Introduction

From SOA to REST: Designing and Implementing RESTful Services [./]
Tutorial at WWW2009 [http://www2009.org/] (Madrid, Spain)

Cesare Pautasso (**Faculty of Informatics**, **University of Lugano**)
Erik Wilde (**School of Information**, **UC Berkeley**)

April 21, 2009

Contents

Abstract (2)

This introduction presents the schedule, the tutorial presenters, and some background for the tutorial. Specifically, we briefly mention all the *OA terms that have been invented in recent years, such as SOA (Services), ROA (Resources), WOA (Web), SynOA (Syndication), and EOA (Event), and briefly set them into context. Our main goal is to explain our notion of SOA for the purpose of this tutorial, and what we perceive as the core tasks when moving from SOA to REST.

Schedule (3)

- 9.30-11.00: Intro & What is REST? [What is REST?]
- 11.30-13.00: RESTful Service Design [RESTful Service Design]
- 14.30-16.00: REST vs. WS-* Comparison [REST vs. WS-* Comparison]
- 16.30-18.00: REST in Practice [REST in Practice]

Presenters

Cesare Pautasso   (5)

- Computer Science at Politecnico di Milano, Italy
  [http://www.polimi.it]
- Ph.D. at ETH Zürich [http://www.ethz.ch/index_EN] (2004)
- Post-Doc at ETH Zürich [http://www.iks.inf.ethz.ch/]
  - Software: JOpera: Process Support for more than Web services
    [http://www.jopera.org]
- Researcher at IBM Zurich Research Lab [http://www.zurich.ibm.com] (2007)
- Assistant Professor at the Faculty of Informatics [http://www.inf.unisi.ch/] (since September 2007)
- Representations: Web [http://www.pautasso.info]

Erik Wilde   (6)

- Computer Science at Technical University of Berlin (TUB)
  [http://www.tu-berlin.de/eng/] (1988-1991)
- Ph.D. at ETH Zürich [http://www.ethz.ch/index_EN] (1992-1997)
- Post-Doc at ICSI, Berkeley [http://www.icsi.berkeley.edu/] (1997/98)
  - book on "Technical Foundations of the World Wide Web [http://dret.net/netdret/publications#wil98]"
- Various activities back in Switzerland (1998-2006)
  - teaching at ETH Zürich [http://www.ethz.ch/index_EN] and FHNW [http://www.fhnw.ch/]
  - working as independent consultant (training, courses, consulting)
  - research focus on Web architecture and XML technologies
- Professor at the School of Information [http://ischool.berkeley.edu/] (since Fall 2006)
  - technical director of the Information and Service Design (ISD) program
    [http://isd.ischool.berkeley.edu/]
- Representations: Web [http://dret.net/netdret/]; blog [http://dret.typepad.com/]; twitter
  [http://twitter.com/dret]

*OA Overload

What is SOA? (8)

- What is *Service-Oriented Architecture*?

1. Alignment of *business objectives* and *IT*
   - ○ can be implemented with any architecture, technology, products
   - ○ SOA explained like this is more for business people
2. Technical architecture (interfaces are exposing *services*)
   - ○ focus on IT and the idea of *services* as the main abstraction
   - ○ still little guidance on how exactly a service is identified and exposed
3. SOA as the high-level explanation for *WS-* Web Services* [http://en.wikipedia.org /wiki/Web_service]
   - ○ this is how SOA as a buzzword started
   - ○ most SOA products are focusing on this view of *Web Services*

What are Web Services? (9)

Definition: A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

"Web Services Architecture", W3C Working Group Note, February 11, 2004 [http://www.w3.org /TR/ws-arch/#whatis]

Resource Oriented Architecture (ROA)   (10)

- More concrete guidelines for Web-based implementations
- Taking *services* and turning them into *RESTful Web services*
    - Query string parameters are appropriate if they are inputs to a Resource which is an algorithm
    - Prefer pragmatic uses of putting data into URI, instead of using HTTP Headers
    - RPC-style APIs are avoided in favor of Resources and protocols
    - A representation of a resource should have many links to the other Resources in the application, so that a client can discover state transitions
    - URI templates provide the technology behind specifying families of URI to clients
- Not really an architecture, more a "set of engineering principles"

Web Oriented Architecture (WOA)   (11)

- "WOA: Putting the Web back in Web Services"
- Unclear distinction from ROA (maybe no HTTP extensions?)
- Questionable conceptual landscape
    - "REST is the protocol most preferred since it's a natural extension of HTTP for the purposes of sharing self-describing information and state." [http://hinchcliffe.org/archive /2006/08/05/8489.aspx]
- Can be safely ignored

Syndication Oriented Architecture   (12)
(SynOA)

- *Syndication* can be regarded as a pattern for information dissemination
- Atom [REST in Practice; Atom (1)] and Atom Publishing Protocol (AtomPub) [REST in Practice; Atom Publishing Protocol (AtomPub) (1)] are existing Web standards for syndication
- SynOA is a pattern for building a SOA architecture
    - services are defined around collections
    - interactions are centered around reading and updating feeds

Event Oriented Architecture (EOA)   (13)

- Riding the SOA wave for even-oriented systems
- Now trademarked by eClient [http://www.eclient.com/]

Conclusions (14)

- *OA is hard to define and very hype-sensitive
- SOA lacks well-defined ways of how to define a service
- Business-level SOA tends to be implemented RPC-oriented/WS-*
- RESTful SOA is a better route for achieving loose coupling [http://dret.net/netdret /docs/loosely-coupled-www2009/]

Schedule (15)

- 9.30-11.00: Intro & What is REST? [What is REST?]
- 11.30-13.00: RESTful Service Design [RESTful Service Design]
- 14.30-16.00: REST vs. WS-* Comparison [REST vs. WS-* Comparison]
- 16.30-18.00: REST in Practice [REST in Practice]