

**Dokumentace k programům**  
**Documentation for programs**

**Bakalářská práce**  
**David Toufar**  
**Vedoucí bakalářské práce: Ing. Václav Novák, CSc.**  
**Jihočeská univerzita v Českých Budějovicích**  
**Pedagogická fakulta**  
**Katedra informatiky**  
**2010**

## **Prohlášení**

Prohlašuji, že svoji bakalářskou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury. Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě pedagogickou fakultou elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách.

V Českých Budějovicích dne 22.4.2010 .....

David Toufar

## **Anotace**

Dokumentace k programům má za základní ideu pomáhat uživatelům a vysvětlovat jak programy, pro které se dokumentace vytvořila, pracují. Zvláštní důraz budu klást na potřeby vycházející z Visual Studia 2005 a XML nástrojů. V práci chci ukázat vytvoření XML dokumentace z VS 2005. Zaměřím se na možnosti při psaní dokumentace v DocBook. V této práci budu chtít ukázat funkce důležité při tvorbě dokumentace k programům a vybraným editorům tyto funkce ohodnotím.

## **Abstract**

The main aim of programs documentation is helping the users and explaining how the programs work, for which the documentation this was created. I will in particular focus on the needs that are based on Visual Studio 2005 and XML tools. In this thesis I want to show the creating of XML documents from Visual Studio 2005. I will also handle the possibilities of writing documentation in DocBook. In this thesis I further want to show functions that are important for creating documentation of programs and selected editors and I will evaluate these functions.

## **Poděkování**

Děkuji za odborné vedení panu Ing. Václavu Novákovi, CSc., dále pak hlavně své rodině za podporu.

## Obsah

<b>1</b>	<b>ÚVOD A CÍL PRÁCE</b> .....	<b>7</b>
1.1	ÚVOD.....	7
1.2	CÍL PRÁCE .....	7
<b>2</b>	<b>ANALÝZA PROBLÉMU</b> .....	<b>9</b>
2.1	METODIKA .....	9
2.2	VÝCHOZÍ STAV .....	12
<b>3</b>	<b>FUNKCE EDITORŮ</b> .....	<b>13</b>
3.1	ZÁKLADNÍ FUNKCE .....	13
3.2	PODPORA JAZYKOVÝCH SCHÉMÁT .....	30
3.3	PODPORA KONKRÉTNÍCH SCHÉMÁT .....	32
3.4	PŘEDNASTAVENÉ VÝSTUPNÍ FORMÁTY .....	33
<b>4</b>	<b>VISUAL STUDIO A XML</b> .....	<b>35</b>
4.1	PSÁNÍ KOMENTÁŘŮ.....	35
4.2	GENEROVÁNÍ DOKUMENTACE .....	36
<b>5</b>	<b>XML EDITORY</b> .....	<b>38</b>
5.1	TYPY XML EDITORŮ .....	38
5.1.1	Textové editory.....	38
5.1.2	Grafické editory.....	39
<b>6</b>	<b>TVORBA XML DOKUMENTACE</b> .....	<b>40</b>
6.1	SEKCE .....	41
6.2	METAINFORMACE.....	41
6.3	BLOKOVÉ ELEMENTY .....	42
6.4	INLINE ELEMENTY .....	44
6.5	TVORBA DOKUMENTU.....	48

<b>7</b>	<b>POROVNÁNÍ EDITORŮ .....</b>	<b>50</b>
7.1	EDITX XMLEDITOR .....	50
7.2	EPCEDIT .....	55
7.3	XMLMIND .....	61
7.4	<OXYGEN/> XML EDITOR.....	67
<b>8</b>	<b>ZHODNOCENÍ VÝSLEDKŮ A ZÁVĚR .....</b>	<b>74</b>
<b>9</b>	<b>REFERENCE .....</b>	<b>67</b>

## **1 Úvod a cíl práce**

### ***1.1 Úvod***

V dnešní době si softwarové programy bez dokumentace nedokáže představit jak uživatel, tak ani jakýkoli programátor, který s programem bude pracovat. Dokumentaci k programům můžeme mít v tištěné podobě, v podobě internetových stránek, dále dokumentace může být v PDF dokumentu, RTF dokumentu, Java Helpu či HTML help apodobně. Dokumentaci k programům můžeme psát samostatně pro každý formát, což je časově náročné při aktualizaci a tak i nevýhodné. Kdybychom psali dokumentaci pro tištěný výstup, a HTML Help musíme vše psát dvakrát. Nyní si představte, že ale píšete dokumentaci pro všechny možné výstupní formáty. Proto je v dnešní době, kdy je času nedostatek, výhodné psát dokumentaci ve formátu XML, kde můžete mít výsledný dokument ve více podobách. Pro psaní dokumentace pomocí XML existuje v dnešní době velice mnoho editorů. Každý z těchto editorů nabízí jiné výstupní formáty (PDF, HTML, XHTML, apod.). Editory nám dokážou zobrazit dokumenty v různých podobách (textové podobě, grafické) a mají různé funkce.

### ***1.2 Cíl práce***

V současném počítačovém světě existuje velké množství XML editorů na tvorbu dokumentace, které nemusejí být veřejnosti známé. Cílem této práce je ukázat jak nastavit Visual Studio 2005, aby se

## *Úvod a cíl práce*

vytvořil XML dokument a popsat možnosti Visual Studia 2005 a co vše můžeme pomocí přednastavených elementů zdokumentovat.

Popíši tvorbu dokumentu (knihy) v DocBooku ve verzi 5. Dále vysvětlím a otestuji na vybraných editorech důležité funkce, podporu schémat a přednastavené výstupní formáty dokumentu. To vše vyhodnotím a doporučím vhodné editory pro začátečníky, pokročilé a profesionály.



## **2 Analýza problému**

### **2.1 Metodika**

Ve výběru XML editorů pro psaní dokumentace k programům, které budu následně popisovat, jsem zvolil kritéria:

- Nejznámější funkce editorů při tvorbě dokumentace, vizualizace dokumentu, přednastavené výstupy XML editorů při tvorbě dokumentace.

Mým motivem vybrat si tato kritéria, byla dlouholetá zkušenost práce s editory. Často jsem používal editory, ať už grafické nebo textové, ve kterých se pracovalo velice dobře, ale chvílemi mi chyběly určité funkce editoru. Kvůli tomu jsem byl občas nucen vyhledat a využít schopnost jiného editoru, který chybějící funkci nahradil. Proto jsem se rozhodl otestovat funkce XML editorů pro tvorbu dokumentace.

Vybral jsem tyto nejběžnější funkce XML editorů na tvorbu dokumentace:

Zvýraznění syntaxe, validace XML souborů, průběžná validace XML souborů, nabídka elementů a atributů, vkládání a vyjímání XML struktury, vytvoření elementu výběrem textu, změna elementu, funkce pro editaci textu, vyhledávání a nahrazování (XPath), kontrola pravopisu v dokumentu (čeština), vícejazyční text (Unicode), MathML, tvorba a

## *Analýza problému*

upravování schémat, vestavěný XSLT procesor, vestavěný XSL-FO procesor, tvorba a upravování XSLT a XSL-FO stylů, práce v kódu, práce v tabulkovém zobrazení, práce ve stromovém zobrazení, práce v grafickém zobrazení, práce s velkými dokumenty (testován bude 10MB soubor), vzdálená editace (FTP), práce s více soubory. Podpora jazyků pro popis schémat (zde zjistím podporu: DTD, W3C Schema, Schematron, RELAX NG). Podpora pro konkrétní schémata (zde jsem se rozhodl otestovat: DocBook, DITA, TEI). Přednastavené výstupní formáty (Vybral jsem základní výstupy, které editory nabízí: PDF, HTML, XHTML, HTML Help, PostScript).

Dále jsem sestavil dle níže uvedených kritérií seznam softwaru na tvorbu dokumentace. Z těchto internetových stránek (dne 2.2.2010):

- <http://www.xml.com/>
- <http://www.kosek.cz/>

Kritéria při výběru editorů:

- Editor bude pracovat s XML
- Bude stavěn minimálně pro operační systém Windows
- Editory budou freeware nebo shareware
- Editory nebudou stejného typu

Po prolistování webových stránek jsem vybral 4 typy editorů. Textový editor, částečný WYSIWYG editor, WYSIWYM editor,

## *Analýza problému*

WYSIWYG editor. Níže prezentuji typy editorů a k nim přiřazené editory na tvorbu dokumentace.

- EditX XML Editor 2010 je označován jako textový editor
- epcEdit je částečný WYSIWYG editor
- XMLMind XML Editor je WYSIWYM editor
- <oxygen/> XML Editor je plně WYSIWYG editor

Tyto programy se pokusím co nejlépe popsat a to podle předloh:

- Instalace (náročnost instalace, instalace zásuvných modulů)
- O programu (licence, nejnovější verze, jazyková podpora, platformy)
- Mého vlastního hodnocení tohoto editoru (přehlednost, ovladatelnost, práci s funkcemi)

Editory budu dále hodnotit a testovat pro operační systém:

- Windows XP professional Service Pack 3

Při testování budu hodnotit funkce, podpora pro různé jazyky schémat, podporu pro konkrétní schémata a výstupní formáty, které odzkouším a ohodnotím. Hodnotit budu 1 bodem funkci, jazyk pro podporu schémat, schéma a výstupní formát editorů, které onu funkci, jazyk pro popis schématu, schéma a výstup budou podporovat. Naopak pokud editor funkci, jazyk pro popis schéma, schéma a výstupní formát podporovat nebude, dostane 0 bodů.

## **2.2 *Výchozí stav***

V dnešní době se vyskytuje veliké množství editorů na tvorbu dokumentace pomocí jazyka XML. Značná část různých editorů a informace o jejich možnostech nejsou běžně známy a z toho důvodu často unikají i velice kvalitní editory na tvorbu dokumentace. Obecně je možné říci, že chybí jakékoli srovnání a porovnání editorů z hlediska jejich možností, funkčnosti a tudíž i jejich využitelnosti.

## **3 Funkce editorů**

### **3.1 Základní funkce**

#### **Zvýraznění syntaxe**

Zvýraznění syntaxe je vlastnost některých editorů, která programátorům usnadňuje práci se zdrojovým kódem. Tato funkce editoru zobrazuje různé části zdrojového kódu v různých barvách, fontech nebo stylu. Kód programu se zvýrazněním syntaxe nijak nezmění. Dále je možné zvýrazněním syntaxe odhalit chybu. [1]

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="C:\Program Files\Wattle
Software\XMLwriter 2\Projects\Examples\Library\Library.xslt"?>
<!-- DOCTYPE declaration, specifying an external DTD file,
Library.dtd.-->
<!DOCTYPE library SYSTEM "C:\Program Files\Wattle
Software\XMLwriter 2\Projects\Examples\Library\Library.dtd">
<library>
    <name>Knihovna</name>
    <book status="available">
</library>
```

## **Validace XML dokumentu v editoru**

Validace XML umožňuje odhalit nekonzistenci dat, které později mohou vadit při dalším zpracování. Validace XML dokumentu ověřuje shodu se schématem. Pokud editor tuto funkci nabízí a my ji spustíme, editor nám potvrdí či vyvrátí validnost dokumentu. Definice uvedená přímo ve specifikaci XML 1.0 říká, že „dokument je validní, pokud je mu přiřazena deklarace DTD a pokud splňuje všechna její omezení.“[2] Validita je vztah mezi dokumentem a schématem.[3]



*Obrázek 3.1: Validace dokumentu*

## **Průběžná validace XML dokumentu**

Naopak průběžná validace v editoru dokument validuje pořád. Při této funkci editoru se uživatel nestará o validaci dokumentu. Dokument je validován pokaždé, když uživatel nějak změnil či upravil obsah dokumentu. Při psaní v dokumentu uživatel stále vidí, zda je dokument psán správně podle schématu. Díky automatické validaci editor uživateli zjednodušuje práci s dokumentem, protože už po napsání začátečního a končícího tagu uživatel vidí, zda mohl tyto tagy v místě použít.

## Funkce editorů

