

Debreceni Egyetem

Informatikai Kar

Szemantikus keresők

Témavezető:

Kuki Attila

egyetemi tanár

Készítette:

Veres Gergő

programtervező matematikus

Debrecen

2010

Tartalom jegyzék

1. Köszönetnyilvánítás.....	4
2. Bevezetés	5
3. Metaadat.....	7
4. Keresők	9
4.1. Tianamo	9
4.2. Hakai	10
4.3. Yebol	11
5. A szemantikus web rétegbesorolása	13
6. Unicode, URI	14
7. XML	17
7.1. XML dokumentumok	19
8. RDF (Resource Description Framework)	20
9. OWL (Web Ontology Language).....	22
10. Logikák	26
11. Bizonyítás	28
12. Bizalom.....	29

13. Digitális aláírás	30
14. Szemantikus keresők	32
14.1. PowerSet	32
15. Összefoglalás	35
16. Irodalom jegyzék	36

1. Köszönetnyilvánítás

Szeretném megköszönni témavezetőmnek, Dr. Kuki Attila egyetemi adjunktusnak, odaadó segítségét és végtelen türelmét, hogy értékes tanácsokkal látott el és így sikeresen elkészülhettem a diplomamunkám. Köszönettel tartozom családomnak, mert lehetővé tették, hogy a Debreceni Egyetemen tanuljak, barátaimnak, akik mindvégig türelmesek és segítőkészek voltak velem szemben.

Végül, de nem utolsó sorban szeretnék köszönetet mondani egyetemi oktatóimnak, hogy olyan sokat tanulhattam tőlük hisz ők azok, akik hozzájárultak elméleti tudásom elsajátításában.

2. Bevezetés

Az Internet... napjaink leggyorsabban fejlődő és leghasznosabb eleme. Hiszen manapság már több millió ember chattel, fórumozik, különböző hírportálokat olvasgat, maileket küld, vagy csak a barátaival veszi fel a kapcsolatot esetleg keres az neten. Aki a netet használja, két dolgot csinál: nem megy ki a nyílt webre, hanem csak az előre lementett címkéi között kattintgat. Szóval ugyanazt a 20-30 oldalt nézi meg minden nap. A második aki meg linkről linkre jár. A cégek sem engedhetik meg maguknak, hogy kimaradjanak ebből a világból, különböző reklámokon, weboldalakon hirdetik magukat. Az internet szerverek és kliensek tömkelege, a kliensek a szervereken lévő információkat. Az egyre nagyobb igények miatt az interneten fellelhető adatmennyiség olyan hatalmas, hogy egyre nehezebb a pontos keresés. 2001 márciusában a „computer” kulcsszóra keresett (Google) találatok száma 33millió volt, ugyanez 2002 végére már 77millió, mondanom sem kell, hogy 2009-ben már meghaladta a 701 milliót is. Egy felmérés szerint egyre hosszabbá válnak a keresőbe beírt kifejezések is, a találatok száma viszont nem csökken sőt, egyre inkább nő, mert ezzel bekerülnek olyan oldalak is a találatok közé, aminek nem sok köze van keresett kifejezéshez. Erre szolgálna a szemantika... Mivel a mostani keresők kizárólag azzal foglalkoznak, hogy fellelhető az adott oldalon a keresett szó és hány előfordulással, tehát hiányzik a szemantika, ennek az oka az, hogy természetes nyelven van leírva (Magyar, angol kínai...), valamint grafikus multimédiás stb. formában, amit a gépek nem értenek meg, tehát nincs gépesítve, itt jön a képbe a szemantikus web. A szemantikus web lényege, hogy keresni tud a különböző meta adatokkal ellátott különböző információk között.

A szemantikus web megvalósításához szükség van az egyes erőforrások egyértelmű elnevezésére, egy az adatok leírására és összekapcsolására szolgáló általános modellre az adatok e modell alapján történő elérésére egy közös szóhasználat definíciójára és egy következtetési rendszerre. A szemantikus web technológia elterjedése még messze van, de már napjainkban is vannak hatékony alkalmazások.

A diplomamunka célja, hogy megmutassa az szemantikus web rétegbesorolását, hogy milyen utat kell végigjárni addig amíg minden ami a neten van dokumentumok, képek, videók, blogok, hírek ...minden mindent metaadatokkal látjuk el.

Fontos belátni, hogy az internetről meglehetősen távol eső dolgokban is használnak meta információkat, például: az MP3-akhoz id3v1 és id3v2 elemeket.

A lényeg, hogy egy sima Word dokumentumhoz is ugyanúgy tudjunk meta információkat csatolni, mint az MP3akhoz.

3. Metaadat

Angol neve „Metadata”, ahogy a neve is mutatja ez egy adat, „adat az adatról”.

Rengeteg metaadat vesz minket körül, ha jobban megnézzük egy könyvtárjegy tele van metaadatokkal az adott könyvről: írója, kelte műfaja, címe... Az Cd-k is tárolnak metaadatokat, pont úgy mint az MP3ak.

A következő példában szemléletesen megmutatom mit is jelent számunkra a metaadat:

Adat	Metaadat
29,46	EUR volt a
30	napra eső bérleti díj, ami
9,7	% -al több mint a múlt havi.

komplett információsomag.

A lényeg, hogy tudjuk honnan jön az adat. Manapság a metaadat az internet számára nagyon fontos lett, mivel más rengeteg adat van az interneten és nehéz kiszűrni a tényleges információkat. Kétféleképpen tárolhatjuk a metaadatokat, magában a fájlban vagy a fájlön kívül.

Meta-keresők- lényegük, hogy más keresőgépekkel kereső keresőgépek, azaz azt az elvet követik, hogy mikor nem találunk meg valamit egyfajta keresővel, gyakran átnavigálunk egy másikra és ott találjuk meg amit szeretnénk.

Vannak kérdés-átalakítókereső is, a feltett kérdést előbb nyelvileg elemezzük, majd készítünk egy másik kérdést, ami reményeink szerint jobb hasznosabb találatokkal szolgál, mint az eredeti.

A legkézenfekvőbb módszer az lenne, hogy az oldalakat meta-információkkal látnánk el és így az oldalak közt is könnyebb keresni, nem jönnének fel keresési találatnak olyan oldalak amiknek semmi köze a keresett szóhoz, valamint az idegen nyelvűséget is egy csapásra megoldhatnánk mivel lehet hogy a keresett információnk valamilyen más nyelven van leírva.

4. Keresők

4.1. Keresés, Tianamo

A 80-as évek óta már senki sem szörfölget a neten senki sem klikkelget csak úgy a nagyvilágba és próbálja megtalálni azt ami éppen érdekli. Mindenki kereső rendszert használ pl: Google.

A Google linkstruktúrát használ belső (saját oldalon belüli linkstruktúra) és külső (más oldalakon szereplő) linkek vannak amelyek nagyban befolyásolják azt, hogy a keresőmotor jelen esetben a Google milyen keresési feltétel mellett hányadik oldalára kerülünk, mivel mint ahogy már az imént elmondtam a felhasználó nem néz meg többet az első 25oldalnál.

