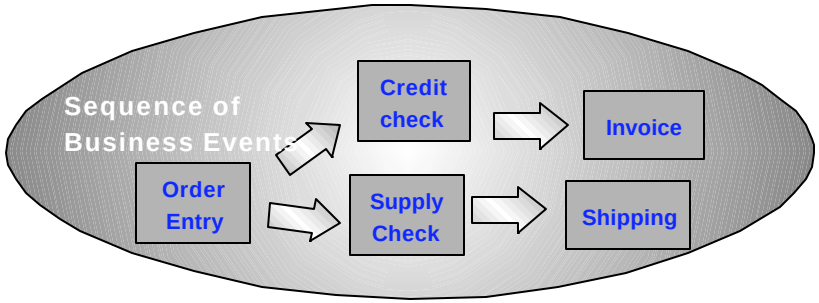
Using MQSeries Workflow and MQSeries Integrator

Frank Skrzypczak
Claudia Zentner

Agenda

- Introduction
  - BPM
  - MQWF and MQSI characteristics
- Interoperability between MQWF and MQSI
  - MQWF's XML message interface
  - Call-In Sample Scenario
  - Call-Out Sample Scenario
  - Event Sample Scenario
  - Support Pacs
- Sample Scenario

Business Process Management



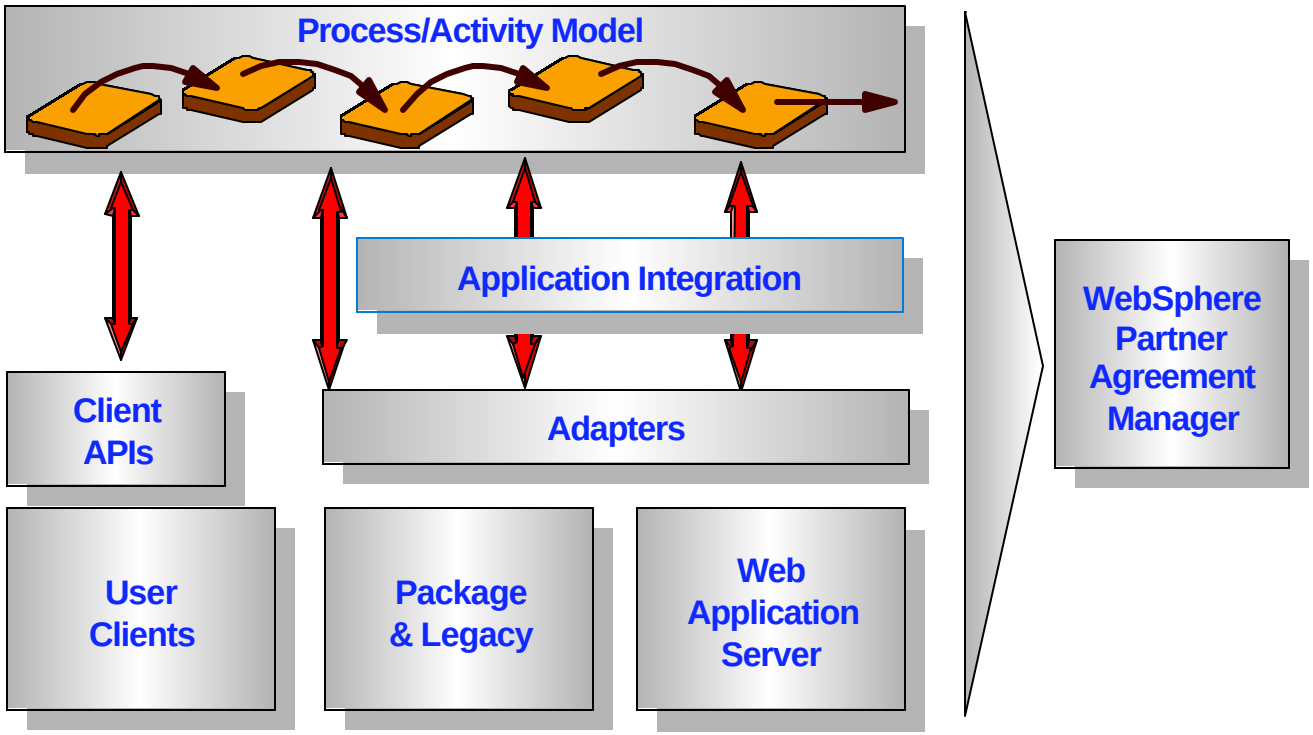
What

Define/Manage processes
(MQSeries Workflow)

How

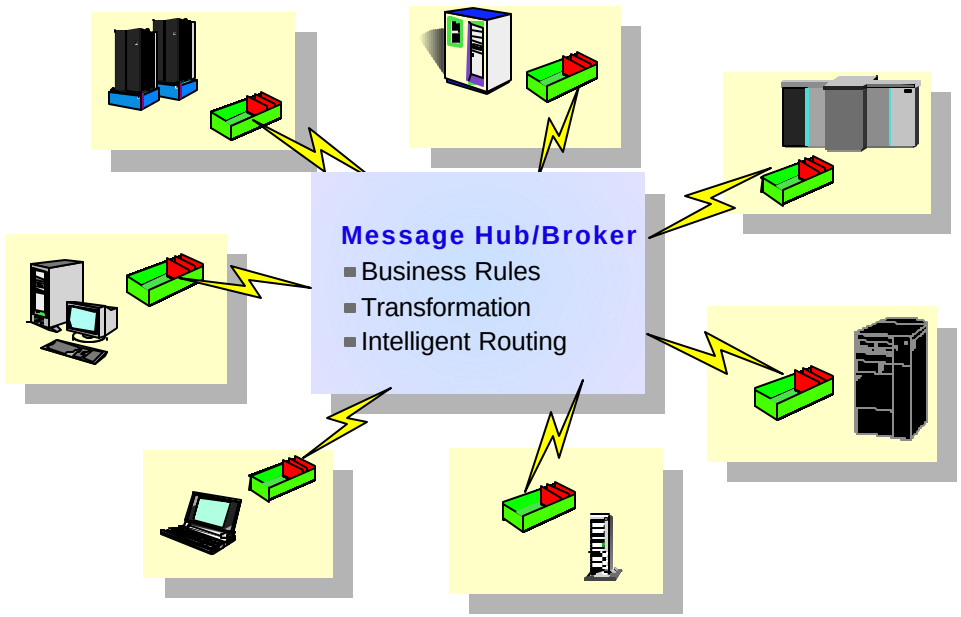
Manage your information
(MQSeries Integrator)

