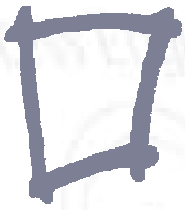


Bruce Silver Associates

BPM and Content Management Advisors

BPMInstitute.org℠
Peer to Peer Exchange for Business Process Management Professionals

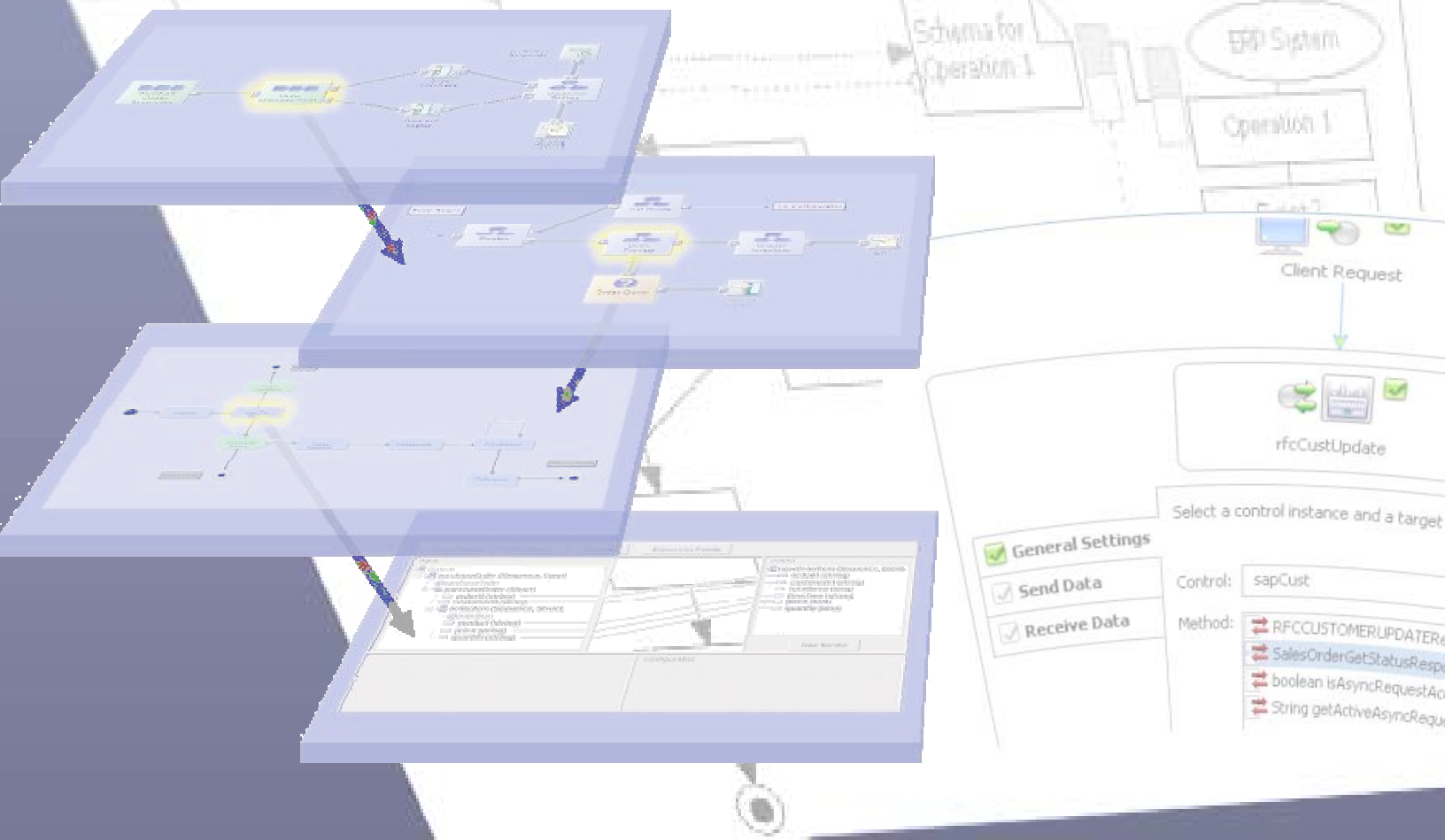
Published in conjunction with BPMInstitute.org

The 2006 BPMS Report:

IBM WebSphere BPM Suite v6.0

Bruce Silver

www.brsilver.com



IBM WEBSPHERE BPM SUITE

1. Vendor and Product Overview

IBM occupies a unique spot in the BPMS vendor landscape: It is the first (and, so far, only) infrastructure software supplier to offer a *complete* BPMS – supporting the end-to-end lifecycle from analytical modeling to performance management and optimization – based on BPEL and service-oriented architecture (SOA). IBM has a long history in the traditionally distinct markets for workflow, application integration, and performance management software, and is today leading the charge toward enterprise transformation through SOA. Previous generations of BPMS technology from IBM, sold under the WebSphere Business Integration brand, were based on a diverse collection of tools and engines, some J2EE-based and some not. IBM's new BPM suite, based on WebSphere Business Modeler, Business Monitor, and Process Server version 6, brings all of these functional elements together in a common service-oriented framework. The architecture provides a common orchestration engine, common data representation, common invocation model, and common design environment for all aspects of business integration solutions.

Business process management is only one style of business integration supported IBM's strategic SOA initiative. WebSphere Process Server and its associated design tools are aimed broadly at SOA developers building composite applications on WebSphere. The BPM suite "wraps" those components in business-oriented modeling and monitoring tools that bring a BPM perspective to service-oriented solution design and performance management.

WebSphere BPM implementations emphasize mission-critical integration-intensive processes in banking and financial services, insurance, electronics manufacturing, retail, energy, and telecommunications.

- A leading global financial institution was able to re-engineer trust services processes and integrate formerly non-integrated back end systems. As a result, account opening cycle time was reduced from six months to six weeks, netting an initial savings of $6 Million. Supported by automation and business rules, fee income increased tenfold.
- A large mortgage lender was able to cut its labor-intensive post-closing process time by 53 percent while achieving a 34 percent increase in efficiency, resulting in $4 Million in annual savings.
- A national telecommunications provider was able to integrate six different provisioning applications and automate the assignment of channels. Time for key provisioning steps decreased 15x – from 45 minutes to under 3 minutes – and developer productivity increased 50% from component reuse.

The WebSphere BPM suite has four primary components:

WebSphere Business Modeler v6 provides analytical modeling of process flow, resources, costs, data, and performance management; simulation of process scenarios; documentation; and export to process design tools. A new feature is the **Business Measures Editor**, which defines key performance indicators and other process metrics used in performance management.

500 Bear Valley Road
Aptos CA 95003 USA
Contact: Bruce Silver, Principal bruce@brsilver.com +1 831 685-8803

WebSphere Integration Developer v6 (WID) provides executable process modeling through service-oriented component assembly. Oriented to non-programmers, WID provides a simplified authoring environment supporting end-to-end design of all types of business integration solutions based on service components, including BPM solutions. As part of the BPM suite, WID imports BPEL and business objects from WebSphere Business Modeler and deploys executable models to the Process Server runtime environment.

WebSphere Process Server v6 is the foundation of IBM's service-oriented architecture. Layered on WebSphere Application Server, its SOA core services include Service Component Architecture, business objects, and the Common Event Infrastructure (CEI) used for process monitoring. In BPMS solutions, Process Server provides the BPEL orchestration engine, business rule engine, and component execution environment, as well as the **WebSphere ESB** and other integration middleware components.

WebSphere Business Monitor v6 is the performance management runtime component. Monitor Server processes *events* emitted by executing service components and writes performance data to a DB2-based operational data store. Using metrics defined in Modeler, both real-time and historical analytics can be displayed in dashboard tables and charts, and event-triggered actions executed through Monitor's **Action Manager.** WebSphere Business Monitor runs as portlets in IBM WebSphere Portal, which is included in the package.

Surrounding these four primary BPMS components, aimed at business analysts and integration specialists, are additional modeling, design tools, and runtime services for enterprise architects, J2EE developers, and legacy system specialists, which may be needed to build reusable services and complete BPMS solutions. Figure 1 shows how the WebSphere BPM suite components correspond to elements of the BPMS component structure described in Chapter 1 of the 2006 BPMS Report.

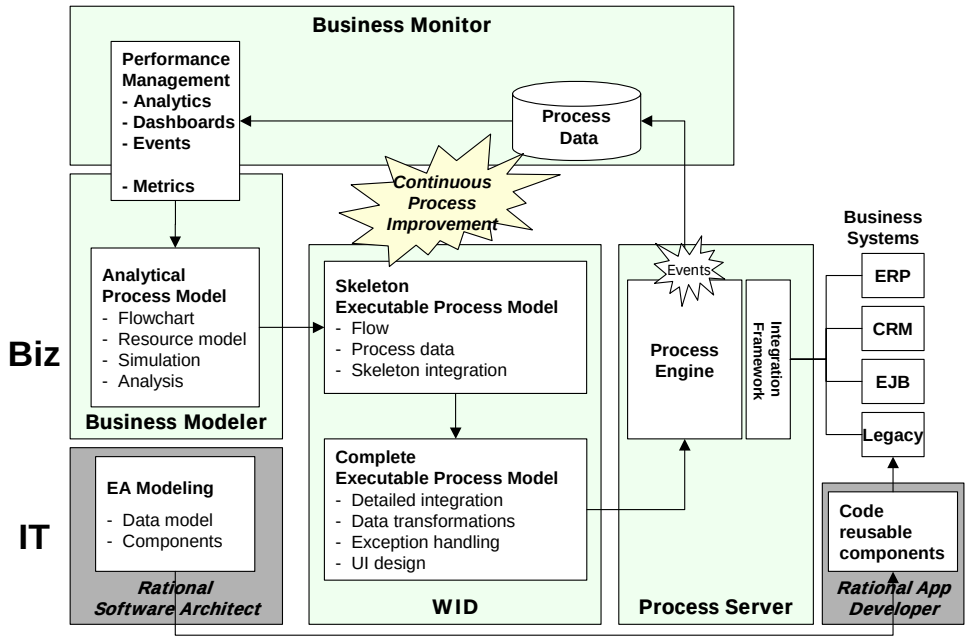


Figure 1. Mapping of WebSphere BPM suite components to the BPMS process lifecycle.

Unlike most BPMS vendors, IBM owns all of its suite components and has combined them in a unified design and execution environment. In fact, the WebSphere BPM suite is the result of a significant reworking of technology either acquired (e.g., Holosofx, Crossworlds, Rational, PureEdge) or developed in its own labs to place them all on a consistent service-oriented

architecture based on WebSphere and industry standards. Figure 2 shows the evolution of the previous generation of WebSphere Business Integration components into the consistent architecture of version 6.

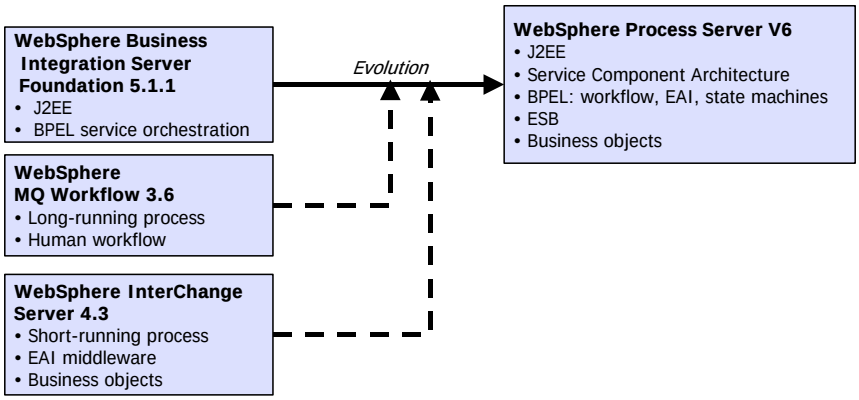


Figure 2. Evolution of previous business integration tools and runtime components to a unified WebSphere-based BPMS platform in v6.