```
para>This is a <code>Docbook</code> figure<inlineequation>  
<špatnyelement></špatnyelement>
```

### Dokument není validní

■ Analysis completed.  
2 Problem(s) Found.  
- E [Xerces] Element type "špatnyelement" must be declared.  
- E [Xerces] Unexpected element "špatnyelement". The content of the parent element type must match "(alt?,(graphic+|inlinemediaobject+))".

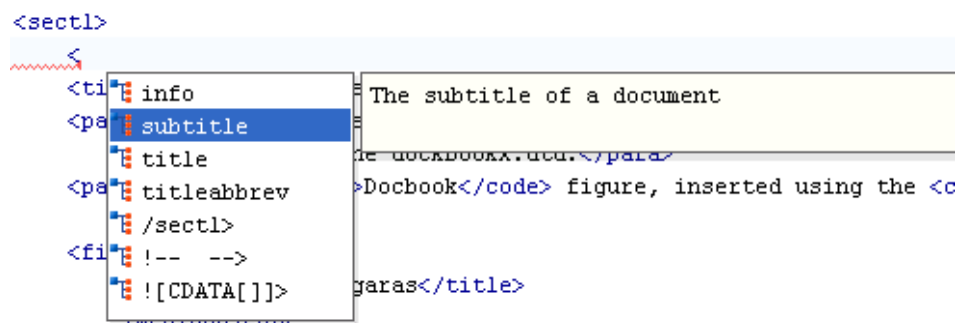
### Dokument je validní



Obrázek 3.1: Automatická validace

## Nabídka elementů a atributů

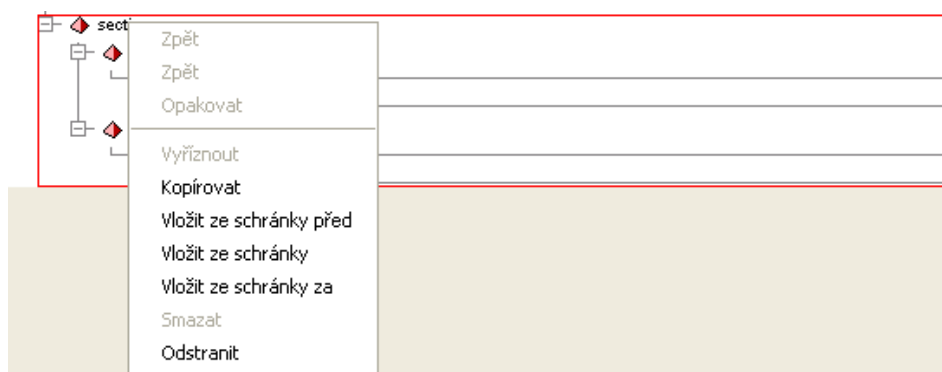
Editory mající tuto funkci, bývají pro uživatele výhodné. Tato funkce nabízí uživateli elementy a atributy, které je možné kdykoli v místě kurzoru do dokumentu vložit. Uživateli to tak zjednodušuje práci při vytváření dokumentu.



Obrázek 3.1: Nabídka elementů a atributů

## **Vkládání a vyjímání XML struktury**

Tato funkce pomáhá uživateli editoru přesouvání, vyjímání a vkládání XML struktury v dokumentu. Uživatel označením určitého tagu vybere vše co je mezi jeho začínajícím a končícím tagem a editor mu umožňuje tuto část jednoduše vyjmout.



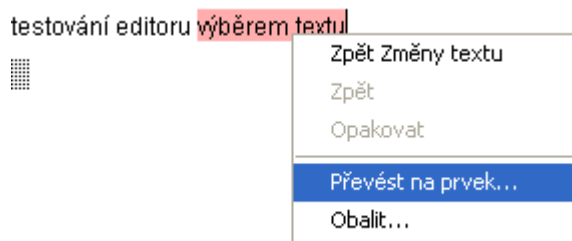
*Obrázek 3.1: Vyjímání XML struktury*

## **Vytvořením elementu výběrem textu**

Tato funkce umožňuje uživateli vložit na kraje uživatelem vybraného textu tagy z výběru elementů, které mu nabízí editor. Uživatel editoru tak nemusí při opravě dokumentu kopírovat text a dávat ho do nově vytvořených tagů, ale může to udělat už při označení textu, který chce zabalit do tagu.



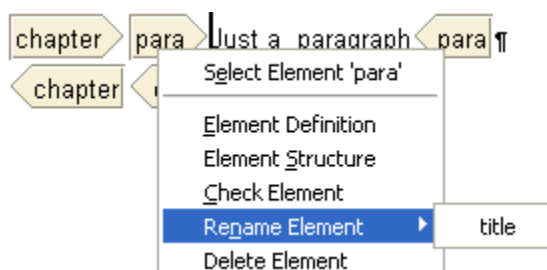
## Funkce editorů



Obrázek 3.1: Vytvoření elementu výběrem textu

## Změna elementu

Editory nabízejí i možnost změnit si element označením a vlastním výběrem nového elementu. Po změně elementu se automaticky změní začínající i ukončovací element.

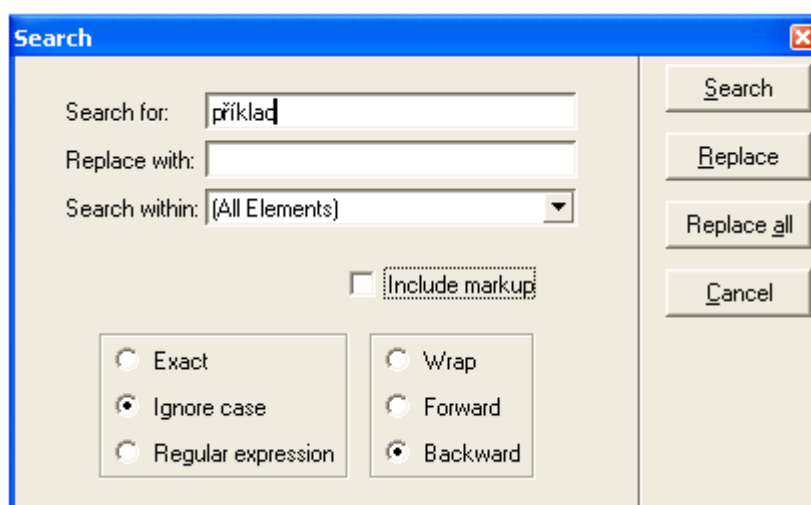


Obrázek 3.1: Změna elementu

## Funkce editorů

### Funkce pro editaci textu

Editory většinou nabízejí základní funkce při editaci textu. Mezi tyto funkce patří vkládání, hledání a nahrazování textu. To je velice pro základní manipulaci s textem.



Obrázek 3.1: Vyhledávání a nahrazení v textu

### Vyhledávání a nahrazování (XPath 2.0)

XPath (XML Path Language) je počítačový jazyk na vybírání jednotlivých elementů z XML a na práci s jejich hodnotami a atributy. XPath je standardem vydaným organizací W3C. Například může nahradit všechny `<b>` které jsou v `<a>` na tag `<c>`. [4]

Zjednodušená možnost

## Funkce editorů

Zjednodušeně Pokročile

Vyhledat prvek: uvnitř aktuálního prvku a za ním (following + descendants) jména:

s atributem:  přesné hodnoty

s textem: přesné hodnoty

Prázdné políčko znamená "jakýkoli"

Stiskem "Escape g" zopakujete poslední vyhledávání.

Obrázek 3.1: Vyhledávání elementu

XPath výraz:

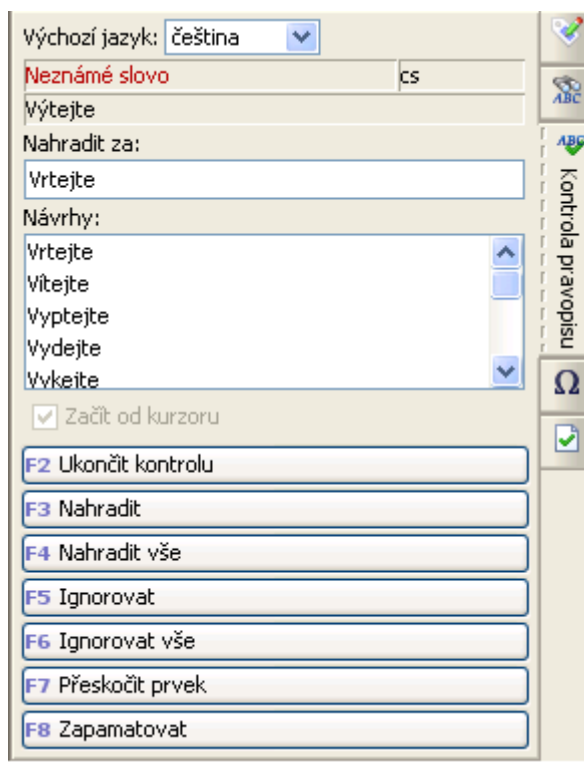
//code

Obrázek 3.1: Vyhledávání pomocí XPath

## Kontrola pravopisu v dokumentu

Funkce umožňuje editoru kontrolovat pravopis v dokumentu. Editor může obsahovat funkci, která importuje ze slovníku pro kontrolu pravopisu, může také obsahovat vkládání slov do slovníku pro kontrolu pravopisu. To napomáhá psát bezchybný text. Velká výhoda je, pokud

editor má český slovník.



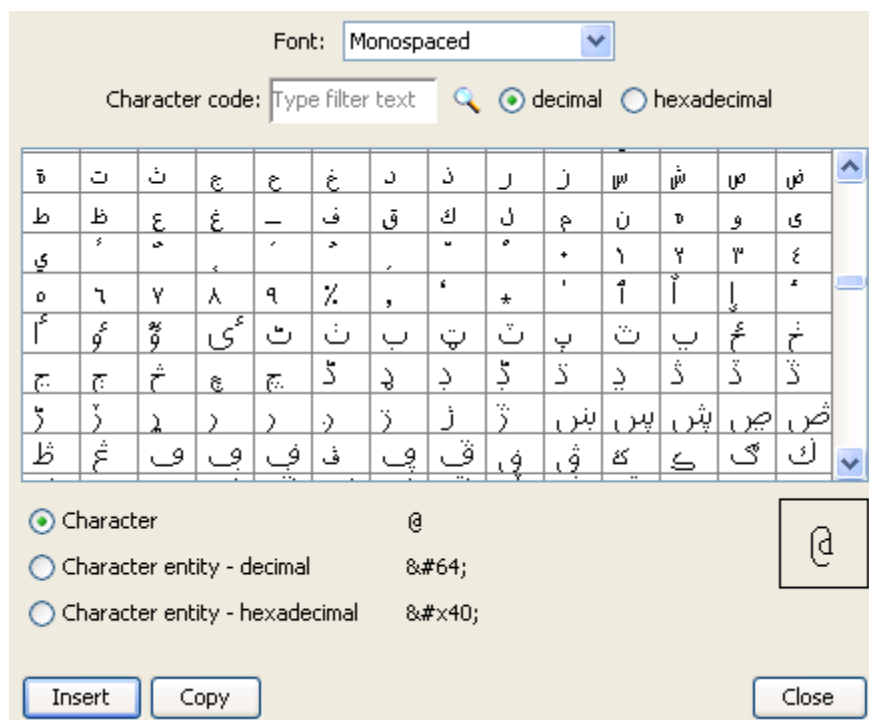
Obrázek 3.1: Kontrola pravopisu a její možnosti

### Vícejazyčný text (Unicode)

Editor obsahuje možnost vkládání vícejazyčného textu a symbolů, může obsahovat tabulku všech existujících abeced. Proto je zde možnost psaní vícejazyčného textu v rámci jednoho dokumentu. “Unicode je globální standard pro kódování znaků, který umožňuje výměnu dokumentů sázených v mnoha různých jazycích a způsobech psaní

## Funkce editorů

(skriptech). V Unicode má každý znak každého skriptu vlastní identifikaci“.[5]



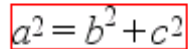
Obrázek 3.1: Tabulka znaků

## MathML

MathML (Mathematical Markup Language) je aplikace XML, s jejímž vývojem začalo konsorcium W3C. Tato aplikace slouží pro popis matematických zápisů a zachycení jejich struktury tak i obsahů za účelem jejich prezentace v elektronické podobě. Editory mající tuto

## Funkce editorů

aplikaci vestavěnou, zjednodušují práci při tvorbě matematických vzorců v dokumentu. Nejnovější verze je MathML 2.0 která vyšla 21. října 2003.[6]


$$a^2 = b^2 + c^2$$

Obrázek 3.1: Matematický vzorec

Tento obrázek nám zobrazí editor na výstupu nebo v grafickém zobrazení, zapsáním kódu dole do dokumentace.

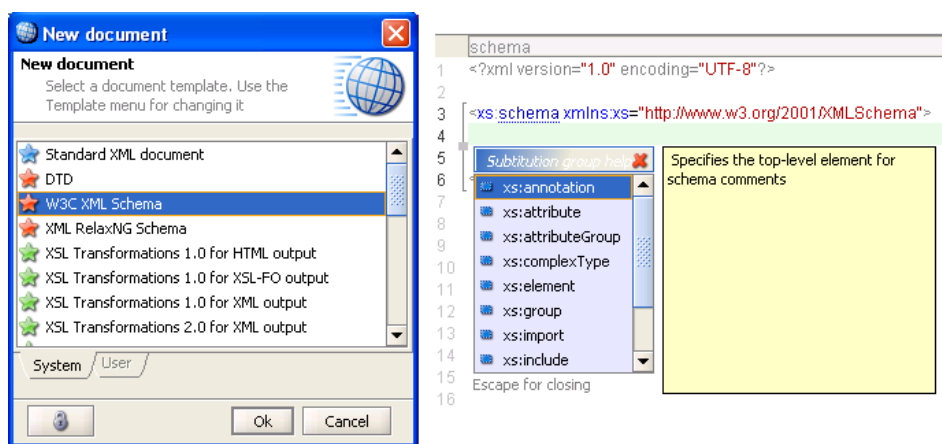
```
<mml:math xmlns:mml="http://www.w3.org/1998/Math/MathML">
  <mml:mrow>
    <mml:msup>
      <mml:mi>a</mml:mi>
      <mml:mn>2</mml:mn>
    </mml:msup>
    <mml:mo>=</mml:mo>
    <mml:msup>
      <mml:mi>b</mml:mi>
      <mml:mn>2</mml:mn>
    </mml:msup>
    <mml:mo>+</mml:mo>
    <mml:msup>
```

## Funkce editorů

