

# Imenski prostori, poti in kazalci v XML

Iztok Savnik

# Imenski prostori v XML

- XML dokument lahko uporablja attribute, elemente in definicije, ki se nahajajo v drugih datotekah
  - Modularna zasnova
  - Ne sme priti do „trkov“ imen
  - Imena se morajo razlikovati
- Imenski prostori
  - Datoteke, kjer so definirana imena (simboli)
  - Namesto, da definiramo svoja imena uporabimo obstoječa
  - Uporaba standardiziranih imenskih prostorov omogoča razumljivost dokumenta

# Imenski prostori v XML (1)

- Imenski prostor v XML definiramo z uporabo vnaprej definiranih atributov
  - xmlns: in xmlns
  - Predpono imenskega prostora definiramo z xmlns:
  - Privzet imenski prostor definiramo z xmlns
- Uporabniki ne smejo uporabljati xmlns nikjer drugje

# Imenski prostori v XML (2)

- Primer definicije imenskega prostora

```
<x xmlns:edi='http://ecommerce.example.org/schema'>  
  <!-- the 'taxClass' attribute's namespace is  
        http://ecommerce.example.org/schema -->  
  <lineItem edi:taxClass="exempt">Baby food</lineItem>  
</x>
```

# Imenski prostori v XML (3)

- Primer uporabe xhtml v okviru XML

```
<?xml version="1.0"?>
<html:html xmlns:html='http://www.w3.org/1999/xhtml'>
  <html:head><html:title>Frobnostication</html:title></html:head>
  <html:body><html:p>Moved to
    <html:a href='http://frob.example.com'>here.</html:a>
  </html:p>
</html:body>
</html:html>
```

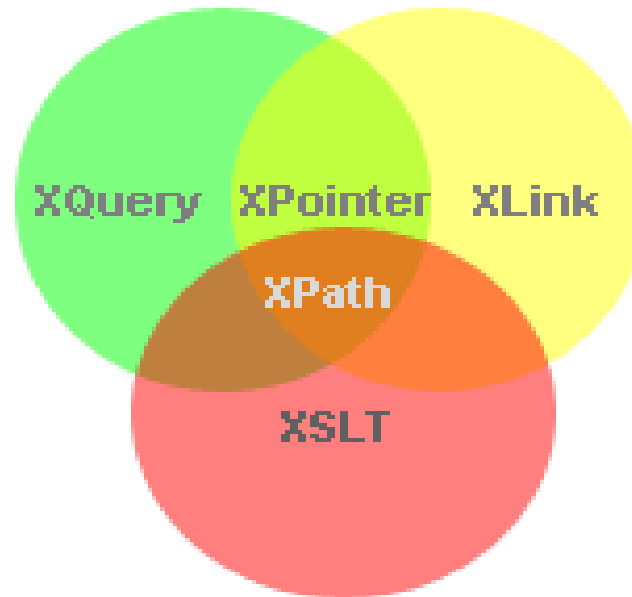
# Imenski prostori v XML (4)

- Primer definicije privzetega imenskega prostora

```
<?xml version="1.0"?>
<!-- elements are in the HTML namespace, in this case by default -->
<html xmlns='http://www.w3.org/1999/xhtml'>
  <head><title>Frobnostication</title></head>
  <body><p>Moved to
    <a href='http://frob.example.com'>here</a>.</p></body>
</html>
```

# X{\*} - struktura

- ★ XPath
- ★ XPointer
- ★ XLink
- ★ XQuery
- ★ XSLT



# X{\*} - struktura (1)

- XLink
  - XLink je okrajšava za XML Linking Language
  - Hiper-povezave med XML dokumenti
  - Katerikoli element XML se lahko obnaša kot povezava
  - Podpira enostavne (html) in kompleksne (razmerja) povezave
  - Povezave so lahko definirane izven povezanih datotek
- XLink je W3C Recommendation



# X{\*} - struktura (2)

- XPointer
  - Okrajšava za XML Pointer Language
  - Kazalci lahko identificirajo specifične dele dokumenta
  - Uporablja XPath izraze za navigacijo po XML dokumentu
- XPointer je W3C Recommendation

# Poti v XML dokumentih

- Identifikacija delov XML dokumenta
- Xpath
  - Omogoča delo z osnovnimi tipi: niz, števila in boolove vrednosti
  - Kompaktna, ne-XML sintaksa jezika
  - Xpath model: drevo vozlišč
  - Različne vrste vozlišč: elementi, atributi in tekst

# Xpath izrazi

- Osnovni sintaktični element Xpath je *izraz*
- Rezultat evaluacije izraza je objekt, ki je enega od naslednjih tipov:
  - Množica vozlišč
  - Boolova vrednost
  - Število
  - Niz znakov

