

XPath szabvány

XPath szabvány

Az előadás anyaga

Prof. Dr. Kovács László: Adatkezelés XML környezetbe és
Kollár Lajos, Sterbinszky Nóra: Programozási technológiák
jegyzete és további irodalom alapján készült el

Témakör kérdései

XPath szabvány - history

Mi az XPath? - services

XPath - expression

XPath szabvány - nodes

XPath – relationship of nodes

XPath – predicates

Sample tasks

Igényelt kompetenciák

- XPath feldolgozási modell megismerése.
- Az XPath kifejezések.
- XPath csomópontjainak megismerése.
- XPath program felépítésének elsajátítása.
- Környezet: XML szerkesztő (Eclipse, Oxygen, JDeveloper, EditIX,)

XML adatok kezelési lehetőségei

Magával az XML-lel kapcsolatos specifikációk

- Az XML lehetőségeit bővítik.
- Lehetővé teszik XML dokumentumok *szerkezetére és tartalmára vonatkozó megszorítások kifejezését (XML sémanyelvek)*.
- Lehetővé teszik XML dokumentumokból *információ kinyerését (lekérdező nyelvek)*.
- Lehetővé teszi XML dokumentumok *más formába alakítását (transzformációs nyelvek)*.

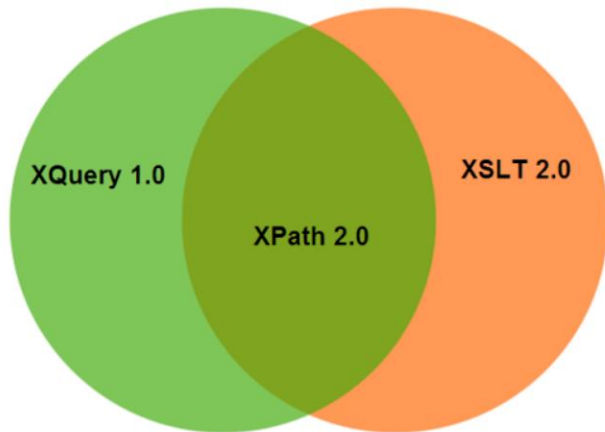
Mi az XPath?

„Az **XPath** (*XML Path Language*) egy lekérdező nyelv - csomópontok kiválasztásához az XML dokumentumból.

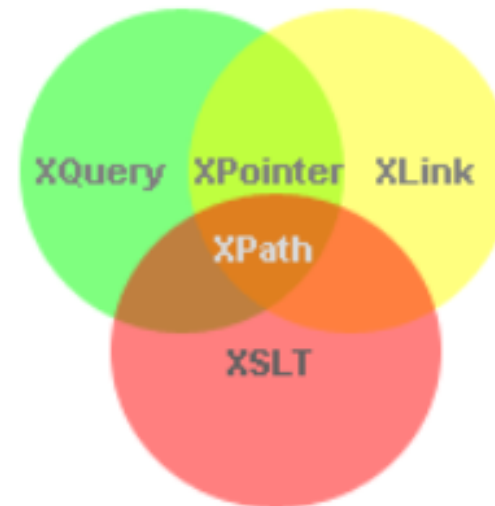
- Az XPath használható pl.: értékek kiszámításához (pl.: sztringek, számok, vagy logikai értékek) XML dokumentum tartalmából.
- Az eredeti elképzelés az volt, hogy *közös szintaxist és viselkedési* modellt teremtsenek az *XPointer* és az *XSLT* között (*Xpointer: a hivatkozások egy XML-dokumentum egy meghatározott részeire mutat*)

XPath nyelv helye parse-k között

- Az *XPath* különböző típusú *kifejezéseket* használ az információk lekérésére az XML dokumentumból.
- Az *XPath szintaxisa* egy XML dokumentum egyes részeinek meghatározásához.



Forrás: <https://www.javatpoint.com/>



https://www.w3schools.com/xml/xpath_intro.asp

XPath szabvány története - version 1.0

Az XPath-nak (*XML Path Language*) három verziója van jelenleg használatban.

- *XPath 1.0* - 1999. novemberben jelent meg.
- Ekkor kezdték el *széles körben implementálni* és használni:
 - *részben ún. API-n keresztül* nyelvekből ilyen pl.: Java, C#, Python vagy JavaScript,
 - részben *beágyazva nyelvekbe*, úgymint XSLT, XML séma.

[URL: https://www.w3.org/TR/xpath/](https://www.w3.org/TR/xpath/)

XPath szabvány története - version 1.0

Cél: lehetővé tegye XML dokumentumok *részeinek kijelölését*.

- Ehhez biztosítja *karakterláncok, számok és logikai értékek* kezelését is.
- Az *XSLT* és az *XPointer* szabványok használják (mindkettő kibővíti).

XPath szabvány története – version 2.0

XPath 2.0 - a változata a nyelvnek:

[URL: https://www.w3.org/TR/xpath20/](https://www.w3.org/TR/xpath20/)

2010. december jelent meg - mint ajánlás.

- Az XPath 2.0 nyelvi *specifikációja* sokkal nagyobb, mint az XPath 1.0-é.
- Változások történtek az *alapvető fogalmát* illetően, mint pl.: a típus rendszer: *Document Order, Effective Boolean Value, Input Sources*.

XPath szabvány története - version 2.0

Kapcsolódó szabványok:

- *XQuery 1.0 and XPath 2.0 Data Model (XDM) (Second Edition)* (W3C ajánlás, 2010. december 14.)
<https://www.w3.org/TR/xpath-datamodel/>
- *XQuery 1.0 and XPath 2.0 Functions and Operators (Second Edition)*
• (W3C ajánlás, 2010. december 14.)
<https://www.w3.org/TR/xquery-operators/>
- *W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures* (W3C ajánlás, 2012. április 5.)
<https://www.w3.org/TR/xmlschema11-1/>

XPath szabvány története – version 3.0

XML Path Language (XPath) 3.0 (W3C ajánlás, 2014. április 8.)

<https://www.w3.org/TR/xpath-30/>

Kapcsolódó szabványok:

- *XQuery and XPath Data Model*

<https://www.w3.org/TR/xpath-30/>

- *XPath and XQuery Functions and Operators* (2014. április 8.) Arithmetic operators:

<https://www.w3.org/TR/xpath-30/>

Operators	Meaning
op:numeric-add	Addition
op:numeric-subtract	Subtraction
op:numeric-multiply	Multiplication
op:numeric-divide	Division
op:numeric-integer-divide	Integer division
op:numeric-mod	Modulus
op:numeric-unary-plus	Unary plus
op:numeric-unary-minus	Unary minus (negation)

Arithmetic operators on numeric values

XPath szabvány története – version 3.0

Operators	Meaning
<u>op:numeric-add</u>	Addition
<u>op:numeric-subtract</u>	Subtraction
<u>op:numeric-multiply</u>	Multiplication
<u>op:numeric-divide</u>	Division
<u>op:numeric-integer-divide</u>	Integer division
<u>op:numeric-mod</u>	Modulus
<u>op:numeric-unary-plus</u>	Unary plus
<u>op:numeric-unary-minus</u>	Unary minus (negation)

XPath szabvány története - version 3.0

- *XQuery 3.0: An XML Query Language* (W3C ajánlás, 2014. április 8.)

<https://www.w3.org/TR/xquery-30/>

- *XSL Transformations (XSLT) Version 3.0* (W3C ajánlás, 2017. június 8.)

<https://www.w3.org/TR/xslt-30/>

XPath szabvány története – version 3.1

XPath 3.1 - 2017. 03. 21-től jelent meg.

<https://www.w3.org/TR/xpath-31/>

- A legjelentősebb változás az, hogy az XPath 2.0 sokkal *gazdagabb típus rendszerrel* rendelkezik.
- Az XPath 2.0 valójában az XQuery 1.0 egy része.
- Ugyanazt a lekérdezés adatmodellt használják (XDM)."

(Forrás: Jeszenszky, DE)

XPath szabvány története – version 3.1

„Gazdag típus rendszere

- *Predefined Schema Types*: `xs:untyped`,
`xs:untypedAtomic`, `xs:dayTimeDuration`.
- *Namespace-sensitive Types*: `xs:QName`, `xs:NOTATION`
- *Typed Value and String Value*: pl.: egy csomópont beírt értéke az *xs:integer* érték 30, akkor a *string* értéke " 30" vagy " 0030" lehet."

[Forrás: https://www.w3.org/TR/xpath-31/#id-typed-value](https://www.w3.org/TR/xpath-31/#id-typed-value)

XPath szabvány - jellemzők

- „Tömör, nem-XML szintaxis (XPath kifejezések).
- XPath *kifejezések* használhatjuk *URI-ban* (egy rövid karaktersorozat, amelyet egy webes erőforrás azonosítására használunk) és *attribútumok értékeiben*.
- XML dokumentumokhoz egy *logikai adatmodell van* definiálva, de eltérések van a DOM adatmodelljétől.
 - *Fa adatmodell*, amelyben minden csomópontához definiált annak *sztring értéke*.

