IBM SOA講堂 - WebSphere 動靜皆宜：WebSphere MQ 7.0新功能介紹及與PM4Data(檔案傳輸服務) & WTX(資料轉換引擎)的整合

Owen Chang
WebSphere Technical Sales Specialist
IBM Taiwan Software Group
owench@tw.ibm.com

Agenda

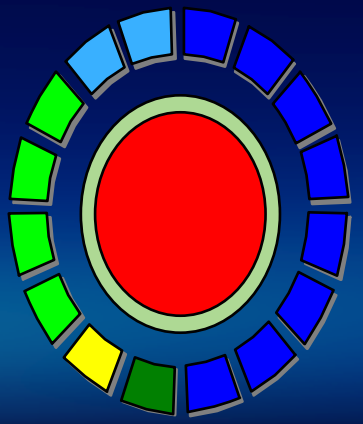
- **WebSphere MQ v7.0 新功能介紹**
- **WebSphere MQ Integrate With WTX(整合協同服務)資料轉換引擎**
- **WebSphere MQ Integrate With PM4DATA(檔案傳輸服務)**
- **Summary**

Next Version Objectives

- Improved support for "standards-based" messaging
  - Underpinning SOA and ESB architectures
  - Ease-of-use
  - Performance
- Extension of publish/subscribe capabilities
  - Leading to simplification with Message Broker
- Easier programming in any environment
  - Some features suggested by JMS requirements are useful in MQI
- Common across Distributed and z/OS codebases

Platform Coverage

- Essentially the same platforms as V6
  - Minor updates to base OS levels
    - Linux based on RH4 or Suse 9; AIX V5.3; z/OS 1.8 (GA 3Q06)
- Drop Linux/zSeries 31-bit
  - 64-bit edition will continue
- Drop Windows 2000
  - Windows XP is base level
  - Will include Vista
- Windows 2003 x86-64 support
  - Adds 64-bit application support
  - Supporting existing 32-bit applications
  - Some exits will require recompiling to support both 32 and 64-bit modes
- Java 1.4 and later

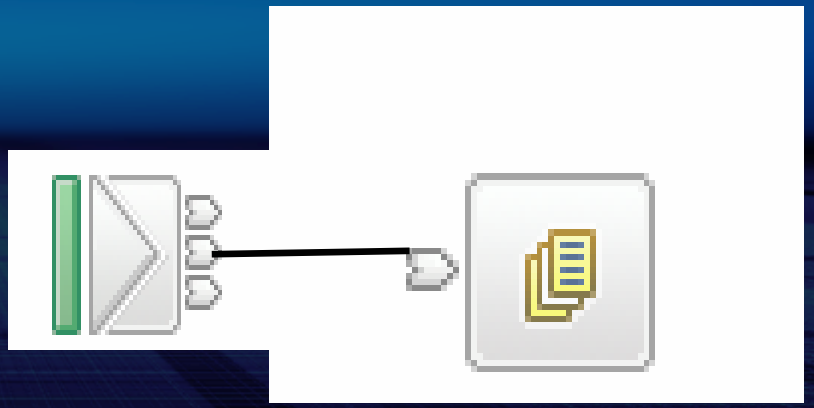


Publish/Subscribe

- 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
- A natural part of the JMS API
  - Combined both Publish/Subscribe and Point-to-Point styles
  - Want to make it a natural part of the native MQI
- V5.3 and V6 (Distributed) included a Publish/Subscribe broker (nee MA0C)
- Also have Message Broker and Event Broker products
- Goal is to simplify and improve ...

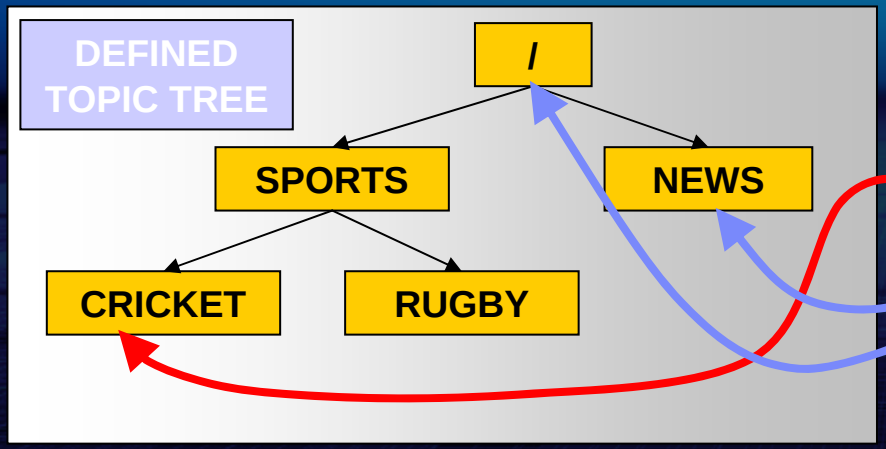
Relationship with Message Broker

- Next version of Message Broker will prereq this version of WMQ for pub/sub
  - But WMB V6 will continue to work with new WMQ releases
  - Migration can be done one piece at a time
- The publish/subscribe engine is removed from the broker
  - The Nodes will call the WMQ engine transparently
  - Message Flows do not need to be changed
- Administration tooling will be integrated with WMQ tooling
  - One place to configure WMQ and WMB resources
  - V6 SupportPacs show direction
- Looking for any other optimisations ...
  - Vision is all TRANSPORT is in WMQ
    - For example, multicast



Publish/Subscribe Admin

- Based on Topics
- 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
- 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 Admin (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 ...
  - No "cleanup" task needed for non-durable
- Subscriptions
  - Able to see who is subscribing to topics: like DISPLAY QSTATUS
  - Able to create subscriptions on behalf of a third party
- Status and Statistics
  - Items such as number of messages published on a topic
- 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 (RACF or OAM)

Publish/Subscribe Topologies

- WMQ V6 publish/subscribe networks based on hierarchies
  - All brokers linked in parent/child tree
- WMB publish/subscribe networks based on hierarchies of collectives
  - All systems in a collective are connected to each other (mesh)
  - Also has "clones“
- New product will have hierarchies and pub/sub clusters (a.k.a. collectives)
  - With interoperability to other pub/sub systems through hierarchies
- Design gives
  - Scalability
  - Availability
  - Ease of administration
- Collectives based on WMQ clustering
  - Cluster can be defined independently of any existing cluster used for queueing

Publish/Subscribe MQI

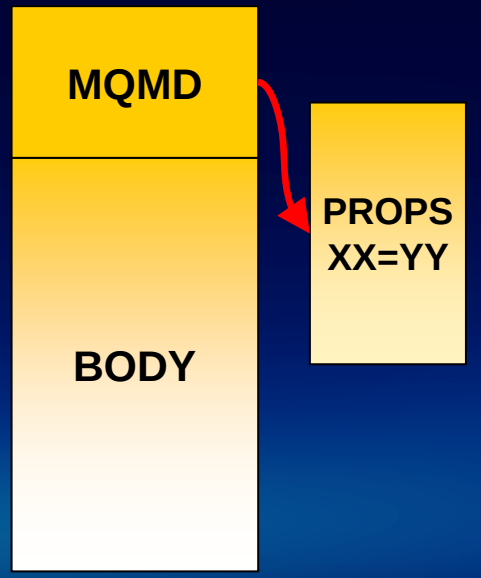
- Cannot change the JMS API
  - But we want to make some of its facilities more easily available in the MQI
  - To improve MQI programming and improve (make thinner) the JMS layer
- New verb for subscribing
  - So you do not need to build RFH or RFH2 headers in the application
  - **MQSUB** registers a subscription
    - Includes information about where messages will be read from
    - Do not need to specify a queue – can be automatically assigned
- New options on existing verbs
  - MQOPEN to get access to a topic
  - MQCLOSE will deregister a subscription
  - MQPUT, MQGET to publish and to receive subscriptions
  - MQSUBRQ (new verb) for initial state
- Conversion of point-to-point applications without code changes
  - Administrative changes to objects

Publish/Subscribe Application Migration

- New verbs remove need for some of the older interfaces
  - Which will be deprecated – though not removed immediately
  - PCF and RFH1 facilities for WMQ publish/subscribe applications: Identity, Streams
- New MQI operations will not be available for old VB and ActiveX programs
  - Use .Net classes instead
- A single application cannot mix new verbs with old options
  - Can't use RFH1(Register Publisher) and MQPUT(topic) in same program
- Separate daemon included to translate MA0C pub/sub commands
  - With some performance cost
  - Not needed when all publish/subscribe applications have been converted
- Migration step to convert existing WMQ and WMB subscriptions
- Most common "application" is IBM-supplied JMS/XMS layer
  - New client design to take advantage of new facilities when they exist
  - So installing new JMS jar or XMS dll will be all that's needed

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 **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
- Appear as RFH2 properties on older queue managers
- Permits explicit statement of relationships between messages
  - eg Message X is a **REPLY** to Message Y
  - Which can then be exploited by products looking for patterns



Using Properties – Possible Scenario

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

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

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

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

Other MQI Enhancements

- Asynchronous Message Reception
  - New verb **MQCB** defines a callback function
  - Automatically Invoked when a message arrives
  - No need for MQGET(WAIT) or MQGET(SIGNAL)

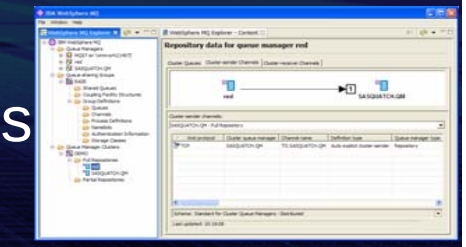
- Selectors
  - Use a SQL92 clause to select messages by properties including MQMD fields
  - Can be specified on MQOPEN, MQSUB for filtering messages
  - Filtering is normally done inside queue manager for efficiency
  - Not filtering on message body contents
    - Message Broker still required for content filtering

New Quality of Service

- Traditional WMQ non-persistent messages more reliable than some need
- Receiving Messages/Subscriptions:
  - Messages sent to a client in advance of MQGET, queued internally
  - Administrative choice – no application changes needed
  - Higher performance in client
- Sending/Publishing Messages:
  - Application can indicate it doesn't want to wait for the return code
    - Maybe pick up return code later – **MQSTAT** verb
  - Maintains transactional semantics
  - Higher performance in client
- Implementation also gives us more heartbeat opportunities
  - Faster failure notification for clients

WMQ Explorer Enhancements

- Grouping
  - Queue Managers can be partitioned into groups within the Navigator
  - eg "Test", "Production"
- Security Configuration
  - Easy to set channel exits, userid/password configurations per queue manager
    - Still recommend a SVRCONN security exit for authentication
  - ... or globally for all queue managers in a group or workspace
- Tighter JMS integration
  - Creating an queue/topic can define a JMS destination at the same time
- 3rd party plug-ins can create new property pages for an object
  - Makes them look even more integrated
- Investigating alternative distribution mechanisms
  - eg Eclipse Update site