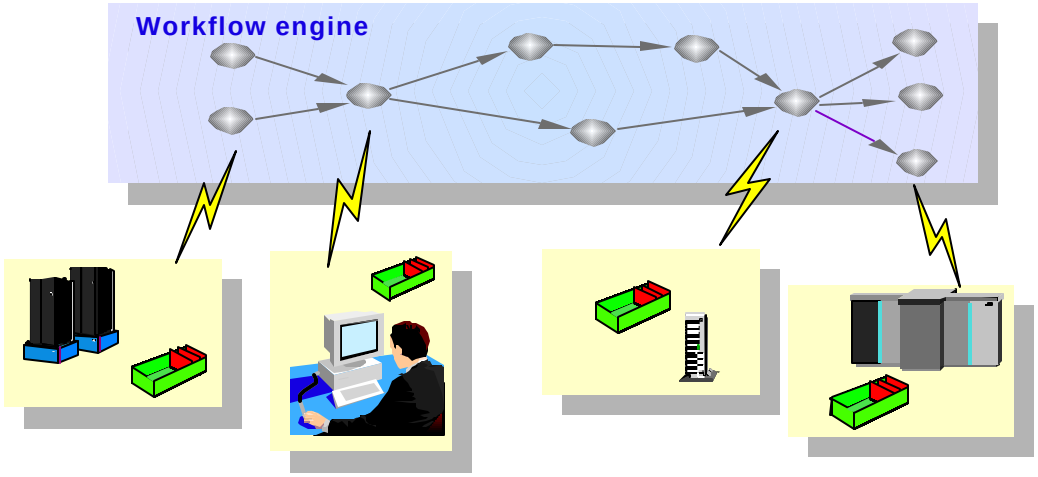
Connect your applications
(MQSeries Adapter Offering)



MQSeries Integrator

- Message brokering hub for processing, transformation and routing of messages (including publish/subscribe)





Business logic above the integrated applications

MQSeries Workflow

- Business process manager orchestrating and automating long-running, potentially interactive business processes

Flow Model characteristics of ...

MQWF

- *Process*
  - Sequence of steps that define a business process
  - Graph consisting of *activities* (program/process/block) and *control/data connectors*
  - Rules specified via start/exit/transition conditions
  - Resource assignment (People, IT) per activity

MQSI

- *Message Flow*
  - Sequence of operations on a message
  - Graph consisting of *message processing nodes* (built-in nodes, custom nodes) and *connectors*
  - Rules specified via special nodes (e.g. filter node)

Today: Separate tooling (MQWF Buildtime, MQSI Control Center)

Tomorrow: Common tooling based on FCM/Eclipse

Runtime characteristics...

Execution model of flow instances (process instance/message flow)

Process Instance

- Each navigation step is a single transaction.
- Multiple threads of execution
- Persistent state in database
- Process monitoring based on instance data and / or audit trail records
- Joins

Message Flow

- Entire flow is a single (transactional) unit of work.
- Single thread of execution
- Transient state within thread of execution, no state maintained across multiple inputs.
- "All-or-nothing" semantics
- Flow instance not observable while running, historic data about execution may be written.

Runtime characteristics... (cont.)

Execution model of activity / node implementations

Process Instance

- Invocation of activity implementations
  - distributed
  - heterogeneous
  - parallel

  (by PEA, PES or UPES)

Message Flow

- Invocation of processing nodes within same address space as message flow
  - local
  - homogeneous
  - serial

Summary

	MQSeries Workflow	MQSeries Integrator
General:	Process Management	Message Broker Hub
Flow Types:	Business Process Flows	Information Flows
Flow Scope:	Across units of work	Within a unit of work
Flow States:	Process State	Stateless
Message Formats:	XML (& internal)	XML & others
People Integration:	Yes	No
Application Integration:	MQSeries message PEA/PES/UPES	MQSeries message build-in nodes

Agenda

- Introduction
  - BPM
  - MQWF and MQSI characteristics
- Interoperability between MQWF and MQSI
  - MQWF's XML message interface
  - Call-In Sample Scenario
  - Call-Out Sample Scenario
  - Event Sample Scenario
  - Support Pacs
- Sample Scenario

Interoperability of MQWF and MQSI via XML messages

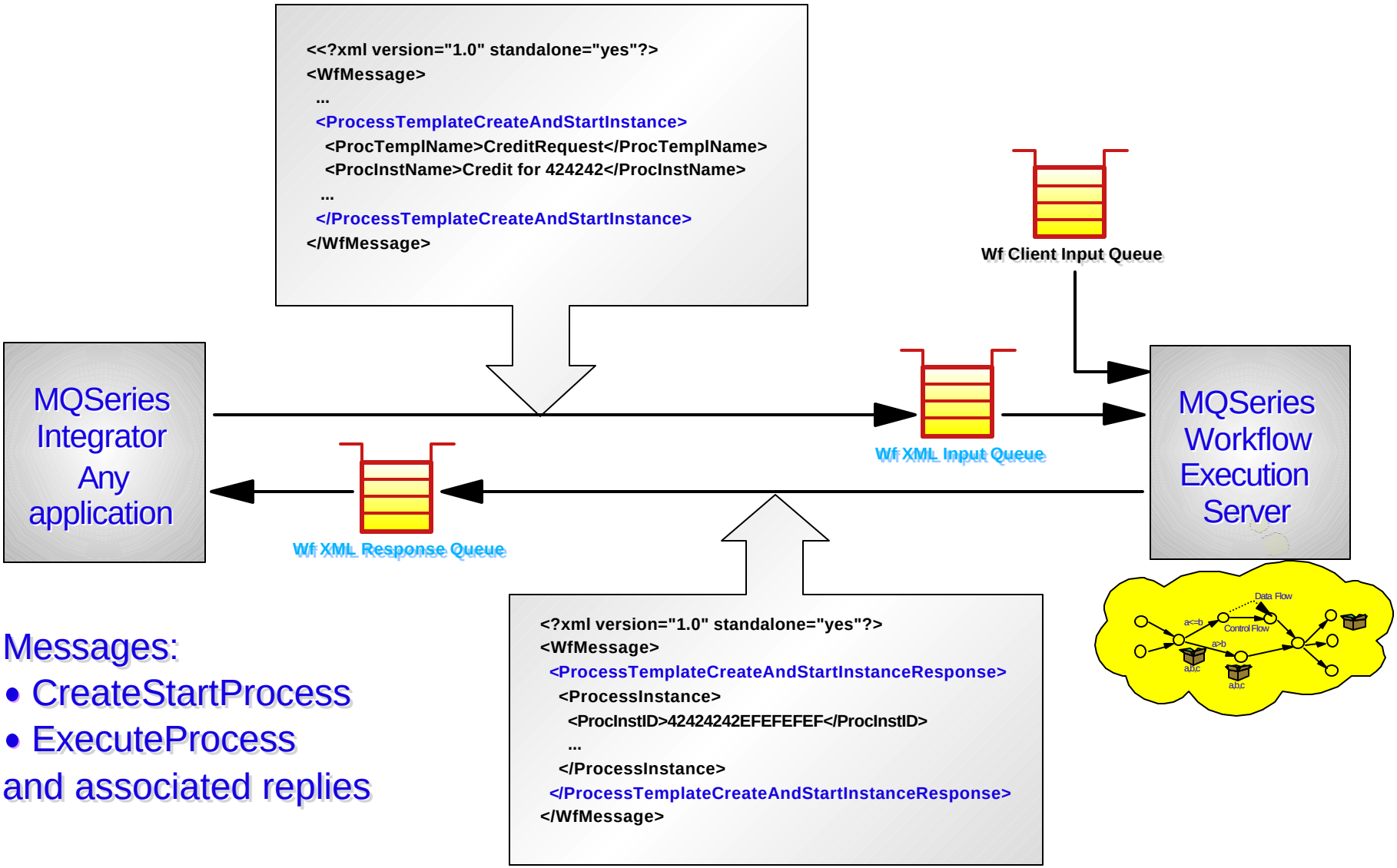
Integration of MQSeries Workflow and MQSeries Integrator by means of XML messages.

