

Apache AXIS Experience

by Simon Zambrovski

Table of contents

1 First steps.....	2
1.1 Installation.....	2
1.2 Running samples.....	2
2 Writing own applications.....	2
2.1 Web services.....	2
2.2 Handler.....	2
3 Problems.....	3
4 Links.....	3

1. First steps

1.1. Installation

During the project definition a question about a convenient Web-Service toolkit occurred. The decision has been met to use Apache Axis. In the time of project implementation Apache Axis 1.1 has reached a stable state. Axis 1.2 was available as a Beta-Release from Apache CVS. The installation of both versions were trivial. It is detailed described in the readme-File.

1.2. Running samples

The most Apache Axis samples, delivered together with the distribution were running as written in the readme. Some of them didn't work at all. (e.g. Proxy-Sample)

2. Writing own applications

2.1. Web services

The project implementations needs several Web Services to be built. First, the sample scenario Web Services has been implemented using Axis 1.1. This was a straight-forward implementation. The usage of Axis for the Web Service endpoints described in the WS-C and WS-BA specifications were not convenient using the Axis built-in serializer/deserializer.

Due to the modular structure of Apache Axis I used Castor 0.956 for the XML-to-Java mapping as described in the article on developerWorks (see [Links](#)). Some difficulties occurred during the setup of this symbiose, so the Apache Axis 1.2 has been obtained from the apache CVS and rebuild with Clover as de-/serializer.

Note:

During the build we noticed, that there were some problems with having Clover on your Classpath. If you did, Clover has been tightly bound in Axis libraries and was needed for runtime execution.

2.2. Handler

For more excited control on the message processing it is useful to implement own handlers. There are samples in AXIS documentation describing how to do it.

Note:

The usage of handlers had the following effect of running in JBoss: the dynamic (in-runtime) undeployment or redeployment

of handler in JBoss is impossible and makes the JBoss restarts necessary. The re-/undeployment works in standalone Tomcat.

3. Problems

The main problem of Axis usage was the lack of sufficient documentation at the time of the project creation. The only useful documentation sources could be found on IBM and Apache AXIS homepages.

4. Links

- [Apache Axis Installation Guide](#)
- [Apache Axis main page](#)
- [Create Web services using Apache Axis and Castor](#)