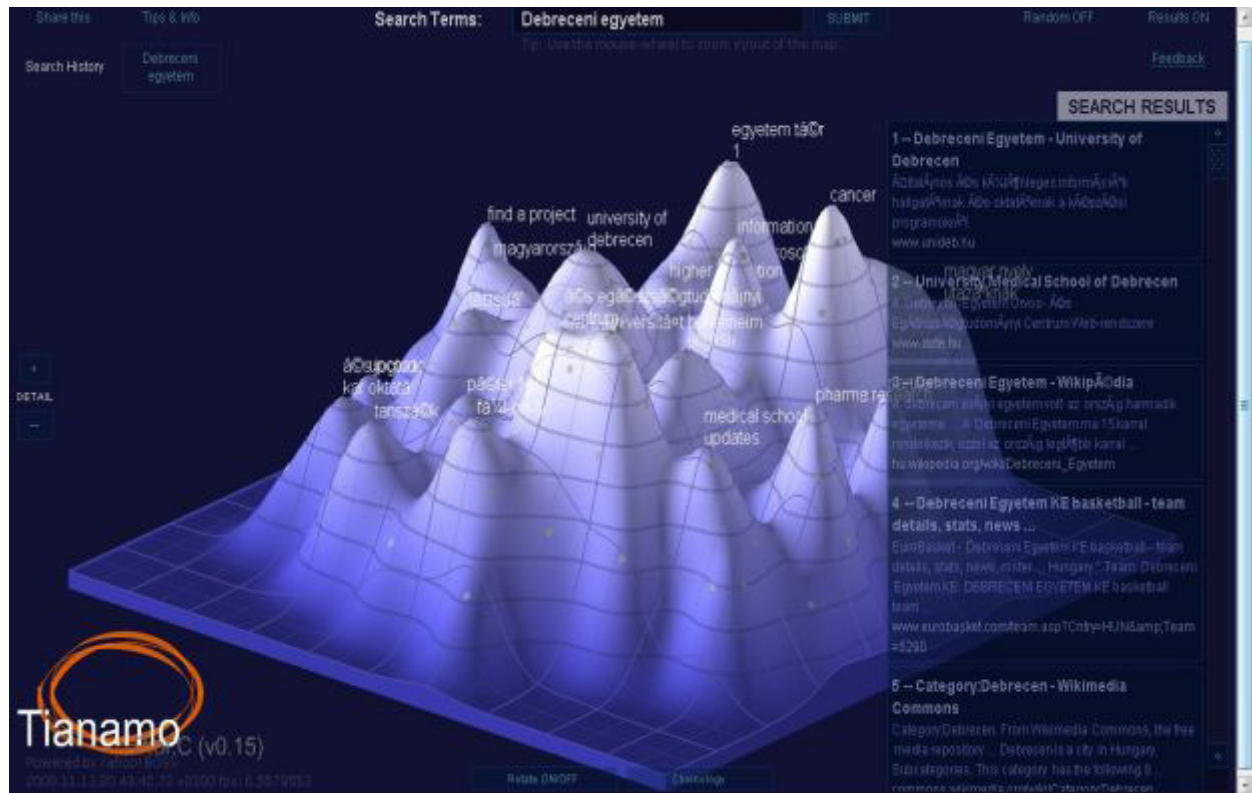
Továbbá figyelünk kell a linkmélységet az oldalról kifelé és befelé mutató linkeket a releváns és nem releváns linkeket. Ezek megfelelő használata nagy szakértelmet igényel. Ezek mellett az új oldalaknak esélyük sincs feltörni az élre, mivel nekik még nem volt idejük kiforrni. Azokat az oldalakat amelyeket csak most regisztrálnak nincs kintről rámutató link, tehát a keresőmotor nehezen találja meg és hátra teszi a listában, itt jön képbe a keresőoptimalizálás vagy más néven SEO (Search Engine Optimization). Ez arra szolgál a friss oldalak is megfelelő pozícióba kerülhessenek. Lehet még befolyásolni a keresőket különböző kulcsszavak beírása a „header” fájlba, de azt már nagyjából kiküszöbölték a linkstruktúrával...

Vannak olyan cégek akik nem szívesen költenek milliókat a marketingre hanem csak megbíznak egy olyan céget akik erre szakosodtak, hogy a SEO-t jól beállítják és egyszeri kifizetés milliókat hozhat amit sokkal több marketinggel lehetett volna elérni.

Megjelent egy új kereső ami „clusterekben” tárolja a találatokat és 3D-ben, hegyek dombok jeleníti meg azokat így nagyobb esélyük van a friss oldalaknak a megtalálásukra.

Neve „Tianamo” (1. ábra), Java platform, de elég jól használható kereső, külön gombok vannak a forgatásra leállításra és hasonlóan a Google vagy a Yahoo hoz itt is megjelennek oldalt a linkek.

A Tianamo Yahoo motorral működik, tehát ne várjunk tőle más megoldásokat, mint amit a Yahooban találnánk, de a megjelenítés sokkal egyedibb és sokkal hasznosabb...



(1. ábra: A Tianamo működés közben)

Egy másik lehetőség lenne a szemantikus világháló...

4.2. Hakai

Ez egy olyan kereső ami feldarabolja a találatokat (2.ábra), külön osztja a blog belüli beszélgetéseket, külön a képeket, az esetleges információkat a keresett szóról vagy szavakról.

Minden esetre érdekes kereső, mivel eltudja dönteni, hogy a keresett szó az egy blog környezetben van benne vagy esetleg egy könyvben, dalszöveggént.

A keresőről annyit lehet tudni, hogy az adatbázisát rakták nagyon össze, a szavak közötti relációkat, összefüggéseket. A lényeg hogy egy nagyszerű kereső a Hakai.



2.ábra

4.3. Yebol

Egy másik nagyon jó kereső a „Yebol” (3.ábra). Ez a kedvencem, a Hakainál jobb találatokat hoz és kinézetre, dizájnilag sem utolsó.

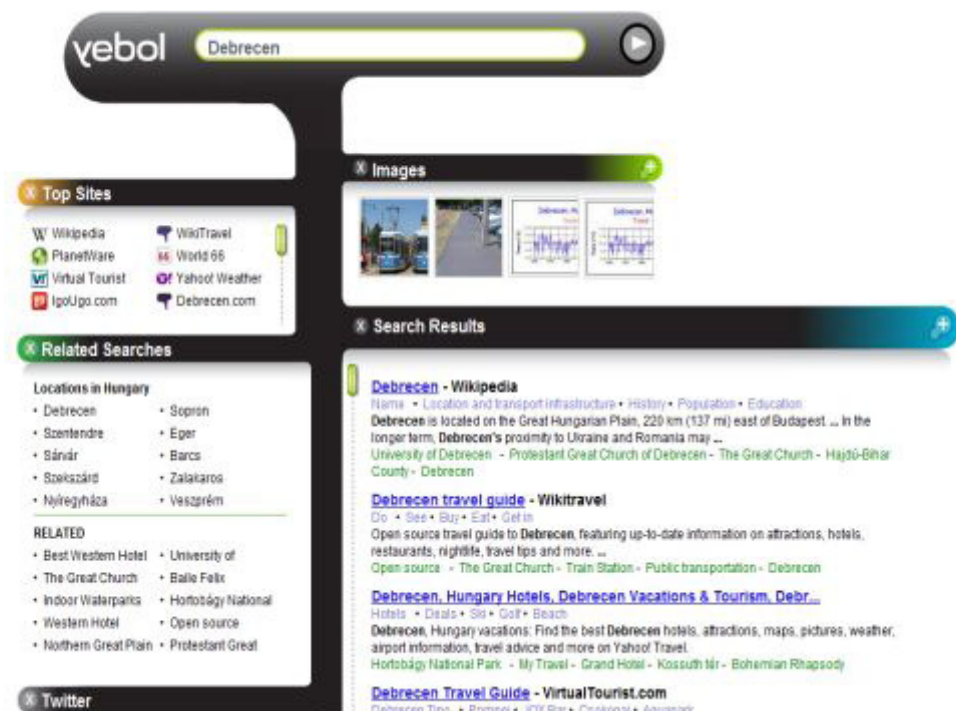


3.ábra

De elemezzük egy kicsit:

Keresés után három fő területünk van (4.ábra).

