

Web 3.0

Szemantikus web

Tartalom

- Web 2.0
 - ...
 - Szoftver mint szolgáltatás
- Web 3.0 – szemantikus web

Szoftver mint szolgáltatás

- A **szoftvert mint szolgáltatást** adják el, nem mint terméket.
 - Pl., a termékfejlesztést össze kell hangolni *a napi működés során szerzett tapasztalatokkal*.
- A szoftver megszűnik hatékonyan működni, hacsak nem *napi szinten frissítik*.
 - Pl. a Google számára kötelező...
 - folyamatosan bejárni a Webet és frissíteni az indexeket,
 - folyamatosan szűrni a link-spam-et és minden más, a keresési eredményt befolyásoló dolgot,
 - folyamatosan és dinamikusan megválaszolni aszinkron felhasználói lekérdezés százmillióit,
 - egyidőben kontextusban illeszkedő hirdetéseket válogatni hozzá.
- A Google rendszeradminisztrátori, hálózati és terhelés kiegyenlítési technikái talán még sokkal inkább féltve őrzött titkok, mint maguk a kereső algoritmusok!

Szoftver mint szolgáltatás

- Google sikeressége ezen folyamatok automatizálásában egyértelműen kulcs előny a költséghatékonyságban a versenytársakhoz képest.
- *Szkipt nyelvek* (mint Perl, Python, PHP, Ruby) fontos szerepet játszanak a Web 2.0-ben
 - ...dinamikus rendszerek felépítésében, amelyek *folyamatos változást követelnek meg.*

Szoftver mint szolgáltatás

- *A felhasználókra mint társ-fejlesztőkre kell tekinteni!*
- A „**folytonos béta**” verzió: a termék fejlesztése a nyilvánosság előtt zajlik, új funkciók beillesztésével havi, heti vagy akár napi szinten (azaz „beta” akár évekig)
- *"Microsoft üzleti modellje szerint mindenki fejleszti a számítógépes környezetét 2-3 évenként.*
- *A Google arra épít, hogy mindenki azt szeretné látni, hogy mi változott a számítógépes környezetében nap mint nap."*

Web 3.0

szemantikus web

The-name-of-the-game...

„A **szemantika** ([magyarul jelentéstan](#), ám ez a terminológia az utóbbi időkben egyre kevésbé pontos) a [nyelvészet](#) egyik részterülete, amely a nyelvi formák (szavak, szimbólumok stb.) jelentésével illetve jelentésváltozásaival foglalkozik. Eredetileg csak szójelentéssel foglalkozott, de a [1960-as évek](#) óta, vagyis a [generatív nyelvészet](#) megjelenése óta a mondatjelentést is vizsgálja.

Hasonlóképpen a [formális nyelvek](#) elméletében a szemantika az adott formális nyelv [szintaxisa](#) által meghatározott [szavak](#) jelentését definiáló szabályok összessége.^[1]”

(Wikipedia)

Mi a Web 3.0?

- Az emberek nem tudnak megegyezni (...az elnevezésben, a definícióban, a létezésében)
- *„The Semantic Web (or Web 3.0) promises to organize the worlds’ information in a dramatically more logical way than Google can ever achieve with their current engine design.”*
(Marc Fawzi, Evolving Trends)
- *„Web 2.0 is a marketing term, and I think you’ve just invented Web 3.0”*
(Eric Schmidt, CEO of Google)
- *„If Web 2.0 was so hot, how about Web 3.0? This has been a recurrent theme of would-be meme-engineers who want to position their startup as the next big thing.”*
(Tim O’Reilly)

Mi lesz a Web 3.0?

- Milyen lehetne a Web 3.0?
 - Hogyan szerveződnek majd a saját adataink?
 - Mi fogunk „szörfölni”, vagy a gép csinálja helyettünk?
 - Ugyanúgy fog kinézni a Web számomra is, mint mindenki más számára?
 - Milyen technológiák lesznek közismertek? Melyek válnak elavulttá?

Szemantikus Web intro

- Alapdilemma:
 - Az Interneten meglévő tartalmak **struktúrálatlanok**.
 - Nincs egy olyan rendszer az egész felett, ami segítene: merre mit találunk, hol vannak hasonló tartalmak, stb.
- Megoldás(?): **szemantikus Web**
 - Internetre emelt **intelligens réteg**;
 - globálisan **skálázható Web**;
 - segít megtalálni a számunkra hasznos és érdekes **tartalmat**.
- *„A szemantikus web **célja** egy olyan **infrastruktúra** létrehozása, amely lehetővé teszi a Weben lévő **adatok integrálását**, a közöttük levő **kapcsolatok definiálását** és **jellemzését**, illetve az adatok **értelmezését**”*

Szemantikus Web (W3C def)

- *„The Semantic Web is about two things. It is about **common formats for interchange of data**, where on the original Web we only had interchange of documents. Also it is about **language for recording how the data relates to real world objects**. That allows a person, or a machine, to start off in one database, and then move through an unending set of databases which are connected not by wires but by being about the same thing.”*

(W3C – Semantic Web Activity - <http://www.w3.org/2001/sw/>)

Felhasznált irodalom

- Herman Iván, „Szemantikus Web: egy rövid bevezetés”, Magyarországi Web Konferencia, 2006.

A szemantikus Web felé...

- A jelenlegi Weben az információk *különböző formákban* állnak rendelkezésre:
 - természetes nyelveken (angol, magyar, kínai, holland, ...)
 - grafikákon, képeken, audió és videó formákban... stb.

- ***Emberek számára ez nem jelent igazán problémát!***

A szemantikus Web felé...

- A Weben gyakran van szükség adatok **kombinálására**, **koherens** módon
 - Pl.: a szállodai és az utazási adatok általában különböző forrásból származnak, habár együtt akarjuk őket használni
 - Pl.: valamely kutatásnak különböző digitális könyvtárak anyagaira van szüksége
 - *terminológiai különbségek!*

- *Ezt is könnyedén megtesszük; egy fogalomról másikra **asszociálni** nekünk egyszerű!*

Gépek, intelligens ágensek...

- ***A gépek buták!***

- **részleges információt** nem tudnak használni
- a képek **értelmezése** még mindig komoly kutatás tárgya
- **analógiákat** nehezen tudnak automatikusan megtalálni
- az adatok **kombinálása** is nehézkes
 - ugyanaz-e az **<abc:alkotó>** mint az **<cba:író>**?
- ...

Gyakorlati példák...

Példa 1: Keresés

- A Google túl sok a hamis/irreleváns találatot ad vissza
- Segítséget jelenthet, ha az adatforrásokhoz valamilyen további (esetleg alkalmazásfüggő) leírást lehetne hozzárendelni.

Gyakorlati példák...

Példa 2: Utazásszervezés

Egy automatikus utaztató rendszer, amely...

- **ismeri** a szokásaimat, kívánalmaimat;
- a múlt alapján további **tudást alakít ki** rólam;
- a helyi információt össze tudja **kombinálni** távoli információkkal, mint például:
 - légitársaság adataival
 - orvosi kérdésekkel, mint diétával, gyógyszerek hozzáférhetőségével
 - naptáradatokkal, állami vagy vallási ünnepek adataival
 - stb.

Gyakorlati példák... (folyt.)

Példa 3: Adatbázisok integrációja

- Adatbázisok **struktúrája**, tartalma nagyon különböző lehet
- Sok alkalmazás alapul adatbázisok kombinációján:
 - pl: cégösszeolvadások; biokémiai, orvosi, genetikai adatok; kormányzati és adminisztratív adatok
- Ezek az adatok legtöbbször a Weben vannak már (habár nem feltétlenül nyilvánosak)
- **Az adatok, adatbázisok *szemantikáját* kell ismerni ahhoz, hogy kombinálhatók legyenek**
 - Az, hogy a szemantika hogy képződik le a konkrét adatbázisra, voltaképpen mellékes.

Mire van szükség?

