



WebSphere software

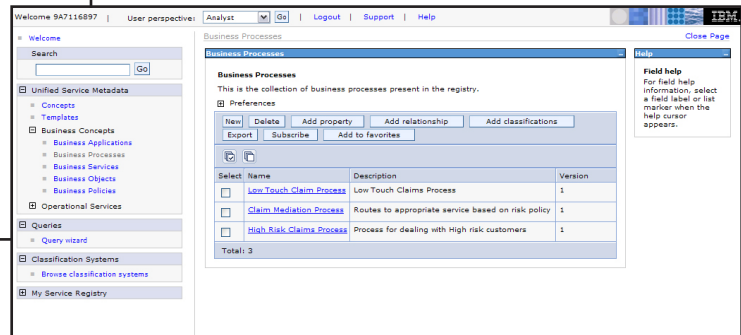
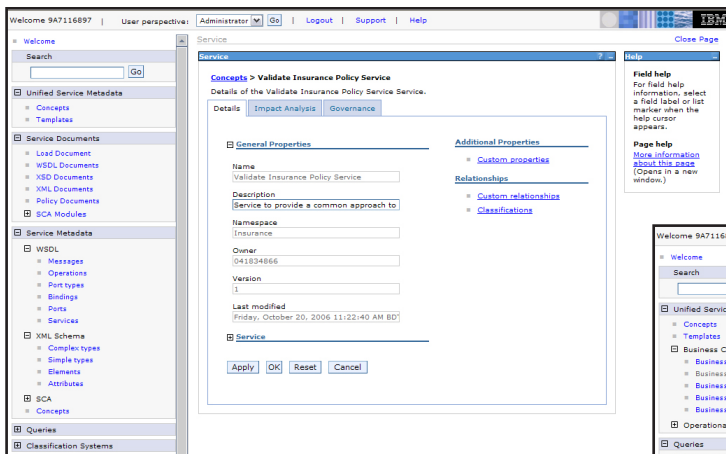
IBM WebSphere Service Registry and Repository, Version 6.0

Highlights

- **Provide clear visibility into service associations and relationships while encouraging reuse of services to avoid duplication and reduce cost**
- **Use other standard registries and repositories for a unified view across a variety of service-information sources**
- **Enrich connectivity with dynamic service selection and binding at run time**
- **Unlock the value of mainframe applications**
- **Enable governance of services throughout the SOA life cycle**
- **Help institute best practices and enforce policies in your SOA deployments**
- **Help ensure interoperability of services with a registry and repository built on open standards**
- **Adopt SOA incrementally by using the product to start your SOA small and grow as needed**

Service oriented architecture (SOA) has the potential to drive business flexibility, performance and innovation by enabling you to better align your information technologies to your business objectives. Making the most of this potential depends on how well you govern and manage the services in your SOA.

IBM WebSphere® Service Registry and Repository, Version 6.0 is an industrial-strength tool that helps you achieve more business value from your SOA by enabling better management and governance of your services. Through its robust registry and repository capabilities and its tight integration with IBM SOA Foundation, WebSphere Service Registry and Repository can be an essential foundational component of your SOA implementation.



IBM WebSphere Service Registry and Repository enables you to store, access and manage information about the services in your SOA.

Store, access and manage information to support a successful SOA

WebSphere Service Registry and Repository enables you to store, access and manage information about the services (commonly referred to as *service metadata*) in your SOA. You can use this information to select, invoke, govern and reuse services as part of a successful SOA.

With WebSphere Service Registry and Repository, you can store information about services in your SOA, or in other organizations' SOAs, that you already use, that you plan to use or that you want to be aware of. This capability helps make your SOA deployment more dynamic and more adaptable to changing business conditions.

WebSphere Service Registry and Repository is a single product that integrates:

- *A service registry that contains information about services, such as the service interfaces, its operations and parameters.*
- *A metadata repository that has the robust framework and extensibility to suit the diverse nature of service usage.*

The unique capabilities of WebSphere Service Registry and Repository enable it to be a critical deployment component of SOA projects. These capabilities include:

- *Publish and find capabilities for greater visibility and reuse*
- *Enrich capabilities for dynamic service interactions*
- *Manage capabilities for policies, classifications, versioning and impact analysis*
- *Governance capabilities to support services throughout their life cycle, helping to ensure that they maintain their vitality and usefulness to your business*

Publish and find for greater reuse

The publish and find capabilities of WebSphere Service Registry and Repository promote asset reuse in SOA projects by providing greater visibility of and easier access to existing services. These capabilities also expose redundant or inefficient services.

For example, when a request for a service need is approved, WebSphere Service Registry and Repository searches to determine if the service is available. If a service exists, the service owner is contacted to approve the reuse

of the service. If the service requires modification, the owner can choose whether to permit the alteration after performing an impact analysis with the metadata in the registry-repository. If the alteration is approved, a new version of the service is published to WebSphere Service Registry and Repository, and a new owner can be designated to support and maintain it. However, if the service does not exist, a new service-development request is initiated and published in WebSphere Service Registry and Repository, and the community is informed to prevent duplicate efforts.