Other Ease-of-use Items

- Looking to produce items such as
  - More sample programs
  - Demonstrations
  - Default configurations
- Default configuration wizard will create all objects needed
  - For JMS and for publish/subscribe operation
- No need for command to start/stop a broker
  - The "broker" process disappears unless needed for compatibility of old apps
  - All queue managers will have publish/subscribe function automatically available

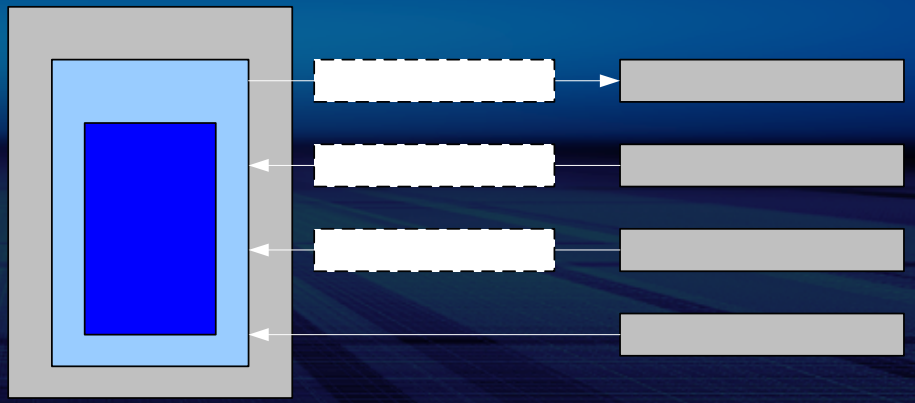
HTTP-MQ Goals

- Simplify access to MQ Apps from Rich Internet Applications
  - Gives AJAX and Web 2.0 access to the Enterprise
    - Submit data direct to queues & topics from Browser
    - Low Latency Web Pub/Sub
      - Stock price update, Sports scoreboard, Airport and Rail Departures / Arrivals notification
- Enable MQ Application Connectivity from *any* Platform or Language with HTTP capabilities
  - Significantly increase range of supported platforms
    - e.g. – Linux distros, POS terminal running Windows Services for Unix environment, RFID reader, Mobile devices
    - No client library installation required
- Lightweight (low qualities of service) messaging

Background: HTTP-MQ is Loosely Modeled on REST

REpresentational State Transfer (Roy Fielding)

- **Everything is modeled as a Resource**
- **Every resource is identified by an address (URI)**
- **Resources have state (representation)**
- **HTTP is used to transfer state to networked application**
- **HTTP verbs operate on the resource**
  - **GET** → retrieves a resource's state representation
  - **POST** → Updates resource (or other processing)
  - **DELETE** → deletes resource
  - **PUT** → Creates / updates resource state



Summary

- Improvements to publish/subscribe
- Improvements to JMS/XMS layer
- Ease-of-use for administrators
- Ease-of-use for MQI programmers
- Performance
- Tighter integration with Message Broker
- Continues to extend the enterprise messaging foundation

Agenda

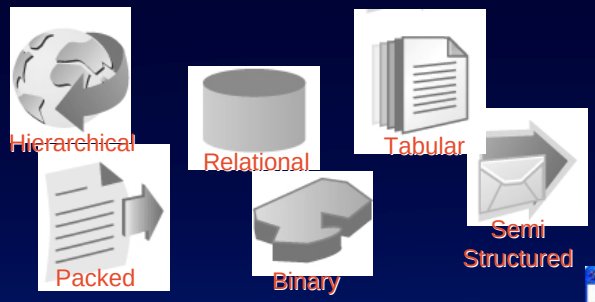
- **WebSphere MQ v7.0新功能介紹**
- **WebSphere MQ Integrate With WTX(整合協同服務)資料轉換引擎**
- **WebSphere MQ Integrate With PM4DATA(檔案傳輸服務)**
- **Summary**

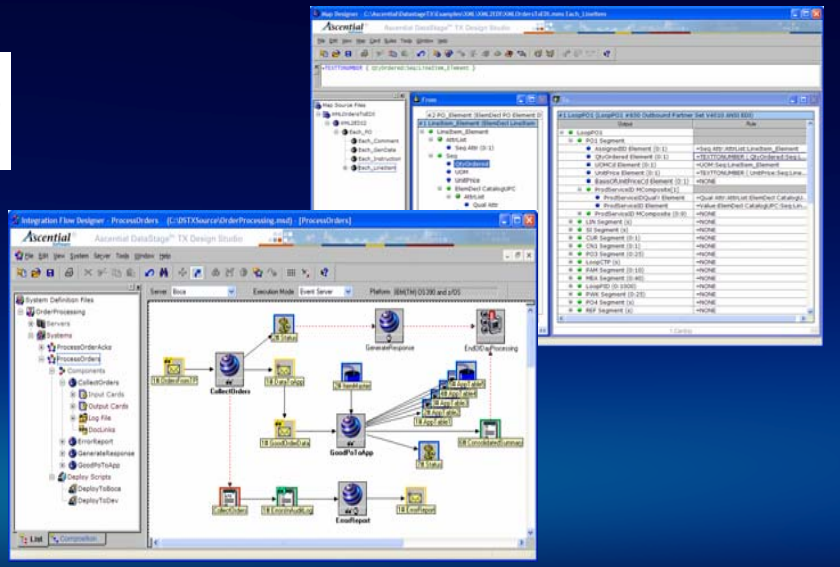
What does WebSphere TX do?

It's a complex transformation engine that extends the value of DS Enterprise Edition

It takes any kind of data from its native form

ANY Data





Processes them together, natively, with no Code

MANY to MANY Integration

And outputs them into their native target formats

ANY Data

Ex. - Hierarchical Data , Binary Data, Packed Data, Tabular Data, Relational Data, Nested Structures, Mixed-Type Data, and on and on…

Transformation Vision - Simplified

- Transform Anything
- Transform Anywhere
- Everyone Transforms

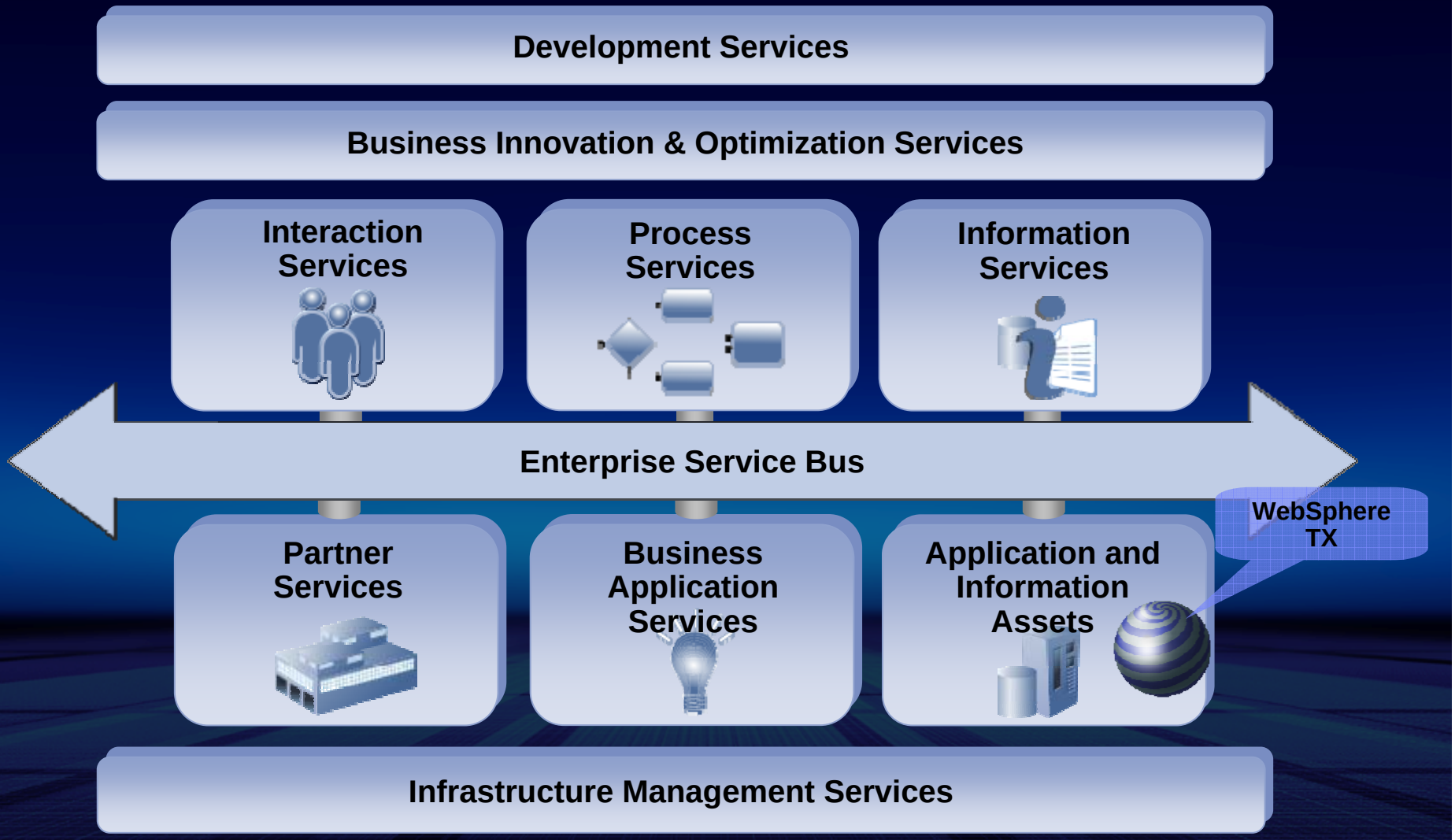
What Makes WebSphere TX so Different?

