WebSphere. software

IBM

IBM Software Group

University of Toronto

Business Process Management & SOA

Soudabeh Javadi
Consulting Technical Sales Support
WebSphere – Process Integration
IBM Software Group

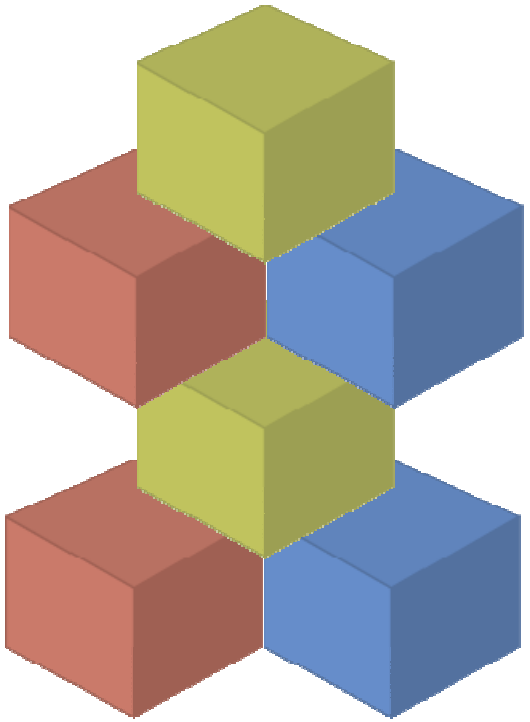
ON DEMAND BUSINESS

Agenda

- **SOA & BPM Introduction**
- **Model**
- **Assemble & Deploy**
- **Manage**

SOA: Service Oriented Architecture

- **An approach for building distributed systems that allows tight correlation between the business model and the IT implementation.**

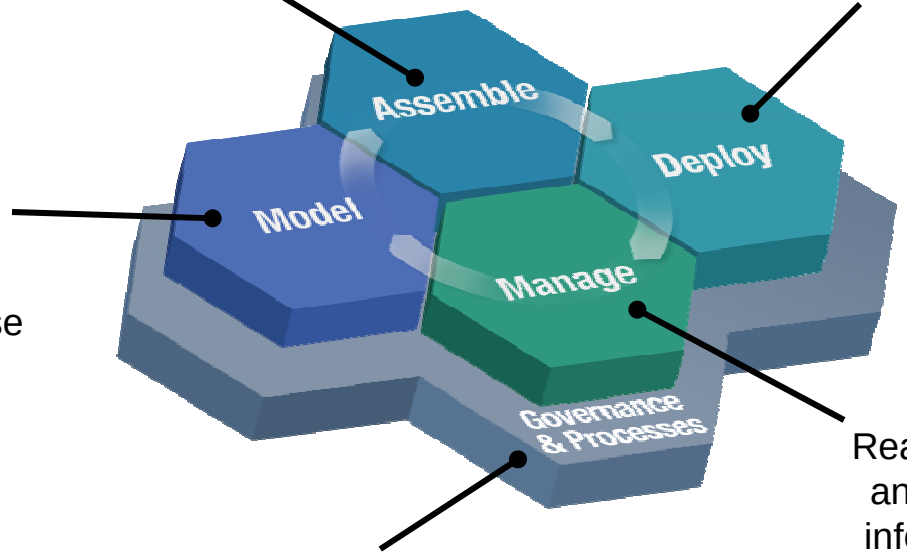


- **Characteristics:**
  - Represents business function as a service
  - Shifts focus to application assembly rather than implementation details
  - Allows individual software assets to become building blocks that can be reused in developing composite applications representing business processes
  - Leverages open standards to represent software assets

The SOA Lifecycle .. For Flexible Business & IT

Assemble
Assemble existing and new assets to execute and manage business processes

Deploy
Deployment of models, policies and assemblies to realize business intent

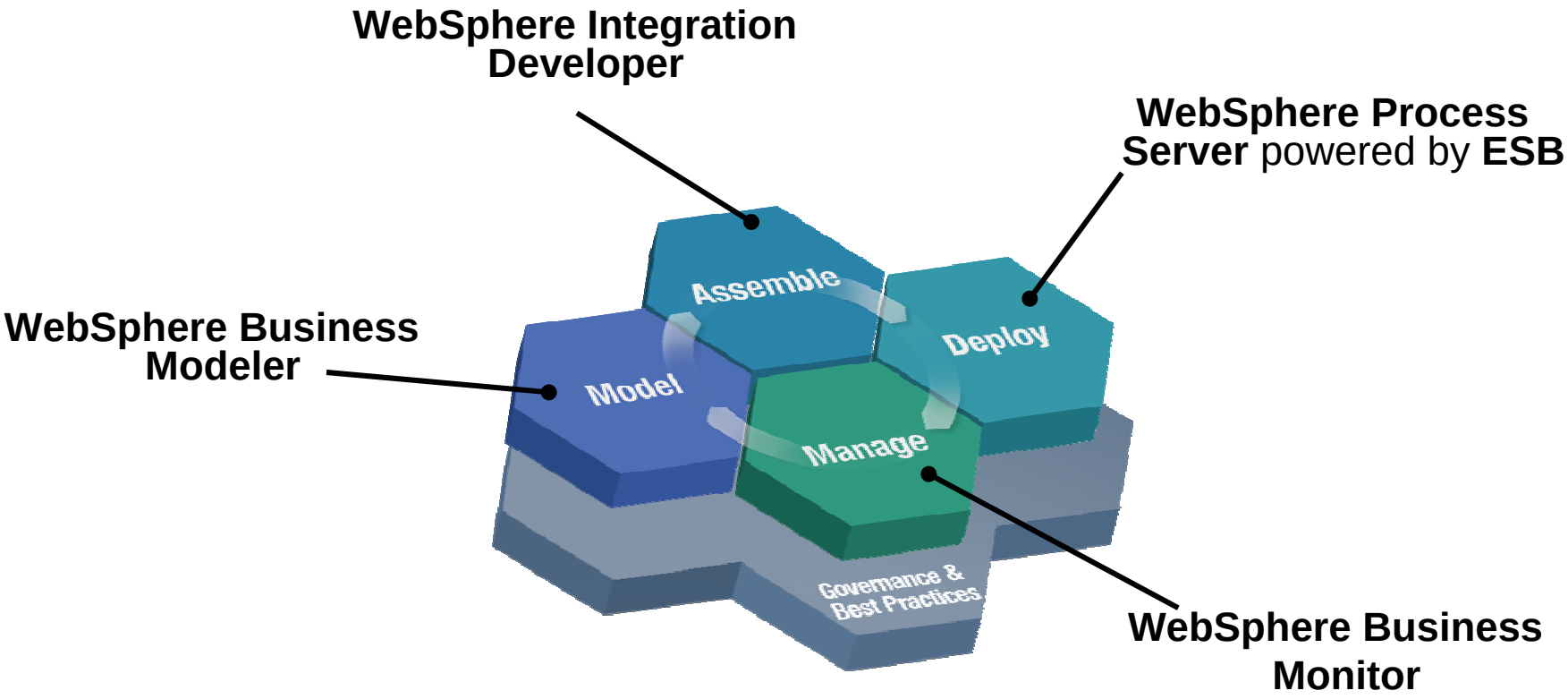


Model
Capture, simulate, analyze, and optimize business models to reduce risk and increase flexibility

Manage
Real-time visibility and analysis of business information for timely and coordinated action

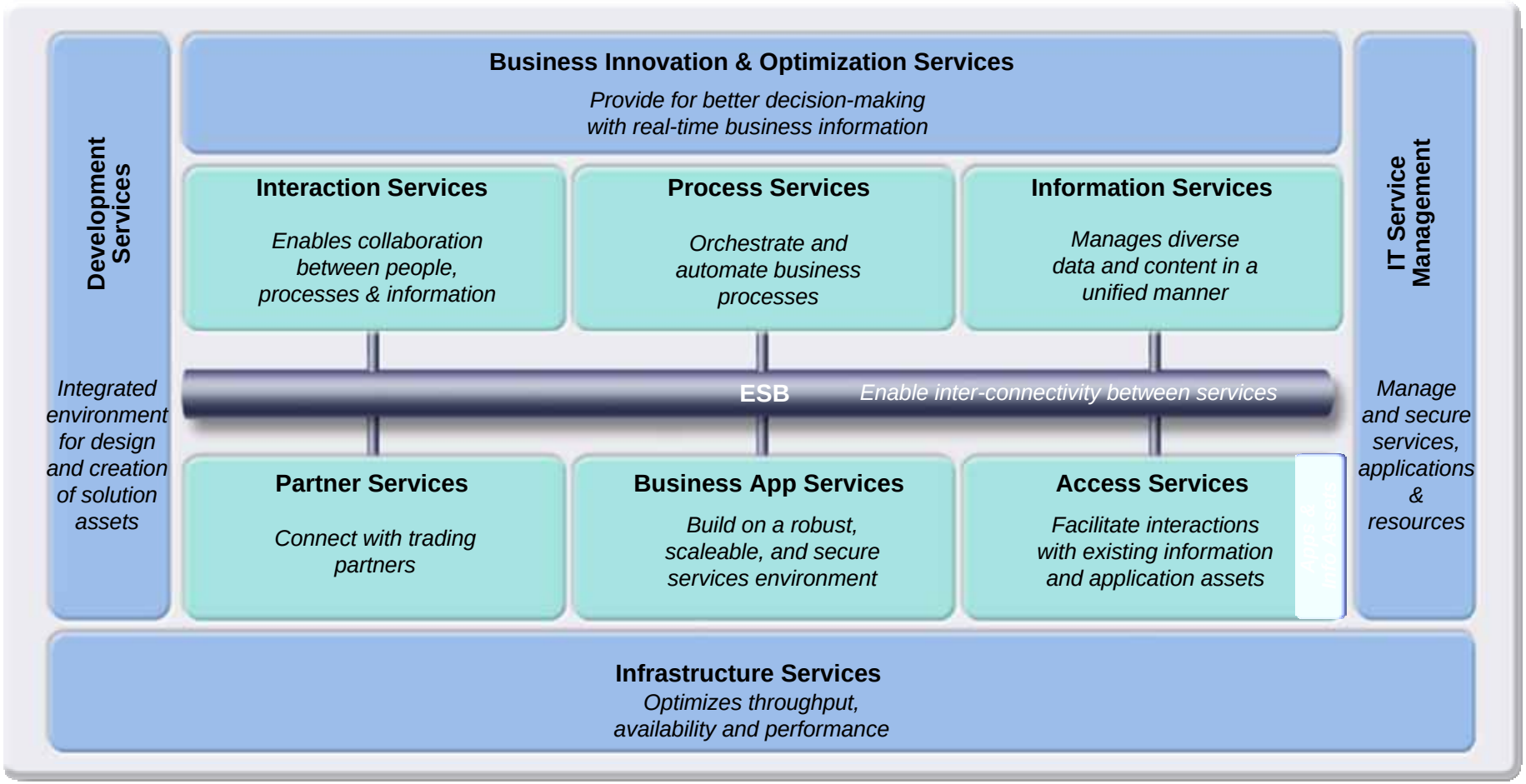
Governance & Processes
Alignment of strategy and operations across business and IT in support of business objectives

IBM WebSphere software supporting BPM & SOA



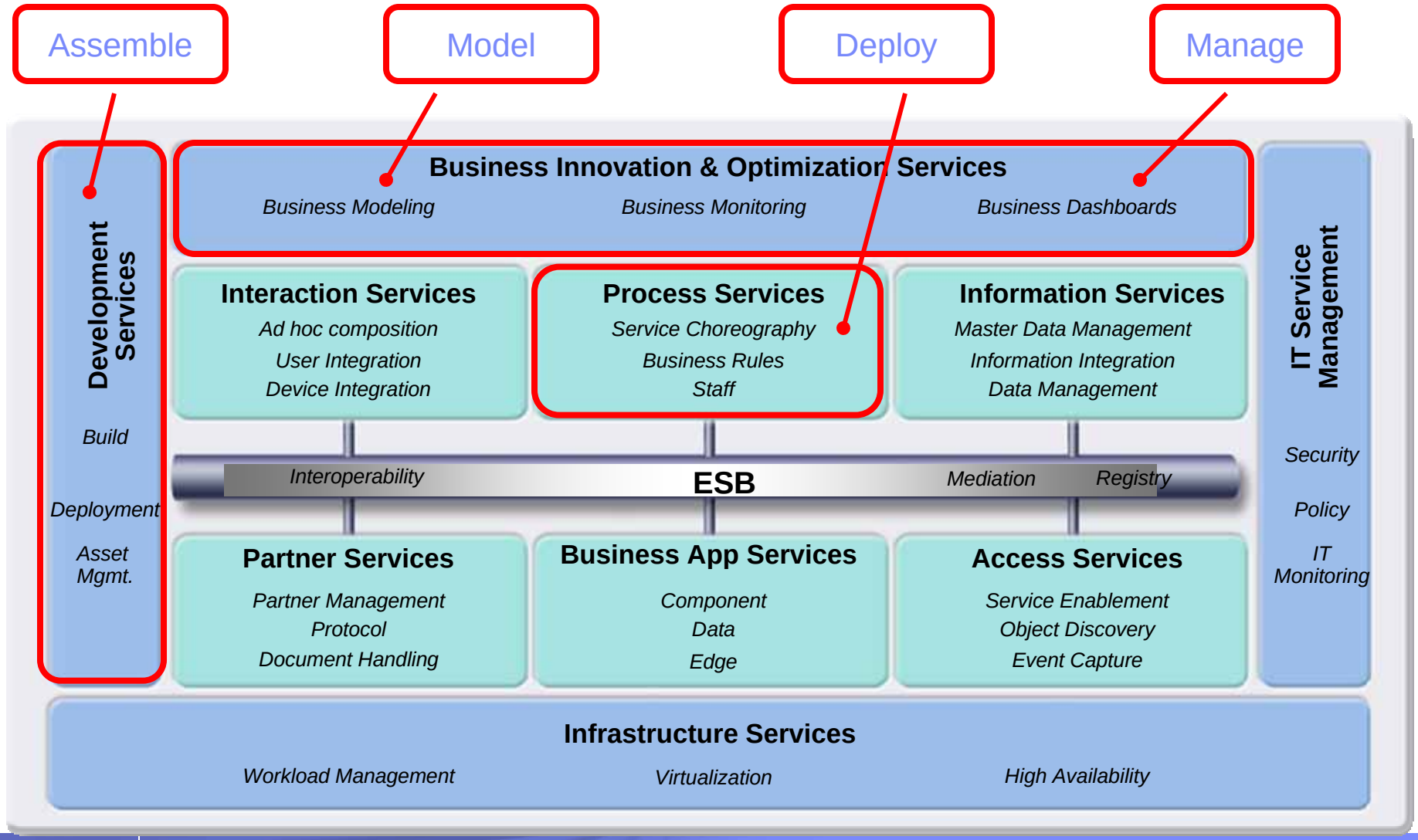
SOA Reference Architecture

Model of the Logical Architecture



SOA Reference Architecture

Comprehensive services in support of your SOA



Agenda

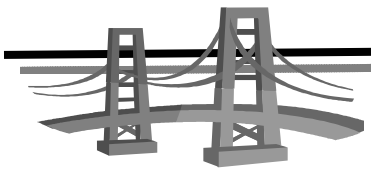
- **SOA & BPM Introduction**
- **Model**
- **Assemble & Deploy**
- **Manage**

Drivers for Business Understanding

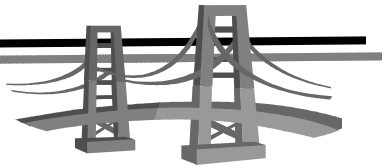
Business Modeling

Customers model processes for many purposes:

- Modeling For Compliance/Documentation
- Modeling For Redesign
- Modeling For Execution



Domain and Tooling Gap

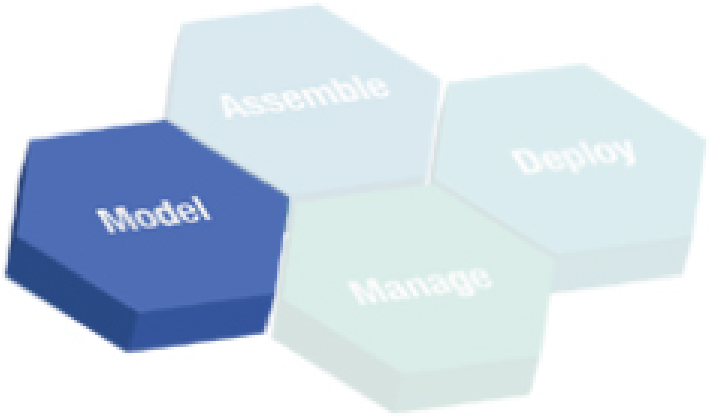


IT Development

- Application Development
- Service Implementations
- Process Choreography and Human Workflow

Model

- **Graphically Model Processes**
  - Simple but Comprehensive Modeling
  - A business tool for business users
  - Model everything you need to design and "sand-box" your business process – Costs, Times, Resources
- **Simulate And Analyze**
  - Simulated execution of the business process with detailed statistical analysis tools
- **Collaborate and Web Publish**
  - Tools to allow multiple people to work as a team on business process work
  - Tools to publish business process work across the business
- **Hand Off To IT**
  - Export business and data models for use in IT deployment
- **Business Perfomance Modeling**
  - Define Key Performance Indicators and metrics



IBM WebSphere Business Modeler

WB Modeler v6 Advanced – at a glance (1)

- Business Process Modeling
  - Easy to use designed and tested with business analysts
  - Multiple user profiles & technology modes
  - Detailed resource, process and data modeling
  - Swimlane Editors
- Robust Analysis
  - Static and Dynamic analysis
  - Powerful simulation engine
- Extended Reporting Capabilities
  - Standard reports based on templates
  - User defined reports (Report Designer)
  - Integrated Crystal Reports

WB Modeler v6 Advanced – at a glance (2)

- Multi-User Support
  - CVS and ClearCase
- Collaboration Support
  - Web-publishing server
- Business Measures Editor (Metrics, KPIs, Aggregations, etc.)
  - Export Business Measures model to WebSphere Business Monitor
  - Import WebSphere Business Monitor Run-Time Metrics
- Integration Support
  - Eclipse based & integration with Rational XDE & RSA (UML), WSAD – IE & WebSphere Integration Developer (BPEL), WMQ Workflow (FDL)
  - Visio, XML, XML schema (XSD), delimited text,…
  - IFW (Information Framework) & IAA (Insurance Application Architecture) – bi-directional bridge
  - Tivoli Business System Manager

Agenda

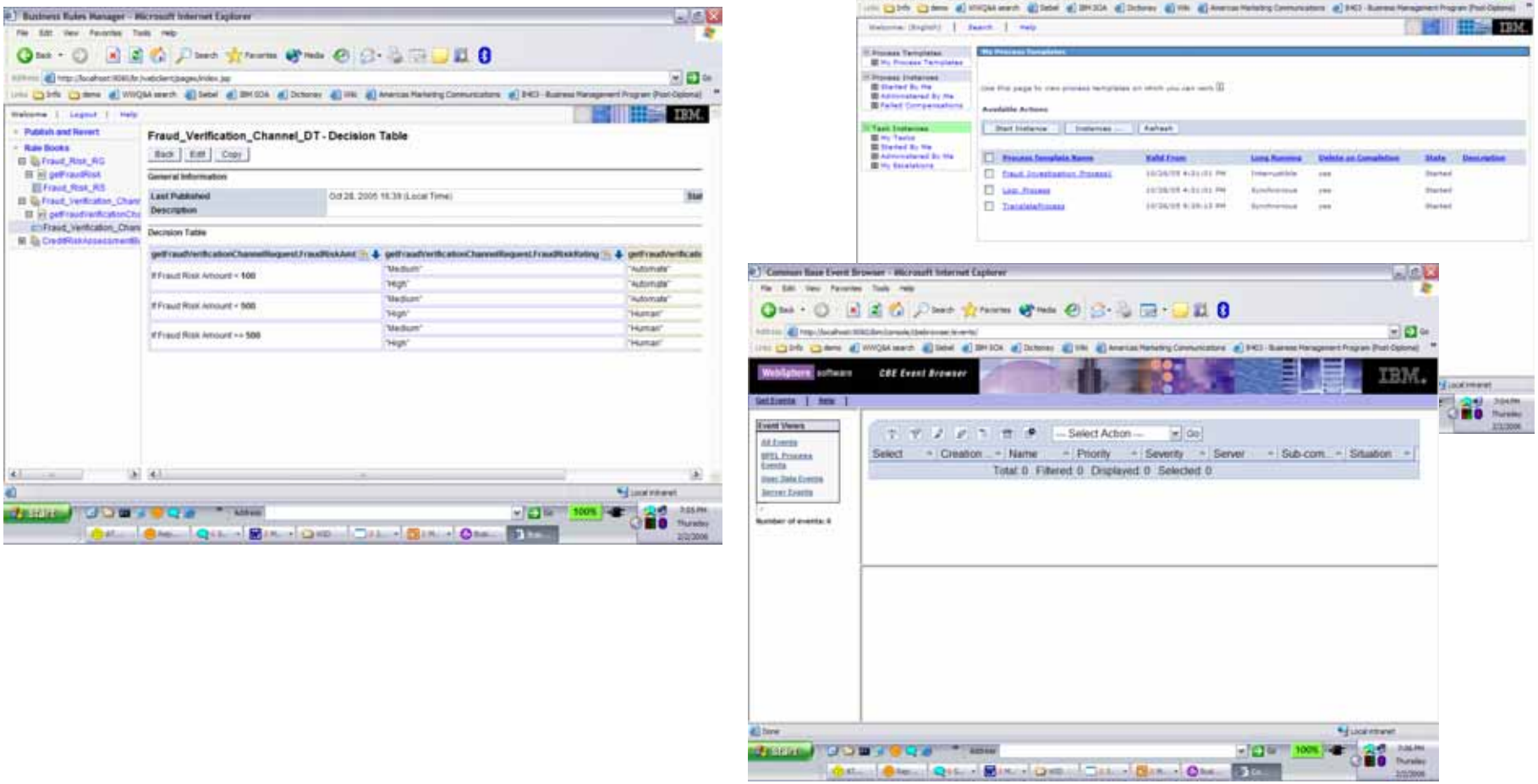
- **SOA & BPM Introduction**
- **Model**
- **Assemble & Deploy**
- **Manage**

Integration in an SOA World

SOA Integration involves

- Common Data Model
  - All Data is represented consistently
- Common Invocation Model
  - All components are represented consistently
  - All components are invoked identically
- Common Connectivity
  - Enterprise Service Bus
- Service Choreography
  - Components can be choreographed independently of their implementation

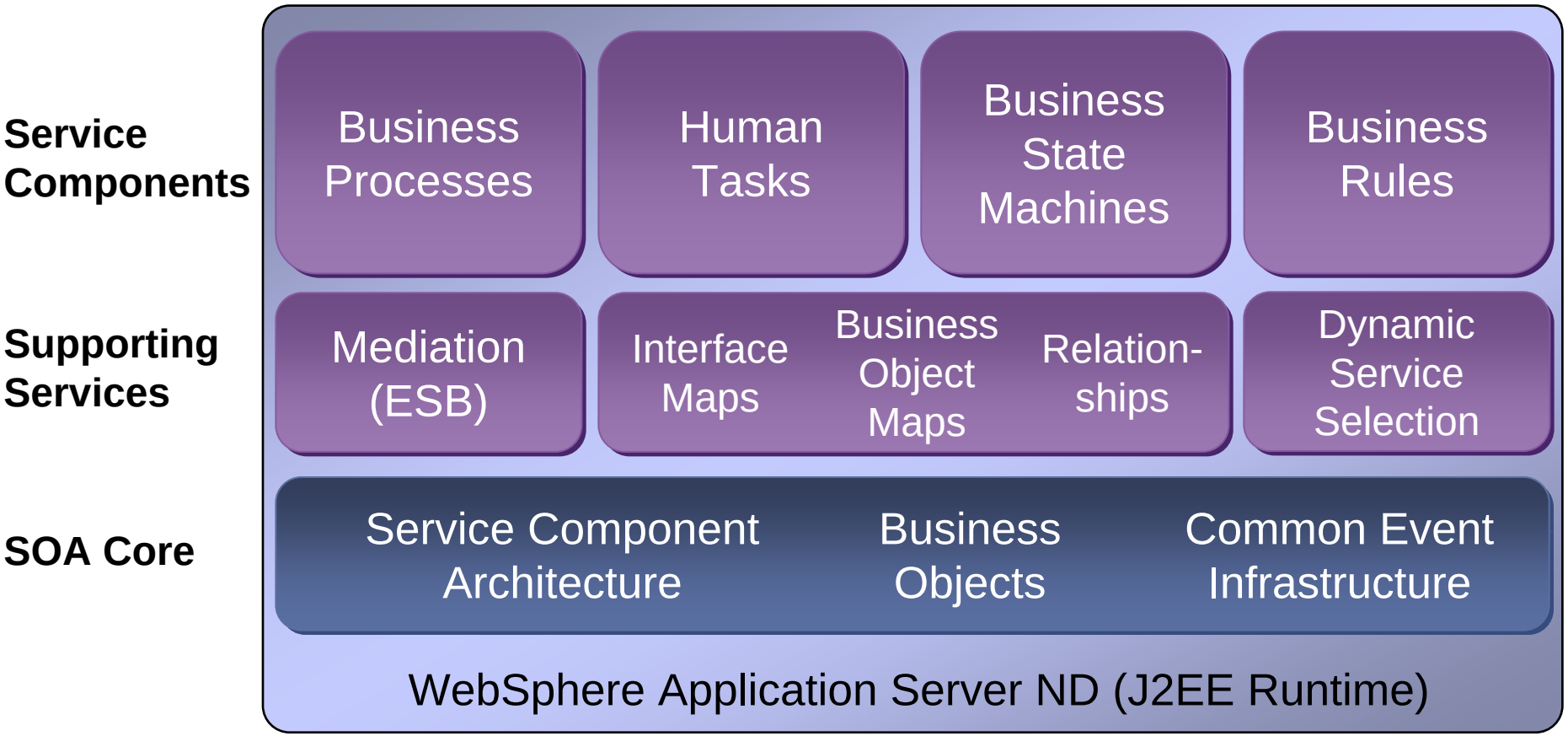
WebSphere Process Server v6



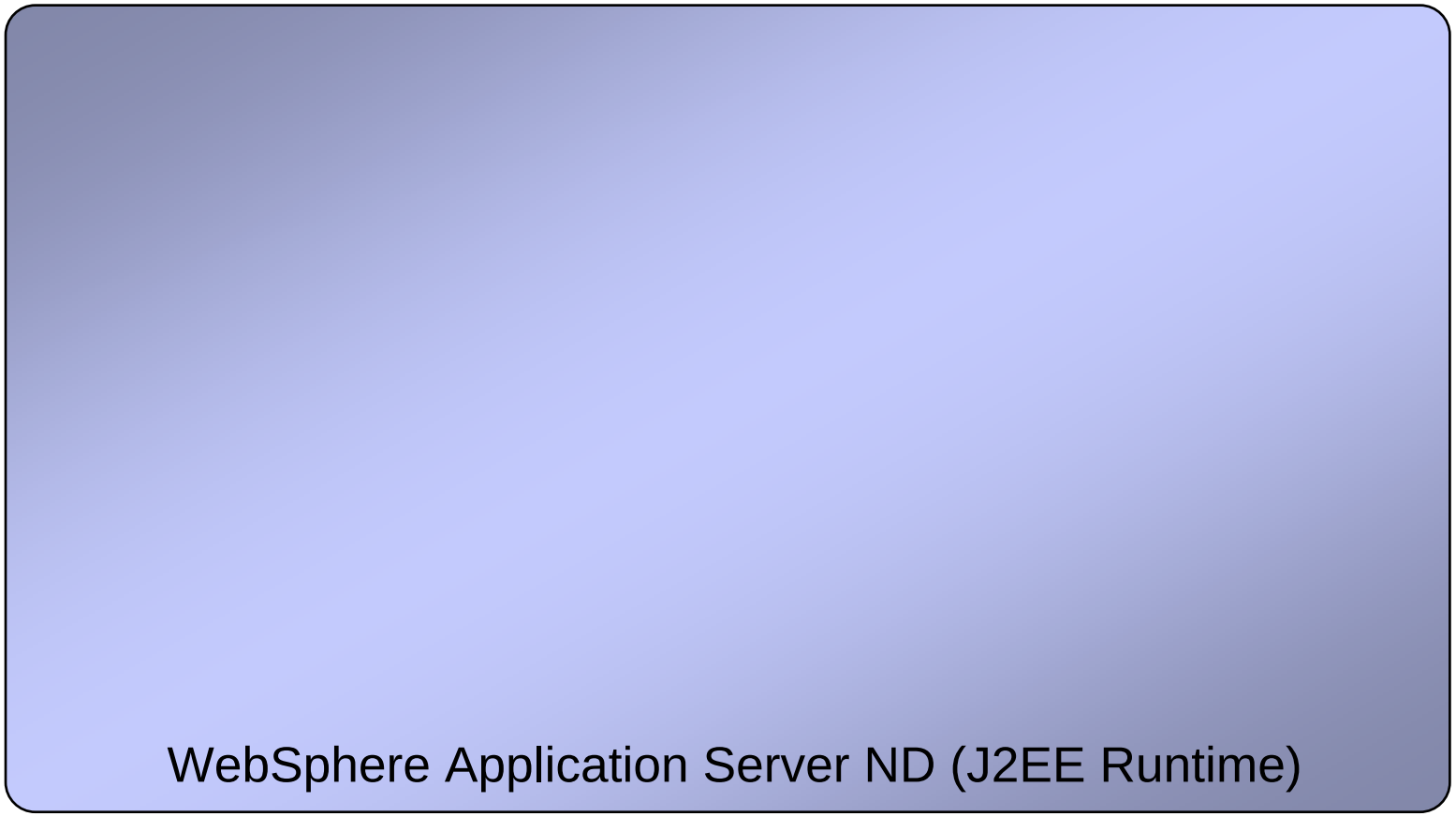
WebSphere Process Server v6 - Highlights

- WebSphere Application Server Foundation
  - Clustering, failover, high availability and robust platform
  - Single administration environment
  - Common Event Infrastructure – Process Management
- Service Oriented Architecture platform
  - A uniform invocation programming model (SCA)
  - A uniform data representation model (Business Objects)
  - Common connectivity (ESB)
  - Powerful tools to build and reuse standard components
- Powerful Staff Components
  - Participating / Originating / Ad-Hoc Tasks
  - Multi-level escalation
  - Client components out-of-the box (JSF)
- Business Processes
  - WS-BPEL standard
- Business State Machines, Business Rules & Transformations
  - Advanced services to build integration solutions
- A single Process Integration platform
  - Reduces complexity and administration cost