```
<mml:mi>c</mml:mi>  
<mml:mn>2</mml:mn>  
</mml:msup>  
</mml:mrow>  
</mml:math>
```

## Tvorba a upravování schémat a DTD

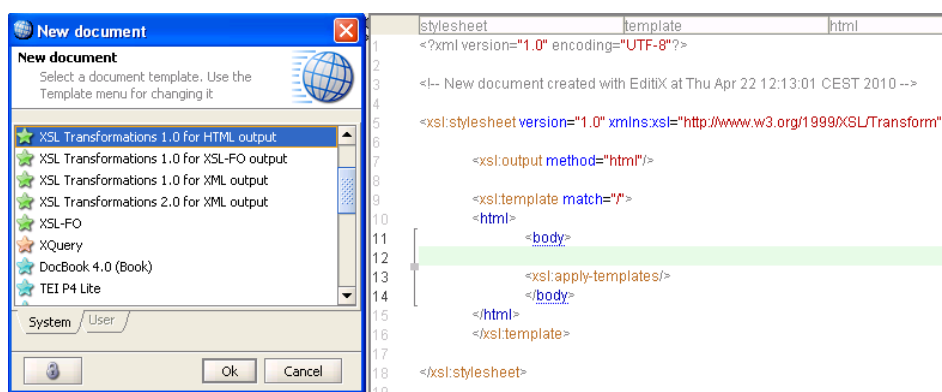
Pro pokročilejší uživatele technologie XML je důležité, aby editor měl podporu pro vytváření a upravování různých schémat a DTD. XML Schémata i DTD jednoznačně definují, jak může xml dokument vypadat. Úpravou schémat i DTD změníme přípustnou strukturu a obsah XML dokumentů podle vlastní potřeby. Také XML schémata i DTD slouží pro validaci dokumentů. [7]



Obrázek 3.1: Tvorba schéma a jeho úprava

## Tvorba a upravování XSLT a XSL-FO šablon

Tato funkce je velice důležitá pro upravování výstupní dokumentace pomocí XSLT a XSL-FO šablon. Šablony říkají XSLT, XSL-FO procesorům jaký způsobem mají jednotlivé značky XML dokumentu přeměnit na výstupní značky.[8]



Obrázek 3.1: Tvorba XSL šablony

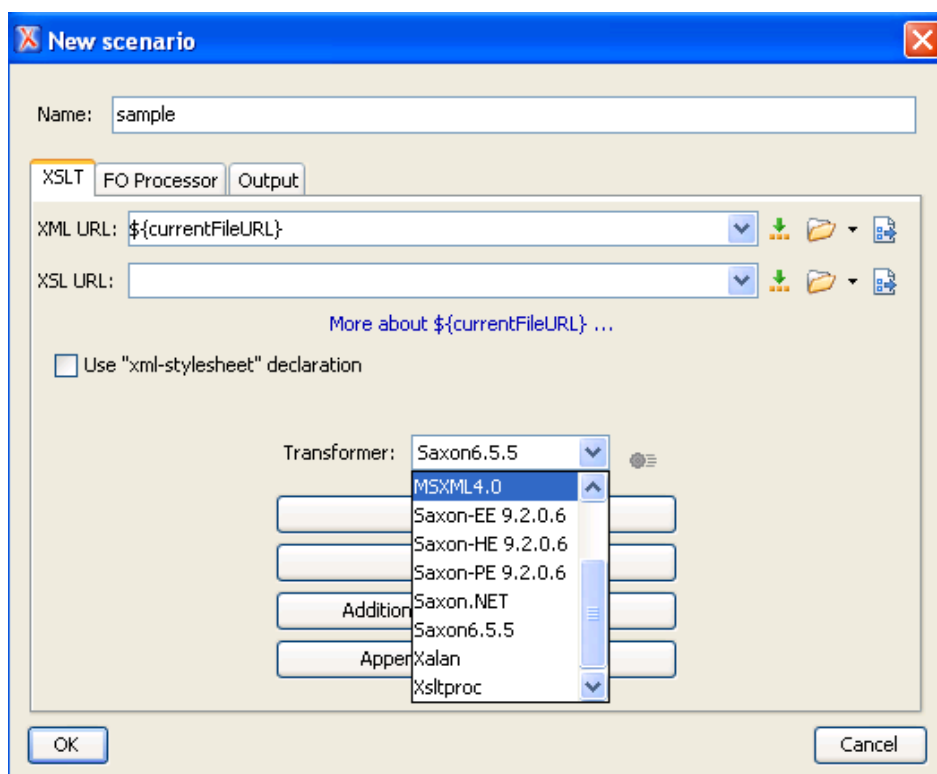
## Vestavěný XSLT a XSL FO procesor

„Procesor XSLT je program, která se používá k transformaci XML dokumentů pomocí jazyka XSLT. XSL dokument (XSL stylesheet) je procesorem použit na vstupní XML dokument. Výsledkem je výstupní dokument, který může být ve speciálním světě formátovaných objektů XSL-FO (Formatting Objects nebo Flow Objects), nebo také dokument ve formátu HTML, XHTML, případně dokument využívající jiný XML prostor (XML namespace). Dnes již existuje celá řada XSLT procesorů, které umožňují zpracování XML dokumentů na základě XSLT stylu, ale



## Funkce editorů

jen několik z nich plně podporuje XSLT 1.0 standard, ba dokonce XSLT 2.0.“[9] Mezi nejznámější XSLT procesory patří MSXML, Xalan, Saxon.

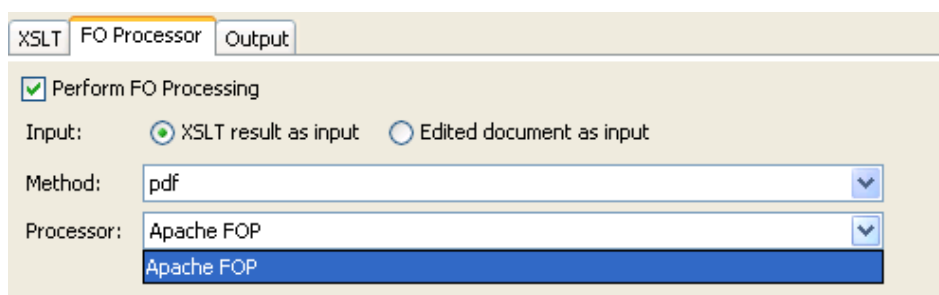


Obrázek 3.1: Podpora procesorů v editoru

„XSL-FO procesor je program pro transformaci XML dokumentu pomocí XSLT šablony, která obsahuje tzv. FO (Formatting Objects) značky. Vznikne mezivýstup XSL-FO dokument, který je transformován na tištěný výstup, jiný výstupní formát nebo je pouze zobrazen na obrazovce. Nejčastějšími výstupními formáty, které umí FO procesory

## Funkce editorů

vytvářet, jsou PDF (Portable Document Format), PS (Post Script), RTF (Rich Text Format). Další možností, jak z XML dokumentu vytvořit např. PDF, je použít samostatný XSLT procesor, pomocí kterého vytvoříme XSL-FO dokument. Tento dokument poté zpracujeme FO procesorem. Doporučil bych používat tuto variantu. XSLT procesory, které jsou součástí FO procesorů, mohou být buď zastaralé, nebo nemusí podporovat některé XSLT značky. Další nespornou výhodou je kontrola FO dokumentu pomocí XML editoru. Třetí možností je přímo napsat FO dokument a převést FO procesorem. “[10] Mezi nejznámější XSL FO procesory patří XEP, FOP.



Obrázek 3.1: Podpora FO procesorů v editoru

## Zobrazení dokumentu

Editory nabízejí různé možnosti zobrazení dokumentu. Různí uživatelé rádi pracují v tom či onom zobrazení. Mezi nejzákladnější zobrazení patří „kódové zobrazení“. V tomto zobrazení uživatel pracuje jen v XML kódu. V dalším zobrazení „grid view“ slouží pro prohlížení a upravování XML tabulkových dat. Existuje také „stromové zobrazení“,

## Funkce editorů

kteřé nám dokument zobrazí s elementy ve stromové struktuře. V „grafickém zobrazení“ uživatel pracuje s XML dokumentem, aniž by viděl používané tagy v dokumentu.[11]