It is important to keep in mind that IBM created key building blocks such as WebSphere Integration Developer and Process Server not exclusively for the BPMS marketplace, but for the broader SOA market as well. In fact, since service-oriented application development is still a bigger market for IBM than BPMS today, IBM's terminology and conceptual framework occasionally aligns itself with that SOA/integration world rather than with the standard nomenclature of BPMS.

2. Environment and Architecture

2.1 BPMS Runtime Architecture

2.1.1 WebSphere Process Server

IBM positions SOA as the new standard upon which all kinds of business integration applications are built, and WebSphere Process Server v6 as its strategic offering for service-oriented integration applications. Based on SOA principles, Process Server provides WebSphere with:

- A universal data representation, called *business objects*
- A universal invocation model, called *Service Component Architecture (SCA)*
- Common orchestration of service components in a business process, based on *BPEL*.
- Universal connectivity fabric to all service components, called *WebSphere ESB*
- *Common Event Infrastructure* (CEI) for monitoring performance of integration solutions

These features are designed to simplify and accelerate development of integration applications, promote agility and reuse, and significantly lower the programming effort and skill required.

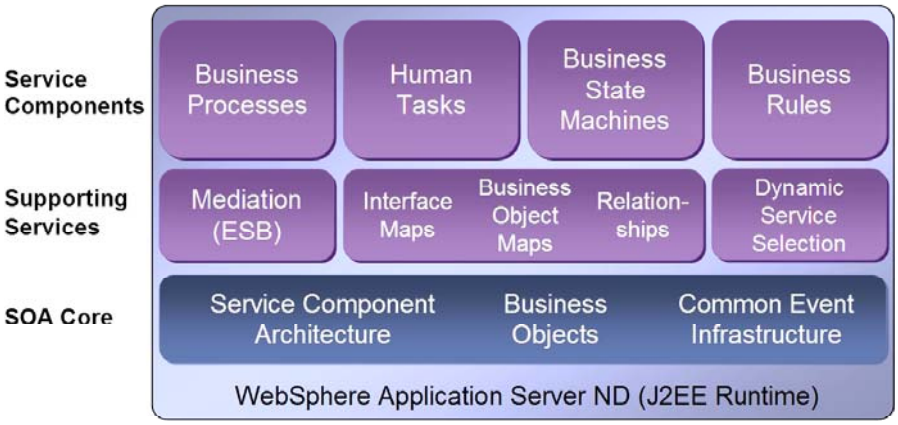


Figure 3. Functional blocks of WebSphere Process Server. Source: IBM

Figure 3 shows the functional blocks of WebSphere Process Server v6. At the lowest level is the SOA Core, including SCA, Business Objects, and CEI. This is all layered on and extends the features of WebSphere Application Server itself, including performance scalability, high-availability, security, transaction management, and basic resource management.

Above the SOA Core are layered the primary and supporting service components themselves. Despite its name, Process Server technically does not "automate business processes," but instead "integrates service components." A business process is just one type of service component. Other types of service components include human tasks, business state machines, and business rules. A business state machine is like a business process, except that instead of being modeled as a flowchart or a UML activity diagram, it is modeled using the event-condition-action paradigm of UML state charts. Both business processes and business state machines are ultimately represented as BPEL. Supporting service components include data transformation, mediation (message transformation and brokering), and selectors (dynamic runtime selection of a particular service endpoint).

2.1.2 Service Component Architecture and Business Objects

IBM describes service component architecture as a programming model that dramatically simplifies the development of business applications by separating business logic from technical infrastructure code. SCA encapsulates software functions in composable units that hide the technical infrastructure and expose only business-level interfaces needed for integration. Those units, called *service components*, are described by their WSDL or Java interfaces, but their implementation can be virtually anything, including Java classes, EJBs, or JCA adapters. Process Server artifacts like business processes, human tasks, business rules, state machines, transformations, and selectors are all represented as service components.

Individual service components are wired together into larger units called *modules*, encapsulating fragments of integration logic for reuse. A change to service components within one module will not impact any other module in the overall solution as long as the interface of the changed module stays the same.

Just as service components are a generalization of web services, *business objects* are a generalization of the XML messages used to invoke web services. In addition to XML, business objects support other data descriptions such as JDBC ResultSet, and carry *metadata* like change history and context (e.g., Update, Create, Delete – the "verbs" familiar to users of the older WebSphere InterChange Server), but ultimately business objects can still be fully described through XML Schema (xsd).

With SCA, a service component is invoked by calling its *interface* (Figure 4). The interface specifies which operations a service component offers and what data (business objects) makes up the parameters to these operations. The invocation can be synchronous or asynchronous through a variety of technologies and protocols, including Java Objects, EJBs, JMS, or WebSphere Adapters.

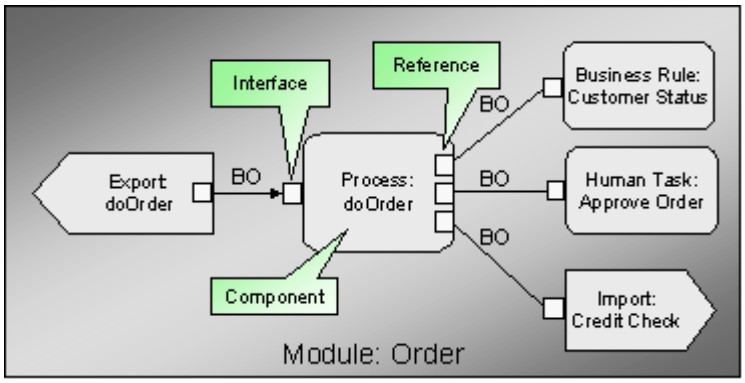


Figure 4. Component *modules* are the fundamental units of reuse. Source: IBM

2.2 Development Roles and Tools

The WebSphere BPM suite is the first serious attempt to bring service-oriented process design and management to a non-programmer audience. At a time when most SOA vendors are focused solely on enterprise architects and skilled developers, IBM has created a suite of tools, each oriented to a specific role and skill set, that can interchange data and process models to create a truly collaborative environment shared by business and IT (Figure 5). The various design components of the suite can work together because they all run in Eclipse 3.0. Eclipse is a popular browser-based integrated development environment, originally developed by IBM and later placed in the public domain. Design tools from various sources can be plugged into Eclipse and share common services. Eclipse also allows specific screen layouts, called *perspectives*, to be tailored to the needs of particular design roles and activities. While Eclipse has historically been mainly a programmer environment, IBM has extended it to business analysts and other non-programmers involved in process modeling and performance management.

Role	Focus and Skills	Tool
Business Analyst	Analytical modeling of business process and performance. Line of business knowledge, not technical.	WebSphere Business Modeler
Integration Specialist	Executable process modeling via graphical component assembly. Some technical knowledge, but not a Java or J2EE programmer.	WebSphere Integration Developer
Software Architect	Technical requirements, UML modeling and solution architecture	Rational Software Architect
J2EE Developer	Design and coding of services, custom user interfaces, and other components	Rational Application Developer
Enterprise Developer	Adaptation of legacy assets to SOA. Knowledge of COBOL/PL1/RPG and some J2EE.	WebSphere Enterprise Developer

Figure 5. IBM allows tools oriented to specific process design roles and skills to share data and models through the Eclipse 3.0 environment.

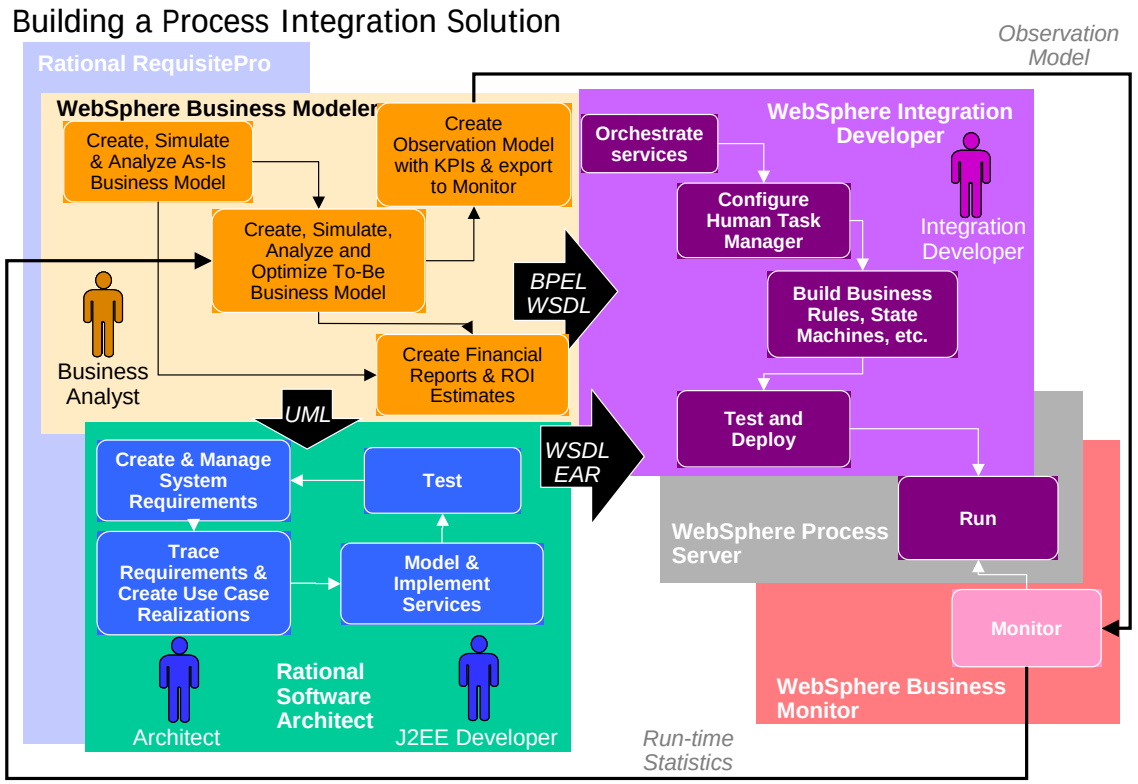


Figure 6. Tools in WebSphere BPM suite interchange model information via UML and BPEL standards. Source: IBM

3. Analytical Modeling

Some BPMS vendors do not even include analytical modeling as part of the offering, but IBM places it front and center in its approach to BPMS. While WebSphere Integration Developer, Process Server, and the Rational tools serve both the BPMS market and the broader SOA application development market, wrapping these tools in WebSphere Business Modeler specifically targets the BPM audience. WebSphere Business Modeler v6 is not only the modeling tool for upfront process capture, simulation, and analysis, but it defines the key performance indicators and other business measures that are calculated and displayed by WebSphere Business Monitor. This section of the report will discuss the upfront modeling and analysis features; discussion of performance management modeling will be deferred to section 10, Performance Management.

WebSphere Business Modeler describes processes in terms of seven separate models. Besides the process model itself, the tool defines separate models for resources and staff roles, organization (reporting relationships), process data (information model), simulation parameters (analysis model), web publishing to non-Modeler users, and business measures used for performance management.