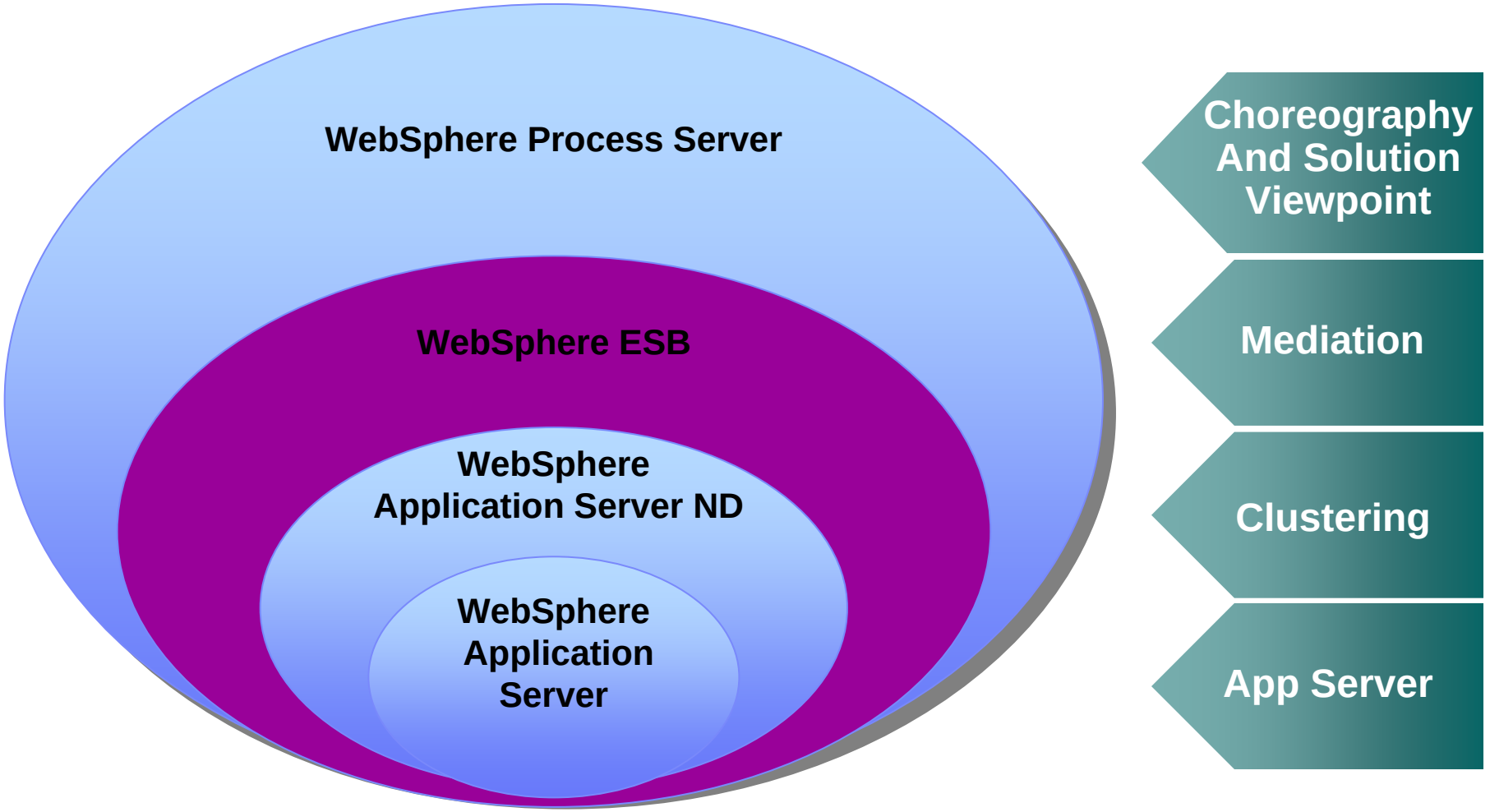
WebSphere Process Server v6 - Components



WebSphere Process Server v6 - Components

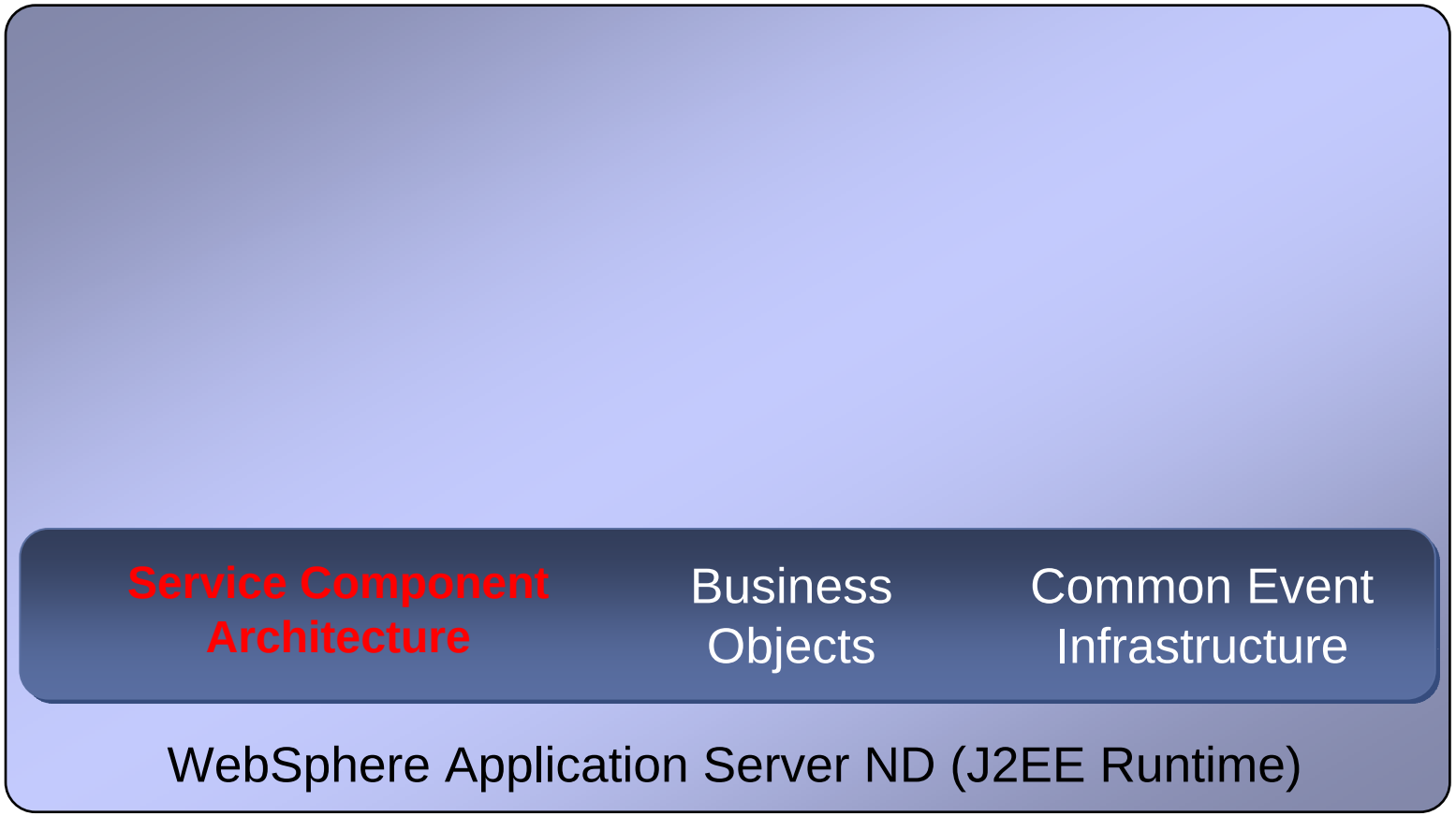


WebSphere Application Server, ESB, and Process Server

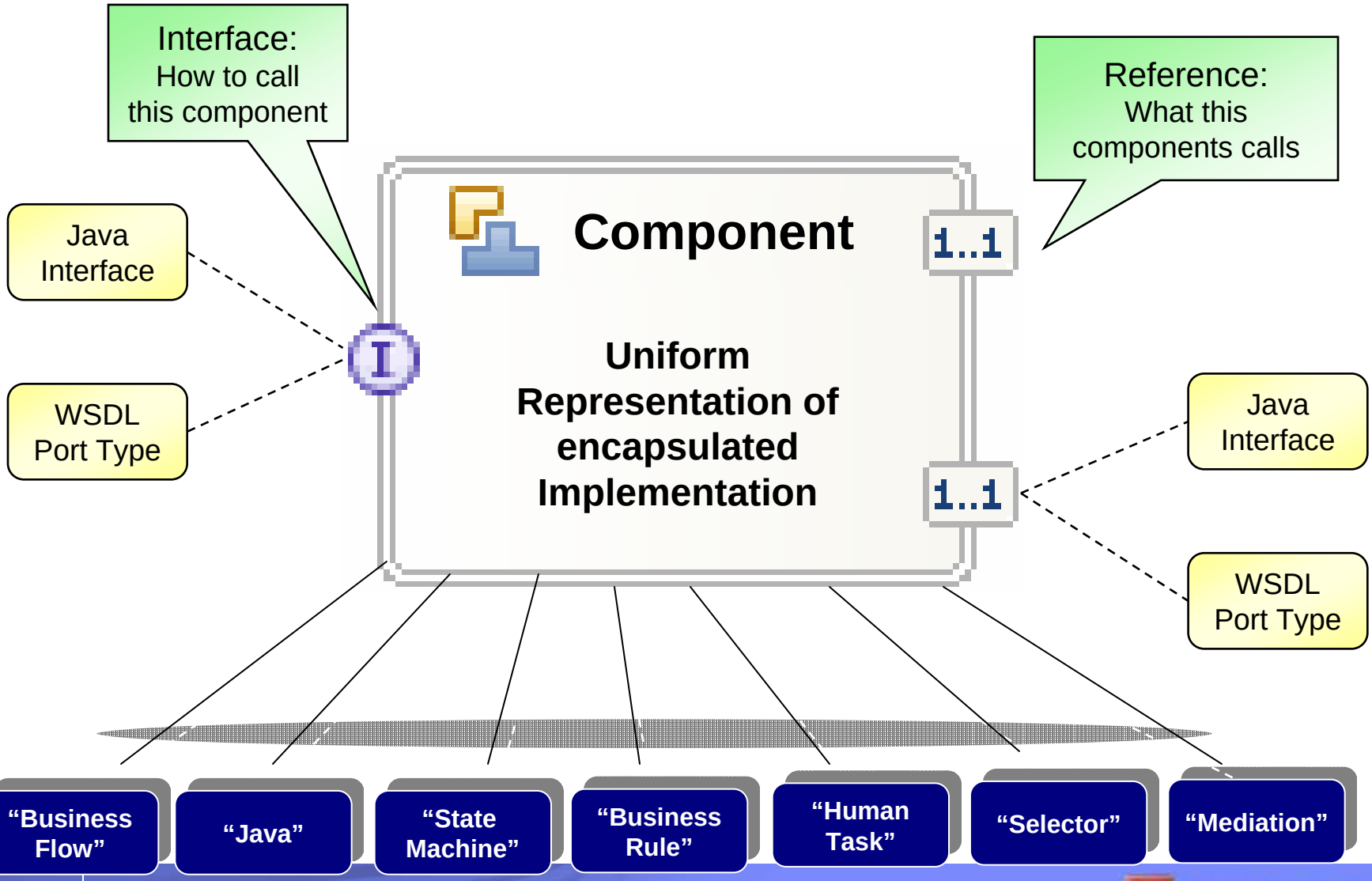


WebSphere Process Server v6 - Components

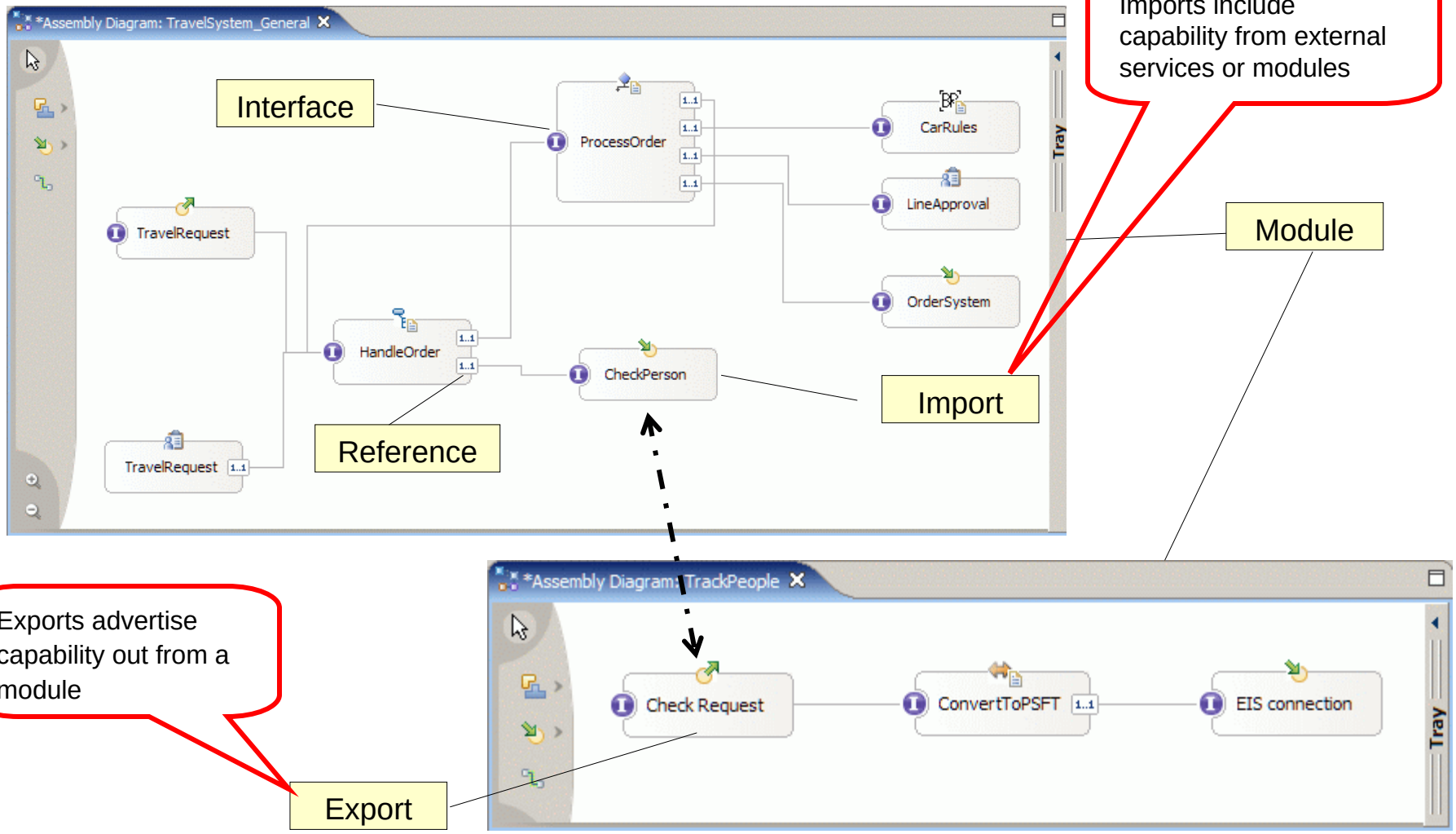
SOA Core



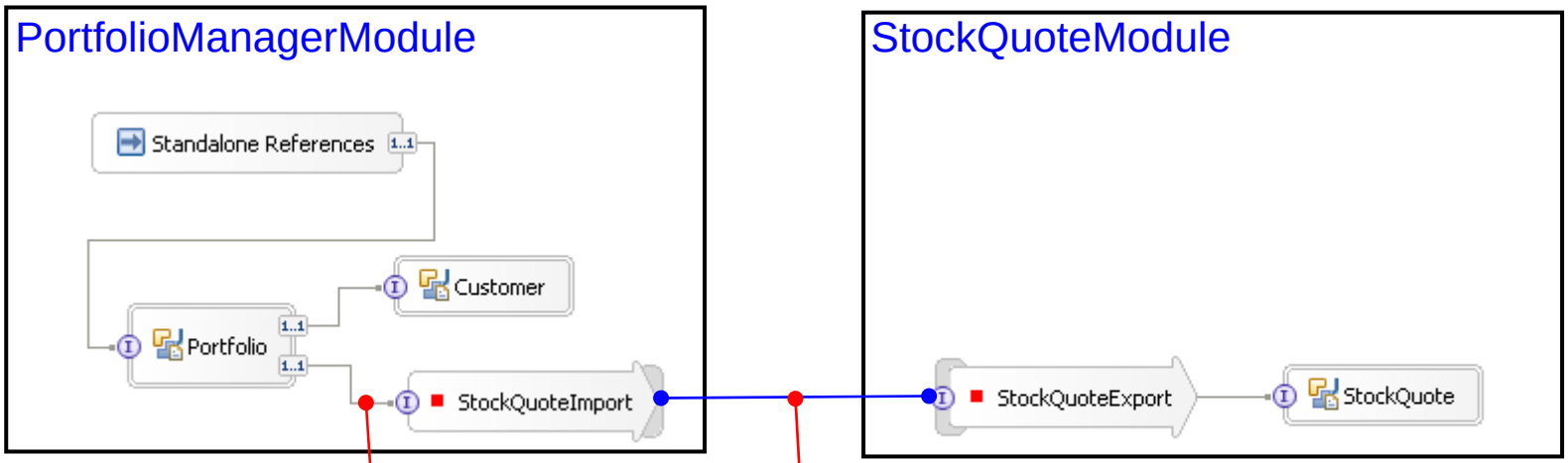
The Common Invocation Model: Service Components



Assembly Editor



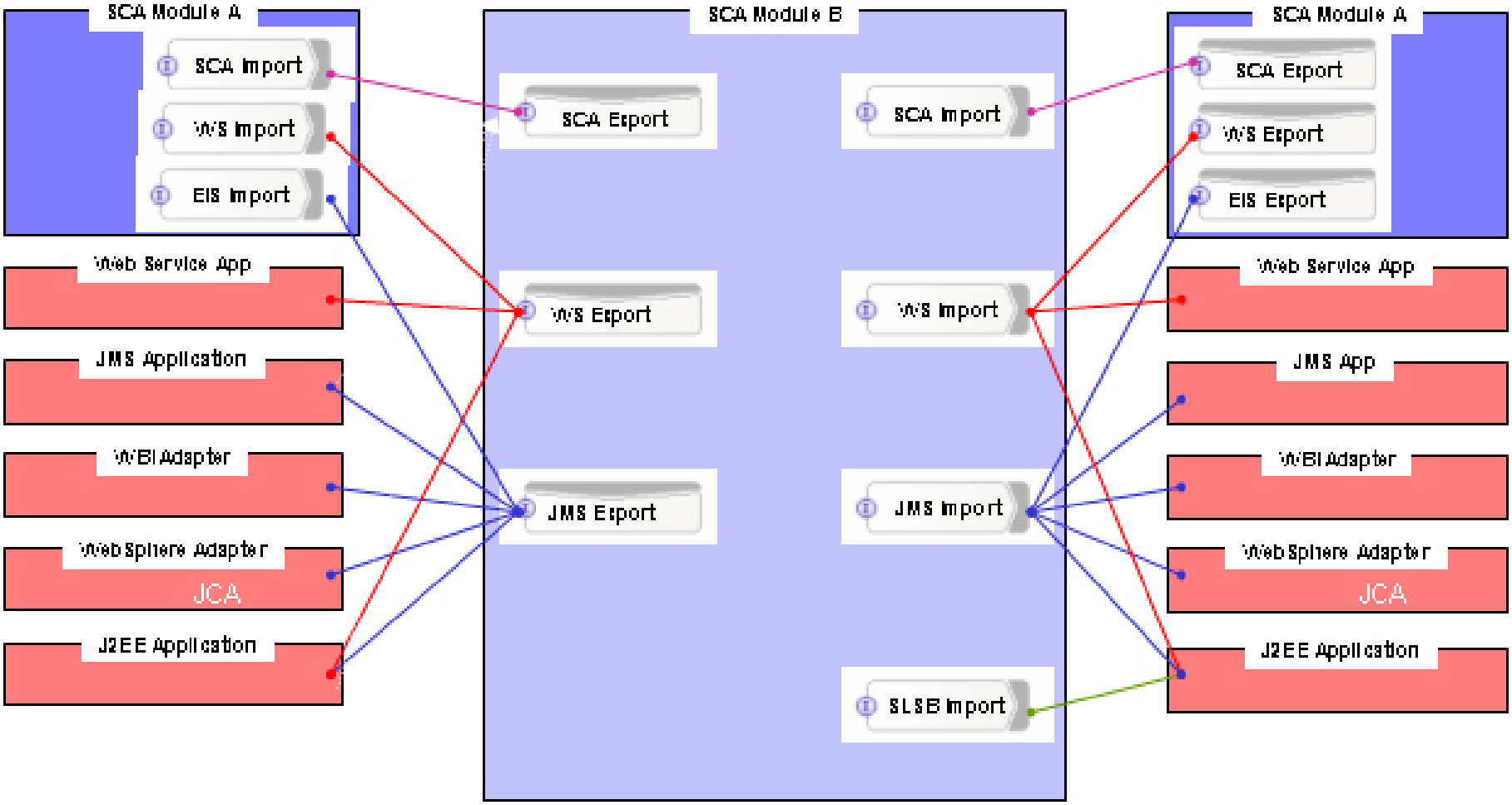
SCA Invocation Models



Invocation Models
Synchronous **(by value)**
Asynchronous – One Way (by value)
Asynchronous – Deferred Response (by value)
Asynchronous – Response with Callback (by value)

Invocation Models
Synchronous **(by ref)**
Asynchronous – One Way (by value)
Asynchronous – Deferred Response (by value)
Asynchronous – Response with Callback (by value)

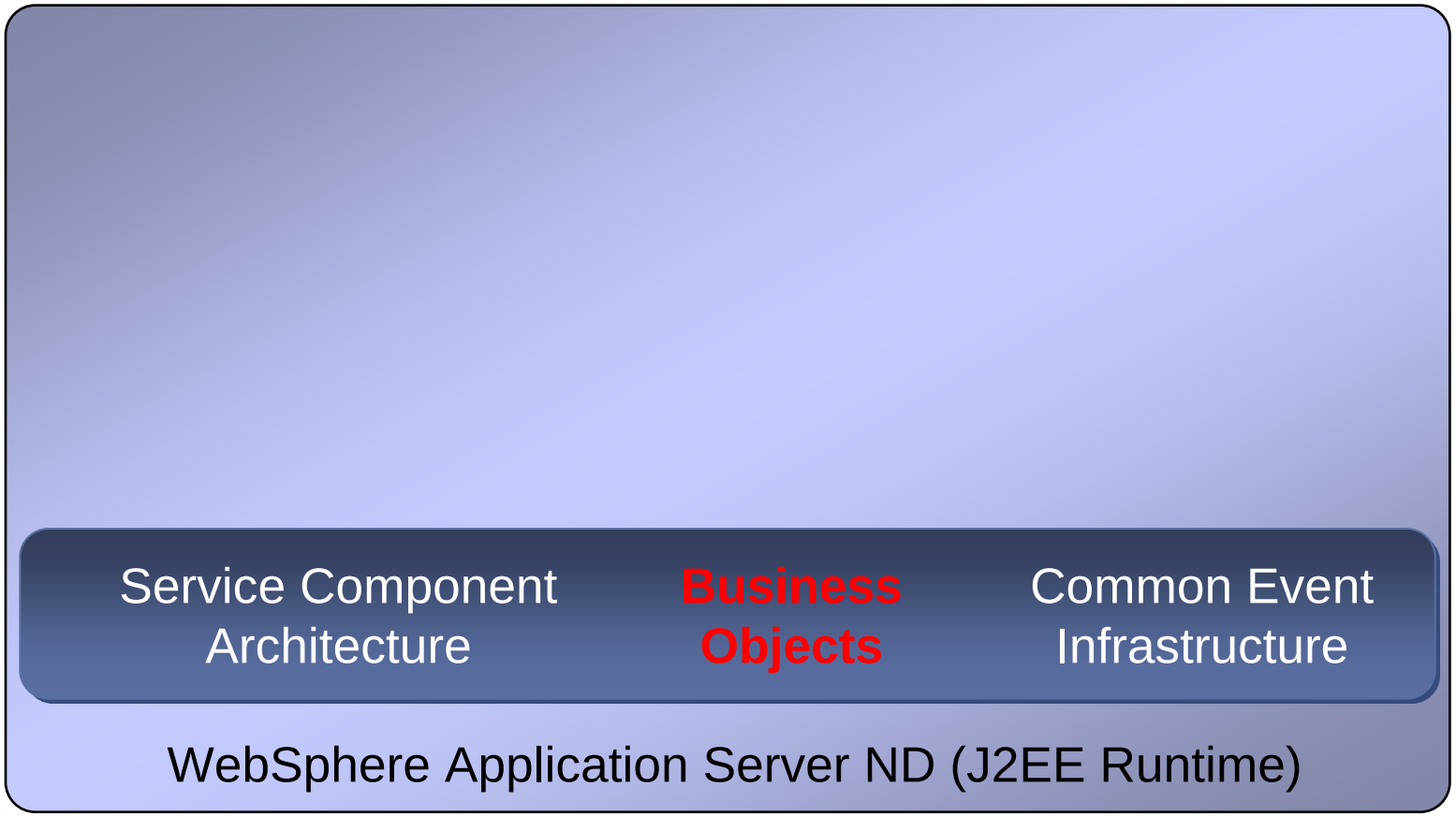
SCA Based Integration - Bindings



- Native 'SCA' bindings for SCA to SCA (modules)
- SCA components may call from (look left) a variety of client programming styles and support SCA programs to view a number of services as WSDL describe SOA services (look right)

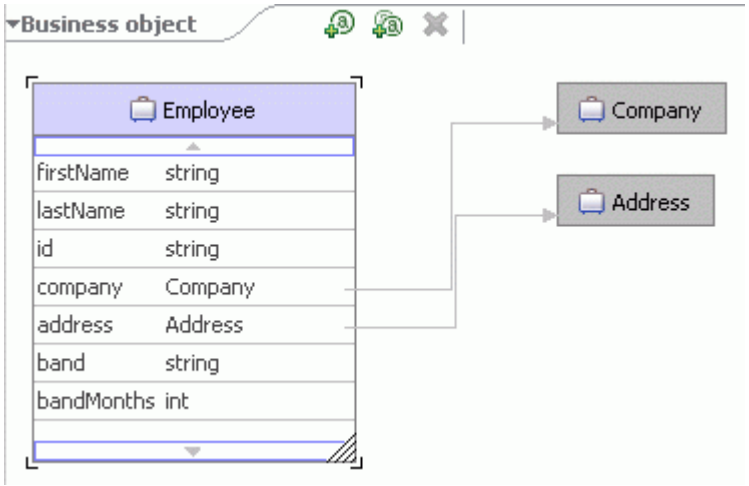
WebSphere Process Server v6 - Components

SOA Core

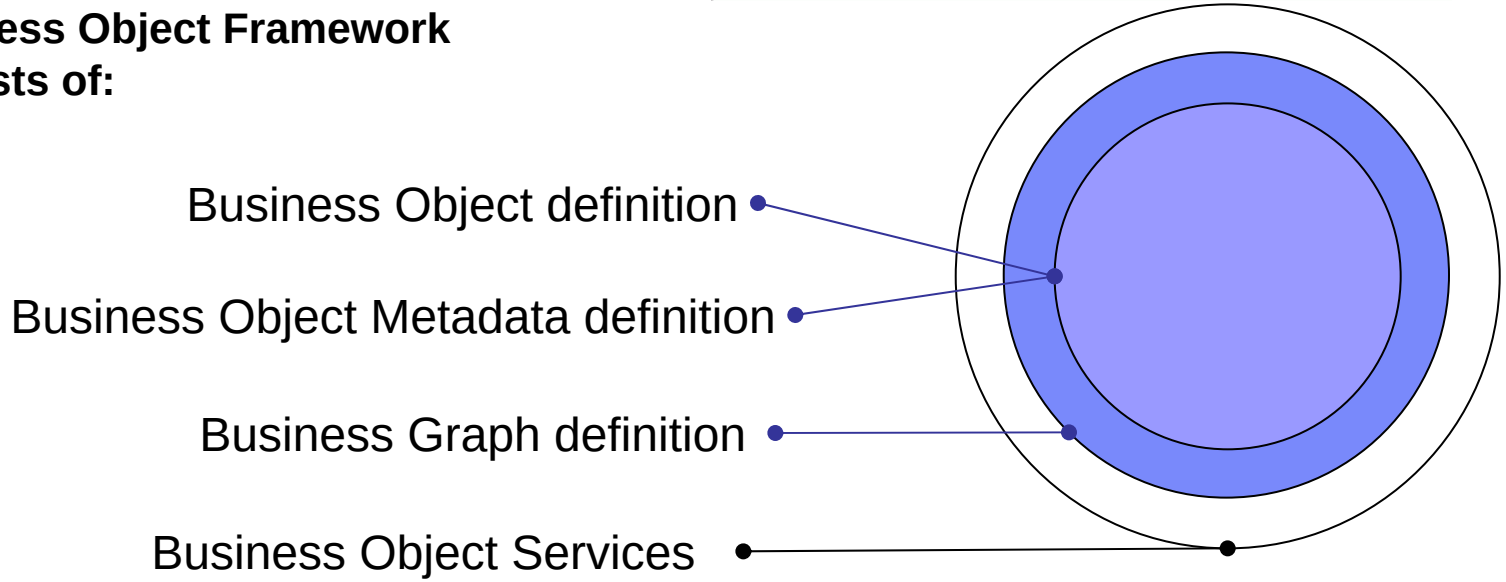


The Common Data Model: Business Objects

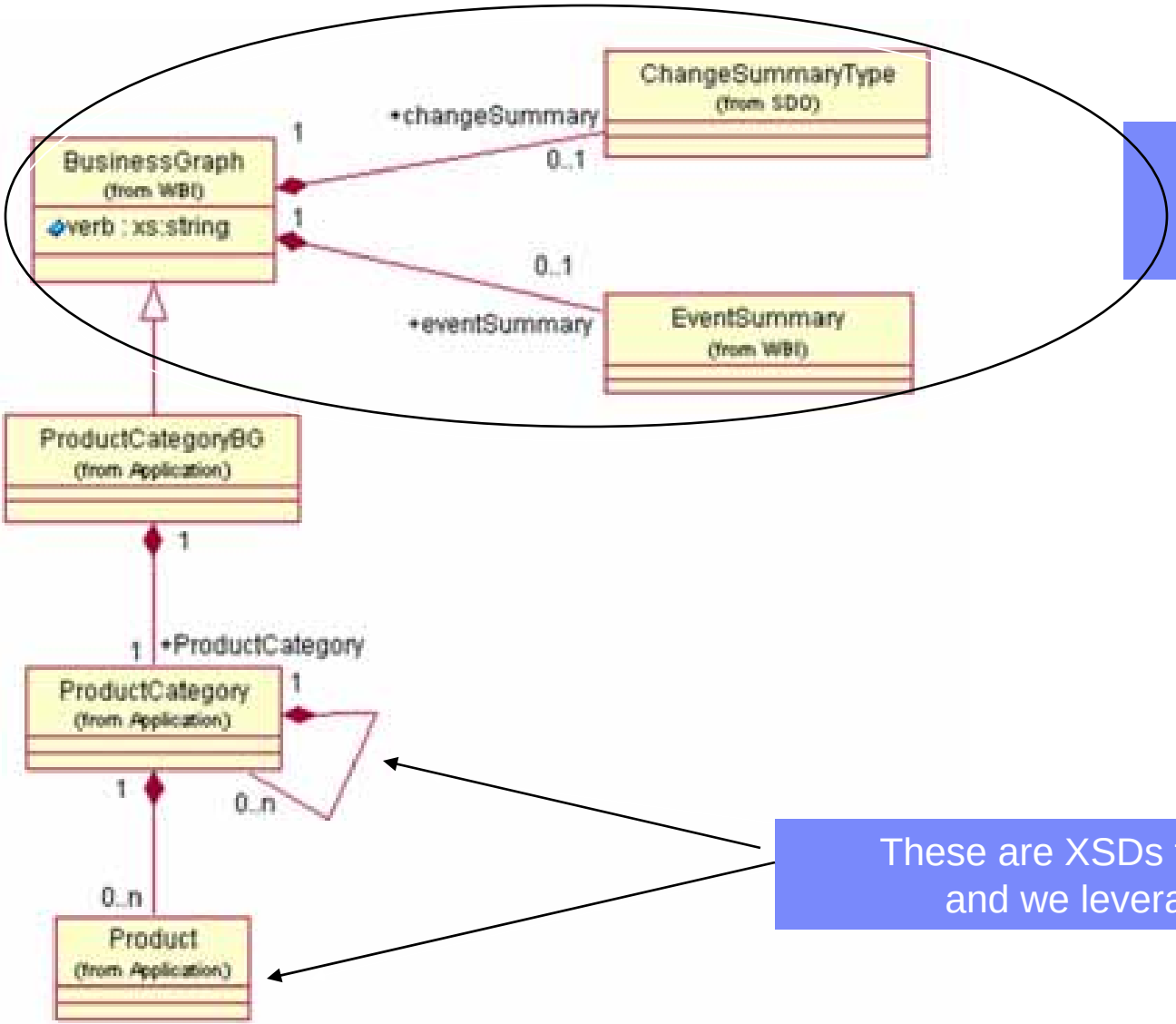
- **Enhanced Service Data Object**
  - Provides some function not available in base SDO specification (close to SDO 2.0)
  - Supports Inheritance and Aggregation
  - Enables import of ‘standard’ XSD



