

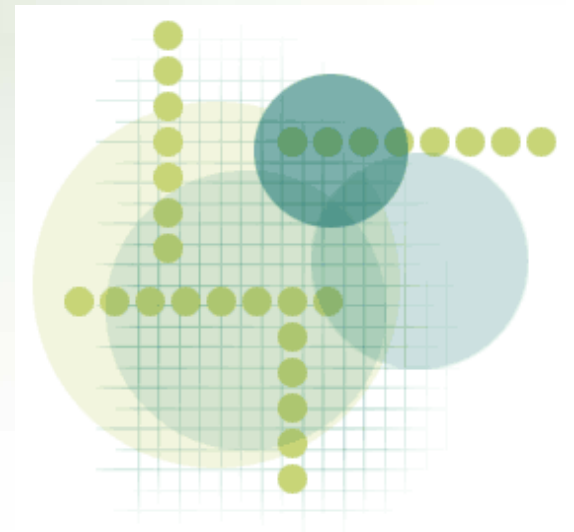
# DocBook v praxi

**Bc. Marián Šimko**



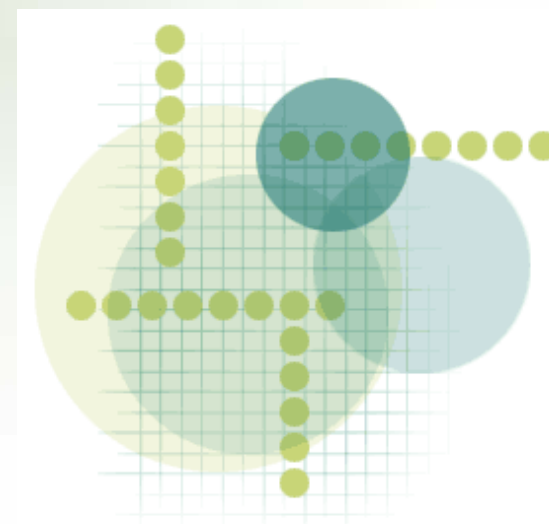
# Obsah

- čo je DocBook
- tvorba DocBook dokumentu
- formátovanie / transformácie
- generovanie výstupu
- Apache Ant
- budúcnosť DocBook-u



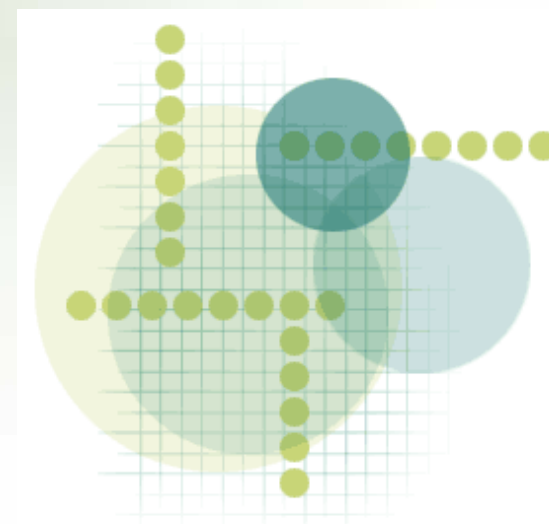
# Čo je DocBook

- štandard pod záštitou OASIS
- DTD, XSD, **RELAX NG**
- DocBook Project (<http://docbook.sourceforge.net>)
  - šablóny
    - HTML
    - PDF
    - JavaHelp
    - RTF
  - nástroje

