

---

# Kapitola 1. Základní standardy rodiny XML

## Obsah

Specifikace XML 1.1 .....	2
XML 1.0 (Third Edition) .....	2
XML - další tutoriály a články .....	2
Terminologie .....	3
Terminologie .....	3
Terminologie - opakování (2) .....	3
Terminologie - opakování (3) .....	3
Znaky v XML dokumentech .....	3
Znaky v XML dokumentech .....	3
Standardy Unicode, ISO 10646 .....	4
Kódování Unicode .....	4
Znaky v XML dokumentech .....	5
Document Type Definition (DTD) .....	5
Document Type Definition (DTD) .....	5
DTD - tutoriály .....	5
DTD - deklarace typu dokumentu podrobněji .....	6
DTD - podmíněné sekce .....	6
DTD - definice typu elementu .....	6
DTD - definice atributu .....	7
DTD - definice typu hodnoty atributu .....	7
DTD - předpis kardinality (počtu výskytů) atributu .....	8
DTD - implicitní hodnota atributu .....	8
Fyzická struktura (entity) .....	8
Entita - deklarace a použití .....	8
<b>Entity obecné (general)</b> - mohou být .....	8
<b>Entity parametrické (parametric)</b> .....	9
Jmenné prostory .....	9
Jmenné prostory (XML Namespaces) .....	9
Prefixy jmenných prostorů, shoda... .....	9
Příklad implicitního jmenného prostoru .....	10
Příklad explicitního jmenného prostoru .....	10
Obtíže se jmennými prostory .....	10
XML Base .....	11
XML Base .....	11
XML Base - příklad .....	11
XML Inclusions .....	11
XML Inclusions (XInclude) .....	11
XInclude: použití .....	11

XInclude: příklad .....	12
XML Catalogs .....	12
XML Catalogs .....	12
XML Catalogs - příklad .....	12
XML Information Set .....	13
XML Information Set (XML Infoset) - cíle .....	13
XML Infoset - struktura .....	13
Kanonický tvar XML .....	14
Kanonický tvar XML dokumentu .....	14
Kanonický tvar - zásady konstrukce .....	14
Potíže při definici kanonického tvaru .....	15

## Specifikace XML 1.1

### XML 1.0 (Third Edition)

- Původní specifikace (W3C Recommendation) XML 1.0 na W3C: <http://www.w3.org/XML/>
- 3rd Edition (aktualizace, ne změny) na <http://www.w3.org/TR/2000/REC-xml-20001006>
- výborná komentovaná verze téhož na XML.COM (Annotated XML): <http://www.xml.com/pub/a/axml/axmlintro.html>
- na XML 1.1 (Candidate Recommendation) [<http://www.w3.org/TR/xml11/>] - změny indukované zavedením *UNICODE 3*, lepší možnosti *normalizace*, upřesnění postupu manipulace se znaky *ukončení řádku*.

### XML - další 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](http://www.zvon.org/xxl/XMLTutorial/General/book_en.html)
- 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>

# Terminologie

## Terminologie

- Opakování: správně utvořený (*well formed*) dokument
- Nové: platný (*valid*) dokument

Platný podle specifikace znamená *přísnější* omezení než správně utvořený.

Obvykle se validitou myslí soulad s *DTD* (Document Type Definition) dokumentu.

## Terminologie - opakování (2)

- 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>

## Terminologie - opakování (3)

- uzel dokumentu
  - ten je nadřazený kořenovému elementu
  - může kromě něj obsahovat též komentáře, instrukce pro zpracování, notaci DOCTYPE atd
- kořenový element

# 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 čínština).
- 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 zapíše 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).

## Znaky v XML dokumentech

- 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) \_ (podtržítka) : 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...