- **Business Object Framework consists of:**



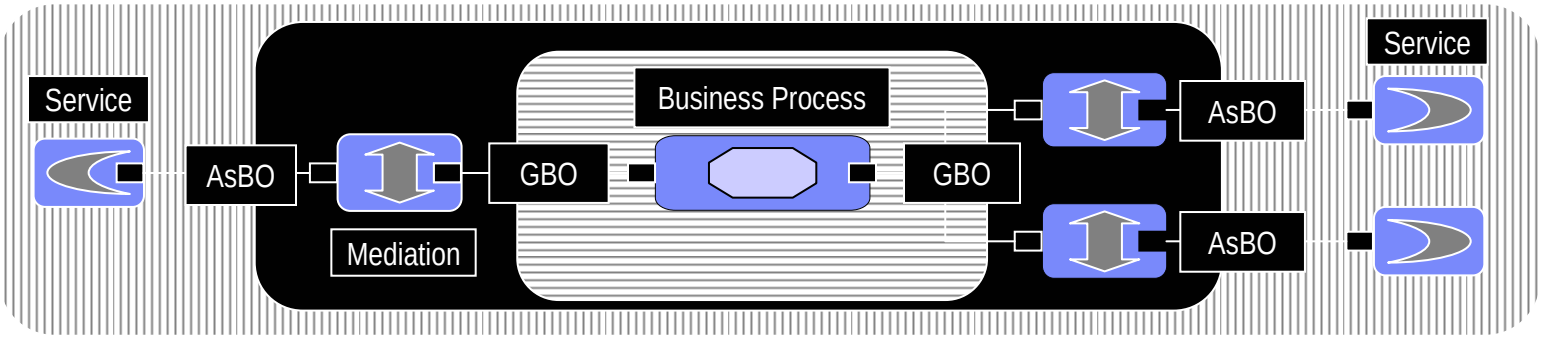
Business Graph Model Sample



We add this to the picture to carry additional value

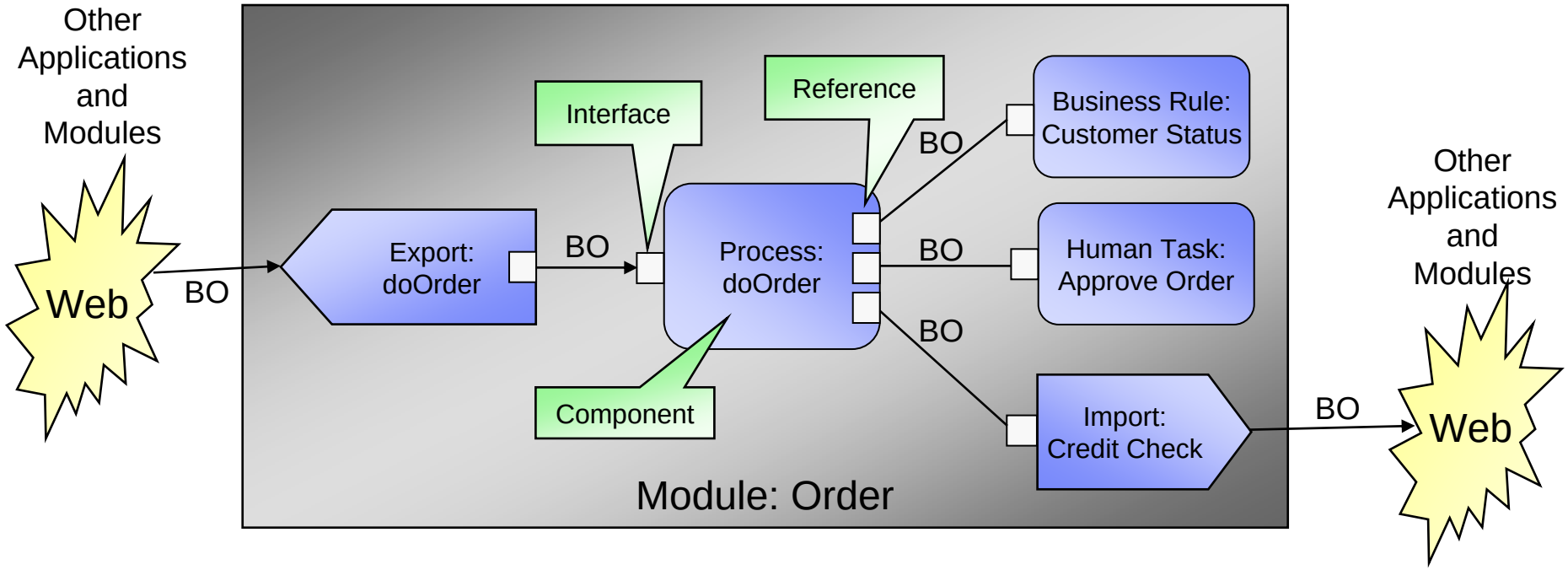
These are XSDs that might be ‘standard’ and we leverage them ‘unchanged’

Business Object – AsBO, GBO



- **Service emits/accepts a Business Object**
  - –Specific to this service -> Application Specific Business Object (AsBO)
- **Mediation on every endpoint**
  - –Convert AsBo to GBO and GBO to AsBo
  - –Cross-Referencing to keep BOs in sync
- **Business Process operates on a Generic Business Object (GBO)**
  - –Superset of all possible AsBOs
  - –Process is independent of the actual services - Services can be replaced without impacting the process or other services

SCA and Business Objects – Conceptual View



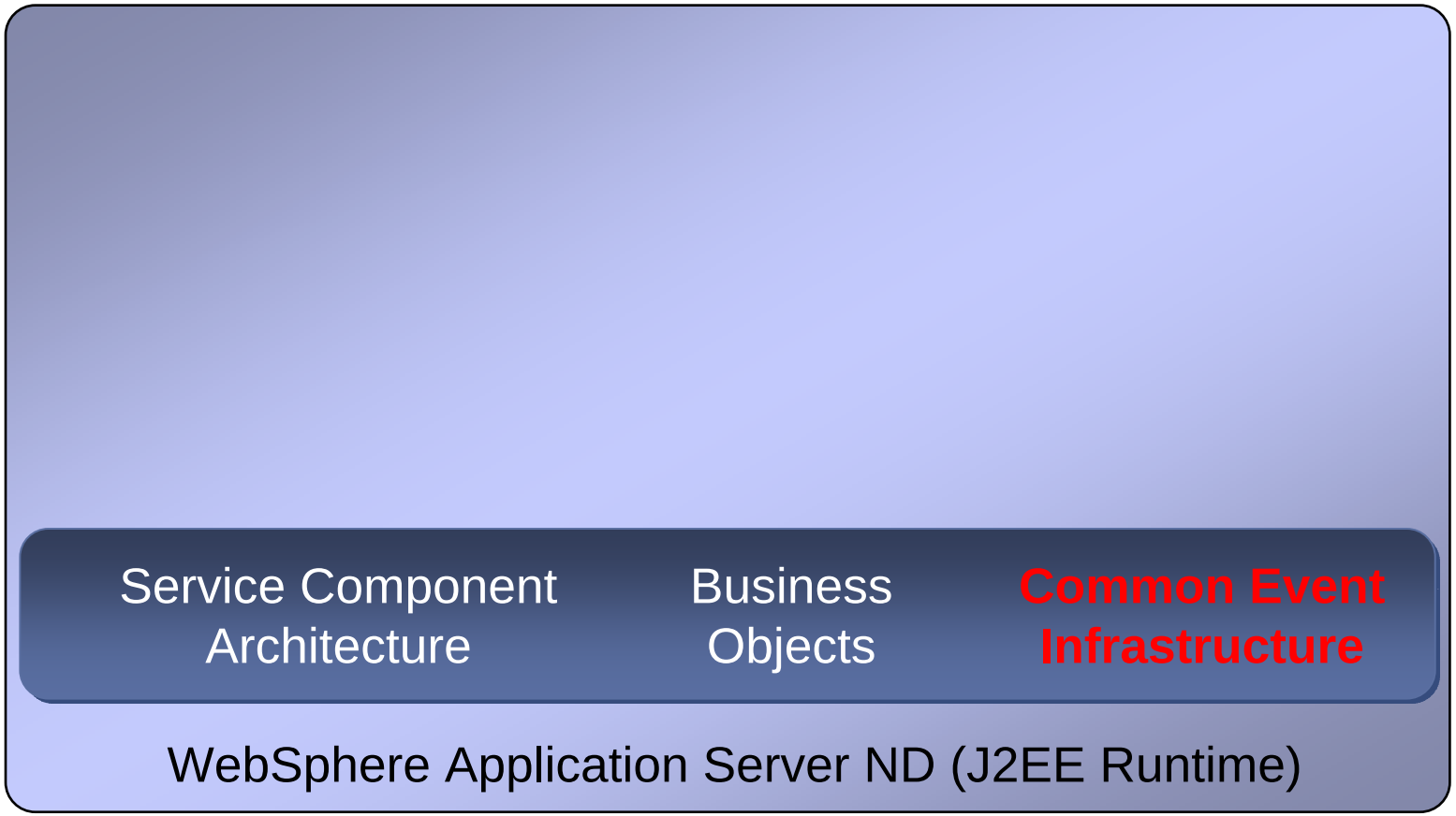
- SCA is the component model
- Components may be wired together
- Business Objects are the data flowing on wires between Components
- Exports advertise capability out from a module
- Imports include capability from external services or modules

SCA / SDO – Industry specification announcement

- Industry leaders endorse SCA
- Industry collaboration announced on Nov 30th, 2005: A series of specifications aimed at developers building solutions and components using Service Oriented Architecture principles:
  - Service Component Architecture (SCA)
  - Service Data Objects (SDO)
- Submitted to the Tuscany project proposal to Apache
- The Service Component Architecture specifications are co-authored by IBM, BEA, Oracle, SAP, Siebel, IONA, and Sybase, with further support from Interface21.
- The Service Data Objects specifications are co-authored by IBM, BEA, Oracle, SAP, Siebel, Xcalia, and Sybase, with further support from Zend.
- The specs on the IBM web site:
  - http://www.ibm.com/developerworks/library/specification/ws-sca/

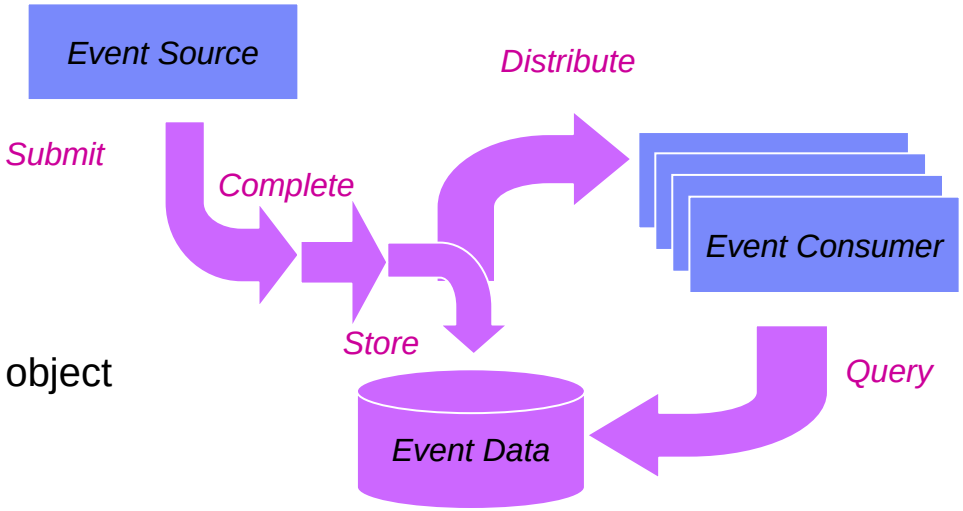
WebSphere Process Server v6 - Components

SOA Core



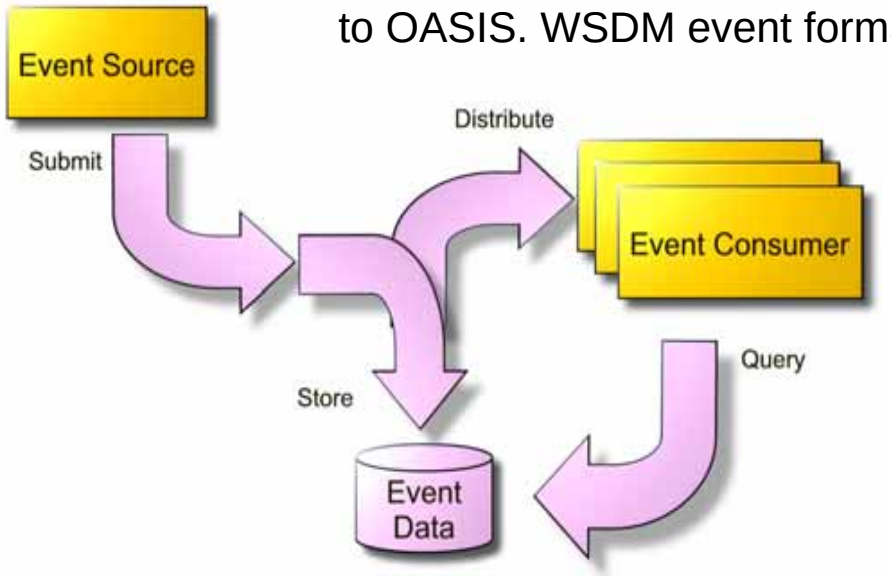
Common Event Infrastructure

- An event occurs when something significant happens in the IT system
- Based on CEI (Common Event Infrastructure)
- Data about the event are captured in an event object
  - Has a standardized format called the Common Base Event (CBE)
  - Application supplies the business data
- All event objects are passed to the event infrastructure to enable:
  - Tracking the progress of a business process
  - Audit trails
  - Coordinating work between independent business processes
  - Monitoring for exceptions in a business process
- Generated by runtime environment
  - API to generate custom CEI events



CBE and CEI

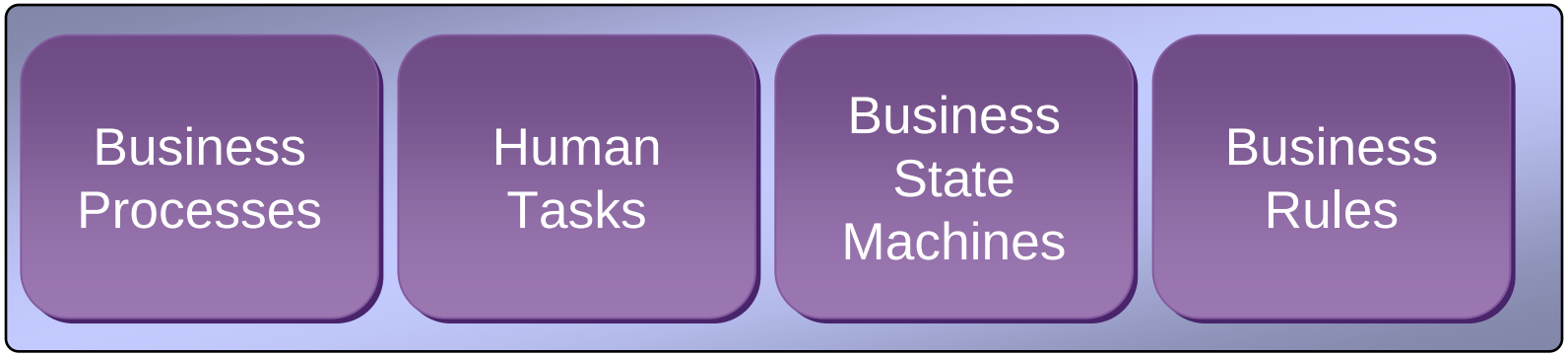
- CEI, Common Event Infrastructure, is IBM's implementation of a consistent approach for the creation, transmission, persistence and distribution of a wide range of business, system and network events, based on common base events.
- CBE, Common Base Event, defines the “event” data format that is generated based on the definition of the business measure. An “Event” is anything interesting that occurs from either a business or an IT perspective. CBE is the event data format IBM has proposed, and was accepted, as a standard to OASIS. WSDM event format (WEF) is OASIS standard version of CBE



A Common Base Event (CBE) flows over the Common Event Infrastructure (CEI)

Service Components

- **Business Processes**
- **Business State Machine**
- **Human Tasks**
- **Business Rules**

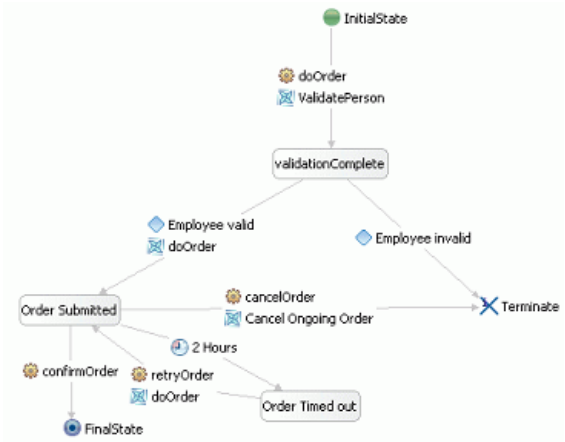


Two Styles of Service Choreography

Business Processes

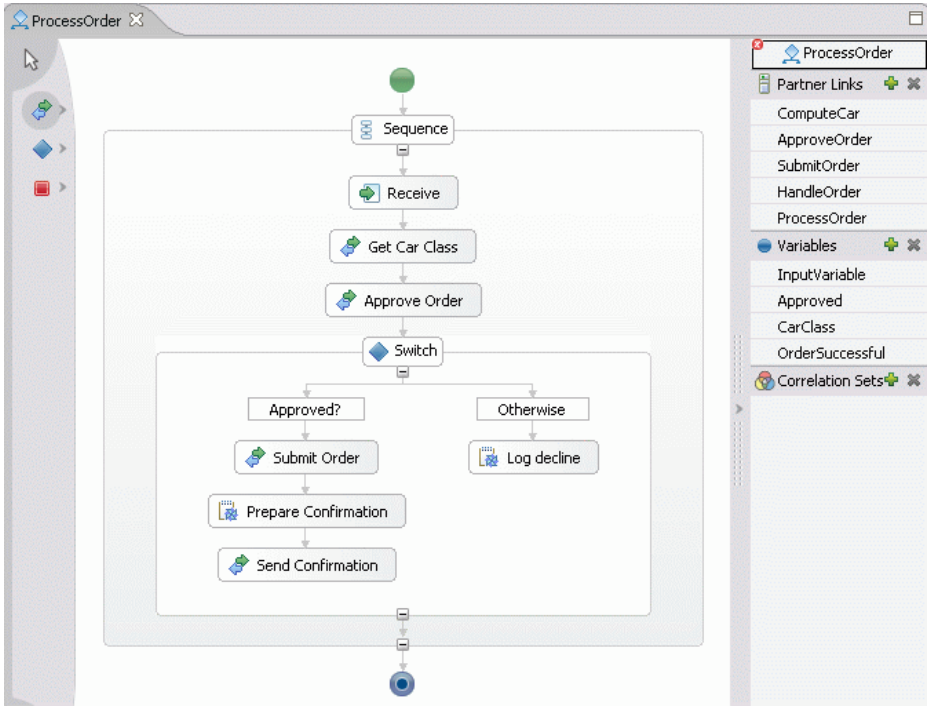
- **Business Processes**
  - Traditional Business Processes
  - Full support for WS-BPEL
  - Import from WebSphere Business Modeler

- **Business State Machines**
  - Event-driven Business Processes
  - Full support for State Machine programming model
    - States, Events, Transitions, Actions, Guards, …
    - State Machine Authoring / Debugging / Logging



Business Process

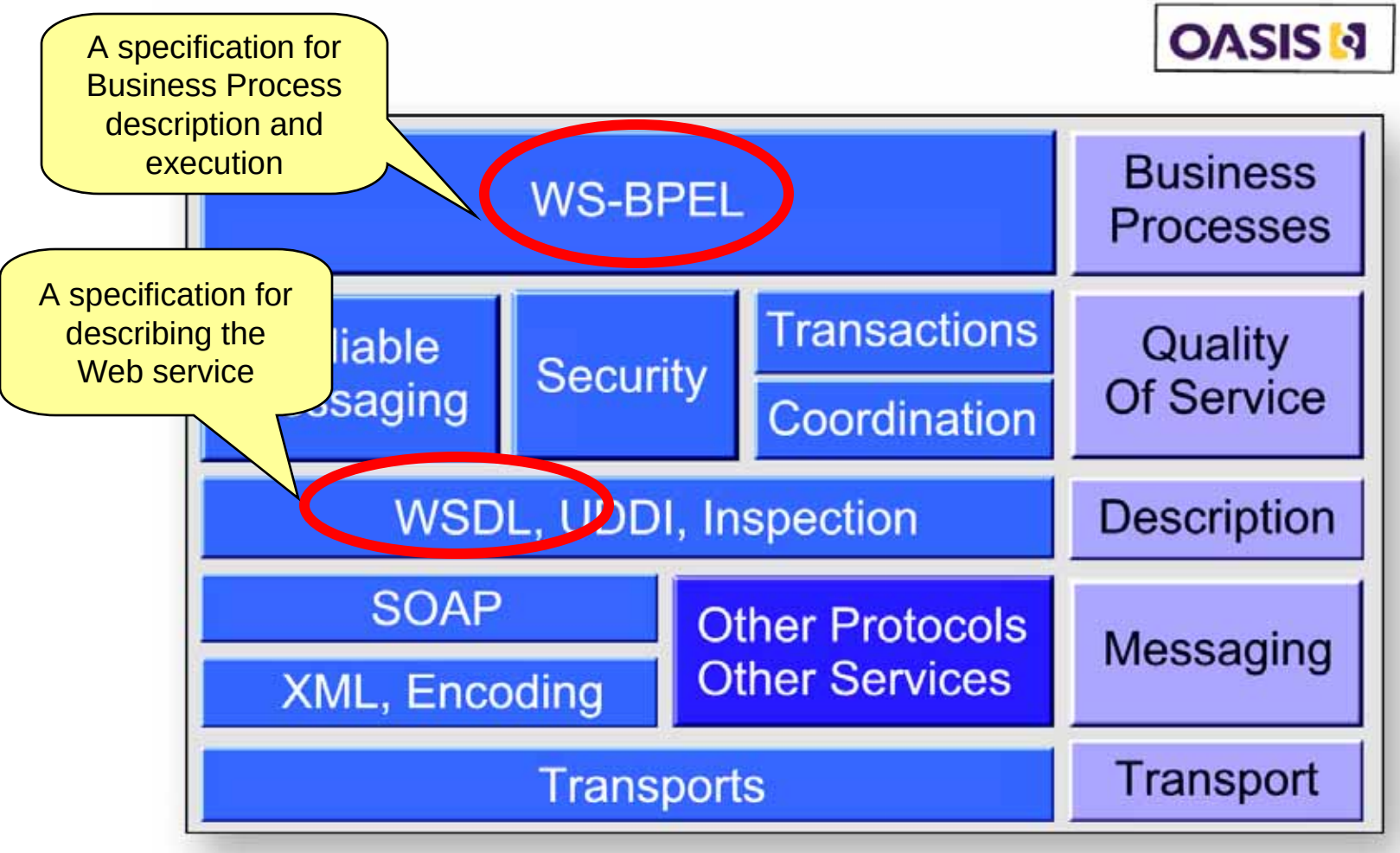
- **WS-BPEL compliant business process engine**
  - WS-BPEL 1.1 + 2.0 Draft
  - Optionally, generated from WebSphere Business Modeler
- **Generic Business Process**
  - Operations / Parameters
  - Service Implementation Details hidden
- **Transactions / Compensation**
- **Full XPath 1.0 Support**
- **Visual Debugger**



What is WS-BPEL?

- Web Services Business Process Execution Language, also called BPEL
- Previously known as BPEL4WS (Business Process Execution Language for Web Services)
- Industry standard for web services choreography
- A language to specify behavior of business processes
  - As Web services
  - Between Web services
- Builds on and extends XML and Web Services specifications

Web Services Standards …



- Web Services Distributed Management (WS-DM) and its Web Event Format (WEF)
- Specification approved on March 9, 2005

Major Standards Organization

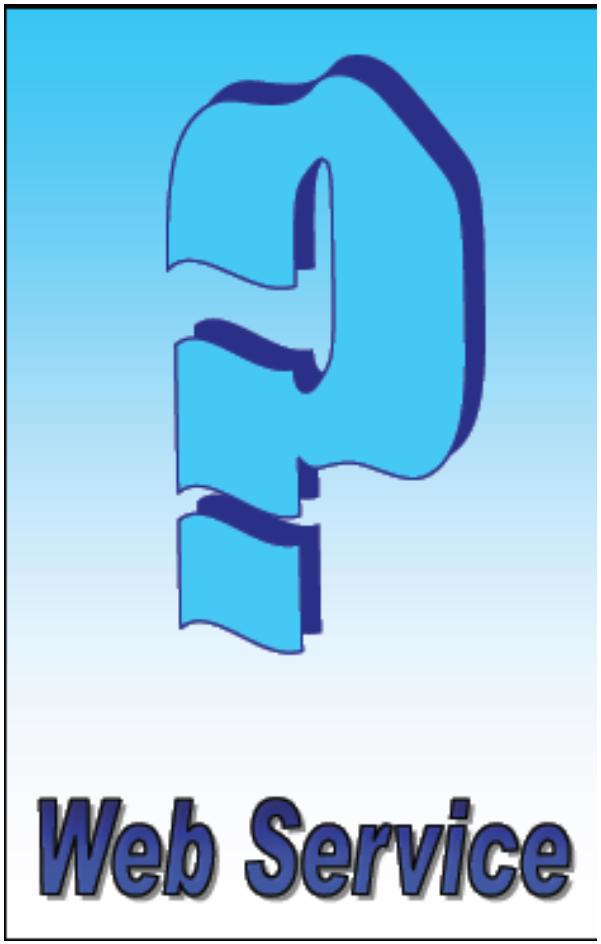
- **W3C (World Wide Web Consortium) – http://www.w3.org**
  - Develops interoperable technologies (specifications, guidelines, software, and tools) to lead the Web to its full potential. Covers HTML, HTTP, XML, SOAP, etc. Founded in 1994 and includes 350 member organizations from around the world
- **OASIS (Organization for the Advancement of Structured Information Standards) - http://www.oasis-open.org**
  - Drives the development and adoption of Web services, security, ebusiness, public sector and application-specific standards (BPEL, ebXML, UDDI, WSRP, WS-Security, etc.). Founded in 1993, has more than 3,000 participants from 600 organizations in 100 countries
- **WS-I**
  - Is an open industry effort to promote Web Services interoperability across platforms, applications, and programming languages. Provides guidance, recommended practices, and supporting resources for developing interoperable Web services (WS Basic Profile)

History of BPEL

- 7/2002: Original 1.0 BPEL4WS proposal from IBM, Microsoft and BEA. Combined ideas from IBM's WSFL and Microsoft's XLang.
- 4/2003: OASIS Technical Committee formed. Standards-based follow-on to earlier BPEL4WS work.
- 5/2003: Revised 1.1 proposal with contributions from SAP and Siebel.
- 4/2004: Implementation of BPEL 1.1 available in IBM's WebSphere Business Integration Server Foundation V5.1
- BPEL V2.0 draft available today, planned for 1Q06
- 9/2005: Implementation of BPEL 1.1 and BPEL 2.0 draft available in IBM's WebSphere Process Server V6

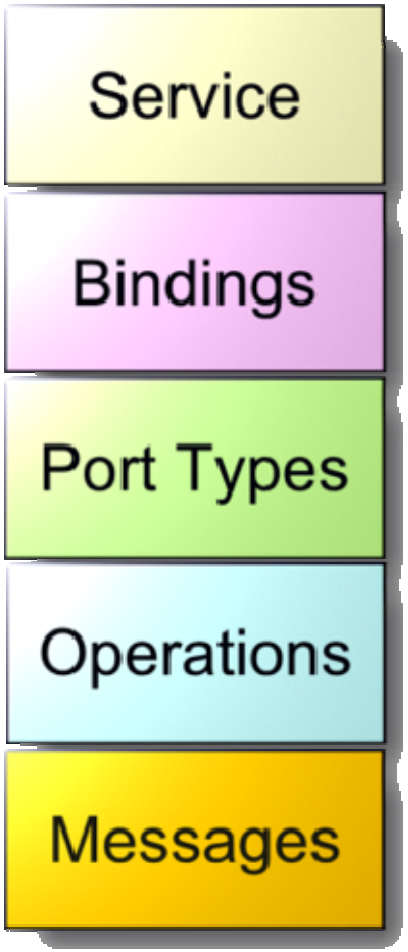
What is a Service …?

