Web Service Description Language (WSDL)

UNIT - IV

Web Service Description Language (WSDL)

- A WSDL describes point of contact for a service provider
- Also called *service endpoint or endpoint*
- *It provides a formal* definition of the endpoint interface
- Contains info to structure request messages, physical location (address) of the service
- *Two categories:*
  - abstract description
  - concrete description

Abstract Description

- An *abstract description - interface characteristics of the Web service without any reference to the technology used to host or* transmit messages

- Integrity of the service description preserved, irrespective of changes occur in underlying technology platform

- Three main parts of abstract description:
  - **portType**
  - **Operation**
  - **message**

Contd...

- PortType Section contains group of functions as **Operations**

- Each operation represents a specific action performed by service

- A service operation is comparable to a public method used by components

- Operations also have input and output parameters

- As Web services rely on messaging-based communication, parameters are represented as messages

- So, operation consists of a set of *input and output messages*

Concrete description

- Web service to execute logic, abstract interface definition should connected to real implemented technology

- To execute service application logic needs communication - so needs to connect to transport protocol

- This connection is defined in *concrete description portion of the* WSDL file

- It consist of 3 parts:
  - **Binding**
  - **Port**
  - **service**

Contd...

- A WSDL description's binding describes about physical connections – a possible transport technology the service can use to communicate

- Most common SOAP, but other protocols are also supported

- A binding can apply to entire interface or just a specific operation

- Port represents the physical address at which a service can be accessed with a specific protocol

- location information maintained independently from other aspects of concrete description

- service refers to group of related endpoints

Metadata and Service Contracts

- WSDL use XSD schemas to formalize the structure of incoming and outgoing messages

- Policy – supplemental service description document

- *Policies can provide rules, preferences, and processing details etc. (Metadata)*

- Service Contract - WSDL definition, XSD schema, Policy + Service Level Agreement (scope, quality, responsibilities agreed between the service provider and the service user)

- Service contract - a set of conditions that must be met and accepted by potential service requestor to enable successful communication

Semantic Descriptions

- Service Metadata does not service's behavioral characteristics such as:
    - how a service behaves under certain conditions
    - how a service will respond to a specific condition
    - what specific tasks the service is most suited for
- service semantics are assessed by either verbally by humans or reading supplementary documentation
- Semantic information need to be in a structured manner – for service requestors to evaluate and choose suitable service providers independently

Contd...

- Semantic information is of greater importance - when global knowledge of web service published is limited, within organization Web services grows

Private and Public Registries

- UDDI specifies standard for structuring registries

- Registries can be searched manually and accessed programmatically via a standardized API

- Each registry record consists of a ***Business Entity*** - *basic profile information about the organization / service provider*

- Also contains one or more ***Business Service areas*** *- description of the services offered by the* business entity

- Registry records store binding information in ***Binding Template,*** each business service can reference one or more binding templates

- In a Web service binding template references a **tModel**, it provides pointers to actual service descriptions

BusinessEntity

Company Info

- `<businessEntity  businessKey="..." operator="http://www.ibm.com" authorizedName="John Doe">`

Contacts

- `<name>Acme Company</name>`
- `<description> We create cool Web services </description> <contacts>`
- `<contact useType="general info">`
- `<description>General Information</description>`
- `<personName>John Doe</personName>`
- `<phone>(123) 123-1234</phone>`
- `<email>jdoe@acme.com</email>`
- `</contact> </contacts>`

Contains information about the company itself, including contact information, industry categories, business identifiers, and a list of services provided

Business Identifier

Business Category

- `<businessServices> ... </businessServices>`
- `<identifierBag> <keyedReference tModelKey="..." name="D-U-N-S" value="123456789" /> </identifierBag>`
- `<categoryBag> <keyedReference tModelKey="...." name="NAICS"  value="111336" /> </categoryBag>`
- `</businessEntity>`

Business Service

- `<businessEntity ...>` → represents an individual web service provided by the business entity
- `<contacts> ...</contacts>`
- `<businessServices serviceKey="UUID:xx" businessKey="UUID:yyy">`
- `<name>Hello World Web Service</name>`
- `<description>A friendly Web service</description>`
- `<bindingTemplates> ... </bindingTemplates>`
- `</businessServices>`
- `<identifierBag> <keyedReference tModelKey="....." />` `</identifierBag>`
- `<categoryBag> <keyedReference tModelKey="...." />`
- `</categoryBag>`
- `</businessEntity>`

what type of web service

Info about how to bind WS

taxonomical categories it belongs to

UUID unique to Business entity / Business Services

Binding Template

- `<bindingTemplate serviceKey="uuid:xxx" bindingKey="uuid:xxx">`
- `<description>`Hello World SOAP Binding`</description>` `<accessPoint` URLType ="http"> http://localhost:8080 `</accessPoint>`
- `<tModelInstanceDetails>` `<tModelInstanceInfo` tModelKey="uuid:xxx">
- `<instanceDetails>`
- `<overviewDoc>`
- `<description>` references the description of the WSDL service definition `</description>`
- `<overviewURL>` http://localhost/helloworld.wsdl `</overviewURL>` `</overviewDoc>`
- `</instanceDetails>`
- `</tModelInstanceInfo>`
- `</tModelInstanceDetails>`
- `</bindingTemplate>`

- represents the actual implementation of the web service
- can have multiple binding templates
- service may specify different implementations of the same service
- each bound to a different set of protocols or a different network address

Technical Model (t-Model)

- tModel is a way of describing various business, service, and template structures stored within the UDDI registry
- Any abstract concept can be registered within the UDDI as a tModel
- Later tModel can be associated with one of that business service's binding templates

- Example:
- <tModel tModelKey="uuid:xyz987…" operator="http://www.ibm.com" authorizedName="John Doe">
- <name>HelloWorldInterface Port Type</name>
- <description> An interface for a friendly Web service </description> <overviewDoc>
- <overviewURL> http://localhost/helloworld.wsdl </overviewURL> </overviewDoc>
- </tModel>