Figure 7 shows a fragment of the Process Model. For nontechnical business users, this model offers a basic editing mode, with additional analyst-oriented functionality available in intermediate and advanced editing modes. It also has separate technology modes for Process Server v6 and previous generations of IBM BPM engines.

The Process Model (Figure 7) describes the process as a flow of activities, each assigned a resource, resource cost, and average duration. At branch points in the flow, the percentage of instances following each branch can be assigned. Connectors between activities also indicate the business object passed between them, defined in the Information Model (Figure 8). Based on resource availability and cost, activity duration, branching ratios, and details of business item attributes, process models can be analyzed dynamically through simulation.

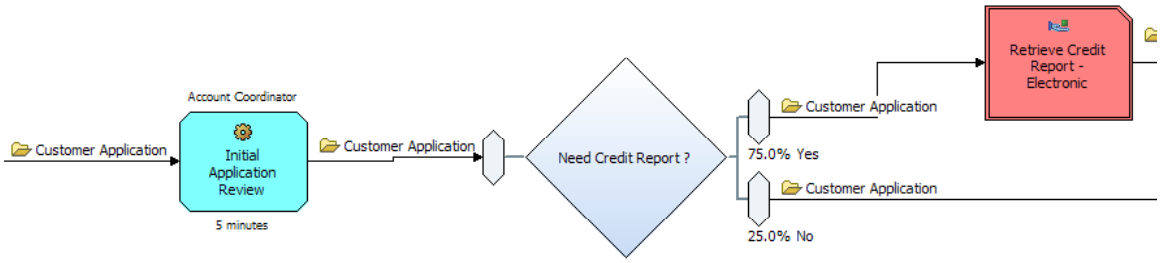


Figure 7. Process Models indicate activities, resources, and information flows. Source: IBM

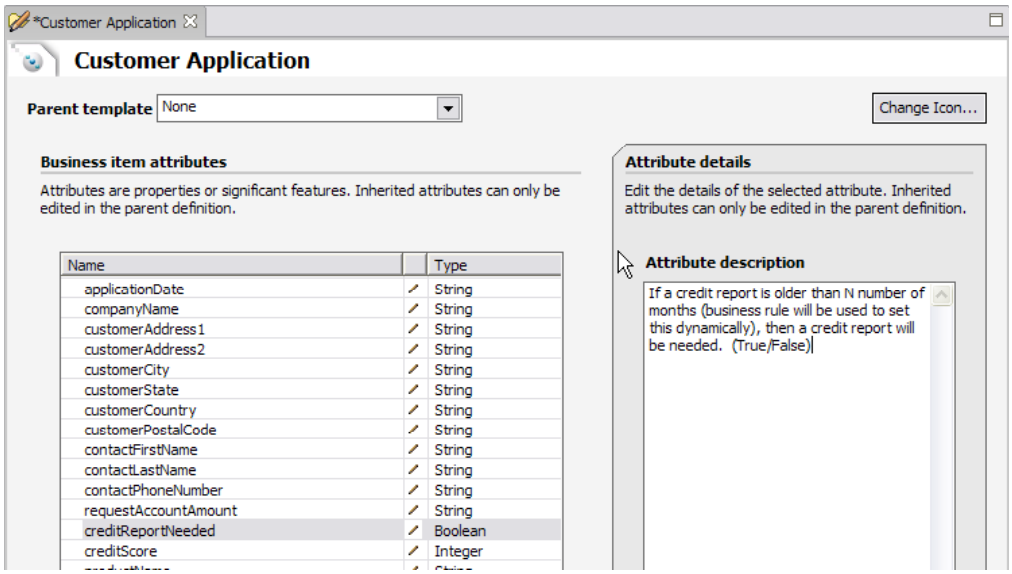


Figure 8. Information Model defines process data elements. Source: IBM

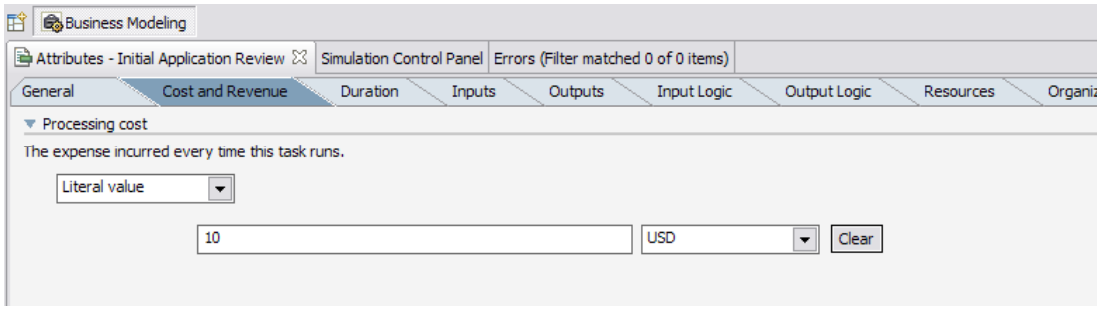


Figure 9. Analysis Model defines simulation parameters. Source: IBM

The Analysis Model (Figure 9), which defines simulation parameters, is extraordinarily detailed. For example, rather than just assigning a cost to a process activity, the model allows the user to

separately specify regular processing cost, startup cost, wait time cost, revenue realized upon task completion, etc. And that's just one tab of the model!

Using the Analysis Model, the process can be run through Modeler's simulation engine using various scenarios, and optimized for cycle time, total cost, resource utilization, or other metrics. As the simulation runs, backlogs at each activity are displayed in real time on the Process Model diagram, and various built-in metrics are calculated. In addition, simulation results can be output to Crystal Reports for customized display and reporting.

4. Process Structure and Data

4.1 Structure of an End-to-End Process

Within WID, the integration specialist wires together components into *modules*. Modules are the actual units of deployment to the Process Server. Modules call each other implicitly through their *exports* (interfaces) and *imports* (references to other modules or components). Import and export bindings include integration adapters (both JCA 1.5 and messaging-based WBI adapters), web services (SOAP over HTTP or JMS), JMS messaging, EJB, or a standalone API.

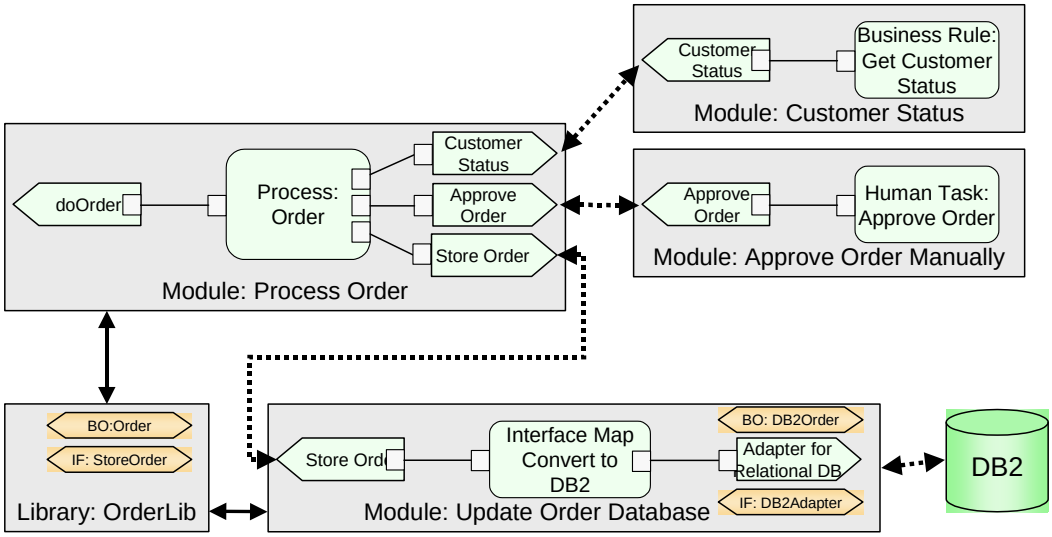


Figure 10. End-to-end processes are composed by interconnecting modules. Source: IBM

In BPMS, it is common to think of an end-to-end process as a single entity, with a defined beginning and end, composed as a hierarchy of nested and chained activities and subprocesses. Essentially this is what WebSphere Business Modeler describes, but WID and Process Server describe integration solutions in terms of interacting components and modules, not a hierarchy of processes and subprocesses. For example, in WID, a BPEL process, a human task, and a business rule are all service components. To a BPM audience, this can be confusing. While recognizing that WID and Process Server can be used for all kinds of integration solutions, when used in BPMS we need to relate service components back to core notions of business process.

Here's the connection: WebSphere Business Modeler, representing a true BPM perspective, exports to WID definitions for the top-level process and any subprocesses as BPEL documents, process interfaces as WSDL documents, and process data as XML Schema documents (xsd). WID imports these artifacts and from them automatically generates service components, interfaces and references, and business objects. Thus *when used in conjunction with Business Modeler*, the top-level process in WID is typically represented by a business process component (i.e., a BPEL process), that invokes other components through their service interfaces. For

example, in BPM the module Process Order in Figure 10 would normally be considered "the process," and the other modules would be considered by Modeler to be process activities or subprocesses. In WID, they're all simply modules.

Libraries are repositories of shared business objects and interfaces, facilitating component reuse. For example, Figure 10 illustrates a process in the form of a component assembly in which the *Update Order Database* module, invoked through the *StoreOrder* interface, converts order data to DB2 and stores it in the database. Because *StoreOrder* and its underlying business object is maintained in a library, the *Update Order Database* module can be replaced by a different *Update Order SAP* module having the same *StoreOrder* interface, without changing any other process components. Backend specific information (*DB2Order*) is encapsulated in the module since there is no benefit of sharing it with other modules.

4.2 Process Data and Data Transformation

The WebSphere BPM suite relies on a common data model for all components based on *business objects*. The Business Object Editor specifies object elements as either base types or other business objects, and allows complex relationships to be defined between various business objects. When a business object is specific to a particular application environment, it is called an *Application Specific Business Object* (ASBO). When a business object definition is independent of application environment it is called a *Generic Business Object* (GBO). GBOs allow canonical representations of common business objects, such as *Customer* or *Order*, which support reusable module interfaces. For example, in Figure 10, *Order* is a GBO and *DB2Order* is an ASBO. While the WebSphere BPM suite does not insist on the use of GBOs, they make processes more flexible and promote component reuse.

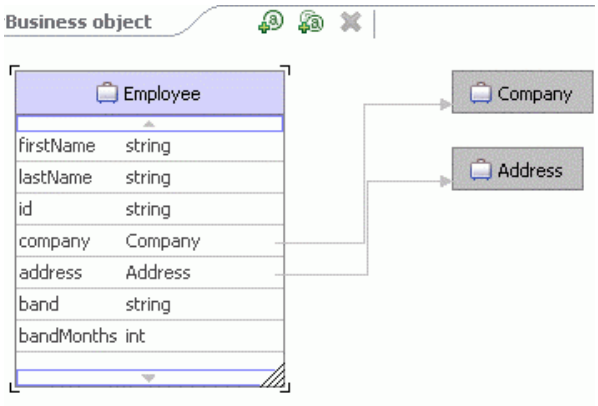


Figure 11. Business objects are the common data representation for IBM's SOA. Source: IBM

Again referring to Figure 10, the module *Update Order Database* contains a Transformation component that maps the *Order* GBO to the *DB2Order* ASBO. In WebSphere Process Server, data transformation is implemented through a *Mediation Framework* that operates at three levels. *Interface mapping* connects different implementation types and parameters, for example, from a BPEL process to a JCA adapter. *Data mapping* transforms the business object elements and metadata (Figure 12). WID provides rich graphical data mapping supporting Move, Join, Extract, and Assign of data fields as well as custom Java mapping. *Relationship mapping* matches identifiers between the two interfaces. These mappings are often used together (Figure 13).