The XML message interface of MQWF V3 allows

- to start a process instance by means of an XML message
- to implement a program activity by sending an XML message to a user defined program execution server (UPES)

MQSI V2 provides XML message support to parse and create generic XML messages.

"Call-In" -- request as XML message



Messages:

- CreateStartProcess
- ExecuteProcess

and associated replies

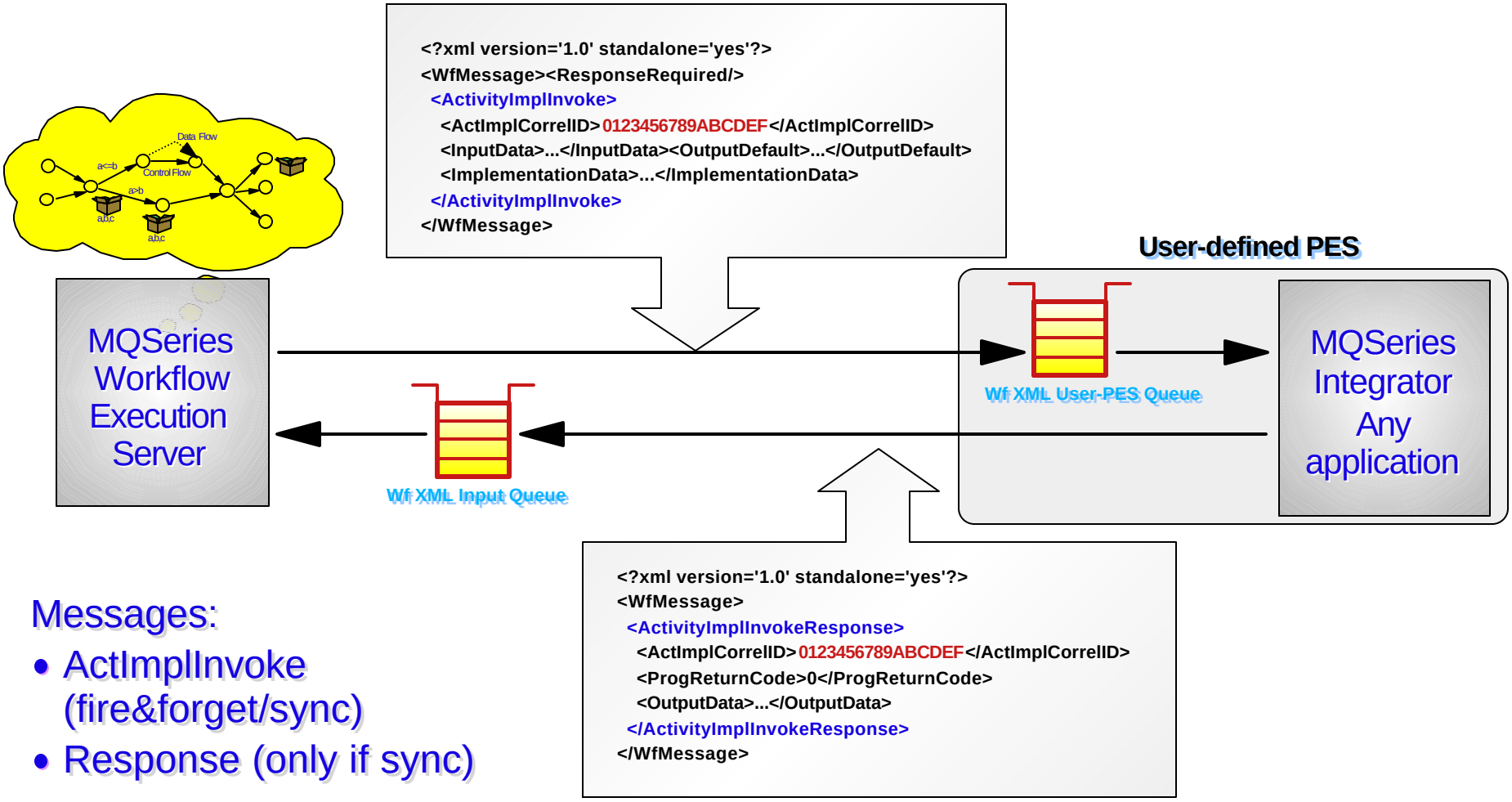
Notes: "Call-In" -- request as XML message

Provide additional interface into Workflow server

Message-based interface

- Enables transactional requests
- Truely asynchronous, sessionless
- Reply addressing via MQ header
- Request/reply correlation via optional user context data

"Call-Out" -- XML message as activity implementation



Messages:

- ActImplInvoke (fire&forget/sync)
- Response (only if sync)

Notes: "Call-Out" -- XML message as activity implementation

Provide additional callback mechanism for activity triggering from Workflow server

- Facilitates transactional activity implementation (save application)
- Synchronous invocation
  - reply required to signal completion
- "Fire and forget" invocation
  - WF server continues process navigation right away

User-defined program execution server (UPES)

- modeled during Buildtime as an abstraction of a MQ queue
- implemented by an application that processes activity implementation invocation messages and generates appropriate responses

