



Building an Enterprise Service Bus with WebSphere Message Broker for z/OS

name



Trademarks

The following are trademarks of the International Business Machines Corporation in the United States, other countries, or both. For a complete list of IBM trademarks please visit www.ibm.com/legal/copytrade.shtml

CICS	IBM Logo	S/390
DB2	IMS	Tivoli
E-business logo	iSeries	VM/ESA
ESCON	MVS	VSE/ESA
eServer	OS/390	WebSphere
FICON	pSeries	z/OS
IBM	Rational	zSeries
	RS/6000	System z

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

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
Microsoft trademark guidelines

Intel is a registered trademark of Intel Corporation in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Other company, product, or service names may be trademarks or service marks of others.



Agenda

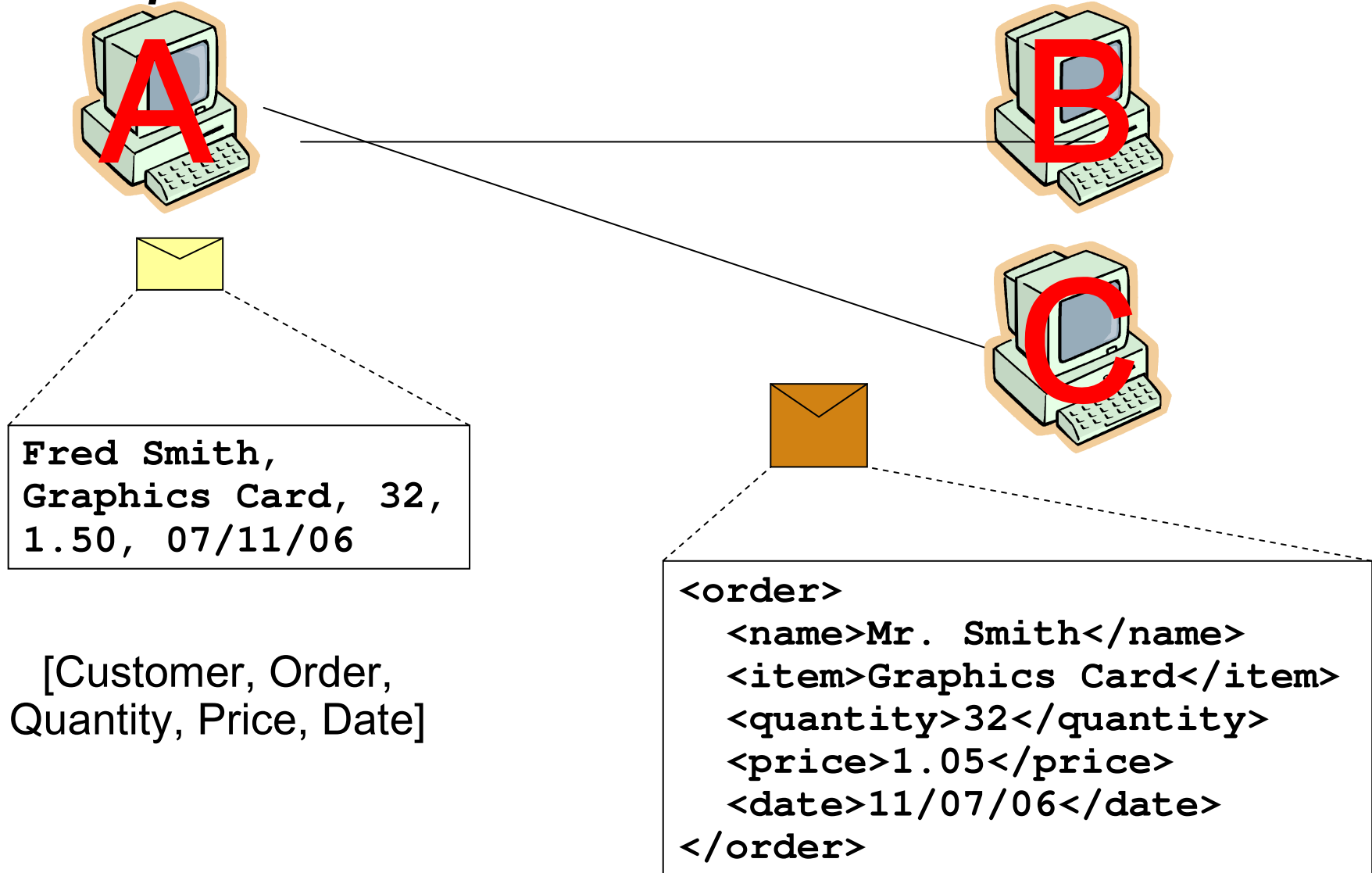
- WebSphere Message Broker Overview
- WebSphere Message Broker as an ESB
- What's new in WebSphere Message Broker for z/OS V6.1



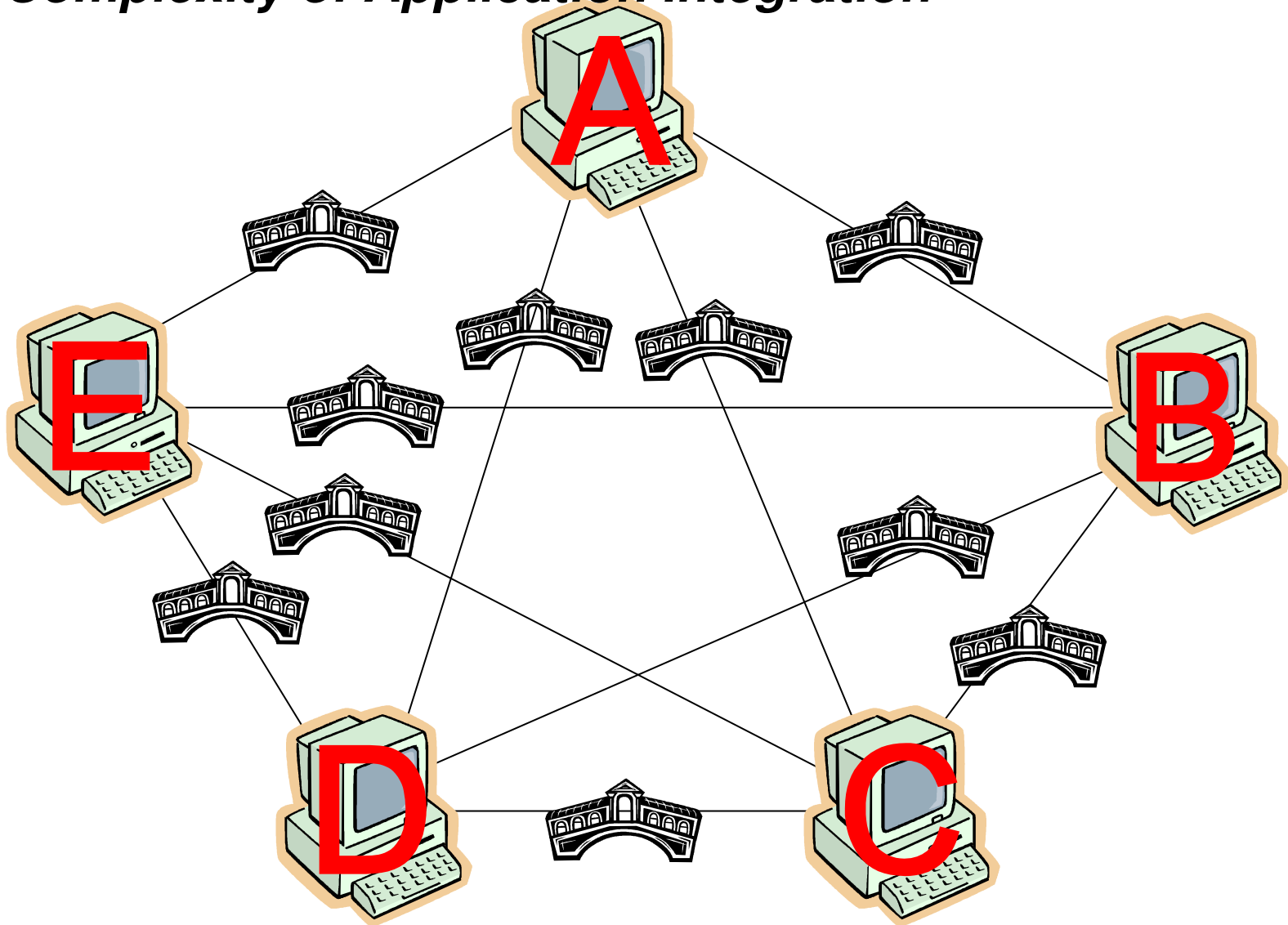
SOA: Unlock business value.
→ New software and services.