XPath szabvány - jellemzők

- *Az XPath kifejezések egy részhalmaza alkalmas mintaillesztésre (illesztés vizsgálata), - azt vizsgáljuk, hogy egy adott csomópont illeszkedik-e egy adott mintára,*
- *Az XPath kifejezés onnan származik, hogy a '/' karakter éppen olyan módon használt egy hierarchikus szerkezetben való navigálásra, mint az állományrendszernél.”*

(Forrás: Jeszenszky, DE)

Mi az XPath? - szolgáltatásai

„Az XPath a World Wide Web Consortium (W3C) hivatalos ajánlása.

Szolgáltatásai:

Az XPath meghatározza a struktúrát - meghatározza az XML-dokumentum részeit, pl: elemet, attribútumot, szöveget, névteret, feldolgozási utasításokat, megjegyzéseket és dokumentum csomópontokat.

Mi az XPath? - szolgáltatásai

- **Az XPath biztosítja az útvonal kifejezését** – elérési útvonalat (kifejezéseket) kínál az XML dokumentumok *csomópontjainak vagy csomópontok listájának* kiválasztására.
- **Az XPath egy szabványos függvény** – gazdag függvénykönyvtárat kínál: a karakterlánc-értékek, numerikus értékek, dátum- és idő-összehasonlítás, csomópont- és szekvencia-manipuláció, logikai értékek stb.

Mi az XPath? - szolgáltatásai

- **Az XPath az XSLT egyik központi eleme** – az XSLT-szabvány egyik fő eleme az XPath.
- **W3C ajánlás** - Az XPath a World Wide Web Consortium (W3C) hivatalos ajánlása.”

Forrás: <https://www.javatpoint.com/>

XPath szabvány

Az XPath szabvány az XML dokumentumrészletek kijelölésére 1999-ben jött létre.

Az XPath nyelv egyik jellemzője, hogy az XML dokumentumot egy dokumentum faként kezeli, amely fa nagyban hasonlít a DOM fa modelljére.

XPath szabvány – mire használjuk?

a.) Az XPath arra használatos, hogy *elérjünk/kiválasszunk bizonyos XML elemeket és attribútumokat.*

A hagyományos értelemben vett *útvonalakhoz hasonlít* (a fájlrendszer is egy fa).

b.) Az XPath kifejezések egy részhalmaza *alkalmas mintaillesztésre (illesztés vizsgálatára - azaz azt teszteljük, hogy egy adott csomópont illeszkedik-e egy adott mintára).*

Több, mint 200 beépített függvény tartozik hozzá.

Az XPath-ot erősen használja az XSLT.

XPath – DOM eltérés

Néhány ponton eltér tőle.

Az XPath fa modellje az alábbi pontokban tér el a DOM modelljétől:

- nincs szimbólum kezelés,
- nincs CDATA kezelés,
- nincs DTD kezelés,
- minden csomóponthoz tartozik egy *string* érték.

Az XPath feldolgozási modell

„A XPath alapvetően az *adatmodell* és a *kifejezés kontextus* feldolgozásán alapul.

Feldolgozási modell lépései:

1. Az *adatmodell* generálása:

- Az XML-elemző *Infoset-et* generál, majd opcionálisan végbemehet egy validáció (PSVI= Post Schema Validation Infoset).
- Ezt az *Infoset-et* vagy *PSVI-t* - *XDM-példánnyá* kell *transzformálni*.

Az XPath feldolgozási modell

2. *Séma import feldolgozás.*

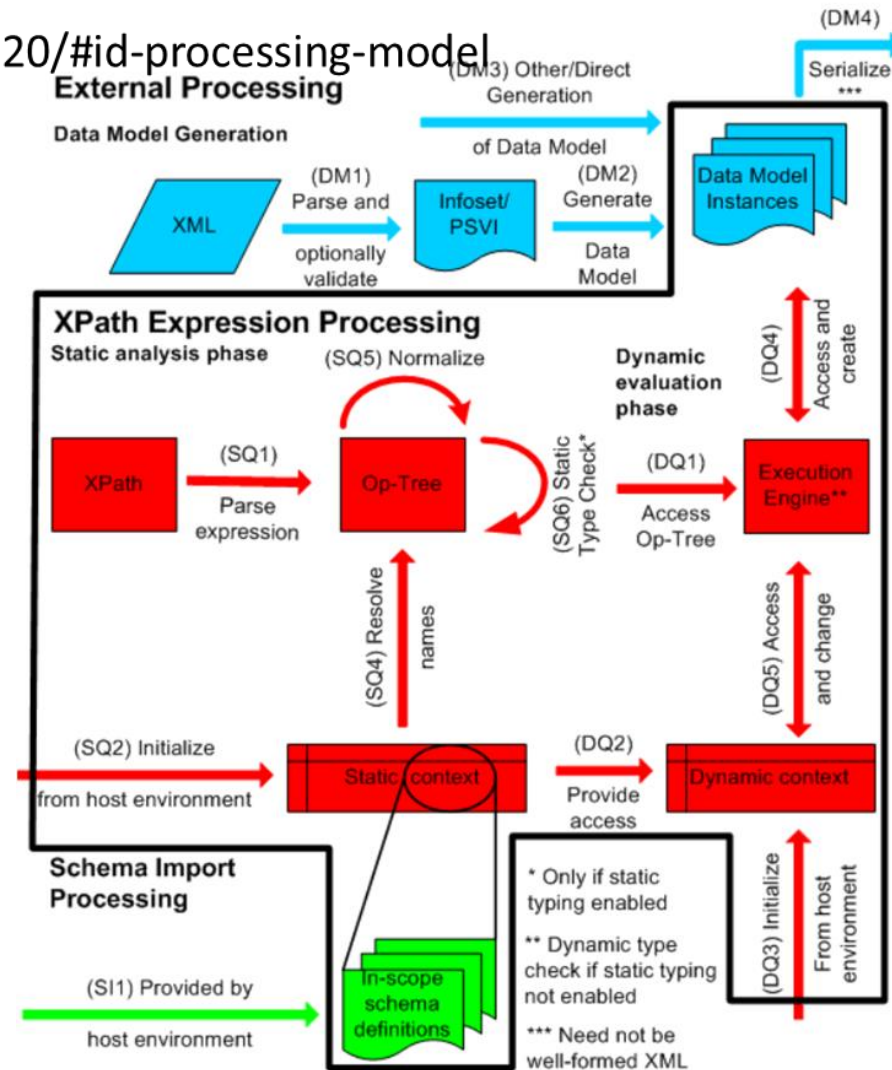
3. *Kifejezések feldolgozása:*

- *statikus elemzés:* névtér-, sémadefiníciók, függvények szignatúrája kiértékelése,
- *dinamikus kiértékelés:* statikus környezet komponensei + környezeti *elem, pozíció, méret.*

4. *Serializáció*”

Az XPath feldolgozási modell

<https://www.w3.org/TR/xpath20/#id-processing-model>



XPath szabvány

Az XML dokumentum *egy fával reprezentálható*, melyben:

- egy elemnek több *azonos nevű gyermeke* is lehet és
- egy *elemnév a dokumentum több különböző helyen is* előfordulhat – ezért,
- szükség van egy olyan mechanizmusra, mely alkalmas a fa *egyes elemeinek hatékony kijelölésére*.

XPath szabvány – részdokumentum kijelölés

Legfontosabb szempontok a *részdokumentum* kijelölésénél:

- tetszőleges *elem* vagy *elemhalmaz* kijelölhető legyen,
- rugalmas *elemkijelölés biztosítása*, melyben az elem kiválasztása történhet:
 - *név* alapján,
 - *pozíció* alapján,
 - *érték* alapján,

XPath szabvány - részdokumentum kijelölés

- *összetett feltételek* támogatása alapján,
- *származtatott kifejezések* támogatása,
- *adattípusok kezelése*.

XPath – kifejezés (expression)

Az XPath kifejezés általában egy mintát határoz meg a csomópontkészlet kiválasztásához.

Alkalmas mintaillesztésre (illesztés vizsgálatára - azaz arra, hogy teszteljük, hogy egy adott csomópont illeszkedik-e egy adott mintára).

Az XPath specifikáció hét csomópontot határoz meg.

- Az XPath egy elérési út kifejezés segítségével **csomópontot** vagy **csomópontok listáját** választja ki egy XML dokumentumból.