- Az adatokat gépi kezelésre is elérhetővé kell tenni
 - egyes esetekben az adat más adatokat ír le (mint a keresés esetén): ezek az ún. **metaadatok**
 - máskor magát az adatokat kell kombinálni, például a naptáram vagy utazási szokásaim esetén
- Az adatokat össze kell tudni olvasztani, **kombinálni**, és mindezt a Web nagyságrendjén
- A gépeknek ***következtetéseket is le kell tudnia vonni az adatokról*** (például hogy a használt terminológia azonos szemantikát takar...)

Mire van szükség (technikailag)?

- A **szemantikus Web** célja egy olyan **infrastruktúra** létrehozása, amely...
 - lehetővé teszi a Weben lévő adatok **integrálását**,
 - a közöttük levő **kapcsolatok definiálását** és jellemzését,
 - az adatok **értelmezését**.

Szemantikus Web technológiák

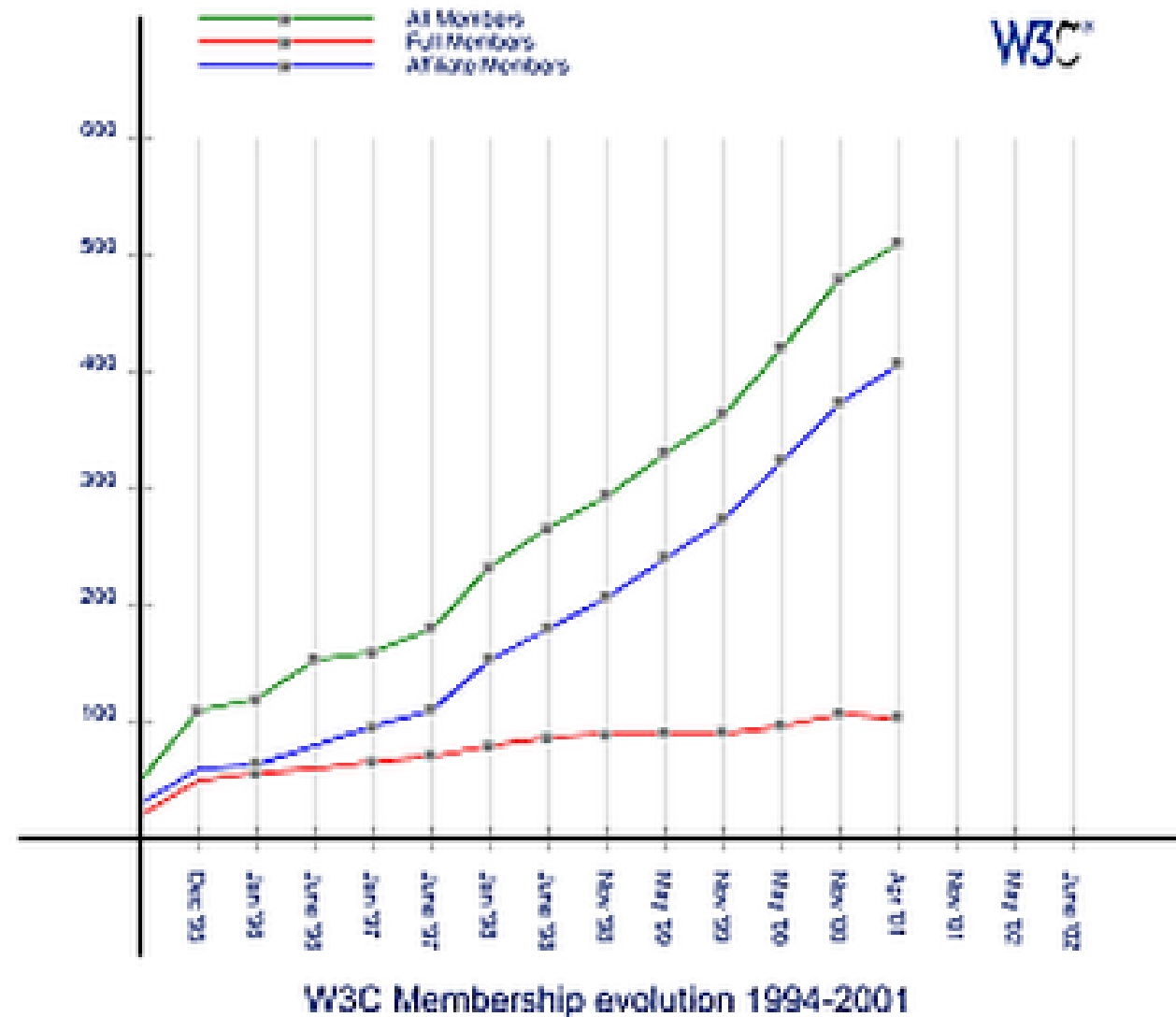
- Mindehhez szükség van:
 1. az erőforrások egyértelmű elnevezésére (azaz **címzése**): ???
 2. az adatok összekapcsolására, leírására szolgáló általános **adatmodellre**: ???
 3. az adatok a modell alapján való elérésére, **lekérdezésére**: ???
 4. a **közös terminológia** definíciójára: ???
 5. logikai **következtetési rendszerekre**: ???

Szemantikus Web technológiák

- Mindehhez szükség van:
 1. az erőforrások egyértelmű elnevezésére (azaz **címzése**): **URI**
 2. az adatok összekapcsolására, leírására szolgáló általános **adatmodellre**: **RDF**
 3. az adatok a modell alapján való elérésére, **lekérdezésére**: **SPARQL**
 4. a **közös terminológia** definíciójára: **RDFS, OWL, SKOS**
 5. logikai **következtetési rendszerekre**: **OWL, Rules**
- Megjegyzés: Az URI nem szemantikus Web-specifikus, hanem egy jól elterjedt korábbi megoldás. A többi (RDF, OWL, ...) a szemantikus Webre lettek kifejlesztve.

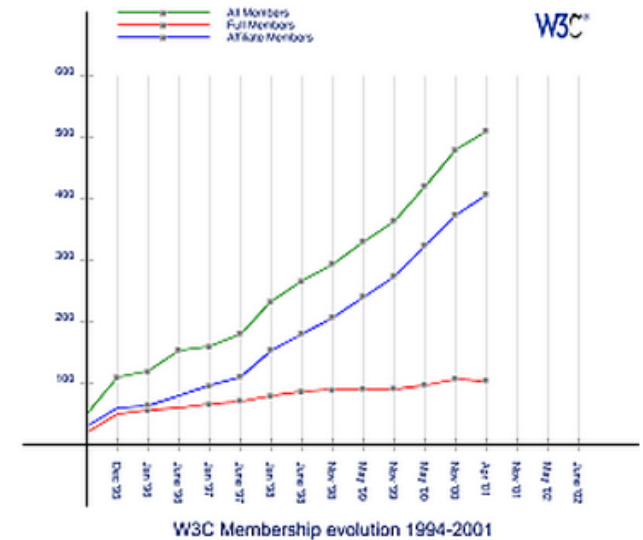
Egy példa...

- Egy „egyszerű” üzleti grafikon a Web-en: a tartalma „világos”, látszólag nem igényel további magyarázatot.
- DE ha ezt az ábrát egy vak felhasználó (pl. ágens?) számára szeretnénk elérhetővé tenni?...



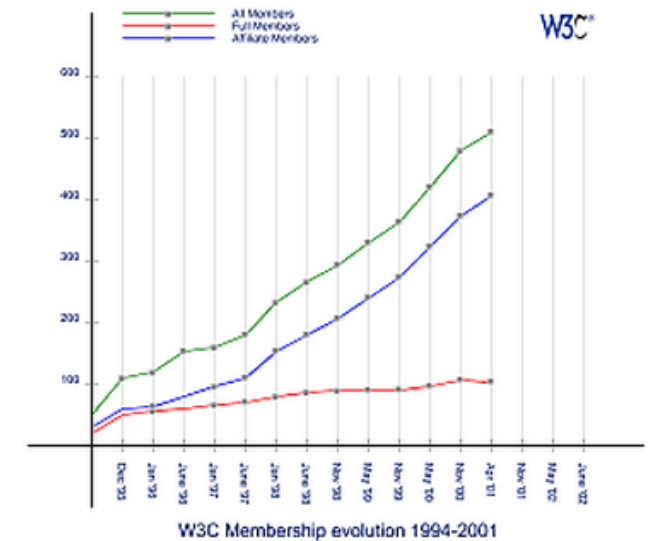
Egy példa (folyt.)