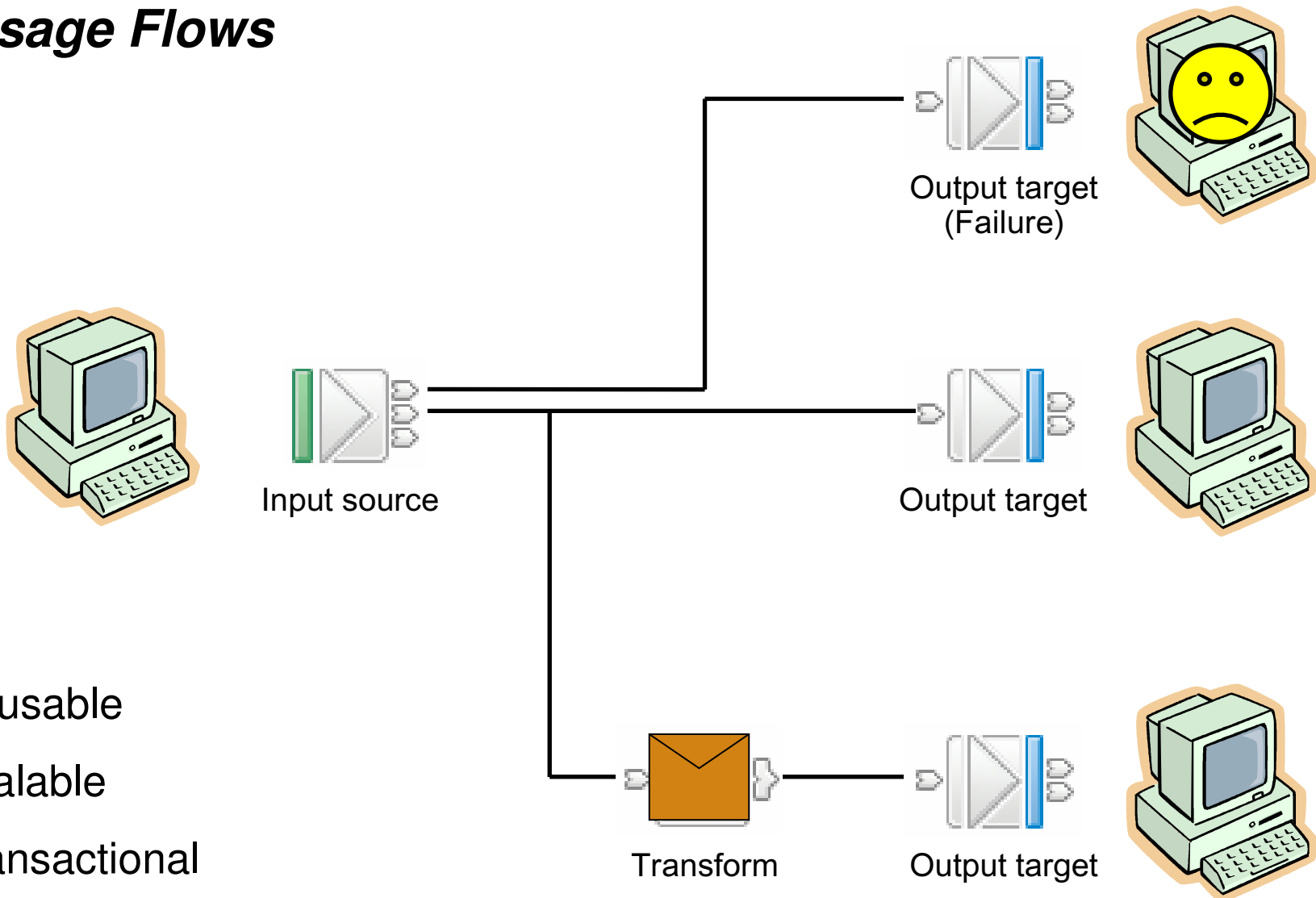
An Example



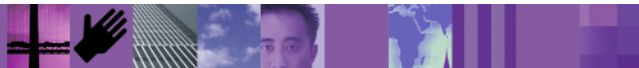
The Complexity of Application Integration



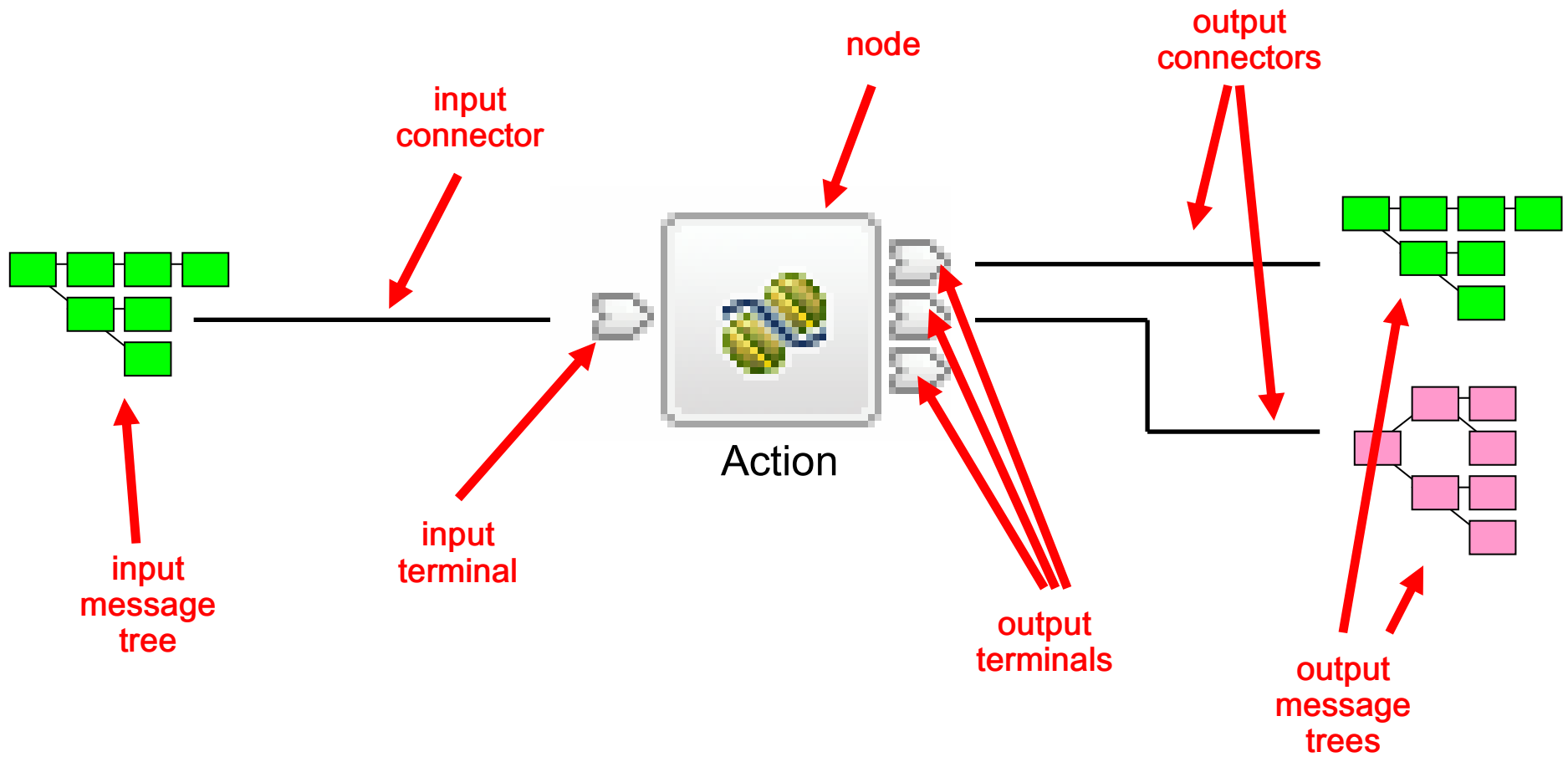
Message Flows



- Reusable
- Scalable
- Transactional

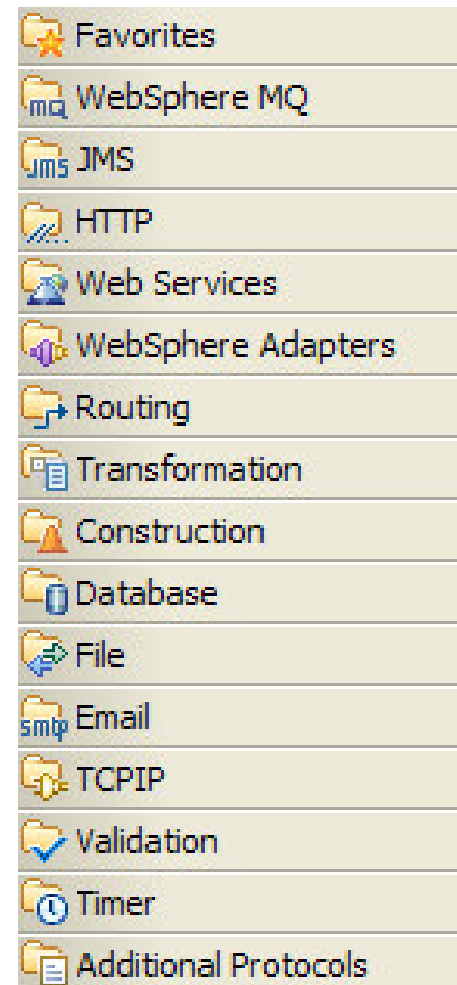


Message Nodes



Nodes

- The building blocks of message flows
- Each node type performs a different (input, output or processing) action
- Many different node types
 - Grouped into logical categories in the message flow editor



Lots of Nodes are Built in [1]

WebSphere MQ

- MQInput
- MQOutput
- MQReply
- MQGet
- MQHeader
- MQOptimizedFlow
- MQeInput (Deprecated)
- MQeOutput (Deprecated)

File

- FileInput
- FileOutput

Email

- EmailOutput

Web Services

- SOAPInput
- SOAPReply
- SOAPRequest
- SOAPAsyncRequest
- SOAPAsyncResponse
- SOAPEnvelope
- SOAPExtract
- Registry Lookup
- Endpoint Lookup

HTTP

- HTTPInput
- HTTPReply
- HTTPRequest
- HTTPHeader

JMS

- JMSInput
- JMSOutput
- JMSReply
- JMSHeader
- JMSMQTransform
- MQJMSTransform

TCPIP

- TCPIPClientInput
- TCPIPClientOutput
- TCPIPClientReceive
- TCPIPServerInput
- TCPIPServerOutput
- TCPIPServerReceive

WebSphere Adapters

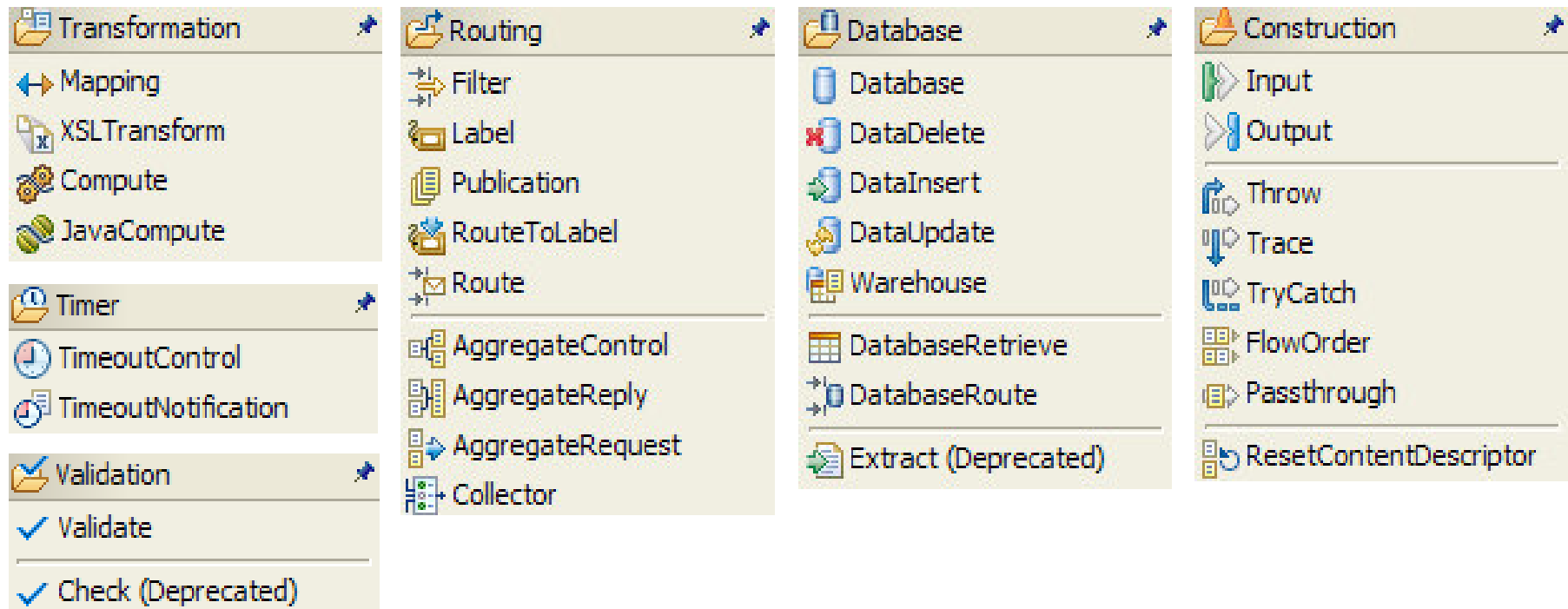
- PeopleSoftInput
- PeopleSoftRequest
- SAPInput
- SAPRequest
- SiebelInput
- SiebelRequest
- TwineBallInput
- TwineBallRequest

