Key WebSphere Service Registry and Repository scenarios in business process management.

By Naveen Sachdeva, Worldwide Technical Sales Leader, IBM, and Marc-Thomas Schmidt, Distinguished Engineer, IBM

Contents

Overview

This paper describes key scenarios for IBM WebSphere® Service Registry and Repository in the business process management (BPM) space. We will discuss how WebSphere Service Registry and Repository can add value across the BPM life cycle of Model, Assemble, Manage and Govern phases to promote reuse and enhance business flexibility. We will also discuss the importance of governance and how WebSphere Service Registry and Repository enables governance throughout the BPM life cycle. In a follow-on paper, we will describe how IBM WebSphere Business Services Fabric enables business process flexibility and responsiveness using composite business applications and business-oriented policies, and leverages WebSphere Service Registry and Repository along with other WebSphere products to provide a comprehensive business process management solution.

BPM in a nutshell

Like any other asset or service, a business process goes through the life cycle phases of Model, Assemble, Deploy, Manage and then back to Model, and IBM BPM solutions support this continuous, end-to-end, business process life cycle.

In the following sections, we describe the full spectrum of tasks in the BPM life cycle. Not all of these steps are implemented in all BPM scenarios (for example, the Model phase might be skipped). The IBM BPM solution components support the following major activities (for details, refer to the IBM Redbook "Business process management: Modeling through monitoring using WebSphere V6.0.2 products" at http://www.redbooks.ibm.com/redpieces/abstracts/sg247148.html). Business analysts can use IBM WebSphere Business Modeler to sketch out new business processes reflecting the needs of their enterprise or modify processes they defined earlier. They identify the main activities to be performed in the process, assign responsibilities for those activities, identify services the enterprise has that can implement the activities, and describe the process flow between activities. Business analysts then identify key performance indicators (KPIs) to be used later to measure performance of the process. They simulate the process, try out alternatives and then hand over the high-level business process design to people who can translate it into implementation.

Integration developers and application developers handle the next step in the life cycle of a BPM application. They use IBM WebSphere Integration Developer (if using IBM WebSphere Process Server as the process server) or IBM FileNet® Process Designer (if using IBM FileNet Process Engine as the process server) to translate business process designs into executable process models (described in Business Process Execution Language [BPEL]/XML Process Definition Language [XPDL]) with actual services implementing the activities and instrumentation of the resulting application to provide input for measuring KPIs. Part of this work is to deal with potential mismatches between requirements of a process on services and real existing services — enterprise service bus (ESB) mediations are used to handle that. They build in points-of-variability into the resulting application, which allows deployers and administrators to tweak behavior of the application to a degree.

During the deployment of an application, service metadata is refined or defined that determine the specific behavior of the application. The process execution engine (WebSphere Process Server or FileNet Process Engine) collaborates with the ESB — such as IBM WebSphere Message Broker, IBM WebSphere Enterprise Service Bus or IBM WebSphere DataPower® SOA Appliance XI50 — running the process and delegating execution of activities to the most suitable service, according to the service metadata defined.

The KPI instrumentation of the composite process application provides events that can be used to monitor and analyze the performance of the process, for example in IBM WebSphere Business Monitor. In addition, service monitoring and management products, such as IBM Tivoli® Composite Application Manager for SOA, record interactions and the status of endpoints involved in those interactions.

WebSphere Service Registry and Repository in a nutshell

WebSphere Service Registry and Repository is a new addition to IBM BPM Portfolio. It is a master store for all the deployed services, related artifacts and the associated metadata (properties, relationships and classifications). It provides value throughout the BPM life cycle through the five key capabilities of *Publish, Find, Enrich, Manage* and *Govern* (see Figure 1). For more information about WebSphere Service Registry and Repository, see the IBM information center Web site at http://publib.boulder.ibm.com/infocenter/sr/v6r1/index.jsp.
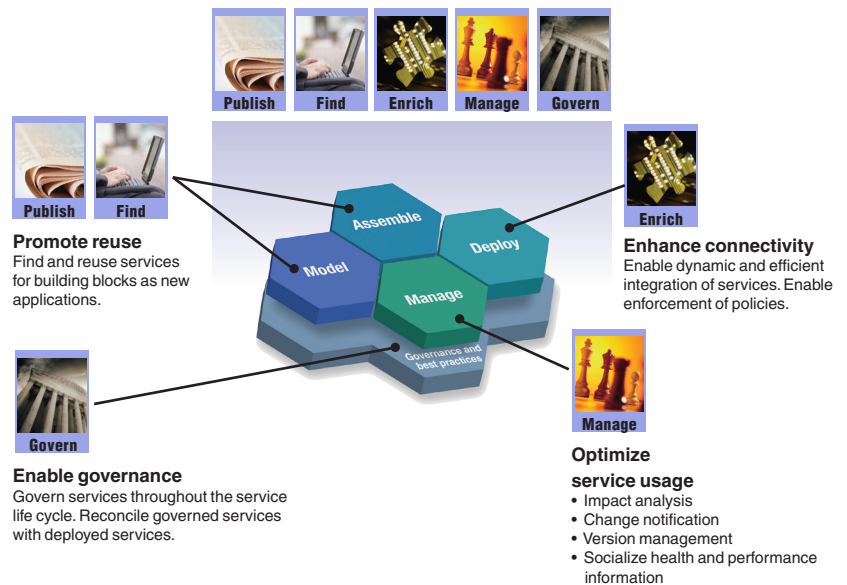

