
Kapitola 1. Navigace v XML datech (XPath), Odkazování a adresace (XLink, XPointer)

Obsah

XML Linking Language (XLink)	6
XLink - úvod	6
XML Linking Language (XLink) - původ standardu	7
XLink - historie a motivace	7
Výhody odkazové infrastruktury na bázi XLink	7
Specifikace, tutoriály	7
XLink - základní principy	8
Integrace XLinku do (schémat) dokumentů	8
XLink - základní typy	8
<i>Simple</i> XLink odkaz	8
<i>Extended</i> XLink odkaz	8
Příklad odkazu XLink (1) - jednoduchý XLink	9
Příklad odkazu XLink (2) - rozšířený XLink	9
Sémantika odkazů XLink	9
Chování aplikací nad odkazy XLink	9
Upřesnění role odkazu pro aplikace	9
Upřesnění role odkazu pro člověka	10
Použití XLink a implementace procesorů XLink	10
Procesory XLink	10
Prohlížeče	10
Příčiny nízkého rozšíření	10
Alternativy k XLink	11
HLink	11
VELLUM	11
VELLUM - přednosti a nedostatky	12
VELLUM - ukázka	12
VELLUM - ukázka (2)	13
VELLUM - ukázka (3)	13
VELLUM - srovnání s RDF	13
Odkazy na další zdroje	13
XPointer	14
XML Pointer Language (XPointer)	14
Aktuální specifikace skupiny XPointer	14
XPointer - terminologie	15
pojem Point	15
pojem Range	15

pojem Location	15
XPointer - ukázky	15
XPointer - ukázka (1)	15
XPointer - ukázka (2)	16
Zvon XPointer Tutorial	16

XPath - aplikační oblasti

- Pokročilá navigace v XML datech
- Transformace (XSLT)
- V "selekční části" XML dotazovacích jazyků
- V některých modelovacích jazycích - Schematron

XPath - hlavní principy

- XPath je syntaxe pro specifikaci *částí* XML dokumentů (uzly, množiny uzlů, sekvence uzlů; nelze specifikovat *části* textových uzlů).
- XPath používá syntaxi obdobnou jako *cesty v souborovém systému*.
- XPath používá knihovnu standardních funkcí (evt. uživatelsky definovaných - v XPath 2.0 nebo i XPath 1.x, ale proprietárně - podle procesorů)
- XPath je od v 1.0 základem pro XSLT, od 2.0 i pro XQuery
- XPath syntaxe *není XML* (bylo by příliš "upovídáné")
- XPath 1.0 je W3C Recommendation - <http://www.w3.org/TR/xpath>

XPath - pojem cesty (paths) a lokace (locations)

Cesta určuje lokaci v dokumentu

Cesty jsou konstruovány podobně jako cesty ve FS, tj.

- relativní - vyhodnocovány vůči kontextovému uzlu (KU), viz dále
- absolutní - od kořene, ale výrazy (predikáty) také vyhodnocovány vůči KU

Syntaktická pravidla:

```
[20] PathExpr ::= AbsolutePathExpr | RelativePathExpr
```

```
[22] AbsolutePathExpr ::= ("/" RelativePathExpr?) | ("//" RelativePathExpr)
[23] RelativePathExpr ::= StepExpr ("/" | "//") StepExpr*
[24] StepExpr ::= AxisStep | GeneralStep
[25] AxisStep ::= (Axis? NodeTest StepQualifiers) | AbbreviatedStep
```

XPath - osy (axes)

Osy jsou množiny prvků dokumentu, vymezené (obvykle relativně) vůči *kontextu*.

Kontext je tvořen především *dokumentem* a *aktuálním (kontextovým) uzlem*.

