



IBM Software Group

IBM® WebSphere® Application Server V6.1 Feature Pack for Web Services

***SOAP with attachments API for Java™
interface 1.3 (SAAJ 1.3)***



@business on demand.

© 2007 IBM Corporation
Updated August 6, 2007

This presentation will provide an overview of the SOAP with Attachments API for Java Interface 1.3 or SAAJ 1.3 technologies provided by the Feature Pack for Web Services.

Agenda

- SOAP with attachments API for Java interface 1.3 (SAAJ 1.3)
- Problem determination

This presentation will explain the SAAJ 1.3 support in the Feature Pack for Web Services, and problem determination for this component.

What is SAAJ

- Normally, in a Web services environment, programmer deals with objects and relies on underlying code (for example JAX-WS, JAX-B) to translate these objects to XML
- SAAJ provides APIs to read, write, send, and receive SOAP Messages
- SAAJ also allows binary content to be attached to a SOAP message
- SAAJ related classes are in the **javax.xml.soap** package



The normal way of processing the data that is exchanged between Web Services and clients is for the programmer to allow the underlying code (for example; JAX-WS, JAXB and the runtime) to handle the translation between the XML data in the SOAP message and the Java objects that the programmer uses. There are, however, times when access to the SOAP message, including any binary content, is needed. SAAJ provides a set of APIs to read, write, send, and receive SOAP Messages and binary data attached to those messages. The SAAJ classes are found in the javax.xml.soap package.

SAAJ 1.3

- Extends the existing SAAJ 1.2 specification
- Existing methods retain same SAAJ 1.2 semantics
 - ▶ Support for DOM Level3
 - ▶ Support for SOAP 1.2
 - SOAPFactories can now create SOAP 1.2 or SOAP 1.1 payloads
- Existing applications written to SAAJ 1.2 will work with SAAJ 1.3
- As a best practice:
 - ▶ Treat SAAJ 1.3 as a secondary programming model
 - ▶ Only use, if necessary, with JAX-WS

SAAJ 1.3 extends the existing SAAJ 1.2 specification so existing applications written to SAAJ 1.2 will work with SAAJ 1.3. Existing methods use the same semantics as in SAAJ 1.2, with added support for DOM Level 3 and SOAP 1.2 bindings. The SAAJ SOAP Factories have been updated to support both SOAP 1.1 and 1.2 payloads. Typically developers will not need to write applications that use the SAAJ APIs, most normal tasks can be accomplished with the JAX-WS programming model instead.

SAAJ problem determination

- Problems **reported by** the code associated with SAAJ are frequently data model related
- The data model is streamed through the various components (axis2, JAX-WS, sandesha, security, and WebSphere Application Server)
- Therefore, the best approach is to use several trace strings:
 - com.ibm.xml.xlsp.*=all
 - org.apache.*=all
 - com.ibm.ws.webservices.*=all
 - com.ibm.ws.websvcs.*=all
 - ▶ open source tracing may be disappointing (that is, insufficient tracepoints)
- Prior to determining the trace string, try to determine if the problem is client or server related. This may not be obvious, for example:
 - ▶ an exception may occur on the server, but the server is responding to invalid information from the client
 - ▶ client may report exception but exception is really occurring on server due to some internal error happening on service side
- Using tcpmon to collect message traces is extremely useful



Most problems related to SAAJ will be related to the Data Model or form of the SOAP message. Since the data is streamed through numerous components it is best to use multiple trace strings, capturing data for each of those components. Prior to running the trace, it is good practice to determine if the problem is specific to the client or service provider. Using a TCP monitor to view the SOAP message can help to determine where the problem originates.

Section

Summary

Summary

- In addition to introducing JAX-WS and JAXB technologies, IBM WebSphere Application Server V6.1 Feature Pack for Web Services supports the SOAP with attachments API for Java interface 1.3 (SAAJ 1.3) specification
- This presentation has given some high level information on these technologies and some guidance for problem determination

In addition to introducing JAX-WS and JAXB technologies, IBM WebSphere Application Server V6.1 Feature Pack for Web Services supports the SAAJ 1.3 specification. This presentation has given some high level information on these technologies and some guidance for problem determination related to the SAAJ 1.3 technologies.

