Úvod do značkovacích jazyků, základní pojmy, logická a fyzická struktura dokumentu

Obsah

Úvod do XML

Co je XML?

- XML je standard (přesněji doporučení konsorcia W3C [http://www.w3.org]) jak vytvářet značkovací jazyky.
- Jedná se tedy o metajazyk.
- Ideově vychází ze staršího standardu SGML (Structure Generalized Markup Language) - XML lze jej téměř chápat jako podmnožinu SGML.
- Se základním standardem úzce souvisí několik dalších, např. XML Namespaces, XInclude, XML Base, XML Infoset.
- Tyto spolu s dalšími standardy (XSLT, XSL-FO, XHTML, CSS...) tvoří "rodinu" standardů XML.

Deset zásad pro specifikaci XML standardů

vyňato z preambule ke specifikaci XML 1.0 (Third Edition)

1. XML shall be straightforwardly usable over the Internet.

   XML bude přímočaře použitelné na Internetu.

2. XML shall support a wide variety of applications.

   XML bude podporovat širokou škálu aplikací.

3. XML shall be compatible with SGML.

   XML bude kompatibilní se SGML.

4. It shall be easy to write programs which process XML documents.

   Tvorba programů zpracovávajících XML bude jednoduchá.

5. The number of optional features in XML is to be kept to the absolute minimum, ideally zero.

   Počet volitelných prvků XML standardu bude málo, optimálně 0.

6. XML documents should be human-legible and reasonably clear.

   XML dokumenty by měly být "lidsky" čitelné a rozumně jednoduché.

7. The XML design should be prepared quickly.

   Návrh XML standardu by měl být rychle hotov.

8. The design of XML shall be formal and concise.

   Návrh XML musí byt formální a správný.

9. XML documents shall be easy to create.

   XML dokumenty bude možné snadno vytvořit.

10.Terseness in XML literal is of minimal importance.

Úspornost XML značkování není podstatná.

Charakteristika XML jazyků

- XML není jeden konkrétní značkovací jazyk; je to specifikace určující, jak mají vypadat značkovací jazyky
- jedná se tedy o "metajazyk"
- konceptuálně jde o zjednodušení SGML standardu, aby se usnadnila práce tvůrcům parserů (analyzátorů) a aplikací

  - například v tom, že každý element musí být uzavřen; pak na přečtení struktury dokumentu nemusíme mít DTD
- XML navazuje na úspěšnou implementaci SGML - jazyk HTML; má podobné charakteristiky z hlediska zaměření na internet
- Vážné diskuse se vedou okolo *binárního XML*, což by měla být rovnocenná reprezentace stejného modelu, jako má "textové" XML.

Aktuální specifikace XML

- Původní specifikace (W3C Recommendation) XML 1.0 na W3C: http://www.w3.org/XML/
- 4th Edition (aktualizace, opravy, ne změny) na Extensible Markup Language (XML) 1.0 (Fourth Edition) [http://www.w3.org/TR/2006/REC-xml-20060816/]
- výborná komentovaná verze téhož na XML.COM (Annotated XML): http://www.xml.com/pub/a/axml/axmlintro.html
- XML 1.1 (Second Edition) [http://www.w3.org/TR/2006/REC-xml11-20060816/] - změny indukované zavedením *UNICODE 3*, lepší možnosti *normalizace*, upřesnění postupu manipulace se znaky *ukončení řádku*. XML 1.1 není už vázaný na konkrétní verzi UNICODE, ale vždy na verzi poslední.

Aktivity W3C

XML Coordination Group	pracovní skupina zprostředkující jakési "rozhraní" mezi jednotlivými skupinami aktivity XML a také navenek
XML Core Working Group	vývoj hlavní specifikace (*XML*) a blízce souvisejících (*Namespaces in XML, XML Information Set, XInclude*)
XSL Working Group	vývoj specifikací Extensible Stylesheet Language (XSL), zahrnující jak *XSL Transformations* (XSLT), tak *XSL Formatting Objects* (XSL/FO). Od zač. 2003 přesunuto pod *W3C Architecture Domain*.
Efficient XML Interchange Working Group	vývoj standardů k efektivní výměně XML dat s důrazem na platformovou přenositelnost a nezávislost na jednotlivých výrobcích (součástí je např. *XML Binary Characterization*)
XML Processing Model Working Group	pracuje na definici skriptovacího jazyka pro XML, na specifikaci operací nad XML daty

XML Linking Working Group	dnes již nefungující skupina pracovala na vývoji *XML Linking Language* (XLink) a *XML Pointer Language* (XPointer).
XML Query Working Group	pracuje na návrhu XML Query Language (*XQuery* a *XPath* - společně s XSL Working Group)
XML Schema Working Group	Připravuje specifikace *W3C XML Schema* k popisu struktury, obsahu, příp. sémantiky XML dokumentů.