Osami jsou:

- child - obsahuje dceřinné uzly kontextového (aktuálního) uzlu
- descendant - obsahuje všechny potomky kontextového (aktuálního) uzlu (dále jen KU). Nepočítají se mezi ně atributy!!!
- parent - obsahuje rodičovský uzel KU (existuje-li)
- ancestor - obsahuje všechny předky - rodiče, "prarodiče"... kořenový element (pokud KU není sám kořenový)
- following-sibling - obsahuje všechny následující sourozence KU (pro NS a atributy je tato osa prázdná)
- preceding-sibling - dtto, ale obsahuje *předchozí* sourozence
- following - obsahuje všechny uzly nacházející se *po* KU (mimo atributů, potomků a NS uzlů)
- preceding - dtto, ale obsahuje *předchozí* uzly (ale mimo předky, atributy, NS!)
- attribute - obsahuje atributy (jen pro uzly - elementy)
- namespace - obsahuje všechny NS uzly KU (jen pro uzly - elementy)
- self - obsahuje samotný KU
- descendant-or-self - obsahuje sjednocení os descendant a self
- ancestor-or-self - obsahuje sjednocení os ancestor a self

XPath - predikáty (predicates)

Určeny k selekci (výběru) z uzlů specifikovaných např. cestou

př.: /article/para[3] - vybere třetí odstavec v článku

Nejjednodušším výrazem v predikátu je specifikace *pozice (blízkosti)* (proximity position) - viz výše

- pozor u reverzních os (ancestor, preceding...) - pozice se počítá v rámci množiny uzlů vždy OD KONTEXTOVÉHO UZLU!
- specifikaci pozice 3 možno nahradit výrazem position()=3

XPath - Výrazy

Určeny k použití v predikátech, k výpočtům, atd.

Výrazy mohou být:

- řetězcové
- numerické (hodnotami jsou floating-point čísla)
- logické (boolean)
- uzly
- sekvence

XPath - zkrácená notace - Příklady

- para selects the para element children of the context node
vybere všechny dceřinné elementy kontextového uzlu jmenující se para
- * selects all element children of the context node
- text() selects all text node children of the context node
- @name selects the name attribute of the context node
- @* selects all the attributes of the context node
- para[1] selects the first para child of the context node
- para[last()] selects the last para child of the context node
- */para selects all para grandchildren of the context node
- /doc/chapter[5]/section[2] selects the second section of the fifth chapter of the doc
- chapter//para selects the para element descendants of the chapter element children of the context node
vybere všechny element para, jež jsou následníky chapter

- //para selects all the para descendants of the document root and thus selects all para elements in the same document as the context node
vybere všechny elementy para z dokumentu
- //olist/item selects all the item elements in the same document as the context node that have an
vybere všechny elementy item, které mají za rodiče olist
- . selects the context node
vybere kontextový uzel
- //para selects the para element descendants of the context node
vybere všechny elementy-potomky kontextového uzlu, které nesou značku para
- .. selects the parent of the context node
vybere rodičovský uzel od kontextového
- ../@lang selects the lang attribute of the parent of the context node
vybere atribut lang rodičovského uzlu od kontextového

XPath - zkrácená notace (2)

Nejpoužívanější zkracování je *osy child* :

- tj. píšeme article/para místo child::article/child::para.
- a *atributu*: píšeme para[@type="warning"] místo child::para[attribute::type="warning"]
- Další používané zkracování je // místo /descendant-or-self::node()/
- a samozřejmě zkratky . a ..
- Pro přehlednost někdy delší formu zachováváme

Informační zdroje k XPath

- XPath na W3C: <http://www.w3.org/TR/xpath>
- Zvon XPath Tutorial: <http://zvon.org/xxl/XPathTutorial/Output/index.html>
- XPath Tutorial na W3Schools: http://www.w3schools.com/xpath/xpath_intro.asp

XPath 2.0

- zatím jako PRACOVNÍ NÁVRH - <http://www.w3.org/TR/xpath20/>
- Změna pohledu na hodnoty vrácené XPath výrazem: vše jsou **sekvence** (byť jednoprvkové)
- ->odstraňuje problémy s "pořadím" uzlů v množině
- Zavádí **podmíněné výrazy a cykly**
- Zavádí možnost uživatelských funkcí (psaných jako dynamicky vyhodnocované výrazy v XPath)
- Lze použít existenční a obecné kvantifikátory, např. `exist student/name="Fred"` nebo `all student/@id`
- Dále viz např. <http://www.saxonica.com/>

XPath 2.0 - příklady

- Řetězcové funkce [<http://www.fi.muni.cz/~tomp/xml03/xpath20/string.html>]
- Numerické funkce [<http://www.fi.muni.cz/~tomp/xml03/xpath20/numeric.html>]
- Funkce nad sekvencemi [<http://www.fi.muni.cz/~tomp/xml03/xpath20/sequence.html>]
- Booleovské funkce [<http://www.fi.muni.cz/~tomp/xml03/xpath20/boolean.html>]

