

eXtensible Markup Language



A szoftver és hardverfüggetlen
adatbázis

Vörös Péter

http://people.inf.elte.hu/vopraai/korszeru_xml/

Példa

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body lang="en">Don't forget me this
  weekend!</body>
</note>
```

Mi az XML?

Adat leíró nyelv

Hardver és Szoftver független

Semmi köze a megjelenítéshez
(Ellenben a HTML-lel, ahol
pont ez a cél)

Építőelemek

- Elemek (Elements)
- Attribútumok (Attributes)
- Entitások (Entities)
- Névterek (Namespaces)
- XML deklaráció (XML declaration)
- Feldolgozási instrukciók (Processing Instruction)
- Megjegyzések (Comments)
- CDATA szekciók (CDATA sections)

Egy XML dokumentumban egy nyitó és megfelelő záró jelölő (tag) közötti részt **elemnek** hívjuk.

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body lang="en">Don't forget me this
weekend!</body>
</note>
```

XML-ben **Attribútumokkal** további információk írhatóak le egy elemről.

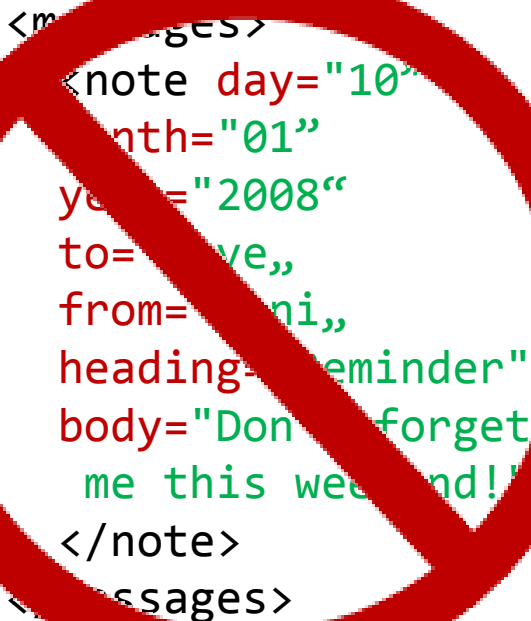
Attribútum csak nyitó tag-ben szerepelhet, és az értékét idézőjelbe kell tenni

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body lang="en">Don't forget me this
weekend!</body>
</note>
```

Túl sok attribútum

```
<messages>
  <note day="10"
    month="01"
    year="2008"
    to="Tove,,
    from="Jani,,
    heading="Reminder"
    body="Don't forget
      me this weekend!">
  </note>
</messages>
```

Az attribútumok helyett, törekedjünk az elemek használatára, ez sokkal áttekinthetőbb



```
<messages>
<note day="10"
      month="01"
      year="2008"
      to="Tove,,
      from="Jani,,
      heading="Reminder"
      body="Don't forget
            me this weekend!"
      </note>
</messages>
```

```
<note>
  <date>
    <year>2008</year>
    <month>01</month>
    <day>10</day>
  </date>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this
weekend!</body>
</note>
```


Az attribútumokat ne vigyük túlzásba, lehetőség szerint hagyjuk meg a metaadatoknak

```
<messages>
  <note id="501">
    <to>Tove</to>
    <from>Jani</from>
    <heading>Reminder</heading>
    <body>Don't forget me this weekend!</body>
  </note>
  <note id="502">
    <to>Jani</to>
    <from>Tove</from>
    <heading>Re: Reminder</heading>
    <body>I will not</body>
  </note>
</messages>
```

Entitás

Az **entitás** egy rövidítés, egy bonyolult vagy hosszú szöveges adat megnevezésére.

Entitások szabadon definiálhatók, a dokumentumban bárhol használhatóak.

Elemzéskor az entitás helyére behelyettesítődik az a szöveg, amit rövidít.

<	<
>	>
&	&
'	'
"	“

Névütközés

1. Dokumentum

```
<table>
  <tr>
    <td>Apple</td>
    <td>Pear</td>
  </tr>
</table>
```

2. Dokumentum

```
<table>
  <name>Coffee Table</name>
  <width>80</width>
  <length>120</length>
</table>
```

N
É
V
Ü
T
K
Ö
Z
É
S

Névterek és minősített nevek

A **névterek** jelölőnevek a tag-ek egyedivé tételére, ezáltal a névütközések elkerülhetőek

A névterekkel egyedivé tett jelölőneveket **minősített neveknek** nevezzük



Névütközés

**1.
Dokumentum**
<table>
...
</table>

**2.
Dokumentum**
<table>
...
</table>

```
<root>
<h:table xmlns:h="http://www.w3.org/TR/html4/">
  <h:tr>
    <h:td>Apples</h:td>
    <h:td>Bananas</h:td>
  </h:tr>
</h:table>

<f:table xmlns:f="http://www.w3schools.com/furniture">
  <f:name>African Coffee
Table</f:name>
  <f:width>80</f:width>
  <f:length>120</f:length>
</f:table>
</root>
```

XML Deklaráció

1.0 verzió esetén opcionális, 1.1 esetén kötelező

Ha nincs megadva akkor az alapértelmezett 1.0-ként azonosítjuk

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

Megjelenítési információt társíthatunk a dokumentumhoz.