## Document Type Definition (DTD)

### Document Type Definition (DTD)

- Definice typu dokumentu (použití této definice je pak **deklarace typu dokumentu**)
- Specifikována přímo standardem XML 1.0
- Popisuje přípustný **obsah elementů, atributů**, jejich implicitní (default) hodnoty, definuje použité **entity**
- Může být uvedena jako **interní** nebo **externí** DTD (*internal and external subset*) nebo "napůl" - tam i tam.
- Dokument vyhovující DTD je označován jako *valid* (platný).

## DTD - tutoriály

- Webreview: [http://www.webreview.com/2000/08\\_11/developers/08\\_11\\_00\\_2.shtml](http://www.webreview.com/2000/08_11/developers/08_11_00_2.shtml)
- ZVON: <http://www.zvon.org/xxl/DTDTutorial/General/contents.html>

- XML DTD Tutorial (101): <http://www.xml101.com/dtd/>
- W3Schools DTD Tutorial: <http://www.w3schools.com> [<http://www.w3school.com>]

## DTD - deklarace typu dokumentu podrobněji

Uvádí se těsně před kořenový elementem konstrukcí

- `<!DOCTYPE jméno-kořenového-elt Externí-ID [ interní část DTD ]>`

**Interní** nebo **externí** část (*internal or external subset*) nemusí být uvedena nebo mohou být uvedeny obě.

**Externí identifikátor** může být buď

- `PUBLIC "PUBLIC ID" "URI"` (hodí se pro "veřejná", obecně uznané DTD) nebo
- `SYSTEM "URI"` - pro soukromá nebo jiná "ne zcela standardizovaná" DTD ("URI" nemusí být jen URL na síti, může být i jméno souboru, vyhodnocení se děje podle systému, na němž se vyhodnocuje)

Význam interní a externí části je rovnocenný (a nesmí si odporovat - např. dvě definice téhož elementu).

Obsahem DTD je seznam deklarací jednotlivých prvků - *elementů*, *seznamů atributů*, *entit*, *notací*

## DTD - podmíněné sekce

Slouží k "zakomentárování" úseků DTD např. při experimentování.

- `<![IGNORE[ toto se bude ignorovat ]]>`
- `<![INCLUDE[ toto se zahrne do DTD (tj. nebude se ignorovat)]]>`

## DTD - definice typu elementu

Popisuje možný obsah elementu, má formu `<!ELEMENT jméno-elementu ... >`, kde ... může být

- `EMPTY` - prázdný element, může být zobrazen jako `<element/>` nebo `<element></element>` - totéž
- `ANY` - povolen je libovolný obsah elementu, tj. text, dceřinné elementy, ...
- může obsahovat **dceřinné elementy** - `<!ELEMENT jméno-elementu (specifikace`

dceřinných elementů)>

- může být **smíšený** (MIXED) - obsahující text i dceřinné elementy dané výčtem <!ELEMENT jméno-elementu (#PCDATA | přípustné dceřinných elementy)\*>. Nelze specifikovat pořadí nebo počet výskytů konkrétních dceřinných elementů. Hvězdička za závorkou je *povinná* - vždy je možný libovolný počet výskytů.

Pro specifikaci dceřinných elementů používáme:

- operátor **sekvence** (*sequence, follow with*) ,
- operátor **volby** (výběru, *select, choice*) |
- závorky ( ) mají obvyklý význam
- nelze kombinovat v jedné skupině různé operátory , |
- počet výskytů dceřinného elementu omezujeme specifikátory "hvězdička", "otazník", "plus" s obvyklými významy. Bez specifikátoru znamená, že je povolen právě jeden výskyt.

## DTD - definice atributu

Popisuje (datový) typ, případně implicitní hodnoty atributu u daného elementu.