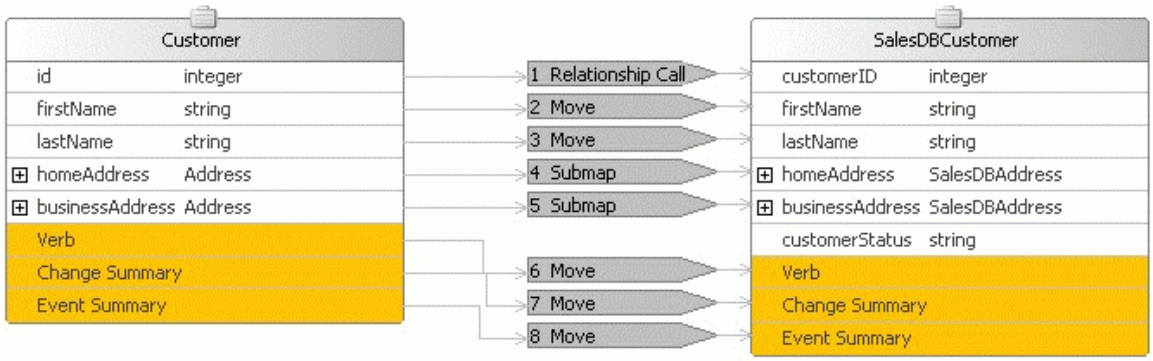


Figure 12. Data Map in WID. Source: IBM

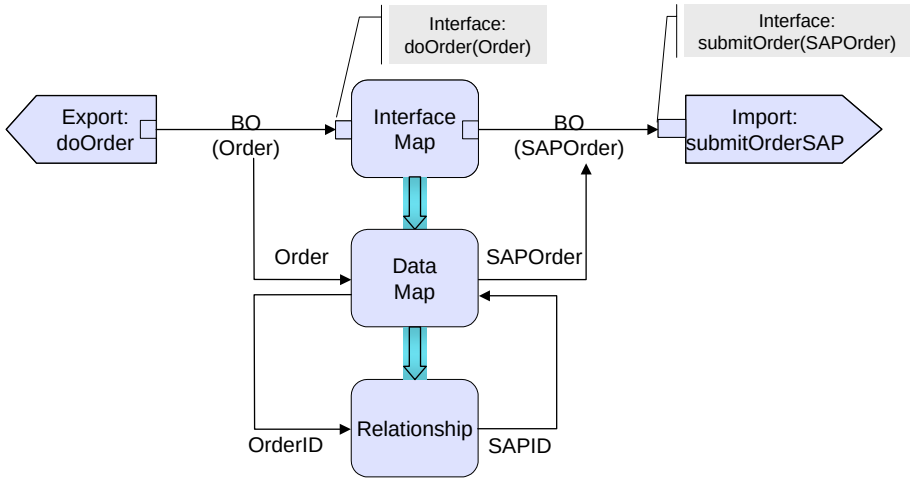
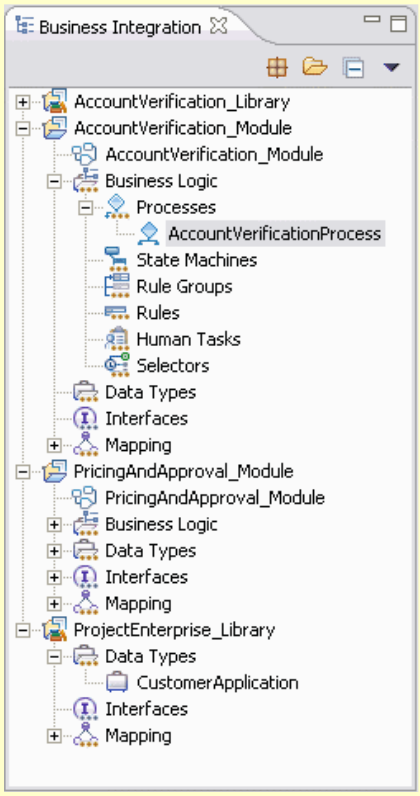


Figure 13. Business object transformation involves a chain of three mappings. Source: IBM

4.3 Integration Solutions in the WID Environment

In the WID environment, a business integration solution is built in the *Business Integration* perspective. The various integration artifacts involved are described in Figure 14.



Solution Folder	Contents
Business Integration Perspective	Contains modules and libraries. Modules contain Service Components and other artifacts. Libraries contain reusable artifacts like business object definitions or interface definitions.
Module Assembly	The module assembly is used to connect the various service components in a module. It is also used to specify QoS characteristics about the various component interactions, to import existing artifacts (like Web Services) and to export the interface of the module (e.g. Web Services or JMS interface).
Business Logic	Contains components. Components are divided by their implementation types into subfolders: Processes (BPEL), Business State Machines, Business Rules (Rule Groups and actual Rules), Human Tasks and Selectors.
Data Types	Contains the Business Objects used in the Solution.
Interfaces	Contains the Interfaces used by the Components that are contained in the Solutions.
Mapping	Contains elements required for transforming data and interfaces: Interface Maps, Business Object Maps, Relationships.

Figure 14. Integration solution artifacts in WID. Source: IBM

4.4 Assembly Editor

The unit of deployment and reuse in the WebSphere BPM suite is a *module*, also called a *component assembly*. Within a module, business logic elements are represented by *service components* interconnected by the exchange of *business objects*. WID provides an *Assembly Editor* (Figure 15) for defining modules and the high-level interconnections between their components. Proxies for external interfaces of modules are represented as *exports* and *imports*.

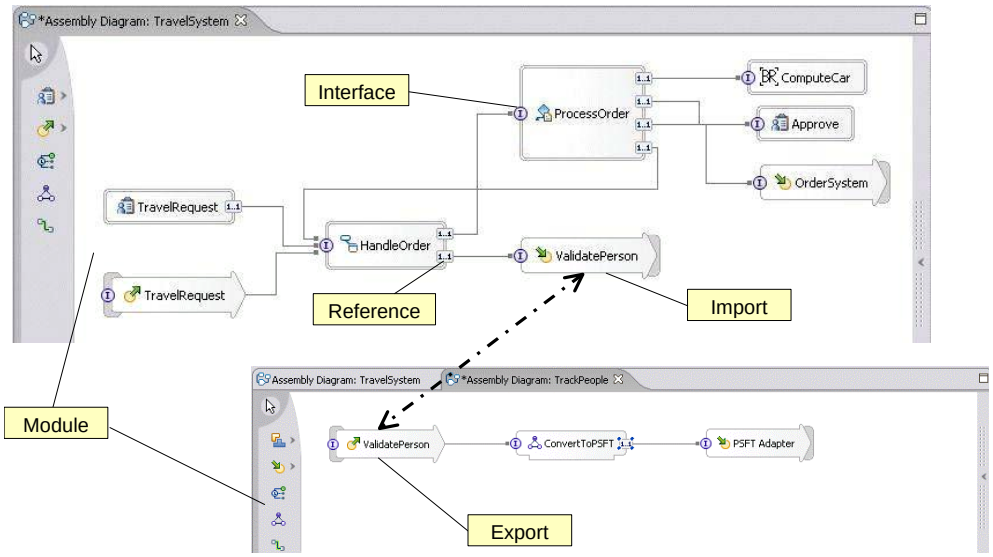


Figure 15. Assembly Editor. Source: IBM

4.5 Business Processes

WID provides a graphical BPEL Process Editor for creating *business process components* (Figure 16). Steps in the process diagram represent BPEL activities, specified as service invocations. In the Process Editor, process component interfaces and references are configured as BPEL partnerLinks, and process data elements are configured as BPEL variables. WebSphere Business Processes support the WS-BPEL 2.0 specification (technically still in Draft stage in OASIS), and adds a number of IBM extensions. Besides the staffing activity extensions mentioned earlier, a frequently used extension allows BPEL to invoke a Java snippet inline, typically used to map data to business object variables, similar to standard *Assign* activities for XML variables. Activities can be grouped into transactional units called *scopes*, with exception and compensation handlers defined at the activity, scope, or process level.

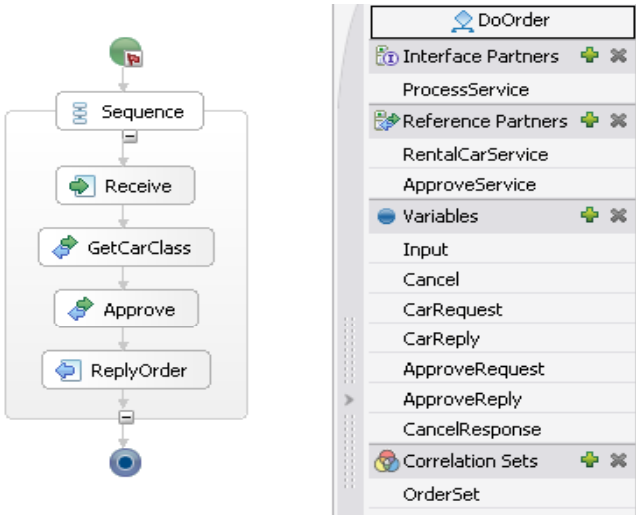


Figure 16. Business process components created in WID with a graphical BPEL editor. Source: IBM

4.6 Business State Machines

In addition to business processes, WID provides an alternative description of BPEL processes called *business state machines*. Appropriate for event-driven processes in which the effect of the event depends on the process state – for example, an order handling process in which *orderCancel* or *orderChange* events might be received – business state machines are modeled in WID with a special editor based on UML state charts (Figure 17). Nodes in these diagrams represent *states*, and *actions* are performed as part of state *transitions* triggered by *events* and *guards* (conditions) or timeouts. While the modeling metaphor is completely different from the business process editor (Figure 16), underneath the covers the business state machine editor also generates BPEL, so the same process could in principle be defined in either one.