Informační zdroje k XML

Základní tutoriály a články

- (výborný úvodní) Koskův seriál o XML pro Softwarové noviny: http://kosek.cz/clanky/swn-xml/index.html
- Seriál o XML na ŽIVĚ [http://zive.cz]
- (obsahuje hodně příkladů) Zvon XML Tutorial: http://www.zvon.org/xxl/XMLTutorial/General/book_en.html
- Tutoriál ke XML na W3 Schools [http://www.w3schools.com/xml/default.asp]
- Microsoft XML Tutorial: http://msdn.microsoft.com/xml/tutorial/
- 101 XML Tutorials: http://www.xml101.com/xml/default.asp
- XML Tutoriály na Beginners.co.uk [http://tutorials.beginners.co.uk]
- Tutoriály na Developerlife.com: http://developerlife.com

Portály k XML

World Wide Web Consortium (W3C)	http://www.w3.org/
XML Startkabel	http://xml.startkabel.nl - aktuality, odkazy *(v angličtině/ nizozemštině - obnovováno cca 1x týdně)*
Zvon	http://zvon.org - asi nejlepší soubor tutoriálů, on-line referencí v mnoha jazycích, místo je hostované v ČR
XML Cover Pages	xml.coverpages.org [http://xml.coverpages.org] - denně aktualizovaný soubor odkazů na články, publikace standardů, software, atd. v oblasti XML. Nejlepší zdroj v této kategorii.
O'Reilly XML.COM	http://xml.com - články, tutoriály atd. na vysoké technické úrovni
IBM DeveloperWorks, sekce XML	http://ibm.com/developer/xml [http://ibm.com/developer/xml/] - články, tutoriály, software atd. na vysoké technické úrovni

Elektronické konference, news, maillisty k XML

XML USENET newsgroup	news:comp.text.xml

XML-DEV	mailto:xml-dev@xml.org - nejznámější maillist ke XML standardům (webový archív [http://lists.xml.org/archives/xml-dev/])

XML software

- IBM AlphaWorks: http://www.alphaworks.ibm.com - alpha-software fy IBM k volnému vyzkoušení
- Free XML Software (L. M. Garshol): http://www.garshol.priv.no/download/xmltools/ - asi nejlepší kolekce odkazů na nekomerční XML software
- XMLSoftware: http://xmlsoftware.com - asi nejlepší kolekce odkazů na obecný XML software (i komerční)

Další odkazy k XML

- Přehled XML aktivit W3C: http://www.w3.org/XML/Activity - specifikace standardů, konference, odkazy na SW, referenční nástroje, odkazy *(obnovováno podle potřeby)*
- *What is XML?* na XML.COM: http://www.xml.com/pub/a/98/10/guide0.html - jeden z úvodních článků ke XML
- XML: XML Quick Syntax Reference Card [http://www.mulberrytech.com] - výborná stručná referenční karta
- výborná komentovaná verze téhož na XML.COM (Annotated XML): http://www.xml.com/pub/a/axml/axmlintro.html

Zdroje k XML na FI

Předměty - podzimní semestr

PA165	Vývoj programových systémů v jazyce Java - T. Pitner, P. Adámek, J. Pavlovič a další
PB029	Elektronická příprava dokumentů - P. Sojka
PV110	Softwarové elektronické publikace I - P. Sojka
PV173	Seminář Laboratoře zpracování přirozeného jazyka

Předměty - jarní semestr

IB047	Úvod do korpusové lingvistiky a počítačové lexikografie - K. Pala, P. Rychlý
PA105	Technologie informačních systémů II - J. Král
PA154	Nástroje pro korpusy - P. Rychlý
PA156	Dialogové systémy - I. Kopeček
PV174	Laboratoř elektronických a multimediálních aplikací - P. Sojka
PV030	Textové informační systémy - P. Sojka
PV113	Softwarové elektronické publikace II - P. Sojka

Software

- Balík **XSLT2** (Jan Pavlovič) Návod k modulu xslt2 [http://www.fi.muni.cz/~xpavlov/xml/index.html]

Struktura XML dokumentů

Syntaxe XML dokumentů

Základním požadavkem kladeným na *každý* XML dokument je, že musí být *dobře utvořen (well-formed)*.

Toto nastane, právě když:

1. Obsahuje *prolog (hlavičku)* a

   právě jeden, tzv. *kořenový element*.

   Dále může před a po kořenovém elementu obsahovat instrukce pro zpracování, komentáře atd. (`Misc`)

2. It meets all the well-formedness constraints given in the specification.

   Musí vyhovovat všem pravidlům pro správné utvoření uvedeným ve specifikaci.

