

XML 1.1 és névterületek

dr. Paller Gábor

XML 1.1

- Az XML 1.1-gyel párhuzamosan létező specifikáció, amely maga is a 2. kiadásnál tart. <http://www.w3.org/TR/2006/REC-xml11-20060816/>
- Az XML-t csak apró részletekben változtatja meg.
- Nevek korlátozásainak lazítása.
- További sorvégjelek bevezetése, melyeket egzotikusabb rendszerekben használnak (IBM mainframe: #x85, Unicode line separator: #x2028).
- Vezérlőkarakterek (#x1-#x1F) engedélyezése XML dokumentumokban karakter referenciákon keresztül (direktben továbbra is tiltva vannak). Ezen felül a #x7F-#x9F tartomány, amelyik az XML 1.0-ban direktben is megengedett volt, az XML 1.1-ben csak karakterreferenciákon keresztül elérhető.

XML 1.1: nevek korlátozásainak lazítása

- "Name" szintaktikus elem az XML specifikációkban: általánosan használt azonosító-token, pl. tag-ek, attribútumok nevei, ID típusú attribútumok értéke, stb.
- "Name" szintaktikus elem az XML 1.0-ban (CombiningChar: kombinálódó ékezet, ami hozzacsatolódik az előtte levő karakterhez, Extender: pl. ismétlésjelző japánban):

```
NameChar ::= Letter | Digit | '.' | '-' | '_' | ':' |  
CombiningChar | Extender  
Name ::= (Letter | '_' | ':') (NameChar)*
```

- "Name" szintaktikus elem az XML 1.1-ben:

```
NameStartChar ::= ":" | [A-Z] | "_" | [a-z] | [#xC0-  
#xD6] | [#xD8-#xF6] | [#xF8-#x2FF] | [#x370-#x37D] |  
[#x37F-#x1FFF] | [#x200C-#x200D] | [#x2070-#x218F] |  
[#x2C00-#x2FEF] | [#x3001-#xD7FF] | [#xF900-#xFDCF] |  
[#xFDF0-#xFFFD] | [#x10000-#xEFFFFF]
```

```
NameChar ::= NameStartChar | "-" | "." | [0-9] | #xB7 |  
[#x0300-#x036F] | [#x203F-#x2040]
```

```
Name ::= NameStartChar (NameChar)*
```

Namespaces

- Amint az XML dokumentumok nőnek, fellép az elemkönyvtárak modularizációjának kérdése.
- Például lehet két dokumentumtípusunk, amelyben a "book" elemet másként definiáltuk. Ha egy olyan dokumentumot akarunk létrehozni, amelyben mindkét dokumentumtípus "book" eleme szerepel, bajba kerülünk, mert a két "book" definíciója összeütközik.
- Megoldás: az XML nevekhez csomagneveket rendelünk. A csomagnév és az XML név együttesen kell egyedi legyen.
- A csomagnevet XML terminológiával *névterületnek (namespace)* hívják.
- Specifikáció: <http://www.w3.org/TR/2006/REC-xml-names-20060816/> (1.0) és <http://www.w3.org/TR/2006/REC-xml-names11-20060816/> (1.1)

Namespaces (2)

- A névterületet egy URI azonosítja. Az XML elemző számára az URI csupán egy karaktersorozat, tehát nem feltétlenül kell létező cím legyen.
- Az URI túl hosszú ahhoz, hogy XML nevekben fordulhasson elő és illegális karaktereket is tartalmazhat, ezért minden XML dokumentumban helyi névterület-azonosítóra képezzük le.
- Az URI világszinten egyedi azonosító, azonban az XML dokumentumban alkalmazott helyi névterület-azonosító csak az adott dokumentumban egyedi, viszont rövid, tehát XML nevekhez (pl. tag-név) hozzáadható.
- Példa:
névterület-azonosító: <http://www.server.org/books>
helyi névterület azonosító: books
tag-név helyi névterület-azonosítóval: <books:published>

Kvalifikált név

- Új fogalom: *kvalifikált név (qualified name)*. Az elem kvalifikált neve teljesen meghatározott, tehát tartalmazza a névterület-meghatározást (ha van neki).
- ```
QName ::= PrefixName | UnprefixedName
PrefixName ::= Prefix ':' LocalPart
UnprefixedName ::= LocalPart
Prefix ::= NCName
LocalPart ::= NCName
/* An XML Name, minus the ":" */
NCName ::= NCNameStartChar NCNameChar*
NCNameChar ::= NameChar - ':'
NCNameStartChar ::= Letter | '_'
```
- A kvalifikált név névterület-meghatározása a helyi névterület-azonosítót tartalmazza, de ennek kötvé kell lennie globális névterület-azonosítóhoz (URI). Valójában tehát a kvalifikált név "Prefix" része a globális URI, bár a dokumentum szintaxisa a helyi névre utal. Vannak elemzők, amelyeknek belső QName reprezentációja "Prefix" alatt a globális névterület-azonosítót adja vissza. Az (URI,LocalPart) párost a specifikáció *bővített névnek (expanded name)* hívja.

