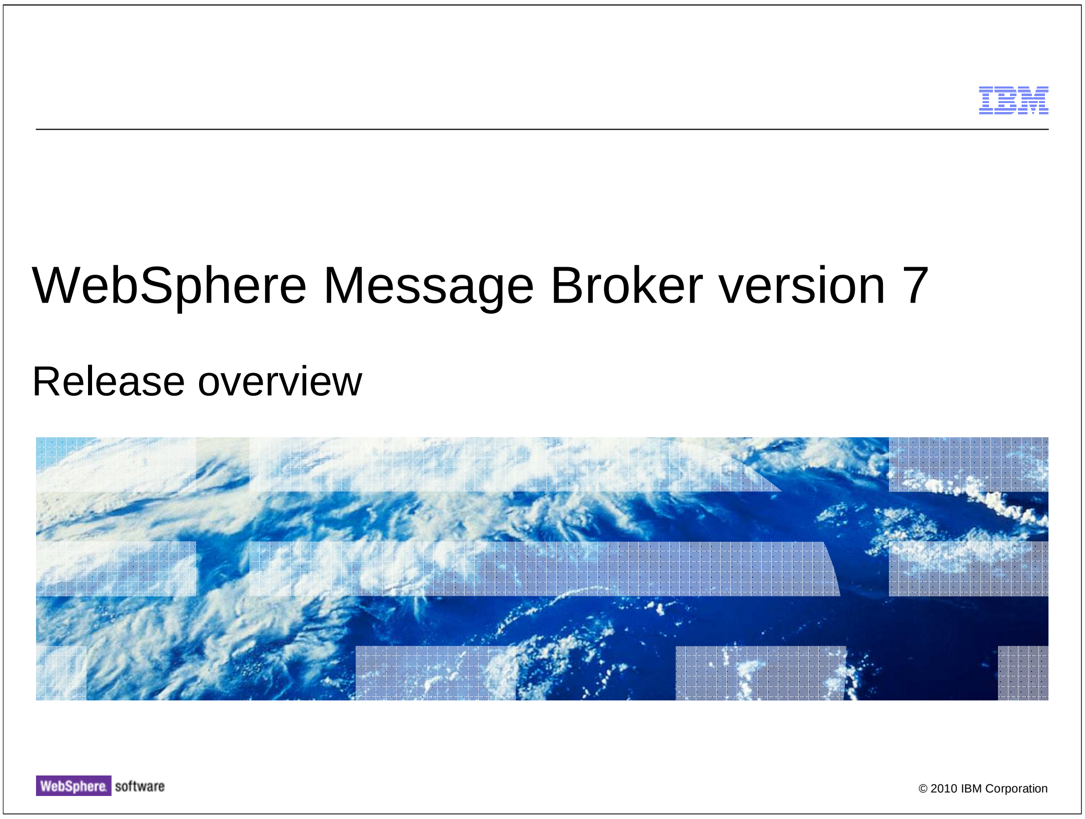


WebSphere Message Broker version 7

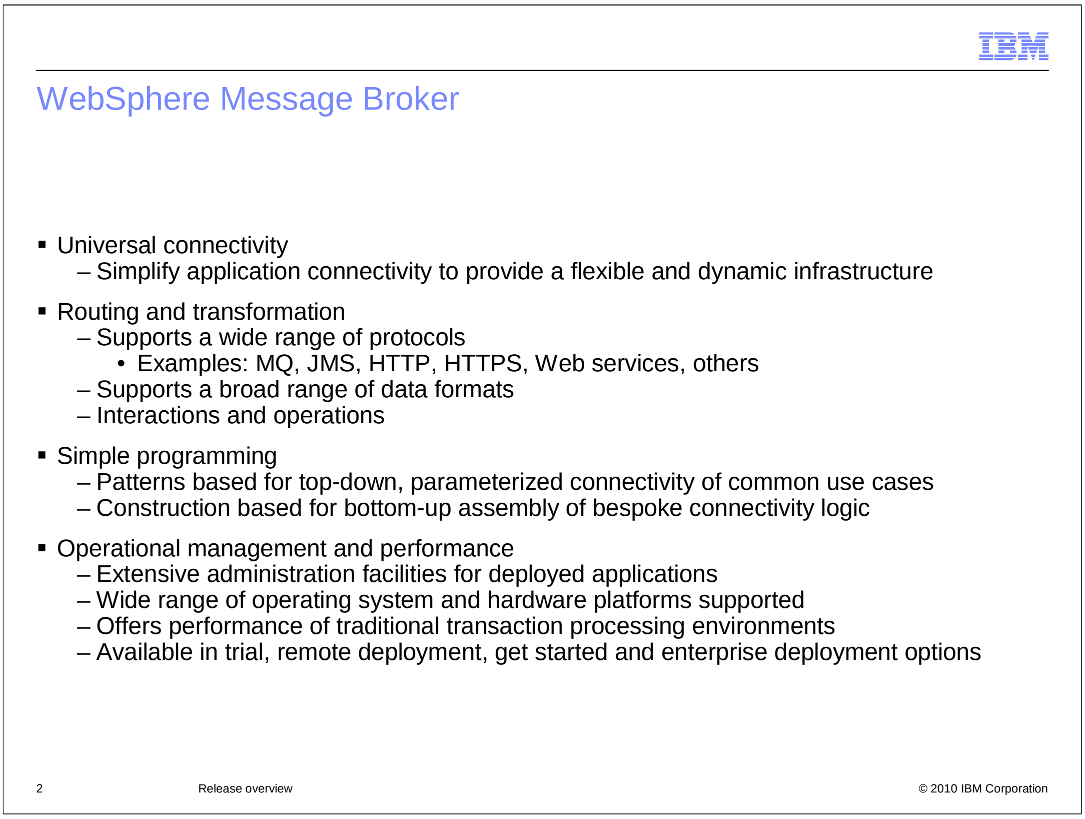
Release overview

WebSphere software

© 2010 IBM Corporation

This session will provide a complete overview of WebSphere Message Broker version 7. It will describe the main changes in this new version at a high level, and will go on to cover most of the new application and management functions in version 7.

Subsequent modules will go into each item in more detail.



WebSphere Message Broker

- Universal connectivity
  - Simplify application connectivity to provide a flexible and dynamic infrastructure
- Routing and transformation
  - Supports a wide range of protocols
    - Examples: MQ, JMS, HTTP, HTTPS, Web services, others
  - Supports a broad range of data formats
  - Interactions and operations
- Simple programming
  - Patterns based for top-down, parameterized connectivity of common use cases
  - Construction based for bottom-up assembly of bespoke connectivity logic
- Operational management and performance
  - Extensive administration facilities for deployed applications
  - Wide range of operating system and hardware platforms supported
  - Offers performance of traditional transaction processing environments
  - Available in trial, remote deployment, get started and enterprise deployment options

2 Release overview © 2010 IBM Corporation

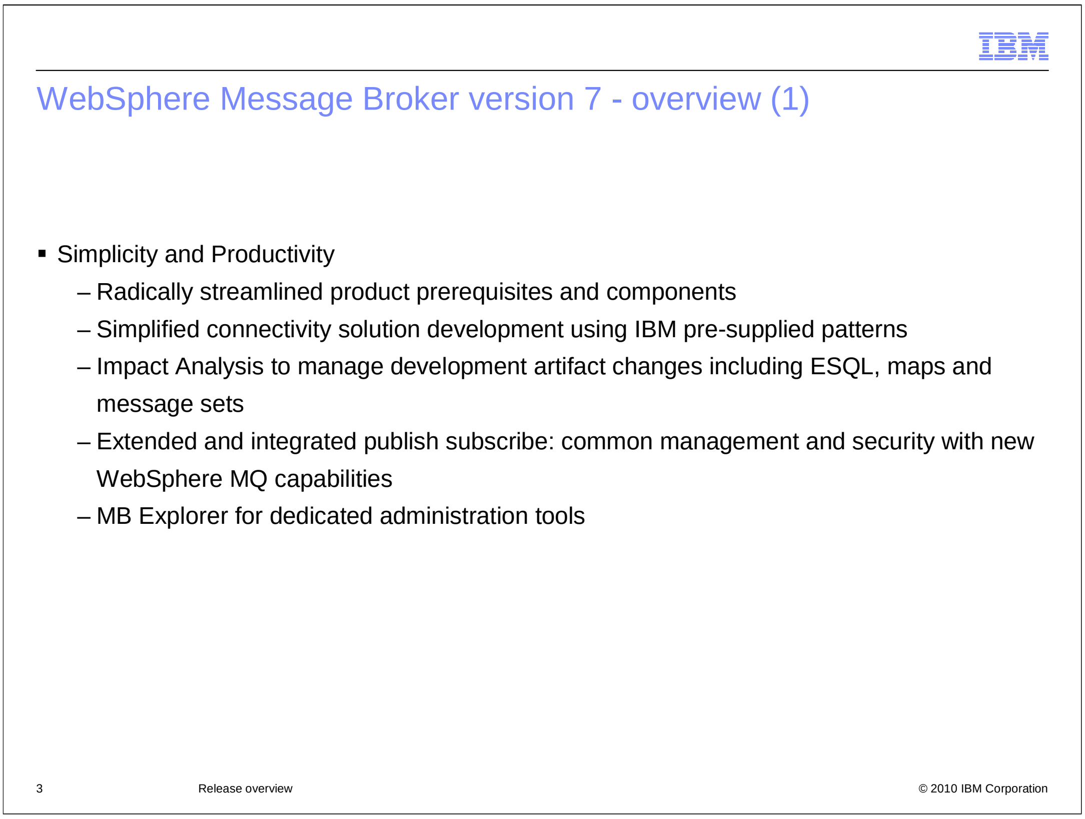
This slide serves as a summary and recap of the primary characteristics of WebSphere Message Broker. Whilst this presentation will not describe each bullet in detail, it is important to observe that it describes the two primary classes of user of Message Broker, namely application developers, and operations and administration.

First, WebSphere Message Broker is a connectivity tool. Its primary purpose is to enable connections between different environments, which might use different network protocols, have different interface styles, and which have different formats of data representation. These characteristics describe the problem that WebSphere Message Broker is addressing, and it is important to appreciate that this continues to be the case in this new version 7, and will continue into the future. For example, since the original implementation using WebSphere MQ messages, support has been added for HTTP and HTTPS, SOAP messages, JMS providers, File and FTP interfaces, and native TCP/IP interfaces. In addition, direct connections to application environments such as SAP, Siebel and PeopleSoft is also provided as base components.

These connections will transport data in a variety of data formats such as binary (for example in C or COBOL format), XML, or comma-separate variables. A variety of industry or application formats like SWIFT in the finance industry, or HL7 in the health industry are also supported.

Second, WebSphere Message Broker allows you to develop integration applications in many ways, and using many programming styles and languages. These range from the long-standing ESQL language, through Java and XSL transformation, with more recent additions like WebSphere Transformation Extender and PHP. The choice of language for writing applications is typically a factor of local considerations, such as skill levels in particular programming styles.