XPath szabvány - csomópont típusai

Az XPath fa *csomópont típusai*:

- *gyökér*: XML-dokumentum *gyökérelem-csomópontja*,
- *elem*: egy- és kéttagú elemeket foglalja magába,
- *elemjellemző*: egy *elemcsomópont* *attribútuma*,
- *szöveg*: *szöveget tartalmazó levél*, *elemcsomópont szövege*,
- *megjegyzés*: *megjegyzést leíró elem*,

XPath szabvány - csomópont típusai

- *feldolgozási utasítás*: feldolgozónak szóló direktívák levele,
- *névtér*: az érvényes névterek definícióját tartalmazza.

Egy XML dokumentum megadható csomópontok fájaként.

A fa legfelső elemét *gyökérelemnek* nevezzük.

XPath - Expression

KIFEJEZÉS	LEÍRÁS
<i>nodename</i>	Kiválasztja az összes node-t, aminek a „ <i>nodename</i> ”
/	Kiválasztja a gyökércsomópontot.
//	Az <i>aktuális node-tól indítja</i> a keresést és kiválasztja az összes egyezést.
.	Kiválasztja az <i>aktuális csomópontot</i> .
..	Kiválasztja az <i>aktuális csomópont szülőjét</i> .
@	Kiválasztja az <i>attribútumokat</i> .

XPath – kifejezések – 1. példa

„Példa: student.xml

class / student	Kiválasztja az összes „ <i>student</i> ” elemet, amely a „ <i>class</i> ” gyermeke.
//student	Kiválasztja az összes „ <i>student</i> ” elemet, függetlenül attól, hogy hol vannak a dokumentumban.

XPath kifejezések – 2. példa

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book>
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
</bookstore>
```

- A `<bookstore>` – a *root node*
- Az `<author>J K. Rowling</author>` - *element node*,
- `lang="en"` - *attribute node*,
- *atomi értéknek* nevezzük az *olyan csomópontokat*, aminek *nincs se gyerek se szülője*. Pl: J K. Rowling (Joanne Kathleen Rowling)

XPath kifejezés – 2. példa

XPath kifejezés	Leírás
bookstore	Kiválasztja az összes bookstore nevű node-ot
/bookstore	Kiválasztja a a gyöker bookstore elemet (A /-el kezdődő XPath mindig abszolút útvonalat jelöl)
bookstore/book	Kiválasztja az összes book elemet, mely a bookstore gyereke.
//book	Kiválasztja az összes book elemet, nem fontos hogy a fa-ban hol vannak.
bookstore//book	Kiválasztja az összes book elemet, amely a bookstore leszármazottja (transzitiv gyereke).
//@lang	Kiválasztja az összes lang nevű attribútumot.

XPath szabvány – 3. példa

```
<book>
```

```
  <title>XML for Dummies</title>
```

```
  <authors>
```

```
    <author id="1">Lucinda Dykes</author>
```

```
    <author id="2">Ed Tittel</author>
```

```
  </authors>
```

```
  <price currency="USD">30</price>
```

```
</book> „
```

https://www.w3schools.com/xml/xpath_nodes.asp

Xpath – csomópontok közötti kapcsolatok

Az egyes csomópontok között különböző kapcsolatok lehetnek:

Szülő: minden *elemnek* és *attribútumnak* egy szülője van.

Pl.: A **book** *elem* a *title*, *authors*, és *price* elemek szülője.

Gyerek: az elem csomópontoknak lehet 0 vagy több gyerek elemük.

Pl.: Az *authors* gyerekelemei *author* elemek.

Xpath – csomópontok közötti kapcsolatok

Testvér: olyan *elemcsomópontok*, melyeknek *ugyanaz a szülője*.

Pl.: Testvérek a *title*, *authors* és *price* csomópontok.

Ős: a csomópont szülője, annak szülője, és így tovább.

Pl.: Az *author* csomópontok ősei az *authors* és a *book*.

Leszármazott: egy csomópont *gyerekei*, annak *gyerekei*, és így tovább.

Pl.: A *book* leszármazottja *minden benne foglalt elem*.

XPath adatmodellje - csomópont típusok

- gyökér csomópont: nincs neve; tartalma a dokumentum szövegcsomópontjainak konkatenációja
- elem csomópont: van neve; tartalma a befoglalt szövegcsomópontok konkatenációja (Betűk vagy szavak összefűzése)
- elemjellemző csomópont: van neve, tartalma a jellemző értéke
- szöveg csomópont: nincs neve tartalma a szövegrész
- feldolgozási utasítás csomópontja: neve a beállított paraméter; tartalma az átadott érték
- megjegyzés csomópont: nincs neve, tartalma a megjegyzés szövege
- névtér csomópont: neve a névtér alias neve; tartalma a tényleges névtér azonosít

Az XPath modell egy csomópontok értékeként az alatta elhelyezkedő szöveges csomópontokból összefűzéséből keletkező szöveges értéket tekinti. Az összefűzésnél a sorrend az úgynevezett dokumentum-sorrenden alapszik.

Forrás: KovácsL

XPath szabvány - csomópont típusai

Példa: XML dokumentum az XPath csomópontok különböző terminológiájára.

```
<? xml version = "1.0" encoding = "UTF-8" ?>
< Könyvtár >
  < könyv >
    < title lang = "hu" > Életem három hibája </ title >
    < szerző > Chetan Bhagat </ szerző >
    < év > 2008 </ év >
    < ár > 110 </ ár >
  </ könyv >
</ Könyvtár >
```

XPath szabvány - csomópontok

Csomópontok a XML dokumentumban.

`<library>` (root element node)

`<author>`Chetan Bhagat`</author>` (element node)

`lang="en"` (attribute node)

```
<? xml version = "1.0" encoding = "UTF-8" ?>
```

```
< Könyvtár >
```

```
< könyv >
```

```
< title lang = "hu" > Életem három hibája </ title >
```

```
< szerző > Chetan Bhagat </ szerző >
```

```
< év > 2008 </ év >
```

```
< ár > 110 </ ár >
```

```
</ könyv >
```

```
</ Könyvtár >
```

XPath szabvány - Relationship of Nodes

Parent Node

Minden *elemnek* és *attribútumnak* van egy szülője, amely az adott elem vagy attribútum legfelső eleme.

Példában a **könyv** elem a *cím*, a *szerző*, az *év* és az *ár* szülője.

```
<book>
  <title lang="en">Three Mistakes of My Life</title>
  <author>Chetan Bhagat</author>
  <year>2008</year>
  <price>110</price>
</book>
```

```
<? xml version = "1.0" encoding = "UTF-8" ?>
< Könyvtár >
  < könyv >
    < title lang = "hu" > Életem három hibája </ title >
    < szerző > Chetan Bhagat </ szerző >
    < év > 2008 </ év >
    < ár > 110 </ ár >
  </ könyv >
</ Könyvtár >
```

XPath szabvány - Relationship of Nodes

Children Nodes: A gyermek csomópontoknak *nulla, egy vagy több gyermekük* lehet.

A példában a *cím*, a *szerző*, az *év* és az *ár* elemei mind a *könyv* elem gyermekei.

```
<book>
  <title lang="en">Three Mistakes of My Life</title>
  <author>Chetan Bhagat</author>
  <year>2008</year>
  <price>110</price>
</book>
```

```
<? xml version = "1.0" encoding = "UTF-8" ?>
< Könyvtár >
  < könyv >
    < title lang = "hu" > Életem három hibája </ title >
    < szerző > Chetan Bhagat </ szerző >
    < év > 2008 </ év >
    < ár > 110 </ ár >
  </ könyv >
</ Könyvtár >
```

XPath szabvány - Relationship of Nodes

Siblings Nodes: azonos szülővel rendelkező csomópontok testvérek.

A példában a *cím*, a *szerző*, az *év* és az *ár* mind testvérek.

```
<book>
  <title lang="en">Three Mistakes of My Life</title>
  <author>Chetan Bhagat</author>
  <year>2008</year>
  <price>110</price>
</book>
```

```
<? xml version = "1.0" encoding = "UTF-8" ?>
< Könyvtár >
  < könyv >
    < title lang = "hu" > Életem három hibája </ title >
    < szerző > Chetan Bhagat </ szerző >
    < év > 2008 </ év >
    < ár > 110 </ ár >
  </ könyv >
</ Könyvtár >
```

XPath szabvány - Relationship of Nodes

Ancestor (ős): A csomópont szülője vagy a szülő szülője meg van adva *ősként*.

A példában a *cím* elem ősei a *könyv* és a *könyvtár* elem.

```
<book>
  <title lang="en">Three Mistakes of My Life</title>
  <author>Chetan Bhagat</author>
  <year>2008</year>
  <price>110</price>
</book>
```

```
<? xml version = "1.0" encoding = "UTF-8" ?>
< Könyvtár >
  < könyv >
    < title lang = "hu" > Életem három hibája </ title >
    < szerző > Chetan Bhagat </ szerző >
    < év > 2008 </ év >
    < ár > 110 </ ár >
  </ könyv >
</ Könyvtár >
```


XPath szabvány - Relationship of Nodes

Descendants: a leszármazottat csomópont gyermekének vagy gyermekeinek nevezzük.

A példában a *könyvtár* elem leszármazottai a *könyv*, a *cím*, a *szerző*, az *év* és az *ár* elemei.