- Web Services Description Language (WSDL) is used to describe a Service or set of Services
- A "Service" is a set of related application functions that can be programmatically invoked over the internet. Businesses can dynamically mix and match Web Services to perform complex transactions with minimal programming.
- Web service is a self-contained, self-describing modular applications that can be published, located and invoked across the Web.



WSDL …

- **Web Services Description Language**
- **Open Standard for describing Interfaces to Services**
- **Characteristics**
  - Describes data expected to be sent and received
  - Describes what the Service can do
  - Describes how to reach the service
- **WSDL description is an XML document that conforms to the WSDL standard**
- **If you have a WSDL file, you know how to interact with a service!!**

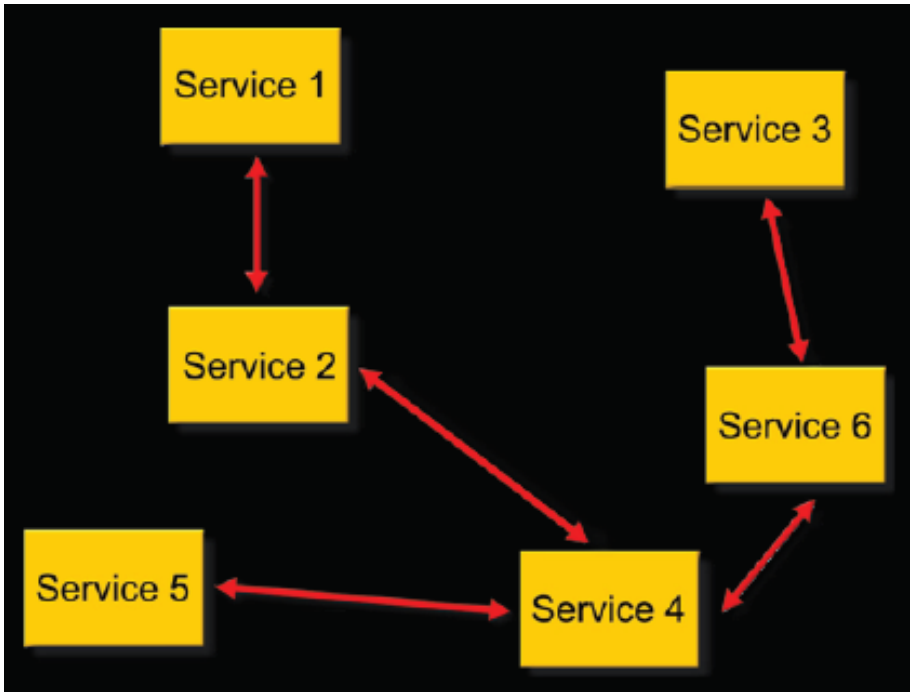


Composing Software Services …

- Software Solutions become composable
- Services become building blocks
- Solutions are now part of a new paradigm:

Service Oriented Architecture

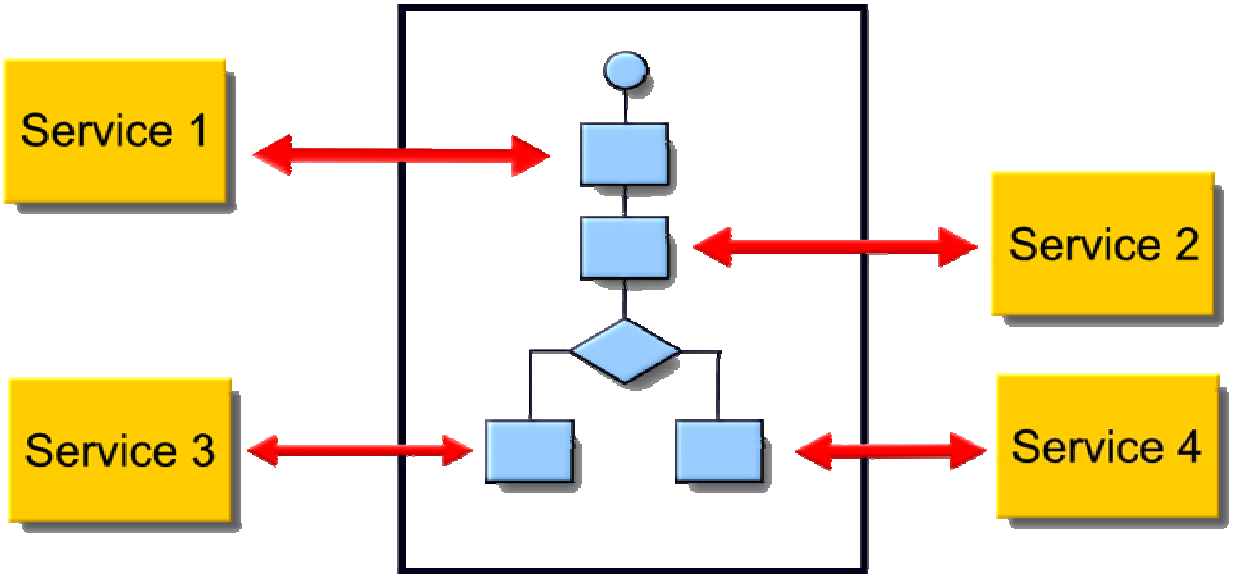
A set of architectural principles and patterns which address characteristics such as modularity, encapsulation, loose coupling, separation of concerns, reuse, composable and single implementation



This new idea of composing software solutions from Service building blocks is called the Service Oriented Architecture

Process Choreography …

- The decision of which Services are invoked
- The decision of what order Services are invoked
- The transformation of data output from one Service and input to another



Process is a “Flow Chart” of execution paths!!

Why WS-BPEL?

- Industry wide language for business processes
  - common programming skill
  - Industry investment (including 3rd party vendors)
- Based on other standards including WSDL and other XML standards
  - WSDL defines interface of composed service and services used by composite
  - XML Schema and XPath for data context handling and business rules specification
- Choreography of services independently of their implementation
- WS-BPEL is vendor independent – allows for portable business processes
- An important bridge between the J2EE and the .Net worlds
- Allows stateful, long-running interactions between service based business partners
- IBM extensions

Elements of a BPEL Process: Activities

A BPEL Business Process is composed of

- **Basic activities**
  - Which are the things that we need to do as part of a business process
  - Receive input, reply to business partners or other business processes, manage exceptions, make decisions
- **Structuring activities**
  - Help us organize and manage the complexity of the flows
  - Typical programming constructs

Basic BPEL activities

	Do a blocking wait for a matching message to arrive
	Send a message in reply to a formerly received message
	Invoke a one-way or request-response operation
	Update the values of variables or partner links with new data
	A “no-op” instruction for a business process
	Generate a fault from inside the business process
	Forward a fault from inside a fault handler
	Immediately terminate execution of a business process instance (“Exit”)
	Wait for a given time period or until a certain time has passed
	Invoke compensation on an inner scope that has already completed

Structured BPEL activities

Contained activities are executed in parallel, partially ordered through control links (“Flow”)

Contained activities are performed sequentially in lexical order

Select exactly one branch of activity from a set of choices (“If then else”)

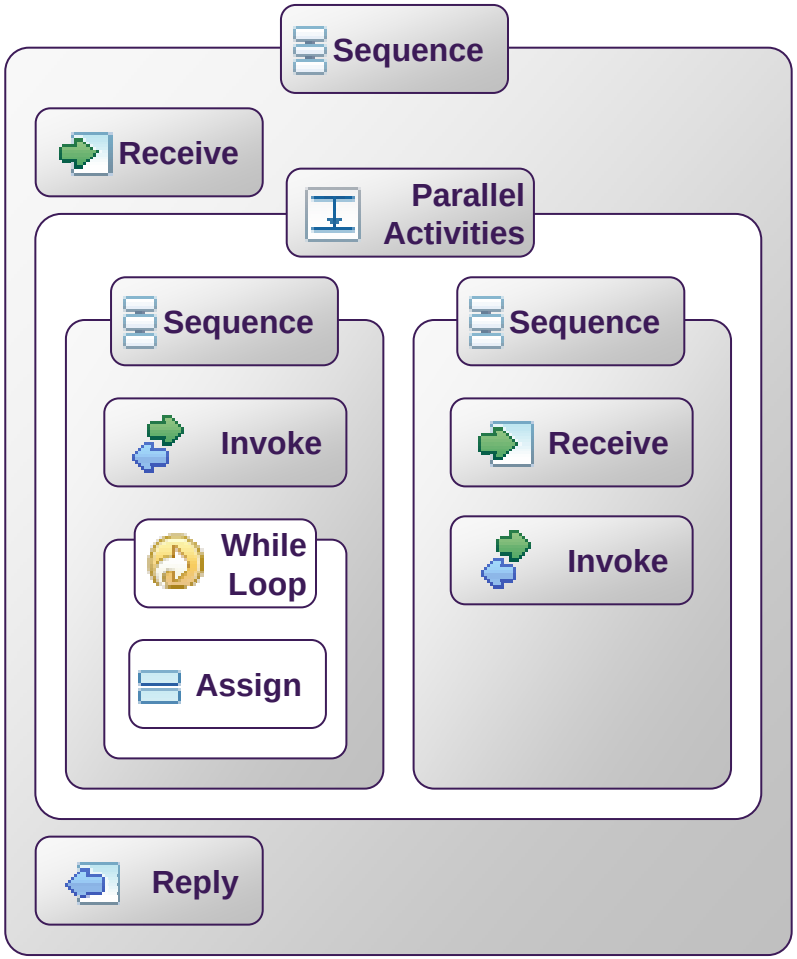
Block and wait for a suitable message to arrive (or time out) (“Pick”)

Contained activity is repeated while a predicate holds

Associate contained activity with its own local variables, fault handlers, compensation handler, and event handlers

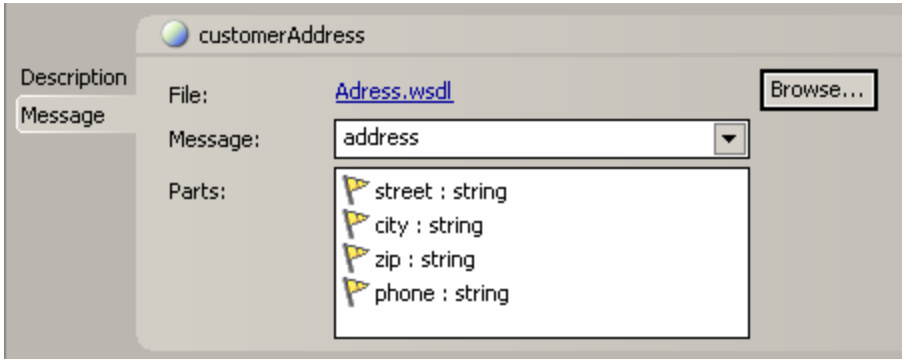
Nesting structured activities

```
<sequence>
   <receive .../>
   <flow>
      <sequence>
         <invoke .../>
         <while ... >
            <assign>...</assign>
         </while>
      </sequence>
      <sequence>
         <receive .../>
         <invoke ... >
      </sequence>
   </flow>
   <reply>
</sequence>
```



Elements of a BPEL Process: Variables

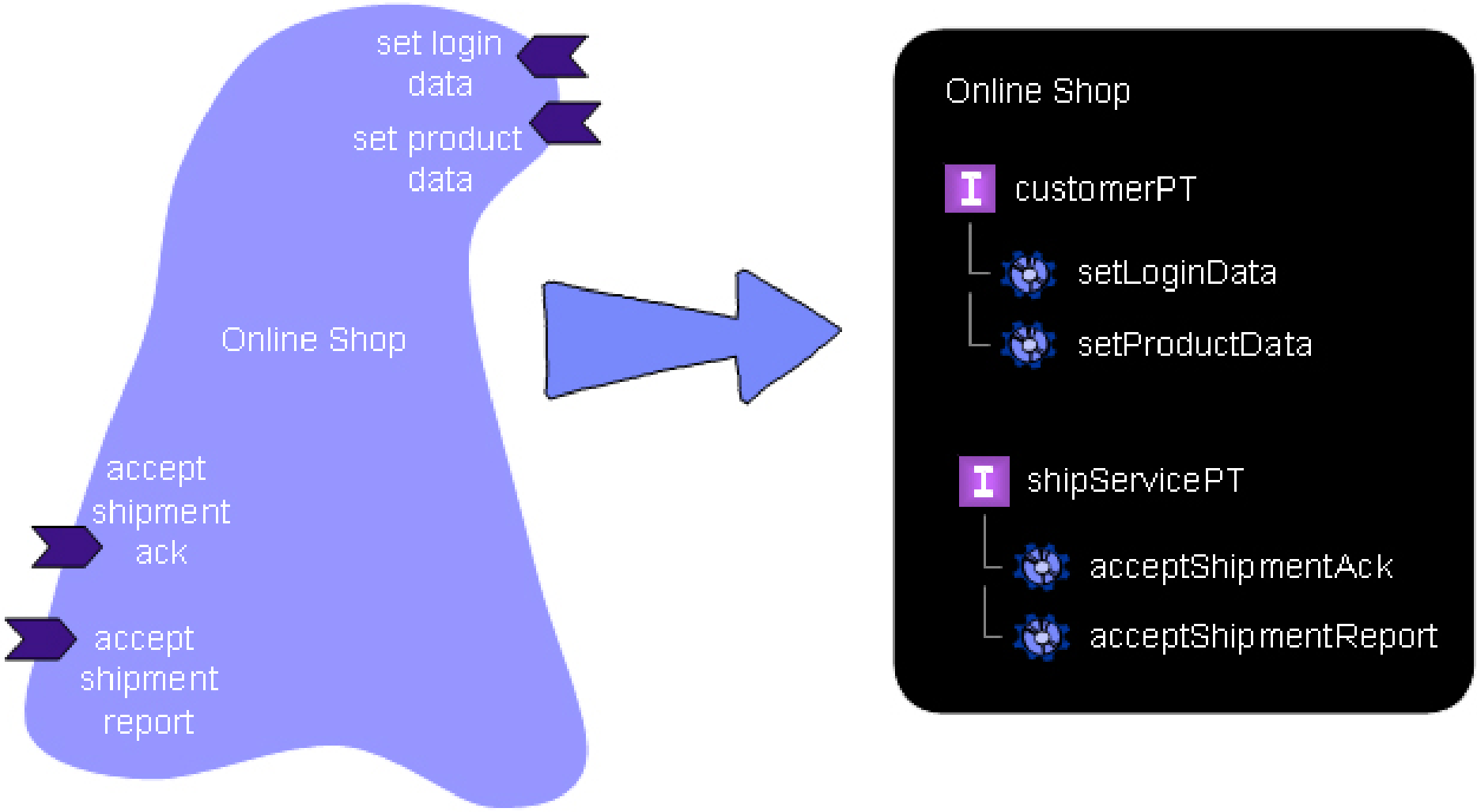
- **Hold data that constitutes the state of a process**
  - May be received from or sent to partners
  - Can be specified as input or output variables for invoke, receive, and reply activities
  - May hold state data related to the process and never exchanged with partners
- **Associated with WSDL message types**



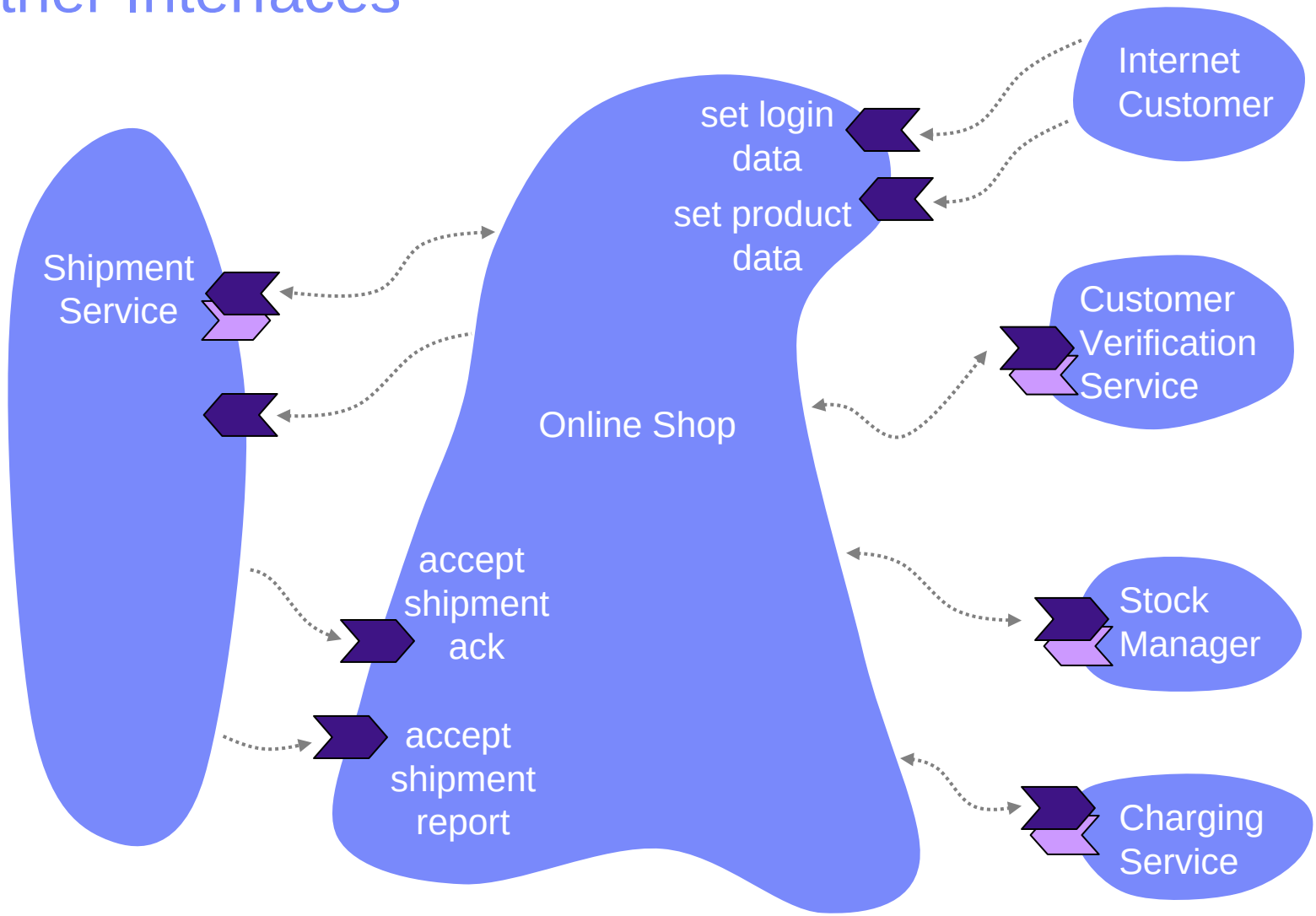
Elements of a BPEL Process: Partner Links

- **Partner: BPEL term for any entity that a process is interacting with**
  - Business Partner, i.e. a web service
  - Internal Service, i.e. an EJB
  - Process Starter, e.g. a web application
  - ...
- **Partner Link: "Placeholder" for a partner**
  - Part of the process definition
  - No need to specify concrete service endpoints within the process model
  - Allows for late binding of partners (at assembly time ↔ build time)
- **Allows for long-running, stateful interactions with a partner**

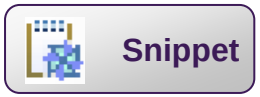
WSDL: Service Interface



Partner Interfaces



Additional basic activities (BPEL extensions)



Java snippet

Human task **(also known as "staff" activity in Version 5)**

Task kind: inline task

Task type: participating

Alternative to "inline task"**:**

→ Standalone task (NEW)

implemented as BPEL invoke activity

and human task SCA component.

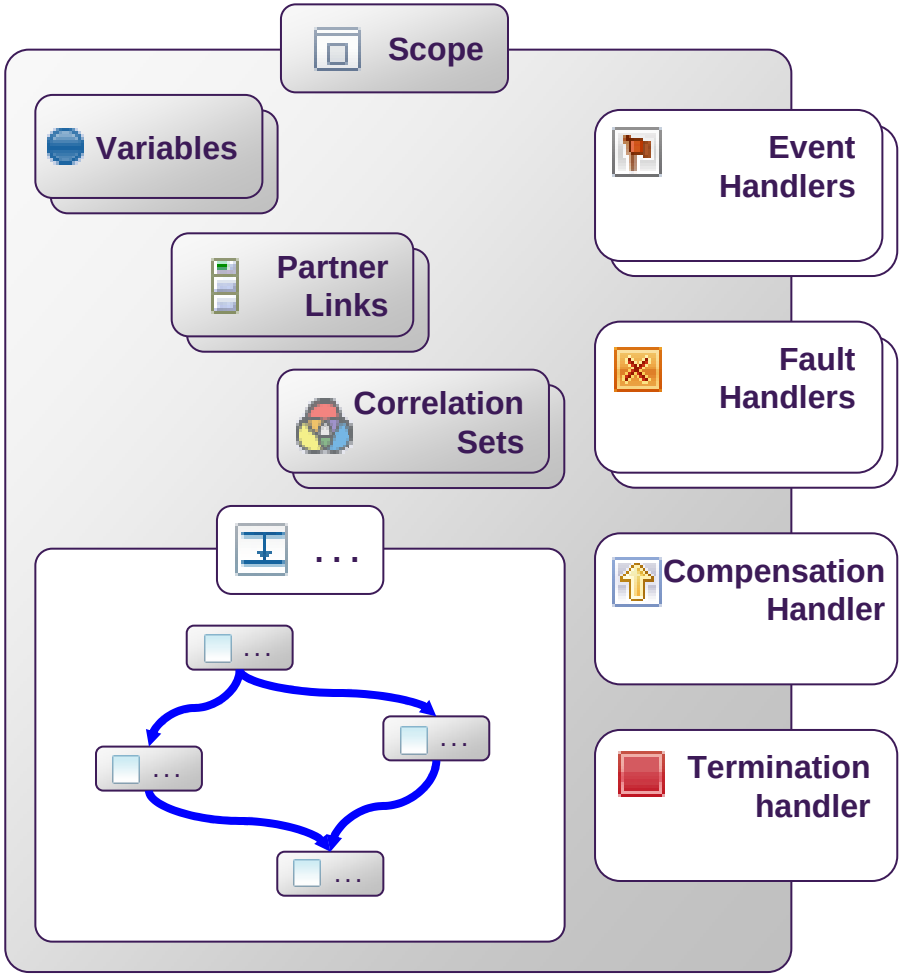
Scopes and handlers

Scope

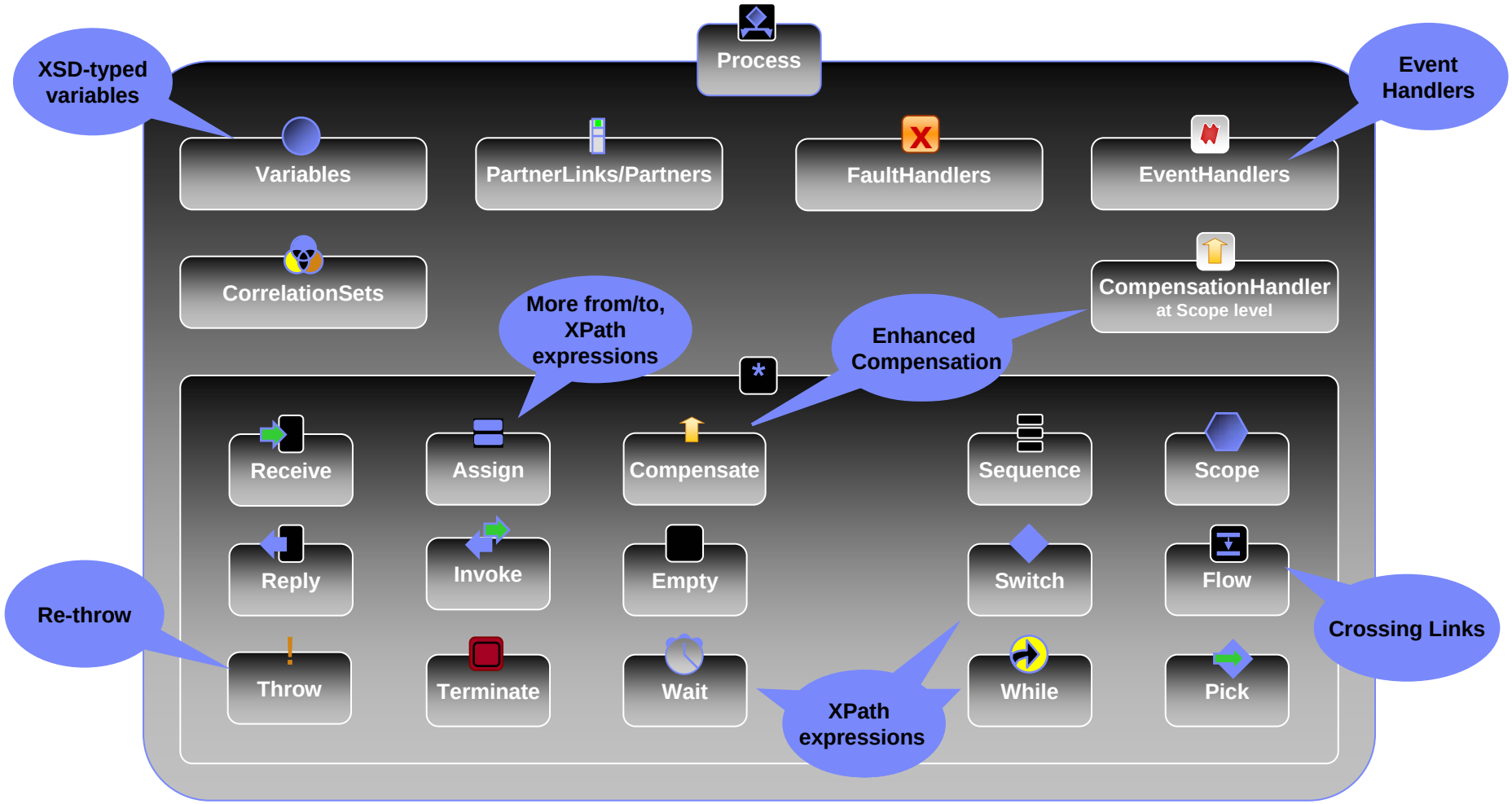
- Local variables
- Local partner links
- Local correlation sets
- Set of activities (basic or structured)

Handlers

- **Event handlers**
  - Message events or timer events (deadline or duration)
- **Fault handlers**
  - Dealing with different exceptional situations (internal faults)
- **Compensation handler**
  - Undoing persisted effects of already completed activities
- **Termination handler**
  - Dealing with forced scope termination (external faults)

