

XML Messaging with Apache XML-RPC

Apache XML-RPC is a Java API that enables remote procedure calls to be marshalled as XML using HTTP as a transport protocol. XML-RPC is a subset of SOAP Web Services and, although easy to programme, lacks much of the richer functionality provided by SOAP. In this lab, we will see how the framework can be used to encode simple and complex types in an XML format and provide access to them through a set of public methods.

1. Compile the source code as follows:

- Linux / OSX: `javac -cp ./lib/* ie/gmit/sw/*.java`
- Windows: `javac -cp ./lib/* ie/gmit/sw/*.java`

You should have compiled class files of the following:

MessageHandler: Provides simple accessor methods for encapsulating a string in a message.

MessageRequestProcessorFactoryFactory: a factory class that ensures that a stateful instance of MessageHandler is created.

FileHandler: Provides a method `byte[] downloadFile(String fileName)` that returns a `byte[]` of a file on the XML-RPC server. The `byte[]` is Base64 encoded for XML transport.

FileRequestProcessorFactoryFactory: a factory class that ensures that a stateful instance of FileHandler is created.

XMLServer: Although XML-RPC is usually deployed as a web application on Tomcat, it can also be used as a stand-alone web server. Each service is attached as a handler to the service, using the PropertyHandlerMapping class, either programmatically or from a properties file. Note that the mechanism for adding handlers in XML-RPC limits how objects can be constructed and thus lacks the extensibility and flexibility required for enterprise computing.

Client: The `XmlRpcClient` class provides an `execute()` method for making XML-RPC calls. The `execute()` method must be parameterised with the handler name/class name and an `Object[]` representing a list of parameters. The latter is still required, even for method that have no parameters.

2. Start the XML-RPC server from a command prompt:

- Linux / OSX: `java -cp ./lib/* ie.gmit.sw.XMLServer`
- Windows: `java -cp ./lib/* ie.gmit.sw.XMLServer`

3. Execute the client application from another command prompt:

- Linux / OSX: `java -cp ./lib/* ie.gmit.sw.Client`
- Windows: `java -cp ./lib/* ie.gmit.sw.Client`

4. Start the TCP Monitor utility, available with the Apache Axis API (on Moodle)

- Linux / OSX: `java -cp ./path/to/axis/lib/* org.apache.axis.utils.tcpmon`
- Windows: `java -cp ./path/to/axis/lib/* org.apache.axis.utils.tcpmon`
- Start the monitor listening on port 7777 and relay requests to the port number the XML-RPC server is running on (8080). Modify the client code to point to port 7777, recompile and execute. You should be able to see the XML-RPC request and responses in XML format using the HTTP protocol.