

Luento 8: XML-tuki ohjelmointikielissä & Web-palvelut

AS-0.110 XML-kuvauskielten perusteet

Janne Kalliola

1

XML-tuki ohjelmointikielissä ja Web-palvelut

- XML-tuki ohjelmointikielissä
 - Java
 - PHP
 - C, C++
 - Perl
 - .NET, C#, Visual Basic
 - Muut kielet
- Web-palvelut
 - SOAP
 - WSDL
 - UDDI
 - Web-palvelujen käyttö

2

XML-tuki ohjelmointikielissä

- Kaikissa olennaisissa ohjelmointikielissä on mahdollisuus käyttää XML-dokumentteja
 - perustasolla tuki perustuu yleensä DOM- ja SAX-malleihin
 - näiden lisäksi useissa kielissä on korkeamman tason rajapintoja, esimerkiksi olioiden sarjallistamiseksi XML-dokumenteiksi ja takaisin

3

Java

- Javassa lienee paras saatavilla oleva XML-tuki
 - useat standardi- ja de-facto-standardi-API:t toteutettu
- Kirjastojen lisäksi Javalla on toteutettu useita hyviä XML-sovelluksia
 - **FOP** - XSL-FO-dokumenttien käsittelyyn
 - **Batik** – SVG-kuvien näyttämiseen, luontiin ja manipulointiin
 - **Cocoon** – dynaamisten julkaisujärjestelmien ja WWW-sivustojen rakentamiseen
- JSP-sivut ja servletit voivat käyttää samoja kirjastoja ja ohjelmistopaketteja kuin normaalitkin Java-ohjelmat

4

PHP

- PHP on skriptipohjainen kieli dynaamisten www-sivujen ja –sivustojen laadintaan
 - kielen rakenteet sijoitetaan yleensä HTML-koodin keskellä
 - PHP:lla voi tuottaa myös dynaamisia XML-dokumentteja
 - PHP:ssa on hyvä tuki HTTP-liikenteelle ja tietokannoille
 - PHP on yleisimpiä WWW-hotelleissa tarjottavia ohjelmointiympäristöjä
- PHP:ssa on useita erilaisia tapoja muokata ja käsitellä XML-dokumentteja
 - osa moduuleista on vielä kokeellisia ja kaikkia ei ole tarjolla hotelleissa
 - PHP 5 selkeyttää tilannetta

5

C

- Koska C-kieli ei ole oliopohjainen, DOM- ja SAX-määrittelyt eivät kelpaa sellaisenaan
 - XML-peruskäsittely löytyy kirjastoista libxml ja libxml2
 - rajapinnat muistuttavat DOM-rajapintoja
 - myös SAX-tyyliset rajapinnat ovat olemassa
- Perustuen lisäksi C:lle löytyy suuri joukko erikoistuneempia kirjastoja
 - libxsl – XSLT-tyylisivujen käyttö
 - xmlrpc-c – XML-RPC -kutsut
- Kaikki PHP:n moduulit pohjautuvat näihin kirjastoihin
 - moduuli tarjoaa yleensä osajoukon kirjaston kyvyistä
 - kaikki PHP:lle tarjolla oleva löytyy siis myös C:lle

6

C++

- DOM-rajapinta löytyy C++
 - esimerkiksi Apache Foundationin Xerces C++
 - sama toiminnallisuus kuin Java-versiossa
 - myös Xalan C++ XSLT:ta varten on olemassa
- SOAP on tarjolla esimerkiksi EasySoap++ -kirjastolla
 - on olemassa muitakin toteutuksia
- C++ voi käyttää myös kaikkia C:n kirjastoja

7

Perl

- Perlin XML-tuki pohjautuu PHP:n tavoin moduuleihin
 - sama toiminnallisuus on saatettu toteuttaa eri tavoilla eri moduuleihin
 - Perliin löytyy ainakin viisi erilaista XML-jäsennintä
 - suureen osaan C-kirjastoista löytyy myös Perl-rajapinta
 - esimerkiksi libxml tai XSLT-kirjastot
 - myös SOAP on tuettu

