

# XML Namespaces

# Razlozi za uvođenje

---

- svako može da definiše sopstvenu XML gramatiku
- ukoliko želimo da koristimo različite gramatike može doći do poklapanja imena

# Razlozi za uvođenje

- primer: podaci o osobi, gde je biografija formatirana po XHTML-u

```
<?xml version="1.0" encoding="utf-8"?>
<person>
  <name>
    <title>Sir</title>
    <first>John</first>
    <middle>Fitzgerald Johansen</middle>
    <last>Doe</last>
  </name>
  <position>Vice President of Marketing</position>
  <resume>
    <html>
      <head><title>Resume of John Doe</title></head>
      <body>
        <h1>John Doe</h1>
        <p>John's a great guy, you know?</p>
      </body>
    </html>
  </resume>
</person>
```

dva elementa title sa  
različitim značenjem

# Razlozi za uvođenje

---

- šta je problem?
  - dva elementa `title` se formalno ne razlikuju
  - aplikacije nemaju način da prepoznaju razliku
  - pretraga po elementu `title` može biti dvosmislena

# Razloži za uvođenje

---

- šta može biti rešenje?
  - učiniti da nazivi svih elemenata budu jedinstveni
  - možemo svakom elementu dodati prefiks
  - zavisno od toga gde su elementi definisani, prefiks će u primeru biti
    - `pers` (za elemente koji služe za opis osobe)
    - `xhtml` (za elemente iz XHTML formata)

# Razlozi za uvođenje

- prethodni primer sa prefiksima

```
<?xml version="1.0" encoding="utf-8"?>
<pers:person>
  <pers:name>
    <pers:title>Sir</pers:title>
    <pers:first>John</pers:first>
    <pers:middle>Fitzgerald Johansen</pers:middle>
    <pers:last>Doe</pers:last>
  </pers:name>
  <pers:position>VP of Marketing</pers:position>
  <pers:resume>
    <xhtml:html>
      <xhtml:head>
        <xhtml:title>Resume of John Doe</xhtml:title>
      </xhtml:head>
      <xhtml:body>
        <xhtml:h1>John Doe</xhtml:h1>
        <xhtml:p>John's a great guy, you know?</xhtml:p>
      </xhtml:body>
    </xhtml:html>
  </pers:resume>
</pers:person>
```

sada nema  
kolizije imena

# Pojam prostora imena

---

- skup elemenata koji imaju isti prefiks nazvaćemo prostor imena (*namespace*)
- u prethodnom primeru, biće dva prostora imena:
  - pers
  - xhtml

# Pojam prostora imena

---

- prostor imena  $\neq$  tip dokumenta
  - prostor imena je skup elemenata
  - jedan tip dokumenta može uključivati elemente iz više prostora
  - jedan element može biti korišćen u više tipova dokumenata



# Identifikacija prostora imena

---

- koncept prefiksa nije dovoljno robustan
  - više ljudi može da se odluči da koristi prefiks `pers`
  - morala bi postojati organizacija koja se bavi administracijom prefiksa
    - slično kao administriranje Internet domena
- umesto da se formira novi sistem za administraciju prefiksa, može da se koristi postojeći - Internet

# Identifikacija prostora imena

---

- prostori imena identifikuju se nazivom
- naziv je niz znakova u URI formatu
- URI - Uniform Resource Identifier

# Uniform Resource Identifier (URI)

---

- resurs je sve što ima identitet
  - fizički objekti (npr. knjiga)
  - digitalni objekti (fajlovi)
  - apstraktni pojmovi (stanje saobraćaja u Novom Sadu)
- resurs može biti dostupan putem Interneta (npr. fajl), ali i ne mora (npr. autor fajla)
- URI standard, IETF RFC 2396, opisuje identifikatore resursa
- URI može biti formiran kao
  - URN
  - URL

# Uniform Resource Locator (URL)

---

- standard definisan u IETF RFC 1738
- predstavlja podatke koji se mogu upotrebiti za dobavljanje resursa
- URL format - protokol:putanja-do-resursa
- primeri  
`http://www.ns.ac.yu/stara/fakulteti.htm`  
`mailto:xyz@gmail.com`

# Universal Resource Name (URN)

---

- IETF RFC 2141
- definiše trajno ime resursa koje ne zavisi od njegove lokacije
- URN format: `urn:nid:nss`
  - `urn`: fiksno
  - `nid`: namespace identifier
  - `nss`: namespace specific string
- primer - recimo da se JMBG brojevi koriste kao URN
  - `nid` bi mogao da glasi `rs.jmbg`
  - `nss` bi bio JMBG broj konkretne osobe
  - `urn:rs.jmbg:3107006805027`