- „Search Results”- keresési találatok: ez mint a már a Google-tól megszokott dizájnbán de teljesen szűrt eredményeket kapunk.
- „Top Sites”- főbb oldalak: itt a fontosabb oldalak vannak felsorolva egymás alá.
- „Related Search”- kapcsolódó keresés: itt a kapcsolatos témák vannak felsorolva, amiket úgy „gondolt” a kereső, hogy esetleg azok is érdekelhetnek.



4.ábra

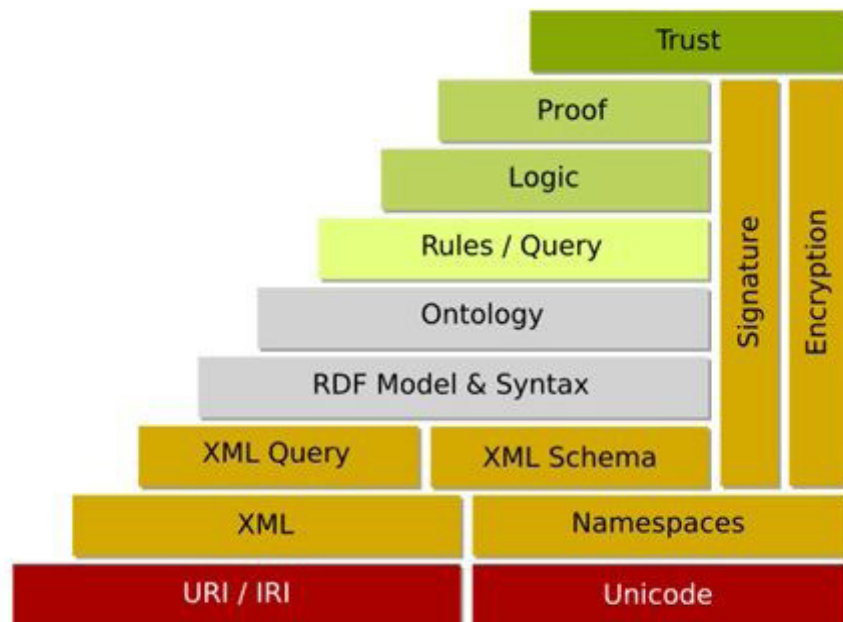
A lényege a keresőnek az, hogy kisgyerekeknek néz minket, mert nem bízva ránk, hogy mit keresünk, nem tételezi fel azt, hogy értjük amit akarunk, hanem ő építi fel az egészet ő keresi meg, és „érti” meg, hogy mit is szeretnénk.

5. A szemantikus web rétegbe sorolása.

Tim Berners Lee feltalált már 10 éve egy olyan eszközt, ami több lépcsőből áll és teljes mértékben körülírja azt amit a szemantikus webnek tudnia kell és biztosítania. Ebben a fejezetben szeretnék Berners Lee sémáján néhány réteg részletesebb külön taglalásával végigmenni a szemantikus web rétegein.

A weben a következtetéseket is le tudjuk kezelni a szemantikus webbel. Akik ezzel foglalkoznak, és kidolgozzák a szabványokat az a W3C „*World Wide Web Consortium*”.

Ami a legfontosabb a lépcsőből, és tényleg ez képezi a szemantikus web motorját, az az RDF „*Resource Description Framework*” és az OWL „*Web Ontology Language*” persze nem ennyire egyszerű a séma. A legkönnyebben ezzel a modellel lehet megmutatni, hogy hogyan is kellene elképzelni a szemantikus webet. (5. ábra).



5.ábra: A szemantikus web modellje

6. URI-UNICODE

Az első helyen az UNICODE és az URI áll. A Unicode nagyon egyszerűen megfogalmazható, ez egy nemzetközi szabvány, ami a különböző írásrendszerek egységes kódolását írja le. Részletesen foglalkozik a karakterek megjelenésével. 16biten tárolt síkokra oszt 100000 karaktert.

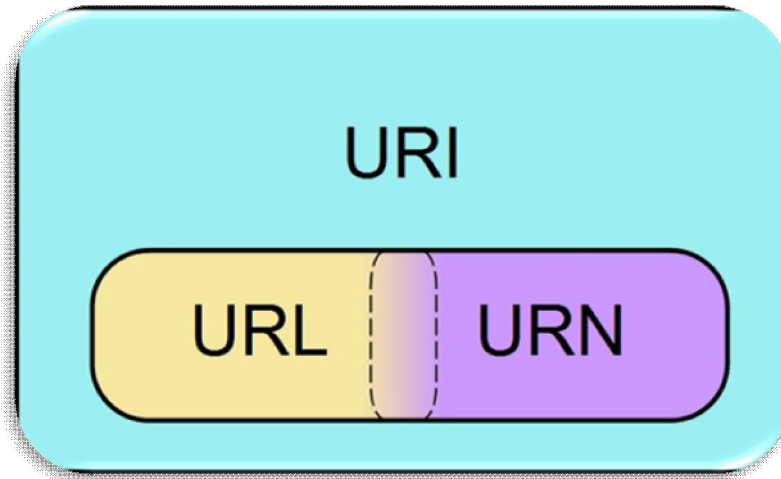
A nemzetközi fonetikai ábécé az IPA (International Phonetic Alphabet). A Unicode egy olyan szabványkódolás, ami nemzetközi karakterkészleteket használ és lehetővé teszi, hogy minden emberi nyelven használható (írás és olvas) a web segítségével egy szabványosított formában.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0250	e	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
0260	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g
0270	u	u	u	u	u	u	u	u	u	u	u	u	u	u	u	u
0280	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
0290	z	z	z	z	z	z	z	z	z	z	z	z	z	z	z	z
02A0	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q
02B0	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n
02C0	z	z	<	>	^	v	^	v	^	v	^	v	^	v	^	v
02D0	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
02E0	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y
02F0
1D00	A	Æ	æ	B	C	D	Ð	E	З	!	J	K	Ł	M	И	O
1D10	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
1D20	V	W	Z	z	z	z	z	z	z	z	z	z	z	z	z	z
1D30	o	e	z	g	h	i	j	k	l	m	n	и	o	o	o	o
1D40	T	U	W	z	z	z	z	z	z	z	z	z	z	z	z	z
1D50	m	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
1D60	o	x	i	r	u	v	z	v	z	z	z	z	z	z	z	z
1D70	h	p	f	f	s	t	z	z	z	z	z	z	z	z	z	z
1D80	b	d	f	g	k	l	m	n	p	r	s	z	z	z	z	z
1D90	q	d	e	z	z	z	z	z	z	z	z	z	z	z	z	z
1DA0	f	j	g	u	i	l	z	z	z	z	z	z	z	z	z	z
1DB0	n	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o

Unicode

Az első helyen a Unicode mellett még ott van az URI (*Uniform/Universal Resource Identifier*) egy karaktorsorozat ami egy nevet vagy egy forrást az interneten azonosítására

használunk. Két komponensből tevődik össze az URL-ből (Uniform/Universal Resource Locator), és az URN -ből (Uniform/Universal Resource Name).



