

# Introduction to Assignment 2

Distributed Systems Lecture  
HS 2011, ETH Zurich

Simon Mayer

[simon.mayer@inf.ethz.ch](mailto:simon.mayer@inf.ethz.ch)



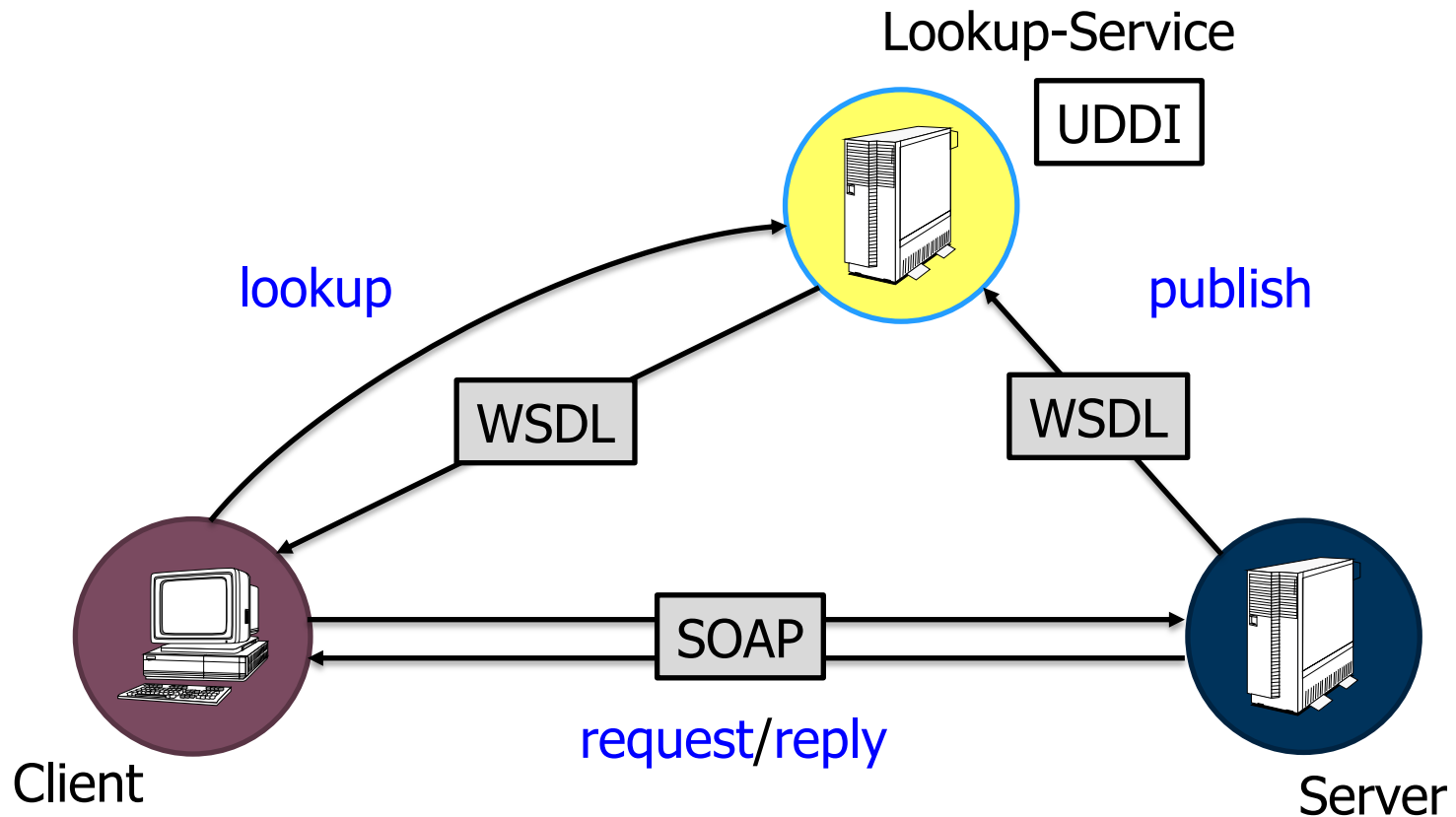


# Today's Menu

- Quick rundown of Web application architectures
  - WS-\* Web Services
  - Representational State Transfer
  