# Identifikacija prostora imena

---

- zašto se u praksi za identifikaciju prostora imena koriste URL?
  - lakše je napraviti jedinstveno ime (firma već poseduje svoj Internet domen)
  - iako se ne može sprečiti zloupotreba (da jedna firma koristi domen druge kada formira URL za identifikaciju prostora imena), izbegavaju se slučajne kolizije
  - DTD koji definiše elemente iz datog prostora imena se može zaista i postaviti na lokaciju na koju pokazuje URL - na taj način definicije elemenata postaju javne i odmah dostupne
- identifikator prostora imena ne mora da nosi nikakvo značenje!

# Identifikacija prostora imena

---

- ako se koristi URL format za identifikator, uvek se koristi puna adresa, dakle nešto kao  
`http://www.mojafirma.com/primer`
- XML aplikacije ne smatraju jednakim sledeća imena:  
`http://www.mojafirma.com/primer`  
`http://mojafirma.com/primer`  
`www.mojafirma.com/primer`

# Identifikacija elemenata u prostoru imena

---

- primer: elementi koji opisuju osobu pripadaju namespace-u sa imenom `http://www.ftn.ns.ac.yu/dtds/person.dtd`
- puno ime elementa `name` iz tog prostora glasilo bi `http://www.ftn.ns.ac.yu/dtds/person.dtd:name`
  - ovo nije validno XML ime
  - vrlo je nezgrapno za pisanje



# Identifikacija elemenata u prostoru imena

---

- elementi se identifikuju **kvalifikovanim** imenima (*qualified name*, *QName*)
- kvalifikovano ime se sastoji iz dva dela
  - lokalno ime
  - namespace
- u XML dokumentima se, umesto naziva prostora imena, koristi (kraći, XML-dozvoljeni) prefiks
- prefiks mora da se poveže sa svojim prostorom imena
- ime prefiksa više nije bitno
  - može biti bilo koje XML-dozvoljeno ime

# Identifikacija elemenata u prostoru imena

- primer

person pripada prostoru  
označenim prefiksom pers

```
<pers:person  
xmlns:pers="http://www.ftn.ns.ac.yu/dtds/person.dtd">
```

prefiks pers zamenjuje URL  
<http://www.ftn.ns.ac.yu/dtds/person.dtd>

prefiks sada ne služi za identifikaciju prostora imena, već samo  
da poveže elemente sa URI identifikatorom!

# Identifikacija elemenata u prostoru imena

- primer

```
<pers:person
  xmlns:pers="http://www.ftn.ns.ac.yu/dtds/person.dtd">
  <pers:name>
    <pers:title>Sir</pers:title>
    <pers:first>John</pers:first>
    <pers:last>Doe</pers:last>
  </pers:name>
  ...
</pers:person>
```



i zatvarajući tag mora imati prefiks!

# Identifikacija elemenata u prostoru imena

```
<?xml version="1.0" encoding="utf-8"?>
<pers:person
  xmlns:pers="http://www.ftn.ns.ac.yu/dtds/person.dtd"
  xmlns:xhtml="http://www.w3.org/1999/xhtml">
  <pers:name>
    <pers:title>Sir</pers:title>
    <pers:first>John</pers:first>
    <pers:last>Doe</pers:last>
  </pers:name>
  <pers:position>VP of Marketing</pers:position>
  <pers:resume>
    <xhtml:html>
      <xhtml:head>
        <xhtml:title>Resume of John Doe</xhtml:title>
      </xhtml:head>
      <xhtml:body>
        <xhtml:h1>John Doe</xhtml:h1>
        <xhtml:p>John's a great guy, you know?</xhtml:p>
      </xhtml:body>
    </xhtml:html>
  </pers:resume>
</pers:person>
```

# Default namespace

- šta raditi sa starim XML dokumentima gde se ne koriste namespace prefiksi?
- postoji pojam **podrazumevanog** (default) prostora imena
  - namespace kao i svaki drugi
  - njegovi elementi se navode bez prefiksa
- primer

```
<person xmlns="http://www.ftn.ns.ac.yu/dtds/person.dtd">
  <name>
    <title>Sir</title>
  </name>
</person>
```

Ovo više nije kvalifikovano ime (ima samo lokalni deo).  
Pojam univerzalno ime (UName) obuhvata QName i ime u default namespace-u

# Deklarisanje prostora imena

---

- u jednom elementu može se deklarirati više prostora imena  

```
<pers:person  
  xmlns:pers="http://www.ftn.ns.ac.yu/dtds/person.dtd"  
  xmlns:xhtml="http://www.w3.org/1999/xhtml">
```
- ne moraju svi prostori imena biti deklarirani u korenskom elementu
- prefiksi imaju značenje samo *u okviru elementa* u kome su definisani!