Integration applications are created using the core components of message flows, message sets and the internal message tree. Again, these are long-standing structures within WebSphere Message Broker, and continue to be the basis for all future product developments. Application development can be achieved with both the existing



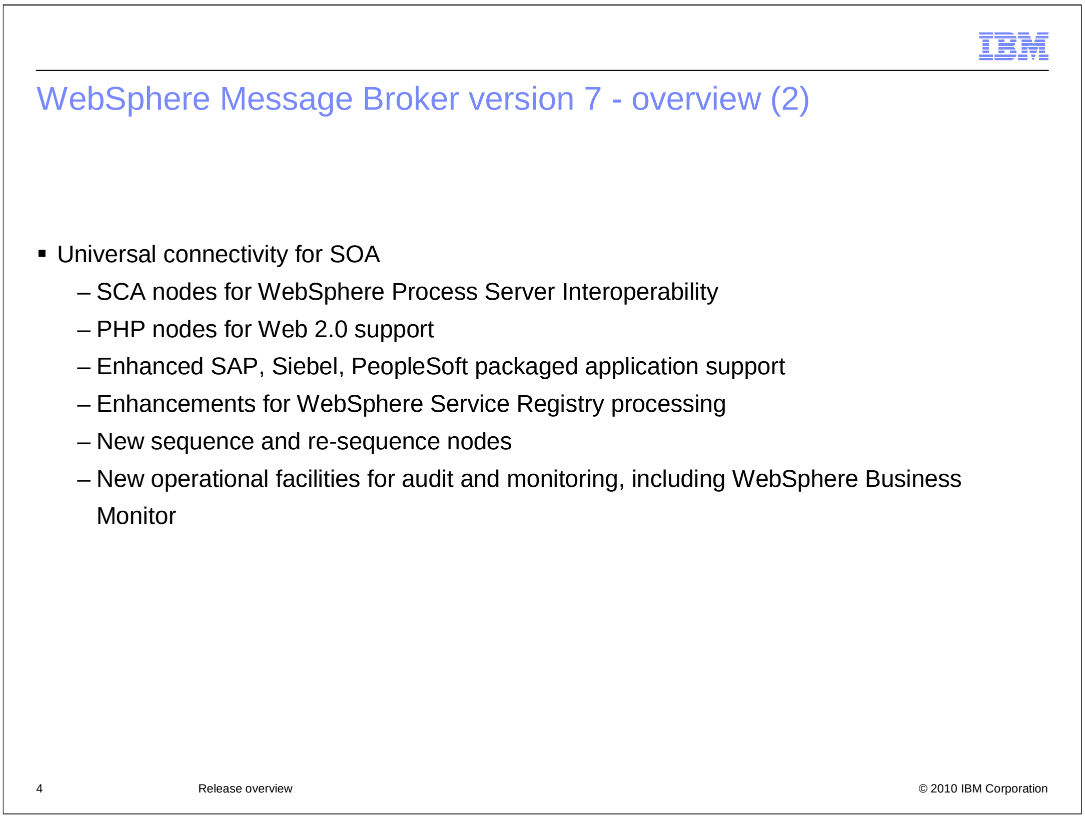
WebSphere Message Broker version 7 - overview (1)

- Simplicity and Productivity
  - Radically streamlined product prerequisites and components
  - Simplified connectivity solution development using IBM pre-supplied patterns
  - Impact Analysis to manage development artifact changes including ESQL, maps and message sets
  - Extended and integrated publish subscribe: common management and security with new WebSphere MQ capabilities
  - MB Explorer for dedicated administration tools

3 Release overview © 2010 IBM Corporation

Message Broker Version 7 delivers a range of new function, developed using several major themes.

First, a key theme in version 7 is the objective to simplify the product architecture, and to enable applications to be developed, tested and deployed more quickly and reliably. Version 7 has removed three major components, namely the Configuration Manager (referred to as the Config Manager in future), the broker database, and the “User Name Server” in publish/subscribe environments. Version 7 has introduced a new way of creating new applications, using a patterns-based approach. The version 7 toolkit has been simplified, and now contains functions to enable impact analysis of changed artifacts. Message Broker version 7 takes full advantage of the WebSphere MQ version 7 publish/subscribe function, and extends the MQ management function with the new “MB Explorer”.



WebSphere Message Broker version 7 - overview (2)

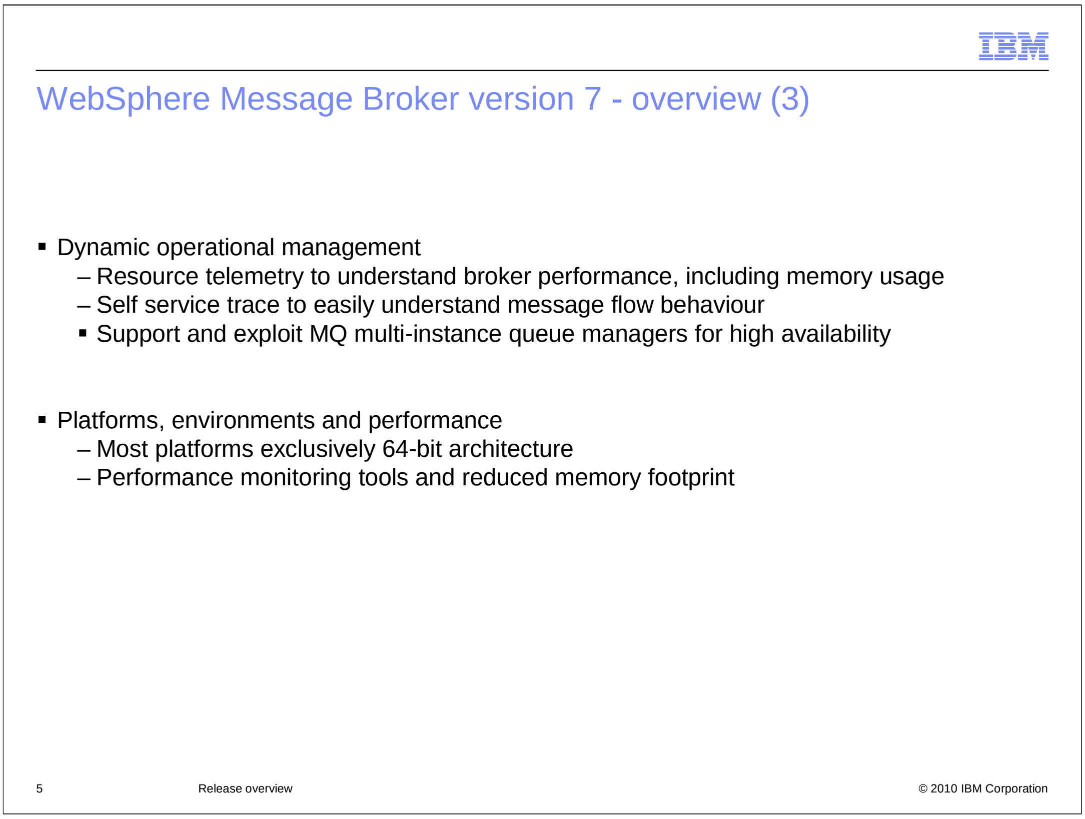
- Universal connectivity for SOA
  - SCA nodes for WebSphere Process Server Interoperability
  - PHP nodes for Web 2.0 support
  - Enhanced SAP, Siebel, PeopleSoft packaged application support
  - Enhancements for WebSphere Service Registry processing
  - New sequence and re-sequence nodes
  - New operational facilities for audit and monitoring, including WebSphere Business Monitor

4 Release overview © 2010 IBM Corporation

Secondly, in support of the broad service oriented architecture principles, connectivity to WebSphere Process Server using the Service Component Architecture, or SCA, has been introduced.

Web 2.0 support is addressed with new a PHP (*pronounced "P H P")* compute node, and the existing application adapters receive new function to enhance their deployment options.

New sequencing nodes for handling messages that require a strict ordering are provided, and finally there are new facilities to provide monitoring data for WebSphere Business Monitor.



WebSphere Message Broker version 7 - overview (3)

- Dynamic operational management
  - – Resource telemetry to understand broker performance, including memory usage
  - – Self service trace to easily understand message flow behaviour
  - ▪ Support and exploit MQ multi-instance queue managers for high availability

- Platforms, environments and performance
  - – Most platforms exclusively 64-bit architecture
  - – Performance monitoring tools and reduced memory footprint

5 Release overview © 2010 IBM Corporation

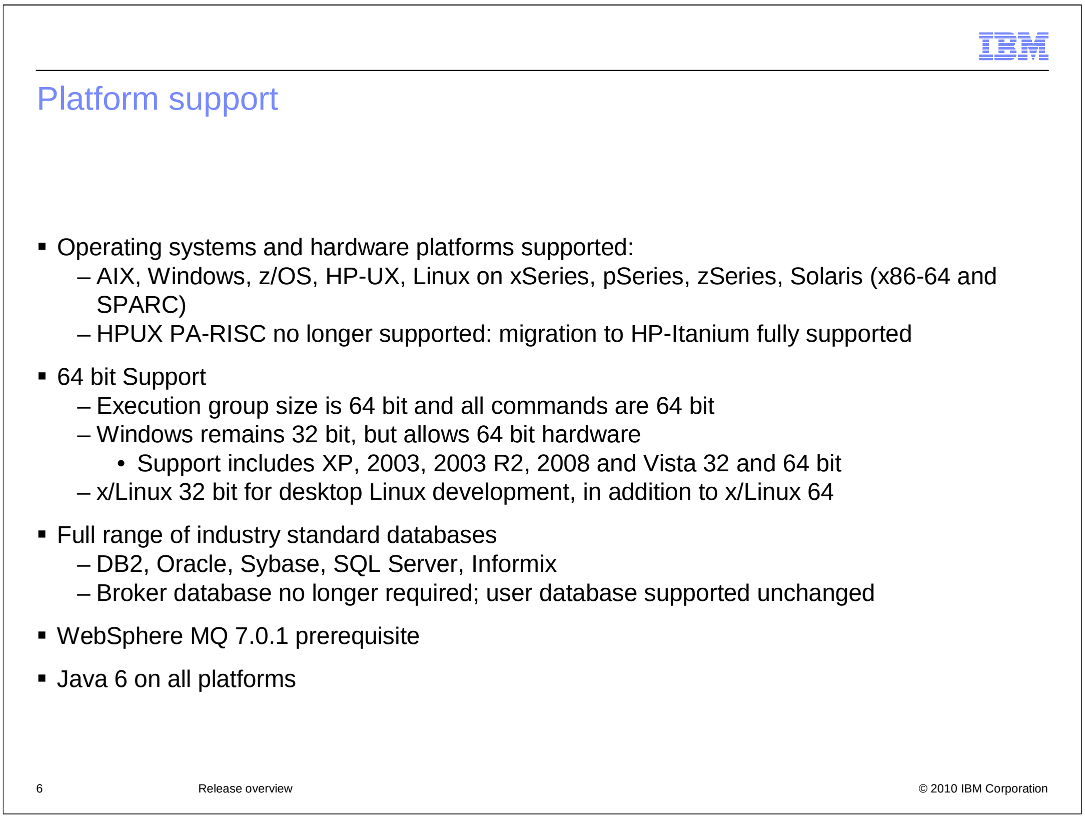
Version 7 introduces several new functions to improve administration and operational management. First, version 7 now provides tools to assist with the internals of the broker’s execution groups. This includes data such as memory usage and handle usage.

Secondly, additional trace facilities are provided to assist in diagnosis of application problems in a message flow.

And thirdly, Message Broker version 7 can now take advantage of the new standby queue managers introduced in WebSphere MQ 7.0.1.

All unix platforms now run only in 64-bit mode, and 32-bit execution groups are no longer supported on unix platforms. Windows platforms remain 32-bit.

Finally, this new version introduces additional performance monitoring tools, in addition to making reductions in the footprint of the installed components.



Platform support

- Operating systems and hardware platforms supported:
  - AIX, Windows, z/OS, HP-UX, Linux on xSeries, pSeries, zSeries, Solaris (x86-64 and SPARC)
  - HPUX PA-RISC no longer supported: migration to HP-Itanium fully supported
- 64 bit Support
  - Execution group size is 64 bit and all commands are 64 bit
  - Windows remains 32 bit, but allows 64 bit hardware
    - Support includes XP, 2003, 2003 R2, 2008 and Vista 32 and 64 bit
  - x/Linux 32 bit for desktop Linux development, in addition to x/Linux 64
- Full range of industry standard databases
  - DB2, Oracle, Sybase, SQL Server, Informix
  - Broker database no longer required; user database supported unchanged
- WebSphere MQ 7.0.1 prerequisite
- Java 6 on all platforms

This slide describes the platforms and pre-requisite software levels of related software products that are required for Message Broker version 7.