```
<?xml-stylesheet type="text/css"  
href="cd_catalog.css"?>
```

Feldolgozási instrukciók

<!-- Ez csak egy komment -->

A megjelenítési információ nem gyakran használt, a W3C által nem is ajánlott, helyette XSLT-t használjunk.

CData

A **CData** szekcióba írt területet az elemző figyelmen kívül hagyja

```
<script>  
  <![CDATA[  
    function matchwo(a,b)  
    {  
      if (a < b && a < 0) then  
        return 1;  
      else  
        return 0;  
    }  
  ]]>  
</script>
```

Jól formált XML

1. Minden tag-nek kell hogy legyen záró párja.
(Szöveg nélküli tag rövidíthető)

```
<p>This is a paragraph.</p>  
<br />
```

2. A tag-ek kis- és nagybetű érzékenyek

```
<Message>incorrect</message>  
<message>correct</message>
```

3. Helyes egymásba ágyazás

```
<b><i>This is incorrect</b></i>  
<b><i>This is correct</i></b>
```

Jól formált XML folytatás...

4. Minden dokumentumnak kell hogy legyen pontosan egy gyökér eleme, ami tartalmazza az összes többi elemet

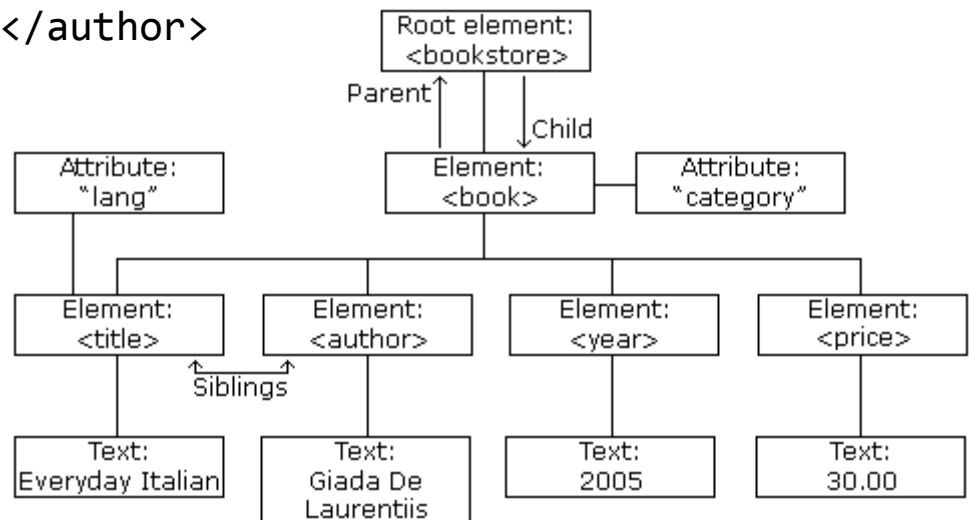
```
<root>  
  <child>  
    <subchild>.....</subchild>  
  </child>  
</root>
```

5. Az attribútum értékeket idézőjelben adjuk meg

```
<title lang=Inglis>HP</title>  
<title lang="en">HP</title>
```

XML fa

```
<bookstore>
  <book category="COOKING">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
  <book category="CHILDREN">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
</bookstore>
```



DTD (Document Type Description)

- A DTD sémaleíró meghatározza az XML dokumentumok szerkezetét.
- DTD használatával specifikáljuk, hogy mikor **érvényes** egy XML dokumentum.
- A DTD azt mondja meg, hogy mikor legális egy elem, illetve milyen beágyazások megengedettek.
- A DTD kifejező ereje nem elég típusok definiálására.

Miért jó a DTD?

- Speciális nyelvet alkothatunk vele
(<http://www.w3.org/TR/html401/sgml/dtd.html>)
- Az alkalmazásunk könnyen ellenőrizni tudja hogy a kapott adat valid-e.

Hogyan?

```
<?xml version="1.0"?>  
<!DOCTYPE note SYSTEM "note.dtd">  
<note>  
  <to>Tove</to>  
  <from>Jani</from>  
  <heading>Reminder</heading>  
  <body>Don't forget me this weekend!</body>  
</note>
```

```
<!ELEMENT note (to,from,heading,body)>  
<!ELEMENT to (#PCDATA)>  
<!ELEMENT from (#PCDATA)>  
<!ELEMENT heading (#PCDATA)>  
<!ELEMENT body (#PCDATA)>
```

```
<!ELEMENT note (to,from,heading,body)>  
<!ELEMENT to (#PCDATA)>  
<!ELEMENT from (#PCDATA)>  
<!ELEMENT heading (#PCDATA)>  
<!ELEMENT body (#PCDATA)>
```

!DOCTYPE note megadja hogy a dokumentum gyökere a note

!ELEMENT note rész megadja hogy a note "to,from,heading,body" részekből áll

!ELEMENT to, from, heading, body megadják hogy azok az elemek szöveges értékűek

a	a elem
e1?	0 vagy 1 előfordulása az e1 kifejezésnek
e1*	0 vagy több előfordulása az e1-nek
e1+	1 vagy több előfordulása az e1-nek
e1, e2	Az e2 az e1 után következik
e1 e2	e1 vagy e2
(e)	Zárójelezés
#PCDATA	Nem elemzendő szöveg
EMPTY	Nincs tartalom
ANY	Tetszőleges tartalom
(#PCDATA a ₁ . . a _n)*	Vegyes tartalom

Attribútumokra is megfogalmazható megszorítás

#REQUIRED - Szükséges az adott attribútum

```
<!ATTLIST person number ID #REQUIRED>  
<person number="5" />  
<person />
```

#IMPLIED - Opcionális az adott attribútum

```
<!ATTLIST contact fax CDATA #IMPLIED>  
<contact fax="555-667788" />  
<contact />
```

#FIXED - Fix érték kell hogy legyen

