BEA WebLogic Server ᴿ JAX-WS Web Service Stack

JAX-WS Enhancements

This release of WebLogic Server contains significant enhancements to the JAX-WS web service stack originally introduced in WebLogic Server 10.0. These features include:

- Support for the JAX-WS 2.1 API and the JAX-WS 2.1.2 Reference Implementation

- XML Catalog support

- Enterprise-quality support for WS-Security (identical to the WS-Security support available in the WLS JAX-RPC stack).

JAX-WS 2.1 Support

WebLogic Server now supports JAX-WS 2.1, which is a maintenance release of the JAX-WS 2.0 (JSR 224) application programming interface (API). JAX-WS 2.1 adds the following features to those found in JAX-WS 2.0:

- Support for the JAXB 2.1 (JSR 222) Data Binding API

- WS-Addressing support

- Dynamic Publishing of Endpoints

- APIs for EndpointReference creation and propagation

- Annotations and APIs to enabled/disable features, such as MTOM and Addressing

For details on these enhancements in the JAX-WS API, please see the JAX-WS 2.1 maintenance release specification, see http://jcp.org/aboutJava/communityprocess/mrel/jsr224/index2.html

The WebLogic Server implementation of JAX-WS is based on the JAX-WS reference implementation (RI), version 2.1.2, and includes enhancements to the tool-layer to simplify the building and deployment of JAX-WS services and to ease the migration from JAX-RPC to JAX-WS. The JAX-WS RI is an open source implementation of the JAX-WS specification, developed within the Metro project. The following features and enhancements are available from the JAX-WS RI 2.1.2:

- .NET 2.0/WSE 3.0 MTOM Interoperability Support

- Significant Performance Improvements through the use of Woodstox StAX Parser

- SOAPAction Based Dispatching

- Integration of JAXB RI 2.1.5

- JAXB Type Substitution Support

- WS-Addressing Support for both W3C (1.0) and Member Submission (2004/08)

- Asynchronous client/server Support

- Dispatch and Provider Support

  - Dispatch<Message> and Provider<Message> Support

  - Allows development of non-WSDL or non-SOAP endpoints, such as REST

As with WebLogic Server 10.0, developers may begin development with either a Java source file or WSDL file. The WebLogic Server ant tasks `<jwsc>` and `<clientgen>` automate the generation of portable data binding classes, creation of deployment descriptors, and packaging.