# Xpath izrazi (1)

- Izraz se ovrednoti glede na dani *kontekst*
  - Kontekst definirata XSLT ali Xpointer
- Kontekst sestavljajo:
  - Kontekstno vozlišče
  - Par pozicija konteksta/velikost konteksta
  - Množica spremenljivk povezanih z vrednostmi
  - Knjižnica funkcij
  - Množica definicij imenskih prostorov

# Xpath izrazi (2)

- Knjižnica funkcij
  - Argumenti in rezultati imajo enega od osnovnih tipov
  - XSLT in Xpointer še razširijo funkcije
- Predikati
  - Spreminjanje konteksta: kontekstno vozlišče, pozicija in velikost
- Lokacijska pot
  - Del Xpath izraza – najbolj pomemben del Xpath
  - Z lokacijsko potjo izberemo objekte
  - Imamo okrajšano in polno sintakso

# Lokacijska pot

- Primer:

```
/child::movies/child::movie[@rating='R']
```

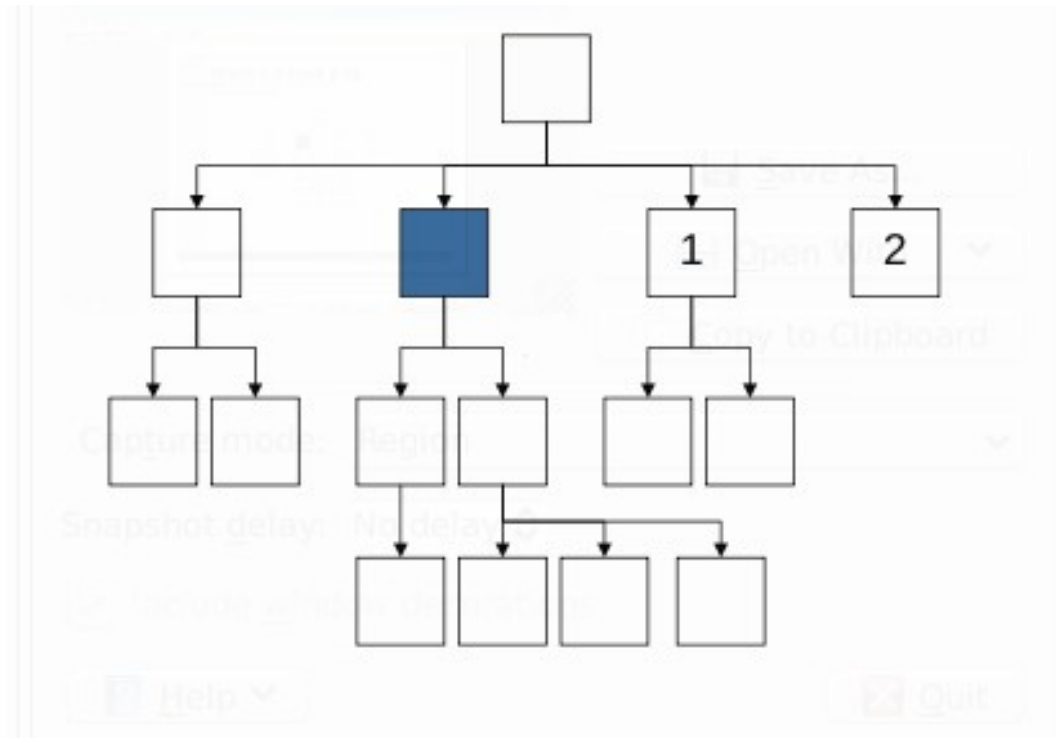
- Korak --- /child::movies
- Os --- child::
- Test vozlišča --- movie
- Predikat --- [@rating='R']

# Lokacijska pot (1)

- Osi:
  - ancestor, ancestor-or-self
  - attribute
  - child
  - descendant, descendant-or-self
  - following, following-sibling
  - namespace
  - parent
  - preceding, preceding-sibling
  - self