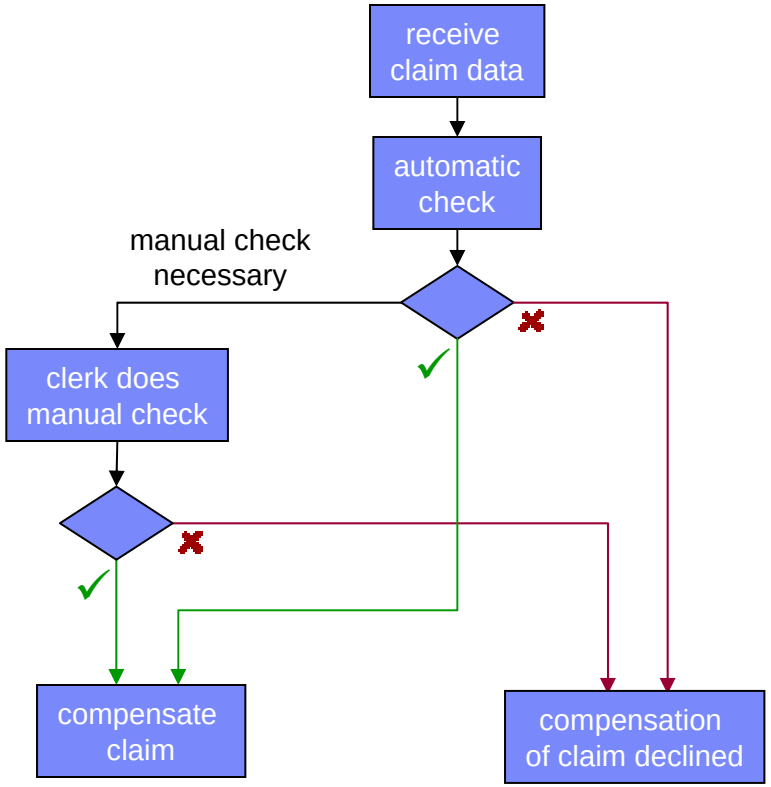


WS-BPEL 2.0 in WPS V6 - Business Flow Manager

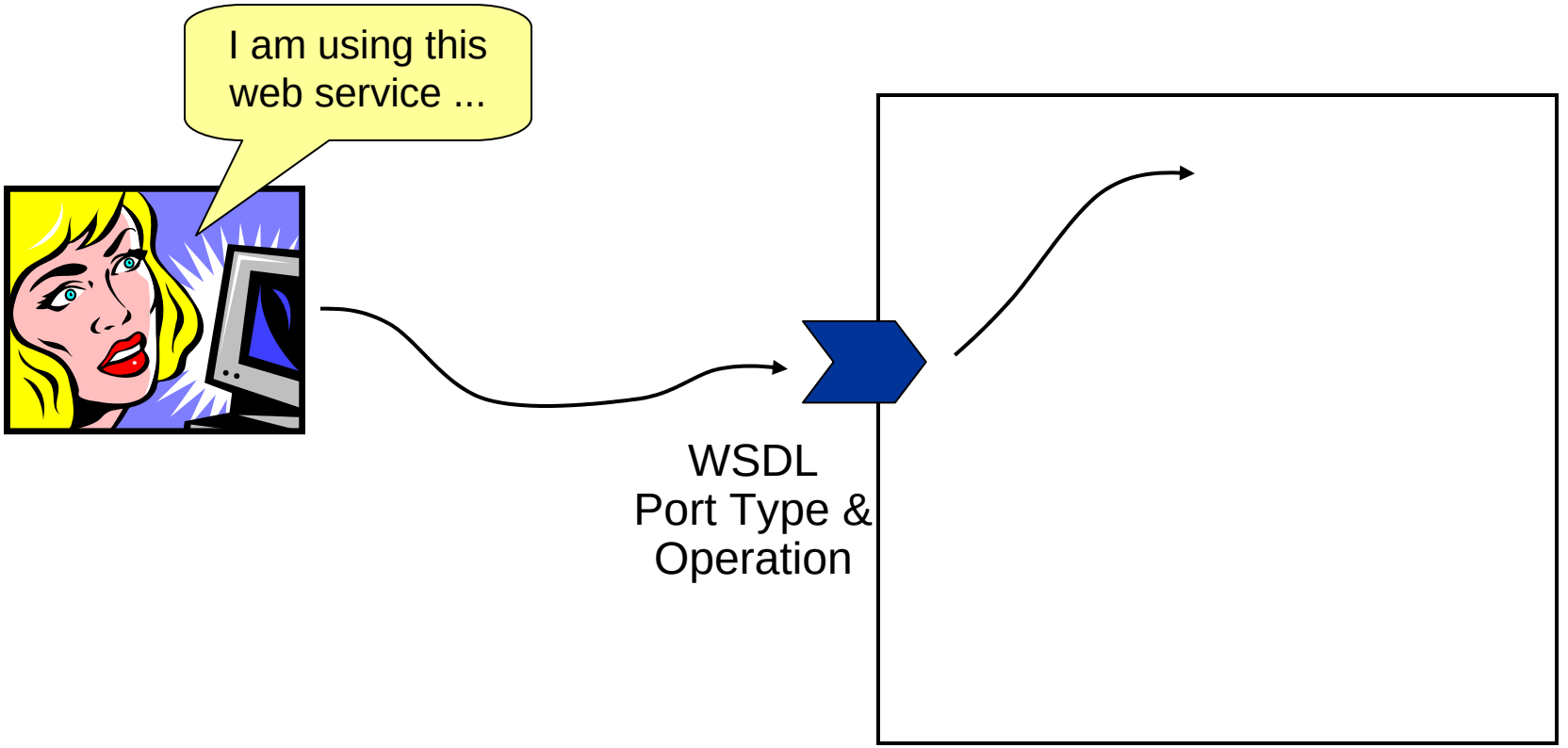


Sample Business Process

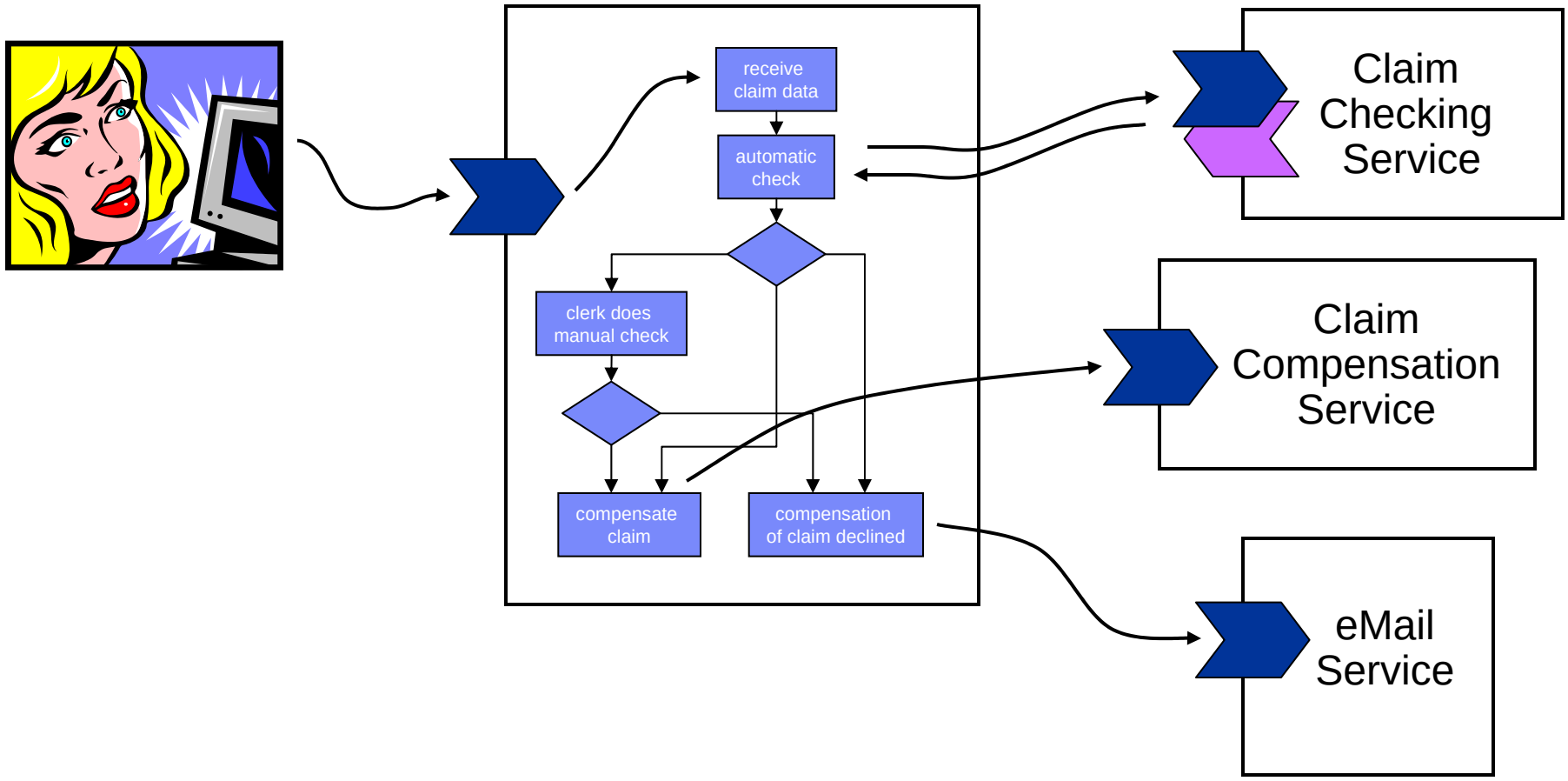
- **Processing of a claim in an insurance company**



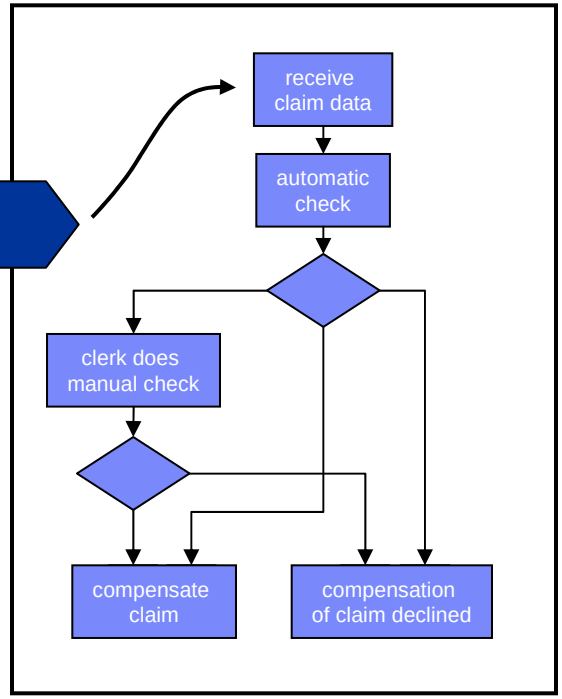
Business Processes as Web Services

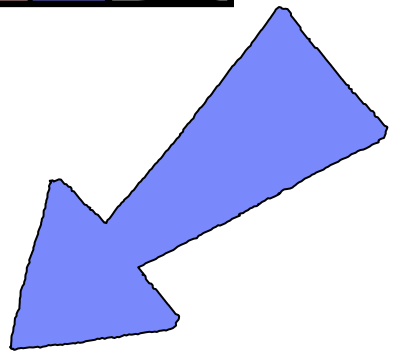


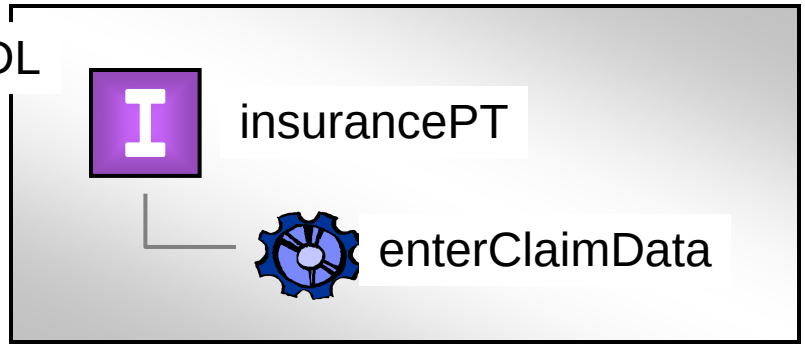
Sample Business Process



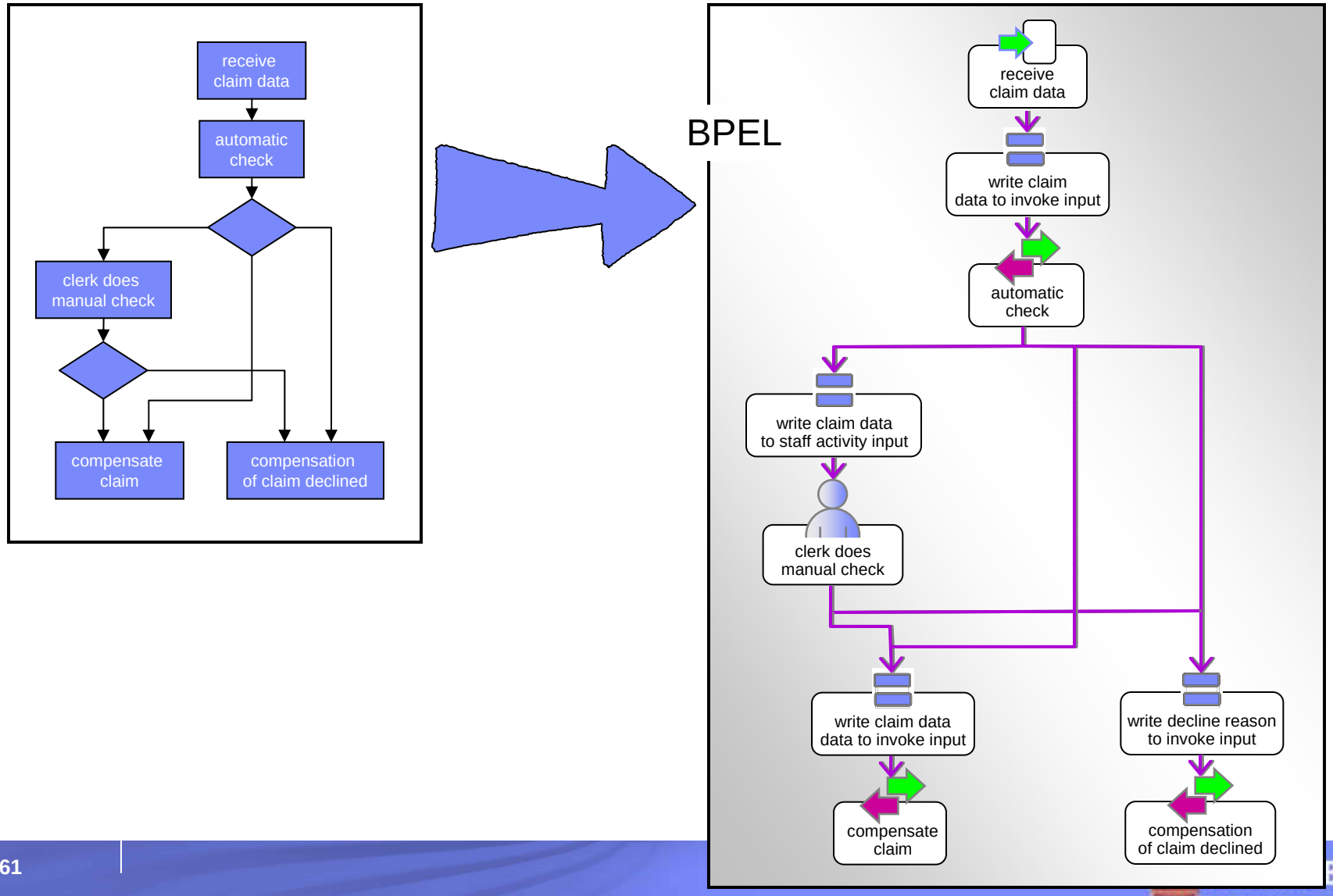
Create Artifacts – WSDL File







Create Artifacts – BPEL File

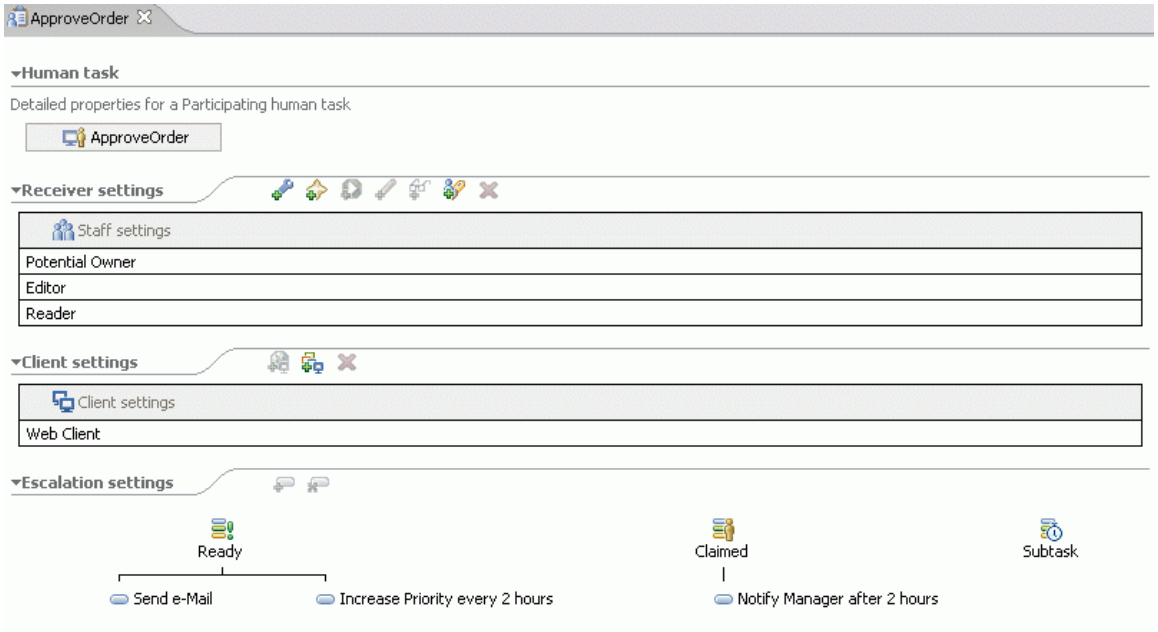


Human Task Manager – Human Tasks

Human Tasks

- **A Standalone Component**
  - Not restricted to just invocation from WS-BPEL Processes
- **Three kinds of Human Tasks**
  - Machine to Human (participating)
    - Invoke humans as a service
    - Component creates a work item for Human interaction (WS-BPEL)
  - Human to Machine (originating)
    - Human interfaces to service
    - Human interaction invokes a Component (i.e. Business State Machine)
  - Human to Human (pure human task)
    - Ad-hoc usage of To-Do list
    - Human interaction invokes a Component which creates a work item for another Human
- **Human Task Components**
  - Implement WSDL interfaces
  - Are implemented as SCA Components
  - Fit the overall SOA Model

Human Task Features



- **Different, rich assignment rules for Editors, Readers, Administrators and Potential Owners**
- **Multiple Clients**
  - Web Client
  - Portal Client
- **Multi-level Escalation Mechanisms**
  - e-Mail
  - Staff Assignment -> Notification Work Item
  - Priority Aging

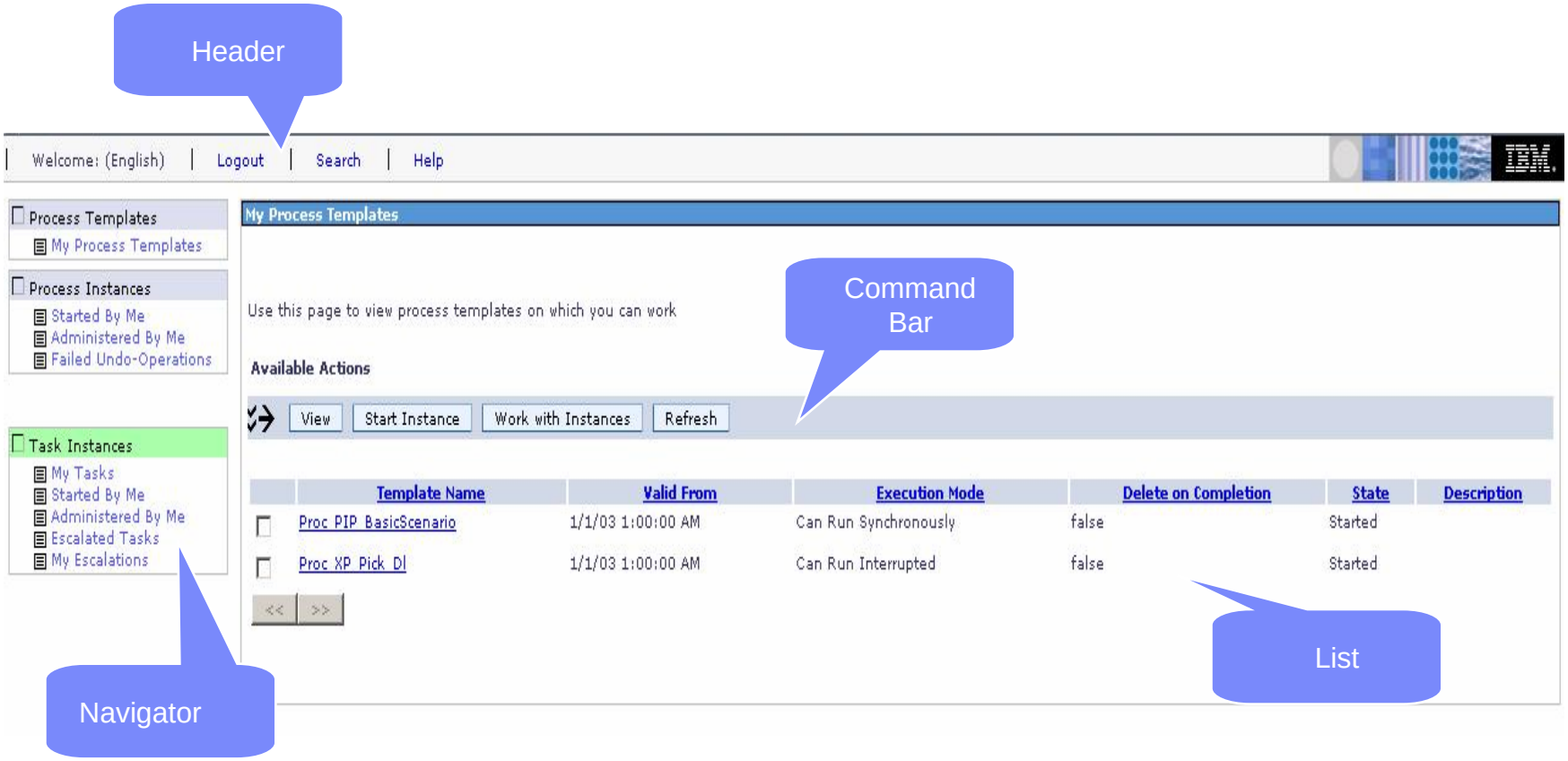
Escalation and Notification

- **Used to handle overdue tasks**
  - Create notifications if a task's progress is behind the expectations
  - Send notification based on time and task state
- **Possible ways to send notifications:**
  - *work items* for a set of users
  - *e-mail* notification
  - sending a notification *event* to a registered consumer (via callback)
- **Escalation structure**
  - One or many escalations per task
  - Single or chained

BPC Explorer – Support for Tasks

- **Execution and administration of Tasks.**
  - Claim / Save / Complete tasks
  - Suspend / resume / terminate tasks
  - Transfer / create / delete work items
- **List of Task templates – “Task Templates”**
  - Tasks that I am allowed to initiate/trigger
- **List of Task instances – “My Tasks”**
  - Tasks to be done by me.
- **Escalations for overdue tasks**

BPC Explorer – JSF Components

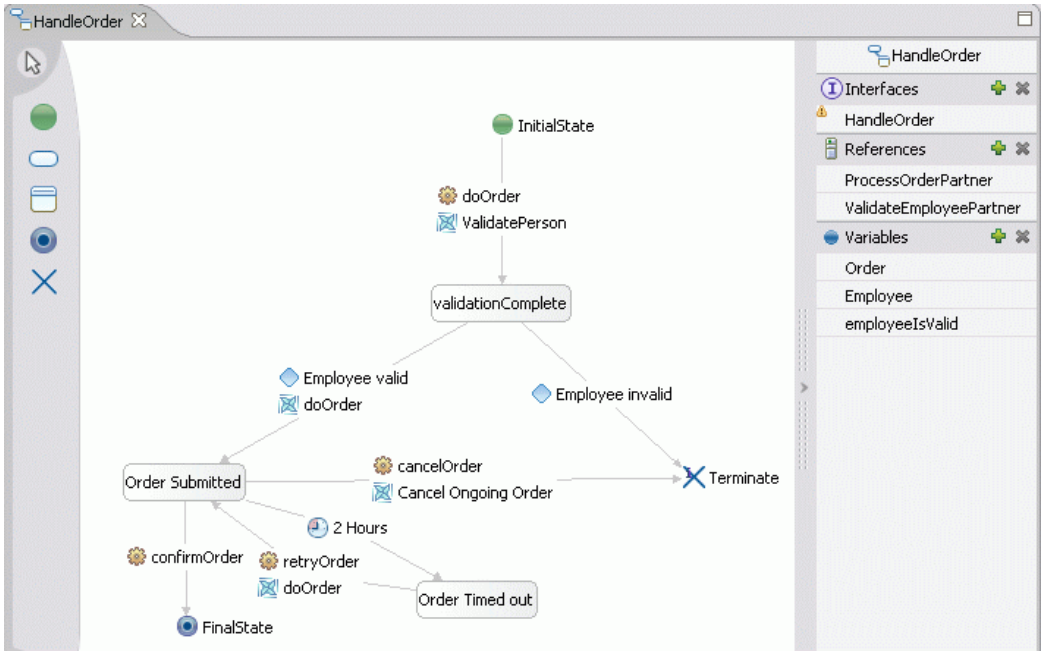


Business State Machines

Business State Machines

A ‘Business State Machine’ is an implementation of a business model that ‘executes’; moving from one state to another state based on real-time events.

- State Machine Implementation
  - Based on UML 2.0 State Machine Models
  - Event driven business processes
  - Creates WS-BPEL under the covers
- Simple/Complex States
  - Entry/Exit
- Transitions
  - Events
  - Actions (invokes)
  - Guards
  - Timeout



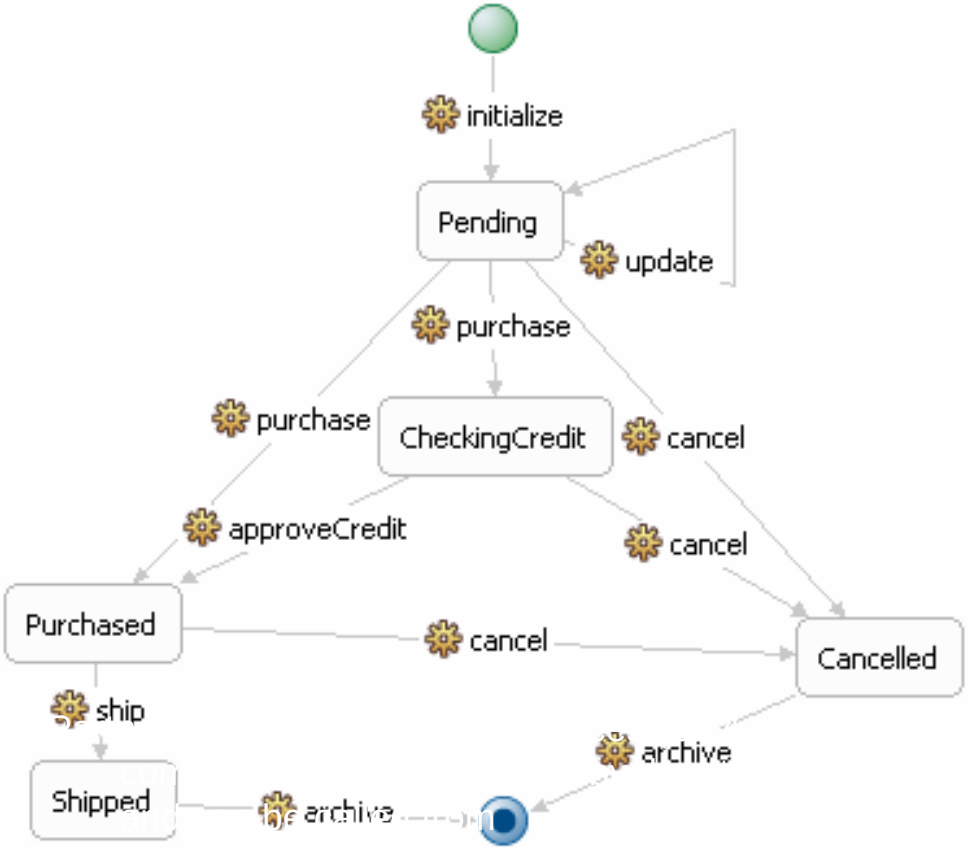
When to use a Business State Machine

- **Use a Business State Machine when:**
  - the business process is heavily event-driven
  - the reaction to these events is dependent on the process state
  - the process may revert to prior states
  - some sequential steps ok

- **Use a BPEL process when:**
  - steps in a process tend to happen in sequence
  - some event handling and looping is ok

Business State Machine

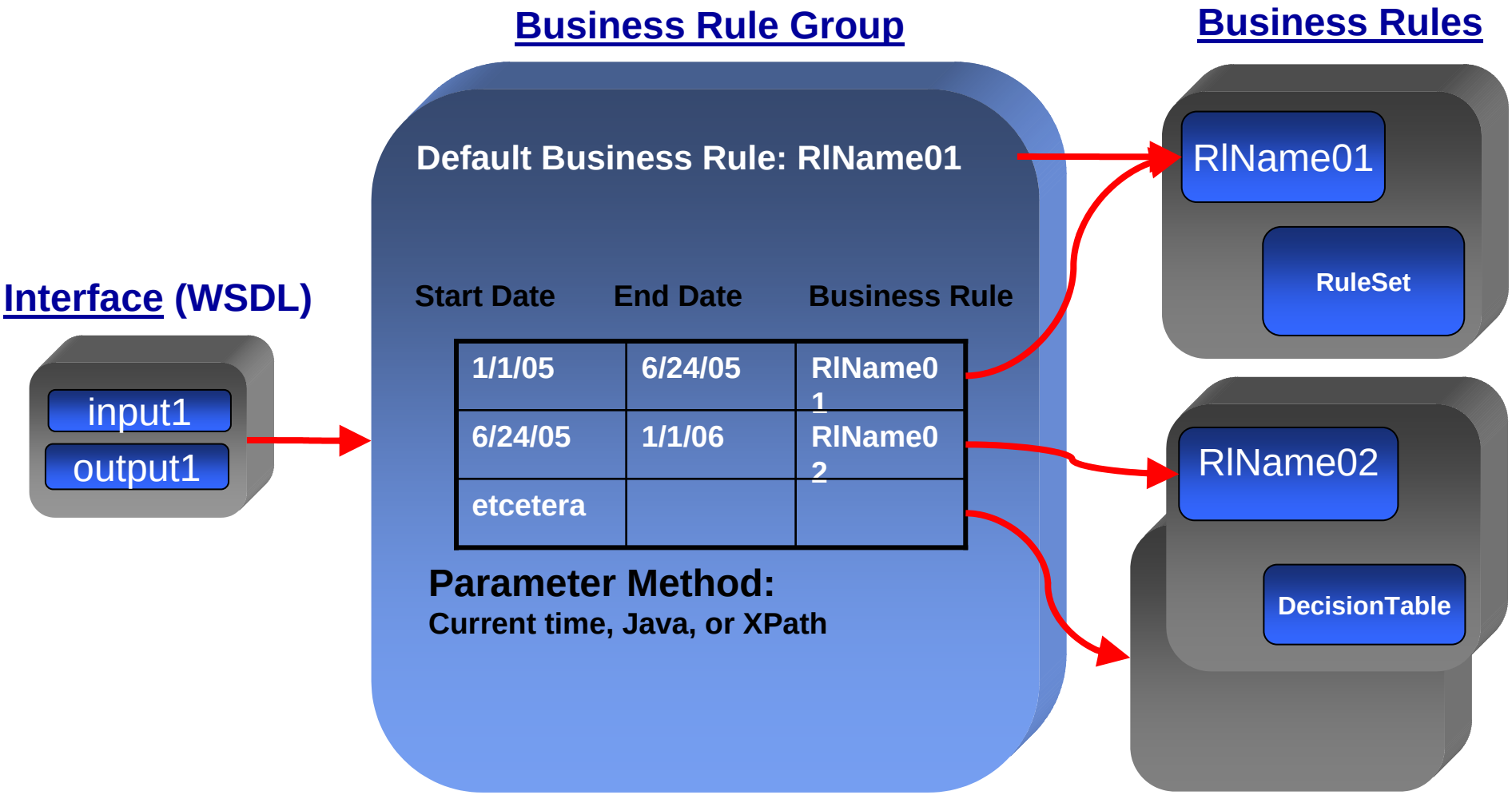
- **States and state transitions frame the process**
- **Logic embedded in the transitions**
- **Based on UML 2.0 State Machine**
- **Basic Pattern/Execution:**
  - Use input parameter for ‘correlation’ to get proper instance and current ‘state’
  - If the **event** isn’t supported by the current state, throw an exception.
  - For each transition that supports the event, check the **guards** (if specified) for a result of ‘true’
  - Process the state exit **action** (if specified)
  - Process the **transition**’s action (if specified)
  - Change the **state**
  - Process the state entry action (if specified)
  - Check for any automatic transitions out of the new state and repeat, or wait for next event



