

XML

T. Pitner, L. Bártek, A. Rambousek, L. Grolig

FI MU Brno, 2020

Obsah

- Úvod do XML
- Motivace
- Principy
- Zdroje (@FI, ostatní)

Co je XML?

- Jedná se o standard [W3 Konsortia](#) popisující jak tvořit značkovací jazyky.
- Jedná se tedy o *metajazyk*.
- Je odvozen ze starších standardů (SGML – [Standard Generalized Markup Language](#))
 - Na XML se dá nahlížet jako na téměř podmnožinu SGML.
- Existuje řada standardů spojených s XML
 - XML Namespaces, XInclude, XML Base, XML Infoset
- Tyto standardy spolu s dalšími (XSLT, XSL-FO, XHTML, CSS, ...) tvoří “rodinu” XML standardů.

Deset principů XML Standardů

Z preambule XML 1.0 (3. vydání)

1. XML by mělo být přímočaře použitelné na Internetu.
2. XML bude podporovat širokou škálu aplikací.
3. XML bude kompatibilní z SGML.
4. Tvorba programů zpracovávajících XML bude jednoduchá.
5. Počet volitelných prvků XML standardů bude malý, optimálně 0.

Deset principů XML standardů

6. XML dokumenty by měly být „lidsky“ čitelné a rozumně jednoduché.
7. Návrh XML standardu by měl být rychle hotov.
8. Návrh XML musí být formální a správný.
9. XML dokumenty bude možné snadno vytvořit.
10. Úspornost XML značkování není podstatná.

Charakteristika XML jazyků

- XML není specifický jazyk, je to specifikace, jak tvořit značkovací jazyky.
 - Je to metajazyk.
- Konceptuálně vychází z SGML.
 - Zjednodušeno kvůli snazší tvorbě parserů.
- Jelikož každý element musí být uzavřený.
 - Dokument nemusí mít definováno DTD, pro zpracování struktury.
- XML staví na úspěšné implementaci SGML – HTML.
 - Má podobné charakteristiky, tzn. zaměření na Internet.
- Probíhá seriózní diskuze ohledně binárního XML.
 - Mělo by být možné ho reprezentovat stejně jako textové XML.

Aktuální specifikace XML

- Původní specifikace (W3C Recommendation) [W3C XML 1.0](#)
- 5. vydání [Extensible Markup Language \(XML\) 1.0 Fifth Edition](#)
- [XML 1.1 \(Second Edition\)](#)
 - Změny vyvolané zavedením
 - UNICODE 3,
 - snažší normalizací,
 - Upřesněním zpracování konců řádků
 - XML 1.1. není vázáno na konkrétní verzi UNICODE, ale vždy na poslední.

Tutoriály a články

- Tutoriál k XML na zvon.org
- Tutoriál k XML na W3Schools
- Tutoriál k XML fy. Microsoft
- Tutoriály k XML na [101 XML](http://101XML)
- XML tutoriál na Beginners.co.uk
- Tutoriály na Developerlife.com

Portály vztažené k XML

- [World Wide Web Consortium \(W3C\)](#)
- [XML Startkabel](#)
- Vynikající sbírka tutoriálů a on-line dokumentace v řadě jazyků, hostovaná v CZ
- [XML Cover Pages](#) - denně aktualizovaný souhrn na materiály vázané k XML
- [O'Reilly XML.com](#) - články a tutoriály na vyšší úrovni.
- [IBM DeveloperWorks, section XML](#) - články, tutoriály, software atd. na vyšší úrovni.

Aktivity W3C

- Pracovní skupina Efficient XML Interchange Working Group
 - Vývoj standardů pro efektivní výměnu XML dat s důrazem na přenositelnost a platformní nezávislost konkrétních produktů.
 - Včetně binární specifikace XML.
- XML Processing Model WG
 - vývoj a definice:
 - skriptovacího jazyka pro XML,
 - operací nad XML daty.
- XML Linking WG
 - Dnes již nefunkční skupina.
 - Pracovala na specifikacích [XLink](#) a [XPointer](#).

Aktivity W3C

- Koordinační skupina pro XML
 - „rozhraní“ pro pracovní skupiny a externími spolupracovníky.
- Pracovní skupina XML Core
 - Vývoj specifikací XML a úzce souvisejících doporučení:
 - XML namespaces
 - XML Information Set
 - XInclude

Další odkazy k XML

- [Aktivity W3C](#): specifikace standardů, konference, odkazy na SW, ukázkové nástroje, odkazy
- [What is XML na XML.com](#) - jeden z úvodních článků k XML
- [XML: XML Quick Syntax Reference Card](#) - výborná, jednoduchá referenční karta.
- [Komentovaná verze specifikace XML na XML.com \(Annotated XML\)](#)

Knihy

- XML in Nutshell od E.R.Harolda

Zdroje k XML na FI (předměty)

- Podzimní semestr
 - PA156 – Enterprise Application in Java
 - PB029 – Elektronická příprava dokumentů
 - PV110 – Základy filmové řeči
 - PV173 – Seminář NLP
- Jarní semestr
 - IB047 – Úvod do korpusové lingvistiky a počítačové lexikografie
 - PA154 – Jazykové modelování
 - PA156 – Dialogové systémy
 - PV174 – Laboratoř elektronických a multimediálních aplikací
 - PV113 – Produkce audiovizuálního díla

Co dále?