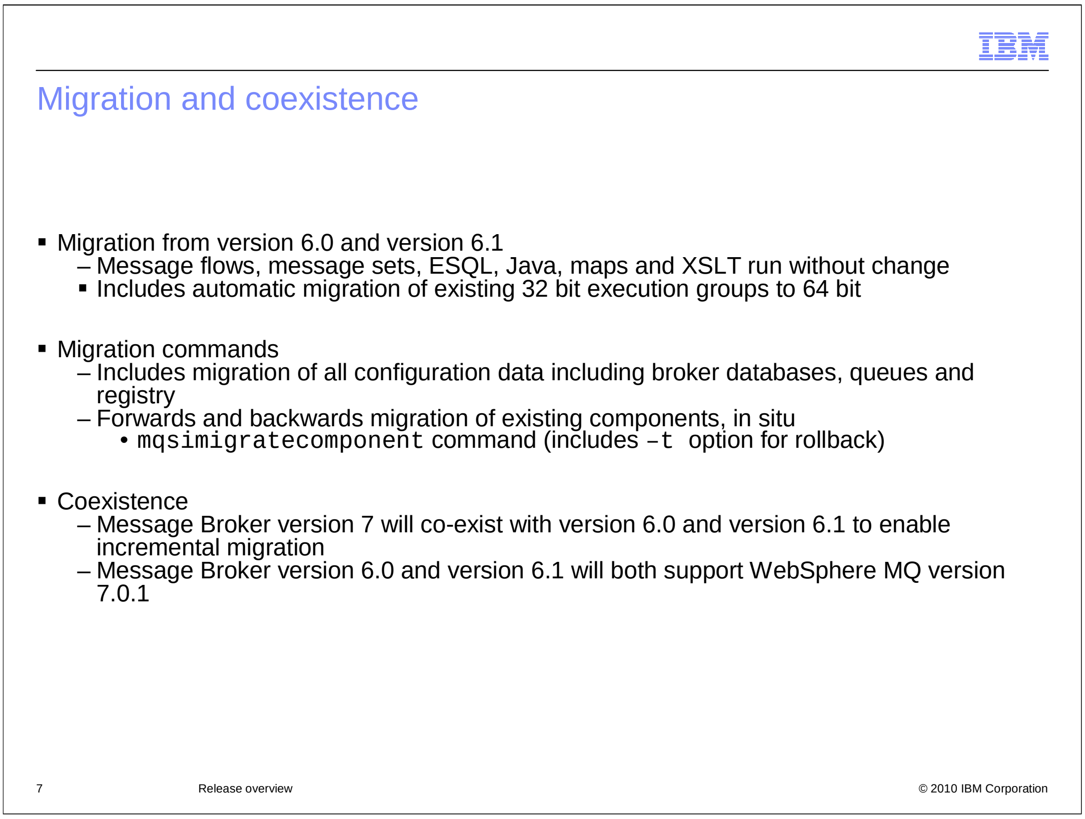
The majority of the operating systems that are supported are unchanged from Message Broker Version 6.1. However, if you plan to run Message Broker on the HP-UX platform on the PA-RISC hardware (*pronounced P A – Risk*), note that HP no longer market this platform. HP have moved to the Itanium platform, and support for Message Broker has therefore followed the same approach.

For all other hardware platforms, the support for Message Broker version 7 is unchanged, although this support is now almost exclusively restricted to 64-bit execution groups, including z/OS. The only exceptions to this are the Windows platform, where support is still restricted to 32-bit, and the 32-bit implementation for Linux on Intel hardware. On all other platforms, including z/OS, execution groups can only be 64-bit. This will typically mean that larger workloads can be accommodated, processing larger amounts of data, or perhaps processing larger messages.

The range of supported databases is the same as in earlier versions. However, a key difference in version 7 is that a broker database is no longer required. The list of databases shown here is for user data. The broker database removal is discussed in more detail in other IBM Education Assistant topics.

Message Broker Version 7 has a pre-requisite level of WebSphere MQ, which is version 7.0.1. This new version has been available since August 2009, and is required for some of the new and updated function in Message Broker version 7, for example publish/subscribe, and the new sequence nodes.

Support for Java 6 is provided on all platform, which gives improvements in the overall performance of the JVM, and its memory requirements.



Migration and coexistence

- Migration from version 6.0 and version 6.1
  - Message flows, message sets, ESQL, Java, maps and XSLT run without change
  - Includes automatic migration of existing 32 bit execution groups to 64 bit
- Migration commands
  - Includes migration of all configuration data including broker databases, queues and registry
  - Forwards and backwards migration of existing components, in situ
    - `mqsimigratecomponent` command (includes `–t` option for rollback)
- Coexistence
  - Message Broker version 7 will co-exist with version 6.0 and version 6.1 to enable incremental migration
  - Message Broker version 6.0 and version 6.1 will both support WebSphere MQ version 7.0.1

7 Release overview © 2010 IBM Corporation

Migration to Message Broker version 7 is provided from version 6.0 and 6.1. All migrated applications will run without change. Applications can be migrated directly from version 6.0; it is not necessary to migrate to version 6.1 first.

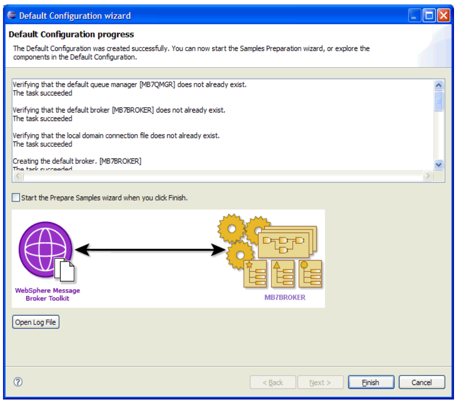
If your installation contains any 32-bit C plug-ins or parsers, then these will need to be recompiled to use a 64-bit compiler.

The migration process also includes migration of the Message Broker runtime components, principally the defined execution groups. If these are defined as 32-bit, then they are redefined as 64-bit in Message Broker version 7.

The “m-q-s-i-migrate-components” command is used to migrate broker environments from earlier versions of Message Broker to version 7. In order to perform the migration process, WebSphere MQ version 7.0.1 must be installed and available to both the old version of the Message Broker installation, and the Message Broker version 7 installation. This is explained in more detail in the Migration topic in the this IBM Education Assistant module.

Product simplification

- Primary components
  - Toolkit
  - Runtime
  - Message Broker Explorer
- File based deployment and configuration
  - Transient state (for example, aggregation) in WebSphere MQ
- Publish / subscribe scenarios
  - Function moved to WebSphere MQ
  - Content-based routing retained in Message Broker
- Changes in operation of Message Broker
  - Toolkit and Explorer can manage multiple brokers



Default Configuration progress

The Default Configuration was created successfully. You can now start the Samples Preparation wizard, or explore the components in the Default Configuration.

```
Verifying that the default queue manager [MB7QMGR] does not already exist.
The task succeeded

Verifying that the default broker [MB7BROKER] does not already exist.
The task succeeded

Verifying that the local domain connection file does not already exist.
The task succeeded

Creating the default broker. [MB7BROKER]
The task succeeded
```

☐ Start the Prepare Samples wizard when you click Finish.

Default Configuration wizard
WebSphere Message Broker Toolkit
MB7BROKER
Open Log File
< Back
Next >
Finish
Cancel

Message Broker version 7 has significantly reduced the number of components parts. In version 7, the two primary components are the Toolkit, for application development, and the Broker Runtime, for running applications. A third component, the Message Broker Explorer, is used for system administration. This will be referred to as “MB Explorer” from now on.

If you are familiar with earlier version of Message Broker, note that the Config Manager, Broker database, and User-Name Server are no longer used. The Rational Agent Controller, used for debugging in version 6.0, is also removed.

The Broker Database has been replaced with a file-based system, which makes the creation, backup and restore of the Broker components much simpler than in earlier versions. Broker artifacts such as message flows, message sets, and any other application components, are now stored directly within the Broker’s file system. Applications that require state information to be held, for example aggregation or the holding of routing information, use queue defined in WebSphere MQ.

Since the broker definitions are now stored directly in a file system, rather than a database, version 7 has introduced a command to backup and restore the broker definitions. The commands are “m-q-s-i-backup-broker” and m-q-s-i-restore-broker”. The backup command creates a zip-file of the broker file system. This command can be run online, whilst the broker is active. The restore command will re-create the broker from the zip-file; in this case, the target broker cannot be used whilst the restore command is active. These commands are described in the session which covers the broker database removal.

The removal of these key components has resulted in some changes in the way that certain functions operate.

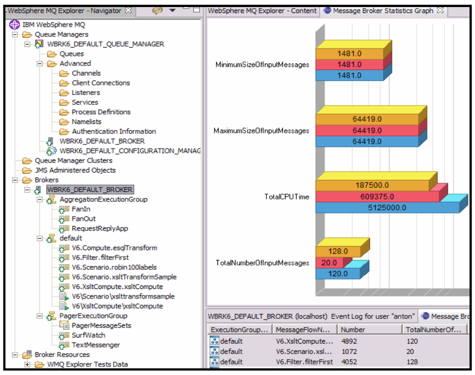
First, the removal of the configuration manager has removed the concept of the broker domain, where multiple brokers were managed by a single configuration manager. In version 7, each Broker runtime instance is entirely separate, and can be managed by any

Message Broker explorer for operations and administration

- Single administration for WebSphere MQ and Message Broker
  - – MB Explorer is a full function, small footprint, stand-alone tool for operational staff
    - Allows MQ and Message Broker artifacts to be managed in single console

- New broker administration features
  - – Broker sets
  - – Deployment to multiple brokers
  - – Deployment from file systems

- Graphical performance monitor


WebSphere MQ Explorer - Navigator
IBM WebSphere MQ
Queue Managers
WBRK6_DEFAULT_QUEUE_MANAGER
Queues
Advanced
Channels
Client Connections
Listeners
Services
Process Definitions
Namelists
Authentication Information
WBRK6_DEFAULT_BROKER
WBRK6_DEFAULT_CONFIGURATION_MANAG
Queue Manager Clusters
JMS Administered Objects
Brokers
WBRK6_DEFAULT_BROKER
AggregationExecutionGroup
FanIn
FanOut
RequestReplyApp
default
V6.Compute.esqlTransform
V6.Filter.filterFirst
V6.Scenario.robin100labels
V6.Scenario.xsltTransformSample
V6.XsltCompute.xsltCompute
V6\Scenario\xslttransformsample
V6\XsltCompute\xsltCompute
PagerExecutionGroup
PagerMessageSets
SurfWatch
TextMessenger
Broker Resources
WMQ Explorer Tests Data
WebSphere MQ Explorer - Content
Message Broker Statistics Graph
MinimumSizeOfInputMessages
1481.0
1481.0
1481.0
MaximumSizeOfInputMessages
64419.0
64419.0
64419.0
TotalCPUTime
187500.0
609375.0
5125000.0
TotalNumberOfInputMessages
128.0
20.0
120.0
WBRK6_DEFAULT_BROKER (localhost) Event Log for user "anton"
Message Bro

ExecutionGroup...	MessageFlowN...	Number	TotalNumberOf...
default	V6.XsltCompute...	4892	120
default	V6.Scenario.xsl...	1072	20
default	V6.Filter.filterFirst	4052	128

© 2010 IBM Corporation

The Message Broker Explorer is a new component which provides a user interface for operational and administration. It is typically known as the “MB Explorer”, and is plug-in to the Eclipse-based MQ Explorer provided with WebSphere MQ version 7.0.1. Version 7.0.1 or later is required to enable the MB Explorer; earlier versions, including version 7.0, are not sufficient.

The MB Explorer can be installed on a Windows or Linux desktop, and can be installed without any other components. The only pre-requisite software is the MQ Explorer, and WebSphere MQ itself does not need to be installed, unless local brokers are going to be defined. The MB Explorer can connect and manage brokers that are defined on remote systems.

The MB Explorer provides full management capability for all broker components. Brokers can be started and stopped, execution groups can be created, started, stopped and deleted, and applications can be deployed, started, stopped and removed. Application can be deployed to several brokers at the same time, using a single deployment operation. The bar-files that are deployed can be imported into the MB Explorer workspace, and deployed from there, or can be deployed directly from a separate file-system by dragging and dropping directly onto the target execution group. Finally, the MB Explorer can connect to a code repository, such as Rational Clearcase, CVS or SVN. Bar-files can be stored in these repositories and used by the MB Explorer for deployment to the broker.

Additionally, the MB Explorer can be used to create local brokers on the Windows or Linux desktop system.

