



Tendances Logicielles

L'architecture pour répondre aux besoins métier

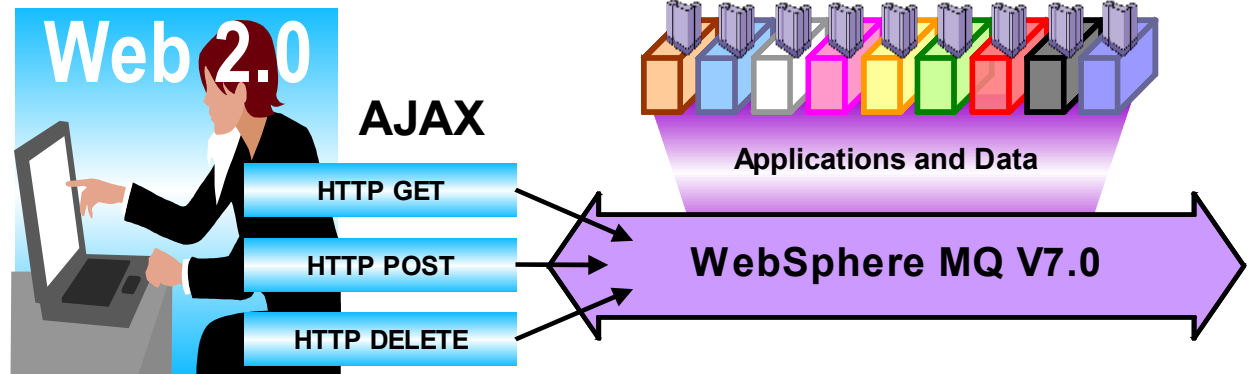
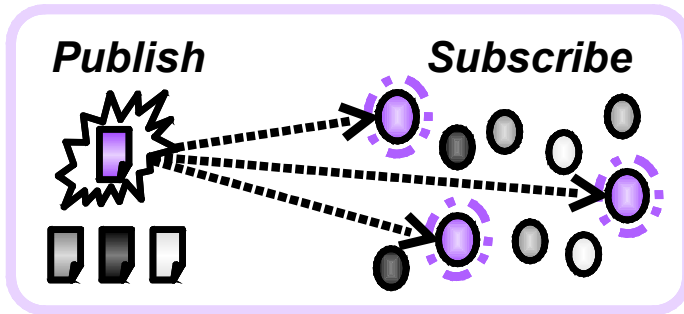
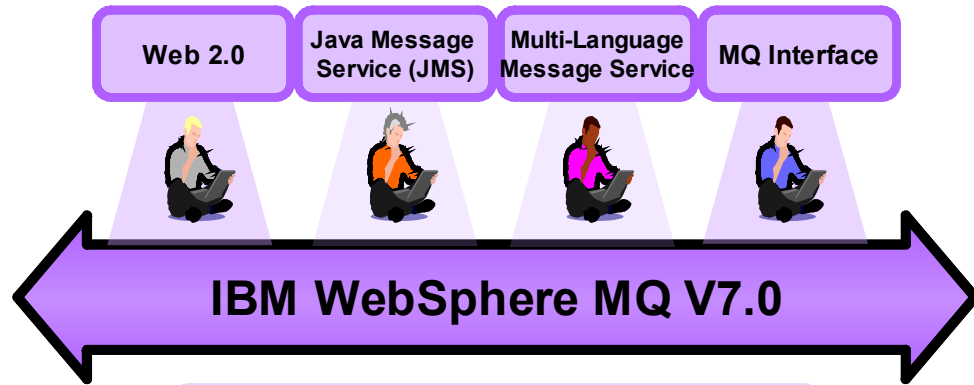
What's New in WebSphere MQ Family : MQ V7, MQFTE

Saad BENACHI
WebSphere TechSales



IBM WebSphere MQ V7.0

*Universal Messaging
Backbone for SOA and Web
2.0*



WebSphere MQ Version 7

- Central requirement was to improve JMS implementation
 - ∞ More applications being written to use this API
 - ∞ Underpins many SOA/ESB solutions needing access to messaging
- Leads to designs involving features such as
 - ∞ Ease-of-use
 - ∞ Performance
- But it also leads to enhancements for ALL applications
 - ∞ Not just JMS users
- Extension of publish/subscribe capabilities
 - ∞ Designed with Message Broker in mind
- Easier programming in any environment
 - ∞ Some features suggested by JMS requirements are useful in MQI
- Administration model and APIs natural evolution of existing interfaces



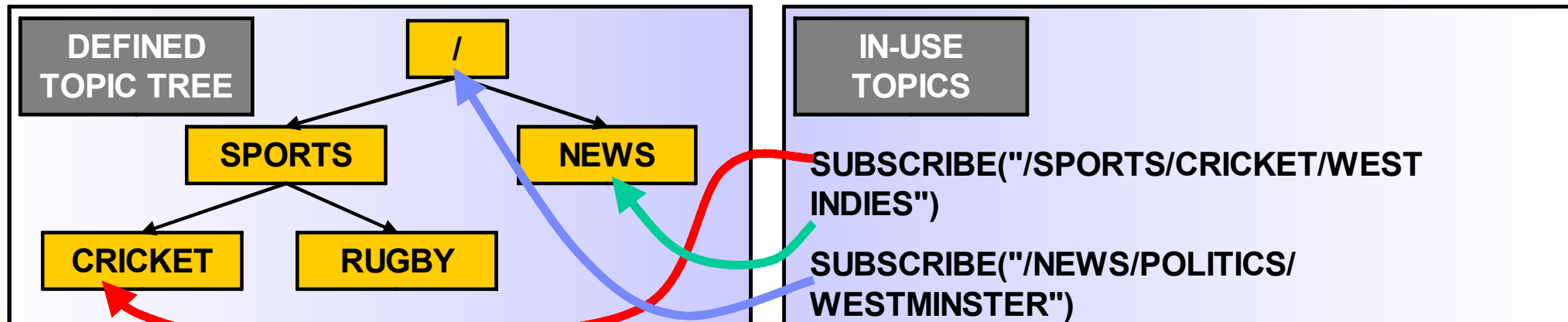
Publish/Subscribe

- A natural part of the JMS API
 - ⌘ Combines both Publish/Subscribe and Point-to-Point patterns
 - ⌘ Now also a natural part of the native MQI
- Point-to-point asynchronous messaging decouples applications
 - ⌘ But still implies a one-one relationship between sender and receiver
- Publish/subscribe is a further stage of decoupling
 - ⌘ Sender has no direct knowledge of how many (if any) apps will see a message
 - ⌘ Link between applications is a **Topic**, not a **Queue**
- WMQ V6 (Distributed) included a Publish/Subscribe broker (formerly MA0C)
 - ⌘ Compatibility mode available in V7
- Implementation substantially improved with V7
 - ⌘ And is available for the first time on z/OS



Publish/Subscribe Administration

- Based on Topic Strings
- Topic Objects
 - ⌘ New object type, like queue or channel definitions
 - ⌘ A 48-character name which has a longer attribute for full **topic string**
 - ⌘ Defines major points in a topic tree
 - ⌘ No additional definitions needed before applications can start using pub/sub
- In-use topics
 - ⌘ The topic strings that applications are publishing or subscribing on
 - ⌘ Inherit attributes (eg security) from the "closest" defined topic object
 - ⌘ Not defined administratively, but can be viewed



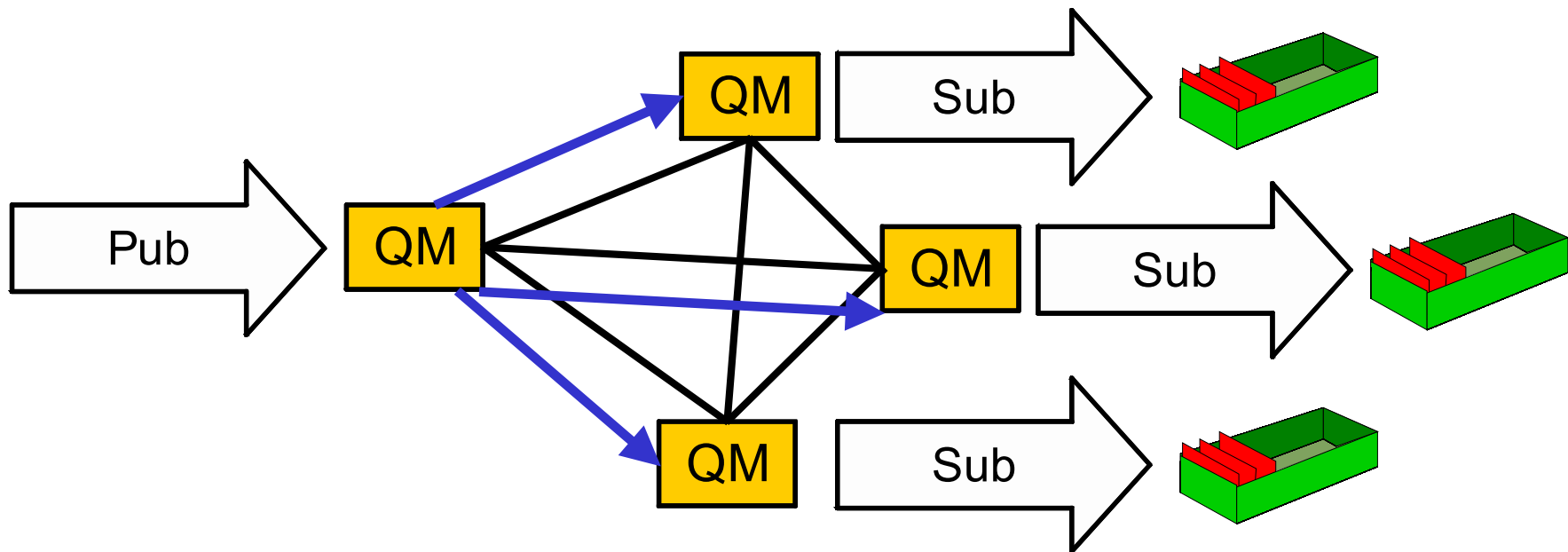
Publish/Subscribe Administration (2)

