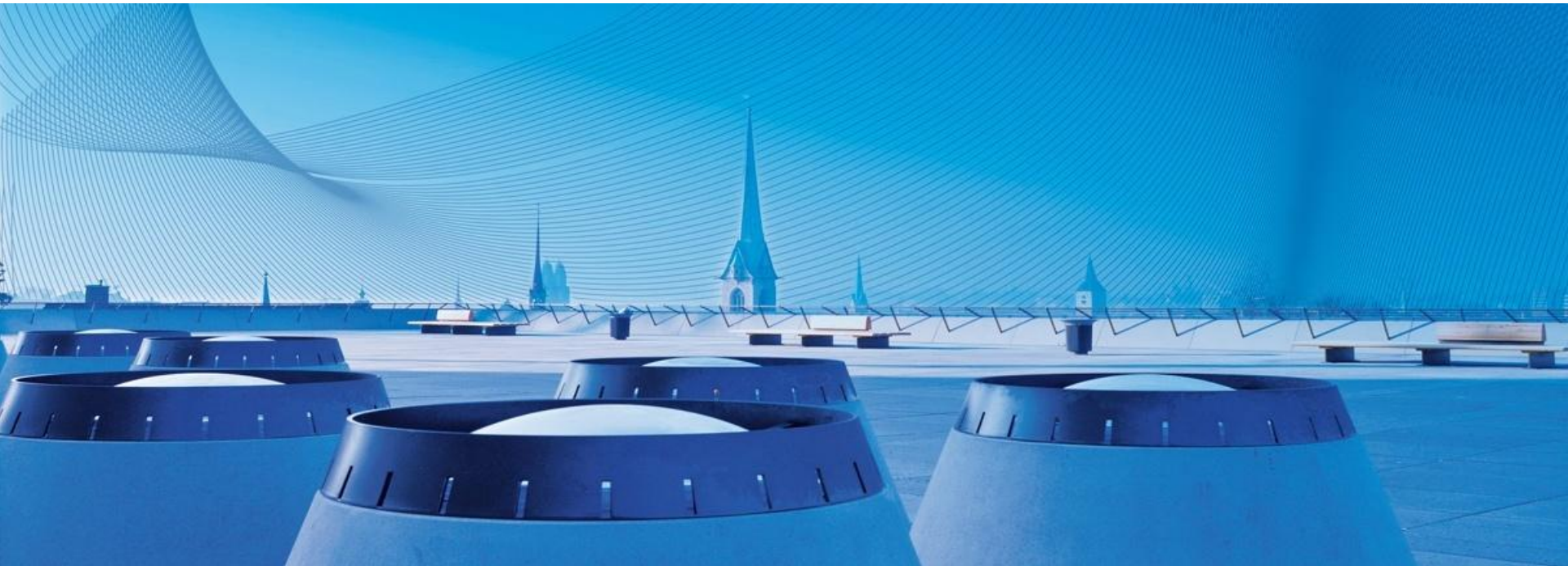


Some Hints for WS-* and REST

Dominique Guinard
Distributed Systems Group



Some Links for WS-*

- The WS-* galaxy is as big as this:

<http://www.innoq.com/resources/ws-standards-poster/>

Hence, you do **not have to understand all** the standards for this assignment. Rather you should understand them on a high level and experiment with them:

- <http://netbeans.org/kb/docs/websvc/intro-ws.html>
gives you a very brief overview of WS-* and REST.
- <http://gdp.globus.org/gt4-tutorial/multiplehtml/ch01s02.html>
gives you more information about Web-services.
- You definitely do not need to know more for solving the assignment.
- If you do want to know more, Google (or any other of your choice) search is your friend!

Some Hints for WS-* and kSOAP2

- Use the patched version of kSOAP2:
<http://code.google.com/p/ksoap2-android/>
The original version will not work because of performances and class availability in Android.
- There are a lot of tutorial and blog posts that can help you solving the task, here is a selection:
 - <http://www.android10.org/index.php/articleslibraries/167-using-ksoap2-for-android-soap-web-service> provides a short tutorial on kSOAP2 for Android (reading the first part should be enough).
 - <http://www.javaworld.com/javaworld/jw-08-2002/jw-0823-wireless.html?page=1> has a longer tutorial on kSOAP2.
 - <http://vimeo.com/9633556> is a step by step video tutorial on how to consume a WS-* service from Android.

Some Links for REST

- Tutorials that can help you understand the basic of REST:
 - <http://www.infoq.com/articles/rest-introduction> should be enough for the assignment.
 - <http://www.ibm.com/developerworks/webservices/library/ws-restful/> is a more complete tutorial from IBM.
 - <http://netbeans.org/kb/docs/websvc/intro-ws.html> gives you a very brief overview of both, WS-* and REST.
 - If you want to know more (not required for the assignment), this book (available at the D-INFK library) is probably the best resource:

RESTful Web Services
by Leonard Richardson, Sam Ruby



Some Hints for the Apache HTTP Client (REST)

- There are several ways of using Apache HTTP Client to make requests, check out the tutorials:
 - <http://tinyurl.com/dyzfzq>
(<http://svn.apache.org/repos/asf/httpcomponents/httpclient/trunk/httpclient/src/examples/org/apache/http/examples/client/ClientWithResponseHandler.java>)
is probably the easiest way to work with it.
 - <http://hc.apache.org/httpcomponents-client-ga/tutorial/html/index.html>
is the most comprehensive manual.

More Questions?

- Check out the given links.
- The community around these two service technologies is big, try «googleing» a little for your problem.
- If you still need help with the implementation, please feel free to contact Dominique Guinard from our group:

dguinard@mit.edu

(allow a few hours for a reply
as he is currently on Boston time)