```
<book>
  <title lang="en">Three Mistakes of My Life</title>
  <author>Chetan Bhagat</author>
  <year>2008</year>
  <price>110</price>
</book>
```

```
<? xml version = "1.0" encoding = "UTF-8" ?>
< Könyvtár >
  < könyv >
    < title lang = "hu" > Életem három hibája </ title >
    < szerző > Chetan Bhagat </ szerző >
    < év > 2008 </ év >
    < ár > 110 </ ár >
  </ könyv >
</ Könyvtár >
```


Az XPath adatmodellje – példa

„A XPath fa szerkezetének szemléltetésére egy XSLT mintaprogram készült, amely bejárja és felfedezi a dokumentumfát.

Mintaként az alábbi forrás XML dokumentumot használjuk fel:
(*Forrás: KL*)

Az XPath minta - példa

```
<autok   tul="misi" >
  <!-- elso auto -->
  <kl:auto xmlns:kl="www-db.iit.uni-miskolc.hu" rsz ="r11">
    <tipus> Fiat  <![CDATA[<a>]]></tipus>
    <ar>21233</ar>
  </kl:auto>
  <auto rsz ="r21">
    <tipus> Opel &gt;12 </tipus>
    <ar> 41233</ar>
  </auto>
</autok>
```

Az XPath minta - példa

```
<autok tul="misi" >
  <!-- else auto -->
  <kl:auto xmlns:kl="www-db.iit.uni-miskolc.hu" rsz ="r11">
    <tipus> Fiat <![CDATA[<a>]]></tipus>
    <ar>21233</ar>
  </kl:auto>
  <auto rsz ="r21">
    <tipus> Opel &gt;12 </tipus>
    <ar> 41233</ar>
  </auto>
</autok>
```

```
[comment: : auto adatbazis ]
[element:autok: Fiat <a> 21233 Opel >12 41233 ] prefix: xml
{
  [text: : ]
  [comment: : else auto ]
  [text: : ]
  [element:auto: Fiat <a> 21233 ] prefix: xml prefix: kl
  {
    [text: : ]
    [element:tipus: Fiat <a>] prefix: xml prefix: kl
    {
      [text: : Fiat <a>]
    }
  }
  [text: : ]
  [element:ar: 21233] prefix: xml prefix: kl
}

: 21233]
]
e:rsz: r11]

to: Opel >12 41233 ] prefix: xml

tul: misi]
```

(Forrás: KL)

Az XPath szintaxisa

Az XPath alkalmazásban az ***útvonal kifejezést*** - *csomópontok* vagy *csomópontkészletek kiválasztására* használják egy XML dokumentumban.

A csomópont kiválasztása egy *útvonal* vagy *lépések megadásával* történik.

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book>
    <title lang="en">Three Mistakes of My Life</title>
    <price>110</price>
  </book>
  <book>
    <title lang="en">Immortals of Meluha</title>
    <price>200</price>
  </book>
</bookstore>
```

Csomópontok kiválasztása

A csomópontok kiválasztásához használt elérési útvonalak a következők:

Index	Kifejezés	Leírás
1)	csomópontnév	Kiválasztja az összes "csomópont" nevű csomópontot
2)	/	Kiválaszt a gyökércsomópontból.
3)	//	Kiválasztja az aktuális csomópontból a dokumentumban azokat a csomópontokat, amelyek megfelelnek a kiválasztásnak, függetlenül attól, hogy hol vannak.
4)	.	Kiválasztja az aktuális csomópontot
5)	..	Kiválasztja az aktuális csomópont szülőjét
6)	@	Kiválasztja az attribútumokat

Csomópontok kiválasztása

Útvonal kifejezései

<i>Path Expression</i>	<i>Eredmény</i>
bookstore	Kiválasztja az összes csomópontot a „bookstore” névvel
/bookstore	Kiválasztja a bookstore-t. Megjegyzés: ha az útvonal perjellel (/) kezdődik, az mindig egy elem <i>abszolút útvonalát</i> jelenti!
bookstore/book	Kiválasztja az összes <i>book element</i> , amely a <i>bookstore gyermekei</i> .
//book	Kiválasztja az összes <i>book element</i> , függetlenül attól, hogy hol vannak a dokumentumban.
bookstore//book	Kiválasztja az összes <i>book element</i> , amely a bookstore elem <i>leszármazottja</i> , függetlenül attól, hogy a bookstore elem alatt hol találhatók.
// @ lang	Kiválasztja az összes lang nevű attribútumot.

Predikátum – példa

A predikátumokat egy ***adott csomópont*** vagy egy ***adott értéket tartalmazó csomópont megkeresésére*** használják.

A predikátum mindig szögletes zárójelbe [] vannak ágyazva.

Path Expression	Eredmény
/bookstore/book[1]	Kiválasztja az első book csomópontot, amely a bookstore elem gyermeke.
/bookstore/book[last()]	Kiválasztja az <i>utolsó</i> könyvelemet, amely a bookstore elem gyermeke.
/bookstore/book[last()-1]	Kiválasztja az utolsó, de egy könyv elemet, amely a bookstore elem gyermeke.

Predikátum – példa

<code>/bookstore/book[position()<3]</code>	Kiválasztja az első két könyv elemet, amelyek a bookstore elem gyermekei.
<code>//title[@lang]</code>	Kiválasztja az összes <i>title</i> elemet, amely rendelkezik <i>lang</i> nevű attribútummal.
<code>//title[@lang='en']</code>	Kiválasztja az összes olyan <i>title</i> elemet, amely rendelkezik "lang" attribútummal és értéke "en".
<code>/bookstore/book[price>100]</code>	Kiválasztja a <i>bookstore</i> elem összes <i>book elemét</i> , amelynek 100-nál nagyobb értékű price eleme van.
<code>/bookstore/book[price>100]/title</code>	Kiválasztja a bookstore elem book elemeinek összes <i>title elemét</i> , amelyeknek 100-nál nagyobb az ára.

Selecting Unknown Nodes

Az XPath Wildcard karaktereket használ ismeretlen XML csomópontok kiválasztására.

Wildcard	Description
*	Bármely elemcsomópontnak megfelel.
@ *	Bármely attribútum csomópontnak megfelel.
node()	Bármilyen csomóponthoz illik.

Selecting Unknown Nodes - example

Path Expression	Result
/bookstore/*	Kiválasztja a bookstore elem összes gyermek elemét.
//*	Kiválasztja a <i>dokumentum</i> összes elemét.
//title[@*]	Kiválasztja az összes <i>title</i> elemet, amely rendelkezik legalább egy bármilyen attribútummal.

Selecting Several Paths

A **|** operátor az XPath kifejezésben **több útvonal kiválasztására** szolgál. *Példa:* elérési kifejezést vagy a kifejezések eredményét.

Path Expression	Result
<code>//book/title //book/price</code>	Kiválasztja az <i>összes book elem összes title és price</i>
<code>//title //price</code>	Kiválasztja a dokumentum összes title and price
<code>/bookstore/book/title //price</code>	Kiválasztja a bookstore elem book elemének összes title elemét és a dokumentumban szereplő összes price elemet

XPath kifejezés - tengely

Az XPath kifejezések célja: a fa csomópontjainak halmazából egy tetszőleges részhalmaz kiválasztása.

Az XPath - **kifejezéseket** használ csomópontok és csomóponthalmazok kijelölésére az XML dokumentumban egy megadott tengely mentén.

XPath kifejezés - tengely

A kiválasztás *egy mintával történik*, az eredménybe azon *csomópontok* kerülnek bele, melyek *illeszkednek a mintára*.

Az illesztési feltétel mindig környezetfüggő, függ a kiértékelési környezettől.

XPath kifejezés - kijelölő minta

A kijelölő minta lehet elemi és összetett.

- Egy *elemi mintát* **elérési útvonalnak** is szokás nevezni. Az **elérési minta** *egy vagy több elérési lépésből* épül fel.
- *Összetett mintát* más mintákra épülve, azok uniójaként állítunk elő.

XPath kifejezés

Az elérési lépés kifejezése három részből áll össze:

- *navigációs tengely*: megadja a keresés fő irányát (pl.: a gyerekek felé vagy a szülő fele mozogni),
- *csomópont szűrés*: az érintett csomópontok halmazát szűkíteni lehet a *csomópont neve* vagy *típusa alapján*,
- *egyedi szelekciós feltétel*: kiválasztott csomópontoknál további, pl.: a tartalmukra vonatkozó szűkítési feltételt jelent.

XPath kifejezés felépítése

Az elérési lépés szintaktikája:

`tengely::típus-szűrés[szelekció]`

A szelekciós kifejezés opcionális és lehet összetett is, logikai operátorokat is tartalmazhat.