- Support for durable and non-durable subscriptions
 - ⌘ With durable, a client can go away and come back later without missing messages
 - ⌘ Durable can cause queues to fill – generating configured depth events as warning
 - ⌘ Non-durable exist only for the lifetime of the application
 - ⌘ No manual "cleanup" task needed when applications end unexpectedly
- Subscriptions
 - ⌘ Able to see who is subscribing to topics: like DISPLAY QSTATUS
 - ⌘ Able to create subscriptions on behalf of a third party
- Security
 - ⌘ Use of a topic is restricted by permissions on the associated topic object
 - ⌘ On z/OS drives need for mixed-case support in RACF
 - ⌘ Follows existing WMQ model for security configuration (SAF or OAM)
- Conversion of point-to-point applications without code changes
 - ⌘ Administrative changes to objects
 - ⌘ A queue alias can point to a topic, not just a local queue

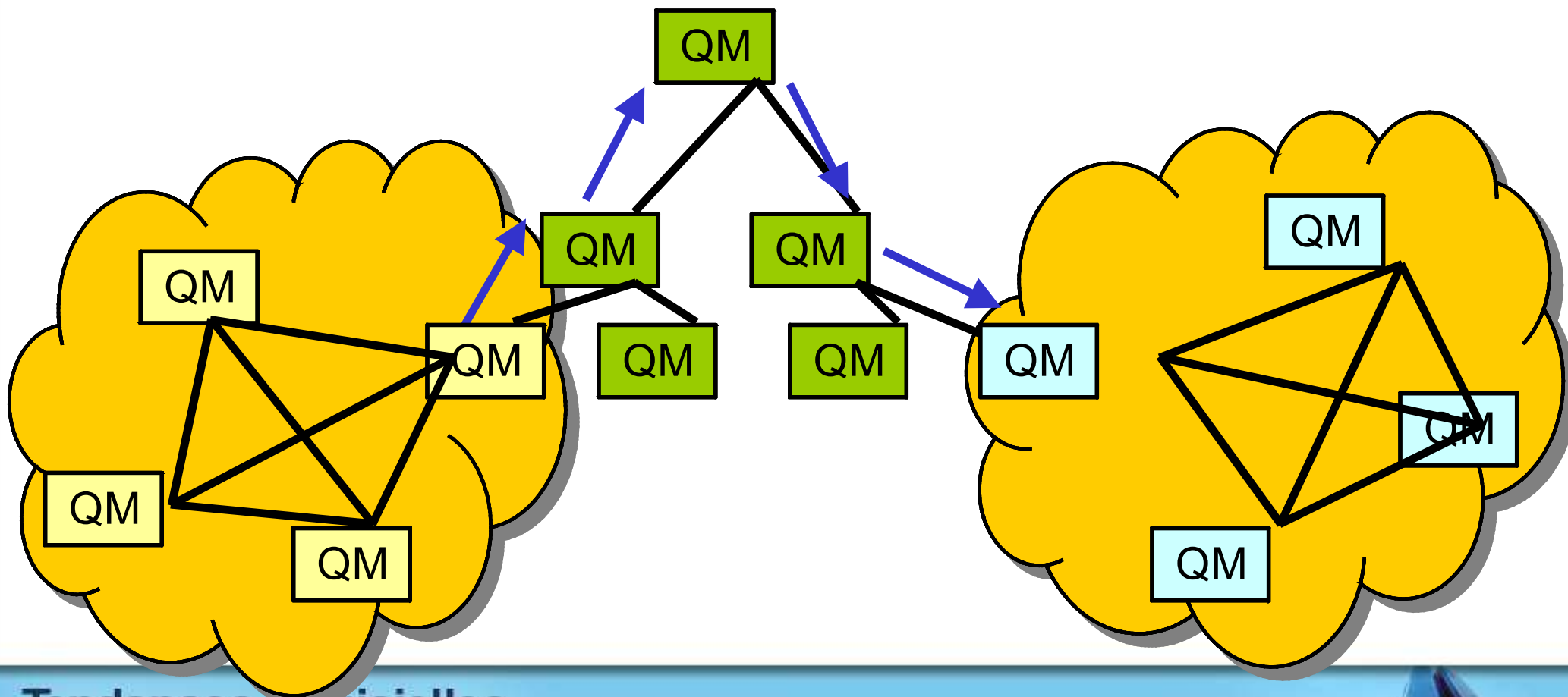


Publish/Subscribe in a Cluster

- Consistent topic definitions in cluster
- Multiple routes across cluster



Publish/Subscribe in Combined Hierarchy & Clusters



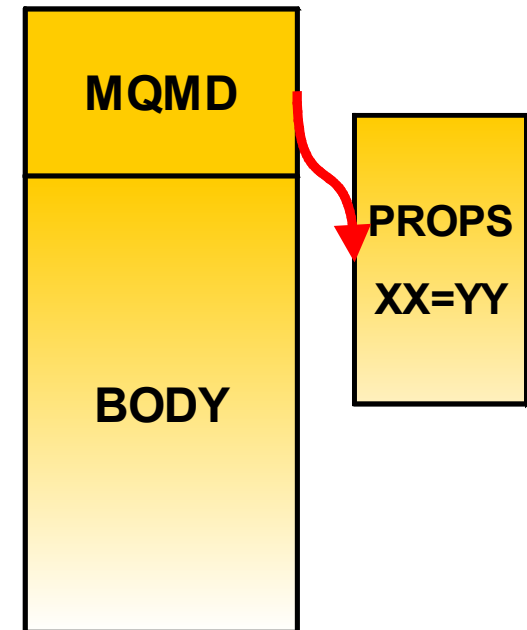
Publish/Subscribe Application Programming

- Cannot significantly change the JMS API
 - ☞ But we want some of its facilities more easily available in the MQI
 - ☞ To improve MQI programming and improve (make thinner) the JMS layer
 - ☞ JMS implementation exploits new MQI functions
- New verb for subscribing
 - ☞ So you do not need to build RFH or RFH2 headers in the application
 - ☞ **MQSUB** registers a subscription
 - Includes information about where messages will be read from
 - Do not need to specify a queue – can be automatically assigned
 - ☞ Retained publications delivered immediately after subscribing
- New options on existing verbs
 - ☞ MQOPEN to get access to a topic
 - ☞ MQCLOSE deregisters a subscription
 - ☞ MQPUT, MQGET to publish and to receive publications
- Sample programs included to demonstrate use



Message Properties

- Arbitrary values associated with the message but not part of the body
 - ⌘ Like a user-extendable MQMD
 - ⌘ Already part of JMS
- New verbs including **MQSETMP** and **MQINQMP**
 - ⌘ Properties can be integers, strings, boolean, etc.
- Easier to use than RFH2 folders
 - ⌘ Receiving apps do not see them unless they want
 - ⌘ No need to parse and skip over message headers
- Configuration options for compatibility
 - ⌘ Queue and channel attributes define behaviour
 - ⌘ Defaults will create RFH2 folders
- Permits explicit statement of relationships between messages
 - ⌘ eg Message X is a **REPLY** to Message Y
 - ⌘ Messages referred to by handles



Other MQI Enhancements

- Asynchronous Message Reception
 - œ New verb **MQCB** defines a callback function
 - œ Automatically Invoked when a message arrives
 - œ No need for MQGET(WAIT) or MQGET(SIGNAL)
 - œ A thread can receive messages from multiple queues
 - œ New verb **MQCTL** to start and stop message delivery to callback
- Selectors
 - œ Use a SQL92 clause to select messages by properties including MQMD fields
 - œ Can be specified on MQOPEN, MQCB for filtering messages
 - œ Selection is done inside queue manager
 - œ Not looking inside message body
 - Message Broker still required for content filtering
- Cooperative Browsing and Message Tokens
 - œ Efficient interface for applications reading from the same queue
 - œ Example: "master" program browses a queue telling "slaves" which message to work with, based on elements within the message
 - œ 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



