Web Services

Beehive Metadata for Web Services

The purpose of JSR-181 "Web Services Metadata for the Java Platform" is to simplify the authoring and deployment of Java Web Services. Beehive's "Web Services Metadata for the Java Platform" subproject (WSM) provides an open source implementation of this specification that (when completed) can be used on on many web service deployment platforms. Note, this project is also known as "WSM".

Note, WSM **is not** another web service runtime. Rather, WSM is used to *author* web services that then execute atop a web service runtime. For example, users can write web services as annotated Java files and execute them atop Axis, Geronimo, and various other web service runtimes.

- Architecture: WsmArchitecture
- Tools: WsmTools
- Feature wishlist: WsmFeatureWishlist

Status

WSM has been undergoing substantial refactoring in the last few months and is getting closer to completion. The three areas left with sbustantial work include:

- type mapping
- .wsdd file generation
- tools

The near-term priority for WSM is passing the JSR-181 TCK. This project has been alive for a long time, but it's still not shipped a usable 1.0. WSM needs to ship before adding substantial new features including additional tools, XMLBean support, and so on.

Features

- Support for authoring annotated web services that execute atop the Axis 1.x web service runtime

Glossary

- Annotation Processor: A processor for JSR-175 annotations; the JSR-181 processor unless stated otherwise.
- Annotated Web Service: A .java source file that complies with JSR-181 and uses JSR-175 metadata.
- Web Service Runtime: The actual web service runtime on which an annotated web service executes. Such implementations might be Apache Axis, WebLogic, Geronimo, and so on.
- WSDL (Web Service Definition Langugae): An XML instance document that defines the interface for a web service.
- WSM: The TLA for this project.
- XSD: An XML Schema file; schemas are often used in web services to define the message shape of a web service's inputs and outputs.
- .class file: executable byte code derived from annotated Java Web Service files file through compilation.
- .jws file: An Axis-specific Java web service.

References

- APT Sun Microsystems Annotation Processing Tool (APT).
- Axis Apache Software Foundation. Web Services Axis.
- JSR-101 R. Chinnici. Java APIs for XML based RPC.
- JSR-109 J. Knutson. Implementing Enterprise Web Services.
- JSR-175 G. Bracha. A Metadata Facility for the Java Programming Language.
- JSR-181 M. Mihic, J. Trezzo. Web Services Metadata for the Java Platform.

Archive

Old documents are linked here:

- JSR-181 Project Plan (Proposal): Jsr181ProjectPlan
- General Proposal: Jsr181Impl
- Requirements for JSR-181 Implementation: Jsr181Requirements
- Architecture for JSR-181 Implementation (Proposal): Jsr181ImplArchitecture