# URI és helyi azonosító összerendelése

- Használat előtt a helyi azonosítót és az URI-t össze kell rendelni. Az összerendelés mindig egy elem *hatáskörében* (*scope*) történik meg és vonatkozik az összerendelő elemre valamint azok gyerekeire.
- Példa: a legfelső "html" tag-ben hozzárendeljük a "html" helyi névmező-azonosítóhoz a <http://www.w3.org/1999/xhtml> globális névmező-azonosítót, majd rögtön használni is kezdjük a "html", "head" és "title" elemeknél.

```
<?xml version="1.0"?>
```

```
<html:html xmlns:html="http://www.w3.org/1999/xhtml">
```

```
 <html:head>
```

```
 <html:title>Frobnostication</html:title>
```

```
 </html:head>
```

```
</html:html>
```

# Alapértelmezett névterület

- Az alapértelmezett névterület beállítható és a deklaráció hatáskörében a Prefix nélküli elemekre vonatkozik.
- Példa: a "html" tag gyerekei (és maga a "html" tag is) a <http://www.w3.org/1999/xhtml> névterülethez tartozik, pedig erről a tag-ek egyenként nem rendelkeznek.

```
<?xml version="1.0"?>
```

```
<!-- elements are in the HTML namespace, in this case
by default -->
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
 <head><title>Frobnostication</title></head>
```

```
</html>
```



# Több névterület egyidejű használata

- Egyszerre több névterület-deklaráció lehet aktív
  - Egy tag egyszerre több névterületet deklarálhat
  - A szülőelemek névterület-meghatározása öröklődik.
- Egy tag egyszerre csak egy alapértelmezett névterületet deklarálhat.
- Példa: az "urn:loc.gov:books" elemei prefix nélkül használhatók, az "urn:ISBN:0-395-36341-6" elemeihez viszont kell az "isbn" prefix.

```
<?xml version="1.0"?>
<!-- unprefixed element types are from "books" -->
<book xmlns="urn:loc.gov:books"
 xmlns:isbn="urn:ISBN:0-395-36341-6">
 <title>Cheaper by the Dozen</title>
 <isbn:number>1568491379</isbn:number>
</book>
```

# Helyi névterület-azonosító felülírása

- Az XML 1.1 névterület-specifikáció rendelkezik a helyi névterület-azonosító felülírásának esetéről. E szerint a gyerekelemben a szülő által definiált névterület felülírható és a gyerekelem hatáskörében az új azonosító érvényes. Az XML 1.0 névterület-specifikációjából ezek a részek hiányoznak (felülírás esetéről a specifikáció nem rendelkezik), tehát XML 1.0-ban ez a gyakorlat rizikós (népszerű Xerces parser például megengedi).
- ```
<?xml version="1.0"?>
<x xmlns:n1="http://www.w3.org">
  <n1:a/>
  <x xmlns:n1="http://www.w4.org">
    <n1:a/>
  </x>
</x>
```

Névterület-azonosító törlése

- **Csak XML 1.1:** lehetőség: egy előzőleg deklarált prefixet egy gyerekelemben törölni lehet. Amennyiben a prefixhez rendelt URI üres string, a helyi névterület-azonosító érvénytelenné válik az elem hatáskörében.
- Példa: a második "n1:a" hivatkozás érvénytelen, mert a szülő "x" tag törölte az "n1" névterület deklarációját.

```
<?xml version="1.1"?>
<x xmlns:n1="http://www.w3.org">
  <!-- legal; the prefix n1 is bound -->
    <n1:a/>
    <x xmlns:n1="">
  <!-- illegal; the prefix n1 is not bound here -->
    <n1:a/>
    <x xmlns:n1="http://www.w4.org">
  <!-- legal; the prefix n1 is bound again -->
    <n1:a/>
    </x>
  </x>
</x>
```

