Web Services and Web Service Flows

TECHNISCHE UNIVERSITÄT DARMSTADT

Dimka Karastoyanova
Mariano Cilia

Introduction

- Why Web services?
- Some Definitions
- Characteristics
- The model
- The Platform

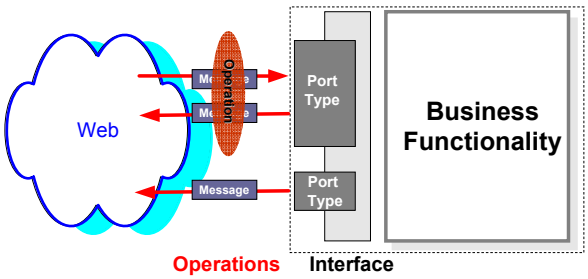
Introduction

- **Web Services**
  - The newest distributed systems technology
  - Reveals the shift in the focus of application providers
    - Traditional approach (build an application in-house, buy an application from a vendor)
    - The "web"-approach
      - Expose a piece of functionality on the Web
      - Access application functionality over the Web
  - Obey the principles of SOA
- **New approach triggered mainly by business requirements**
  - Time
  - Low cost
  - Low risk and complexity
  - Possibility to leverage existing software
    - From a technical viewpoint
    - Investment leverage
    - Changes in small steps with minimum risk and expenses

Web Services - Definitions

- **A Web service represents a unit of business, application, or system functionality that can be accessed over the Web.**
- Web services are **a new breed of Web applications**. They are **self-contained**, **self-describing**, **modular** applications that can be **published**, **located**, and **invoked across the Web**. Web services **perform functions**, which can be anything from simple request to complicated business processes… Once a Web service is deployed, other applications (and other Web services) can discover and invoke the deployed service. [IBM]
- A Web-Service is **an interface** that **describes a collection of operations that are network accessible through standardized XML messaging** [Microsoft]
- …

What is a Web Service?



General Characteristics (1)

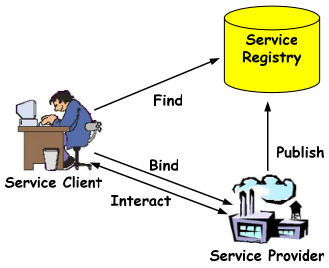
- **Web Services are fully XML based**
- **Expose interface**
  - Clients access the service functionality through interfaces
  - Communication between applications as opposed to communication between users
    - Computer-to-computer communication
- **Self-describing modular units**
- **Accessible over the Web**
- **Interaction**
  - XML based **messages** over standard Web protocols
  - Loose coupling
- **Platform, Language and Protocol independent**

© 2005 Dimka Karastoyanova

General Characteristics (2)

- Registered and discoverable at a Web service registry
- Compositions of Web Services are possible
- Web Services exclusively focus on
  - Interface description
  - Search & discovery
  - Protocol independent method call over the Web
- Web Services deliberately omit
  - Functionality implementation details
  - Abstraction
- Use: Implement the Service Oriented Architecture (SOA)

The SOA Model



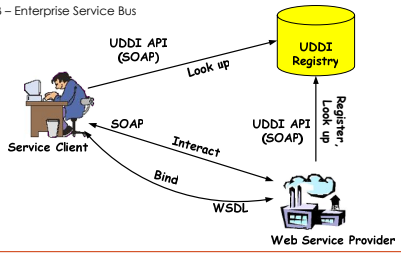
- Service Provider
  - Offers a service
- Service Requestor
  - Looks for and
  - Uses a service
- Service Broker:
  - Provides service registry
  - Allows services to be registered (by the services provider)
  - Allows service look-up (for the service requestor)

The Platform

- Basic platform: HTTP and XML
  - Access & Invocation
  - Description
- The basic Web Service Platform: WSDL + SOAP + UDDI
  - Augmenting the Web with Web Service specific support
    - SOAP - XML Communication
    - WSDL - Description
    - UDDI – Registration and Discovery
  - Full functionality
- Enhanced concepts – enterprise strength
  - Composition (Web Service Flows, Processes)
  - Web Service Transactions, Coordination
  - Security (authentication, encryption, etc.)
  - …

The Basic Web Service Standards

- SOAP
  - Communication protocol
- WSDL
  - Web Service (interface) description
- UDDI
  - Web Service discovery
- Platform
  - ESB – Enterprise Service Bus



WSDL

- Motivation
- Structure of WSDL Document
  - Abstract Part
  - Concrete Part
- Examples