- A kép tartalmát szövegesen is le szeretnénk írni (fontos az elérhetőséghez):
 - adjunk meg **metaadatot**, amely leírja a képet
 - legyen egy *eszköz*, amely (a metaadat alapján) egyszerű *kimenetet* állít elő
 - használjunk *szabványosított metaadat formalizmust*
- Megjegyzés: Az ábra SVG-ben (vagyis egy grafikus XML leírás), így a grafika minden eleme megcímezhető egy URI segítségével!



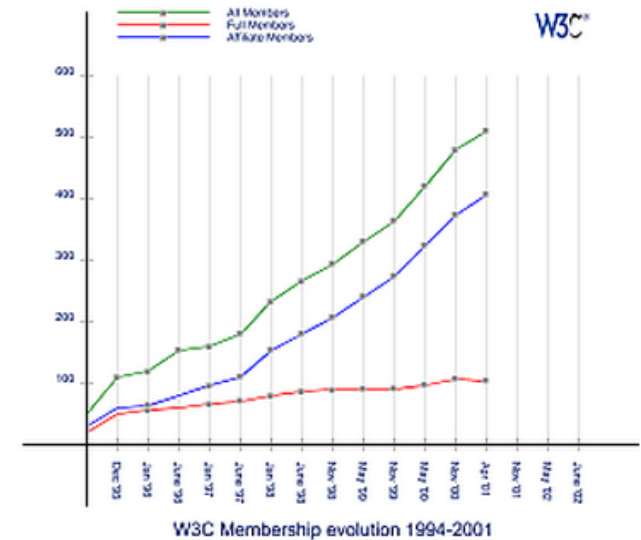
Egy példa (folyt.)

- A metaadat **állítások** sorozata:
 - „a teljes ábra típusa egy grafikon, a grafikon «vonalas» típusú”
 - „a grafika felirata egy (SVG) szövegelem”
 - „a jelmagyarázat egyben egy hiperlink”
 - „a hiperlink erre és erre a URI-ra mutat”
 - „a teljes ábra a jelmagyarázatból, tengelyekből és adatgörbékből áll”
 - „a görbék a teljes jogú és társult tagokat, illetve az összes tagot ábrázolják”



Egy példa (folyt.)

- Az állítások **hármás tagolást** (triplet) mutatnak:
 - *Vmről állítunk* valamit (pl. „üzleti grafika”)
 - *összekapcsoljuk* egy szövegelemmel (pl. „vonalgrafika”) vagy külső adattal (URI)
 - maga a *kapcsolat* is *szemantikával* bír (pl. „típusa”).



RDF

- RDF – (erő)forrásleíró keretrendszer (Resource Description Framework)
- Az **állításokat** a következő módon modellezhetjük:
 - **Forrás:** egy elem, egy URI, literál, ...
 - **Kapcsolat:** *irányított* reláció két forrás között
 - **Állítás:** két forrás egy őket összekapcsoló relációval
- *RDF* az ilyen típusú **állítások általános modellje**
 - kifejezhető XML-ben, de más szintaxissal is (pl. n3)

RDF hármások

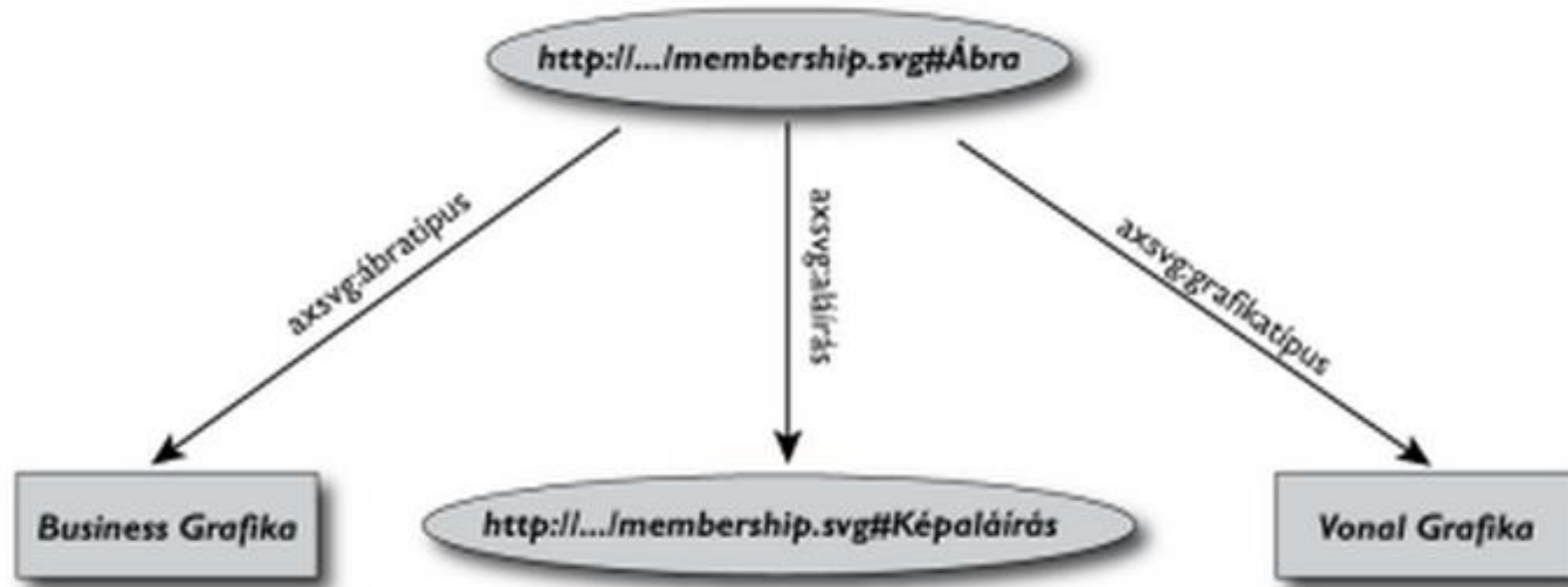
- Az angol terminológia:
 - „*triplets*”, „*triples*”, vagy „*statement*”
 - magyarul: „hármás”, vagy „állítás”
 - „*subject*”, „*predicate*” vagy „*property*”, „*object*”
 - magyarul: „alany”, „állítmány” vagy „tulajdonság”, és „tárgy”
- Egy RDF hármás (s,p,o):
 - „s”, „p” URI-k; „o” egy URI vagy egy literál
 - jelentése: a „p” összekapcsolja az „s”-t az „o”-val
 - az elnevezések/cimkék eszközei szintén a URI-k: `http://.../abra.svg`
 - Példa: íme a teljes hármás:

(`http://.../abra.svg; http://.../tipus; „vonalgrafika”`)

RDF hármások (folyt.)

- *Bármely* URI használható; vagyis egy XML fájl-ba is lehet címezni, nemcsak a teljes anyagra, pld:
 - `http://www.example.org/file.xml#xpointer(id('naptár'))`
 - `http://www.example.org/file.html#naptár`
- RDF a hármások általános modellje:
 - lényegében egy **irányított, címkézett gráf**
 - *csomópontok: alanyok és tárgyak*
 - *élek: állítmányok (tulajdonságok)*
 - **gépileg olvasható** formátumokkal (RDF/XML, Turtle, n3, RXR, ...)
 - **RDF/XML** a „hivatalos” XML alapú formátum
 - Nem a szintaxis a fontos, hanem a mögötte álló adatmodell!