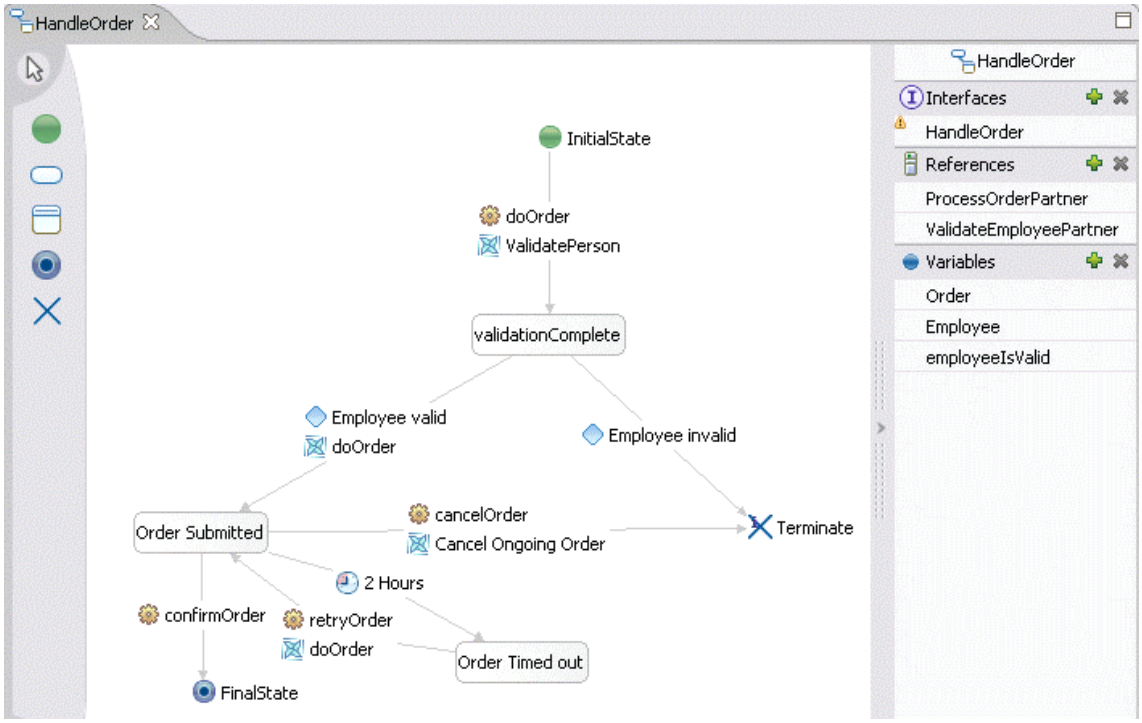


Figure 17. Business state machine components are event-driven BPEL processes specified as UML state charts. Source: IBM

5. Human Workflow

5.1 Human Tasks

In WebSphere Business Integration Server Foundation 5.1, IBM made a clear distinction between human and automated activities, and used BPEL extensions to invoke so-called *staff activities*. WID v6 still allows that, but has added a new *Human Task component*, invoked with standard BPEL, as the preferred alternative. To the BPEL process a human task component is represented simply by its interface and the operations it supports (e.g., approve order). This allows a nominally human approval activity to be replaced by an automated business rule, web service, or other component simply by changing the wiring in the assembly diagram, without changing the calling process.

IBM has featured an elaborate set of human workflow capabilities dating back to the earliest days of MQ Workflow, and these have been carried over to WebSphere Process Server v6. The

software supports a variety of human task types with different capabilities in a business process (Figure 18):

- Originating Task (oTask) can invoke another service component, such as another business process. In Figure 18, *BookOrdering* and *WaitForBook* are oTasks.
- Administrative Task (aTask) allows a user to administer another service component.
- Participating Task (pTask) is invoked by a service component, typically a business process.
- Pure Human Task (hTask) can be created like a To Do reminder and have WebSphere Process Server track this task.

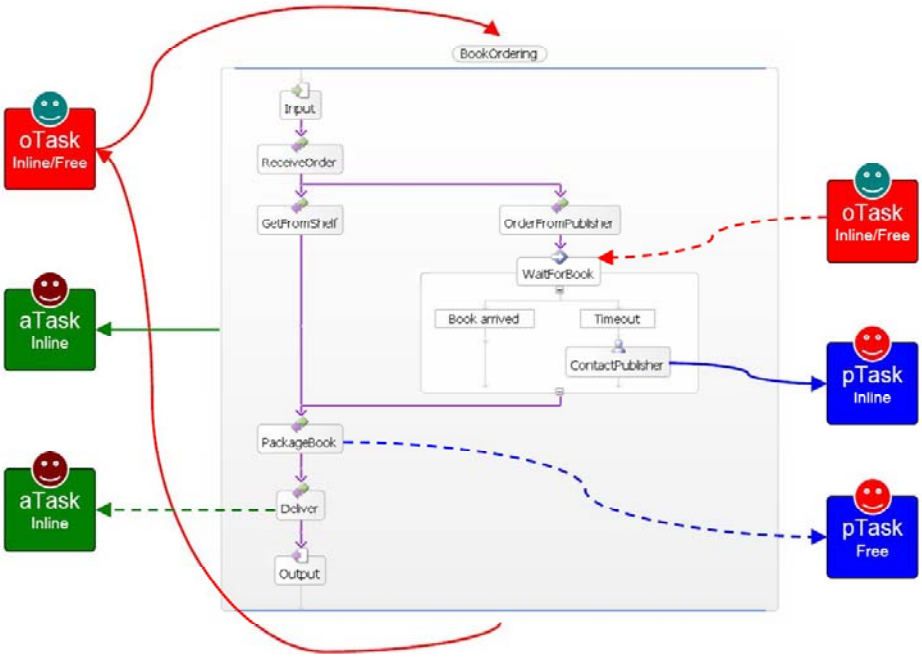


Figure 18. Human tasks in a BPEL process. Source: IBM

WID provides a Human Task Editor that configures tasks by means of *Task Templates*. A Task Template defines settings for Staff, Client, and Escalation. Staff settings define who is authorized to start, claim (i.e. perform), modify, and monitor a task. IBM has long featured an elaborate task staffing model. Each authorization may be assigned to defined roles, groups, specific individuals, managers of other task performers, or even via other complex queries.

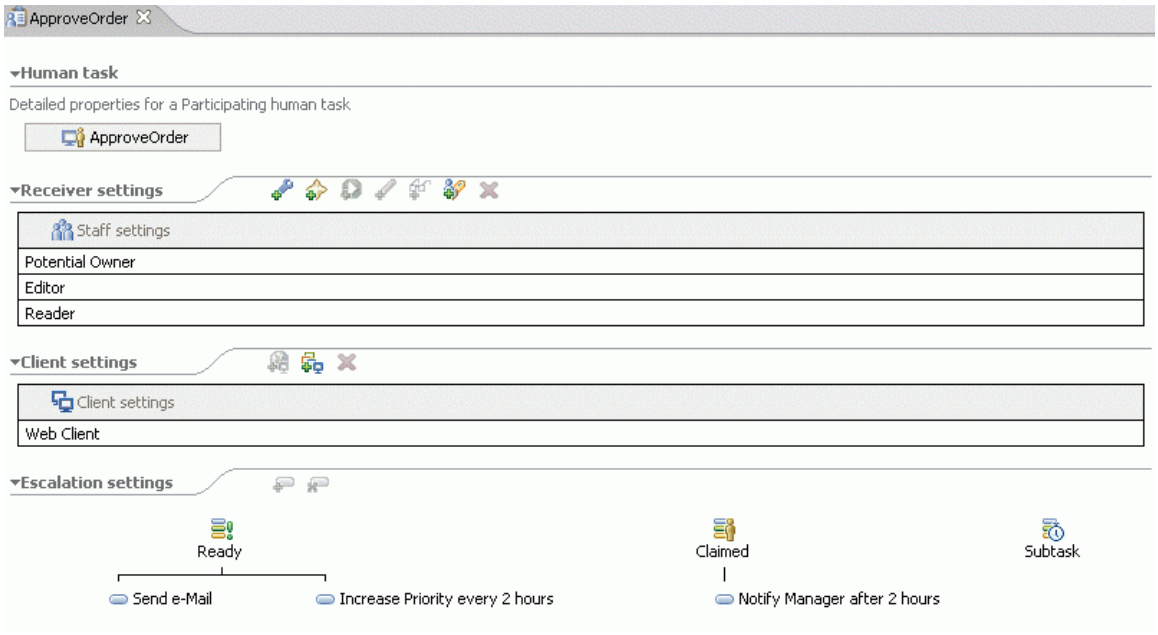


Figure 19. Human Task Editor. Source: IBM

5.2 Web Client

Client settings specify the task presentation, either the standard web client, custom web client, or in a portlet. WebSphere Process Server's web client, called *BPC Explorer* (Figure 20), is built on a set of reusable Java Server Faces (JSF) components that can be used to create custom clients or embed human task functionality into other web applications.

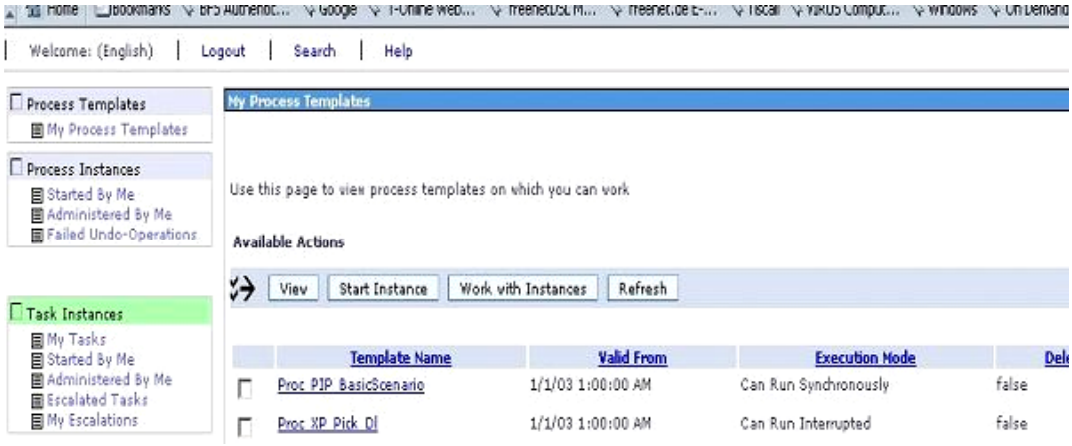


Figure 20. BPC Explorer provides worklist management and task user interface. Source: IBM

Escalation settings define deadlines and triggered escalation actions, including notifications, generated tasks, or invocation of an observer event that allows user-determined actions, and which can be extended to elaborate multi-stage escalation flows. Also, every time an escalation fires the priority of a task can be increased.

The BPC Explorer includes a worklist manager that enables end users to claim, complete or transfer human tasks. When a human task is started, a pre-defined Java Server Page is launched. The full set of APIs is available for the JSP developer to access and display process data. At this point it is also possible to integrate other applications like e-forms applications or document

management applications. Alternatively WebSphere Portal includes a task list portlet that can be used as well.

6. Integration Framework

The WebSphere BPM suite offers undoubtedly the most comprehensive integration framework of any BPMS, capable of connecting virtually any application or information system, J2EE or legacy, to the business process. In WebSphere's service orchestration architecture, integration actions are modeled in WID as invocations of service components.

6.1 ESB

In SOA, the connectivity fabric linking service requestors and providers is called an *enterprise service bus* (ESB). IBM's ESB architecture relies on the *Service Integration Bus* built into WebSphere Application Server v6, which may be used in conjunction with a WebSphere MQ message bus backbone spanning the enterprise (Figure 21). This Service Integration Bus, sometimes referred to as *WebSphere Messaging Resources*, provides the infrastructure for WebSphere ESB and is based on the Java Message Service (JMS) standard. A service component can send XML and SOAP messages to JMS queues or URLs over the bus, or invoke operations on external systems by sending business objects to J2EE Connector Architecture (JCA) v1.5 integration adapters. It also allows process components to receive invocations and events from J2EE applications via SOAP or JMS. The ESB is also the medium through which the Process Server communicates to human process participants, through web clients, WebSphere Portal, or WebSphere Everyplace Access.

In addition to transport, WebSphere ESB provides *mediation services*, with capabilities of filtering, logging, and transforming messages, and performing database lookups. These message flows are modeled in WID as *mediation components* using WebSphere-supplied primitives.