WSDL

- WSDL = Web Services Description Language (1.1)
  - A standard language for defining Web Service interfaces
    - Therefore expect the constructs for defining interfaces
      - Interfaces (port types)
      - Methods (operations)
      - Parameters and their data types (type, messages, parts)
  - WSDL helps
    - to generate automatically stubs/skeletons for Web Services, regardless of the platform
    - To make dynamic calls to WSs (no stubs needed)
- A WSDL interface:
  - Describes a service in terms of the messages it accepts and generates
  - Represents a service as a collection of ports at which certain functionality is made available via a concrete binding

SOAP

- Introduction
- The Protocol
- Components of the SOAP Message
- Examples

SOAP

- **SOAP**
  - A **lightweight** protocol for exchange of information in a decentralized, distributed environment
  - SOAP is not a self-sufficient protocol
    - It is bound to the underlying transport protocols (http, smtp, jms, etc.)
- **SOAP defines a model for using simple request / response messages written in XML as the basic protocol for communication**
  - Focuses on the format and content of the request/ response message (not on the implementation)
- **Used to be referred to as "the platform-neutral remote procedure call"**
  - Message contains a method name and arguments

SOAP - The Protocol

- **Provides:**
  - a modular **packaging model** and
  - **encoding mechanism** for encoding data within modules using XML and XML namespaces
- **The protocol has 3 parts**
  - Envelope – contains the message components
  - Set of encoding rules (encoding to transport protocols)
  - Conventions for representing remote procedure calls
- **The "package" is called message**
  - SOAP messages are one-way transmissions, but can be combined in request/response pattern
  - Data to be transported is called payload

SOAP Message

- **SOAP Envelope:**
  - what is in a message
  - who should process it and
  - is it optional or mandatory to process the message content
- **SOAP Header – optional:**
  - Used to extend the exchanged information without prior knowledge of the communicating parties
  - Possible extensions
    - Security - Authentication, Encryption
    - Transactions – Transaction Contexts, Transactions (IDs)
    - Addressing/Routing – WS-Addressing
    - Other possibilities – tokens for correlation, brokered communications etc.
- **SOAP Body:**
  - Provides a mechanism for exchanging mandatory information intended for the ultimate recipient of the message

UDDI

- UDDI Registry
- The Model
- UDDI Specification
  - Structure of the UDDI Registry
  - UDDI API
- Examples

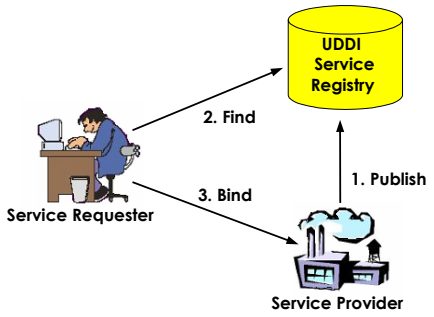
UDDI Service Registry

- **UDDI = Universal Description, Discovery and Integration Specification**
- **Defines a way to publish and discover information about businesses and their Web services**
- **The name UDDI is used to refer to:**
  - The specification (UDDI Service Registry Specification)
  - The registry itself as a service (UDDI Registry Service)
- **UDDI Registry is logically centralized, physically distributed service**
  - Multiple nodes replicate data with each other
  - All nodes are synchronized once in 24 hours
  - Once a service is registered at a node, the data is automatically shared with other UDDI root nodes

UDDI Service Registry (2)

- **It is not a full-fledged discovery service**
  - It enables **technical discovery** of services.
  - UDDI registries were initially developed to be used by portals able to access it
  - User specifies keys, provider name, service name etc.
  - **Semantic description is missing**
    - Some preliminary approaches exist already
- **Advanced discovery features are under development**
  - **Business discovery** is not yet supported
  - For example:
    - Geographical proximity, price, time zone etc. are not considered as search criteria
    - Give me all services implementing **Currency Converters** used in **Financial Applications** for **Accounting**

The Model



Web Services and WS-flows

- Business Processes
- Workflows
- BPEL4WS

Business Processes

- **Business process:**
  - Collection of activities
  - Performed by human users or software applications
  - To achieve a particular business objective
- **Process model - describes:**
  - Structure of a real world business process
  - Actions to be performed
  - Rules, alternative paths
- **Process Instance**
  - The instance of the process model is a process
  - Is carried out according to the rules and paths of the model
  - Put into context
  - Humans and/or programs perform tasks
  - Executed not only on a computer

Business Processes (2)