Egy egyszerű RDF példa (RDF/XML)



```
<rdf:Description rdf:about="http://...#Ábra">
```

```
  <axsvg:ábratípus xml:lang="hu">Business grafika</axsvg:ábratípus>
```

```
  <axsvg:aláírás rdf:resource="http://...#Képaláírás"/>
```

```
  <axsvg:grafikatípus xml:lang="hu">Vonal  
grafika</axsvg:grafikatípus>
```

```
</rdf:Description>
```

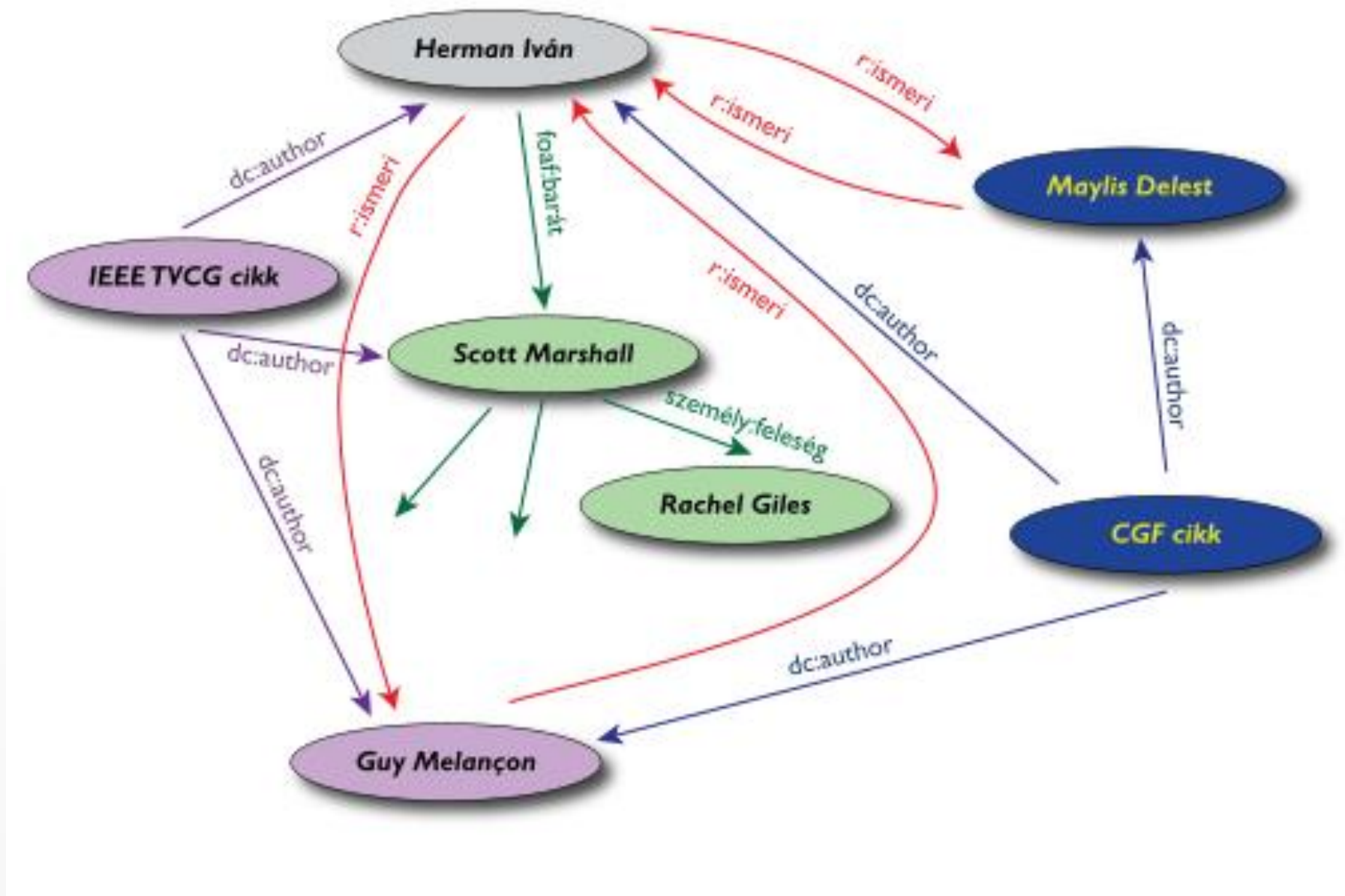
A URI-k alapvető szerepe

- RDF állítások *bármilyen* adatra vonatkozhatnak a Weben.
- A forrásokat *egyértelműen* azonosítják.
- *Bárki* kreálhat (meta)adatot *bármely* Web–erőforrásról, nem csak annak létrehozója.
 - Pl.: ugyanazt az XML–alapú állományt le lehet írni egymástól eltérő terminológiákkal.
- *Végeredményben: A URI-k „ágyazzák” az RDF-et a Webbe*
 - *így lesz a „Szemantikus Web”... „szemantikus web”* 😊

URI-k: összevonás

- URI-k használatának jelentős előnye: könnyűvé válik az adatok (logikai) **összevonása**
- Az összevonás megtehető az *azonos* URI-k alapján
 - Egy gráfban: azonos URI-val rendelkező csomópontok egymással azonosíthatóak → triviális módon összevonhatóak.
- Ez az összevonás az RDF–modell *nagyon* fontos jellemzője
 - a leírásokat **különböző személyek**, csoportok hozhatják létre, de ...
 - ...az alkalmazás **egységként kezelheti** őket
 - egyike azon területeknek, ahol az RDF–modell sokkal könnyebben használható, mint az XML

Példa az összevonásra



RDF megvalósítása

- A gyakorlati alkalmazások milliós nagyságrendben kezelnek RDF hármassokat!
 - → szükség van nagy **adatbázisokra**
 - → szükség van **lekérdező nyelvre** bonyolultabb információk lekérdezésére
- Adatbázisok
 - Többféle RDF adatbázis rendszer létezik
- **Lekérdező nyelv**
 - Az adott adatmodellhez kapcsolódik
 - Pl.: SQL a relációs adatmodellhez, XML Query az XML adatmodellhez
 - RDF-hez: **SPARQL** (**SPARQL Protocol and RDF Query Language**)

Lekérdezések: SPARQL

- Az alapvető ötlet: Egy lekérdezés egy *gráfmintát* tartalmaz:

```
SELECT ?név
```

```
WHERE {
```

```
    ?x abc:naptáram ?y.
```

```
    ?x abc:személynév ?név.
```

```
}
```

- Vagyis, körülbelül: *„Add meg mindazoknak a nevét, akiknek a naptára a Weben van”*
- A specifikáció még nem teljes, de már nagyon sok implementáció és alkalmazás létezik...

Az RDF nem elegendő...

- A kapcsolatok létrehozása és programból való használata működik, feltéve, hogy a program *tudja*, hogy milyen **terminológiát** használhat!
- Például használtuk a következő fogalmakat:
 - naptáram, családnév, személynév, ...
- Ismertek-e ezek? Korrektek-e?
 - A probléma egy kicsit hasonló egy adatbázis rekordtípus definiálásához

Megoldandó kérdések

- Mely terminológiák, szavak használhatók? Ismert-e a **terminológia**?
- Korrekt módon használjuk-e a tulajdonságokat? Van-e értelmük az adott erőforrások esetén?
- Lehet-e következtetéseket levonni? Például:
 - *„ha »A« »B«-től balra van, »B« »C«-től balra van, akkor balra van-e »A« »C«-től?”*
 - *nekünk nyilvánvaló, de egy programnak nem ...*
 - *... vagyis: levonhatják-e a programok ezeket a következtetéseket?*
- Ha valaki más definiál egy állításhalmazt: ugyanaz-e, mint a mienk?

Ontológiák

- A Szemantikus Webnek szüksége van **ontológiákra**:

Def: „**Ontológia**: Egy adott tudásterület leírására használt fogalmak és összefüggések definíciója”

Ontológiák

- Szükség van egy **Webontológia nyelvre**, amellyel definiálni lehet:
 - az adott kontextusban **használható fogalmakat**
 - a tulajdonságokra érvényes **korlátozásokat**
 - a tulajdonságok **logikai jellemzőit**
 - a fogalmak és tulajdonságok **ekvivalenciáját** (vagy különbözőségét)
 - stb.
- Az erre szolgáló specifikációk:
 - **RDFS** (RDF **Sémák**) – RDF szókészlet leírónyelv
 - **OWL** (**Webontológia nyelv**)

Osztályok, erőforrások

- Gondoljunk az ismert, tradicionális ontológiákra:
 - ismerjük az „emlős” fogalmát (valahonnan)
 - „minden delfin emlős”
 - “Flipper egy delfin”
 - stb.
- Az RDFS definiálja az **erőforrás** és az **osztály** fogalmát:
 - az *RDF* számára minden egy „erőforrás”
 - egy *osztály* erőforrások („egyedek”) lehetséges összessége
 - Pl: „emlős”, „delfin”, ...