- One Engine, Multiple Deployment Options
- In-Process Data Validation
- Code-Free Design and Deployment
- Ontological Data Model for all Data Types
- High Throughput Execution of Complex Transformations and Enhancements

Examples of Complex Transformation Challenges Addressed by WebSphere TX

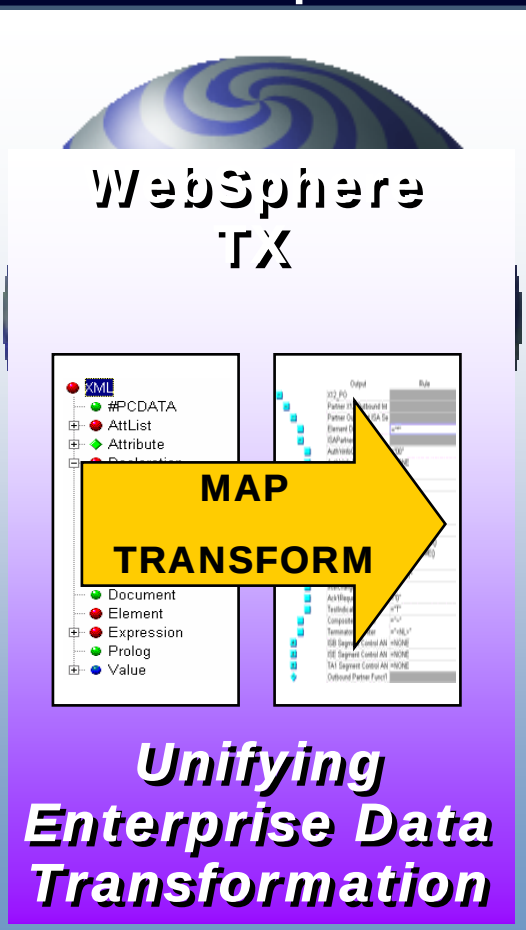
- Data Enhancement
  - Lookups
  - Data Logic and Routing
  - Data Validation
  - Context Based Data and Usage Rules
- Many to Many Transformation
  - Single-Transaction, interdependent data sets, conversions and logic
  - Mixed Data and Source/Target Types
  - Dependent Result Sets, Nested Structure Dependencies
- Complex Data Transformation
  - Nested, Semi-structured And Hierachical Data Types
  - Dependent Inputs And Outputs
  - Binary, Packed, EBCDIC, ASCII, Mixed Character Data

WebSphere Integration Reference Architecture



WebSphere TX: A Unified Transformation Environment across our portfolio

- ONE Transformation Engine for the Enterprise!
  - Powerful transform capabilities without coding
  - Natively handles any data type
  - Solves really hard transformation problems
- Complements existing



For batch environments (as a standalone engine) — Available Now

For WebSphere Message Broker — Available Now

For WebSphere Process Server

For WebSphere ESB

For WebSphere Partner Gateway

For 3rd Party Integration Products (as an embeddable engine) — Available Now

WebSphere : A Powerful, Modular Platform

Process Mgmt	= WebSphere PS
Event Mgmt	= WebSphere MB
Transformation	= WebSphere TX
Transport	= WebSphere MQ

WebSphere TX across the WebSphere Platforms

AIM Offerings

WTX - embedded
WebSphere Message Broker
Available Now

WTX – add-on
WebSphere Process Server

WTX – add-on
WebSphere ESB

WTX – add-on
WebSphere Partner Gateway B2B

WTX Server For Unix (AIX, Solaris, HPUX, Linux)
Available Now

WTX Server For z/OS (IMS, CICS, Batch, USS)
Available Now

WTX Server For Windows
Available Now

WTX Launcher - cross-platform - (For Existing Customers)

WTX SOA and Extended SOA Edition (RTI – Remains in IIS)

IIS WTX Offerings

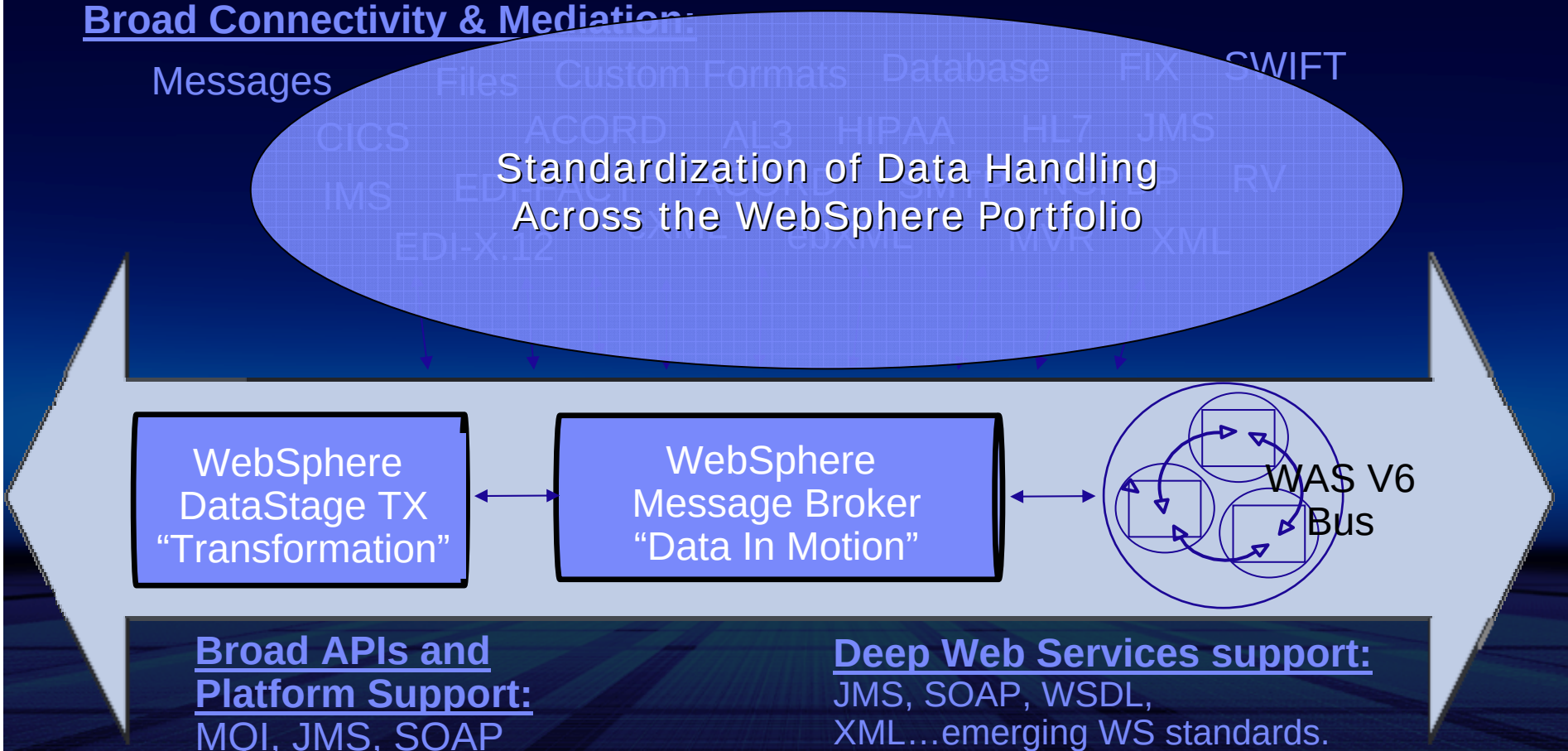
MapStage (WTX OEM version)
WebSphere DataStage
Available Now

WTX Packs are available for all offers

- SWIFT
- HIPAA
- NCPDP
- HL7
- EDI,EDIFACT
- TRADACOMS
- ODETTE
- Etc..

Available Now

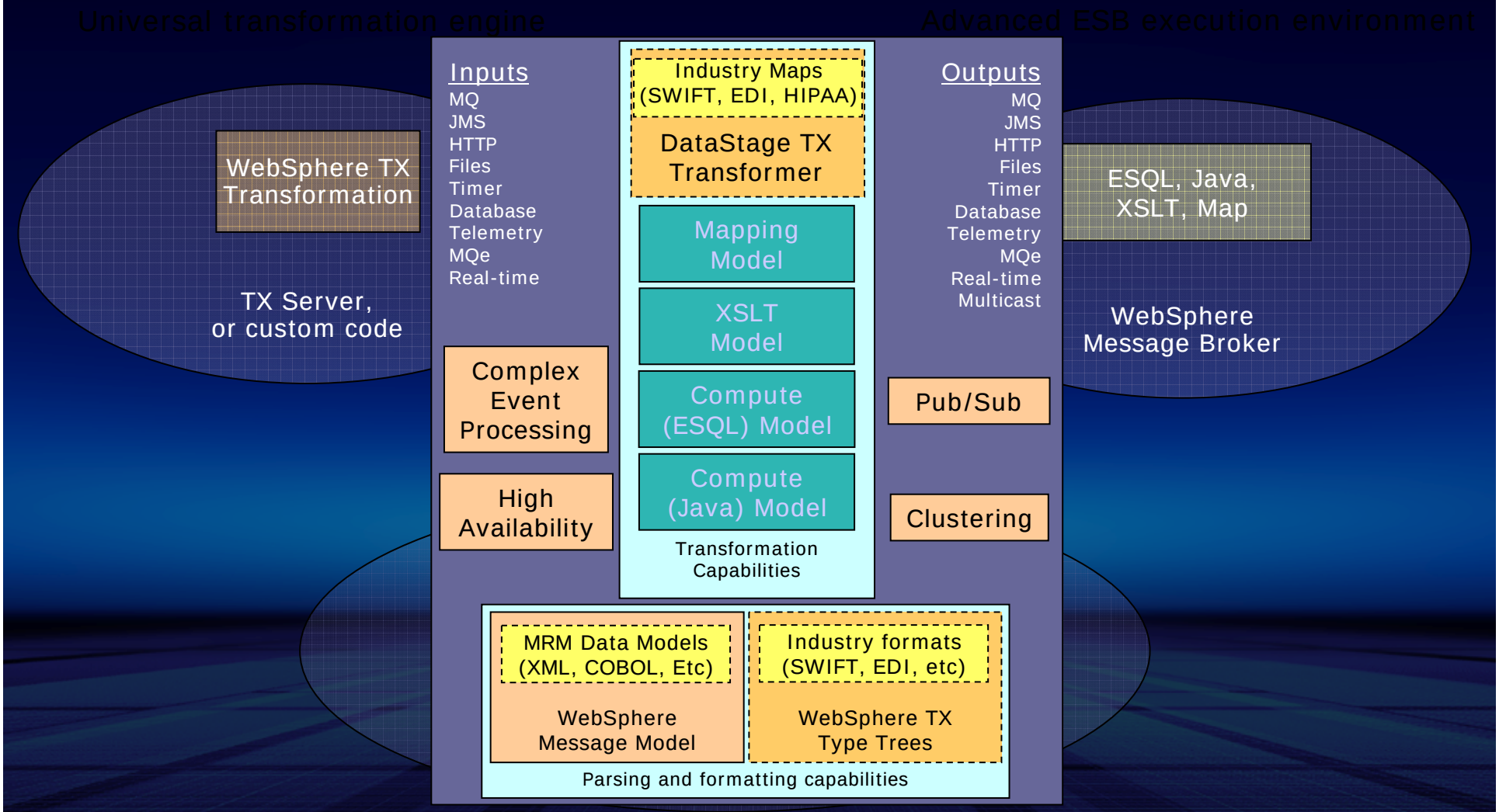
WebSphere TX and
The IBM Enterprise Service Bus Advanced Edition



WebSphere TX Adds a Powerful Transformation Environment to our ESB Portfolio

- Provides a universal transformation engine
  - Powerful transform capabilities without coding
  - Natively handles any data type
  - Solves really hard transformation problems in
- Complements existing portfolio of products
  - WebSphere MQ
  - WebSphere Message Broker
  - WebSphere ESB
- Continues to complement IIS portfolio of products (MapStage offering)
- Delivers new opportunities for sales to customers with mainframe, batch, and third party environments