6.ábra: „Venn diagram URI=URL+URN

Egy **URI** -t lehet minősíteni (URL) néven vagy (URN) néven vagy mindkettő. A Uniform Resource Name (URN) úgy működik, mintha a személy nevét írná le, míg a Uniform Resource Locator (URL) olyan mintha a személy postacímét írná le. Más szóval: az URN meghatároz egy elem azonosítót, míg az URL ad egy módszert az elem megtalálására.

Az URI jellemzők:

- **Egységes:** Az egységesség számos előnnyel szolgál. Ez lehetővé teszi a különböző típusú erőforrás azonosítók használatát ugyanebben az összefüggésben. Lehetővé teszi, hogy az új típusú erőforrás-azonosítókat használjuk, ahogyan a létező azonosítókat kell használni, beavatkozás nélkül.
- **Forrás:** Ez a specifikáció nem korlátozhatja, hogy mi lehet a forrás, hanem a "forrás" kifejezést használja az általános értelemben bármivel lehet azonosítani egy URI –t. Családias példák közé tartozik az elektronikus dokumentum, egy kép, egy információforrás, konzisztens céllal.
- **Azonosító:** Az azonosító testesíti meg a szükséges információkat, megkülönbözteti őket.

Az XML használata során szükség van egy szabványos sémára, mellyel sikerül beazonosítani dokumentumokat, helytől függetlenül. URL -eket rengeteg helyen használunk főleg XML szabványban.

Az neten lévő dokumentumok szabványos módon való meghatározásának igénye az URL (Egységes Erőforrás lokátor, Uniform Resource locator) séma kifejlődéséhez vezetett. Ez a séma ugyanolyan jól működik interneten mint sajátgépen.

Az URL igazából egy szövegsztring. Ez a sztring bármilyen formában megjelenhet. Kinyomtathatjuk papírra és akkor is URLnek hívjuk, de elektronikus formában hasznosabb...főleg annak a szoftvernek amelynek el kell érnie egy dokumentumot a megadott URL alapján.

Az „abszolút” URI -k erőforrásokat azonosítanak, a „relatív”-nak pedig csak az adott környezetben van értelmük. Látszólag teljesen egyformák, annyi különbség van köztük, hogy a szintaktikájukban egy sémaazonosító hiányzik a relatív URI elejéről.

Egy “relatív” URI feloldása közben az “abszolút” URI a “bázis” URItól örökli a sémát.

7.XML

Az XML-elemek összetevőkre bontják, és a szabvány alapvető részeit alkotják. A további funkciói a szabványnak, más periférikus szabványokkal együtt támogatószerepet játszanak, emellett az összetevők finomítására, vezérlésére összpontosítanak.

Az XML neve az „Extensible Markup Language”(Kiterjeszhető jelölőnyelv) kifejezés rövidítése (az „E” helyett jobban mutat az „X”). Nem tartozik a kereskedelmi érdekeltségekhez. A W3C (a World Wide Web Konzorcium) fejlesztette ki, majd a korábbi jelölőnyelveken alapuló tapasztalat eredményeként fejlődött.

Az XML-t szinte mindenhol fel lehet használni, alkalmazni lehet, viszont van két terület is amiben nagyon fontos szerepet tölt be, az adatcsere-formátum és a dokumentum szerkesztési formátum. E két alkalmazási mód gyakran kiegészíti egymást, mivel az adatok emberek által történő felhasználásra is formázhatók.

Az XML- adatformátumot percek alatt elsajátíthatjuk és úgy érezhetjük, hogy csak ennyiből áll az egész, azonban számos megszorítás és rejtett komplikáció van. Ettől függetlenül kevés XML-el is sok mindent megtehetünk.

Az XML-adatformátum általában támogat valamilyen más szövegformátumot. Az XML tehát kiaknázza azokat a hatalmas előnyöket, amelyeket e megközelítés nyújt az információcsere számára.

A névterek dokumentumokat segítő szabványt ismertet, melyeknek különböző tartományokból, külön DTD -ben vagy XML sémákban definiált összetevőket tartalmaznak.

Séma

Manapság a legjobb az XML-sémaszabvány, ez a legfigyelemreméltóbb a jelen állás szerint, persze vannak más DTD helyettesítők. Bár a névtereket használó XML-dokumentumok DTD -vel validálhatóak, mégis van néhány szigorú korlátozás.

A DTD -nek az összes névtér valamennyi elemének és attribútumának összes lehetséges kombinációját ismernie kell, és az alkalmazott előtagokat előre kell definiálni.

Általában az összes modellező nyelvre a séma kifejezést alkalmazzák, mivel ez egy átfogja az összes nyelvet. Az elnevezés az informatikából, pontosabban az adatbázis technológiák terminológiájából származik. A séma az adatbázisok tábláinak és mezőinek, az egyes mezők értékeire vonatkozó megszorításokra és a táblák közötti kapcsolatok leírására szolgál.

Az lenne jó ha egy univerzálisan elfogadott szabvány alakulna ki, mert most többen is versengnek az elsőbbségért: Schematron, TREX, XML-séma.

Az XML-sémák 2001 májusában a W3C kiadta az XML-sémaszabványt. A szabvány nagyon hosszú munka eredménye, ami nem meglepő, mivel a cél egy minden érdekelt számára kielégítő modellező nyelv kifejlesztése volt. A nyelvnek nagyon rugalmasnak és erőteljesnek kellett lennie, úgy, hogy közben könnyen implementálható maradjon. A funkciók tekintetében az XML-sémamodellek lefelé kompatibilisek a DTD –kel.

Jelenleg két XML verzió létezik az 1.0 és az 1.1. Az XML 1.0-t sokáig fejlesztették első kiadása 1998-ban volt és azóta is azt javítgatják, 2004 februárjában kiadták a harmadik verziót az 1.0 –ból és egy időben adták ki az 1.1-et is, később majd 2006 augusztus 16-án kiadták a negyedik verziót is az 1.0-ból.

A különbség nem annyira eltérő mivel az XML 1.0 Unicode 2.0 –t használ amiben jelen vannak a legtöbb világnyelvek csak pár nyelv maradt ki mint pl.: kambodzsai, burmai...

Az XML 2.0 szabványról bár már régóta beszélgetnek de nem látható előre, hogy egyáltalán létre fog e jönni.