Má tvar <!ATTLIST jméno-elementu jméno-atributu typ-hodnoty implicitní-hodnota>

## DTD - definice typu hodnoty atributu

Přípustné *typy hodnot* jsou:

- CDATA
- NMTOKEN
- NMTOKENS
- ID
- IDREF
- IDREFS
- ENTITY
- ENTITIES
- výčet hodnot - např. (hodnota1 | hodnota2 | hodnota3)

- výčet notací - např. NOTATION (notace1|notace2|notace3)

Atribut (i nepovinný) může mít implicitní hodnotu:

- "implicitní hodnota" - atribut je nepovinný, ale není-li uveden, chápe se to, jako by měl hodnotu `implicitní hodnota`

## DTD - předpis kardinality (počtu výskytů) atributu

Atributy mohou mít předepsán (povinný) výskyt:

- #REQUIRED - atribut je povinný
- #IMPLIED - atribut je nepovinný
- #FIXED "pevná-hodnota" - atribut je povinný a musí mít právě hodnotu `pevná-hodnota`

## DTD - implicitní hodnota atributu

Atribut (i nepovinný) může mít implicitní hodnotu:

- "implicitní hodnota" - atribut je nepovinný, ale není-li uveden, chápe se to, jako by měl hodnotu `implicitní hodnota`

## Fyzická struktura (entity)

### Entita - deklarace a použití

Rozlišuje se:

- deklarace
- reference (tj. použití) dané (již deklarované) entity.

### Entity obecné (general) - mohou být

- *parsované* - soubory se (správně utvořeným) značkováním,
- *neparsované* - např. binární soubory,
- *znakové* - znaky, např. `&gt;` je referencí na znakovou entitu



## Entity parametrické (parametric)

- mohou být použity *jen v rámci DTD*
- hodí se při např. deklaracích *seznamu atributů* (pokud je dlouhý a vícekrát použitý, nahradíme ho referencí na parametrickou entitu)
- viz např. DTD pro HTML 4.01 - <http://www.w3.org/TR/html4/sgml/dtd.html>
- definicí parametrické entity je např. `<!ENTITY % heading "H1|H2|H3|H4|H5|H6">`

## Jmenné prostory

### Jmenné prostory (XML Namespaces)

- XML Namespaces (W3C Recommendation): <http://www.w3.org/TR/REC-xml-names>
- Existuje také nové *Namespaces in XML 1.1 W3C Recommendation* [<http://www.w3.org/TR/xml-names11/>] 4th February 2004. Andrew Layman, Richard Tobin, Tim Bray, Dave Hollander
- Definiují "logické prostory" jmen (elementů, atributů) v XML dokumentu.
- Dávají uzlům ve stromu XML dokumentu "třetí dimenzi".
- Logickému prostoru jmen odpovídá jeden globálně ("celosvětově") jednoznačný identifikátor, daný URI (URI tvoří nadmnožinu URL).
- NS odpovídající danému URI nemá nic společného s obsahem nacházejícím se případně na tomto URL ("nic se odnikud automaticky nestahuje" - nedochází k tzv. dereferenci daného URI).

### Prefixy jmenných prostorů, shoda...

- V rámci dokumentů se místo těchto URL používají zkratky, *prefixy* těchto NS namapované na příslušné URI atributem `xmlns:prefix="URI"`.

Jméno elementu či atributu obsahující dvojtečku se označuje jako *kvalifikované jméno*, *QName*.

- Dva NS jsou stejné, jestliže se jejich URI shodují po znacích přesně (v kódování UNICODE).
- NS neovlivňují význam textových uzlů.
- Element/atribut nemusí patřit do žádného NS.

- Deklarace prefixu NS nebo implicitního NS má platnost na všechny dceřinné uzly rekurentně, dokud není uvedena jiná deklarace "přemapující" daný prefix.
- Jeden NS je tzv. *implicitní (default NS)*, deklarovaný atributem `xmlns=`
- Na atributy se *implicitní NS nevztahuje!!!*, čili atributy bez explicitního uvedení prefixu nejsou v *žádném NS*.