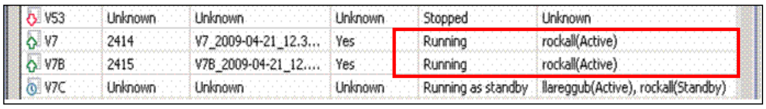
The MB Explorer can also be used to define and manage broker configurable services. New configurable services can be created from a set of templates, and can be copied to create further new entries. These can then be applied to any of the broker instances. Similarly, the Web services policy sets can be defined and managed in the same way.

High availability using MQ 7.0.1 standby queue managers

- Message Broker exploits MQ 7.0.1 multi-instance queue manager capability

- Active and stand-by queue managers
  - Start multiple instances of a queue manager on different machines
  - One is “active” instance; other is “standby” instance
  - Shared data is held in shared networked storage but owned by active instance

- Exploitation by Message Broker
  - If standby broker is not running; MQ will restart broker once MQ recovery complete
  - If standby broker is running, initialization will complete MQ recovery complete



V53	Unknown	Unknown	Unknown	Stopped	Unknown
V7	2414	V7_2009-04-21_12.3...	Yes	Running	rockall(Active)
V7B	2415	V7B_2009-04-21_12....	Yes	Running	rockall(Active)
V7C	Unknown	Unknown	Unknown	Running as standby	llareggub(Active), rockall(Standby)

© 2010 IBM Corporation

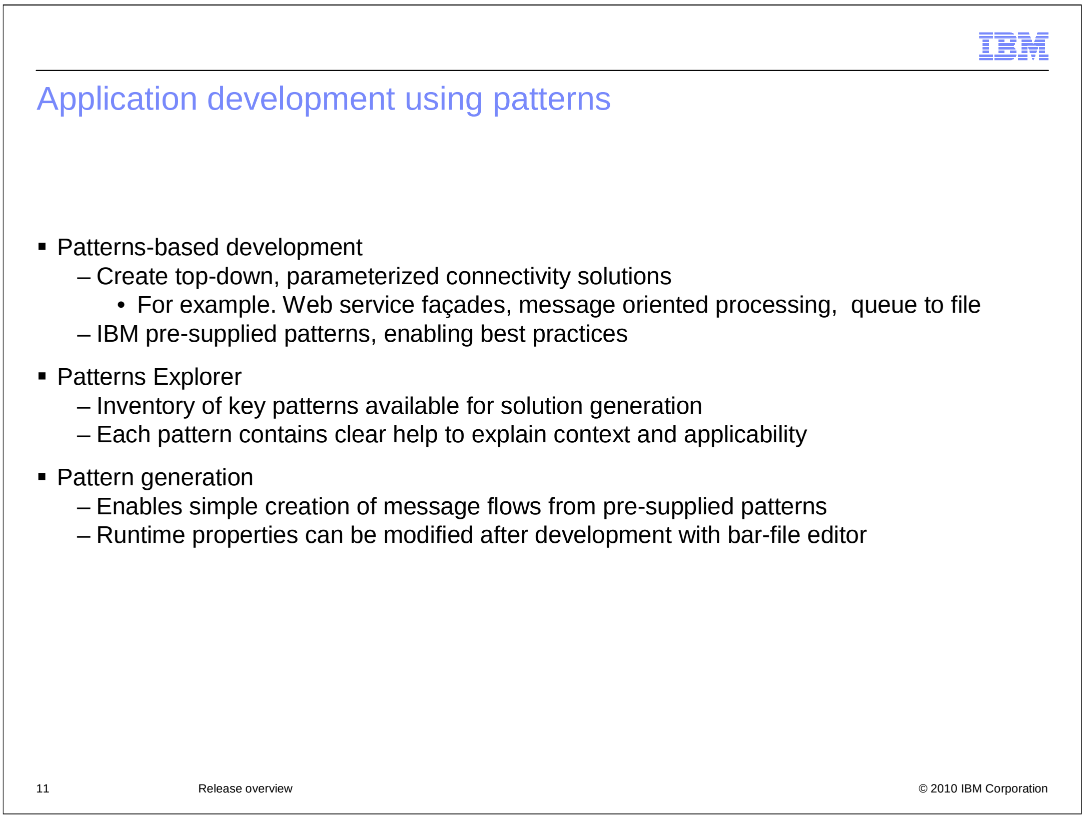
WebSphere MQ version 7.0.1 has introduced a new facility addressing high availability. This feature provides a multi-instance queue manager, where a stand-by queue manager is used to provide a backup system for the primary queue manager.

In some scenarios, this can replace the requirement for alternative hardware-based high availability approaches.

So, with WebSphere MQ version 7.0.1, you can define a queue manager to be a multi-instance queue manager. This means that it will have a active queue manager, and a standby queue manager. These two queue managers are connected by using a shared storage device, and will monitor each other’s health with a heart beat. In the event the active queue manager fails, the standby queue manager will become fully initialized, and will take over processing.

WebSphere Message Broker version 7 will complement this facility by providing an active and standby broker. As with MQ, these will be connected using a shared storage device. The active broker will normally process all requests. If the underlying queue manager detects a problem, then the standby queue manager will takeover, the active broker will be terminated, and the standby broker will complete initialization and takeover processing of requests.

In addition, when used in combination with SAP, the SAP input node is now capable of switching between broker instances to provide high availability in a SAP scenario.



Application development using patterns

- Patterns-based development
  - Create top-down, parameterized connectivity solutions
    - For example. Web service façades, message oriented processing, queue to file
  - IBM pre-supplied patterns, enabling best practices
- Patterns Explorer
  - Inventory of key patterns available for solution generation
  - Each pattern contains clear help to explain context and applicability
- Pattern generation
  - Enables simple creation of message flows from pre-supplied patterns
  - Runtime properties can be modified after development with bar-file editor

Message Broker version 7 has introduced a major new function to assist with application development. Constructing new applications based on a set of supplied patterns will make the creation of such applications much more reliable and repeatable.

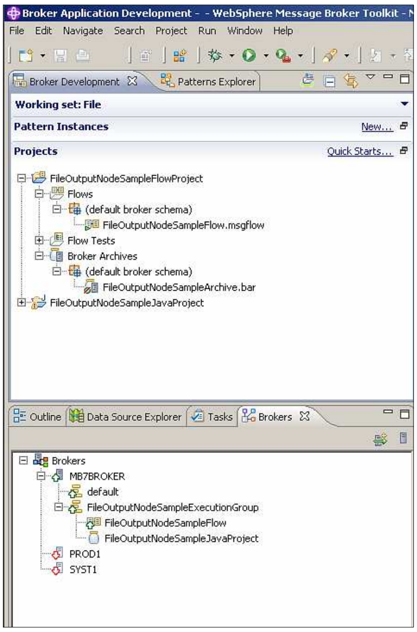
Patterns development allows you to create a new message broker application, based on a set of supplied templates. Different patterns are provided for a variety of different application styles, from simple messaging, Web service facades, request-reply, queue to file, and many others.

These patterns are provided in the patterns explorer, a new component of the Message Broker toolkit. Each pattern contains a full description of the target application, with clear help to explain the context and how to create the application.

When you have selected the most appropriate pattern, the pattern generation facility allows you to specify certain parameters such as the inputs and outputs to the new application. You can also extend the generated application to enable it for your specific application requirements.

Toolkit enhancements and changes

- Impact analysis to manage asset change
- Simple compile, test, fix life cycle
- Mapping enhancements
- More XPATH expression editor functions
- Comprehensive samples gallery
- Enhanced barfile editor


Broker Application Development - - WebSphere Message Broker Toolkit - N
File Edit Navigate Search Project Run Window Help
Broker Development
Patterns Explorer
Working set: File
Pattern Instances
New...
Projects
Quick Starts...
FileOutputNodeSampleFlowProject
Flows
(default broker schema)
FileOutputNodeSampleFlow.msgflow
Flow Tests
Broker Archives
(default broker schema)
FileOutputNodeSampleArchive.bar
FileOutputNodeSampleJavaProject
Outline
Data Source Explorer
Tasks
Brokers
Brokers
MB7BROKER
default
FileOutputNodeSampleExecutionGroup
FileOutputNodeSampleFlow
FileOutputNodeSampleJavaProject
PROD1
SYST1

The Message Broker Toolkit has been improved in many areas in version 7, and this slide discusses some of the more significant changes.

First, the toolkit now has only one primary Eclipse perspective, the application development perspective. The administration perspective that was used in all earlier versions, has been removed, with most of the function being replaced with the “MB Explorer”. The application development perspective has been extended to allow you to connect to brokers, and to deploy applications to these brokers, using the new “View Brokers” pane, a shown on this slide.

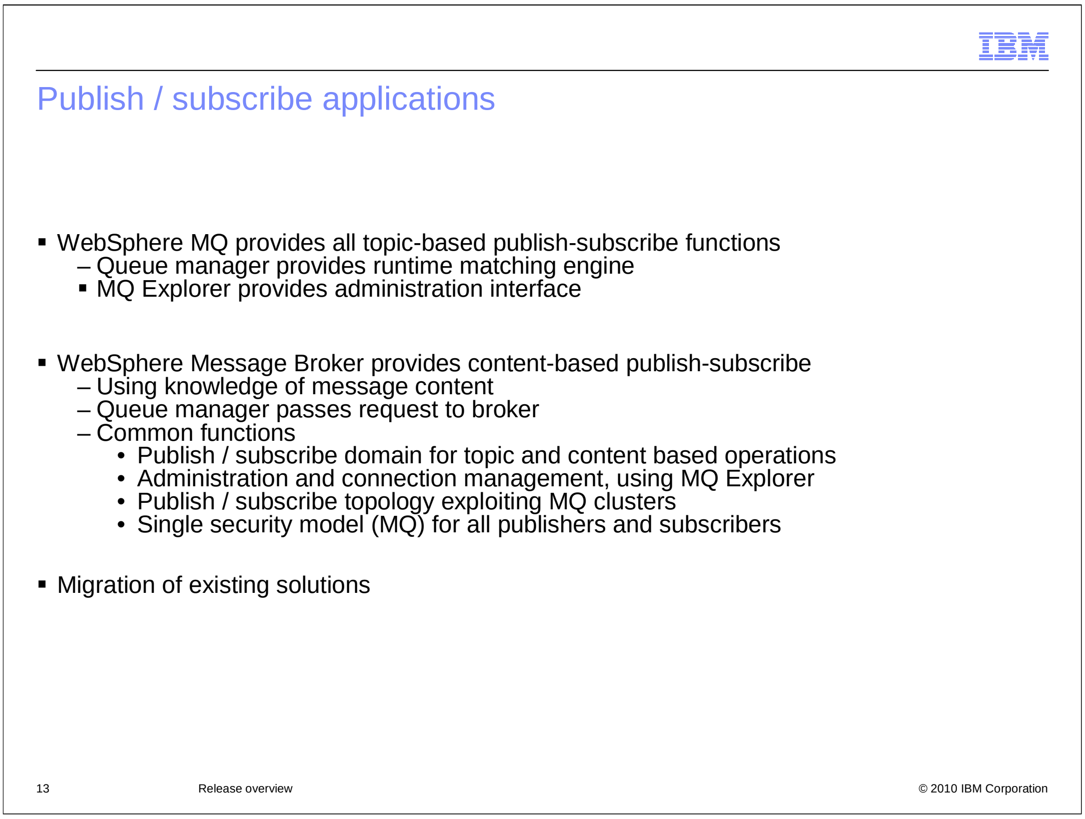
In addition, it is now no longer necessary to create a bar-file to deploy an application. You can drag the message flow from the project navigator directly onto the target execution group. The toolkit will generate a bar-file during this process, and this is stored within the project of the message flow.

Secondly, the toolkit now provides an impact analysis tool, enabling you to identify and manage any broker artifacts that are affected by changing an item in the toolkit workspace. These items cover ESQL files, maps, XML data, message set definitions, message flow and message nodes.

Next, the mapping node has a several improvements, covering access to DB2 and Oracle stored procedures, and provision of an intelligent auto-map function and rules-based mapping.