```
<!ATTLIST sender company IDREF #FIXED "Microsoft">  
<sender company="Microsoft" />  
<sender company="W3Schools" />
```

Entitás létrehozás

```
<!ENTITY NEWSPAPER "Vervet Logic Times">
```

```
<!ENTITY PUBLISHER "Vervet Logic Press">
```

```
<body>&NEWSPAPER;&PUBLISHER;</body>
```

Érvényesség DTD szerint

Egy XML dokumentum **érvényes** egy adott DTD szerint, ha

- a dokumentum illeszkedik a reguláris kifejezésekre
- az attribútumok típusai megfelelőek
- az azonosítók és hivatkozások használata szabályos.

Feladat

`http://people.inf.elte.hu/vopraai/korszeru_xml/DTD_feladat.txt`

Ellenőrzéshez:

`http://www.xmlvalidation.com/`

XSD - XML séma

Nagyon hasonlít a DTD-re

Vannak típusok

- xs:string
- xs:decimal
- xs:integer
- xs:boolean
- xs:date
- xs:time

XSD példa

```
<?xml version="1.0"?>
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="note">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="date" type="xs:date"/>
          <xs:element name="to,, minOccurs=„1" maxOccurs="unbounded" .../>
          <xs:element name="from" type="xs:string"/>
          <xs:element name="heading" type="xs:string"/>
          <xs:element name="body" type="xs:string"/>
        </xs:sequence>