Enriching connectivity in your SOA

WebSphere Service Registry and Repository helps improve the agility of your SOA by identifying metadata users and notifying them when changes occur. The product's role-based (such as an administrator, developer, architect or analyst) access contributes to proper governance of your SOA. For runtime environments, the service registry and repository enables dynamic, efficient and security-rich access to services information by enabling service end-point selection, service availability management and policy enforcement.

For service end-point selection, an enterprise service bus (ESB) mediation locates requestor metadata and searches WebSphere Service Registry and Repository for candidate provider end points. The mediation applies a selection algorithm and routes the request to the selected end point. However, if the provider end point doesn't respond because of a failure, the ESB mediation can search WebSphere Service Registry and Repository for other services that might meet requirements and are approved for use. For policy enforcement, a policy-infrastructure component uses policies stored in WebSphere Service Registry and Repository, and the request is either forwarded or rejected based on these policies.

Help maximize service efficiency and resilience

Management capabilities available with WebSphere Service Registry and Repository enable you to manage service metadata information, as well as service interactions, dependencies, and redundancies. You can classify services into meaningful groupings based on business objectives, manage policies for service usage and monitor how services are changed and versioned. You can also analyze service usage, history and business impact to promote and encourage optimal services usage.

For example, after a service is developed and deployed, WebSphere Service Registry and Repository shares service data with other operational data stores. You can determine how services are going to be used and how they interact — and indicate these parameters in the repository. Then, WebSphere Service Registry and Repository can keep you informed if services perform outside these parameters, enabling you to more effectively monitor and manage quality-of-service objectives of these services.

Manage services throughout their life cycles

With WebSphere Service Registry and Repository, you can enable governance of services throughout their life cycles. And because these capabilities span the entire life cycles of services, you can manage changes to services and create policies for publishing, using and retiring them.

For example, when you develop a service, its artifacts are stored in WebSphere Service Registry and Repository. Governance processes then promote the service from testing to production, and this is recorded in WebSphere Service Registry and Repository. After the service has been deployed and used, governance processes can determine when the service is no longer needed, and retire it, without affecting subscribers. Metadata in WebSphere Service Registry and Repository is used to assess the impact of changes to services.

Playing a vital role in the SOA life cycle

WebSphere Service Registry and Repository plays a major role in each stage of the SOA life cycle.

Model

During the development stages of the service life cycle, WebSphere Service Registry and Repository is used to locate the copies of record for candidate service-interaction end points or mediating intermediaries and policies governing the interactions, as well as service taxonomies, vocabularies and XML schemas.

Assemble

WebSphere Service Registry and Repository is also used to assemble solutions from new or existing components to create new composite applications. Integration developers can search for service end-point information in WebSphere Service Registry and Repository to model the mediations necessary to facilitate interactions between services.

Deploy

After the service is deployed, runtime environments, such as an ESB, can discover service end-point information stored in WebSphere Service Registry and Repository. This capability enables dynamic end-point selection and binding at run time. You can also use WebSphere Service Registry and Repository to dynamically retrieve and enforce the policies that are in effect for a service interaction in the areas of logging, filtering, data transformation or routing. WebSphere Service Registry and Repository stores the policy definitions, as well as information about which service-interaction end points that they can be applied to.

Manage

Operational management and resilience within the SOA is enhanced by sharing the service metadata that exists in WebSphere Service Registry and Repository with operational data stores, enabling management and monitoring dashboards to present a more-comprehensive view of the managed service environment. Summary information about service performance can be fed back into the WebSphere Service Registry and Repository and used by the runtime environment to affect the selection of the best-fit provider.

A range of SOA governance capabilities

WebSphere Service Registry and Repository plays an enabler role for SOA governance by providing:

- *Control of visibility over and access to service metadata for sharing and reuse.*
- *Support for the tracking of service metadata as it makes its way through its governed life cycle, including approvals, deprecation and retirement, in development, test, staging and production environments.*
- *The ability to perform synchronous and asynchronous user-defined validation of state transitions.*
- *An impact-analysis function to assist in the assessment of the impact of a change. WebSphere Service Registry and Repository captures many dependencies automatically.*
- *Change notifications through user-defined notification schemes, basic Java™ Message Service (JMS) publication of events providing basic information about what happened, as well as an e-mail-based notification feature.*

Federation with other SOA repositories

WebSphere Service Registry and Repository federates with other SOA repositories to enable governance and management of the service life cycle. At the model and assemble phase, WebSphere Service Registry and Repository is complemented by repositories that specialize in managing SOA development artifacts. For example, development artifact-management systems, such as reusable asset repositories, would handle service and composite-application components such as source code, service-interface declarations, software-architecture models or business-process models that are under construction. A reusable assets manager and asset repository can manage bundles of artifacts that describe assets according to the reusable asset specification (RAS) standard.