- Exercise 2
  - Overview
  - Tasks
  - Hints



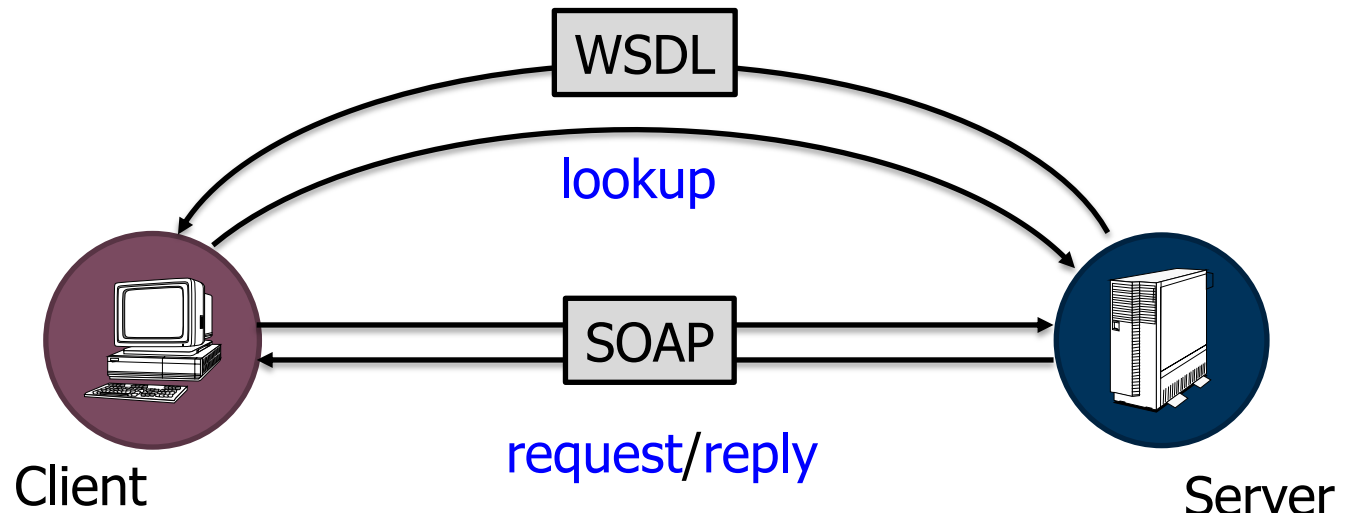
# Web Services in a Nutshell





# Web Services in a Nutshell

- For the exercise, we let the service publish its WSDL without going through a UDDI...





# Web Services - WSDL Overview

Exercise: <http://vswot.inf.ethz.ch:8080/SunSPOTWebServices/SunSPOTWebservice>

- *types, messages, portType, binding, port, service*



# Quick WS-\* Overview on next slide...

[http://en.wikipedia.org/wiki/List\\_of\\_web\\_service\\_specifications](http://en.wikipedia.org/wiki/List_of_web_service_specifications)



### Interoperability Issues

- Basic Profile
- Basic Profile
- Basic Profile
- Attachments Profile
- Simple SOAP
- Basic Security Profile
- WS-Trust Profile
- SAML-Trust Profile
- Confidence Data Attachment Mechanism
- Reliable Agreements

### Business Process Specifications

- Business Process Execution Language for Web Services 1.1
- WS-Choreography Model
- Web Service Choreography Interface
- Web Service Choreography Interface
- Business Process Management Language
- BPMN
- XBML Process Definition
- XBML

### Management Specifications

- Management Data Web Services
- Management of Web Services
- WS-Management
- Service Modeling Language

### Presentation Specifications

- Web Services for Remote Portals

### Metadata Specifications

- WS-Policy
- WS-Policy Assertions
- WS-Policy Attachment
- WS-Dictionary
- WS-Metadata Exchange
- External Description
- Web Service Description Language 2.0
- Web Service Description Language 1.2

### Reliability Specifications

- WS-ReliableMessaging
- WS-ReliableMessaging Policy Assertion
- WS-Reliability

### Security Specifications

- WS-Security
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions

### Transaction Specifications

- WS-Transaction
- WS-Transaction Activity
- WS-Atomic Transaction
- WS-Composite Application
- WS-Coordinator
- WS-Participant
- WS-Transaction Participant
- WS-Transaction Participant
- WS-Transaction Participant

### Resource Specifications

- WS-Security
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions
- WS-Security Assertions

### Messaging Specifications

- WS-Notification
- WS-OrderedNotification
- WS-RetrieveNotification
- WS-Eventing
- WS-Addressing - Core
- WS-Addressing - SOAP
- WS-Addressing - SOAP
- WS-Addressing - SOAP

