Script generated by TTT

Title: Distributed_Applications (17.06.2014)
Date: Tue Jun 17 14:32:44 CEST 2014
Duration: 46:13 min
Pages: 18
A Web Service is a standardized way of integrating Web-based applications.

**Informal Definition**

Integration
- allows integration of application functionality within organizations
- between business partners across organizational boundaries

**Features of Web Services**

**Web Services - Distributed Objects**

**Features of Web Services**

**Web Services - Distributed Objects**

---

**Definition**

A Web service (WS) 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 may then interact with the Web service in a manner prescribed by its definition, using XML based messages conveyed by Internet protocols.

A Web Service is a standardized way of integrating Web-based applications using XML, SOAP, WSDL and UDDI open standards over an Internet protocol backbone.

**Simplified view**: a web service is a remote procedure call over the Internet using XML messages.

**Web Services Interoperability Stack**

**Basic Architecture**

**Roles**

**Operations of the Web Service Architecture**

**Basic Standard Technologies**

**Message Exchange Patterns**
Web Services Interoperability Stack

<table>
<thead>
<tr>
<th>Compositional</th>
<th>BPEL4WS, WS-Notification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quality of Experience</td>
<td>WS-Security, WS-Transactions, ...</td>
</tr>
<tr>
<td>Description</td>
<td>WSDL, UDDI, WS-Policy, ...</td>
</tr>
<tr>
<td>Messaging</td>
<td>XML, SOAP, WS-Addressing</td>
</tr>
<tr>
<td>Transport</td>
<td>HTTP, SMTP, ...</td>
</tr>
</tbody>
</table>

Basic Architecture

defines an interaction between software components as an exchange of messages between service requesters and service providers.

**Functions of the architecture**

- exchanging messages,
- describing Web services,
- publishing and discovering Web service descriptions.

The service: a Web service is an interface; implementation of it is the service.
The service description: details of the interface and the implementation of the service.

Roles

The basic Web service architecture models the interactions between three roles

**Service Provider**

processes a Web service request.

**Service Discovery Agency**

agency through which a Web service description is published and made discoverable.

**Service Requestor**

requests the execution of a Web service.

Operations of the Web Service Architecture

**Publish**: a service needs to publish its description such that a requestor can subsequently find it.

**Find**: the requestor queries a registry for the required service and retrieves a service description.

**Interact**: a service needs to be invoked and the results are returned.
Web services are based on 3 basic standards:

- **WSDL**: Web Services Description Language
- **UDDI**: Universal Description, Discovery and Integration
- **SOAP**: Simple Object Access Protocol

**Steps involved in providing and consuming a service**

1. A service provider describes its service using WSDL.
2. A service requestor queries the directory to locate a service and determine how to communicate with that service.
3. (Optional) Read service description to service requestor.

**Message Exchange Patterns**

Define the sequence of one or more messages exchanged between service requestor and service provider. Examples are: one-way, request/response, broadcast.

The Web service architecture may support different interaction scenarios.

- **Peer-to-Peer**
- **Direct Interaction**
- **Intermediary**
In the peer-to-peer scenario, each Web service instance serves in both the service requestor and service provider role.

Intermediaries may perform additional functions (besides the operations defined by the message exchange patterns) with a message such as routing, security, management.

Web services provide a standard means of communication among distributed software applications based on the Web technology. Standardization by the W3C community.

**Motivation - Example**

- **Service Oriented Architecture - SOA**
- **Web Services - Characteristics**
- **Web Services Architecture**
- **Simple Object Access Protocol (SOAP)**
- **Web Services Description Language (WSDL)**
- **Universal Description, Discovery, and Integration (UDDI)**
- **REST**
- **Web Service Composition**
- **Adopting Web Services**
- **Mashups**