Promote reuse
Find and reuse services for building blocks as new applications.

Enhance connectivity
Enable dynamic and efficient integration of services. Enable enforcement of policies.

Enable governance
Govern services throughout the service life cycle. Reconcile governed services with deployed services.

Optimize service usage
• Impact analysis
• Change notification
• Version management
• Socialize health and performance information

Figure 1. WebSphere Service Registry and Repository provides value throughout the BPM life cycle.

Publish refers to the capability to publish services, related artifacts and the associated metadata. *Find* refers to the capability to find the services using the metadata. *Enrich* refers to the capability to enhance the run time and, as a result, business flexibility, through efficient APIs for runtime lookup. *Manage* refers to the capability to manage services (and related entities, such as polices captured using WS-Policy document) through impact analysis, change notification, version management and automatic update of runtime metadata (status and average response time) through integration with the service monitoring products. *Govern* refers to the capability to control the service and related metadata throughout the service life cycle through fine-grained access control, enforcement of metadata validation policies and automatic discovery of deployed services.

A day in the life

So let's illustrate the use case for WebSphere Service Registry and Repository by following a business process from: the moment when it is initially sketched out in the Model phase; to the point when it is implemented in the Assemble phase; to the phase when it is deployed and implemented; to the phase when it is monitored; and then back again to modifications of the process resulting from observed bottlenecks.

Model: Process creation

A business analyst uses a business-process analysis tool, such as IBM WebSphere Business Modeler, to model and simulate a new business process. WebSphere Business Modeler has built-in integration with WebSphere Service Registry and Repository, which allows the business analyst to find existing services that can be used to automate business tasks in the business process, download the corresponding interface Web Services Description Language (WSDL) files and plug the services into the business process. The business analyst (with the help of the integration developer) can then export the business process (as well as the corresponding XML Schema Definition [XSD] and WSDL files) in the BPEL or the XPDL format, as shown in Figure 2, and put them in a Source Control Management (SCM) system, such as IBM Rational® ClearCase®.

At this point, the analyst could make the WebSphere Service Registry and Repository user community aware of the envisioned dependency of the new process on the existing service by creating a process entity in WebSphere Service Registry and Repository, establishing a relationship to the existing service, making the entity governable and indicating that its life-cycle state is *under construction*. The analyst could also capture the link to the process artifacts in WebSphere Service Registry and Repository as a property on the process entity and transition the process entity to the Assemble phase. WebSphere Service Registry and Repository can verify that the link to process artifacts has been specified and then send a notification to an integration developer.
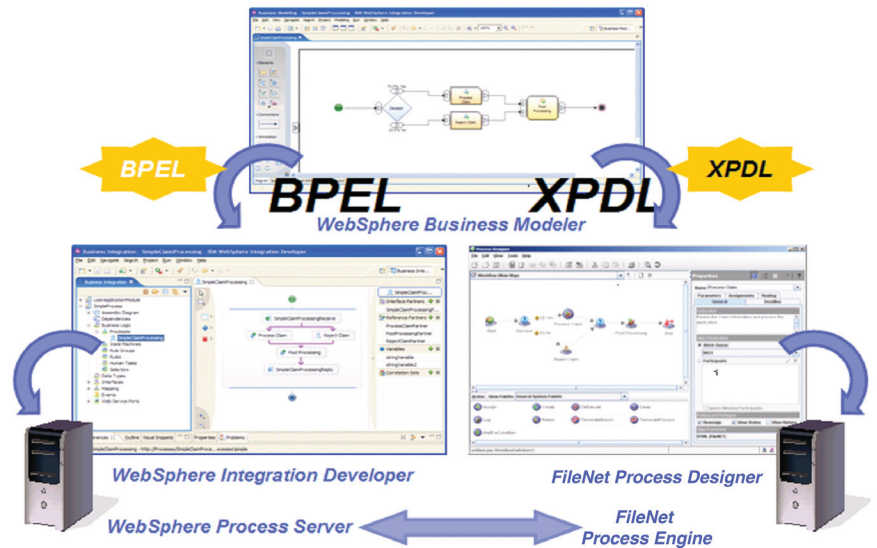

Figure 2. The business analyst can export the business process in the BPEL or the XPDL format.