8

.NET

- .NET on Microsoftin määrittelemä viitekehys, joka käyttää Web-palveluita järjestelmien kytkentään
 - .NET on tarjolla sekä kehitystyökaluina (Visual Studio yms.) ja palvelimina (IIS ja Exchange)
 - kaikki Microsoftin ohjelmointiympäristöt voivat käyttää .NET-palveluja
 - Visual Basic, C++ ja C#
 - Microsoft tarjoaa valmiit kirjastot
- .NET ei ole suljettu
 - palveluja käytetään normaaleilla SOAP-mekanismeilla
 - Java ja Unix-ohjelmat voivat kytkeytyä .NET Web-palveluihin ja päinvastoin

9

C#

- C# on Microsoftin kehittämä uusi ohjelmointikieli
 - muistuttaa ominaisuuksiltaan Javaa, tiettyjä asioita on parannettu
 - C#-ohjelmat käännetään Common Language Infrastructuureen (CLI) ja ne suoritetaan CLI-virtuaalikoneessa
 - CLI-ympäristöön voidaan kääntää myös muillakin ohjelmointikielillä
- C# sisältää sisäisen XML-tuen System.XML -joukossa (assembly)
 - tuki DOM- ja SAX-tyyliselle jäsenyykselle
 - tapahtumamallin sijaan käytetään pyyntömallia (pull mechanism) XML:n jäsentämiseen
 - sama malli löytyy myös Javalle (XPP2)
 - samanlaisia kirjastoja ei ole muissa kielissä

10

Visual Basic

- Visual Basicia käytetään laajalti Windows-ympäristöissä
 - piilottaa järjestelmien monimutkaisuudet ja ohjelmoija voi keskittyä sovellukseen liittyvien ongelmien ratkintaan
 - ohjelmat voidaan suorittaa joko palvelimissa tai omina sovelluksinaan
 - Visual Basicista löytyy sidonnat .NET-ympäristöön
- MSXML-kirjaston avulla Visual Basic voi käyttää DOM- ja SAX-rajapintoja
 - myös XSLT-tuki on samassa kirjastossa
 - SOAP-tuki löytyy erikseen

11

Muut kielet

- Suuressa osassa kielistä on sidonta joukkoon C-kielen XML-kirjastoja
 - esimerkiksi Python, Pascal, Ruby, Tcl
- Aina koko kirjaston toiminnallisuus ei ole käytettävissä
 - riippuu osittain myös kielen ilmaisuvoimasta
 - kaikkia C:n rakenteita ei voida helposti siirtää rajoitetummille kielille
 - osoittimet, structit ja unionit

12



HOHDE

Web-palvelut

- Web-palveluissa kaksi sovellusta vaihtaa tietoa verkon yli käyttäen hyväkseen XML:a ja jotakin yleiskäyttöistä protokollaa
 - SOAP määrittää Web-palvelujen toimintamallin
 - WSDL määrittää palveluiden rajapinnan
- Näiden lisäksi voidaan puhua laajennetusti Web-palveluista, joissa pyyntö lähetetään normaalina HTTP GET-pyyntö ja vastaus saadaan XML-dokumenttina
 - XML over HTTP

14

This slide features a background image of a large tree with green and yellow leaves. The HOHDE logo is in the top right corner, and the title 'Web-palvelut' is in the top left. A bulleted list describes web services and XML over HTTP. The number '14' is in the bottom right corner.

SOAP

- Simple Object Access Protocol on XML-pohjainen protokolla koneiden väliseen kommunikointiin tietoverkoissa
 - SOAP on W3C:n kehittämä ja tällä hetkellä suosituksen versio on 1.2

15

Sidonta protokollaan

- SOAP ei määritä mitään tiettyä protokollaa viestien siirtämiseen
 - yleensä käytetään HTTP:ta
- SOAP-viestintä sidotaan tiettyyn protokollaan (binding)
 - tämä sidonta määrittää, kuinka viestit siirretään verkon solmulta toiselle
- SOAP-viesti voi kulkea usean SOAP-solmun kautta
 - SOAP määrittää joukon elementtejä ja attribuutteja viestinnän kontrollointiin

16

SOAP-viesti ja -kirjekuori

- SOAP-viesti (message) on XML-dokumentti, joka lähetetään sovellukselta toiselle
 - yleensä verkon yli
 - viesti on yksisuuntainen, toinen sovelluksista (SOAP node) on lähettäjä (SOAP sender) ja toinen vastaanottaja (SOAP receiver)
- SOAP-viesti lähetetään SOAP-kirjekuoressa (envelope)
 - kirjekuori sisältää valinnaiset otsikkotiedot (SOAP header) ja itse viestin (SOAP body)