- **A business process specifies:**
  - The potential execution order of operations (from a collection of applications)
  - The data to be shared
  - The partners' involvement in the business process
  - Joint exception handling
  - How multiple parties and services participate
  - Might reflect organizational structures
  - Might specify the exact application programs involved in the workflow
  - Enforce constraints, business rules …
  - …

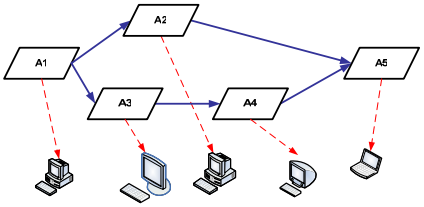
Workflows

- **Parts of processes run on computers:**
  - Called **workflows** (process instances)
  - Workflow model (process model)
- **Workflow**
  - executable description of a business process
- **Executed by a Workflow Management System (WfMS)**

© 2005 Dimka Karastoyanova

What is a Workflow?

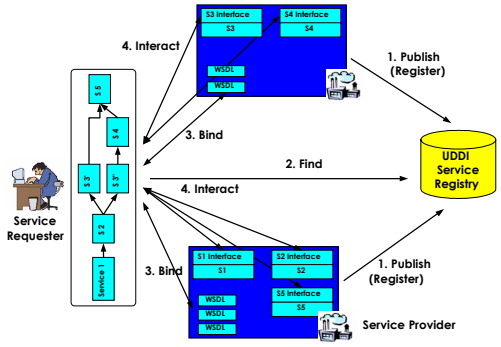
- **A description of:**
  - Tasks to be performed
  - By participants (human, programs, resources)
  - In a certain order (sequential, parallel, etc.)
  - Taking into account alternative paths → control flow
  - Data to be exchanged → data flow
  - Work items assigned to participants



Application of Workflow

- **Image/document processing**
- **Groupware – Coordination of team work**
- **Project Management**
- **Manufacturing**
- **Supply chain management**
- **Customer relationship management**
- **…**

Workflows and Web Services



BPEL4WS

- **Concepts**
- **Business Process Parts**
  - WSDL Description
  - Process Definition
- **Advanced Issues**

Business Process Execution Language for Web Services

- **BPEL4WS = Business Process Execution Language for Web Services (1.1)**
  - BPEL4WS is a flow language
  - Microsoft and IBM
  - A combination of WSFL and XLang, replacing them
    - WSFL – support for graph oriented processes
    - XLang – structural constructs of processes
- **BPEL4WS describes business processes**
  - XML based Web Service Flow definition language
  - A process is a combination of activities, involving multiple WS
  - The process itself is a Web Service, too
- **Goal**
  - Standard language for Web Service compositions

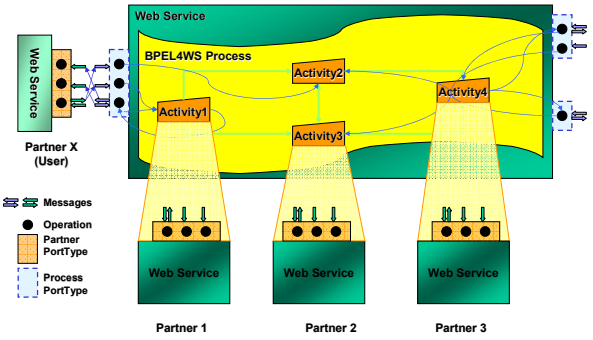
BPEL4WS Concepts

- **BPEL4WS Business Process Elements**
  - Process
  - Participants, relationships
  - Activities
    - Simple
    - Complex
  - Messages, Containers/Variables, Correlation data
  - Fault handling and compensation
  - Business protocols and views
- **All elements are put together in two groups**
  - Two XML documents are needed to deploy a BPEL4WS Process
    - WSDL document – interface description
    - BPEL document – process description

BPEL4WS Processes

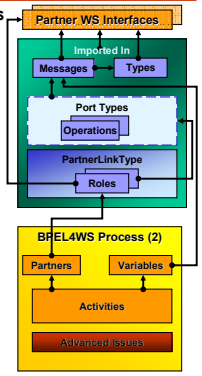
- **BPEL4WS Process – a composition of Web services**
  - Execution order of activities
  - Process's and partners' roles within an interaction
  - Data flow
  - Control flow
  - Error Handling
  - Compensation activities
- **Process interface description – Web service interface**
  - A WSDL document
  - portTypes and operations the process exposes
  - References to partners' Web services taking part in the interaction
  - Types of relationships to partners (ServiceLinkTypes)
- **Executable and abstract BPEL4WS processes**