Osztályok, erőforrások

- Az erőforrások és az egyedek között **relációk** létesíthetők:
 - „**típus**” („*typing*”): vagyis **egy egyed egy adott osztályhoz tartozásának jelzése** („*Flipper egy delfin*”)
 - „**alosztály**” („*subclassing*”): **reláció két osztály között**: az egyik osztály egyedei automatikusan a másiknak is egyedei („*minden delfin emlős*”)
- Az *RDFS* ezeket a (tradicionális) fogalmakat formalizálja

RDFS

- Az RDFS fogalmak és definíciók RDF-ben vannak megfogalmazva.
 - Az „osztály” egy speciális *erőforrás*, meghatározott URI-val
 - A „típus” és „alosztály” egy-egy RDF *tulajdonság*...
 - ...jól meghatározott és *szabványosított szemantikával*.

- Például:

```
(http://...#emlős, rdf:type, rdfs:Class)
```

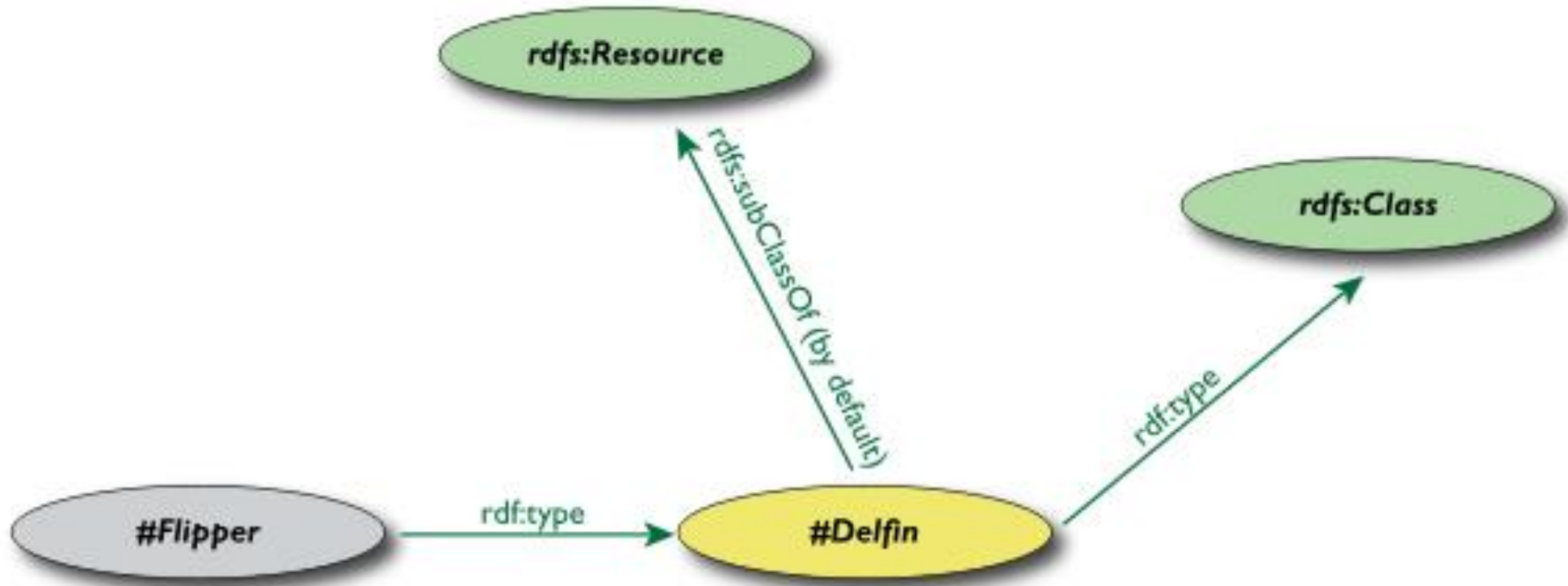
```
(http://...#delfin, rdf:type, rdfs:Class)
```

```
(http://...#delfin, rdfs:subClassOf, http://...#emlős)
```

```
(http://...#Flipper, rdf:type, http://...#delfin)
```

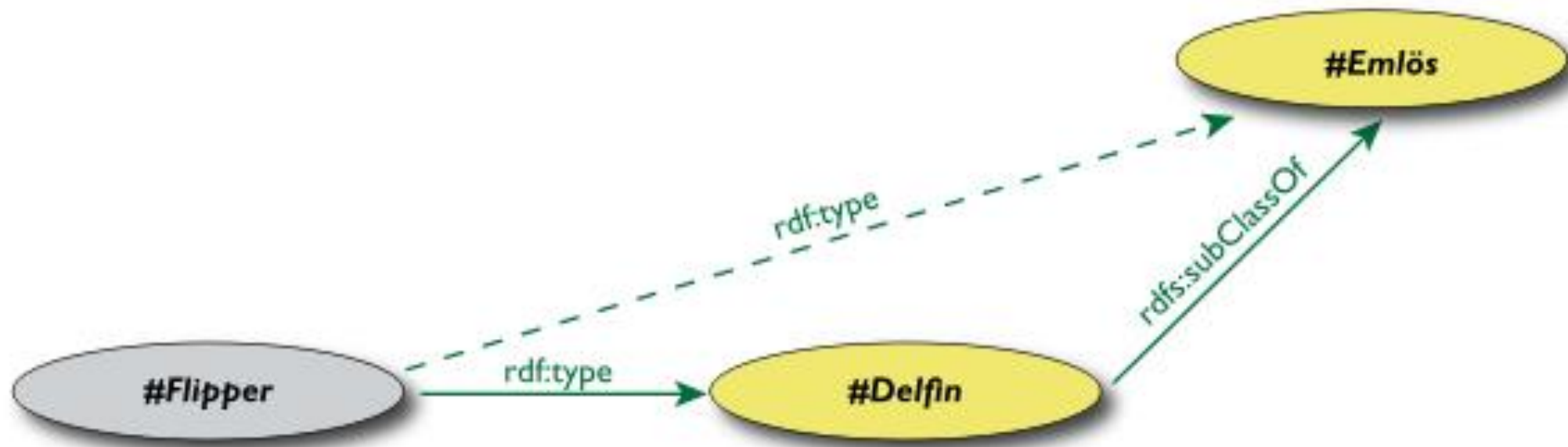
Osztályok, erőforrások (RDF)

- Az RDFS definiálja a `rdfs:Resource`, `rdfs:Class`, `rdf:type`, `rdfs:subClassOf` fogalmakat
 - *(ezek mind speciális, az ábrán névterekkel rövidített URI-k)*



Következtetett tulajdonságok

- (`#Flipper rdf:type #Emlős`) *nem* része az eredeti RDF adathalmaznak...
- ...de ki lehet *következtetni* az RDFS szabályokból
- Jobb RDF környezetek ezt az állítást is tartalmazzák