Attribútumok és névterületek

- Az attribútumok is tartozhatnak névterülethez.
- Az attribútumokra azonban nem vonatkozik az alapértelmezett névterület. Az attribútumot tartalmazó elem szabadon dönthet, mi szerint értelmezi a prefix nélküli attribútumokat.
- Példa: a "taxClass" attribútum a lineItem elemben az "edi" névterületből jön.

```
<x xmlns:edi='http://ecommerce.example.org/schema'>  
<!-- the 'taxClass' attribute comes from the "edi"  
namespace -->  
  <lineItem edi:taxClass="exempt">Baby food</lineItem>  
</x>
```

Névterületek és DTD

- A DTD semmilyen módon nem ad plusz támogatást a névterületekhez.
- Minthogy a QName-ek érvényes XML Name-ek egyben, a DTD definiálhatja a névterületeket teljes formájukban és #FIXED eszközzel fixálhatja a hozzárendelésüket. Ez kétségkívül egy béna megoldás, mivel a DTD-k éppen emiatt nem modularizálhatók (hozzá kell nyúlni a DTD-khez, ha egy dokumentumba akarjuk foglalni az egyes DTD-k által leírt elemeket és explicit már a DTD-ben kötni kell a globális és a helyi névterület-azonosítókat).
- Konklúzió: a névterületek túlmutatnak a DTD-n.
- Példa: a "checkboxook" elem egy másik névterületből származó "deposit" elemet használ.

```
<!ELEMENT checkbox ( f:deposit | payment )*>
<!ATTLIST checkbox
  xmlns:f CDATA #FIXED "http://schemas.arent.net/soap/file/">
<!ELEMENT f:deposit (payor, amount, date,
description?)>
<!ATTLIST f:deposit
  type (cash|check|direct-deposit|transfer) #REQUIRED>
```

Gyakorlat

- A korábbi gyakorlatban bemutatott "books" adatbázist egy könyvesbolt akarja felhasználni készletlistának. A bolt azonban CD-eket is árul.
- Egy CD-nek címe van, kiadója és kiadási éve.
- Tervezze meg a CD-adatbázis formátumát és a hozzá tartozó DTD-t!
- Ez után hozza létre a CD-k és könyvek közös adatbázisát úgy, hogy a CD-khez és könyvekhez tartozó elemek külön névterületbe kerüljenek, majd tervezzen ezt validáló közös DTD-t!

cds.xml

- ```
<?xml version="1.0"?>
<!DOCTYPE cds SYSTEM "cds.dtd">
<cds>
 <cd id="id1"
 title="Best of"
 published="1999"
 publisher="Best of Music Inc." />
 <cd id="id12"
 title="Even Better of"
 published="2000"
 publisher="Best of Music Inc." />
</cds>
```

# cds.dtd

- ```
<!ELEMENT cds (cd)*>
<!ELEMENT cd EMPTY>
<!ATTLIST cd
    id ID #REQUIRED
    title CDATA #REQUIRED
    published CDATA #REQUIRED
    publisher CDATA #REQUIRED>
```


stock.xml

- ```
<?xml version="1.0"?>
<!DOCTYPE books SYSTEM "stock.dtd">
<stock xmlns:cd=http://www.products.org/cd
 xmlns:book="http://www.products.org/book">
 <book:book id="id1">
 <book:title>My first XML book</book:title>
 <book:published>2001</book:published>
 <book:publisher><XML> Publishing,
Inc.</book:publisher>
 <book:author>John Doe</book:author>
 <book:author>Richard Doe</book:author>
 <book:abstract>XML for dummies</book:abstract>
 </book:book>
 <cd:cd id="id12"
 title="Even Better of"
 published="2000"
 publisher="Best of Music Inc."/>
</stock>
```

# stock.dtd

- ```
<!ELEMENT stock (book:book | cd:cd)*>
<!ATTLIST stock xmlns:book CDATA #FIXED
  "http://www.products.org/book"
                xmlns:cd CDATA #FIXED
  "http://www.products.org/cd">
<!ELEMENT book:book
(title,published,publisher,author+,abstract?)>
<!ATTLIST book:book
  id ID #REQUIRED>
<!ELEMENT book:title (#PCDATA)>
<!ELEMENT book:published (#PCDATA)>
<!ELEMENT book:publisher (#PCDATA)>
<!ELEMENT book:author (#PCDATA)>
<!ELEMENT book:abstract (#PCDATA)>
<!ELEMENT cd:cd EMPTY>
<!ATTLIST cd:cd
  id ID #REQUIRED
  title CDATA #REQUIRED
  published CDATA #REQUIRED
  publisher CDATA #REQUIRED>
```