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

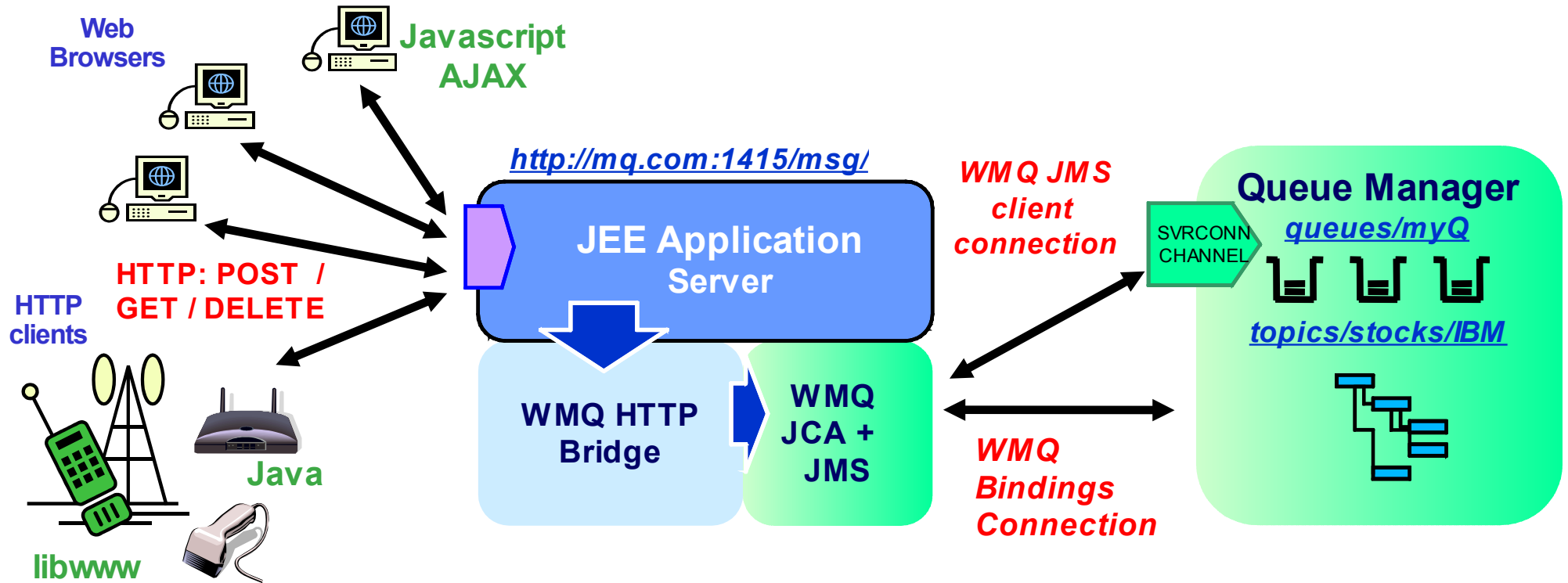


Client Connection Management

- 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
- Free connections to z/OS for administration programs like WMQ Explorer
 - ⌘ Limited number of clients permitted by V7 license without CAF



WebSphere MQ Bridge for HTTP - Architectural Overview



- Key features of the WebSphere MQ Bridge for HTTP -
 - Maps URIs to queues and topics
 - Enables MQPUT and MQGET from
 - Web Browser
 - Lightweight client
- Alternative non-servlet implementation available as MA94



HTTP-MQI Verb / Resource Mapping

- Define URI to identify queue (or topic)
- Modelled on REST principles
 - Simple translation of HTTP to MQI
- Message Format:
 - Header fields (MQMD) conveyed in HTTP headers
 - Body is passed in HTTP entity body
 - Message type is conveyed in HTTP Content-Type
 - “text/plain” or “text/html” equate to WMQ string messages (MQFMT_STRING)
 - All other media types map to WMQ binary messages (MQFMT_NONE)

HTTP verb mapping

Resource

Sample URIs

GET

POST

PUT

DELETE

Messages

<http://host/msg/queue/qname/>
http://host/msg/topic/topic_path/

MQGET w.
browse

MQPUT

-

MQGET



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



Some Performance Information

- 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 throughput improved up to 45%
 - ∞ Latency also improved
- Measurements taken from pre-release code
- Performance reports will be published as usual on SupportPac site



Managed File Transfer with WebSphere MQ File Transfer Edition



What's Driving Your Business Today?

Business Demands

Support up-to-the-minute 24/7 decision making & forecasting

Reduce disruption, cost & time wasted resolving errors in partner & customer transactions

Meet Regulatory Compliance or other audit obligations by demonstrating integrity of financial or sensitive data to avoid penalties

Reduce cost & time to market of new business offerings

Streamline unnecessary investments

Make changes & absorb surprises without impacting ability to continue executing

Exploit best practices, processes & tools across organization

IT Challenges

Reduce "batch window" or enable continuous stream of updates

Improve reliability of data exchange between IT systems & eliminate sources of error

Preserve integrity of data and secure it – especially when moving it between IT systems

Accelerate new development by avoiding duplication of function

Consolidate & reuse IT infrastructure across enterprise

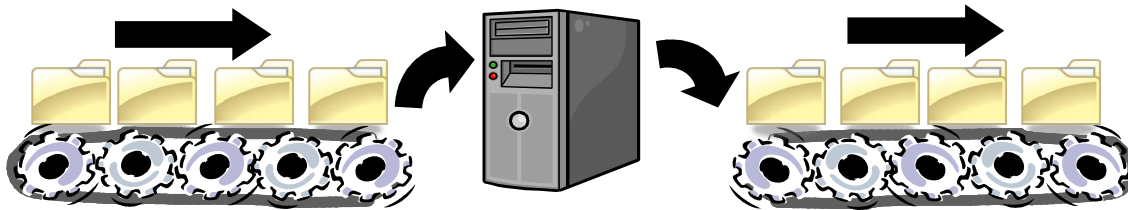
Leverage SOA capabilities across the entire IT Infrastructure

Enable widespread use of IT infrastructure & reduce dependency on IT specialists

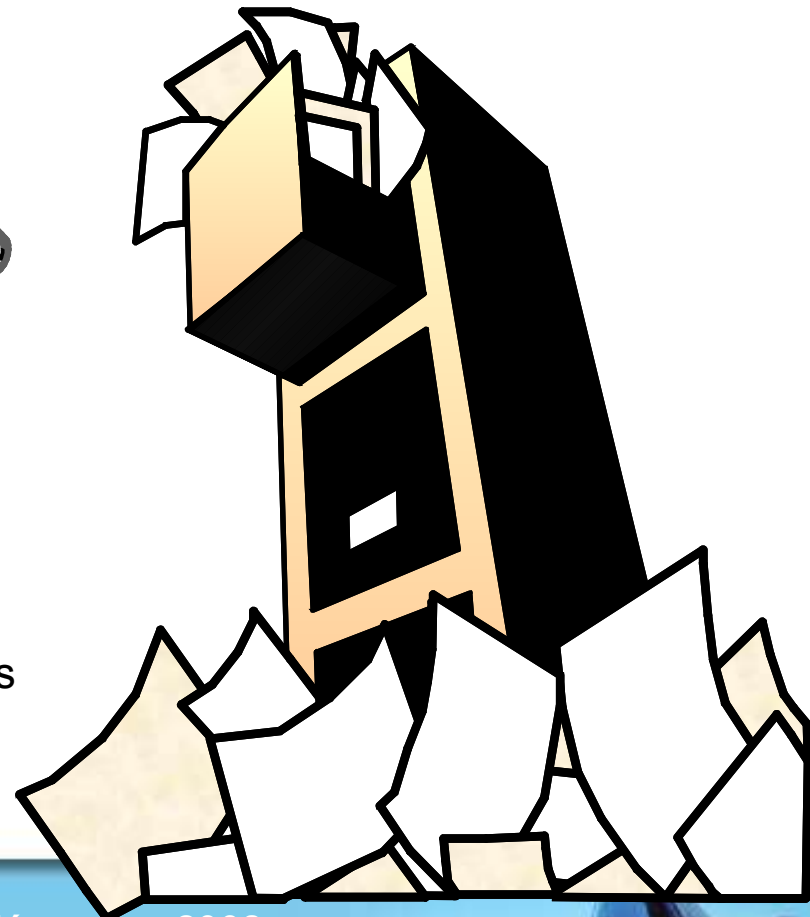


How Are Most Organizations Moving Files Today?

- Currently, many business critical applications connect by exchanging files
 - ▶ Most organizations have *several* products, and different techniques for doing file transfer
 - ▶ Typically there is a mix of FTP, homegrown, and other file transfer products



- Why is FTP use so widespread?
 - ▶ Lowest common denominator
 - ▶ Quick fix – repent at leisure
 - ▶ Simple concepts – low technical skills to get started
 - ▶ FTP products are “free”, simple, intuitive and ubiquitous



How Do You Move Files?