Combined power of two industry leading technologies

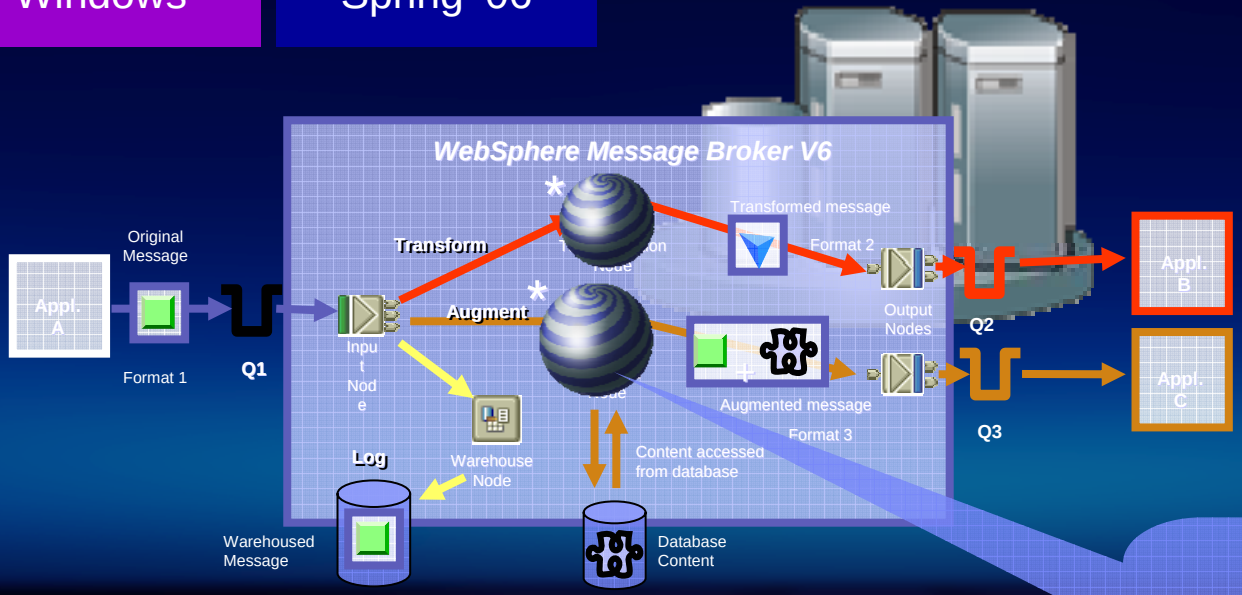


WebSphere TX Extender for WebSphere Message Broker

Unix or Windows

USS Available Spring '06

"Powerful ESB with Powerful, Universal Transformation"



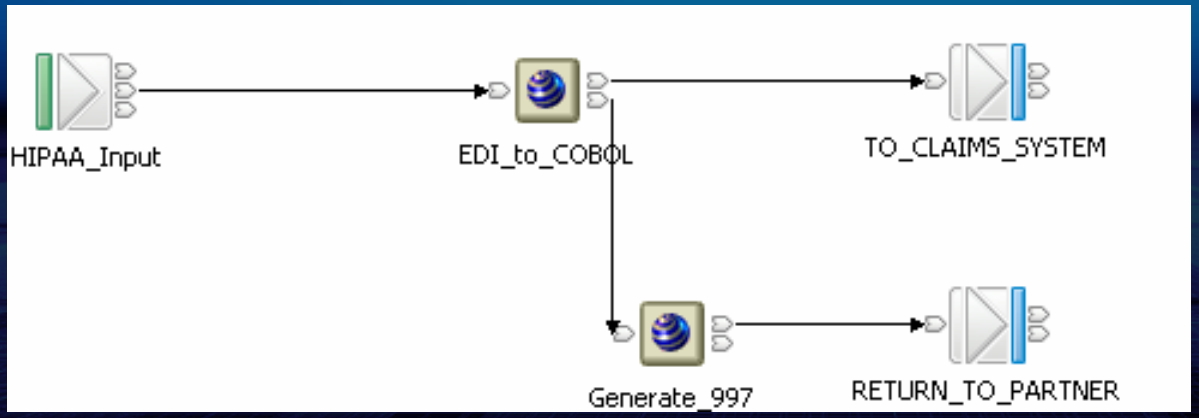
*WebSphere TX Extender for WebSphere Message Broker

One method of transformation regardless of data type or transformation challenge

It is the Power of WTX transformation on top of the Power of the WMB Platform

WMB Plug-in components

- Parser
  - Transforms bit streams to Message Trees (MQInput) and Message Trees to bit streams (MQOutput)
  - Transforms Message Trees to Message Trees (RCD)
- Plug-in node
  - Transforms anything to anything
  - Icon installs into Message Flow palette
  - Easy configuration

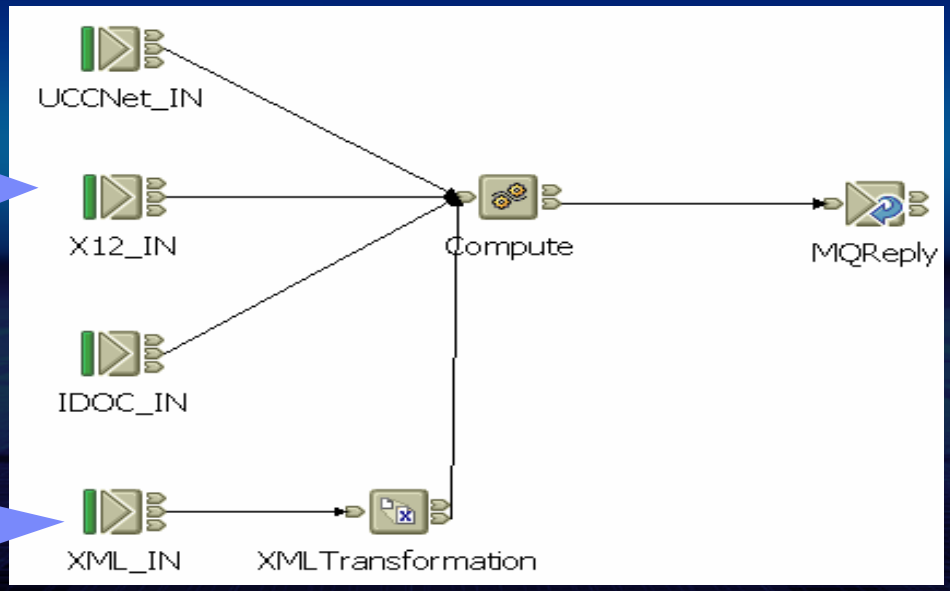


WMB Parser Use Cases

- Convert any input stream to standard Message Tree
- Reuse existing ESQL
- Mix and match WTX and standard parsers
- No need for custom written (c coded) Parsers

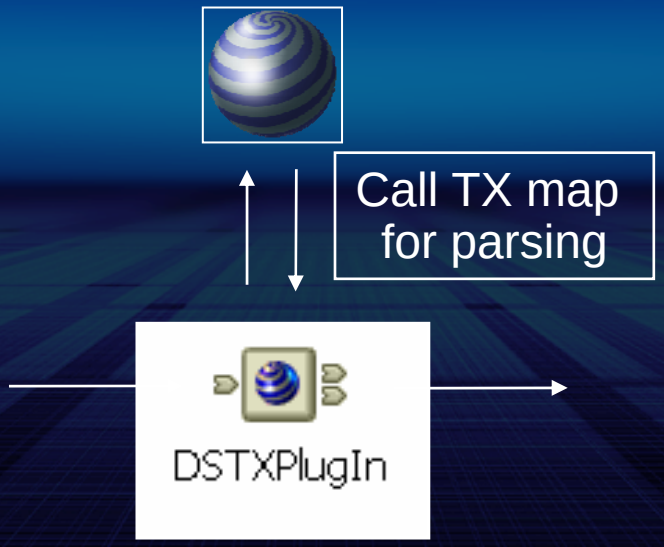
Custom WTX Parsers

Standard XML Parser



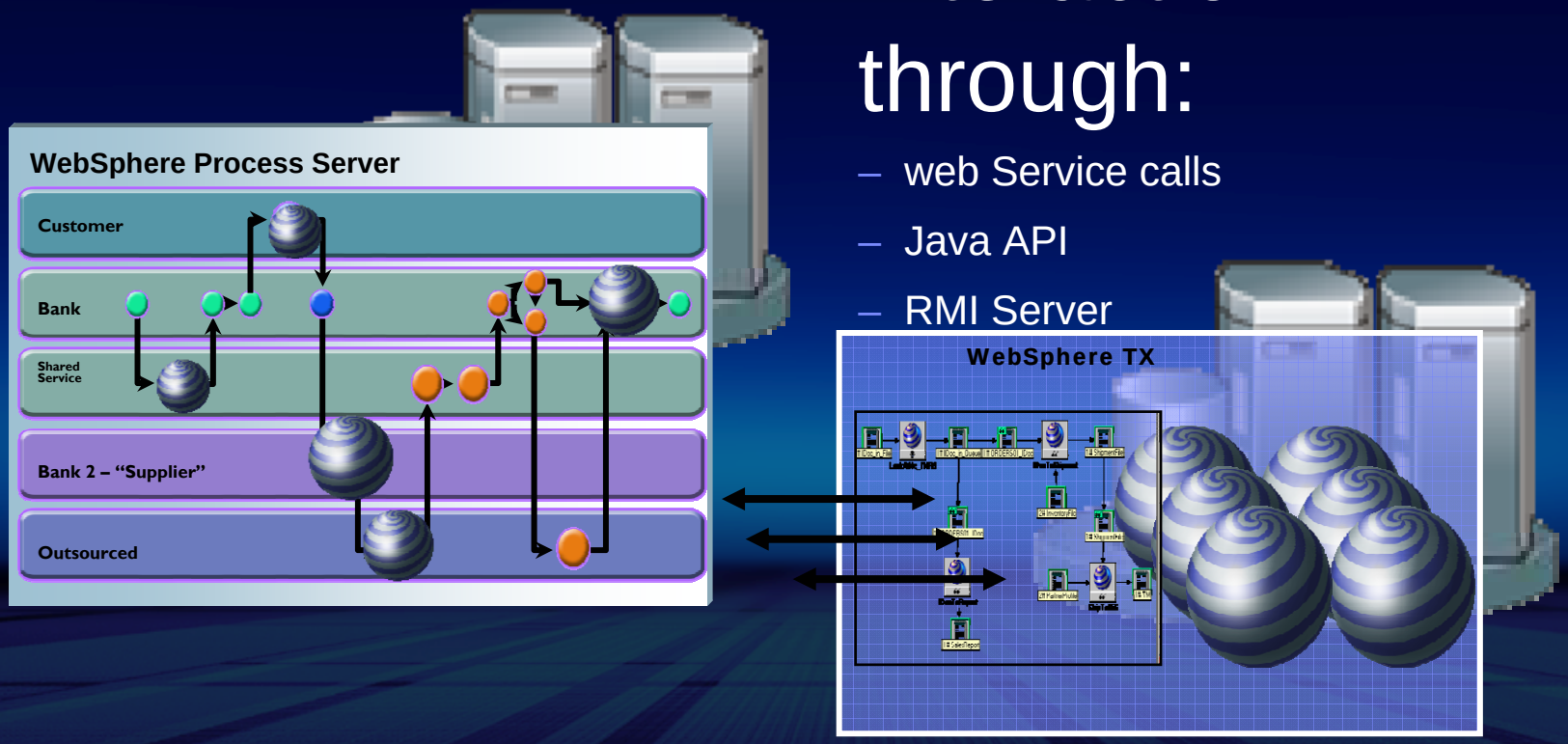
WebSphere TX plugin node for WebSphere Message Broker

- Can be used to call any WebSphere TX map from within a Message Broker flow
- Transaction Control of message broker is maintained
- Message Broker passes BLOB to WebSphere TX map
- Plugin can pass BLOB back to message broker, or can parse to any other format: XML, MRM, ...