Assemble: Process assembly

Depending on the type of process, the integration developer will use either
WebSphere Integration Developer or FileNet Process Designer to capture the
technical details and integrate the process with the services. In either tool, the
integration developer can retrieve the process artifacts, identify the services
that the business analyst wanted to reuse, download the corresponding service
WSDL files from WebSphere Service Registry and Repository, plug them into the
process, expose the process as a service and publish the service in WebSphere
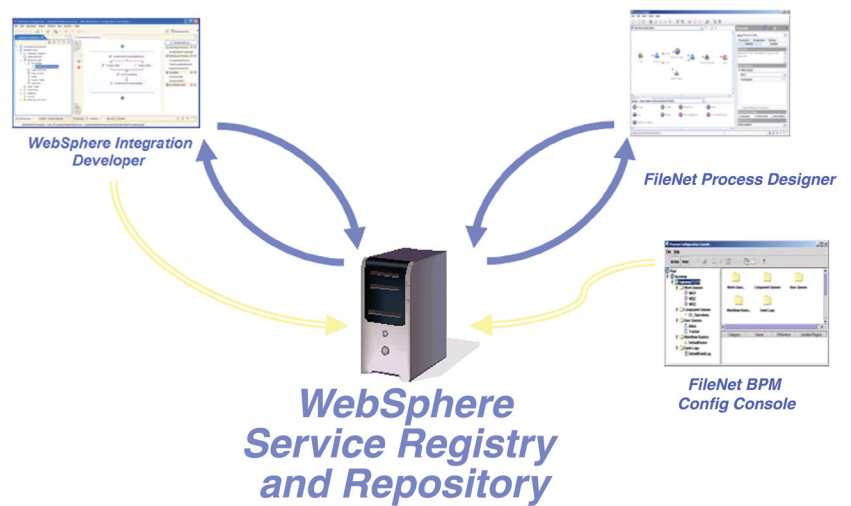Service Registry and Repository (see Figure 3).

Figure 3. The integration developer can use either WebSphere Integration Developer
or FileNet Process Designer to retrieve or publish services from or to WebSphere Service Registry
and Repository.

In WebSphere Integration Developer, the integration developer can also choose to export the project as a service component architecture (SCA) module and publish it in WebSphere Service Registry and Repository. The solution that the developer builds could also include mediation modules that facilitate mapping between the service requirements in their processes and the existing services. The integration developer can choose to create a proxy service (in this case, a mediation flow) with dynamic endpoint lookup using the WebSphere Service Registry and Repository specific primitive called EndpointLookup and binding, and then link the process to the proxy service instead of to the actual service (see Figure 4).
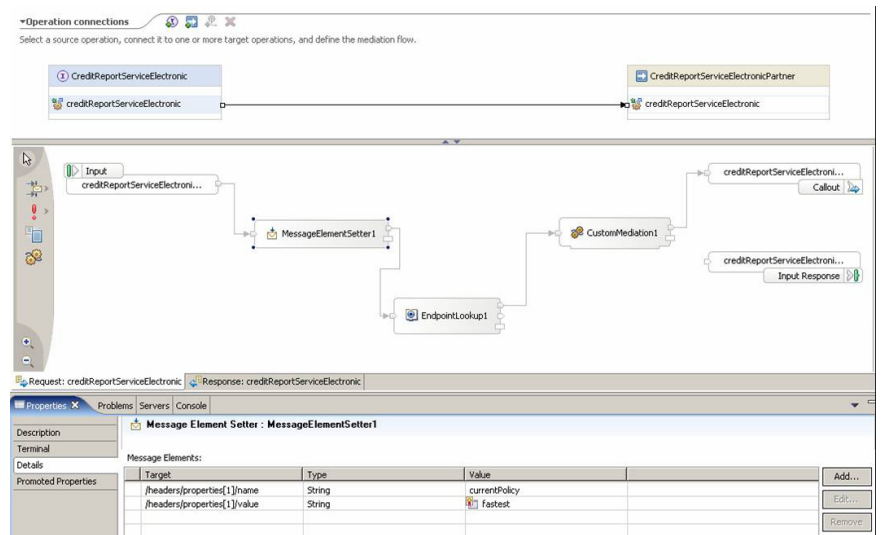

Figure 4. The integration developer can create dynamic mediation flows using WebSphere Integration Developer.

