



Ontologie a WWW

Vojtěch Svátek

Vysoká škola ekonomická v Praze
katedra informačního a znalostního inženýrství
svatek@vse.cz

Koncepce výkladu

- Pojem *ontologie* velmi široký; zde jen:
 - význam používaný v *informatice*
(X širší filosofické souvislosti)
 - v jeho rámci důraz na *konceptuální* ontologie
(X čistě terminologické)
- Vztah ontologií a *WWW* chápán převážně v intencích tzv. *sémantického webu*
- (Neumělá) snaha o porovnání s principy *databázových systémů*

Obsah tutoriálu

1. Základní pojmy ontologického inženýrství
2. Jazyky pro reprezentaci ontologií
 - „**Syntaktická**“ odbočka: RDF
 - „**Sémantická**“ odbočka: Deskripční logiky
3. Aplikace ontologií
4. Problémy a perspektivy ontologií v rámci sémantického webu



Základní pojmy ontologického inženýrství

Pojem ontologie

- Ve filosofii:
 - „nauka o bytí“
 - „univerzální soustava znalostí o světě“
(tak, jak je, nezávisle na usuzování o něm...)
- V informatice:
 - „soustava znalostí o světě“
 - bez nároku na ucelenost (mnoho dílčích ontologií)
 - často účelově vzniklý artefakt

Definice ontologie

- (T. Gruber, 1993) „explicitní specifikace konceptualizace“
- (W. Borst, 1997) „*formální* specifikace *sdílené* konceptualizace“
- (B. Swartout, 1997) „hierachicky strukturovaná množina *termínů* popisujících určitou věcnou oblast...“
- a mnoho dalších

„Komunitní“ pohledy na ontologie

1. Rozvinutí *tezaurů*
 - strukturace *termínů* přirozeného jazyka
2. Rozvinutí *DB schémat*
 - sémantická nadstavba nad primárními *datovými* zdroji
3. Rozvinutí reprezentace znalostí v *umělé inteligenci*
 - *logické teorie*, důraz na „plné“ odvozování
4. Sémantická nadstavba *webových metadat*
5. *Matematické* struktury
 - zvl. *konceptuální svazy* (FCA)

Předmět konceptualizace

- Věcná oblast → *doménové* ontologie
- Obecná problematika napříč věcnými oblastmi → *generické* ontologie
- Problematika řešení určitého typu úloh → *úlohové* ontologie
- Jazyk → *reprezentační* ontologie
- Konglomerát konceptuálních znalostí pro potřebu konkrétního projektu → *aplikační* ontologie

Míra formalizace

- Obvykle se předpokládá použití formálního jazyka
- Neformální „ontologie“: glosáře
- Semi-formální ontologie: částečně strukturovaný volný text, např. (Enterprise Ontology, 1996):

[LEGAL OWNERSHIP:] a *Relationship* between a LEGAL ENTITY and an *Entity* whereby the LEGAL ENTITY has certain rights with respect to the *Entity*.

Součásti ontologií

- Třídy, koncepty, kategorie...
- Individua, instance, objekty...
- Relace, funkce, sloty, vlastnosti, atributy
- Meta-sloty, omezení na sloty, facety
- Primitivní hodnoty a datové typy
- Axiomy, pravidla

Třídy

- Též koncepty, kategorie...
- Zpravidla uspořádány *hierarchicky*
- Na rozdíl od OO neobsahují *metody* (ale mohou obsahovat atributy...)
- Interpretovány jako *unární relace* na množině objektů
- Příslušnost ke třídám může být *definována* pomocí nutných a postačujících podmínek

Dědičnost tříd v ontologiích vs. OO

- OO: cílem je efektivní znovupoužití *kódu*
- Ontologie: cílem je zachycení *reality*
- Příklad (Farquhar, 1997):
 - třída *kruh* s atributy: *souř-X*, *souř-Y*, *šířka*
 - třída *elipsa*: *kruh*, který má navíc atribut *výška*
 - ale *elipsa* v realitě není speciálním případem *kruhu*!

Individua

- Též instance, objekty
- Na rozdíl od OO mohou příslušet k více třídám nebo naopak (provizorně) k žádné!
- Ontologie primárně neslouží k uchovávání individuí; zařazují se zejména pokud jsou potřebná pro *definici tříd*:
 - „X je instancí třídy *anglicky-hovořící-člověk* právě tehdy, když pro něj vlastnost *mluví-jazykem* nabývá hodnoty ***angličtina***“

Třída nebo individuum?

- Rozlišení tříd a instancí často jen pragmatické
- ontologie chování zvířat:
 - třídy *Klokan*, *Kráva*
 - [individua *Skippy*, *Stračena*]
- ontologie vývoje druhů:
 - třídy *Čeled'*, *Rod*, *Druh*
 - individua *Klokanovití*, *Klokan*, *Klokan rudý*

Relace

- Vyjadřuje vztah mezi n objekty
- Konkrétní n -tice jsou *instancemi relací*
- Příslušnost k relacím může být *definována* pomocí nutných a postačujících podmínek
- Mnoho jazyků omezeno na *binární* relace
 - nazývané *sloty* či *vlastnosti*
 - avšak existují nezávisle na třídách!
- Relace/sloty mohou mít nad sebou definovanou *hierarchii* (jako relaci 2. řádu!)

Funkce, atributy

- Relace, u které je hodnota n -tého argumentu určena předchozími $n-1$
- Funkční *slot* (unární funkce) se označuje jako *atribut*

Omezení na sloty

- [Srv.: obecné n-ární relace mohou být vymezeny libovolným výrazem...]
- Též *meta-sloty*
- *Globální omezení*: vlastnosti slotu jako takového
- *Lokální omezení*: omezení hodnot slotu v případě aplikace na určitou třídu z definičního oboru

Typická globální omezení

- Hierarchický vztah:
 - *je-otcem* IS-A *je-předkem*
 - *sousedem-vpravo* IS-A *sousedí-s*
- Vzájemně inverzní sloty:
 - *je-rodičem* INVERSE *je-dítětem*
 - *má-součást* INVERSE *je-součástí*
- Tranzitivita, symetrie, funkčnost...
- Definiční obor, obor hodnot

Typická lokální omezení

- Též *facety* (či „šablony facet“) slotů: ternární (meta-)relace mezi třídou, slotem, a hodnotou slotu
- Hlavní typy:
 - Omezení oboru hodnot
 - Omezení kardinality
- Příklad (kombinované omezení):
 - „Hodnotou slotu *má-otce* aplikovaného na třídu *osoba* je **právě jedna** instance **třídy *osoba***“

Primitivní hodnoty a datové typy

- Argumenty relací (resp. hodnotami slotů) mohou být nejen objekty, ale i *primitivní hodnoty*
- Obor hodnot *dato-typových* slotů:
 - základní typy – string, integer, float...
 - stanovení výčtem (např. pro barvy...)
- I dato-typové sloty jsou a-priori *relace*:
 - nutno deklarovat jako funkční...
 - ...jinak mohou nabývat více hodnot!