WebSphere TX and WebSphere Process Server

- Interaction through:
  - web Service calls
  - Java API
  - RMI Server



WebSphere Message Broker Adds Rich Enterprise Service Bus Connectivity Services

Transport Services

- ✓ Assured delivery
- ✓ Secure delivery
- ✓ Transactional delivery
- ✓ Manageable delivery
- ✓ Delivery replay
- ✓ Modifiable qualities of transport.

Event Services

- ✓ Event detection
- ✓ Event triggering
- ✓ Event distribution
- ✓ Complex Event Processing (CEP).

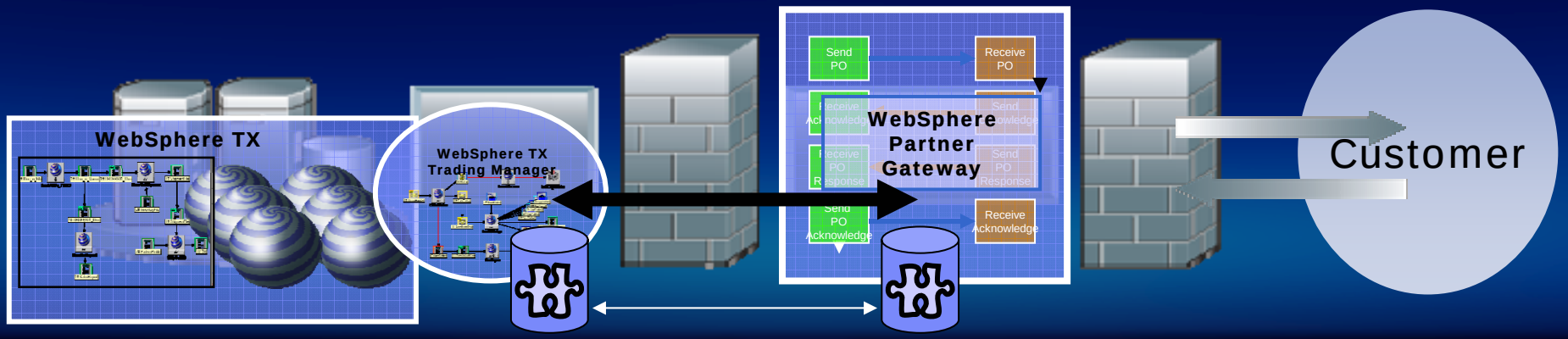
Mediation Services

- ✓ Routing
- ✓ Transport switching
- ✓ Programming model switching
- ✓ Transformation & content augmentation
- ✓ Customized communications.

Message Broker brings these to WTX

WebSphere WPG + TX
Combined Implementation Offering

Full B2B Solution Implementation



Agenda

- **WebSphere MQ v7.0 新功能介紹**
- **WebSphere MQ Integrate With WTX(整合協同服務)資料轉換引擎**
- **WebSphere MQ Integrate With PM4DATA(檔案傳輸服務)**
- **Summary**

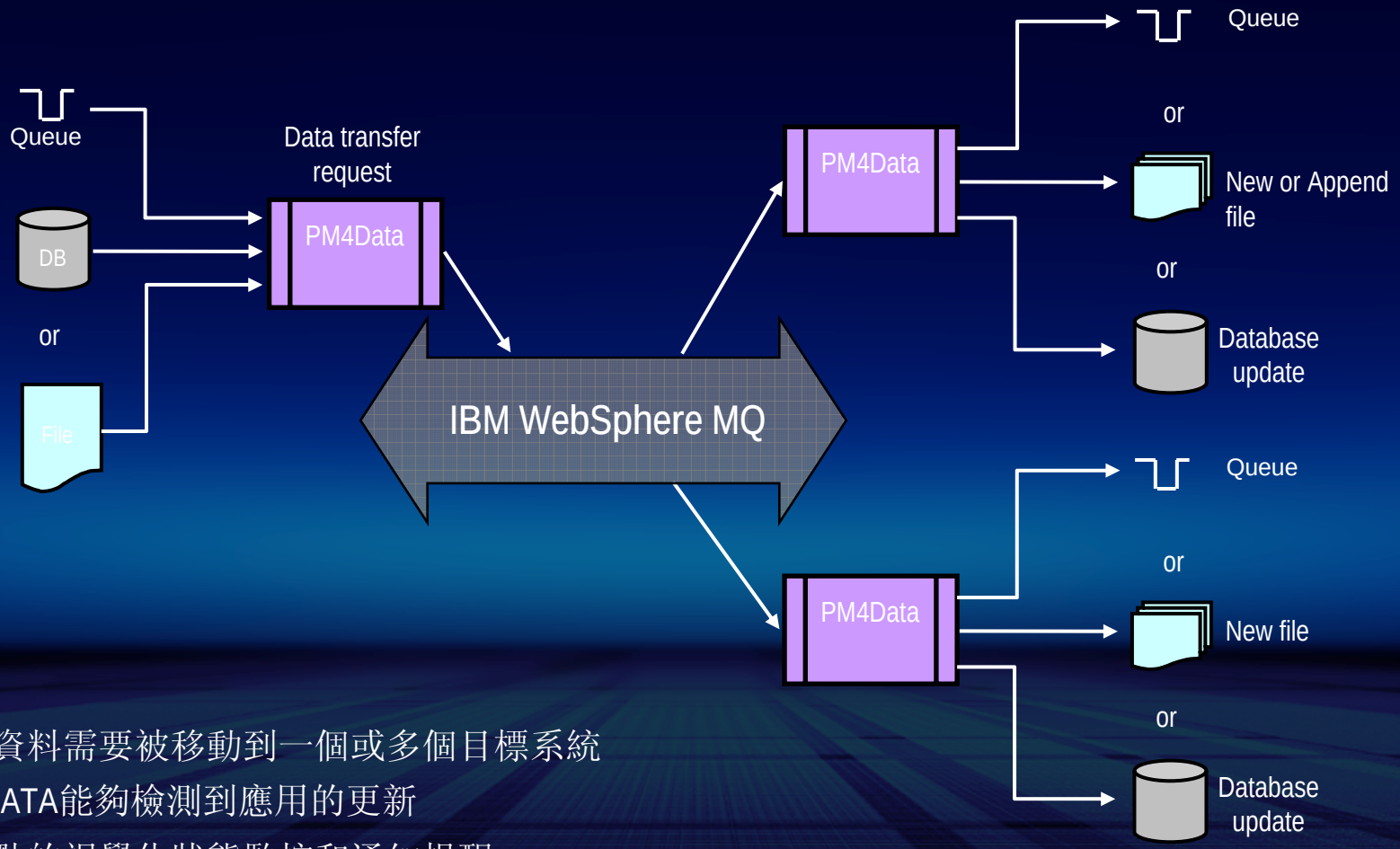
內容

- 什麼是Process Manager for Data?
- 特點和功能
- 可管理的資料交換
- 工具
- 支援的平臺
- 支援的檔案類型
- 如何與PM4DATA整合

什麼是Process Manager for Data (PM4Data)?

- 基於WebSphere MQ企業級解決方案，用於解決資料和檔案傳輸問題
- 架構具有可擴展性
- PM4Data提供如下服務的保證:
  - 高效能，雙向多文件並行傳輸
  - 安全性
  - 多平臺覆蓋
  - 集中監控和點對點的視覺化管理
  - 提供資料的追蹤和監控、例外處理和容錯機制
  - 大資料量傳輸基礎架構
  - 利用成熟技術提供自動的整合解決方案
  - 該解決方案可以增強現有檔案和資料爲基礎的技術資產，提供更多業務價值
  - 強大的企業資料整合平臺，可擴展的強大能力實現資料移動的整合、管理和控制以及跨各類異構平臺的整合任務

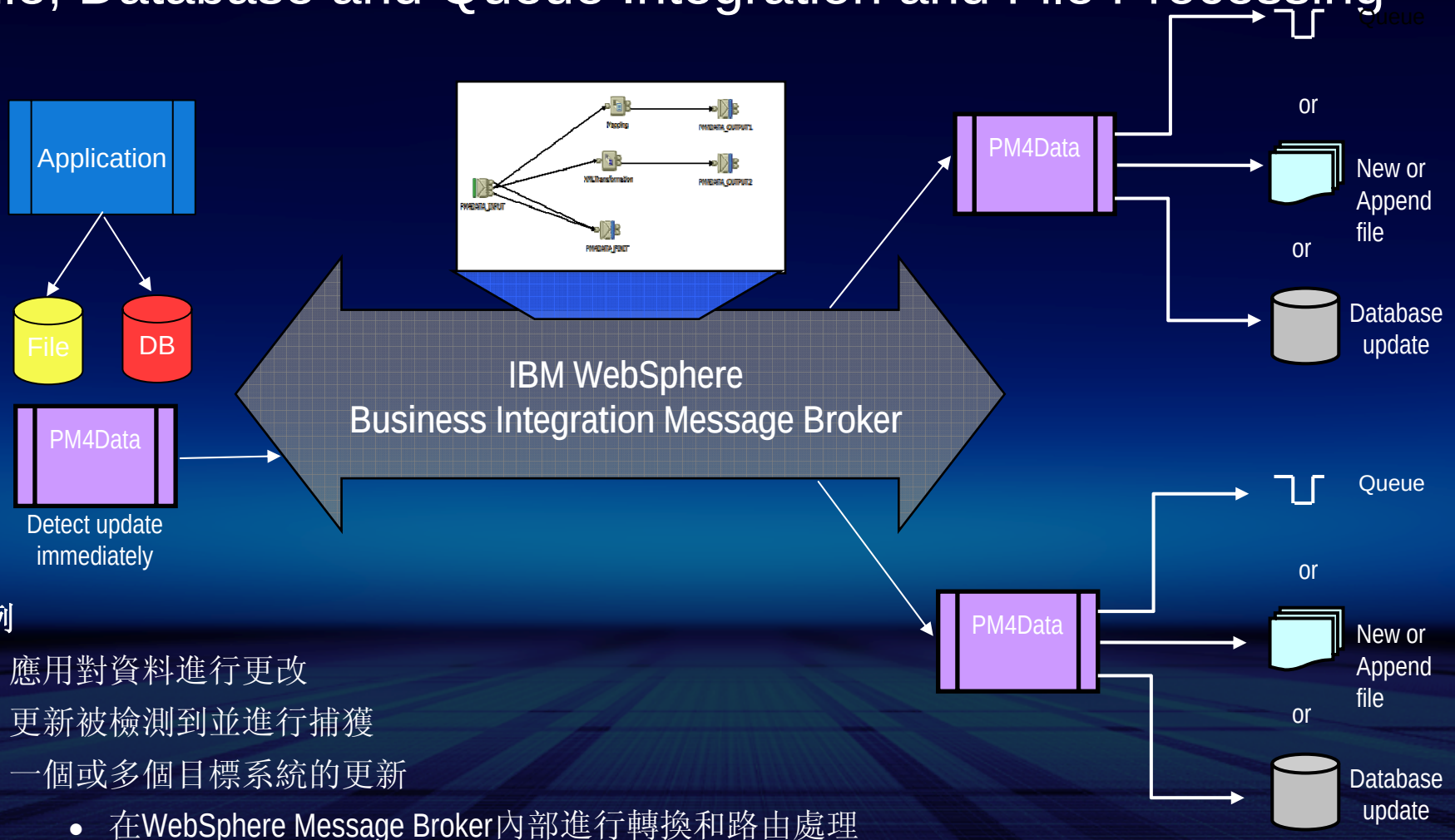
PM4DATA
File, Database and Queue Integration