```
2 <!DOCTYPE article PUBLIC "-//OASIS//DTD DocBook XML V4.4//EN"
3     "http://www.docbook.org/xml/4.4/docbookx.dtd">
4 <article>
5     <title>Welcome to Docbook</title>
6     <sect1>
7         <title>Inline Markup and Images</title>
8         <para>This is a sample showing that <code>&lt;oxygen/agt;</code> can be used to edit documents in
9             conformity with the dockbookx.dtd.</para>
10        <para>This is a <code>Docbook</code> figure, inserted using the <code>imagedata</code>
11            tag:</para>
12        <figure>
13            <title>Lake in Fagaras</title>
14            <mediaobject>
15                <imageobject>
16                    <imagedata fileref="../lake.jpeg" scale="100"/>
17                </imageobject>
18            </mediaobject>
19        </figure>
```

Obrázek 3.1: Kódové zobrazení

<?xml version="1.0" encoding="UTF-8"	
<!DOCTYPE article	
article	title Welcome to Docbook
sect1 (2 rows)	title
	para
1 Inline Markup and Images	para (2 rows)
2 CALS Tables and Lists	para
	figure
	table

Obrázek 3.1: Tabulkové zobrazení

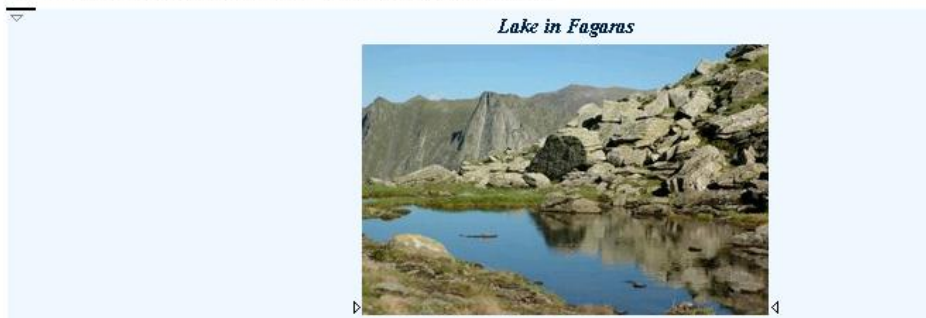
## Article: Welcome to Docbook

---

### Section 1: Inline Markup and Images

This is a sample showing that `<math>x^2 + y^2 = z^2</math>` can be used to edit documents in conformity with the docbookx.dtd.

This is a Docbook figure, inserted using the `<img alt="Lake in Fagaras" data-bbox="448 274 705 396" />` tag:



Obrázek 3.1: Grafické zobrazení

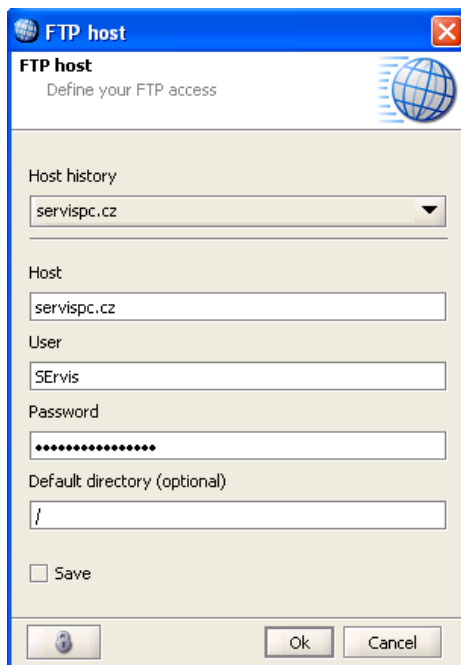
### Práce s velkými dokumenty (10MB)

Používání velkých XML dokumentů sebou nese velké nevýhody. Některé editory nezvládnou dokument ani otevřít. Jiné editory dokument otevřou, ale poté se zpomalí natolik, že práce v nich se stává neefektivní. Nejlepší jsou ty editory, které dokument otevřou a jakoukoli práci v nich (vyhledávání a nahrazování) zvládnou rychle za provozu.

### Vzdálená editace

Některé editory nabízejí vzdálenou editaci XML dokumentů, pomocí FTP protokolu. Vzdálené soubory mohou být přidány do projektu a mohou být použity pro XSL a FO transformaci. [12]

## Funkce editorů



Obrázek 3.1:Vzdálené připojení pomocí FTP

## Práce s více soubory

Pro uživatele pracující s více dokumenty najednou je tato možnost editorů velice výhodná. Uživatelé zde mohou spustit více dokumentů a při práci se mezi soubory přepínat bez potřeby jednotlivé dokumenty ukládat a vypínat.



Obrázek 3.1:Pracování ve více souborech

### **3.2 Jazyky pro popis schémat**

#### **DTD**

Definice typu dokumentu neboli DTD (z anglického document type definition) je jazyk určený pro popis XML případně SGML dokumentů. DTD přesně určuje strukturu daného dokumentu, vymezuje povolené prvky (elementy i atributy) a určuje tak standardy, kterými se dokument řídí. DTD také používají validátory, které právě podle ní rozhodují, zda dokument odpovídá definovaným pravidlům (tzv. je validní). Aby to bylo ještě trochu obtížnější, kromě veřejně známých DTD navíc existují i DTD systémové, tj. definice, které si vytvoříme sami. [13]

#### **W3C XML Schema**

W3C XML Schema je schematický jazyk z dílny konsorcia W3C, který je určen k popisu struktury a datového obsahu XML dokumentů. W3C XML Schema umožňuje mnohem přesnější definici obsahu jednotlivých částí dokumentu, než DTD. Také je ale komplikovanější než DTD. W3C XML Schema nenahlíží na XML dokument jako na textové sdělení, ale jako na sadu prvků (elementy a atributy) různých datových typů. V podstatě každý element je považován za prvek určitého datového typu s definovanými vlastnostmi. Datové typy se dají v W3C XML

## *Funkce editorů*

Schema odvozovat a jejich vlastnosti dědit. Celý princip se velmi podobá objektově orientovaným programovacím jazykům.[14]

### **RELAX NG**

„Relax NG je schémový jazyk pro XML, to znamená, že popisuje nějaký model. Je jednoduchý a snadno naučitelný. Dále má vestavěnou podporu jmenných prostorů, což je bezesporu jeho velkou výhodou. Jedním z hlavních rysů Relax NG je to, že popisuje stejně elementy a atributy, pokud je to možné. Má omezenou podporu smíšeného a neuspořádaného obsahu. Relax NG se dá prokládat jiným anotačním jazykem. Syntaxe Relax NG je ve dvou základních typech. První syntaxe se nazývá XML syntaxe, které je přístupná a transformovatelná pro XML programové nástroje. Druhá se nazývá kompaktní a je zručnější, srozumitelnější pro zpracování lidmi. Já se zde budu zabývat pouze kompaktní syntaxí. Výhodou této syntaxe je použití jednoduchého mechanismu, využívajícího tzv. vzorů, ale také rozšiřitelnost pomocí anotací. Velká nevýhoda Relax NG je, že nelze nastavit implicitní hodnoty. V dalších kapitolách bude popisována právě tato kompaktní syntaxe.“[15]

### **Schematron**

„Dříve zmíněné schémové jazyky jako DTD, W3C XML Schema a RELAX NG víceméně definují gramatiku pro dokument XML. Schematron je odlišný. Pomocí Schematronu jde zapsat tvrzení

## *Funkce editorů*

o přítomnosti nebo absenci určitých vzorů v dokumentu. Validace oproti Schematronu pak vrací seznam tvrzení, který vznikne kontrolou vzorů oproti dokumentu. Pro zápis vzorů se přitom používá dobře známý jazyk XPath. To má dvě velké výhody – máme k dispozici poměrně silné vyjadřovací prostředky XPathu a pro validaci dokumentu nám stačí XSLT procesor, protože schematronové schéma lze převést na XSLT transformaci.“ [16]

### ***3.3 Podpora pro konkrétní schémata***

#### **DocBook**

DocBook je sémantický značkovací jazyk určený pro technickou dokumentaci. Původně byl určený pro psaní softwarové a hardwarové dokumentace, ale nyní je možné psát v něm jakoukoli práci. Umožňuje svým uživatelům vytvořit obsah dokumentu ve formě, která zachycuje logickou strukturu obsahu, tento obsah pak může být zveřejněn v různých formátech, aniž by uživatelé museli provádět změny zdroje. Nejnovější verze DocBook je DocBook 5. [17] [18]

#### **TEI**

„TEI je rozsáhlý projekt, jehož cílem je vytvoření skupiny standardizovaných DTD vhodných pro uchovávání a výměnu knih. Na několika univerzitách ve světě probíhají projekty, které převádějí



## *Funkce editorů*

klasická díla (např. Shakespearova) do elektronické podoby právě pomocí TEI. Kromě elementů pro popis struktury textů nalezneme v TEI speciální elementy, který umožňují v dokumentech vyznačit údaje důležité pro literární, historické nebo lingvistický výzkum.“ [19] Poslední verze TEI je P5.

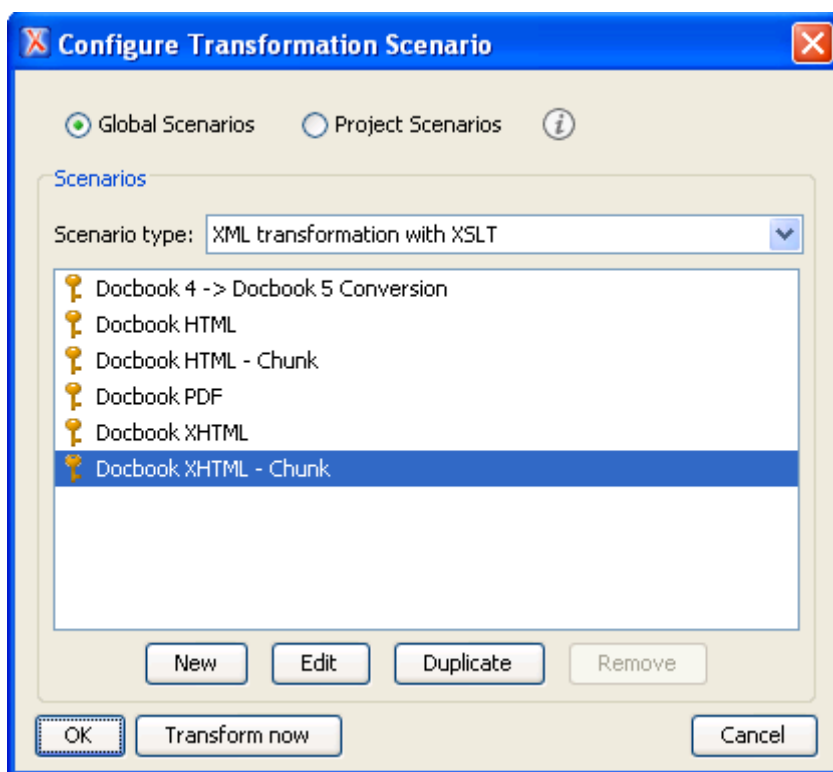
## **DITA**

„DITA (Darwin Information Typing Architecture) zjednodušuje psaní, korektury, publikování a výměnu informací v XML orientovaných na témata a vytváření mapování DITA. Technické dokumenty lze nyní snadno udržovat a aktualizovat pomocí výkonných funkcí pro vytváření knih a rozvržení stránek, včetně automatického číslování a křížových odkazů. Nová funkce dovolí technickým autorům zvýraznit provedené úpravy textu v dokumentu, což zjednoduší proces korektury.“ [20]

### ***3.4 Přednastavené výstupní formáty***

Jde o formáty, do kterých půjde XML dokument editorem transformovat. Každý editor nabízí určité množství nadefinovaných výstupních formátů. To uživateli velice zjednodušuje práci. Hlavně pokud jde o začátečníky, mohou si díky nastaveným možnostem výstupních formátů rychle vytvořit náповědu z XML dokumentu. Mezi nejznámější formáty patří: PDF, HTML, z něho vycházející XHTML, PostScript či HTML Help. [21]

## Funkce editorů



Obrázek 3.4: Přednastavené výstupní formáty

## 4 Visual Studio 2005 a XML

Ve Visual Studiu 2005 lze dokumentovat zdrojový kód pomocí speciálního typu komentářů. Dokumentace se píše ke třídám, jejich metodám, vlastnostem a dalším prvkům. V programu lze pak vygenerovat kostru dokumentace automaticky, kde se následně při kompilaci vytvoří XML soubor. Tímto XML souborem s dokumentací vytvořeným Visual Studiu 2005 můžeme ve stejném softwaru vytvořit i jednoduché webové stránky nebo XML soubor můžeme přiložit k dll knihovně, kde se z tohoto souboru načítají data pro Intellisence. Dokumentace se píše pomocí tří lomítek (///) před jakýkoli datový typ a field nebo pomocí /\*\* \*/ , kde na první řádek dokumentace zadáte /\*\* a pokračujete v psaní dokumentace, pokud dokumentace bude pokračovat i na další řádek, je nutno přidat na tento řádek \* a pokračovat v psaní dokumentace. Po dokončení dokumentace musíte zadat \*/.[22][23]

### 4.1 Psaní komentářů

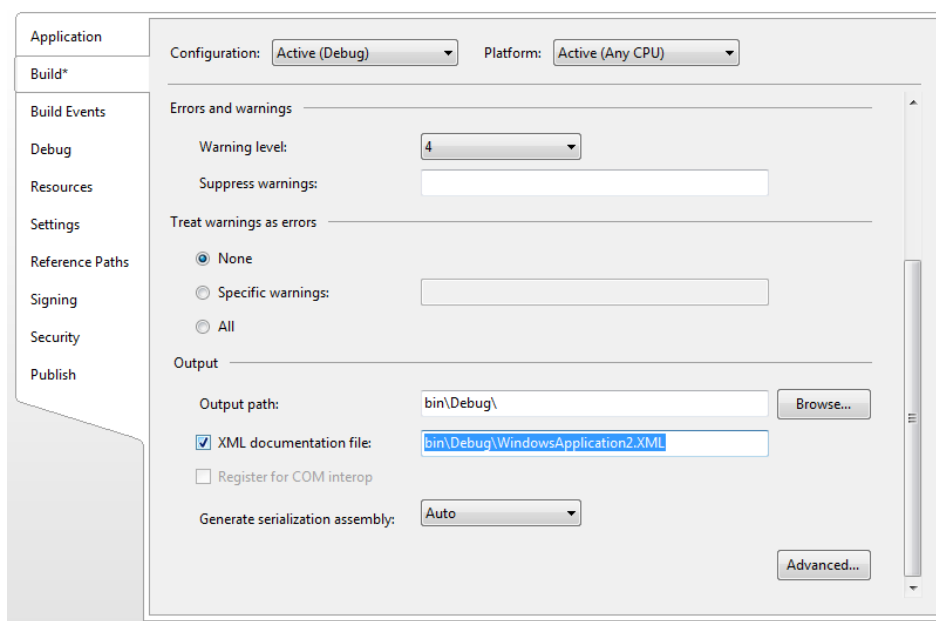
Psaní komentářů ve Visual Studiu 2005 jde jen ve Visual C#. Při vypsání /// nebo /\*\* nám řádek zezelená, při přechodu na další řádek Visual C# automaticky doplní oddělovače dokumentace (/// nebo \*). Při psaní komentářů se používají základní tagy, které Visual C# nabízí. Mezi nejzákladnější tagy patří <summary>, tento tag by neměl u žádné třídy ani metody chybět. Do tagu <summary> se dává popis metody. Popsat

popis k jednotlivým parametrům se používá tag `<param>`. Pro popis návratové hodnoty, používáme tag `<returns>`. Při popisování doplňujících informací nebo jakýchkoli záludností máme tag `<remarks>`, rozdělení textu do více odstavců rozdělíte pomocí tagu `<para>`, který tvoří odstavec. Tag `<paramref>` vytváří odkaz na parametr metody, podobný tag je `<see>`. U tagu `<see>` se odkaz zadává pomocí atributu *creff*, který je při kompilaci o funkčnosti kontrolován. Mezi jedny z důležitých patří tag `<example>`, který je určen na psaní příkladů. Zdrojový kód se vloží přímo, pomocí tagu `<code>`. Mezi poslední tagy patří tag `<c>`, který zvýrazní jednoslovné části kódu (třídu, metodu, vlastnost). Při použití tagu `<exception>` popisujete výjimky, ke kterým může dojít při volání a vysvětlit situace při kterých se vyhodí výjimka. Pokud budeme chtít vložit do dokumentace seznam, použily bychom tag `<list>`. [22][24]

## 4.2 Generování dokumentace

Generování dokumentace ve Visual Studiu 2005 není automatické, proto je nutné nastavit možnosti projektu, aby se nám vytvořila i XML dokumentace. Ve vlastnostech projektu si v menu vlevo najdeme položku „Build“. Otevřeme ji a v části „Output“ si zakřížkujeme „XML documentation file“. Nastavíme si, kam si dokumentaci budeme ukládat a uložíme. Nyní kdykoli dáme „Start debugging“ (tlačítkem F5) se nám vytvoří XML soubor. Výsledný XML soubor navíc ještě obsahuje tagy `<doc>`, `<assembly>`, `<members>`, `<member>`. Element *doc* je

kořenový element obsahující element *assembly* a poté element *members*, ve kterém je uložena samotná dokumentace. Individuální části dokumentace naleznete uvnitř tagů *member*, které mají pomocí atributu *name* stanoveno k jaké části kódu patří. Pro příklad *name="F:WindowsFormsApplication1.Form1.components"* znamená, že dokumentace patří k typu (F- field) *WindowsFormsApplication1.Form1.components* (další předpony jsou například T – type, M - metohod).[1]



Obrázek 4.2: Nastavení Visual Studia 2005 pro dokumentaci

## 5 XML Editory

Při psaní dokumentace pomocí XML se používají editory podporující XML. XML dokument je textový soubor doplněný o tagy, můžeme použít kterýkoli textový editor. Přijdeme tak ovšem o spousty užitečných funkcí, které by nám práci mnohonásobně usnadnily. Lepší editory nám kontrolují strukturu dokumentu a radí, které atributy a elementy můžeme vložit na dané místo, mohou nabízet možnosti dokument zformátovat do dalších formátů. Editory mohou poskytovat při tvorbě grafické zobrazení dokumentace.[25]

### 5.1 Typy XML editorů

V dnešní době se editory dají rozdělit do dvou skupin:

- Textové editory
- Grafické editory

#### 5.1.1 Textové editory

Nabízejí jen práci ve zdrojovém kódu. Textové XML editory jen zvýrazňují syntaxi, jiné editory zvýrazňují syntaxi a automaticky kontrolují strukturu dokumenty a nabízející elementy a atributy. Mezi textové editory patří XML editor EditX XML Editor 2010. Tento editor zvýrazňuje jak syntaxi tak nabízí elementy v místě kurzoru. Editory mohou obsahovat i stromové zobrazení dokumentu.[25]

