

Znalostní modelování

Vojtěch Svátek, Martin Kavalec, Martin Labský

KIZI VŠE

Tématické okruhy vystoupení

- Pojem a historie znalostního modelování (5')
- Modely znalostních úloh/aplikací (15')
- Ontologie jako informační artefakt (30')
- Znalostní inženýrství a meta-modelování (5')
- Aktuální problémy tvorby a využívání ontologií (20')

Celkem 75 minut vč. průběžné diskuse ???

Osnova vystoupení (1)

- Modely znalostních úloh (přístup KADS)
 - tří/čtyřvrstvá architektura modelu expertizy
 - znovupoužitelné šablony modelů
 - tvorba *formálních/proveditelných specifikací*
 - komplexní *metodika* (CommonKADS)

Osnova vystoupení (2)

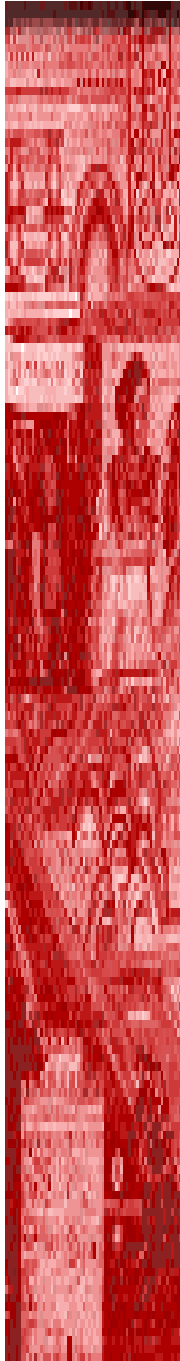
- Ontologie jako informační artefakt
 - posun vůči *filosofickému chápání pojmu*
 - typické prvky ontologií, srovnání s *UML*
 - Podobnosti a rozdíly jazyků
 - Rozšíření meta-modelu UML ve směru k ontologiím
 - *grafická reprezentace*
 - formální základ ve výpočtové logice
 - role v sémantickém webu (syntaxe *XML/RDF*)

Osnova vystoupení (3)

- Znalostní inženýrství a *meta-modelování*
 - meta-modelování znalostních metod a nástrojů?
 - ontologie jako podpora meta-modelování?

Osnova vystoupení (4)

- Aktuální problémy tvorby a využívání ontologií
 - standardizace jazyka
 - zajištění adekvátního obsahu



Základní pojmy ontologického inženýrství

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)

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 definovanu *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-rodicem* 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

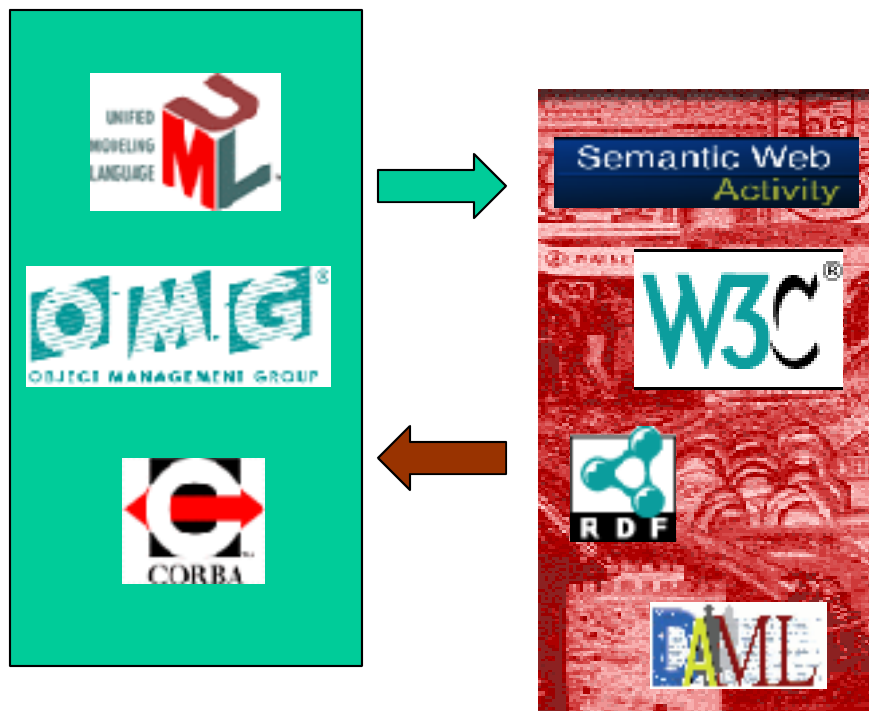
- Vedle definic a omezení lze do ontologií zařazovat „volné“ formule:
- *Axiomy* – např.:
 - ekvivalence/subsumpce tříd/relací
 - disjunktnost tříd
 - rozklad třídy na podtřídy
 - ...
- *Pravidla*
 - dopředné a/nebo zpětné řetězení...