- Ani XML není univerzální řešení všech problémů při strojové výměně dat.
- Vývoj pokračuje.
- Pro pokročilé (rich) webové aplikace s intenzivní komunikací client-server:
 - Potřeba lepší interoperability a menšího množství dat
 - Formáty jako JSON (JavaScript Object Notation).
 - YAML – ruční popis struktovaných dat.
- Budou probrány později. Hlavní důraz bude na XML a jeho zpracování.

Koncepty a Struktura XML dokumentů

Obsah

- Logická a fyzická struktura dokumentu

Struktura XML Dokumentů

- Základní požadavek na XML dokumenty – musí být dobře utvořené:
 - Obsahují XML prolog a právě jeden kořenový element.
 - Před a za kořenovým elementem mohou být instrukce pro zpracování.
 - Splňuje požadavky na dobře utvořené dokumenty ze specifikace.
 - Každá přímo nebo nepřímo odkazovaná parsovatelná entita je dobře utvořená.
- Další možný požadavek na XML dokument – být validní.

Struktura XML dokumentů (další informace)

- Viz tutoriál [XML Fundamentals](#) (anglicky)
- [Obsah](#) k XML na [zvon.org](#)

Struktura XML dokumentů

- Rozlišujeme:
 - fyzickou a
 - logickou strukturu.
- Aplikační programátory obvykle zajímá pouze logická struktura,
- zatímco pro autory obsahu a editorů může být důležitá i fyzická struktura.

Fyzická a logická struktura

- Logická struktura:
 - dokument se skládá z:
 - elementů – jeden je kořenový,
 - atributů,
 - textových uzlů,
 - instrukcí pro zpracování,
 - symbolů
 - komentářů.

Fyzická a logická struktura

- Fyzická struktura
 - Jeden logický dokument může být uložen ve více fyzických entitách – vždy v alespoň jedné.

Prvky logické struktury

- Uzlem (generickým prvkem) může být:
 - Element – občas nekorektně nazývaný „tag/značka“ (značka je počáteční a koncové značkování ne celý element).
 - Atribut – vždy součást elementu.
 - Textový uzel – text mezi značkami.
 - Instrukce pro zpracování – neobsahuje textový obsah nebo atribut, pouze pro účely zpracování.
 - Komentáře – určeny pro lidské čtenáře.

Elementy

- Objekty ohraničené počáteční a koncovou značkou:

```
<body background="yellow">
```

```
  <h1>text node — content of element h1</h1>
```

```
  <p>text node — content of element p</p>
```

```
</body>
```

Prázdné elementy

- Pokud je element prázdný, píšeme značku prázdného elementu
 - Neobsahuje dceřiné uzly (elementy, textový obsah)

```
<hr width="507"/>
```

- Nebo ekvivalent (z pohledu logické struktury)

```
<hr width="507"></hr>
```

Atributy

- Vždy v počáteční značce elementu

```
<hr width="507">
```

- Fyzické pořadí není důležité a obecně není bráno v potaz.
- Atributy jsou pouze „připojeny“ k elementu a nesou další informaci:
 - ID,
 - požadované formátování (styl) u (X)HTML,
 - odkazy na další elementy.

Atributy vs. Elementy

- Atributy lze nahradit elementy
 - používají se ke zlepšení čitelnosti.
- Obsah atributu nelze dále strukturovat.
- Hodnota atributu není podle standardu strukturovaný.
 - Ačkoliv to není doporučeno, tak aplikace jí mohou strukturova.

Jak zapisovat atributy

- Atribut se skládá z jeho *jména* a *hodnoty*.
- Atributy se vkládají do počáteční značky.
 - Může být prázdná.
- Hodnota atributu je vždy v apostrofech (') a nebo úvozovkách (") a od jména je oddělena =.
- Zápis `width='800'` resp. `Width="800"` znamená to samé.
- Pro jména atributů platí to stejné jako pro názvy elementů.
 - V jedno elementu nemůže být více atributů se stejným jménem.
 - V případě použití jmenných prostorů nemohou být ve stejném jmenném prostoru v jednom elementu dva atributy se stejným jménem.

Textové uzly

- Nesou textovou informaci, textový obsah.
- V následujícím příkladu je textovým uzlem ‚ahoj!‘ ne celý element em.

```
<em>‘ahoj!’</em>
```

Instrukce pro zpracování

- Instrukce pro zpracování se zapisují

`<?target content?>`

- Informují aplikaci o očekávaném

- zpracování
- nastavení.