The XPATH (*pronounced X-path*) editor has been improved, the bar-file editor now provides better filtering, navigation and over-ride capability, and the samples gallery has been enhanced.

And finally, improvements have been made to the bar-file editor, enabling deployment properties to be over-ridden and deployed more easily.



Publish / subscribe applications

- WebSphere MQ provides all topic-based publish-subscribe functions
  - Queue manager provides runtime matching engine
  - MQ Explorer provides administration interface
- WebSphere Message Broker provides content-based publish-subscribe
  - Using knowledge of message content
  - Queue manager passes request to broker
  - Common functions
    - Publish / subscribe domain for topic and content based operations
    - Administration and connection management, using MQ Explorer
    - Publish / subscribe topology exploiting MQ clusters
    - Single security model (MQ) for all publishers and subscribers
- Migration of existing solutions

13 Release overview © 2010 IBM Corporation

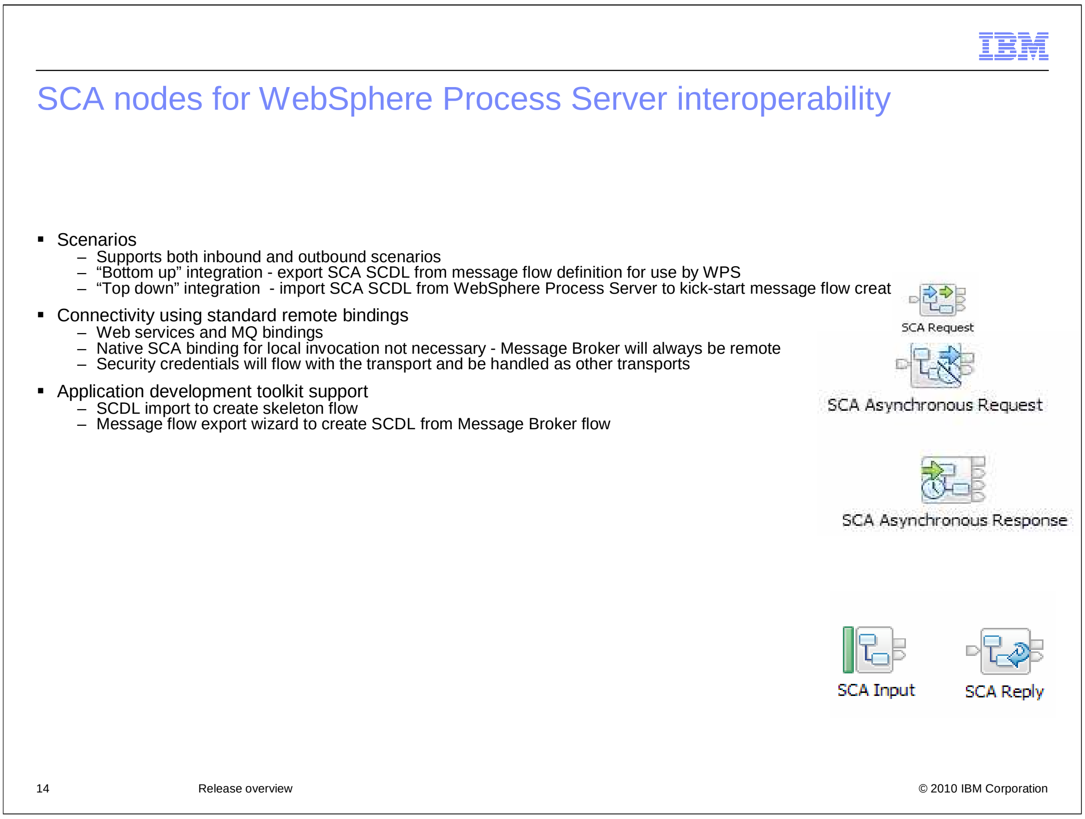
Message Broker version 7 has seen some significant changes in the implementation of publish / subscribe applications. The most important change is that the majority of all publish / subscribe applications are now provided by WebSphere MQ. As mentioned earlier, this must be using version 7.0.1.

WebSphere MQ now provides a full-function publish / subscribe capability, including management of all subscriptions, use of MQ Explorer for administration, and use of the MQ security tools access control. All topic-based publish / subscribe applications should now been written to use only MQ, and the management of such applications, subscriptions and retained publications are done using only the MQ queue manager.

If you want to extend this scenario by adding publications which need access to the content of published messages, then this is known as content-based publish / subscribe, and is managed by Message Broker. If the queue manager determines that a particular publication needs access to the content of the incoming message, it will pass the request to the broker. The broker will perform the appropriate matching algorithms, before passing the result back to the queue manager for final publishing to the subscriber.

This extended publish / subscribe scenario still uses the MQ facilities for management and administration, providing a common set of tools across MQ and Message Broker.

Migration of publish / subscribe applications is straightforward, and all existing subscriptions can be retained across the migration.



SCA nodes for WebSphere Process Server interoperability

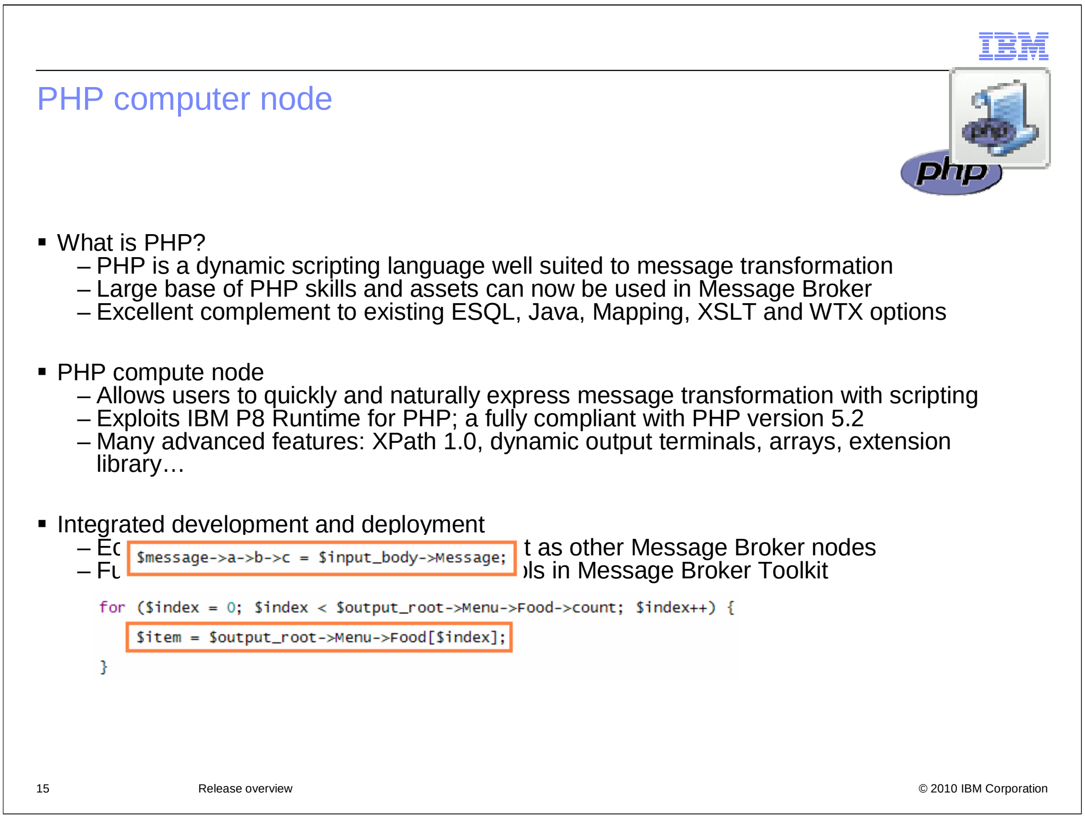
- Scenarios
  - Supports both inbound and outbound scenarios
  - “Bottom up” integration - export SCA SCDL from message flow definition for use by WPS
  - “Top down” integration - import SCA SCDL from WebSphere Process Server to kick-start message flow creat
- Connectivity using standard remote bindings
  - Web services and MQ bindings
  - Native SCA binding for local invocation not necessary - Message Broker will always be remote
  - Security credentials will flow with the transport and be handled as other transports
- Application development toolkit support
  - SCDL import to create skeleton flow
  - Message flow export wizard to create SCDL from Message Broker flow

14 Release overview © 2010 IBM Corporation

One of the major new items in version 7 in the area of connectivity is the support provided for the Service Component Architecture, more commonly referred to as SCA (*pronounced S C A*). Message Broker version 7 introduces five new nodes, called SCA nodes. These are designed to make connectivity with WebSphere Process Server very easy, and to enable simple interaction between message broker applications and process server applications.

For example by using the SCA request node, a message flow can invoke a process server application, by populating the properties of the SCA request. The developer of the message flow still uses the techniques that are familiar within the message broker environment, and does not need to become familiar with the terminology of the WebSphere Process Server environment. Similarly, the process server application can invoke a message broker flow by invoking an SCA component. The corresponding invoked message flow is initiated with an SCA input node, and terminated with an SCA reply node.

Hence, using the four SCA nodes, connectivity in both directions is possible. An application running in WebSphere Process Server can invoke a service running in Message Broker. Conversely, a Message Broker application can invoke a service in WebSphere Process Server. These invocations are achieved by using one or more of the new SCA nodes shown here.



IBM

PHP computer node

- What is PHP?
  - PHP is a dynamic scripting language well suited to message transformation
  - Large base of PHP skills and assets can now be used in Message Broker
  - Excellent complement to existing ESQL, Java, Mapping, XSLT and WTX options
- PHP compute node
  - Allows users to quickly and naturally express message transformation with scripting
  - Exploits IBM P8 Runtime for PHP; a fully compliant with PHP version 5.2
  - Many advanced features: XPath 1.0, dynamic output terminals, arrays, extension library…
- Integrated development and deployment
  - Ec t as other Message Broker nodes
  - Fu ›ls in Message Broker Toolkit

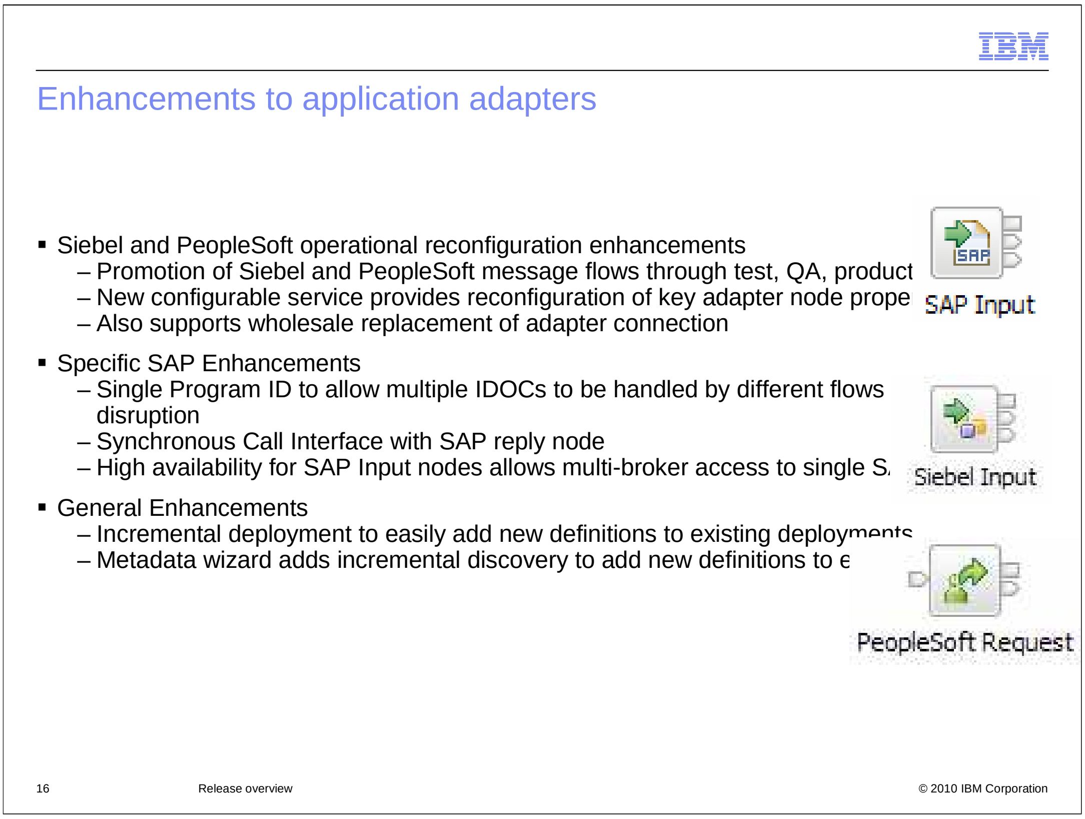
```
$message->a->b->c = $input_body->Message;
```

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

© 2010 IBM Corporation

An increasingly common language for Web developers is the PHP scripting language. Support for PHP was introduced in Message Broker version 6.1 on the Windows platform. Version 7 has now extended this support to all unix platforms.