IT Challenges

Reduce “batch window” or enable continuous stream of updates

Improve reliability of data exchange between IT systems & eliminate sources of error

Preserve integrity of data and secure it – especially when moving it between IT systems

Accelerate new development by avoiding duplication of function

Consolidate & reuse IT infrastructure across enterprise

Leverage SOA capabilities across the entire IT Infrastructure

Enable widespread use of IT infrastructure & reduce dependency on IT specialists

How do you transfer files?

- Can you finish ever larger batches of file transfers overnight?
- Can you transfer updates continuously throughout the day?

- Can you move files reliably across your distributed IT systems?
- Can you restart file transfers that haven't completed properly?
- Can you automate & schedule transfers to avoid human-errors?

- Can you prove that files only went where were supposed to?
- Can you detect & recover whenever files are partially sent?
- Can you prevent unauthorized access to files?

- Can you avoid developing code to improve file transfers?
- Can you avoid duplicating file transfer logic across apps?

- Can you use a single infrastructure for all traffic including files?
- Can you reduce your administration & maintenance costs?

- Can you apply ESB capabilities to files e.g. transformation?
- Can you involve files as part of your business processes?
- Can you include file-oriented applications in your SOA?

- Can you enable more IT staff to use a common infrastructure?
- Can you enable less skilled staff to use your IT infrastructure?
- Can you bring service-oriented & batch/file systems together?



Shortcomings of basic FTP

■ **Limited Reliability**

- ▶ Checkpoint restart facilities not always available – files might be lost
 - Not transactional in nature
- ▶ Transfers or batches of transfers may terminate without notification
 - Partial files or incomplete batches could be used in subsequent business processes causing issues with integrity of applications and data downstream
- ▶ Files data could be unusable after transfer (ASCII/Binary transfer)

■ **Limited Flexibility**

- ▶ All resources usually have to be available concurrently
- ▶ Often only one FTP-based transfer can run at a time
- ▶ Typically transfers cannot be prioritized

■ **Limited security**

- ▶ In some cases usernames/ passwords are sent with data – as plain text!
- ▶ Non-repudiation often lacking
- ▶ Privacy, authentication and encryption may not be available

■ **Limited visibility and traceability**

- ▶ Typically transfers cannot be monitored and managed centrally or remotely
- ▶ Logging capabilities may be limited and may only record transfers between directly connected systems
- ▶ How to track the entire journey of the file – not just from one machine to the next but from the start of its journey to its final destination



What is Managed File Transfer?

- Enables *managed* movement of files and documents between IT systems
 - ▶ Auditable
 - ▶ Reliable
 - ▶ Secure
 - ▶ Any size file
 - ▶ Automated : Eliminating need to manually detect transfer problems and restart transfers
 - ▶ Backbone : Across distributed IT systems that need not be directly connected
 - ▶ Time-independent : Without requiring IT systems and network to be constantly available
 - ▶ Centralized control : Enabling remote management and monitoring of all aspects of transfer

- **Managed File Transfer is a strategic part of an organization's IT infrastructure**
 - ▶ Should be aligned with other transport mechanisms e.g. messaging
 - ▶ Managed File Transfer should work with and re-enforce SOA initiatives
 - Including applying ESB capabilities to files



IBM's Vision – SOA Messaging Backbone

- Addressing full spectrum of universal transport requirements



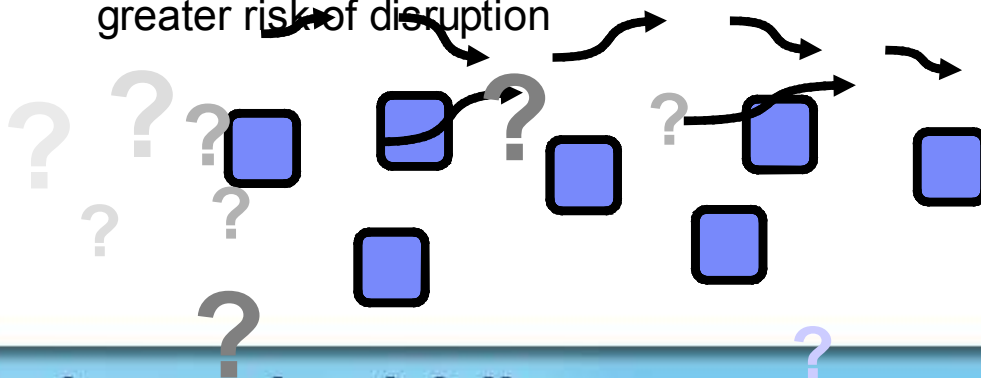
Auditing File Transfers

■ Pain points

- ▶ File transfers impossible to audit – exposing risks of non-compliance with regulations e.g. Sarbanes-Oxley, MiFID, HIPAA etc
- ▶ Unreliable nature of FTP means transfer failures need to be detected (by writing application code) and re-sent (consuming network bandwidth) increasing the batch-window needed to transfer files
- ▶ Inflexible nature of FTP means that development and maintenance costs resulting from application changes are spiraling so that each additional change is more costly, takes longer and carries greater risk of disruption

■ Need ability to track movement of files end-to-end as these move around organization

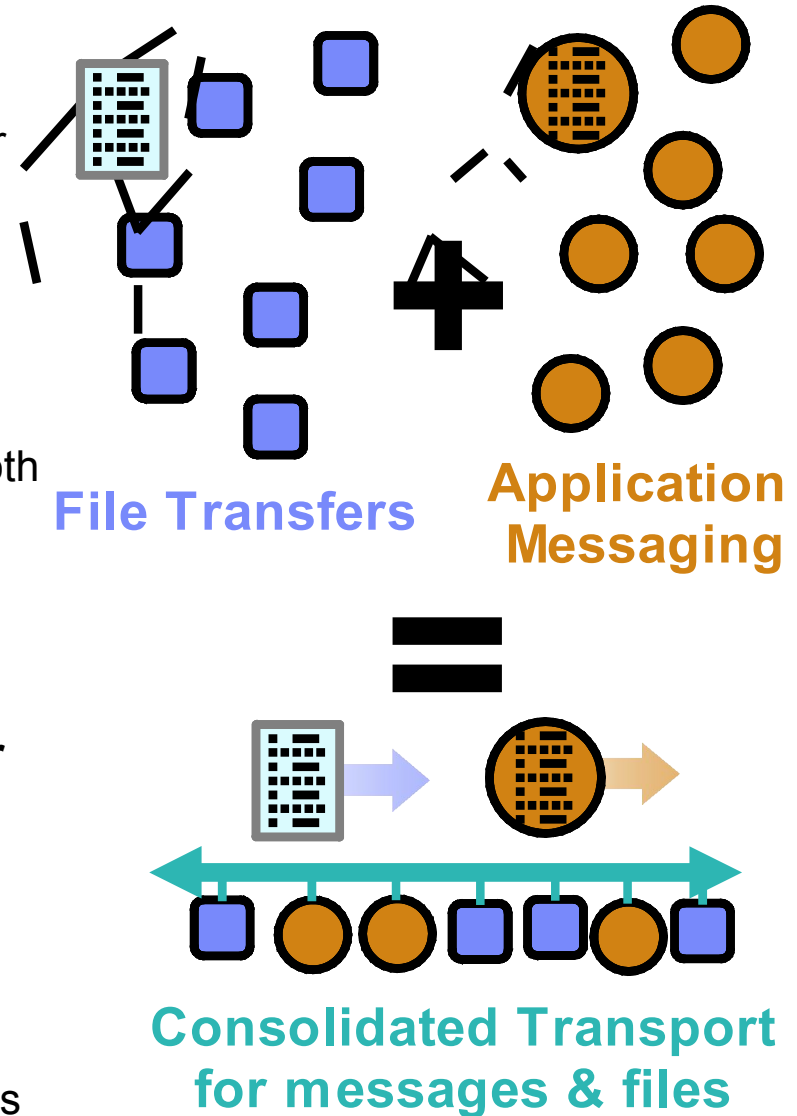
- ▶ Improving ability to meet regulatory compliance obligations in demonstrating the integrity of data in motion and of business data used to compile financial reports
- ▶ Improving the reliability of file transfers so that applications no longer need to detect transfer failures, corrupted or partially transmitted files and avoiding need to always re-transfer entire batches of files when failures occur
- ▶ Increasing flexibility of infrastructure so that changes to hardware, O/S, applications and networks + re-working of code



*Where has this file come from?
Where has it been before it got here?
Has it been changed? By Who?
When?
How can I quickly make changes?
Is the file complete?*