## Příklad implicitního jmenného prostoru

V následující ukázce je pro celý úryvek platný deklarovaný implicitní jmenný prostor charakterizovaný URI (URL) `http://www.w3.org/1999/xhtml`

### Příklad 1.1. Implicitní jmenný prostor

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
  <body>
    <h1>Huráááá</h1>
  </body>
</html>
```

## Příklad explicitního jmenného prostoru

V následující ukázce je deklarován a přiřazen prefixu `xhtml` jmenný prostor charakterizovaný URI (URL) `http://www.w3.org/1999/xhtml`

### Příklad 1.2. Jmenný prostor mapovaný na prefix

```
<xhtml:html xhtml:xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
  <xhtml:body>
    <xhtml:h1>Huráááá</xhtml:h1>
  </xhtml:body>
</xhtml:html>
```

## Obtíže se jmennými prostory

Dosud ne všechny parsery dokážou rozpoznávat NS. ...*i když problémy jsou s tím dnes výjimečné...*

NS jsou nekompatibilní s DTD (DTD přísně rozlišuje např. jméno `xi:include` a `include`, přestože patří do stejného NS a mají tedy z hlediska aplikace obvykle stejnou interpretaci/význam).

# XML Base

## XML Base

- XML Base, W3C Recommendation 27 June 2001: <http://www.w3.org/TR/xmlbase/>
- Standard pro vyhodnocování relativních URL v odkazech z/na XML dokumenty.
- Definiuje použití vyhrazeného atributu `xml:base` označujícího základ pro vyhodnocování relativních URL.
- Doplnuje se se standardem *XLink*.
- Respektuje princip "překrývání" bázové adresy nastavené v nadřazeném elementu.

## XML Base - příklad

### Příklad 1.3. `xml:base` určuje základ pro relativní URL



#### Poznámka

Všimněte si použití vyhrazeného prefixu `xml:`

# XML Inclusions

## XML Inclusions (XInclude)

- XML Inclusions (XInclude) Version 1.0 W3C Working Draft 10 November 2003, <http://www.w3.org/TR/xinclude/>
- XInclude umožňuje vkládání (částí) XML dokumentů do dokumentů.
- Je ortogonální k entitám (lze použít oboje v rámci jednoho dokumentu, "nevadí si").
- Nezávislé na DTD (zpracování XInclude probíhá až po validaci)
- Nezávislé na XML Schema

## XInclude: použití

- Specifikace definuje *jmenný prostor* a v něm jeden *element* `<xi:include>` s *atributy*:

- href= - vkládaný dokument
- parse= - hodnota je buď "text", pak se obsah vkládá jako (neparsovaný) text, nebo "xml", pak se hodnota vkládá jako značkový obsah
- encoding= - v případě encoding="text" specifikuje (je-li to nutné) kódování vkládaného textu
- a dalšími atributy (xpointer, accept, accept-charset, accept-language...) viz specifikace.
- Na FI je k dispozici interpret rozšířeného XInclude - xinclude-fi [http://www.fi.muni.cz/~tomp/xinclude-fi], který umí vkládat části textových souborů.

## XInclude: příklad

### Příklad 1.4. Vložení textového souboru (jako textového uzlu)

```
<xhtml:html xhtml:xmlns="http://www.w3.org/1999/xhtml"
           xml:lang="en" lang="en">
  <xhtml:body>
    <xhtml:h1>Huráááá</xhtml:h1>
    <xi:include xmlns:xi="http://www.w3.org/2001/XInclude"
              href="obsah.txt" encoding="Windows-1250"/>
  </xhtml:body>
</xhtml:html>
```

## XML Catalogs

### XML Catalogs

- Vycházejí ze starších SGML katalogů
- Jde o prostředek, jak se jednotně odkazovat na entity (dokumenty) umístěné na různých systémech na různých místech.
- Dovoluje také praktické použití identifikátorů URI typu PUBLIC, které neodkazují na žádnou reálnou lokaci na internetu.
- Existuje několik formátů pro katalogy - bohužel.

### XML Catalogs - příklad

### Příklad 1.5. Katalog pro styly značkování DocBook Slides