17

Kirjekuoren koodaus

- SOAP-kirjekuori on XML-dokumentti, jonka juurielementti on `env:Envelope`
 - nimiavaruuden URI on <http://www.w3.org/2003/05/soap-envelope>
- SOAP-otsikot sijoitetaan elementin `env:Header` sisään
- Vastaavasti, viesti on elementin `env:Body` sisässä

18

Esimerkki kirjekuoresta

```
<?xml version='1.0' ?>
<env:Envelope xmlns:env="http://www.w3.org/
2003/05/soap-envelope">
  <env:Header>
    ...
  </env:Header>
  <env:Body>
    ...
  </env:Body>
</env:Envelope>
```

19

Header ja body

- SOAP-otsikko mahdollistaa sovelluksen lähettää itse viestistä riippumatonta tietoa vastaanottajalle
 - tällä tavalla voidaan koodata esimerkiksi sovelluskohtaisia toimintaohjeita
 - otsikkotiedoissa voidaan käyttää omaa XML-pohjaista syntaksia
- SOAP-viesti sisältää XML-sirpaleen, jossa määritellään viestin sisältö
 - yleensä omassa nimiavaruudessaan

20

SOAP-viestien vaihto

- Yksisuuntaiselle kommunikaatiolla ei tee käytännössä juuri koskaan mitään
 - tämän takia keskustelevat sovellukset ovat sekä lähettäjiä että vastaanottajia
 - SOAP-viestissä saapuvaan kyselyyn tai pyyntöön vastataan erillisessä SOAP-viestissä lähetettävällä vastauksella

21

Virheviestit

- SOAP määrittää myös toimintoja virhetilanteisiin
 - koska SOAP ei ota kantaa käytettyyn protokollaan, kaikkia virhetilanteita ei välttämättä edes huomata
 - järjestelmien täytyy kyetä toimimaan epäluotettavassa ympäristössä tai protokolla täytyy vaihtaa luotettavampaan
- Virhetilanne kirjataan viestin sisään elementillä `env:Fault`, jonka sisällä täytyy olla elementit `env:Code` ja `env:Reason`
 - `env:Code` määrittää virhekoodin
 - `env:Reason` kertoo virheen syyn
 - näiden lisäksi on mahdollista käyttää elementtiä `env:Detail` kuvaamaan sovelluskohtaista tietoa
 - `env:Node` voi kertoa sovelluksen URI:n

22

Esimerkki virheviestistä

```
<env:Envelope
  xmlns:env="http://www.w3.org/2003/05/soap-envelope"
  xmlns:rpc='http://www.w3.org/2003/05/soap-rpc'>
  <env:Body>
    <env:Fault>
      <env:Code>
        <env:Value>env:Sender</env:Value>
        <env:Subcode>
          <env:Value>rpc:BadArguments</env:Value>
        </env:Subcode>
      </env:Code>
      <env:Reason>
        <env:Text xml:lang="en-US">Processing error</env:Text>
      </env:Reason>
      <env:Detail>
        <e:myFaultDetails
          xmlns:e="http://travel.example.org/faults">
          <e:message code="999">Name does not match
card</e:message>
        </e:myFaultDetails>
      </env:Detail>
    </env:Fault>
  </env:Body>
</env:Envelope>
```

23

SOAP:n käyttö

- SOAP-viestejä ei yleensä tulkita itse, vaan käytetään erillistä SOAP-kirjastoa
 - lukee viestin sisään ja tarjoaa sen sisällön jossakin tietorakenteessa
 - ja vastaavasti osaa muuntaa tietorakenteita SOAP-viesteiksi
 - kirjastoissa on tuki myös virhetilanteille

24

WSDL

- Web Serviced Description Language (WSDL) on kieli, jolla voidaan määrittää Web-palveluiden rajapintoja
 - tässä Web-palvelulla viitataan konekäytettävään palveluun (web services)
 - WSDL sisältää rajapinnan kuvauksen lisäksi tietoa palvelun semantiikasta ja käytännön asioista (kytkeytymistapa, palvelun osoite)
- WSDL-määrittelyn avulla Web-palvelun rajapintaa vastaan voidaan laatia sovellus
 - sovellus kytkeytyy palveluun ja käyttää sen tarjoamia rajapintoja