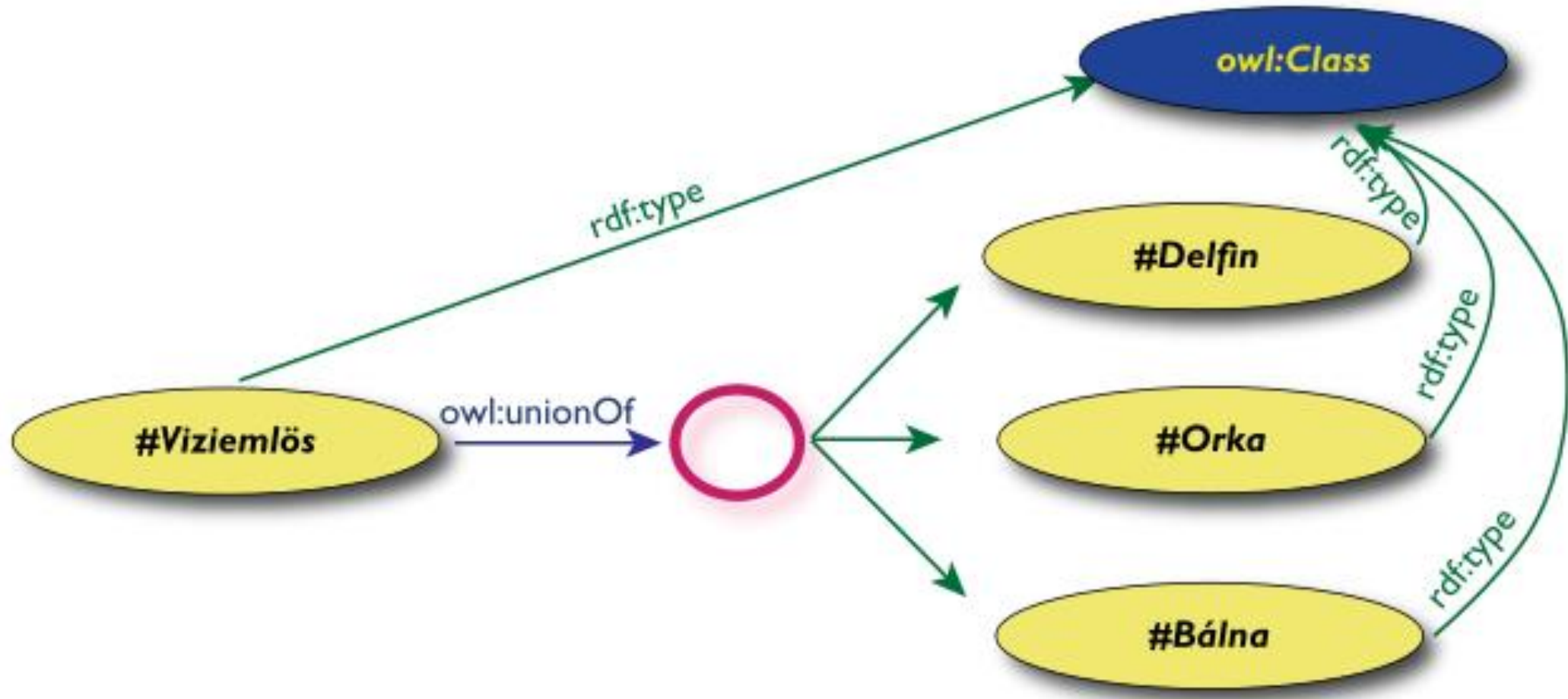


RDFS és OWL

- Az RDFS az ontológiák definiálásának *alapszintje* csak!
- → Komplikáltabb definíciók esetén szükség van a következő szintre: **Web Ontology Language (OWL)**
- Az OWL hozzáad bonyolultabb lehetőségeket, mint például:
 - **osztályok konstrukciója** (a meglévő osztályokból kiindulva)
 - unióként, metszetként, komplementként, stb.
 - egyedek felsorolásával
 - tulajdonságok segítségével
 - **a tulajdonságok logikai jellemzése** (pld. tranzitivitás, szimmetria, függvény)
 - stb.

OWL: osztályok úniója

- Lényegében egy **halmazelméleti únió** (lehetne metszet, komplement, stb):



OWL: osztály definíciója tulajdonságokból

- Delfin ontológia: „*A delfin olyan emlős amely vízben él.*”
- A fenti kijelentés a delfinek osztályát definiálja:
 - Vegyük az *emlősök osztályát*.
 - Vegyük az *él tulajdonságot*.
 - Tekintsük az *emlősök osztályának azon egyedeit*, amelyekre az *él tulajdonságot* alkalmazva a hármasként a víz.
- Ez az **osztály konstrukció** az OWL-ban az ún. **értéktartomány-korlátozás**.

OWL: tulajdonságok logikai jellemzése

- Tulajdonságokat lehet *szimmetrikusnak, tranzitívnek, függvénynek* stb. **deklarálni**.
- Ezen definíciók alapján egy „OWL-t értő környezet” egy sor logikai **nem triviális következtetést** képes levonni!
 - Pl.: A „balra van” tulajdonság tranzitivitásának definiálása, majd következtetés levonása: „Ha »A« »B«-től balra van, »B« »C«-től balra van, akkor »A« »C«-től balra van”
 - Nekünk nyilvánvaló, de egy programnak nem!

Az OWL további lehetőségei

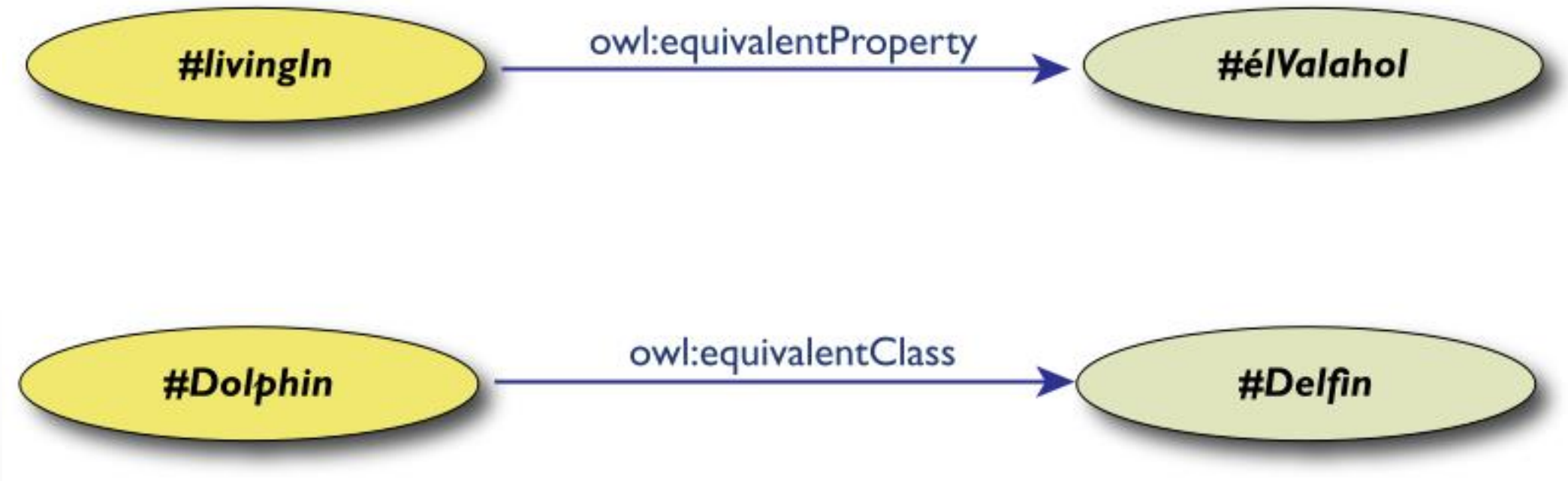
- Az ontológiák nagyon nagyok lehetnek:
 - nagy figyelmet kell fordítani a karbantartásukra
 - több részből (modulból) állhatnak
 - a részeknek különböző eredetük lehet, melyeket integrálni kell
- Ezek **Webontológiák**. Vagyis
 - az alkalmazások több, egymástól **különböző ontológiát** használhatnak, vagy...
 - ... ugyanazon ontológiát, de **különböző nyelveken**
 - vagyis a **terminológiák ekvivalenciája** fontos kérdéssé válhat

Az OWL további lehetőségei

- OWL lehetőséget ad az osztályok/tulajdonságok *ekvivalenciájára*, verziókontrollra, stb.