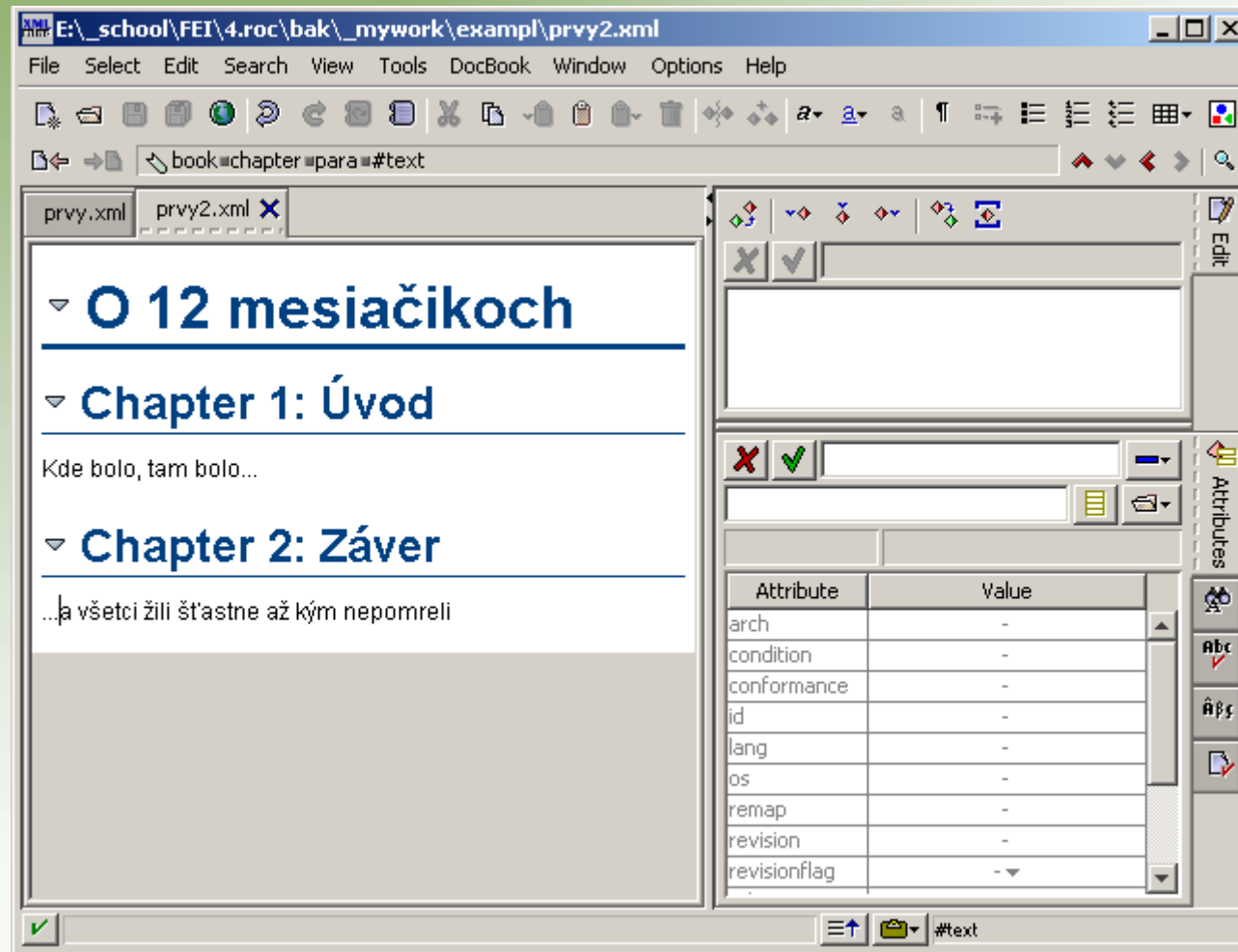


# Tvorba DocBook dokumentu

- podpora tvorby
  - XML editory
    - „WYSIWYG” - XMLMind's XML Editor, Morphon XML Editor
    - iné - Altova XML Spy, <oXygen/> XML Editor
  - OpenOffice
  - Emacs
- validácia
  - MSV (Sun Multi-Schema Validator)



# XMLMind's XML Editor



The screenshot displays the oXygen XML Editor interface. The main window shows the XML document being edited, with line numbers on the left. The document content includes a title, a citation, and a listitem. The left sidebar shows a project tree and an outline of the document structure. The right sidebar contains panels for Attributes, XSLT input, and Schema Components. A status bar at the bottom shows the current file path and document information.

Project: thesis.xml\*

Attributes

Attribute	Value
condition	
conformance	
dir	
id	
lang	
os	
override	
remap	
revision	
revisionflag	
role	
security	
userlevel	
vendor	

XSLT input

No XSL editor is currently selected.

Schema Components

I:\doc\docbook\thesis.xml U+000A 254:17 Modified

# Formátovanie / Transformácie

- použitie šablón
  - DocBook Project
    - modularita
    - parametrizovateľnosť
  - vlastné šablóny
    - tzv. „customization layer”
- procesory
  - XSLT
  - XSL-FO



# Procesory

- XSLT

- „volné”

- Saxon
    - Xalan
    - xsltproc

- komerčné

- MSXML
    - System.Xml.Xsl

- XSL-FO

- „volné”

- FOP
    - Passive TeX
    - xmlroff

- komerčné

- E3
    - XEP
    - XML2PDF





# „Customization Layer“

- vlastné XSL šablóny
  - integrácia do projektu DocBook Project
    - `<xsl:include>`
    - `<xsl:import>`
  - „prepisovanie“ predlôh
    - atribúty *match*, *priority* (`<xsl:template>`)
  - zmena parametrov transformácie
    - `<xsl:param>`
- jednoduchá modifikácia transformačných vlastností bez nutnosti tvorby rozsiahlych šablón