A kifejezésekben az azonosító nevek és konstansok mellett használhatók XPath függvények is.

XPath kifejezés

Elemi XPath kifejezés általános alakja:
tengely::csomópont-szűrés[szelekció]
Összetett XPath kifejezés:
elem_kif/elemi_kif/...

XPath kifejezés értéke:

- csomópont-halmaz
- logikai értékű érték
- numerikus érték
- szöveges érték

- (navigációs) tengely:
megadja a keresés fő irányát (lehet például a gyererek felé vagy a szülő felé mozogni)
- csomópont-szűrés:
az érintett csomópontok halmazát szűkíteni lehet a csomópont neve vagy típusa alapján
- szelekció :
az előző lépésekben kiválasztott csomópontoknál további, rendszerint a tartalmukra vonatkozó szűkítés

XPath adattípusai – kifejezés értékei

A Java-ban az *XPath* kifejezés a következő *adattípusok* egyikét adhatja vissza:

- **node-set** - *csomópontkészletet* képvisel, a készlet lehet *üres*, vagy *akármennyi csomópontot* is tartalmazhat.
- **node** - *egyetlen csomópontot* képvisel, ez lehet *üres*, vagy *akármennyi gyermekcsomópontot* tartalmazhat.

XPath adattípusai – kifejezés értékei

- **boolean** - az igaz vagy hamis értéket jelenti.
- **number** - lebegőpontos számot képvisel.
- **szöveg** - *nulla vagy több karaktert képvisel*, az XML specifikációban meghatározottak szerint.

XPath – abszolút és relatív útvonal

Egy hely elérési útja egy csomópont helyét határozza meg:

- *abszolút: a hely elérési útja gyökércsomóponttal vagy '/' jellel kezdődik,*
- *relatív útvonal: a hely elérési útja a kiválasztott csomópontnál kezdődik.*

*A tengelyeket arra használják, hogy az elemeket kapcsolatuk alapján azonosítsák, pl.: **szülő, gyermek, testvér** stb. elem.*

XPath – abszolút útvonal

Forrás: <https://www.javatpoint.com/xpath-comparison-operators>

Példa az elemek *abszolút elérési út* használatával.

„Hozzon létre egy XML dokumentumot **students.xml** és ahhoz egy *stíluslap dokumentumot*: **students_abs.xsl**, amely az *XPath kifejezéseket* használja.

a) `<xsl:for-each select = "/class/student">`

b) `<p><xsl:value-of select = "/class/student/firstname"/></p>`

XPath – relatív útvonal

Példa az elemek *relatív elérési út* használatával.

Hozzon létre egy XML dokumentumot **students.xml** és ahhoz egy *stíluslap dokumentumot*: **students_rel.xsl**, amely az *XPath kifejezéseket* használja.

XPath kifejezés - Tengely

A **csomópont helyét** *abszolút vagy relatív útvonal* segítségével határozzuk meg.

Az *elemek kapcsolatuk alapján történő azonosítására az XPath tengelyeket* használják pl.: például *szülő, gyermek, testvér* stb.

A tengelyek arra a *tengelyre utalnak*, amelyen az elemek egy elemhez képest hol helyezkednek el (fekszenek).

XPath kifejezés - Tengelytípusok

Sor. sz.	Tengely és leírás
1.	ancestor - Az <i>aktuális csomópont</i> <i>őseit</i> képviseli, amelyek a szülőket a gyökércsomópontig tartalmazzák.
2.	ancestor-or-self - Az aktuális csomópontot vagy <i>őseit</i> képviseli.
3.	attribute - Az <i>aktuális csomópont attribútumait</i> képviseli.
4.	child - Az <i>aktuális csomópont gyermekeit</i> képviseli.

XPath kifejezés - Tengelytípusok

5.	descendant - Az aktuális csomópont <i>leszármazottait</i> képviseli. Az utódok közé tartoznak a csomópont gyermekei a levélcsomópontig.
6.	descendant-or-self - Az <i>aktuális csomópontot</i> vagy annak <i>leszármazottait</i> .
7	following - A <i>következő csomópontot képviseli</i> , amely az aktuális csomópont után következik.
8.	following-sibling - A kontextuscsomópont <i>következő testvéreit</i> képviseli.

XPath kifejezés - Tengelytípusok

9.	Namespace - Az <i>aktuális csomópont</i> névterét képviseli.
10.	Parent - Az aktuális csomópont szülőjét képviseli.
11.	Preceding - Az olyan csomópontot képviseli, amely az <i>megelőző csomópont</i> (azaz a nyitó címke előtt).
12.	Self - Az aktuális csomópontot képviseli.
13.	Preceding - sibling: megelőző és testvér

XPath kifejezés - Tengelytípusok

A tizenhárom tengelytípust támogat az Xpath szabvány:

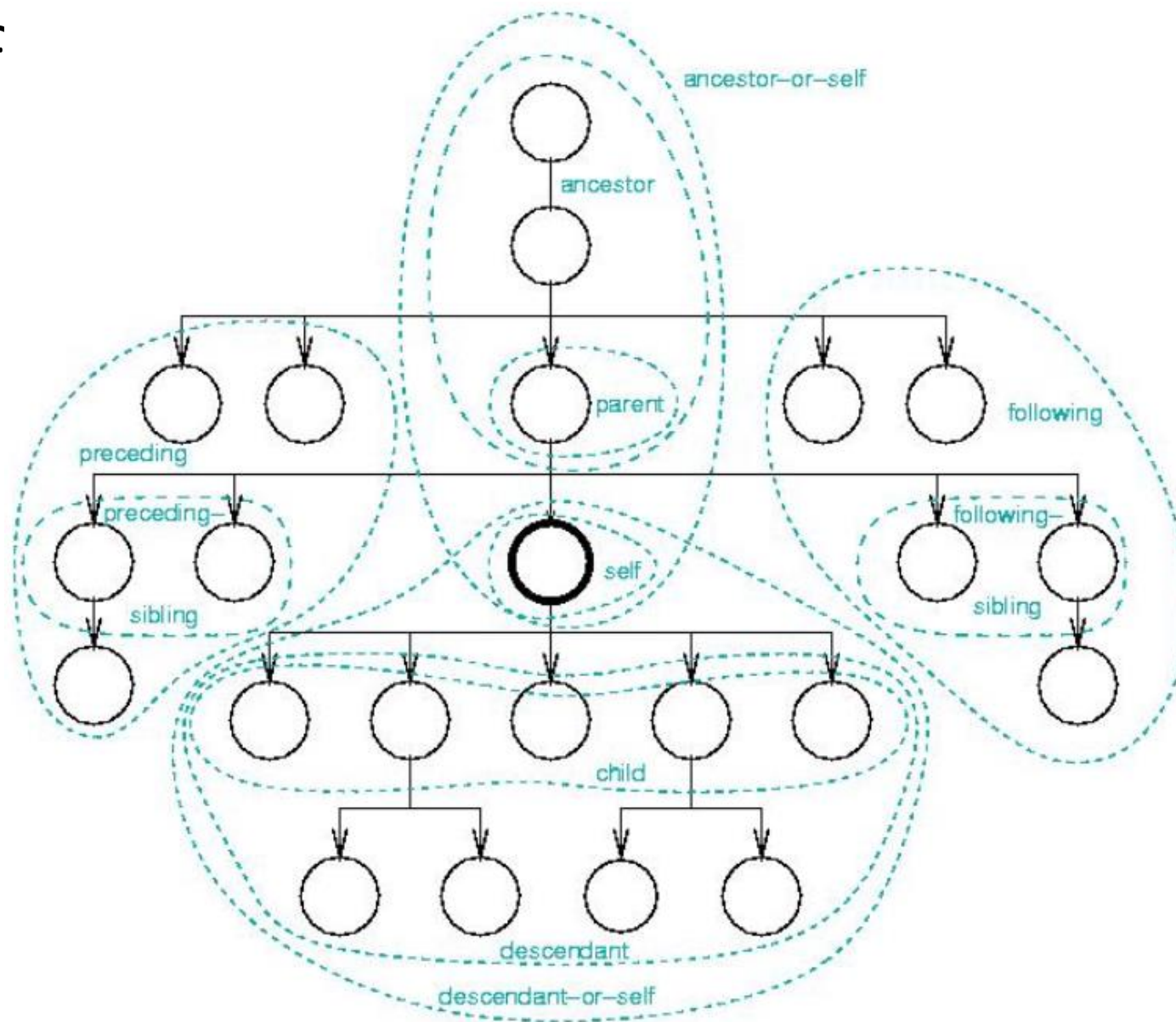
- self: maga a kontextus csomópont
- child: gyerek csomópontok (nem elemjellemző nem névtér)
- descendant: befoglalt csomópontok, tetszőleges mélységben
- descendant-or-self: a befoglalt csomópontok + kontextus
- parent: a szülőt tartalmazó csomópont
- ancestor: befoglaló csomópontok
- ancestor-or-self: a befoglaló csomópontok és a kontextus
- preceding: megelőző csomópontok (nem elemjellemző nem névtér)
- preceding-sibling: megelőző és testvér
- following: követő csomópontok (nem elemjellemző nem névtér)
- following-sibling: követő és testvér
- attribute: elemjellemző
- namespace: névtér leíró csomópontok

