

Service-Oriented Architecture Module: *Pre-study*

Paul Fremantle, paul@fremantle.org, November 2013

Since the World-Wide Web opened up distributed computing for a huge range of users and applications, programming models to support that kind of distribution have been multiplying rapidly. The range of technologies involved in delivering such software can be bewildering, but some relatively straightforward principles for service orientation are emerging.

In this course we will study enough of the technologies of web services (SOAP and REST) to enable us to implement, call, and intermediate simple services, but also try to engage with the broader issues of service orientation: how to design and engineer services and web APIs, orchestration, security and management.

The textbook *Service-Oriented Architecture: Concepts, Technology and Design* by Thomas Erl is excellent in where it goes, but it is a little dated. Unfortunately, there isn't a single book that covers everything I'd like. This is definitely the best book for the overall vision as well as the WS-* side of SOA. There are other excellent books that cover RESTful concepts but don't have the same overall introduction.

An ideal preparation for the course would be to read the whole of Chapters 1-3 (and also Ch. 4 if you get time). In addition I recommend Chs. 8 & 9. Of course you are welcome to read more!

To balance out the very WS-* focused approach of that text book, I suggest you also do some pre-reading around RESTful design approaches.

I recommend the following free resources:

- <http://www.infoq.com/articles/rest-introduction>
- <http://www.infoq.com/articles/webber-rest-workflow/>

The **practicals** will be managed with open source tools using the following technologies:

- **XML** including XML namespaces and XML schema
Appendix A of the textbook provides the necessary material: if you are new to XML, read this before (or, if absolutely necessary, in preference to) reading the early chapters of the book.
- **JSON**
JSON is a very simple model for data interchange.
The following article is a reasonably good introduction.

<http://msdn.microsoft.com/en-us/library/bb299886.aspx>

- **Java**

Most of the practical tasks will rely on the Java programming language. We will make them as accessible as possible for those who are not experts in the language, but if you have an opportunity to review some of the details of Java before the course, please do.

There is some use of Eclipse as well as the Linux BASH command line utility. I would highly recommend you take a quick look at both beforehand.

Finally, we may get the opportunity during the course to make use of Amazon Web Services, so it would be helpful if you could sign up in advance for an AWS account at <http://aws-portal.amazon.com/gp/aws/developer/registration/>. Even if we don't get a chance, this is a good thing to do anyway!