

INFS 3204/7204 Service-Oriented Architecture



A/Prof Heng Tao SHEN
ITEE, UQ
Semester 2, 2011.

M5: Web Service Basic

1

M5 Topics

- Basic concepts
- Deploying Web Services
- Key Web Service technologies
 - SOAP
 - WSDL
 - UDDI

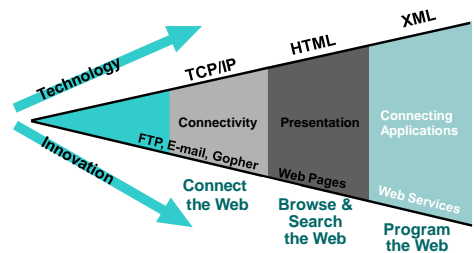
2

Web Evolution

- Success of Web attracted distributed computing systems to reap the ubiquitous benefits of Web for service oriented applications
 - B2B
 - B2C
 - EAI/EDI
- Problems of interoperability between different systems in different platforms require a platform-neutral solution
 - Web Service

3

Technology development



4

Web Service

- One of the hot new topics, about to "revolutionise the Web"
 - Connect to services on the Web and interact with them directly
 - E.g., Car rental quotes; stock quotes, etc
- Core Web Services technologies
 - SOAP: Message exchange protocol
 - WSDL: Service description
 - UDDI : Service registries
 - XML: The core. It is utilised for data setting, formatting, structuring and validation.

5

W3C Definition

- A Web Service is a software system identified by a URI, whose **public** interfaces and bindings are defined and described using **XML**
- Its definition can be **discovered** by other software systems
- These systems then **interact** with the Web Service in a manner prescribed by its definition, using XML based messages conveyed by **Internet** protocols

6

A briefer Definition

- A briefer definition
 - Web Services are loosely coupled, contracted components that communicate via XML-based interfaces and using Internet protocols
- A closer look
 - **Loosely coupled**: Web Services and programs that invoke them can be changed independently
 - **Contracted**: a Web Service's behaviour, its input/output parameters and how to bind to it are publicly available
 - **Component**: encapsulated code whose internal implementation is hidden
 - **XML**: human-readable, text-based format that is firewall friendly and self-describing

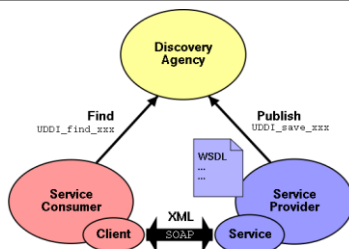
7

Web service benefits

- Promotes interoperability by minimizing the requirements for shared understanding
 - XML and its vocabularies
- Enables interoperability of legacy and new applications
 - Standards
- Enables just-in-time integration
 - Collaborations in Web Services are bound dynamically at runtime
- Reduces complexity by encapsulation
 - A WSDL document is the mechanism to describe the behaviour encapsulated by a service

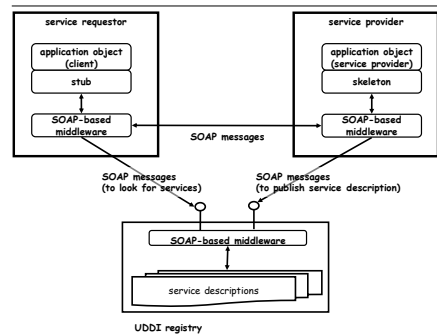
8

SOA architecture



9

A closer look with SOAP



10

4 Phases to Deploy Web Services

- Service Implementation
- Service publication
- Service Discovery
- Service Invocation

11

Phase 1: Service Implementation

- For providing new services
 - Write code (in any language you want), define interface (in WSDL), publish the interface (in UDDI) and deploy the service as a Web Service
- For providing alternative services when there is an existing service interface
 - Find an existing service by using the Web Services Registry, generate a service as above but to comply with the existing interface
- For integration of legacy systems
 - Develop a new service interface for an existing application

12

Phase 2: Publication

- Author the Web Service description document
 - Describe in WSDL what the service will do, where it can be found, and how to invoke it
- Publish the existence of your doc in a UDDI registry
 - Can be global, closed groups or intranet
- Publish the description doc on a Web server so your desired audience can access it
 - Host the service

13

Phase 3 & 4: Discovery and Invocation

- Discovery
 - Any application can discover your service and locate the Web Services description doc you published
 - UDDI supports pattern queries for automatic lookups and return the location of the WSDL file for the desired service using URI
- Invocation
 - Find the service on the server
 - Request the WSDL file based on URI
 - Invoke Web Service dynamically at run time