        <xs:attribute name="ID" type="xs:integer" use="required"/>
      </xs:complexType>
    </xs:element>
  </xs:schema>
```

DTD Tuning

Feladat:

`http://people.inf.elte.hu/vopraai/
korszeru_xml/XSD_feladat.txt`

Validálás:

`http://www.utilities-
online.info/xsdvalidation/`

XPath

XPath kifejezéseit az XML fa alapján kell kiértékelni.

- Navigálás
- Kiválasztás
- Általános információ kinyerés

Csúcs

Az XML fa csúcsai a következő csúcstípusúak lehetnek:

- Dokumentum
- Elem
- Attribútum
- Szöveg
- Instrukció
- Megjegyzés
- Névtér

Csúcs

Egy csúcsnak lehet **neve**, vagy **értéke**, vagy **mindkettő**

- egy **Elem** csúcsnak neve van, de értéke nincs;
- egy **Szöveg** csúcsnak nincs neve, de karaktersorozatú értéke van;
- egy **Attribútum** csúcsnak neve és értéke is van.

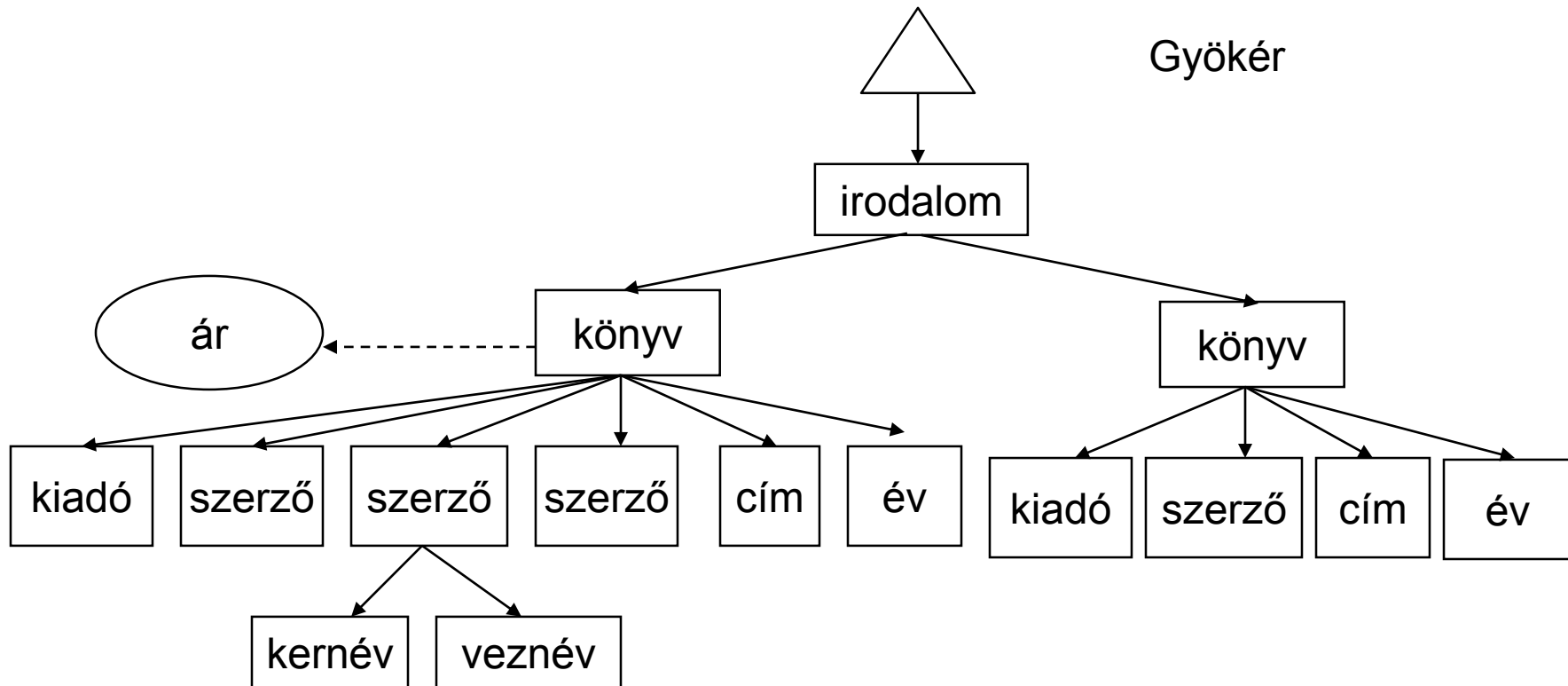
Csúcs sorrend

- A csúcsok közti rendezést a dokumentum sorrend definiálja, mely a pre-order bejárásnak felel meg.
 - Egy szülő csúcs megelőzi a gyerekeit és attribútumait.
 - A testvér csúcsok közül az attribútumok jönnek előbb és csak utána a más típusúak.
 - Az attribútumok sorrendje implementáció függő.

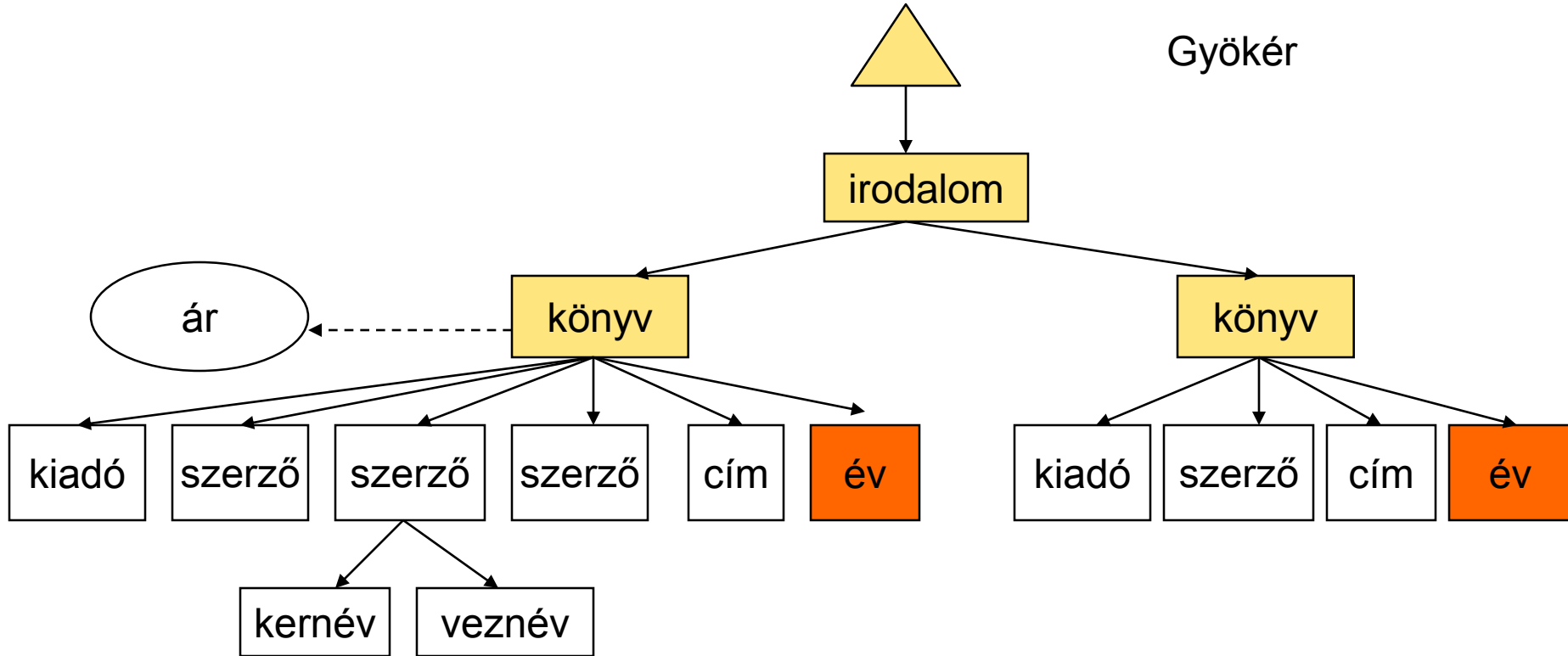
Példa Xpath lekérdezésre