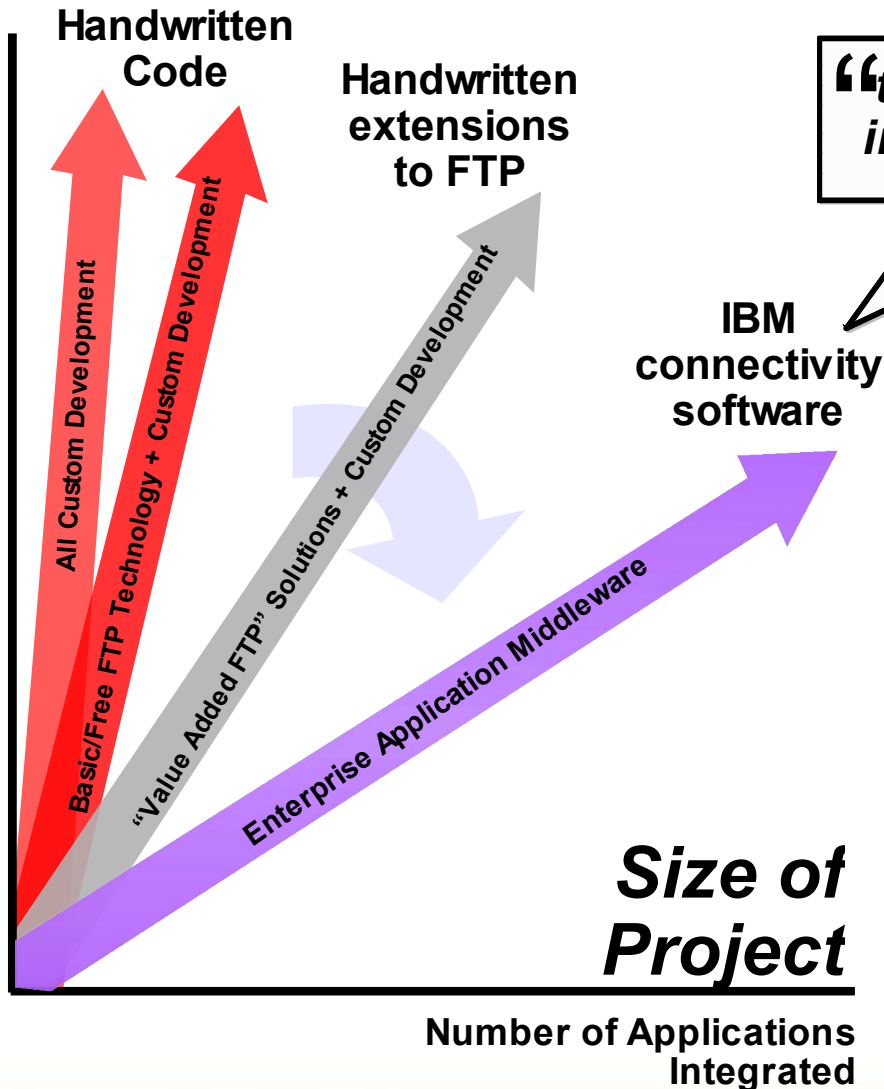
Consolidating IT Infrastructure

- **Developing and maintaining entire parallel infrastructures**
 - ▶ One for files – typically built on FTP – and one for application messaging – based on WebSphere MQ or similar
- **Pain points**
 - ▶ Duplicate maintenance burden is sucking time and resources from IT team and inhibiting progress with other initiatives e.g. SOA
 - ▶ Operations teams are duplicating administration of both infrastructures with poor traceability for file transfers and for data passing between these infrastructures
- **Need ability to consolidate these duplicate infrastructures into one single universal transport for both messages and files**
 - ▶ Operational savings and simplification
 - ▶ Reduced administration effort
 - ▶ Reduced skills requirements and maintenance
 - ▶ Improved reliability, auditability and security of File Transfers
 - ▶ Infrastructure that can be re-used as basis for SOA as opposed to FTP



Cut IT integration cost and maintenance 2-4 times

Cos
Building & Maintaining



“the more applications you integrate the more you save”

*“Custom-built, in-house, hard-coded integration solutions...
...often take 2 to 4 times the time and effort to build
...require a similar multiple of ongoing maintenance and support effort...
... IBM application integration costs 2-4 times less”*

Source: “Enterprise Integration Challenge,” Software Strategies, 2006

Software Strategies



What is WebSphere MQ File Transfer Edition?

- Newest member of growing WebSphere MQ family
 - Builds upon WebSphere MQ's proven transport backbone
 - z/OS product pre-reqs WebSphere MQ for z/OS
 - Distributed product includes WebSphere MQ and trade-up option

■ Robust solution for Managed File Transfer

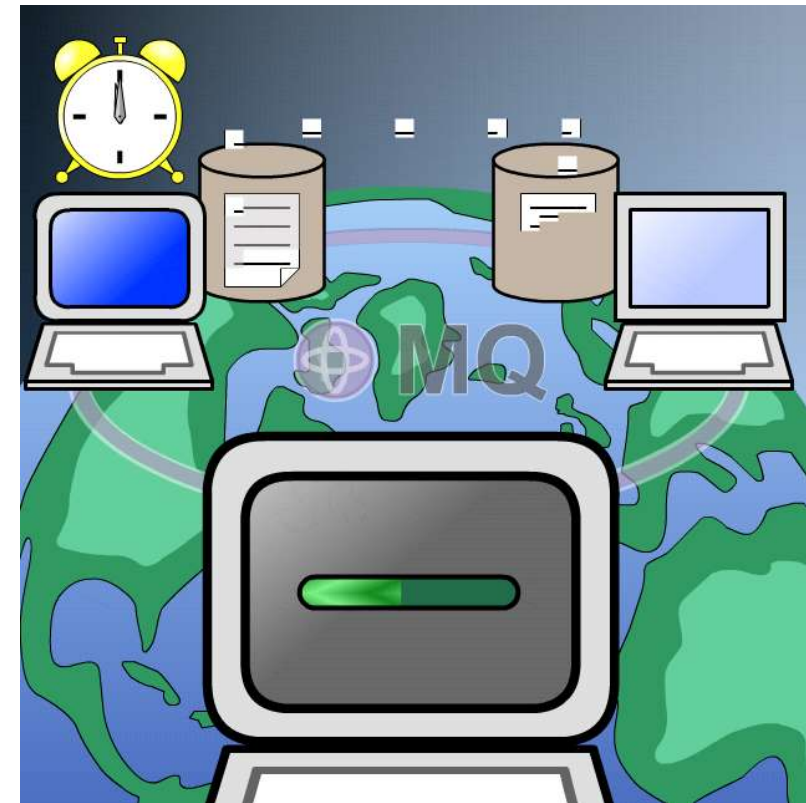
- ▶ Enable control of all aspects of file movement between IT systems
- ▶ Provide file delivery reliability
- ▶ Optimized for both small and massive files
- ▶ Provides audit trail of transfers

■ Designed to integrate with IBM's SOA portfolio

- ▶ Enables files to be delivered to WebSphere Message Broker for File Processing

■ Planned availability 4Q 2008

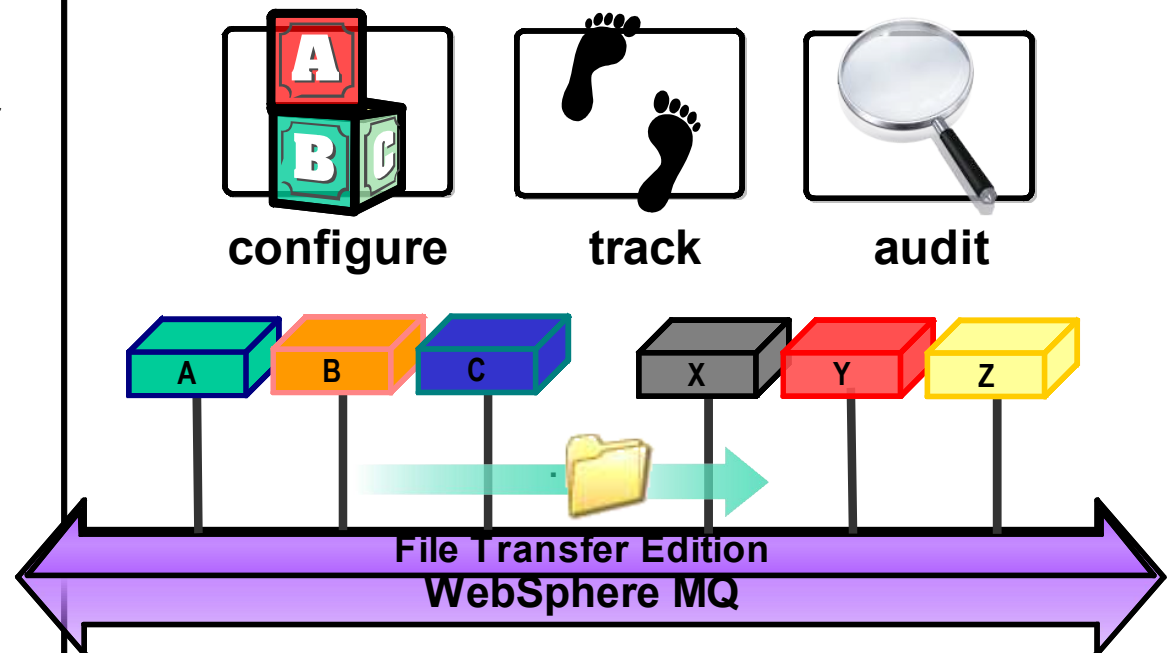
Product roadmap, timeframes and features subject to change and not to be viewed as IBM commitments