Child::*

XPath kifejezés - Tengelytípusok

Tengely	Leírás	Írány	Látható csomópont típusok
child::	A környezetcsomópontból közvetlenül leszármazó elemek.	Forward	Elem, komment, feldolgozó utasítás és szöveg csomópontok
parent::	Közvetlenül a környezetcsomópont feletti elem.	Reverse	Gyökér, elem csomópontok
descendant::	A child:: tengely kiterjesztése. A környezetcsomópontból leszármazó elemek.	Forward	Elem, komment, feldolgozó utasítás és szöveg csomópontok
ancestor::	A parent:: tengely kiterjesztése a csomópont fán felfele.	Reverse	Gyökér, elem csomópontok
descendant-or-self::	Mint a descendant:: tengely, de tartalmazza a környezetcsomópontot.	Forward	Attribútum és névtér nem lehet.
ancestor-or-self	Mint a ancestor:: tengely, de tartalmazza a környezetcsomópontot.	Reverse	Attribútum és névtér nem lehet.
following::	Minden csomópont, ami a környezetcsomópontot követi, kivétel a leszármazottjai.	Forward	Gyökér csomópont, attribútum és névtér nem lehet.
preceding::	Minden csomópont, ami a környezetcsomópontot megelőzi, kivétel a felmenői.	Reverse	Gyökér csomópont, attribútum és névtér nem lehet.
following-sibling::	Minden csomópont, ami a környezetcsomópontot követi és közös a szülőcsomópontja.	Forward	Gyökér csomópont, attribútum és namespace nem lehet.
preceding-sibling::	Minden csomópont, ami a környezetcsomópontot megelőzi és közös a szülőcsomópontja.	Reverse	Gyökér csomópont, attribútum és namespace nem lehet.
attribute::	A környezetcsomópont attribútjai.	Forward	Csak attribútum.
namespace::	A környezetcsomópont névtére.	Forward	Csak névtér.
self::	A környezetcsomópont.		Bármelyik csomópont lehet.

XPath kif



Forrás:

<https://gyires.inf.unideb.hu/GyBITT/12/ch05.html#idp69470128>

XPath kifejezés – Tengely - példa

Készítsünk egy *students.xml* dokumentumot és annak a stíluslap dokumentumát a *students_axes.xls* – amely használja az XPath kifejezést.

XPath expression - Csomópont szűrés

A megadott tengely mentén elhelyezkedő csomópont halmazra történő elemi szűrés

Név alapján

név

névtér:név

*

névtér:*

Típus alapján

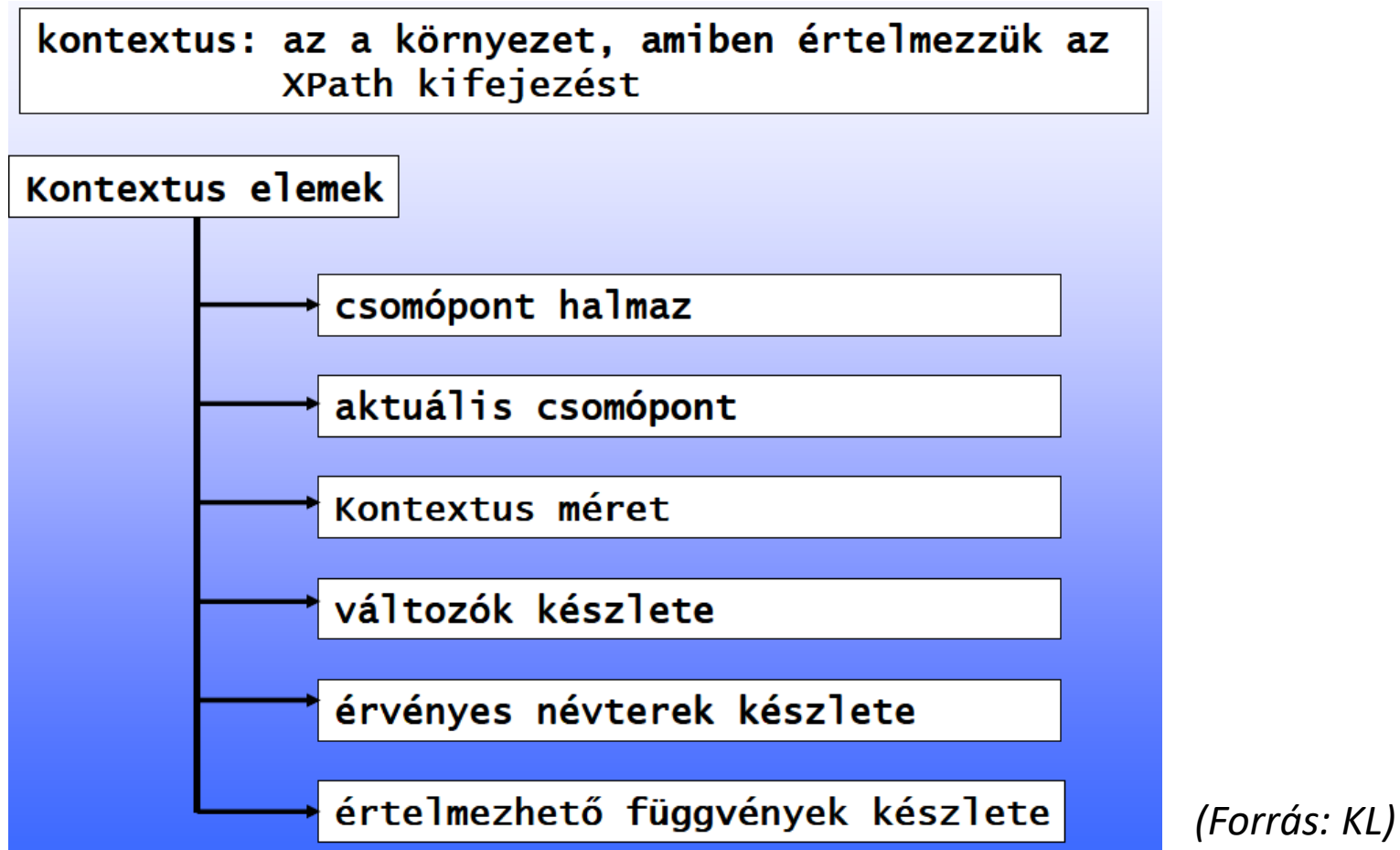
node()

text()

processing instruction()

comment()

XPath expression - Kifejezés kontextusa



Xpath – helyettesítő karakterek

A wildcard kifejezések akkor használják, amikor ismeretlen csomópontokra kell illeszkedni.

Sor sz.	WildCard & Description
1.	* - bármely csomóponthoz illeszkedik.
2.	. - az aktuális csomópont kontextusba illesztésére szolgál.
3.	@* - bármely attribútumhoz illeszkedik.
4.	node() - bármilyen típusú csomópont illesztésére szolgál.

Wildcard kifejezések - példa

A wildcard kifejezések akkor használatosak amikor ismeretlen csomópontokra kell illeszkedni.

Wildcard	Leírás
*	Minden elemre illeszkedik.
@*	Minden attribútumra illeszkedik.
node()	bármilyen node-ra illeszkedik

Példák:

XPath kifejezés	Leírás
/bookstore/*	A bookstore összes gyerekeit választja ki
//*	Az összes elemet kijelöli a dokumentumban.
//title[@*]	Azokat a title elemeket választja ki amiknek van legalább egy attribútuma

XPath - Operators

Az XPath meghatározza a **csomópontok**, **karakterláncok**, **számok** és **logikai típusok** *operátorait és funkcióit*.

Sor sz.	Operators/Functions & Description
1.	Comparison Operators - Összehasonlító operátorok
2.	Boolean Operators - Logikai operátorok a „and”, „or” és „not”
3.	Number Functions/Operators Numerikus függvények/Operátorok
4	String Functions – String függvények.
5.	Node Functions/Operators – Csomópont függvények/Operátorok

XPath – Operators - kitérő

Operator	Description
	It is used to computes two node-sets.
+	Addition Operator
-	Subtraction Operator
*	Multiplication Operator
div	Division Operator
=	Equal Operator
!=	Not Equal Operator
<	Less than Operator

XPath – Operators - kitérő

<=	Less than or Equal to Operator
>	Greater than Operator
>=	Greater than or Equal to Operator
or	Or Operator
and	And Operator

XPath Boolean Operators - example

Index	Operator	Description
1)	and	mindkét feltételnek teljesülnie kell.
2)	or	a feltételeknek bármelyikének teljesülnie kell.
3)	not()	Megadja a nem teljesítendő feltétel ellenőrzésére szolgáló funkciót.