Další části ontologie

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

Objektové modely a znalostní ontologie

podobnosti a rozdíly



Vojtěch Svátek
Martin Labský

Vysoká škola ekonomická
v Praze

katedra informačního
a znalostního inženýrství

svatek@vse.cz

labsky@vse.cz

Struktura přednášky

- Ontologie a sémantický web
 - Pojem ontologie
 - Role RDF a ontologií v sémantickém webu
- Objektové modely a ontologie
 - Podobnosti a rozdíly
 - Možnosti spolupráce objektové a ontologické komunity

Ontologie

- Původně (ve filosofii) věda o “bytí”
- V informatice se *ontologií* nazývá určitý soubor informací - tzv. *formální specifikace sdílené konceptualizace*
 - *konceptualizace*: abstraktní model určité oblasti - soubor pojmů a vztahů mezi nimi
 - *formální*: vyjádřená ve formálně-logickém jazyce, zpracovatelná počítačem
 - *sdílená*: je výsledkem dohody více subjektů
- <http://nb.vse.cz/~svatek/onto-www.pdf>

Koncepce sémantického webu

- Tim Berners-Lee: aby web nebyl jen pro lidi, ale i pro počítače, musí být schopen *formálně reprezentovat* informace a definovat jejich *význam*
- Jádrem současné koncepce sémantického webu jsou data reprezentovaná v jazyce *RDF*, s významem definovaným pomocí *ontologií*

RDF

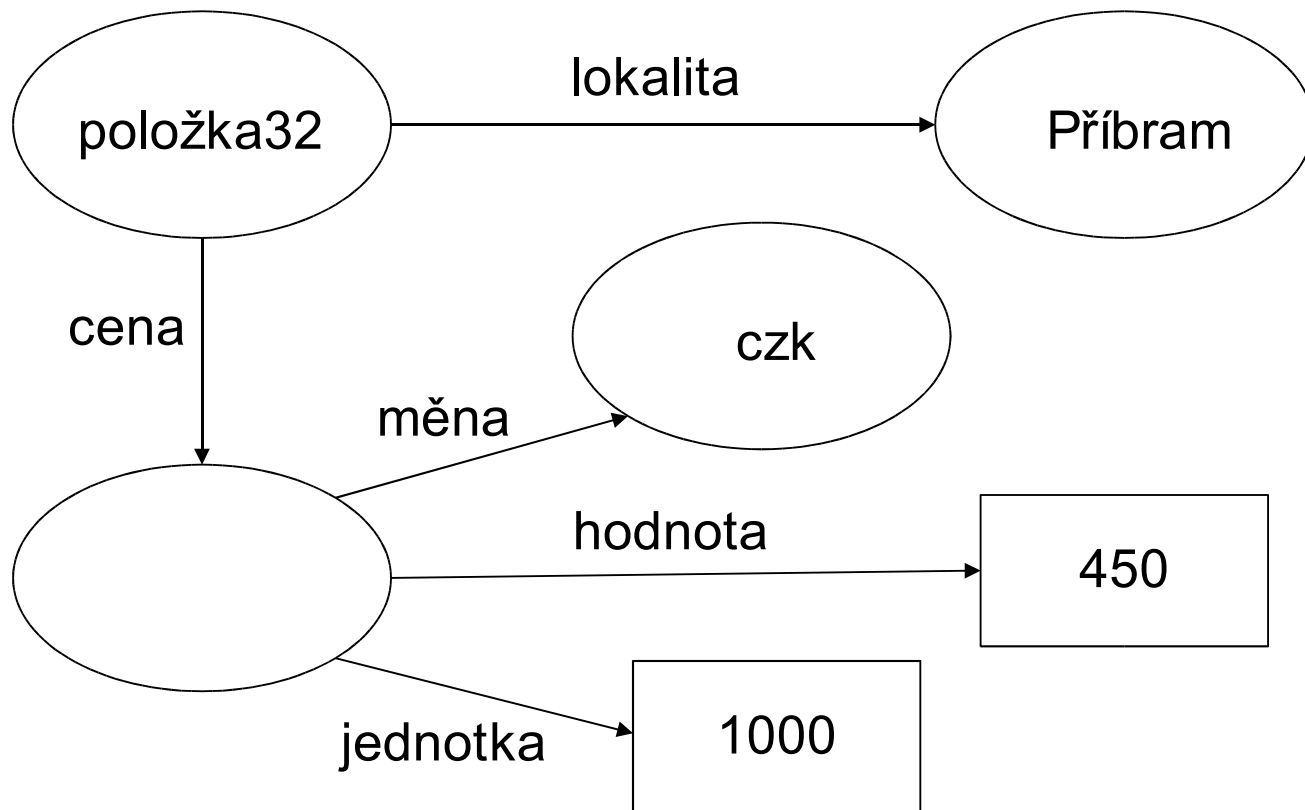
- “Resource Description Framework”
- Doporučení konsorcia W3C
<http://www.w3.org/RDF/>
- Jednoduchý jazyk, v němž je možné vyjádřit tvrzení typu “Zdroj X nabývá pro vlastnost Y hodnoty Z” - tzv. *trojice* (“triple”) *subjekt-predikát-objekt*