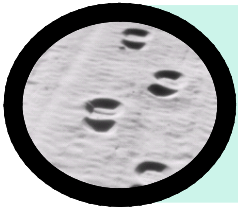
WebSphere MQ File Transfer Edition

- Adds file transfer services to WebSphere MQ to enable movement of files – regardless of size – in a managed way (reliable, auditable, secure)
- Multi-purpose infrastructure – for both files and messages

- ✓ Flexible backbone for transfers – not a single-hop solution like FTP
- ✓ Multi-purpose – use for messages and files
- ✓ Auditable with logging subsystem that tracks transfer at source and at destination for audit purposes
- ✓ Massive files – larger than MQ messages
- ✓ Reliability leveraging the MQ transport
- ✓ Integration with MQ-enabled apps and ESBs
- ✓ No need to program – no need to use APIs
- ✓ Simple graphical tooling enabling remote configuration
- ✓ Automatic file conversion and compression
- ✓ Security - of file payload using SSL
- ✓ Visual transfer status reporting
- ✓ Support for many supported MQ environments



Key Themes – WebSphere MQ File Transfer Edition



Auditable

- Audit logs of transfers at source and target
- Audit data persisted to MQ queues and/or relational database.
- Captures time-stamped log at source and target



Ease-of-Use

- Remote console for transfer initiation, unattended operation, scripting, scheduling, restart policies, status display
- Integrated with MQ Explorer configuration tooling



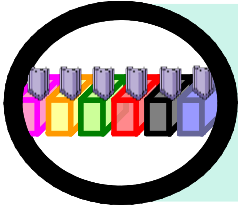
Simplicity

- Small footprint, fast install
- No need to write code or use API to configure transfers – Enabled via GUI
- Leverages WebSphere MQ – no other technology pre-reqs



Security

- Access to individual files subject to file system permissions
- Link level security (inheriting MQ SSL security)



Breadth

- Support WebSphere MQ V6 and V7 for transfers
- Core Platform support (z/OS, Linux (32 Bit), Solaris, AIX, HP, Windows)
- Good file type support (ASCII/EBCDIC, CR/LF, Flat files, z/OS QSAM, BPAM, VSAM)



Automated Transfers

- Transfers can be scheduled to repeat at predetermined intervals
- Transfers can be triggered by range of file system events e.g. new files, updated file, etc.

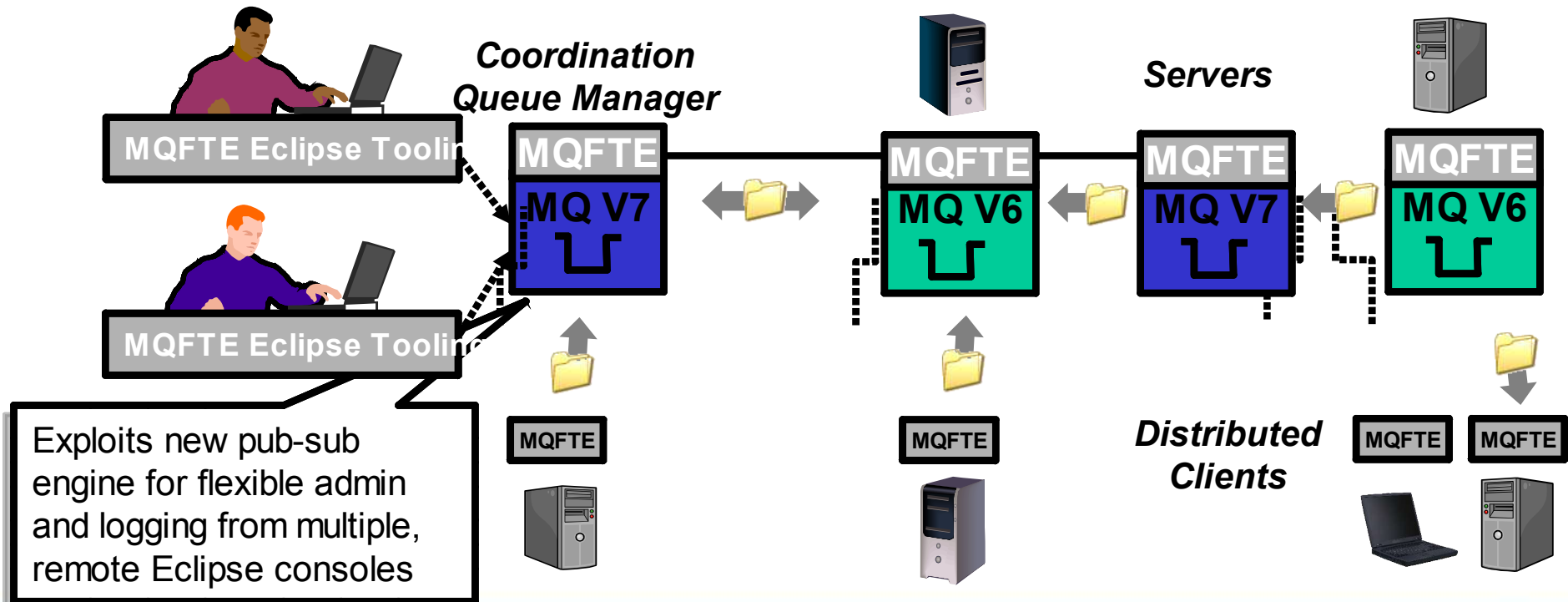
Features & Benefits

File Transfer Backbone	Simplifies configuration, administration & auditing
Time-Independent File Transfer	Improves productivity of applications
Reliable File Transfer	Reduces business disruption by helping preserve integrity of file data
Event-Driven File Transfer	Enables flexible distribution of file data and alerting
Centralized Configuration	Remote management of the whole file transfer backbone
Remote status reporting	Enables transfer status to be viewed remotely
Scheduling	Enables transfers to be scheduled at intervals
Automation	Enables transfers to be triggered based on file events
Scripting	Enables programmatic control of transfers
Audit Log	Enables auditing of file movements at source and target
Zero coding	Accelerates solution deployment and reduces skills requirements
Custom Exits	Enables additional user function to be added pre- and post-transfer
ESB Connectivity	Enables mediation, transformation and content-based routing to be applied to files using WebSphere Message Broker



Architecture

- Enables remote GUI configuration and admin using same tooling as MQ
- Tooling publishes transfer requests to Backbone
- “Agents” running alongside Queues managers publish audit trail to Coordination Center
- “Agents” monitor file directories, load/unload files & perform pre- & post-transfer activities
- Coordination Queue manager publishes transfer status, process and audit trail
- Coordination Queue manager requires MQ V7.0
- Multiple Coordination Queue manager could control transfers, capture audit log and publish status



Time-Independent File Transfer

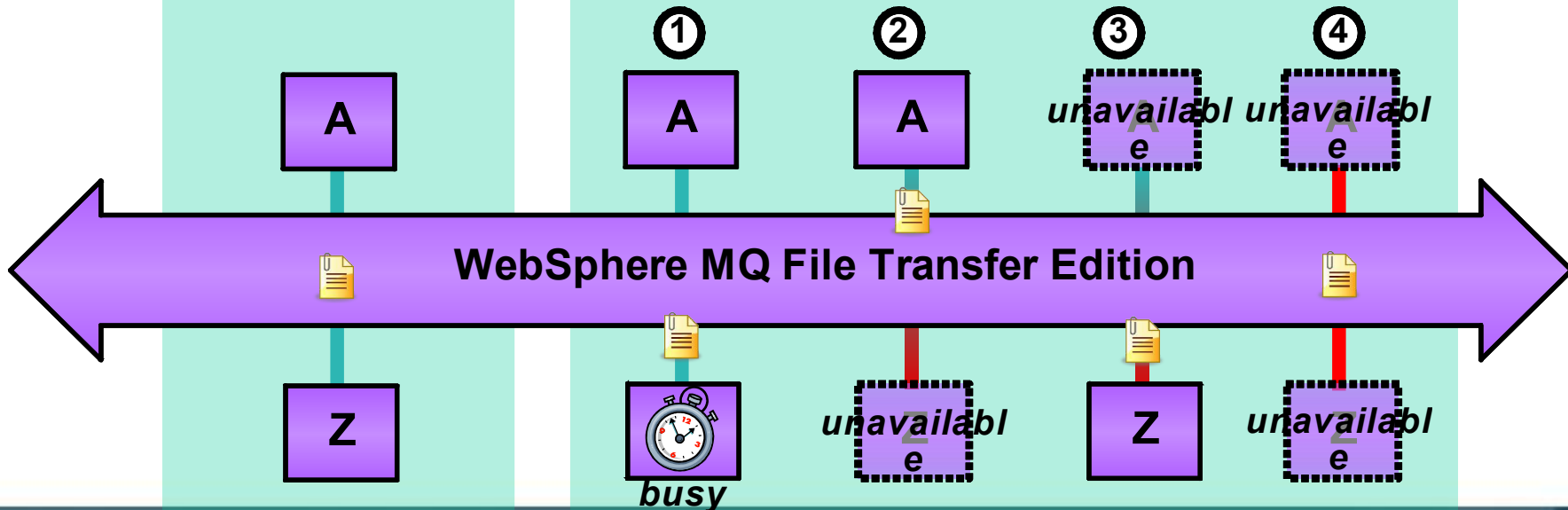
- Transfer files regardless of when solution components are free or available

Seems like this...