This facility extends the range of transformation options.



Enhancements to application adapters

- Siebel and PeopleSoft operational reconfiguration enhancements
  - Promotion of Siebel and PeopleSoft message flows through test, QA, product
  - New configurable service provides reconfiguration of key adapter node prope
  - Also supports wholesale replacement of adapter connection
- Specific SAP Enhancements
  - Single Program ID to allow multiple IDOCs to be handled by different flows disruption
  - Synchronous Call Interface with SAP reply node
  - High availability for SAP Input nodes allows multi-broker access to single S.
- General Enhancements
  - Incremental deployment to easily add new definitions to existing deployments
  - Metadata wizard adds incremental discovery to add new definitions to e

The WebSphere Adapters for SAP, Siebel and PeopleSoft were introduced in Message Broker version 6.1. One of the fixpack releases of version 6.1 added capability allowing the SAP adapter properties to be configured outside the main adapter definition, using configurable services. Properties such as the host name, client ID, system number, user ID and password can now be specified in a configurable service.

Message Broker version 7 has introduced some enhancements to these adapters. First, the Siebel and PeopleSoft adapters now provide the ability to set properties using configurable services, bringing equivalent function to the SAP adapter.

Secondly, the SAP adapter itself has new function to support the synchronous call interface, with the new SAP reply node. Next, the “Single Program ID” facility allows multiple SAP IDOCs to be handled by different message flows. High Availability for SAP is provided by allowing more than one broker instance to connect to the same SAP system. Integrity is maintained by using a common queue manager to control the flow processing state.

General enhancements include the incremental definition of existing adapter definitions. This will allow new or changed definitions to an existing adapter, whilst not making changes to existing components.

Integration with WebSphere Business Monitor

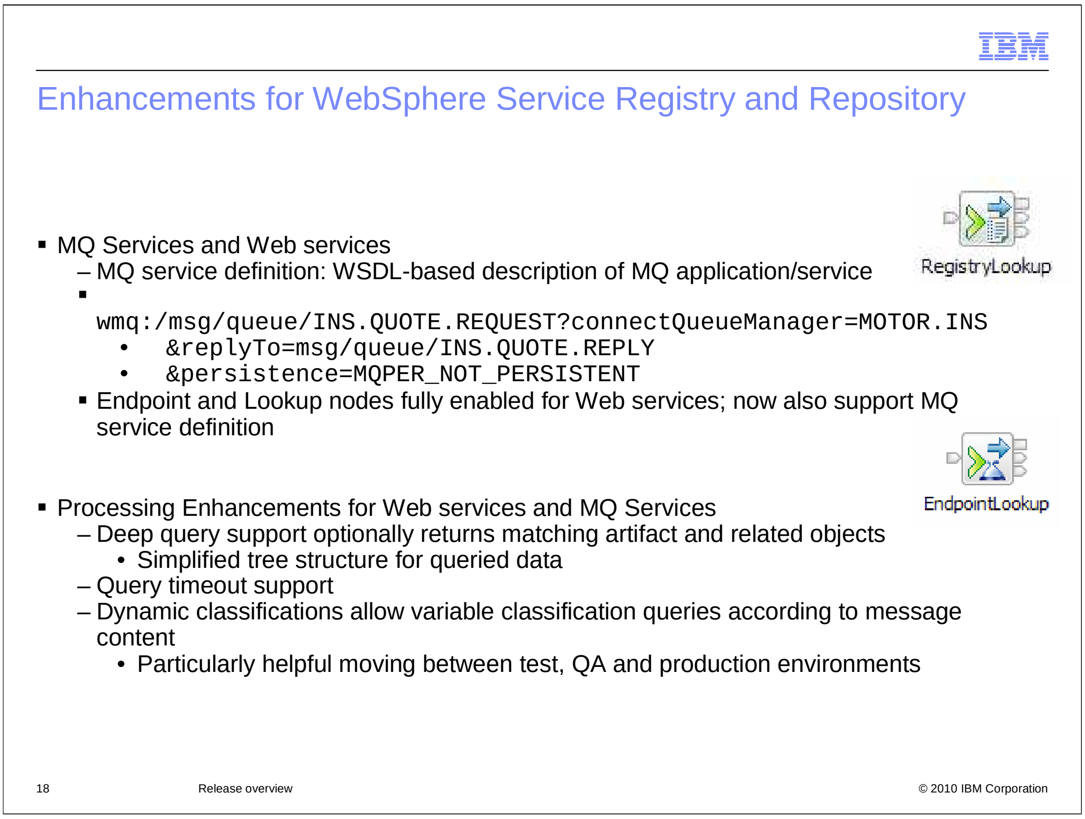
17 Release overview © 2010 IBM Corporation

Message Broker version 6.1 introduced and extended the facilities to provide monitoring information about flow instances. This enables a message flow to publish application data from every instance of a message flow, which can then be analyzed and presented to a business or application user. For example, if the message flow processes order requests from gold and regular customers, the event monitoring data can show how much discount has been given to the gold customers. This can be presented in a Web browser, using WebSphere Business Monitor.

These events can be configured to define the precise data that should be collected, how it can be correlated with other published data, and how the events should be published. It is also possible to specify filters, which will check whether a particular event should be collected or not.

Version 7 has introduced new tools which will automatically generate the monitor models. These are required by the WebSphere Integration Developer Monitor plug-in, and can be easily imported in this toolkit.

This facility is supported by a detailed sample in the Message Broker samples gallery.


IBM
RegistryLookup
EndpointLookup

Enhancements for WebSphere Service Registry and Repository

- MQ Services and Web services
  - – MQ service definition: WSDL-based description of MQ application/service
    - ▪

```
wmq:/msg/queue/INS.QUOTE.REQUEST?connectQueueManager=MOTOR.INS
  •   &replyTo=msg/queue/INS.QUOTE.REPLY
  •   &persistence=MQPER_NOT_PERSISTENT
```

    - ▪ Endpoint and Lookup nodes fully enabled for Web services; now also support MQ service definition
- Processing Enhancements for Web services and MQ Services
  - – Deep query support optionally returns matching artifact and related objects
    - Simplified tree structure for queried data
  - – Query timeout support
  - – Dynamic classifications allow variable classification queries according to message content
    - Particularly helpful moving between test, QA and production environments

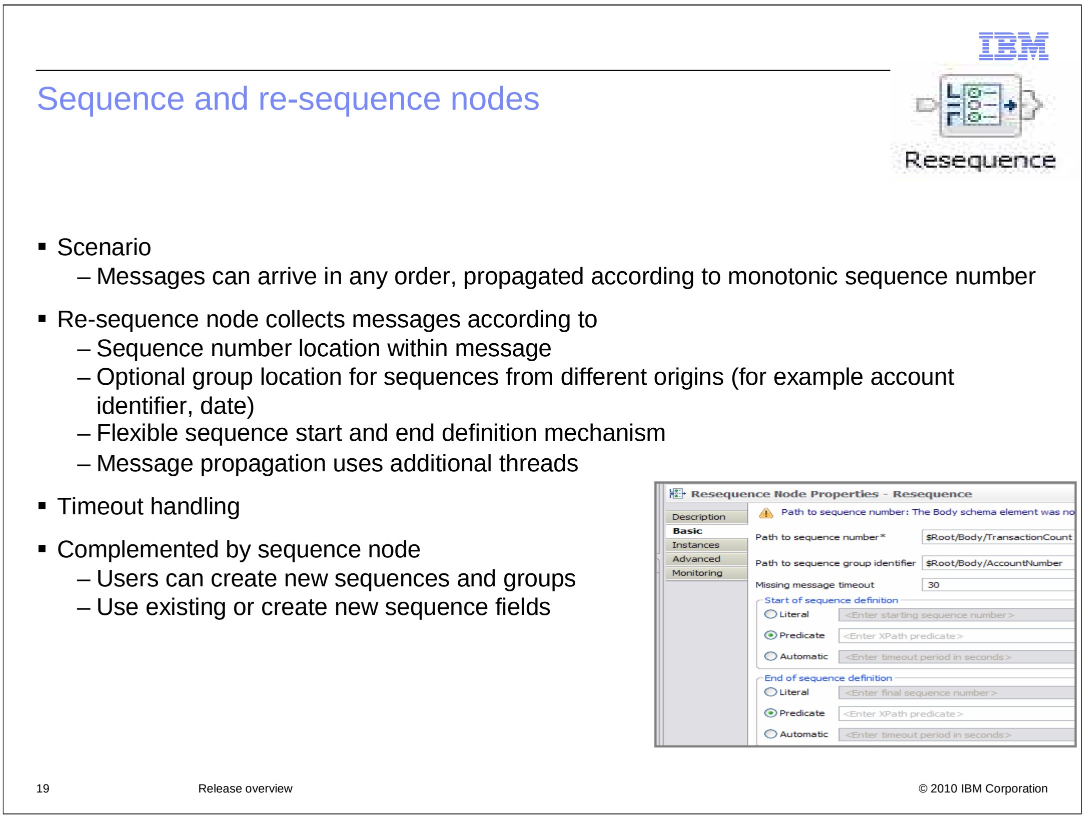
18 Release overview © 2010 IBM Corporation

The Registry Lookup and Endpoint Lookup nodes for WebSphere Service Registry were introduced in Message Broker version 6.1. The nodes and associated functions have been extended to allow the definition of MQ-based services in the service registry, and these definitions can now be used by a message flow that needs to invoke such a service.

In Message Broker version 7, two enhancements have been made to this interface. First, you can now retrieve more information from the registry, using the deep query options. Secondly, a timeout is provided, so that if the service registry does not respond, the affected message flow will not hang. In this case, the message flow will terminate with a timeout failure.

Another feature that can be used is the classification feature. This has been available for some time, but it is highlighted here to make sure that you are aware of its capabilities. This feature can be used to move an application between different systems, whilst not making changes to the message flow itself. As it moves through the development process, the message flow can be designed to invoke different versions of a target service using the classification feature.

WebSphere Service Registry and Repository now supports the definition of MQ-based services, using a syntax similar to that shown on this slide. The definition of these services can be retrieved from the registry by a message flow, and used by the flow to invoke MQ services.



Sequence and re-sequence nodes

- Scenario
  - – Messages can arrive in any order, propagated according to monotonic sequence number
- Re-sequence node collects messages according to
  - – Sequence number location within message
  - – Optional group location for sequences from different origins (for example account identifier, date)
  - – Flexible sequence start and end definition mechanism
  - – Message propagation uses additional threads
- Timeout handling
- Complemented by sequence node
  - – Users can create new sequences and groups
  - – Use existing or create new sequence fields

Resequence Node Properties - Resequence

Description | **Basic** | Instances | Advanced | Monitoring

Path to sequence number: The Body schema element was no

Path to sequence number* $Root/Body/TransactionCount

Path to sequence group identifier $Root/Body/AccountNumber

Missing message timeout 30

Start of sequence definition

○ Literal <Enter starting sequence number>

● Predicate <Enter XPath predicate>

○ Automatic <Enter timeout period in seconds>

End of sequence definition

○ Literal <Enter final sequence number>

● Predicate <Enter XPath predicate>

○ Automatic <Enter timeout period in seconds>

19 Release overview © 2010 IBM Corporation

Message Broker version 7 has introduced two nodes new nodes manage messages that have a requirement for maintaining the order of the messages.