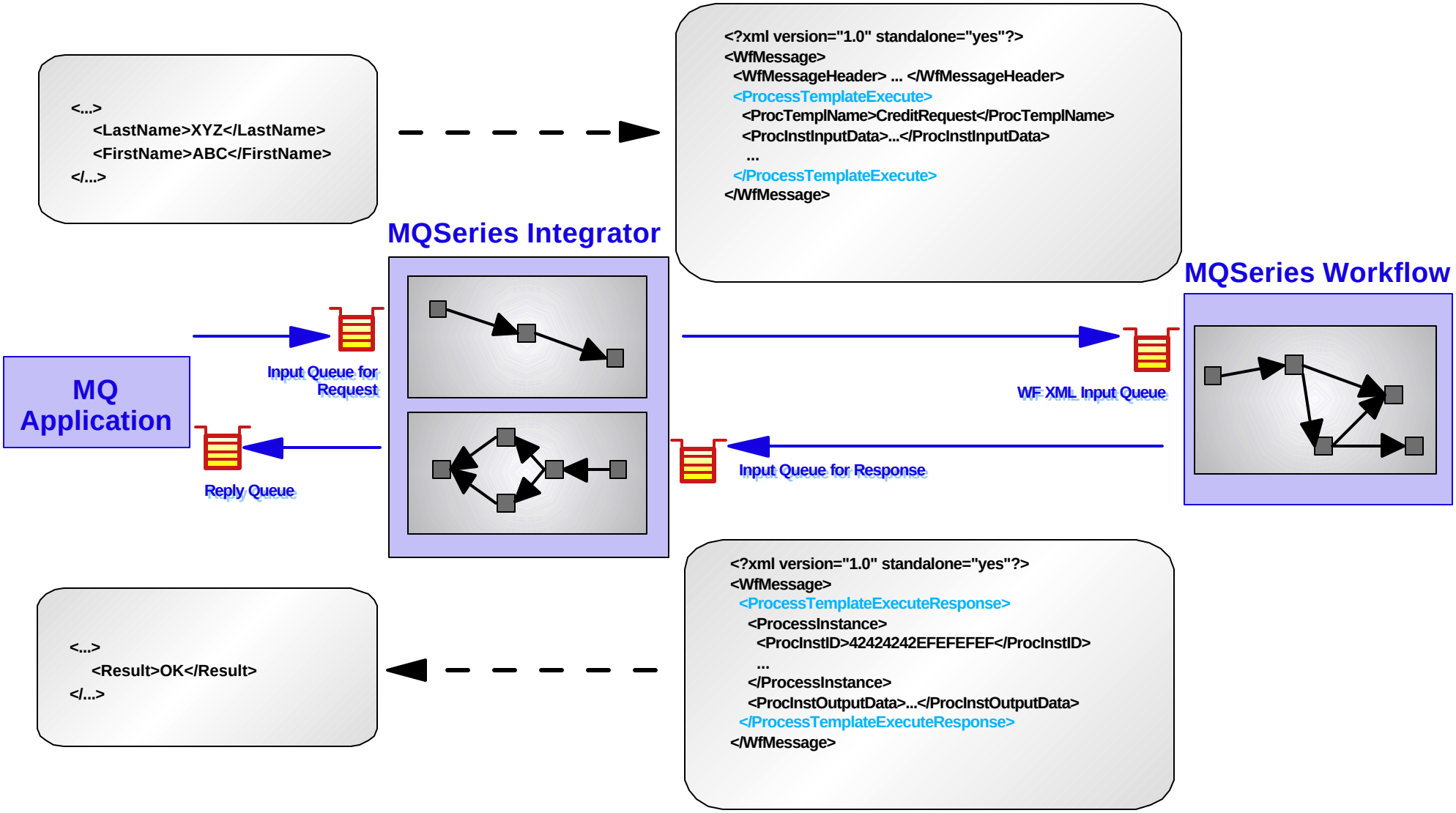
Summary of MQWF's XML support

- **Direct interface with MQSeries applications**
  - Start/Execute process instance
  - Activity implementation invocation (request&reply, fire&forget)
- **Interoperability with MQSeries Integrator**
- **Integrate applications on all MQSeries platforms**
  - Exploitation of MQSeries adapters
- **Transactional application integration via MQSeries**
  - Chaining of workflow server and application transactions
  - Assured message delivery
- **Truly asynchronous, sessionless interface**
  - Request/reply correlation via optional user context data
- **XML as standard language for message content**
  - Attractive to industry solution developers and emerging application standards
  - Supported in new web browsers, and web-based AD tools

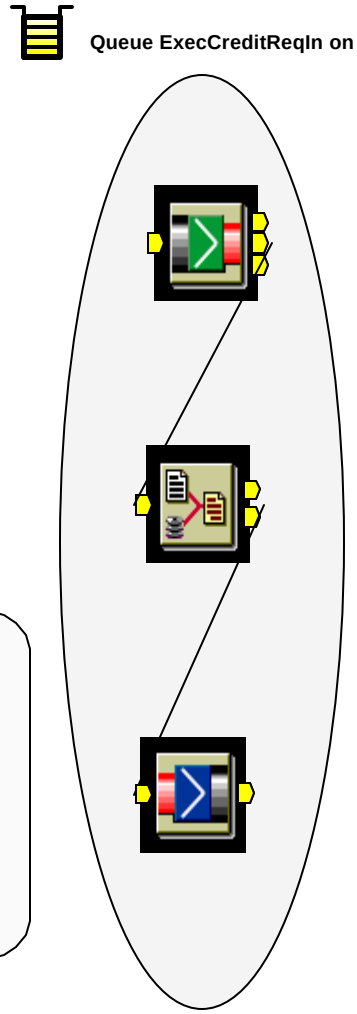
Agenda

- Introduction
  - BPM
  - MQWF and MQSI characteristics
- Interoperability between MQWF and MQSI
  - MQWF's XML message interface
  - Call-In Sample Scenario
  - Call-Out Sample Scenario
  - Event Sample Scenario
  - Support Pacs
- Sample Scenario

Call-In Scenario



Message Flow ExecCreditReq




```
<ExecCreditReq.>
    <LName>Potter</LName>
    <FName>Harry</FName>
</ExecCreditReq>
```

MQInput Node
reads message from input queue of message flow

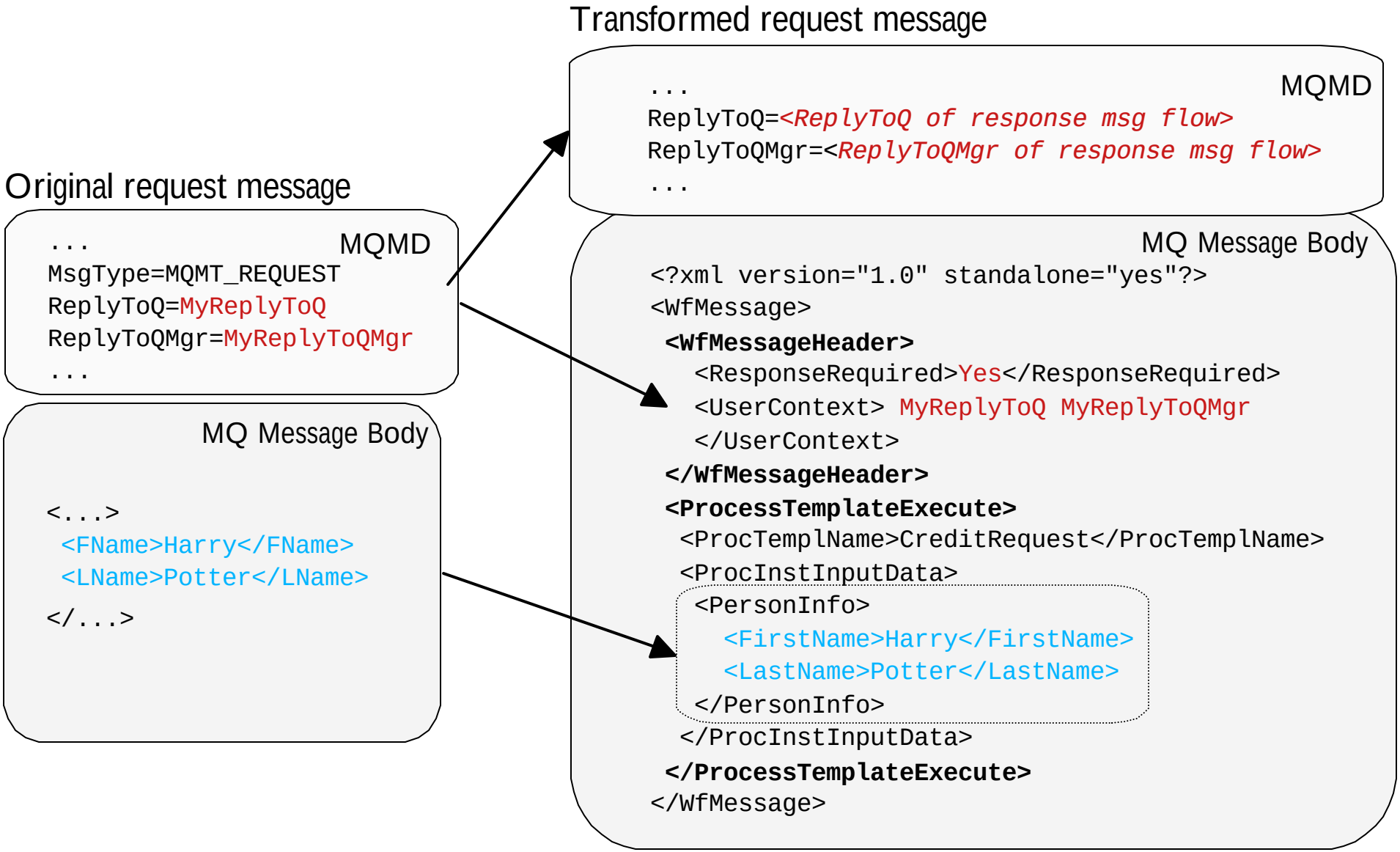
Compute Node
transforms input message into XML message ProcessTemplateExecute