Applications transferring files can assume that the sender, receiver & network will always be constantly available

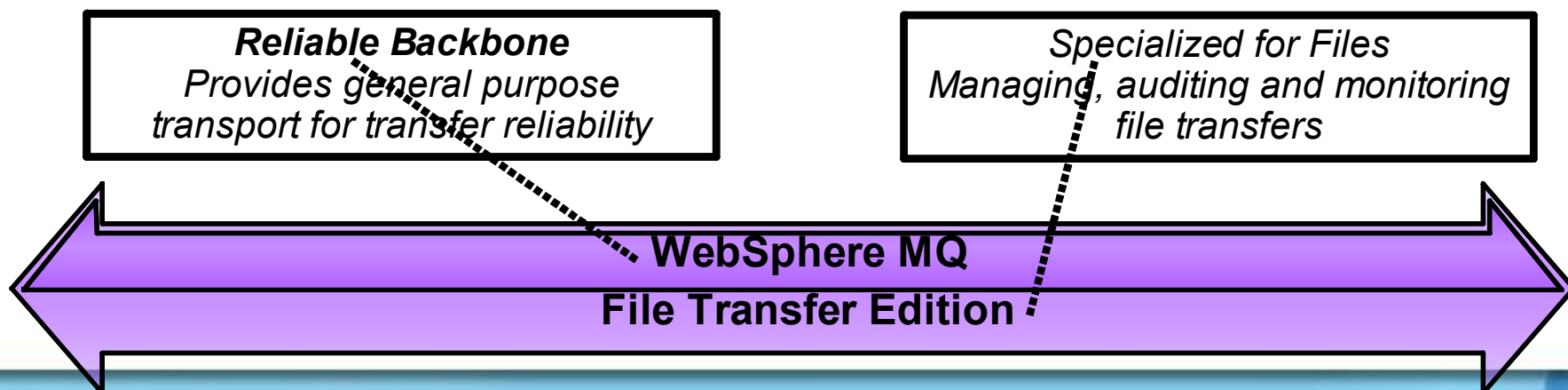
...Even when this might be happening!

- Sender application does not need receiver to be available in order to send files
- Sender application can continue doing useful work while files are being transferred
- Backbone handles network interruptions & recovers transfer once network resumes
- Senders & receivers can continue useful work without waiting for transfers to finish



Reliable File Transfer

- Basic FTP protocol lacks capability need to ensure data is delivered
 - ▶ No integrity checking on the receiver side
 - ▶ No way of verifying whether files received are complete or not
- Traditional Managed File Transfer suites are based on FTP
 - ▶ Need to augment FTP protocol with functions to address these inherent issues
 - ▶ Ability to integrate enterprise applications and databases is dependant on extensions from vendor
- IBM Managed File Transfer starts with industry's leading connectivity backbone: WebSphere MQ
 - ▶ **Reliability** – Patented technology & well-grounded two-phase commit techniques
 - ▶ **Trusted** – 10,000 client sites worldwide moving \$trillions worth of data every day
 - ▶ **Proven** – Leader in messaging connectivity for over 15 years
 - ▶ **Integrated** – with IBM's SOA portfolio including ESB and BPM software



Centralized Configuration & Administration

- Logically centralized configuration of remote, distributed backbone
- Remotely view & configure entire backbone – including on z/OS



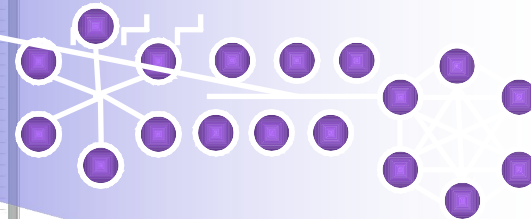
- Create
- Delete
- Display
- Modify
- Deploy
- Test

The screenshot shows the IBM WebSphere MQ Explorer interface. On the left is a tree view of the configuration. The main area displays a table of Transfer Templates:

Name	Source Host	Source File	Target Host	Target File	Scheduled Start	Trigger Events
Tx001	TAgent1	home/reg/cont.db	RAgent1	home/reg/cont.db	20080416-23:22:0000	Schedule
Tx001	TAgent1	home/reg*.*	RAgent1	home/reg*.*		
Tx001	TAgent1	home/reg/abc.txt	RAgent1	home/reg/abc.txt		
Tx002	TAgent2	home/reg/recv	RAgent2	recv/reg	20080416-23:22:0000	Schedule
Tx003	TAgent2	home/reg/recv	RAgent2	recv/reg	20080416-23:22:0000	Schedule, Filesize
Tx004	TAgent3		RAgent3			Filecheck, Periodic

Below the table is a 'Summary transfer progress' section with a table:

Name	Source	Dest	File number	Current file	Rate	Started (local)
Tx001	agent1	..sourceDir agent2	..destDir 1 / 1	ubuntu-716.iso (320M / 699 MB)	346B/s	20080416-23:22:0000
Tx002	agent1	..sourceDir agent2	..destDir 38 / 754	MQ2-license.txt (98 / 13 kB)	284B/s	20080416-23:40:0000



SOA MESSAGING BACKBONE

- Visual display at a glance
- Eclipse-based environment
- Extensible and customizable
- Remote connection from Linux x86 and Windows
- SSL secured connections



Eclipse-based GUI integrated into MQ Explorer

The screenshot displays the WebSphere MQ Explorer interface. On the left, a tree view shows the hierarchy of resources, with 'Managed File Transfer' under 'MyQueueMgr' circled in red. The main pane shows the 'Managed File Transfer' section, including a description of the product, a list of icons, and a summary transfer progress table.

Managed File Transfer

WebSphere MQ File Transfer Edition is new product in the WebSphere MQ product family offering Managed File Transfer capabilities, including:

- Moving files between IT systems reliably and securely regardless of their size.
- Robustness and resilience to failures
- Logging of file movements for audit purposes
- Automation of all aspects of transfers. (Eliminating the need to manually detect transfer problems and restart transfers.)
- Scheduling of transfers

[Help on Managed File Transfer](#)

Managed File Transfer icons

- Managed File Transfer
- Collection of scheduled transfers
- Collection of Managed File Transfer agents
- An agent

[Help on WebSphere MQ icons](#)

Scheme: Default for Saved Transfers
Last updated: 13:23:23

Summary transfer progress

Name	Source	Dest	File number	Current file	Rate	Started (local)
Tx001	agent1 .../sourceDir	agent2 .../destDir	1 / 1	ubuntu-710.iso (320kB / 699 MB)	34kB/s	20080416-23:22.00000
Tx002	agent1 .../sourceDir	agent2 .../destDir	38 / 754	MQ7-license.txt (5B / 13 kB)	28kB/s	20080416-23:40.00000

Scheme: Default for Saved Transfers
Last updated: 13:23:23



Creating File Transfers

The screenshot shows the 'Create Transfer' wizard in IBM WebSphere MQ Explorer. The wizard is divided into two main sections: 'Basic' and 'Advanced'.

Basic Settings (Step 2):

- From:** *Host and *Path fields with 'Find...' and 'Browse...' buttons. A checked checkbox for 'Include subdirectories' is visible.
- To:** *Host and *Path fields with 'Find...' and 'Browse...' buttons.
- Filename:** 'Use original filename' dropdown and 'Append' dropdown (set to 'None').
- Basic Settings:**
 - Mode: Text transfer (ASCII/EBCDIC & CR/LF automated), Binary transfer (no conversion).
 - 'Add to transfer...' button.

Advanced Settings (Step 3):

- File Handling:**
 - A failure of any file in a <set> means that the <set> has failed
 - Remove files from target on failure and write error to audit stream
 - Leave successful files and write the error to audit stream
 - Overwrite files on the target file system that have the same name
 - Append source file content to file of same name on target filesystem
 - Replace the file of the same name on target filesystem
 - Move files
 - Remove source files on successful completion of whole group
 - Remove source files on successful completion of each file
- Maximum number of files expected:** 150
- Transfer Priority:** High | Default | Low
- Settings below are speculative only:**
 - Only transfer changed files (Modified timestamp has changed)
 - Keep source file's attributes (owner, group, timestamp, permissions)
- For text transfers from a z/OS machine:**
 - Create new lines from z/OS ASA control characters
 - Ignore z/OS ASA control characters
 - Lock all files in a group at the start of the group transfer

Navigation buttons: Previous, Next, Finish.