```
<?xml version="1.0"?>
<catalog xmlns="urn:oasis:names:tc:entity:xmlns:xml:catalog">
<!-- Slides DTD locations -->
<group xml:base="schema/dtd/"
      id="slides-dtd"
      prefer="public">
  <public
    publicId="-//Norman Walsh//DTD Slides Custom V3.1.0//EN"
    uri="slides-custom.dtd"/>

  <public
    publicId="-//Norman Walsh//DTD slides Full V3.1.0//EN"
    uri="slides-full.dtd"/>
</group>

  <rewriteURI
    uriStartString="http://docbook.sourceforge.net/release/xsl/current/"
    rewritePrefix="file:/c:/devel/docbook-xsl-1.62.4/">

<!-- Map web references to DocBook 4.2 DTD -->
  <nextCatalog catalog="file:/c:/devel/docbook4.2/catalog.xml" />
</catalog>
```

## XML Information Set

### XML Information Set (XML Infoset) - cíle

- *XML Infoset 2nd Edition W3C Recommendation* First published on 24 October 2001, revised 4 February 2004, John Cowan, Richard Tobin, <http://www.w3.org/TR/xml-infoset/>
- Infoset popisuje "jaké všechny informace lze o uzlu (elementu, dokumentu, atributu...) získat"
- Jinými slovy: aplikace by neměla spoléhat na informace z XML dokumentu, které se po analýze (parsingu) neobjeví v Infosetu.
- Každý správně utvořený XML dokument vyhovující standardu pro jmenné prostory má Infoset.

### XML Infoset - struktura

- Infoset se skládá z *Information items*

- Infoset se týká dokumentu s již expandovanými entitami
- Rozlišuje se infoset *dokumentu, elementu, atributu, znaku, instrukci pro zpracování, neexpandované entitě, neanalyzované entitě, notaci*
- Podrobněji viz specifikace.

## Kanonický tvar XML

### Kanonický tvar XML dokumentu

- Canonical XML Version 1.0, W3C Recommendation 15 March 2001, <http://www.w3.org/TR/xml-c14n>
- Smyslem je popsat kritéria (a algoritmy), které pomohou rozhodnout, zda jsou dva XML dokumenty ekvivalentní, lišící se pouze fyzickou reprezentací (entity, pořadí atributů, kódování znaků)
- Kanonizace "setře" rozdílly mezi takovými dokumenty, k nimž se analyzátor "bude jistě chovat stejně", tj. z pohledu aplikace jsou totožné.
- Použití kanonického tvaru je nutné např. u *elektronického podpisu XML dat* (při výpočtu hodnoty *digest*).
- Bylo by možné nad XML dokumenty definovat i jiné relace ekvivalence než je *Canonical XML*.

### Kanonický tvar - zásady konstrukce

Hlavní zásady konstrukce kanonického tvaru XML dokumentu:

- kódování v UTF-8
- zlomy řádků (CR, LF) jsou normalizovány podle algoritmu uvedeného v std. XML 1.0
- hodnoty atributů jsou normalizovány
- reference na znakové a parsované entity jsou nahrazeny jejich obsahem
- CDATA sekce jsou nahrazeny jejich obsahem
- hlavička "xml" a deklarace typu dokumentu jsou odstraněny
- bílé znaky mimo kořenový element jsou normalizovány
- jiné bílé znaky (vyjma normalizace zlomu řádků) jsou zachovány
- hodnoty atributů jsou uvozeny "

- speciální znaky v hodnotách atributů a textovém obsahu elementů jsou nahrazeny referencemi na entity
- nadbytečné deklarace jmenných prostorů jsou z každého elementu odstraněny
- implicitní hodnoty atributů jsou dodány do každého elementu (kde je to relevantní)
- na pořadí atributů a deklarací jmenných prostorů se uplatní lexikografické řazení

## Potíže při definici kanonického tvaru

Ztráta řady informací (typicky pocházejících z DTD):

- neparsované entity (např. binární entity) jsou po kanonizaci nepřístupné
- notace
- typy atributů (vč. implic. hodnot)