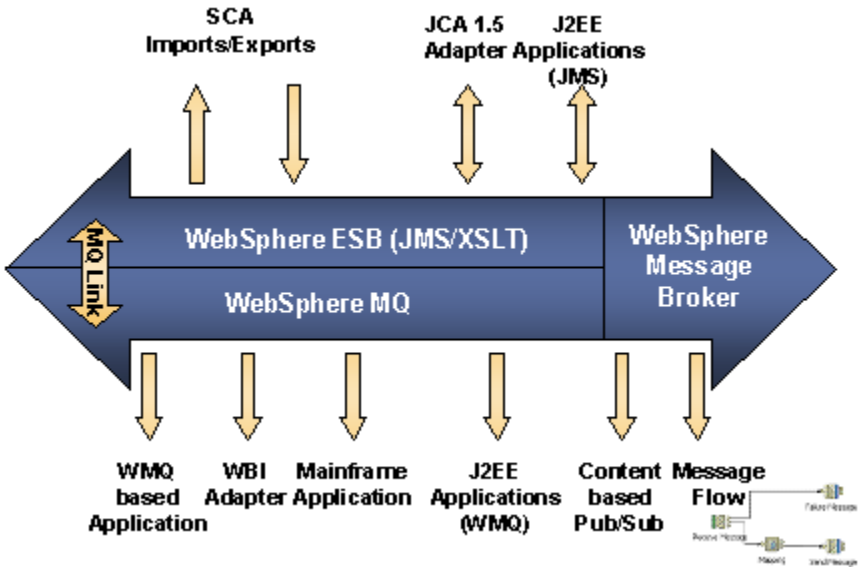


Figure 21. Service connectivity and mediation are provided by a combination of WebSphere's built-in ESB and WebSphere MQ. Source: IBM

WebSphere MQ is an IBM-proprietary message bus long established for high-performance enterprise integration. In addition to XML it supports legacy data formats such as COBOL Copybook, and is accessible not only from J2EE clients but from any JMS or WebSphere MQ

client. Beyond the MQ message bus itself, *WebSphere Message Broker* (formerly called WebSphere MQ Integrator) provides its own set of mediation services and message flows.

6.2 Adapters

WebSphere Adapters allow service components to communicate with a wide variety of external applications and information systems using a consistent SCA programming model. IBM provides two types: JCA 1.5-based adapters and JMS-based adapters. JCA-based adapters for WebSphere Process Server support SAP, PeopleSoft, Siebel, JDBC, and flat files, plus CICS and IMS. JMS-based adapters support these plus a long list of packaged applications, middleware, and mainframe programs. WebSphere adapters essentially provide SCA-compatible *interfaces* and *business objects* for external system functions and events. An outbound adapter makes external business functions accessible as an *import* in an SCA module. An inbound adapter allows an SCA module to listen for service requests, i.e. defines an *export* in the module.

WID provides an *Enterprise Metadata Discovery Wizard* (Figure 22) that uses WebSphere Adapters to introspect external business systems and create the interfaces, business objects, exports, and imports for the selected function or event *automatically*. These artifacts are then used in business process design. For example, business objects generate BPEL input and output variables, and interfaces define BPEL partnerLinks. Imports are used in assembly diagrams for the process component.

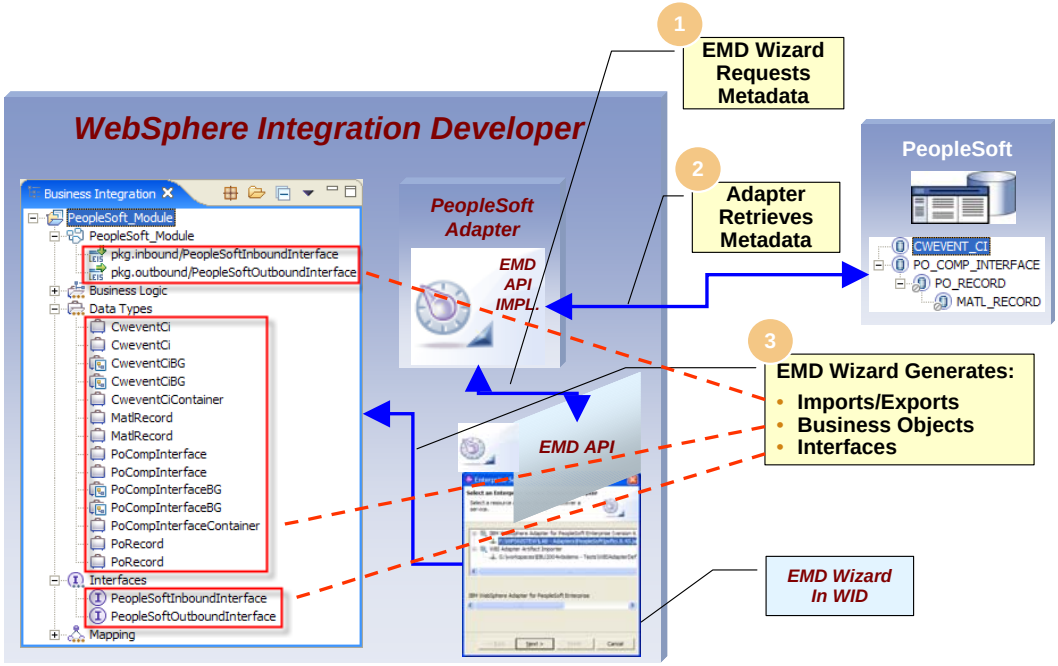


Figure 22. Enterprise Metadata Discovery Wizard introspects business systems and generates integration artifacts automatically. Source: IBM

6.3 Web Services Support

In many BPMS offerings, the ability for a process to invoke a web service or to be invoked itself by a web service request is a special integration feature. In WebSphere Process Server – as in any BPMS based on the service orchestration paradigm – it is fundamental to the basic process architecture. Any external resource already implemented as a web service, or newly created as a web service using Rational Application Developer or other programming tools, is by default a

valid implementation of a WebSphere service component. Similarly, any WebSphere business process module can be invoked as a web service by an external client, simply by defining its *export* as a web service. The necessary web service infrastructure is native to WebSphere Application Server. In WebSphere Process Server SOAP-based integration is almost trivial; the power of SCA is that it allows the process to be invoked as a more generalized "service," specified by a WSDL interface, over the client's choice of communications protocols – Java, JMS, IIOP, etc.

7. Business Rules

While most BPMS offerings implement business rules by integrating a third party rule engine and toolset, WebSphere Process Server provides a business rule management capability native to the BPMS itself, saving considerable cost and integration effort. Business rules implement complex decision logic on business objects, and can change without versioning the business process. Business rules are modeled in WID either as simple *if/then rulesets* (decision trees) or *decision tables* for multi-dimensional rule calculations. Rulesets and decision tables may be aggregated in *rule groups*, configured so that only one rule in the group is in effect at any given time. Business rule logic is deployed as a *Business Rule Group component*, so it can be invoked by a business process or any other service component. The Business Rule Group encapsulates the actual rule implementation into a service component. All the consumer needs to know about the component is the interface – not whether the business rules are implemented as rulesets, decision tables or a combination of the two.

The result of a business rule is returned as a business object element value. A business process invoking the rule then uses the returned result to determine any triggered action required. During their execution business rule components can invoke other service components, for example, to retrieve data from an external system.

Business rule designers may parameterize rules so that the rule logic may be modified at runtime. *Business rule templates* (Figure 23) define the rule parameters. With rule templates, a runtime web interface called a Rule Management Application allows authorized business users to adjust parameter values on the fly. The new rule logic takes effect immediately, even for instances in flight.

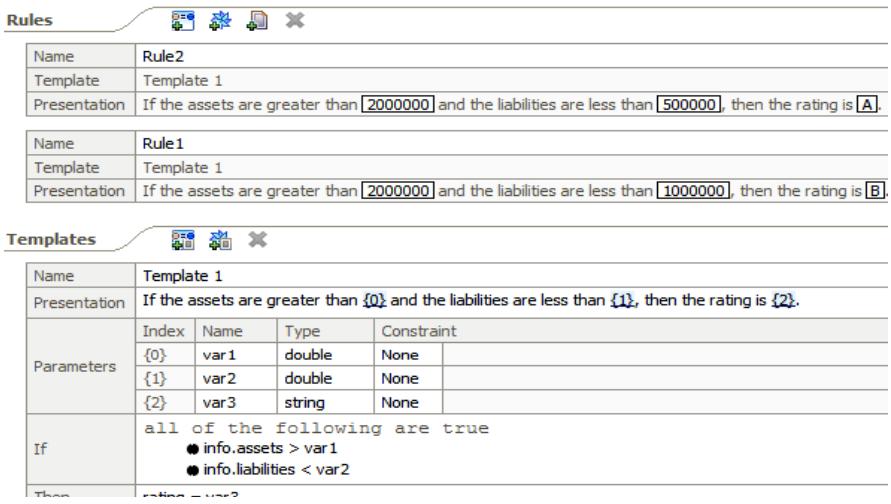


Figure 23. Rule templates allow business rule parameter values to be adjusted on the fly. Source: IBM

In addition to Process Server's native rule engine, the ILOG business rule engine is also integrated with Process Server. ILOG provides a wizard that automatically generates Java components that allow Process Server to call ILOG rulesets. These ILOG rule components are then interchangeable with IBM-native rule components in WID.

8. Content, Collaboration, and Case Management

IBM is also a leader in the enterprise content management market and provides a variety of ECM offerings that can be integrated with the BPM suite.

DB2 Content Manager is an enterprise-scalable ECM platform supporting document imaging, report management, revisable document management, archiving, and records management. The current version integrates with WebSphere Business Integration v5, and a future version of DB2 Content Manager will provide integration with Process Server v6.

IBM Workplace provides a horizontal platform for team collaboration and document management, as well as a suite of collaborative applications. Workplace integrates with WebSphere Business Monitor through portlets, using WebSphere Portal technology. IBM *Workplace for Business Strategy and Execution* provides portlets for business performance management will include WebSphere Business Monitor. *Workplace for Business Controls and Reporting* provides a solution for Sarbanes-Oxley and similar compliance applications, and also integrates with Monitor via portlets.

9. Events and Exceptions

9.1 Business Events

WebSphere Process Server has a comprehensive set of facilities to respond to business events, both those generated by other WebSphere components and those generated by external business systems. In a sense, Process Server solutions are inherently event-driven internally, since each module is triggered by receipt of a *business object* – an event – typically from another module.

In addition, a Business State Machine component – a kind of BPEL process – is explicitly modeled as event-triggered transitions. In the InterChange Server of previous versions of WebSphere Business Integration, the same business object could represent different events depending on its "Verb" metadata element. A similar idea is carried over to version 6 business objects. Business State Machines are in some sense the v6 analog of the older InterChange Server "collaborations." Events are also integrated into Business Process components using the standard BPEL *Receive* and *Pick* activities.

Important sources of external business events in WebSphere Process Server include:

- **WebSphere Adapters.** One of the key differences between the new JCA 1.5-based adapters and the older JCA 1.0 is support of inbound event adapters. Events in an external business system such as SAP or Siebel are exposed to the business process as XML messages or business objects, where they trigger business processes, complete BPEL *Receive* or *Pick* activities, or trigger state transitions in a Business State Machine component. IBM also provides JMS-based event adapters.
- **WebSphere Message Broker.** Message Broker provides high-performance publish/subscribe brokering of business events received as JMS or MQ messages. Message flows in the broker support filtering, content-based routing, and message transformation (mediation). Message Broker output can be received by the business process as a business object event.

9.2 Exception Handling and Transaction Management

Exception handling within a BPEL process component is modeled using standard BPEL constructs, including fault handlers and compensation handlers. These handlers are connected to the BPEL process at any level, from an individual simple activity to a block of activities enclosed in a scope, to the entire BPEL process.

WebSphere Process Server offers full ACID transaction support for business processes, both for short-running (one transaction end to end) and long-running processes (multiple transactions). Transaction boundaries can be modified in WID to group multiple process steps in single transaction. The software also supports compensation of completed activities within a failed long-running transaction.