Additional Protocols

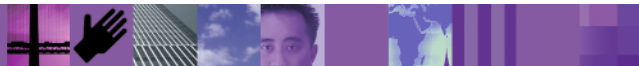
- SCADAInput
- SCADAOutput
- Real-timeInput
- Real-timeOptimizedFlow



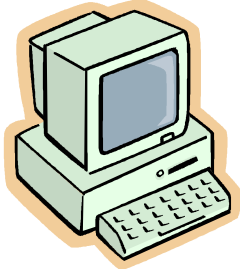
Lots of Nodes are Built In [2]



- Many other nodes available through product extensions and supportpacs
 - E.g. WebSphere TX, CICS, VSAM, QSAM
- Write your own User-Defined Nodes in C or Java



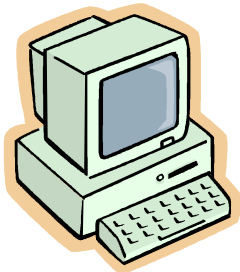
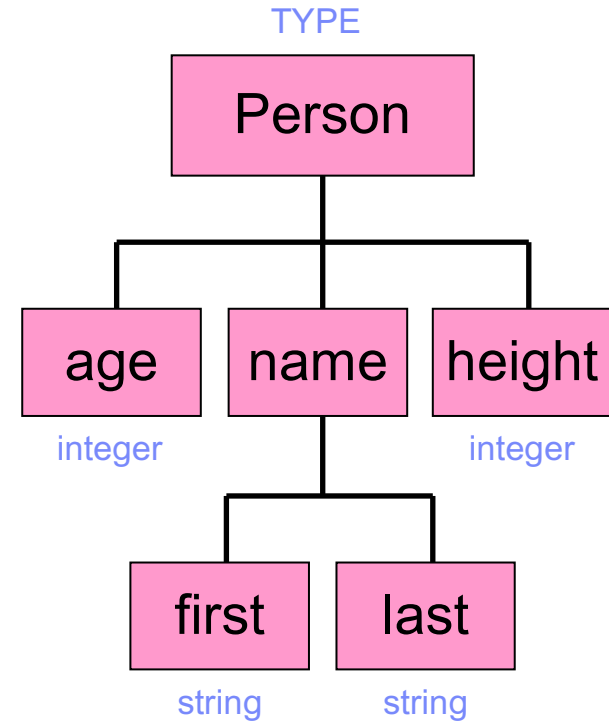
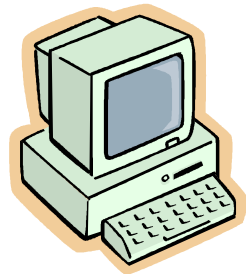
The Logical Message Model – Message Sets



```
<Person age='32' height='172'>
  <name>
    <first>Fred</first>
    <last>Smith</last>
  </name>
</Person>
```

```
struct {
  int height;
  int age;
  char firstname[24];
  char lastname[24];
} Person;
```

172	32	Fred	Smith
-----	----	------	-------



PER + 172 + 32 + Fred Smith

Message Processing Examples



DataInsert

```
IF Body.Person.height > 183 THEN
  INSERT INTO Database.TallPeople
    (Name, Height, Age)
  VALUES (Body.Person.Name,
    Body.Person.height,
    Body.Person.age);
ENDIF;
```



Compute

```
IF (XML format required) THEN
  OutputRoot.Properties.MessageFormat = 'XML';
ELSE IF (custom format)
  OutputRoot.Properties.MessageFormat = 'CWF';
ELSE IF (SWIFT format)
  OutputRoot.Properties.MessageFormat = 'TDS';
ENDIF;
```



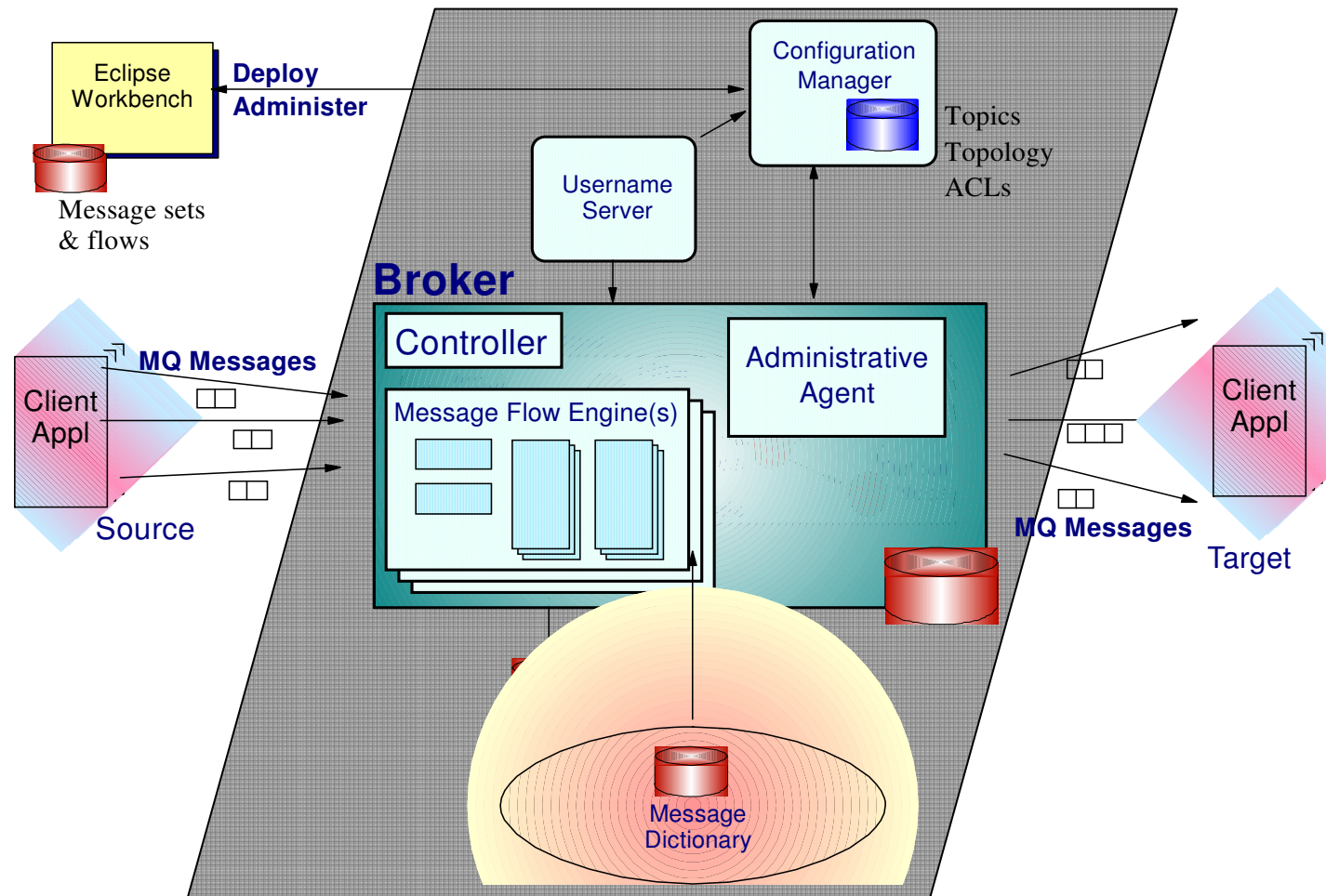
Java Compute

```
public class jcn extends MbJavaComputeNode {
  public void evaluate(MbMessageAssembly assembly)
    throws MbException
  {
    ...
    String personAge =
    (String) assembly.getMessage().evaluateXPath("/Body/Person/Age");
    ...
  }
}
```