XPath Boolean Operators - example

Készítsünk egy *employee_bool.xml* dokumentumot és annak a stíluslap dokumentumát az *employee_bool.xls* – amely kiválasztja a **@id01 or @id03 rekordokat** és kiírja e két rekordot böngészőbe egy táblázatba.

XPath Number Operators/ Functions

Index	Operator	Description
1)	+	It is used for addition operation.
2)	-	It is used for subtraction operation.
3)	*	It is used for multiplication operation.
4)	div	Osztas operation.
5)	mod	Moduláris műköedés.

XPath Number Operators/ Functions

Készítsünk egy *employee_num.xml* dokumentumot és annak a stíluslap dokumentumát az *employee_num.xls* – amely kiválasztja a 200000 alatti ill. feletti fizetést és kiírja a böngészőbe az egyes fokozatokat (low, medium, high) egy *grade* neve mezőbe táblázatba.

XPath String Functions

Index	Function	Description
1.	starts-with(string1, string2)	Igaz értéket ad vissza, amikor az első karakterlánc a második karakterlánccal kezdődik.
2.	contains(string1, string2)	Igaz értéket ad vissza, ha az első karakterlánc tartalmazza a második karakterláncot.
3.	substring(string, offset, length?)	Visszaadja a karakterlánc egy szakaszát. A szakasz a megadott hosszig eltolással indul.
4.	substring-before(string1, string2)	Visszaadja a string1 részét a string2 első előfordulása előtt.
5.	substring-after(string1, string2)	Visszaadja a string1 részét a string2 első előfordulása után.
6.	string-length(string)	Visszaadja a karakterlánc hosszát.

XPath String Functions

7.	<code>normalize-space(string)</code>	A elemekről levágja a vezető és a hátsó helyet.
8.	<code>translate(string1, string2, string3)</code>	A string1-et adja vissza, miután a string2 bármely megfelelő karakterét kicserélték a string3 karaktereire.
9.	<code>concat(string1, string2, ...)</code>	Minden elem összefűzésére szolgál.
10.	<code>format-number(number1, string1, string2)</code>	Visszaadja a number1 formázott verzióját, miután a string1-et alkalmazta formátum-karakterláncként. A String2 opcionális területi beállítás.

XPath String Functions - example

Készítsünk egy *students_string.xml* dokumentumot és annak a stíluslap dokumentumát az *students_string.xls* – amely kiválasztja a vezetékek és keresztnévet és megszámolja a karakterek hosszát és kiírja a táblázatba.

XPath Node Functions

Index	Operator	Description
1)	/	Egy adott csomópont alatt található csomópont kiválasztására szolgál.
2)	//	Csomópont kiválasztására szolgál a gyökércsomópontból.
3)	[...]	A csomópont értékének ellenőrzésére szolgál.
4)		Két csomópontkészlet egyesítésére szolgál.

XPath Node Functions

Az XPath kifejezéssel együtt használandó *csomópontok függvényeinek*

Index	Function	Description
1)	node()	Mindenféle csomópont kiválasztására szolgál
2)	processing-instruction()	Az utasításokat feldolgozó csomópontok kiválasztására szolgál.
3)	text()	Szöveges csomópont kiválasztására szolgál.
4)	name()	It is used to provide the name of the node.
5)	position()	A csomópont helyzetének megadására szolgál.
6)	last()	Az utolsó csomópont kiválasztására szolgál az aktuális csomóponthoz képest.
7)	comment()	Megjegyzésként csomópontok kiválasztására szolgál.

XPath Node Functions - example

Készítsen egy *students_node.xml* dokumentumot és annak a stíluslap dokumentumát a *students_node.xsl* – amely kiszámítja a hallgatói csomópont helyzetét, majd megjeleníti a `students` részleteit sorozat számmal az első mezőbe a böngészőbe egy táblázatba.

XPath Wildcard - example

Index	Wildcard	Expression
1)	*	Bármely csomópont egyeztetésére szolgál.
2)	.	Az aktuális csomópont kontextusba illesztésére szolgál.
3)	@*	Az aktuális csomópont kontextusba illesztésére szolgál.
4)	node()	Bármilyen típusú csomópont egyeztetésére szolgál.

XPath Wildcard - example

Készítsen egy *students.xml* dokumentumot és annak a stíluslap dokumentumát a *students_wild.xls* – amely a hallgatók jellemzőit egymás alá más-más színekkel.

XPath kifejezés - összefoglalás

A legfontosabb *kifejezések és tengelyek*:

Kifejezés	Tengelynév	Leírás
<code>nodename</code>	-	Minden <code>nodename</code> nevű csomópont kijelölése.
<code>/</code>	-	Abszolút útvonal kijelölése a gyökércsomópontból kiindulva.
<code>//</code>	<code>descendant</code>	A kontextuscsomópont minden leszármazottjának kijelölése.
<code>.</code>	<code>self</code>	A kontextuscsomópont kijelölése.
<code>..</code>	<code>parent</code>	A kontextuscsomópont szülőjének kijelölése.
<code>@</code>	<code>attribute</code>	A kontextuscsomópont attribútumainak kijelölése.

XPath Predicate

A Predicate szögletes zárójelbe írja az XPath kifejezéseket.
Bizonyos feltételekre meghatározott csomópont kiválasztott csomópontjainak korlátozására szolgál.

XPath Predicate - example

Készítsünk egy *students_pred.xml* dokumentumot és annak a stíluslap dokumentumát az *students_pred.xls* – amely pl.: predikátumot használ az *employee node* helyzetének kiszámításához, majd az *employee* adatainak kinyomtatásához.

XPath kifejezés - Fontosabb rövidítések

<code>child::</code>	
<code>attribute::</code>	<code>@</code>
<code>descendant-or-self::node()</code>	<code>//</code>
<code>self::node()</code>	<code>.</code>
<code>parent::node()</code>	<code>..</code>
<code>[position()=2]</code>	<code>2</code>

```
child::/adatbazis/  
descendant::jatekos[csapat[attribute::p=3]/tipus/text  
( ) = 3]/child::nev
```

```
/adatbazis//jatekos[csapat[@p=3]/tipus/text() = 3]/nev
```

Forrás: KovácsL

XPath kifejezés

Az XPath kifejezés nemcsak útvonal kifejezéseket tartalmazhat

+ : összeadás
- : kivonás
* : szorzás
div : osztás
mod : moduló
| : csomópont-halmaz egyesítés
and : logikai és
or : logikai vagy
not() : logikai tagadás
= : egyenlő
!= : nem egyenlő
< : kisebb
> : nagyobb

Forrás: KovácsL

XPath kifejezés - Függvények

<code>count(cs-halmaz)</code>	csomópontok darabszáma
<code>position()</code>	csomópont pozíciója
<code>last()</code>	kontextus mérete
<code>name(cs-halmaz)</code>	csomópont neve
<code>namespace-uri(cs-halmaz)</code>	csomópont névtére
<code>string(kif)</code>	Konverzió szövegre
<code>concat(s1,s2,...)</code>	Szövegek összefűzése
<code>contains(s1,s2)</code>	Rész-szöveg vizsgálat
<code>String-length(s1)</code>	Szöveg hossza
<code>Sum(cs-halmaz)</code>	Összeg
<code>Floor(n)</code>	Egészrész
<code>Round(n)</code>	Kerekítés egészre

Forrás: KovácsL

XPath mintapéldák

Minta állományként a *kiado*, *könyv* és *szerzo* adatait tároló XML állomány szolgál. (Forrás: KovácsL)

XPath mintapéldák

Feltétel: az aktuális kontextus csomópont a dokumentum gyökér eleme.

1. A kiadók városnevei:

1a: descendant-or-self::kiado/varos

2. A könyv első vagy második kulcsszavai:

2a: descendant-or-self::konyv/kulcsszo[position()=1 or position()=2]

3. A 2008-ban kiadott könyv címe:

3a: descendant-or-self::konyv[number(ev) = 2008]/cím”

(Forrás: KL)

Java XPath Parser - Parse XML Document

Az XPath használatának lépései:

- Import XML-related packages.
- Create a DocumentBuilder.
- Create a Document from a file or stream.
- Create an Xpath object and an XPath path expression.

Java XPath Parser - Parse XML Document

- Az XPath kifejezést az *XPath.compile ()* metódussal *fordítja le*, és az *XPath.evaluate ()* útján lefordított kifejezés történik a csomópontok *listájának kiértékelése*.
 - For ciklussal végig iterálás a csomópontok listáján.
 - Vizsgálja az attribútumokat.
 - Vizsgálja az subelement (gyerek elemeket)
 - Hibakezelés.