BPEL4WS Process



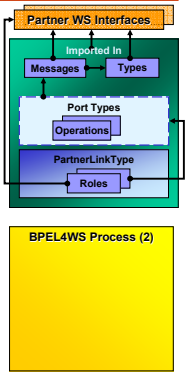
BPEL4WS Processes Parts

- **Any BPEL4WS Process consists of two+ parts**
- **The (+) part**
  - WSDL interface descriptions of all partners
  - Imported in the definitions
- **Definitions part (1)**
  - Interface of the process
    - As a web service (WSDL)
  - Types of Relationships between the process and the Web Services it uses
- **Process description part (2)**
  - BPEL document
  - Partners and containers
  - Definition of the process itself



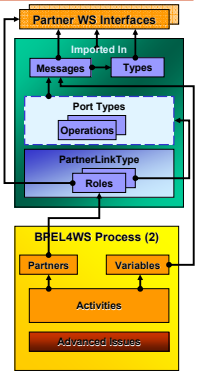
Process Interface (WSDL)

- **BPEL4WS Processes are exposed as Web Services**
- **Definitions = abstract definitions**
  - Partner Interfaces
    - Namespaces of partners' services are imported
    - Location of the WSDL file locally is specified
  - Types (optional)
    - Refer to types defined by partners' Web services
    - Define new process types (complex types)
  - Messages (optional)
    - Reference Types (default, complex or imported)
    - Used to define: Operation, Containers, Correlations …
  - PortTypes, Operations
    - Describe the BPEL process as a Web service
    - Used to define: Partners, Service Link Types, Roles, Process activities
  - PartnerLinkTypes
    - Types of relationships to partners
    - Define types of links/relations between partners and process
- **Implementation Specific**
  - Concrete definitions are generated by the BPEL process engine



Process Definition (BPEL)

- **Variables / Containers**
  - Store messages
  - Typed
- **Partners**
  - Represent the WSs advertised by partner organisations and
  - Used by the process
- **Activities – units of work**
  - Simple – cannot be decomposed
  - Complex – control flow, data manipulation, alternatives
- **Advanced Issues**
  - Failure handling
  - Transactional support
  - Compensation



Literature and References

Literature (1)

- **Web Services Essentials**
  - Ethan Cerami, O'Reilly & Associates, ISBN: 0596002246, 2002
- **Simple Object Access Protocol (SOAP)**
  - http://www.w3.org/TR/2002/WD-soap12-part0-20020626/
- **Programming Web Services with SOAP**
  - James Snell, Doug Tidwell, Pavel Kulchenko O'Reilly, Sebastopol, 2002, ISBN: 0596000952
- **Web Services Description Language - specification**
  - http://www.w3.org/TR/wsdl12/
- **Web Services Description Language (WSDL)**
  - Erik Christensen and Greg Meredith, Microsoft, Francisco Curbera and Sanjiva Weerawarana, IBM Research
  - http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwebsrv/html/wsdl.asp

Literature (2)

- **www.uddi.org, www.microsoft.com, www.w3c.org, etc.**
- **Understanding Web Services**
  - Eric Newcommer, Addison Wesley, ISBN: 0201750813, 2002
- **Understanding WSDL in a UDDI registry**
  - Peter Brittenham, Francisco Cubera, Dave Ehnevuske, Steve Graham, IBM Developer Works, September
- **UDDI Executive White Paper**
  - http://www.uddi.org/pubs/UDDI_Executive_White_Paper.pdf
- **UDDI Technical White Paper**
  - http://www.uddi.org/pubs/Iru_UDDI_Technical_White_Paper.pdf

Literature (3)

- **BPEL4WS Specification**
  - http://www.ibm.com/developerworks/library/ws-bpel/
- **Business Process with BPEL4WS**
  - Sanjiva Weerawarana, Francisco Curbera
  - http://www-106.ibm.com/developerworks/webservices/library/ws-bpelcol1/
- **Business processes in a Web services world**
  - Frank Leymann, Dieter Roller
  - http://www-106.ibm.com/developerworks/webservices/library/ws-bpelwp/?dwzone=webservices
- **Business Process Execution Language for Java Run Time**
  - http://www.alphaworks.ibm.com/tech/bpws4j
- **Reference guide for creating BPEL4WS documents**
  - http://www-106.ibm.com/developerworks/webservices/library/ws-bpws4jed/
- **ReFFlow Project**
  - http://www.informatik.tu-darmstadt.de/GK/participants/dimka/ReFFlow/index.htm
- …