# Lokacijska pot (2)

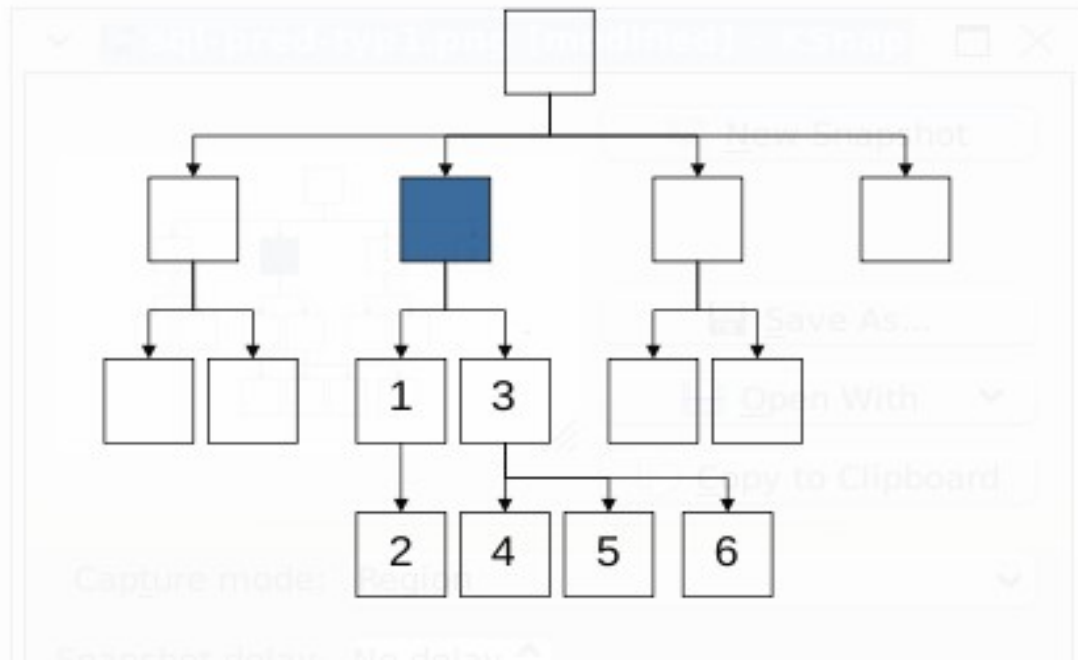
following-sibling





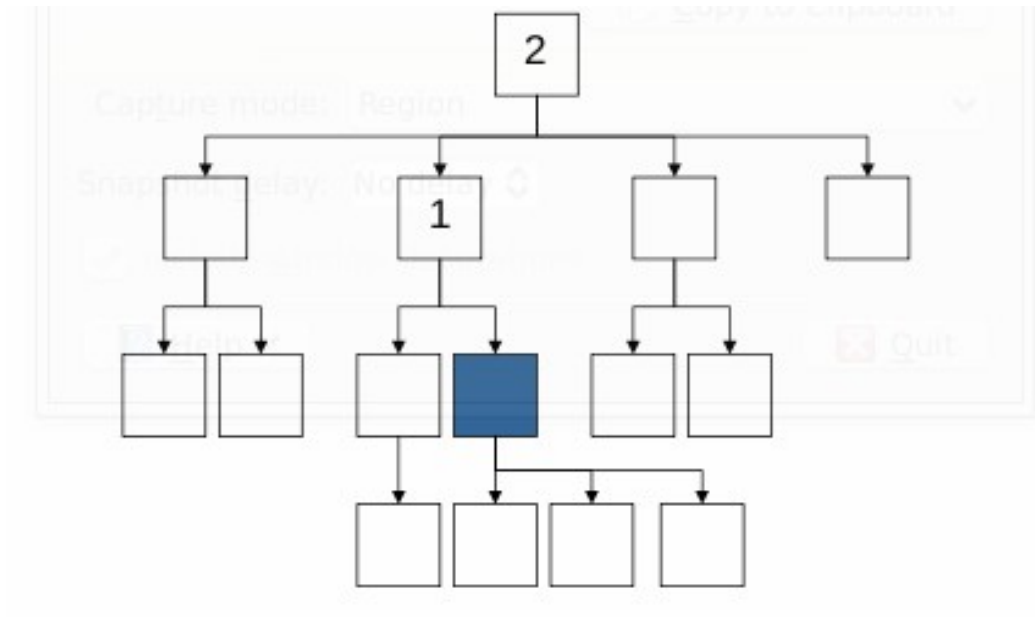
# Lokacijska pot (3)

descendant



# Lokacijska pot (4)

ancestor



# Lokacijska pot (5)

- Test vozlišča
  - foo
  - foo:bar
  - foo:\*
  - \*
  - node()
  - comment()
  - text()
  - processing-instruction()

# Podatkovni tipi in funkcije

- boolean
- number
- string
- node-set
- external object

```
true(), false()  
=, !=, <, >, <=, >=  
and, or  
not()
```

```
=, !=, <, >, <=, >=  
+, -, *, div, mod, -  
floor(), ceiling()  
...
```

```
count(), current(),  
last(), name(),  
position(), sum()  
|  
...
```

```
"foo", 'foo'  
concat(), contains(), starts-with(),  
String-length(), substring(),  
substring-after(),  
substring-before(), translate()  
...
```

