unique features of MQ…
  - Provides a 2PC syncpoint manager for distributed MQ and full 2PC **transactional** participation on z (CICS, IMS, RRS)
  - Support for high-availability and load-balancing via MQ Cluster as well as **Shared queues** with Sysplex on Z
  - Support for **point to point and pub/sub** messaging
  - Support for message grouping and segmentation
  - Support for SSL authentication and encryption
  - Support for message compression
  - Exploits each platform, and in particular z/OS, taking full advantage of RACF, Sysplex, SMF, CICS & IMS Bridges, etc.



WebSphere MQ – always advancing



Why WebSphere MQ ?

Over 17 years of proven experience	Leader in Messaging technology innovation
Connect virtually anything	Broad coverage of platforms, technologies, languages Draw skills from a larger pool – use who you have today Over 9,300 certified developers for IBM Messaging alone
Most widely deployed Messaging Backbone	Over 10,000 customers using IBM Messaging Backbone Over 90% of the Fortune 50 and 9 of the Fortune 10 Over 80% of the Global 25 and 7 of the Global 10
Entrusted with Tens of billions of messages each day	Government client sends 675 million messages per day* Banking client handles over 213 million messages per day on z/OS alone*
Relied upon as the mission-critical Backbone	Financial Markets client handles $1 trillion worth of traffic per day on one MQ network* Banking client sends $7-$35 trillion worth of traffic per day on just one MQ-based SWIFT gateway*
Continuously Investing and Innovating	Over 120 patents and filings within messaging space New WebSphere MQ family products Regular enhancements, updates and new releases

** Results reported from actual MQ implementations*

WebSphere MQ for z/OS v7+ Proof-Of-Technology (POT) workshop

What's new with MQ v7.0 z/OS?

IBM EMEA panIMT Team zWebSphere team
(with thanks to IBM US ATS team for the original "Wildfire" material)

V7.0 - the big picture

WebSphere MQ v7.0 General availability
- Distributed platforms: 16 June 2008
- z/OS: 27 June 2008

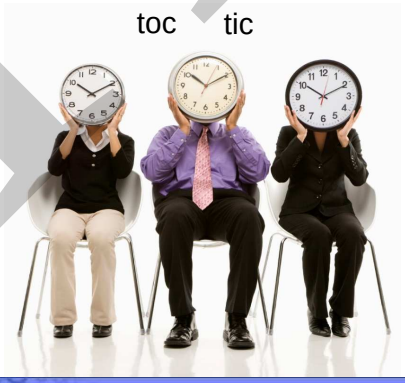
- Extension of **publish/subscribe** capabilities
  - Market asking for more Pub/Sub to further decouple applications
  - Pub/Sub motor now fully integrated into QM
- Improved support for "standards-based" messaging (translate: **JMS**)
  - Underpinning SOA and ESB architectures
  - Ease-of-use
  - Performance
  - Easier programming in any environment
    - Some features suggested by JMS requirements are useful in MQI
- Enhanced performance options
- MQ 7.0.1 added Multi-instance QMs (Distributed only)
- Many other features

Platform support

- Essentially the same platforms as V6, but….
  - **z/OS v1.8 or later**
  - Linux for System p Red Hat Enterprise (RHEL) v4 or v5, SUSE Linux (SLES) v9 ,V10 or v11 (all on 64-bit POWER processor)
  - Linux for System x Red Hat Enterprise (RHEL) v4 or v5, SUSE Linux (SLES) v9, V10 or v11, NLPOS9 FP1, IRES v2, or Red Flag Data Centre v5 (all on 32-bit or 64-bit processor)
  - Linux for System z Red Hat Enterprise (RHEL) v4 or v5, SUSE Linux (SLES) v9, V10 or v11 (in 64-bit, all on System z9 or z10, the best hardware known to mankind!)
  - AIX 5.3 plus TL04 or 6.1
  - HP-UX Itanium or PA-RISC 11i v2 or v3
  - Sun Solaris x86 64-bit v10 or Sun Solaris SPARC v9 or v10
  - i5/OS ("System I", "AS/400") V5R3 , V5R4 or V6R1
  - Windows XP Pro (SP2) or XP Pro x64 or Server 2003 (SP1) or Server Standard or Enterprise x64 Edition or Server 2008 or Vista (Business, Enterprise or Ultimate Edition, 32-bit or 64-bit) (note: No Windows 2000 support!)
- For up-to-date details, see http://www.ibm.com/software/integration/wmq/requirements/index.html
- Java 1.4.2 and later

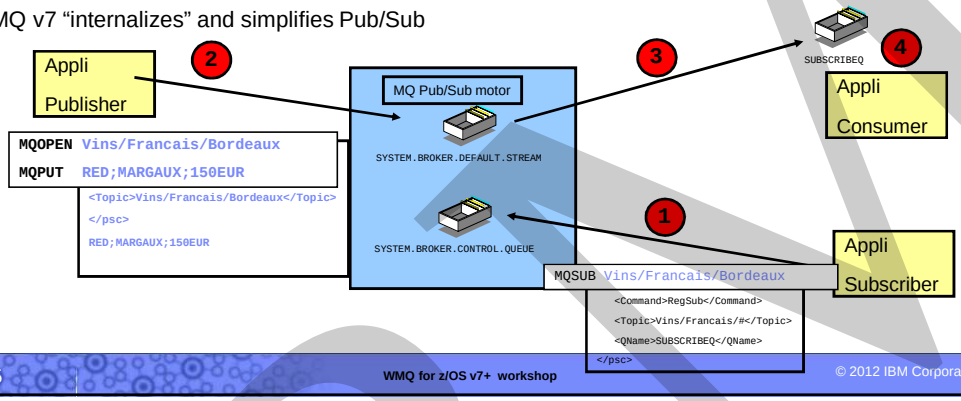
MQ v6.0 – Replaced by MQv7

- IBM announced that WebSphere MQ v6.0 (eg. Windows, Unix, Linux, z/OS will arrive at end of support on 30 September ~~2011~~ ***2012*** ***– see ENUS211-072***
- Avoid the holiday rush… start thinking about migrating to MQv7 now!
- For more details, see http://www.ibm.com/software/websphere/support/lifecycle/



Publish/Subscribe – how does it work in MQ?

- Pub/Sub is a function that allows:
  - Publishing applications to make information (eg. Messages) available to a list of « interested » applications
  - Subscribing application can receive information
  - Fuller “decoupling” model of source (“Publisher”) and target (“Consumer”) applications
- Typical uses of the Pub/Sub model:
  - Document distribution, alert notification, newsgroups, and any application where the the distribution list tends to be dynamic
- MQ v7 “internalizes” and simplifies Pub/Sub

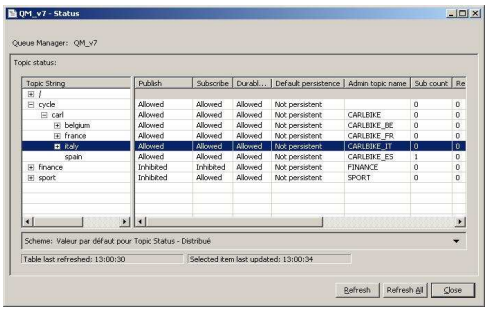


Major Pub/Sub enhancements in v7

- Pub/Sub now fully integrated into MQ, all major platforms, including z/OS
- New MQ “Topic” object
  - Publication direct to a Topic now supported
  - Not absolutely necessary for Pub/Sub, but gives administrative structure, security, etc.
- Complete administration of Pub/Sub (create, modify, status, statistics, security…)
- Interoperability with traditional Pub/Sub (MA0C) as well as WMB
- New MQI verbs – Pub/Sub for COBOL, PL/I…. Non-JMS
  - MQSUB – Register a subscription with a MQSD
  - MQSUBRQ – Request retained publication
- Modified MQI verbs for Publishing
  - MQOPEN, MQPUT with a TOPIC MQOD

Think z !

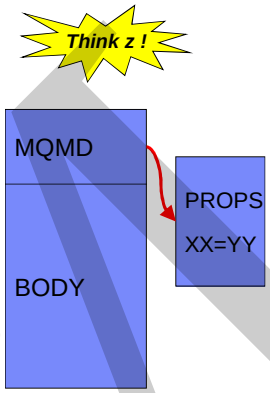
```
DEFINE TOPIC('CARLBIKE_BE') TOPICSTR('cycle/carl/belgium')
     1 : DEFINE TOPIC('CARLBIKE_BE') TOPICSTR('cycle/carl/belgium')
AMQ8690: WebSphere MQ topic created.
DISPLAY TOPIC('CARLBIKE_BE')
     2 : DISPLAY TOPIC('CARLBIKE_BE')
AMQ8633: Display topic details.
   TOPIC(CARLBIKE_BE)                      TYPE(TLOCAL)
   TOPICSTR(cycle/carl/belgium)            DESCR( )
   CLUSTER( )                              DEFPBSCP(ASPARENT)
   DEFSBSCP(ASPARENT)                      DURSUB(ASPARENT)
   MONTOP(ASPARENT)                        STATTOP(ASPARENT)
   PUB(ASPARENT)                           SUB(ASPARENT)
   DEFPSIST(ASPARENT)                      DEFPRTY(ASPARENT)
   DEFPRESP(ASPARENT)                      ALTDATE(2008-02-09)
   ALTTIME(12.49.16)                       PMSGDLV(ASPARENT)
   NPMSGDLV(ASPARENT)                      PROXYSUB(ASPARENT)
   PSSCOPE(ASPARENT)                       WILDCARD(PASSTHRU)
   MDURMDL( )                              MNDURMDL( )
```



You’ll see a lot more Pub/Sub details in another presentation….

Other JMS inspirations!

- Support for Message Properties – metadata associated with the message
  - Inspired from JMS, available via MQI to all languages
  - New verbs: MQCRTMH, MQSETMP, etc.
- Asynchronous Message Reception
  - New verb MQCB defines a callback function
  - Gives a J2EE Message Driven Bean (MDB)-like facility to the rest of us non-Java people!
  - No need for MQGET(WAIT) or MQGET(SIGNAL)
  - New verb MQCTL to start/stop callback
- Selectors
  - Use a SQL92 clause to select messages by properties including MQMD fields,
    eg. JMSType='DeptAccounting' AND Salary > 50000
  - Can be specified on MQOPEN, MQSUB, MQCB for filtering messages
  - Filtering can now be done inside queue manager for efficiency
  - Not filtering on message body contents
    - Message Broker still required for content filtering



Message Properties and Handles – Source Code

```
// This is a router app - get a message and work with it
// 1. Initial setup and read input
MQCRTMH(hConn, &CrtMsgHOpts, &hRequestMsg, &RC, &RC);
GetMsgOpts.MsgHandle = hRequestMsg;
MQGET(hConn, hObj, &MsgDesc, &GetMsgOpts, BufLen, &Buffer, &DataLen, &CC, &RC);
```

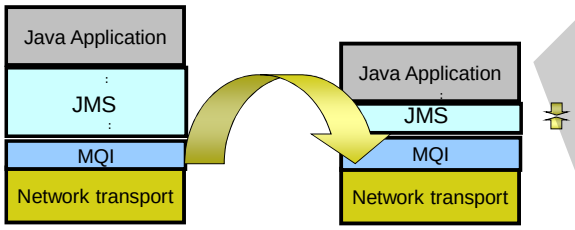
```
// 2. Forward request unchanged to a server app, named in the message
PutMsgOpts.Action = MQACTP_FORWARD;
PutMsgOpts.OriginalMsgHandle = hRequestMsg;
MQPUT(hConn, hServerObj, &MD, &PutMsgOpts, DataLen, &Buffer, &CC, &RC);
```

```
// 3. Tell requester message has been dealt with by updating existing property
Name.VSPtr = "RequestStatus";
Name.VSLength = MQVS_NULL_TERMINATED;
MQSETMP(hConn, hRequestMsg, &SetPropOpts, &Name, &PropDesc, MQTYPE_STRING, "REQUEST
  RECEIVED", 16, &CC, &RC);
PutMsgOpts.Action = MQACTP_REPLY;
PutMsgOpts.OriginalMsgHandle = hRequestMsg;
MQPUT(hConn, hReplyObj, &MD, &PutMsgOpts, DataLen, &Buffer, &CC, &RC);
```

```
// 4. Also put a completely unrelated message to a logging queue
PutMsgOpts.Action = MQACTP_NEW;
PutMsgOpts.OriginalMsgHandle = MQMH_NONE;
MQPUT(hConn, hLogObj, &MD, &PutMsgOpts, DataLen2, &LogMsgBuf, &CC, &RC);
MQCMIT(hConn,&CC,&RC);
```

WMQ Performance

Moving more JMS function into the MQI allows more function to run in compiled code

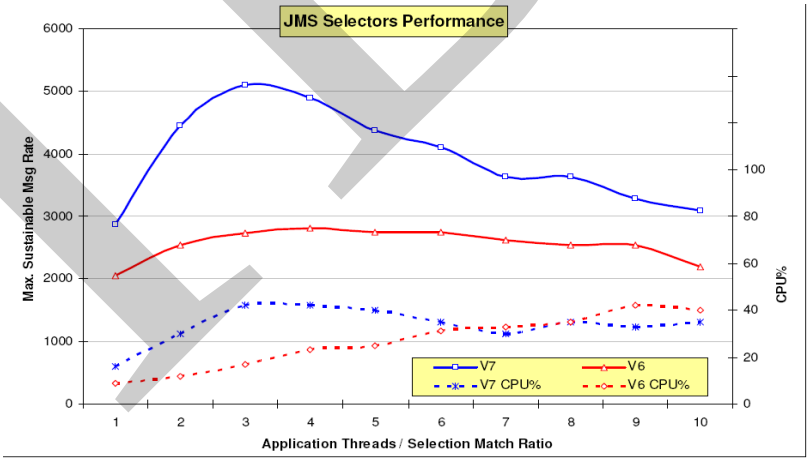


- Persistent pub/sub throughput increased up to 60%
- Non-persistent client throughput increased up to 300%
- JMS Selector rates improved up to 250%
- Message Listener (MDB) throughput improved up to 45%
  - Latency also improved

Think z !

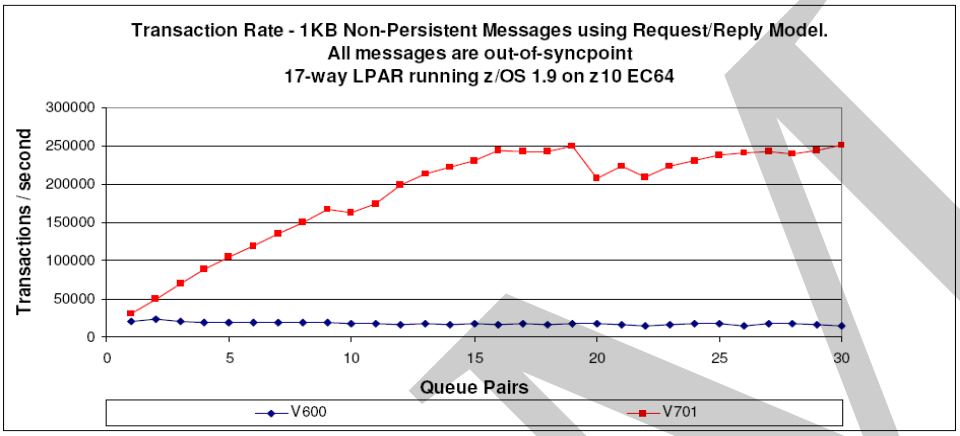
Measurements taken from pre-release code ; performance reports now available at http://www.ibm.com/software/integration/support/supportpacs/

Performance: JMS Selectors



For details, see SupportPac MP07 at
http://www.ibm.com/support/docview.wss?uid=swg24022778&myns=swgws&mynp=OCSSFKSJ&mync=E

Performance: Maximum Request/Reply messaging



For details, see SupportPac MP1G at
http://www.ibm.com/support/docview.wss?rs=171&uid=swg24024589&loc=en_US&cs=utf-8&lang=en

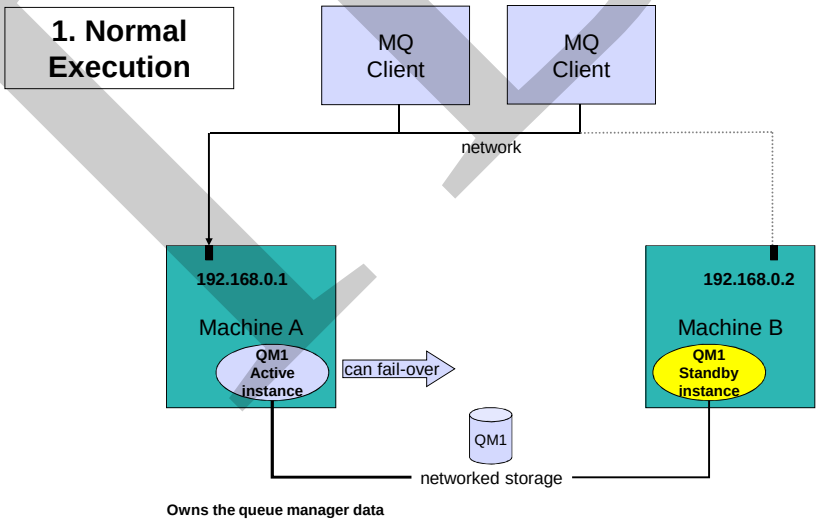
Other MQI Enhancements

- Cooperative Browsing and Message Tokens
  - Efficient interface for applications reading from the same queue
  - Example: "master" program browses a queue telling "slaves" which message to work with, based on elements within the message
  - No races – messages locked but available to any cooperating process

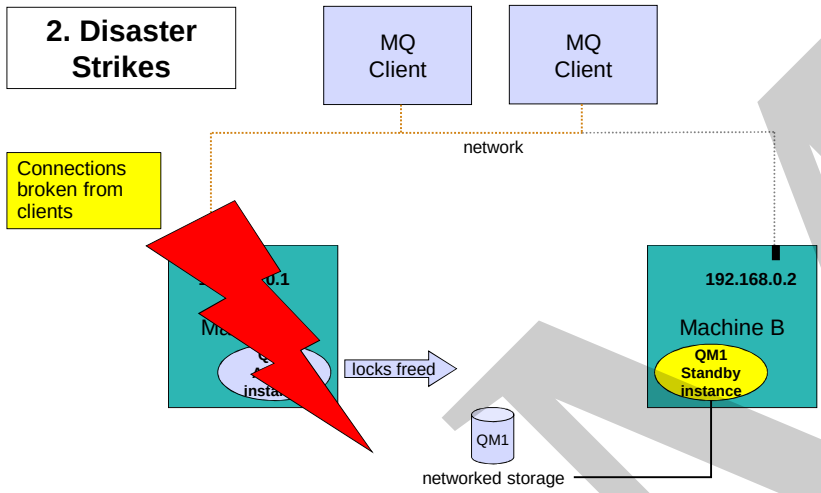
Programming in Java

- JMS read/write access to all MQMD fields as properties
  - Have to explicitly enable this in the application program
  - Allows the application to go beyond the JMS specification
- JMS access to the raw message content
  - Can treat the whole body as a byte array property
  - Can see RFH2 folders that would normally be stripped
- Message Header Classes for Java
  - Updated and supported version of MS0B SupportPac
  - Makes it easy to build and parse PCF structures
  - Extended to handle other MQI message header formats
    - eg MQCIH, MQDLH classes

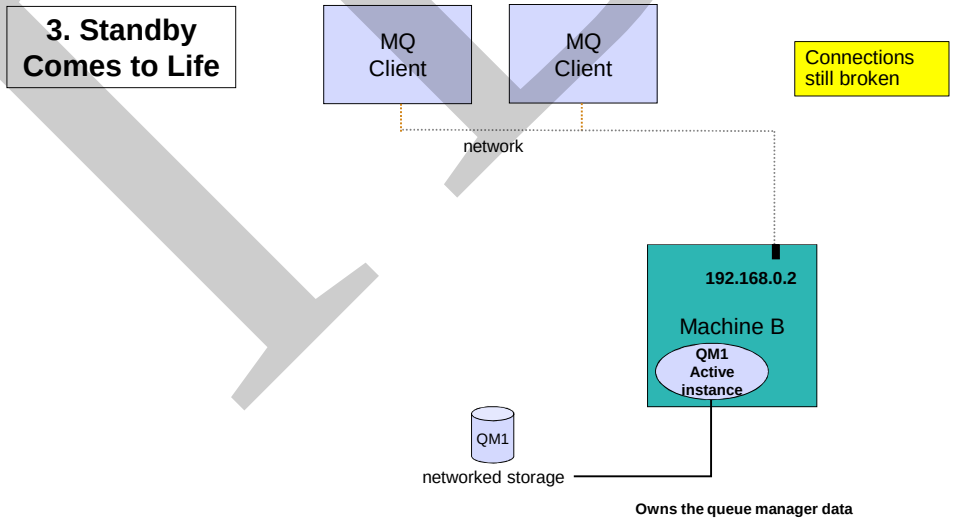
Multi-instance Queue Managers (Distributed MQ only, MQ v7.0.1)



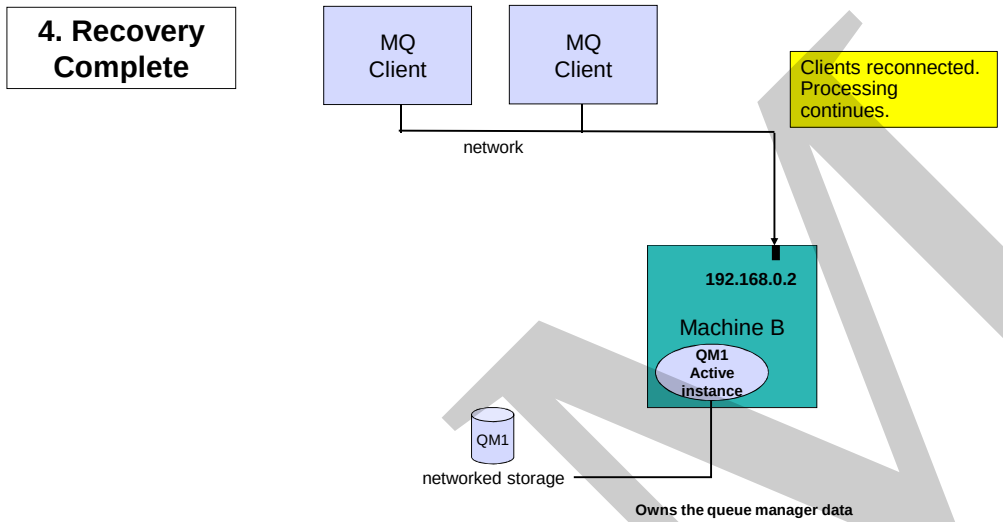
Multi-instance Queue Managers



Multi-instance Queue Managers



Multi-instance Queue Managers



Multi-instance queue managers: Details

- MQ is NOT becoming an HA coordinator
  - Generally, if other resources also required, use an HA coordinator such as HACMP
  - Service objects can restart applications with qmgr but limited control
  - Message Broker integrates with and exploits this MQ function
- The IP address is not taken over
  - Channel config needs all possible addresses unless you use external IPAT or intelligent router
  - CONNAME('host1(port1),host2(port2)') syntax extension on all platforms including z/OS
- Support for networked storage over modern network file system protocols
  - For example, NFS v4 (not v3)
  - Tool shipped to validate configuration
- New options for crtmqm/strmqm/endmqm to control operations
  - Cannot guarantee which instance becomes the primary
- Removes need for MC91, which will be withdrawn
  - crtmqm now does equivalent of MC91's hacrtmqm

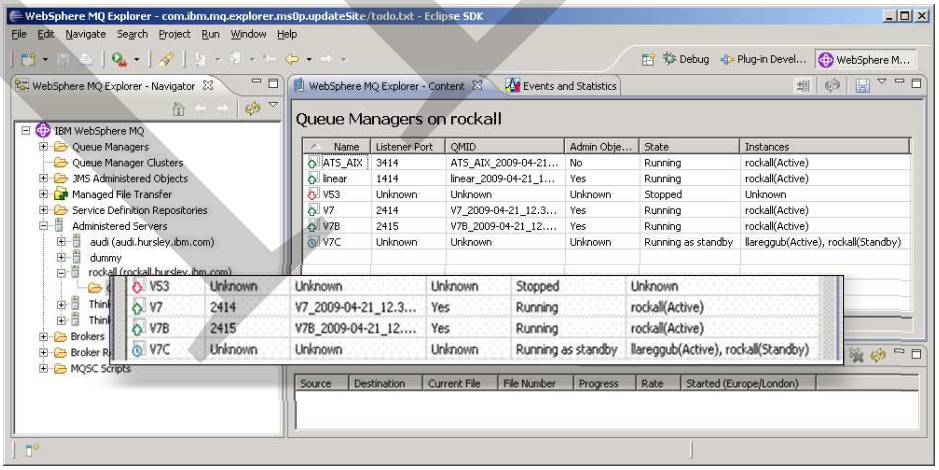
Multi-instance queue managers: How it looks

- Enhanced dspmq
- New option for dspmq to output English-only text
  - Useful for programmable parsing

```
$ hostname
rockall
$ dspmq -x
QMNAME(V7)       STATUS(Running)
    INSTANCE(rockall)   MODE(Active)
QMNAME(V7B)      STATUS(Running)
    INSTANCE(rockall)   MODE(Active)
QMNAME(V7C)      STATUS(Running as standby)
    INSTANCE(llareggub) MODE(Active)
    INSTANCE(rockall)   MODE(Standby)
```


Multi-instance queue managers: How it looks

- As a graphical example, SupportPac MS0P V7.0.1



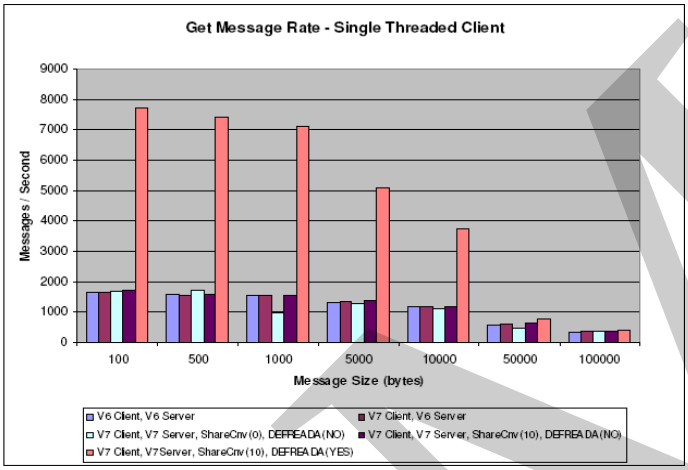
Distributed Platforms: New Client Features

- New client features to improve performance and to provide greater availability have been added with V7.0.x
- While not strictly applicable to z/OS, these features have generated a great deal of interest.

Client Performance

- Traditional WMQ non-persistent messages more reliable than some need
- "Read Ahead" for Receiving Messages/Publications:
  - Messages sent to a client in advance of MQGET, queued internally
  - Administrative choice – no application changes needed
  - Higher performance in client
- "Asynchronous Put" for Sending/Publishing Messages:
  - Application can indicate it doesn't want to wait for the real return code
    - Maybe look for return code later – **MQSTAT** verb
  - Maintains transactional semantics
  - Higher performance in client

Performance: MQ Client Read-ahead



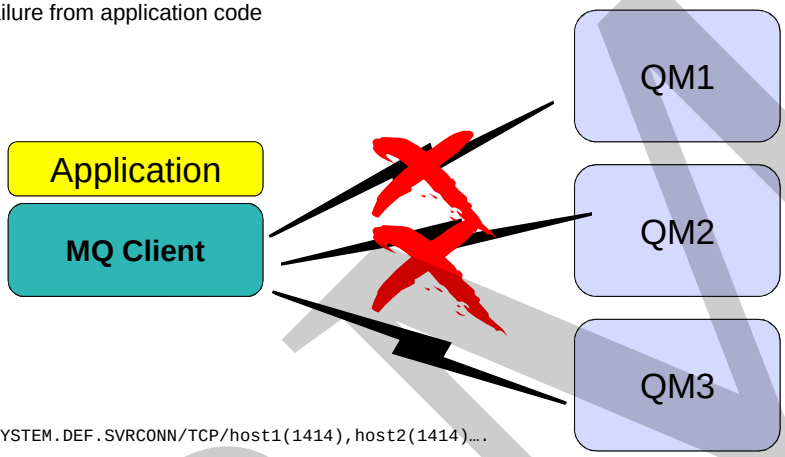
For details, see SupportPac MP1F at
http://www.ibm.com/support/docview.wss?rs=171&uid=swg24024589&loc=en_US&cs=utf-8&lang=en

Client Connection Management

- Shared Client Conversations
  - Several connections from the same process can be handled on the same socket
  - Faster startup for the second and subsequent connections
- Implementation also gives us more heartbeat opportunities
  - Faster failure notification for clients
- Client Connections
  - Automatic workload distribution via CCDT
  - Control number of connected clients at a queue manager (MAXINST, MAXINSTC)
- Free connections to z/OS for administration programs like WMQ Explorer
  - Limited number of clients permitted by V7 license without CAF

Automatic Client Reconnection

- Client library provides necessary reconnection logic on detection of a failure
- Hides failure from application code




```
MQSERVER=SYSTEM.DEF.SVRCONN/TCP/host1(1414),host2(1414)….
```


Automatic Client Reconnection

- Tries to hide queue manager failures by restoring current state automatically
  - For example, if MQPUT returns error, client reruns MQCONN/MQOPEN/MQPUT internally
- Uses the list of addresses in CONNAME to find queue manager
  - MQSERVER environment variable also understands list
  - MQSERVER=SYSTEM.DEF.SVRCONN/TCP/host1(1414),host2(1414)
- Can reconnect to the same or different Queue Manager
- Re-opens queues and other qmgr objects, re-establishes subscriptions
- Reconnection interval is backed off exponentially on each unsuccessful retry
  - Total timeout is configurable – default 30 minutes.

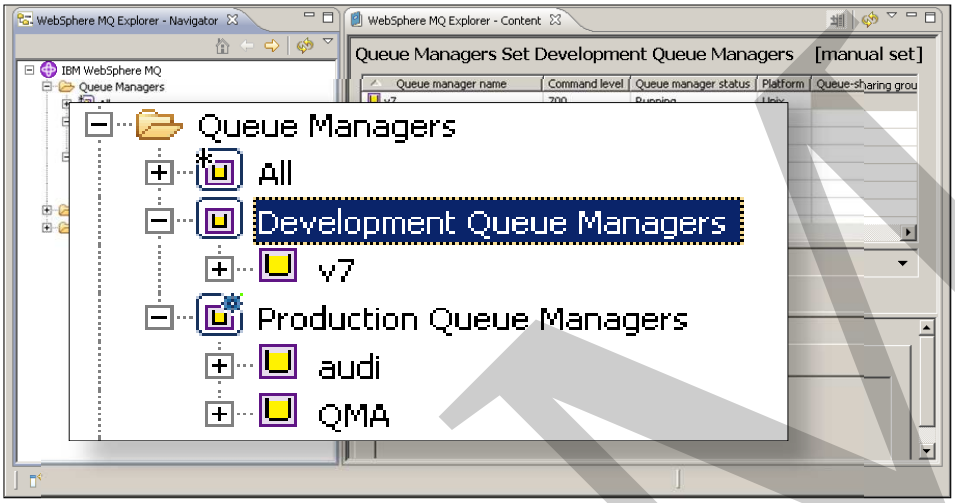
Automatic Client Reconnection: Details

- Enabled in application code or ini (eg. mqclient.ini) file
  - Event Handler callback shows reconnection is happening if app cares
- Tries to keep dynamic queues with same name
  - So replies may not be missed
- Not all MQI is seamless, but majority repaired transparently
  - eg a browse cursor would revert to the top of the queue, non-persistent messages will have been lost during restart, non-durable subscriptions may miss some messages, in-flight transactions backed out, hObj values maintained
- Some MQI options will fail if you have reconnection enabled
  - Using MQGMO_LOGICAL_ORDER, MQGET gives MQRC_RECONNECT_INCOMPATIBLE
- Initially just in MQI and JMS – not the other OO classes
  - Requires both client and server to be V7.0.1 level with SHARECNV>0
  - Server can be z/OS

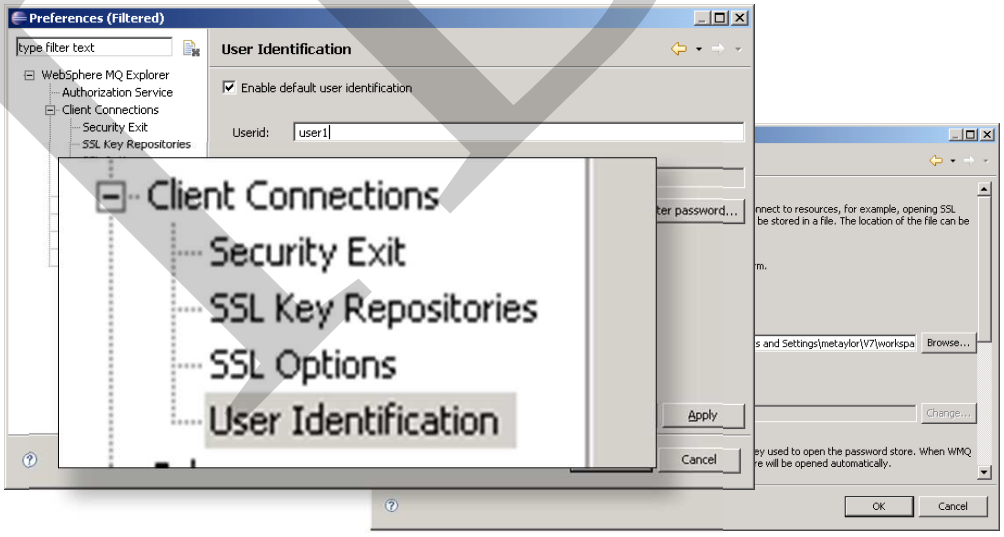
WMQ Explorer Enhancements

- Sets
  - Queue Managers can be partitioned into sets within the Navigator
  - For example "Test", "Production"
- Security Configuration
  - Easy to define channel exits, userid/password configurations
  - Configured for each queue manager or for all queue managers in a set
  - Password manager included
  - Still recommend security exit or service for authentication at the server
- Tighter JMS integration
  - Creating an queue/topic can define a JMS destination at the same time
- Message browser configuration
  - Number, size of messages
- Plug-in Migration
  - Explorer now based on Eclipse 3.3 – compatibility not guaranteed
  - Major change is availability of supported PCF classes
- Using MQ Explorer with WebSphere MQ for z/OS
  - you can use the MQ Explorer, which runs on the Windows and Linux on x86 platforms, to manage the z/OS parts of your WebSphere MQ network without having to buy the Client Attach Feature