14

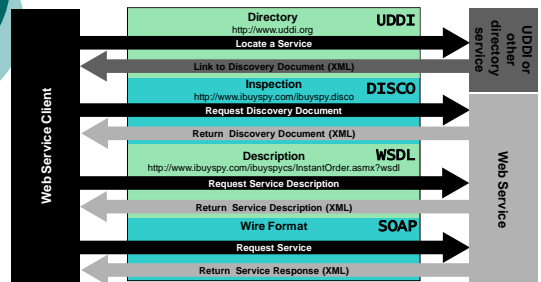
Underlying technologies

Directory: Publish & Find Services:	UDDI
Inspection: Find Services on server:	DISCO
Description: Formal Service Descriptions:	WSDL
Wire Format: Service Interactions:	SOAP
Universal Data Format:	XML
Ubiquitous Communications:	Internet

Simple, Open, Broad Industry Support

15

Web service stacks



16

Key technologies

- SOAP
- WSDL
- UDDI

17

SOAP overview

- Guiding principle: "Invent no new technology"
- Builds on key Internet standards
 - SOAP ≈ HTTP + XML
- The SOAP specification defines:
 - The SOAP message format
 - How to send messages
 - How to receive responses
 - Data encoding

18

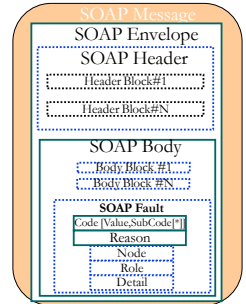
SOAP

- What is SOAP (Simple Object Access Protocol)
 - Simple messaging framework for transferring information between peers over Web in a distributed environment using XML
 - A messaging framework
 - Envelope contains Header (optional) and Body (mandatory)
 - An encoding format (how to encode, serialise and decode objects)
 - An RPC mechanism to call remote objects
- SOAP messages are text-based XML documents
- SOAP is implemented over HTTP
 - Non-proprietary standard (unlike RMI and DCOM)
 - Text based (unlike CORBA/IIOP)
 - Has the advantages of HTTP (efficiency and security)

19

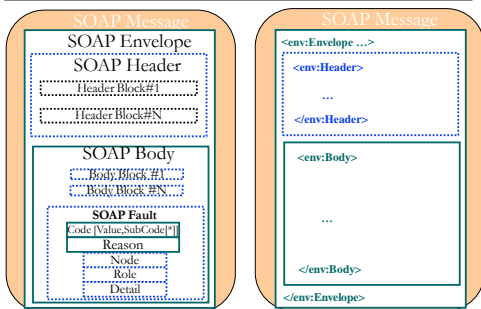
SOAP Message Structure

- **Envelope:** This is top level root element of a SOAP Message, which contains the Header and Body element
- **Header:** A collection of zero or more SOAP header blocks each of which might be targeted at any SOAP receiver within the SOAP message path
- **Body:** A collection of zero or more element information items targeted at an ultimate SOAP receiver in the SOAP message path



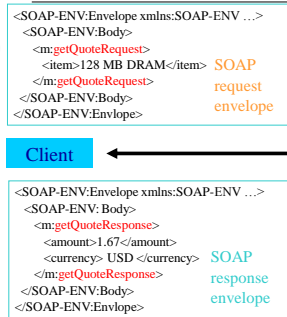
20

SOAP Message Structure



21

How SOAP Works



22

SOAP Messages

- SOAP request message
 - The envelope defines various namespaces used
 - The header (optional) can carry authentication, payment, etc
 - The body carries the payload of the message
 - For RPC, it contains procedure/methods name and the arguments
- SOAP response message
 - Just like a request but the body carries the results

23

Calling SOAP Methods

- A client that knows the format of a SOAP request and response can just make an HTTP request
- Pseudo code
 - Read the SOAP request from a file
 - Open HTTP connection to web services
 - POST request to service
 - Write response to stdout

24

SOAP Implementation

- Microsoft (<http://msdn.microsoft.com/soap>)
 - Languages: VB, C#
- Apache (<http://xml.apache.org/soap>)
 - Language: Java
 - Software required:
 - Java 1.1 and above
 - Tomcat 3.2.1 or later
 - Apache Xerxes XML parser 1.2.3
- SOAP::Lite (<http://soaplite.com>)
 - Language: Perl

25

WSDL

- An XML-based means for expressing the interface to a given Web service
 - Service functionality
 - Binding to a physical protocol
 - <http://www.w3c.org/TR/wsd1.html>
- A WSDL is a contract between a client and a server
 - Using XML to describe Web Services as collections of communication **endpoints** that can exchange messages with each other

26

WSDL features

- XML schema for describing Web Services
 - Service interface definition
 - Abstract semantics for Web Service
 - Service implementation definition
 - Concrete end points and network addresses where Web Service can be invoked
- Clear delineation between abstract and concrete messages