XML Linking Language (XLink)

XLink - úvod

XLink pracuje na úrovni XML Infoset

Dovoluje odkazovat se:

- v rámci dokumentu,
- z jednoho dokumentu na další,
- mezi dokumenty (odkaz je uložen mimo ně)

Rozlišuje kategorie:

- **adresa** (též *zdroj* nebo *lokace*, tj. odkazovaný objekt - element, skupina elementů, text...)
- a vlastní **odkaz**

- Odkaz pak představuje vazbu mezi zdroji s případným upřesněním sémantiky odkazu.

XML Linking Language (XLink) - původ standardu

- W3C Recommendation 27 July 2001
- Obecný mechanismus na propojování XML zdrojů (=dokumentů a jejich částí)
- Inspirován zejména std. **HyTime** (<http://www.hytime.org/>)
- Ortoagonální k entitám (lze použít současně oboje)
- mnohem bohatší sémantika než u entit,
- entity se vyhodnocují (resolve) při parsingu,
- XLink odkazy většinou až v aplikaci

XLink - historie a motivace

- Vychází z pokročilých hypertextových technik - HyTime (pro SGML - viz <http://info.admin.kth.se/SGML/Anvardarforening/Arbetsgrupper/HyTime/Reports/tr1v1.html>), TEI...
- Konstruován tak, že rozšiřuje a upřesňuje syntaxi i sémantiku HTML odkazů

Výhody odkazové infrastruktury na bázi XLink

- Na rozdíl od SGML linků (např. v HyTime) můžeme v XML využívat *jmenné prostory*. Nenutí nás to upravovat DTD při použití odkazů.
- Infrastruktura odkazů může existovat (v jiném NS) nezávisle na schématu (struktuře) XML dokumentů, v nichž se odkazy vyskytují.
- Odkazy mohou být fyzicky mimo soubory s odkazovanými lokacemi.

Specifikace, tutoriály

- XML Linking Language (XLink) Version 1.0 - Specification [<http://www.w3.org/TR/xlink/>]
- Zvon simple XLink tutorial [http://www.zvon.org/xxl/xlink/OutputExamples/xlinksimple_intro.html]
- Zvon extended XLink tutorial

[http://www.zvon.org/xxl/xlink/xlink_extend/OutputExamples/xlinkextend_intro.html]

XLink - základní principy

Integrace XLinku do (schémat) dokumentů

- Atributy a elementy XLinku mají vlastní jmenný prostor
- Jako prefix NS se obvykle používá `xlink`
- Příslušnost k XLinku je dána deklarací daného elementu/atributu v XLink NS
- to umožňuje flexibilní *integraci stávajících schémat (modelů) a XLinku* (nemusíme měnit jména "linkovacích" elementů)
- XLink odkaz "dovnitř" dokumentů používají standard XPointer (<http://www.w3.org/TR/xptr>).

XLink - základní typy

- Jednoduchý (`simple`) - vždy "*in-line*"
- Rozšířený (`extended`) - může být "*out-of-line*" (i v jiném dokumentu/souboru/databázi odkazů)

Simple XLink odkaz

- Váže jeden lokální zdroj na jeden vzdálený (*single local -> single remote entity*);
- Je vyznačen atributem `xlink:type="simple"`;

Extended XLink odkaz

- Váže **jeden nebo více lokálních** a **jeden nebo více vzdálených** zdrojů
- Může mít přesněji definovanou sémantiku (nad rámec výše uvedených možností)
- Odkazy mohou být uloženy mimo odkazující zdroj(e)
- je vyznačen atributem `xlink:type="extended"`

extended odkaz se může odkazovat na

- lokální zdroje - vyznačené atributem `xlink:resource` nebo
- vzdálené zdroje - vyznačené atributem `xlink:locator`

Směr odkazu a role participujících zdrojů mohou být upřesněny (nad rámec výše uvedených možností):