```
<irodalom>
  <könyv ár='55'>
    <kiadó>Addison-Wesley</kiadó>
    <szerző>Serge Abiteboul</szerző>
    <szerző><kernév>Rick</kernév>
      <veznév>Hull</veznév>
    </szerző>
    <szerző>Victor Vianu</szerző>
    <cím>Foundations of Databases</cím>
    <év>1995</év>
  </könyv>
  <könyv> <kiadó>Freeman</kiadó>
    <szerző>Jeffrey D. Ullman</szerző>
    <cím>Principles of Database and Knowledge Base Systems</cím>
    <év>1998</év>
  </könyv>
</irodalom>
```

Az XML fa



/irodalom/könyv/év

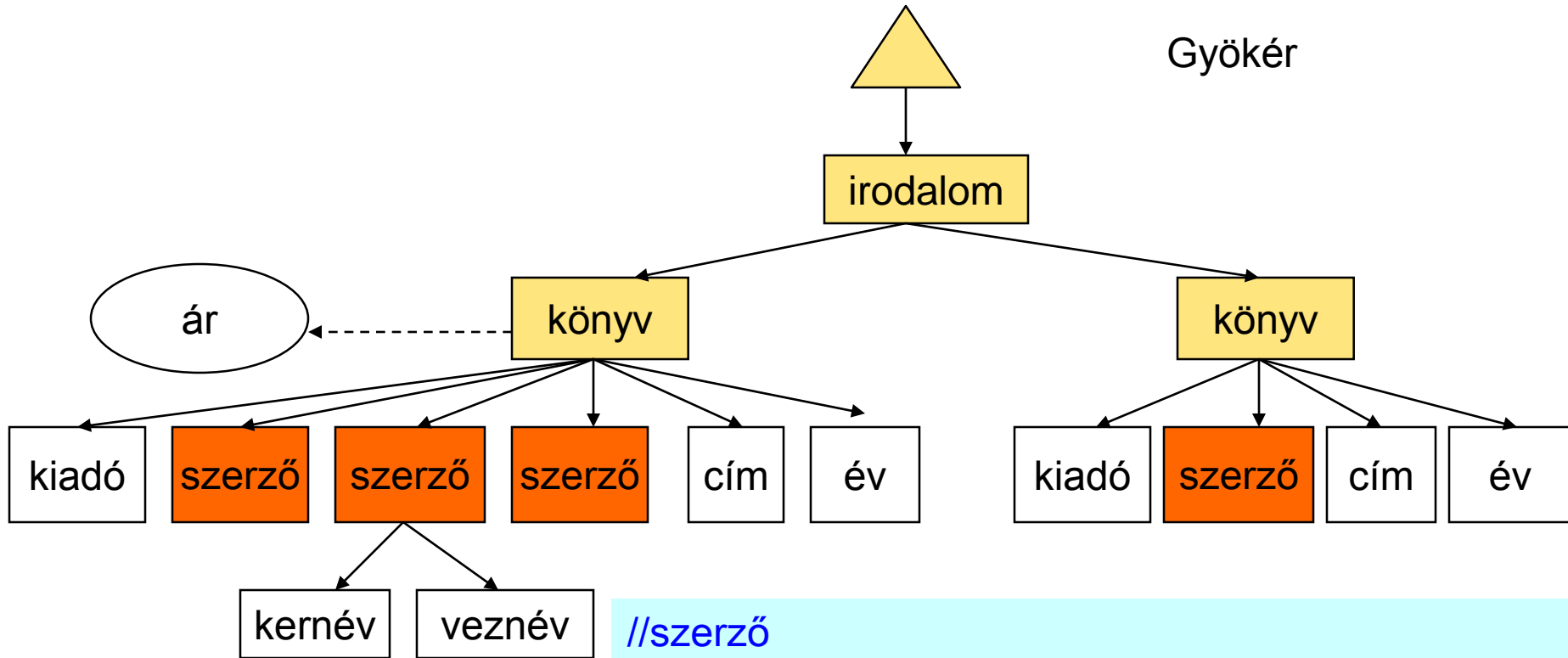


/irodalom/könyv/év

Eredmény: <év>1995</év>
<év>1998</év>