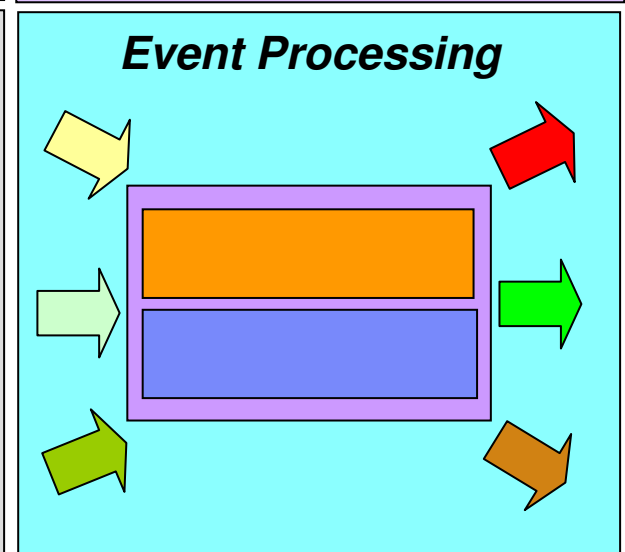
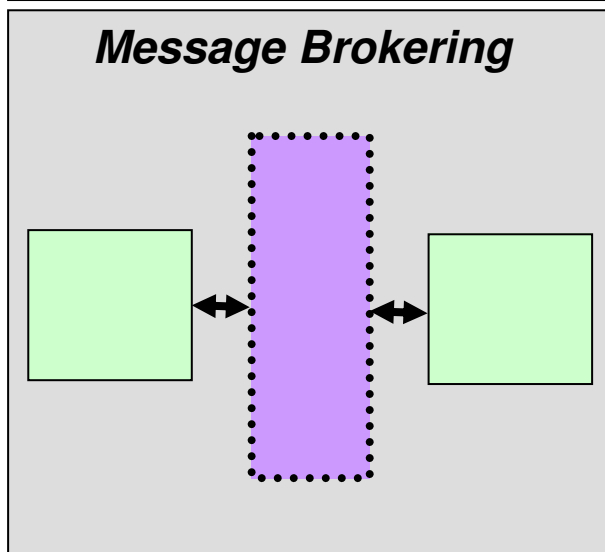
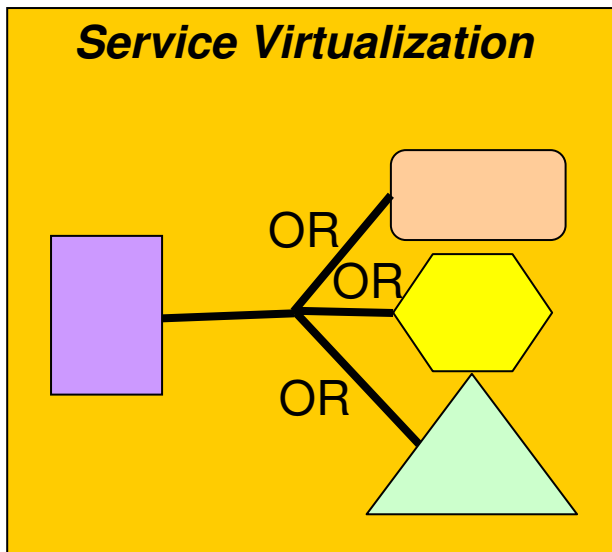
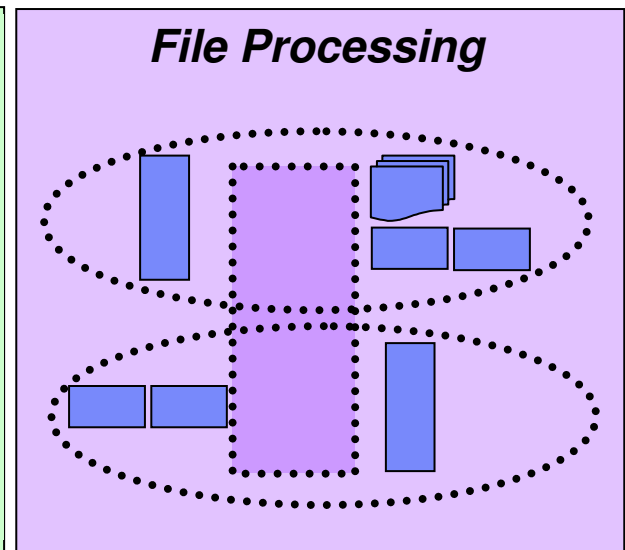
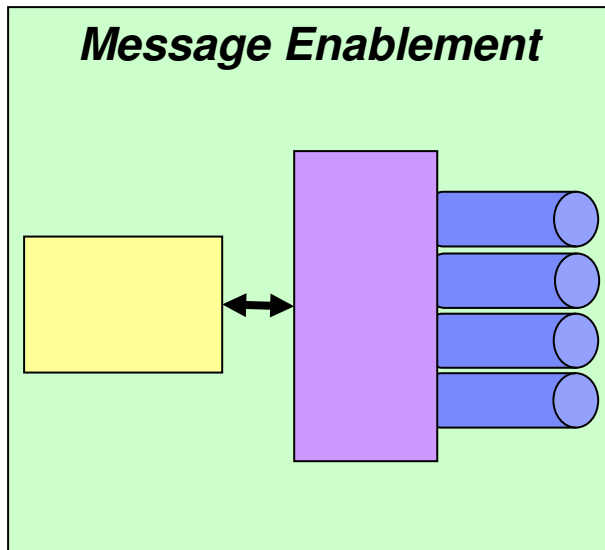
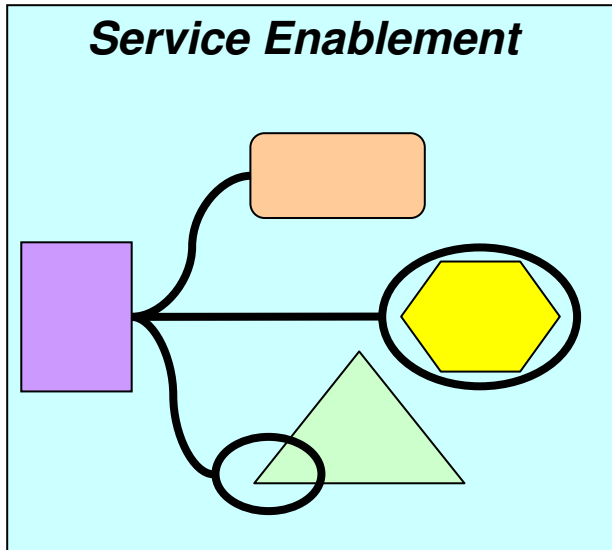


Message Broker Overview

Message Sets and flows are built in Eclipse and deployed to the runtime broker. The user name server sends principals to the Config Manager and broker. The z/OS Broker and Config Manager have full family capabilities, as well some z/OS specific features.



Usage Patterns with Message Broker

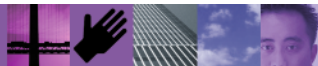


Agenda

- WebSphere Message Broker Overview
- WebSphere Message Broker as an ESB
- What's new in WebSphere Message Broker for z/OS V6.1








SOA: Unlock business value.
→ New software and services.

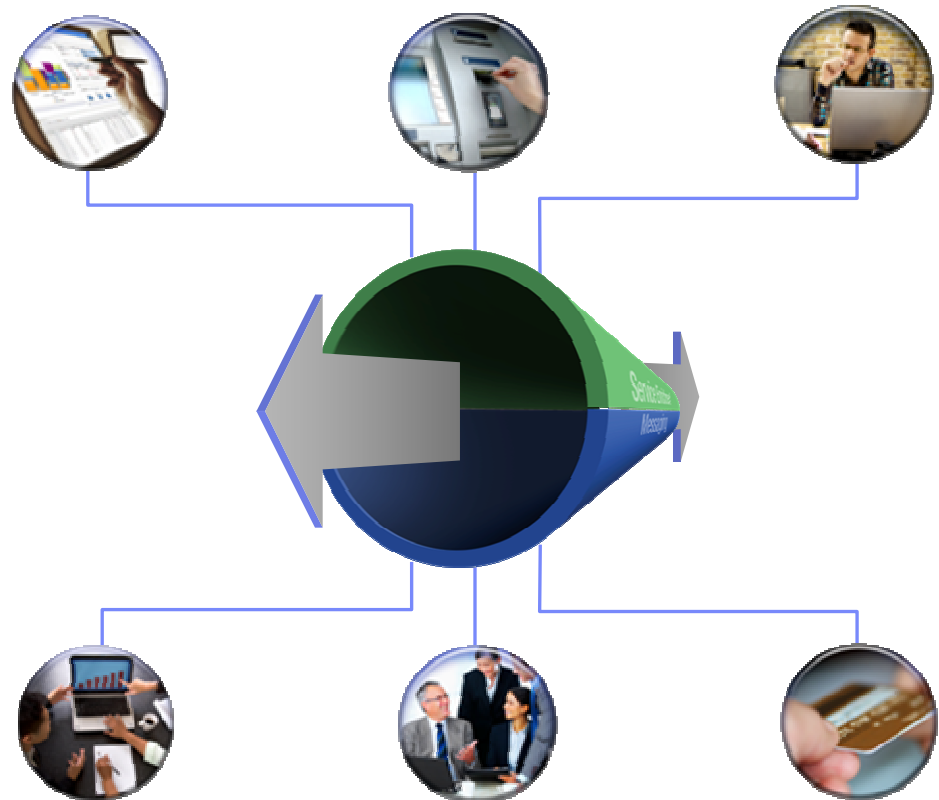




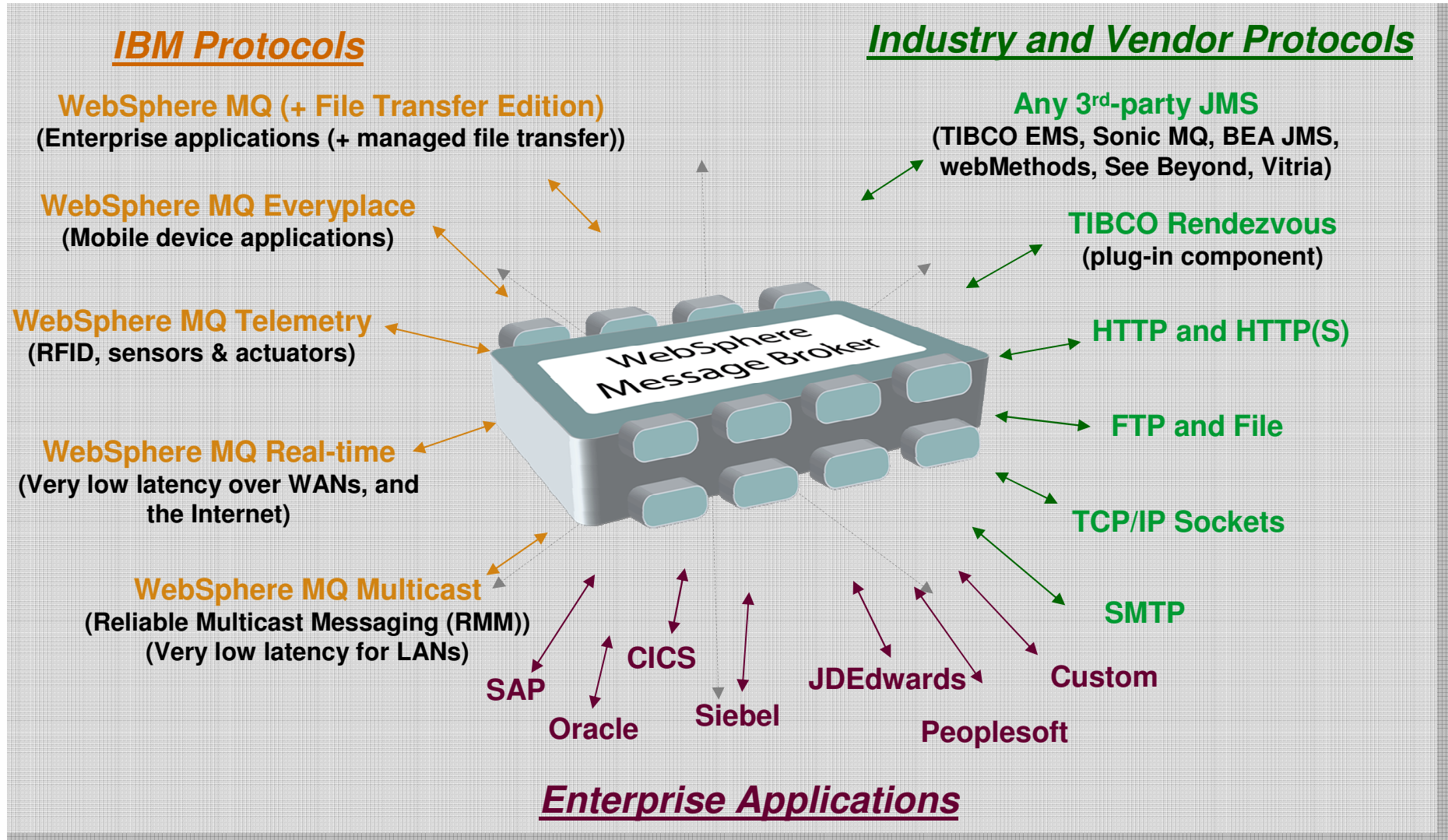
Integrating business applications through an ESB