Axiomy, pravidla

- Vedle definic a omezení lze do ontologií zařazovat „volné“ formule:
- *Axiomy* – např.:
 - ekvivalence/subsumpce tříd/relací
 - disjunknost tříd
 - rozklad třídy na podtřídy
 - ...
- *Pravidla*
 - dopředné a/nebo zpětné řetězení...

Další části ontologie

- dokumentační položky k prvkům ontologie
- údaje o vytvoření ontologie
- odkazy na importované ontologie



Jazyky pro reprezentaci ontologií

Konceptuální grafy (CG)

- J. Sowa (1976)
 - viz: John F. Sowa, *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, Brooks Cole Publishing Co., 2000.
<http://users.bestweb.net/~sowa/cg/>
- Logický formalismus zahrnující třídy, relace, individua, kvantifikátory...
- Binarizace relací pomocí *reifikace*
- Abstraktní syntaxe, grafická (DF) a textová (LF) notace, formální jazyk (CGIF)

CG – příklady (1)

„A cat is on a mat.“

DF:

LF: $[Cat] \rightarrow (On) \rightarrow [Mat] .$

CGIF: $[Cat: *x] [Mat: *y] (On ?x ?y)$

CG – příklady (2)

„John is going to Boston by bus.“

DF:

LF: [Go] -

(Agnt) -> [Person: John]

(Dest) -> [City: Boston]

(Inst) -> [Bus].

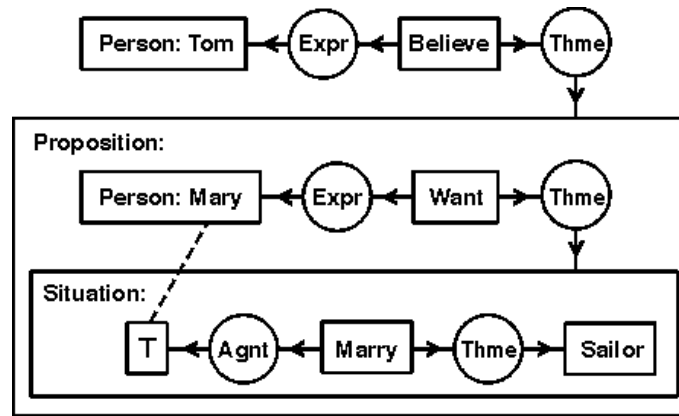
CGIF: [Go *x] (Agnt ?x [Person: John])

(Dest ?x [City: Boston]) (Inst ?x [Bus])

CG – příklady (3)

„Tom believes that Mary wants to marry a sailor.“

DF:



LF: [Person:Tom]←(Expr)←[Believe]→(Thme) -
 [Proposition:[Person: Mary *x]←(Expr)←[Want] →(Thme) -
 [Situation:[?x]←(Agn)←[Marry]→(Thme)→[Sailor]]].

CGIF: [Person: *x1 'Tom'] [Believe *x2] (Expr ?x2 ?x1)
 (Thme ?x2 [Proposition: [Person: *x3 'Mary'] [Want *x4]
 (Expr ?x4 ?x3) (Thme ?x4 [Situation: [Marry *x5] (Agn
 ?x5 ?x3) (Thme ?x5 [Sailor])])])

CycL

- Jazyk vyvinutý pro projekt Cyc:
 - rozsáhlá báze znalostí typu „common sense“
 - jednotlivá tvrzení uspořádána do „mikroteorií“
- Má plnou sílu predikátového kalkulu, obsahuje i prvky vyšších řádů

```

#$ist #NaiveStateChangeMt
  (#$implies
    (#$and
      (#$isa ?FREEZE #Freezing)
      (#$outputsCreated ?FREEZE ?OBJ))
    (#$stateOfMatter ?OBJ
     #SolidStateOfMatter)))

```

Ontolingua

- *Knowledge Systems Lab (KSL), Stanford*
- Jazyk pro sdílení ontologií mezi znalostními systémy
- Základem definice *tříd, relací a funkcí*
- Těla definic vyjádřená v jazyce *KIF*
- Snaha o podchycení nadmnožiny konstruktů různých jazyků reprezentace znalostí (založených na *rámcích*)
→ omezené možnosti přímého odvozování

Ontolingua – definice třídy

```
(Define-Class Sale-Offer (?X)
```

```
"A For-Sale situation with a  
Specified-Potential-Customer"
```

```
:Iff-Def
```

```
(And
```

```
(For-Sale ?X)
```

```
(Exists (?Le)
```

```
(Specified-Potential-Customer  
?X ?Le)))
```

Ontolingua – definice relace

```
(define-relation excretes (?a ?b)
```

```
"some body system or part embodies a function  
which has a body substance as product."
```

```
:def
```

```
(exists (?c) (and (embodies ?a ?c)  
(has-product ?c ?b)))
```

```
:constraints
```

```
(or (anatomical-structure ?a) (organism ?a))
```

```
:axiom-def
```

```
(arity excretes 2)
```

```
:axiom-constraints
```

```
(range excretes body-substance))
```

OCML

- „Operational Concept Modelling Language“ (*Knowledge Media Institute, Open University, UK*)
- Návrh jazyka spojen s efektivní implementací odvozovacích algoritmů:
 - Prologovské zpětné řetězení
 - Dědění v hierarchii tříd
- Nadstavba LISP