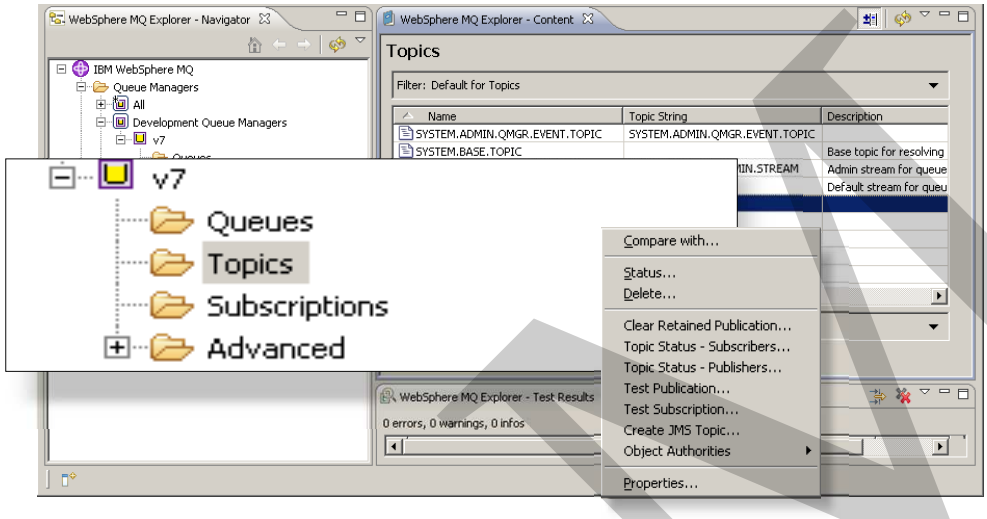
Queue Manager Sets



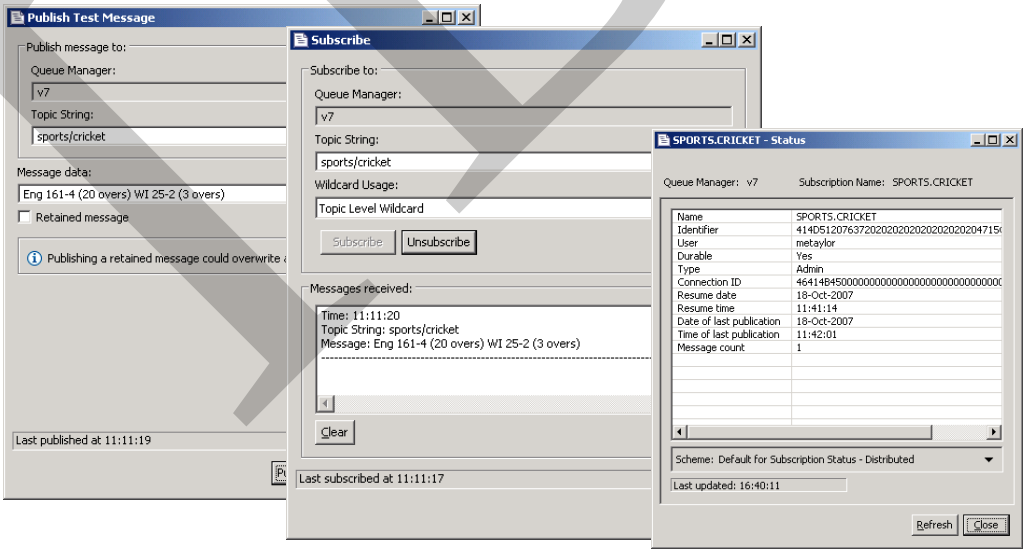
WMQ Explorer Preferences



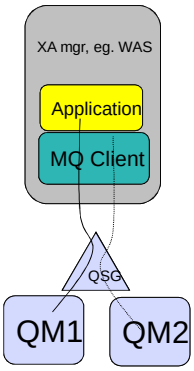
Publish/Subscribe in the WMQ Explorer



Testing Publish and Subscribe

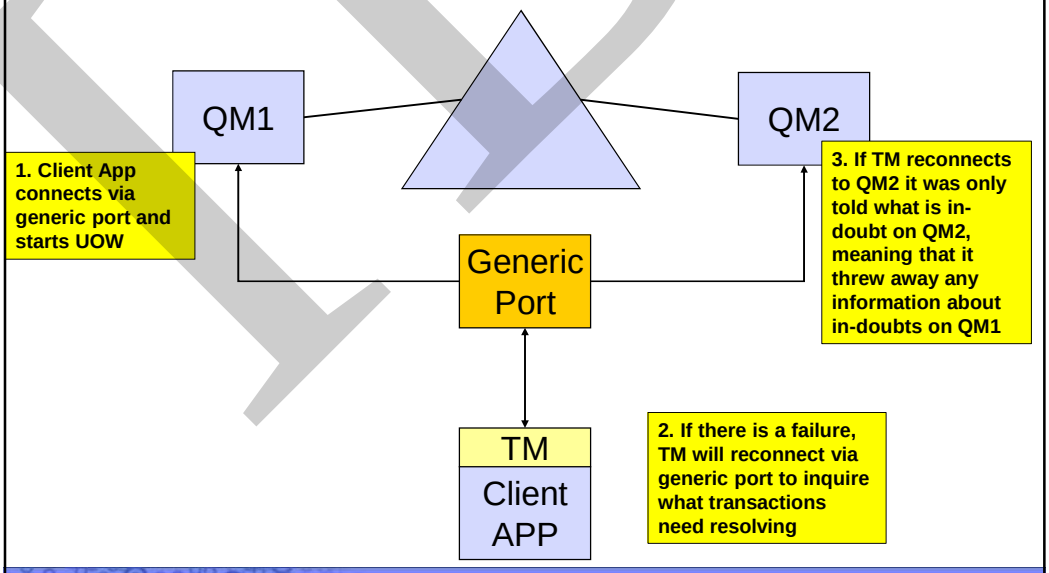


Group-level Units of Recovery for z/OS

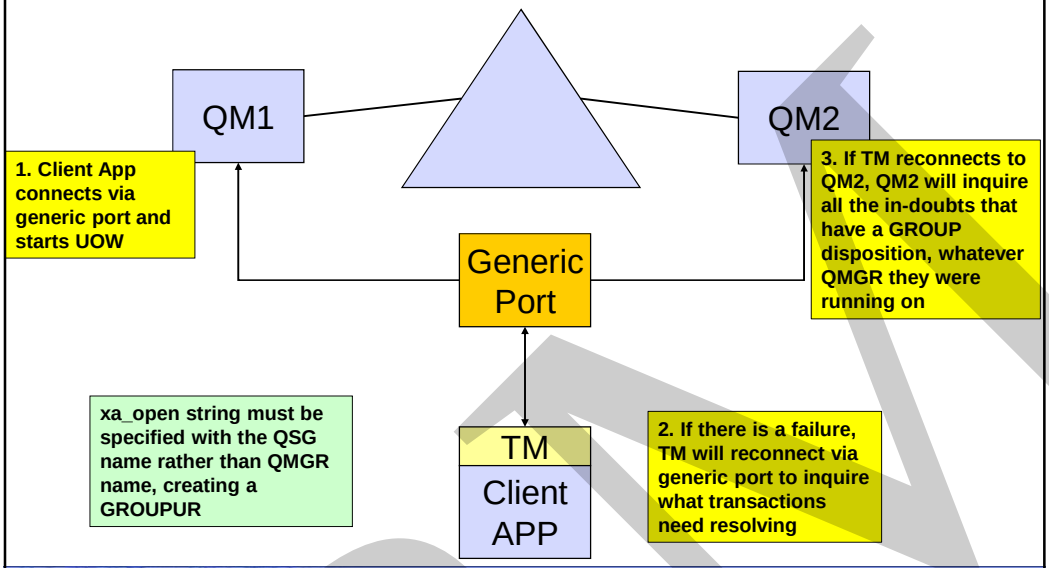


- A client's two-phase/global transaction can now be owned by a QSG
  - Instead of by individual queue managers
- These in-doubt transactions can be resolved on any QMGR in the QSG.
  - Required for support 2-phase commit resolution while connected to the QSG
- Requires use of the Extended Transactional Client
  - For example, from WAS
  - Configure the WAS client connection with the QSG name rather than the QMGR name

GROUPUR: The Problem (pre-MQ v7.0.1)



GROUPUR: The Solution (V7.0.1)



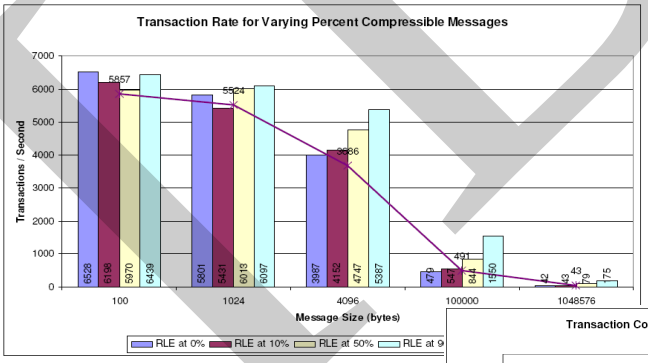
z/OS Constraint Relief – 'Above the bar' exploitation

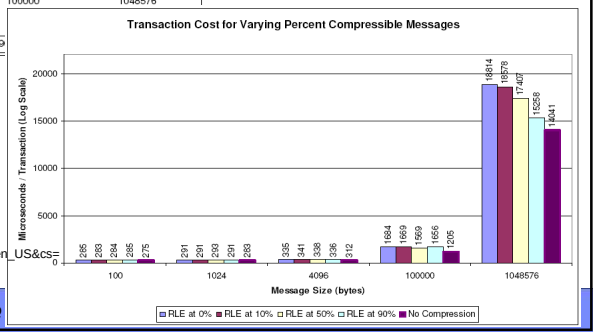
- In V7.0, the queue manager started to exploit 64-bit addressing
  - New Pub/Sub features
  - Intra-Group Queuing (IGQ) buffer
- In V7.0.1 more Queue Manager storage moves to 64-bit
  - 64-bit Queue Indices
  - 64-bit Lock Manager
  - 64-bit Security Cache
- Can have more open queues, more messages on indexed queues etc

z/OS Log Compression

- Can increase the throughput possible for persistent messages
- May reduce the size of your logs
  - Dependent on your message body content
  - Useful if you are DASD or I/O bandwidth constrained.
- RLE (run-length encoding) of “insert” log records for private queue messages
  - Will not compress shared queue log records
  - SMF 115 records updated to show compression rates achieved etc
- Controlled via zPARM option at queue manager level.
  - COMPLOG(NONE) or COMPLOG(RLE) in CSQ6LOGP
  - Can also be viewed/controlled via DISPLAY LOG / SET LOG

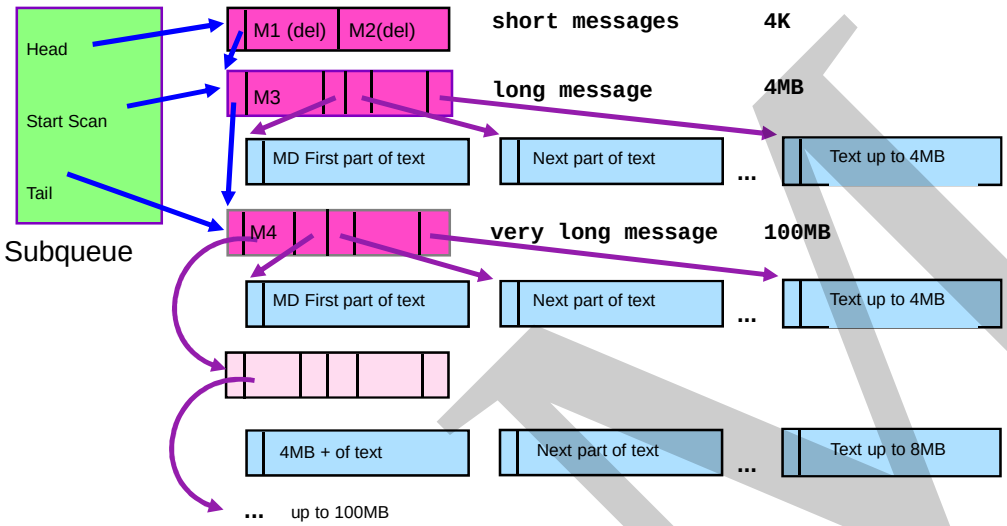
MQ Log Compression – transaction rate & cost



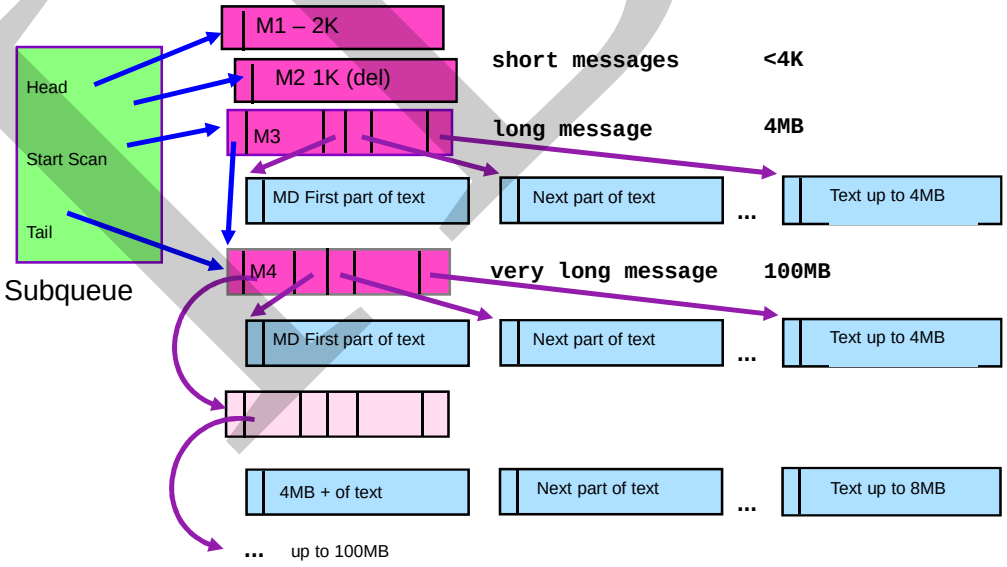


•For details, see SupportPac MP1G at http://www.ibm.com/support/docview.wss?rs=171&uid=swg24024589&loc=en_US&cs=utf-8&lang=en

Small message handling: Pre-V7.0.1 Local Message Queue Storage



Small message handling: Post-V7.0.1 Local Message Queue Storage



MQv7 Migration

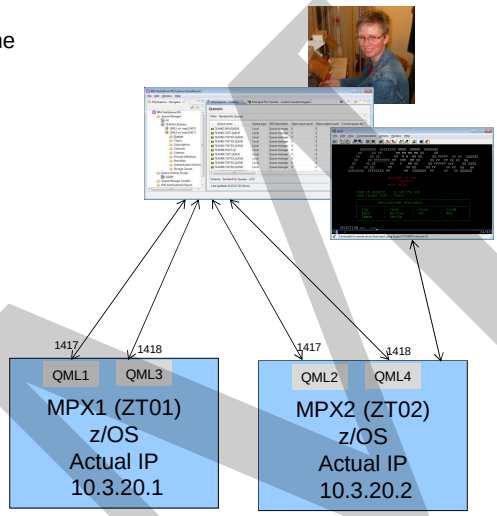
- Basically, migration to MQv7 is just like all other version migrations
- MQv7 is “upward compatible” with MQv6. Programs work as usual. A few notable exceptions…
  - Note that most JMS error codes are now different as a result of the new “layering” of MQ’s JMS
  - RFH2 headers generated by JMS are no longer presented to MQv7 applications by default; the header information is now included in the Message Properties
  - The new MQv7 integrated Pub/Sub motor is PSMODE(ENABLED) by default for a new QM. Set to PSMODE(COMPAT) if using WMB Pub/Sub (default for a migration)
- On Distributed, you (still) can only have one MQ version active at a given time.
  - You can have multiple MQs active (on different platforms) with MQ Clusters, and stop/start channels for a non-stop migration
  - Typically migrate MQ Cluster Full Repositories first
- On z/OS, you can have multiple versions active
  - You can have multiple MQs active with Shared Queues, and just stop/restart to migrate a QM
  - You can migrate backwards if necessary – see technotes to enable this, BUT…
  - Be sure to set the OPMODE=(NEWFUNC,701) in CSQ6SYSP if you want/need all the newest functions (but this can prevent backward migration!)
- Programs compiled/linked with MQv7 and LE may need to be re-LinkEdited for MQ 7.0.1
- MQ v7.0.1 uses by default 1 message per page, unlike earlier MQ versions that placed multiple messages in a single page for small messages. This increases the size requirement for pagesets (but was done to enhance performance).
- See SC34-6948 (csqzao02) MQv7 Migration Information, and SG24-7583 Redbook
- There are two good Technotes that provide extra information on migration to MQ7.0.1 (see http://www.ibm.com/support/docview.wss?uid=swg21406779 and swg21408135)

Summary

- WMQ V7.0 was an extremely rich series of product updates, such as
  - Several new verbs
  - JMS-like facilities for the non-Java developers
  - A fully integrated Pub/Sub motor
  - Many MQ Explorer enhancements
  - Multi-instance queue managers
  - MQ Client performance and HA enhancements
  - Many other goodies!
- The stage has now been set for new functionality to be added in more granular increments.

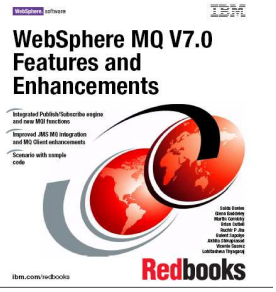
Lab time

- Lab Connectivity and environment
  - A very simple lab just to get you set up
  - You'll configure your MQ Explorer to view the 4 x z/OS queue managers
  - You'll verify that you can logon to TSO and view queues
- Lab MQ v7 Explorer enhancements
  - You'll explore a few of the new MQ v7 Explorer enhancements
- Userids: TEAMXX
- Passwords: _____



MQ Bibliography

- GC34-6926 WebSphere MQ v7 for z/OS Concepts and Planning Guide
- SC34-6929 WebSphere MQ v7 for z/OS System Administration Guide
- SC34-6927 WebSphere MQ v7 for z/OS System Setup Guide
- SC34-6939 WebSphere MQ v7 Application Programming Guide
- SC34-6940 WebSphere MQ v7 Application Programming Reference
- SC34-6932 WebSphere MQ v7 Security
- SC34-6933 WebSphere MQ v7 Queue Manager Clusters
- SC34-6942 WebSphere MQ v7 PCF and Administration Interface
- SC34-6948 WebSphere MQ v7 Migration Information
- GI13-0529 WebSphere MQ for z/OS v7.0 Program Directory
- SG24-7583 WebSphere MQ v7 Features and Enhancements
- SG24-7839 High Availability in WebSphere Messaging Solutions



The complete MQ library is available in PDF at
http://www.ibm.com/software/integration/wmq/library/
Or online at
http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp

WebSphere MQ for z/OS v7+ Proof-Of-Technology (POT) workshop

Publish/Subscribe

IBM EMEA panIMT Team zWebSphere team
(with thanks to IBM US ATS team for the original “Wildfire” material)

Agenda

- Concepts of publishing and subscribing
- Topics
- Subscriptions
- Administration

Publish/Subscribe

- **Publish/Subscribe** (or pub/sub) is an asynchronous messaging paradigm where senders (publishers) of messages are not programmed to send their messages to specific receivers (subscribers). Rather, published messages are characterized into classes, without knowledge of what (if any) subscribers there may be.
- Subscribers express interest in one or more classes, and only receive messages that are of interest, without knowledge of what (if any) publishers there are. This decoupling of publishers and subscribers can allow for greater scalability and a more dynamic network topology.
- Pub/sub is a sibling of the Message Queue paradigm, and is typically one part of a larger Message-Oriented Middleware solution. Most messaging systems support in their API (e.g. JMS) both the pub/sub and Message Queue models.

Point to Point Examples

- Post Card
  - Goes to just the person I send it to.
- eMail
  - Might go to lots of people but I get to choose exactly who gets it.
- Message Queuing
  - If I put a message it will go to one consumer.

With Point to Point, the sender (MQPUTer) explicitly defines the receiver (MQGETer) at MQOPEN time

Publish / Subscribe Examples

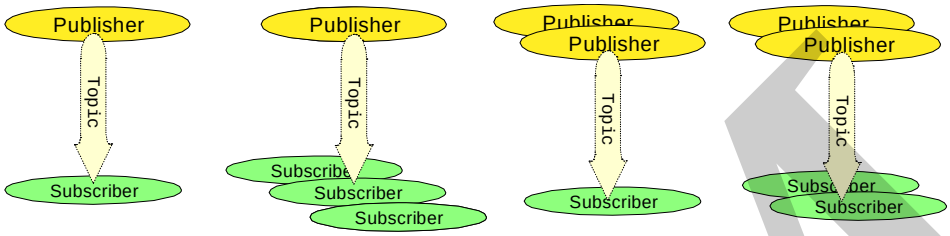
- Magazine Publishing
  - In the US – almost 10,000 titles published in US (only 2,000 are considered 'major publications')
- Airline Departure Boards
  - Boards might display (subscribe to)
    - All departures
    - Departures at this Terminal
    - Departures by this Airline
- RSS News Feeds

With Publish/Subscribe, the sender (MQPUTer) never explicitly defines the receiver; he never even knows if there are any receivers!

Moving Toward Loose Coupling

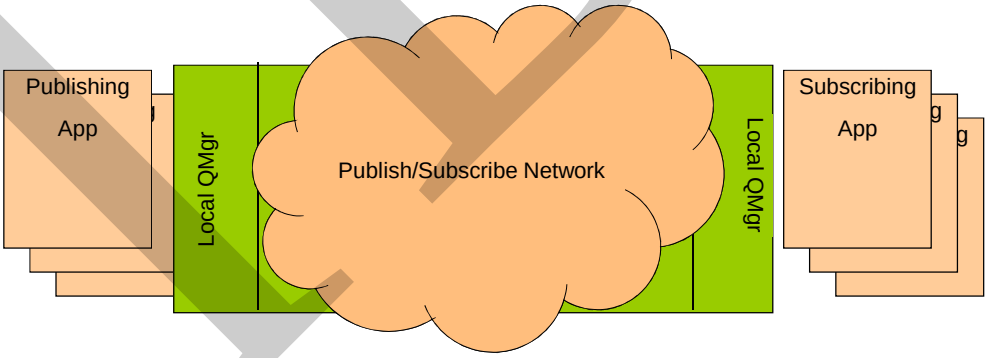
- Monolithic
  - Work is all on one computer!
- Client-Server
  - Work is distributed
  - Detailed knowledge of topology and connectivity needed
  - All components must be available for the application to work.
- Message Queuing
  - Work is distributed
  - Cooperating components can be available at different times
  - Work will queue at busy times or if a component is down
  - Connected by Queue Names.
- Publish/Subscribe
  - Cooperating components exchange data through an infrastructure that identifies the "subject" of the data.

Loose coupling with Pub/Sub



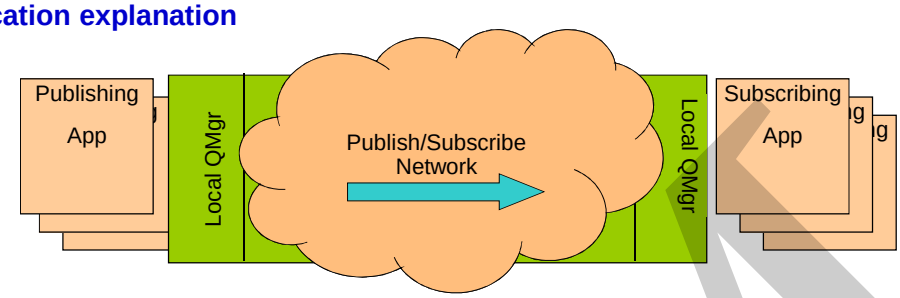
- No need to create a logical link between provider(s) and consumer(s)
  - Point to point does
- New providers can be transparently added ☺
- New consumers can be transparently added ☺ ☺ ☺
- No disruption to existing providers to consumers relationship

Publish / subscribe applications



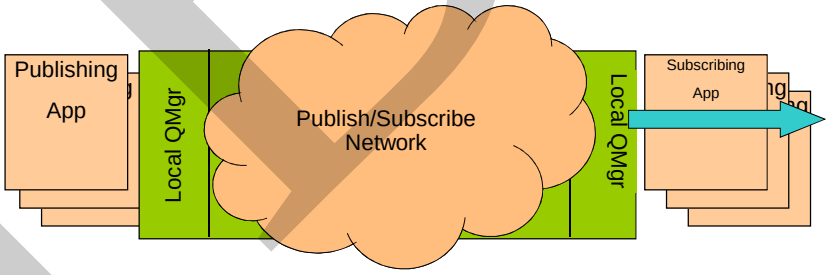
- Applications connect to local queue manager
- Publish/Subscribe asymmetry compared to Put/Get
- Can use JMS or any MQI language
- Can also interact with MQ V6 queued Pub/Sub command messages

Publication explanation



- Publication is MQPUT to a topic string
- Topic string can be created in three ways
  - Specify whole topic string in application.
  - Pre-defined topic object contains the topic string.
  - Concatenation of pre-defined string from topic object plus partial topic string from application.

Subscription explanation



- Subscription
  - A subscription involves two key elements
    1) Identifying the topic string, including wildcards, being subscribed to.
    2) Identifying the destination that qualifying messages are to be placed on.
  - Note the subscription call MQSUB does not get any messages. These must be read from the destination queue with an MQGET call.

Publish/Subscribe using MQI - Summary

- The verbs used are:-
  - MQOPEN
  - MQPUT
  - MQSUB (new in v7)
  - MQGET
  - MQSUBRQ (new in v7)
  - MQCLOSE
- New structures to accompany new verbs
  - MQSUB – MQSD – Subscription Descriptor
  - MQSUBRQ – MQSRO – Subscription Request Options
- Additionally, the MQOPEN, MQPUT and MQCLOSE were extended to include TOPICs

Agenda

- Concepts of publishing and subscribing
- **Topics**
- Subscriptions
- Administration

WebSphere MQ V7 take on Pub/Sub

- Central concept is the TOPIC STRING.
- Messages are “published” to a TOPIC STRING.
  - Each publication is to a single TOPIC STRING.
- Subscribers subscribe to TOPIC STRINGS.
  - Subscribers may be subscribing to many topics, and may use wild cards in topic names.
- TOPIC STRINGS are organized into a logical hierarchy.
- TOPIC STRINGS can be secured

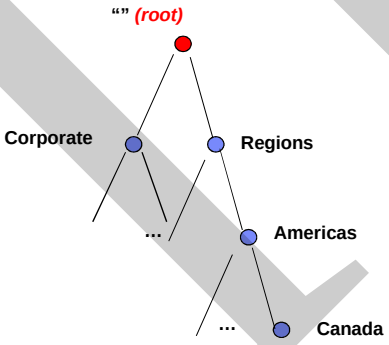
Concepts – Topic String

- A topic string “connects” publishers and subscribers.
- Has “structure” and semantics.
- The hierarchical structure is supplied by the “/” slash character.
- A topic string is practically unlimited in length - a 10,000 byte string.
- The semantics are implied by the use of wild cards (“#” ‘+’ ‘*’) in subscriptions.

Concepts – Topic String

- **Topic Sting Hierarchies – order is important**
  - Corp/HR/Americas is not the same thing as Corp/Americas/HR
- **Case sensitive**
  - Corp/HR/Americas is not the same thing as CORP/Hr/Americas

Concepts – Topic Tree



- A topic tree is a representation of the topic hierarchy.
- It has a root node at the very top.
- It is implied from the complete set of topic strings in use.
- A topic string may be defined, published to, subscribed to by applications or admin tools.
- There is **not necessarily** a one-to-one mapping between topic objects and nodes in the tree.

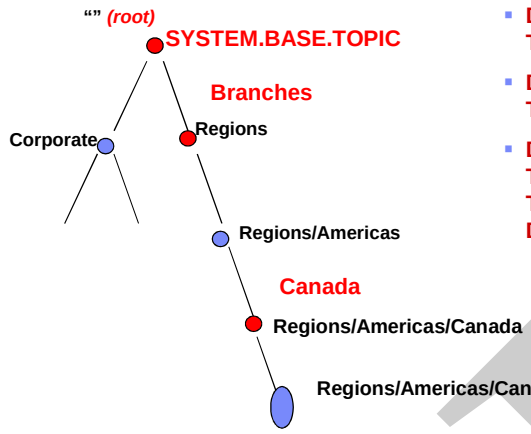
Concepts – Topic Object

- A Topic Object is an *administrated* control point for the Topic Tree.
- The topic object name follows the same rules as other WMQ objects (48 characters for the name).
- Topic string is a required property of a Topic Object.
- Topic Objects may have additional properties.
- Security can only be set on Topic Objects.

Concepts – Topic Object

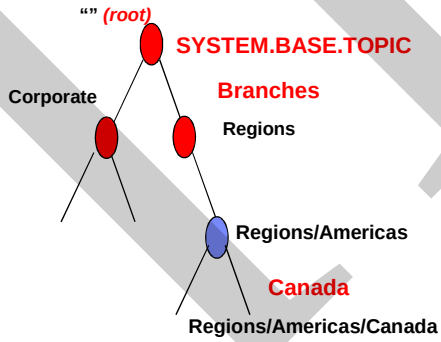
- Application programs using the MQI can use the topic object as a full or partial alternative to topic string when publishing or subscribing.
- A Topic Object name does not have to be logical or meaningful.
  - You can define a Topic Object with the name FRED with a topic string of 'Corp/Americas/HR'.
  - DEFINE TOPIC(*FRED*) TOPICSTR('Corp/Americas/HR')
- You can define a topic object to your QSG.
  - In effect the topic is defined to all members of the QSG

Concepts – Building the Topic Tree



- DEFINE TOPIC(Branches) TOPICSTR('Regions')
- DEFINE TOPIC('Canada') TOPICSTR('Regions/Americas/Canada')
- DEFINE SUB('MySub') TOPICOBJ('Canada') TOPICSTR('Ontario') DEST('MY.QUEUE')

Topic Objects



- Topic object parameters include:
  - **TOPICSTR** – Required - Topic string, up to 10K characters long TOPICSTR('/tv/films/horror')
  - **PUB** – Whether applications can publish to this topic PUB(*ASPARENT* | *ENABLED* | *DISABLED*)
  - **SUB** – Whether applications can subscribe to this topic SUB(*ASPARENT* | *ENABLED* | *DISABLED*)

Agenda

- Concepts of publishing and subscribing
- Topics
- **Subscriptions**
- Administration

Subscriptions

- When you subscribe to a topic string with WebSphere MQ you can specify where you want the messages delivered.
- Your subscription can be durable, or not.
  - Non-durable subscriptions mean messages are delivered to you only while you are connected.
  - Messages for durable subscriptions are delivered to you when you next connect.
- Subscriptions can be created programmatically or administratively.

Concepts – Subscription

- An application subscribes to a topic string.
  - Wild cards may be used.
  - Corresponds to a node in the topic tree.
- A subscription can be durable or non durable.
- A subscription will usually be set up by an application.
  - They can also be created administratively.

Concepts – Subscription - continued

- A subscription identifies a destination (queue) which receives a copy of every publication sent to the topic string.
- A subscription may use a 'managed destination'
  - The queue manager will create the destination queue from a model when the subscription is activated
  - The destination queue may be a temporary or permanent dynamic queue
    - If the subscription is durable, the queue created must be a permanent dynamic queue.
    - Using temporary dynamic queues on z/OS is costly and discouraged.
  - These are sometimes referred to as 'managed subscriptions'.

Wildcards

- **Wildcards may be used to replace parts of a topic string when subscribing or displaying the topic status.**
- **You specify the wildcard type on the subscription**
- **WMQ V7+ uses topic level wildcards**
  - '#' – is the multi-level wildcard
    - Subscribing to 'Branches/#' will subscribe to everything published to 'Branches', 'Branches/Americas','Branches/Americas/Canada', 'Branches/Americas/USA/', etc.
  - '+' – is the single level wildcard
    - Subscribing to 'Branches/+' will subscribe to everything published to 'Branches/Americas', 'Branches/Europe', etc.
    - It will not subscribe to 'Branches/Americas/Canada', 'Branches/Americas/USA' etc.

Durable vs. Non-Durable Subscriptions

- **Durable Subscriptions**
  - A Durable Subscription remains in effect when the subscribing application disconnects.
  - After a disconnect, all publications will be available to the application when it reconnects.
  - Durable subscriptions must have a unique subscription name
- **Non-Durable subscriptions**
  - Non-durable subscriptions are removed when the application disconnects. Subsequent publications are not available to the application.
  - Non-durable subscriptions may have a unique subscription name. Note that a unique subscription name makes it easy to identify in an admin display.

Publishing application

- MQOPEN a topic
- MQOD describes a topic to publish to:
  - ObjectType
    - MQOT_Q for point-to-point
    - MQOT_TOPIC for publish
  - ObjectString/ObjectName
    - Where to publish
- MQPUT a message

```
OpnOpts = MQOO_OUTPUT
        | MQOO_FAIL_IF_QUIESCING;
MQOPEN( hConn,
        &ObjDesc,
        OpnOpts,
        &hObj,
        &CompCode,
        &Reason);
MQPUT ( hConn,
        hObj,
        &MsgDesc,
        &pmo,
        strlen(pBuffer),
        pBuffer,
        &CompCode,
        &Reason);
```

```
MQOD   ObjDesc = {MQOD_DEFAULT};

ObjDesc.ObjectType              = MQOT_TOPIC;
ObjDesc.Version                 = MQOD_VERSION_4;
ObjDesc.ObjectString.VSPtr      = "Price/Fruit/Apples";
ObjDesc.ObjectString.VSLength   = MQVS_NULL_TERMINATED;
```


Subscribing application

- MQSUB verb
- Subscription Descriptor (MQSD) describes the topic
  - MQSD.ObjectString
  - MQSD.ObjectName
- MQGET to consume publications from the returned hObj
  - when MQSO_MANAGED used

```
MQSUB ( hConn,
        &SubDesc,
        &hObj,
        &hSub,
        &CompCode,
        &Reason);

MQGET ( hConn,
        hObj,
        &MsgDesc,
        &gmo,
        strlen(pBuffer),
        pBuffer,
        &DataLength,
        &CompCode,
        &Reason);
```

```
MQSD   SubDesc = {MQSD_DEFAULT};
SubDesc.ObjectString.VSPtr     = "Price/Fruit/Apples";
SubDesc.ObjectString.VSLength  = MQVS_NULL_TERMINATED;
SubDesc.Options                = MQSO_CREATE
                               | MQSO_MANAGED
                               | MQSO_FAIL_IF_QUIESCING;
```

Agenda

- Concepts of publishing and subscribing
- Topics
- Subscriptions
- **Administration**

Administering Topic Objects

- **MQSC**
  - DEFINE TOPIC
  - DELETE TOPIC
  - ALTER TOPIC
  - DISPLAY TOPIC
- **PCF**
- **MQ Explorer**
- **Limited support from z/OS menus (CSQOREXX) due to potential Topic string length**
- **Like other WMQ objects**
  - They can be secured
  - You will be able to migrate between releases
  - Follows the usual naming rules

TOPIC administration syntax example

```
ALTER TOPIC( topic_name )
  [ CLUSTER( cluster_name ) ]          [ COMMINFO( comminfo_name ) ]
  [ CUSTOM( string ) ]
  [ DEFPRESP( ASPARENT | SYNC | ASYNC ) ]
  [ DEFPRTY( ASPARENT | integer ) ]
  [ DEFPSIST( ASPARENT | NO | YES ) ]    [ DESCR( string ) ]
  [ DURSUB( ASPARENT | YES | NO ) ]
  [ MCAST( ASPARENT | ENABLED | DISABLED | ONLY ) ]
  [ MDURMDL( q_name ) ]                [ MNDURMDL( q_name ) ]
  [ NPMSGDLV( ASPARENT | ALL | ALLDUR | ALLAVAIL ) ]
  [ PMSGDLV( ASPARENT | ALL | ALLDUR | ALLAVAIL ) ]
  [ PROXYSUB( FIRSTUSE | FORCE ) ]
  [ PUB( ASPARENT | ENABLED | DISABLED ) ]
  [ PUBSCOPE( ALL | ASPARENT | QMGR ) ]
  [ SUB( ASPARENT | ENABLED | DISABLED ) ]
  [ SUBSCOPE( ALL | ASPARENT | QMGR ) ]
  [ TYPE( LOCAL ) ]                    [ USEDLQ( ASPARENT | NO | YES ) ]
  [ WILDCARD( BLOCK | PASSTHRU ) ]
```


TOPIC admin example

dis topic(*)

```
AMQ8633: Display topic details.
  TOPIC(RedWine)                  TYPE(LOCAL)
AMQ8633: Display topic details.
  TOPIC(SYSTEM.BASE.TOPIC)        TYPE(LOCAL)
  :
AMQ8633: Display topic details.
  TOPIC(SYSTEM.DEFAULT.TOPIC)     TYPE(LOCAL)
AMQ8633: Display topic details.
  TOPIC(SYSTEM.FTE)               TYPE(LOCAL)
AMQ8633: Display topic details.
  TOPIC(WhiteWine)                TYPE(LOCAL)
dis topic('WhiteWine') all
    5 : dis topic('WhiteWine') all
AMQ8633: Display topic details.
  TOPIC(WhiteWine)          TYPE(LOCAL)          TOPICSTR(/Wine/White)    DESCR( )
  CLUSTER( )                DURSUB(ASPARENT)     PUB(ASPARENT)            SUB(ASPARENT)
  DEFPSIST(ASPARENT)        DEFPRTY(ASPARENT)    DEFPRESP(ASPARENT)       ALTDATE(2012-10-27)
  ALTTIME(20.59.13)         PMSGDLV(ASPARENT)    NPMSGDLV(ASPARENT)       PUBSCOPE(ASPARENT)
  SUBSCOPE(ASPARENT)        PROXYSUB(FIRSTUSE)   WILDCARD(PASSTHRU)       MDURMDL( )
  MNDURMDL( )               MCAST(ASPARENT)      COMMINFO( )              USEDLQ(ASPARENT)
  CUSTOM( )
```

Topic Object Parameters

Several options are controlled (eg. whether messages can be published or subscribe to this topic) individually or "as their parent". Options are:

- **ASPARENT** - Whether messages can be published/subscribe to the topic is based on the setting of the closest parent administrative topic object in the topic tree. This is the default supplied with WebSphere MQ, but your installation may change it.
- **ENABLED** Messages can be published/subscribed to the topic (by suitably authorized applications).
- **DISABLED** Messages cannot be published/subscribed to the topic.