Látjuk, hogy az XML csak az adatok feldolgozására használhatjuk a szemantika ami igazán fontos az még nincs meg, hogy az alkalmazások egymás közt kommunikáljanak anélkül, hogy előtte ne kelljen le ellenőrizni az információkat.

Ennek a szemantikának az elérésére hozták létre az RDF (*Resource Description Framework*)-et az erőforrás leíró keretrendszer, ezzel a keretrendszerrel a következő fejezetben fogok részletesebben foglalkozni.

7.1. XML dokumentumok

Nem használjuk a dokumentum kifejezést csak esetleg adatobjektumokkal kapcsolatban, mivel az XML olyan korábbi szabványokon alapul, amelyeket elsődlegesen elbeszélő szöveg közzétételre való előkészítésére alkalmaztak. Ez a terminológia még ma is használatos, annak ellenére, hogy amikor az XML-t szoftveralkalmazások közti adatcserére használjuk, sohasem tároljuk adatfájlban, és sohasem tesszük közzé vagy mutatjuk be más módon az embereknek. Természetesen még mindig gyakori, hogy az XML tárolt adatfájlokban jelenik meg, bizonyos esetekben pedig ezek a fájlok terjesztésre és közzétételre szánt elbeszélő szöveget tartalmaznak.

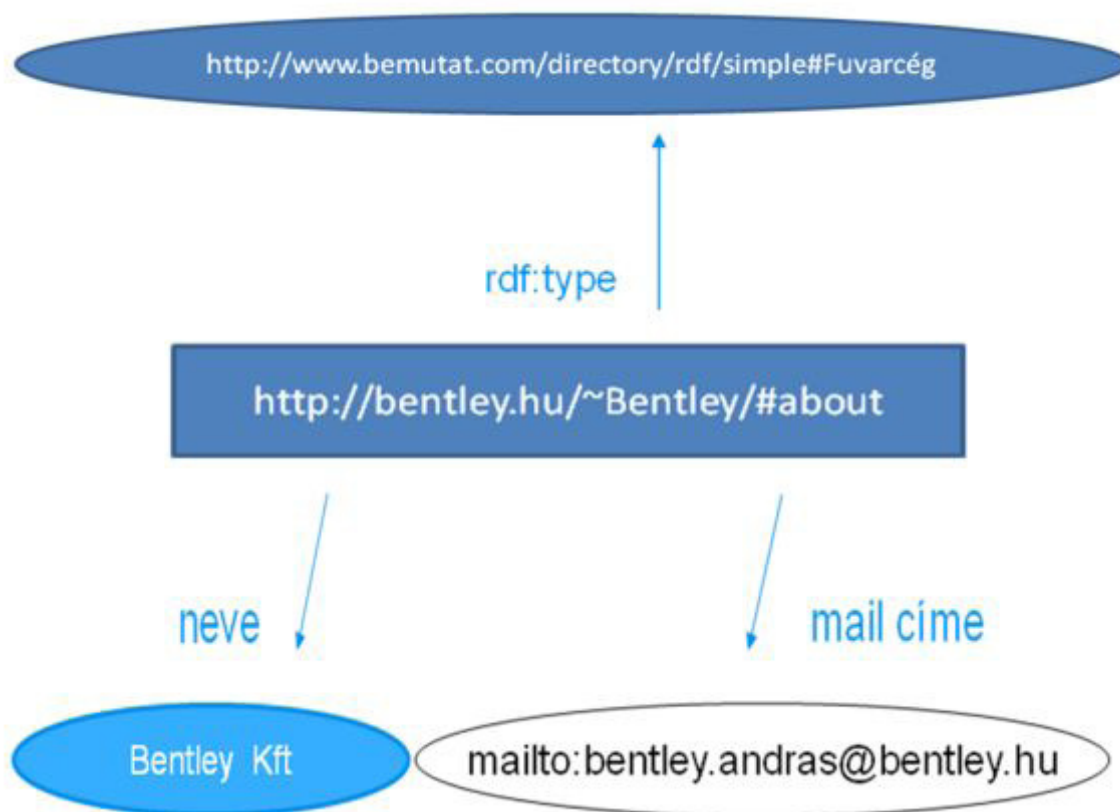
Az XML dokumentumot akkor tekintjük helyesnek ha helyesen van formázva azaz ha megvan mind a három része, pl.: ha van nyitó tagja de nem rendelkezik záróval akkor az elemző nem fogja azt feldolgozni.

Pl.:<DMunka> ... </DMunka>

8.RDF

Az RDF (*Resource Description Framework*) leírás metaadatok összessége.

Lényege, hogy azokat az erőforrásokat amelyeket URI -vel azonosítottak összekapcsolja más erőforrásokkal, literálokkal (7.ábra).



7.ábra RDF gráf

Könnyebb gráfban gondolkodni, gráfokkal leírni az állításokat.

Az RDF rengeteg állításból tevődik össze, az állítások halmaza egy irányított, címkézett gráf, a csúcsa az alany és a tárgy a gráf élei pedig az állítmány.

A „Semantic Web Interest Group” jelenleg megbeszéléseket tart arról, hogy újításra van-e szükség? Ha igen akkor milyen formában valósuljon meg, mivel 2009 végén már voltak viták a lehetséges RDF 2.0 irányában.

Az RDF- ek bármiről szólhatnak, a lényege az, hogy az URI- k mire utalnak. Az URI ágyazza be az RDF- et a szemantikába.

Az URI- k végzik a munka nagy részét ebben a felállásban, mivel minden URI mutat valamire ezért előfordulhat az is, hogy két dologra egy azon URI- vel hivatkozunk, ezek azonosak.

Ami még fontos az RDF- ben, hogy megadhat definiálhat metaadatot, összevonásukat meg akármelyik alkalmazás tudja.

Az „RDF Vocabulary Description Language” és az „Ontológia Web Ontology Language” (OWL) típus definíciók.

Az RDFS nem más, mint osztályok és források összessége. ezek valamilyen módon kapcsolatban állnak egymással, függ attól, hogy az egyed milyen osztályokba tartozik. Lehet típus és alosztály. Az egyik függ attól, hogy az egyed egy osztályba tartozik, a másik pedig ha beletartozik az egyikbe akkor a másikba is.

A tulajdonságokat is ugyanúgy két csoportba osztjuk: bal és jobb csoportba. A bal csoportot értelmezési tartománynak is nevezhetjük, a jobbat pedig értékészletnek.

9. OWL

Sajnos az RDF nem fed le mindent amit kellene, szükségünk van egy következő szintre, ahol már a következtetéseket is levonhatjuk. Itt jön képbe az OWL Web Ontology Language. Például halmazoknál ha B eleme A-nak, C eleme B-nek, akkor C eleme-e A-nak? Nekünk nyilvánvaló a kérdés viszont a számítógép számára már nem.

Az OWL pont, mint az RDF URI -k tömkelegéből áll. Nem mondhatjuk, hogy túl régóta van a „pályán”

Ezt a szemantikus nyelvet DAML+OIL -nak nevezték. Az OIL (*Ontology Interface Layer*) és a DAML -ONT (*DARPA Agent Markup Language*). A sok foltozgatás, alakítgatás után átnevezték OWL -re. Az osztályokat, tulajdonságokat írhatjuk le ebben a nyelvben.