### 5.1.2 Grafické editory

Grafické editory rozdělujeme do tří druhů. Editory se rozdělují podle možnosti zobrazení dokumentu. První typ editoru dokáže dokument vytvořit v částečném WYSIWYG zobrazení. Tento editor zobrazuje text dokumentu zformátovaný, ale jsou v něm vidět jednotlivé tagy.[25] Do tohoto druhu editoru patří editor epcEdit. Mezi další typ grafických editorů patří WYSIWYM editory. WYSIWYM editory se soustředí na zachování smyslu každého prvku. Například hlavičky, nadpisy, odstavce jsou označeny odpovídajícími XML tagy a celá stránka se tak v prohlížeči zobrazí zamýšleným způsobem. WYSIWYM je „*What You See Is What You Mean*“ v překladu „*Co vidíte, to máte na mysli*“. Do WYSIWYM editorů patří XMLmind XML Editor. Mezi poslední typ editorů patří úplné WYSIWYG editory. Při tomto typu editorů editor zobrazuje verzi dokumentu na obrazovce takovou, jakou ve skutečnosti bude i výsledná verze dokumentu, kdyby byla vytištěna na papíře.[26] WYSIWYG je „*What You See Is What You Get*“, v překladu to znamená „*co vidíš, to dostaneš*“. Do takového typu editoru patří <oxygen/> XML Editor. [27]

## 6 Tvorba XML dokumentace

Nástrojů pro tvorbu dokumentace existuje mnoho, ale jedním z nejpoužívanějších je formát DocBook. Tento formát je normativně definován pomocí jazyka Relax NG. Používá se pro zápis textových dokumentů, které jsou používány jako dokumentace a nápověda k softwaru. Tento formát však není sestaven jen na dokumentaci, jeho možnosti se dají použít i při psaní knih, článků a všech možných textových dokumentů. Mezi nejčastější typ dokumentů je kniha. Proto si popíšeme tvorbu knihy v DocBooku ve verzi 5.0 která je primárně založena na RELAX NG místo DTD, ze kterého se budou dále generovat ostatní formáty schémat (DTD, W3C XML Schema).[28] Tento typ dokumentu je velice volný, proto vyhovuje většině požadavků. Kniha se skládá dále ze čtyř druhů elementů:

- Věnování - Stránka s věnováním se téměř vždy vyskytuje na přední straně knihy
- Navigační komponenty - do tohoto druhu elementů prvky určené pro navigaci: *toc* pro rejstříky a seznamy titulů (seznamy tabulek, obrázků, příkladů atd.) a *index* pro indexy
- Divize - divize jsou na první hierarchické úrovni nižší než kniha, obsahují části a reference. Části knih se skládají z jednotlivých komponent. Reference naopak obsahuje odkazy.
- Komponenty - do této části patří elementy, které dělí knihu na úrovni kapitol – poděkování z knihy nebo jiné komponenty



(*acknowledgements*), dodatek v knize nebo článek (*appendix*), článek (*article*), bibliografie (*bibliography*), kapitoly (*chapter*), Text na zadní straně knihy (*colophon*), věnování knihy (*dedication*), slovníček pojmů (*glossary*), index (*index*), část (*part*), úvod (*preface*), sbírka údajů (*reference*), obsah (*toc*). [29]

## **6.1 Sekce**

Komponenty knih a článků můžeme strukturovat s různými elementy:

*Sect1*, *sect5* jsou sekce přesně po sobě jdoucí, kterých je pět a nejde nějakou vynechat nebo přeskočit (*sect2* musí být po *sect1*). *Section*, tento element je alternativou k číslovaným sekcím a sekce se dají vnořit do libovolné hloubky. Je to totiž rekurzivní element. Za to *Simplesect* je element pro sekce které nejde dále rozdělit na podsekce. Sekce se neobjevuje v obsahu. *Bridgehead* je element pro samostatný nadpis oddílu bez obsahu sekce. *Refsect1... refsect3* jsou sekce, které umožňují členit položky reference. Existují pouze 3 úrovně číslované části elementů v sekci. *Glossdiv*, *bibliodiv* a *indexdiv*- elementy pro členění slovníčku, seznam literatury a rejstříku. Na rozdíl od sekcí tyto elementy nejsou specifikované. [29]

## **6.2 Metainformace**

Většina elementů v sekcích a výše, a mnoho dalších elementů obsahují obal pro metainformace o obsahu. Tento element se jmenuje

*info*. Ve starších verzích DocBooku se tento element jmenoval *bookinfo*, *chapterinfo*. Do metainformace můžete přidat bibliografické informace jako je autor (author), název (title), copyright (copyright), aj. [29]

### **6.3 Blokové elementy**

Blokové elementy jsou ty, co text dělí na odstavce a podobné úseky. Tvoří samostatný odstavec. Odstavce, seznamy, tabulky a blokované citace jsou jedny z nejběžnějších blokových prvků. [29]

#### **6.3.1 Seznamy**

V DocBooku ve verzi 5 je 8 elementů na tvorbu seznamů. *Itemizedlist* značí neseřazený seznam s odrážkami. Naopak *orderedlist* je číslovaný seznam. *Variablelist* je spíše seznam pojmů a definic nebo popisů. Mezi méně používané seznamy patří *Simplelist*, *segmentedlist*, *glosslist*, *bibliolist*, *calloutlist*. [29]

#### **6.3.2 Upozornění**

Existuje pět elementů pro upozornění či výstrahu: *caution*, *important*, *note*, *tip* a *warning*. [29]

#### **6.3.3 Elementy zachovávající řádkování**

Tato prostředí zachovávají mezery a konce řádků ve zdrojovém textu. Například *address* je element pro zápis poštovní adresy, nabízející další elementy jako *city*, *country*, *fax*, *phone*, *postcode*, atd. *Literallayout*

zachovává konce řádků a zachovává šířku a font písma. Element *programlisting* je určený pro zařazování výpisu programů. *Screen* element zachycuje nějaký výstup z obrazovky. *Screenshot* je speciální element pro zařazování sejmutých obrazovek. *Synopsys* obsahuje popis syntaxe funkce nebo příkazu. [29]

#### **6.3.4 Příklady, čísla a tabulky**

Pro vkládání příkladů, tabulek nebo čísel jsou podporovány elementy *gymple*, *informalexample*, *figure*, *informalfigure*, *table* a *informaltable*. [29]

#### **6.3.5 Odstavce**

Existují tři elementy na tvorbu odstavců. *Para*, *simpara*, *formalpara*. [29]

#### **6.3.6 Rovnice**

Jsou dva blokové elementy na tvorbu rovnic. *Equation* a *infomalequation*. Rovnice se vkládají jako alternativní text nebo obrázky. [29]

#### **6.3.7 Grafika (obrázky)**

V grafice se nejčastěji používají elementy *figure* a *screenshot*. Mediální objekty mohou obsahovat 5 druhů elementů. *Audioobject*, *imageobject*, *imageobjectto*, *videoobject*, *textobjec*. [29]

### 6.3.8 Otázky a odpovědi

*Gandaset* je element vhodný pro často kladené otázky a jiné soubory otázek. *Qaudaentry* obsahuje otázky a její odpovědi. Soubor otázek a odpovědí může být rozdělen do sekcí pomocí *qandadiv*. [29]

### 6.3.9 Úkoly a postupy

Element *procedure* obsahuje kroky jak postupovat (*substep*, *stepalternatives*). [29]

### 6.3.10 Synopse

DocBook obsahuje řadu elementů pro popis příkazu, funkce a třídy přehledy. *Cmdsynopsis*, *funcsynopsis*, *classsynopsis*. [29]

### 6.3.11 Různé blokové elementy

Pokud budeme chtít psát delší citaci, použijete *blockquote*. *Epigraf* použijeme pro krátký úvod (nejčastěji citaci), na začátku dokumentu. Na řadu navazujících chybových zpráv, použijete *msgset*. Pro sidebar použijete *sidebar*. [29]

## 6.4 Inline elementy

Pro uživatele DocBooku je k dispozici nepřeborné množství elementů. Inline elementy přiřazují význam jednotlivým částem textu a proto záleží na uživateli, jak přesně bude dokument značkovat. [29]

### 6.4.1 Obecné elementy

Tyto inline identifikátory jsou ty, které se obecně vyskytují při psaní. *Abbrev* se označuje zkrácené slovo, chceme-li napsat zkratku, využije se element *acronym*. Při zvýraznění textu zas *emphasis*, pokud chceme napsat poznámku pod čarou, využijeme *footnote*. *Phrase* označíme část textu a chceme-li dát text do uvozovek, použijeme prvek *quote*. Do elementu *trademark* se dává obchodní značka. [29]

### 6.4.2 Odkazy

Element *Anchor* označuje místo v dokumentu. Pokud máme citaci položky ze seznamu literatury, bude se nám hodit element *station*. *Citerefentry* se používá, pokud chceme napsat citaci ze stránky. Název citace zapíšeme do *citetitle*. *Firstterm* se využije při prvním výskytu pojmu. Element *glossterm* použijeme, když máme pojem ze slovníku. Odkazuje-li na jiné místo v dokumentu, použije se *link*. *Olink* odkazuje na jiný dokument zapsaný nepřímou. *Ulink* nám vytvoří odkaz na URL adresu a poslední element *trf* nám odkazuje na jinou část dokumentu. [29]

### 6.4.3 Značkování

Používají se pro označení textu pro speciální prezentace. Element *foreignphrase* se využije při psaní fráze v cizím jazyce. *Wordasword* potřebujeme, pokud chceme napsat slovo, které je jen slovo a nezastupuje nic jiného. Při výstupu generovaném z počítače se využije

*computeroutput*. Text chápaný jako literál zapíšeme pomocí elementu *literal*, *markup* se použije při zapsání značkování které má být zobrazeno tak jak je napsáno. Znak nebo řetězec označující začátek vstupního pole na obrazovce počítače zapíšeme pomocí elementu *prompt*. Obsah, který může být uživatelem změněn, zapíšeme s využitím elementu *replaceable*. Element *tag* využíváme při psaní součástí XML(nebo SGML) značkování. *Userinput* se píše pro údaje zadané uživatelem. [29]

#### **6.4.4 Matematické výrazy**

Mezi matematické elementy patří celkem 4. *Inlinenequation* se zapisují matematické rovnice nebo výrazy. *Mathphrase* se zapíše matematická věta, výraz, která může být zastoupeny s obyčejných textovým a malým množstvím značek. Dolní index se zapisuje elementem *subscript* a naopak horní index pomocí *superscript*. [29]

#### **6.4.5 Uživatelské rozhraní**

Tyto elementy popisují aspekty uživatelského rozhraní. *Accel* označuje grafické uživatelské rozhraní (GUI), klávesové zkratky. Text na tlačítku v GUI napíšeme pomocí elementu *quibutton*. *Guiicon* je element označující text nebo obrázek vystupující jako ikona v GUI. Text na popisce (štítku) popíšeme *quilabel* a pomocí *guimenu* zapíšeme název menu. Zapsat jde i název menu terminálu v GUI (*guimenuitem*), název submenu (*guisubmenu*), text napsaný na klávese (*keycap*), kód určené klávesy (*keycode*), kombinace kláves (*keycombo*), symbolický název na

klávesnici (*keysym*), výběr nebo serie výběrů z menu (*menuchoice*), Tlačítko myši (*mousebutton*), klávesová zkratka odpovídající v menu (*shortcut*). [29]

#### 6.4.6 Programování

Mnohé z technických inline elementů v DocBooku se vztahují k programování. Můžeme zde popsat název třídy pomocí elementu *classname*. Pomocí *constant* můžeme popsat konstantu, error code zapíšeme elementem *errorcode*, jméno chyby *errorname* a klasifikaci chybové zprávy zas *errortype*. Název funkce nebo procedury zapíšeme *function*, *literalem* zas text jako literál, text hlášení zapíšeme *msgtext*. Hodnotu nebo symbolický odkaz na hodnotu pomocí *parameter*. Jednotku dat spojenou s určitou částí počítačového systému zapíšeme *property*, obsah který může nebo musí nahradit uživatel *replaceable*, Hodnotu vrácenou funkcí *returnvalue*, název který je hodnotou vrácen před zpracováním symbol, jednotka informací *token*, klasifikace hodnoty element *type* a posledním typem je název proměnné *varname*. [29]

#### 6.4.7 Operační systémy

Název aplikace zapíšeme elementem *application*, název spustitelného programu nebo jiný softwarový příkaz *command*, Proměnnou prostředí *envar*, Název prostředí *filename*, text hlášení *msgtext*, možnost softwarového povelů *option*, hodnotu nebo symbolický odkaz na hodnotu *parameter*. Opět řetězec označující vstupní

pole na obrazovce *prompt* a poslední je položka nebo pojem svázaný se systémem *systemitem*. [29]

#### 6.4.8 Pro všeobecné použití

Mezi poslední inline elementy patří elementy pro všeobecné použití. Název aplikace (*application*), název nebo část databáze (*database*), emailovou adresu (*email*), název souboru (*filename*), fyzickou součást počítačového systému (*hardware*). Literál zapíšeme pomocí *literal*, možnost softwarových povelů *option*, Nepovinné informace *optional*, obsah který může nebo musí být nahrazen uživatelem *replaceable*, název který je nahrazen hodnotou před zpracováním symbol, jednotka informací *token*, klasifikace hodnoty *type*. [29]

### 6.5 Tvorba dokumentu

Vytvoření dokumentu je velice jednoduché. Každý dokument by měl mít na začátku dokumentu hlavičku oznamující, o jaký typ dokumentu se jedná. V případě XML by tato hlavička měla mít tvar

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

Kde *version="1.0"* značí verzi XML dokumentu a *encoding="UTF-8"* určuje typ použitého kódování.