# Deklarisanje prostora imena

- primer

```
<?xml version="1.0" encoding="utf-8"?>
<pers:person
  xmlns:pers="http://www.ftn.ns.ac.yu/dtds/person.dtd">
  <pers:name>
    <pers:title>Sir</pers:title>
    <pers:first>John</pers:first>
    <pers:last>Doe</pers:last>
  </pers:name>
  <pers:position>VP of Marketing</pers:position>
  <pers:resume>
    <xhtml:html xmlns:xhtml="http://www.w3.org/1999/xhtml">
      <xhtml:head>
        <xhtml:title>Resume of John Doe</xhtml:title>
      </xhtml:head>
      <xhtml:body>
        <xhtml:h1>John Doe</xhtml:h1>
        <xhtml:p>John's a great guy, you know?</xhtml:p>
      </xhtml:body>
    </xhtml:html>
  </pers:resume>
</pers:person>
```

# Deklarisanje prostora imena

- primer 2 - koristi se default namespace

```
<?xml version="1.0" encoding="utf-8"?>
<person xmlns="http://www.ftn.ns.ac.yu/dtds/person.dtd">
  <name>
    <title>Sir</title>
    <first>John</first>
    <last>Doe</last>
  </name>
  <position>VP of Marketing</position>
  <resume>
    <xhtml:html xmlns:xhtml="http://www.w3.org/1999/xhtml">
      <xhtml:head>
        <xhtml:title>Resume of John Doe</xhtml:title>
      </xhtml:head>
      <xhtml:body>
        <xhtml:h1>John Doe</xhtml:h1>
        <xhtml:p>John's a great guy, you know?</xhtml:p>
      </xhtml:body>
    </xhtml:html>
  </resume>
</person>
```



# Deklarisanje prostora imena

- primer 3 - redefiniše se default namespace

```
<?xml version="1.0" encoding="utf-8"?>
<person xmlns="http://www.ftn.ns.ac.yu/dtds/person.dtd">
  <name>
    <title>Sir</title>
    <first>John</first>
    <last>Doe</last>
  </name>
  <position>VP of Marketing</position>
  <resume>
    <html xmlns="http://www.w3.org/1999/xhtml">
      <head>
        <title>Resume of John Doe</title>
      </head>
      <body>
        <h1>John Doe</h1>
        <p>John's a great guy, you know?</p>
      </body>
    </html>
  </resume>
</person>
```

# Deklarisanje prostora imena

---

- isključenje default namespace-a

```
<p xmlns="http://www.w3.org/1999/xhtml">  
  I've worked with <name xmlns="">Jane Doe</name>  
  for over a <em>year</em> now.  
</p>
```

# Deklarisanje prostora imena

---

- koji način je bolji?
  1. uvek koristiti prefiks
  2. koristiti jedan namespace kao default, a ostale navoditi sa prefiksom
  3. svuda redefinisati default namespace
- za XML aplikacije je svejedno
- možda nam je bitno da znamo i koji je prefiks korišćen
  - tada ćemo koristiti varijantu 1

# Identifikacija atributa u prostoru imena

---

- primer  
`<pers:name id="i25">`
- da li atribut nasleđuje namespace od svog elementa?
  - u principu ne, ali
  - može, ako aplikacija to tako interpretira (i ako namespace nije eksplicitno naveden)
- primer: formalno, sledeća dva atributa nisu ista  
`<pers:name id="i25">`  
`<pers:name pers:id="i25">`
- ali mogu biti ako aplikacija to tako interpretira

# Namespaces & DTDs

---

- DTD je formalno definisan pre XML Namespaces standarda
- DTD ne podržava potpuno namespaces
- elementi u DTD-u se moraju definisati zajedno sa prefiksom
  - prefiks se mora zadati unapred i mora biti fiksni
- namespace deklaracije se u DTD-u tretiraju kao atributi
  - nije svejedno gde deklarišemo namespace