This allows the service request to be dynamically routed to a new endpoint without making any changes to the code (for example, routing to a failover endpoint or routing to a new, backward-compatible version of the service). Mediations can consider service interfaces and metadata associated with corresponding endpoints, such as a property reporting on the status of the endpoint, in their decision-making process.

Deploy: Dynamic processes using mediations
WebSphere Service Registry and Repository content can play a major role in the execution of deployed process services if the composite application was configured to make use of that information, as indicated in the previous section. The ESB layer is key to making that happen. Endpoints that were not available at design and deploy time can become targets of service requests issued by the business process through dynamic lookup. Endpoint selection can be changed by updating metadata about services in WebSphere Service Registry and Repository without having to go through a development cycle. Services that are not meeting response-time and SLA criteria can be excluded from the selection. This significantly increases the flexibility of the process and allows adjustments to changes in the execution environment.

Manage: Monitoring processes

During the Manage phase, business metrics for business processes as well as the services they use, are monitored by WebSphere Business Monitor. Examples of these metrics include the number of new accounts each month, average account creation time and so on. IT metrics, such as the status of the credit check service and average response time for the credit check service, are monitored by IBM Tivoli Composite Application Manager for SOA, respectively. If these metrics breach the predefined limits, the monitoring products can raise the situation events so that corrective actions can be taken by automatically routing the requests to faster services. WebSphere Service Registry and Repository enables this by capturing the situation events as properties on the corresponding processes and services, which can be discovered and acted upon by the ESB dynamically (as shown in Figure 5).
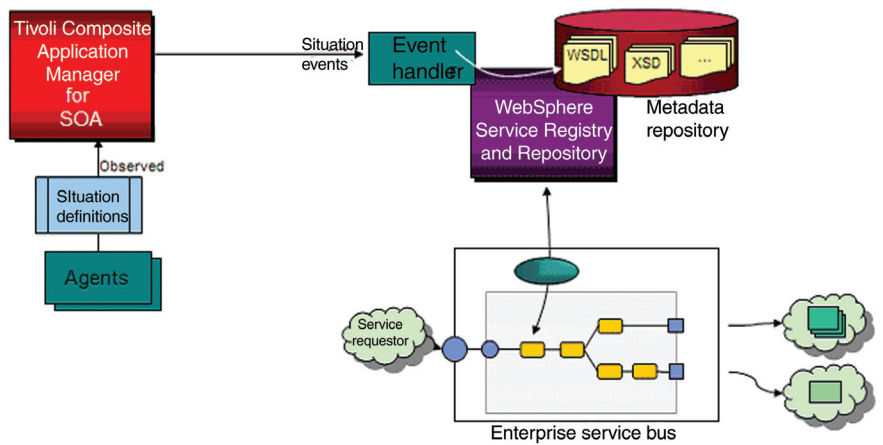

Figure 5: Metadata in WebSphere Service Registry and Repository can be updated by the monitoring products.

Govern: Ensuring compliance

So far, we discussed how WebSphere Service Registry and Repository can add value across the BPM life cycle, for example how WebSphere Service Registry and Repository can promote reuse and enhance business flexibility. But in order to realize this value, the governance needs to be in place as indicated by the Governance and Processes layer below the BPM life-cycle layer. For example, before the business analyst can find the appropriate services, the services need to be decorated with the appropriate metadata for proper identification. How do you ensure that the metadata is added to the services as they go through the service life cycle? How do you ensure that the business analyst can only see those services that have been approved for reuse? How do you ensure that the integration developer is notified if the particular version of the service that is being used in the process is deprecated? How do you ensure that a service version is not retired if it is still being used by another service or process?

Here also WebSphere Service Registry and Repository adds value through these key capabilities:

- Classification, *which allows the business analysts to define business domains and then use them to classify processes and services so that they can be easily discovered.*
- Fine-grained access control, *which allows the administrator to define access control based on the roles and entity metadata. For example, the administrator can define that BusinessUnit1_BusinessAnalyst role can only retrieve those services that are classified as "Published" and "BusinessUnit1_Service" or "Shared_Service."*
- Life-cycle management, *which allows the administrator to capture the states and transitions that the services or the related entities should go through, as well as the roles that can perform those transitions. For example, the administrator can add multiple review states between the Assemble and Deploy phases and set up the access control so that the integration developer can transition the service from Assemble to Review. But Review to Deploy can only be performed by the reviewer or governance manager, who can manually review the process or service metadata before transitioning the process or service to the deploy phase.*
- Validation framework, *which allows the administrator to define the metadata validation policies so that the metadata can be automatically validated as the process or service is transitioned from one state to another. For example, the administrator can set up an IT policy that will prevent a service from transitioning to the retired state if a process has a dependency on this service.*
- Notification framework, *which allows the administrator to set up the subscriptions so that the interested parties can be notified automatically. For example, the administrator can create the subscriptions for the services that are used in the process so that the integration developer will receive an e-mail if the services are deprecated.*