Az OWL -ben három alnyelvet definiálhatunk.

- „OWL Lite” – Olyan felhasználók számára, akik nem akarnak sok mindent kiszámítani, csak sima korlátozásokat akarnak.
- „OWL DL” – Azoknak a felhasználóknak készült akik biztosak akarnak lenni az eredményben, sok minden kiszámítható ezzel az alnyelvvvel, leíró logika, minden kiszámítható, minden művelet véges időn belül befejeződik, mindent tud amit az OWL tud de csak korlátok közt.
- „OWL Full” – Azoknak a felhasználóknak készült akik kitudják használni az OWL-t teljesen, a hátránya viszont az, hogy nem kiszámítható.

Előnye az Full -nak, hogy együtt tudjuk alkalmazni az OWL és az RDF sémát. Nem érdekli, hogy szétvan-e választva az osztályok, egyedek, tulajdonságok...

Szeretnék ebben a fejezetben megmutatni egy konkrét eszközt amellyel saját ontológiát alkothatunk. Az ontológiai szerkesztők olyan alkalmazások amelyeknek a célja, hogy segítséget nyújtsanak az ontológiák létrehozásában és szerkesztésében.

Az interneten keresgélve rengeteg ontológiakészítő programot találtam. Például: „Chimaera” (Stanfordi egyetem), „DOME” (DERI Ontology Management Environment), „HOZO” (Java

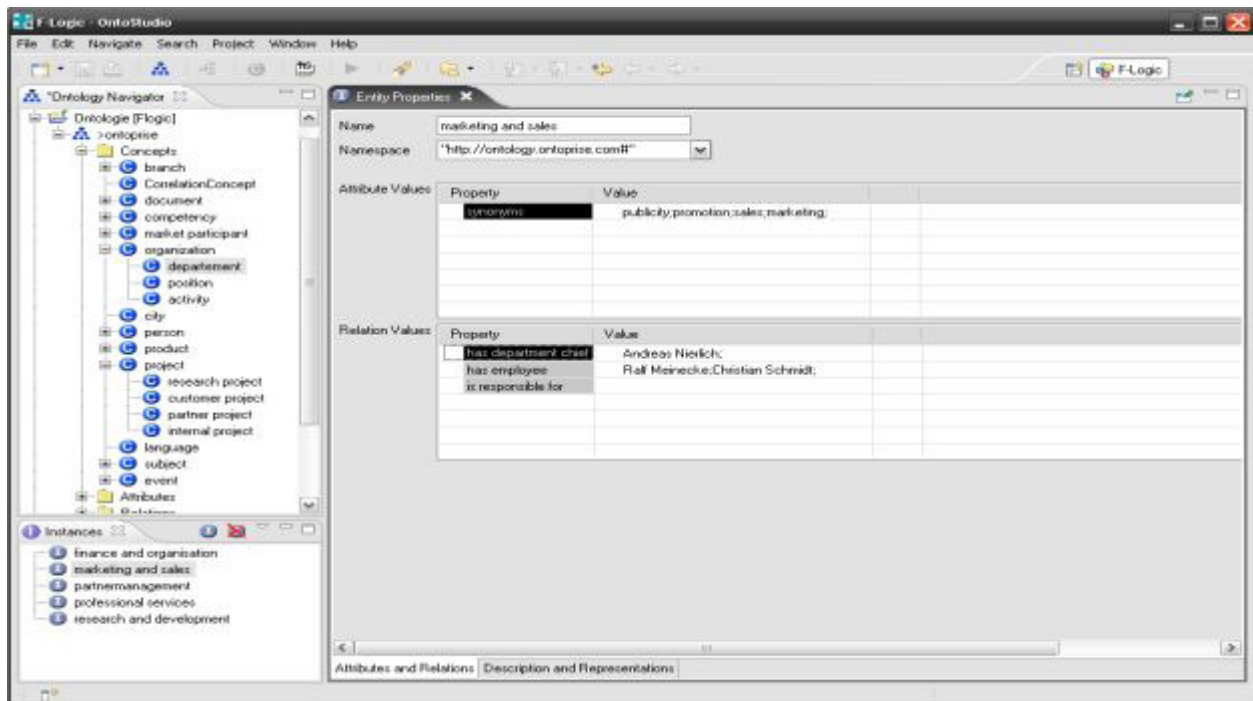
lapu, Osaka egyetem), „Protégé” (Java alapú, Stanfordi egyetem), „NeOn Toolkit”, „OBO-Edit, OilEd”, „Swoop”, „Model Futures OWL Editor” (ingyenes), „CmapTools Ontology Editor” (COE Florida egyetem) és sok egyéb, én szeretném bemutatni a „Ontoprise”-tól az „F-Logic Onto Studio” programot.

Az „F-Logic Onto Studiot” ingyenesen letölthetjük és három hónapos kipróbálási lehetőségünk van. Az „OntoStudio” is képes importálni sok struktúrát, sémát és modellt. A legfontosabb feladata a leképezés eszköze, a grafikus szerkesztő és az integrált tesztkörnyezet. A grafikus szerkesztő szabály és az integrált teszt biztosítja a környezet minőségének a modellezés.

Az „OntoStudio” támogatja az „F-Logic”, RDF, OXML formátumokat. Az OXML fájl az az „OntoStudio” saját XML-formátumú adatok tárolására szolgáló fájl.

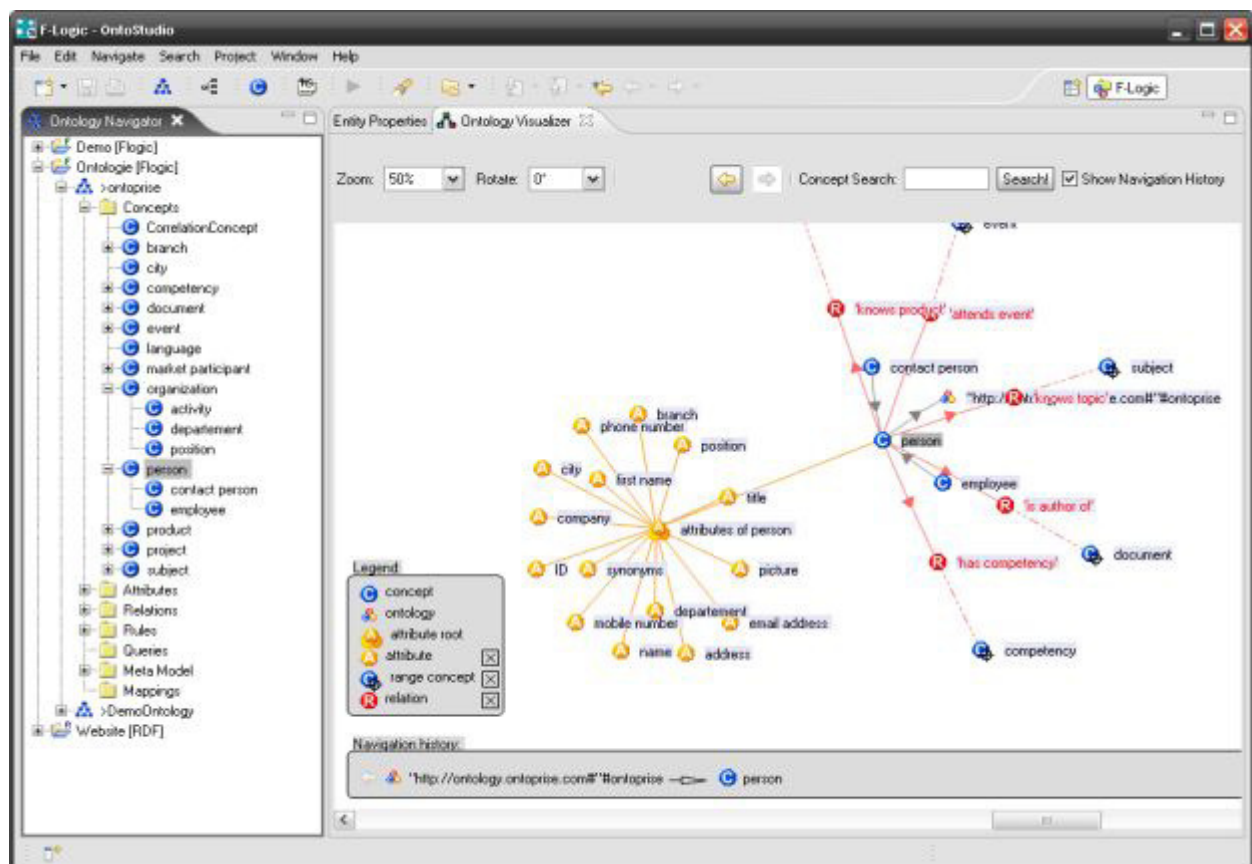
Importálni tudja az UML 2.0-t, adatbázismintái: Oracle, MS-SQL, DB2, MySQL, Excel táblázatok, fájlrendszerben levő könyvtárszerkezetet.