- uvedením podelementů `arc` v `extended` odkazu

Příklad odkazu XLink (1) - jednoduchý XLink

```
<zvon:logo xmlns:zvon = "http://www.zvon.org"
           xmlns:xlink = "http://www.w3.org/1999/xlink"
           xlink:type="simple" xlink:href="zvon.gif" />
```

Příklad odkazu XLink (2) - rozšířený XLink

```
<?xml version="1.0"?>
<zvon_tutorial xmlns:xlink="http://www.w3.org/1999/xlink">
  <extendedlink xlink:type="extended"> Any content here </extendedlink>
</zvon_tutorial>
```

Sémantika odkazů XLink

Chování aplikací nad odkazy XLink

Co má aplikace udělat s odkazovaným dokumentem/prvkem dokumentu je specifikováno atributem `xlink:show`

- `new` - chápat jako nový dokument (např. otevřít v novém okně)
- `replace` - nahradit jím zdrojový dokument
- `embed` - vložit odkazovaný obsah do zdrojového dokumentu

"kdy to má udělat" - specifikováno atributem `xlink:actuate`

- `onLoad` - tj. ihned při zpracování zdrojového dokumentu (ale kdy? při parsingu?)
- `onRequest` - na požádání (např. po kliknutí na zobrazený obsah odkazujícího elementu)

Upřesnění role odkazu pro aplikace

roli odkazu lze upřesnit v podobě *srozumitelné aplikací*:

- atributem `xlink:role` - u `simple` i `extended` odkazu
- atributem `xlink:arcrole` - u elementu `arc` v `extended` odkazu

Upřesnění role odkazu pro člověka

Lze upřesnit **roli odkazu** v podobě *srozumitelné člověku*:

- atributem `xlink:title` - u `simple` i `extended` odkazu i u elementu `arc` v `extended` odkazu

Použití XLink a implementace procesorů XLink

Procesory XLink

Zatím nepříliš rozšířené (řádově jednotky aplikací). Nejznámějšími procesory jsou:

- Fujitsu XLink Processor (XLiP) - <http://www.labs.fujitsu.com/free/xlip/en/>
- Open source projekt `xlinkit` - <http://www.xlinkit.com/xtoox/index.html>
- X2X (komerční, velmi drahý produkt) podporuje XLink - http://www.stepuk.com/products/prod_X2X.asp

Prohlížeče

podpora *simple* odkazů

- Mozilla (open source) a Netscape (komerční)
- DocZilla (komerční)
- MSIE ani Opera (vč. posledních verzí) XLink *nepodporují*

Příčiny nízkého rozšíření

- Vývoj trval dlouho (cca 5 let), mezitím aplikace začaly používat vlastní řešení
- Sémantika XLinku je:

- **příliš obecná** (aplikace raději používaly vlastní, na míru šitý odkazovací jazyk, viz (X)HTML) a současně
- **málo obecná** (např. RDF metadata popíší vztahy mezi zdroji podrobněji, s typovou kontrolou přes RDF Schemata, atd.)

XLink je také nahrazován aktivitou směřující ke specifikaci *XML Topic Maps*

Alternativy k XLink

HLink

- Cílem je adaptovat XLink pro potřeby XHTML, kde "nelze použít přímo XLink"
- viz HLink - Link recognition for the XHTML Family - W3C Working Draft 13 September 2002 [<http://www.w3.org/TR/hlink/>]
- specifikace je navržena *W3C Working Group*, jí oponovala *W3C Technical Architecture Group (TAG)*
- Princip: mapuje vybrané atributy na jejich přesně dané XLink protějšky
- tzn. i to, co normálně nemůže být XLinkem, se na XLink může "automaticky" mapovat
- Tento standard je kritizován jako "hybrid, který jedinež dokazuje praktickou nevhodnost XLinku"
- dále viz např. diskuse na xmlhack.com [<http://xmlhack.com>]

VELLUM

- Nezávislá alternativa Simona St. Laurenta
- viz Very Extensible Linking Language Unafraid of Markup (VELLUM) [<http://www.simonstl.com/projects/vellum/>]
- Nevýhodou je poněkud rozvleklejší ("upovídaný", *verbose*) zápis, než u XLink
- VELLUM is aimed exclusively at "external" or out-of-line linking. This frees VELLUM from the constraints imposed by developer expectations for inline linking.
- VELLUM accepts the cost of a verbose form in exchange for the extensibility and precision that element forms and indirection can offer.
- VELLUM does permit the use of existing abbreviations (like URIs and URI references), but simultaneously allows developers to specify more information than is carried in those identifiers.