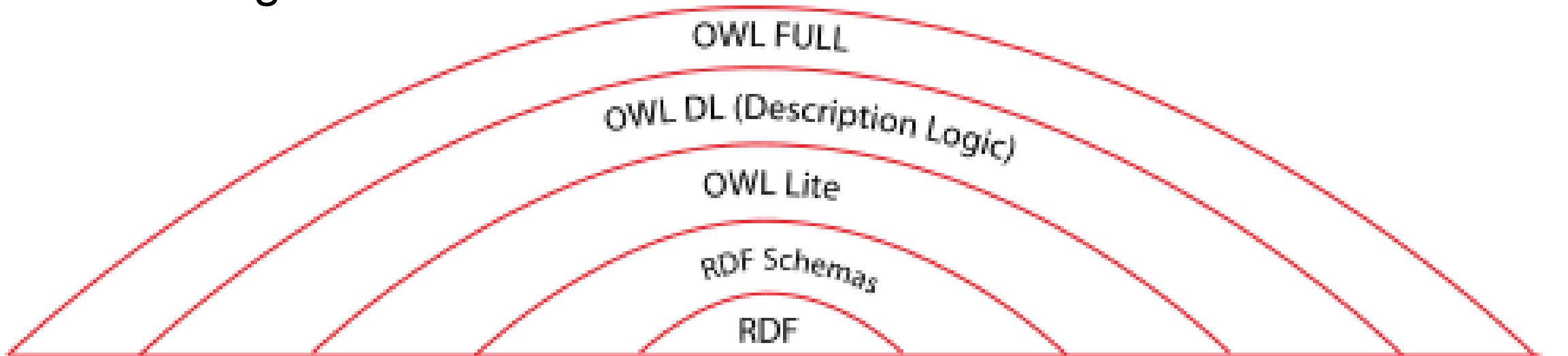
```
(http://...#delfin, owl:equivalentClass,  
http://...#dolphin)
```

Példa: kapcsolat az angol és a magyar között



De: az ontológiák bonyolultak!

- Nehéz egy teljes ontológiarendszert implementálni
 - és egyes alkalmazások számára felesleges is lehet
- Innen az egyre bonyolultabb specifikációk „réteges” modellje, különböző megkötésekkel
- De: az RDFS, OWL-Lite és OWL-DL *kiszámítható*, míg ez nem igaz OWL Full-ra



OWL fejlesztési irányok

- Az OWL nem teljes, nem mindent lehet vele leírni, pl...
 - nem képes valószínűségeket kifejezni: „ez a kapcsolat adott valószínűséggel áll fenn”
 - analóg, fuzzy kapcsolatok kifejezése hiányzik
- Bizonyos következtetési láncok definiálása és kifejezése sem lehetséges.

(Dávid, apja, Iván) (Iván, testvér, János)

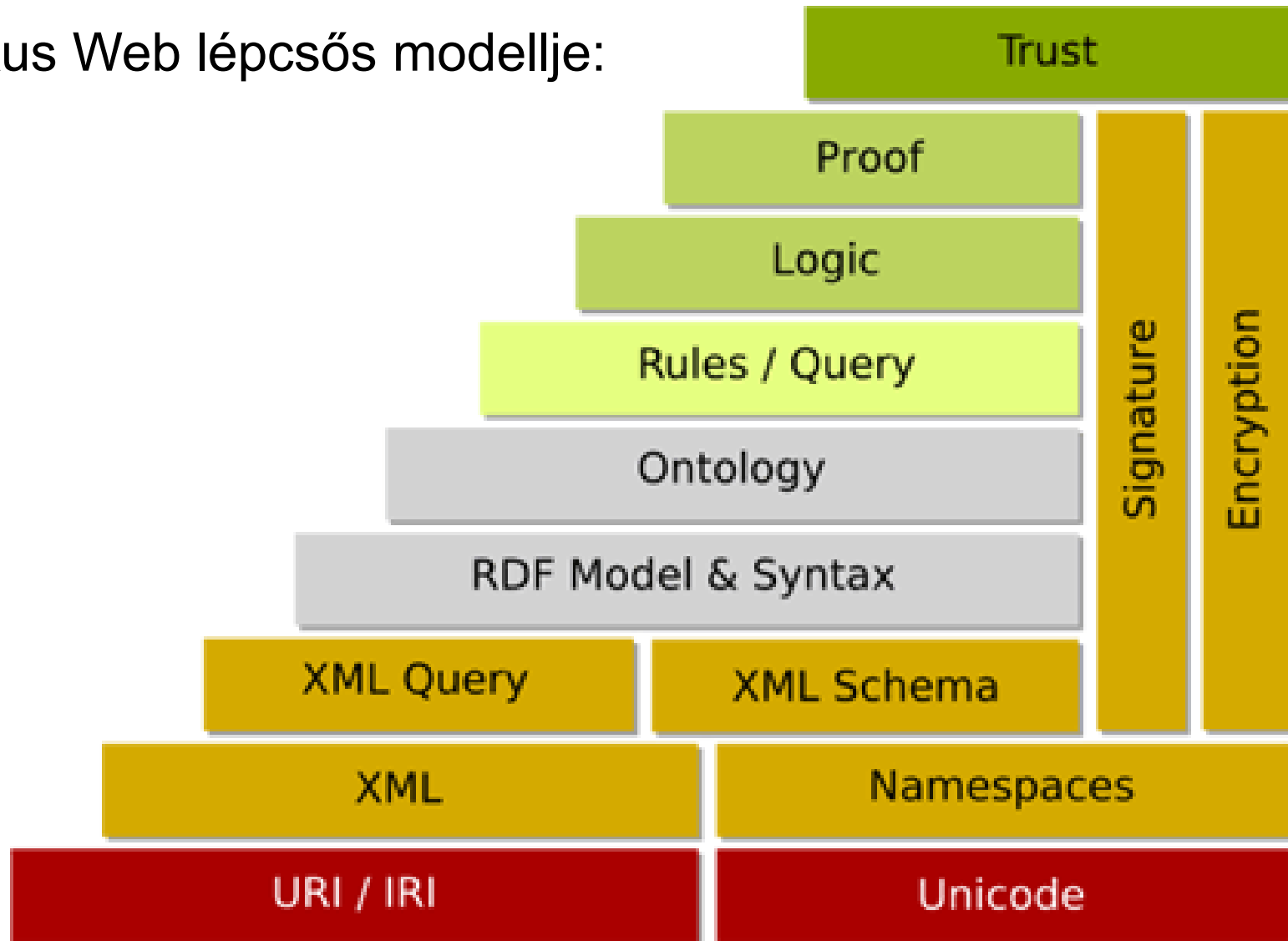
- akkor szeretnénk, ha a rendszer a következő hármásra is következtetne:

(János, nagybácsi, Dávid)

- Sajnos egy ilyen szabályt nem lehet az OWL-ban leírni, ez az ún. Horn következtetési szabályokat igényli, amely másfajta logikákhoz kapcsolódik.

A szemantikus Web öszsképe

- A szemantikus Web lépcsős modellje:



Szemantikus web lépcsős modell

- Unicode és URI
 - biztosítják az egységes nemzetközi karakterformátumot és az egységes címezhetőséget.
- XML réteg
 - biztosítja, hogy a szemantikus web definíciókat egyéb XML alapú szabványokkal integrálhassuk. Önleíró, érvényesíthető dokumentumformátum jön létre.
- RDF és RDFS réteg
 - biztosítja, hogy állításokat tehessünk az objektumokról URI-k segítségével, és olyan szótárakat készítsünk, amelyekre ezen URI-k hivatkozhatnak.
- Az ontológiai réteg
 - lehetőséget ad bonyolultabb szótárak felépítésére. Itt már a különböző fogalmak közötti összefüggések is megadhatók.
- A digitális aláírás réteg
 - a dokumentumok hitelességét igazolja.

Alkalmazási példák

szemantikus web

Alkalmazási példák

1. Közösségi webportálok
2. Multimédia gyűjtemények
3. Vállalati webportál-kezelés
4. Web 3.0 keresőmotorok
5. Ágensek és szolgáltatások

<http://www.w3c.hu/forditasok/OWL/REC-webont-req-20040210.html>

1. alkalmazás: Közösségi webportálok

- A webportál egy olyan webhely, mely
 - **információtartalmat publikál** valamilyen közös témáról,
 - a tartalmat tipikusan a **közösség tagjai szolgáltatják...**
 - akik gyakran **indexelik** is ezeket valamilyen téma alatt,
 - de ez még kevés...

