

Web servisi

UDIS 2011/2012

WSDL

- WSDL (*Web Services Description Language*) je jezik za opisivanje sučelja Web servisa u XML formatu.
<http://www.w3.org/2002/ws/>
- WSDL sadrži četiri vrste informacija:
 - Informacije koje opisuju sve javno dostupne funkcije u sučelju servisa
 - Informacije o tipovima podataka za upite i odgovore
 - Informacije o vezivanju za transportni protokol koji se treba koristiti
 - Informacije o lokaciji servisa

Npr. <http://soap.amazon.com/schemas2/AmazonWebServices.wsdl>

WSDL struktura

- WSDL dokument sadrži 6 osnovnih elemenata:
 1. *definitions* - izvorni element svakog WSDL dokumenta je element *definitions* unutar kojeg se navodi naziv web servisa i imenski prostori.
 2. *types* - element definira sve tipove podataka potrebne za slanje i primanje poruka. WSDL nije ograničen na neki određen način kodiranje podataka, ali ima *defaultni* način kodiranja koji je određen XML shemom.
 3. *message* - element se može pojaviti više puta u WSDL dokumentu. Sastoji se od jednog ili više *part* elemenata. *Part* elementi referenciraju tipove definirane unutar *types* elementa. *Part* elementi ustvari opisuju sadržaj poruke.

WSDL struktura

4. *portType* - predstavlja skup operacija. Sadrži elemente *operation*. *portType* može npr. udružiti jednu *request* i jednu *response* poruku u jednu *request/response* operaciju.
5. *binding* - svi elementi prije *binding* elementa predstavljaju apstraktni opis web servisa. S *bindingom* počinje opis konkretne implementacije. *Binding* definira način prijenosa poruka, tj. protokol koji će se pritom koristiti.
6. *service* - element sadrži atribut *name*, koji predstavlja naziv web servisa i element *port*. *Port* element sadrži *binding* atribut i unutar njega se navodi adresa web servisa.

WSDL struktura

- Dva dodatna pomoćna elementa su:
 1. *documentation* - element sadrži dokumentaciju za bolje razumijevanje web servisa.
 2. *import* - element služi za uvođenje novih dokumenata unutar WSDL dokumenta koji su onda sadržani unutar početnog WSDL dokumenta. To omogućuje modularnu strukturu WSDL dokumenta. Na taj način više WSDL dokumenata može koristiti neka osnovne elemente iz zajedničkog jedinstvenog dokumenta.

Primjer WSDL-a

- Amazon web servis nudi 23 operacije, npr.:
 - AuthorSearchRequest operacija za pretraživanje knjiga po autoru
 - KeywordSearchRequest operacija za pretraživanje po ključnoj riječi
- <http://soap.amazon.com/schemas2/AmazonWebServices.wsdl>

WSDL

```
<?xml version="1.0"?>
<definitions name="GoogleSearch"
targetNamespace="urn:GoogleSearch"
xmlns:typens="urn:GoogleSearch"
xmlns:xsd="http://www.
w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.
org/wsdl/soap/"
xmlns:soapenc="http://schemas.xmlso
ap.org/soap/encoding/"
xmlns:wsdl="http://schemas.xmlsoap.
org/wsdl/"
xmlns="http://schemas.xmlsoap.org/
wsdl/">
...
</definitions>
```

- Element *definitions* je root element WSDL dokumenta.
- Definira imenski prostor dokumenta preko atributa *targetNamespace*.
- Imenski prostor ima istu ulogu kao i kod XML dokumenta, da se kod korištenja više WSDL dokumenata ne narušava jedinstvenost identifikatora.

WSDL

- Element *definitions* pored atributa *name* i *targetNamespace* može imati dodatne reference na imenske prostore koji se koriste u WSDL dokumentu. Tu su još standardni imenski prostori od WSDL, SOAP,...
- Nakon definiranja atributa *name* sa *GoogleSearch* i elementa *targetNamespace* sa *urn:GoogleSearch* dodjeljuje se *targetNamespaceu* skraćeni naziv *typens* koji nam omogućuje direktno pozivanje elemenata unutar našeg dokumenta.

WSDL

```
<types>
<xsd:schema
  xmlns="http://www.w3.org/2001/XMLSchema"
  targetNamespace="urn:GoogleSearch">
<xsd:complexType name="GoogleSearchResult">
<xsd:all>
<xsd:element name="documentFiltering"
  type="xsd:boolean"/>
<xsd:element name="searchComments"
  type="xsd:string"/>
<xsd:element name="estimatedTotalResultsCount"
  type="xsd:int"/>
<xsd:element name="estimatedIsExact"
  type="xsd:boolean"/>
<xsd:element name="resultElements"
  type="typens:ResultElementArray"/>
<xsd:element name="searchQuery" type="xsd:string"/>
<xsd:element name="startIndex" type="xsd:int"/>
<xsd:element name="endIndex" type="xsd:int"/>
<xsd:element name="searchTips" type="xsd:string"/>
<xsd:element name="directoryCategories"
  type="typens:DirectoryCategoryArray"/>
<xsd:element name="searchTime" type="xsd:double"/>
</xsd:all>
</xsd:complexType>
...
</xsd:schema>
</types>
```

- Unutar elementa *types* definiraju se tipovi podataka koji će se koristiti u WSDL dokumentu.
- Standardni način kodiranja je opisan [XML shemom Datatypes](http://www.w3.org/2001/XMLSchema) (<http://www.w3.org/2001/XMLSchema.xsd>).
- Moguće je korišćenje i proizvoljnih načina kodiranja.