XLink kifejezés

XLink kifejezés

„Az *XML szabvány* gyökere az *SGML jelölőnyelvben* rejlik, melyhez kapcsolódik a *hypertext koncepció* fogalma is.

A *hypertext dokumentum fő jellemzője*, hogy a dokumentumból, annak egy eleméből *gyorsan és egyszerűen* át lehet lépni egy *másik dokumentum részletre*.

A *hypertext koncepció* egyik jól ismert megvalósulása a HTML nyelv, melynek szintaxisa

```
<A href=URI ..> ... </A>
```

XLink kifejezés

*Egy új elem, funkció jelent meg az XML világában is, amely lehetővé teszi más XML dokumentumokra történő **hatékony és rugalmas hivatkozást**.*

A kidolgozott szabványt nevezik **XLink rendszernek 1.0**, melynek első hivatalos W3C verziója 2001-ben jelen meg.

URL: <https://www.w3.org/TR/xlink/>

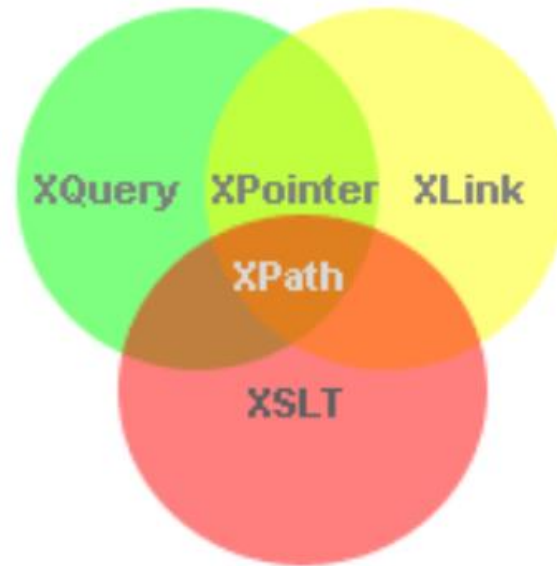
XML Linking Language (XLink) Version 1.1

W3C Recommendation 06 May 2010.

XLink rendszer helye

Az XLink hiperhivatkozások létrehozására szolgál XML dokumentumokban.

- Az XML-dokumentum bármely *elem* hivatkozásként viselkedhet.
- Az XLink a W3C ajánlása.



XLink rendszer célja

Az XLink rendszer célja: egy újabb eszköz a dokumentumrészek közötti kapcsolatok rugalmas és hatékony leírására.

XLink rendszer alapfogalmai:

- *a kapcsolat a dokumentumok erőforrás elemeit köti össze,*
- *a kapcsolat lehet elemi és összetett,*
 - *elemi kapcsolat: egy belső erőforrásból érhető el egy külső erőforrás,*

XLink rendszer célja

- *összetett kapcsolat*: több erőforrás kötődhet egymáshoz egyidejűleg,
- a kapcsolatokhoz *tetszőleges bejárési irányok* köthetők,
- a kapcsolatokhoz több *metaadat* is kapcsolható,
- a kapcsolatrendszer leírása *tárolható külső dokumentumban* is,
- a *leírás igazodik* az XML formátumhoz.

XLink rendszer

Az XLink rendszer sajátossága: hogy a kapcsolati paraméterek megadása *saját elemjellemezőkön* keresztül.

A kapcsolat megadására *tetszőleges elemeket* lehet felhasználni, szerepük azáltal válik *egyértelművé*, hogy *tartalmazzák a megadott elemjellemezőket*.

Az XLink-hez rendelt névtér értéke:

`http://www.w3.org/1999/xlink`

XLink kifejezés - kapcsolat megadása

A modell az alábbi *elemjellemzőket* definiálja:

```
<kapcsolat xmlns:ll="http://www.w3.org/1999/xlink"
  ll:type="simple" ll:href="masik.xml">
  Hivatkozási szöveg
</kapcsolat>
```

XLink névtere: <http://www.w3.org/1999/xlink>

type: a kapcsolat leíró elem jellege
href: erőforrás azonosító, URI alakban jelöljük ki az érintett elemet
from: hivatkozás kiinduló pontja
to: hivatkozás célpontja
title: kapcsolat szöveges jellemzése
label: kapcsolati elem azonosítása
role: a kapcsolat szemantikai tartalmának kijelölése
actuate: a kapcsolat mentén történő ugrás kiváltásának módja
show: a hivatkozott dokumentum megjelenítésének módja

(Forrás: KL)

XLink kifejezés – kapcsolattípusok

Az értelmezett **típusok**:

simple: egyszerű kapcsolat
extended: összetett kapcsolat
locator: külső erőforrás azonosítása
resource: belső erőforrás azonosítása
arc: kapcsolati irány kijelölése
title: leírás megadása

Az XLink modell kétféle kapcsolat fajtát különböztet meg:
Elemi kapcsolat

a kapcsolat két erőforrást köt össze
egyik elem a forrás a másik a célhely
az induló elem egy belső erőforrás
a célelem egy külső erőforrás

```
<kapcsolat xmlns:ll="http://www.w3c.org/1999/xlink"
  ll:type="simple" ll:href="masik.xml">
  Hivatkozási szöveg
</kapcsolat>
```

XLink kifejezés – kapcsolattípusok - show

`show` attribútum: a hivatkozáshoz tartozó célerőforrás hogyan jelenik meg a felhasználó számára:

- `replace`: célerőforrás lecseréli az aktuális dokumentumot (alapértelmezett érték).
- `new`: cél erőforrás egy új ablakban jelenik meg.
- `embed`: cél erőforrás a hivatkozás helyére kerül be az aktuális dokumentumba.

XLink kifejezés – kapcsolattípusok - `actuate`

`actuate` attribútum: hogyan induljon el egy hivatkozás, és általában a következő értékek egyikére van beállítva:

- `onRequest`: a linket a felhasználónak kézzel kell bejárnia (alapértelmezett érték).
- `onLoad`: a hivatkozás automatikusan bejárható a forrásdokumentum betöltésekor.

XLink kifejezés – Összetett kapcsolat

Összetett kapcsolat

több belső erőforrás is szerepelhet
több külső erőforrás is megjelenhet
tetszőleges irányítás dedfiniálható az elemek között

```
<kapcsolat xmlns:ll="http://www.w3c.org/1999/xlink"
  ll:type="extended">
  <vevo ll:type="locator" ll:href="www.w1.org/w1"
    ll:label="KL1" />
  <gyarto ll:type="locator" ll:href="www.w1.org/p1"
    ll:label="GY1" />
  <szallito ll:type="locator" ll:href="www.w1.org/t1"
    ll:label="TR1" />
  <termek ll:type="locator" ll:href="www.w1.org/w1"
    ll:label="TR1" />
  <termek ll:type="locator" ll:href="www.w1.org/w2"
    ll:label="TR2" />
</kapcsolat>
```

XLink kifejezés - példa

Példa: egy ***rendelés*** kapcsolatrendszerét adjuk meg.

A **rendelés** kapcsolat összefog egy *vevőt*, egy *gyártót*, egy *szállítót* és *három terméket*, valamint egy *összérték leíró*t.

A *vevő*, *gyártó* és *szállító*k *külső dokumentumban* tárolódnak.

Az érték *lokális erőforrás*.

A kapcsolat alapján a *vevő* és a *szállító* között lehet mozogni, mégpedig mindkét irányba.

XLink kifejezés - példa

Az összetett kapcsolat a bináris kapcsolatok mellett támogatja az n -es kapcsolatokat is, ahol az alkotó elemek között szabadon definiálhatók az ugrási irányok.

XLink kifejezés - példa

A rendelés kapcsolat példa egy *összetett kapcsolat definiálását* mutatja be:

Felhasznált irodalom

- Kovács László: Adatkezelés XML környezetbe (lásd: moodle).
- Kollár Lajos, Sterbinszky Nóra: Programozási technológiák – Jegyzet, 2014.

<https://gyires.inf.unideb.hu/GyBITT/21/ch06.html#idp46260096>

- Dr. Adamkó Attila: Fejlett Adatbázis Technológiák - Jegyzet, 2013, DE.

Felhasznált irodalom

- XPATH COVER PAGE

<https://www.w3.org/TR/xpath/>

- Jeszenszky Péter: XML, 2019, DE.

- XPath Tutorial

https://www.w3schools.com/xml/xpath_intro.asp

- XML Linking Language (XLink) Version 1.1

<https://www.w3.org/TR/xlink/>

További URL oldalak a XPath-hoz

- <https://docs.oracle.com/javase/tutorial/jaxp/dom/index.html>
- <https://howtodoinjava.com/xml/read-xml-dom-parser-example/>
- <http://tutorials.jenkov.com/java-xml/dom.html>
- <https://www.javatpoint.com/xml-dom>