OCML – příklady (1)

```
(def-relation date-difference
  (?date-1 ?date-2 ?diff ?diff1))
```

```
(def-rule date-difference
  ((date-difference ?date-1 ?date-2 ?diff ?diff1)
   if (and
       (date ?date-1 day ?d1 month ?m1 year ?y1)
       (date ?date-2 day ?d2 month ?m2 year ?y2)
       (not (== ?date-1 ?date-2))
       (= ?diff1 (+
                  (* (- ?y2 ?y1) 365)
                  (* 30 (- ?m2 ?m1))
                  (- ?d2 d1))))
      (> ?diff1 ?diff))))
```

OCML – příklady (2)

```
(def-class disease ()  
  (  
    (has-name :type string)  
    (is-severe :type boolean)  
  ))
```

```
(def-relation has-disease (?p ?d)  
  :constraint (and (patient ?p) (disease ?d)))
```

```
(def-class seriously-ill (patient) ?p  
  :sufficient  
  (and (has-disease ?p ?d)  
        (disease ?d is-severe yes)  
  ))
```

OCML – příklady (3)

```
(def-class measurement ()
  ((of-patient :type patient)
   (of-parameter :type parameter)
   (in-location :type (or "office" "home"))
   (in-day :type (kappa ?year ?month ?day))
   (has-value :type (or string (kappa ?v1 ?v2))))
))
```

```
(def-class hypertension (blood-pressure) ?bp
  :iffdef
  (and
    (?bp sbp ?sbp)
    (?bp dbp ?dbp)
    (or (>= sbp 140) (>= dbp 90))))
```

OKBC

- *Open Knowledge Base Connectivity*, vznikl od r. 1993
- API pro
 - předávání konstruktů znalostních bází (třídy, individua, sloty, facety...)
 - volání operací nad konstrukty (např. dotazy na hodnoty slotů)
- Snaha o pokrytí (v té době) všech hlavních modelů znalostních systémů založených na *rámčích*

OML

- University of Washington (1999)
- Poprvé syntaxe XML!
- Sémantika *konceptuálních grafů*
 - silnější avšak méně intuitivní než rámcové modely?
 - nepodporuje přímo facety slotů
 - nízká dostupnost nástrojů pro odvozování

XOL

- Vznikl v r. 1999, motivován potřebami *bioinformatické* komunity
- *Rámcový* model (z *OKBC-Lite*), syntaxe *XML*

```
<class>  
  <name>osoba</name>  
</class>
```

```
<slot>  
  <name>věk</name>  
  <domain>osoba</domain>  
  <value-type>integer</value-type>  
  <numeric-max>150</numeric-max>  
</slot>
```

Ontologické jazyky pro webová metadata

- Snaha zabezpečit sémantiku webovým metadatům hlavním stimulem rozvoje ontologických formalismů od konce 90.let
- „Průkopnické“ jazyky: SHOE, Ontobroker
- „Pokročilé“ jazyky: založené na RDF (RDFS, OIL, DAML+OIL, OWL)
- Hlavní aktivity: tvorba ontologií, anotování stránek, sběr metadat, odvozování

SHOE a Ontobroker

- SHOE (Simple HTML Ontology Extension)
 - 1986 J. Hendler, UMBC
 - cíleno na běžné uživatele WWW
 - distribuovaná koncepce: metadata i ontologie umístěny na stránkách, společný jazyk
- Ontobroker
 - 1987 D. Fensel et al., Karlsruhe
 - cíleno na odborné komunity
 - centralizovaná koncepce: metadata od poskytovatelů sbírána do centrální databáze

SHOE - příklad (1)

```
<DEF-CATEGORY NAME="Organization" ISA="base.SHOEntity">
<DEF-CATEGORY NAME="Person" ISA="base.SHOEntity">
<DEF-CATEGORY NAME="Publication" ISA="base.SHOEntity">
<DEF-CATEGORY NAME="ResearchGroup" ISA="Organization">
<DEF-RELATION NAME="member">
  <DEF-ARG POS="1" TYPE="Organization">
  <DEF-ARG POS="2" TYPE="Person">
  </DEF-RELATION>
<DEF-RELATION NAME="publicationAuthor">
  <DEF-ARG POS="1" TYPE="Publication">
  <DEF-ARG POS="2" TYPE="Person">
  </DEF-RELATION>
<DEF-RELATION NAME="publicationDate">
  <DEF-ARG POS="1" TYPE="Publication">
  <DEF-ARG POS="2" TYPE=".DATE">
```

SHOE - příklad (2)

```
<INSTANCE KEY="http://www.cs.umd.edu/users/george/">
  <USE-ONTOLOGY ID="cs-dept-ontology"
    URL="http://www.cs.umd.edu/projects/plus/SHOE/onts/cs.html"
    VERSION="1.0" PREFIX="cs">

  <CATEGORY NAME="cs.GraduateStudent">
  <RELATION NAME="cs.age"> <ARG POS=TO VALUE="52">

  <INSTANCE KEY="http://www.cs.umd.edu/users/george/#BRUNHILDA">
```

Ontobroker - příklad

```

Person :: Object.
Researcher :: Person.
Person [lastName =>> STRING;
      ...;
      publication =>> Publication].
FORALL Per, Pub
  Pub:Publication [author ->> Per]
  <-> Per:Person [publication ->> Pub]

```

```

<a onto="page:Researcher"></a>
<a onto="page[affiliation=href]"
  href="http://www.iiia.csic.es/">
  IIIA - Artificial Intelligence Institute </a>
<a onto="page[firstName=Body]">Richard</a>

```

Sémantický web

- Vize Tima B. Lee: spojení otevřené webové architektury s technologií inteligentních agentů
- Informační zdroje sémantického webu: 5 vrstev
 - syntaxe *XML*
 - datová struktura *RDF*
 - sémantika („meaning“) specifikována pomocí *ontologií*
 - *logické odvozování* realizováno *pravidly*
 - mechanismus zabezpečení *věrohodnosti* („trust“)
- Podpora W3C, DARPA, EC...