In the deploy and manage phase, WebSphere Service Registry and Repository can work with a configuration-management database (CMDB) to acquire and manage detailed information about the environment and topology in which service end points run, and service management products like the IBM IT Service Management suite can use and update that information and drive governance of processes that provision and configure the underlying infrastructure.

WebSphere Service Registry and Repository and CMDB federation enable users to get detailed information about the environment and runtime status of a service, whereas CMDB users can obtain detailed descriptions of shape and semantics of service end points from WebSphere Service Registry and Repository.

Service monitoring and management products like IBM Tivoli® Component Application Manager for SOA provide instrumentation of service interactions that enable you to monitor service interactions and service end-point behavior; summary information about service behavior can be pushed into WebSphere Service Registry and Repository to decorate service end-point metadata with implementation data. This information can be used to configure policy-enforcement points that implement the service-level agreements users want to enforce in service interactions.

WebSphere Service Registry and Repository technical overview

WebSphere Service Registry and Repository is a Java 2 Platform, Enterprise Edition (J2EE) application based on IBM WebSphere Application Server. Interactions with WebSphere Service Registry and Repository are provided through both Java application programming interfaces (APIs) and SOAP APIs. Basic create, retrieve, update, delete (CRUD) operations,

as well as governance operations and a flexible query capability based on XPath, are delivered through both APIs. When the SOAP API is used, content is communicated using XML data structures; when the Java API is used, content is communicated using service-data object (SDO) data graphs.

A Web user interface is the main way for users representing different roles to interact with WebSphere Service Registry and Repository. A fine-grained access-control model enables you to define which roles can perform what actions on which artifacts. You can also define and import classification systems from simple classifier sets to taxonomies and classification hierarchies. An Eclipse plug-in is also provided to support search, retrieval and publishing of service metadata in the context of Eclipse technology-based development tools or management consoles. A relational database (such as IBM DB2® or Oracle databases) is used as a backing store for service metadata.

WebSphere Service Registry and Repository administration interfaces support importing and exporting content for exchange with other metadata repositories. The interfaces also provide a Java Management Extensions (JMX) technology-based API for configuration and basic administration.

Governance functions include modeling a life cycle for governed entities using a state machine that describes life-cycle states, valid transitions between them, plug invalidators to guard the transition and (notification) actions to be taken as a result of the transition. WebSphere Service Registry and Repository also provides interfaces to analyze change impact to the registry and repository content, and features to audit these changes.

For more information

To learn more about IBM WebSphere Service Registry and Repository, Version 6.0, contact your IBM representative or IBM Business Partner, or visit:

ibm.com/software/integration/wsrr

To join the Global WebSphere Community, visit:

www.websphere.org



IBM WebSphere Service Registry and Repository, Version 6.0 at a glance

Hardware requirements

Any hardware that is explicitly compatible and fully capable of running the specified operating system, all the corresponding supporting software and any associated applications unmodified. The responsibility to provide a statement of full compatibility between machines lies with the original equipment provider (if not IBM).

- IBM AIX®: IBM System p™ machines
- HP-UX: Hewlett-Packard HP-9000 systems
- Linux® x86 platform: 32-bit Intel® PC hardware
- Sun Solaris Operating Environment: Sun SPARC processor systems
- Microsoft® Windows®: x86-compatible PC hardware

Software requirements

Server platforms

- IBM AIX, Version 5.3
- HP-UX 11.23 (on PA-RISC platform)
- Windows 2003 Enterprise Server (32 bit)
- Red Hat Enterprise Linux AS, Version 4.0
- Sun Solaris Operating Environment, Version 2.9 or 2.10
- SUSE Linux Enterprise Server, Version 9.0

Repository databases

- IBM DB2 Universal Database™ Enterprise Server Edition, Version 8.2 with Fix Pack (FP) 5 (supplied with WebSphere Service Registry and Repository)
- Oracle 10g Enterprise Edition Release 10.2.0.1.0 (Sun Solaris systems only)

Application server

IBM WebSphere Application Server, Version 6.0.2 with FP 11 (supplied with WebSphere Service Registry and Repository)

Compilers

- AIX: XL C/C++ Enterprise Edition, Version 7.0 or 8.0
 - HP-UX (PA-RISC): HP aC++ A.03.63
 - Linux: gcc 3.23 or gcc 3.4 (gcc 3.4 included with Red Hat Enterprise Linux AS, Version 4.0)
-

© Copyright IBM Corporation 2006

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States of America
11-06
All Rights Reserved

AIX, DB2, DB2 Universal Database, IBM, the IBM logo, System p, Tivoli and WebSphere are trademarks of International Business Machines Corporation in the United States, other countries or both.

Intel is a trademark of Intel Corporation in the United States, other countries or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries or both.

Other company, product or service names may be trademarks or service marks of others.