An ESB is a flexible connectivity infrastructure for integrating applications and services

-  Connects Everything to everything
-  Matches & Routes Communications between services
-  Converts Between different transport protocols
-  Transforms Between different data formats
-  Distributes Business events

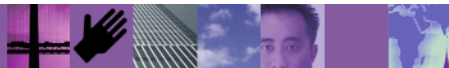


WebSphere Message Broker – Protocols and Applications



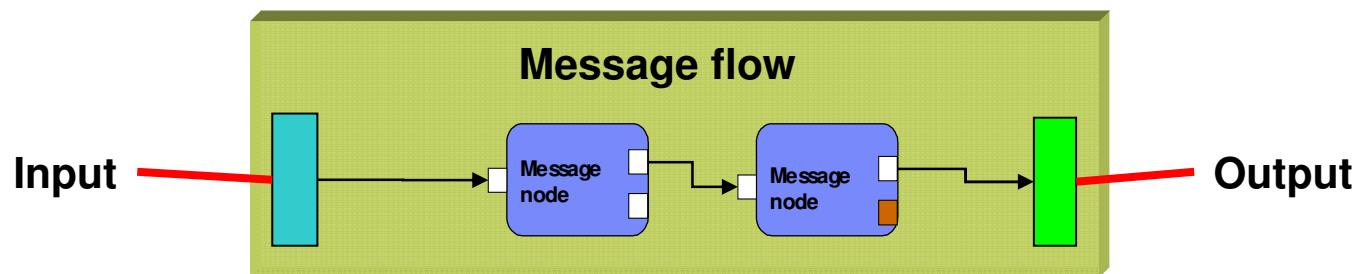
WebSphere Message Broker on System z at a glance

- Universal Connectivity
 - Simplify application connectivity to provide a flexible and dynamic infrastructure
- Routes and transforms messages **FROM** anywhere, **TO** anywhere
 - Supports a wide range of protocols
 - MQ, JMS 1.1, HTTP(S), Web Services, File, EIS (SAP, PeopleSoft...), TCP/IP, User Defined
 - Supports a broad range of data formats
 - Binary (C/COBOL), XML, Industry (SWIFT, EDI, HIPAA...), User Defined
 - Interactions and Operations
 - Route, Filter, Transform, Enrich, Monitor, Distribute, Decompose, Correlate, Detect...
- Simple programming
 - Message Flows to describe application connectivity comprising...
 - Message Nodes which encapsulate required integration logic which operate on...
 - Message Tree which describes the data in a format independent manner
 - Transformation options include Graphical mapping, Java, ESQL, XSL and WTX
- Operational Management and Performance
 - Extensive Administration and Systems Management facilities for developed solutions
 - z/OS and z/Linux natively supported
 - Offers performance of traditional transaction processing environments



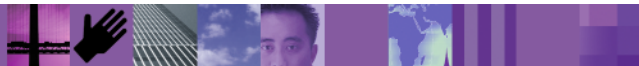
Concepts

- Broker = “container” for message flows
- Message flow = deployable unit of mediation, definition of ‘one-way’ mediation
 - Input = means of getting message into message flow
 - Output = means of getting message out of message flow
- Message node = atomic unit of message processing



Message Broker ESB Functions - Routing

- Ability to change or filter destination of messages
 - Support for one-way and request-response interaction patterns
 - Database, queue and in-memory caches for persisting transient state
 - Comprehensive composition and flow-of control capabilities
 - Branching, ordering, exception processing, tracing
 - Support for creation of new messages during processing
- Full range of messages can be routed and filtered
 - XML – for XML payloads (full schema support)
 - Record based – for COBOL, C payloads
 - Industry standard – SWIFT, EDI...
 - User defined – RYO formats
- Different routing and filtering rules
 - Message based
 - Message Header and Body routing are supported
 - External reference
 - Database lookup, PLUS in-memory cache for infrequently changing data
 - User algorithm
- Partial Parsing for efficient routing of large messages



Message Broker ESB Functions - Protocol Support

- de facto Standard protocols
 - MQ, HTTP, JMS, TCP/IP
 - Native input nodes provides
- “Standards” protocols
 - Web Services via SOAP/HTTP, SOAP/JMS
 - Native HTTP(S), Any JMS 1.1 connectivity
 - SOAP and MIME parsers provided
- Technology interactions
 - Files, VSAM, QSAM, CICS to access traditional data sources
- Industry protocols
 - SCADA, MQe for mobile devices and sensors
- User Defined protocols
 - For User ISV provided extensions, full user defined node interface.



Message Broker ESB Functions - Transformation

- Message Broker models all data via “Message Tree”
 - All data formats (XML and non-XML) understood natively by message broker
 - Message Tree created by native parser from source input message format
 - Transformed using appropriate transformation language
 - Output tree used to generate required target output message format
 - ...leads to high performance
 - Native formats means data is only transformed when it necessary
 - Partial parsing only reads required fields
- Use transformation languages to match skill set
 - Graphical mapping for non-programmers
 - XSLT for XML orientation
 - SQL for database and declarative orientation
 - Java for widely available skill set benefits
 - ESQL
 - WTX to leverage existing maps –
 - **WebSphere Transformation Extender for Message Broker**
 - Transformation languages can be mixed and matched within same message flow



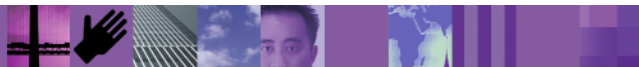
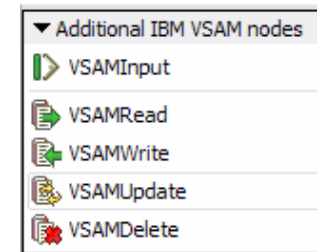
Message Broker ESB Functions – Event Handling

- Processing external events
 - From simple message exchange to message pattern detection
 - Ability to distribute information to interested parties
 - Publish Subscribe and more complex event processing (CEP)
- Timer based processing
 - Processing triggered by timed event
 - Fixed or variable timers
 - Ability to generate new timer events from current processing
- Message Validation
 - Confirm that messages are well formed as they enter and leave the bus
 - e.g. SWIFT validation, XML schema validation



Why run WMB on z/OS?

- WMB on z/OS takes full advantage of the z/OS Sysplex
 - The only platform supporting MQ Shared Queues for high-availability and optimized “pull” load-balancing with the very best in scalability
- WMB on z/OS utilizes many of the other inherent features of z/OS
 - RACF security
 - WLM resource optimisation
 - SMP/E installation control
 - SMF performance recording
 - RRS transaction management for speed and reliability
 - ARM restart
- WMB on z/OS offers several specific interfaces to the z/OS platform
 - CICSRequest, QSAM and VSAM nodes
 - Using WMB on z/OS will typically be much more efficient if some of the interfaced applications are co-located on z/OS
- Running on the z/OS and System z platform brings all the security and robustness of the the mainframe platform
 - Looking for the highest scalability and high-availability for a strategic corporate ESB



Agenda

- WebSphere Message Broker Overview
- WebSphere Message Broker as an ESB
- What's new in WebSphere Message Broker for z/OS V6.1



SOA: Unlock business value.
→ New software and services.



z/OS Improvements for Version 6

- Configuration Manager
- JES SPOOL for ALL output
- Audit Trail Messages for all broker and configuration manager operations
- 50 % Performance improvement in message throughput
- CICS Node for access to CICS via EXCI
- VSAM Nodes to input, read, write, update and delete records from VSAM files
- MQ Shared queues Serialized Application support
- All Commands, and all in JCL
- Unicode database support
- Restart broker without DB2 restart
- zAAP offload for Java Compute, XSLT, Realtime & multicast, Config Manager
- XML Toolkit for consistent XML processing
- MVS Dumps rather than coredumps in USS ;)



Key Supportpacs

- Broker Explorer (IS02)
- CICS/VSAM Nodes (IA12/IA13)
- SOAP Envelope/SOAPExtract Nodes (IA9O)
- Message Broker Client for WSRR (IA9Q)
- Importer for formatted text message models (IA7D)
- Complex Event Processing Nodes (IA0S)