Příklad aplikace

- Automatizace transakcí B2B, např.:
 - vyhledávání partnera na základě metadat na jeho WWW stránkách
 - zjištění věrohodnosti metadat pomocí metadat z referenčního zdroje
 - výměna formalizovaných kontraktačních principů (ve formě věcných pravidel)
 - automatizované sladění návrhů v několika iteracích

Resource Description Framework (RDF)

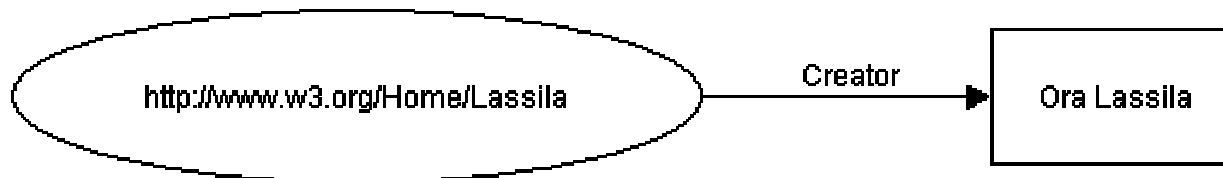
- Standard webových metadat proponovaný W3C (doporučení z února 1999)
- Zahrnuje jednoduchou konceptuální nadstavbu – *RDF Schema* (posl. verze duben 2002)
- Obecný model; v praxi využívá syntaxi XML, vedle toho existuje
 - grafická notace
 - jednoduchá textová notace (N3)

RDF – základní pojmy

- *Zdroj* („resource“):
entita, o které chceme pomocí RDF něco sdělit
 - identifikován pomocí *URI*
(Universal Resource Identifier – zobecnění URL)
 - nemusí být přístupný na webu!
- *Tvrzení* („statement“):
trojice *subjekt-predikát-objekt*
 - „vlastnost <predikát> nabývá pro <subjekt> hodnoty <objekt>“
 - subjektem je vždy zdroj
 - objektem je buď zdroj nebo primitivní hodnota (literál)

RDF - příklad

- Ora Lassila is the creator of the resource <http://www.w3.org/Home/Lassila>.



```

<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description
    about="http://www.w3.org/Home/Lassila">
    <dc:Creator>Ora Lassila</dc:Creator>
  </rdf:Description>
</rdf:RDF>
  
```

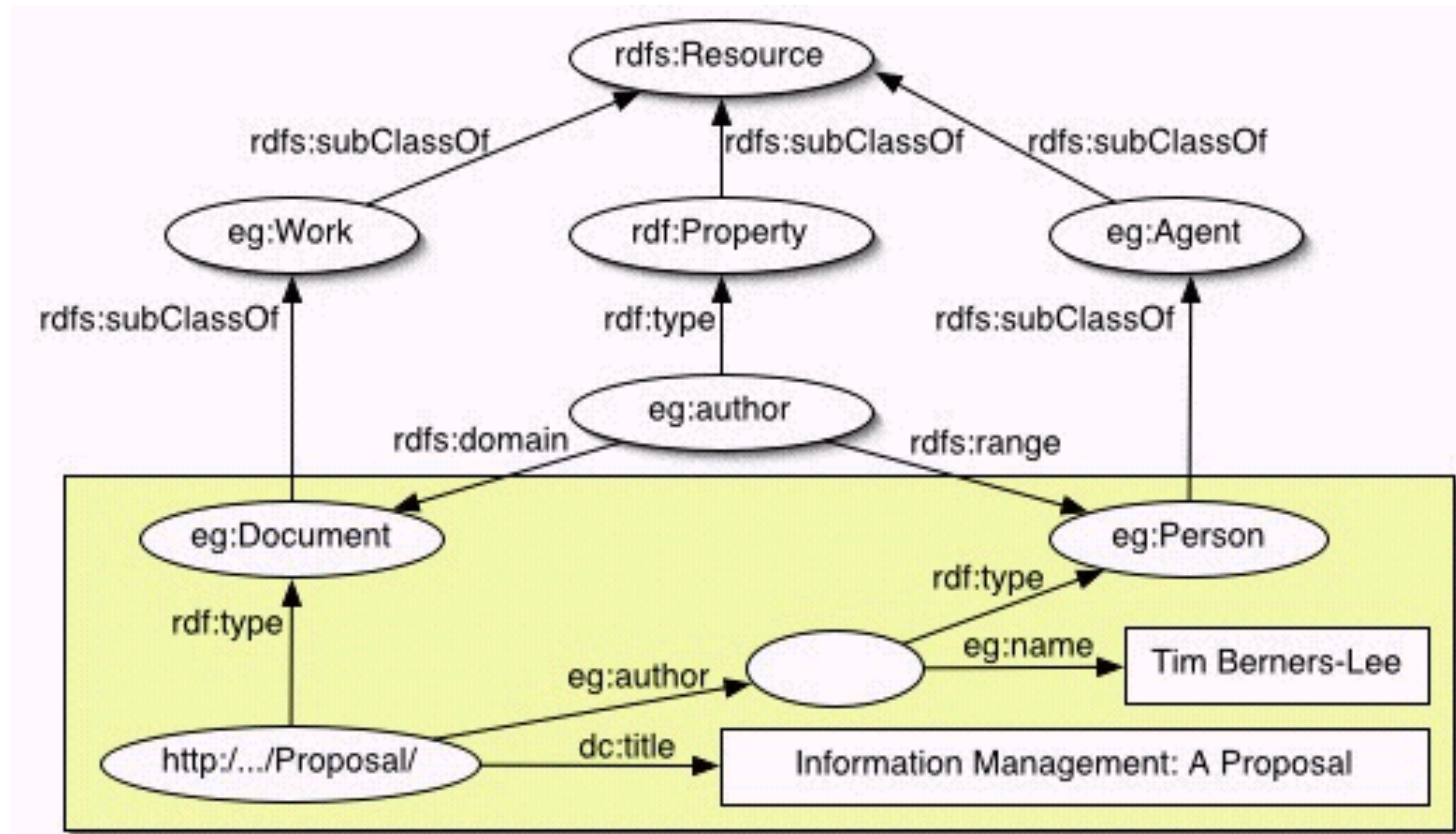
RDF - další možnosti

- sdružování zdrojů do *kolekcí* (“container”)
- *reifikace* - možnost formulovat tvrzení o tvrzeních
- zachycení relací o vyšší aritě (např. odlišení “hlavní” hodnoty)
- “typování” zdrojů (rozdělení do tříd) pomocí *RDF Schema*