- VELLUM also looks forward to the prospect of VELLUM processors which maintain state across multiple traversals and permit the creation of interactive hypertexts, not just collections of connections.

VELLUM - přednosti a nedostatky

- VELLUM attempts to provide a *general-purpose solution* to linking which addresses the complexities raised by the W3C's XPointer and XLink specifications by taking a very different approach. VELLUM *does not assume that URIs and URI references are adequate to the task of identifying resources*, representations, and fragments of representations, and strives to put XML hyperlinking on firmer but still approachable foundations. VELLUM supports and uses URIs and URI references, but offers *options that extend those capabilities*.
- VELLUM is *not a general-purpose solution to hypertext linking*. VELLUM is intended to be used in cases where *precision is important and verbosity is not a problem*. While VELLUM could conceivably be mixed with other vocabularies and used to define links within them, it is not designed explicitly for such use. VELLUM is more appropriate for use in cases like *external links and linkbases*, where the links are stored separately from the resources they connect. (VELLUM's designer hopes that a simpler mechanism for in-line linking will emerge to complement VELLUM.)
- VELLUM both builds on the *URI framework and goes beyond* the URI framework. URIs and URI references may be used within the VELLUM framework if the level of precision they provide is adequate, but developers can specify more information about issues like *content-negotiation* within the VELLUM framework if they choose. VELLUM also makes it possible to explicitly specify whether a connection *involves an abstract resource or a particular concrete representation*.
- VELLUM also makes it possible for developers to *create metadata which applies to their links in a local context*. While XLink uses URIs for everything from href to arcrole, VELLUM lets developers use more intelligible identifiers whose meaning is defined within a particular VELLUM context. While VELLUM may be more verbose than a comparable XLink linkbase, it should (if designed thoughtfully) be more readable.

VELLUM - ukázka

```
<piece xmlns="http://simonstl.com/ns/vellum" >
  <connections>
    <traverse>
      <from href="http://www.w3.org/TR/REC-xml#sec-common-syn" />
      <to href="http://www.w3.org/TR/REC-xml-names/#ns-qualnames" />
    </traverse>
    <traverse>
      <from href="http://www.w3.org/TR/REC-xml-names/#ns-qualnames" />
      <to href="http://www.w3.org/TR/REC-xml#sec-common-syn" />
    </traverse>
  </connections>
</piece>
```

VELLUM - ukázka (2)

```
<piece xmlns="http://simonstl.com/ns/vellum">
  <connections>
    <set id="namespaceUses" >
      <member href="http://www.w3.org/TR/REC-xml#sec-common-syn" />
      <member href="http://www.w3.org/TR/xmlschema-2/#QName" />
    </set>
    <traverse>
      <from ref="namespaceUses"/>
      <to href="http://www.w3.org/TR/REC-xml-names/#ns-qualnames" />
    </traverse>
  </connections>
</piece>
```

VELLUM - ukázka (3)

```
<piece xmlns="http://simonstl.com/ns/vellum">
  <targets>
    <target id="_xml-names" representation="http://www.w3.org/TR/REC-xml#sec-co
    <target id="_xml-schema-qnames" representation="http://www.w3.org/TR/xmlsch
    <target id="_xmlns-qual-names" representation="http://www.w3.org/TR/REC-xml
  </targets>
  <connections>
    <set id="namespaceUses" >
      <member ref="_xml-names" />
      <member ref="_xml-schema-qnames" />
    </set>
    <traverse>
      <from ref="namespaceUses" />
      <to ref="_xmlns-qual-names" />
    </traverse>
  </connections>
</piece>
```

VELLUM - srovnání s RDF

koncept VELLUM out-of-line odkazů je podobný metadatům podle rámce RDF (odkazy jsou také meta-data...)

proto se k VELLUM vrátíme po prostudování RDF

Odkazy na další zdroje

- XLink Tutoriál na Zvonu (<http://zvon.org>)
- Specifikace XLink (<http://www.w3.org/TR/xlink>)
- IBM Developerworks/XML
- Portál XML.COM (např. článek *What is XLink?* <http://www.xml.com/lpt/a/2000/09/xlink/index.html>)
- xmlhack.com [<http://xmlhack.com>]