```
< book xmlns="http://docbook.org/ns/docbook" version="5.0" >
```



Tento řádek obsahuje kořenový element dokumentu - kniha. Všechny prvky jsou v DocBook XML balíčku, takže kořenový element má atribut `xmlns` pro nastavení jmenného prostoru "`http://docbook.org/ns/docbook`". Dále musí být kořenový element dokumentu DocBook mít verzi `version="5.0"`, která určuje verzi formátu na kterém je dokument postaven. `xmlns="http://docbook.org/ns/docbook"` slouží pouze jako identifikátor. [29]

Toto je náš první dokument v DocBook verzi 5:

```
<?xml version="1.0" encoding="UTF-8"?>
<book xmlns="http://docbook.org/ns/docbook" version="5.0">
<title>První dokument v Docbook 5</title>
  <article>
    <title>PODNADPIS</title>
    <para>Zde je text prvního článku v knize</para>
  </article>
</book>
```

## **7 Porovnání editorů**

### **7.1 *EditX XML Editor***

EditX XML Editor je výkonný a snadno ovladatelný editor XML, pro Windows, Linux a Mac OS X. EditX poskytuje uživateli širokou škálu funkcí XML. Tento editor je freeware patřící do skupiny textových editorů.

#### **7.1.1 Instalace**

Z internetových stránek výrobce máte na výběr, jestli chcete instalovat program s Javou VM nebo bez Javy VM. Poté zkopírovat instalační balíček „editx2010-win.exe“ na disk a spustit instalaci, následně náš průvodce provede instalaci. Nainstaloval jsem EditX XML Editor 2010

#### **7.1.2 O programu**

**Výrobce:** JAPISoft

**Licence:** Freeware

**Nejnovější verze:** EditX XML Editor 2010

**Podpora jazyků:** Angličtina

**Pro platformy:** Windows, Unix/Linux, Mac OS

**Web:** [www.editix.com](http://www.editix.com)

**Dokumentace:** Editor nabízí dokumentaci na jediné HTML (s menu na začátku) stránce výrobce. Nápověda tedy bude vždy aktuální. V případě že nejste připojeni k internetu, to je nevýhodné.

### 7.1.3 Odzkoušení programu

Odzkouším editor a zapíši do tabulky kladný nebo záporný výsledek.

Funkce editoru	Hodnocení
Zvýraznění syntaxe	ANO
Validace XML souboru v editoru	ANO
Průběžná validace	NE
Nabídka elementů a atributů	ANO
Automatické dokončování elementů a atributů	NE
Vkládání a vyjímání XML struktury	ANO
Vytvoření elementu výběrem textu	NE
Změna elementu	NE
Funkce pro editaci textu	ANO
Vyhledávání a nahrazování (XPath 2.0)	ANO
Kontrola pravopisu v dokumentu	NE
Vícejazyčný text (Unicode)	ANO

MathML	ANO
Tvorba a upravování schémat a DTD	ANO, ANO
Vestavěný XSLT a XSL FO procesor	ANO, ANO
Tvorba a upravování XSLT a XSL-FO šablon	ANO, ANO
Práce v kódu	ANO
Práce v tabulkovém zobrazení	NE
Práce v stromovém zobrazení	NE
Práce v grafickém zobrazení	NE
Práce s velkými dokumenty (10MB)	ANO
Vzdálená editace	ANO
Práce s více soubory	ANO

Podpora jazykových pro popis schémat	Hodnocení
DTD	ANO
W3C XML Schema	ANO
Schematron	NE
RELAX NG	ANO

Podpora pro konkrétní schémata	Hodnocení
DocBook	ANO
TEI	ANO
DITA	ANO

Přednastavené výstupní formáty	Hodnocení
PDF	ANO
HTML	ANO
XHTML	ANO
HTML Help	ANO
PostScript	ANO

#### 7.1.4 Hodnocení editoru

Instalace tohoto editoru je velice jednoduchá. Stačí stáhnout soubor na disk a nainstalovat a vše se pomocí průvodce nainstaluje. Zvýraznění syntaxe je pohodlné a uživatel si barevnost může sám nastavit. Velice výhodná je nabídka elementů a atributů, která nabízí v místě jen ty elementy a atributy které jsou povoleny. Dokument musíme validovat ručně, a když dokument není validní, tak se u

řádkování dokumentu zobrazí červeně znaménko o chybě a dole v programu nám vyjede okno o zprávách (chybách). Editor podporuje vkládání a vyjímání XML struktury. V dokumentu jsem se pokusil vybrat text, abych tento text ohraničil tagy, ale tato verze funkce nepodporuje a odkazuje na stránky výrobce. Takto editor odkazuje i při změně elementu. Editor pro editaci textu má základní funkce, které splňují základní požadavky. Možnost vkládání vícejazyčného textu také funguje a nabízí uživateli v okně celou tabulku znaků. Bohužel matematické vzorce v této funkci nejsou povoleny. Jedině pokud bych si chtěl vytvářet zvlášť dokument pak je tato funkce možná. Mě osobně by to ale bylo trošku nepříjemné. V editoru si mohu otevřít více souborů a mezi soubory v pohodlí pracovat. Editor nabízí tvorbu schémat, DTD a jejich úpravu. V šablonách najdeme i XSLT a XSL FO styly ve více možnostech, to se mi na editoru velice líbí, člověk nemusí nic hledat nebo nově vytvářet. Dokumenty je možno zobrazit jak v textovém zobrazení tak i ve stromovém. Při testování velkého souboru editor prošel na jedničku. Byl velice rychle načten a pracovalo se s ním stejně rychle jako s nově vytvořeným prázdným dokumentem. Pro vzdálené připojení k souboru je v této verzi povolen jen FTP přístup. Podpora jazyků pro popis schémat je dobrá, editor podporuje DTD, W3C XML Schema a RELAX NG. DocBook je zde podporován ve verzi 4, bohužel při výběru z šablony editor nabízí jen pro tvorbu v DocBooku knihu. Dále je zde podporována DITA a TEI. Editor nabízí velice příjemnou škálu výstupních formátů. Pro DocBook je to PDF, HTML, HTML Help,

XHTML, Java Help, FO, XML, PRINT, PCL, PS, TXT a SVG. Sice sám osobně bych využil jen pár formátů, ale v žádném případě to není na škodu.

### 7.1.5 Bodové ohodnocení

Funkce editoru	18b
Podpora jazyků pro popis schémat	3b
Podpora pro konkrétní schémata	3b
Přednastavené výstupní formáty	5b
Moje hodnocení editoru	9b
Celkem bodů	38b

## 7.2 *epcEdit*

Je Německý plně vybavený XML/SGML editor, který využívá rozsáhlé možnosti knihovny TkSGML, která je součástí instalačního balíčku *epcEdit*. Editor má být v budoucnosti k dispozici pod open source licenci. Nyní aby byl použit zdarma, je důležité při instalaci stáhnout aktivační klíč ze stránek výrobce.

### 7.2.1 Instalace

Zkopírovat instalační balíček „epcedit-126-win32.exe“ na disk a spustit instalaci, následně náš průvodce provede instalaci. Nainstaloval jsem epcEdit 1.2.6. Poté co jsme nainstalovali epcEdit bylo nutné si stáhnout z webových stránek výrobce aktivační kód. Stažený soubor je nutné rozbalit a při spuštění epicEdit editoru spustit instalaci licence, kde je nutné zadat adresu souboru, který jsme rozbalili. Nyní už je instalace epcEdit editoru XML/SGML dokončena.

### 7.2.2 O programu

**Výrobce:** Heinz Detlev Koch & Roman Halstenberg

**Licence:** Freeware

**Nejnovější verze:** epcEdit 1.2.6.

**Podpora jazyků:** Angličtina

**Pro platformy:** Linux (Intel), Solaris (SPARC), Microsoft Windows (ME, 2000, XP, NT)

**Web:** [www.epcedit.com](http://www.epcedit.com)

**Dokumentace:** EpcEdit má uživatelskou příručku obsahující podrobné informace o všech aspektech používání epcEdit. Navíc je k dostání i manuál TkSGML, který je určen pro programátory, kteří chtějí změnit a rozšířit epcEdit. Uživatelská příručka je buď to v menu programu, nebo na webových stránkách výrobce. Je dělaná v PDF a HTML.



### 7.2.3 Odzkoušení programu

Odzkouším editor a zapíši do tabulky kladný nebo záporný výsledek.

Funkce editoru	Hodnocení
Zvýraznění syntaxe	ANO
Validace XML souboru v editoru	ANO
Průběžná validace	NE
Nabídka elementů a atributů	ANO
Automatické dokončování elementů a atributů	NE
Vkládání a vyjímání XML struktury	ANO
Vytvoření elementu výběrem textu	ANO
Změna elementu	ANO
Funkce pro editaci textu	ANO
Vyhledávání a nahrazování (XPatch 2.0)	NE
Kontrola pravopisu v dokumentu	ANO
Vícejazyčný text (Unicode)	NE
MathML	NE
Tvorba a upravování schémat a DTD	NE, ANO

Vestavěný XSLT a XSL FO procesor	NE, ANO
Tvorba a upravování XSLT a XSL-FO stylů	NE, NE
Práce v kódu	ANO
Práce v tabulkovém zobrazení	NE
Práce v stromovém zobrazení	NE
Práce v grafickém zobrazení	NE
Práce s velkými dokumenty (10MB)	ANO
Vzdálená editace	NE
Práce s více soubory	NE

Podpora jazyků pro popis schémat	Hodnocení
DTD	ANO
W3C XML Schema	ANO
Schematron	NE
RELAX NG	NE

Podpora pro konkrétní schémata	Hodnocení
DocBook	ANO

TEI	ANO
DITA	NE

Přednastavené výstupní formáty	Hodnocení
PDF	ANO
HTML	NE
XHTML	NE
HTML Help	NE
PostScript	NE

#### 7.2.4 Hodnocení editoru

Při stažení tohoto editoru jsem si všimnul složitějších věcí. Editor po instalaci potřebuje soubor, který musíte stáhnout v balíčku Zip a poté na disk rozbalit a v editoru zadat cestu k tomuto souboru. Program nabízí základní nastavení syntaxe. Editor nabízí funkční validaci souboru, ale bohužel zde není automatická validace. Editor je výborný při nabízení elementů a atributů v částečném WYSIWYG zobrazení. Bohužel, tato možnost už chybí v textovém zobrazení. Editor v textovém zobrazení nenabízí žádné výjimečnosti než zobrazení zdrojového kódu se

zvýrazněnou syntaxí. V grafické zobrazení editor umožňuje jednoduše vkládat a vyjímat XML strukturu a taktéž je velice lehké označením textu v dokumentu vytvořit element, navíc editor vždy ukáže elementy, který jde zrovna vložit. To, nabízí u změny elementu. Stačí pravým tlačítkem kliknout na element, který chcete změnit vybrat se v menu změnit a vybrat kterým elementem chcete stávající změnit. Funkce pro editace textu je standardní, jako u všech editorů, trošku výhodné však je, že když budeme hledat slovo v určitém tagu, můžeme při vyhledávání tento tag zadat a editor nám bude postupně vyhledávat/měnit slova v tomto tagu. Vyhledávání a nahrazování elementu jsem v editoru vůbec nenašel. Dokonce i tento editor co jsem testoval, umožňuje kontrolu češtiny. Bohužel je na to potřeba nainstalovat speciální program, který editor potřebuje a poté ještě stáhnout slovník. Editor dokáže převádět dokument pouze do PDF dokumentů. Editor umožňuje nad výběrem elementu zobrazit i stromovou strukturu dokumentu. Při práci s velkým dokumentem editor se spustil, ale trvalo velice dlouho dobu, než se s programem mohlo pracovat. Editor podporuje XML a SGML. Podporuje DTD. DocBook verze 4 a TEI.