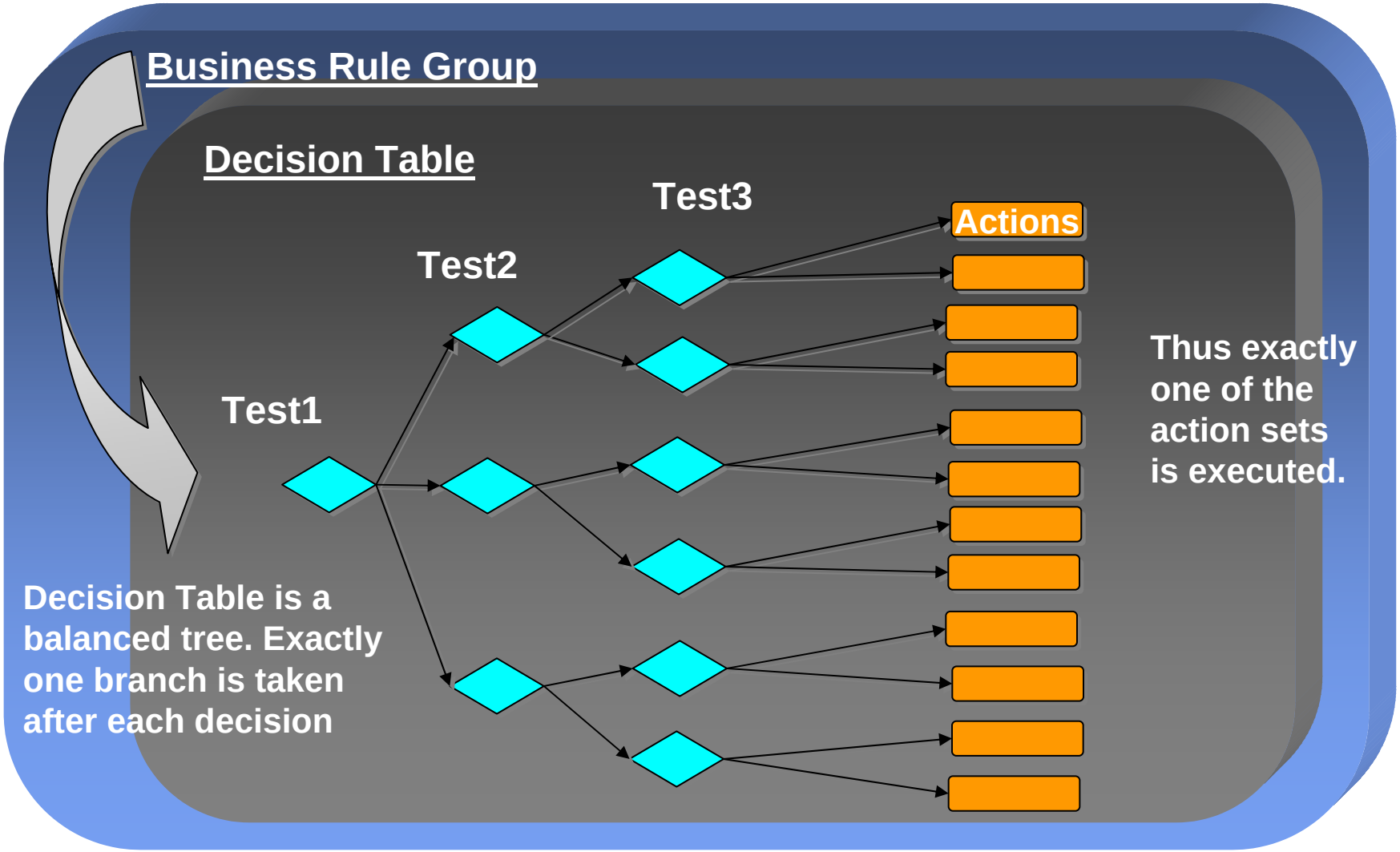
Business Rules

- **Externalize Business Logic from an application (business process)**
  - Easy change of logic that may change
- **Dynamically Update Rules in Runtime on the fly through Web Interface**
  - NLS enabled free text representation for rules
- **Most-requested Business Rule Functionality**
  - Decision Tables
  - Rule Sets (If/Then Rules)
  - Rule Templates
  - Action Rules
- **Ease of Use**
  - Rule Group: detailed implementation encapsulated in a component with a well defined interface

Business Rules – Logical Organization



Business Rules – Decision Table



Business Rules – Decision Table

Business Rule Group

Decision Table

Conditions		
input1.creditScore	input1.requestAccountAmount	output1.responseCode
>500	>50000	"REQ"
	>10000	"APP"
	>=0	"APP"
>300	>50000	"REQ"
	>10000	"REQ"
	>=0	"APP"
>=0	>50000	"REJ"
	>10000	"REJ"
	>=0	"REQ"

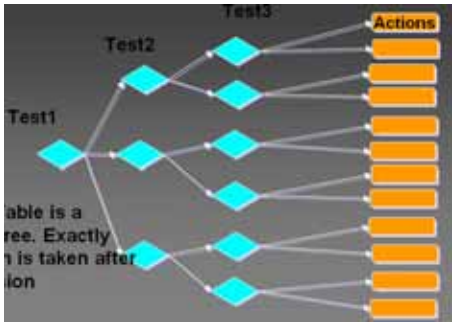
Business Rules – Decision Table

Business Rule Group

Decision Table

Conditions			
input1.creditScore	>500	>300	>=0
input1.requestAccountAmount	output1.responseCode	output1.responseCode	output1.responseCode
>50000	"REQ"	"REQ"	"REJ"
>10000	"APP"	"REQ"	"REJ"
>=0	"APP"	"APP"	"REQ"
			Actions

Business Rules – Decision Table - Summary

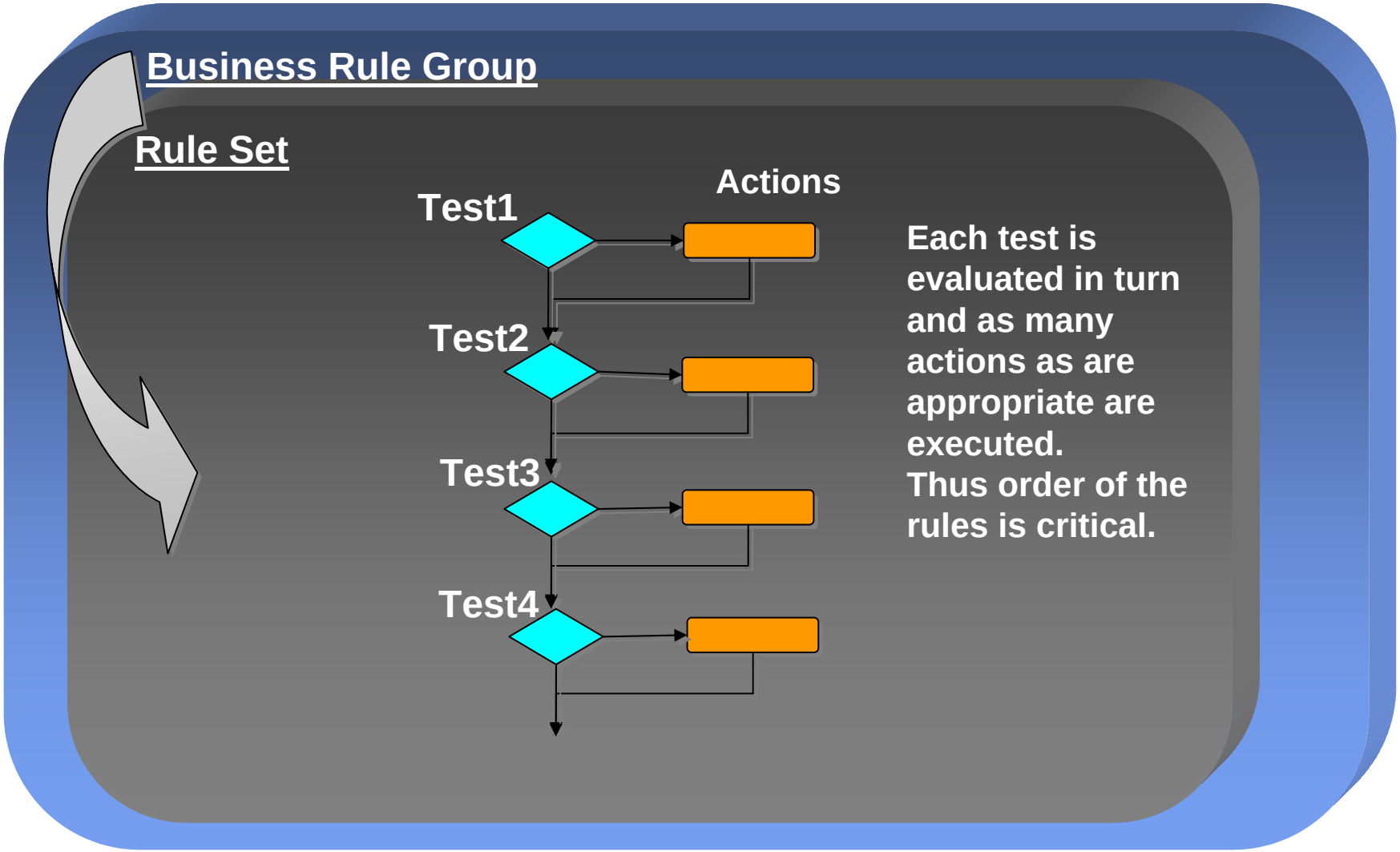


- **Decision Table represents a decision tree structure**

- **Each level of the tree is like a case (or select) statement.**
  - Exactly one branch is taken at each level (the first listed that evaluates true so order is important here too).
  - If no case statement is satisfied at any level an exception occurs.

- **The number of "total cases" is the product of the number of cases at each level.**
  - So Level1 choice of 3, level2 choice of 2, level3 choice of 4 would give 3*2*4=24 cases in total.
- **Only one set of actions will be fired.**
  - The actions often consist of setting values for elements in the output. However more complex actions involving other SCA components are possible.

Business Rules – Rule Set



Business Rules – Rule Set

Business Rule Group

Name	Rule4
Presentation	One of the Final rules - all template rules should come before this
If	all of the following are true ● creditScoreNeeded>0 ● input1.creditScore>=(creditScoreNeeded*0.9)
Then	output1.responseCode="REQ" output1.responseReason="Request Approval - Cretid score within 10% of auto approval target"

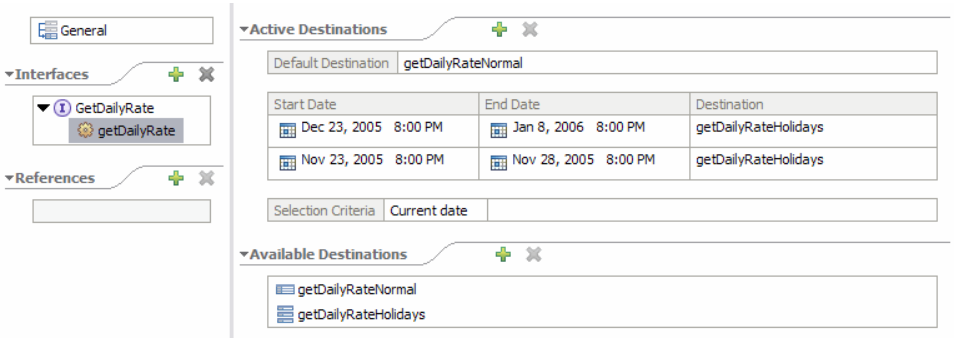
Name	Rule5
Presentation	One of the final rules - all template rules should come before this.
If	all of the following are true ● creditScoreNeeded>0 ● input1.creditScore>=creditScoreNeeded
Then	output1.responseCode="APP" output1.responseReason="Automatic Approval"

Business Rules – Rule Set - Summary

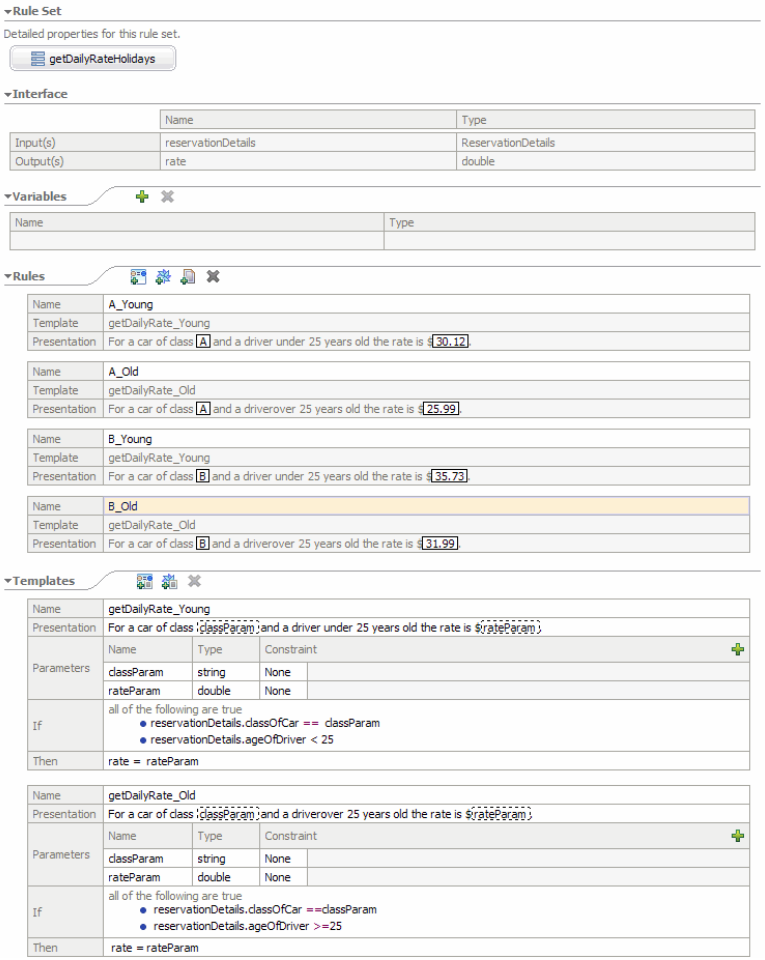
- **Rule Set is a set of 1 or more if-then condition/action statements that are processed sequentially**
  - Evaluated from first listed to last, so order is important.
  - Able to choose any number of Action/If-Then conditions
  - An Action need not have an If condition associated. Simple actions are allowed, these are always executed. Example: Initializing variables.
- **Evaluates multiple conditions and can fire multiple rules**
  - All the actions that have conditions that evaluate to true are needs to be done
- **Local variables can be defined for “COMPLEX” processing**
  - These can be of any datatype.
- **Actions generally set values for output BOs,**
  - But can invoke other SCA modules.

Business Rules

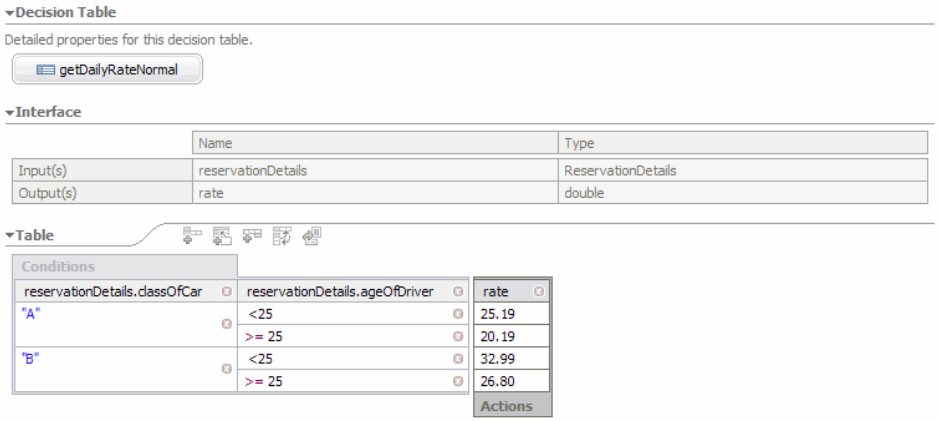
Rule Group



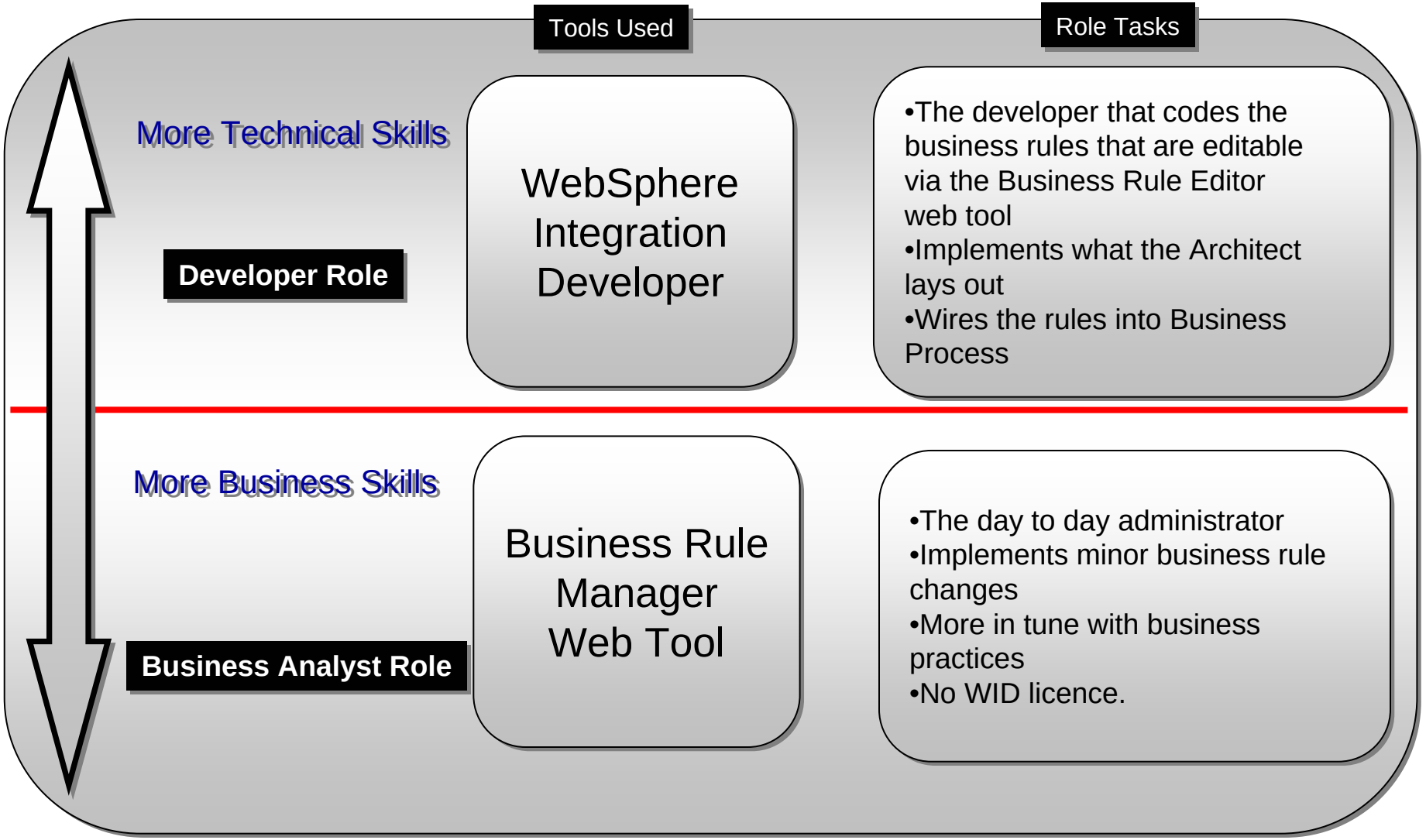
Rule Set



Decision Table



Business Rules – Roles - Tooling



Business Rules – Presentation Information

- **Enables rules to be displayed in the web based tooling with a more natural language view**
  - if invoice.purchase() >= 100.00 then discount = .05
  - When the customer purchase is $ 100 or more then give the customer a discount of 5 percent.
- **Defined by the application developer**
- **Can be translated**

Tooling – Templates - WID

- **As well as using WID to create the rules and rule groups we have discussed, WID can be used to create paramatized rule templates. From these templates new rule instances can be created in WID or the**

Templates

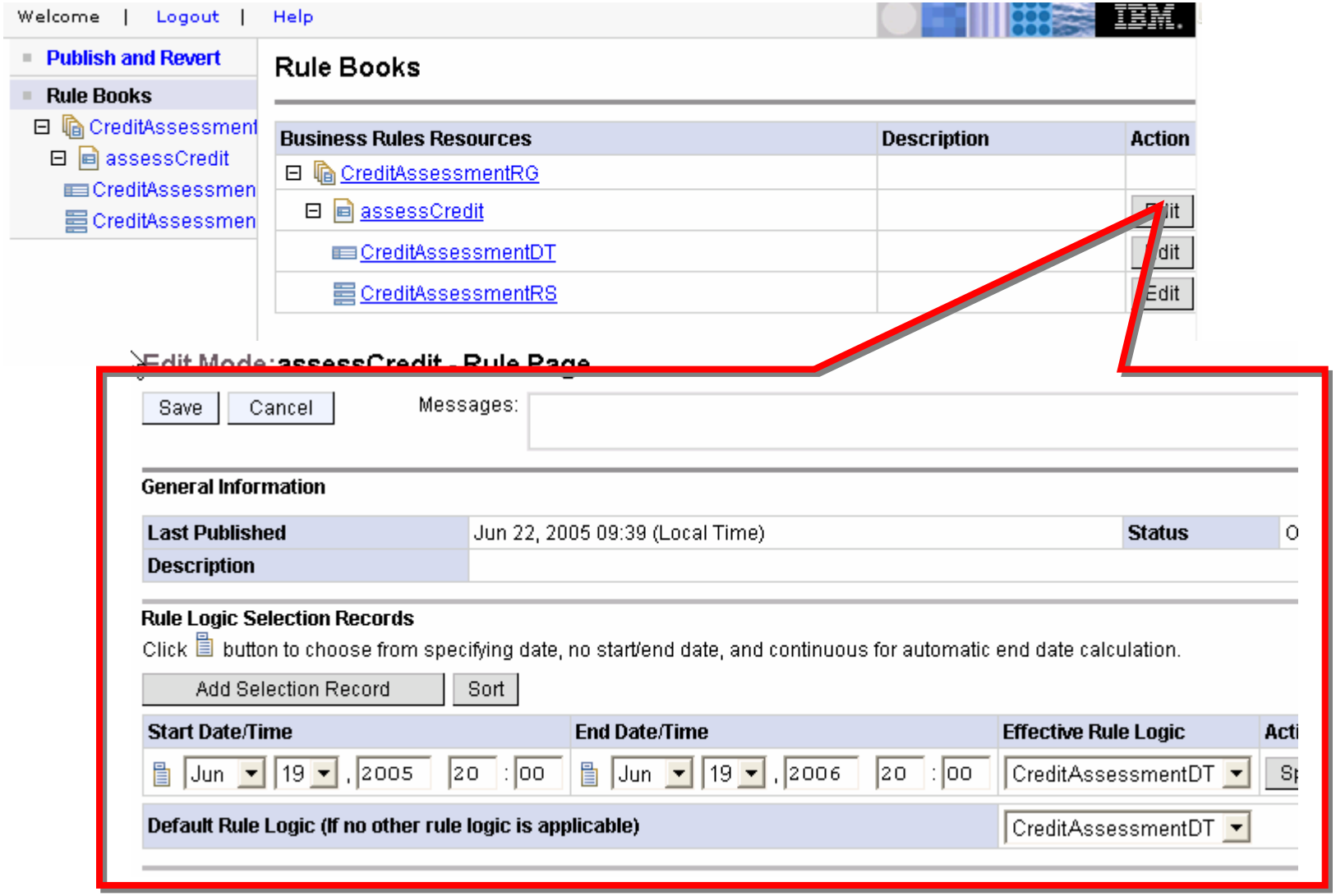
TEMPLATE

Name	Template 1			
Presentation	For Country pCountry an account greater than pAccountSize requires a credit score greateror equal to pCreditScore			
Parameters	Name	Type	Constraint	
	pCountry	string	None	
	pAccountSize	int	None	
	pCreditScore	int	None	
If	all of the following are true ● input1.customerCountry==pCountry ● input1.requestAccountAmount>creditScoreNeeded ● creditScoreNeeded<pCreditScore			
Then	creditScoreNeeded=pCreditScore			

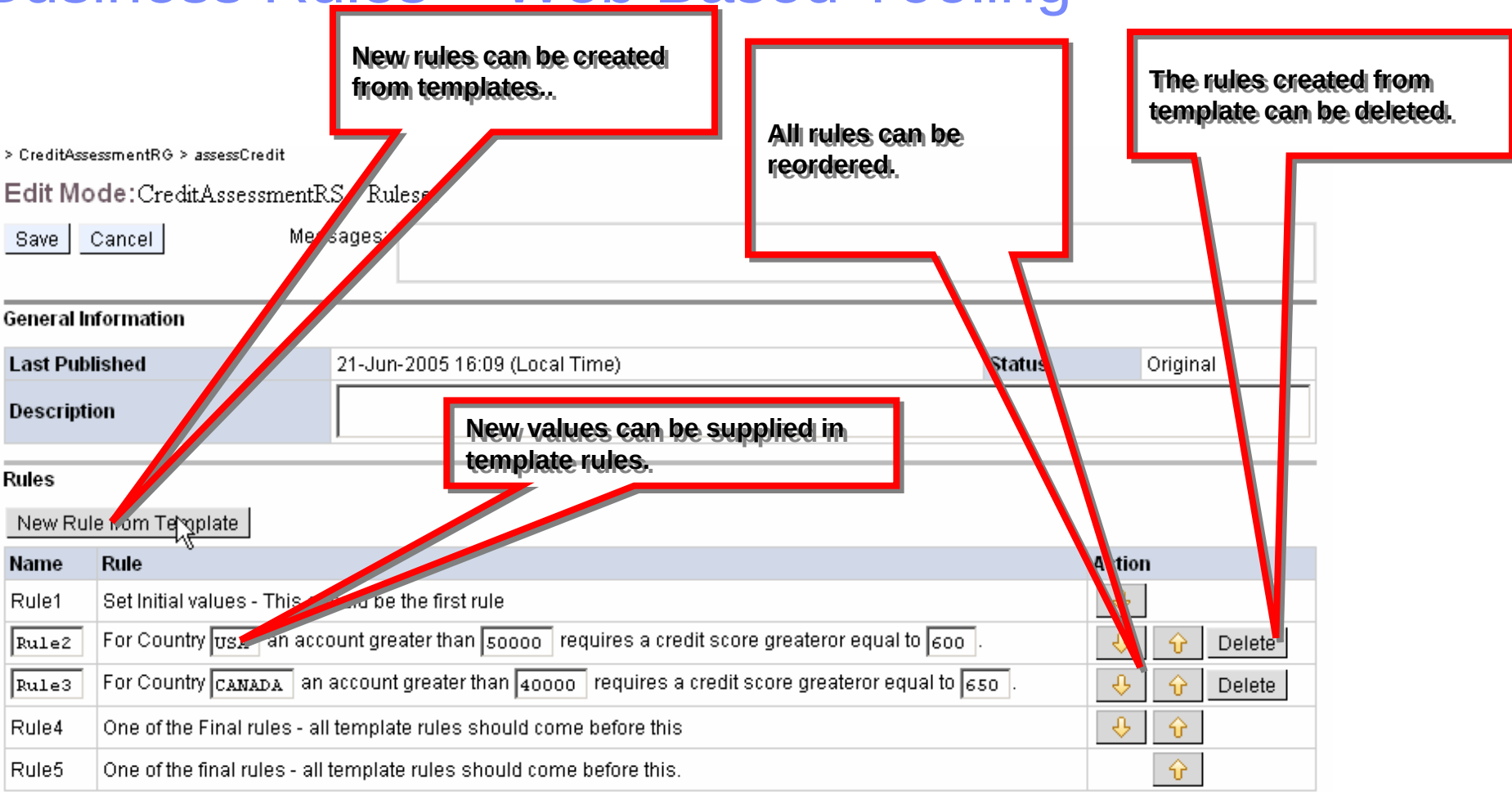
RULE from TEMPLATE

Name	Rule2
Template	Template 1
Presentation	For Country USA an account greater than 50000 requires a credit score greateror equal to 600.