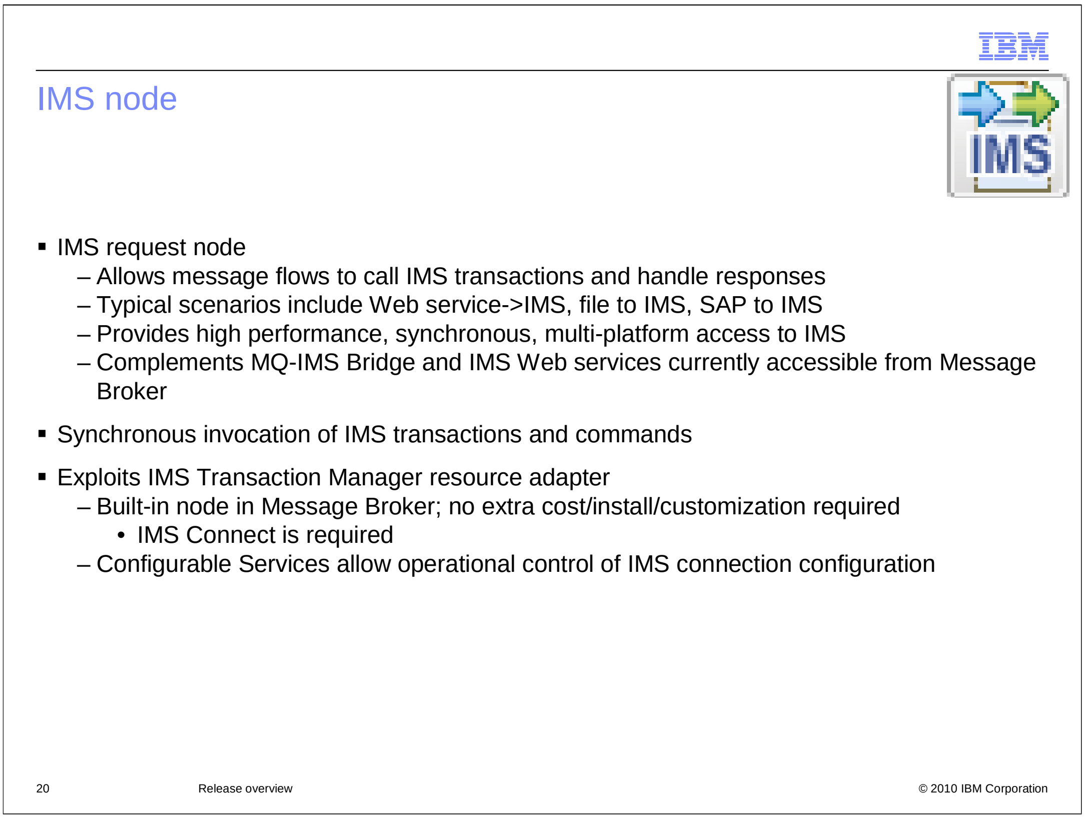
Using the re-sequence node, messages arrive in any order, and are propagated according to monotonic sequence number. The sequence groups have a defined start and end, and are propagated in order within the group.

The re-sequence node collects messages according to the sequence number location within message, or an optional group location for sequences from different origins.

You can define a flexible sequence start and end definition mechanism.

If messages do not arrive in the expected sequence, and arrive late, there is a timeout handling process. The missing message timer monitors sequence gaps, and late messages propagated according to the message flow design. Duplicate messages raise exceptions.

The sequence node allows you to create new sequences and groups. You can use existing fields or create new sequence fields.



IMS node

- IMS request node
  - Allows message flows to call IMS transactions and handle responses
  - Typical scenarios include Web service->IMS, file to IMS, SAP to IMS
  - Provides high performance, synchronous, multi-platform access to IMS
  - Complements MQ-IMS Bridge and IMS Web services currently accessible from Message Broker
- Synchronous invocation of IMS transactions and commands
- Exploits IMS Transaction Manager resource adapter
  - Built-in node in Message Broker; no extra cost/install/customization required
    - IMS Connect is required
  - Configurable Services allow operational control of IMS connection configuration

The IMS request node was provided in one of the fixpack releases of Message Broker version 6.1, but is discussed here as a reminder.

The IMS node supports a broad range of IMS facilities, including IMS message processing programs, batch programs (BMPs) and Fast Path transaction regions.

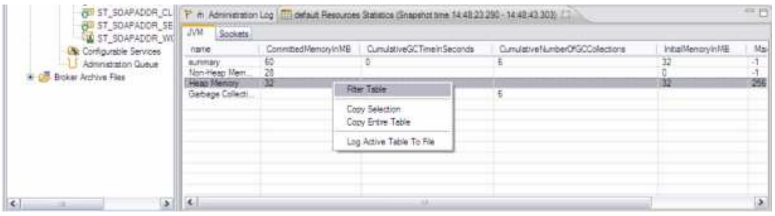
The IMS node uses commit mode 0 or 1, and Sync-level NONE, or CONFIRM.

It can handle single or multi-segment IMS messages.

The IMS node is provided as a standard component of WebSphere Message Broker, and there are no additional licensing considerations. The IMS node uses IMS Connect to connect to the target IMS region, and the IMS TM Resource Adapter is used within the IMS system. There are no other pre-requisite components.

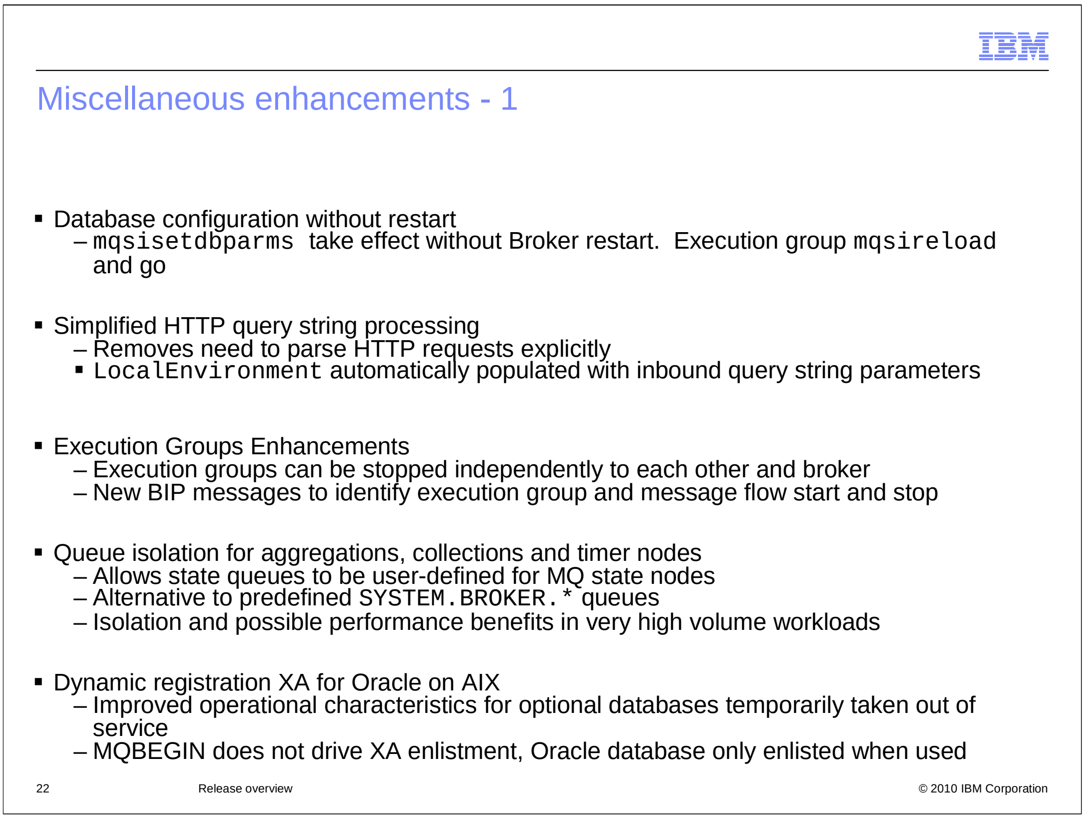
Broker runtime – Measurements and reporting

- Broker telemetry
  - Allows operational personnel to non-invasively understand behaviour of deployed solutions
    - Understand operational characteristics without reference to development artifacts
    - Logical extension of current existing accounting and statistics information
  - Reporting mechanisms
    - Configurable reporting interval
    - Graphically reported through MB Explorer
- Available resource report metrics
  - JVM: memory usage; message elements created/deleted; parser count
  - HTTP: socket host/port open; bytes sent, bytes received



name	CommittedMemoryInMB	CumulativeGCTimeInSeconds	CumulativeNumberOfGCCollections	InitialMemoryInMB	Ma
summary	60	0	6	32	-1
Non-Heap Mem...	28			0	-1
Heap Memory	32			32	256
Garbage Collecti...			6		

The Message Broker Explorer has been extended to provide the capability to report on resource usage within the JVM of a broker execution group. Both memory usage and socket usage are reported and the collection of these statistics can be started and stopped as required. This information is independent of any specific user application.



Miscellaneous enhancements - 1

- Database configuration without restart
  - `mqsisetdbparms` take effect without Broker restart. Execution group `mqsireload` and go
- Simplified HTTP query string processing
  - Removes need to parse HTTP requests explicitly
    - `LocalEnvironment` automatically populated with inbound query string parameters
- Execution Groups Enhancements
  - Execution groups can be stopped independently to each other and broker
  - New BIP messages to identify execution group and message flow start and stop
- Queue isolation for aggregations, collections and timer nodes
  - Allows state queues to be user-defined for MQ state nodes
  - Alternative to predefined `SYSTEM.BROKER.*` queues
  - Isolation and possible performance benefits in very high volume workloads
- Dynamic registration XA for Oracle on AIX
  - Improved operational characteristics for optional databases temporarily taken out of service
  - MQBEGIN does not drive XA enlistment, Oracle database only enlisted when used

22 Release overview © 2010 IBM Corporation

In addition to the major new items in Message Broker version 7, several smaller items have also been included.

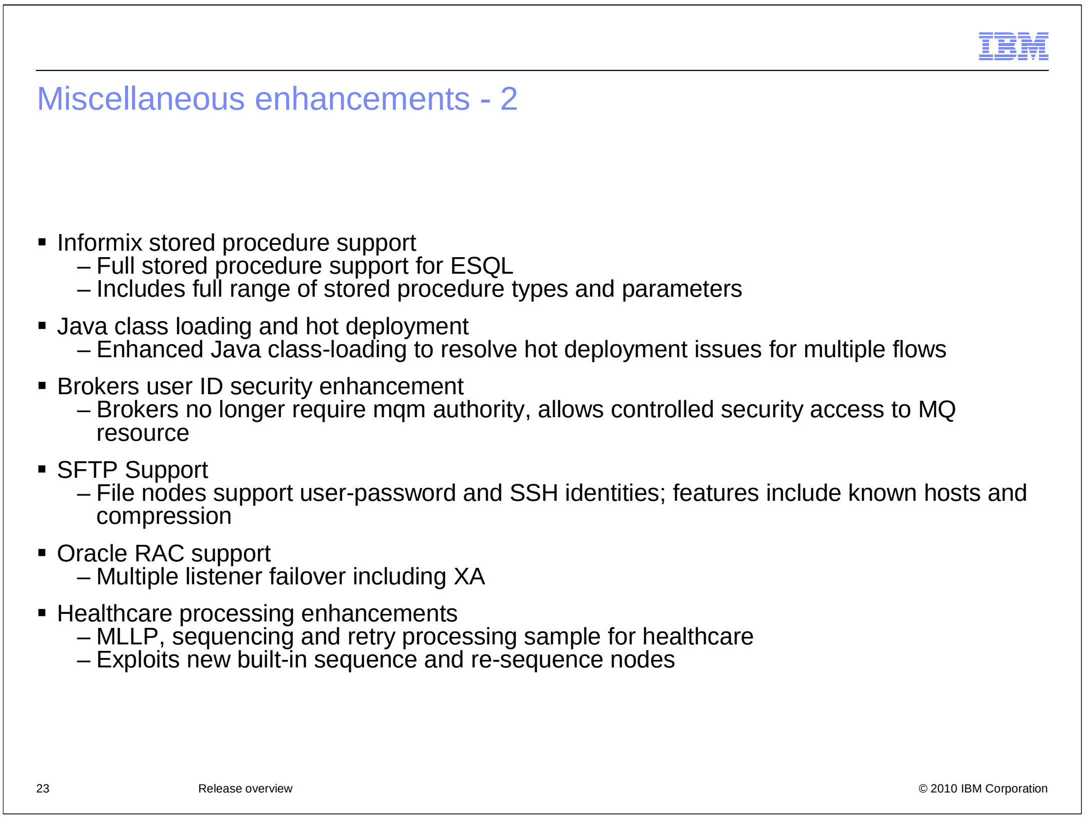
The mqsisetdbparms (*pronounced "m q s i set db parms"*) command has been improved so that it is no longer necessary to restart the broker for this to take effect. All that is needed is to restart the affected execution group.