### 7.2.5 Bodové ohodnocení

Funkce editorů	12b
Podpora jazyků pro popis schémat	2b
Podpora pro konkrétní schémata	2b

Přednastavené výstupní formáty	1b
Moje ohodnocení editoru	4b
Celkem bodů	21b

### 7.3 XMLmind

Je velice pokročilý editor XML dokumentů, který při tvorbě a editaci využívá zcela vizuální prostředí podobné známým textovým editorům (MS Word). Je to WYSIWYM editor.

#### 7.3.1 Instalace

Zkopírovat instalační balíček „xe-perso-4\_5\_2-setup.exe“ na disk a spustit instalaci, následně náš průvodce provede instalaci. Nainstaloval jsem XMLmind 4.5.2. Poté co jsme nainstalovali XMLmind. Máme nainstalovanou základní verzi programu. Nyní pro lepší práci s programem můžeme v menu „Options“ vyberete „Install plugin“ a zde si vybereme češtinu jako jazyk pro práci s editorem a také slovník. Nyní máme nainstalovaný český editor pro tvorbu dokumentace.

#### 7.3.2 O programu

**Výrobce:** Pixware SARL

**Nejnovější verze:** 4.5.2

**Licence:** Shareware

**Podpora jazyků:** Angličtin, Španělština, Ruština, Italština, Polština, Čeština

- **Pro platformy:** Windows XP/Vista/7, Mac OS X, Linux 2.4

**Web:** [www.xmlmind.com](http://www.xmlmind.com)

**Dokumentace:** Editor nabízí dokumentaci v programu. Bohužel není v češtině jako program. Popisuje i základní postup při tvorbě dokumentace.

### 7.3.3 Odzkoušení programu

Odzkouším editor a zapíši do tabulky kladný nebo záporný výsledek.

Funkce editoru	Hodnocení
Zvýraznění syntaxe	NE
Validace XML souboru v editoru	ANO
Průběžná validace	NE
Nabídka elementů a atributů	NE
Automatické dokončování elementů a atributů	NE
Vkládání a vyjímání XML struktury	ANO

Vytvoření elementu výběrem textu	ANO
Změna elementu	ANO
Funkce pro editaci textu	ANO
Vyhledávání a nahrazování (XPath 2.0)	NE
Kontrola pravopisu v dokumentu	ANO
Vícejazyčný text (Unicode)	ANO
MathML	ANO
Tvorba a upravování schémat a DTD	ANO, NE
Vestavěný XSLT a XSL FO procesor	ANO, ANO
Tvorba a upravování XSLT a XSL-FO stylů	ANO, ANO
Práce v kódu	NE
Práce v tabulkovém zobrazení	NE
Práce v stromovém zobrazení	ANO
Práce v grafickém zobrazení	ANO
Práce s velkými dokumenty (10MB)	ANO
Vzdálená editace	NE
Práce s více soubory	ANO

Podpora jazyků pro popis schémat	Hodnocení
DTD	ANO
W3C XML Schema	ANO
Schematron	ANO
RELAX NG	ANO

Podpora pro konkrétní schémata	Hodnocení
DocBook	ANO
TEI	ANO
DITA	NE

Přednastavené výstupní formáty	Hodnocení
PDF	ANO
HTML	ANO
XHTML	NE
HTML Help	ANO
PostScript	ANO



### 7.3.4 Hodnocení editoru

Tento editor nenabízí práci v textovém zobrazení. Při práci v grafickém zobrazení (i stromovém zobrazení) si člověk musí zvyknout na jeho menu pro vkládání elementů. Validovat dokument jde jen ručně a automatické validování zde není možné. Editor ale velice dobře nabízí elementy a atributy vhodné na místě kurzoru. Pomocí menu (Vložit prvek, vložit prvek před, vložit prvek za) se po zvyknutí práce s editorem stává pohodlná. Navíc po pravé straně v editoru vybíráme elementy a máme možnost popsat i atributy. Dobře se mi pracovalo i při vkládání a vyjímání struktury, vytvoření elementu výběrem textu i změnou elementu. Jednoduše se označil celý element a při změně prvku editor nabídl jen ty elementy, které byly možné na místě stávajícího elementu. Vynikající menu opět ukázalo, že při editaci textu editor obsahuje historii příkazů, která ukazuje, co vše se dělalo zpětně. Při vyhledávání elementů editor ukazuje velké množství možností vyhledávání. Editor podporuje XPath 1.0. Nejvíce co se mi na práci líbilo, byla jednoduchá aktualizace pluginů ve které byla i čeština a jde zapnout i automatická kontrola. S kontrolou pravopisu jsem byl nadmíru spokojen, měla všechny možnosti a byla jednoduše dostupná v menu vpravo. Menu vpravo také umožňuje jednoduše ukazovat a vkládat znaky. Při tvorbě matematických vzorců pomocí MathML jde ale podle mě je to velice složité. Editor nabízí tvorbu a upravování schémat ale práce s tímto souborem mi nepřišla moc jednoduchá. Editor navíc obsahuje XSLT procesor a pomocí pluginu můžete stáhnout XSL- FO procesor. Při vytváření

dokumentů se musíte podřídít šablonám editoru a pro tvorbu schémat je jen jedna šablona W3C XML Schema. Editor podporuje práci s více soubory a při práci s velkým 10 MB souborem dokument otevřel, ale práce s ním byla velice zpomalená. Editor podporuje DTD, W3C XML Schema, Schematron. Editor podporuje i konkrétní schémata jako je DocBook, TEI. Docbook je ve verzi 5 a starší verzi 4. Editor má velikou škálu výstupních formátů. To vše po jednoduché instalaci a stáhnutí pluginů. Mezi výstupní přednastavené formáty je výstup v PDF, HTML, HTML Help, PostScript, Java Help, Eclipse Help, Epub. Je ale nutné znát přípony souborů abychom mohli výstup uložit. Editor u některých formátů nabízí více výstupů, např.: HTML, HTML jedna strana, HTML jedna strana a bez obsahu.

### 7.3.5 Bodové ohodnocení

Funkce editoru	17b
Podpora jazyků pro popis schémat	4b
Podpora pro konkrétní schémata	2b
Přednastavené výstupní formáty	4b
Moje ohodnocení editoru	10b
Celkem bodů	37b

## **7.4 <oXygen/> XML Editor**

XML editor je kompletní multiplatformní XML editor. XML Editor obsahující všechny dnešní XML technologie nabízející podporu pro každý dokument, práci s W3C XML Schema, DTD, Relax NG a NRL. Tento editor nabízí uživatelům velkou podporu pro transformaci dokumentů jako HTML, PS, PDF atd. Tento editor patří do třídy WYSIWYG editorů.

### **7.4.1 Instalace**

Stačí zkopírovat instalační balíček „oxygen.exe“ na disk a spustit instalaci. Je důležité, aby se program neinstaloval do stejné složky, ze které se spustil instalační balík. Při instalaci nás průvodce provede v klidu instalací. Nainstalovali jsme <oXygen/> 11.2. Po spuštění stačí zkopírovat licenční klíč do programu, který přišel emailem po registraci.

### **7.4.2 O programu**

**Výrobce:** SyncRO Soft

**Licence:** Shareware

**Nejnovější verze:** 11.2.

**Podpora jazyků:** Angličtina, Francouzština, Němčina, Italština, Japonština

**Pro platformy:** Microsoft Windows, Mac OS X, Linux (Linux 32 bit a Linux 64 bit), Eclipse

**Web:** [www.oxygenxml.com](http://www.oxygenxml.com)

**Dokumentace:** Editor nabízí kompletní nápovědu v Editoru pomocí klávesy F1. Nabízí i dynamickou nápovědu, kterou najdete v menu Help pod názvem Show Dynamic help view.

### 7.4.3 Odzkoušení programu

Odzkouším editor a zapíši do tabulky kladný nebo záporný výsledek.

Funkce editorů	Hodnocení
Zvýraznění syntaxe	ANO
Validace XML souboru v editoru	ANO
Průběžná validace	ANO
Nabídka elementů a atributů	ANO
Automatické dokončování elementů a atributů	ANO
Vkládání a vyjímání XML struktury	ANO
Vytvoření elementu výběrem textu	ANO
Změna elementu	ANO
Funkce pro editaci textu	ANO

Vyhledávání a nahrazování (XPatch)	ANO
Kontrola pravopisu v dokumentu	ANO
Vícejazyčný text (Unicode)	ANO
MathML	ANO
Tvorba a upravování schémat a DTD	ANO, ANO
Vestavěný XSLT a XSL FO procesor	ANO, ANO
Tvorba a upravování XSLT a XSL-FO stylů	ANO, ANO
Práce v kódu	ANO
Práce v tabulkovém zobrazení	ANO
Práce v stromovém zobrazení	ANO
Práce v grafickém zobrazení	ANO
Práce s velkými dokumenty (10MB)	ANO
Vzdálená editace	ANO
Práce s více soubory	ANO

Podpora jazyků pro popis schémat	Hodnocení
DTD	ANO
XML Schema	ANO

Schematron	ANO
RELAX NG	ANO

Podpora pro konkrétní schémata	Hodnocení
DocBook	ANO
TEI	ANO
DITA	ANO

Přednastavené výstupní formáty	Hodnocení
PDF	ANO
HTML	ANO
XHTML	ANO
HTML Help	NE
PostScript	NE

#### 7.4.4 Hodnocení editoru

Instalace byla velice jednoduchá a nebylo potřeba stahování dalších doplňků. Pro uživatele je důležité vlastnit emailovou schránku, na kterou přijde email s registračním kódem pro aktivaci. Nyní program už fungoval. Oproti ostatním editorům, <oxygen/> XML editor nabízí možnost změny syntaxe pro jakýkoli jazyk např.: XML, CSS, DTD. Validace dokumentu byla velice příjemná. Validovat dokument se může validovat automaticky nebo ručně. Velice výhodná mi přišla automatická validace, která automaticky ukazuje validnost při úpravě souboru, místo chyby ukáže v postraní liště a informace o chybě se uživateli objeví v angličtině. Při nabízení elementů a atributů editor nabízí všechny elementy ale také popis elementů. Editor elementy a atributy automaticky ukončuje. U nastavování atributů, jsem viděl v postranním okně všechny možné atributy elementu a mohl jsem si je nastavit, aniž bych na nějaký zapomněl. Standardní pro mě bylo vyjímání a vkládání XML struktury, stejně tak vytvoření elementu výběrem textu a změna elementu. Při vyhledávání nebo změně nabízel editor vynikající funkce. Při vyhledávání všech elementů a atributů se vytvoří okno, kde se seřadí a popíše všechny nalezené elementy. Při změně a vyhledávání editor v možnostech vyhledání v elementech, attributech a jejich hodnot, CDATA, Doctype aj., navíc XPath je podporován s verzí 1.0, 2.0, 2.0 SA. Při kontrole pravopisu v základní nabídce je jen angličtina. Pokud si uživatel bude chtít stáhnout češtinu, čeká ho dlouhý postup, než češtinu nastaví. Slovník má plno možností, například se do slovníku uloží slovo,