/irodalom/cikk/év eredménye üres, mert nincs cikk.

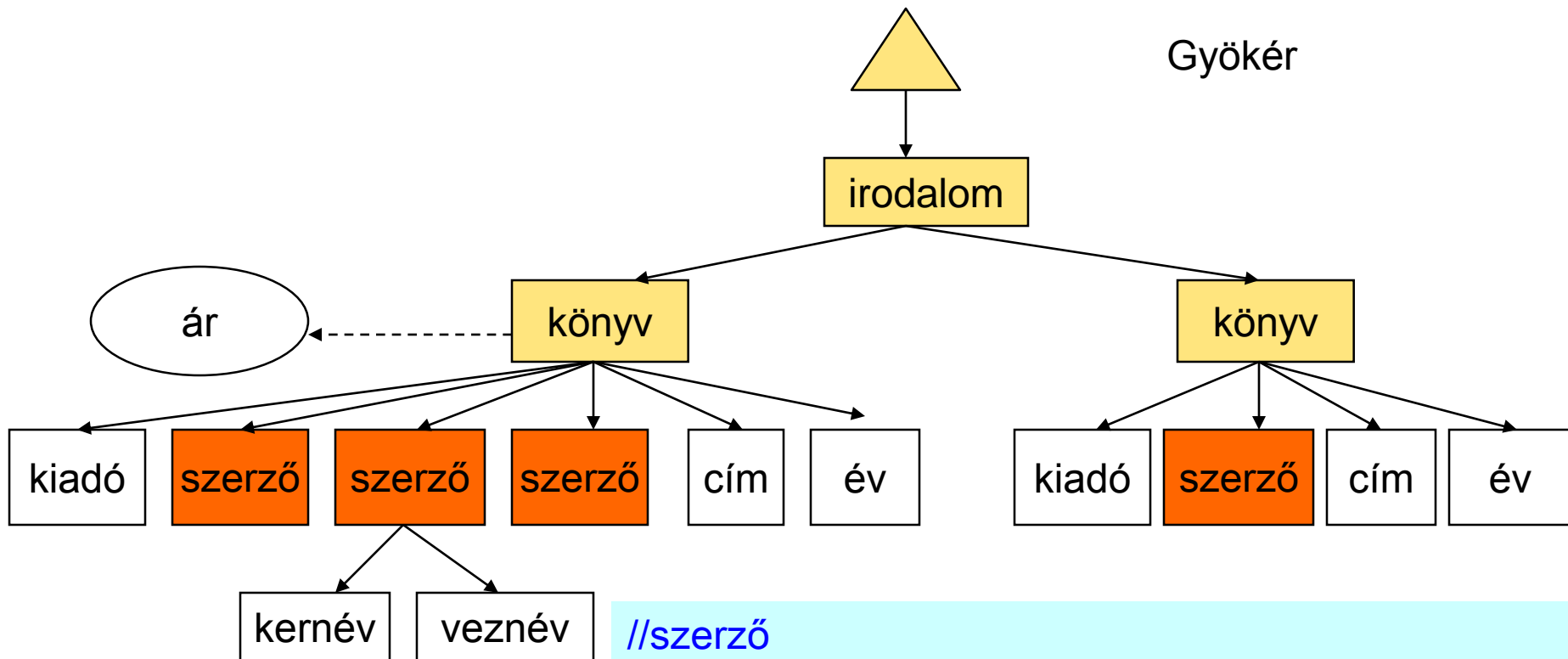
//szerző



//szerző

Eredmény: `<szerező>Serge Abiteboul</szerező>`
`<szerező><kernév>Rick</kernév>`
`<veznév>Hull</veznév>`
`</szerező>`
`<szerező>Victor Vianu</szerező>`
`<szerező>Jeffrey D. Ullman</szerező>`

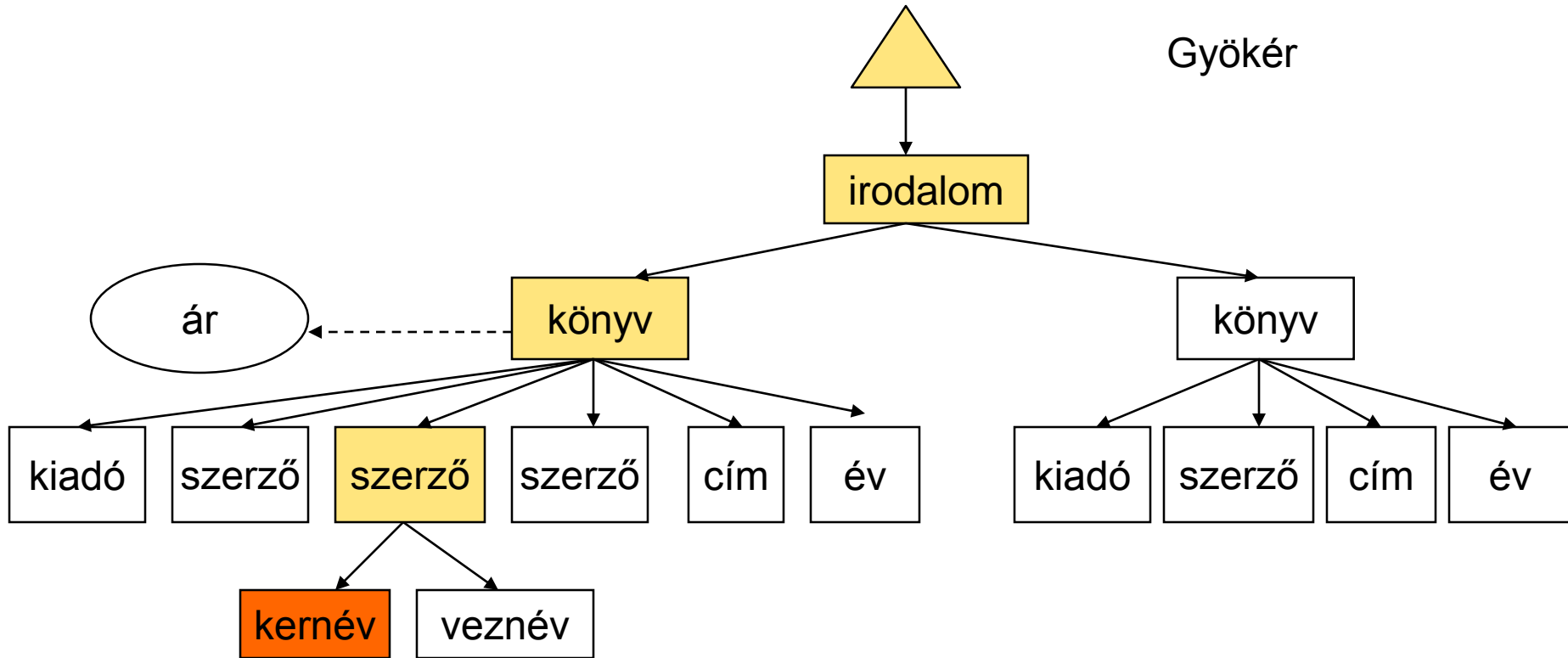
//szerző



//szerző

Eredmény: `<szerző>Serge Abiteboul</szerző>`
`<szerző><kernév>Rick</kernév>`
`<veznév>Hull</veznév>`
`</szerző>`
`<szerző>Victor Vianu</szerző>`
`<szerző>Jeffrey D. Ullman</szerző>`

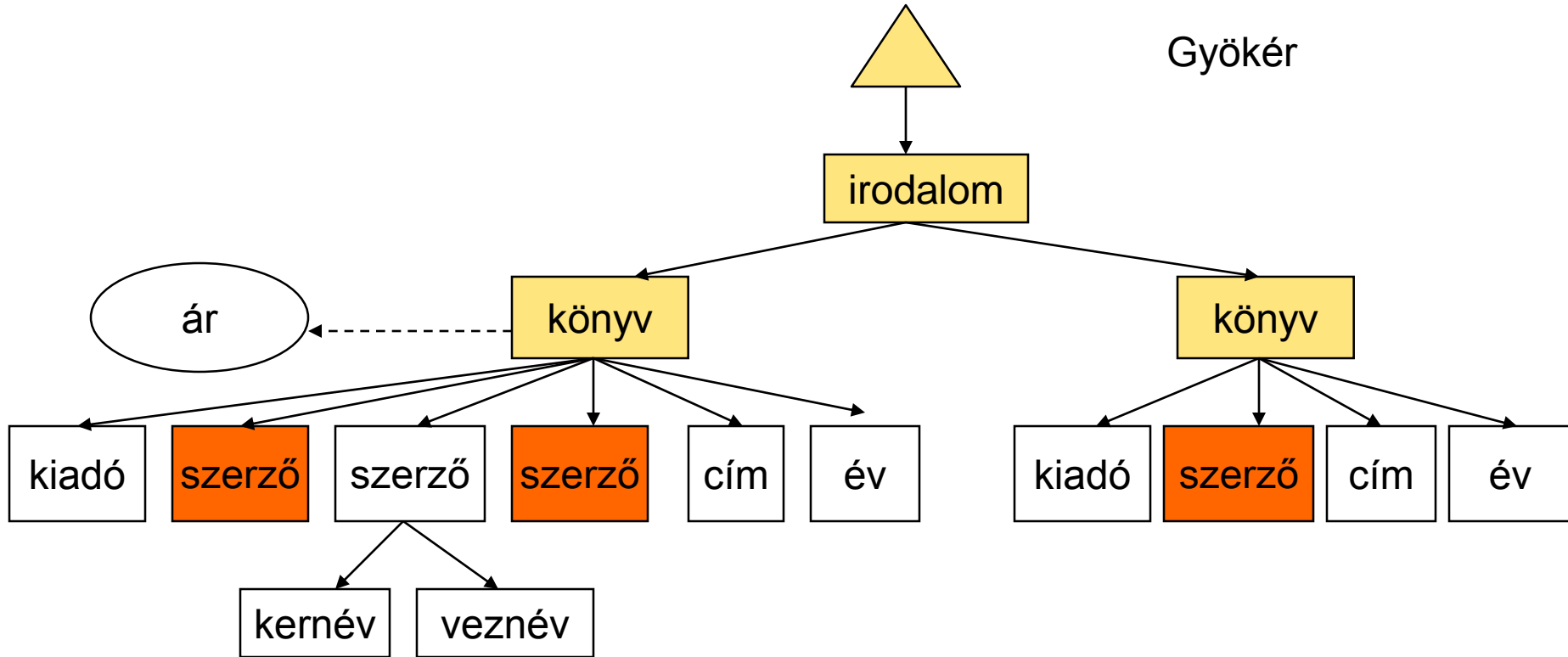
/irodalom//kernév



/irodalom//kernév