Maga a program arra szolgál, hogy itt adhatjuk meg azokat az adatokat amik leírnak egy ontológiát.



8.ábra:F-Logic

Az elkészített ontológiákat grafikusan is szemügyre vehetjük:



9.ábra: F-Logic

Az „OntoStudio” a legelterjedtebb kereskedelmi az ontológiák modellezési és létrehozási területén. Átfogó funkciók, intuitív ontológia. Az „OntoStudio” több felhasználó is képes egyszerre szerkeszteni az ontológiát az „OntoBroker Collaboration” szerveren.

- „Taxanómia” – osztályozást jelent vagy a dolgok hierarchikus osztályozása. Mindent szinte, tárgyakat, helyeket, eseményeket osztályozhatunk, rendezhetjük őket valamilyen keretrendszerbe.

A tulajdonságokkal tudunk objektumokat, osztályokat jelzővel, tulajdonságokkal ellátni.

A tulajdonságokat két részre oszthatjuk:

- objektum tulajdonság: ez bináris relációt képes felállítani két példány között.
- adattípus tulajdonság: ez arra szolgál, hogy összekössön egy literált egy sémával.

OWL2- Ez az OWL ontológiai nyelv továbbfejlesztett változata, a W3C ajánlását 2009 október 27 én kapta meg.

Az OWL 2 egy új dokumentum, három különböző profilja van, amik függetlenek egymástól.

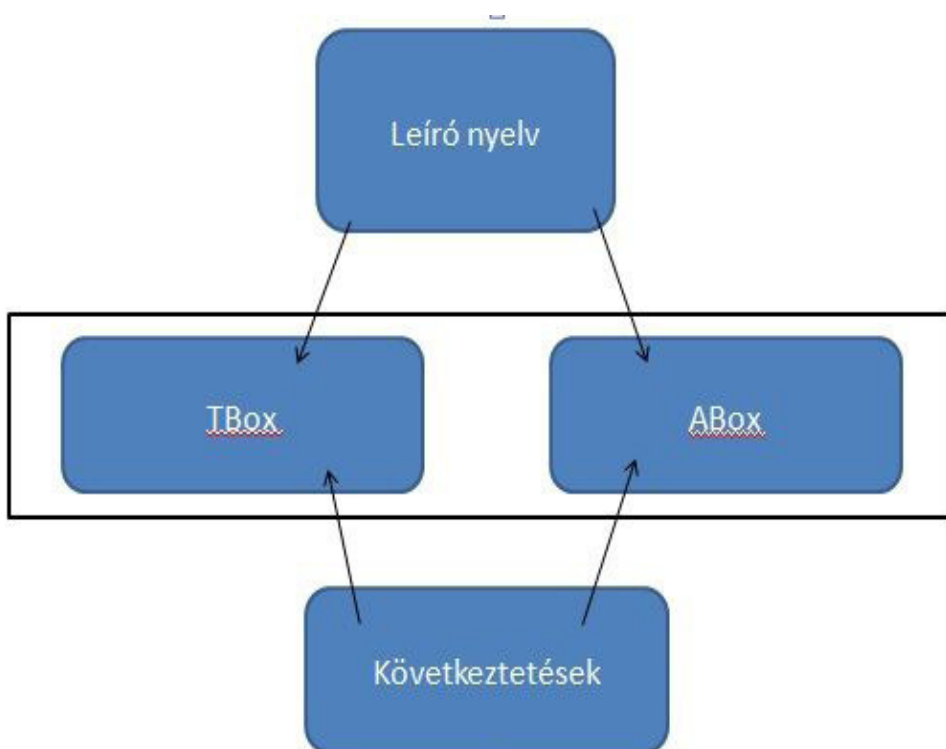
- OWL2 EL – különösen hasznos az alkalmazások számára. Az EL tükrözi a profil alapján az EL leíró logikát. Egzisztenciális mennyiség.
- OWL2 QL – Nagymennyiségű adatok számára, hagyományos relációs adatbázis rendszereken, konjunktív lekérdezés. A QL- a kérdés megválaszolását ebben a profilban lehet végrehajtani.
- OWL2 RL – Az RL arra utal, hogy az érvelést ebben a profilban lehet megvalósítani.

10. Logikák

A logika szó egy görög szóból ered. Ez egy olyan tudományág ami csak érvényes következtetéseken alapul.

A diagram következő lépcsője a Logikák, azon belül is a leíró logikák. Ezek az elsőrendű logika résznyelvei képezik az OWL hátterét. A leíró logikák lényege, hogy velük könnyen modellezni tudjuk ennek a területnek, vagy bármely terület rendszerét.

A Logika szemantikai lépcső egyik legfontosabb része, ez mondja meg a kifejezések szemantikai értékét. Pont mint az RDF- ben csak két argumentumú kapcsolatokat lehet használni.



10.ábra

Itt a fogalmak az egyedeknek egy halmazát jelöli, például az állatok fogalma az összes állat fogalmát jelöli.

A tudásbázis az A és a T doboz.(Knowledge base).

- T: terminológiai doboz, itt a fogalmak vannak, a terminológiai halmaz.
- A: adataxiómák halmaza.

Az A doboz olyan, mint egy relációs adatbázis, amelyben csak olyan táblák vannak amelyben csak egy két oszlopú tábla van, azzal a különbséggel, hogy itt nyílt az adatbázis.