když dáme učení. Příště to slovo bere jako správné. Jde zapnout i automatická kontrola a také jen kontrola textu. MathML editor nabízí v grafickém zobrazení, práce při tvorbě vzorce byla nejlepší ze všech editorů, kterých jsem testoval. Pro pokročilejšího uživatele je zde možnost úpravy XML schémat pro všechny jeho podporovaná schémata. Také upravování XSLT a XSL FO stylů. S dokumenty v editoru jde pracovat ve více zobrazeních. Dávám přednost zdrojové podobě, ale když mohu dokument zkontrolovat i v grafické je to pro mě pohodlnější. Velice vynikající práce v grafickém zobrazení. Editor nabízí vzdálené editace přes FTP/SFTP/WebDAV. To beru jako výhodu pro dokumentaci protože odtud si ji mohou programy samy nahrát a tak je velice dokumentace vždy aktuální. Editor má nejnovější DocBook verzi 5 a 4. Editor také nabízí přednastavené výstupy po transformaci- obyčejnou stránku HTML, vícestránkové HTML stránky, to vše i v HTML, možnost výstupu v PDF a také můžeme udělat převod ze staré verze do novější (např.: DocBook 4 na DocBook 5).

#### 7.4.5 Bodové ohodnocení

Základní funkce	26b
Podpora jazykových schémat	4b
Podpora pro konkrétní schémata	3b
Přednastavené výstupní formáty	3b



Moje ohodnocení editoru	8b
Celkem bodů	44

## 8 Zhodnocení výsledků a závěr

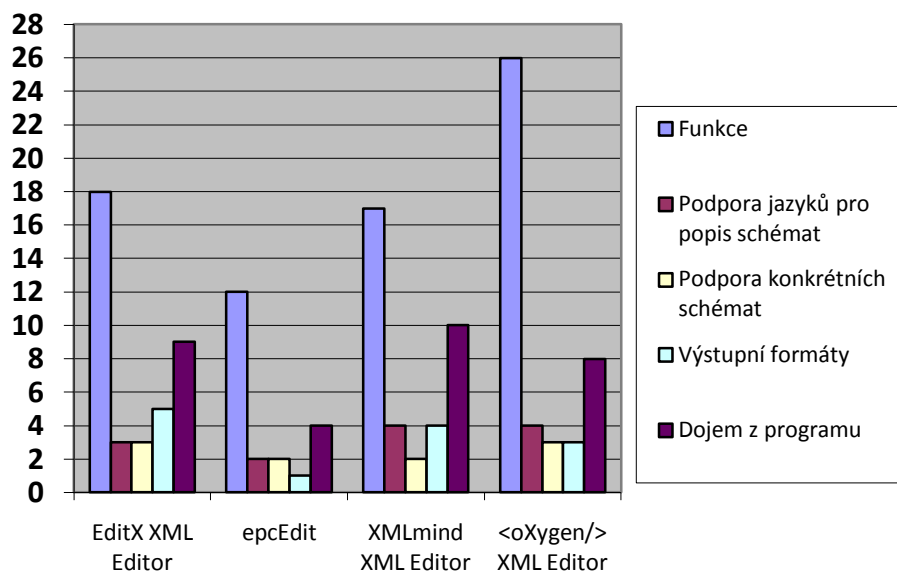
Ve své bakalářské práci jsem zpracoval téma týkající se tvorby dokumentace z Visual Studia 2005 a tvorby dokumentu v DocBook ve verzi 5 která není v českém jazyce moc zdokumentována. Také jsem otestoval XML editory na tvorbu dokumentace. <oxygen/> XML Editor, XMLmind XML Editor, EditX XML Editor a epcEdit jsem testoval na operačním systému Windows XP Professional Service Pack 3. Editorům na tvorbu dokumentace jsem otestoval jejich funkce, výstupní formáty, podporované schémata, Konkrétní podporovaná schémata. Při hodnocení editorů na tvorbu dokumentace jsem shledal důležité věci při výběru editoru na tvorbu dokumentace. Dle mého názoru je, že pro začátečníka v psaní dokumentace pomocí XML je důležité, aby při výběru editoru na tvorbu dokumentace rozhodoval podle výstupních formátů, které editor nabízí a jestli je možnost práce v editoru v českém jazyce. Také velice důležitou věcí je zobrazení dokumentu v XML editoru. Myslím si, že začínajícímu uživateli editoru by vyhovoval grafický editor, kde uvidí, jak může vypadat výsledná dokumentace. Další funkcí při výběru editoru pro začátečníka nabídka elementů a atributů v editoru. Pro pokročilejšího uživatele psaní dokumentace je důležité při výběru editoru zjistit, jaké editor podporuje schémata a jestli je možné tyto schémata upravovat nebo vytvářet. Profesionál by se měl ještě informovat jaké má editor

XSLT a XSL-FO procesory a zda mohu vytvářet a upravovat šablony pro převod dokumentace do výstupních formátů.

V následující tabulce je zachyceno bodové ohodnocení editorů na tvorbu dokumentace za: funkce, podporu jazyků pro popis schémat, podpora konkrétních schémat, výstupní formáty, dojem z programu, dále pak celkový počet bodů, který editor obdržel.

<b>Editory</b>	<b>EditX</b>	<b>epcEdit</b>	<b>XML mind</b>	<b>&lt;oXygen/&gt;</b>
<b>Funkce editorů</b>	18b	12b	17b	26b
<b>Podpora jazyků pro popis schémat</b>	3b	2b	4b	4b
<b>Podpora pro konkrétní schémata</b>	3b	2b	2b	3b
<b>Výstupní formáty</b>	5b	1b	4b	3b
<b>Dojem z programu</b>	9b	4b	10b	8b
<b>Celkem</b>	<b>38b</b>	<b>21b</b>	<b>37b</b>	<b>44b</b>

Z Výsledných dat předešlé tabulky hodnocení editorů na tvorbu dokumentace jsem sestavil graf:



*Graf 8: Výsledné hodnocení editorů*

V následující tabulce je uvedeno výsledné pořadí editorů na tvorbu dokumentace.

1. Místo	<oXygen/> XML Editor
2. Místo	EditX XML Editor 2010
3. Místo	XMLmind XML Editor
4. Místo	epcEdit

Z výsledného hodnocení editorů jsem v tabulce rozdělil editory pro tvorbu dokumentace podle mého názoru vhodné pro začátečníky, pokročilé a profesionály.

XML Editor pro profesionály	<oXygen/> XML Editor
XML Editor pro pokročilé	EditX XML Editor
XML Editor pro začátečníky	XMLmind XML Editor

## 9 Reference

- [1] KOKEŠ, Ondřej. *Lupa.cz* [online]. 2006-2-8 [cit. 2010-04-20]. Nepoužívejte Poznámkový blok. Dostupné z WWW: <<http://halogan.blog.lupa.cz/2006/02/08/nepouzivejte-poznamkovy-blok/2> <http://www.w3.org/TR/REC-xml/>>.
- [2] *W3C.ORG* [online]. 2008-11-28 [cit. 2010-04-20]. Extensible Markup Language (XML) 1.0 (Fifth Edition). Dostupné z WWW: <<http://www.w3.org/TR/REC-xml/>>.
- [3] *KOSEK.CZ* : XML [online]. 2005 [cit. 2010-04-20]. Validace dokumentů XML. Dostupné z WWW: <<http://www.kosek.cz/xml/2005vecery/foil32.html>>.
- [4] *W3C.ORG* [online]. 2007-1-23 [cit. 2010-04-20]. XML Path Language (XPath) 2.0. Dostupné z WWW: <<http://www.w3.org/TR/xpath20/>>.
- [5] *TYPO.CZ* [online]. 2009 [cit. 2010-04-20]. Unicode. Dostupné z WWW: <<http://www.typo.cz/databaze/pismolijny-a-distributori/tvorba-a-editace-fontu/unicode/>>.
- [6] *W3C.ORG* [online]. 2003-10-21 [cit. 2010-04-20]. Mathematical Markup Language (MathML) Version 2.0 (Second Edition). Dostupné z WWW: <<http://www.w3.org/TR/MathML2/>>.
- [7] *XML schémata* [online]. 2005-8-18 [cit. 2010-04-20]. Dostupné z WWW: <<http://www.kosek.cz/xml/schema/>>.

- [8] VŠB | Katedra informatiky FEI VŠB-TUO [online]. 2007 [cit. 2010-04-20]. 6. XSLT. Dostupné z WWW: <<http://www.cs.vsb.cz/behalek/vyuka/pcsharp/text/ch08s06.html>>.
- [9] FENCL, Dušan. *XML na papír : XSLT procesory* [online]. České Budějovice, 2010. 78 s. Bakalářská práce. Jihočeská Univerzita.
- [10] FENCL, Dušan. *XML na papír : XSL-FO procesory* [online]. České Budějovice, 2010. 78 s. Bakalářská práce. Jihočeská Univerzita.
- [11] Oxygen.com[online]. 2010 [cit. 2010-04-21]. XML Editor. Dostupné z WWW: <[http://www.oxygenxml.com/xml\\_editor.html](http://www.oxygenxml.com/xml_editor.html)>.
- [12] [online]. 2010 [cit. 2010-04-21]. Remote Editing over FTP, HTTP/WebDAV and HTTPS/WebDAV. Dostupné z WWW: <[http://www.oxygenxml.com/ftp\\_webdav.html](http://www.oxygenxml.com/ftp_webdav.html)>.
- [13] *Adaptic.cz* [online]. 2010 [cit. 2010-04-21]. DTD. Dostupné z WWW: <<http://www.adaptic.cz/znalosti/slovnicek/dtd.htm>>.
- [14] *Xml.ic.cz* [online]. 2006 [cit. 2010-04-21]. XML Schema. Dostupné z WWW: <<http://xml.ic.cz/chapter5.html>>.
- [15] MORAVEC, Jaroslav. *Jeem.eu* [online]. 2008 [cit. 2010-04-21]. Relax NG v příkladech. Dostupné z WWW: <<http://www.jeem.eu/relax-ng-v-prikladech/1-co-jsou-schemata-relax-ng.html>>.

- [16] *KOSEK.CZ* [online]. 2004 [cit. 2010-04-21]. Kapitola 5. Schematron. Dostupné z WWW: <<http://www.kosek.cz/xml/schema/sch.html>>.
- [17] *DocBook.org* [online]. 2008 [cit. 2010-04-21]. DocBook V5.x. Dostupné z WWW: <<http://www.docbook.org/schemas/5x>>.
- [18] KOSEK, Jiří. *KOSEK.CZ* [online]. 2007 [cit. 2010-04-21]. Úvod. Dostupné z WWW: <<http://www.kosek.cz/xml/db/uvod.html#d4e133>>.
- [19] KOSEK, Jiří. *XML pro každého* [online]. Praha : Grada Publishing, spol. s.r.o., 2000 [cit. 2010-04-21]. Dostupné z WWW: <<http://www.kosek.cz/xml/xmlprokazdeho.pdf>>.
- [20] *Lupa.cz* [online]. 2007-7-25 [cit. 2010-04-21]. Adobe uvádí významný upgrade FrameMakeru. Dostupné z WWW: <<http://www.lupa.cz/tiskove-zpravy/adobe-uvadi-vyznamny-upgrade-framemakeru/>>.
- [21] KOSEK, Jiří. *KOSEK.CZ* [online]. 2007 [cit. 2010-04-21]. Kapitola 8. Výstupní formáty generovatelné z DocBooku. Dostupné z WWW: <<http://www.kosek.cz/xml/db/vystupy.html>>.
- [22] *Tomasz.net* [online]. 2006-4-9 [cit. 2010-04-21]. C# - Dokumentace pomocí XML komentářů. Dostupné z WWW: <<http://tomasz.net/articles/xmldoc.aspx>>.



- [23] *Msdn.microsoft.com* [online]. 2010 [cit. 2010-04-21]. Delimiters for Documentation Tags. Dostupné z WWW: <[http://msdn.microsoft.com/enus/library/5fz4y783\(v=VS.71\).aspx](http://msdn.microsoft.com/enus/library/5fz4y783(v=VS.71).aspx)>.
- [24] *Msdn.microsoft.com* [online]. 2010 [cit. 2010-04-21]. Recommended Tags for Documentation Comments. Dostupné z WWW: <[http://msdn.microsoft.com/enus/library/5ast78ax\(v=VS.71\).aspx](http://msdn.microsoft.com/enus/library/5ast78ax(v=VS.71).aspx)>.
- [25] KOSEK, Jiří. *KOSEK.CZ* [online]. 2001 [cit. 2010-04-21]. Editory. Dostupné z WWW: <<http://www.kosek.cz/clanky/swn-xml/ar02s17.html>>.
- [26] *Freesoftwaremagazine.com* [online]. 2005-04-11 [cit. 2010-04-21]. XML: WYSIWYG to WYSIWYM. Dostupné z WWW: <[http://www.freesoftwaremagazine.com/articles/practical\\_applications\\_xml?page=0%2C1](http://www.freesoftwaremagazine.com/articles/practical_applications_xml?page=0%2C1)>.
- [27] *Jakpsatweb.cz* [online]. 2002, 2009 [cit. 2010-04-21]. Editory HTML stránek. Dostupné z WWW: <<http://www.jakpsatweb.cz/editory.html>>.
- [28] *DocBook.org* [online]. 2008 [cit. 2010-04-21]. DocBook V5.x. Dostupné z WWW: <<http://www.docbook.org/schemas/5x>>.
- [29] *Docbook.org* [online]. 2008 [cit. 2010-04-21]. Creating DocBook Documents. Dostupné z WWW: <<http://www.docbook.org/tdg5/en/html/ch02.html>>.