27

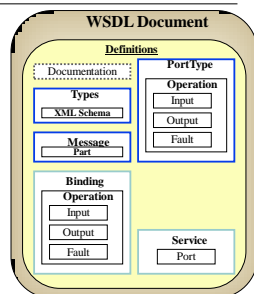
WSDL features

- WSDL provides a means of specifying
 - Data types used
 - Messages
 - Endpoints of a Web Services
- Endpoints are defined by a set of
 - Input messages
 - Output messages
 - Fault messages
- These end points are then bound to a messaging framework such as SOAP
 - The messaging framework is finally bound to a concrete instance of the service by specifying the addressing and the transport protocols, e.g., HTTP based URIs

28

WSDL Document Structure

- Definitions:** Associate the web service with its namespaces
- Documentation:** human readable
- Types:** A container for data type definitions
- Message:** An abstract, typed definition of the data contained in the message
- PortType:** The set of operations
- Operation:** An abstraction description of the web service
- Binding:** A spec of the protocol and data format for a particular portType
- Port:** An endpoint, defined in terms of binding and URL
- Service:** A collection of related endpoints



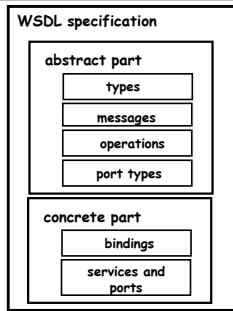
29

WSDL Concepts

- **Message** and **types**: data communicated to a service is typed
- **Operation**: an action provided by a web service, described by input and output messages
- **Port**, **portType** and **binding**: how dialog occurs between caller and service
 - A **binding** is how to access an operation using a particular protocol (SOAP or HTTP)
 - A **port** is a named association of a network address with a binding
 - **The caller sees only the port and its binding when making a request of a service**
- **Service**: a combination of related ports, specifies details about the implementation

30

Clear delineation



31



32

WSDL Implementations

- Some popular WSDL implementations
 - Microsoft SOAP toolkit
 - Sun's JAX-RPC
 - IBM Web Services Toolkit (WSTK)
- No ONE writes anything in WSDL**

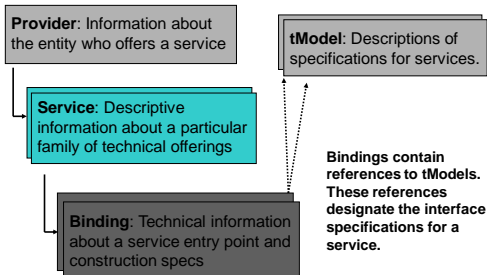
33

UDDI

- "Ultimate directory for online business"
 - A repository for WSDL docs
 - Not tied to WSDL
 - It's supported by an XML schema
 - Standard Querying API and publishing API (www.uddi.org)
- A business can register itself with a UDDI registry giving info such as
 - Business information:** about your company (name, descriptions, URL, contacts...)
 - Service information:** about your services (high level)
 - Binding information:** about how to invoke your services (technical)
 - Specification of services information:** further technical information about the services

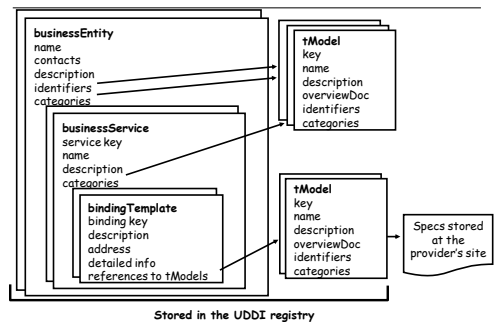
34

UDDI – Information model



35

A closer look



36



How UDDI work - tModel

- tModel = Technology Model
- Generic meta-data structure to uniquely represent any concept or construct
- Also includes interface protocol definitions
- Powerful abstraction modeling system

37



UDDI Features

- Neutral in terms of protocols – as a registry, it can contain pointers to anything
- Can search by business, service, Web Service (tModel), binding
- Usage of Globally Unique Identifiers (GUIDs)
- Specification allows public and private nodes
- Delineation between interface and implementation

41



Public/Private Registries

- IBM, Microsoft, SAP ... run public registries
 - uddi.ibm.com, uddi.microsoft.com
- They are synchronised so all UDDI registries contain the same info
 - A distributed database
- Private registries can be run within LANs or closed groups
 - IBM and Sun provide tools to build
-

42



Summary

- This week:
 - Basic concepts
 - Deploying Web Services
 - Key Web Service technologies
 - SOAP
 - WSDL
 - UDDI
- Next week:
 - **Web Service Composition**

43



References

- Web Services:
 - <http://www.w3.org/2002/ws/>
 - SOAP: www.w3.org/TR/soap
 - WSDL: www.w3.org/TR/wsd.html
 - UDDI: www.uddi.org

44