RDF Schema

- Definuje *třídy* zdrojů a sémantiku jejich *vlastností*
- Standard zahrnuje možnost specifikovat:
 - definiční obor a obor hodnot vlastnosti
 - vztah třídy a podtřídy, vlastnosti a “podvlastnosti”
- Jednoduchý ontologický jazyk, postrádá zvl. specifikaci *omezení*

RDFS - příklad



OIL

- Výsledek evropské spolupráce (2000)
- První ontologický jazyk založený na kombinaci rámcového pohledu s *deskripční logikou*
- Oproti RDFS zahrnuje *lokální omezení*, i meta-sloty typu “inverse”, “transitive”...
- Vrstvená architektura: maximální využívání vlastností definovaných v RDF/S
- Spojením s DAML-ONT (DARPA, USA) vznikl *DAML+OIL*

Deskripční logika

- Vlastně logiky - rodina formalismů, jejichž jádrem je vymezení konceptů pomocí jejich formálních popisů (deskripcí)
- Na základě popisů zjišťována *subsumpce konceptů*
- Též *terminologická* logika

DAML+OIL

- Momentálně asi nejpopulárnější ontologický jazyk
- Ontologie je souborem *axiomů* o *třídách*
- *Konstruktory tříd* zhruba odpovídají konceptuálním výrazům v DL, navíc však umožňují pracovat s instancemi

DAML+OIL - příklad (1)

```

<?xml version='1.0' encoding='ISO-8859-1'?>
<!DOCTYPE rdf:RDF [
  <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
  <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
  <!ENTITY daml 'http://www.daml.org/2001/03/daml+oil#'>
  <!ENTITY old 'http://www.daml.org/xyz/20000911.daml#'>
  <!ENTITY xsd 'http://www.w3.org/2000/10/XMLSchema#'>
]>
<rdf:RDF xmlns:rdfs="&rdfs;"
         xmlns:rdf="&rdf;"
         xmlns:daml="&daml;">

<daml:Ontology rdf:about="">
  <daml:versionInfo>xyz</daml:versionInfo>
  <rdfs:comment>Ontology to describe organizations and      individuals
    participating in a government R&D program.</rdfs:comment>
</daml:Ontology>

```

DAML+OIL - příklad (2)

```
<rdfs:Class rdf:ID="Agency">
  <rdfs:subClassOf rdf:resource="#Organization"/>
  <daml:sameClassAs rdf:resource="#old;Agency"/>
  <rdfs:subClassOf>
    <daml:Restriction>
      <daml:onProperty rdf:resource="#name"/>
      <daml:toClass
rdf:resource="http://www.w3.org/2000/10/XMLSchema#string"/>
    </daml:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <daml:Restriction>
      <daml:onProperty rdf:resource="#partOf"/>
      <daml:toClass rdf:resource="#Agency"/>
    </daml:Restriction>
  </rdfs:subClassOf>
</rdfs:Class>
```

OWL - Web Ontology Language

- Vyvíjen W3C Ontology Working Group
- Vychází z DAML+OIL
- Verze *OWL Lite* přináší zjednodušení
 - chybí anonymní třídy
 - omezení na kardinalitu 0-1

UML jako ontologický jazyk

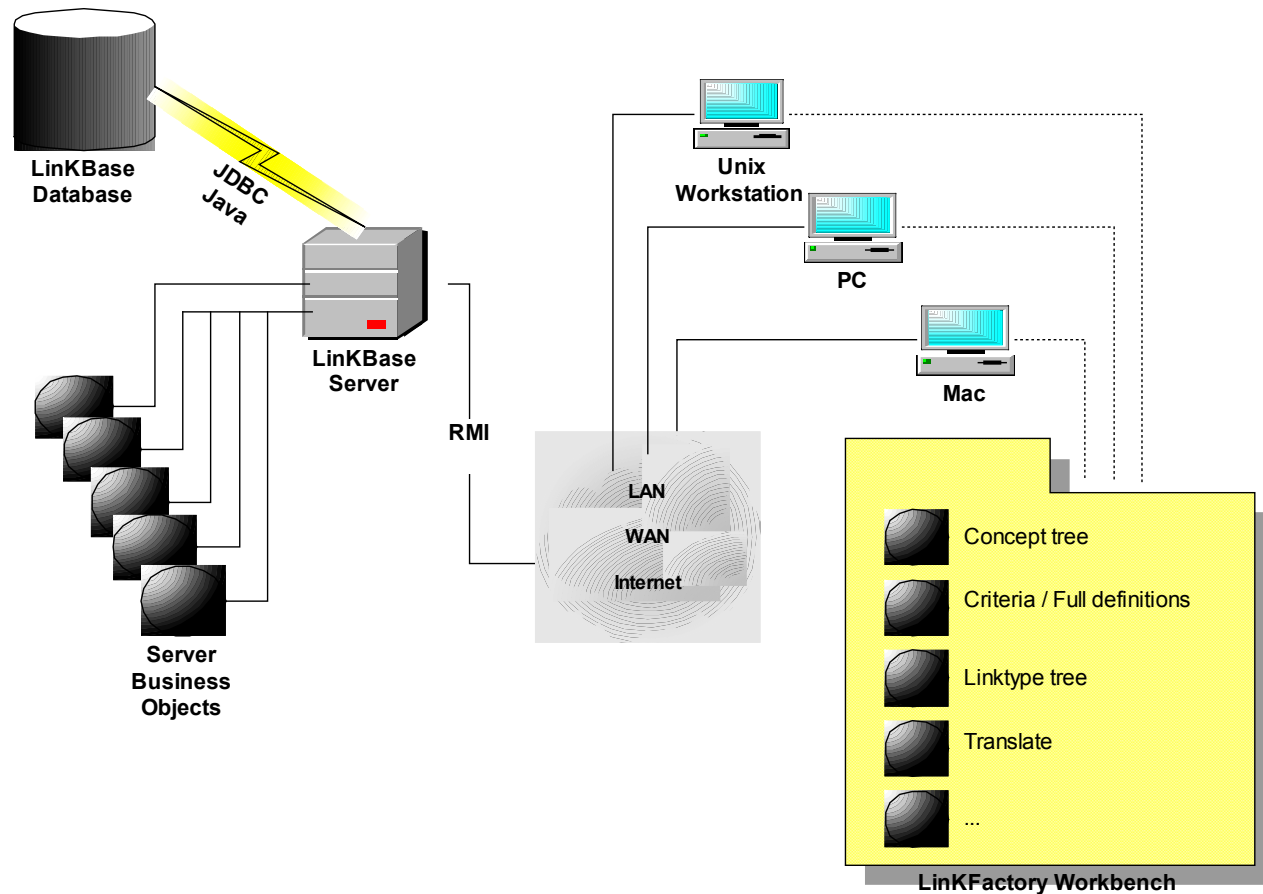
- Třídy (bez metod) s dědičností, relace (asociace), sloty (atributy)
- Výhodou rozšířenost
- Nevýhody:
 - malá schopnost vyjádřit omezení (zvl. lokální) aj. meta-informace
 - závislost vlastností a instancí na třídách
- Závěr: lze použít v úvodní fázi modelování, ale nemůže nahradit formální jazyk