If you have a requirement to parse the incoming HTTP string, then this information is now populated in the input local environment, and can be reference directly from a compute node.

In the operational area, execution groups can now be independently stopped and started. Stopping an execution group stops all the message flows contained in the execution group.

If you use the collector, aggregation and timer nodes, then you can now specify different MQ queues to hold the user state of applications using these nodes. This is likely to be of most use in high-volume scenarios.

And finally, the operation of Oracle database access has been improved, in particular for applications which have a transactional requirement.



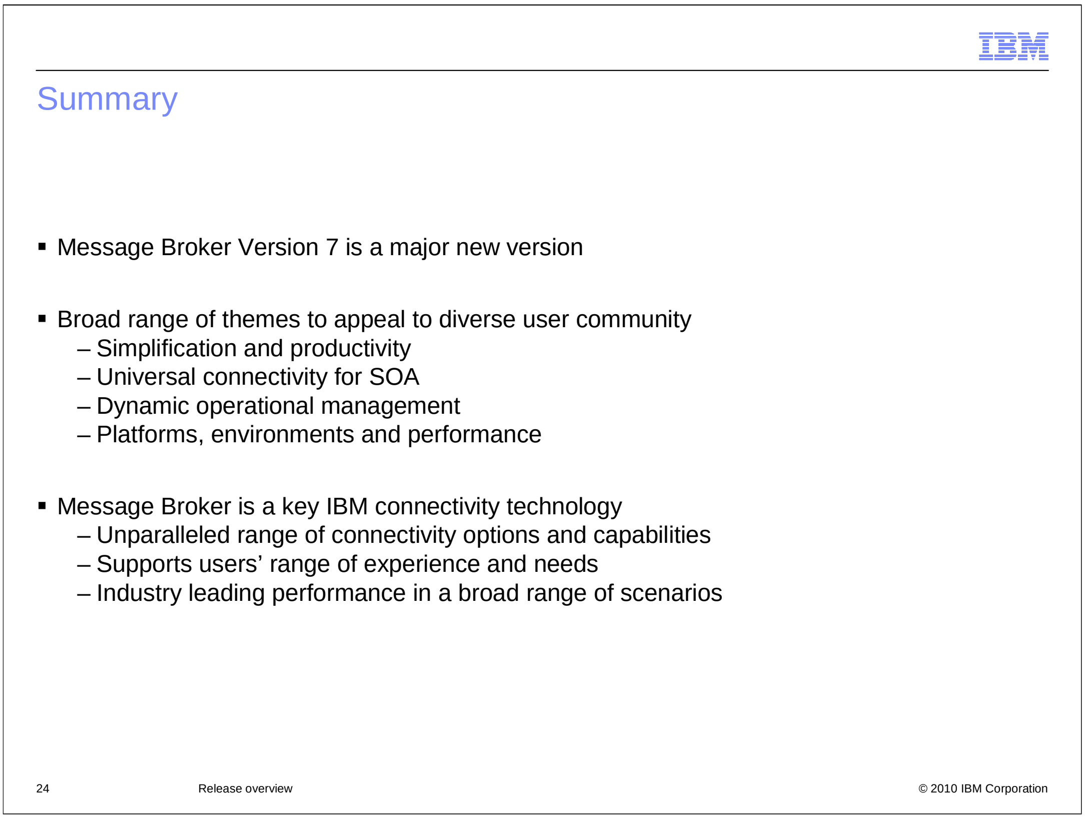
Miscellaneous enhancements - 2

- Informix stored procedure support
  - Full stored procedure support for ESQL
  - Includes full range of stored procedure types and parameters
- Java class loading and hot deployment
  - Enhanced Java class-loading to resolve hot deployment issues for multiple flows
- Brokers user ID security enhancement
  - Brokers no longer require mqm authority, allows controlled security access to MQ resource
- SFTP Support
  - File nodes support user-password and SSH identities; features include known hosts and compression
- Oracle RAC support
  - Multiple listener failover including XA
- Healthcare processing enhancements
  - MLLP, sequencing and retry processing sample for healthcare
  - Exploits new built-in sequence and re-sequence nodes

© 2010 IBM Corporation

This final slide provides a summary of the remainder of the smaller enhancements in version 7. They range from support for new database functions, Java enhancements, and operational improvements, to specific applications in the healthcare industry.

These functions are all explained in depth in the Message Broker version 7 information center.



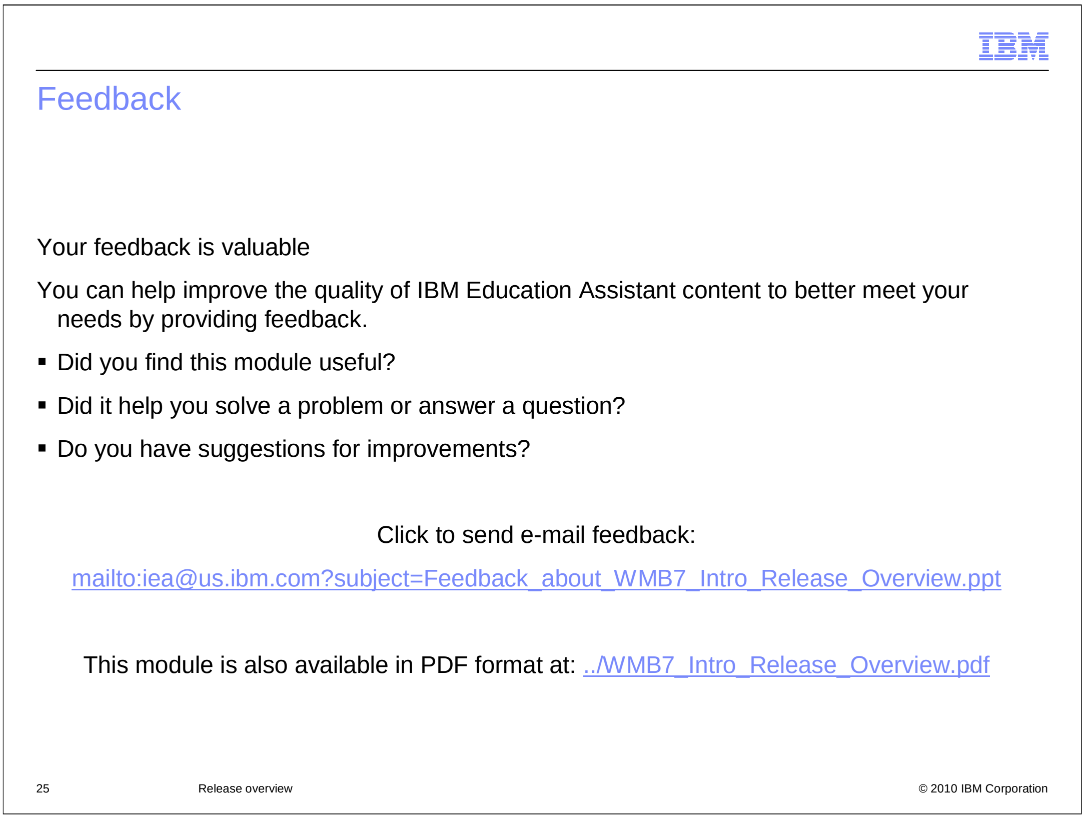
Summary

- Message Broker Version 7 is a major new version
- Broad range of themes to appeal to diverse user community
  - Simplification and productivity
  - Universal connectivity for SOA
  - Dynamic operational management
  - Platforms, environments and performance
- Message Broker is a key IBM connectivity technology
  - Unparalleled range of connectivity options and capabilities
  - Supports users' range of experience and needs
  - Industry leading performance in a broad range of scenarios

© 2010 IBM Corporation

In summary, WebSphere Message Broker version 7 is a major new version of IBM's key connectivity software. It has been significantly improved, with several simplification items covering configuration, application development and operations.

In addition, many new application programming tools have been provided, driven by the needs of the market.



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

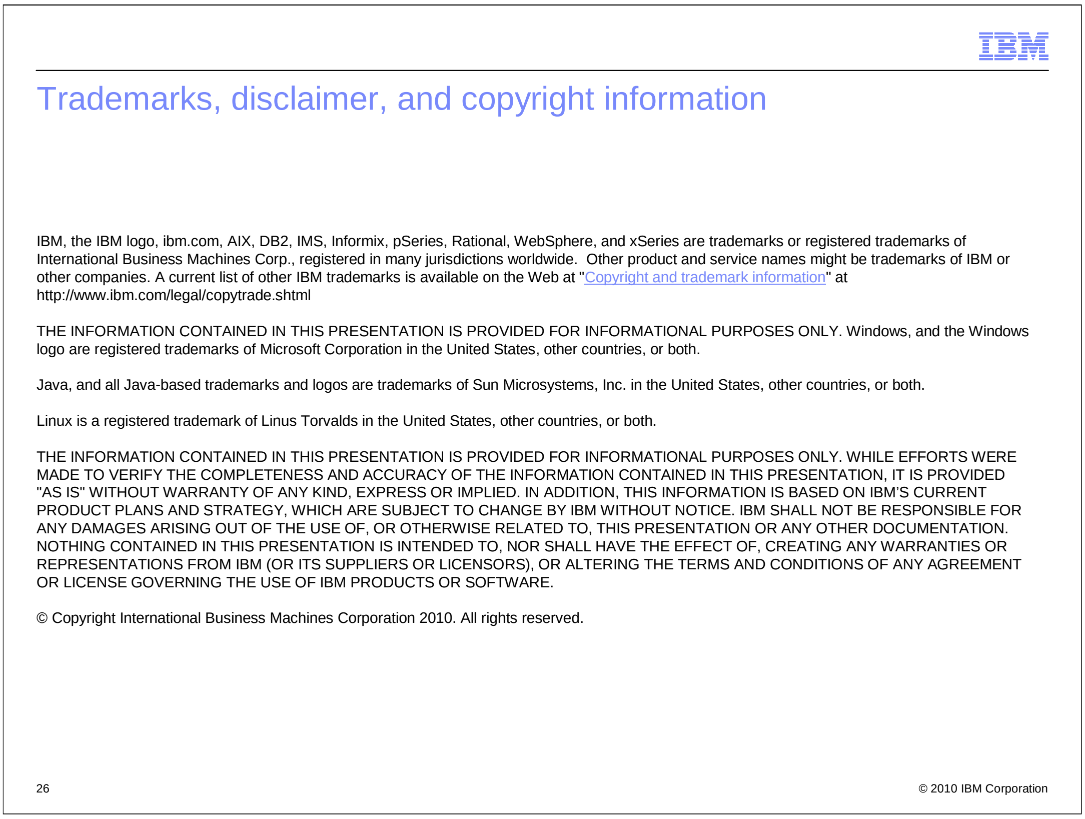
Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WMB7_Intro_Release_Overview.ppt

This module is also available in PDF format at: ../WMB7_Intro_Release_Overview.pdf

© 2010 IBM Corporation

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, AIX, DB2, IMS, Informix, pSeries, Rational, WebSphere, and xSeries are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. Windows, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2010. All rights reserved.