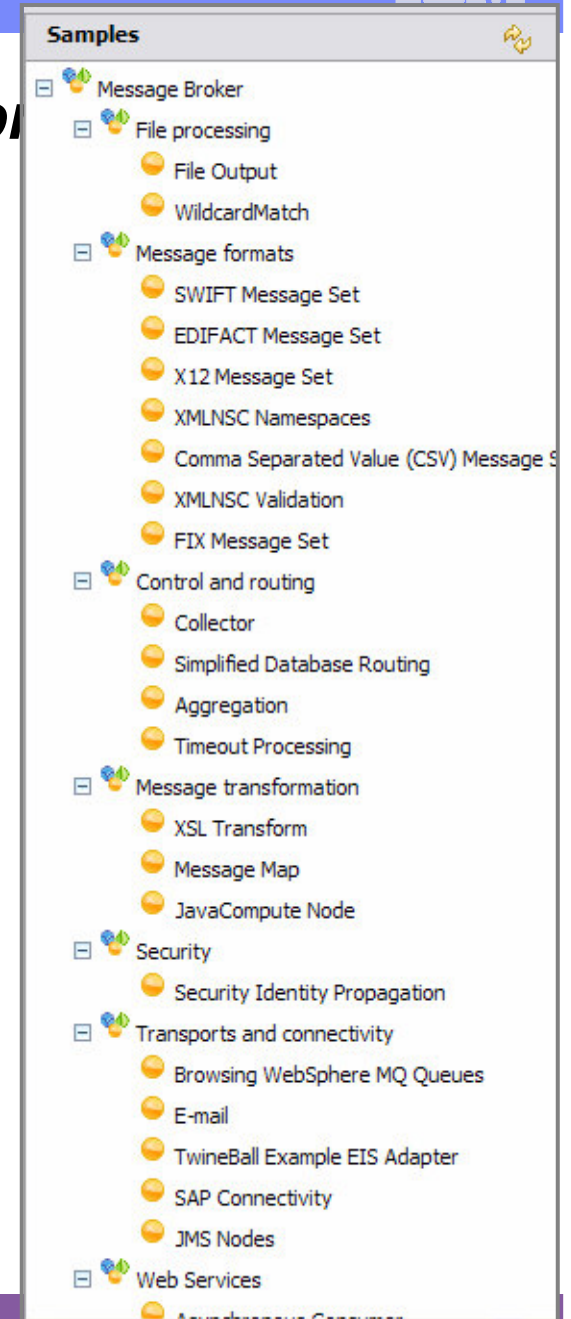
Version 6.1 Feature Overview for System z

- **Ease of Use and Productivity**
 - Reducing the time to get started with Message Broker
 - Simplifying development tasks including debug; reducing the time to create working solutions
- **Enhanced SOA support**
 - Supporting Web Services natively with WS-Security and WS-Addressing
 - DataPower SOA appliance for WS-Security
 - Integration and enhancement of WSRR support
- **Extended Connectivity**
 - Built-in nodes for EIS access
 - SAP for z/OS, SAP and PeopleSoft for zLinux
 - WebSphere Adapters delivered “out of the box” as built-in nodes
 - Native support for very large file processing, including FTP
 - New SMTP and TCP nodes
- **Administration & Systems Management**
 - Enterprise-wide identity, authentication and authorization with Tivoli and LDAP
 - MB Explorer Eclipse administration
 - Numerous manageability improvements
- **Platform Support and Performance**
 - 64 bit zLinux; Java 5
 - Ultra High Performance XML parser including schema validation
 - Compacted memory footprint; Real-time graphical performance analysis
 - Performance improvements
- **Message Broker V6.1.0.2 GA 27 June 2008**
 - Two new attractively priced deployment options on Linux on System z and Getting Started Sub-capacity pricing for z/OS (http://www.ibm.com/servers/eserver/zseries/library/swpriceinfo/ipla_exe.html)
 - Enhanced WebSphere TX support
 - New TCP/IP nodes and TLOG Sample
 - Common Event Infrastructure support enabling WebSphere Business Monitor integration



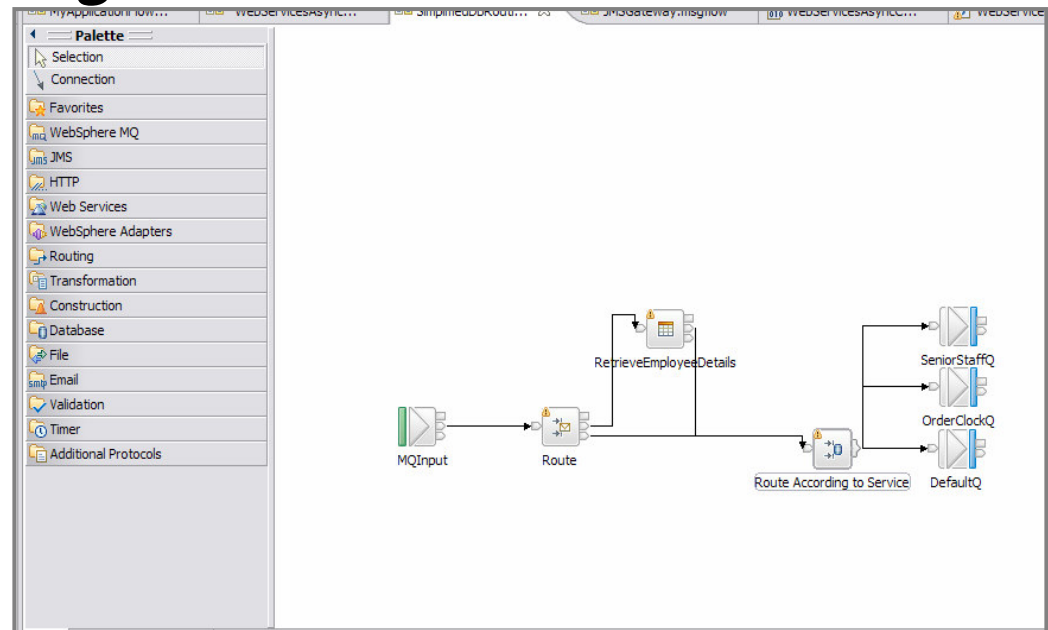
Platform Support and Product Simplification

- Simplified Offering
 - Single Message Broker Offering focussing on Advanced ESB functionality
 - Event Broker version 6 customers entitled to Message Broker 6.1
 - Rules and Formatter available for existing users only
- 64 bit Support
 - zLinux has 64 bit capability
 - Default execution group size is 64 bit & commands are all 64 bit
 - z/OS remains 31 bit for V6.1
- Databases
 - DB2 for z/OS, DB2 and Oracle for zLinux as broker databases
- Java
 - JDBC XA for z/Linux
 - z/OS RRS for JDBC XA to follow in future release
 - Java 5 on all platforms
- Removal of RAC prerequisite – native debug
- ISMP installer for zLinux, SMP/E for z/OS
- Default configuration
 - Allows you to understand broker components and configuration
 - Quickly create a working system for development
- Samples gallery
 - Comprehensive “Samples Gallery” for all new and existing function
 - Single click to install and run using default configuration
 - New sample message sets e.g. CSV
 - Learn how to use all 6.1 capabilities using realistic, working samples
- Product Prerequisites
 - MQ V6 or above
 - A production database
 - Cloudscape provided for development and test
 - DB2 supplied as production database



Powerful, Easy to Use Tooling

- Full function Toolkit in smallest ever install
- Builds on advances in Toolkit 6.0.2
- Wizards
 - Guides you through solution creation
 - Novice and expert modes
- WSDL Drag Drop
 - Quickly create Web Services solutions
- Drag and Drop Mapping
 - Now includes calling Java from map
- Many new ease of use features
 - “Message Viewer” visualizes expressions
 - Discovery wizards for SAP and PeopleSoft
 - BAR file rebuild
- Integrated Test Facility
 - Unit Test License included
 - Test Client to test flows
 - Direct debug using Java Debug Protocol
 - “Component Trace” to follow message path



Message Flow Test Events

- Invoke Message Flow
- Invoke Message Flow
- Invoke Message Flow
- Invoke Message Flow
 - Starting
 - Sending Message to MQ Queue "inq"
 - Message from [MQInput : out] to [Filter : in]
 - Message from [Filter : true] to [MQOutput : in]
 - MQ Queue Monitor "outq"
 - Stopped

General Properties

Detailed Properties

Message Flow: /flowProject/testMQ.msgflow

Source message node: MQInput

Source terminal: out

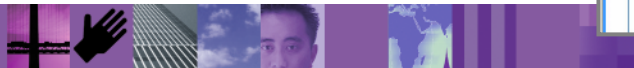
Target message node: Filter

Target terminal: in

Message

- Header
- Properties

Viewer: XML Structure



Support for Web Services

- Support provider and consumer scenarios
 - Provider:
 - SOAP input & SOAP reply
 - Consumer:
 - Synchronous SOAP request
 - Asynchronous SOAP request and reply
 - Can be combined to provide Web Service intermediar
 - SOAP Extract and SOAP Envelope nodes
 - Simplify processing of SOAP payload and headers

- Support WS-Security and WS-Addressing “out of the box”
 - Support for WS-Addressing Endpoint References and Message addressing properties
 - Support for WS-Security authentication, encryption and signing
 - Username password, X509 certificates for authentication
 - Comprehensive encryption and signing algorithms (from JSSE/JCE)
 - Configuration using Policy Sets
 - Policy Set editor enables declaration of WS-Security capabilities