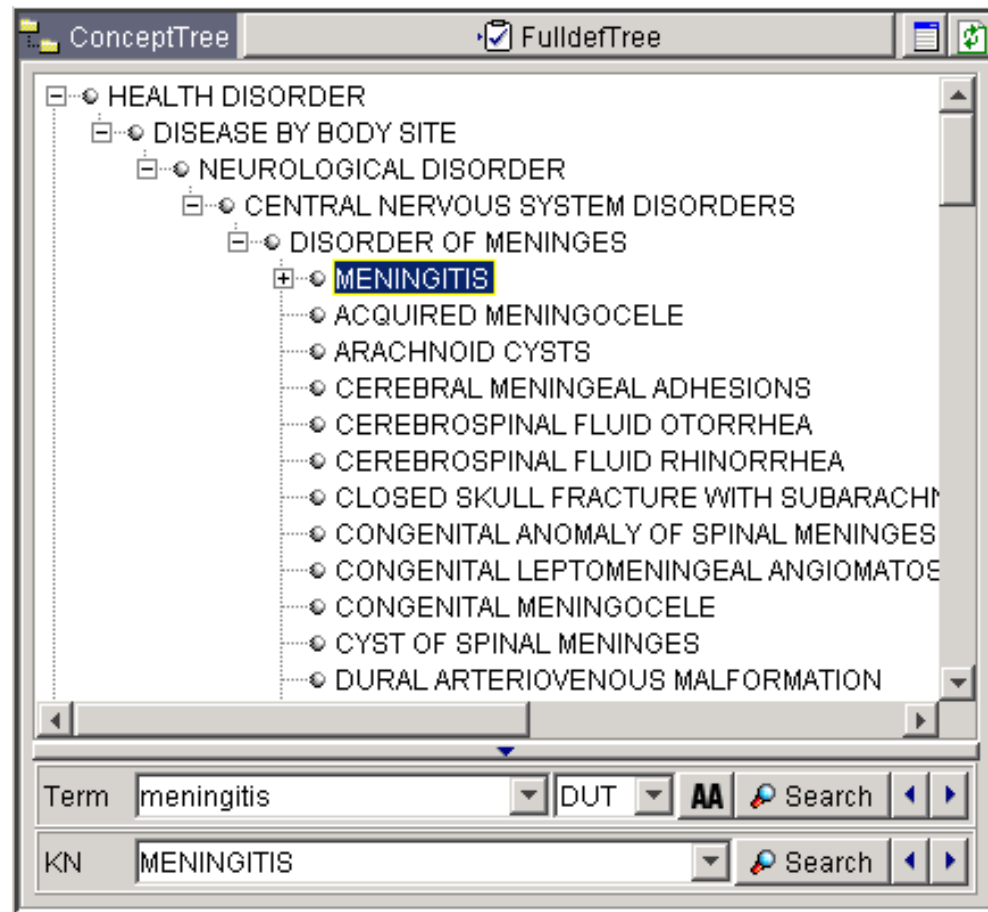
Aplikace ontologií

LinkFactory[©]