# „Customization Layer“ - príklad

```

<?xml version='1.0'?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
  xmlns:fo="http://www.w3.org/1999/XSL/Format"
  version='1.0'>

  <xsl:import href="../docbook-xsl-1.71.1/fo/docbook.xsl"/>
  <xsl:include href="../docbook-xsl-bibliography/fo/biblio.xsl"/>
  <xsl:include href="../docbook-xsl-thesis/common/thesis.common.xsl" />

  <!-- veľkosť papiera -->
  <xsl:param name="paper.type" select="'A4'"/>

  <!-- okraje -->
  <xsl:param name="page.margin.inner" select="'3cm'"/>
  <xsl:param name="page.margin.outer" select="'2.5cm'"/>
  ....

  <!-- poznámka pod čiarou -->
  <xsl:template match="footnote">
    <fo:footnote>
      <fo:inline font-size="75%" baseline-shift="super"><xsl:number level="any" count="footnote"/></fo:inline>
      <fo:footnote-body font-family="serif,Symbol,ZapfDingbats"
        font-size="8pt" font-weight="normal" font-style="normal"
        text-align="justify" margin-left="0pc">

        <fo:block>
          <fo:inline font-size="75%" baseline-shift="super"><xsl:number level="any" count="footnote"/></fo:inline>
          <xsl:value-of select="para"/>
        </fo:block>
      </fo:footnote-body>
    </fo:footnote>
  </xsl:template>
  ...

</xsl:stylesheet>

```

# Generovanie výstupu

- potreba automatizácie
  - zostavovacie nástroje
    - Apache Ant
    - Make
  - použitie v programovom kóde
  - XSL servery
- možnosť parametrizácie projektu



# Apache Ant

- vychádza z XML
- koreňový element - `<project>`
- projektové ciele - `<target>`
- úlohy (angl. task)
  - široké spektrum zabudovaných
  - možnosť definovania vlastných
- explicitné definovanie parametrov



# Apache Ant - příklad

```
<project name="DocBookT" default="pdf" basedir=".">
  <description>transformacia DocBook XML dokumentu do formatu PDF</description>
  <property file="build.prop"/>
  <taskdef name="fop" classname="org.apache.fop.tools.anttasks.Fop">
    <classpath>
      <pathelement location="${fop.path}\build\fop.jar"/>
    </classpath>
  </taskdef>

  <target name="init">
    ...
  </target>

  <target name="validate" depends="init">
    <java jar="${relames.path}/relames.jar" fork="true">
      <jvmarg value="-Xss512K"/>
      <arg line="${docbook.path}\rng\docbook.rng ${dir.src}\${file.src}"/>
    </java>
  </target>

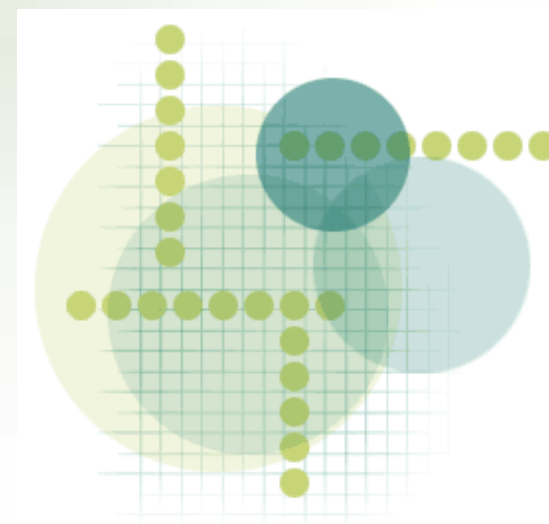
  <target name="html" depends="init,validate"
    description="urob html">
    <exec executable="xsltproc" searchpath="true" vmlauncher="false">
      <env key="Path" path="${env.Path};${xsltproc.path}"/>
      <arg line="-output ${dir.dst}\${TODAY.NOW}.html ${dir.src}\biblio.html.xsl ${dir.src}\${file.src}"/>
    </exec>
  </target>

  <target name="fo" depends="init,validate"
    description="urob fo">
    <exec executable="xsltproc" searchpath="true" vmlauncher="false">
      <env key="Path" path="${env.Path};${xsltproc.path}"/>
      <arg line="-output ${dir.tmp}\temp.fo ${dir.src}\biblio.fo.xsl ${dir.src}\${file.src}"/>
    </exec>
  </target>

  <target name="pdf" depends="fo"
    description="urob pdf (fo -> pdf)" >
    <fop format="application/pdf" userconfig="${dir.conf}\fop\config.xml"
      fofile="${dir.tmp}\temp.fo" outfile="${dir.dst}\${TODAY.NOW}.pdf" />
  </target>
</project>
```

# DocBook – klady a zápory

- platformovo nezávislý formát
- jednoduché možnosti automatizácie rutínnej práce
- generovanie širokého spektra formátov
- jednoduchý dokument-processing
- komplikované rozširovanie použiteľných elementov
- nevhodný pre malé projekty



# Quo vadis DocBook?

(DocBook vs. ODF vs. OOXML)

- DocBook
    - „silná“ sémantická vrstva
  - ODF (+ OOXML)
    - dôraz na zobrazenie zdrojového textu
    - mediálna podpora
    - WYSIWYG priorita
- problémy s konverziou ODF do DocBook
- hlasy po zmene...



Ďakujem za pozornosť!

[ariquis@gmail.com](mailto:ariquis@gmail.com)