11. Bizonyítás

A leíró logikákban az állítások helyességére is szükség van. A következtetések eredményét szeretnénk megosztani a weben, és a webre kitett állítások helyességéről bárki meggyőződhet. Ahhoz, hogy a lekérdező félnek ne kelljen elvégezni még egyszer a bizonyítást, csatolnunk kell azokat az információkat mikből a következtetést levontuk.

A PML (Proof Markup Language) egy kezdetleges bizonyítási nyelv az RDF alapján nyugszik és egy bizonyítás leírására szolgál, az osztályrését viszont az OWL-től vette át az owl: Class-t.

A technológia még fejlesztési fázisban van és a W3C ajánlásától még messze áll, de ha kész lesz akkor jelentősen megkönnyíti a bizonyítékok megosztásának folyamatát.

12. Bizalom

A következő lépcső a rétegbe az a bizalom, ez a lényege az egésznek, hogy tudjunk együtt működni, mivel az internet nem csak abból áll, hogy naponta megnézzük a híreket vagy esetleg letöltünk egy Mp3-at.

A W3C próbálja megerősíteni az emberekben a bizalmat az internet, a web iránt. Ez lenne az egyik legfontosabb és legnehezebb lépcsője a szemantikus webnek.

13. Digitális aláírás

A digitális aláírás is egyfajta a mi hitünket megerősíteni hivatott, lényege, hogy az aláírásunkat helyettesíteni tudjuk, szóval bárki aki azt az iratot felhasználja vagy elolvassa, fogja tudni, hogy kitől származik, ugyanakkor ha módosul az irat akkor bizonyítani tudjuk, hogy nem mi módosítottuk.

Az aláírás tartalmaz egy ellenőrzőséget, amihez tartozik egy „hash”- függvény. Egy összetett függvény ami az aláírás idejéből, az aláíró nevéből, az algoritmus nevéből... áll. Ezek után az aláíró leködolja a titkos kulccsal.

A digitális aláírás (Digital Signature) a legegyszerűbb protokoll az RSA –ból vezethető vissza.



11.ábra

A „hash” függvények egyirányú de nem egyértelműen deklaráható függvények:

$$h: \{0,1\}^* \rightarrow \{0,1,\dots,d-1\}, d=2^{160}$$

Attila kiszámítja a $h(I)$ -t és a $J=h(I)^d \bmod m$ -et és (I,J) -t elküldi Beának.

Bea kiszámítja a $I'=J^c \bmod m$ -et és a $h(I)$ -t.

Ha $h(I)=I'$ akkor elfogadja az aláírást különben elutasítja.

14. Szemantikus keresők

A keresés a szemantikus keresővel általában szemantikus információkkal, metaadatokkal ellátott adat.

Ezeknél a keresőknél nincs nyelvi ütközés ami a nagy problémája a mostani keresőknek, itt arra mentek rá hogy az oldalak szövegének a gépi lefordítását érik el, azaz a gép „értse” meg miről is szól az oldal.

Mivel ez a jövő, vannak ígéretes keresők, igaz még fejlesztés alatt vannak de már alkalmasak bizonyos kérdések megválaszolására. Ilyen például az „iGlue” amit a Microsoft becslések szerint 100 millió dollárért vásárolt fel, vagy a „PowerSet”, „RicerGlass”, „Hakia”, „TextWise”. Ezek közül szeretném bemutatni a PowerSet-et, a Hakait már bemutattam még a diplomamunkám, elején..

14.1. PowerSet

A PowerSet egy San Francisco-i cég Californiában, 2005-ben alakult azzal a céllal, hogy a természetes nyelvű keresést kiépítse, azaz, hogy lehetővé tegye a konkrét kérdésre adó választ. Ellentétben a kulcsszó alapú keresőkkel. A cég 2008 Július 1 óta Microsoft tulajdonban van.



Maga amire keresünk egy szöveg, mivel szövegre tudunk szűrni a meglévő adathalmazból, a szintaktikája a keresőnek nagyon egyszerű, lényege, hogy nem fontos, hogy mikor publikálták az adott folyóiratot, vagy, netán milyen nyelven van.

Ha változtatunk valamit egy mondatban, más szóösszetételt használunk, akkor már a mondat jelentése megváltozhat, és a szemantikus keresőnek reagálnia kell erre, hogy sikeresen letudjuk védeni ezt a részét is a keresésnek, következtetéseket kell bevezetni ami függ a szövegtől, szóval egy szónak vagy egy mondatnak akár több értelme is lehet, amit nem biztos, hogy logikával lehet ábrázolni. Emiatt más ábrázolási módot, a szövegkörnyezetet figyelembe vevő módot kell alkalmazni

A szemantikus keresők már kézzel foghatóak de sajnos még nincsenek abban a fázisban, hogy széles körben elterjedjenek. Még kell néhány év és valószínűleg átszoknak az emberek a szemantikus keresőkre mivel jelentősen megkönnyítenék az életünket...

15. Összefoglalás

A szemantikus web egy feltörő ágazat amit bárhogy is próbálunk elkerülni, erre szükségünk van. A web 1.0 a megjelenéséről szólt, a 2.0 a közös tartalomról, 3.0 elvileg a szemantikus webről szólna. Ennek a célja, hogy az interneten fellelhető rengeteg információ, adat, tudás ne csak olvasható legyen a számítógépek számára hanem megérthető is, mivel iszonyatos iramban emelkednek a világhálón az oldalak száma ezért már az ember képtelen felügyelni azt a több milliárd oldalt amit át kellene látni. Bár úgy tűnik, hogy mindez csak a jövő és, hogy csak feltételezés de annak az oka, hogy még nem terjedtek el ezek a keresők, nem a keresők kezdeti állapota miatt hanem a szemantika hiánya, a metaadatok hiánya a világhálóról nem teszi ezt még lehetővé. Mivel egy egyszerű felhasználó egyáltalán nem ismeri az RDF és az OWL nyelveket, hogy ezeket a hibákat ki tudja javítani, ami másrésről meg időigényes.

Diplomadolgozatom elkészítésével szerettem volna betekintést nyújtani a szemantikus web felépítésébe struktúrájába, az alapoktól kezdve egészen a végső lépcsőig. Barnes Lee diagramján végig haladva, bemutattam az egyes lépcsőket, történetüket, működésüket példával illusztrálva.

Irodalomjegyzék

- [1] W3C Semantic Web Activity
<http://www.w3.org/2001/sw/>
- [2] Szemantikus Web bevezető <http://www.w3.org/2002/Talks/2409-Budapest-IH/OverviewPrint.html>
- [3] Gottdank Tibor: Szemantikus web – Bevezetés a tudásalapú internet világába, ComputerBooks, 2005
- [4] Herendy Csilla: Web 3.0 - A szemantikus web?:
<http://damjanovich.hu/cikkek/web3.0-a-szemantikus-web.html>
- [5] W3C <http://www.w3.org/RDF/>
- [6] RDF Query Specification <http://www.w3.org/TandS/QL/QL98/pp/rdquery.html>
- [7] W3C Consortium - RDF Validation Service: <http://www.w3.org/RDF/Validator>
- [8] PowerSet <http://www.powerset.com/>
- [9] Szeredi Péter – Lukácsy Gergely – Benkő Tamás: A szemantikus világháló elmélete és gyakorlata, TYPOTEX, 2005