### SOAP

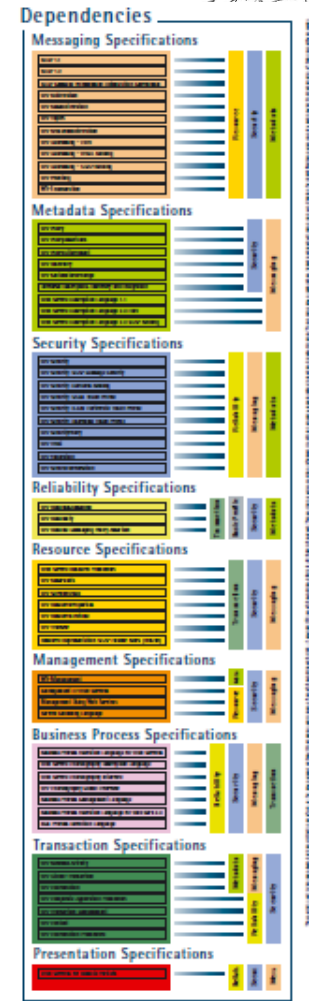
- SOAP
- SOAP
- SOAP
- SOAP
- SOAP
- SOAP

### XML Specifications

- XML 1.1
- XML 1.0
- Namespace in XML
- XML Information Set
- XML Schema
- XML Schema
- XML Schema
- XML Schema
- XML Schema

#### Standards Bodies

ISO/IEC JTC1 SC30  
OASIS  
W3C  
IEEE



innoQ  
innoQ Deutschland GmbH  
Häselstraße 17  
D-40880 Ratingen  
Phone +49 21 02 7793-100  
info@innoq.com - www.innoq.com

innoQ Schweiz GmbH  
Gewerkstrasse 11  
CH-6300 Cham  
Phone +41 41 743 01 11





# REST in a Nutshell

- Every resource gets an ID

e.g., a sensor node: <http://vswot.inf.ethz.ch:8081/sunspots/>



[<http://code.google.com/p/hcsfsp/>]

- Resources are linked
- Resources have multiple representations
- Resources provide a uniform interface
- Stateless communication





# REST and HTTP...

- REST and HTTP are not equivalent!
- REST uniform interface
  - REST nouns and verbs
    - HTTP: GET, POST, PUT, DELETE, OPTIONS, HEAD,...
  - WS-\* verbs



# Assignment 2 – Overview

## ■ Objectives – Exercise 2

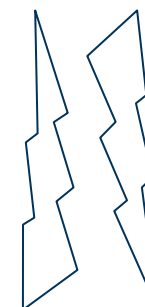
- Learn to develop distributed Web applications
- Use the two different paradigms seen in the lecture:
  - Representational State Transfer (REST)
  - Web Services (WS-\*)

## ■ Dates

- Exercise begins: **Now (October 10, 2011)**
- Exercise is due: **9:00am, October 24, 2011**



[<http://code.google.com/p/hcsfsp/>]



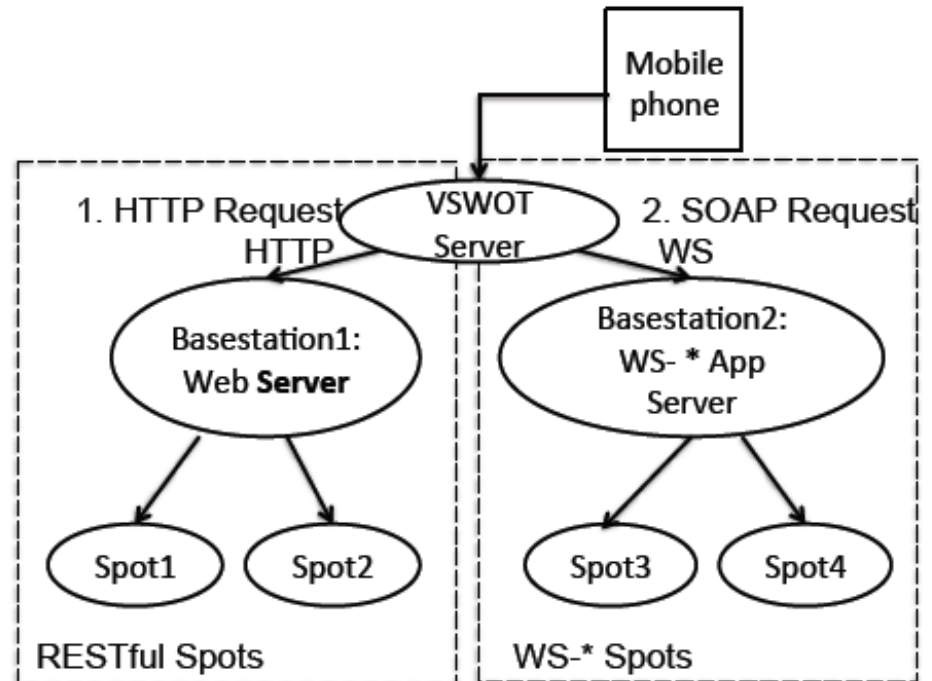


# Assignment 2 – System Setup

- Access Sun SPOTs through WS-\*/REST
- Sun SPOTs: Wireless sensor nodes (temp, acc, light,...)