1. alkalmazás: Közösségi webportálok

- A webportálok definiálhatnak egy **ontológiát** a közösség számára, amely...
 - egyrészt terminológiát nyújthat a tartalom leírására,
 - másrészt axiómákat bocsáthat rendelkezésre, amelyek...
 - új fogalmakat definiálhatnak az ontológia más fogalmainak felhasználásával. Pl:
 - terminológia: "újságcikk", "publikáció", "személy,, "szerző,,
 - definíciók: "minden újságcikk publikáció", "minden publikáció szerzője ember,,
 - Ezek a definíciók lehetővé teszik olyan újabb tények kikövetkeztetését, amelyek szükségszerűen igazak.
 - Ezek a következtetések azután segítik a felhasználókat abban, hogy olyan találatokat kapjanak, amelyek nem lennének elérhetőek a hagyományos kereső rendszerekkel.

2. alkalmazás: Multimédia gyűjtemények

- A gépek számára még a természetes nyelvű szövegekből történő információszerzésnél is nehezebb feladat az értékes szemantikai tartalom kiemelése a multimédiás anyagokból.
- Ontológiák segítségével **szemantikai annotációkkal** láthatjuk el a képek, hangok és egyéb, nem szöveges objektumok gyűjteményeit.

2. alkalmazás: Multimédia gyűjtemények (folyt.)

- A multimédia ontológiák kétféle típusba sorolhatók:
 - média-specifikusak, vagy
 - tartalom-specifikusak.
- A média-specifikus ontológiák
 - taxonómiaszerűen osztályozhatják a médiatípusokat, és
 - leírhatják a különböző médiumok tulajdonságait. (Pl. videó hossza)
- A tartalom-specifikus ontológiák
 - leírják a forrás témáját (pl. szereplők).

2. alkalmazás: Multimédia gyűjtemények (folyt.)

- Példa: antik bútorokat ábrázoló kép-archívum
- Egy antik bútorokra vonatkozó ontológia segíthetné az ilyen archívumban való keresést!
- **Taxonómia:** különböző bútor típusok osztályozása.
- Az ontológia **tárgyi tudást** is ábrázolhat.
 - Pl. az indexelő személy, egy antik komóddal kapcsolatban a *stílus/korszak* tulajdonsághoz a "késő georgiánus" értéket rendeli.
 - Ez lehetővé teszi annak **kikövetkeztetését**, hogy a "készítés ideje" tulajdonság érték valahol 1760 és 1811 közötti, valamint, hogy a bútor a brit kultúrához tartozik.

2. alkalmazás: Multimédia gyűjtemények (folyt.)

- A háttér-információk rendelkezésre állása nagymértékben megnövelheti a keresés hasznosságát.
- Egy másik opció az "alapértelmezett tulajdonságok" ábrázolása.
 - Pl: A "késő György-korabeli komódok", egyéb információ hiányában úgy tekintendők, hogy mahagóni fából készültek.

3. alkalmazás: Vállalati webportál

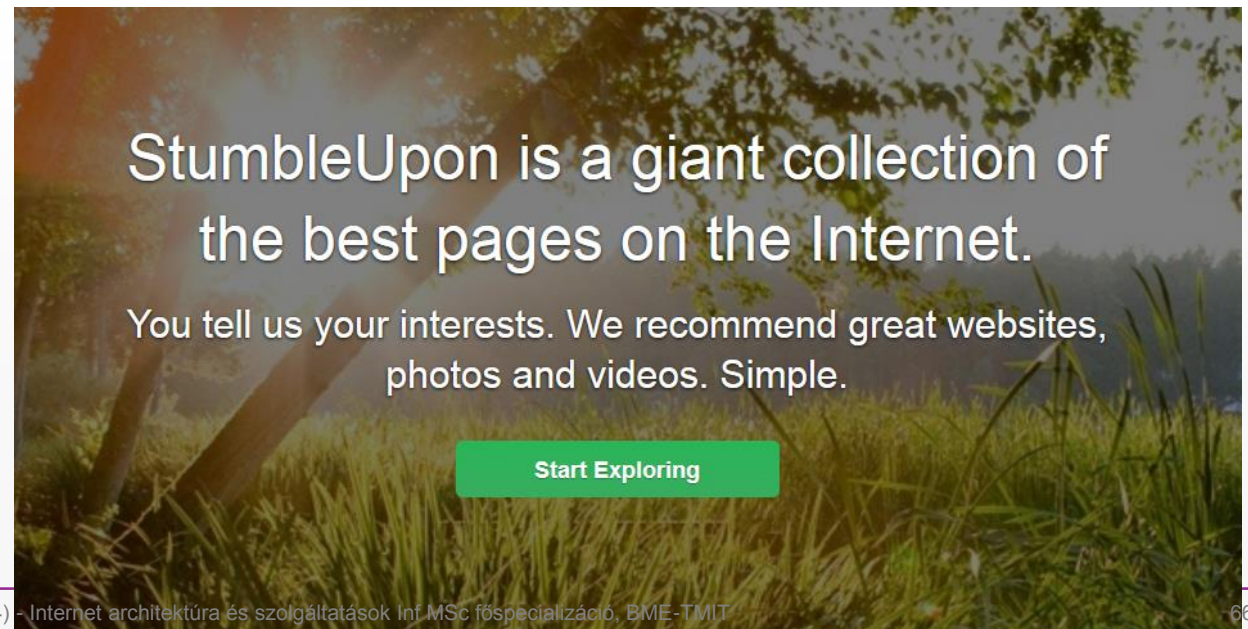
- A nagyobb cégek jelentős számú weblapot használnak
 - Pl.: sajtónyilatkozatok, áruajánlatok, esettanulmányok, ügyintézési eljárások, belső produktumismertetőik és összehasonlítások, brosrák és folyamatleírások. ..
- A felhasználó tipikus problémája, hogy ő esetleg más terminológiát használ:
 - egy cégen belüli kereskedő, aki olyan reklám brosrát keres, mely megfelel az eladási céljainak.
 - egy cégen belüli műszaki ember, aki meghatározott műszaki ismeretek és korábbi, részletes tapasztalatok után kutat.
 - egy projektvezető, aki a cég korábbi tapasztalatából merítve mintát keres egy bonyolult, többfázisú projekt előkészítéséhez és végrehajtáshoz.

3. alkalmazás: Vállalati webportál

- Hasznos lenne, ha...
 - a felhasználók minden kategóriája egy **külön taxonómiával** rendelkezne, mely megfelel a saját szaknyelvének, és
 - ezek a taxonómiák **összekapcsolhatók** lennének, hogy a fordítás **automatikusan** megtörténhessék.

4. alkalmazás: Web 3.0 keresőmotorok

- Szociális és nyelvi keresők
 - Személyre szabott keresések
 - Érdeklődési körnek megfelelő találatok
 - Ajánlatok a többi „hasonló” felhasználótól
 - Csak az érdekes oldalak megjelenítése
 - A web eldugott, de számunkra releváns részének felderítése és megmutatása
 - Pl.: StumbleUpon



5. alkalmazás: Ágensek és szolgáltatások

- A szemantikus Web lehetővé teszi az **ágenseknek** (intelligens ügynökprogramoknak), hogy *megértsék* és *integrálják* a különböző forrásokból származó információkat.
- PI: szabadidő tevékenységet tervező ágens
 - A felhasználó preferenciáiból kiindulva (pl., hogy milyen filmeket kedvel, milyen ételeket fogyaszt szívesen) megtervezi annak esti programját.
 - Megvizsgálhatja a filmekre és éttermekre vonatkozó elégedettségi statisztikákat és kritikákat is.

5. alkalmazás: Ágensek és szolgáltatások (folyt.)

- Ehhez szükség van:
 - egy **ágazati ontológiára**
 - amely az éttermek, szállodák stb. hálózatát ábrázolják
 - egy **szolgáltatási ontológiára**
 - mely azokat a fogalmakat reprezentálja, amelyeket az egyes szolgáltatásoknál használnak.

Összegzés

- A WEB 3.0 még nem a mesterséges intelligenciáról szól
- Habár jobb és okosabb alkalmazásokat lehet rá építeni, mint a WEB 2.0-ra
- A távoli jövő
 - A WebOS idősza
 - Intelligens és személyes rendszerek
 - A.I. az interneten?!
 - Web 4.0
- Ha a **WEB 2.0 = Facebook,**
- és a **WEB 3.0 = StumbleUpon,**
- akkor a **WEB 4.0 = SkyNet ???**