舉例

- 來源資料需要被移動到一個或多個目標系統
- PM4DATA能夠檢測到應用的更新
- 點對點的視覺化狀態監控和通知提醒

PM4DATA
File, Database and Queue Integration and File Processing



舉例

- 應用對資料進行更改
- 更新被檢測到並進行捕獲
- 一個或多個目標系統的更新
  - 在WebSphere Message Broker內部進行轉換和路由處理
- 點對點的視覺化狀態監控和通知提醒

可管理的資料移動

- 複雜的、可靠的、安全的、高效能的、大資料量的可管理的資料傳輸解決方案，利用WebSphere MQ可支援非同步的、交易性的資料傳輸能力
- 在傳輸操作中，自動處理平臺和檔案類型的差異
- 通過客製化出口提供可擴展的能力
- 高安全性和資料壓縮能力
- 點對點的資料傳輸視覺化
- 多叫用方法，包括XML訊息介面

可管理的資料移動

Source Platform
MQSeries
Queue
Manager
PM
ata
File
HTML/XML
HTTP/S
Managing
Platform
Status
WebSphere MQ Channels
Target Platform
MQSeries
Queue
Manager
PM4Data
點對點詳細的傳輸狀態

可管理的資料移動

Source Platform
MQSeries
Queue
Manager
PM
ata
File
HTML/XML
HTTP/S
Managing
Platform
Status
WebSphere MQ Channels
Target Platform
MQSeries
Queue
Manager
PM4Data
點對點詳細的傳輸狀態

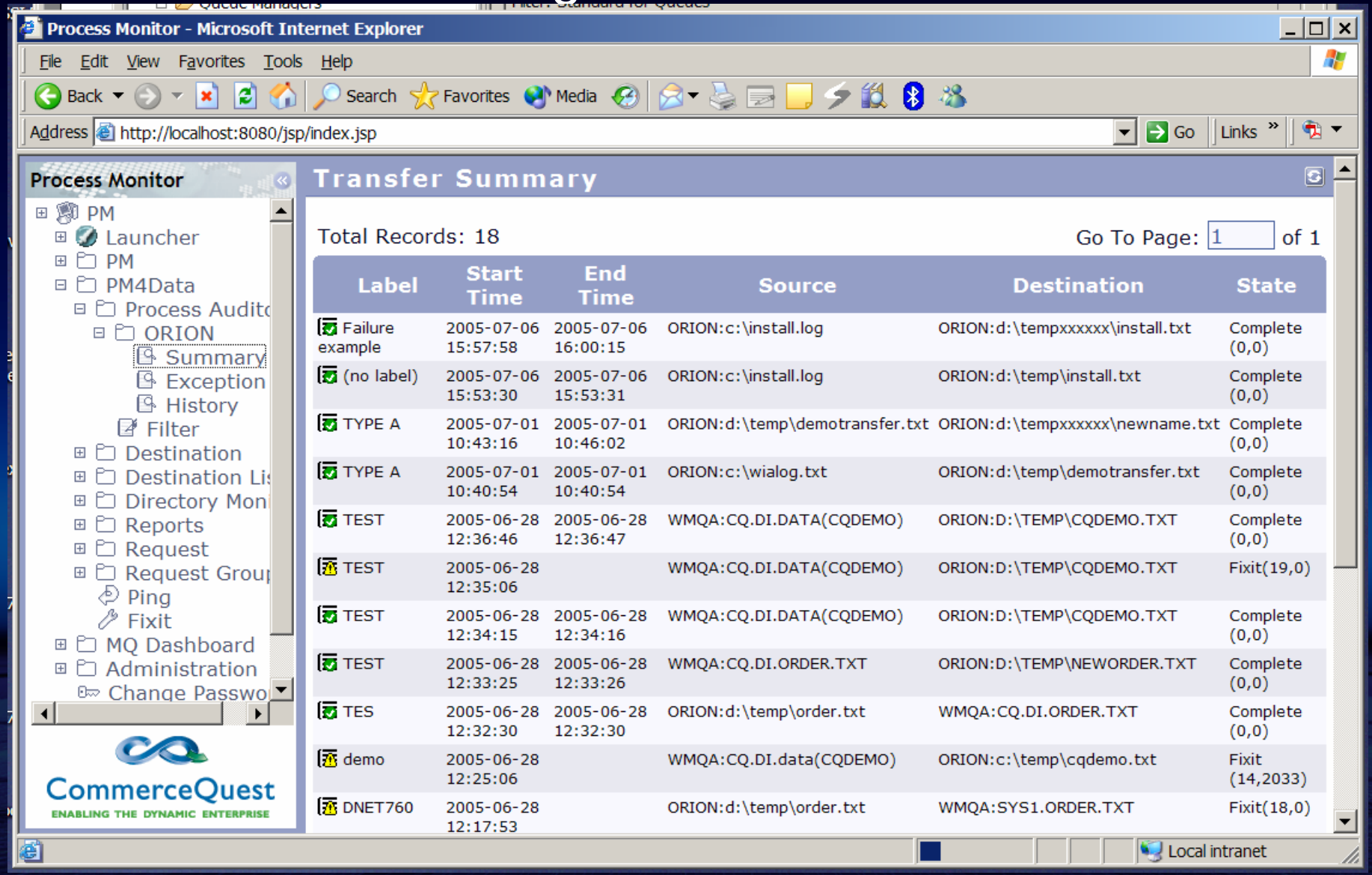
可管理的資料移動

Source Platform
MQSeries
Queue
Manager
PM4Data
File
HTML/XML
HTTP/S
Managing
Platform
Status
WebSphere MQ Channels
Target Platform
MQSeries
Queue
Manager
PM4Data
點對點詳細的傳輸狀態

可管理的資料移動

Source Platform
MQSeries
Queue
Manager
PM4Data
File
HTML/XML
HTTP/S
Managing
Platform
Status
WebSphere MQ Channels
Target Platform
MQSeries
Queue
Manager
PM4Data
點對點詳細的傳輸狀態

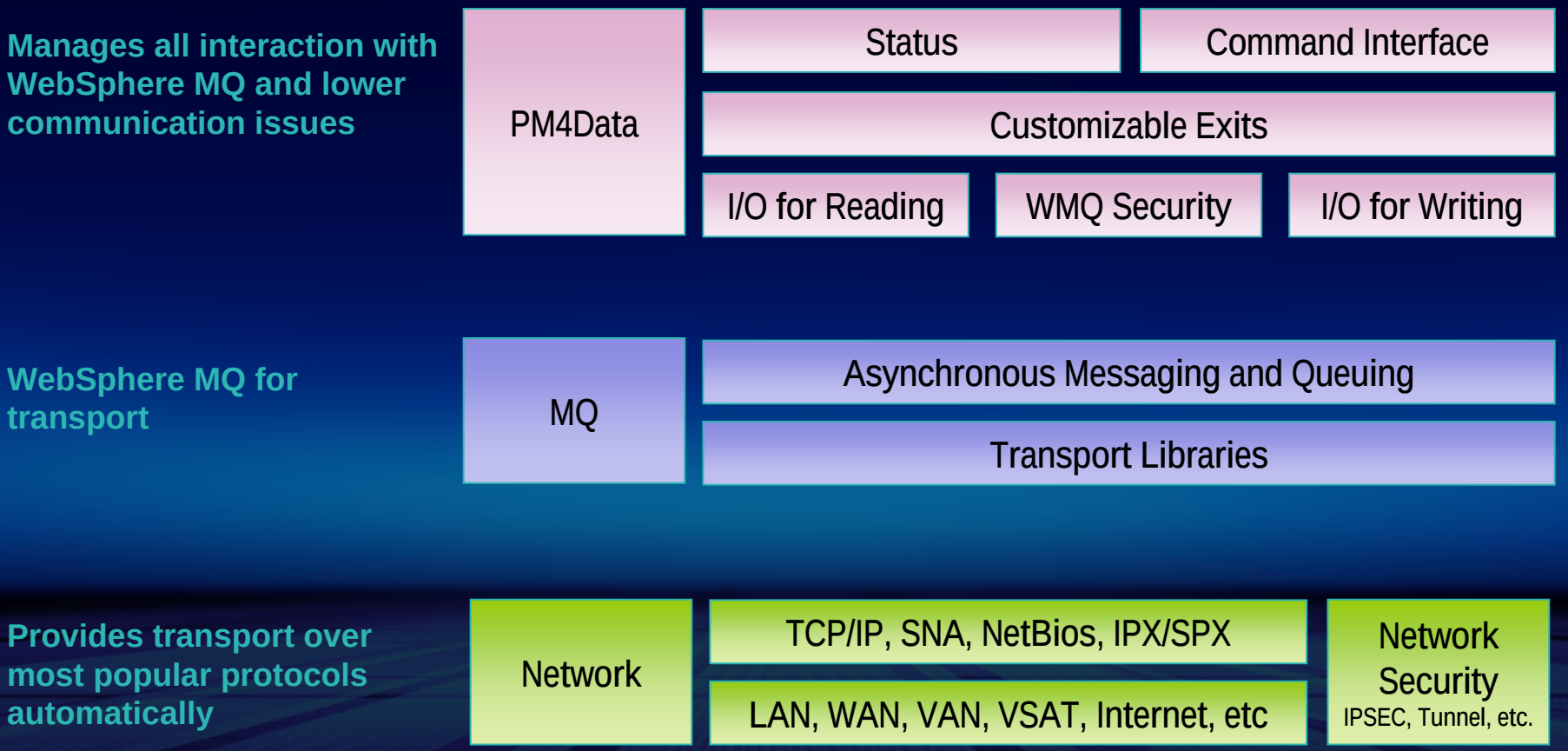
PM4Data Managed Data Movement Status



可管理的資料移動 – 優點

- 服務質量保證 (Quality of Service)
  - 可靠性
  - 安全性
  - 高效能的壓縮資料傳輸
  - 大量數據
  - 非同步傳輸
  - 事物完整性
- 在傳輸操作中，自動處理平臺和檔案類型的差異性
- 通過客製化出口提供可擴展能力
- 多種介面叫用方式，例如XML介面叫用
- 點對點的視覺化傳輸
- 傳輸全過程監控和統計