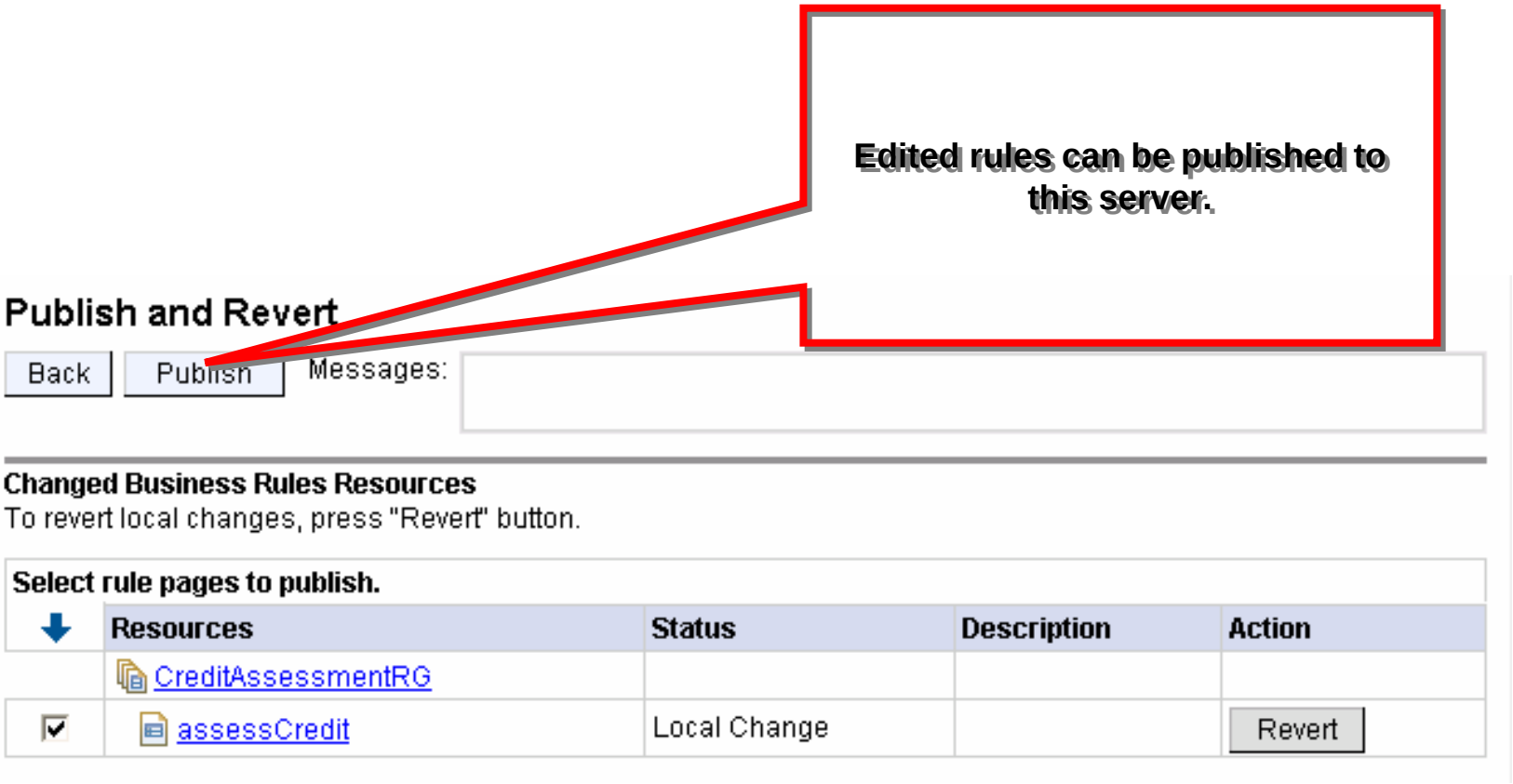
Business Rules – Web Based Tooling



Business Rules – Web Based Tooling



Business Rules – Web Based Tooling

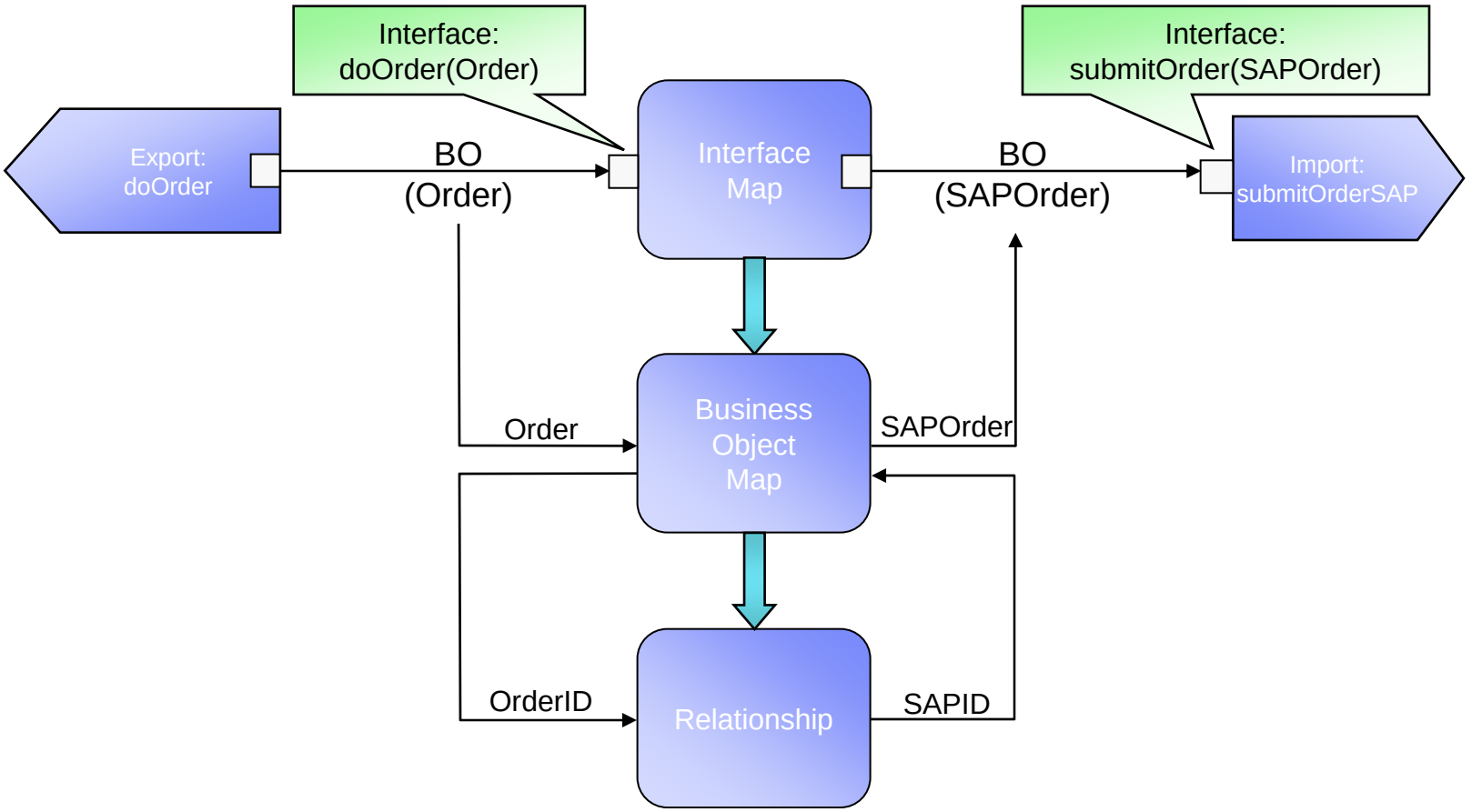


Supporting Services – Transformation summary

- **Business Object Maps**
  - Translate one Business Object into another
- **Relationships**
  - Maintain Key Relationships for Business Data
- **Interface Maps**
  - Translate one interface into another
  - For Interfaces semantically identical but not syntactically
- **Selector**
  - Invoke different component based on time
  - All components have the same interface
- **Mediation**
  - Implementation of mediation logic

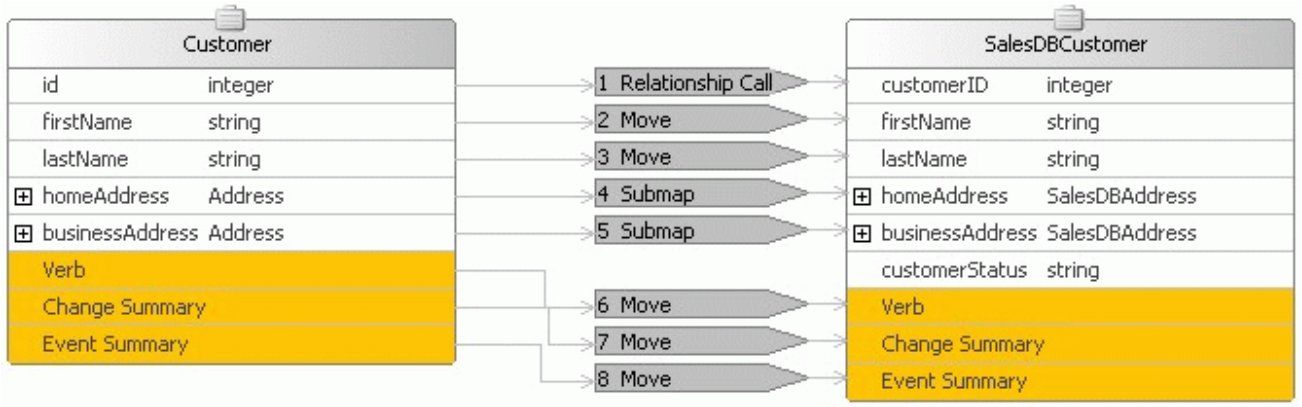
Transformation Components

Interface Maps | Business Object Maps | Relation-ships

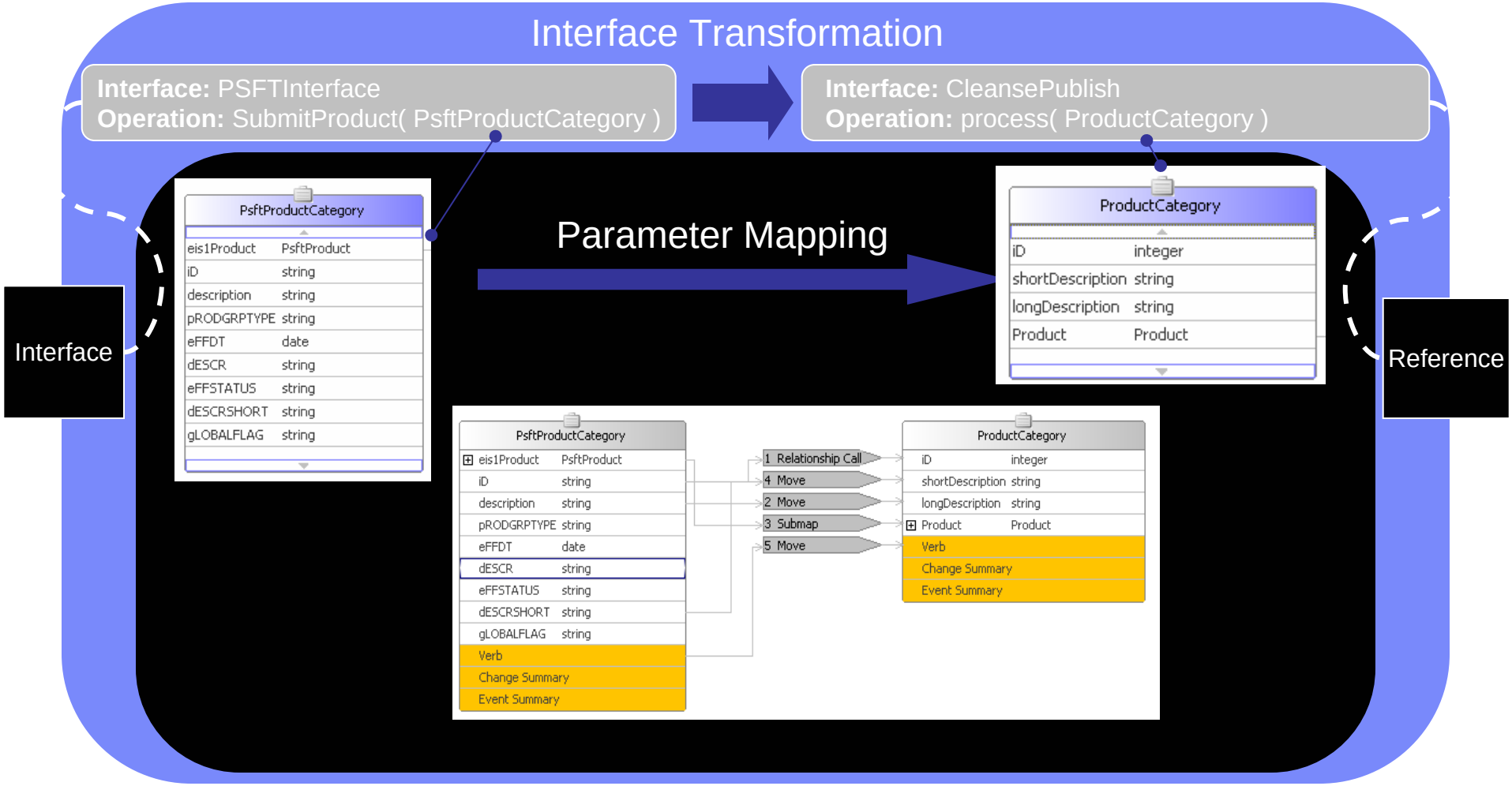


Transformation – Maps

- **Map Service is a 'system' Service**
- **Graphically convert one Business Object into another**
- **Mapping Functions**
  - Move, Extract , Join
  - Relationship Call
  - Assign
  - Submap
  - Custom
    - Activity Editor
    - Java Code



Interface Transformation as a SCA component



Transformation – Relationship Service

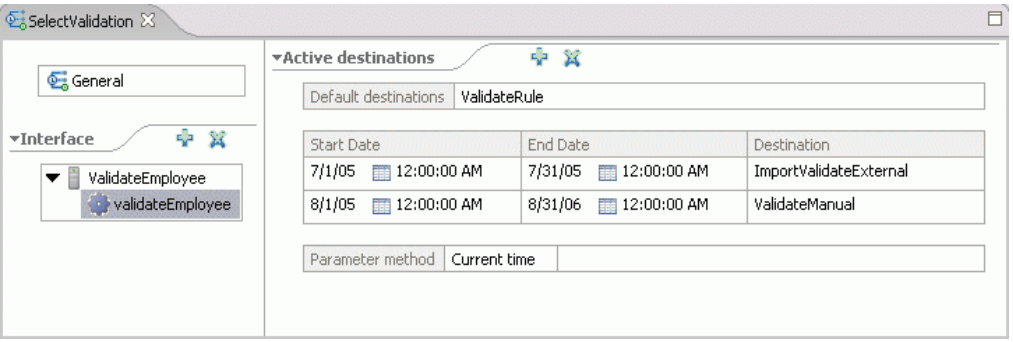
- **Relationship Service is a ‘system’ Service**
- **Relationship Data stored in a database**

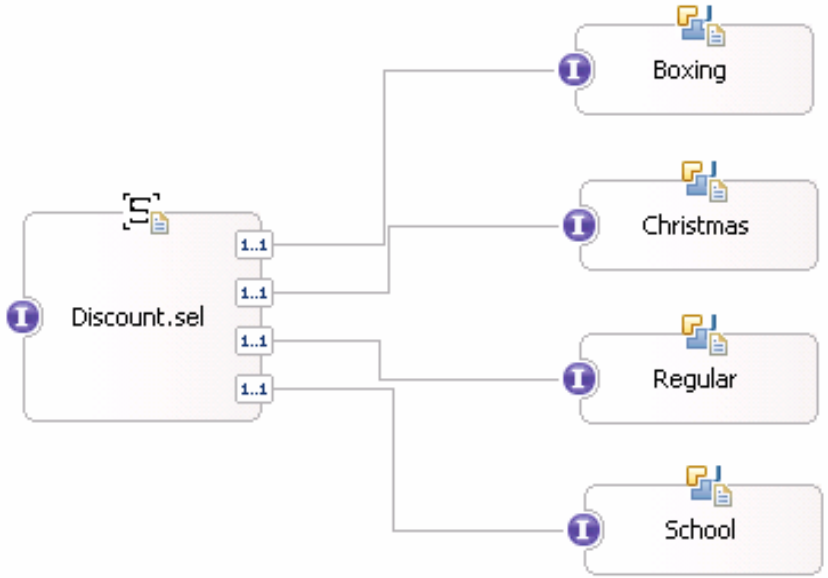
- **Two types of Relationships:**
  - Identity Relationships
    - Customer ID, Order ID, …
  - Lookup Relationship
    - 1 = AL = Alabama, 2 = AK = Arkansas, 3 = AR = Arizona, …
    - 1 = Monday, 2 = Tuesday, 3 = Wednesday, …

Dynamic Service Selection

Selector

- **Client**
  - makes a call to the Selector Component
- **Selector Component**
  - chooses which target destination to invoke using a declared selection implementation
- **Destination(s)**
  - for each operation on the Selector Component are associated with the Selector Component
- **Web-based Administration**

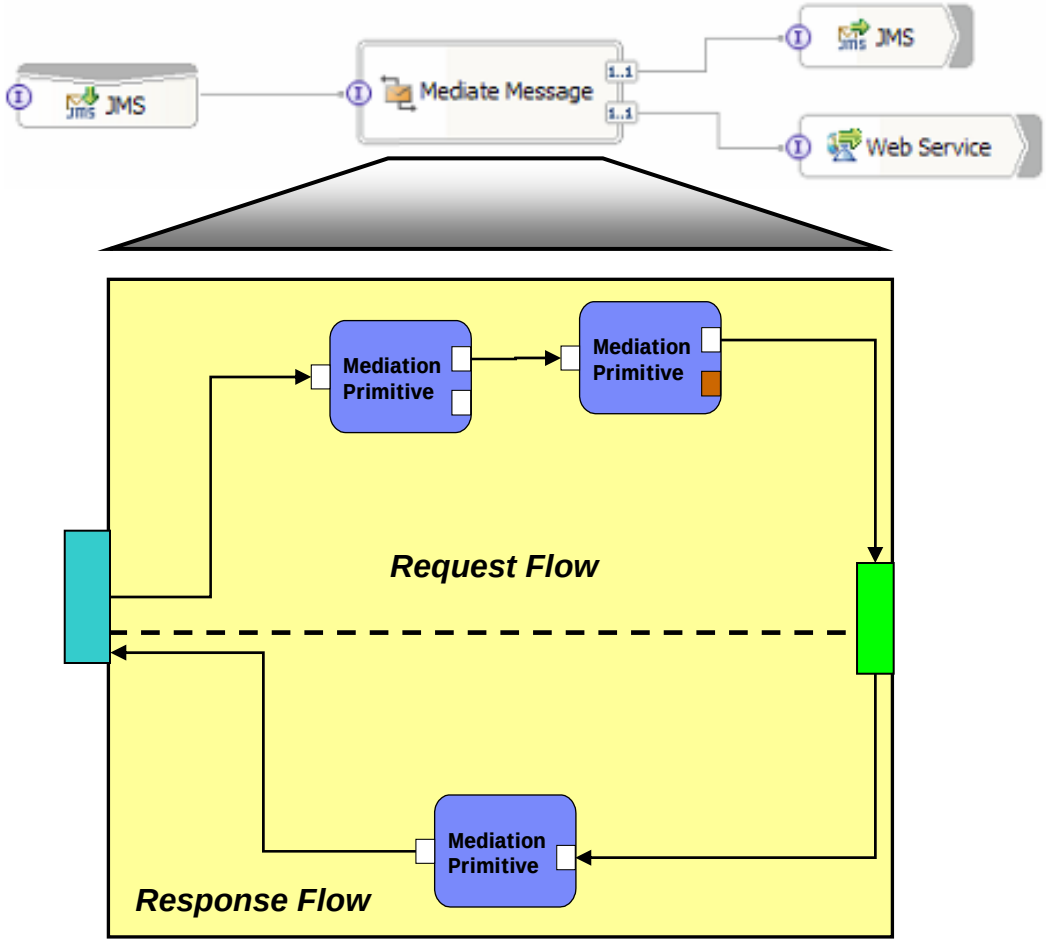




ESB Mediation Component

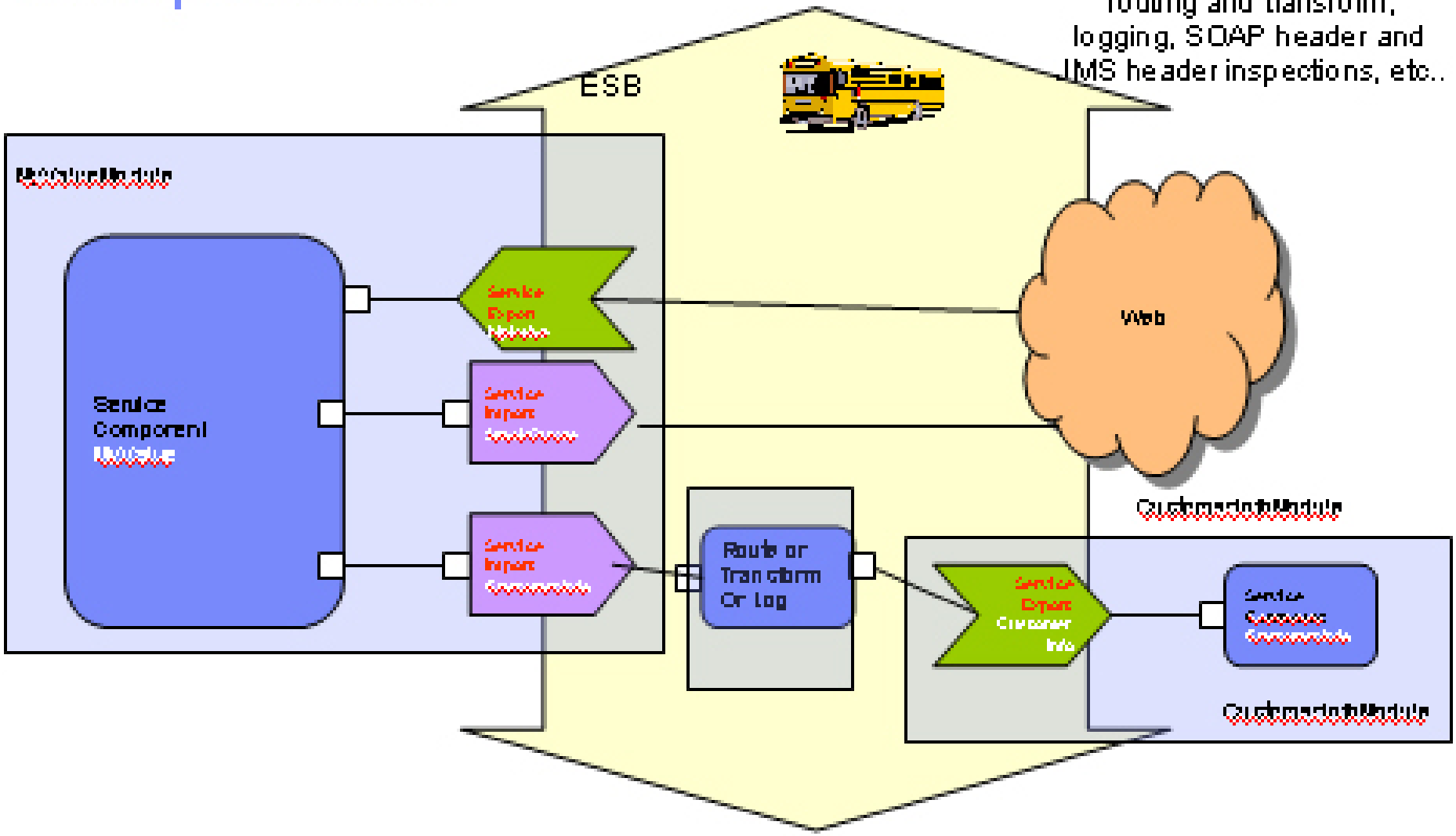
Mediation (ESB)

- **Provide the Implementation of mediation "logic"**
  - "flows" that operate on messages/events as they are processed by the system
  - Operate on both One-Way and Request-Response interactions
- **Pre-Supplied primitives allow flows to be visually composed**
  - XSLT Transformation
  - Message Logger
  - Message Filter
  - Fail
  - Stop
  - Database Lookup
  - Custom (Java) Component
  - CEI Emitter (Post GA)



WebSphere ESB

Programming Model, Authoring and Admin/Config tools and Primitives to enable message routing and transform, logging, SOAP header and JMS header inspections, etc..



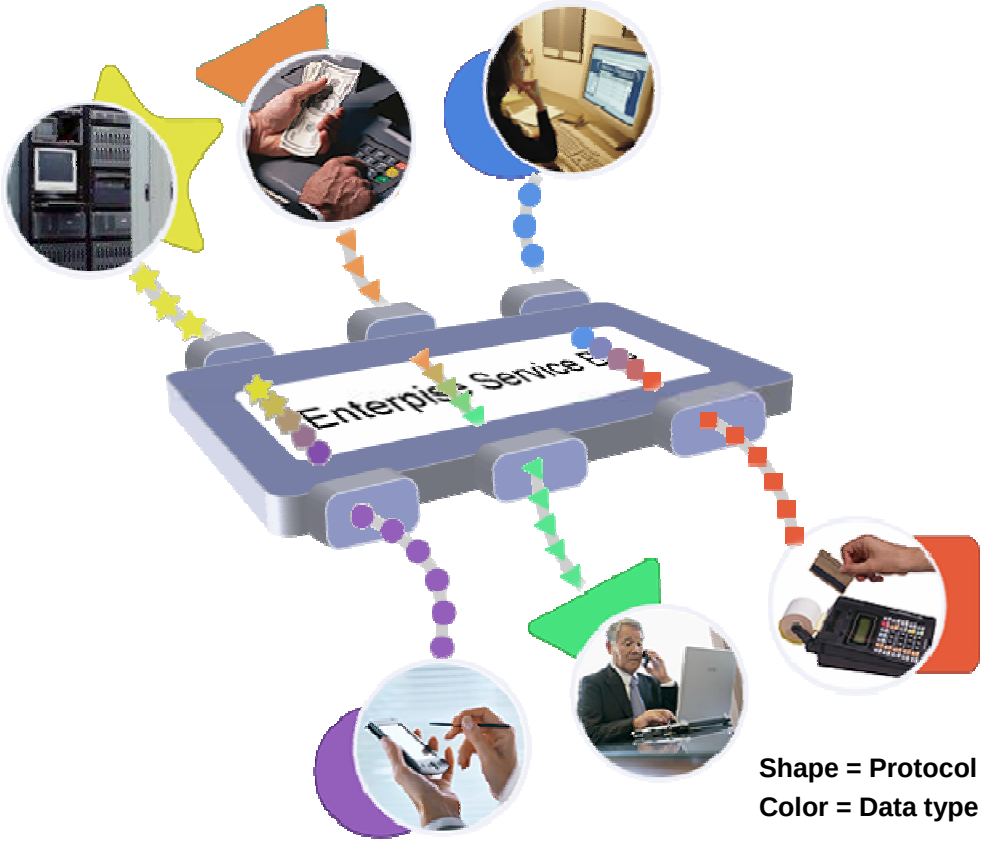
Common Connectivity: Enterprise Service Bus

An Enterprise Service Bus (ESB) is a flexible connectivity infrastructure for integrating applications and services.

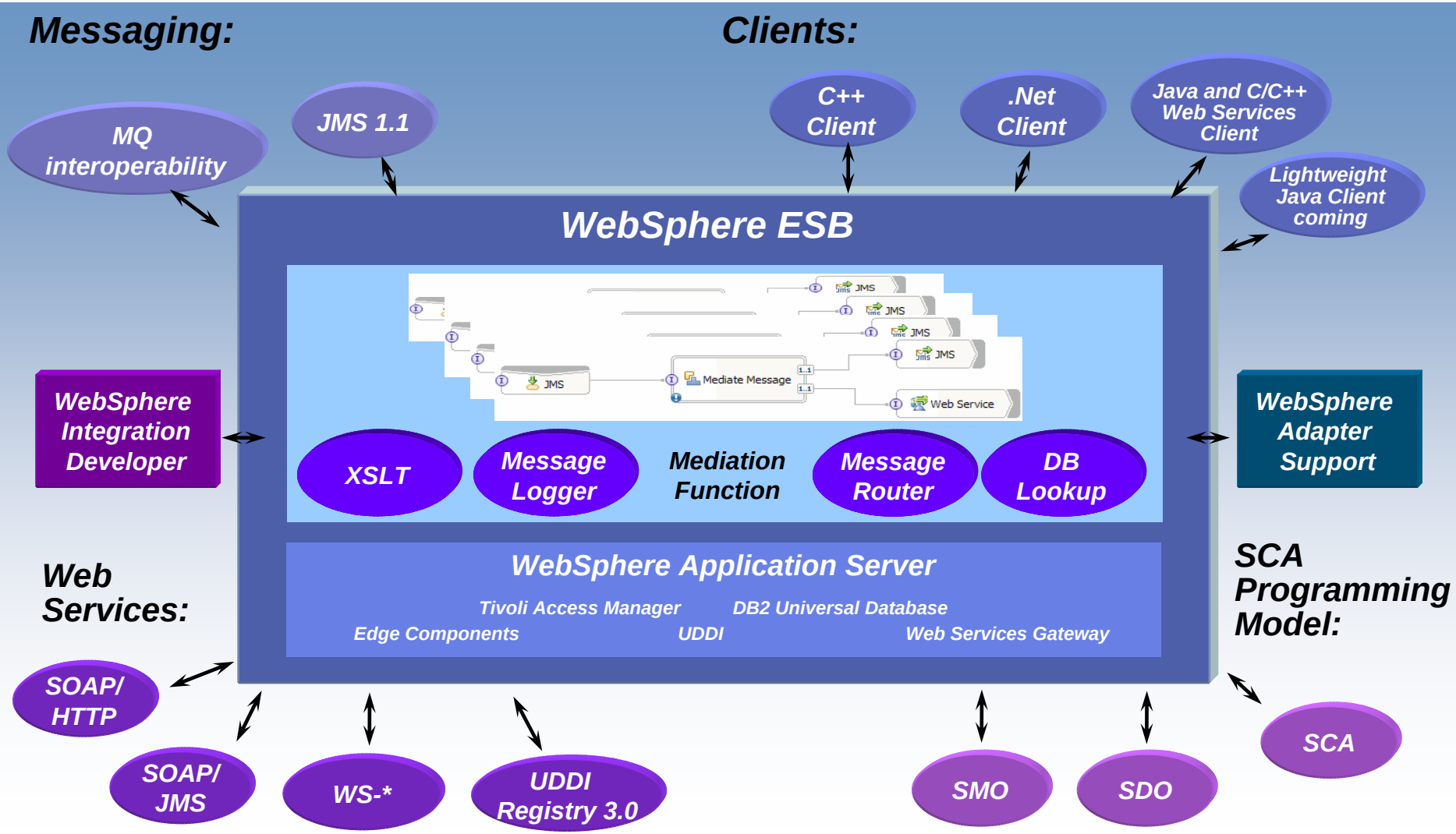
An ESB powers your SOA by reducing the number, size, and complexity of interfaces.