XPointer

XML Pointer Language (XPointer)

W3C Working Draft 16 August 2002

- Definiuje mechanismus **adresování** (tj. "pointing" - ne odkazování!) **v rámci XML dokumentu**
- Rozšiřuje koncept *kotvy* (anchor) známé z HTML - *#jménoKotvy*
- Umožňuje **explicitní adresování navigací** i **implicitní adresování dotazováním**
- Umožňuje adresování i v rámci textových uzlů
- Umožňuje rozlišit adresu **bodu** (point) a **úseku** (range) v XML dokumentu
- Jako základ pro specifikaci adresy bere jazyk XPath

Aktuální specifikace skupiny XPointer

POZOR - specifikace XPointer se v současnosti rozpadla na 4 následovníky:

- *XPointer Framework* - This specification defines the XML Pointer Language (XPointer) Framework, an extensible system for XML addressing that underlies additional XPointer scheme specifications. The framework is intended to be used as a basis for fragment identifiers for any resource whose Internet media type is one of text/xml, application/xml, text/xml-external-parsed-entity, or application/xml-external-parsed-entity. Other XML-based media types are also encouraged to use this framework in defining their own fragment identifier languages.
- *XPointer element() Scheme* - The XPointer `element()` scheme is intended to be used with the *XPointer Framework* to allow basic addressing of XML elements.
- *XPointer xmlns() Scheme* - The XPointer `xmlns()` scheme is intended to be used with the *XPointer Framework* to allow correct interpretation of namespace prefixes in pointers, for instance, namespace-qualified scheme names and namespace-qualified element or attribute names appearing within scheme data.

- *XPointer xpointer() Scheme* - The XPointer xpointer() scheme is intended to be used with the *XPointer Framework* to provide a high level of functionality for addressing portions of XML documents. It is based on XPath, and adds the ability to address strings, points, and ranges in accordance with definitions provided in DOM 2: Range.

XPointer - terminologie

XPointer - terminologie

pojem Point

point - bod, daný svým kontejnerem (container) a celočíselným indexem (pozicí) v rámci kontejneru

bod může mít *explicitní* nebo *implicitní* hranici

může být typu node-point nebo character-point

dále viz specifikace - typ Point [<http://www.w3.org/TR/xptr-xpointer/#dt-point>]

viz také tutoriál node-point [http://zvon.org/xxl/xpointer/tutorial/OutputExamples/xml30_out.xml.html]

a character-point [http://zvon.org/xxl/xpointer/tutorial/OutputExamples/xml40_out.xml.html]

pojem Range

range - je obsah mezi dvěma body V RÁMCI jednoho dokumentu nebo jedné externí analyzované entity

dále viz specifikace - typ Range [<http://www.w3.org/TR/xptr-xpointer/#dt-range>]

pojem Location

location - "místo" - může být DOM *uzel*, point, range

dále viz specifikace - typ Location [<http://www.w3.org/TR/xptr-xpointer/#dt-location>]

XPointer - ukázky

XPointer - ukázka (1)

```
<link xmlns:xlink="http://www.w3.org/2000/xlink"
      xlink:type="simple"
      xlink:href="mydocument.xml#xpointer(//AAA/BBB[1])" />
```

XPointer - ukázka (2)

Xpointer výraz `b2/3` nebo `xpointer(id('b2')/*[3])` vybere z následujícího dokumentu

```
<AAA>
  <BBB myid="b1" bbb="111"> Text in the first element BBB.</BBB>
  <BBB myid="b2" bbb="222">
    Text in another element BBB.
    <DDD ddd="999"> Text in more nested element.</DDD>
    <DDD ddd="888"> Text in more nested element.</DDD>
    <DDD ddd="777"> Text in more nested element.</DDD>
  </BBB>
  <CCC ccc="123" xxx="321"> Again some text in some element.</CCC>
</AAA>
```

...třetí uzel `DDD ddd="777"`...

Zvon XPointer Tutorial

Zvon XPointer Tutorial - http://www.zvon.org/xxl/xpointer/tutorial/OutputExamples/xpointer_tut.html