Exception management across process components is provided by WebSphere Process Server's Recovery Manager and Recovery Console. Failed instances of a business process will be detected and allow an administrator to manage their recovery from the Recovery Console.

10. Performance Management

Performance management is a major emphasis of the WebSphere BPM suite. Design of metrics, reports, and dashboards is done in WebSphere Business Modeler. The performance management runtime is called WebSphere Business Monitor.

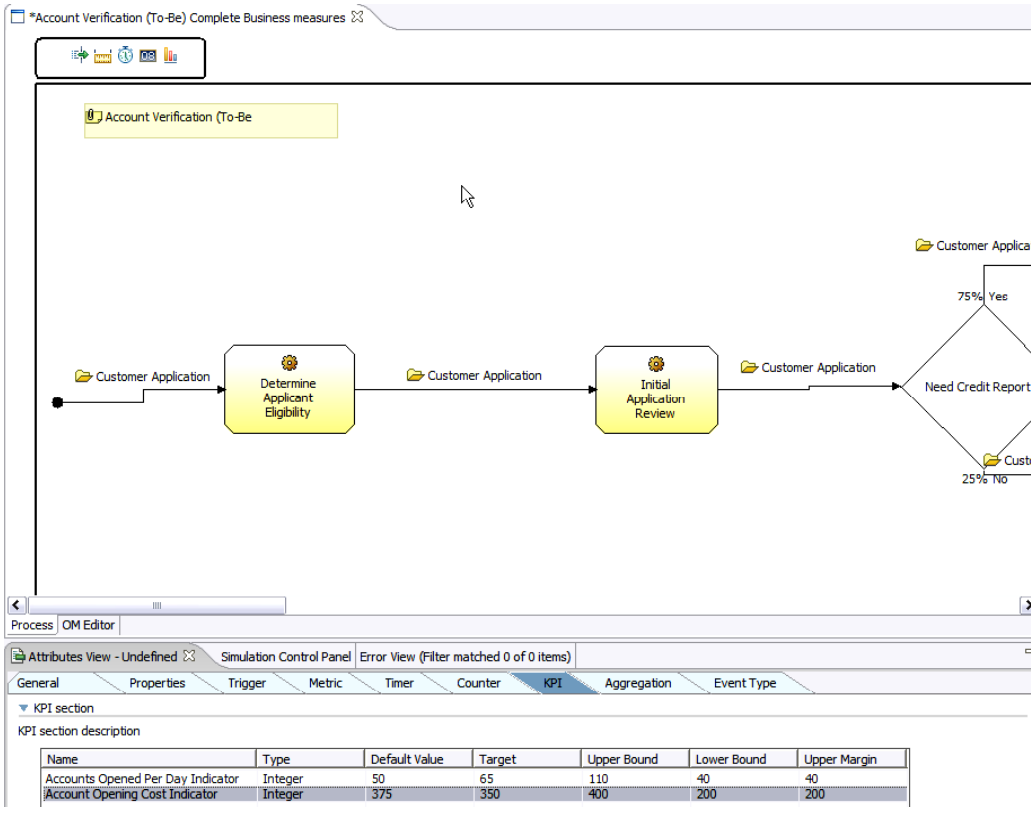


Figure 24. Business Measures Editor in WebSphere Business Modeler. Source: IBM

10.1 Business Measures Editor

Figure 24 illustrates the *Observation Model* in WebSphere Business Modeler, specified in the *Business Measures Editor*. This tool allows business analysts to design the metrics, KPIs,

counters, timers, triggers, and other performance management artifacts used at runtime by WebSphere Business Monitor. A KPI is a special kind of metric defined with a target value and range. When the calculated KPI value goes out of the target range, the performance management runtime issues a *situation event* that can initiate a notification, a business process, or some other action. KPIs and metrics are defined both at a business process and activity level of granularity.

Calculation of metrics, KPIs, counters, and timers is initiated by various triggers defined in the Business Measures Editor. Trigger events include when activities start or finish, when the state of an activity or process changes, when a specified metric or KPI changes, etc. Measures calculated on instances of the same Modeler business process are then *aggregated* (e.g., total, average, min and max) as specified in the Business Measures Editor (Figure 25). These specifications determine the OLAP schema of the measure in the WebSphere Monitor data store.

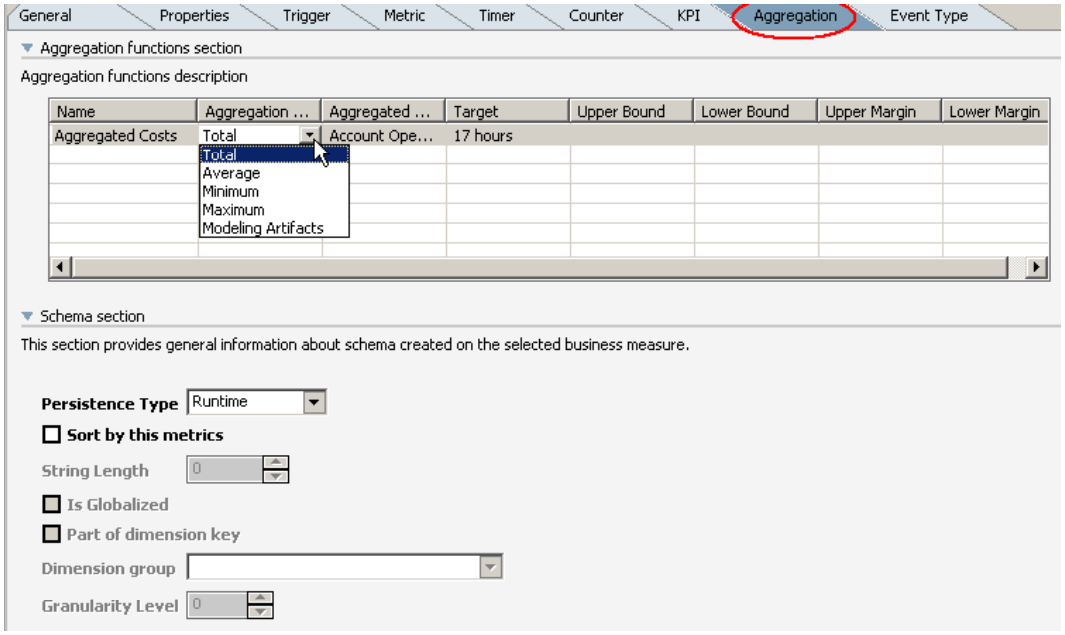


Figure 25. Business measure aggregation attributes in WebSphere Business Modeler.

10.2 Business Monitor Architecture

The Observation Model created in the Business Measures Editor is deployed to the WebSphere Business Monitor for execution. Monitor works by processing the *events* generated by Process Server automatically upon state changes of running instances of service components, a function of Process Server's Common Event Infrastructure (CEI). CEI provides 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). The CBE standard was submitted to the OASIS standards organization and IBM is hoping for a greater level of adoption by the industry.

As defined in the Observation Model, Monitor looks for specific events and uses them to calculate the business measures (Figure 26). Events are stored in Monitor's State database and replicated to other operational data stores used by Monitor dashboards, one for real-time and another for historical reporting. When metrics and KPIs exceed target levels, Monitor's Action Manager publishes *situation events* that can trigger other Process Server components or system management infrastructure such as IBM Tivoli.

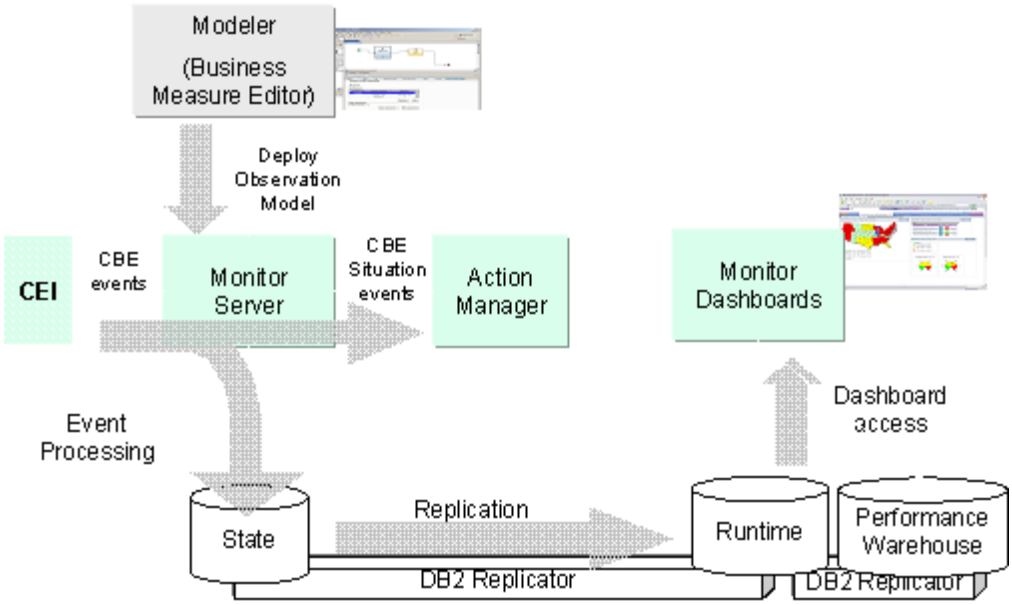


Figure 26. WebSphere Business Monitor relies on Process Server's Common Event Infrastructure. Source: IBM

10.3 Analytics and Dashboards

From the information in its data stores, Monitor calculates and displays the specified business measures in dashboards accessible through the WebSphere Portal. Monitor Dashboards (Figure 27) are based on DB2 AlphaBlox, business intelligence technology acquired by IBM in 2004. In addition to powerful analytics, they provide users with configurable wizards to define graphical views of KPIs, scorecards, alerts and prompted actions, tables and reports, and display them in web portlets.