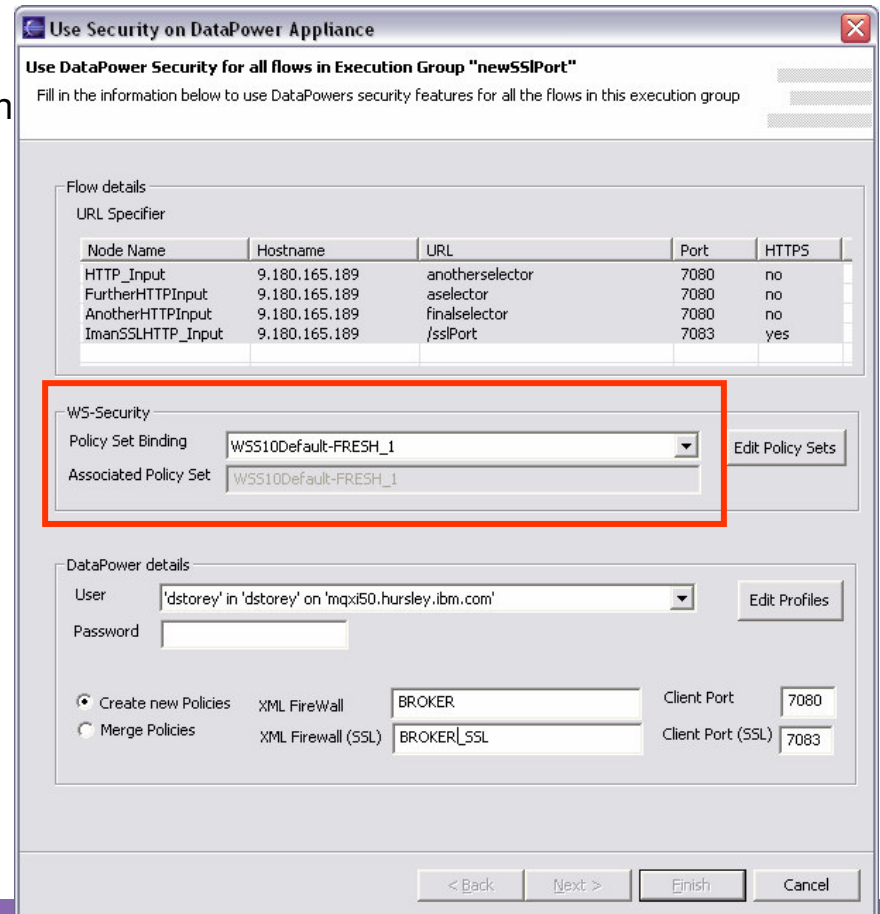
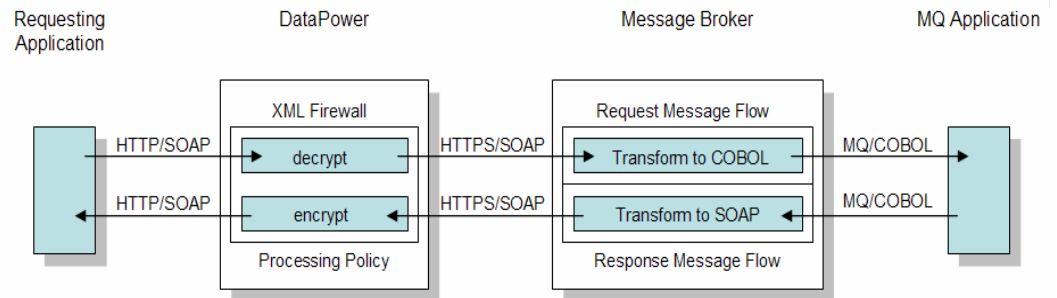


DataPower Appliance

- Exploit DataPower for Web Services security
 - Single tool and security policy description
 - Security best practices
 - WS-Security at appropriate point in topology
 - Built-in XML threat protection; Hardened device
 - Built-in service level management
 - Manage traffic using policy; WSDM and WS-Man
 - Scale as volumes increase
 - Enhanced performance with SOA appliance
 - Add capacity when necessary

- Administration User Experience
 - Operational reconfiguration only
 - Applications and Message Flows unchanged
 - Right click on flow and select “Use DataPower”
 - DataPower performs WS-Security processing
 - Forwards processed request to MB

- Initial focus is on XML and WS-Security processing
 - June 2007 preview
 - Other functions may follow



WebSphere Service Registry and Repository

- Integrated support for WSRR
 - Registry contains variety of “entities” (documents) such as WSDL, XSD...
 - Includes entity category, its relationships and its associated user properties
 - True governance achieved through registry determining MB processing
 - Development and runtime usage aspects which can be used together

- Development Activity
 - Use WSRR AD plug-in to search registry for particular entity
 - Entity can “kick start” message flow and message set creation
 - E.g. Retrieve WSDL and drag-drop to configure external Web Service call

- Runtime interactions
 - Message flows can query and/or select specific registry entities
 - Information cached for high performance access
 - Registry changes result in cache refresh via built-in PubSub mechanism
 - 2 new nodes to support most popular processing scenarios
 - Query – retrieve entity details; other nodes can act on this
 - Select – choose a specific service instance via node matching criteria
 - WSRR interactions can be overridden dynamically based on message content
 - Expanded expression support to include literals and variables



RegistryLookup

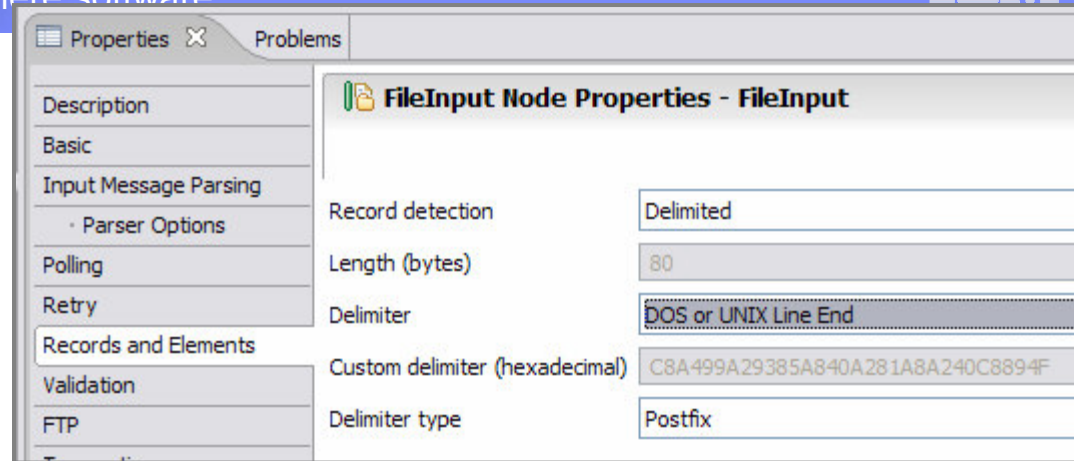


EndpointLookup



File Processing Built-in

- Local and remote (FTP) files
- Advanced file processing within ESB
 - File input and File output nodes
 - Combine with other MB nodes
 - (e.g.) File to MQ, File to database, File record filtering
- Large file handling
 - Allows very large files (gigabyte) to be processed without using excessive storage
 - Appropriate broker parsers have been enhanced to request data on demand
- Comprehensive support for record detection
 - Simple: LF, EOL, CRLF, Fixed Length, Whole-file, User-defined
 - Parser: Use an existing message definition to identify record boundaries



FileInput



FileOutput



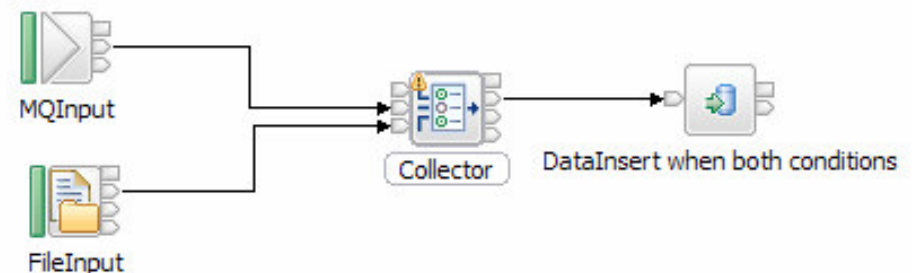
Transport Headers and Triggering

- Process Transport headers without programming (6102)
 - Shows most common transport header properties for MQ, JMS and HTTP
 - New users can easily understand and modify common transport properties

- SMTP node for email generation
 - e.g. “operator” notification of alert conditions

- TCPIP nodes for legacy integration (6102)
 - Client and server side sockets support, inbound and outbound

- Collector node for more advanced triggering scenarios
 - Coordinate message from multiple, disjoint sources
 - Wait for multiple input conditions
 - Process when all satisfied



Collector Node Properties - Collector

Collection definition

Terminal	Quantity	Timeout	Correlation path	Correlation pattern
MQInput	1	0	\$Root/MQMD/CorrelId	
FileInput	1	0	\$LocalEnvironment/File/Name	

High Performance

- Significant Performance improvements on ALL platforms
 - Major throughout improvement across a broad range of scenarios
 - Builds on real world, customer-verified, scenarios on version 6 usage
 - No need to change flows or assets to receive gains – “for free”

- Highlights
 - Significant XML performance
 - Up to 150% improvement processing more complex XML documents
 - XML validation performance
 - Up to 3 times improvement validating XML documents
 - Binary and String parsers improved
 - Industry and legacy message formats will benefit
 - XSLT performance
 - Solid improvements

- Storage reduction
 - Compacted runtime storage, significant reduction in runtime footprint



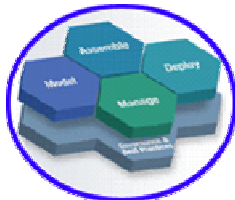


Version 6.1.0.2 Feature Overview



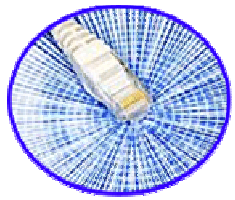
Ease of Use and Productivity

- Unit Test Client Enhancements
- Mapping Node Enhancements for Stored Procedures
- New Modes of Operation on distributed: Trial, Starter, Remote Adapter Host & Enterprise
- Getting Started Sub-capacity pricing on z/OS



Enhanced SOA support

- WS-Security Enhancements
- WS-Addressing Enhancements



Extended Connectivity

- TCP/IP nodes
- WebSphere Transformation Extender Launcher support
- MQ, JMS and HTTP Transport header nodes, TLOG Sample



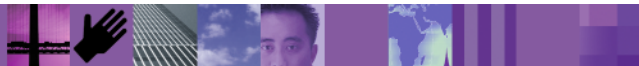
Administration & Systems Management

- Support for BEA WebLogic JMS XA
- Monitoring and Auditing Support
- ITCAM for SOA support



Platform Support and Performance

- CVP/IVP for ALL platforms
- Citrix support for remote operations

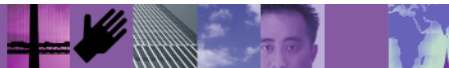


WebSphere Message Broker Summary

- Message Broker is a key IBM integration technology
 - Industry leading performance in a broad range of scenarios
 - Unparalleled range of integration options and capabilities
 - Supports users' range of experience and needs

- Five key themes satisfying a broad range of customer requirements
 - Ease of Use and Productivity
 - Enhanced SOA support
 - Administration & Systems Management
 - Extended Connectivity
 - Platform Support and Performance

- Builds on success of Version 6
 - Introduces significant new opportunities for high performance integration