可管理的資料移動 - 設計架構



支援的平臺

- Microsoft Windows 2000 and Windows 2000 Server
  - w/SP1 or SP2
- Microsoft Windows Server 2003
  - Standard Edition
  - Enterprise Edition
  - Web Edition
- Microsoft Windows XP Professional w/SP1 or SP2
- Hewlett-Packard HP-UX 11.00
- IBM AIX 5.1, 5.2 or 5.3
- IBM z/OS V 1.4 or 1.5 (Not all features supported)
- IBM OS/400 V5 R1 or higher (Not all features supported)
- Red Hat Enterprise Linux ES 3 for Intel x86
- Sun Solaris 8 and 9
- SuSe LINUX Enterprise Server 9

支援的資料庫

Platform	Supported Databases
AIX 5.1, 5.2 and 5.3	DB2 UDB 7.2, 8.1 and 8.2 Oracle 8i R3, 9i R2 and 10g R1
HP-UX 11.0	DB2 UDB 7.2, 8.1 and 8.2 Oracle 8i R3, 9i R2 and 10g R1
Solaris 8 and 9	DB2 UDB 7.2, 8.1 and 8.2 Oracle 8i R3, 9i R2 and 10g R1
OS/400 V5R1	DB2 for iSeries V5R1
Windows 2000 and XP	DB2 UDB 7.2, 8.1 and 8.2 Oracle 8i R3, 9i R2 and 10g R1 SQL Server 2000 with Service Packs 1, 2 and 3
z/OS V1.4 and V1.5	DB2 UDB for z/OS V6.1 and V7.1

支援的檔案類型

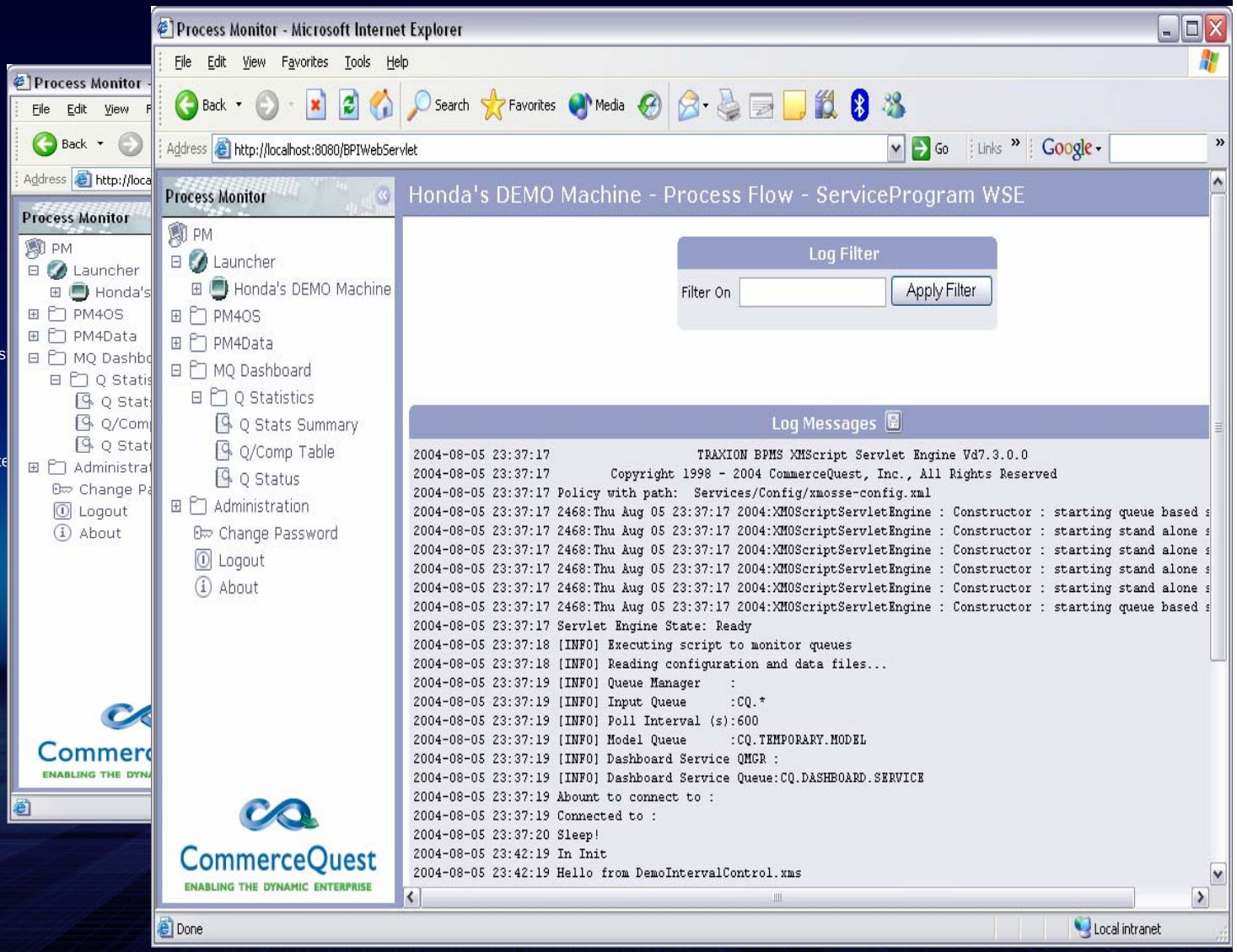
- Windows and UNIX
  - Text
  - Binary
- MVS
  - Physical Sequential Data Sets (PS)
  - Partitioned Data Sets (PDS)
  - Partitioned Data Sets Extended (PDSE)
  - VSAM Data Sets
  - Generation Data Groups (GDG)
  - Hierarchical File System (HFS)/UNIX System Services
  - ASA Data Sets
- OS/400
  - Logical Files
  - Physical Files (including Source Physical Files)
  - Integrated File System (IFS)

內容

- 什麼是Process Manager for Data?
- 特點和功能
- 可管理的資料交換
- 工具
- 支援的平臺
- 支援的檔案類型
- 如何與PM4DATA整合

流程監控 – 傳輸管理

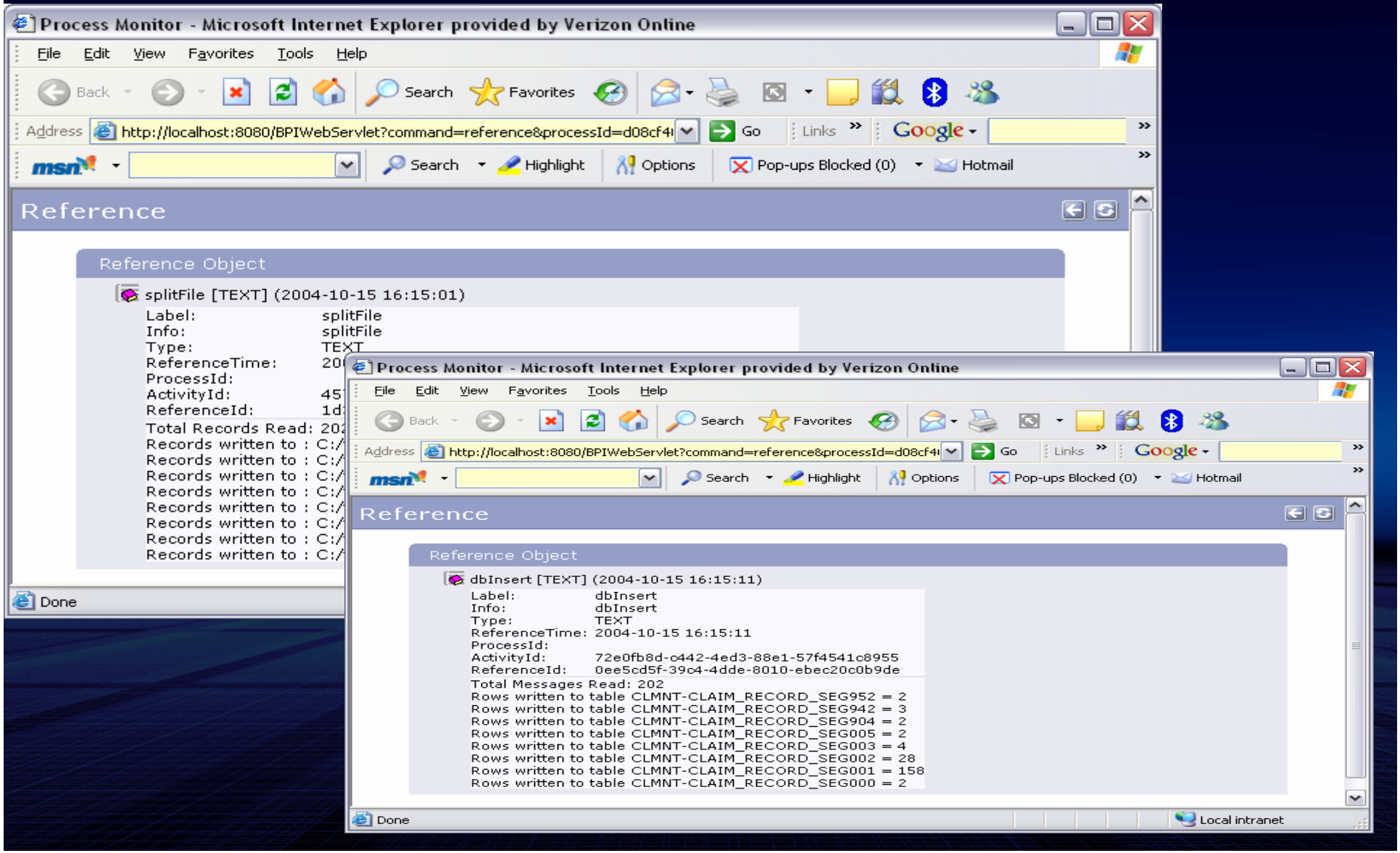
- Web 伺服器支援: jboss(included)
  - Status
  - User information
  - Stored requests
- 狀態顯示
  - End-to-End transfer status
- 訪問控制
  - User management, with roles & rights
- 傳輸請求管理
  - Ping nodes
  - Create, save, and submit transfer reques
  - Transfer grouping
  - Directory monitoring
- 管理報告
  - Measure transfer success and failure rate