And what happened the next day

As we mentioned earlier, if the business or IT metrics don't satisfy business goals, it might be possible to take the corrective actions dynamically. But that is not always feasible. If there are bottlenecks in the business process, the business analysts need to go back to the drawing board (in this case, IBM WebSphere Business Modeler) to identify those bottlenecks through simulations. Integration between WebSphere Business Monitor and WebSphere Business Modeler allows the business analysts to use the information collected by WebSphere Business Monitor for simulation and create a new version of the business process model. WebSphere Service Registry and Repository supports coexistence of multiple versions of processes and services, which can be taken through the life cycle independently.

If the process interface has changed, the business analyst or the integration developer can perform the impact analysis on the process (see Figure 6) and identify the consumers who might be affected and notify them.
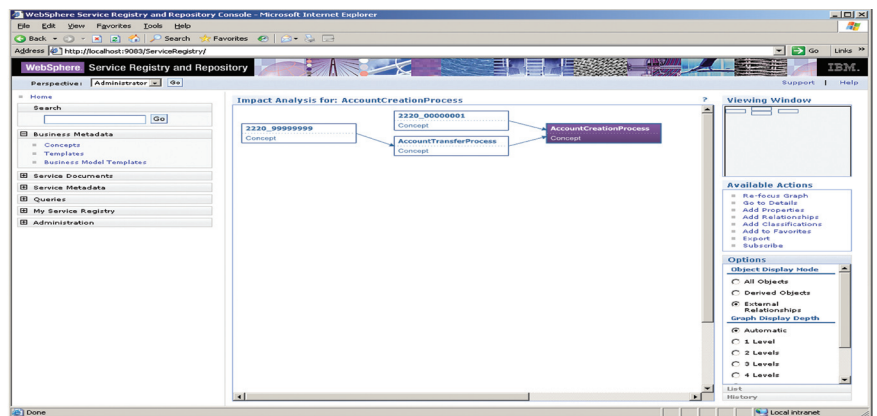

Figure 6. Impact analysis using WebSphere Service Registry and Repository

Switching over from the older version to the newer version of the process can be controlled in WebSphere Process Server through the process-start and the process-expiration dates. Consumers of the older process version can be notified of this change by updating the process-expiration date in WebSphere Service Registry and Repository, which will automatically send e-mail notification to the process consumers who have subscribed for process updates.

Summary
WebSphere Service Registry and Repository adds value across the BPM life cycle as well as the underlying governance layer through the five key capabilities of Publish, Find, Enrich, Manage and Govern. It allows the business analyst and the integration developer to publish and find services. It allows the mediation flows, created by the integration developer, to retrieve the service metadata at run time and thereby adapt dynamically. WebSphere Service Registry and Repository can also receive situation events from the monitoring products and automatically update the process or service metadata to be retrieved by the mediation flows in the ESB layer. In addition, WebSphere Service Registry and Repository provides governance capabilities that include fine-grained access control, impact analysis, life-cycle management, business modeling, and validation and notification frameworks. In a new white paper, we will describe scenarios demonstrating WebSphere Business Services Fabric capabilities in managing business services and related metadata, and how it leverages WebSphere Service Registry and Repository to provide a comprehensive business process management solution to our clients.

For more information

To learn more about WebSphere Service Registry and Repository, visit:
ibm.com/software/integration/wsrr

Additional resources

IBM Redbooks publication, WebSphere Service Registry and Repository
Handbook: http://www.redbooks.ibm.com/abstracts/SG247386.html?Open

Service oriented architecture (SOA): **ibm.com**/soa

SOA governance: **ibm.com**/soa/gov

BPM enabled by SOA: http://www-304.ibm.com/jct03002c/software/info/bpmsoa

IBM FileNet Content Manager: http://www-306.ibm.com/software/data/content-management/filenet-content-manager

IBM WebSphere Process Server: http://www-304.ibm.com/jct03001c/software/integration/wps/

IBM WebSphere Business Services Fabric: http://www-306.ibm.com/software/integration/wbsf/

IBM

WSW14010-USEN-00