Topic Status – current usage of topics

- **DISPLAY TPSTATUS(*topic-string*)**
  - **Used to display the status of one or more topic nodes in a topic tree.**
  - Topic attributes (TYPE(TOPIC)) (the default)
    - Resolved topic ASPARENT attributes
    - Number of publishers and subscribers
    - Whether a retained publication exists
  - Subscriber info (TYPE(SUB)):
    - Which subscription IDs are currently subscribed to this topic with associated connection ID
    - Last resume date of subscription and last message delivery time
    - Number of messages sent to this subscriber
  - Publisher info (TYPE(PUB)):
    - Which connection IDs are currently publishing to this topic
    - Last publish time
    - Number of messages published

Topic Status example

- **DISPLAY TPSTATUS – TYPE(TOPIC)**

```
:dis tpstatus('/Wine/Red/#')

AMQ8754: Display topic status details.
   TOPICSTR(/Wine/Red)                ADMIN(RedWine)    CLUSTER( )          MDURMDL(SYSTEM.DURABLE.MODEL.QUEUE)
   MNDURMDL(SYSTEM.NDURABLE.MODEL.QUEUE)  DEFPSIST(NO)    DEFPRTY(0)                DEFPRESP(SYNC)
   DURSUB(YES)                        PUB(ENABLED)    SUB(ENABLED)                PMSGDLV(ALLDUR)
   NPMSGDLV(ALLAVAIL)                 RETAINED(NO)    PUBCOUNT(0)                 SUBCOUNT(1)
   PUBSCOPE(ALL)                      SUBSCOPE(ALL)   USEDLQ(YES)

AMQ8754: Display topic status details.
   TOPICSTR(/Wine/Red/Carbon d'Artigues)   ADMIN(CarbonDArtigues)    CLUSTER( )    MDURMDL(SYSTEM.DURABLE.MODEL.QUEUE)
   MNDURMDL(SYSTEM.NDURABLE.MODEL.QUEUE)   DEFPSIST(NO)    DEFPRTY(0)                DEFPRESP(SYNC)
   DURSUB(YES)                        PUB(ENABLED)    SUB(ENABLED)                PMSGDLV(ALLDUR)
   NPMSGDLV(ALLAVAIL)                 RETAINED(NO)    PUBCOUNT(0)                 SUBCOUNT(1)
   PUBSCOPE(ALL)                      SUBSCOPE(ALL)   USEDLQ(YES)
```

- **DISPLAY TPSTATUS – TYPE(SUB)**

```
dis tpstatus('/Wine/Red/#') type(sub)

   28 : dis tpstatus('/Wine/Red/#') type(sub)
AMQ8754: Display topic status details.
   TOPICSTR(/Wine/Red)  SUBID(414D5120514D5F6361726C3531302020AD2E8C502000210F)
   SUBUSER(MUSR_MQADMIN)              RESMDATE( )    RESMTIME( )                 LMSGDATE( )
   LMSGTIME( )    ACTCONN(000000000000000000000000000000000000000000000000)
   DURABLE(YES)                       SUBTYPE(ADMIN)    MCASTREL( , )             NUMMSGS(0)

AMQ8754: Display topic status details.
   TOPICSTR(/Wine/Red/Carbon d'Artigues)    SUBID(414D5120514D5F6361726C353130202028ABB0502000CE04)
   SUBUSER(cfarkas)                   RESMDATE(2012-11-24)    RESMTIME(13:18:25)          LMSGDATE( )
   LMSGTIME( )    ACTCONN(414D5143514D5F6361726C353130202028ABB0502000CE01)
   DURABLE(NO)                        SUBTYPE(API)    MCASTREL( , )             NUMMSGS(0)
```


Subscription Status – current subscriptions

- DISPLAY SUB(topic-object-name)

```
 dis sub('Sub*') all
AMQ8096: WebSphere MQ subscription inquired.
   SUBID(414D5120514D5F6361726C353130202028ABB05020035C5C)
   SUB(SubCarbon)     TOPICSTR(/Wine/Red/Carbon d'Artigues)
   TOPICOBJ(CarbonDArtigues)          DEST(CarbonUpdates)    DESTQMGR( )               PUBAPPID( )
   SELECTOR( )                        SELTYPE(NONE)    USERDATA( )
   PUBACCT(1601051500000000EF2C5BD66544E29F111125BEE03000000000000000000000B)
   DESTCORL(414D5120514D5F6361726C353130202028ABB05020035C5C)
   DESTCLAS(PROVIDED)                 DURABLE(YES)    EXPIRY(UNLIMITED)           PSPROP(MSGPROP)
   PUBPRTY(ASPUB)                     REQONLY(NO)    SUBSCOPE(ALL)                SUBLEVEL(1)
   SUBTYPE(ADMIN)                     VARUSER(ANY)    WSCHEMA(TOPIC)              SUBUSER(MUSR_MQADMIN)
   CRDATE(2012-11-24)                 CRTIME(17:02:20)    ALTDATE(2012-11-24)     ALTTIME(17:02:20)
AMQ8096: WebSphere MQ subscription inquired.
```

- DISPLAY SBSTATUS(*topic-object-name*)

```
 41 : dis sbstatus('SubCarbon') all
AMQ8099: WebSphere MQ subscription status inquired.
   SUB(SubCarbon)    SUBID(414D5120514D5F6361726C353130202028ABB05020035C5C)
   SUBUSER(MUSR_MQADMIN)              RESMDATE(2012-11-24)
   RESMTIME(17:02:20)                 LMSGDATE(2012-11-24)    LMSGTIME(17:14:37)
   ACTCONN(000000000000000000000000000000000000000000000000)
   DURABLE(YES)                       MCASTREL( , )    NUMMSGS(1)                 SUBTYPE(ADMIN)
```

Retained Publications

- A retained publication is a message that provides a base level of information for a new subscriber.
- Each node on the topic tree may have one (1) and only one retained publication.
- The retained publication can be requested at any time.
- A Retained Publication is determined at publish time by the application program.
- A subscriber can choose to get retained publications every time they are published
- In addition a subscriber can request, at anytime, the current retained publication at a node (eg. MQSUBREQ).
  - If the subscription contains wildcards many retained publications from many nodes may be sent.

Retained Publications

- Example Stock Price - CurrentPrice is a retained publication
  - A subscriber can be notified whenever it changes – ticker tape
  - Or can request it at any time

Pub/Sub Security – Why it works the way it does

- To those of us on z/OS pub/sub security can seem odd
  - Pub/Sub Security looks for 'yes'
    - Looking for both a deny/allow is very complex
    - Wildcard subscriptions provide a bypass, so searching to the top is necessary
    - Security model is the same on all platforms

Security - Generics

- When you define topics, it might be a good idea to remember that MQ can use generic profiles.
- So instead of defining profiles like:
  - SPORT_FOOTBALL_RESULTS
  - SPORT_FOOTBALL_PLAYERS_HURSLEY
- It might be a good idea to do it like this:
  - SPORT.FOOTBALL.RESULTS
  - SPORT.FOOTBALL.PLAYERS.HURSLEY
- This allows you to grant access like this:
  - setmqaut –n SPORT.FOOTBALL.* . . .

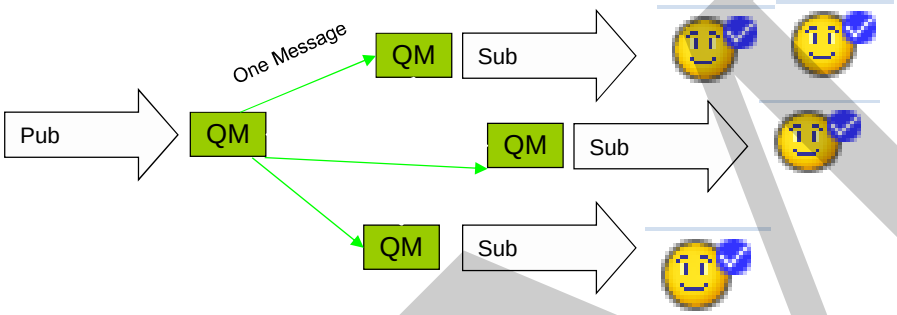
When is security checked?

- Security in the API
  - When publishing, a single check is performed at the MQOPEN of the topic.
  - When subscribing, checks are performed as follows:
    - Subscriber is allowed to subscribe to the topic objects
      - This check 'walks' the tree to find a yes as described earlier
    - Subscriber must also have authority to put to the destination for the subscription – actual 'publish' check.
      - Managed - Access permitted to the system generated temp queue
      - Non_Managed – Access permitted to the permanent queue

Pub/Sub Topologies

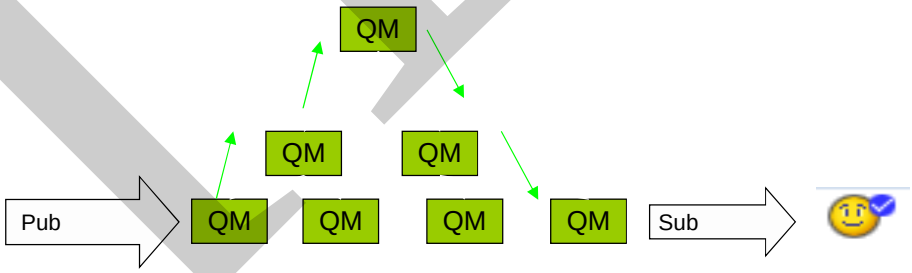
- We recognize three types of Pub/Sub Topologies.
- Clustered Pub/Sub.
  - When queue managers are in a cluster, it is possible that publications from any queue manager in the cluster may be subscribed to applications attached to another queue manager in the cluster.
  - In addition the multiple paths provided by the cluster topology can eliminate single points of failure.
- Hierarchical Pub/Sub
  - Queue managers are arranged in a Parent/Child Hierarchy.
  - Each Queue manager may have only one Parent.
  - Queue managers may exchange Pub/Sub information directly between only their parent or children.
  - This type of pub/sub often requires multiple hops
  - Each queue manager in the hierarchy is a single point of failure.
- Plus the composite

Pub/Sub in a cluster

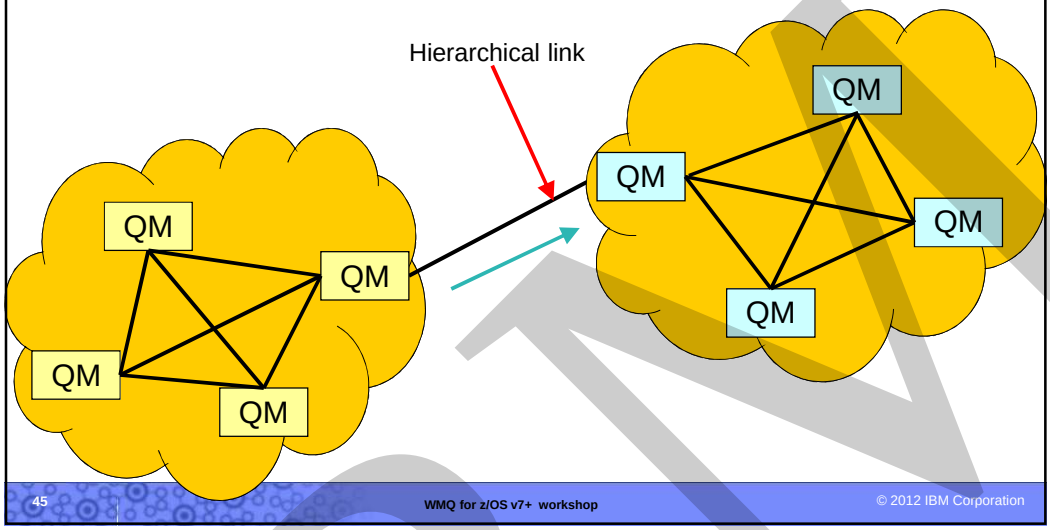


- Consistent definitions in cluster
- Multiple routes across cluster

Pub/Sub using Hierarchical Distributed Queue Managers

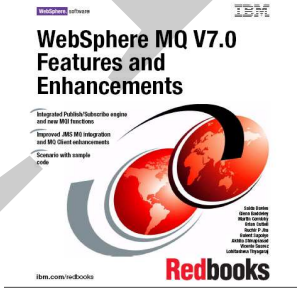


Publish/Subscribe in Combined Hierarchy & Clusters



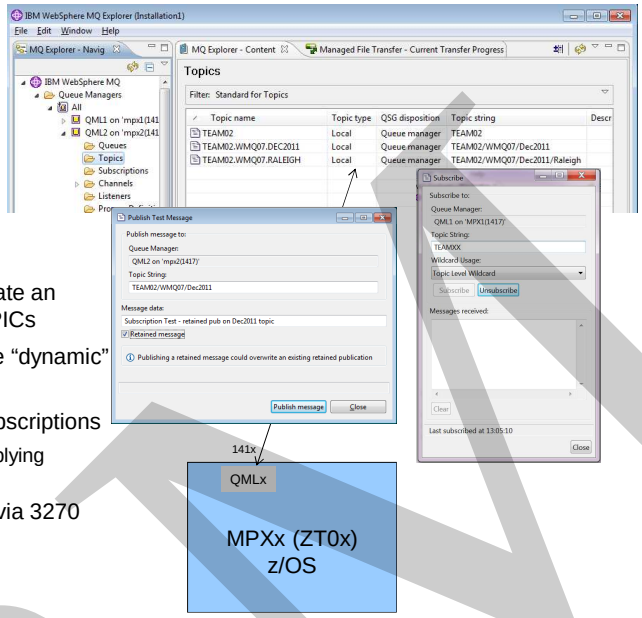
Bibliography

- SC34-6950 MQv7 Publication/Subscribe User's Guide
- SG24-7583 WebSphere MQ v7 Features and Enhancements



The complete MQ library is available in PDF at
http://www.ibm.com/software/integration/wmq/library/
Or online at
http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp

Lab time



- Using the MQ Explorer, you'll create an "administerable" hierarchy of TOPICs
- You'll test the Pub/Sub and create "dynamic" subscriptions
- You'll create some permanent subscriptions
  - You'll see a new "feature" for multiplying messages
- You'll take a glimpse at Pub/Sub via 3270

WebSphere MQ for z/OS v7+ Proof-Of-Technology (POT) workshop

What's new with MQ v7.1 z/OS? (and MQ v7.5 Distributed)

IBM EMEA panIMT Team zWebSphere team
(with thanks to IBM US ATS team for the original "Wildfire" material)

V7.1 - the big picture

WebSphere MQ v7.1 General availability
- Distributed platforms: 11 Nov 2011
- z/OS: 25 Nov 2011

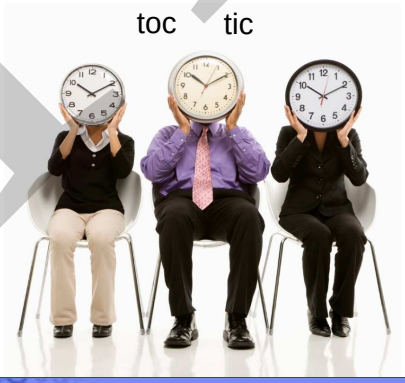
- For everyone: Out of the box MQ security
  - Channel Authentication
- For Distributed MQ: ease of migration
  - Multi-version support
- For z/OS: enhanced scalability and robustness
  - SMDS
  - Shared queue offload rules
  - Shared queue reconnection
- And, of course, loads of other nice goodies…

MQ v7.1 platform support

- **z/OS v1.11 or later**
  - For up-to-date details, see http://www.ibm.com/support/docview.wss?uid=swg27023056
- Linux Red Flag 5.0 DC Server x86-32
- Linux Red Hat Enterprise Linux (RHEL) 5 or 6 POWER, System z, x86-32, x86-64
- SUSE Linux Enterprise Server (SLES) 10 SP1 POWER, SP1 System z,SP3 x86-32, SP3 x86-64 or SLES 11 POWER, System z, x86-32, x86-64
- AIX 6.1 or 7.1HP-UX Itanium or PA-RISC 11i v2 or v3
- HP-UX 11i v3
- Sun Solaris 10 SPARC or x86-64
- IBM i v6.1 or 7.1 POWER System
- Windows 7 Enterprise, Professional, Ultimate, Server 2003 R2 Enterprise Edition, Server 2003 R2 Standard Edition, Server 2003 SP1 Enterprise Edition, Server 2003 SP1 Standard Edition, Server 2008 Enterprise Edition, Vista Business, Vista Enterprise, Vista Ultimate and XP Professional on x86-32 or x86-64
- For up-to-date details on Distributed MQ platforms, see http://www.ibm.com/support/docview.wss?uid=swg27023058

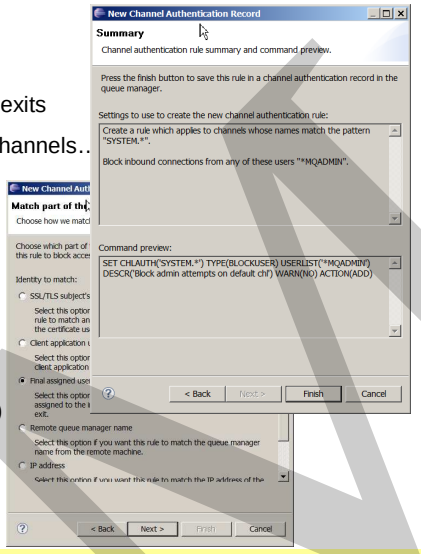
MQ v6.0 – Replaced by MQv7, v7.1, v7.5….

- IBM announced that WebSphere MQ v6.0 (eg. Windows, Unix, Linux, z/OS will arrive at end of support on 30 September ~~2011~~ ***2012*** ***– see ENUS211-072***
- Avoid the holiday rush… start thinking about migrating to MQv7 now!
- For more details, see http://www.ibm.com/software/websphere/support/lifecycle/



Security: Channel Access Control

- Simplifying configuration for channel access
  - MQ Clients and Queue Managers
- Significantly reduce the need for custom channel exits
- New CHLAUTH definitions control who can use channels…
  1. Name mapping
  2. Access blocking
- Rules are based on
  - Partner IP address
  - Partner Queue Manager name
  - SSL Distinguished Name
  - Asserted identity (including *MQADMIN option)
  - Derived identity from DN mapping
- MQ Explorer configuration via wizard



You'll see a lot more Channel Authentication details in another presentation….

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 can declare authorisations for non-local queues
  - Including clustered queues
  - Now consistent with z/OS
  - Also consistent with Topic authorisations
- 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
- Also works for applications which explicitly open queue@qmgr
  - Common pattern when using ReplyTo information

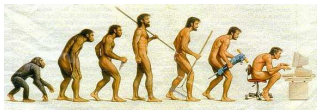
```
setmqaut –m QM1 –t queue –n SYSTEM.CLUSTER.TRANSMIT.QUEUE -p mquser +put
```

Becomes….

```
setmqaut –m QM1 –t rqmname –n QM2 –p mquser +put
```


Multi-Version Installation on MQ Distributed

- MQ on Unix and Windows can install multiple levels on a system (which has always been supported on z/OS! ☺)
  - 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

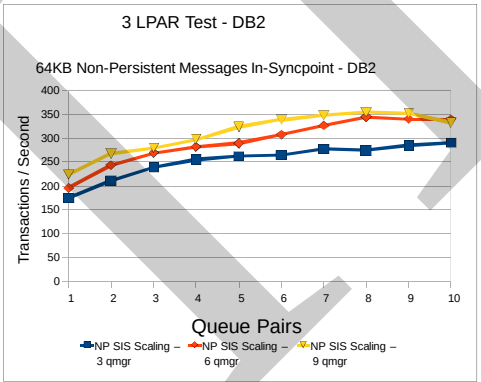
You’ll see more on Multi-version support in another presentation….

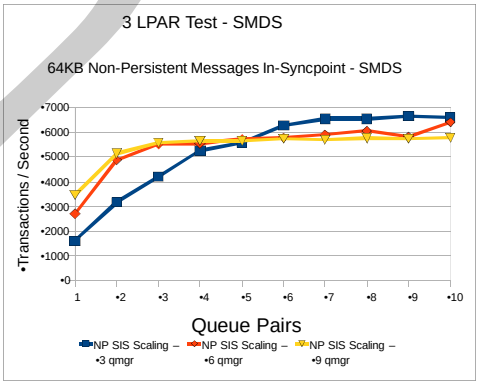
Large Shared Queue Messages

- **Shared Message DataSets (SMDS) *alternative* to DB2 for large message storage**
- Using DB2 BLOBs to store large (>63KB) messages is expensive
  - Both CPU and pathlength
  - DB2 still needed for storing shared definitions
  - Coupling Facility 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 now control offload message size and % Structure-full offload trigger**
  - Set per CF structure,
    - eg. DEFINE CFSTRUCT(MYSTRUCT) … OFFLD1TH(65) OFFLD1SZ(32K) means "For MYSTRUCT, when it gets over 65% full, offload all messages over 32K"
- All QSG members must be at new level to access queues with this capability
- The rules can be set up for both SMDS and DB2 offloading

You'll see a lot more Shared Queue details in another presentation….

SMDS Performance Improvement

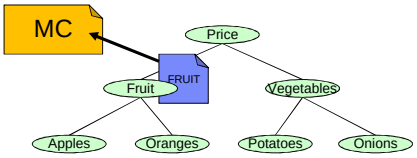




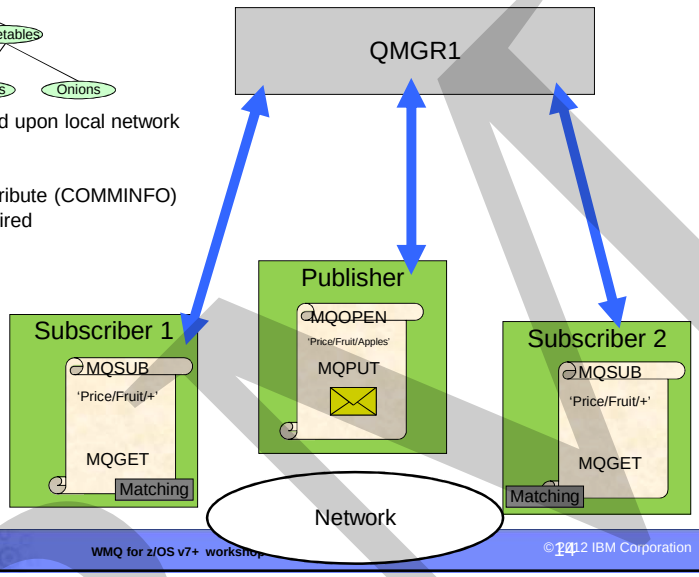
- Early Test Results on z196
- Tests show comparable CPU savings making SMDS a more usable feature for managing your CF storage
- SMDS per CF structure provides better scaling than DB2 BLOB storage

Transactions/second are an order of magnitude better with SMDS!

Multicast Publish/Subscribe (for MQ Clients)



- Super high-speed pub/sub based upon local network multicast
  - Uses LLM technology
- Topics defined with multicast attribute (COMMINFO)
- Normal MQ connection still required
- ***Messages flow directly between clients***
- Subscriber matching done at each client
- MQ queue manager can also publish out to Multicast subscribers
- Statistics sent to QM
  - Distributed MQ only
- Requires MQ Client
- Classic compiled programs only today; no Java or .Net yet



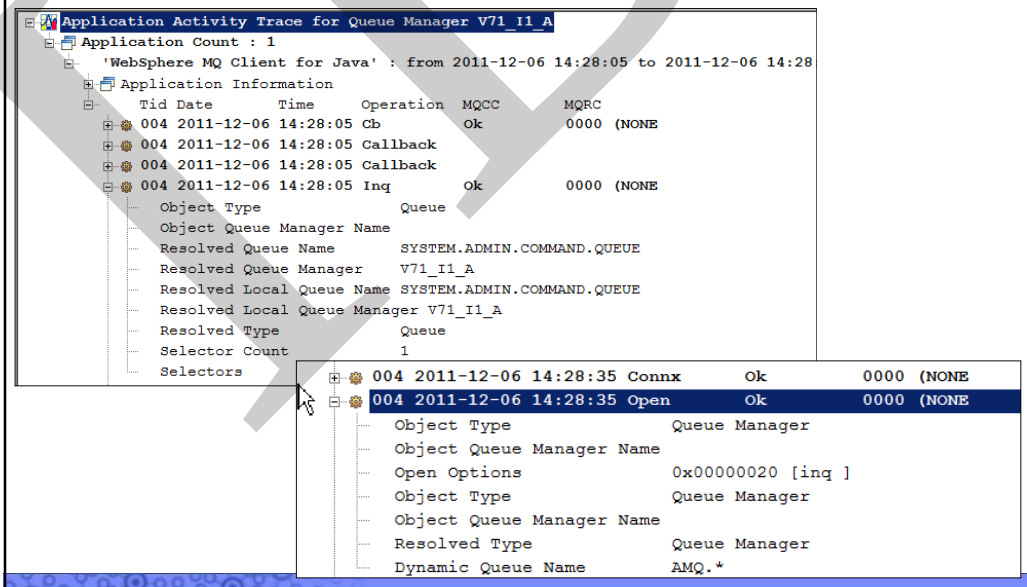
Application Activity Reports (MQ Distributed only)

- 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
```


SupportPac MS0P V7.1

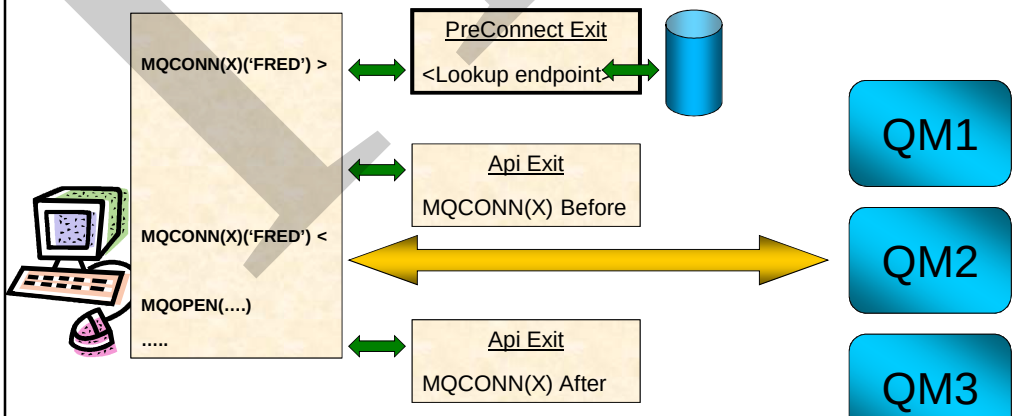


MQ Cloud Support: Pre-Connect Exit

- Supports movement by some to “Utility Compute”, Private Cloud configs, etc.
  - Rapid provision of applications allied with need to further decouple Client/Server connectivity
  - Server applications might move location – new addresses or queue managers
- MQ Client connects to a “service” rather than specific Queue Manager
- Can transparently change location of MQ server-side applications
  - No client code changes needed
  - No configuration files need to be updated at the client machine
  - JMS/XMS applications already do this via JNDI lookup
- Exit run during MQCONN queries a repository to discover real location
  - MQ V7.1 incorporates the LDAP implementation from SupportPac MA98

PreConnect Exits

- Actually shipped as part of MQ V7.0.1 fixpack 4
- Ability to dynamically choose which server to connect to
- Only invoked in the client leg
- Same interface regardless of MQCONN or MQCONNX
- V7.1 adds the LDAP exit implementation that was previously a SupportPac



RPRODUCT & RVERSION – Channel Status

- Channel status shows product and version of remote connection

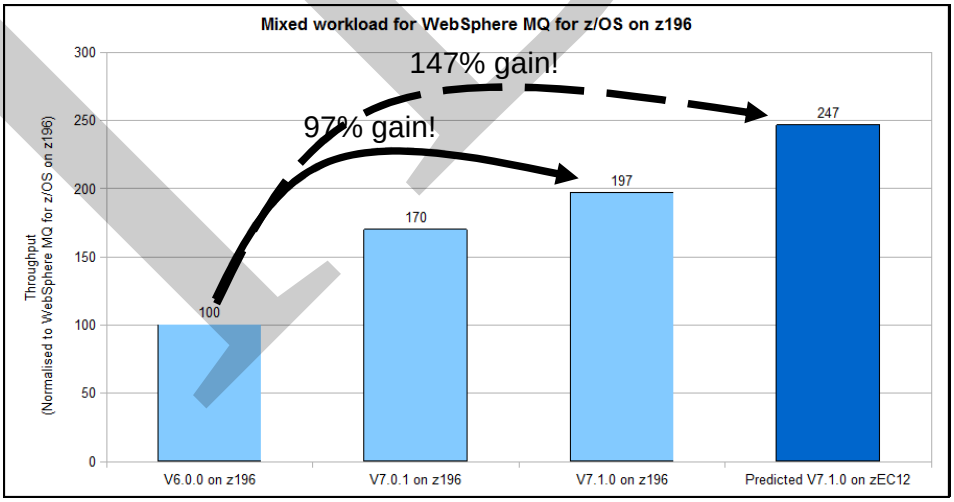
```
STC23876  CSQM201I  QML1 CSQMDRTC  DIS CHSTATUS DETAILS  906
          CHSTATUS(TO.QML4)
          CHLDISP(PRIVATE)
          XMITQ(QML4)
          CONNAME(192.168.17.253)
          CURRENT
          CHLTYPE(SDR)
          STATUS(RUNNING)
          SUBSTATE(MQGET)
          STOPREQ(NO)
          RQMNAME(QML4)
          RPRODUCT(MQMV)
          RVERSION(07010000)
           END CHSTATUS DETAILS
```

- Applies to both MCA channels and SVRCONNs
- Remote system must be MQ V7.0 or higher – otherwise blank is shown

MQMM	Queue Manager Distributed
MQMV	Queue Manager z/OS
MQCC	MQ C Client (including XA Client)
MQNM	MQ .NET managed client
MQJB	Java Base Client
MQJM	Java JMS V7-leg
MQJN	Java JMS V6-leg

MQJU	JMQI
MQJW	JMQI used by WAS
MQXC	XMS Client C/C++ V7-leg
MQXD	XMS Client C/C++ V6-leg
MQXN	XMS Client .NET V7-leg
MQXM	XMS Client .NET V6-leg

WebSphere MQ for z/OS – Version to version performance



Measurements based on SupportPac MP1H run on dedicated WebSphere MQ performance sysplex (2817-779) running z/OS v1r13. Predicted performance based on LSPR tables.

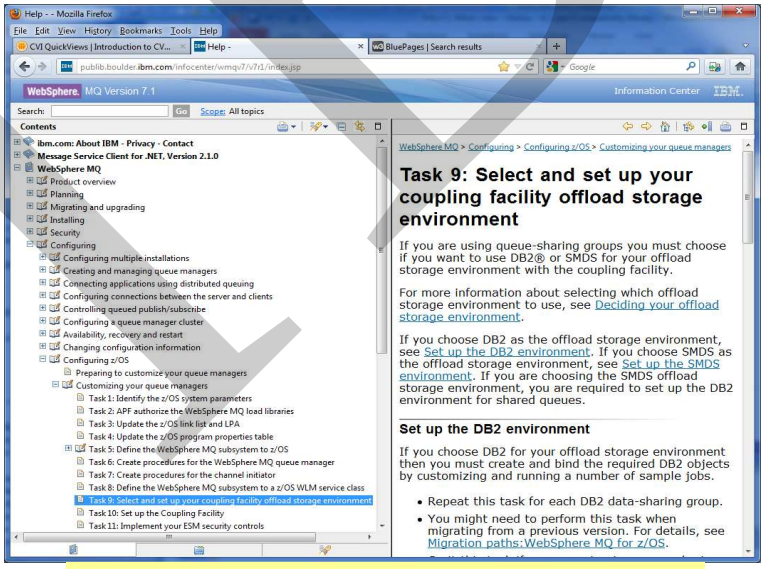
Scalability & Performance & Function – MQ Explorer

- 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

- Some nice new features…
  - Multiple object delete
  - Suppress those annoying confirmation messages

MQ InfoCenter - restructured



http://publib.boulder.ibm.com/infocenter/wmqv7/v7r1/index.jsp

Other MQ v7.1 goodies…

- New “Bind on group” for MQ Cluster
  - All messages within a logical group are routed to the same queue manager
  - Workload balancing is done for each group
- MQ Client enhancements
  - A client is now available on System i
  - API Exits available in C clients
    - Same interface as available for local binding applications
- Dead-letter queue setting *per channel* or *per topic*
- MQ Hypervisor version available – pre-installed for clouds
- New *dmpmqcfg* command (Distributed MQ) to extract MQ configuration (replaces *saveqmgr* , the wonderful MS03 SupportPac)
- New *RUNMQSC SET AUTHREC* for equivalent *setmqaut* commands
- Etc

WMQ for z/OS Installation Information

- **Standard SMP/E Installation**
  - Program Directory for WMQ for z/OS V7.1 - GI13-0572-00
  - System Requirements for WMQ V7.1 for z/OS
    - http://www-01.ibm.com/support/docview.wss?uid=swg27023056#z_OS_System_z_hw
  - FMID HMS7100 – z/OS Base
    - FMI Subsets:
      - **JMS7101 - U.S. English (ENU)**
      - JMS7102 - Japanese (JPN)
      - JMS7103 - Simplified Chinese (CHS)
      - JMS7104 - Upper Case English (ENP)
      - JMS7105 - French (FRA)
      - **JMS7107 - Client Attachment**
      - **JMS7108 - Unix System Services Components**

WMQ for z/OS Installation Information

- **Software Requirements:**
  - Conditional Requirements (continued):
    - Cryptographic Services:
      - System SSL (FMID HCPT380) for SSL channels
      - Security Level 3 (FMID JCPT381) for some CipherSpecs
    - Shared Queues
      - DB2 at the at the minimum level
      - Coupling Facility (CF) Level 9
      - Resource Recovery Services (RRS) active
    - Java Runtime
      - IBM 31-bit SDK for z/OS, Java(TM) Technology Edition, version 5 or later
      - IBM 64-bit SDK for z/OS, Java(TM) Technology Edition, version 6 or later
    - RACF:
      - RACF for z/OS Version 1.8 with APAR OA23043 or Version 1.9 or higher
      - RACF mixed case security profiles and TOPIC security

Changed SCSQPROC members

- **CSQ4INSG**
  - New parameters added for channel models
    - BATCHLIM
    - USEDLQ
  - New Queue added
    - SYSTEM.ADMIN.PUBSUB.EVENT
  - New Model for JMS
    - SYSTEM.JMS.TEMPQ.MODEL
  - New Topic parameters
    - NPMSGDLV (new with 7.0.1)
- **CSQ4MSTR**
  - New members in the concatenation
    - CSQ4INSA – default rules for channel authorization
    - CSQ4INST – sample subscriptions (new with 7.0.1)
    - CSQ4INYT – sample commands to follow pub/sub initialization
- **CSQ4SMFJ**
  - New output file QESD
- **CSQ4S100**
  - Changed CICS sample definition parameters

Changed SCSQPROC members

- **CSQ4ZPRM**
  - IDBACK – removed
  - IDFORE – removed
- **CSQ45BPL**
  - DB2 package names changed
- **CSQ45CTB**
  - New columns added
- **CSQ45CTS**
  - New note on SEGSIZE for older versions of DB2
- **CSQ45GEX**
  - New names for the DB2 GRANTs
- **CSQ7IPCS**
  - New dump exit program name for V7.1

New and removed SCSQPROC members

- **New Members:**
  - CSQ4INSA – Default rules for channel authorization
  - CSQ4INST – Sample subscriptions (new with 7.0.1)
  - CSQ4INYT – Sample commands to follow pub/sub initialization
  - CSQ4SMDS – Shared Messages Data Set definition and format
  - CSQ4UZPR – USERMOD control
  - CSQ455TB – Migration for table definitions from WMQ 5.3.1 to WMQ 7.1
  - CSQ456TB – Migration for table definitions from WMQ 6 to WMQ 7.1
  - CSQ457TB – Migration for table definitions from WMQ 7.0.1 to WMQ 7.1
- **Removed Members:**
  - CSQ45ATB – Migration for table definition from WMQ V6 to WMQ V7.0
  - CSQ45MTB – Migration for table definitions from WMQ 5.3.1 to WMQ 7.0

MQv7.1 gotchas

- CSQ6SYSP in ZPRM no longer uses the parameters CTHREAD, IDFORE, IDBACK (ignored)
- New suggested default names MQTR.DEAD.QUEUE, MQTR.DEFXMIT.QUEUE in CSQ4INYG
- A few WMQ.V7R1M0.SCSQPROC changes…
  - CSQ4INSA for defining system objects and default rules for Channel Authentications
  - CSQINPT added to run CSQ4INST for initialisation after pub/sub has started
  - CSQOUTT output added for pub/sub
- For additional notes, see http://www.ibm.com/support/docview.wss?uid=swg21596103

To make customization a bit easier!

- **You can use a sample REXX to make customization simpler**
  - Includes most of the '++' variables used in the sample JCL to create a queue manager and channel initiator