Eredmény: <kernév>Rick</kernév>

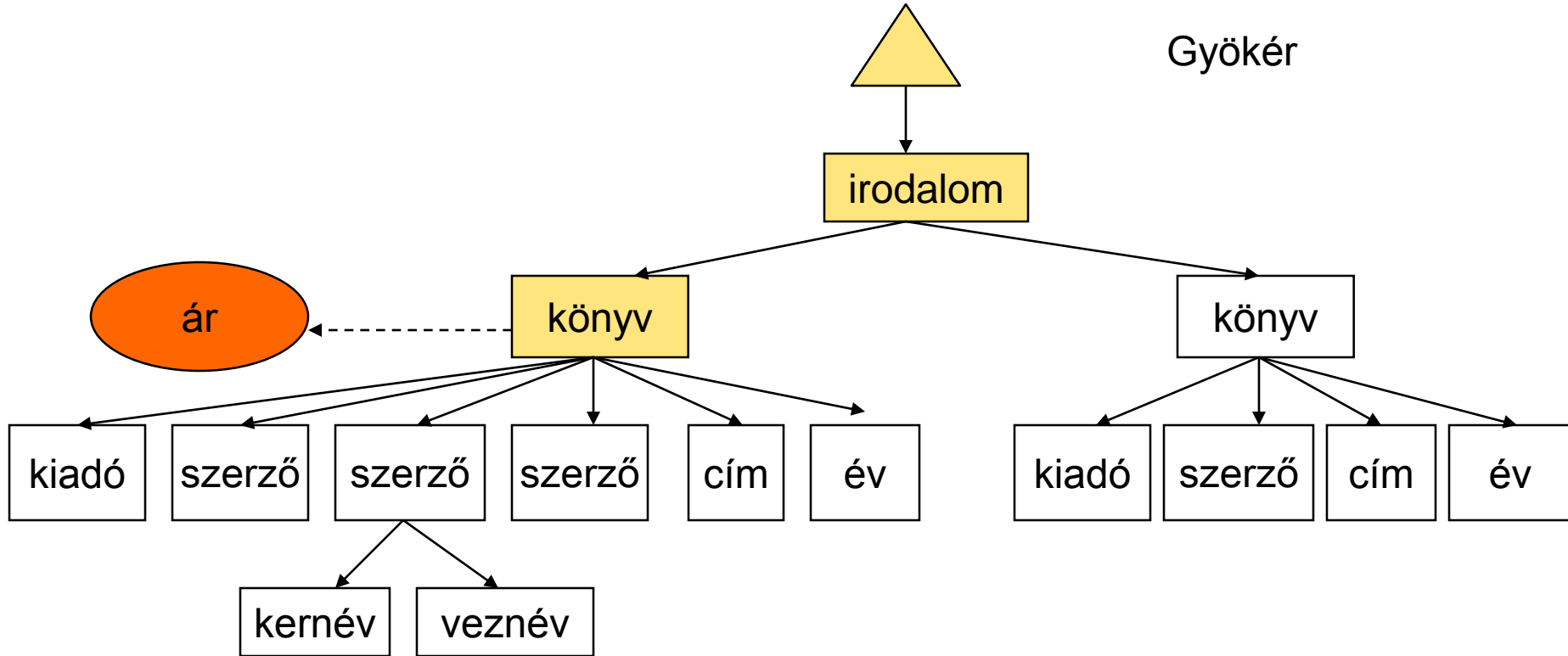
/irodaalom/könyv/szerző/text()



text() = visszaadja a szövegcsúcs értékét
node() = visszaadja csúcsot
name() = visszadja a jelölő nevét

/irodalom/könyv/szerző/text()
Eredmény: Serge Abiteboul
Victor Vianu
Jeffrey D. Ullman

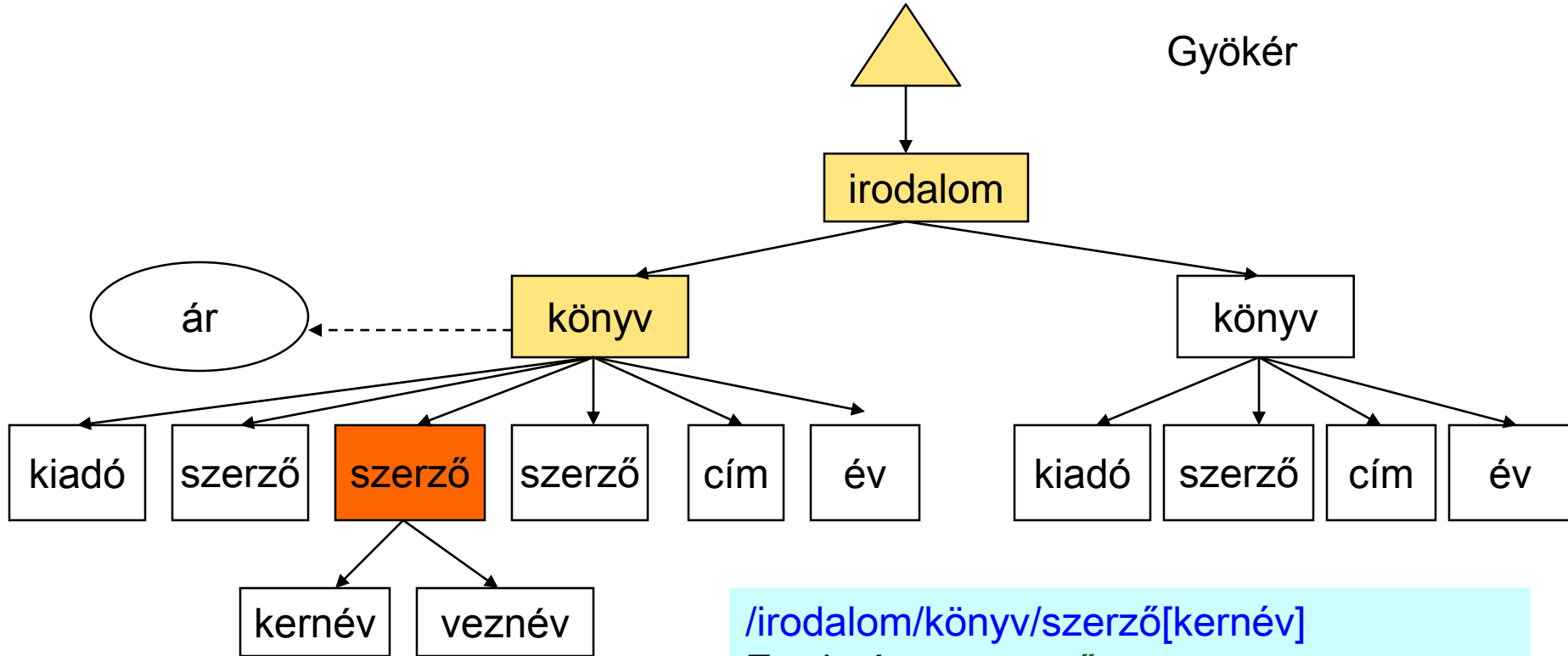
/irodalom/könyv/@ár



/irodalom/könyv/@ár

Eredmény: 55

/irodalom/könyv/szerző[kernév]



/irodalom/könyv/szerző[kernév]

Eredmény: <sz^{er}ző>

<kernév>Rick</kernév>

<veznév>Hull</veznév>

</sz^{er}ző>

[kernév] igaz, ha a *sz^{er}ző* elemnek van *kernév* gyereke.

További példák

/irodalom/könyv[@ár < 60]
az ára kisebb mint 60

/irodalom/könyv[szerző/kernév = "Rick"]
egyik szerzőjének kerneve Rick

/irodalom/könyv[szerző/text()]
van szöveggel megadott szerzője

/irodalom/könyv[2]
a 2. könyv



Halmazműveletek

- A következő halmazműveleteket lehet használni:
 - Egyesítés: $|$, union
 - Metszet: intersect
 - Kivonás: except
- Ezek a műveletek eltávolítják a többszörös előfordulásokat.

XPath Feladat

`http://people.inf.elte.hu/vopraai/korszeru_xml/xpath_feladat.txt`

`http://people.inf.elte.hu/vopraai/korszeru_xml/kolcs_onzesek.xml`

Ellenőrzéshez:

`http://www.xpathtester.com/xpath`