# Xpath: primeri

- `child::para` izbere elemente z imenom `para`, ki so otroci kontekstnega vozlišča (KV)
- `child::*` izbere vse otroke kontekstnega vozlišča
- `child::text()` izbere vsa tekstovna vozlišča, ki so otroci KV
- `child::node()` izbere vse otroke KV neglede na tip
- `attribute::name` izbere imena atributov KV
- `attribute::*` izbere vse attribute KV
- `descendant::para` izbere vsa vozlišča, ki so nasledniki KV
- `ancestor::div` izbere vse predhodnike KV z imenom `div`
- `ancestor-or-self::div` izbere vse predhodnike KV skupaj z KV, ki imajo ime `div`
- `descendant-or-self::para` izbere naslednike KV vključno z KV, ki imajo ime `para`

# Xpath: primeri (1)

- `/descendant::olist/child::item` izbere elemente item, ki imajo starša olist, ki je pod KV
- `child::para[position()=1]` izbere prvega otroka KV z imenom para
- `child::para[position()=last()]` izbere zadnjega otroka KV za imenom para
- `child::para[position()>1]` izbere vse otroke KV z imenom para razen prvega
- `following-sibling::chapter[position()=1]` izbere vse naslednike (brate/sestre) KV chapter
- `preceding-sibling::chapter[position()=1]` izbere prvega predhodnika (brat/sestra) KV
- `/descendant::figure[position()=42]` izbere 42. element figure med vsemi naslediki KV

# Xpath: primeri (2)

- `/child::doc/child::chapter[position()=5]/child::section[position()=2]` izbere 2. sekcijo iz 5. poglavja elementa doc
- `child::para[attribute::type="warning"]` izbere otroke para KV, ki imao tip atributa warning
- `child::para[attribute::type='warning'][position()=5]` izbere 5. otroka KV z imenom para, ki imaa tip atributa warning
- `child::para[position()=5][attribute::type="warning"]` izbere 5. otroka KV, če ima vozlišče atribut tipa wrning
- `child::chapter[child::title='Introduction']` izbere poglavje z imenom chapter, ki ima naslov 'Introduction'
- `child::chapter[child::title]` izbere otroke z imenom chapter, ki imajo vsaj en naslov
- `child::*[self::chapter or self::appendix]` izbere otroke KV, ki bodisi vozlišča chapter ali appendix

# Okrajšave v Xpath

- `child::` ≡ `"`
- `attribute::` ≡ `@`
- `/descendant-or-self::node()/` ≡ `//`
- `self::node()` ≡ `.`
- `parent::node()` ≡ `..`



# Xpath: primeri z okrajšavami

- para izbere elemente para, ki so otroci KV
- \* izbere vse otroke (elemente) KV
- text() izbere vse otroke KV, ki so tekstovna vozlišča
- @name izbere imena atributov KV
- @\* izbere vse attribute KV
- para[1] izbere 1. otroka KV z imenom para
- para[last()] izbere zadnjega otroka KV, ki ima ime para
- \*/para izbere vsa podrejena vozlišča KV z imenom para
- /doc/chapter[5]/section[2] izbere 2. sekcijo 5. poglavja dokumenta
- chapter//para izbere elemente para, podrejene chapter
- //para izbere vse podrejene elemente para korena dokumenta

# Xpath: primeri z okrajšavami (1)

- `//olist/item` izbere vse elemente `item`, ki imajo starša `olist`
- `.` izbere KV; `..` izbere starša KV
- `./para` izbere vsa naslednike KV z imenom `para`
- `../@lang` izbere atribut `lang` starša KV
- `para[@type="warning"]` izbere otroke KV z imenom `para`, ki imajo vsaj en atribut `type` z vrednostjo „warning“
- `para[5][@type="warning"]` izbere 5. otroka KV z imenom `para`, ki ima atribut `type` z vrednostjo „warning“
- `chapter[title="Introduction"]` izbere otroke KV z imenom `chapter`, ki imajo naslov „Introduction“
- `chapter[title]` izbere otroke KV z imenom `chapter`, ki imajo naslov
- `employee[@secretary and @assistant]` izbere otroke KV z imenom `employee`, ki imajo tajnico in pomočnika

# Literatura

- <http://www.w3.org/TR/xpath>
- David J. Malan, XML with Java, Java Servlet, and JSP, Course CS E-259, Harvard University, 2007
- <http://www.w3.org/TR/xml-names/>