Footer text: Scheme: Default for Saved Transfers, Last updated: 13:23:23



Auditing File Transfers

The screenshot displays the IBM WebSphere MQ Explorer interface. On the left is a tree view of the MQ environment, including 'Managed File Transfer' and 'Transfer Agents'. The main pane shows the 'Transfer History' table, which lists individual transfer records with columns for Name, Source Agent, Source File, Target Agent, Target File, Started (UTC), Completion State, State recorded (UTC), and Owner. A context menu is visible over the table, offering options like 'Delete', 'Del', 'Archive selected...', and 'Archive all...'. Below the history table is a 'Summary transfer progress' table, which provides a high-level view of ongoing transfers, including Name, Source, Dest, File number, Current file, Rate, and Started (local).

Name	Source Agent	Source File	Target Agent	Target File	Started (UTC)	Completion State	State recorded (UTC)	Owner
Tx001	TxAgent1	/home/greg/recipe	RxAgent1	/recipes/greg	20080416-23:22.00000	✓ Completed	20080822-23:59.00000	Bob Builder
Tx002	TxAgent2	/home/greg/recipe	RxAgent2	/recipes/greg	20080416-23:22.00000	✗ Failed	20080822-23:59.00000	Bob Builder
Tx003	TxAgent2	/home/greg/recipe	RxAgent2	recipes/greg	20080416-23:22.00000	▣ Partial - in progress	20080822-23:59.00000	Bob Builder
Tx004	TxAgent3	/home/greg/recipe	RxAgent3	/recipes/greg		⊙ Not yet started	20080822-23:59.00000	Bob Builder

Name	Source	Dest	File number	Current file	Rate	Started (local)
Tx001	agent1 .../sourceDir	agent2 .../destDir	1 / 1	ubuntu-710.iso (320kB / 699 MB)	34kB/s	20080416-23:22.00000
Tx002	agent1 .../sourceDir	agent2 .../destDir	38 / 754	MQ7-license.txt (5B / 13 kB)	28kB/s	20080416-23:40.00000

- Captures log at Source and Target
- Can be viewed remotely using MQ Explorer
- Applications can subscribe to audit information (or portions of it)
- Log records can be loaded into other systems e.g. SQL database



Scheduling File Transfers

Create Transfer
New Edit Transfer
Enter Schedule and Event Settings (step 3)

Schedule Triggers

Schedule transfer Transfer on completion of wizard

Hours: 00 01 02 03

Days: Mon 12th Tue 13th 14th 15th 16th 17th 18th

Weeks: 17 18 19 20 21 22 23 24 25

Months: Mar Apr May Jun Jul Aug

Years: 2007 2008 2009 2010 2011 2012

Transfer Name
Start: date time
Repeat:
Every: 14 days
Until: Forever|date time

Transfer **Transfer Name** at *hh:mm*, starting *dd/mm/yyyy*
Repeat every *nn minutes|hours|days|weeks|months|years*
Until *hh:mm* on *dd/mm/yyyy*

Edit Schedule

Previous Next Finish

Triggering File Transfers

Create Transfer

New Edit Transfer

Enter Schedule and Event Settings (step 3)

Schedule Triggers

Trigger File Trigger

Enable File DetectionTrigger

File

*Host Find...

*File Browse...

Trigger Attribute

Trigger when file:

Appears

Disappears

Exceeds 1 kB

Unlocks

Previous Next Finish

Monitoring File Transfer Progress

The screenshot displays the IBM WebSphere MQ Explorer interface. On the left, a tree view shows the hierarchy of the MQ environment, including 'Managed File Transfer' and 'Current Transfers'. The main window, 'Transfer Progress Detail', shows a table of active transfers with columns for Name, Source, Dest, File number, Current file, Rate, and Started (local). A context menu is visible over the table, offering 'Pause...' (Ctrl+P) and 'Cancel' (Del) options. Below the table, there are fields for 'Scheme: Default for Saved Transfers' and 'Last updated: 13:23:23'. At the bottom, a 'Summary transfer progress' window provides a condensed view of the same data.

Name	Source	Dest	File number	Current file	Rate	Started (local)
Tx001	agent1 .../sourceDir	agent2 .../destDir	bundle of 3	ubuntu-710.iso (320kB / 699 MB)	34kB/s	20080416-23:22.00000
Tx001.1		agent2 .../destDir	1	ubuntu-710.iso (320kB / 699 MB)	34kB/s	20080416-23:22.00000
Tx001.2		agent2 .../destDir	2	ubuntu-710.iso (320kB / 699 MB)	34kB/s	20080416-23:22.00000
Tx001.3		agent2 .../destDir	3	ubuntu-710.iso (320kB / 699 MB)	34kB/s	20080416-23:22.00000
Tx002	agent1 .../sourceDir	agent2 .../destDir	38 / 754	MQ7-license.txt (5B / 13 kB)	28kB/s	20080416-23:40.00000

- Displays visual progress of transfers
- Current progress of remote transfers
- Transfer progress can be subscribed
- Enables 3rd party and bespoke applications to monitor or react to events e.g. alerts for a stalled transfer



Scripting

- Scripting language will provide automated, programmatic control of transfers
- Transfer commands can be invoked from the supported Operating Systems shell environment
- Developers can use any native scripting language on the OS that can invoke these commands
- Examples:
 - ⌘ `fteCreateTransfer` line Starts a new file transfer from the command line
 - ⌘ `fteStartAgent` command line Starts a File Transfer agent from the command line
 - ⌘ `fteStopAgent` way Stops a File Transfer agent in a controlled way
 - ⌘ `fteShowAgentDetails` Transfer agent Displays the details of a particular File Transfer agent
 - ⌘ `fteShowAgents` agents Displays the status of all known Transfer agents



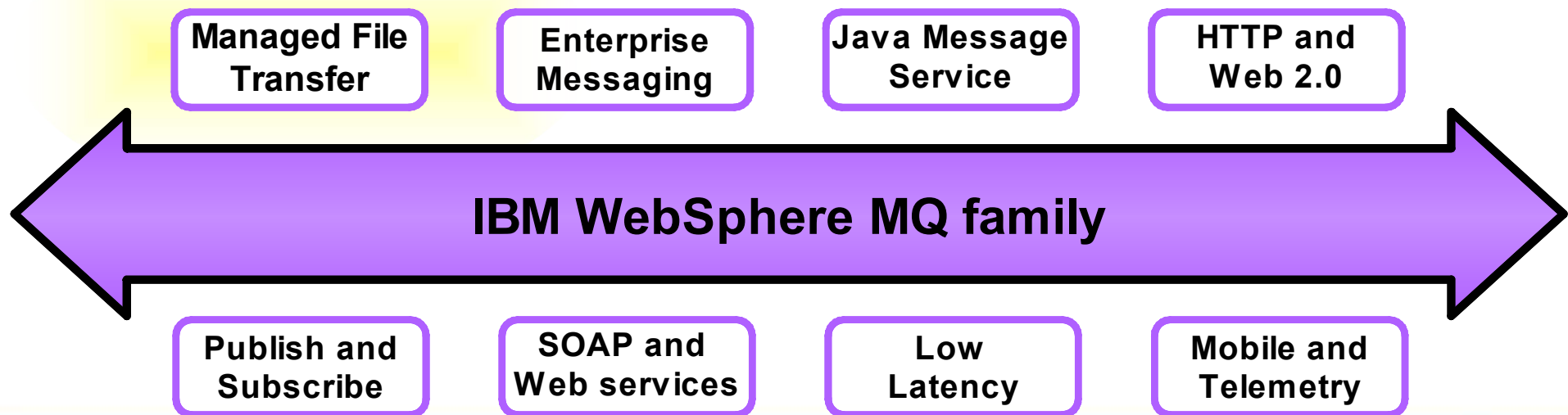
Planned initial platform Coverage & Support

- Core platforms targeted for initial release:
 - ☞ WebSphere MQ File Transfer Edition (Distributed)
 - AIX
 - Linux x86
 - Sun Solaris
 - HP-UX
 - Microsoft Windows
 - ☞ WebSphere MQ File Transfer Edition for z/OS
 - z/OS
- WebSphere MQ Versions supported:
 - ☞ V6.0
 - ☞ V7.0



Consolidated Transport Backbone

- Combined solution for transferring messages and files via a single consolidated infrastructure
 - ▶ Reducing operational costs through synergies and lowering skills requirements
- A Managed File Transfer solution that can be leveraged in SOA
 - ▶ A one-two punch – Solve today's file problem while building a foundation for the future
 - ▶ Single Universal Connectivity solution bringing together file- message- service- and event-oriented applications and Web 2.0 traffic
 - ▶ Apply ESB capabilities to file data – transformation, mediation, content-based routing



Why IBM?

Over 15 years of proven experience

- Over 15 years leadership 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 the messaging and ESB space
- New WebSphere MQ family products
- Regular enhancements, updates and new releases

Results reported from actual IBM WebSphere MQ implementations