WSDL

```
<message
  name="doGoogleSearchRes
  ponse">
  <part name="return"
    type="typens:GoogleSearch
    Result" />
</message>
```

- WSDL dokument može imati više *message* elemenata.
- Unutar svakog *message* elementa nalazi se više *part* elemenata. Oni opisuju dijelove poruke, tj. parametre ili podatke koji su sadržani u poruci i koji se prenose s njom. Podaci se opisuju preko atributa *type*.

WSDL

```
<portType
  name="AmazonSearchPort"
>
<operation
  name="KeywordSearchRequest">
  <input
    message="typens:Keyword
      SearchRequest"/>
  <output
    message="typens:Keyword
      SearchResponse"/>
</operation>
```

- Element *portType* sadrži ustvari definiciju metoda koje nudi servis (definirano porukama između klijenta i servera).
- Svaki *portType* element zadrži jedan ili više *operation* elemenata.

WSDL

- Unutar *operation* elementa nalazi se jedan *input* i/ili *output* element. Kroz različite načine uporabe *input* i *output* elemenata određuju se četiri vrste *portType* operacija:
 1. *Input-Output* operacije (*Request-Response*): Operacija u klasičnom RPC stilu. Klijent postavi pitanje, a server vrati nazad odgovor ili poruku o greški.

```
<operation name=..>  
  <input message= .../>  
  <output message= .../>  
</operation>
```
 2. *Input-Only* operacije (*One-Way*): Klijent samo pošalje poruku serveru.

```
<operation name=..>  
  <input message= .../>  
</operation>
```

WSDL

3. *Output-Input* operacije (*Solicit-Response*):
Server postavi klijentu pitanje i klijent vrati
natrag odgovor ili poruku o greški.
 <operation name=..>
 <output message= .../>
 <input message= .../>
 </operation>
4. *Output-Only* operacije (*Notification*): Server
samo pošalje poruku klijentu.
 <operation name=..>
 <output message= .../>
 </operation>

WSDL

- Element *binding* se koristi za definiranja protokola koji će se koristiti za prijenos poruka.
- Unutar jedne *binding* komponente smije se koristiti samo jedan protokol.
- Ako želimo opisati web servis koji je u stanju koristiti različite protokole za svaki protokol treba definirati poseban *binding* element. Svaki *binding* element sadrži atribut *type* koji referencira jedan prethodno definiran *portType* element.

WSDL

```
<binding
  type="typens:AmazonS
earchPort"
  name="AmazonSearch
Binding">
```

```
<soap:binding
  transport="http://sche
mas.xmlsoap.org/soap
/http" style="rpc"/>
```

- WSDL ima ugrađene ekstenzije za SOAP *binding*.
- *Soap:binding* element definira uporabu SOAP protokola. Atribut *style* određuje opći stil SOAP poruka. U ovom primjeru to je RPC.

WSDL

- Ako koristimo *style* rpc to znači da će tijelo *SOAP request* i *response* poruke sadržavati XML element s nazivom funkcije. Unutar tog elementa nalazit će se vrijednosti parametara koji se predaju funkciji odnosno povratna vrijednost funkcije.
- Da je *style* atribut postavljen na *document*, tada bi se *SOAP request* i *response* poruke sastojale od XML dokumenata.

WSDL

- Atribut *transport* određuje način transporta SOAP poruka.
- URL <http://schemas.xmlsoap.org/soap/http> definira transport SOAP poruka preko HTTP protokola.

WSDL

```
<service
  name="AmazonSearchService">
  <port
    name="AmazonSearchPort"
    binding="typens:AmazonSearchBinding">
    <soap:address
      location="http://soap.amazon.com/onca/soap2"/>
    </port>
  </service>
```

- Element *service* određuje lokaciju web servisa i sastoji od jednog ili više *port* elemenata.

WSDL

- Svaki *port* element ima atribut *binding* koji referencira neki *binding* koji je definiran prije u dokumentu.
- Osim toga unutar *porta* mora uvijek biti definirana adresa. Definiranje adrese se izvodi preko posebnih elemenata koji ovise o korištenom protokolu. Radi korištenog SOAP protokola, u primjeru se za definiranje adrese koristi `soap:address` element.