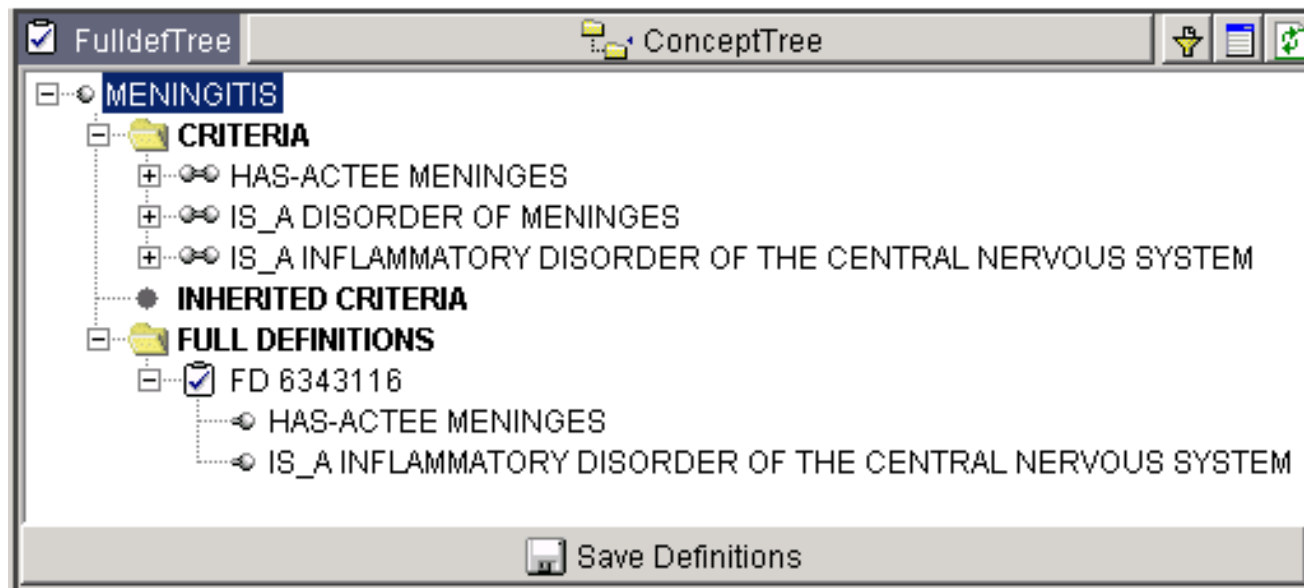
Language & Computing, Belgie



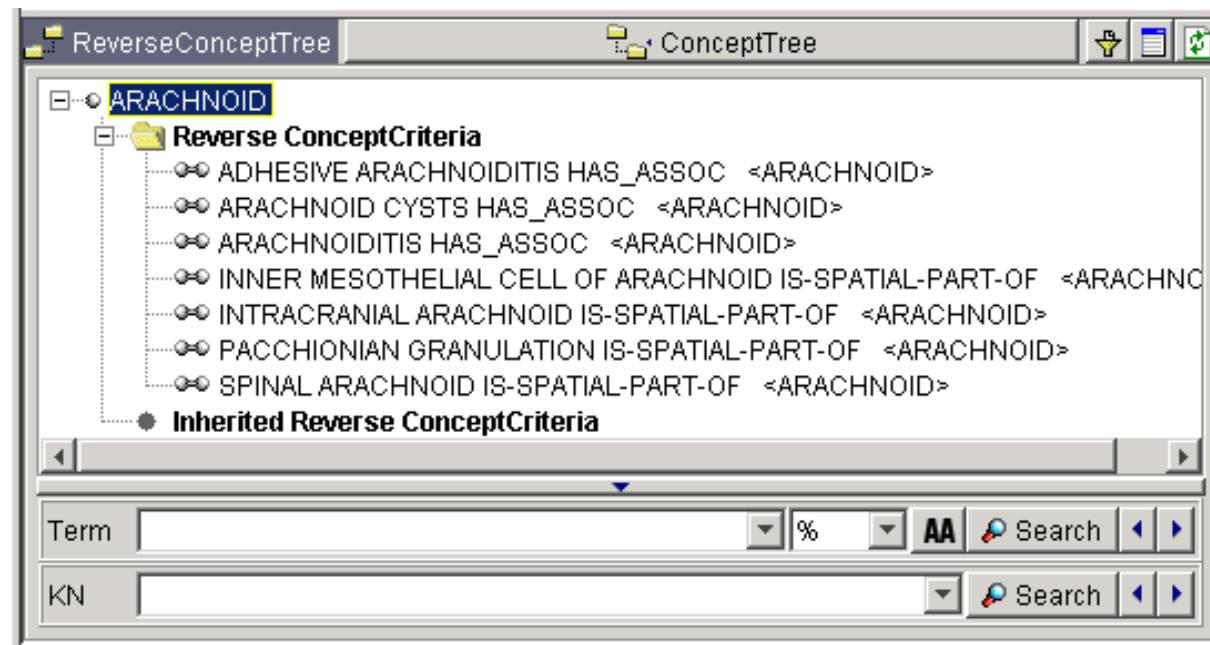
LF – concept tree bean



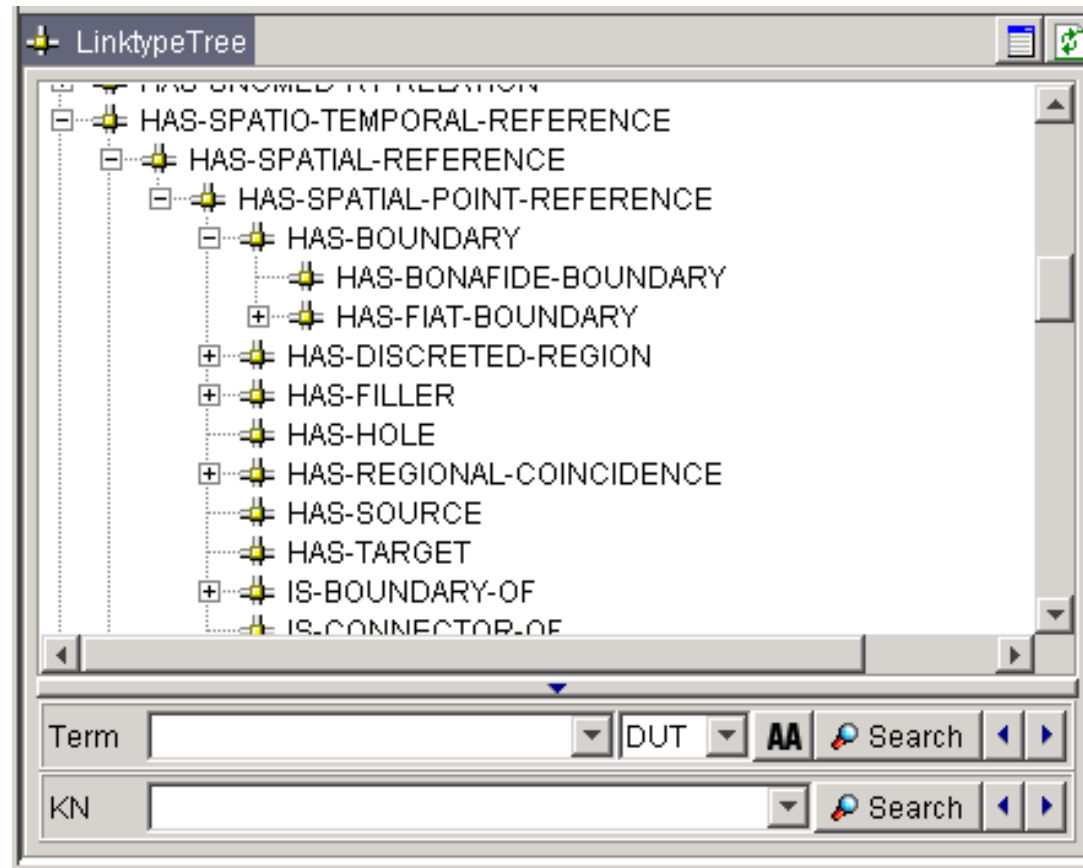
LF – full definition bean



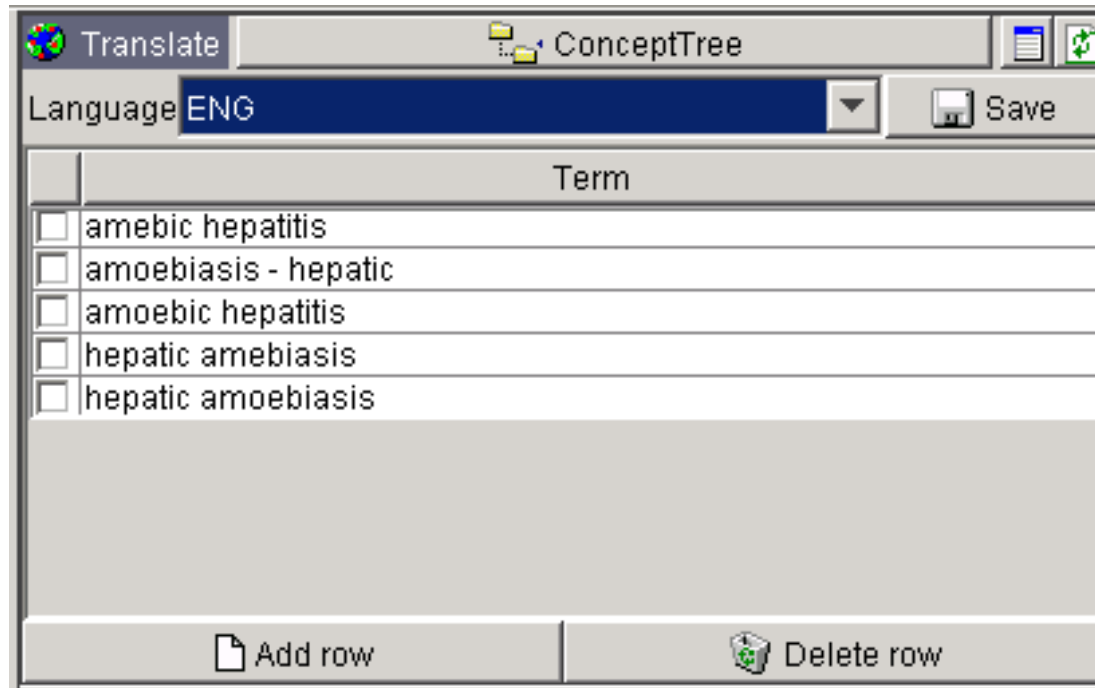
LF – reverse concept bean



LF – link type tree bean



Translation bean





Problémy a perspektivy ontologií v rámci sémantického webu

Vybrané problémy

- RDF versus “čistý” XML?
- Vyjadřovací síla jazyků
- Role automatické inference
- Koexistence RDF a HTML?
- Využití existujících terminologických zdrojů

RDF versus “čistý” XML?

- Zápis ontologických konstruktů vyžaduje dvě úrovně kódování:
 - RDF (trojice)
 - XML (serializace do stromové struktury)
- P.Patel-Schneider (Bell Labs, USA):
 - XML nemá sémantiku XML, ale ani RDF sémantiku RDF (!) - “dark triples”...
 - V XML mnohem více dat, než v RDF
 - Propojit data v XML pomocí URI zdrojů RDF!

Příklad “dark triples”

```
<oil:DefinedClass rdf:about=
  "http://x.y.com/onto.rdfs#small_company">
  <oil:hasPropertyRestriction rdf:resource="_anon1"/>
  <rdfs:subClassOf rdf:resource=
    "http://x.y.com/onto.rdfs#company"/>
</oil:DefinedClass>

<oil:HasValue rdf:about="_anon1">
  <oil:toClass rdf:resource="_anon2"/>
  <oil:onProperty rdf:resource=
    "http://x.y.com/onto.rdfs#ann-turnover"/>
</oil:HasValue>

<oil:Max rdf:about="_anon2"
  oil:integerValue="1000000"/>
```

Vyjadřovací síla jazyků

- Tradiční ontologické jazyky *rozšířením* predikátového kalkulu, webové jazyky spíše jeho *restrikcí* (binární relace...)