```
 File  Edit  Edit_Settings  Menu  Utilities  Compilers  Test  Help
------------------------------------------------------------------------------
VIEW       WMQ710.QML#.SCSQPROC(QML#EDIT) - 01.00          Columns 00001 00072
Command ===>                                                 Scroll ===> PAGE
****** ***************************** Top of Data ******************************
000001 ISREDIT MACRO NOPROCESS
000002 ADDRESS ISREDIT
000003 "change '++THLQUAL++' 'WMQ710' all"
000004 "change '++HLQ++' 'SYS1.MQM710' all"
000005 "change '++LANGLETTER++' 'E' all"
000006 /* **** BSDS CHANGES ****  */
000007 "change '++VOLBSDS1++' 'Q70001' all"
000008 "change '++VOLLOG1A++' 'Q70002' all"
000009 "change '++VOLLOG1B++' 'Q70003' all"
000010 /* **** PAGE CHANGES ****  */
000011 "change '++VOL0++' 'Q70001' all"
000012 "change '++VOL1++' 'Q70001' all"
000013 "change '++VOL2++' 'Q70002' all"
000014 "change '++VOL3++' 'Q70003' all"
000015 "change '++VOL4++' 'Q70002' all"
000016 /* **** ZPRM CHANGES ****  */
000017 "change '++HLQ.USERAUTH++' 'WMQ710.++QML#++.AUTHLIB' all"
000018 "change '++NAME++' 'QML#ZPRM' all"
000019 /* **** MSTR CHANGES ****  */
```

  - See our optional Customization exercise if you have time…..

In summary… MQv7.1 brings significant new value to the customer

- Major **security enhancements** in MQ 7.1
  - Typically eliminate the need for supporting custom channel exits
- Major improvements to **Shared Queue support** on z/OS
  - SMDS, offload threshold rules, more robust auto-reconnection
- **Multiple MQ version** support for all platforms
  - Simplify migrations

V7.5 - the small picture

WebSphere MQ v7.5 General availability

- Distributed platforms: 20 June 2012 (note: System I not included)
- z/OS: Not yet

- Simplify deployment for those using MQ "stack" products
  - (Optional) Integration of FTE, AMS and MQTT
- Slip in some goodies…
  - Multiple cluster transmission queues ("split transmit queues")
  - Extended transactional functionality incorporated into MQ Client
  - Etc.

Consolidation of MQ, AMS and FTE capabilities

With MQ V7.5 we have updated the physical packaging for the existing MQ, AMS and FTE capabilities to make them easier to consume

Before

WebSphere MQ V7.1
- MQ Queue manager
- MQ Client
- MQ Documentation

AMS V7.0.1.2
- Advanced Msg Security
- AMS Documentation

File Transfer Ed
- FTE Serv
- FTE Client
- FTE Docs + tools

After

W .Next
- e manager
- ced Message Security
- anaged File Transfer Service *(formerly FTE Server)*
- + Managed File Transfer Logger *(formerly part of FTE tools)*
- + MQ Explorer comes with built-in AMS and managed file transfer support
- MQ Client
- + AMS Enablement
- Managed File Transfer Agent *(formerly the FTE Client)*
- + AMS Enablement
- MQ Documentation
- + AMS docs
- + Managed File Transfer docs

I didn't talk about the price!

AMS is integrated with the queue manager and client. FTE is renamed to "MQ Managed File Transfer" and offers capability which is more tightly integrated with the MQ Queue Manager

Clustering – Split Transmit Queue

- With V7.5 a queue manager can automatically define a PERMANENT-DYNAMIC queue for each CLUSSDR channel.
  - Dynamic queues based upon new model queue "SYSTEM.CLUSTER.TRANSMIT.MODEL"
  - Well known queue names: "SYSTEM.CLUSTER.TRANSMIT.<CHANNEL-NAME>"
- Controlled via attribute affecting all cluster-sdr channels on the queue manager

```
ALTER QMGR DEFCLXQ( SCTQ | CHANNEL )
```

- Also have manual definitions
  - Multiple queues can be defined to cover all, or a subset of the cluster channels.

```
DEFINE QLOCAL(APPQMGR.CLUSTER1.XMITQ)
CHLNAME(CLUSTER1.TO.APPQMGR) USAGE(XMITQ)
```

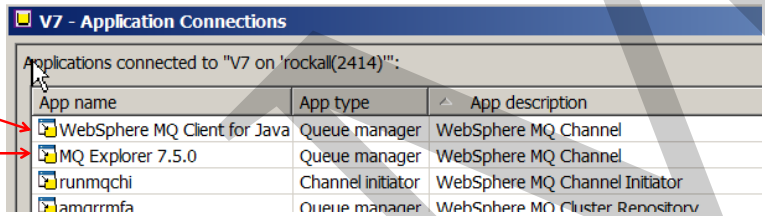
- Automatic and Manual are not mutually exclusive
  - They could be used together

Java Application Identification

- Java client applications now fill in APPLTAG field
- No longer appear as “WebSphere MQ Client for Java”
- Application-provided property
- Or the Main class

V7 Explorer

V7.5 Explorer



MQ Bibliography

- GI13-0572 MQ v7.1 z/OS Program Directory
- SG24-8087 WebSphere MQ V7.1 and V7.5 Features and Enhancements (draft)

The complete MQ library is available in PDF at
http://www.ibm.com/software/integration/wmq/library/
Or online at
http://publib.boulder.ibm.com/infocenter/wmqv7/v7r1/index.jsp

WebSphere MQ for z/OS v7+ Proof-Of-Technology (POT) workshop

Channel Authentication

IBM EMEA panIMT Team zWebSphere team
(with thanks to IBM US ATS team for the original “Wildfire” material)

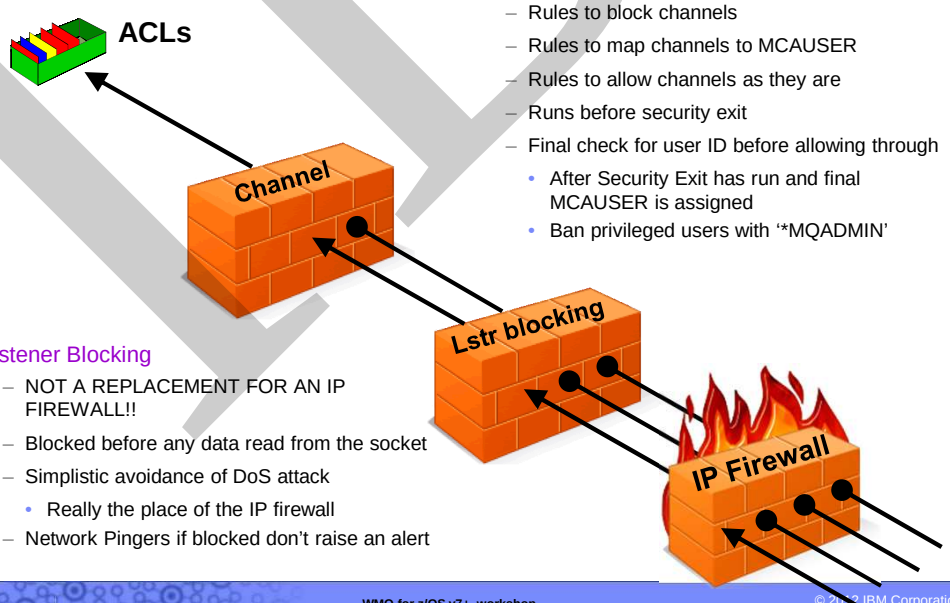
Agenda

- CHLAUTH
  - Configuration
  - IP Blocking
  - Address mapping
  - examples

Channel Authentication Records

- Set rules to control how inbound connections are treated
  - Inbound Clients
  - Inbound QMgr to QMgr channels
  - Other rogue connections causing FDCs
- Rules can be set to
  - Allow a connection
  - Allow a connection and assign an MCAUSER
  - Block a connection
  - Ban privileged access (“*MQADMIN”)
  - Provide multiple positive or negative SSL Peer Name matching
- Rules can use any of the following identifying characteristics of the inbound connection
  - IP Address
  - SSL/TLS Subject’s Distinguished Name
  - Client asserted user ID
  - Remote queue manager name

Channel Access Blocking Points



Channel Blocking/Mapping
- Rules to block channels
- Rules to map channels to MCAUSER
- Rules to allow channels as they are
- Runs before security exit
- Final check for user ID before allowing through
  - After Security Exit has run and final MCAUSER is assigned
  - Ban privileged users with ‘*MQADMIN’

Listener Blocking
- NOT A REPLACEMENT FOR AN IP FIREWALL!!
- Blocked before any data read from the socket
- Simplistic avoidance of DoS attack
  - Really the place of the IP firewall
- Network Pingers if blocked don’t raise an alert

Block Point: MQ Listener

- List of IP address patterns
- One single list

SET CHLAUTH(*) TYPE(BLOCKADDR) ADDRLIST('9.20.*', '192.168.2.10')

- NOT A REPLACEMENT FOR AN IP FIREWALL
  - Temporary blocking
  - Blocking until IP firewall updated
  - Shouldn't be many entries in the list
- Blocked before any data read from the socket
  - i.e. before SSL Handshake
  - Before channel name is known
- Avoiding DoS attack
  - Really the place of the IP firewall
  - Simplistic 'hold' of inbound connection to avoid reconnect busy loop
- Network Pingers if blocked don't raise an alert
  - Immediate close of socket with no data not considered a threat

Mapping Point: MQ Channel

- Basic model is to map an attribute to MCAUSER
  - In conjunction with the channel name
- Identifying attributes are
  - SSL Peer Name pattern (most specific)
    - Precedence defined for partial patterns
  - Remote queue manager name pattern (MCA channels)
  - Client asserted user ID (MQI channels)
    - No pattern matching on this
  - IP address pattern
- Mapping done before calling security exit
- Defines where user ID (MCAUSER) comes from
  - Provided on command
  - Flowed or defined on channel as today
    - Combine this with BLOCKUSER list
  - Blocked

Order	DN Substring	Name
1	CN=	Common name
2	T=	Title
3	OU=	Organizational unit
4	O=	Organization
5	L=	Locality
6	ST=, SP=, S=	State or province name
7	C=	Country

Order	Identity mechanism	Notes
0	Channel Name	
1	SSL Distinguished Name	
2=	Client asserted User ID	Clearly several different user IDs can be running on the same IP address.
2=	Queue Manager Name	Clearly several different queue managers can be running on the same IP address
4	IP address	

Channel Authentication Records – Configuration

- Create rules using
  - MQSC: SET CHLAUTH
  - PCF
  - MQ Explorer GUI Wizard
- Precedence matching
  - Most specific rule is matched
  - Within SSL Peer Name matching
    - Most specific substring is matched
- Pattern matching
  - Channel Name: Beginning, middle, end
  - IP addresses (IPV4 or IPV6)
    - '*' in any segment
    - '-' ranges in any segment
  - SSL Peer Name
  - QMgr Name: Beginning, middle, end

Order	Identity mechanism
0	Channel Name
1	SSL Distinguished Name
2=	Client asserted User ID
2=	Queue Manager Name
4	IP address

Restricting the Mappings

- Restrict where an SSL Certificate can be used from
  - Specific IP address
- Restrict where a queue manager or client user ID can come from
  - Specific IP address

Mapped \ Restrict By	SSL Peer	QM Name	Client User	IP Address
SSL Peer		X	X	✓
QM Name				✓
Client User				✓
IP Address				

```
SET CHLAUTH(*) TYPE(SSLPEERMAP)
SSLPEER('L="Hursley"') MCAUSER(HURUSER) ADDRESS('9.20.*')

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

SSL DN Precedence Mapping Example

```
SET CHLAUTH(*) TYPE(SSLPEERMAP) SSLPEER('OU="MQ Devt"') MCAUSER(MQUSER)

SET CHLAUTH(*) TYPE(SSLPEERMAP) SSLPEER('L="Hursley"') MCAUSER(HURUSER)
```

Order	DN Substring	Name
1	CN=	Common name
2	T=	Title
3	OU=	Organizational unit
4	O=	Organization
5	L=	Locality
6	ST=, SP=, S=	State or province name
7	C=	Country

Most Specific Match

CN=Morag Hughson.OU=MQ Devt. O=IBM UK.L=Hursley.C=UK

IP Address Pattern Matching

- Single Address
  - 9.20.4.6
- Wildcard at the end
  - 9.20.*
- Wildcard in the middle
  - 9.20.*.6
- Ranges
  - 9.20.4.1-10
- IPV4 or IPV6
  - 3ffe:1900:4545:3:200:f8ff:fe21:67cf
- IPV6 wildcarded
  - 3ffe:1900:4545:3:200:*
- IPV4 will also block IPV6 and vice versa
  - 0:0:0:0:0:ffff:192.1.56.10

How should I use this?

```
SET CHLAUTH(*) TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS) WARN(YES)

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"

What happens if…?

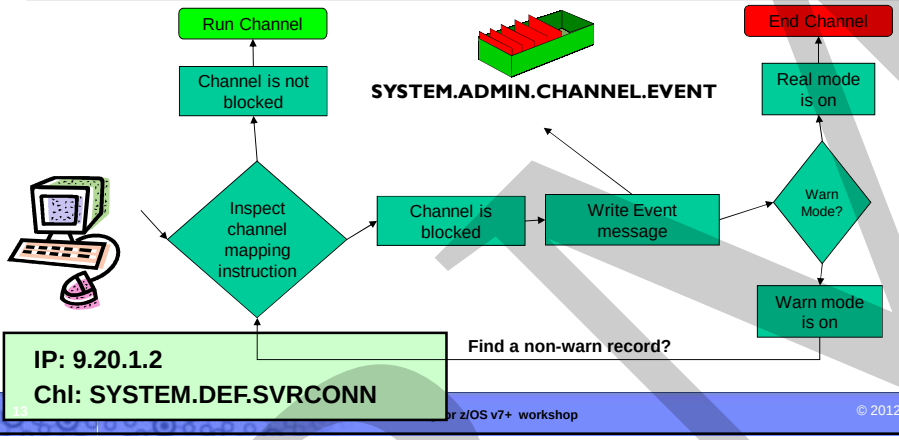
```
DISPLAY   CHLAUTH(SYSTEM.ADMIN.SVRCONN) MATCH(RUNCHECK)
SSLPEER('CN="Morag Hughson", O="IBM UK"')
          CLNTUSER('mhughson')
          ADDRESS('9.180.165.163')
returns ===>
          CHLAUTH(SYSTEM.ADMIN.SVRCONN)
          TYPE(ADDRESSMAP)
          ADDRESS('9.180.165.163') MCAUSER(MORAG)
```

Order	Identity mechanism
0	Channel Name
1	SSL Distinguished Name
2=	Client asserted User ID
2=	Queue Manager Name
4	IP address

Chl: SYSTEM.ADMIN.SVRCONN
DN: CN=Morag Hughson.O=IBM UK
UID: mhughson
IP: 9.180.165.163

Warning mode

```
ALTER QMGR CHLEV(ENABLED)

SET CHLAUTH(SYSTEM.*) TYPE(ADDRESSMAP) ADDRESS('9.20*') USERSRC(NOACCESS) WARN(YES)
SET CHLAUTH(SYSTEM.*) TYPE(ADDRESSMAP) ADDRESS('9.*') MCAUSER(HUGHSON)
SET CHLAUTH(SYSTEM.*) TYPE(ADDRESSMAP) ADDRESS(*) USERSRC(NOACCESS) WARN(NO)
```



Out of the Box

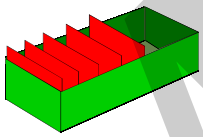
- We supply these rules out-of-the-box.
  - For all channels, ban the assertion of privileged users by inbound channels.
  - For all SYSTEM channels except SYSTEM.ADMIN.SVRCONN (the MQ Explorer GUI channel), ban anyone from using them.

```
SET CHLAUTH(*) TYPE(BLOCKUSER) USERLIST(*MQADMIN)

SET CHLAUTH(SYSTEM.*) TYPE(ADDRESSMAP)
ADDRESS(*) USERSRC(NOACCESS)

SET CHLAUTH(SYSTEM.ADMIN.SVRCONN) TYPE(ADDRESSMAP) ADDRESS(*)
USERSRC(CHANNEL)
```

  - Difficult to supply any default rules regarding IP addresses and SSL Peer Names since they are very installation specific.
- Enabling Switch ALTER QMGR CHLAUTH(ENABLED|DISABLED) different for Migrated or New Queue Manager

Events

- Command events (as normal) for SET CHLAUTH commands
- Configuration events (as normal) for SET CHLAUTH commands
- Channel events when channel is blocked due to a rule
- Controlled by existing switch
  - Considered to be an EXCEPTION
- Written to existing queue
- One event for each type of connection refusal
- MQRC_CHANNEL_BLOCKED
  - MQRQ_CHANNEL_BLOCKED_ADDRESS
  - MQRQ_CHANNEL_BLOCKED_USERID
  - MQRQ_CHANNEL_BLOCKED_NOACCESS



SYSTEM.ADMIN.CHANNEL.EVENT

ALTER QMGR CHLEV(ENABLED|EXCEPTION)

CHLAUTH – An example problem

- I have a clean queue manager with the following default rules: -

```
CSQM293I !MQ25 CSQMDRTC 3 CHLAUTH FOUND MATCHING REQUEST CRITERIA

CSQM201I !MQ25 CSQMDRTC  DIS CHLAUTH DETAILS
CHLAUTH(*)
TYPE(BLOCKUSER)
USERLIST(
 *MQADMIN
 )
 END CHLAUTH DETAILS

CSQM201I !MQ25 CSQMDRTC  DIS CHLAUTH DETAILS
CHLAUTH(SYSTEM.*)
TYPE(ADDRESSMAP)
USERSRC(NOACCESS)
 END CHLAUTH DETAILS

CSQM201I !MQ25 CSQMDRTC  DIS CHLAUTH DETAILS
CHLAUTH(SYSTEM.ADMIN.SVRCONN)
TYPE(ADDRESSMAP)
USERSRC(CHANNEL)
 END CHLAUTH DETAILS
CSQ9022I !MQ25 CSQMDRTC ' DIS CHLAUTH' NORMAL COMPLETION
```

- My queue manager runs under the user 'markw1' and my CHINIT runs under 'mq25usr'
- I have a simple put program on a client (with address 9.20.137.15) which runs under the user 'markw1'.

The actual problem

- When I run the program I see: -

```
+CSQX500I !MQ25 CSQXRESP Channel SYSTEM.DEF.SVRCONN started connection :ffff:9.20.237.15
+CSQX777I !MQ25 CSQXRESP Channel SYSTEM.DEF.SVRCONN from ::ffff:9.20.237.15 has been
blocked due to USERSRC(NOACCESS),
  Detail:
       CLNTUSER(markw1)
+CSQX501I !MQ25 CSQXRESP Channel SYSTEM.DEF.SVRCONN no longer active
 connection ::ffff:9.20.237.15
```


A solution

- Map channel SYSTEM.DEF.SVRCONN from address 9.20.237.15 and user markw1 to mcauser markw2.
- Use the command: -

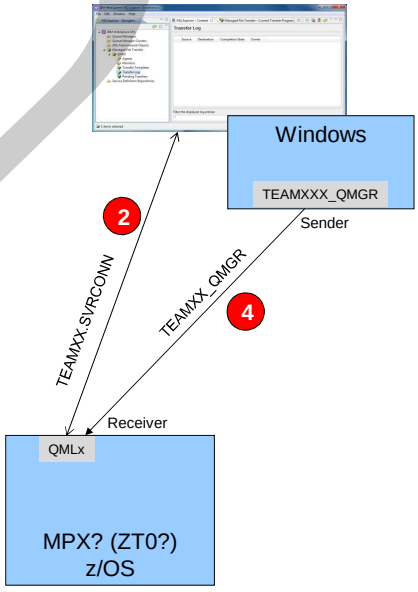
```
SET CHLAUTH(SYSTEM.DEF.SVRCONN)
    TYPE(USERMAP)
    ACTION(ADD)
    DESCR('Allow Mark in')
    ADDRESS('9.20.237.15')
    MCAUSER(MARKW2)
    CLNTUSER(MARKW1)
```

Bibliography

• For details on the Channel Authentication rules, see the MQ InfoCenter at
http://publib.boulder.ibm.com/infocenter/wmqv7/v7r1/topic/com.ibm.mq.doc/zs14190_.htm

Lab time

- Channel Authentication lab
  1. Enable Channel Authentication on your z/OS QMGR, and verify that it is running.
  2. Create a “private” SVRCONN channel to access your z/OS QMGR
  3. Limit access to your private channel.
  4. Control access to a RECEIVER channel
- Userid: TEAMXX
- Passwords: _____



WebSphere MQ for z/OS v7+ Proof-Of-Technology (POT) workshop

Beyond MQ: WebSphere Message Broker

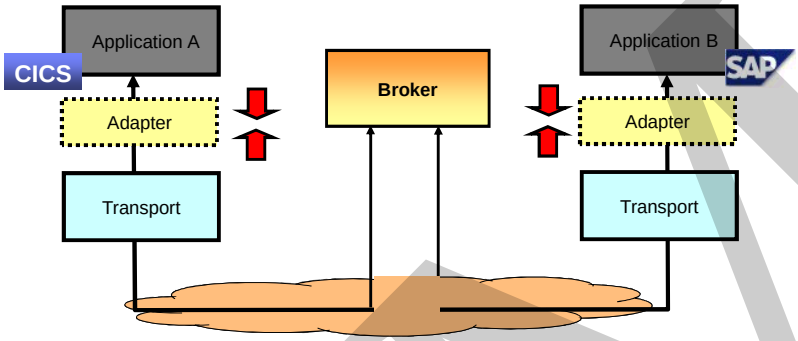
IBM EMEA panIMT Team zWebSphere team

Agenda

- What's an ESB?
- What is IBM WebSphere Message Broker?
  - Development
  - Administration
  - Architecture
- WMB on z/OS
- WMB v8

What is an “Enterprise Service Bus (ESB)”?

"Gartner estimates that up to 30% of the cost of implementing an application is related to the development of the interfaces” (Gartner, January 2000)



- The “Broker” (or “hub” or “gateway” or …) is central and the intermediary between the applications
- It typically has responsibility for routing and transforming data between the applications
- A Broker must be polyvalent with support for multiple data formats and protocols, extremely reliable and scalable

Business value of an ESB

- Traditional communications
  - Point to point
  - Each application must adapt the data to its particular need
  - The number of transformations grows exponentially n*(n-1)
- Communications with a ESB
  - Hub ‘n spoke logic
  - Maintenance can be centralised
  - Promotes reutilisation
  - Scalability and high-availability: multiples physical hubs supported

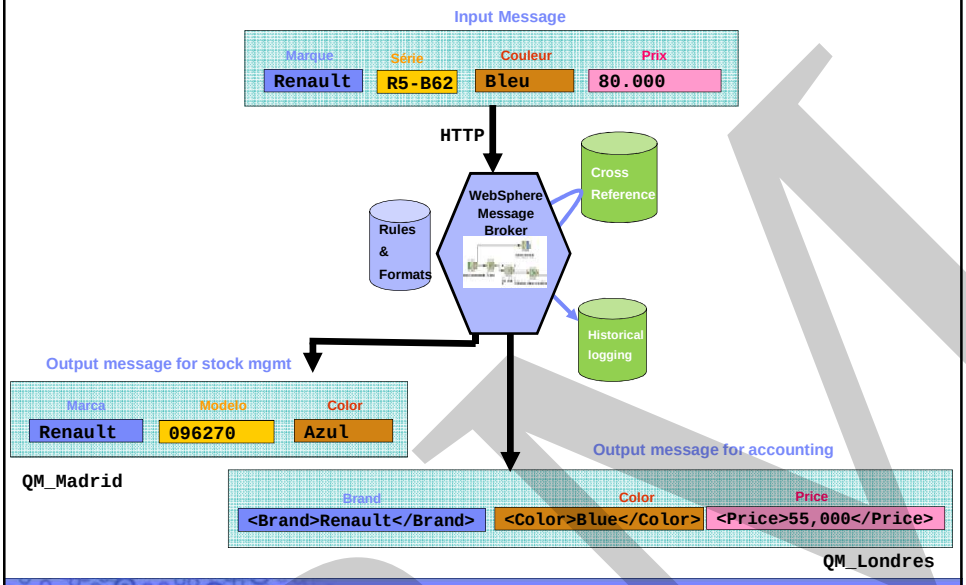
Agenda

- What's an ESB?
- What is IBM WebSphere Message Broker?
  - Development
  - Administration
  - Architecture
- WMB on z/OS
- WMB v8

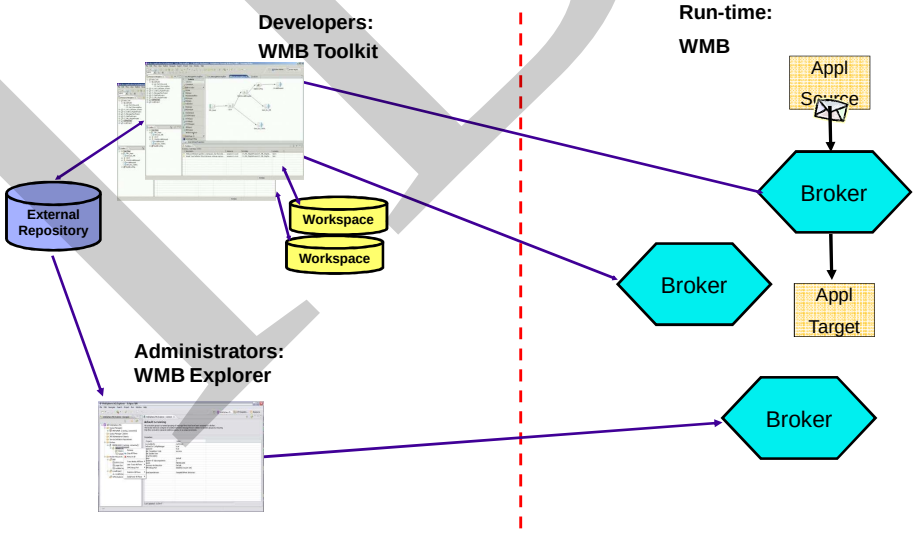
IBM WebSphere Message Broker…

- **Routes, Transforms, Augments** "messages"
- Supports **multi-format** (XML, SOAP, fixed, variable length, tagged, SWIFT, IDOC, etc.)
- Accepts **multi-protocol** (HTTP, JMS, MQ, SOAP, TCP/IP, files, FTP, etc.)
- Offers full **database** support (DB2, Oracle, Informix, Microsoft SQL Server, Sybase, etc.)
- Supports common **ERP** and **EIS** interfaces (CICS, IMS, SAP, PeopleSoft, Siebel, etc.)
- Provides a drag 'n drop **visual development** based upon Eclipse, supporting a variety of **development languages** (Java, eSQL, XSLT, PHP, .Net, etc.)
- Offers a **scalable, high-performance, resilient, low-latency** "execution container"
- Provides **transactional** (2PC) support (MQ, DB2)
- Supports Point-To-Point, Pub/Sub, Event, Synchronous and Asynchronous message processing styles
- Includes full life-cycle tooling (development, administration, runtime)
- Is extensible with open parser, node & administration interfaces
- Supports all major hardware and O/S platforms

WMB: Routing and Transformation

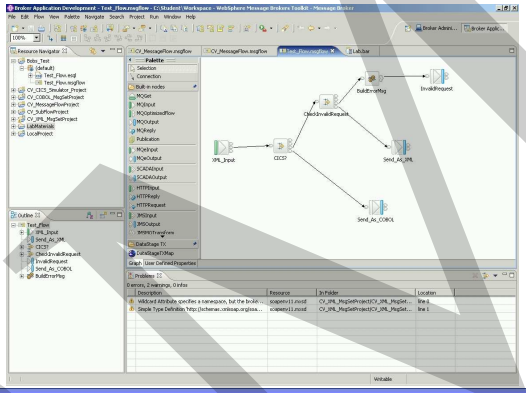


WMB Components & Lifecycle

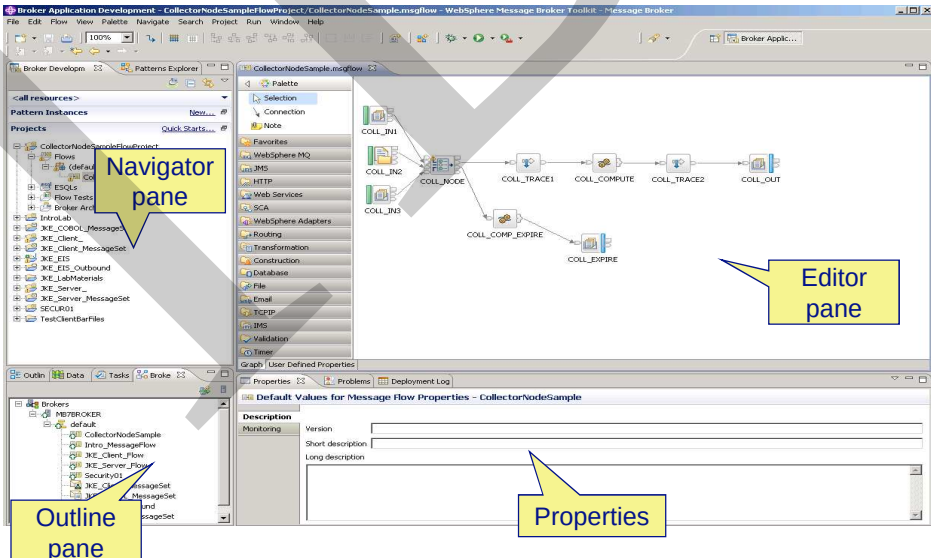


The WebSphere Message Broker Toolkit

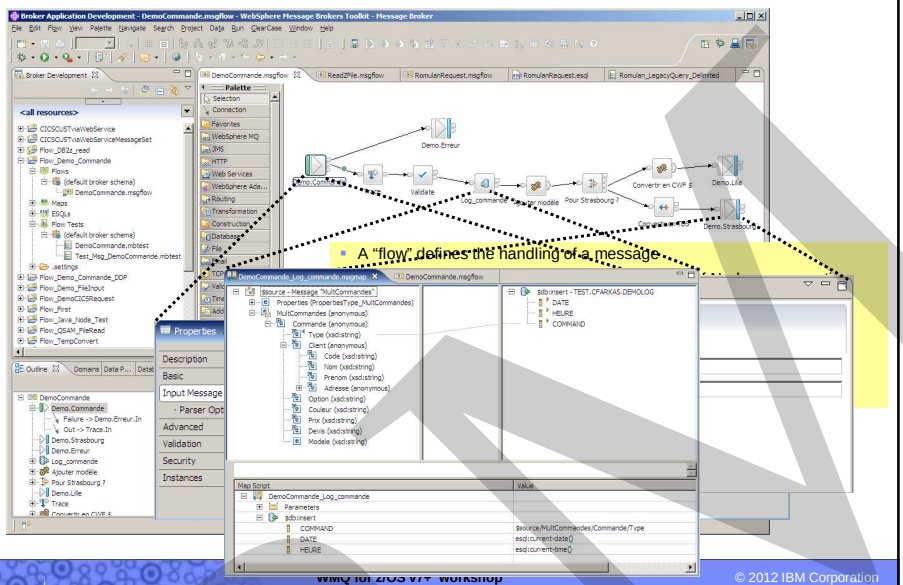
- The GUI used for all ESB development tasks
- eclipse based
- Provides various “perspectives” for different tasks to be performed
  - Message Flow Developer Perspective
  - Debug Perspective
  - Java Perspective
  - etc…
- Let’s take a look at it…….



The WMB Toolkit tour



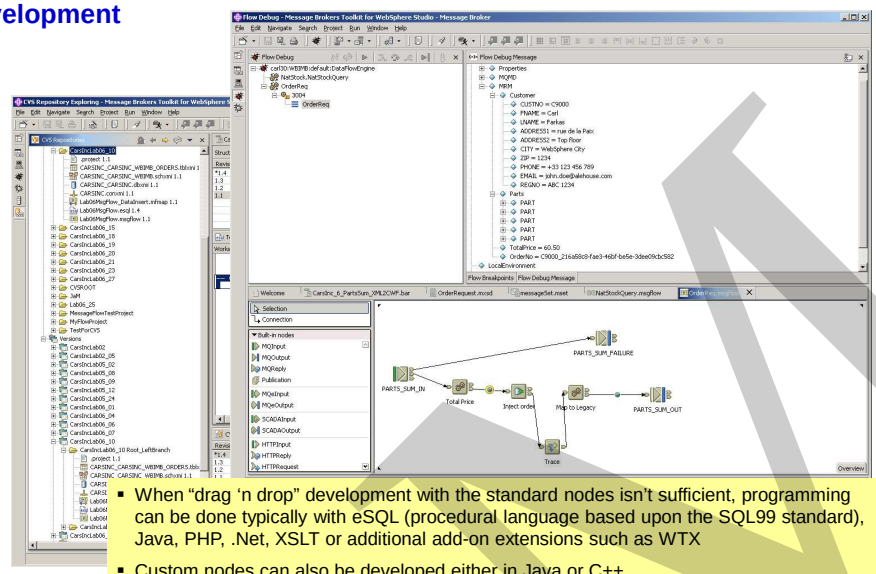
WMB Toolkit - Flow definition



The short list of the standard nodes…

- MQInput
- MQOutput
- MQReply
- SOAP Input
- SOAP Reply
- SOAP Request
- HTTP Input
- HTTP Reply
- HTTP Request
- TCPIP Input
- TCPIP Output
- Real-time Input
- Optimized Real-time Flow
- JMSInput
- JMSOutput
- Aggregate Control
- Aggregate Reply
- Aggregate Request
- Publication
- Validate
- Sequence
- TryCatch
- Throw
- Trace
- TimeOut Notification
- Compute
- Java Compute
- Mapping
- XSLT
- Transformation Extender (WTX)
- Reset Content Descriptor
- WSRR Registry Lookup
- e-mail
- Label
- RouteToLabel
- Collector
- Filter
- Database
- DataInsert
- Warehouse
- SAP
- Siebel
- PeopleSoft
- FileRead
- FileWrite
- FTE Nodes
- CICS
- IMS
- VSAM
- QSAM
- User/Third Party

Development



- When “drag ‘n drop” development with the standard nodes isn’t sufficient, programming can be done typically with eSQL (procedural language based upon the SQL99 standard), Java, PHP, .Net, XSLT or additional add-on extensions such as WTX
- Custom nodes can also be developed either in Java or C++
- WMB Toolkit supports tracing and interactive debugging
- Team development and administration is supported by scripting and/or standard market plugin extensions to the WMB Toolkit, eg. CVS, ClearCase, PVCS, TeamCode, etc.

eSQL

DataInsert

```
IF Root.XML.Person.Taille > 183 THEN
INSERT INTO Database.MesGrandsCopains
       (Name,Height,Age)
 VALUES (Body.Person.Nom,
         Body.Person.Taille,
         Body.Person.Age);
ENDIF;
```

Compute

```
IF (Body.Person.Name = 'Carl') THEN
    OutputRoot.Properties.MessageFormat = 'XML';
ELSE IF (Body.Person.Name = 'Rudi' )
    OutputRoot.Properties.MessageFormat = 'CWF';
ELSE IF (Body.Person.Name = 'SWIFT')
    OutputRoot.Properties.MessageFormat = 'TDS';
    ENDIF;
```

Data types
INTEGER
FLOAT
DECIMAL
STRING
DATETIME
BOOLEAN
REFERENCE
NULL
...

Operators
- + * /
||
AND OR NOT
= <> > >= < <=
IN BETWEEN
LIKE
IS EXISTS
...

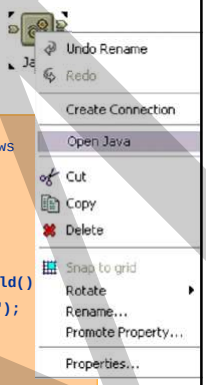
Statements
Basic
DECLARE
SET
IF ENDIF
WHILE
Tree
MOVE
CREATE
DETACH
ATTACH
Database
INSERT
DELETE
UPDATE
PASSTHRU
EVAL
Node
PROPAGATE
RETURN
THROW
...

Functions
String
LENGTH
TRIM LTRIM RTRIM
OVERLAY
POSITION
SUBSTRING
UCASE LCASE
Numeric
ABS
BITAND NOT (X)OR
MOD ROUND
SQRT
TRUNCATE
EXTRACT
Datetime
EXTRACT
CURRENTDATE
CURRENTTIME
Field
BITSTREAM
CARDINALITY
FIELDTYPE
SAMEFIELD
Complex
CAST
SELECT
...

Java Compute Node

- Support for Java as WMB transformation language (Java Compute Node)
  - Complete support for Java JSE 1.6 integrated into WMB
  - Java classes deployed in the WMB Broker Archive with the flow (.BAR)
  - "Wizards" used to simplify the development specific for WMB
  - Classes provided for XPATH message tree navigation
  - zAAP support for Message Broker on z/OS!