An ESB performs the following between requestor and service

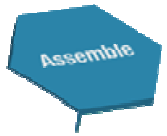
- **ROUTING** messages between services
- **CONVERTING** transport protocols between requestor and service
- **TRANSFORMING** message formats between requestor and service
- **HANDLING** business events from disparate sources

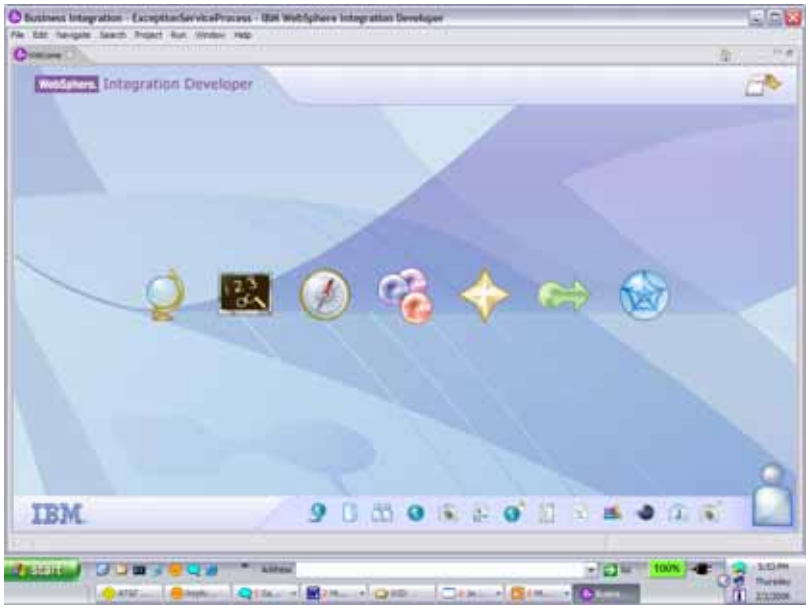


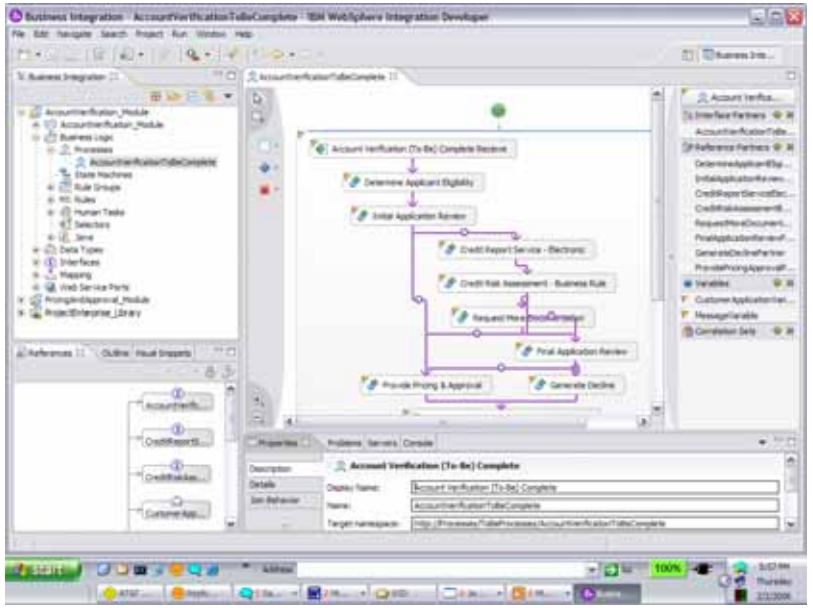
WebSphere ESB



WebSphere Integration Developer V6

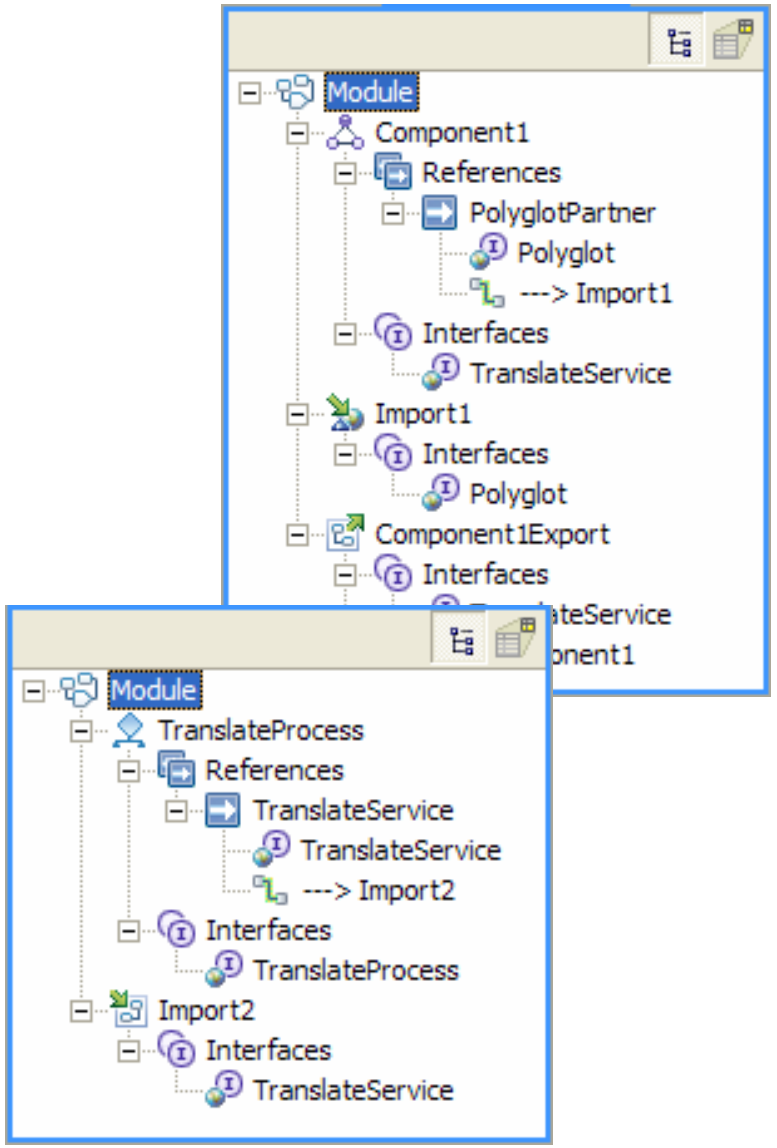




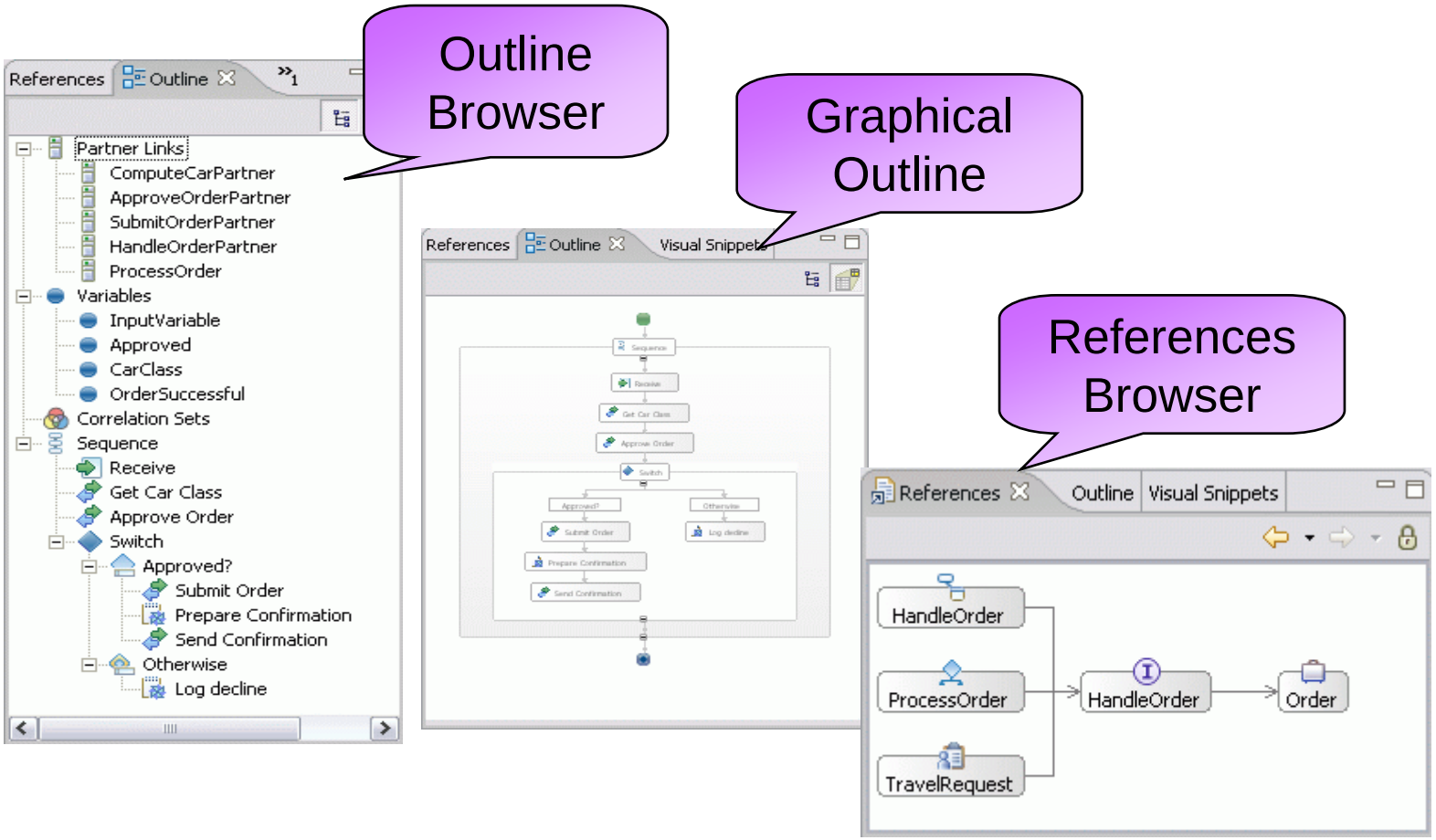


WebSphere Integration Developer V6

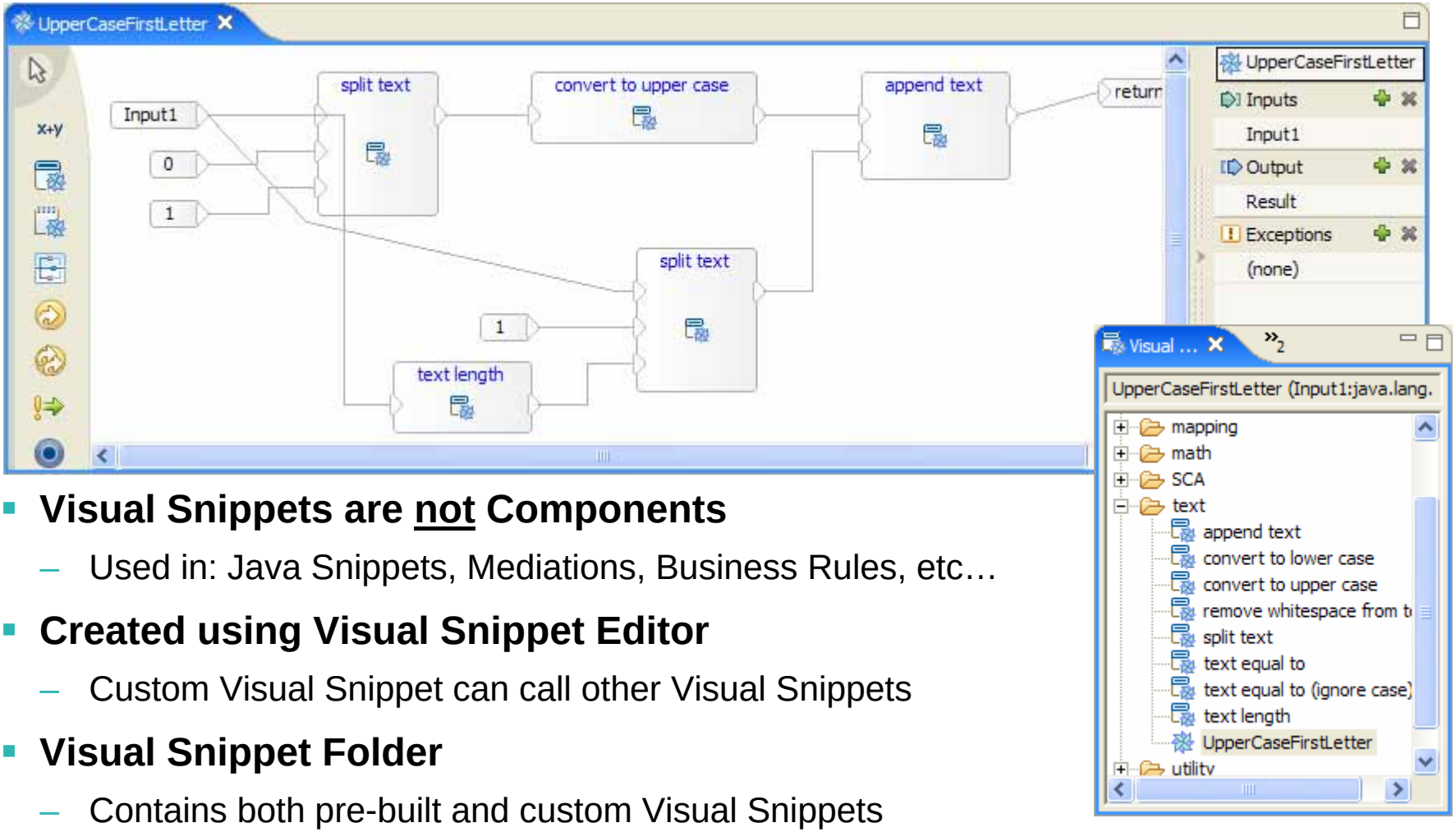
- Reduced complexity
  - Provide a layer of abstraction over the J2EE programming model
- Solution building blocks based on integration-level concepts and patterns
  - Process Choreography, Mediation, Relationships, Business Rules, etc…
- Not J2EE Artifacts
  - EJBs, RARs, EARs, WARs, etc..
- Application assembly
  - Enable solution assembly from components
- Test and debug: Integration test client and integration debuggers



Development Tool Navigation

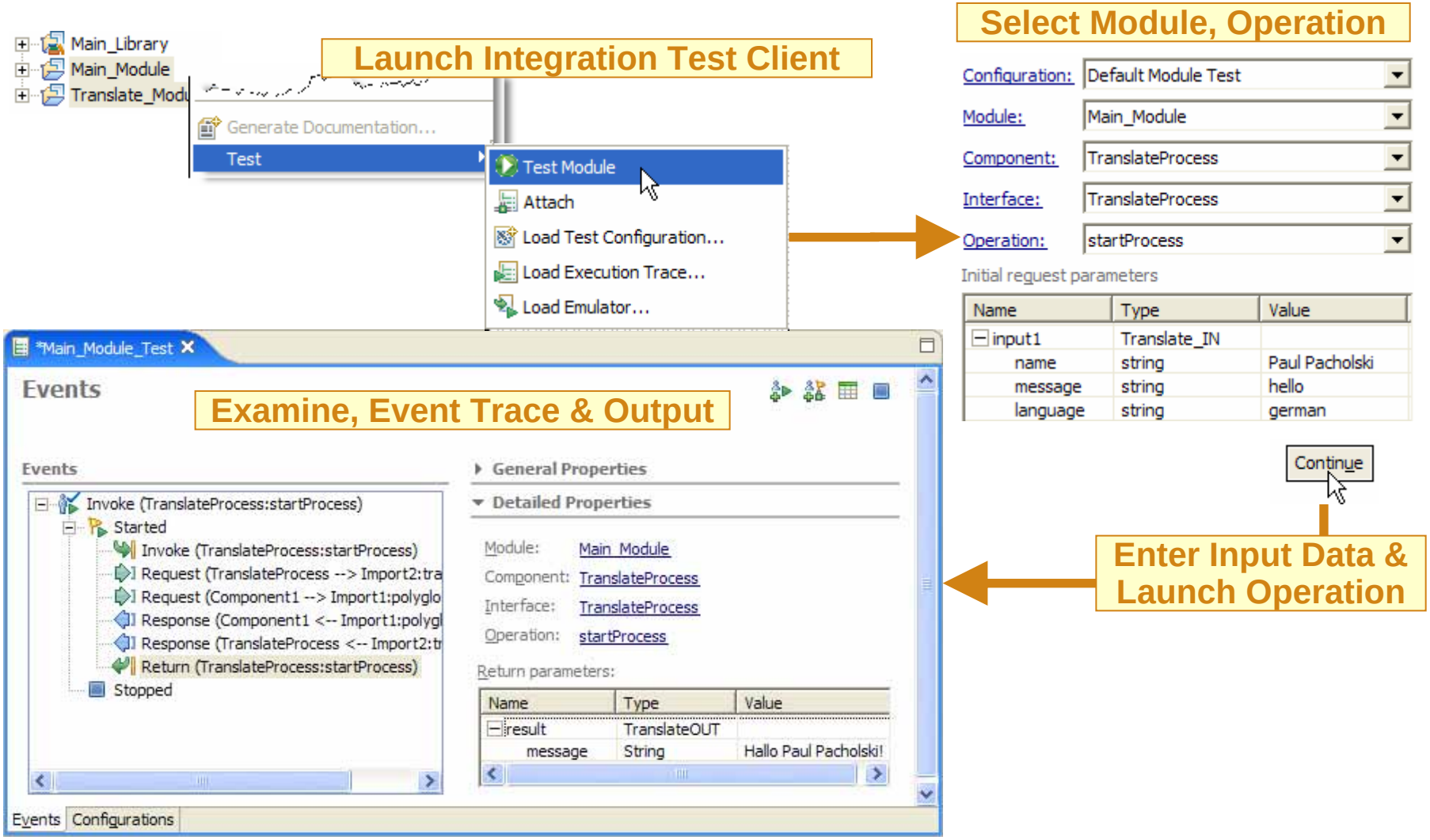


Visual Snippets



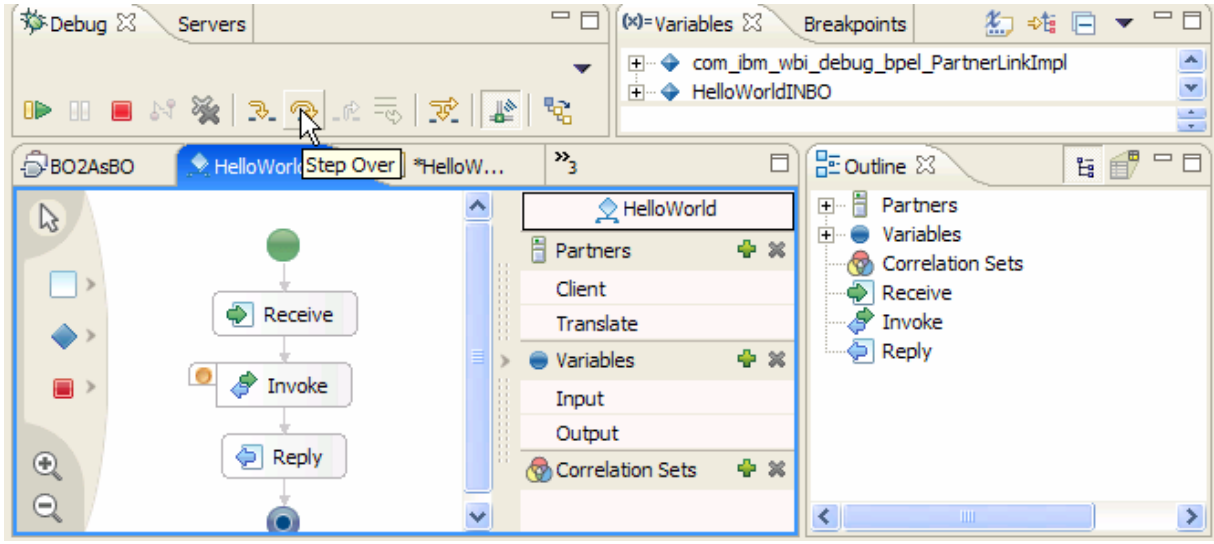
- **Visual Snippets are <u>not</u> Components**
  - Used in: Java Snippets, Mediations, Business Rules, etc…
- **Created using Visual Snippet Editor**
  - Custom Visual Snippet can call other Visual Snippets
- **Visual Snippet Folder**
  - Contains both pre-built and custom Visual Snippets

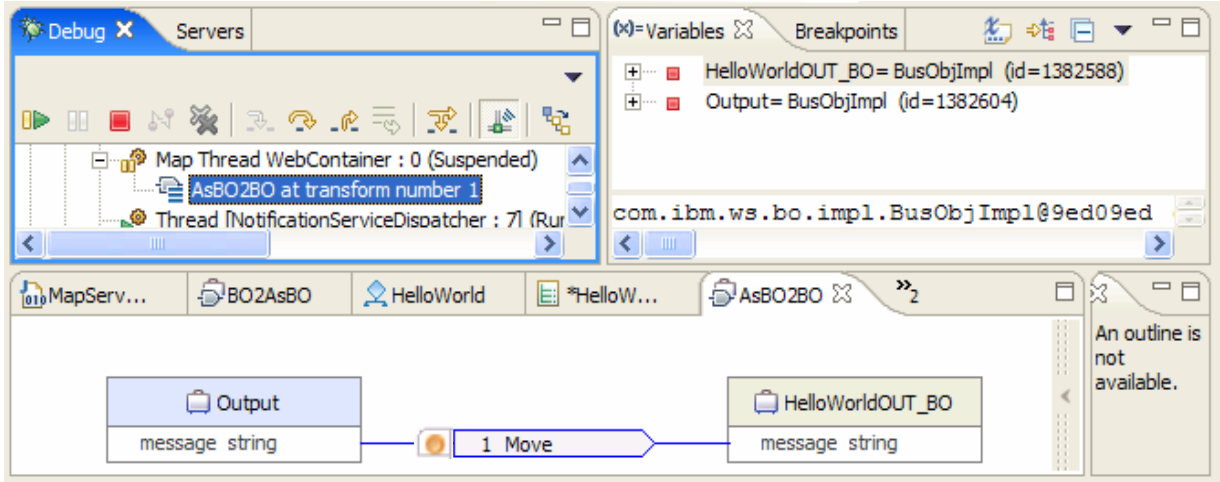
Test and Debug – Integration Test Client



Test and Debug – Integration Debuggers

- **Server must be started in the Debug Mode**
- **Debugger runs in the Debug Perspective**
- **Capabilities**
  - Set breakpoints in a component
  - Step through the component
  - Change the values of its variables
  - Step into source code





Business Driven Development – Roles and Tools …

Business Analyst

- Defines, models Processes
- Optimizes Processes through simulations

WebSphere BI Modeler

Solution Architect

- Defines business contract, business and system use cases
- Models Service Implementation

Rational Software Architect

Integration Developer

- Implements Processes and Composite Applications
- Defines Services

WebSphere Integration Developer

J2EE Developer

- Implements Services
- Constructs other J2EE artifacts

Rational Application Developer

Agenda

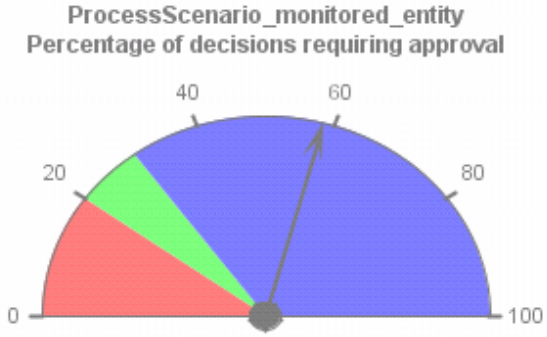
- **SOA & BPM Introduction**
- **Model**
- **Assemble & Deploy**
- **Manage**

Manage - View and modify your business in real time

- Report on how the business is performing as measured against defined objectives – a scorecard view implemented through Key Performance Indicators

- Track and modify business process flows
  - Eliminate redundancies or inefficiencies
  - Identify bottlenecks – balance workloads
  - Reduce latencies

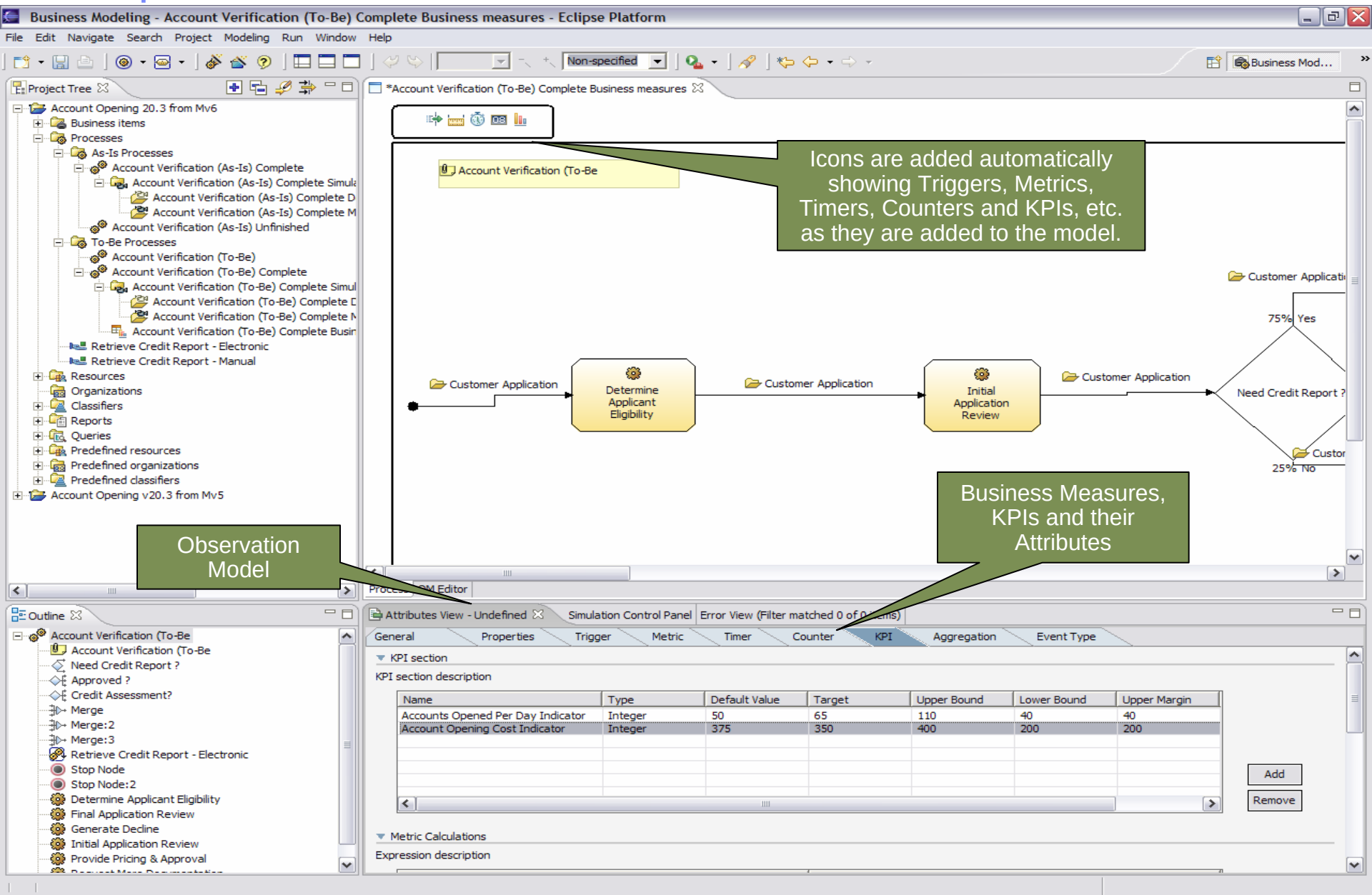
- Intervene in deployed processes
  - Set situational triggers and notifications
  - Dynamically respond to these alerts
- Monitor business process metrics
  - Cost, time, resources
  - Make modifications based upon simulation data sent back to the WebSphere Business Modeler
- Mitigate risks by letting the WebSphere Business Monitor provide you with the relevant real-time data required run your business



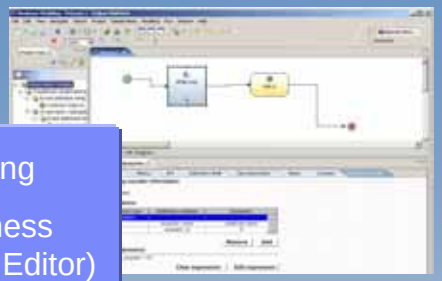
WebSphere Business Monitor v6

- Open standard event infrastructure – Common Base Event (CBE)
- Uses WB modeler as the toolkit to model business measures and KPIs
- Key Features
  - Manage in flight processes
    - Monitor executing processes (i.e. Status, duration, cost, execution path, inspect process instance data)
    - Administer process instances (i.e. Start/Stop, transfer work items)
    - Export actual process data to Modeler for process re-engineering
  - Monitor the Business Performance of active processes
    - Business measures and KPIs calculated from live process data
  - Detect Business Situations and take action
    - Notifications sent for manual response: Email, Pager, SMS messages (future)
    - Invoke automated responses: a BPEL process, a Web Service
  - Gather Business Intelligence from collected process data with the ability to
    - Analyze business metrics over time to identify trends
    - Discover previously hidden patterns using dimensional analysis
  - Render information in role-based dashboards and scorecards to provide actionable insight

WebSphere Business Modeler: Business Measures Editor



How Business Monitor makes it happen



Tooling

(Business Measure Editor)

Define BMs

Business Measures Model

CEI

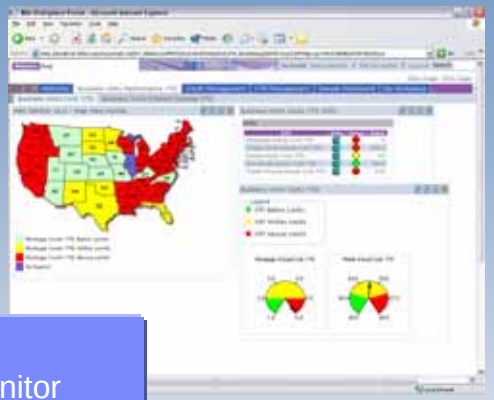
Monitor Server

Action Manager

CBE events

CBE Situation events

Event Processing



Monitor Dashboards

Dashboard access

State

Replication

Runtime

Performance Warehouse

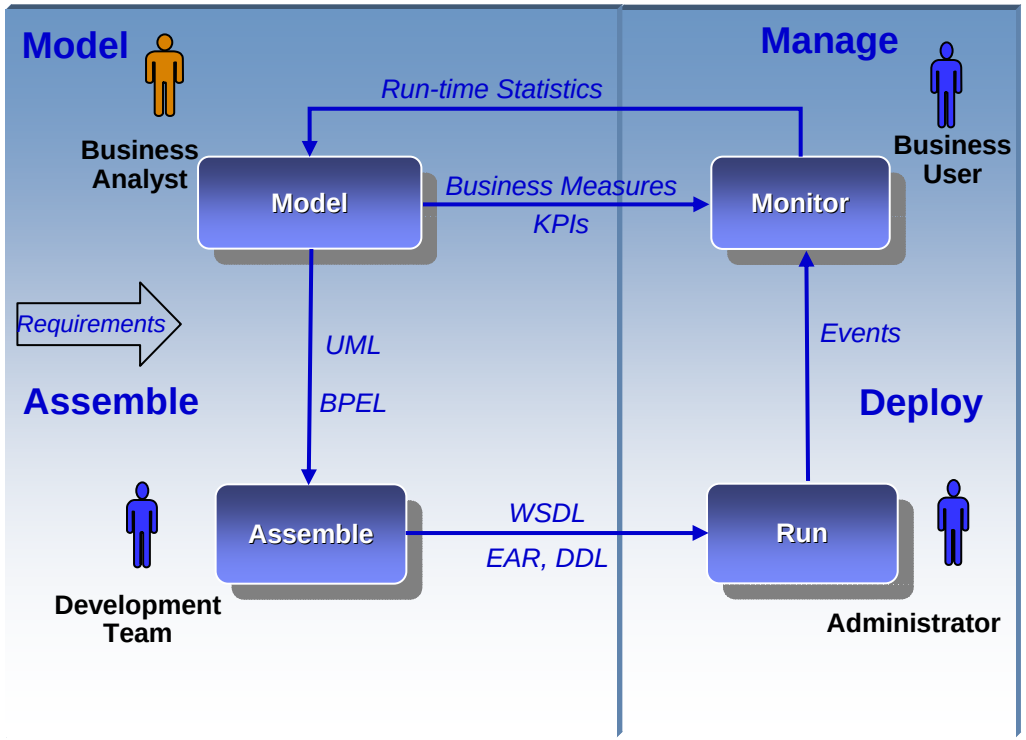
DB2 Replicator

DB2 Replicator

Monitor setup & usage scenario

1. Define Process Model

9. Monitor the Process



2. Define & Edit Business Measures

3. Complete runtime process models

8. Assign Users to Roles and Specify ACL

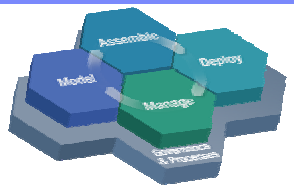
7. Deploy Dashboard Components

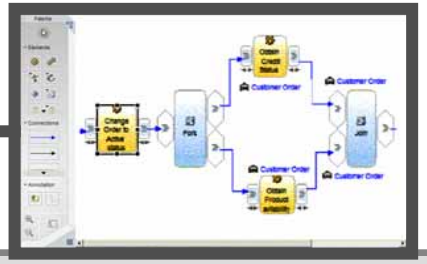
4. Deploy the Business Process

5. Deploy Business Measure Model, Export Database Schemas

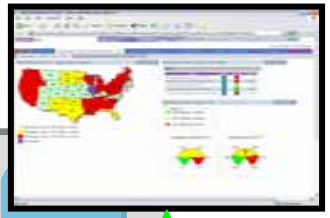
6. Deploy Data Schemas, Refine Cube models

Service Oriented Development





Business dashboard



Business Innovation & Optimization Services

Development Services

Interaction Services

Portal

Process Services

Information

Federated Query

IT Service Management

Partner Services

Community Manager

Business App Services

App EJBs

Access Services

Oracle Adapter

SAP Adapter

DB Access

Infrastructure Services

IT impact on processes

Thank you!