25

Web-palvelun määrittäminen

- Yksi WSDL-dokumentti voi määrittää useita Web-palveluita
 - jokainen palvelu määritetään elementillä portType
 - palvelun yhteydessä määritetään käytössä olevat operaatiot ja niihin liittyvät viestit
- Operaatio voi olla yksi- tai kaksisuuntainen
 - one-way – operaatio ottaa vastaan viestejä, mutta ei vastaa
 - request-response – operaatio ottaa vastaan viestejä ja vastaa niihin
 - solicit-response – operaatio voi lähettää viestin ja odottaa vastausta
 - notification – operaatio voi lähettää viestin ja ei odota vastausta

26

Esimerkki yksisuuntaista palvelusta



```
<message name="newTermValues">
  <part name="term" type="xs:string"/>
  <part name="value" type="xs:string"/>
</message>

<portType name="glossaryTerms">
  <operation name="setTerm">
    <input name="newTerm"
      message="newTermValues"/>
  </operation>
</portType >
```

27

Esimerkki kaksisuuntaisesta palvelusta



```
<message name="getTermRequest">
  <part name="term" type="xs:string"/>
</message>

<message name="getTermResponse">
  <part name="value" type="xs:string"/>
</message>

<portType name="glossaryTerms">
  <operation name="getTerm">
    <input message="getTermRequest"/>
    <output message="getTermResponse"/>
  </operation>
</portType>
```

28

SOAP-sidonta

- Rajapinta kytketään SOAP-malliin käyttämällä elementtiä binding
 - jokaista operaatiota varten täytyy määrittää SOAP-toiminto
 - samalla määritetään kuinka toimintoon liittyvät viestit koodataan SOAP-viestin sisään
 - tässä yhteydessä määritetään myös käytettävä siirtoprotokolla
 - palvelun URI asetetaan

29

Esimerkki sidonnasta

```
<binding type="glossaryTerms" name="b1">
  <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
  <operation>
    <soap:operation
soapAction="http://example.com/getTerm"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>
```

30

UDDI

- Universal Description, Discovery and Integration on luettelopalvelu, jonne voidaan rekisteröidä Web-palveluita
 - UDDI sisältää Web-palveluiden WSDL-kuvaukset
- Periaatteessa voisi olla olemassa julkisia UDDI-palvelimia, joista voisi pyytää Web-palveluiden osoitteita ja rajapintakuvauksia
 - käytännössä nykyään Web-palveluita ei rekisteröidä (ainakaan julkisesti) UDDI:iin
 - sen sijaan, Web-palvelun dokumentaatio antaa ohjeet palveluun kytkeytymiseen

31

Web-palvelujen käyttö

- Käytännössä Web-palvelut toteutuaan puoliautomaattisesti generoimalla koodia
 - Web-palvelun WSDL-kuvauksen pohjalta laaditaan valmis toteutus, joka pystyy lähettämään ja vastaanottamaan SOAP- viestejä
 - tämä toteutus käyttää sisäisesti edellisen kalvon kirjastoa
 - jokainen WSDL:ssa kuvattu palvelu muuntuu metodiksi
 - tätä metodia voidaan kutsua ohjelmassa normaalisti
 - SOAP:n käyttö saattaa lisätä joitakin poikkeustilanteita, joita ei normaalisti olisi
- Web-palveluun liittymisen voi tehdä myös käsin
 - kaikille ohjelmointikielille tai ympäristöille ei ole välttämättä automaattista generointia tai muutakaan SOAP-tukea

32

Web-palvelujen laadinta

- Vastaavasti Web-palvelujen laadinnassa ei tarvitse enää kirjoittaa WSDL-dokumenttia
 - sovellukseen laaditaan normaali rajapinta ohjelmointikielen omilla primitiiveillä
 - tämä rajapinta muunnetaan SOAP-rajapinnaksi erillisellä generaattorilla
 - tuotetaan WSDL-kuvaus
 - rakennetaan komponentti, joka tulkitsee SOAP-viestejä ja kutsuu niiden pohjalta rajapinnan metodeja
- Mikään ei tosin estä toteuttamasta kaikkea käsin

Kysymyksiä? Kommentteja?