3. Each of the *parsed entities* which is referenced directly or indirectly within the document *is well-formed*.

   Totéž platí pro každou *analyzovanou (parsovanou) entitu* přímo nebo nepřímo odkazovanou v dokumentu.

Podívejte se na tutoriál základů XML v češtině [http://zvon.org/xxl/XMLTutorial/General_cze/book.html]

Rejstřík (glossary) pojmů ke XML [http://zvon.org/index.php?nav_id=173]

Struktura XML dokumentu

- U XML dokumentů rozlišujeme strukturu *fyzickou* a *logickou*.
- Aplikační programátory zajímá většinou jen struktura *logická*,
- autory obsahu, XML editorů, procesorů, atd. může zajímat i struktura *fyzická*.

Fyzická a logická struktura

Struktura logická	*dokument* členíme na *elementy* (jedne z nich je *kořenový - root*), jejich *atributy*, *textové uzly* v elementech, *instrukce pro zpracování*, *notace*, *komentáře*
Struktura fyzická	jeden logický dokument může být uložen ve více fyzických jednotkách - entitách; vždy alespoň v jedné - tzv. *entitě dokumentu - document entity*.

Prvky logické struktury

- uzel (element, atribut, textový uzel, instrukce pro zpracování, komentář)
- element
- atribut

- textový uzel
- instrukce pro zpracování
- komentář

Dále viz např. Koskův seriál o XML na http://kosek.cz/clanky/swn-xml/index.html

Elementy

Jsou objekty *ohraničené počáteční a koncovou značkou - start and end tag*, tedy obecně:

```
<tagname ...tag_attributes...>
  tag_content
</tagname>
```

Příklad 1. Příklad elementu s obsahem (neprázdného)

```
<body background="yellow">
   <h1>textový uzel - obsah elementu h1</h1>
   <p>textový uzel - obsah elementu p</p>
</body>
```

Elementy - prázdné

Je-li obsah prázdný (žádné dceřinné elementy ani textový obsah), pak píšeme pouze *značku prázdného elementu - empty element tag*, např.:

```
<tagname ...tag_attributes... />
```

Příklad 2. Příklad elementu bez obsahu (prázdného)

```
<hr width='50%'/>
```

Z hlediska logického je ekvivalentním zápisem

Příklad 3. Příklad elementu bez obsahu - obě značky

```
<hr width='50%'></hr>
```

Atributy

- Nesou "dodatečné informace" k elementu - např. jeho ID, požadované formátování - styl, odkazy na další elementy...
- Konceptuálně by bylo možné atributy nahradit elementy, ale z důvodu přehlednosti obvykle používáme obojí.
- Obsah atributu na rozdíl od obsahu elementu není nijak (na úrovni obecných zásad XML standardů) dále strukturován.

  Občas se u některých značkování vyskytne snaha o strukturaci obsahu atributů, to však obecně vede k více problémům, než řeší...

- Rovněž fyzické pořadí zápisu více atributů v jednom elementu nemá na logický model vliv.

Atributy - zápis

- Atribut je tvořen *jménem* a *hodnotou*.
- Atributy zapisujeme do počáteční značky elementu (který může být i prázdný).
- Hodnota je *vždy* vložena v uvozovkách či apostrofech a od jména ji dělí znak rovnítka (=).
- Pro *názvy* atributů platí stejná omezení jako pro názvy elementů.
- V rámci jednoho elementu *nejsou* přípustné dva atributy se stejným jménem.

Poznámka

Pokud se používají jmenné prostory, nejsou v jednom elementu přípustná ani dva různě prefixované atributy patřící do stejného jmenného prostoru.

Atributy - příklad

Příklad 4. Atribut 'width' v prázdném elementu

```
<hr width='50%'/>
```

Příklad 5. Atribut 'border' v neprázdném elementu

```
<table border='1'>
   <tr><td>jedna</td><td>dve</td></tr>
   <tr><td>tri</td><td>ctyri</td></tr>
</table>
```

Textové uzly

Nesou textovou informaci.

Např. v následující ukázce je *text* `ahoj!` (nikoli celý element `em`!) textovým uzlem:

```
<em>ahoj!</em>
```

Instrukce pro zpracování

Instrukce pro zpracování (*processing-instruction*) píšeme do značek `<?target content?>`

Informují aplikaci o postupu či nastavení nutném pro zpracování daných XML dat. Nepopisují (nepředstavují) obsah, ale *zpracování* dokumentu.

```
<?xsl-stylesheet href="mystyle.xsl"?>
```

Poznámka

`href` v příkladu *neznamená* atribut; atributy nejsou u instrukce pro zpracování možné!

Notace

Notace (*notation*) píšeme do značek `<!NOTATION name declaration >`

Slouží zejména k popisu binárních (non-XML) entit - např. obrázků GIF, PNG,...

Jde vlastně o *deklaraci způsobu zpracování*.