- Nenesou obsah.

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

- href není atribut,
- instrukce pro zpracování neobsahují atributy.

Notace

- Notace se zapisují
 - `<!NOTATION nazev deklarace>`
- Většinou se používají k popisu binárních/ne XML entit
 - Obrázky, videa, ...
- Jedná se o deklaraci, jak zpracovat binární data.

Komentáře

- Podobné jako v HTML
 - Uzavřené do `<!-- komentár -->`
- Obsah komentář je obsah
 - Nikoliv celý komentář včetně značkování
- Komentáře se většinou nezpracovávají
 - Závisí na aplikaci
 - Např. Server Side Includes je využívají
- Parsery by měly být schopné je předat do aplikace
- SAX parser je ve verzi 1 ignoruje
 - Verze 2 je předává

Entity

- *Entita* – základní jednotka fyzické struktury dokumentu.
- Odpovídá
 - znaku,
 - řetězci,
 - celému souboru.
- Parser pracuje s entitami tak, že aplikace se o nich nedozví.

Uzel dokumentu

- Uzel dokumentu
 - Rodič kořenového elementu.
 - Může obsahovat:
 - instrukce pro zpracování
 - notace
 - DTD
 - ...
- Kořenový element
 - Jádro celého dokumentu.
 - V každém souboru smí být pouze jeden.

Uzel dokumentu

- Rozlišujeme
 - Uzel dokumentu
 - Rodič kořenového elementu
 - Může obsahovat instrukce pro zpracování, notace, DOCTYPE, a
 - Kořenový element
 - Základní část XML dokumentu.
 - V každém dokumentu je právě jeden.

Znaky v XML

- Obsah
 - Znakové sady a jejich kódování
 - Unicode a jeho kódování
 - Znaky v XML
 - Znakové entity

Znaky v XML dokumentech

- Specifikace umožňuje v určitých částech dokumentu jen určité znaky
 - jména elementů
 - obsah atributů.
- Znaková sada (charset)
 - sada znaků, resp. jejich kódů/čísel
 - přiřazení znaku jeho ordinární hodnotě (Unicode).
- Kódování znaků (v dané znakové sadě)
 - např. UTF-8, UTF-16, UTF-32
 - ordinární hodnota zakódovaná do posloupnosti bytů
 - Nepatrně odlišné v XML 1.0 a XML 1.1

Unicode a Standard ISO 10646

- Oba standardy se snaží vyřešit problém:
 - „Znakové sady s více jak 256 není možné zakódovat jako 1 byte – jeden znak.“
- Původně 16bitový Unicode
 - Až 64k znaků
 - dostatek pro všechny evropské abecedy
 - málo pro světové abecedy (čínské, ...)
- 32bitový Unicode
 - Dostatek pro „všechny“ abecedy na světě.
 - Dnes se z 32bitového Unicode používá Basic Multilanguage Plane (BMP)
 - Pokrývá většinu typických jazyků
 - Lze použít pro jména v XML (non-terminal Qualified Name – QName)
 - Pro ostatní znaky lze používat libovolný znak z Unicode.

Kódování Unicode

- UCS 2 – přímé kódování Unicode
 - Znak z BMP kódován přímo jejich ordinální hodnotou.
- UCS 4 – totéž, ale všechny znaky z Unicode na 4 bytech včetně US-ASCII a evropských jazyků – neefektivní.
- UTF – nejdůležitější pro XML
 - Zejména UTF-8, ale parsery by měly znát obě.
 - 1. byte – buď přímo znak z ASCII a nebo číslo roviny, ve které se má znak hledat.

Povolená kódování

- Pokud není uvedeno v prologu, tak UTF-8 resp. UTF-16
 - Např. `<?xml version="1.0" encoding="windows-1250"?>`
- Odlišení podle prvních dvou bytů entity dokumentu
 - Tzv. Byte-order-mark – xFFFE.
 - Pokud chybí předpokládá se UTF-8
 - UTF-8 je implicitní kódování XML

Znakové entity

- Umožňují zápis znaků, které nejsou obsaženy v uživatelském fontu.
- Význam např. při zápisu znaků s jiným zvláštním významem
 - Např. Značkování:
 - >
 - <
 -

Zápis entit

- Odkaz na znak v Universal Character Set (UCS) – formát:
 - `&#nnnn;`
 - dekadický zápis ordinální hodnoty znaku
 - `&#xhhhh;`
 - hexadecimální zápis ordinální hodnoty znaku
 - x musí být malé
 - hexadecimální čísla mohou být jak velká, tak malá.
- Pojmenované entity - `&name;`
 - předdefinované ve standardu
 - explicitně deklarované v DTD
 - *name* je case-sensitive název entity, středník je povinný.