[<http://code.google.com/p/hcsfsp/>]





# Assignment 2 – Tasks

1. Experimenting with RESTful Web Services (2P)
  - Create an HTTP request
    - a) “by hand” (i.e., without the use of any HTTP library)
    - b) Using *org.apache.http.\**
  - Use HTTP content negotiation to get machine-readable data
  - Connect to a Sun SPOT and retrieve the temperature value.
2. Experimenting with WS-\* Web Services (2P)
  - Explore WSDL, create SOAP requests
  - Connect to a Sun SPOT and retrieve the temperature value.
3. Assessing Web Service Technologies (1P)



# Assignment 2 – Tasks

## 4. Cloud Services (1P)

- Visualization of retrieved measurements
  - a) Using the *native Android* graphics libraries
  - b) Using the *Google Chart API*

## 5. Your Phone as a Server (2P)

- Implement a Web Server on your phone that allows to access the phone's sensors and actuators

## 6. Report (2P)



# Assignment 2 – Feedback & Study

- Study on REST and WS-<sup>\*</sup>
  - [https://docs.google.com/spreadsheet/viewform?hl=en\\_US&pli=1&formkey=dFFsbS1OVUVSaV9ld1dUYjZ1N0Jsdmc6MA#gid=0](https://docs.google.com/spreadsheet/viewform?hl=en_US&pli=1&formkey=dFFsbS1OVUVSaV9ld1dUYjZ1N0Jsdmc6MA#gid=0)
  - Perceived easiness/speed of learning of technologies
  - Feedback: Previous knowledge, time spent for assignment
  - Anonymous & individual



# Assignment 2 Hints - Relevant Terminology

- Media types: HTML, XML, JSON
  
- ROA – Resource-oriented Architecture
- REST – Representational State Transfer
  
- SOA – Service-oriented Architecture
- ~~SOAP – Simple Object Access Protocol (deprecated)~~
- WSDL – Web Services Description Language





# REST Hints

- <http://www.infoq.com/articles/rest-introduction>
- RESTful Web Services (Leonard Richardson und Sam Ruby)
  - Available at D-INFK library



- Apache HTTP library (simplest sample code alive... 😊)
  - <http://svn.apache.org/repos/asf/httpcomponents/httpclient/trunk/httpclient/src/examples/org/apache/http/examples/client/ClientWithResponseHandler.java>



# WS-\* Hints

- Patched version of kSOAP2
  - <http://code.google.com/p/ksoap2-android/>
  
- Short tutorial on kSOAP2 for Android
  - <http://www.android10.org/index.php/articleslibraries/167-using-ksoap2-for-android-soap-web-service>



# Visualization Hints

- Google charts API example:

<https://chart.googleapis.com/chart?chs=250x100&chd=t:60,40&cht=p3&chl=Hello|World>

- Getting started:

[http://code.google.com/apis/chart/image/docs/making\\_charts.html#usingthewizard](http://code.google.com/apis/chart/image/docs/making_charts.html#usingthewizard)



# Submission

- Same as for Assignment 1
  - Programs/Code, Report

+ Assignment form:

[https://docs.google.com/spreadsheets/viewform?hl=en\\_US&formkey=dDIPSFruV1BocjNCTIA0d1FpMERGRIE6MA#gid=0](https://docs.google.com/spreadsheets/viewform?hl=en_US&formkey=dDIPSFruV1BocjNCTIA0d1FpMERGRIE6MA#gid=0)

+ Feedback form (individual, anonymous):

<https://docs.google.com/spreadsheets/viewform?formkey=dFFsbS1OVUVSaV9ld1dUYjZ1N0Jsdmc6MA#gid=0>

# Introduction to Assignment 2

Distributed Systems Lecture  
HS 2011, ETH Zurich

Simon Mayer

[simon.mayer@inf.ethz.ch](mailto:simon.mayer@inf.ethz.ch)