Komentáře

Podobně jako v HTML - komentář (*comment*) píšeme do značek `<!--text komentáře-->`

Obsahem komentáře je `text komentáře`, nikoli celý komentář i se značkami

Komentář nebývá obvykle pro zpracování významný, ale záleží na aplikaci - může např. uchovávat značky *Servlet-side Includes (SSI)*

Parsery by proto *měly* komentáře zpracovávat - předávat dále.

Např. *SAX parsery* však tak nečiní!!! (resp. činí až v rozšířené verzi SAX2, v Javě balík `org.xml.sax.ext`).

Entity

Entita je základní jednotkou *fyzické* stavby dokumentu. Odpovídá řetězci, souboru...

Parsery by měly entity zpracovávat tak, aby aplikace nemusela o entitách "nic tušit".

Uzel dokumentu

Rozlišujeme:

Uzel dokumentu (Document Node)	• nadřazený kořenovému elementu • může kromě něj obsahovat též komentáře, instrukce pro zpracování, notaci DOCTYPE atd. a
Kořenový element (Root Element)	je nutnou součástí XML souboru. V každém souboru je právě jeden kořenové element.

Podrobněji...

Podrobné informace k syntaxi se dozvíme v následující kapitole Standardy rodiny XML [../standards/index.html]

Znaky v XML dokumentech

Znaky v XML dokumentech

Specifikace povoluje na určitých místech v XML dokumentech (např. název elementu, obsah atributu...) jen některé znaky.

Vzhledem k internacionalizaci a nutnosti zvládnout i exotické jazyky je třeba znát, co se čím myslí.

Musíme rozlišovat:

- *znakové sady* (množiny znaků s pořadovými čísly), tj. přiřazení ordinální hodnoty znaku (např. Unicode) a
- *kódování znaků* (z dané sady), např. UTF-8, tj. ordinální hodnota znaku se kóduje do posloupnosti bajtů

Standardy Unicode, ISO 10646

Oba standardy se zabývají podobnými problémy: řeší znakové sady s více než 256 znaky.

- Původní návrh tzv. 16bitového Unicode: až 64 K znaků, stačí pro evropské, nestačí pro světové jazyky (např. dnes frekventovaná čínstina).
- 32bitový Unicode: pokrývá znaky už "na věky".

V současnosti se z 32bitové škály většinou používá jen tzv. Basic Multilingual Plane (BMP) pokrývající většinu jazyků.

V XML je možné pro názvy (nonterminál *kvalifikovaná jména* - QName) použít znaky z BMP.

Jinak lze v XML dokumentech používat všechny znaky Unicode.

Kódování Unicode

Všechny aplikace XML (zejména aplikace univerzální, parsery) musejí být schopny zpracovat znaky Unicode bez ohledu na kódování.

Přesto je dobré znát nejběžnější kódování:

- osmibitová, tradiční: US-ASCII, ISO-8859-2 (ISO Latin 2), Windows-1250 (=Cp1250) - kódování jen vybrané podmnožiny Unicode.
- UTF-8: kódování všech znaků Unicode, každý znak na 1-6 bajtech, US-ASCII na jednom bajtu, "čeština" na dvou.
- UTF-16: princip stejný jako UTF-8, ale základní ukládací jednotkou je dvoubajtové slovo (16 bitů)
- UCS-2: přímé kódování Unicode, čísla znaků z BMP se zapíší přímo jako dva bajty
- UCS-4: dtto, ale pro celý Unicode a na 4 bajtech - neúsporné, 4 bajty i pro US-ASCII, evropské jazyky...

Pro XML mají klíčový význam UTF kódování, zejména UTF-8 (ale parsery musejí umět obě).

Přípustné znaky

- Přípustné jsou jakékoli UNICODE znaky po x10FFFF (kromě xFFFE, xFFFF a rozmezí xD800 - xDFFF).
- *jména* (*names*) musí být složena ze nemezerových znaků: číslice, písmena, `.` (tečka) − (pomlčka, minus) _ (podtržítko) `:` a dalších, musí začínat písmenem nebo _ `:`
- Kódování těchto UNICODE znaků není podstatné.
- Jako implicitní - není-li v prologu (hlavičce), např.

```
<?xml version="1.0" encoding="Windows-1250"?>
```

uvedeno jinak - se používá UTF-8 nebo UTF-16.

- Rozlišení UTF-8 a UTF-16 se děje pomocí prvních dvou bajtů dokumentové entity (tj. souboru), pomocí tzv. byte-order-mark xFFFE
- Není-li uvedena, předpokládá se UTF-8, čili UTF-8 je implicitní kódování UNICODE znaků v XML dokumentech.

Teoreticky by tedy bylo možné z obsahu souboru rozpoznat přesně, o jaké kódování se u XML dokumentu jedná...