- Jednoduchost podmínkou širšího využívání (srv. SGML versus XML!)
- Nutnost existence více verzí jazyka (OWL vs. OWL Lite...)

Role automatické inference

- Nutnost kompromisu mezi “konceptuálním” a “logickým” pohledem (OIL vs. DAML)
- Otevřené odvozování vs. předpřipravená schémata

Koexistence RDF a HTML?

- RDF je pro koncového uživatele netriviální, nutno vyjít z jeho znalosti HTML
 - (Radikální názor - O. Etzioni:) začít od “lokálních sémantických webů” s proprietárními značkami, postupně sjednocovat
 - (Umírněný názor - S. Haustein:) generovat RDF jako vedlejší produkt tvorby HTML

Využití existujících zdrojů

- Většinou nelze pro sémantický web vytvářet ontologie na zelené louce, nutno využít existující zdroje!
- Standardy:
 - knihovnická schémata (Dublin Core), produktové katalogy (UNSPSC), lékařské tezaury (UMLS) apod.
 - často nekonzistentní - nutno předzpracovat
- Ad hoc zdroje:
 - informační stránky, webové katalogy apod.
 - ještě méně spolehlivé; nelze převzít, ale jen využít pro “učení ontologií”, “dolování sémantického webu” apod.

Další příležitost slyšet něco o ontologiích a sémantickém webu:

Frank van Harmelen (VU Amsterdam):
The Semantic Web

Zvaná přednáška na konferenci Znalosti 2003

Ostrava, 19.-21.2.2003

<http://www.cs.vsb.cz/zna12003>

(Možnost nabídnout referát na tuto konferenci, do 31.10.!))



Děkuji za pozornost

Podrobnější text k problematice:

<http://nb.vse.cz/~svatek/onto-www.doc>

Ještě dotazy?