Resources

- Which ESB on System z? Selection Guidelines for WebSphere Message Broker, WESB and DataPower XI50 <http://www.ibm.com/software/os/systemz/telecon/30jul/>
- Message Broker for z/OS V6.1.0.2 article <http://www-01.ibm.com/software/tivoli/features/ccr2/ccr2-2008-09/message-broker-6102.html>
- Considerations for making System z your ESB deployment platform: ftp://ftp.software.ibm.com/software/websphere/integration/wmq/WSW11335-USEN-00_esb_onZ_wp.pdf
- RedBook: Implementing an ESB using IBM WebSphere Message Broker V6 and WebSphere ESB V6 on z/OS <http://www.redbooks.ibm.com/abstracts/sg247335.html?Open>
- The Value of WebSphere Message Broker Version 6 on z/OS <http://w3-103.ibm.com/software/xl/portal/viewcontent?type=doc&srcID=XT&docID=M831619I05799D48>



Thank
YOU





Additional Information: Message Broker V6.1 What's new



Easy to Move to Version 6.1

▪ Migration

- Support migration from V5 and V6
 - Event Broker V5 and V6 migrated to Message Broker 6.1
- Compatibility
 - Message flows, message sets, ESQL, Java, Maps and XSLT run without change
 - Including 64 bit execution groups
- Rollback support
 - Migrate back to previous release with single command if necessary

▪ Coexistence

- V6.1 will co-exist with V5 and V6
- Enables incremental migration

▪ Production ready at GA (Nov 2007)

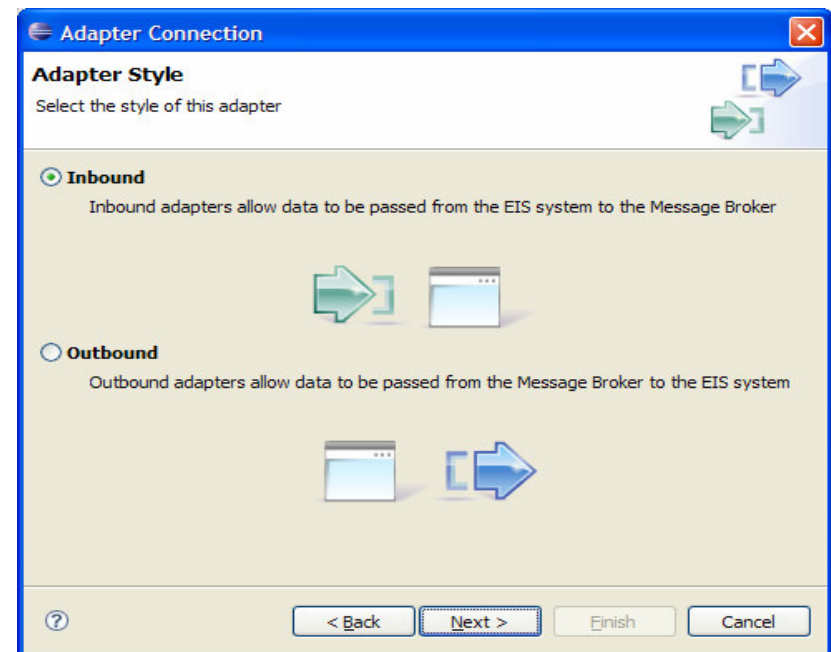
- Less defects than any existing version of Message Broker

Integrated Support for Major EIS Systems

- SAP for z/OS, SAP and PeopleSoft for zLinux
- WebSphere Adapters delivered “out of the box” as built-in nodes
 - Simplifies management and improves performance for key integration scenarios
 - These are the JCA based WebSphere adapters
 - Adapter license still required
- Support for inbound and outbound scenarios
 - Message-to-EIS and EIS-to-message scenarios
 - Adapter nodes integrate with all built-in MB nodes
- Enterprise Metadata Discovery (EMD)
 - Significant tooling support
 - Simplify for key data structure discovery
 - Accelerates generation of message sets
- High Performance access
 - Adapters access native message broker tree



SAP Input

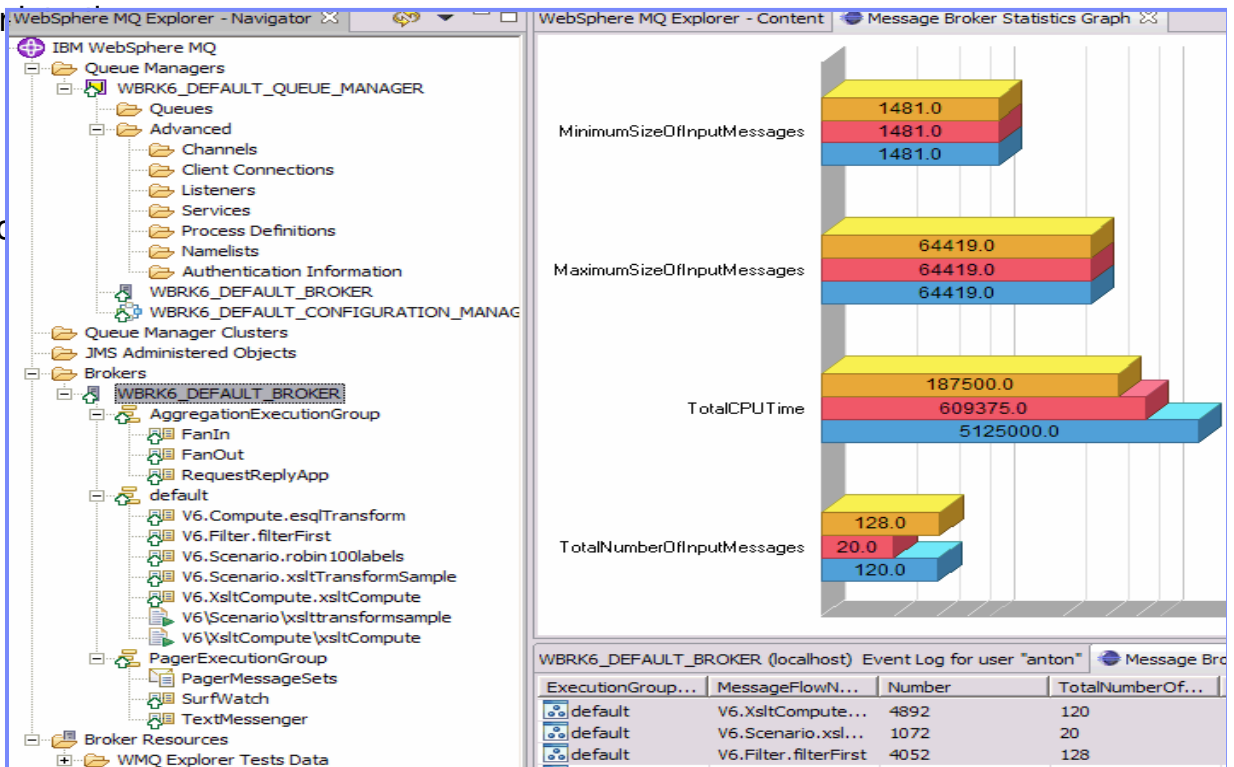


PeopleSoft Request



MB Explorer Eclipse Administration

- **Alternative MQ Explorer based administration**
 - Simplifies administration of MQ,MB networks in single Explorer console
 - Uses MQ Eclipse 'extension points' to provide seamless experience
- **Comprehensive administration facilities**
 - All features in Broker Administration
 - Also includes new features
- **IS02 Cat 3 support Pac**
 - Fully supported in production
 - Initial release July 2006
- **Performance Monitor**
 - Easily view CPU, IO and other metrics in Eclipse
 - Available March 2007



Advanced Security Features

- Message Broker now has powerful runtime security model
 - Supports cross domains security processing
 - Identity, Authentication and Authorization are native capabilities
 - MQ, HTTP, JMS, Web Services transports can all provide identity
 - Attributes on input and output nodes
 - Eclipse editor for security profile administration
- Policy decision points technologies: LDAP, TFIM

	TFIM	LDAP
Authorization	Yes	Yes
Authentication	Yes	Yes
Identity mapping	Yes	No

- Rich identity context supported
 - Type can be Username/password or X509 certificates
 - Token from default or user defined message location
 - e.g. {type=USERNAME, token=user, issuedBy=org, appliesTo=flow}
 - IssuedBy can be default or user defined
 - AppliesTo is fully qualified flow name resource Broker.ExecutionGroup.Flow
 - Identity appears in Message Tree
- ALSO in 6.1: Simplified Basic Authentication for WS and HTTP request nodes
 - TFIM can add username/password certificate to request

MQInput Node Properties - MQInput

Identity token type

Identity token location

Identity password location

Identity issuedBy location

Treat security exceptions as normal exceptions

MyApplicationFlow.cmf

Additional Instances

Commit Count

Commit Interval

Consumer Policy Set

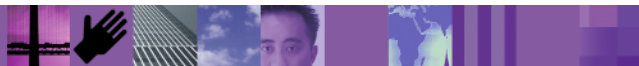
Consumer Policy Set Bindings

Coordinated Transaction

Provider Policy Set

Provider Policy Set Bindings

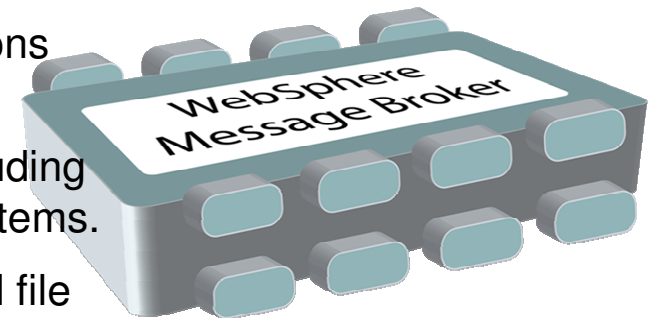
Security Profile Name



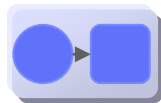
WebSphere Message Broker on System z

Built for universal connectivity and transformation in heterogeneous IT environments

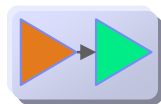
- Connect FROM anything TO anything: the broadest range of transport, protocol, data format and transformation capabilities
- Flexible and function rich ESB: address a wide range of requirements encompassing both existing & new, applications and services.
- Tightly integrated and optimized for the z/OS platform, including specific features for MQ, DB2, CICS, IMS and RRS subsystems.
- Advanced features such as Complex Event Processing and file based integration including VSAM.



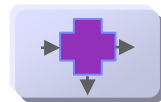
Exploits the unparalleled reach and reliability of the WebSphere MQ enterprise messaging backbone



Integrates everything through standard protocols, WebSphere Adapters for enterprise applications, and specialized connectivity options



Enables transformation between a wide range of data formats, including XML, legacy, and industry standards, and custom formats



Optimized for high-volume processing and rapid time to value for complex mediation requirements with a robust set of pre-built mediation function