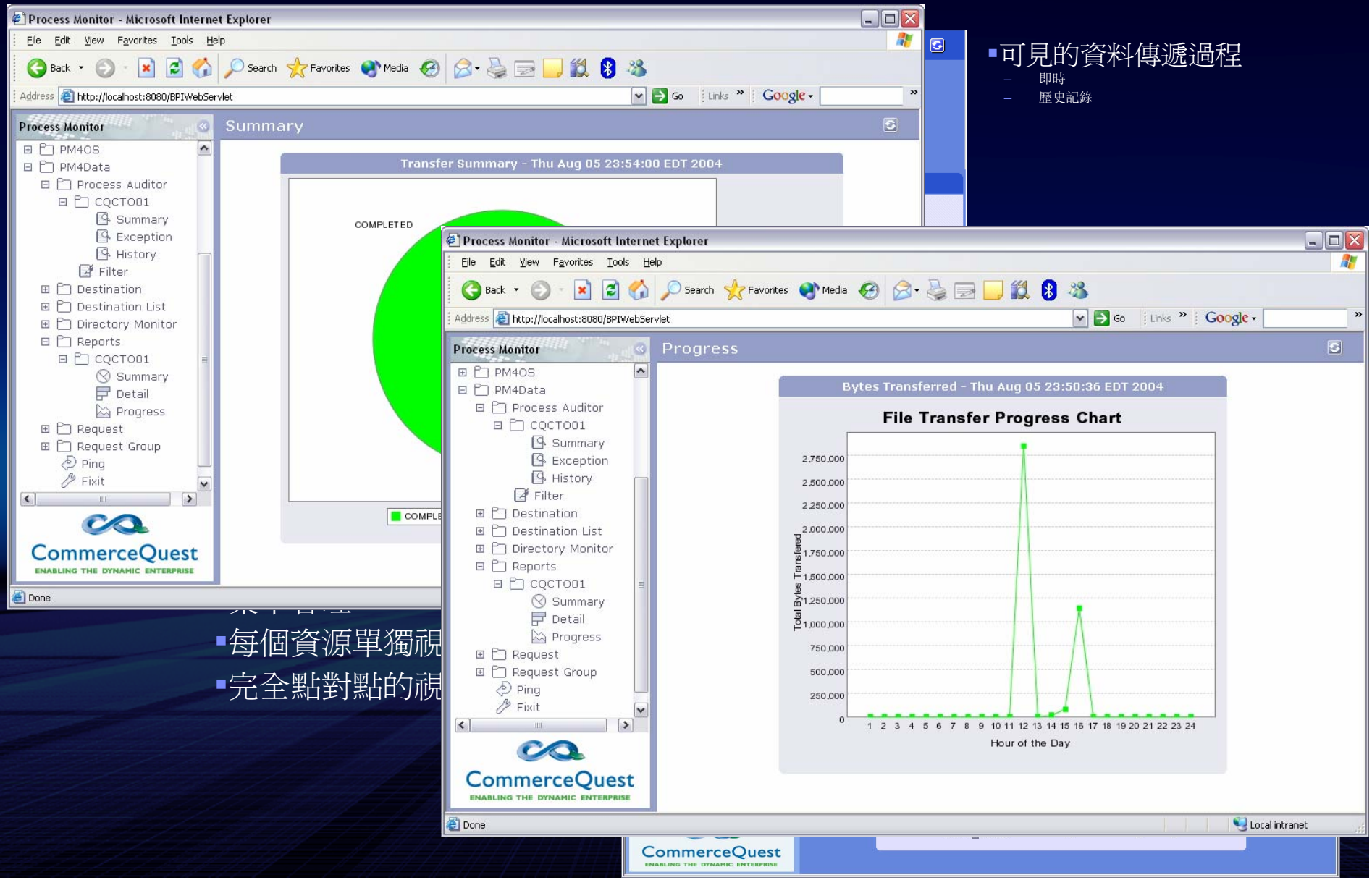
狀態監控

State	Description	Icon
In progress	Transfer request has been submitted, but is not complete. This state can occur when processing is ongoing, or when the transfer has hit a barrier, such as an inactive channel.	
Completed	Transfer request has been submitted and successfully completed.	
Warning	Transfer has not failed, but is not completing successfully.	
Failed	Transfer has been set to a state of Failed in FixIt.	

狀態報告


Process Monitor - Microsoft Internet Explorer provided by Verizon Online
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/BPIWebServlet?command=reference&processId=d08cf4
Go Links Google
Search Highlight Options Pop-ups Blocked (0) Hotmail
Reference
Reference Object
splitFile [TEXT] (2004-10-15 16:15:01)
Label: splitFile
Info: splitFile
Type: TEXT
ReferenceTime: 20
ProcessId:
ActivityId: 45
ReferenceId: 1d
Total Records Read: 202
Records written to : C:/
Records written to : C:/
Records written to : C:/
Records written to : C:/
Records written to : C:/
Records written to : C:/
Records written to : C:/
Records written to : C:/
Done
Process Monitor - Microsoft Internet Explorer provided by Verizon Online
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/BPIWebServlet?command=reference&processId=d08cf4
Go Links Google
Search Highlight Options Pop-ups Blocked (0) Hotmail
Reference
Reference Object
dbInsert [TEXT] (2004-10-15 16:15:11)
Label: dbInsert
Info: dbInsert
Type: TEXT
ReferenceTime: 2004-10-15 16:15:11
ProcessId:
ActivityId: 72e0fb8d-c442-4ed3-88e1-57f4541c8955
ReferenceId: 0ee5cd5f-39c4-4dde-8010-ebec20c0b9de
Total Messages Read: 202
Rows written to table CLMNT-CLAIM_RECORD_SEG952 = 2
Rows written to table CLMNT-CLAIM_RECORD_SEG942 = 3
Rows written to table CLMNT-CLAIM_RECORD_SEG904 = 2
Rows written to table CLMNT-CLAIM_RECORD_SEG005 = 2
Rows written to table CLMNT-CLAIM_RECORD_SEG003 = 4
Rows written to table CLMNT-CLAIM_RECORD_SEG002 = 28
Rows written to table CLMNT-CLAIM_RECORD_SEG001 = 158
Rows written to table CLMNT-CLAIM_RECORD_SEG000 = 2
Done
Local intranet

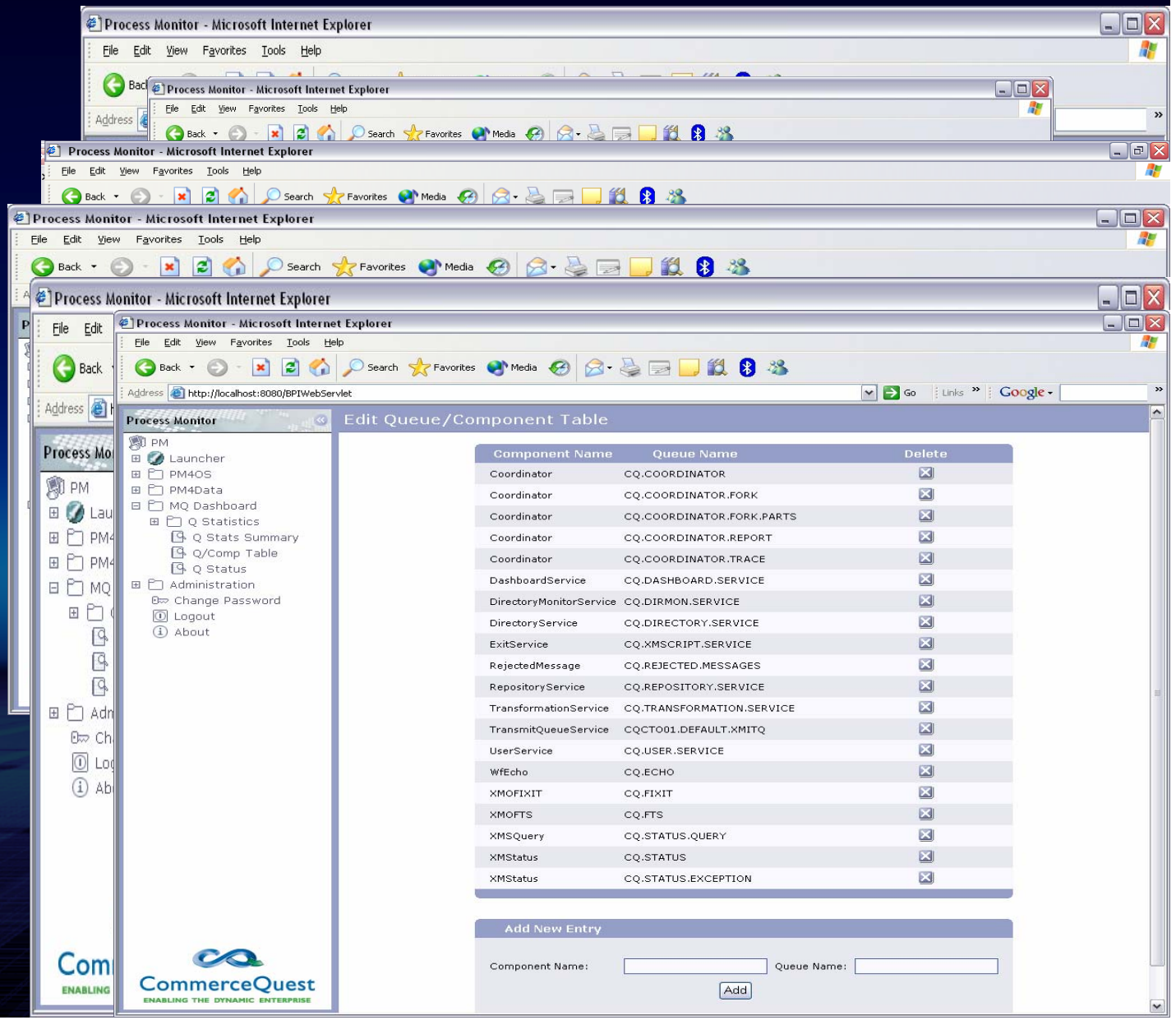
流程監控 – 效能



- 可見的資料傳遞過程
  - 即時
  - 歷史記錄
- 每個資源單獨視
- 完全點對點的視

WMQ Dashboard

- WebSphere MQ Enterprise Dashboard
  - Component / Application Level Queue Statistics
  - Graphical views per Application / Queue
  - Daily Summary
  - Operational Views
  - Queue Content views
  - Problem determination aids.
  - Application Queue Grouping



PM4Data的優勢

- 平臺獨立
- 可靠
- 可恢復
- 高可用
- 可擴展
- 低客戶影響
- 可審計
- 充分利用WebSphere MQ的能力
- 易維護
  - 資料改變對客戶的影響最小
  - 將處理從後臺系統和元件中獨立出來
- 充分利用WebSphere MQ的投資
- 非同步處理流程
- 分散式系統
- XML支持
- 數據轉化

Smart SOA

धन्यवाद
Hindi

多謝
Traditional Chinese

Grazie
Italian

ขอบคุณ

Спасибо
Russian

Obrigado
Brazilian Portuguese

Merci
French

Gracias
Spanish

شكراً
Arabic

多谢
Simplified Chinese

Danke
German

நன்றி
Tamil

ありがとうございました
Japanese

감사합니다

WebSphere software