MQOutput Node
writes message to XML input queue of MQS-WF

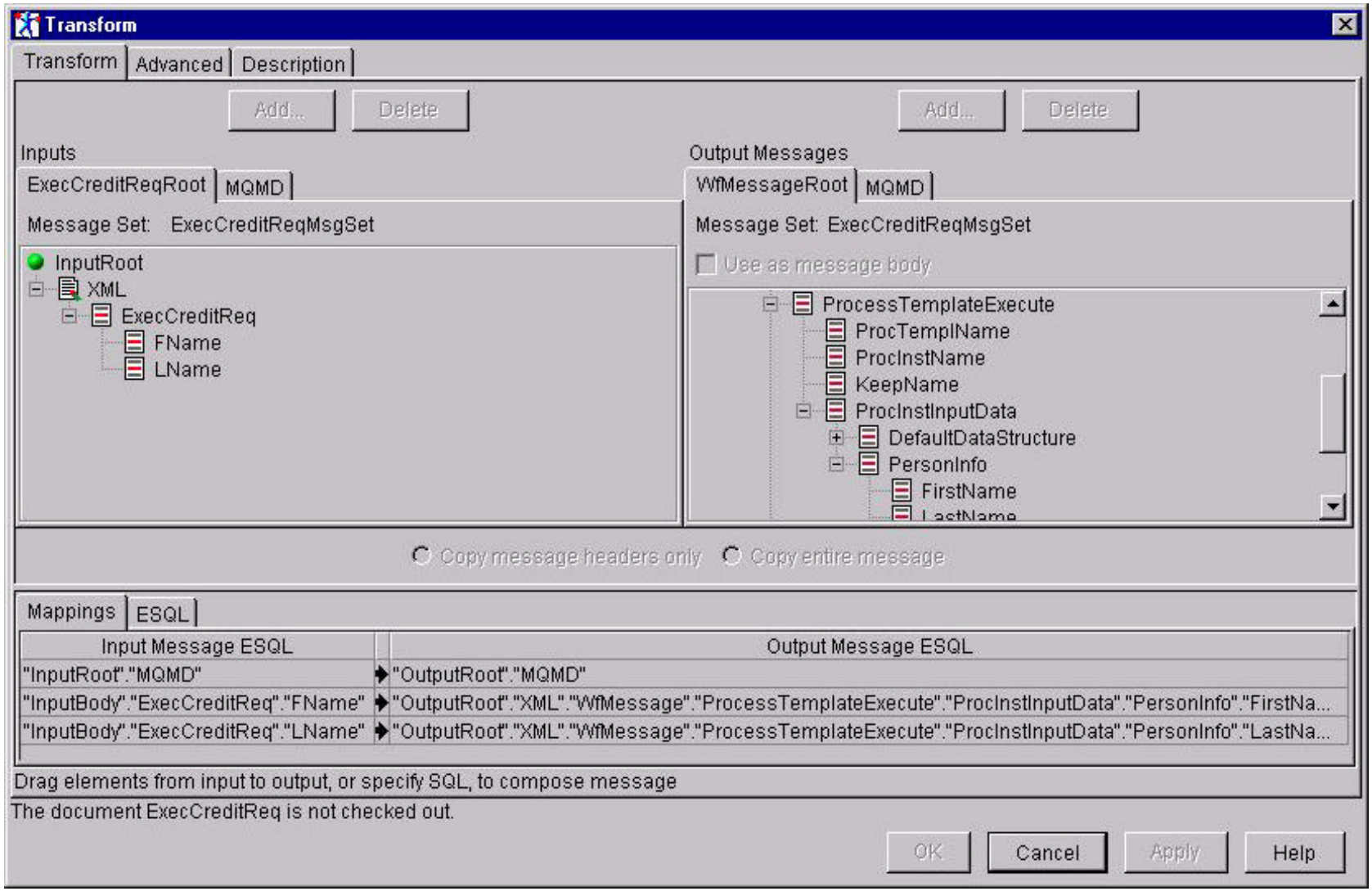
XML message to trigger execution of process CreditRequest with input container

```
<PersonInfo>
  <LastName>Potter</LastName>
  <FirstName>Harry</FirstName>
</PersonInfo>
```