```
public class CarlJNode extends MbJavaComputeNode {
   public void evaluate(MbMessageAssembly inAssembly, MbInputTerminal inTerm) throws
   MbException {
         :
    MbMessage outMessage = new MbMessage(inAssembly.getMessage());

    // Add user code below
    MbElement cadet =
            outMessage.getRootElement().getLastChild().getFirstChild().getLastChild()
    rc = cadet.createElementAfter(MbElement.TYPE_NAME, "NewElm","mon truc chouette");
   // End of user code
    MbMessageAssembly outAssembly =
          new MbMessageAssembly(inAssembly, outMessage);
    getOutputTerminal("out").propagate(outAssembly);
    outMessage.clearMessage();
  }
}
```

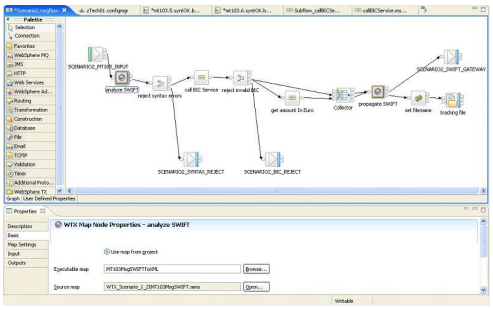


Other integrated transformation options…

- Standard XML Transformations via XSLT
- PHP (Hypertext pre-processor) support

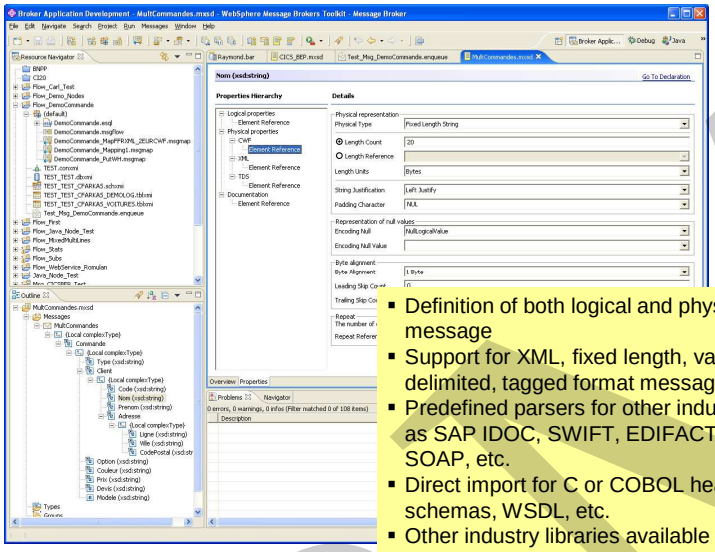
```
$message->a->b->c = $input_body->Message;

for ($index = 0; $index < $output_root->Menu->Food->count; $index++) {
    $item = $output_root->Menu->Food[$index];
}
```

- WebSphere Transformation Extender (WTX) integration - for bulk transformations
  - ✓ Full Development-time integration into toolkit
  - ✓ Full Runtime integration
    - Invoke WTX parser
    - WTX mapping



- .Net integration with WMB v8

WMB Toolkit – Message modelling



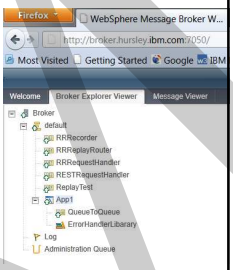
- Definition of both logical and physical format of message
- Support for XML, fixed length, variable length, delimited, tagged format messages, etc.
- Predefined parsers for other industry standards such as SAP IDOC, SWIFT, EDIFACT, ACORD AL3, SOAP, etc.
- Direct import for C or COBOL headers, DTD, schemas, WSDL, etc.
- Other industry libraries available

Agenda

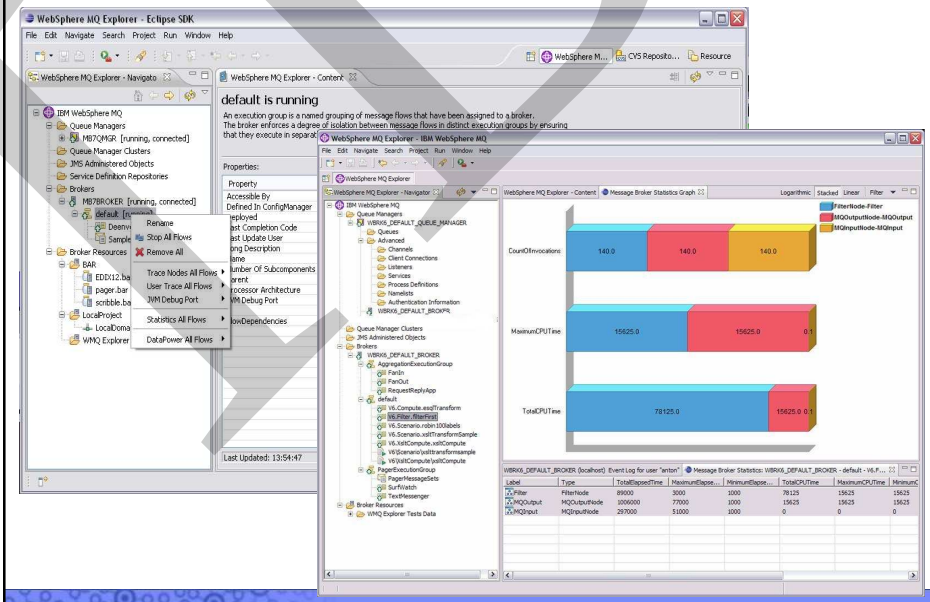
- What's an ESB?
- What is IBM WebSphere Message Broker?
  - Development
  - Administration
  - Architecture
- WMB on z/OS
- WMB v8

WebSphere Message Broker Administration

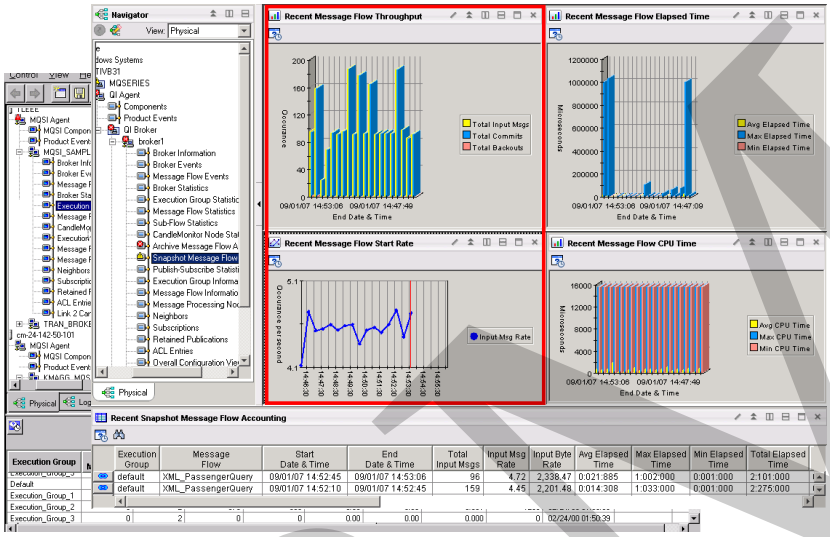
- WebSphere Message Broker offers a wide variety of flexible administration tools
  - This reflects its underlying nature as a production strength tool
  - 3rd party administration tools are also available (BMC, CA, IBM Tivoli…) for integrating into a corporate administration
- ✓ WMB Explorer is the visual (GUI) interface for casual administration
  - eclipse based plugin for WMQ Explorer
  - Available on Linux and Windows
- ✓ Lightweight Web Administration for zero-footprint visual administration
  - Supports IE, Firefox, Safari, Chrome….
- ✓ A full Command Line interface is also available
  - Consistent interface on all platforms
  - Useful for administration from scripting environments
  - SDSF commands also provided for WMB on z/OS
- ✓ The Message Broker Java API is the underlying administration API
  - Available to any administrator to develop routines
  - A solid base for corporate, repeatable, controlled administration
- ✓ REST-based administration interface supporting HTTP clients
  - Compatible with Java API



The WebSphere Message Broker Explorer

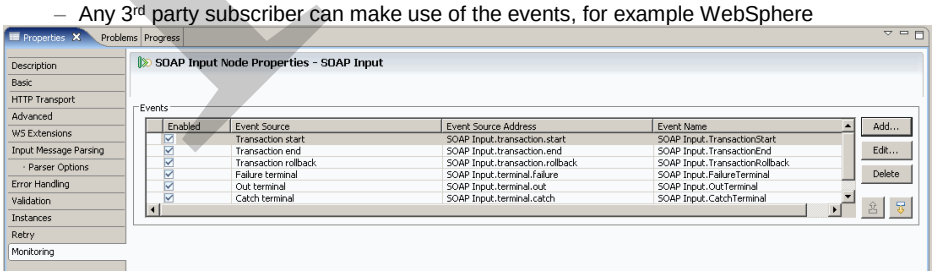


Industrial strength administration – OMEGAMON XE



WebSphere Message Broker business monitoring

- A message flow can be configured to emit events through Monitor Properties on each node
- Events are available for transaction start, transaction end, transaction rollback for Input nodes, and for a message passing into or out of any terminal on any node
- Each event can be manually added – then they can be enabled or disabled.
  - Events can also be defined by the administrator using a monitoring profile
- These Events are for functional & business monitoring, eg. KPIs
  - The entire message or selected fields can be included in the event
- Events are published to a WebSphere MQ topic
- Any subscribed application will receive the events
  - Any 3rd party subscriber can make use of the events, for example WebSphere



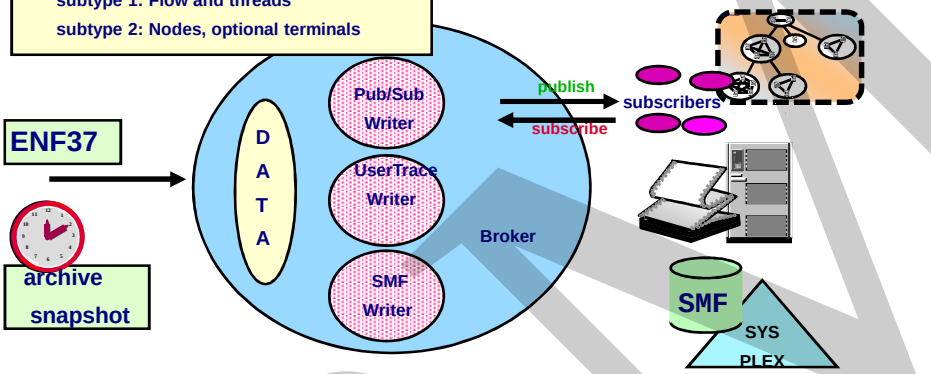
Statistics – technical monitoring

Possible outputs

- MQ(XML) by Pubish/Subscribe
- UserTrace
  - BIP2380/2381/2382/2383 Messages
- z/OS SMF
  - Type 117
    - subtype 1: Flow and threads
    - subtype 2: Nodes, optional terminals

Reporting Scope

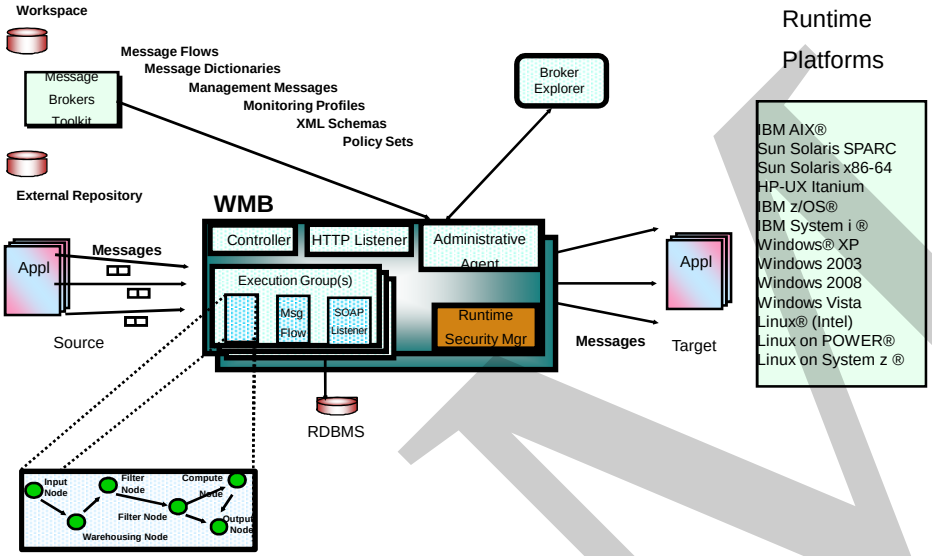
- Flow
- Execution Group
- Broker
- Domain



Agenda

- What's an ESB?
- What is IBM WebSphere Message Broker?
  - Development
  - Administration
  - Architecture
- WMB on z/OS
- WMB v8

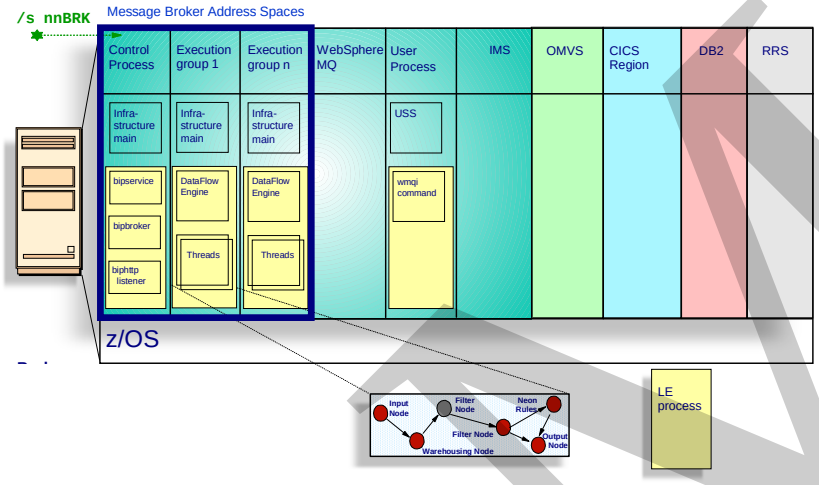
WMB System Architecture



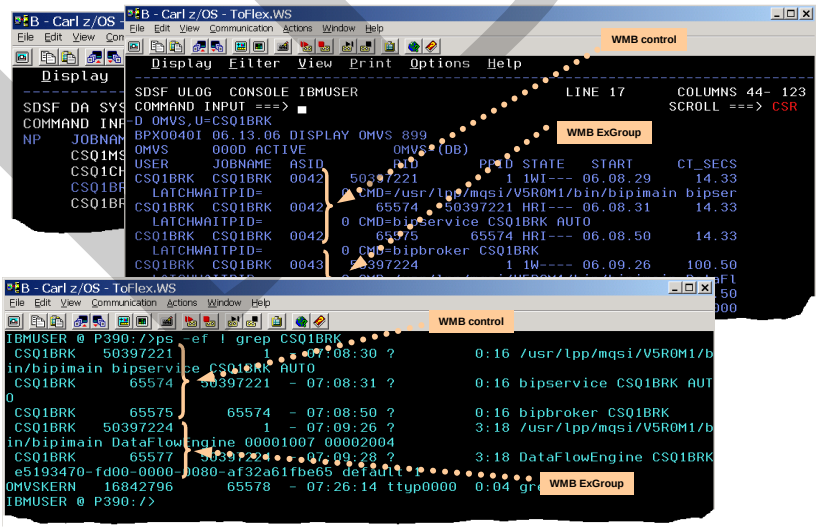
Agenda

- What's an ESB?
- What is IBM WebSphere Message Broker?
  - Development
  - Administration
  - Architecture
- WMB on z/OS
- WMB v8

WMB on z/OS



The Broker address spaces

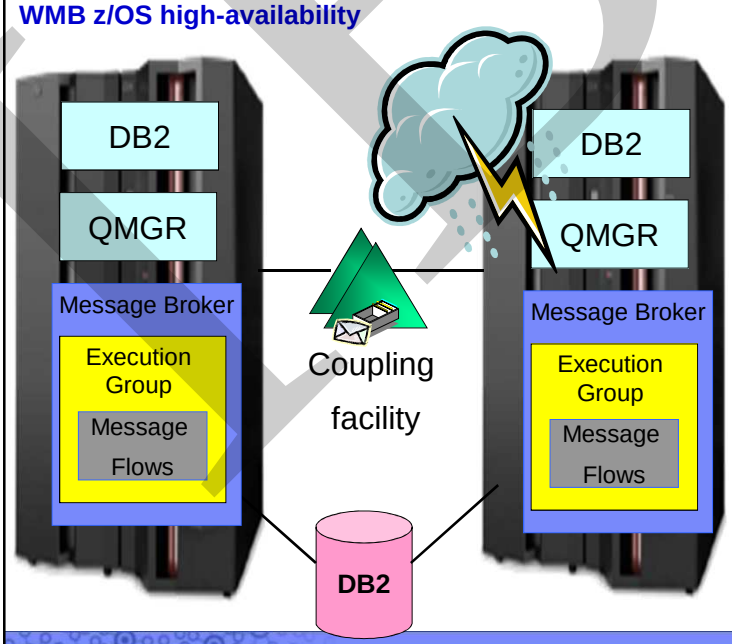


What's different on z/OS?

- The WebSphere Message Broker is essentially identical on all platforms as far as the facilities and any APIs.
  - Almost all flows run "as-is" regardless of the platform
- WMB z/OS does benefit, however, from a few z/OS-specific nodes for better exploiting z/OS:
  - VSAM nodes for direct access to VSAM datasets
  - QSAM nodes for direct access to QSAM datasets
- WMB z/OS also leverages several other z/OS-exclusive features…
  - z/OS is the only platform that can offer the highest levels of **scalability and high-availability** by taking full advantage of the z/OS Parallel Sysplex and WebSphere MQ **Shared Queue** technology
  - WMB uses z/OS **ARM** feature used to auto (re-)start in case of failure
  - z/OS **RRS** is used to ensure WMB transactionality
  - WMB z/OS takes advantage of **WLM** and corporate business goals can be assigned to Execution Groups
  - WMB z/OS makes full use of **SMF** for performance monitoring

Customers choose WMB on z/OS typically when interfacing with host data and/or when they require the best QOS only found on z/OS

WMB z/OS high-availability

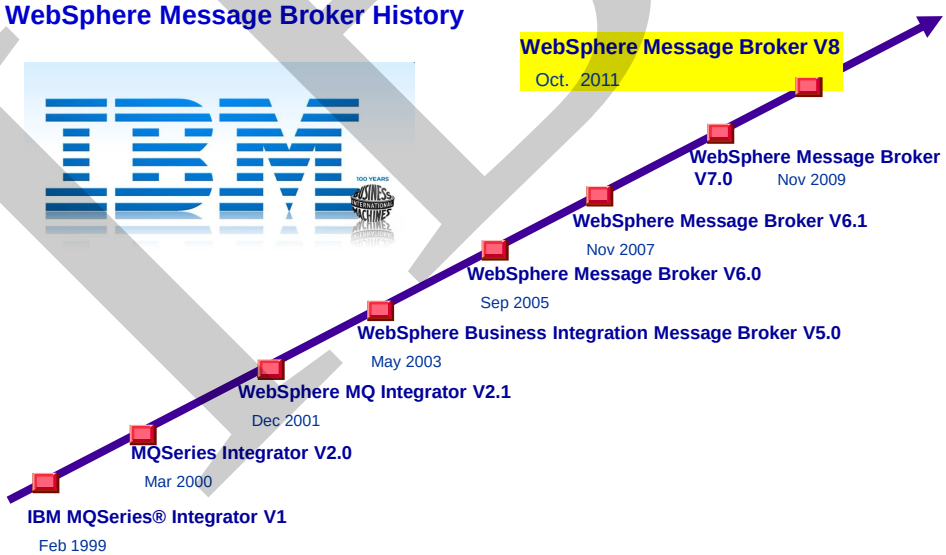


- Any Broker in the QSG can access messages
  - "Natural" load-balancing based upon availability
- Any Broker can recover messages in case of an outage
- z/OS ARM can restart any stopped component

Agenda

- What's an ESB?
- What is IBM WebSphere Message Broker?
  - Development
  - Administration
  - Architecture
- WMB on z/OS
- WMB v8

WebSphere Message Broker History



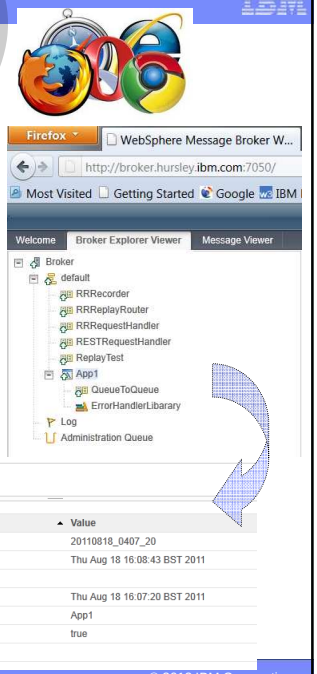
WMB v8.0 - the big picture

WebSphere Message Broker MQ v8.0
General availability: 9 Dec 2011

- Simplify for developers and administrators
  - Web admin
  - Message record and replay
  - Activity Logging for operators & administrators
  - Application and Library organisation
  - Enhanced pattern management
- Continue to lead the industry with up-to-date technology integration
  - DFDL industry-standard data modeling
  - IBM Graphical Data Mapper (GDM) integration
  - .Net embraced
  - Sterling Connect:Direction integration
  - Healthcare HL7 connectivity pack
  - WS-Reliable messaging support
  - Additional DB support

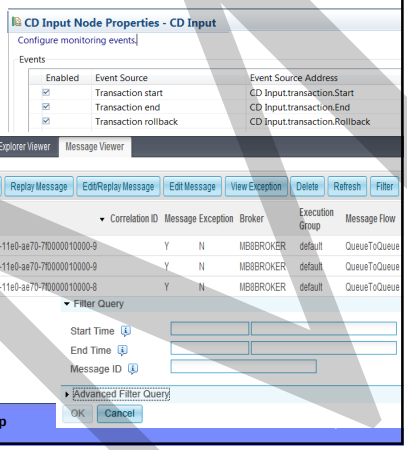
Web Administration for Universal Access

- Web Administration Console
  - Objective is to provide comprehensive web management interface
    - Focus on non-administrators to understand brokers & resources
  - Supports all major browsers Firefox, IE, Opera, Safari, Chrome
  - Designed as users as a complement MBExplorer
    - MB Administrators can users continue to use MB Explorer
- Easy to configure
  - No extra moving parts - uses internal HTTP listener to serve data
    - Web admin started by default on port 7050
  - Role based access provides custom class user control
    - Default is read-only access to MB resources
    - More authority required to create, change or delete resources
- Using Web Admin
  - Intuitive tree view shows hierarchy of MB resources
  - View resource details with click or button
  - Includes full suite of resources
    - Apps, Libs, Flows, Configurable services etc
- Web Admin & MB Explorer
  - MBX & web admin designed for concurrent use
  - Web admin requires MB8 broker
  - Explorer can manage both MB8 & MB7 brokers



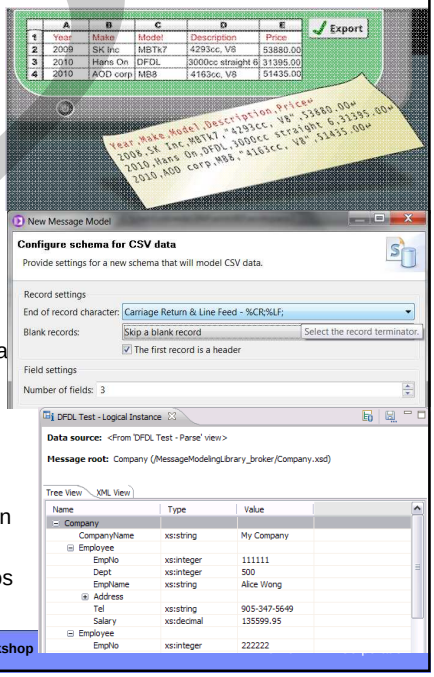
Record & Replay

- Enable Record, Edit and Replay of In-flight Data
  - Comprehensive audit of messages, web, ERP, file & other data
  - Flexible topology: single or multiple brokers for recording, capture & replay
- Data Recording, Capture & Store
  - Graphically configure binary, text, XML payload capture, including whole, partial & multi-field data
- Web Tooling to View, Query, Edit data
- Friendly editors to view, query & edit payloads
  - Key data fields, including application data
  - Independent web admin & capture for scalability
    - Configure multiple EG listeners for web
- Replay for redelivery or flow reprocessing
  - Replay selected data to flows or applications
  - MB admin configures logical destinations
  - User selects destinations from auto-populated drop-down list



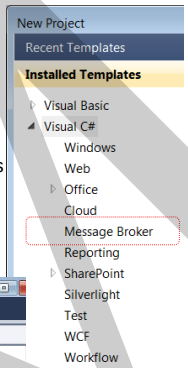
Easy Data Modelling with DFDL

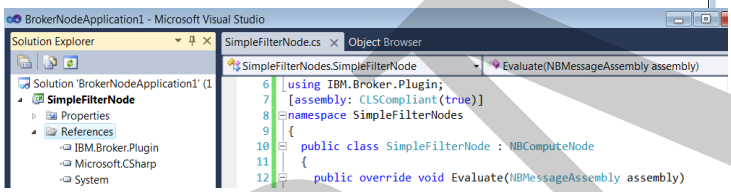
- Simple & powerful standard for data modelling
  - New standard for binary, text & industry data formats
    - Logical structure with physical annotations
    - e.g. endian, ASCII/EBCDIC, padding, justify…
  - Data Format Description Language (DFDL)
    - For use in IBM and non-IBM products
    - forge.gridforum.org/projects/dfdl-wg
- Built-in facilities to model data easily
  - Quick wizards for (e.g.) CSV, record oriented data
  - Auto-model importers (e.g.) COBOL copybooks
  - DFDL editor for power users
    - Create logical model & physical refinements
- Test parsing and test data generation
  - Test whether sample data fits with DFDL definition
    - Parse trace provide success & error case explanation
  - Auto-generate test data for test & debug scenarios



Deep Integration for .NET

- New node for native .NET program invocation
  - Call .NET programs directly via CLR V4; includes app domains for isolation
  - C#, VB .NET (COM), JScript & F# programming available natively in MB
    - Extensive range of .NET data types supported for easy integration
- Integrated Visual Studio Development
  - Create .NET nodes in Visual Studio; Native MB assemblies simplifies process
    - e.g. packages, templates, #using, debug, content assist etc
  - Visual studio compiled resources available without redeploy
- Call .NET programs from new and existing MB nodes
  - Toolkit can introspect .NET assemblies to dynamically discover available methods
    - Automatically create appropriate language signatures to simplify invocation
  - (e.g.) ESQL can now directly invoke .NET programs using simple procedure call





Summary

- Universal Enterprise Service Bus
  - WMB supports a wide range of different data formats, protocols and transformation “languages”
    - Universal connectivity includes standards, de facto standards, industry and custom systems

- Simple & Productive
  - WMB provides a complete interface for defining, modifying and routing data
    - Learn, Develop, Deploy, Manage, Migrate quickly and easily

- Dynamic, Managed & Secure
  - WMB allows the creation of dynamic and governed solutions
    - Design solutions for easy, flexible change with appropriate control
- High Performing & Scalable
  - WMB provides a platform and technology neutral connectivity option
    - Work on the widest possible range of hardware, software and virtualized environments

Bibliography

- GI13-0582 WebSphere Message Broker for z/OS V8.0 Program Directory
- WebSphere Message Broker home page - http://www.ibm.com/software/integration/wbimessagebroker/
- SG24-7090 WebSphere Business Integration Message Broker Basics
- SG24-7006 Implementing and Administering WebSphere Business Integration Server
- SG24-6088 WebSphere Business Integration Pub/Sub Solutions
- SG248020 Using WebSphere Message Broker V8 in Mid-Market Environments
- REDP3894 Implementing High Availability for WBI Message Broker on z/OS
- SG24-7283 Managing WMB Resources in a Production Environment
- SG24-7826 Connecting Your Business Using IBM WebSphere Message Broker V7 as an ESB
- WP101617 Migration to WebSphere MQ V7.0.1 and WebSphere Message Broker V7.0 on z/OS
- SG24-7335 Implementing an ESB using WMBv6 and WESBv6 on z/OS
- REDP4644 WebSphere Message Broker V7.0 Integration with WebSphere Adapter for SAP

The full Broker documentation is available online at

http://publib.boulder.ibm.com/infocenter/wmbhelp/v8r0m0/index.jsp

WebSphere MQ for z/OS v7+ Proof-Of-Technology (POT) workshop

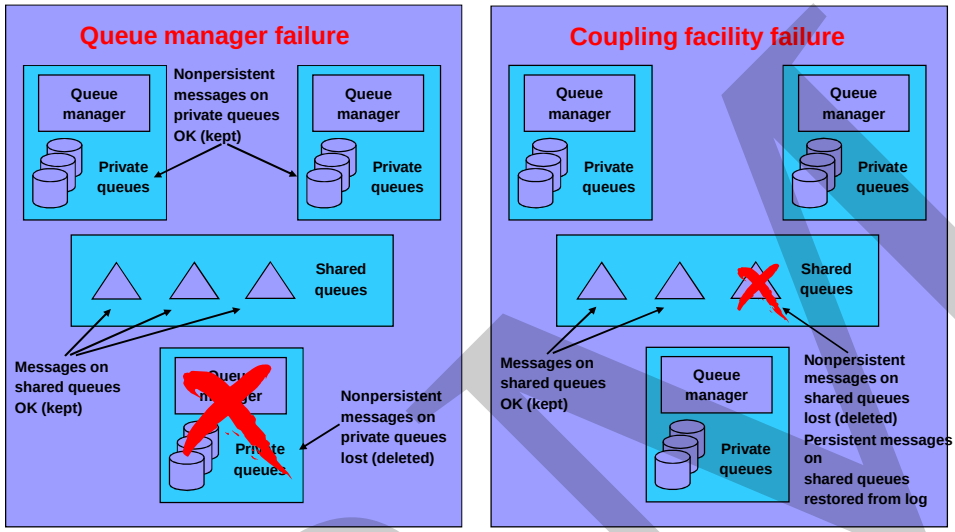
Shared Queue Changes in WMQ v7

IBM EMEA panIMT Team zWebSphere team
(with thanks to IBM US ATS team for the original “Wildfire” material)

Agenda

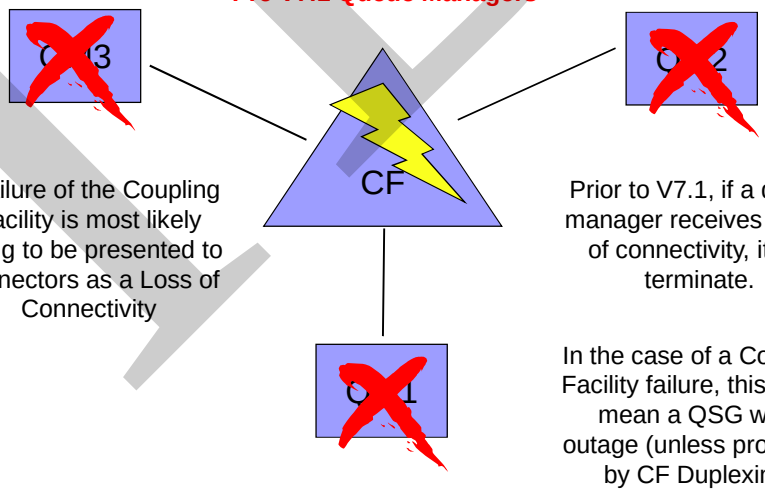
- **Failure and Message Persistence**
- **CF Loss of Connectivity Tolerance**
- **Group UoR**
- **How messages are stored**
  - The Coupling Facility
  - DB2 Blobs
  - Shared Message Data Sets (SMDS)
  - Offload Rules
- **Administration Tasks**
  - Creating the Queue Sharing Groups
  - Setting up the DB2 tables
  - Setting up the SMDS
- **Preliminary Performance Numbers**

Failure and message persistence



CF Loss of Connectivity Tolerance

Pre V7.1 Queue Managers

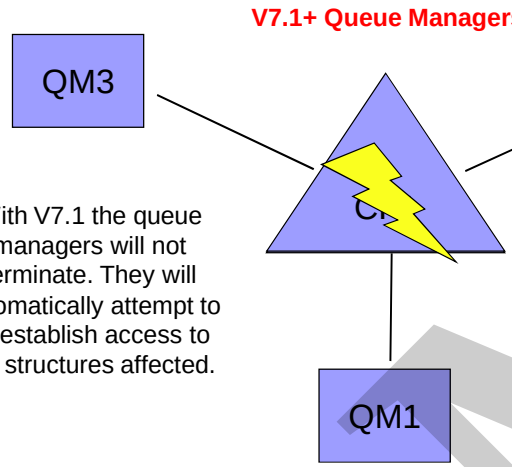


A failure of the Coupling Facility is most likely going to be presented to connectors as a Loss of Connectivity

Prior to V7.1, if a queue manager receives a loss of connectivity, it will terminate.

In the case of a Coupling Facility failure, this would mean a QSG wide outage (unless protected by CF Duplexing)

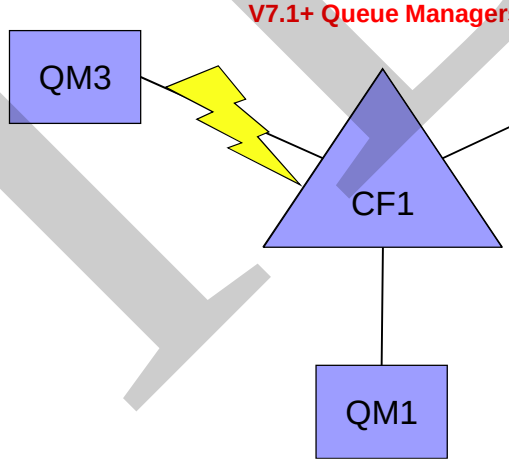
CF Loss of Connectivity Tolerance



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)

CF Loss of Connectivity Tolerance



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

CF Loss of Connectivity Tolerance

- QMGR CFCONLOS(TERMINATE|TOLERATE)
  - Specifies whether loss of connectivity to the admin structure should be tolerated
  - Default is TERMINATE
  - Can only be altered to TOLERATE when all QSG members are at 7.1
- CFSTRUCT CFCONLOS(TERMINATE|TOLERATE|ASQMGR)
  - Specifies whether loss of connectivity to application structures should be tolerated
  - Only available at CFLEVEL(5)
  - Default is ASQMGR for new CFLEVEL(5) structures, and TERMINATE for structures altered to CFLEVEL(5)
- CFSTRUCT RECAUTO(YES|NO)
  - Specifies whether application structures should be automatically recovered
  - Only available at CFLEVEL(5)
  - Default is YES for new CFLEVEL(5) structure, and NO for structures altered to CFLEVEL(5)

Safeguarding against CF failure

- **Administration structure updates are logged so that this structure can be restored.**
- **Coupling Facilities are very rugged (zSeries processor technology).**
- **CF can have its own separate power supply.**
- **CF can have nonvolatile memory (battery power backup).**
- **Lost application structures can be restored from backups and logs. (can use BACKUP CFSTRUCT(*) at V7.0.1)**

How Shared Messages are stored

- **Each message is composed to two components:**
  - Message Header(s)
  - Message Body
- **The message headers are always stored in the CF structure**
- **If the message is over 63K, the message body must be offloaded**

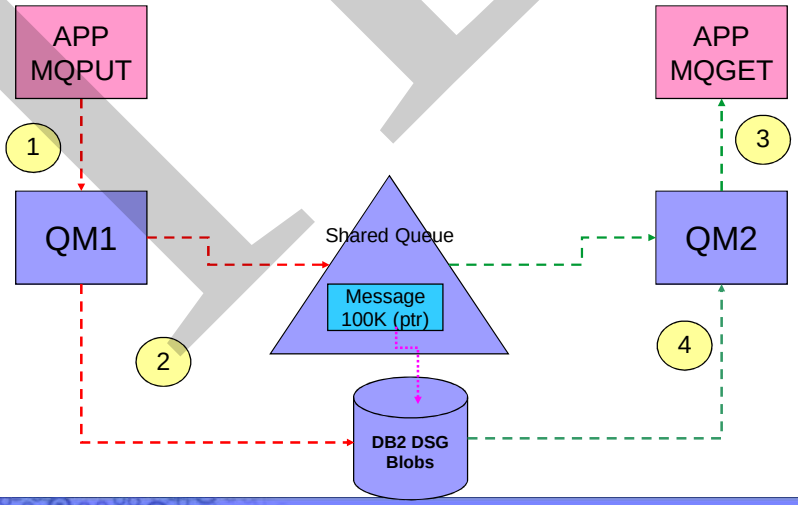
MQ Use of the Coupling Facility

- The CF is a precious resource
- Total size is typically 32G or less
  - MQ only gets a piece of this
- The number of messages that can be stored has traditionally been limited
  - If all messages are large, there can be fewer on a queue
  - Application and channel outages can fill the available storage quickly
  - Applications accustomed to having 64G of storage for a private queue may start seeing 2192 codes

WMQ 7.1 - Options for Message Body Storage

- Options for message body storage
  - CF for messages up to a specified size
    - Defaults to 63K
  - DB2 for messages body
  - New - Shared Message Data Sets (SMDS) for message body
  - New - Three offloading rules
    - MQ detects the CF structure full level
      - If the pre-defined threshold has been reached, the message body of any message over the specified size will be off loaded

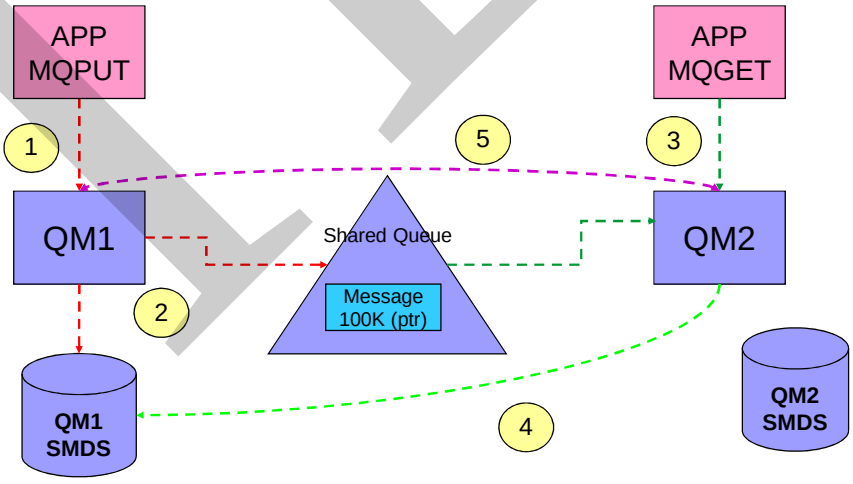
Message Offloading with DB2



Shared message data set concepts

- Offloaded message data for shared messages is stored in data sets.
- Each application structure has an associated group of shared message data sets, with one data set per queue manager.
  - Named using DSGROUP parameter on CFSTRUCT definition.
- Each queue manager owns a data set for each structure, opened for read/write access, which it uses to write new large messages.
- Each queue manager opens the data sets for the other queue managers for read-only access, so it can read their message data.
- When a message with offloaded data needs to be deleted, it is passed back to the queue manager which originally wrote it, so that the queue manager can free the data set space when it deletes the message.

Message Offloading Using Shared Message Data Sets (SMDS)

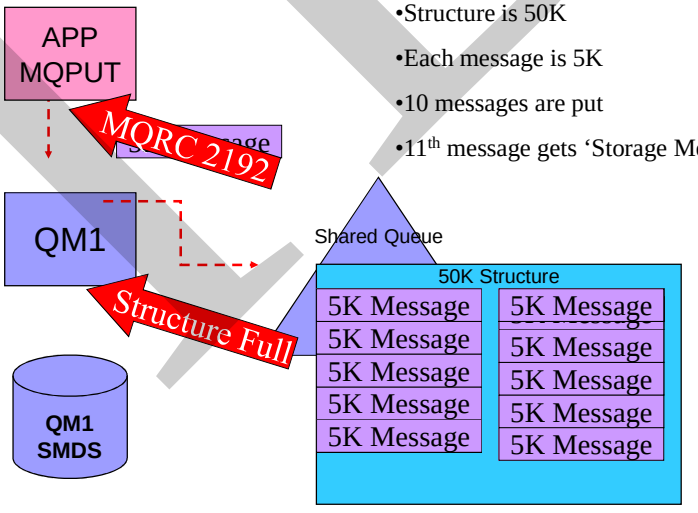


Offload Rules

- Three offload rules are provided:
  - All rules have two parameters
    - The CF full level (%)
    - The maximum message size to be stored in CF
  - Ideally they will be progressive
    - No logic checking is done

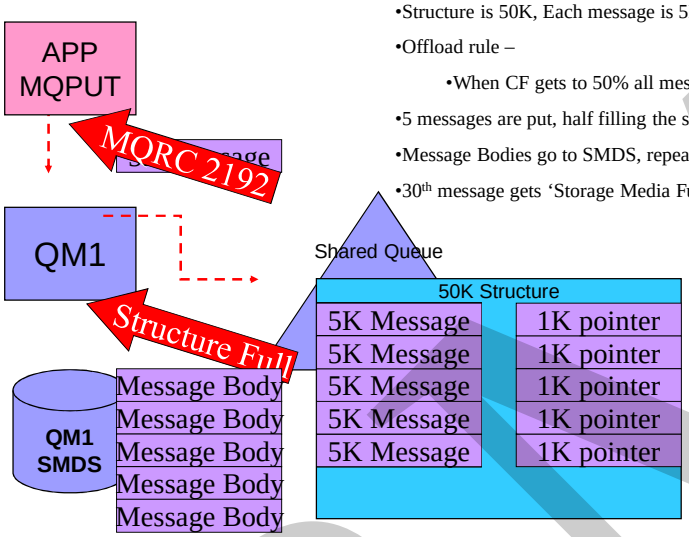
No Offload Rules – Filling the CF Structure

- Simple Example:
  - Structure is 50K
  - Each message is 5K
  - 10 messages are put
  - 11th message gets 'Storage Media Full' Return Code



Offload Rules – Filling the CF Structure

- Simple Example:
  - Structure is 50K, Each message is 5K
  - Offload rule –
    - When CF gets to 50% all message bodies are offloaded
  - 5 messages are put, half filling the space
  - Message Bodies go to SMDS, repeats until the 26th message
  - 30th message gets 'Storage Media Full' Return Code



Admin - Creating a queue-sharing group

Use the CSQ5PQSG utility to create a QSG:

1 Add the QSG into the DB2 tables:

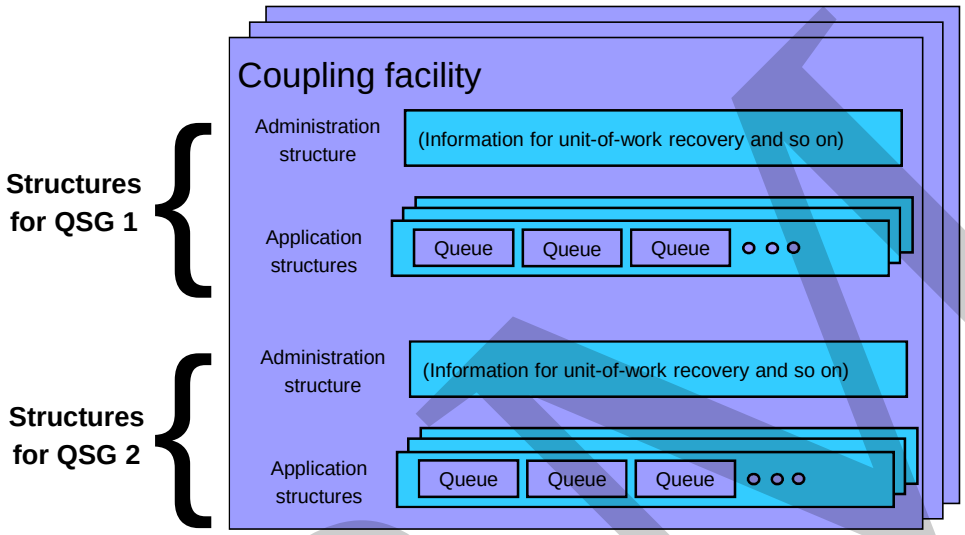
```
//stepname EXEC PGM=CSQ5PQSG,
//    PARM='ADD QSG,qsg-name,dsg-name,DB2-ssid'
```

2 Add the queue managers into the DB2 tables as members of the QSG:

```
//stepname EXEC PGM=CSQ5PQSG,
//    PARM='ADD QMGR,qmgr-name,qsg-name,dsg-name,DB2-ssid'
```

qsg-name	Name for the queue-sharing group
qmgr-name	Name of the queue manager
dsg-name	Name of the DB2 data-sharing group
DB2-ssid	DB2 subsystem ID

CF Structures for shared-queues



Creating CF structures and shared queues

- Define a structure to z/OS (not to WebSphere MQ) by updating the CFRM policy (see *System Setup Guide*):
  - Structure is known to WebSphere MQ by its 12-character *str-name*.
  - Structure is known to z/OS by the 16-character name formed by:
    - *qsg-name* || *str-name* (Application structures)
    - *qsg-name* || CSQ_ADMIN (Administration structure)
- Define the application CFSTRUCT to WMQ
  - DEFINE CFSTRUCT(*str-name)*
    - *CFLEVEL(3/4/5)*
    - ***CFCONLOS(ASQMGR/TERMINATE/TOLERATE)***
    - *RECOVER(YES/NO)*
    - *RECAUTO(YES/NO)*
    - *OFFLOAD(SMDS/DB2)*
      - *SMDS is the default if CFLEVEL(5)*
    - *OFFLDnSZ(?k) OFFLDnTH(?)*
      - *Where n = 1, 2 or 3*
      - *SZ is the size of messages to be offloaded*
      - *TH is the CF percent full that drives the offload*
    - DSGROUP('data.set.name.*')
    - DSBLOCK(256K)
    - DSBUFFS(100)
    - DSEXPAND(YES/NO)

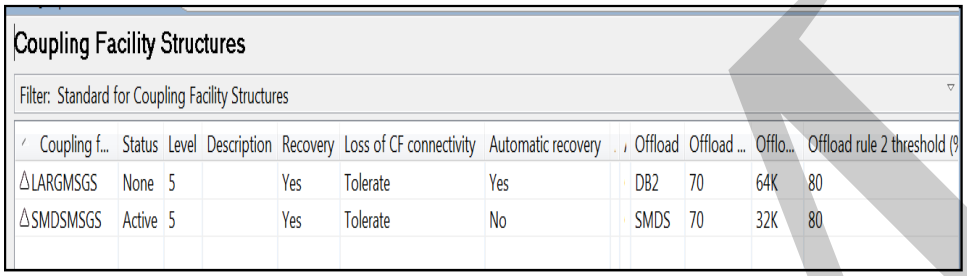
Structure display

```
15.28.43 STC21308  CSQM201I QML1 CSQMDRTC  DIS CFSTRUCT DETAILS  486
   486             CFSTRUCT(SMDSMSGS)
   486             DESCR( )
   486             CFLEVEL(5)
   486             RECOVER(YES)
   486             OFFLOAD(SMDS)
   486             OFFLD1TH(70)
   486             OFFLD1SZ(32K)
   486             OFFLD2TH(80)
   486             OFFLD2SZ(4K)
   486             OFFLD3TH(90)
   486             OFFLD3SZ(0K)
   486             DSGROUP(WMQ710.*.SMDSMSGS.SMDS)
   486             DSBLOCK(256K)
   486             DSBUFS(100)
   486             DSEXPAND(YES)
   486             RECAUTO(NO)
   486             CFCONLOS(TOLERATE)
   486             ALTDATE(2011-10-18)
   486             ALTTIME(17.30.20)
   486              END CFSTRUCT DETAILS
15.28.43 STC21308  CSQ9022I QML1 CSQMDRTC ' DIS CFSTRUCT' NORMAL COMPLETION
```


Structure Display – Part 2

```
16.00.24 STC21308  CSQM201I QML1 CSQMDRTC  DIS CFSTRUCT DETAILS  491
   491             CFSTRUCT(LARGMSGS)
   491             DESCR( )
   491             CFLEVEL(5)
   491             RECOVER(YES)
   491             OFFLOAD(DB2)
   491             OFFLD1TH(70)
   491             OFFLD1SZ(64K)
   491             OFFLD2TH(80)
   491             OFFLD2SZ(64K)
   491             OFFLD3TH(90)
   491             OFFLD3SZ(64K)
   491             DSGROUP( )
   491             DSBLOCK(256K)
   491             DSBUFS(100)
   491             DSEXPAND(YES)
   491             RECAUTO(YES)
   491             CFCONLOS(TOLERATE)
   491             ALTDATE(2011-10-18)
   491             ALTTIME(21.41.24)
   491              END CFSTRUCT DETAILS
16.00.24 STC21308  CSQ9022I QML1 CSQMDRTC ' DIS CFSTRUCT' NORMAL COMPLETION
```

Structure display



Coupling Facility Structures

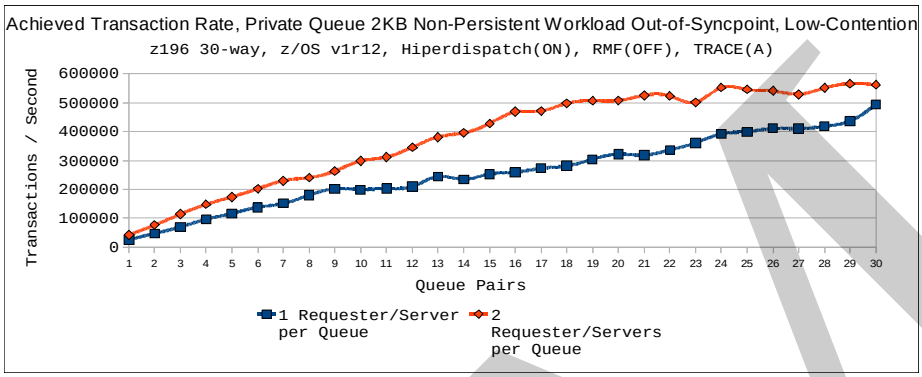
Filter: Standard for Coupling Facility Structures

Coupling f...	Status	Level	Description	Recovery	Loss of CF connectivity	Automatic recovery	Offload	Offload ...	Offlo...	Offload rule 2 threshold (%
LARGMSGS	None	5		Yes	Tolerate	Yes	DB2	70	64K	80
SMDSMSGS	Active	5		Yes	Tolerate	No	SMDS	70	32K	80

MQ V7.1 on z/OS – Highlights so far

- **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
  - MQ SMDS (Shared Message Data Sets) providing an alternative to DB2 for offloading shared queue message data
    - Provides TBs of offload message body storage
    - Significant throughput improvements and CPU reduction
    - The ability to customise the definition of “large” (no longer statically defined as 63KB) allows customers greater control over their CF usage
- **Availability**
  - MQ structure rebuild capability for CF Connectivity loss scenarios significantly improve availability of Shared Queues
  - GroupUR function introduced in MQ V7.0.1 for Distributed QSG connections is now available for CICS usage
    - CICS 4.2 can utilise this to improve the existing MQ Group Attach in CICS 4.1

Private Queue, non-persistent, Out-of-Syncpoint

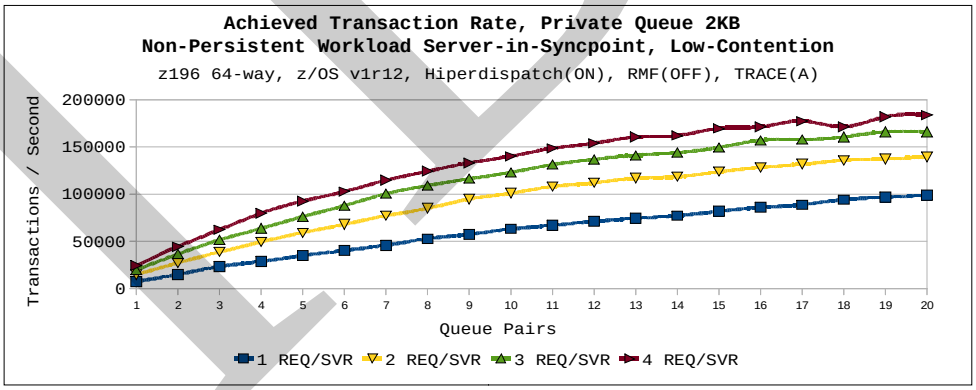


This measurement shows that MQ is able to exceed 550,000 transactions per second on a single queue manager running on a 30-processor LPAR
- and was repeated with similar results on a 64-processor LPAR.

(Note that with current MQ V7.0.1, we max out in this scenario at 330,000 tps)

Each transaction involves a requester task putting a message, a server getting the message and putting a reply and the requester getting the specific reply message - i.e. 2 MQPUT/MQGET pairs.
So a single queue manager is able to support a message rate of **1.1 million messages / second !!**

Private Queue, non-persistent, In-Syncpoint

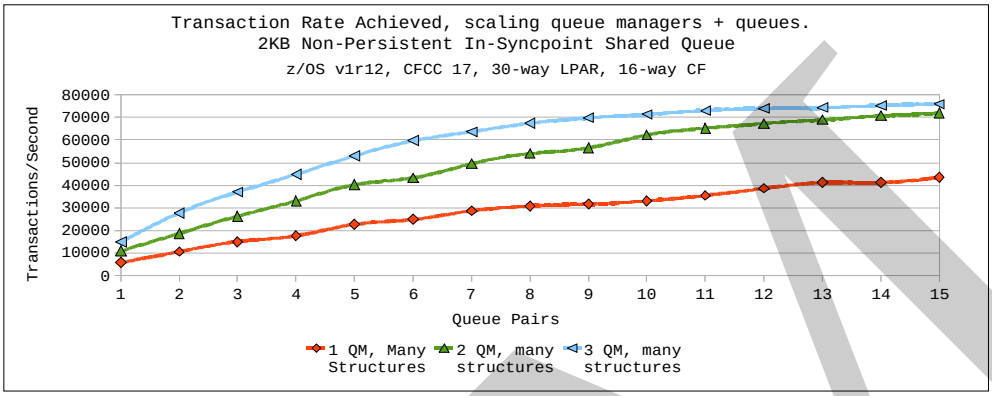


This measurement shows that when the message is put and gotten in-syncpoint, the transaction rate can hit 180,000 transactions a second (360,000 messages per second).

In this case, the measurement was driving the 64-processor LPAR at 77% of capacity, i.e. 50 processors-worth.

The measurement on the 30-processor LPAR peaked at 150,000 transactions per second

Shared Queue, non-persistent, In-Syncpoint



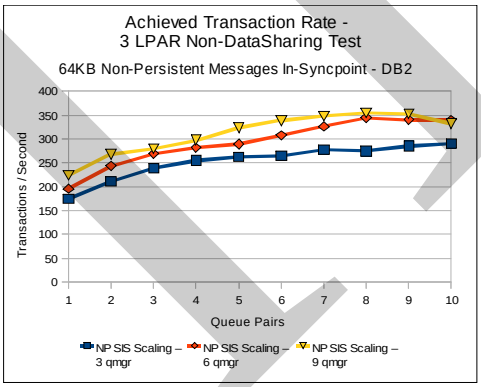
This measurement is run against a single coupling facility with the maximum number of single CF processors defined (16), but using up to 15 application structures within that coupling facility.

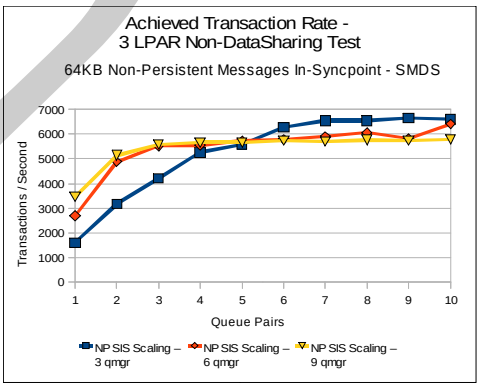
Both the LPAR and the CF are running at 90% of capacity at the peak transaction rate.

Currently investigating 75,000 tps – 150,000 messages/second – CF CPU constraints

Persistent messaging results suggest that we can sustain the above rate – 150K msgs/sec - with 5-6 Qmgrs

MQ V7.1 on z/OS SMDS Test results on z196





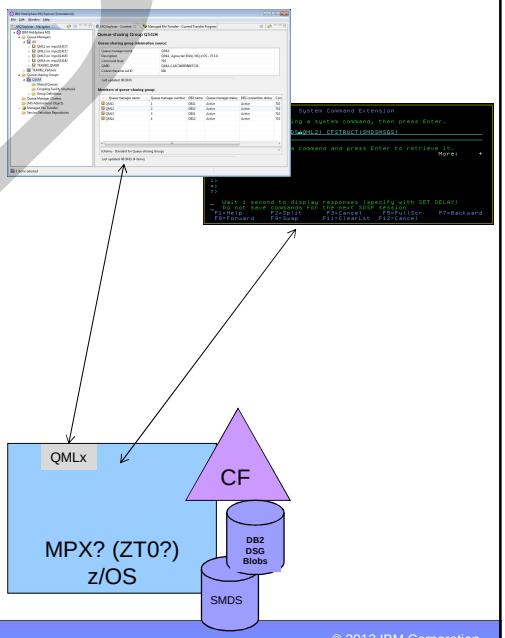
- **Tests show comparable CPU savings making SMDS a much more usable feature for managing your CF Storage usage**
- **SMDS per CF structure provides better scaling than DB2 BLOB storage**
- **zBLC Req. WRBC0407-1050 (RBC) (CF Flash Offload will complete the story !)**

Bibliography

- GC34-6926 WebSphere MQ for z/OS v7 Concepts and Planning Guide
- MQ InfoCenter for Shared queues
  - http://publib.boulder.ibm.com/infocenter/wmqv7/v7r1/topic/com.ibm.mq.doc/zc10420_.htm
- SupportPacs MP16, MP1E, MP1F, MP1G and MP1x
  - http://www.ibm.com/software/integration/support/supportpacs/perfreppacs.html
- REDP-3636 – Redpaper WebSphere MQ Queue Sharing Group in a Parallel Sysplex environment
  - http://www.redbooks.ibm.com/redpieces/pdfs/redp3636.pdf

Lab time – Shared queues

- Using your 3270 emulator as well as the MQ Explorer, you'll experiment with some of the Shared Queue administration commands
- You'll become familiar with the IP13 MQ z/OS load injector
- You'll run some simple performance tests comparing…
  - Private queues vs. shared queues
  - Persistent vs. non-persistent messages
  - DB2 vs. SMDS offloading



WebSphere MQ for z/OS v7+ Proof-Of-Technology (POT) workshop

Beyond MQ: Advanced Message Security

IBM EMEA panIMT Team zWebSphere team

Agenda

- What is MQ AMS?
- Key features
- Pre-requisites and runtime environment
- Logical architecture
- Components
- Installation & configuration

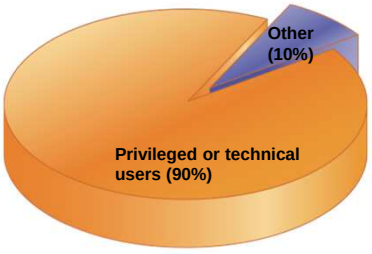
Why use message-level security?

- MQ networks : difficult to prove security of messages
  - Against message injection / message modification / message viewing
  - Prevalence of sub-contractors
  - Increasing levels of partnerships
- More and more data subject to standards compliance
  - Credit card data protected by PCI
  - Confidential government data
- Remember that base WebSphere MQ only provides message encryption when the MQ messages *are in transit over channels*. Without AMS, ***MQ messages have never been encrypted while they are sitting in the queues with standard MQ***!

The *"thief"* is inside the gate!

- **The enemy is us:**
  - **90% of insider incidents are caused by privileged or technical users**
  - **Most are inadvertent violations of:**
    - Change management process
    - Acceptable use policy
    - Account management process
  - **Others are deliberate, due to:**
    - Revenge (84%)
    - "Negative events" (92%)
  - **Regardless, too costly to ignore:**
    - Internal attacks cost 6% of gross annual revenue or 9 dollars per employee per day

Who Causes Internal Incidents?

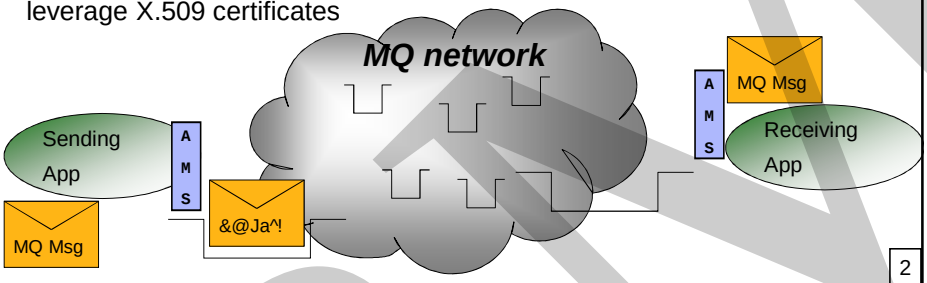


Sources: Forrester research, IdM Trends 2006; USSS/CERT Insider Threat Survey 2005/6; CSI/FBI Survey, 2005; National Fraud Survey; CERT, various documents.

What is MQ AMS?

WebSphere MQ Advanced Message Security V7.0.1

- New product, announced and available Oct 5, 2010
- Provides security for MQ messages, end-to-end with no application changes
- It is a simple “add-on” product that enhances WebSphere MQ v6 or v7
- Security policies are used to define the security level required which leverage X.509 certificates



2

AMS Key Features

- Secures sensitive or high-value MQ messages
  - Privacy via message content encryption
  - It leverages digital certificates (X.509) and Public Key Infrastructure (PKI) to protect MQ messages
- Detects and removes rogue or unauthorized messages before they are processed by receiving applications
  - Authentication via certificate *above and beyond* operating system
  - Authorization to queue *above and beyond* MQ OAM or SAF
- Verifies that messages are not modified in transit
  - Message Integrity via digital signature of message content
- Protects messages not only when they flow across the network but when they are at rest in queues
- Messages from existing MQ applications are transparently secured using “interceptors”
  - No application changes are necessary
- No pre-requisite products other than MQ

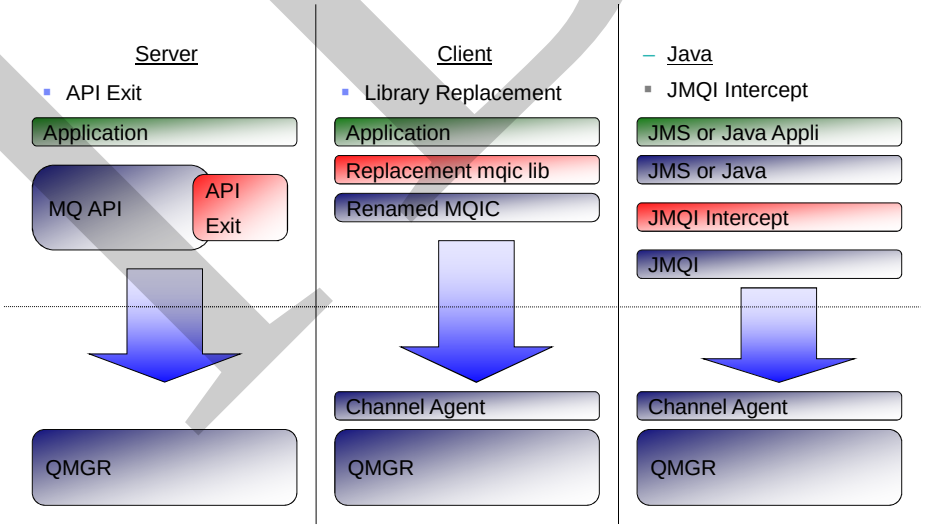
Platforms supported (5724-Z94 for Distributed, 5655-W50 for z/OS)

- Windows (32 & 64-bit, XP Pro, Server 2003, Server 2008, Vista)
- AIX for System p (v5.3, v6.1)
- HP-UX Itanium & PA-RISC (11i v2 & v3)
- Linux for System p (64-bit, RHEL v4 & v5, SLES v9, v10, v11)
- Linux for System x (32 & 64-bit, RHEL v4 & v5, SLES v9, v10, v11)
- Linux for System z (64-bit, RHEL v4 & v5, SLES v9, v10, v11)
- Solaris for Intel X86 (64-bit, v10)
- Solaris for Sun SPARC (64-bit, v9 & v10)
- z/OS for System z (z/OS v1.8) (IBM SSL v1.8 is also required; ICSF is required if CryptoExpress2 or 3 used)

For complete details, see:

http://www.ibm.com/software/integration/wmq/advanced-message-security/reqs/

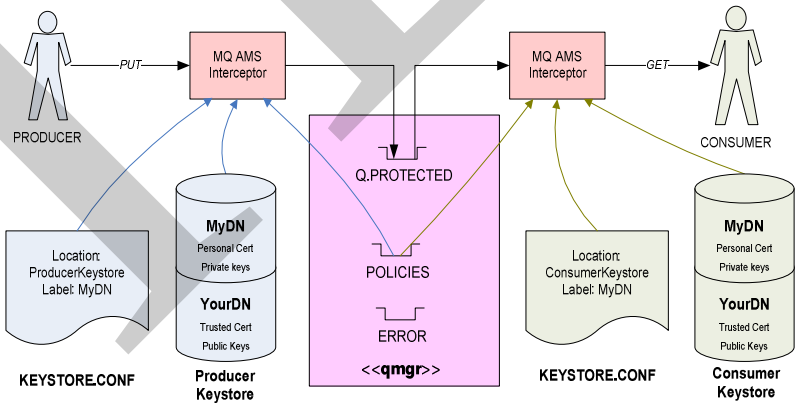
Interceptors



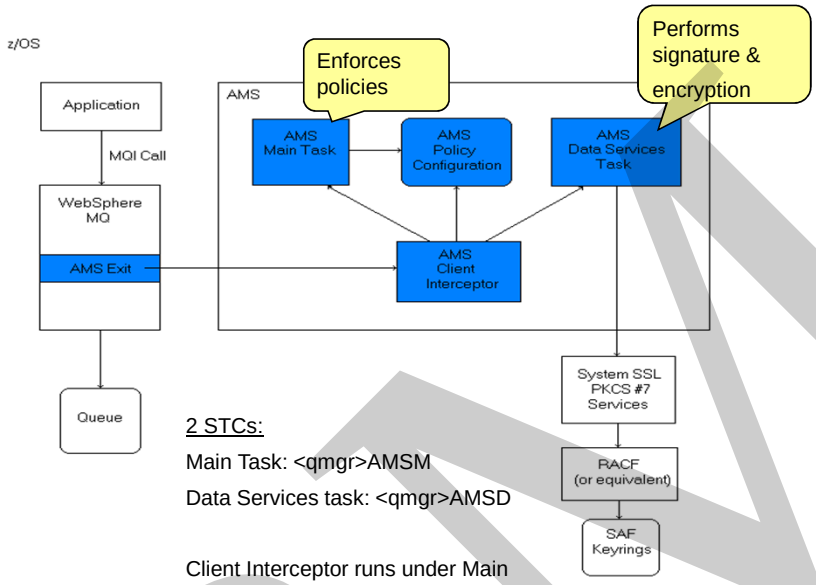
Environments supported

- MQ AMS functionality is implemented in “interceptors”
  - There are no long running processes or daemons (except in z/OS)
  - Existing MQ applications **do not require changes**
- Three interceptors are provided:
  1. **MQ Server interceptor** for local (bindings mode) MQI API and Java applications.
     - Implemented as standard QM API exit on distributed, and “private” API exit on z/OS
     - Requires MQ v6.0.2.8 or 7.0.0.1 as well as GSKit 7.0.4.23 (minimum versions)
     - Note that MQ v7 is required for the AMS MQ Explorer plugin
  2. **MQ Client API interceptor** for remote (client mode) MQ API applications.
     - MQ AMS interceptor imbedded in MQ client code
     - Requires MQ v6.0.2.8 or 7.0.1.1 as well as GSKit 7.0.4.23 (minimum versions)
  3. **MQ Java client interceptor** for remote (client mode) MQ JMS and MQ classes for java applications (JEE and JSE).
     - MQ AMS interceptor imbedded in MQ java client code.
     - Requires MQ Java v7.0.1 as well as IBM Java 1.4..2 (minimum versions)

Logical Architecture Design – Distributed Platforms



Logical Architecture Design – z/OS



2 STCs:

Main Task: <qmgr>AMSM

Data Services task: <qmgr>AMSD

Client Interceptor runs under Main

Message protection policies

- Created or updated or removed by command '`setmqspl`'
  - or by MQ AMS plug-in for MQ Explorer (GUI)
- Policies are stored in queue '`SYSTEM.PROTECTION.POLICY.QUEUE`'
- Each protected queue can have only one policy
- For distributed queuing, protect the queue locally (source QM) as well as the remotely (target QM)
- Two types of policies:
  - Message Integrity policy
  - Message Privacy policy
- Display policies with command '`dspmqspl`'
- "Compromised messages" in queue '`SYSTEM.PROTECTION.ERROR.QUEUE`'
- Extra queue on z/OS '`SYSTEM.PROTECTION.SYNC.QUEUE`'

Message integrity policy definition

- There are two message signing algorithms: SHA1 and MD5
- The list of authorized signers is optional
  - If no authorized signers are specified then any application can sign messages.
  - If authorized signers are specified then only messages signed by these applications can be retrieved.
  - Messages from other signers are sent to the error queue
- On z/OS, same setmqspl program and parms used as SYSIN DD for PGM=DRQUTIL

Syntax:

```
setmqspl
-m <queue_manager>
-p <protected_queue_name>
-s <SHA1 | MD5>
-a <Authorized signer DN1>
-a <Authorized signer DN2>
    :
```

Example:

```
setmqspl -m MYQM
-p MY.Q.INTEGRITY
-s SHA1
-e NONE
-a "CN=cfarkas,O=ibm,C=FR"
```


Message privacy policy definition

- Encryption algorithms: RC2, DES, 3DES, AES128* and AES256*.
- Message privacy requires that encrypted messages are also signed.
- The list of authorized signers is optional.
- It is mandatory to specify at least one message recipient
- Messages retrieved by un-authorized recipients cause messages to be sent to the SYSTEM.PROTECTION. ERROR.QUEUE.

– * Note: z/OS System SSL does not support AES algorithms today.

Syntax:

```
setmqspl
-m <queue_manager>
-p <protected_queue_name>
-s <SHA1 | MD5>
-e <encryption algorithm>
-a <Authorized signer DN1>
-a <Authorized signer DN2>
-r < Message recipient DN1>
-r < Message recipient DN2>
```

Example:

```
setmqspl -m MYQM
-p MY.Q.PRIVACY
-s SHA1
-e AES128
-a "CN=carl,O=ibm,C=US"
-r "CN=ginger,O=catunion,C=JP"
-r "CN=saadb,OU=WBI,O=IBM,C=FR"
```

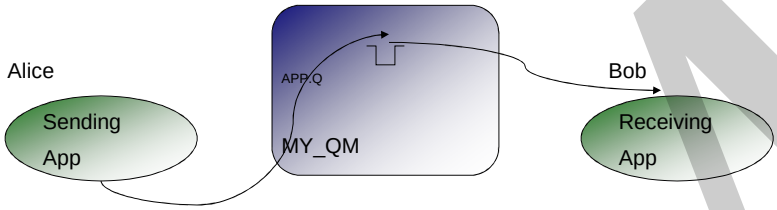
Keystores and X.509 certificates

- **Each MQ application** producing or consuming protected messages **requires access to a keystore** that contains a personal X.509 (v2/v3) certificate and the associated private key.
- The keystore and certificate is accessed by the MQ AMS interceptors.
- The keystore must contain trusted certificates to validate message signers or to obtain the public keys of encrypted message recipients
- Keystore can be the same as that used for MQ SSL
- Several types of keystore are supported (Distributed): CMS, JKS and JCEKS.
- On Distributed MQ, the IBM Key Management (iKeyman, part of GSKit) is provided to create and do simple management of local keystores
- On z/OS, standard SAF product (eg. RACF) used to create certificates which are SAF-managed and must be on a keyring named "`drq.ams.keyring`"
- 3rd party software is available to provide more robust, industrialisation of keystore maintenance.

MQ AMS configuration file (distributed AMS)

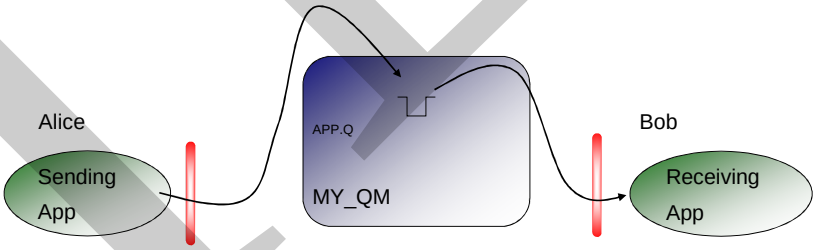
- MQ AMS interceptors require a configuration file, eg. `KEYSTORE.CONF`, which contains:
  - Type of keystore: CMS, JKS, JCEKS
  - Location of the keystore.
  - Label of the personal certificate.
  - Passwords to access keystore and private keys (or `.sth` stash for CMS format)
- Interceptors locate the configuration file using one of the following methods:
  - Environment variable MQS_KEYSTORE_CONF=<path to conf file>.
  - Checking default locations and file names.
    - Platform dependent. For example in UNIX: "$HOME/.mqs/keystore.conf"

WebSphere MQ AMS – install/config example, 0



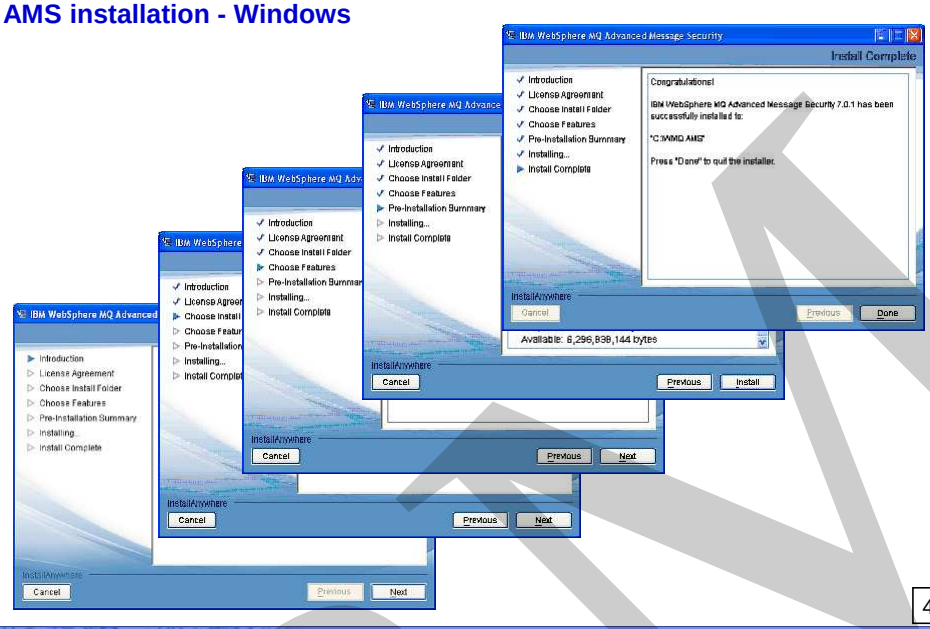
- For a good step-by-step guide, see the AMS InfoCenter at
- http://publib.boulder.ibm.com/infocenter/mqams/v7r0m1/index.jsp
- and search for "Quick start"

WebSphere MQ AMS – install-run example, step 1



1.Install AMS Interceptor

AMS installation - Windows

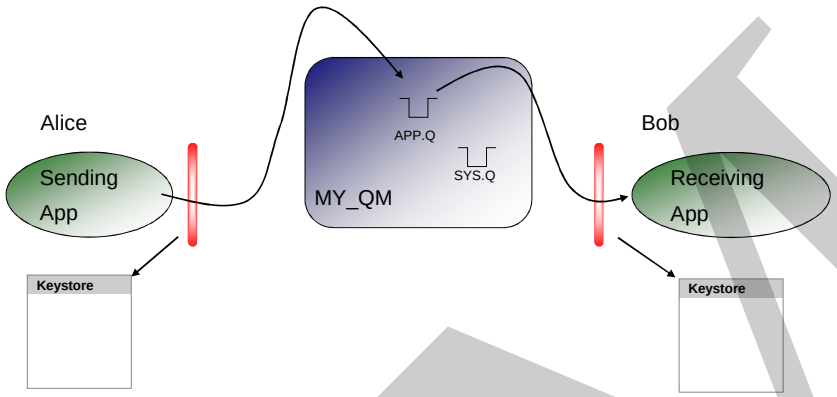


4

AMS installation - z/OS

- SMP/E installation
  - Program 5655-W50, FMID HMS7701
  - Requires 120 tracks (Target), 150 tracks (Distribution), 1MB (zFS)
- Post-installation tasks
  - Update LPA for AMS modules in SDRQLINK
  - Update Authorized Program Facility (APF) list for SDRQLOAD
  - Update Program Properties Table (PPT) update
  - Update Product Registration list (IFAPRDxx)
  - Possible update to DIAG member for allocating in user storage key
- Create AMSM & AMSD procedures for the two AMS STCs
- Create profiles for STCs
  - Set up the userid(s) for the STCs, give SAF permissions,
- Note: userids that will be putting & getting protected messages will require:
  - An OMVS segment associated with their userid (or set default with FACILITY class, BPX.DEFAULT.USER)
  - SAF UPDATE permission for the FACILITY class, IRR.DIGTCERT.LISTRING

WebSphere MQ AMS – install-run example, step 2



1. Install AMS Interceptor
2. Enable AMS

AMS enable - distributed AMS

1. Enable AMS system queues

```
runmqsc MY_QM < "C:\WMQ AMS\bin\defineqs.mqs“
DEFINE QLOCAL(SYSTEM.PROTECTION.POLICY.QUEUE)
MAXDEPTH(999999999)       MAXMSGL(4194304) DEFSOPT(SHARED)
SHARE DEFPSIST(YES)
DEFINE QLOCAL(SYSTEM.PROTECTION.ERROR.QUEUE)
MAXDEPTH(999999999)       MAXMSGL(4194304) DEFSOPT(SHARED)
SHARE DEFPSIST(YES)
All valid MQSC commands were processed.
```

2. Activate AMS interceptors

```
cfgmqs -enable -server MY_QM
DRQDT3052I   The IBM WebSphere MQ Advanced Message Security
  server interceptor has been enabled successfully
```

3. Set up Environment variable to point to AMS key database configuration

```
MQS_KEYSTORE_CONF=C:\AMSStuff\Carl\keystore.conf
```

4. Create the AMS key database configuration file, eg.

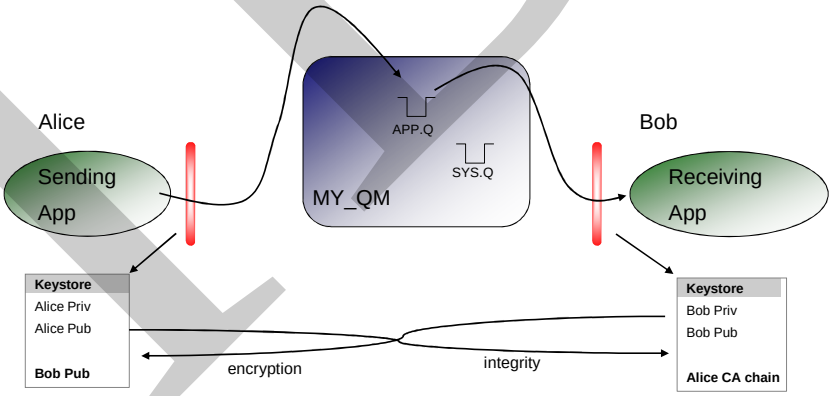
```
C:\AMSStuff\Carl\keystore.conf
cms.keystore=C:/AMSStuff/Carl/carlkey
cms.certificate=Carl_Cert
```

AMS enable - z/OS

```
//CFAMSQM JOB 'Make MQ AMS queues',CLASS=A,MSGLEVEL=(1,1),
// NOTIFY=&SYSUID
/*JOBPARM SYSAFF=ZT01
//*****************************************************************
//*    Define MQ Advanced Message Service (AMS) system queues     *
//*****************************************************************
//STEP1    EXEC PGM=CSQUTIL,PARM='QZ0X',REGION=1M
//* STEPLIB  DD  DSN=WMQ.V7R0M1.SCSQAUTH,DISP=SHR
//*          DD  DSN=WMQ.V7R0M1.SCSQANLE,DISP=SHR
//OUTDEF   DD  DUMMY
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
COMMAND DDNAME(CMDDEF) FAILURE(CONTINUE)
/*
//CMDDEF   DD  DSN=WMQ.AMS.V7R1.SDRQSAMP(DRQQDEFS),DISP=SHR
//
```

AMS supplies the MQ object definitions

WebSphere MQ AMS – install-run example, step 3

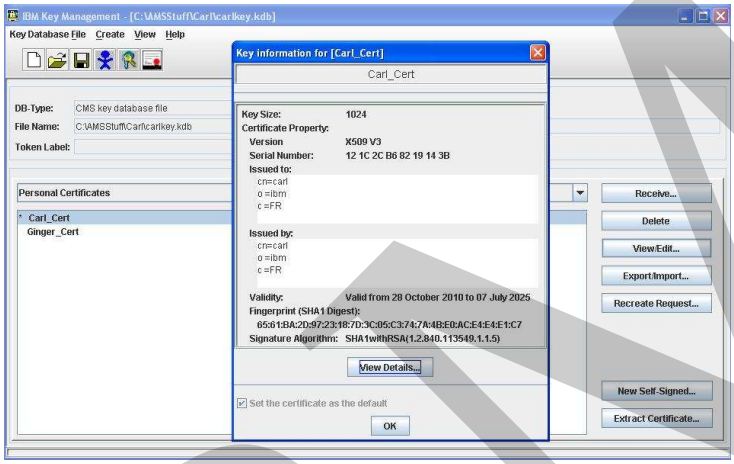


1. Install AMS Interceptor
2. Enable AMS
3. Setup keystores with public / private key pairs
   a) copy sender's CA key chain to receiver's keystore for integrity
   b) copy receiver's public key to sender's keystore for encryption

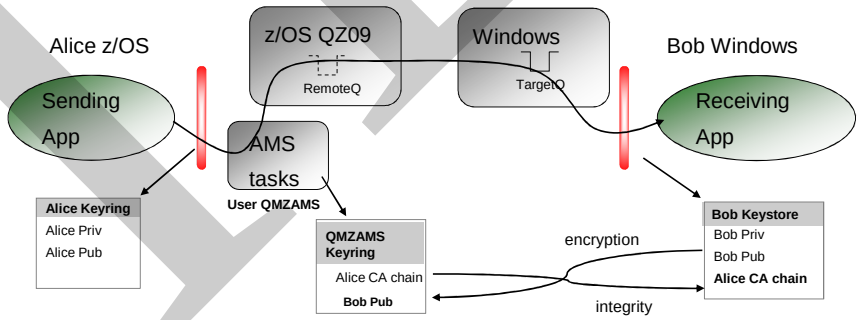
2

Certificate management

- WebSphere MQ supplies iKeyman with GSKit
- Line-mode commands also available (eg. gsk7capicmd)
- On z/OS, RACF commands perform certificate management



WebSphere MQ AMS – Setup z/OS -> Windows



1. Install AMS Interceptor
2. Enable AMS
3. Setup keystores/keyrings with public / private key pairs
   a) copy sender's CA key chain to receiver's keystore for integrity
   b) copy receiver's public key to sender's keyring for encryption

2

RACF on z/OS (sender side)

```
//SYSTSIN  DD *

  RACDCERT ID(QMZAMS) ADDRING(drq.ams.keyring)                    Integrity

/* Create a CA certificate */
RACDCERT CERTAUTH GENCERT -
  SUBJECTSDN( CN('AMS CertAuth') O('IBM') ) WITHLABEL('AMSCA') -
  KEYUSAGE(CERTSIGN) TRUST NOTAFTER(DATE(2018/12/31) )

/* Connect the CA certificate to the AMSD Data task STC */
RACDCERT ID(QMZAMS) CONNECT(CERTAUTH LABEL('AMSCA') -
   RING(drq.ams.keyring)

 RACDCERT EXPORT( LABEL('AMSCA') ) CERTAUTH FORMAT(CERTB64) -
    DSN('ALICE.AMSCA.CERT')

/* Make a keyring for a userid that will be a MQPUTer or MQGETer */
RACDCERT ID(ALICE) ADDRING(drq.ams.keyring)

/* Create a Certificate for a MQPUTer or MQGETer id */
RACDCERT ID(ALICE) GENCERT -
 SUBJECTSDN( c('FR') O('IBM France') CN('Alice on Z') ) -
 WITHLABEL('ALICEONZ') SIGNWITH(CERTAUTH LABEL('AMSCA') )  -
 NOTAFTER(DATE(2018/12/31) ) -
 KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

  RACDCERT ID(ALICE) CONNECT(ID(ALICE) LABEL('ALICEONZ') -
 RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))

  RACDCERT ID(QMZAMS) ADD('ALICE.BOBPUB.CERT') TRUST -
    WITHLABEL('BOBPUB.CERT')
  RACDCERT ID(QMZAMS) CONNECT( ID(QMZAMS) LABEL('BOBPUB.CERT') -
    RING(drq.ams.keyring) USAGE(SITE) )

 SETROPTS RACLIST(FACILITY) REFRESH                               Privacy
```

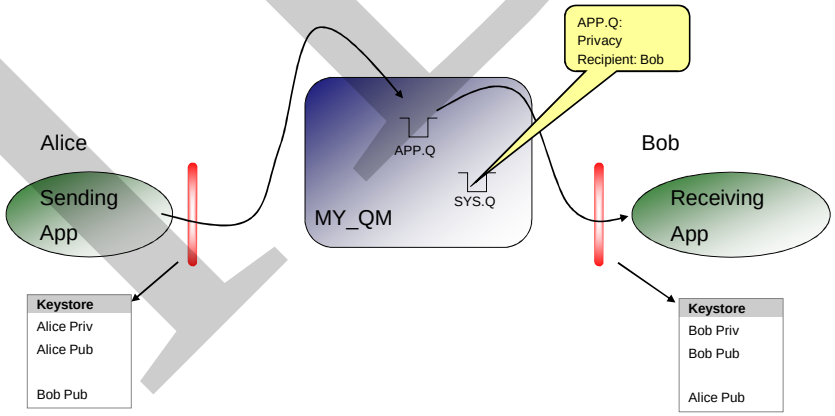
Data task on sender uses CA to validate

Export this CA so others can check my certif

PUTer userid on sender uses certif to sign

Import the Public certif of the Receiver so I can encrypt using this

WebSphere MQ AMS – install-run example, step 4

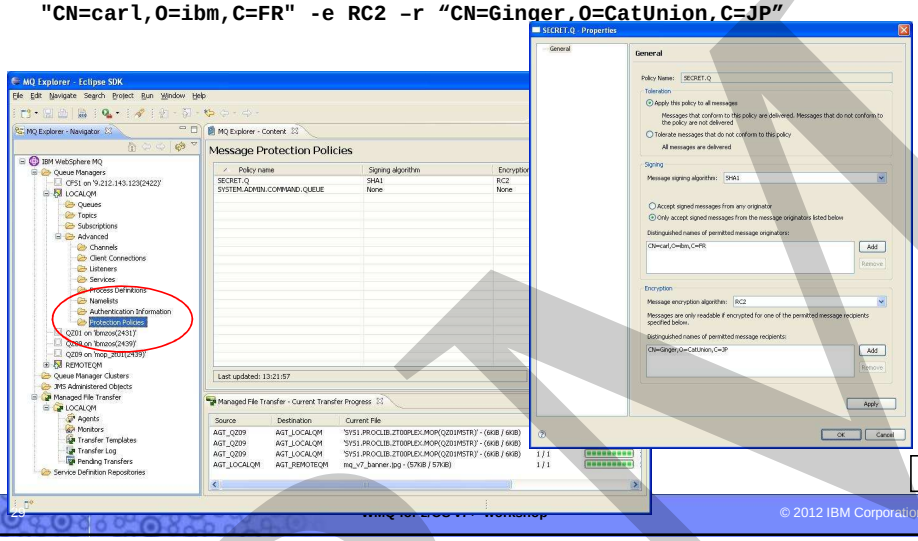


1. Install AMS Interceptor
2. Enable AMS
3. Setup keystores with public / private key pairs
4. Configure protection policy for the queue (setmqspl)

Queue policy definition

- **Use either GUI or line-mode**

```
setmqspl -m LOCALQM -p SECRET.Q -s SHA1 -a
  "CN=carl,O=ibm,C=FR" -e RC2 -r "CN=Ginger,O=CatUnion,C=JP"
```



1

AMS DRQUTIL commands on z/OS

Point to parameters

Execute AMS admin commands

```
//CFAMSAD JOB 'Make MQ AMS queues',CLASS=A,MSGLEVEL=(1,1),
// NOTIFY=&SYSUID
/*JOBPARM SYSAFF=ZT01
//*****************************************************************
//*    Administer MQ Advanced Message Service (AMS)               *
//*****************************************************************
//         SET DIR='/u/farkas'
//         SET FN='drqdserv.envars'
//*
//DRQUTIL  EXEC PGM=DRQUTIL,
//         PARM='ENVAR("_CEE_ENVFILE=&DIR./&FN") /'
//STEPLIB  DD DSN=WMQ.AMS.V7R1.SDRQLOAD,DISP=SHR
//         DD DSN=WMQ.V7R0M1.SCSQANLE,DISP=SHR
//         DD DSN=WMQ.V7R0M1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
 setmqspl -m QZ09
   -p TO.SECRET.FROMZ
   -s SHA1
   -e RC2 -r "CN=carl,O=ibm,C=FR"
/*
//
```

drqdserv.envars

```
_DRQSERV_QMGR=QZ09
_DRQSERV_MSG_LOGGING=stderr_logging
_DRQSERV_MSG_LEVEL=*.i
_DRQSERV_MSG_FOLDING=no
_DRQ_INIT_THREADS=20
_DRQ_MAX_THREADS=100
NLSPATH=/usr/lpp/mqmese/V7R0M1/lib/nls/msg/%L/…£LANG=En_US.IBM-1047
TZ=EST5EDT
```

Error handling

- AMS returns a Rc=2063 if the application tries to access (MQGET) a message for which it is not authorized

```
c:\result>amqsgbr SECRET.Q LOCALQM
Sample AMQSGBR0 (browse) start
LOCALQM
MQGET ended with reason code 2063
Sample AMQSGBR0 (browse) end
```

- The event is also logged in the <AMS installation>\log*.log file

- For destructive MQGET requests, the message is also transferred to the `SYSTEM.PROTECTION.ERROR.QUEUE.` The original message remains there with a DLQ header for administrative handling.

Encrypted message examples

Encrypted message (via Q alias)

```
c:\result>amqsbcg SECRET.Q.ALIAS LOCALQM

AMQSBCG0 - starts here
**********************

 MQOPEN - 'SECRET.Q.ALIAS'


 MQGET of message number 1
****Message descriptor****

  StrucId  : 'MD  '  Version : 2
  Report   : 0  MsgType : 8
  Expiry   : -1  Feedback : 0
  Encoding : 546  CodedCharSetId : 437
  Format : '        '
  Priority : 0  Persistence : 0
  MsgId : X'414D51204C4F43414C514D2020202020D403CF4C201A0402'
  CorrelId : X'000000000000000000000000000000000000000000000000'
  BackoutCount : 0
  ReplyToQ       : '                                                '
  ReplyToQMgr    : 'LOCALQM                                         '
  ** Identity Context
    :
****   Message      ****

 length - 1242 bytes

00000000:  5044 4D51 0200 0200 6800 0000 6800 0000 'PDMQ....h...h...'
00000010:  0800 0000 B501 0000 1100 0000 0000 0000 '................'
00000020:  4D51 5354 5220 2020 0000 0000 0000 0000 'MQSTR   ........'
00000030:  0000 0000 0000 0000 0000 0000 0000 0000 '................'
00000040:  0000 0000 0000 0000 0000 0000 0000 0000 '................'
00000050:  0000 0000 0000 0000 0000 0000 0000 0000 '................'
00000060:  0000 0000 0000 0000 3082 046E 0609 2A86 '........0é.n.   '
00000070:  4886 F70D 0107 03A0 8204 5F30 8204 5B02 'Hå.....áé._0é.[.'
00000080:  0100 3181 D730 81D4 0201 0030 3D30 3131 '..1ü.0ü....0=011'
00000090:  0B30 0906 0355 0406 1302 4A50 3111 300F '.0   ..U....JP1.0.'
000000A0:  0603 5504 0A13 0843 6174 556E 696F 6E31 '..U....CatUnion1'
000000B0:  0F30 0D06 0355 0403 1306 4769 6E67 6572 '.0...U....Ginger'
000000C0:  0208 C1C7 B970 3999 57D6 300D 0609 2A86 '.....p9ÖW.0..   '
000000D0:  4886 F70D 0101 0105 0004 8180 649E 822A 'Hå........üÇd.é*'
000000E0:  C090 A27B 16BE E9BD 916C 8F50 C239 5B9E '.Éó{...ælÅP.9[.'
000000F0:  5C87 4F22 2A9F 0839 6B9D C11C 27B9 53D3 '\çO"*ƒ.9k¥..'.S.'
00000100:  2AC3 C929 B5D8 FB71 4D2B 8F39 A8B2 381D '*..)...qM+Å9¿.8.'
00000110:  31C8 C29D 7608 0891 D6B8 744B 8012 A9DF '1..¥v....tK....'
```

DLQ of encrypted message (via Q alias)

```
c:\result>amqsbcg SYSTEM.PROTECTION.ERROR.QUEUE LOCALQM

AMQSBCG0 - starts here
**********************

 MQOPEN - 'SYSTEM.PROTECTION.ERROR.QUEUE'


 MQGET of message number 1
****Message descriptor****

  StrucId  : 'MD  '  Version : 2
  Report   : 0  MsgType : 8
  Expiry   : -1  Feedback : 0
  Encoding : 546  CodedCharSetId : 437
   :
****   Message      ****

 length - 1398 bytes

00000000:  444C 4820 0100 0000 0F08 0000 5345 4352 'DLH ........SECR'
00000010:  4554 2E51 0000 0000 0000 0000 0000 0000 'ET.Q............'
00000020:  0000 0000 0000 0000 0000 0000 0000 0000 '................'
00000030:  0000 0000 0000 0000 0000 0000 4C4F 4341 '............LOCA'
00000040:  4C51 4D20 2020 2020 2020 2020 2020 2020 'LQM             '
00000050:  2020 2020 2020 2020 2020 2020 2020 2020 '                '
00000060:  2020 2020 2020 2020 2020 2020 2202 0000 '            "...'
00000070:  B501 0000 2020 2020 2020 2020 0B00 0000 '....        ....'
00000080:  433A 5C57 4D51 5637 5C62 696E 5C61 6D71 'C:\WMQV7\bin\amq'
00000090:  7370 7574 2E65 7865 2020 2020 3230 3130 'sput.exe    2010'
000000A0:  3131 3135 3136 3236 3533 3530 5044 4D51 '111516265350PDMQ'
000000B0:  0200 0200 6800 0000 6800 0000 0800 0000 '....h...h.......'
000000C0:  B501 0000 0600 0000 0000 0000 4D51 5354 '............MQST'
000000D0:  5220 2020 0000 0000 0000 0000 0000 0000 'R   ............'
000000E0:  0000 0000 0000 0000 0000 0000 0000 0000 '................'
000000F0:  0000 0000 0000 0000 0000 0000 0000 0000 '................'
00000100:  0000 0000 0000 0000 0000 0000 0000 0000 '................'
00000110:  0000 0000 3082 045E 0609 2A86 4886 F70D '....0é.^.   *åHå..'
00000120:  0107 03A0 8204 4F30 8204 4B02 0100 3181 '...áé.O0é.K...1ü'
00000130:  D730 81D4 0201 0030 3D30 3131 0B30 0906 '.0ü....0=011.0
00000140:  0355 0406 1302 4A50 3111 300F 0603 5504 '.U....JP1.0...U.'
00000150:  0A13 0843 6174 556E 696F 6E31 0F30 0D06 '...CatUnion1.0..'
00000160:  0355 0403 1306 4769 6E67 6572 0208 C1C7 '.U....Ginger....'
00000170:  B970 3999 57D6 300D 0609 2A86 4886 F70D '.p9ÖW.0..   *åHå..'
```

Known limitations today

- Pub/Sub is not supported today
- Channel data conversion is not supported
- Distribution lists are not supported
- IMS Bridge not supported (nor IMS programs in SRB mode)
- Non-threaded applications using API exit on HP-UX
- Java (JMS and Java “base” classes) only supported with MQv7
- AMS does not yet support MQ message properties (introduced in MQv7) on z/OS

- Note that AMS increases message length
  - New Message Size = 1280 + [old msg length] + (200 x [# of recipients])
- AMS usage will increase CPU requirements

Summary

WebSphere MQ Advanced Message Security V7.0.1

- Protects message integrity and/or privacy
- Supports all current versions of MQ
- Supports MQ Server, MQ Client and JMS
- “Light weight” product - No pre-requisites, easy installation, easy configuration
- Existing MQ applications do not require changes

Bibliography

- WMQ AMS InfoCenter at http://publib.boulder.ibm.com/infocenter/mqams/v7r0m1/index.jsp
- GI13-0559 Program Directory for IBM WebSphere MQ Advanced Message Security for z/OS
- GC34-7142 WebSphere MQ AMS Administration Guide
- SG24-8069 Secure Messaging Scenarios with WebSphere MQ (draft)

WebSphere MQ for z/OS v7+ Proof-Of-Technology (POT) workshop

WebSphere MQ File Transfer Edition (FTE)

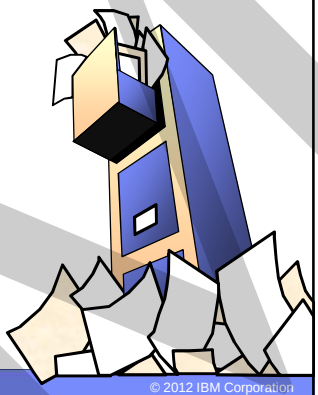
IBM EMEA panIMT Team zWebSphere team
(with thanks to IBM US ATS team for the original “Wildfire” material)

Agenda

- What is WebSphere MQ File Transfer Edition (WMQFTE)?
- Components and Topology
- WMQFTE interface
- MQ usage
- WMQFTE packaging and installation
- Integration

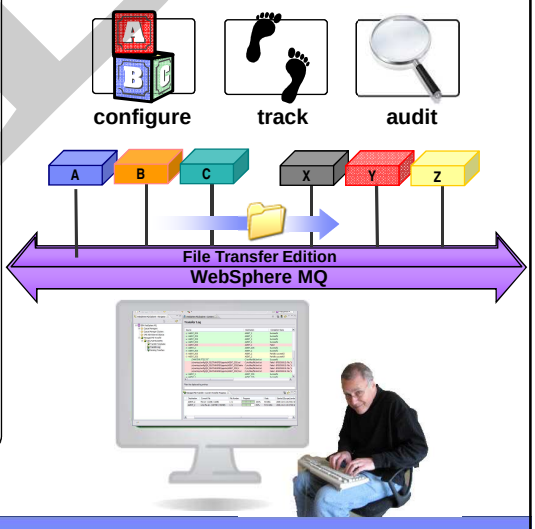
What is WebSphere MQ File Transfer Edition?

- Many applications communicate using files
  - Because they are easy to implement
  - It is the lowest common denominator for application integration
- Traditional ways of moving files often have short-comings:
  - Poor reliability
  - Difficult to manage and monitor
  - Provide no audit capabilities
  - Difficult to integrate with operational batch processes
  - Unsupported or badly supported software
  - Not commonly used or available
- *Managed file transfer* software addresses these problems



Introducing WebSphere MQ/FTE (File Transfer Edition)

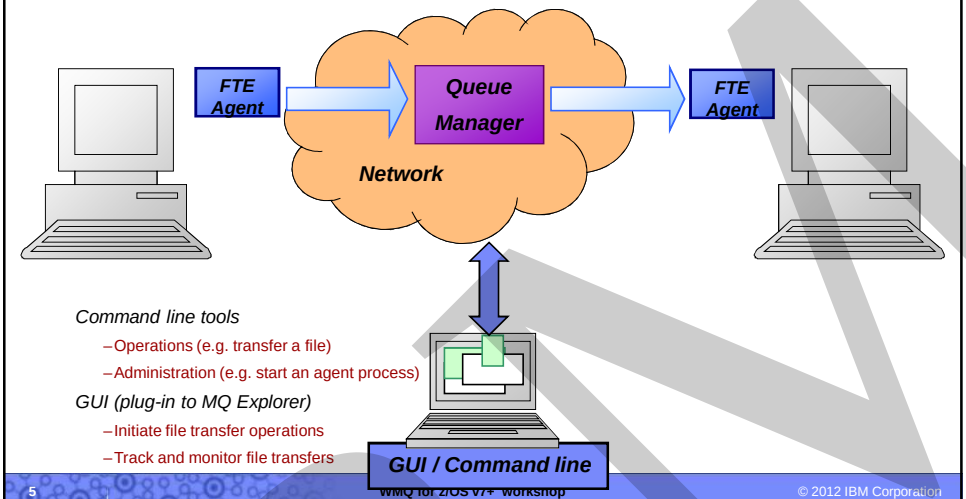
- Flexible backbone for transfers – not a single-hop solution like FTP
- Reliability leveraging the MQ transport
- Centralized monitoring and management
- Auditable with logging subsystem that tracks transfer at source and at destination for audit purposes
- Massive files – larger than MQ messages
- Integration with MQ-enabled apps and ESBs
- Simple graphical tooling enabling remote transfer invocation
- Flexibility with line-mode, API or message-driven invocation
- Automatic file conversion and compression
- Security of file payload using SSL
- Support for major platforms, including z/OS native file support (eg. QSAM, PDS, GDG..)



FTE Components

FTE Agents

- Act as the end-points for file transfers – installed "near" the files
- Run local to the files being transferred at the source and the destination
- Can bind to a local queue manager, or connect using MQ client bindings



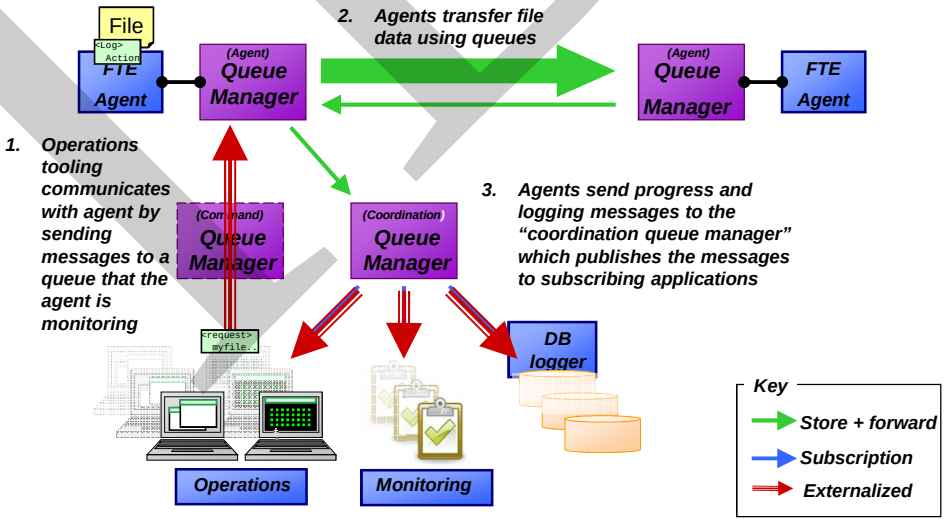
Command line tools

- Operations (e.g. transfer a file)
- Administration (e.g. start an agent process)

GUI (plug-in to MQ Explorer)

- Initiate file transfer operations
- Track and monitor file transfers

WMQ/FTE file transfers



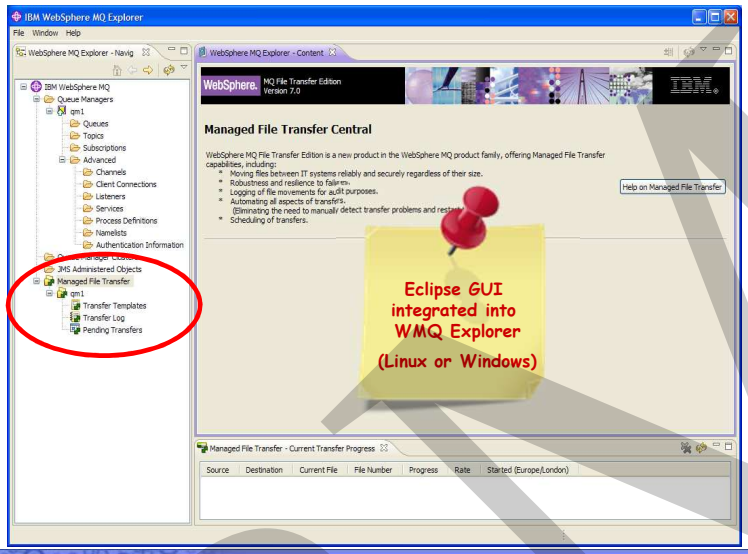
The Queue Manager roles

- *Coordination Queue Manager (1)*
  - A Management and Monitoring Queue Manager
  - It collects information about ...
    - What FTE agents are defined
    - The progress of active transfers
    - Transfer log history "Audit" type information about transfers
  - Must be an MQ v7 Queue Manager to support publish and subscribe
  - Not an FTE "component"! A QM performing an FTE role. There is no FTE code running on the Coordination QM.
  - Never a bottle-neck or single-point-of-failure!
- *Agent Queue Manager (n)*
  - Source or Destination Queue Manager for File Transfers
  - Agents run connected to this Queue Manager
  - In MQ Bindings mode (a direct connection using memory to memory bindings) on the local machine
  - In MQ Client mode (using and MQ Client channel connection) from a remote machine
  - Many agents can share the same queue manager
  - Can be MQ v6 or v7
- *Command Queue Manager (optional, n)*
  - Optional intermediary queue manager that is used for injecting commands
  - Can be MQ v6 or v7

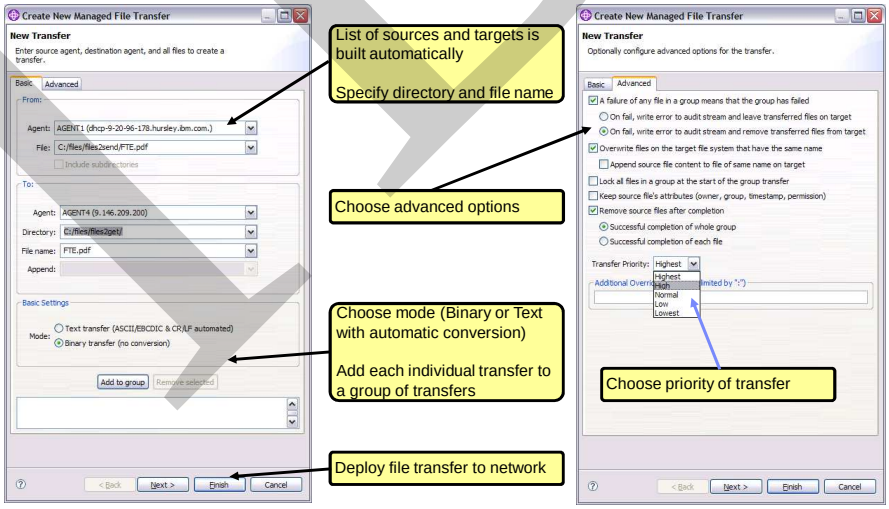
How to start a File Transfer?

- **Through a Graphical User Interface**
  - Eclipse plug-in for WebSphere MQ Explorer
- **Using commands invoked from the supported Operating Systems shell environment**
  - Command line interface
- **Using any native scripting language on the OS that can invoke the IBM WebSphere MQ File Transfer Edition commands**
  - Shell, Bat, CMD etc.
- **Using an ANT script**
- **By putting a properly formatted XML request on the source FTE Agent "command" queue**
- **Automatic submission of a request (Trigger or Monitor)**

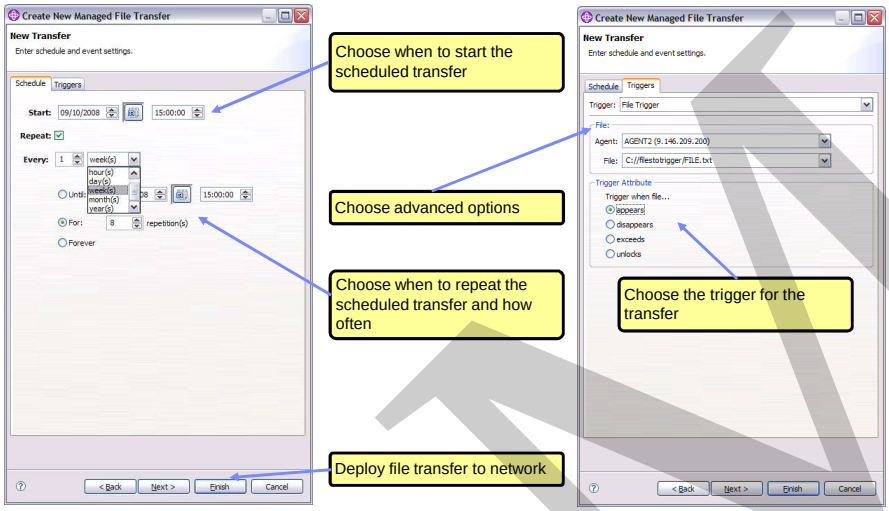
Centralized Configuration via MQ Explorer



Creating File Transfers



Scheduling & Triggering File Transfers

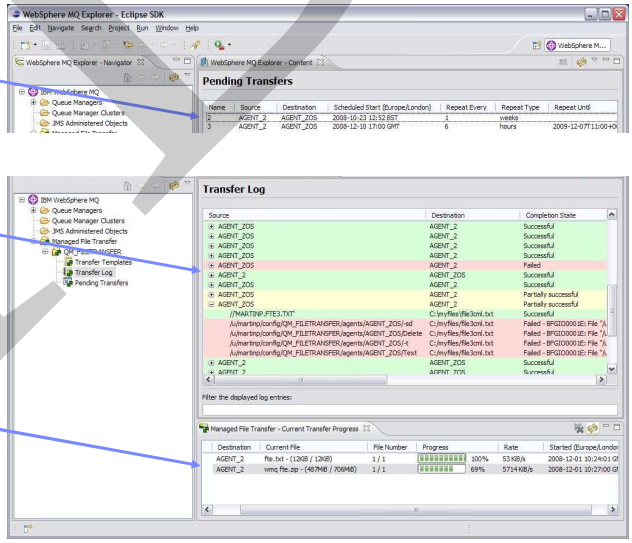


Auditing & Monitoring File Transfers

View or cancel transfers that will run by schedule and/or are triggered

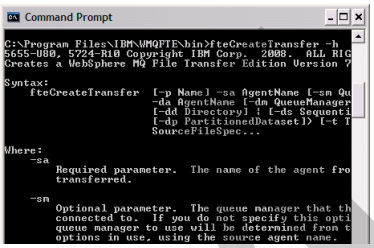
View audit log of all transfers and groups of transfers

View progress of all transfers currently taking place



Configuring File Transfers and Agents from Command Line

- Command Line Interface is consistent across all supported platforms
- Transfer commands can be invoked from the supported Operating Systems shell environment
- Commands can be invoked from anywhere across the file transfer
  - *e.g. Transfers could be invoked from a remote Windows machine for transfers taking place between z/OS and Unix machines*
- Developers can use any native command line language on the OS that can invoke these commands (shell, bat, cmd, etc.)
- Application Programs can place a request using a messaging interface in XML



Examples:

- **fteCreateTransfer** *Starts a new file transfer from the command line*
- **fteStartAgent** *Starts a File Transfer agent from the command line*
- **fteStopAgent** *Stops a File Transfer agent in a controlled way*
- **fteShowAgentDetails** *Displays the details of a particular File Transfer agent*
- **fteListAgents** *Displays all known File Transfer agents*

Line-mode examples

- fteCreateTransfer – request a file transfer