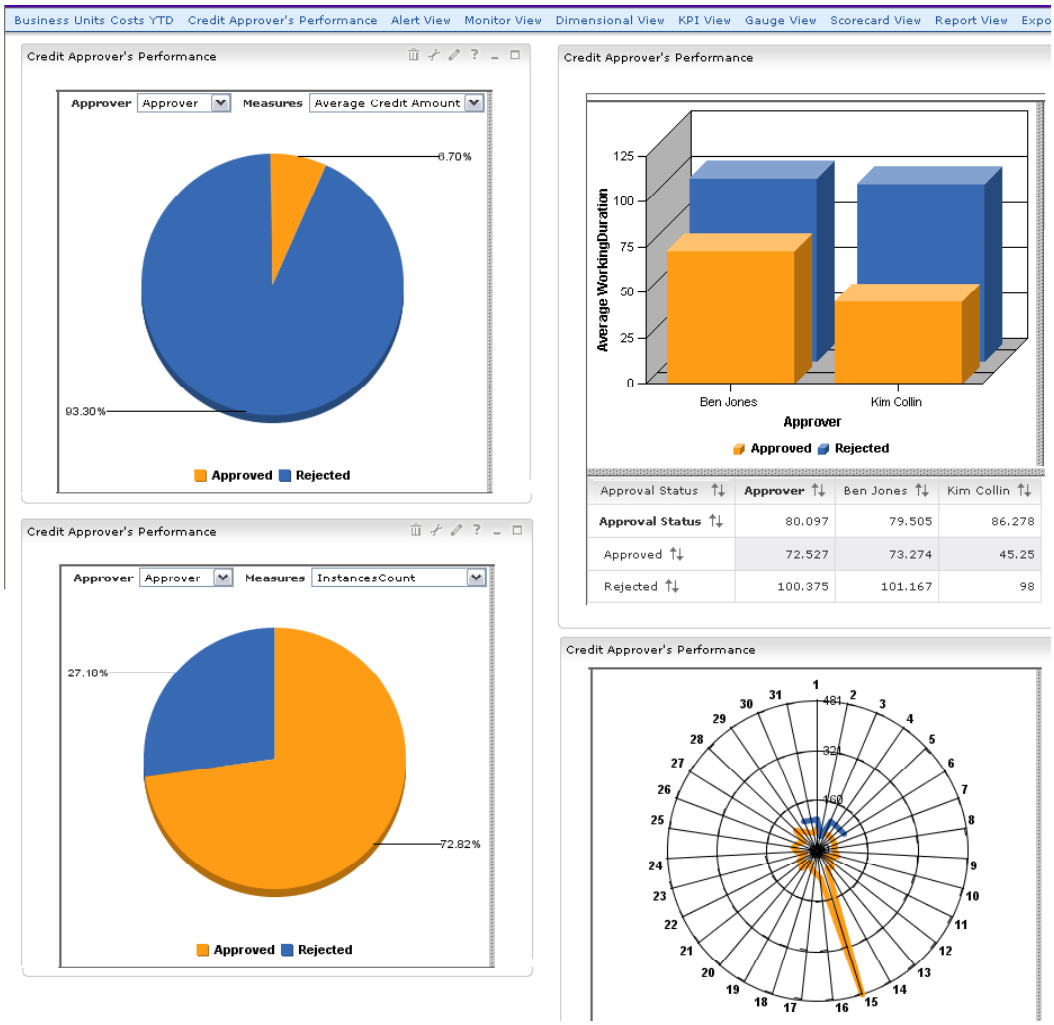


Figure 27. WebSphere Business Monitor displays charts, tables, scorecards, and alerts in web portlets. Source: IBM

11. Industry Solutions and Services

One of the key strengths of WebSphere Business Integration v5, IBM's previous BPM suite offering, was the long list of available vertical solution components implemented as InterChange Server collaborations and business objects. *Collaboration templates* represent business process fragments (in XML), typically system-to-system interactions. *Collaboration objects* represent the actual instances of the business processes (with inputs/outputs bound to connectors or other collaboration objects). IBM sells pre-packaged collaboration templates specifically tailored to common processes for Procurement, Customer Relationship Management, Financial and HR, Order Management, Life Insurance, Property and Casualty Insurance, Telecommunications, HIPAA Transaction, Healthcare, Consumer Products Item Synchronization, Retail Message Manager, and Product Information Management. In addition, users can build their own custom collaborations.

IBM will provide migration tools and services offerings to make much of this collaboration functionality available on WebSphere Process Server v6. The tools will convert collaboration objects into BPEL and web services artifacts. The tools could potentially be used to convert collaboration templates as well, although some additional development may be required. IBM

does not plan to move these pre-packaged collaboration templates to Process Server v6 as supported products. Instead, the IBM strategy is to encourage IBM Business Consulting Services (BCS) and business partners to build industry solutions on top of the WPS v6 platform. In addition, IBM's vertical industry teams have been chartered to move the appropriate assets from IBM's existing industry solutions and create some new vertical "SOA accelerators" on Process Server v6. Besides packaged and supported software products, SOA accelerators also may include collateral, service offerings, "SupportPacs," and other reusable assets.

In addition, IBM has mobilized its own extensive solution development and professional services resources in a major strategic initiative called *Business Innovation and Optimization* (BIO), which places WebSphere BPM Suite in the broader context of "business performance transformation." IBM describes BIO as "a methodology for delivering value" combining industry expertise, best practices, and software. The methodology is based on three basic steps:

1. Identify and understand the key business *goals* and associated *processes* to achieve them.
2. Create *KPIs* to measure progress toward those goals.
3. Use *alerts* to track and manage performance, analyze results, and continuously improve the business.

WebSphere Business Modeler and Monitor are critical to the methodology, and WebSphere Process Server also plays a central role. In addition to WebSphere BPM Suite components, BIO building blocks also include IBM products such as DB2 AlphaBlox for deep analytics and BAM, Workplace collaborative solutions, and Tivoli business systems management technology, as well as third party tools and components.

12. Analysis

12.1 Overall Assessment

IBM has achieved the nifty trick of marrying a business-oriented BPM sensibility, emphasizing modeling and performance management, with the same hard-core developer-oriented toolset and execution platform it is using in its strategic push into SOA based on the WebSphere platform. While other BPMS vendors give lip service to BPEL – the widely proclaimed "standard" for service-oriented process definition – IBM may be the only one that *actually uses it* for the end-to-end process. But instead of trying to pass off a graphical BPEL designer as a business modeling tool, IBM starts with a real modeling and simulation tool for business analysts and generates BPEL (and other executable components) from it under the covers. The Business Modeler also defines the KPIs used for analyzing executable performance, allowing true round-trip optimization.

The handoff between the business analyst-oriented Modeler and Monitor and the programmer-oriented WID and Rational tools is "clean" and largely automated. Business analysts view the process from the lens of Modeler and Monitor, while developers view it from the lens of service components and assemblies, with drilldown to BPEL, Java, or other implementation technologies. A business analyst might be hard pressed to even recognize his own model as it is imported into WID components, but the two views are in fact based on the same underlying model, and run in the same Eclipse environment.

This approach to BPM is not for everyone. Implementation is unashamedly oriented to professional developers, but based on the new SOA-based composite application paradigm: Once the service components have been created by Java programmers, they can be orchestrated and assembled in WID by developers with more widely available skills, and far more rapidly than with old-style Java programming. This style of BPM solution development is more familiar, and probably more attractive, to many IT shops than the point-click-and-deploy style favored by many BPMS vendors.

Also, while IBM's offering is standards-based it is most decidedly *not* platform-independent. It relies on the special capabilities and programming model extensions of the WebSphere v6 platform. In fact, WebSphere Process Server *is* the service orchestration layer of the WebSphere platform. In the evolution of SOA adoption, most companies are still in the stage of creating reusable services, either by coding them in Java or wrapping existing applications in service-oriented adapters. IBM provides the platform and tools to build those services, to interconnect them, and to orchestrate them in composite applications. In that sense, the Rational tools, WID, and Process Server are crucial to the overall SOA push at IBM. For BPM, representing just one segment of the SOA market, IBM wraps those core implementation technologies with Modeler and Monitor.

The WebSphere BPM Suite now combines all essential elements of a BPMS, from modeling and simulation to workflow and integration, business rules, and performance analytics, within a unified environment all using IBM technology. None of it is begged or borrowed from third parties (although IBM is happy to help customers integrate third party components through its services arm). All that is a major plus. While integrated, however, the suite has many moving parts, and development is more complex than with many other tools.

As a result, while you could certainly use it for basic workflow, it might not be worth the programming effort for a simple implementation. The sweet spot for WebSphere BPM Suite is really heavily automated straight-through processing and production workflow, high-volume

processes driven by rules and events and requiring sophisticated modeling and performance management.

It's also important to realize that WebSphere BPM Suite v6 is still a new product with a new architecture, and has not yet had time to fully integrate the wealth of technology riches from the farflung corners of IBM, things like DB2 Content Manager (a scalable ECM repository) or IBM Workplace (a team collaboration environment), or even replicate the long list of prebuilt industry solutions available on the old InterChange Server architecture. When that day comes, IBM will likely be a formidable BPMS contender in every use case category.

12.2 Use Case Fit

12.2.1 Basic Workflow

Fit: ● ● ●

Checklist:

- ***Rich forms user interface.*** Yes, but it requires Java and JSP programming. IBM provides a full set of APIs for task UI development, and allows integration of e-forms.
- ***Process design and maintenance by non-programmers.*** Not really. IBM's design paradigm focuses non-programmers on business modeling and performance management.
- ***Easy to deploy and maintain.*** Yes, relative to traditional J2EE development.

12.2.2 Content Lifecycle

Fit: ● ●

Checklist:

- ***Support for document attachments and viewers.*** These can be integrated, but not out-of-the-box.
- ***Support for content management library services: check-in/check-out, versioning, metadata search.*** IBM provides content lifecycle capabilities in both DB2 Content Manager and IBM Workplace, but these have not yet been integrated with WebSphere BPM Suite.
- ***Support for scalable ECM repositories.*** IBM DB2 Content Manager is such a repository, but not yet integrated.

12.2.3 Complex Collaborative

Fit: ● ● ● ◗

Checklist:

- ***Support for collaborative document review and discussion.*** WebSphere Business Monitor integration with collaborative Workplace solutions – Workplace for

Business Strategy and Execution and Workplace for Business Controls and Reporting – appears to be more advanced.

- ***Support for unstructured/ad hoc flow and offline work.*** See above.
- ***Integration of online team rooms.*** See above.

12.2.4 Case Management

Fit: ● ● ●

Checklist:

- ***Electronic case folder of independent work objects***. Can be supported with custom programming, but not provided out-of-the-box.
- ***Ability to add case objects and flows at runtime.*** See above.
- ***Content management integration.*** Not yet.

12.2.5 Production Workflow

Fit: ● ● ● ●

Checklist:

- ***Support for shared queues and rule-driven task assignment.*** Yes, a strength. IBM has long experience in production workflow.
- ***Performance optimization through simulation, analytics, and real-time escalation.*** Another strength, with optimization metrics built into Modeler and Monitor.
- ***Business rules.*** IBM provides a native rule engine and integrates with third party as well.
- ***High-performance document retrieval/parsing.*** IBM has these capabilities in DB2 Content Manager, but not well integrated with WebSphere BPM Suite.

12.2.6 Transactional/STP

Fit: ● ● ● ● ●

Checklist:

- ***Rich integration infrastructure, including adapters and enterprise service bus.*** Yes. IBM is the undisputed leader here.
- ***Complex business objects, data transformation, and business rules.*** Yes, very rich features here.
- ***Comprehensive event management and automated exception-handling.*** Again, a key strength. IBM is trying to promote industry adoption of its Common Base Event standard.

- ***Industry solution templates with prebuilt objects, transformations, adapters, protocols, performance metrics and reports corresponding to industry standards and best practices.*** This has been a strength of previous-generation BPM from IBM, some of which will be migratable to WebSphere BPM v6. However, IBM's current posture appears to emphasize partner-built solutions and Business Consulting Services more than out-of-the-box solution components.

12.3 Process Lifecycle Fit

WebSphere BPM Suite articulates a clean handoff between two distinct groups of users. Modeler and Monitor are assumed to be used by business analysts, and describe the process using familiar flowcharting constructs. WID and the Rational tools are intended for developers, and describe processes in terms of interacting service components, consistent with SOA design principles but not completely aligned with BPM ideas and terminology. Integration between the business- and IT-oriented design components keeps everything "in sync," but the lifecycle is based on handoffs rather than collaborative design.

About the Author

Dr. Bruce Silver is an independent industry analyst and consultant specializing in BPM and content management technology. He has advised both users and vendors of BPMS technology for many years, as VP at the analyst firm BIS Strategic Decisions (which became Giga Information Group, now Forrester Research) and later through his own firm, Bruce Silver Associates. He currently writes the monthly *BPMS Watch* column on bpminstitute.org and the *Change Agent* column for Intelligent Enterprise magazine. He is also the BPMS Technology Track chair at the Brainstorm BPM Conference series and a frequent speaker on BPMS product selection. In addition to the 2006 BPMS Report, Bruce Silver Associates offers numerous free white papers on BPMS and content management technology, available from www.brsilver.com.

Comments on this report as well as requests for future reports are welcome. Write to bruce@brsilver.com.