Sample Request Messages



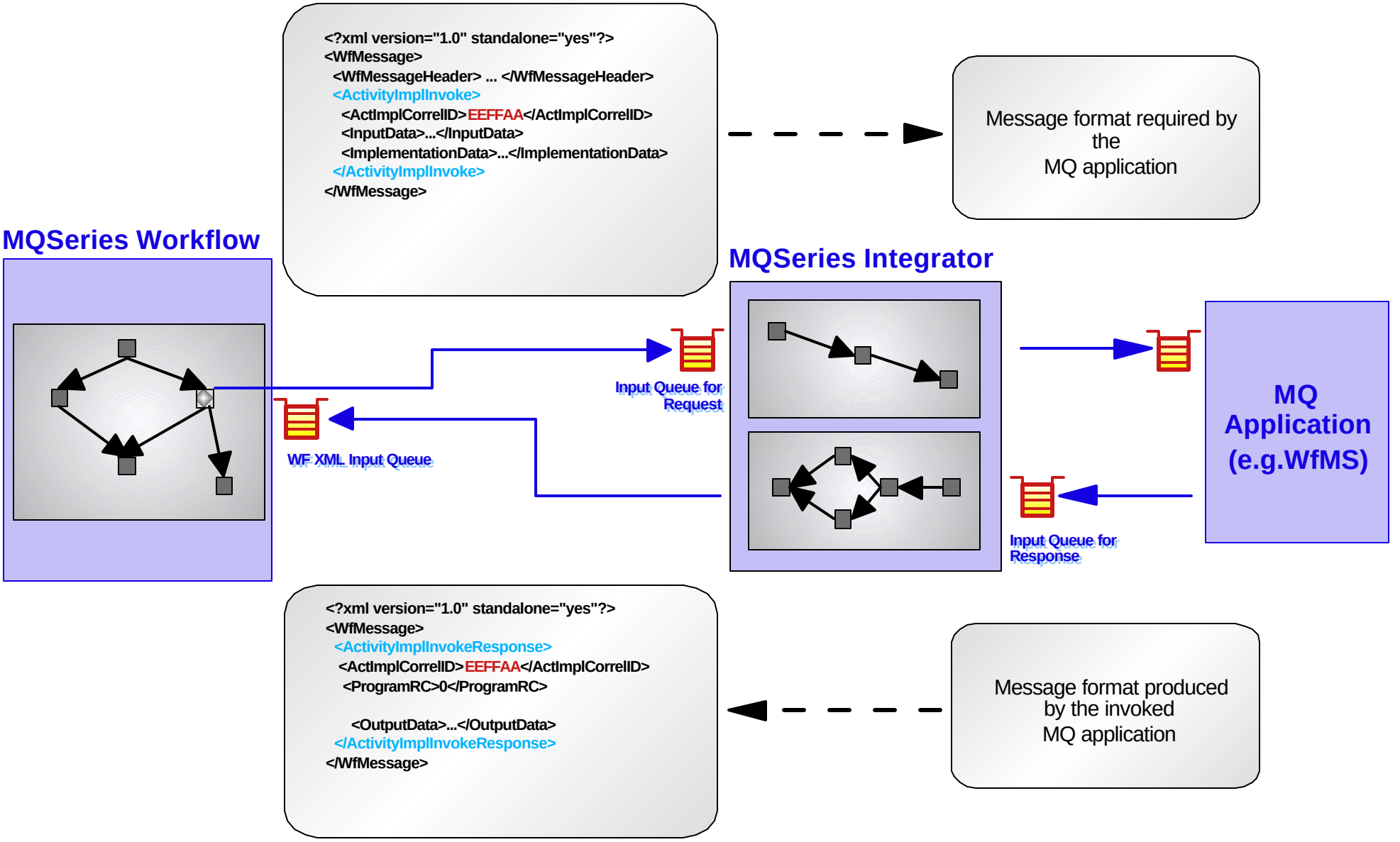
Compute Node Properties



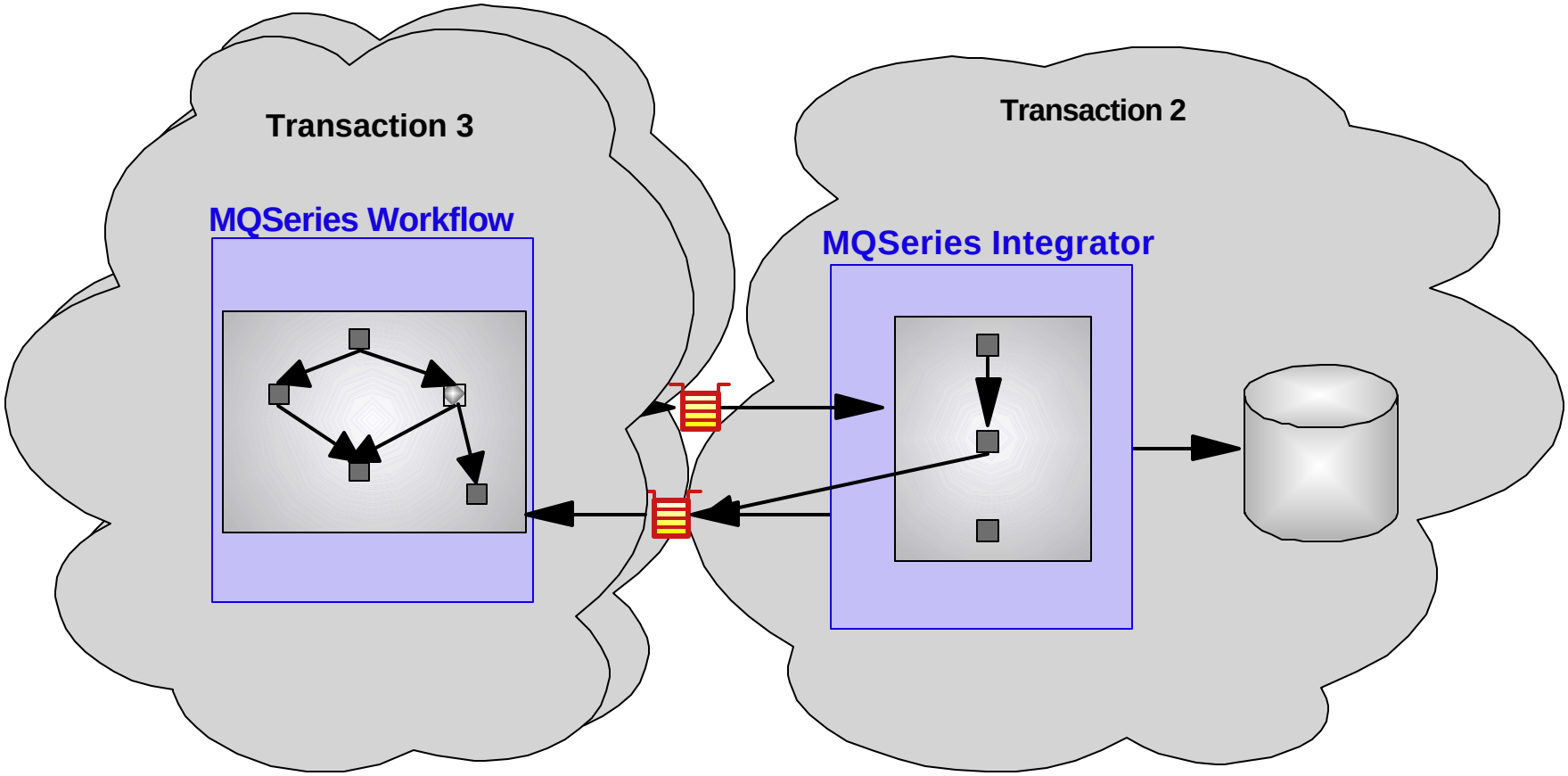
Message Transformation by Compute Node (ESQL)

```
SET OutputRoot.MQMD = InputRoot.MQMD;

SET OutputRoot.MQMD.ReplyToQ='ExecCreditReqResponseIn'; -- ReplyToQ of response msg flow
SET OutputRoot.MQMD.ReplyToQMgr='FMCQM';                -- ReplyToQMgr of response msg flow

SET OutputRoot.XML.(XML.XmlDecl)='';
SET OutputRoot.XML.(XML.XmlDecl).(XML.Version)='1.0';
SET OutputRoot.XML.(XML.XmlDecl).(XML.Standalone)='yes';

IF ( InputRoot.MQMD.MsgType = MQMT_REQUEST ) THEN
  SET OutputRoot.XML.WfMessage.WfMessageHeader.ResponseRequired='Yes';
  SET OutputRoot.XML.WfMessage.WfMessageHeader.UserContext=
  TRIM(InputRoot.MQMD.ReplyToQ) || ' ' || TRIM(InputRoot.MQMD.ReplyToQMgr);
ELSE
  SET OutputRoot.XML.WfMessage.WfMessageHeader.ResponseRequired='No';
END IF;

SET OutputRoot.XML.WfMessage.ProcessTemplateExecute.ProcTemplName='CreditRequest';
SET OutputRoot.XML.WfMessage.ProcessTemplateExecute.ProcInstInputData.PersonInfo.LastName=
InputBody.ExecCreditReq.LName;
SET OutputRoot.XML.WfMessage.ProcessTemplateExecute.ProcInstInputData.PersonInfo.FirstName=
InputBody.ExecCreditReq.FName;
```

