

WPUB - XSLT

4. cvičenie

18.3. 2021

Eduard Kuric

XSLT

- W3C štandard
- **XML Stylesheet Language for Transformations**
- používa sa hlavne na transformáciu značkovacích jazykov (napríklad XML) do inej formy, napríklad HTML, JSON alebo dokonca iného dokumentu XML
- pôvodne
 - transformácia XML dokumentov za účelom ich vizualizácie
- v súčasnosti
 - jazyk takmer s rovnakými možnosťami ako XQuery (XML query language)

XPath vs XQuery

- XPath
 - Lokalizácia konkrétnych elementov v XML-based (hierarchickej) štruktúre
- XQuery
 - Využíva XPath - môžeme realizovať XPath dopyty, je komplikovanejší, poskytuje viac možností – funkcionálny programovací jazyk, napr. umožňuje definovať a volať vlastné funkcie, podmienky, cykly.
- XSLT vs XQuery
 - document-driven vs. program-driven
 - XSLT je XML dokumentom, XQuery nie je

I/O

- Vstup: jeden a viac XML dokumentov
- Výstup: jeden a viac dokumentov (nie iba XML)

I/O

```
<?xml version="1.0"?>
<order id="01" date="03/18/2021">
  <customer id="C1">
    <name>Peter Malý</name>
    <email>peter.maly@xyz.com</email>
  </customer>
  <items>
    <item id="I1">
      <name>Pekáreň v Brooklyne</name>
      <amount>1</amount>
      <price>15</price>
    </item>
    <item>
      ...
    </item>
  </items>
</order>
```

I/O

```
<?xml version="1.0"?>
<html>
<head><title>Order no. 1 - Peter Malý</title></head>
<body>
  <table>
    <tr>
      <td>Pekáreň v Brooklyne</td>
      <td>15 EUR</td><td>1 item</td>
    </tr>
  </table>
  <div>Total price: 15 EUR</div>
</body>
</html>
```

I/O

- **Základom je XSLT skript**
 - Musí spĺňať pravidlá XML súboru
- Skript obsahuje šablóny - **templates**
- Šablóna sa aplikuje na element vstupného XML dokumentu a výstupom je konkrétna transformácia
 - Môže spustiť aplikáciu iných šablón na rovnakom elemente alebo iných elementov
 - Môže načítať údaje zo vstupného dokumentu alebo iných dokumentov

XSLT skript

- XML dokument, koreňovým elementom je `stylesheet`

```
<?xml version="1.0" encoding="UTF-8"?>  
<stylesheet>  
...  
</stylesheet>
```


Menný priestor - namespace

- v XML sú súčasťou špecifikácie
- vyriešiť situácie, keď sa v 1 dokumente vyskytnú elementy/atribúty, ktoré majú rovnaký názov, ale rôznu sémantiku
- Menný priestor je identifikovaný URI
 - Spravidla URL , smerujúca na dokumentáciu, príp. XSD, ale nie nutne
- Skrátený tvar: prefix + URI

Menný priestor – namespace /2

```
<?xml version="1.0" encoding="UTF-8"?>  
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"  
  xmlns="http://www.w3.org/1999/xhtml"  
  version="1.0">  
</xsl:stylesheet>
```

Výstup

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns="http://www.w3.org/1999/xhtml"
version="1.0">
  <xsl:output method="xml" indent="yes" />
</xsl:stylesheet>
```

- xml, pdf, text, html, ...
- indent = odsadenie výstupu

XSLT template

```
<?xml version="1.0" encoding="windows-1250"?>
<xsl:stylesheet ... >
  <xsl:template> ... </xsl:template>
  <xsl:template> ... </xsl:template>
  ...
</xsl:stylesheet>
```

- potomok elementu `stylesheet`
- opisuje jednu šablónu
- skript zvyčajne pozostáva z viacerých šablón

XSLT template /2

- Vstupom je XML element, ktorý vie byť vybratý/získaný prostredníctvom dopytu Xpath
 - element, atribút, text
- Výstupom je napr. XML fragment, alebo ľubovoľný text (HTML, JSON, CSV, PDF, ...)

Nepomenovaná šablóna

- Template s atribútom match

```
<xsl:template match="[xpath path ['|' xpath  
path] *]">
```

```
...
```

```
</xsl:template>
```

Pomenovaná šablóna

- Template s atribútom name

```
<xsl:template name="[template name]">
```

```
...
```

```
</xsl:template>
```

XSLT skript

- Je vykonaný programom tzv. XSLT procesorom
 - Saxon, XSLTProcessor
 - Súčasťou prehliadačov

Vývojové platformy

- PHP runtime – na cvičení
- Java
 - <https://roytuts.com/xml-to-html-using-xslt/>

PHP skript

```
$xml = new \DOMDocument();  
$xml->load('xml/data.xml');
```

```
$xsl = new \DOMDocument();  
$xsl->load('xml/hello.xsl');
```

```
$xsltProc = new \XSLTProcessor();  
$xsltProc->importStyleSheet($xsl);  
return $xsltProc->transformToXML($xml);
```