```
C:\Junk> fteCreateTransfer -sm QMCF -sa ZOSFTECARL -t text -de overwrite -da WINFTE -df
"c:\Junk\Out\SYS1.PROCLIB.line.txt" "//'SYS1.PROCLIB(QMCFMSTR)'

5655-U80, 5724-R10 Copyright IBM Corp. 2008. ALL RIGHTS RESERVED
BFGCL0035I: Transfer request issued. The request ID is: 414d512057696e514d202020202020205b90924920003e02
```

- fteCreateAgent – create a new FTE Agent

```
ITSO:/u/carlf: > fteStartAgent ZOSFTECARL
5655-U80, 5724-R10 Copyright IBM Corp. 2008. ALL RIGHTS RESERVED
BFGCL0030I: Request to start agent 'ZOSFTE' on this machine has been submitted.
BFGCL0031I: Agent log files located at: /u/carlf/MQFTE_MyConfig/WinQM/agents/ZOSFTECARL
```

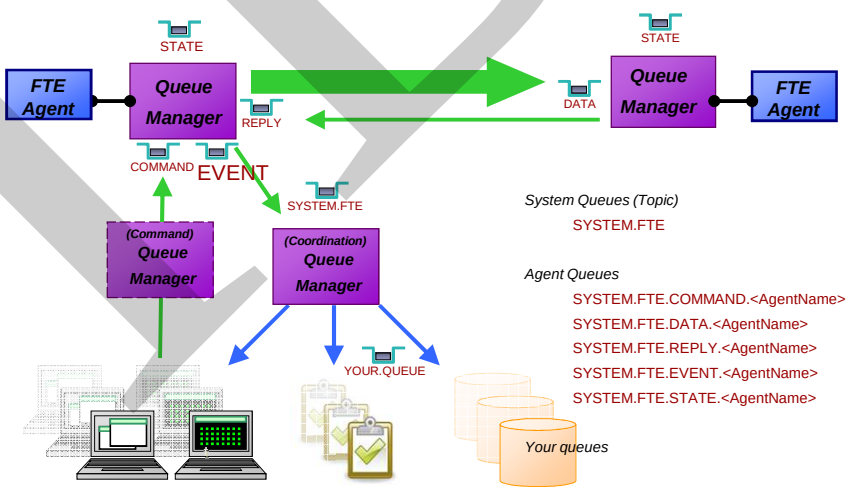
- fteStartAgent – start an FTE Agent

```
ITSO**:/u/carlf: > fteCreateAgent -agentName ZOSFTECARL -agentQMgr QMCF

5655-U80, 5724-R10 Copyright IBM Corp. 2008. ALL RIGHTS RESERVED
BFGCL0128I: The agent requires a number of WebSphere MQ objects to be defined to queue manager QMCF. Ensure these
definitions are present before starting the agent.
BFGCL0071I: Direct the following MQSC definitions for agent 'ZOSFTE' to queue manager 'QMCF'.
  :
```

Invoking File Transfers via JCL on z/OS

```
//IBMUSER JOB …….
//STEP1 EXEC PGM=USERPGM,REGION=4M
//INPUT DD DSN=FILE1.INPUT,DISP=SHR
//OUTPUT DD DSN= PRDB1.SALES.DATA,DISP=OLD
//SYSOUT DD *
//STEP2 EXEC PGM=BPXBATCH,REGION=0M, PARM='PGM /bin/sh
  /usr/mqm/mqfte/bin fteCreateTransfer'
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//STDIN DD *
-sa AGENT1 –sm CSQ6 –da AGENT4 –dm SUNQM1  -df  /tmp/
  prd1sales.dat PRDB1.SALES.DATA
//
//*
```


MQ queues



System Queues (Topic)

SYSTEM.FTE

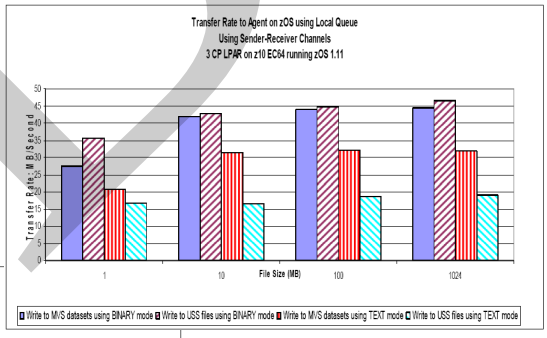
Agent Queues

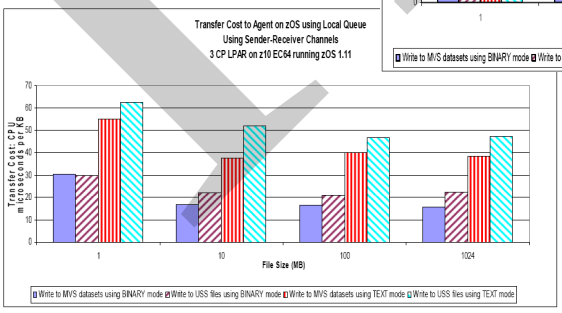
SYSTEM.FTE.COMMAND.<AgentName>
SYSTEM.FTE.DATA.<AgentName>
SYSTEM.FTE.REPLY.<AgentName>
SYSTEM.FTE.EVENT.<AgentName>
SYSTEM.FTE.STATE.<AgentName>

MQ network usage and implications

- *File data is transferred as non-persistent messages*
  - Minimises impact on MQ logs and infrastructure
  - File transfer application protocol compensates for non-delivery
- *Rate of transfer is paced*
  - There is a feedback loop which prevents the source agent out-performing the receiving agent and flooding the MQ infrastructure with data
  - Sending block sizes can be adjusted via parameters
- *Transfers are check-pointed*
  - Agents take periodic check-points of their progress
  - These can be used to restart in case of a failure
  - Persistent message check-point mechanism used
- *Traffic Management*
  - Using priority settings you can manage the importance and speed of messaging
    - For messaging applications
    - For file transfers
  - Default priority for File transfer data messages is 0 – the lowest priority
  - It can be changed via the command line or the GUI

Performance

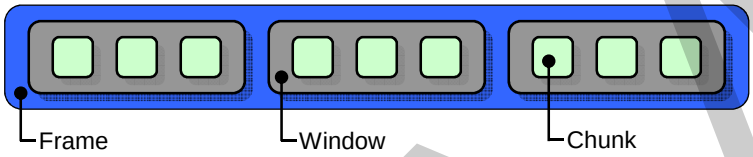




- Example of Linux to z/OS
- For more details, see FP11: WebSphere MQ FTE Performance Report,
http://www.ibm.com/support/docview.wss?rs=171&uid=swg24025452&loc=en_US&cs=utf-8&lang=en

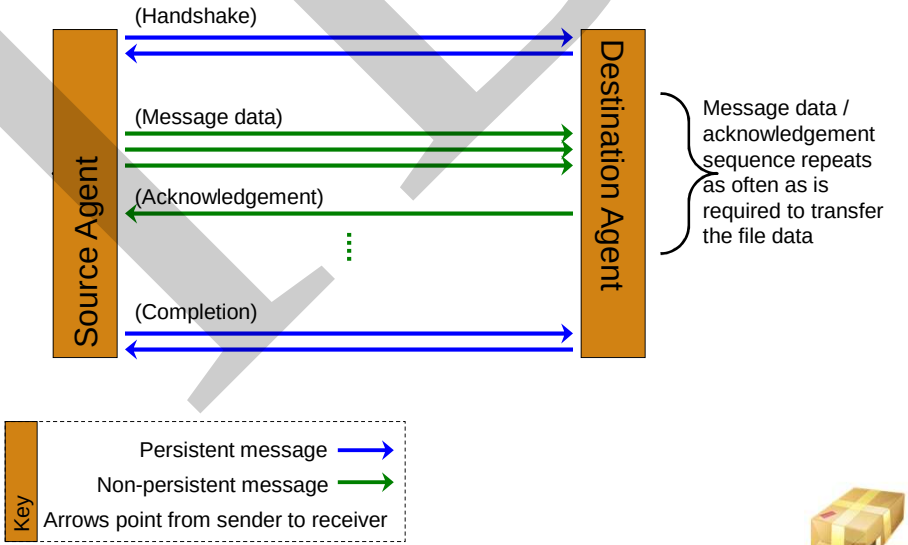
Transfer Protocol

- The frame size determines the maximum amount of data in progress.
- A Frame is split into windows. A frame is moved along the data a window at a time.
  - Normally acknowledgements are only sent back when a window is full not for every chunk
- A window is comprised of a number of chunks sent to the source to destination before an "ack" is sent .
  - Can contain data from one or more files
- A Chunk defines the maximum MQ Message size



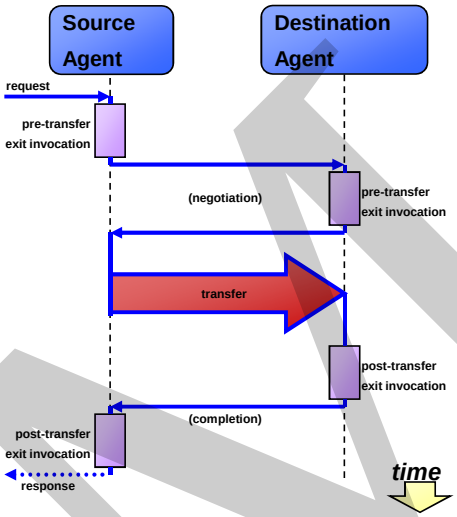
Property Name	Default Value	Description
agentFrameSize	**5**	**Number of windows for the transfer frame**
agentWindowSize	**10**	**Number of chunks for each window**
agentChunkSize	**256kb**	**Size of each transfer chunk, actual MQ message data size**
agentCheckpointInterval	**1**	**Number of completed frames after which a checkpoint. If a transfer fails, recovery occurs at checkpoint boundaries.**

How the Transfer Process Takes Place



User exit routines

- *User supplied Java code can be invoked*
  - Prior to starting a transfer at both the source and destination.
  - After completing a transfer at both the source and destination.
- *Code can:*
  - Cancel the transfer.
  - Change which files are transferred.
  - Read and update transfer meta-data.
- *Any changes are recorded so they can form part of the auditing information that is collected.*



WebSphere MQ File Transfer Edition Versions

- *WebSphere MQ File Transfer Edition Client*
  - Connects to a Queue Manager using MQ Client bindings
  - Typically a remote machine using the network to connect to the Queue Manager
- *WebSphere MQ File Transfer Edition Server*
  - Allows MQ/FTE Agents to connect in bindings mode to the local Queue Manger
  - Allows MQ/FTE Clients to connect via Client bindings over the Network
- *WebSphere MQ File Transfer Edition for z/OS®*
  - Allows MQ/FTE Agents to connect in bindings mode to MQv6 or MQv7 Queue Managers
- *Remote Tools and Documentation CD*
  - Included with all three components
  - This can be installed on as many computers as the customer likes

WebSphere MQ File Transfer Edition – platforms

- Microsoft Windows XP Pro SP2, Windows 2000 Server or Professional, Vista (Business Edition), Windows v7 (Pro or Ultimate)
- Microsoft Windows Server 2003 with SP2 or later (32-bit or 64-bit)
- Microsoft Windows Server 2008 (32-bit or 64-bit, Standard or Enterprise Editions)
- , 5.3 plus TL04 or AIX 6.1 and appropriate firmware,
- HP-UX 11i V2 or v3 on Itanium,
- HP-UX V11.11 on PA-RISC
- Sun Solaris , 10 on SPARC
- I5/OS, OS/400, System I v5.4 or v6.1
- z/OS v1.6 or higher

– O/S in grey are FTE Client only

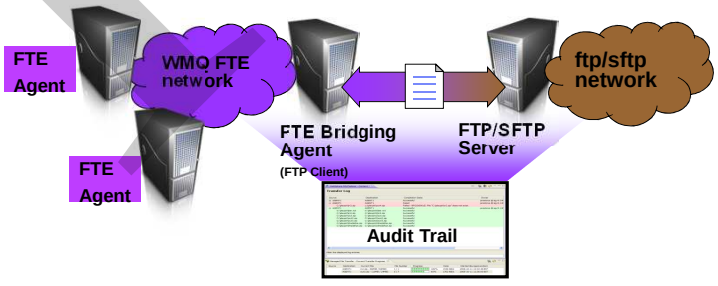
- Linux for System x:
  - Red Hat Enterprise Linux (RHEL) , v5.0 thru 5.4 (32-bit),
  - Red Hat Enterprise (RHEL) , 5.3 thru v5.4 (64-bit)
  - SUSE Linux Enterprise Server (SLES) V9 SP4 thru V11 (32-bit and 64-bit),
- Linux for System p
  - Red Hat Enterprise (RHEL) v5.0.2 thru v5.4 (64-bit)
  - SUSE Linux Enterprise Server (SLES) V9 SP4 thru V11 (64-bit),
- Linux for System z
  - Red Hat Enterprise (RHEL) v5.0.2 thru v5.4 (64-bit)
  - SUSE Linux Enterprise Server (SLES) V10 thru V11 (64-bit),

For current supported platform information, see:
http://www.ibm.com/software/integration/wmq/filetransfer/requirements/index.html

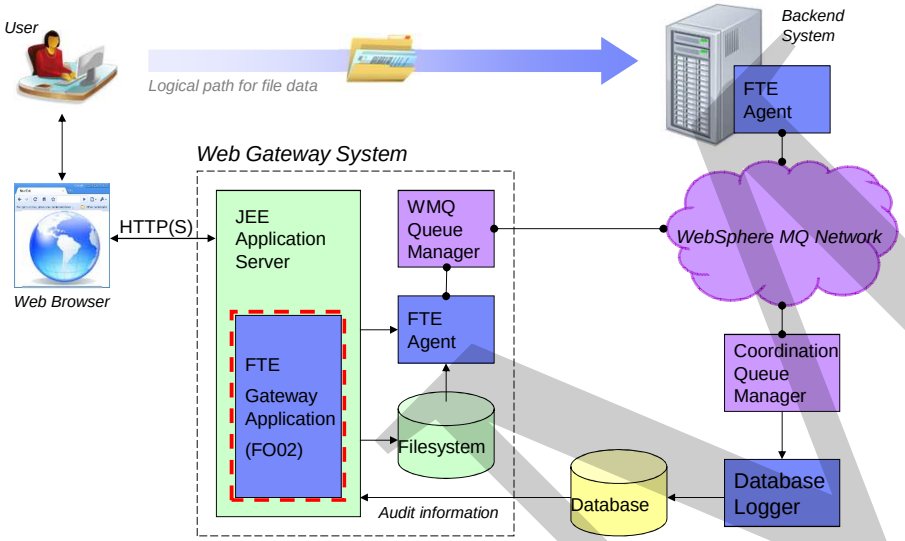
FTE Protocol bridging - FTP/SFTP

- Support for transferring files located on FTP or SFTP servers
  - The source and/or destination for a file transfer can be an FTP or SFTP server
- Enables incremental modernization of FTP-based home-grown solutions
  - Provides auditability of transfers across FTP/SFTP to central audit log
  - Ensures reliability of transfers across FTP/SFTP with checkpoint restart
- Fully integrated into graphical, command line and XML scripting interfaces

Managed file transfers between MQ and FTP/SFTP networks



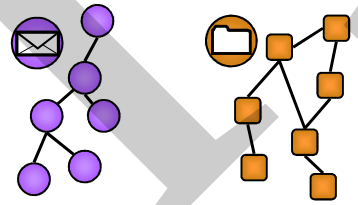
FTE HTTP Gateway support (FO02 SupportPack)



Summary

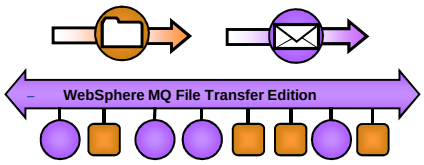
– Simplification means Savings

High cost in building and maintaining dual infrastructures to move business data



Messaging File Transfer

Reduce costs with a single consolidated transport for both files and messages

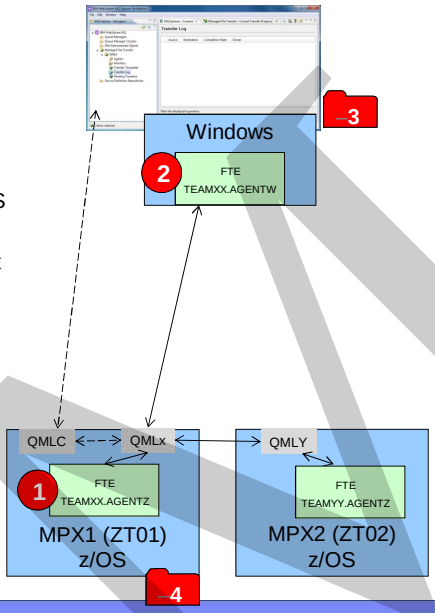


- Re-use what you already have
- Reap operational savings from simplification
- Reduce administration effort – use one tool to manage the whole solution
- Reduce skills requirements and maintenance

– **WebSphere MQ + File Transfer = WebSphere MQ File Transfer Edition**

Lab time

- FTE lab
  1. Configure and start an FTE Agent on z/OS, TEAMXX.AGENTZ
  2. Configure and start an FTE Client Agent on Windows, TEAMXX.AGENTW
  3. Transfer a file from Windoze to your z/OS Agent XX
  4. Transfer a file from your z/OS FTE Agent XX to a different z/OS FTE Agent YY
- Userid: TEAMXX
- Passwords: _____



Backup material

Bibliography

- FTE InfoCenter - http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/index.jsp
- REDP-4532 WebSphere MQ File Transfer Edition Overview (Redpaper)
- SG24-7760 Getting Started with WebSphere MQ File Transfer Edition (Redbook)
- SG24-7128 WebSphere MQ V6 Fundamentals (Redbook)
- REDP-4533 Managed File Transfer for SOA using WebSphere MQ File Transfer Edition (Redpaper)
- Trial version: http://www.ibm.com/software/integration/wmq/filetransfer/

Conditional transfers - scheduling

- At a certain date and time
  - Scheduled events stored by the Agent
  - Event list polled every 30 seconds
- All options available in GUI

- fteCreateTransfer parameters…

```
-ss <yyyy-MM-ddThh:mm> or –ss <hh:mm>        FTR has a deferred start
   -oi <minutes | hours | days | weeks | months | years>   Interval units
   -of <positive integer> Interval frequency (default 1)

   -es <yyyy-MM-ddThh:mm> or –es <hh:mm> Expiration
              or
   -oc <positive integer>     Occurrences count (default unlimited)
```

```
-ss 15:45 –oi hours –of 2 –oc 3
```

Transfer requested starting 15:45, every 2 hours, 3 times (17:45, 19:45)

Conditional transfers - triggering

- Given certain file system environment events (triggering)
  - for example the existence of a file within the file system
  - Or when a file reaches a certain size
- All options available in GUI

- fteCreateTransfer

-tr <condition>,<name list>
<condition> can be

file=exist	**At least one of the files in the name list does exist.**
file!=exist	**At least one of the files in the name list does not exist.**
filesize>= <size>	**At least one of the files in the name list is greater or equal to <size>**

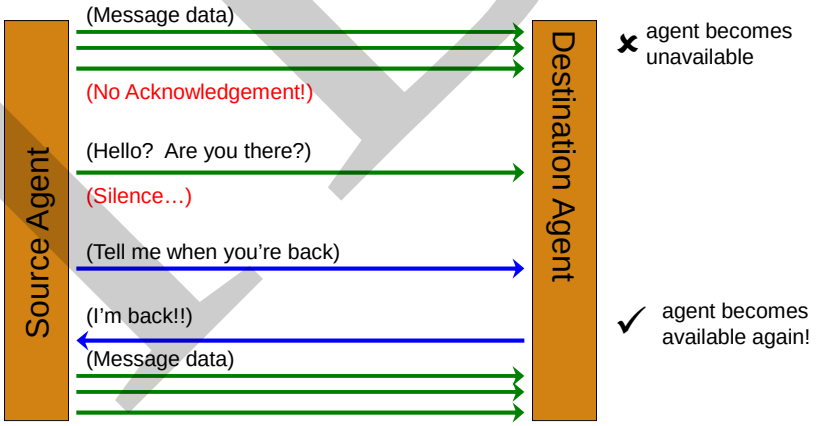
<size> is an positive integer with an optional units in KB, MB, GB. No units implies bytes.
<namelist> a comma separated list of filenames located on the source agent's machine.
The –tr option can appear more than once in the same fteCreateTransfer, but all triggers conditions must evaluate to true for the FTR to be processed.

-tl <yes|no>
defines whether a log message is generated each time the trigger condition fails, giving the first trigger that failed. The default is yes.

```
-tr file=exist,C:\Junk\myTrigger.on
```

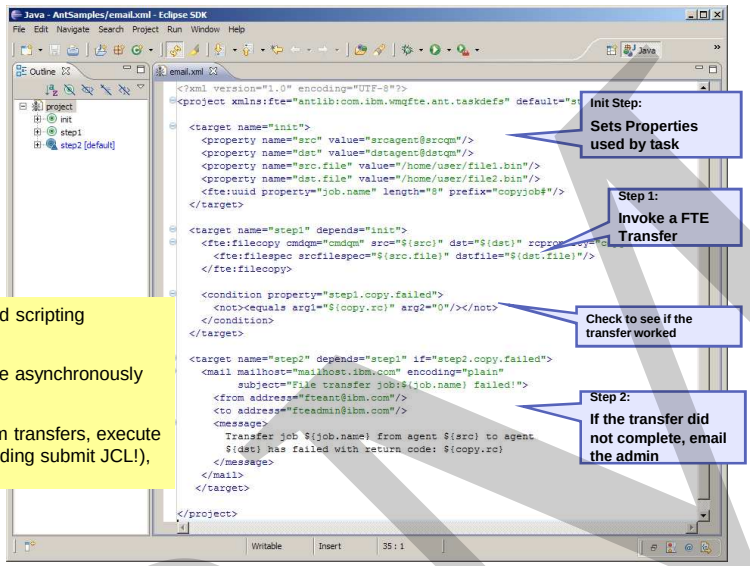
Transfer of (any) file requested if c:\Junk\myTrigger.on is found

Interrupts to the Transfer Process



- *Data is never lost!*
  - Original source file is only (optionally) deleted when placed at target
  - Checkpoint's occur to minimize amount of data retransmitted in case of interruption

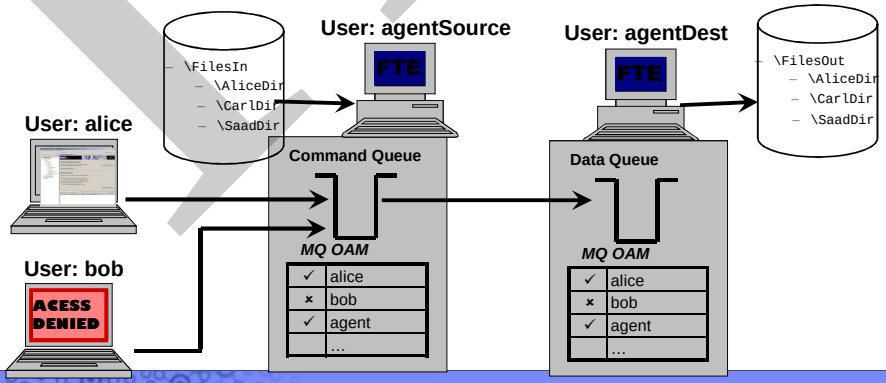
Built-in scripting support with ANT



- Standards-based scripting language
- Ability to execute asynchronously or synchronously
- Ability to perform transfers, execute commands (including submit JCL!), etc.

FTE Security

- MQ security checks (eg. RACF, SSL) are in place to protect channel and queue accesses
  - Can Alice or Bob start a channel to agentSource's QM? Can they access the Command queue?
  - Can agentSource's QM start a channel to agentDest's QM and send messages?
- The Operating System (eg. RACF) protects file access
  - Can agentSource id access \FilesIn directory? Can agentDest access \FilesOut directory?
- Finally (but often most importantly!) FTE can be configured to verify userid (or groupid)
  - Access to the directories (on either side)
  - Ability to execute commands such as file transfers, schedule transfers, monitor, etc.



Minimum supported levels for FTE 7.0.3

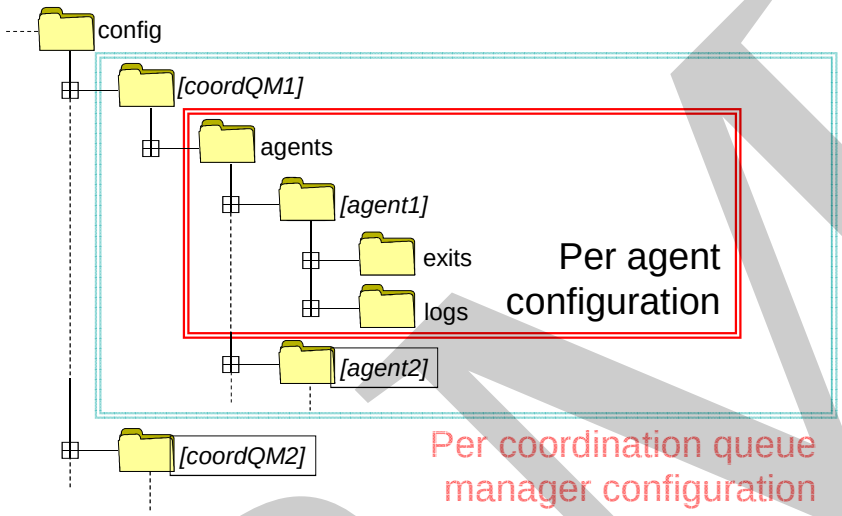
	To act as Coordination queue manager	To act as Agent or Command queue manager only
Distributed	**WebSphere MQ, Version 7.0.0.1 Java 5.**	**WebSphere MQ Version 6.0 with Fix Pack 6.0.2.4, or later. Java 5.**
z/OS	**WebSphere MQ for z/OS, Version 7.0.1.2 with PM03853 & PM00654, Java 5 SR8.**	**WebSphere MQ for z/OS®, Version 6.0.2.10 with fix PM17286, Java 5 SR8.**

For current supported platform information, see:
http://www.ibm.com/software/integration/wmq/filetransfer/requirements/

Basic installation sequence

1. Install binaries (SMP or TAR for z/OS, setup.exe for Windows….)
2. Setup environment, eg.
   - ✓ Verify Java version
   - ✓ Setup MQ
   - ✓ Setup environment (eg. .profile)
3. Install Remote Tools and documentation on a Windows or Linux machine
4. Instantiate runtime components (optionally done with setup on Windows)
   1. fteSetupCoordination
   2. fteSetupCommands
   3. fteCreateAgent
5. Customize MQ from generated scripts
6. Start FTE Agent(s)
   - Check the log
7. Test a file transfer

Configuration Directory Structure



FTE Protocol bridging - B2B

- Documented and tested configurations for integrating with DataPower Appliances
  - WebSphere DataPower XB60 B2B Appliance – for B2B connectivity
  - WebSphere DataPower IX50 Integration Appliance – for ESB connectivity
- Enables sending files to trading partners over a range of protocol transports
  - via DataPower Appliances acting as B2B gateways