Call-Out Scenario



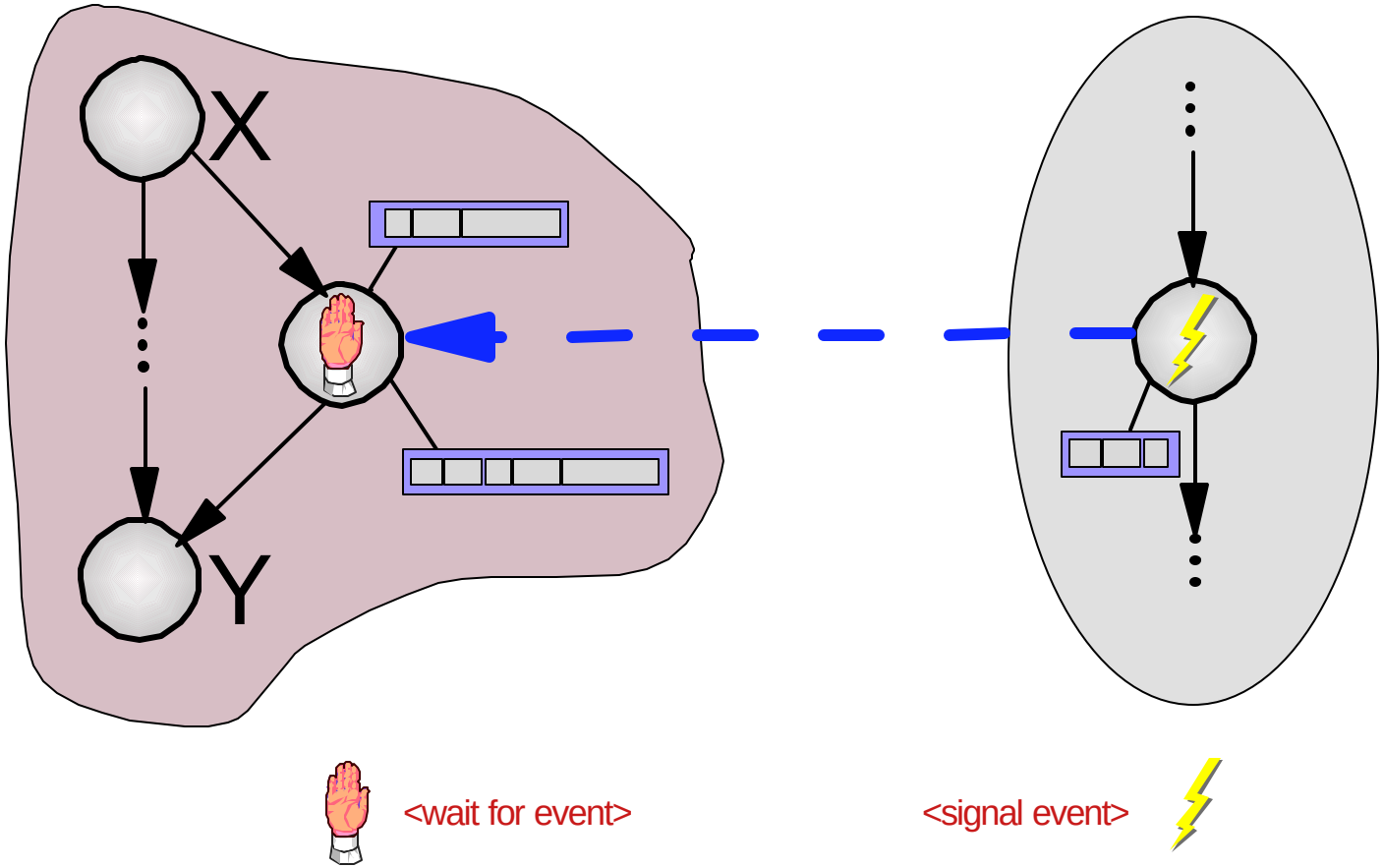
Transactional Activity Implementation



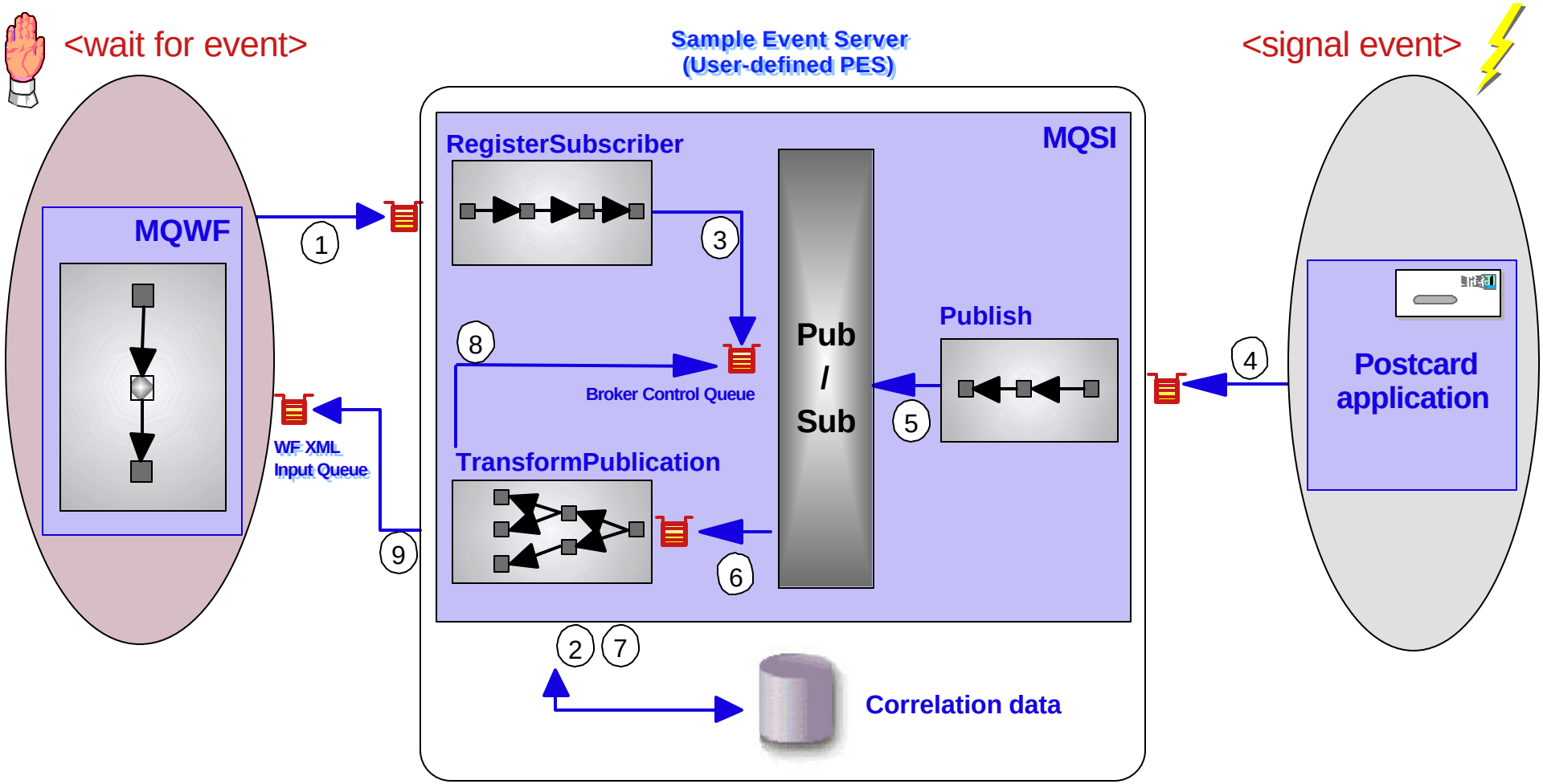
Agenda

- Introduction
  - BPM
  - MQWF and MQSI characteristics
- Interoperability between MQWF and MQSI
  - MQWF's XML message interface
  - Call-In Sample Scenario
  - Call-Out Sample Scenario
  - Event Sample Scenario
  - Support Pacs
- Sample Scenario

MQSeries Workflow - Events



Sample Event Scenario



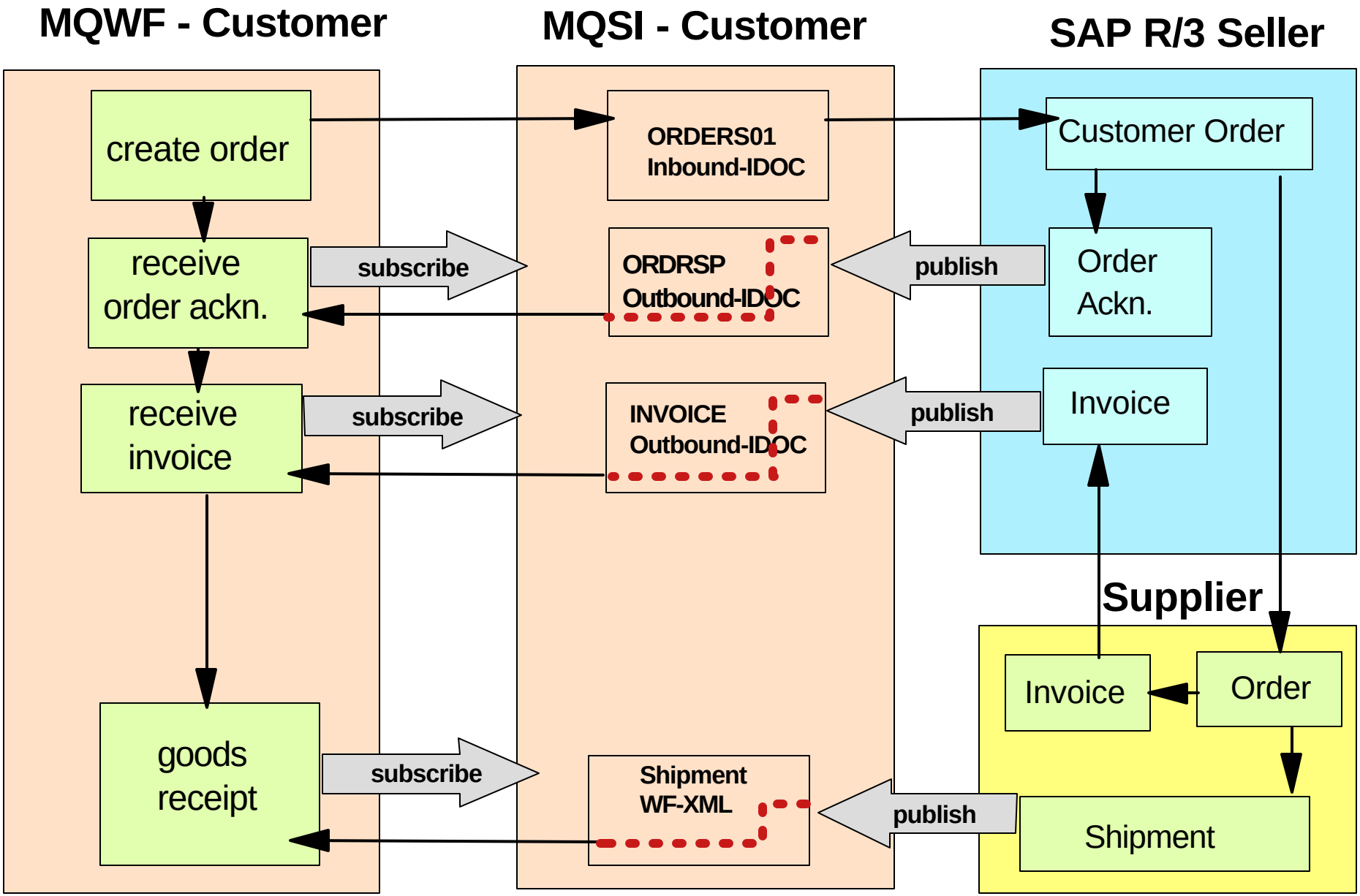
Sample Support Pacs

- **MQSeries Workflow - MQSeries Integrator interoperability sample scenario (WA02)**
- **MQSeries Workflow - Event Server sample using MQSeries Integrator (WA06)**
- **MQSeries Workflow - Web Credit Example (WA82)**
- **availabe at:**
  - **http://www-4.ibm.com/software/ts/mqseries/txppacs**

Agenda

- Introduction
  - BPM
  - MQWF and MQSI characteristics
- Interoperability between MQWF and MQSI
  - MQWF's XML message interface
  - Call-In Sample Scenario
  - Call-Out Sample Scenario
  - Event Sample Scenario
  - Support Pacs
- Sample Scenario

MQWF/MQSI - SAP R/3 Integration



Thank You!

Visit us at the MQSeries Workflow homepage
http://www.software.ibm.com/ts/mqseries/workflow

IBM