- Např.:

	<i>subjekt</i>	<i>predikát</i>	
	položka32	lokalita	
Příbram			
	položka32	cena	X32
	X32	měna	czk
	X32	1	150

RDF

grafická notace



RDF versus XML

- *modulární* (trojice na sobě nezávislé)
- subjekty, predikáty i některé objekty jsou *zdroje* s jednoznačným *identifikátorem* - URI (Uniform Resource Identifier)
- trojice = fakta o světě, kterým lze přiřadit *pravdivostní hodnotu*; nejde jen o strukturu dat jako v případě XML stromů
- samotné RDF ovšem stále nestačí pro strojové *odvozování* nových informací!

XML syntaxe RDF

- RDF lze zapisovat (serializovat) pomocí XML, např.:

```
<rdf:RDF>
  xmlns:r="http://www.reality.cz/
  <rdf:Description
    about="http://www.real-
a.cz/polozka32">
    <r:Lokalita
      rdf:resource="http://www.mistopis.cz/Pří-
      bram"/>
    </rdf:Description>
</rdf:RDF>
```

Subjekt

Predikát

Objekt

RDF a ontologie

- Nová tvrzení můžeme odvodit tehdy, když konkrétní zdroje přiřadíme k obecným *třídám* jakožto jejich *instance* → vlastnosti tříd pak musí platit pro všechny instance!
- Struktura tříd a jejich vlastnosti mohou být definovány v *ontologiích*
- Hlavní jazyky pro reprezentaci webových ontologií:
 - *RDF Schema*: jednoduchý jazyk pro definování hierarchií tříd a vlastností
 - *OWL*: jazyk s bohatšími vyjadřovacími možnostmi, založen na *deskripční logice*

Příklad části ontologie v OWL

```
<owl:Class rdf:ID="2+1">
  <rdfs:subClassOf rdf:resource="Byt" />
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty
        rdf:resource="ma_soucast"/>
      <owl:someValuesFrom
        rdf:resource="Kuchyň"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Třída “2+1” je podtřídou třídy “Byt”, a každá její instance musí být spojena relací “ma_soucast” s alespoň 1 instancí třídy “Kuchyň”

Příklad 1: Vyhledávání

- *Zákazník* hledá levnější byt ve středních Čechách, s kuchyní
- *Realitka* má k dispozici 2+1 v Příbrami za 450 tisíc Kč
- Vyhledávání klíčovými slovy “*střední Čechy AND levné AND kuchyně*” má malou šanci uspět!
- Logickým odvozováním s využitím ontologie lze položku zhodnotit jako vyhovující požadavku:
 - Příbram je ve středních Čechách, a relace “být v” je tranzitivní
 - všechny 2+1 obsahují kuchyň

Příklad 2: Mapování katalogů

- Realitky se rozhodly vyměňovat si informace z katalogů

Realitní kancelář A

```
<polozka>
  <typ>3+1</typ>
  <lokalita>
    Praha-
    Vršovice
  </lokalita>
  <cena
mena="czk">
2 200 000
</cena>
</polozka>
```

Realitní kancelář B

```
<polozka>
  <typ>prodej</typ>
  <druh>2+1</druh>
  <okres>Příbram</okre
s>
  <lokalita>
    tichá, dobrý
    přístup
  </lokalita>
  <cena mena="czk">
450
```

- Mapování katalogů podle názvu položek v XML by ztotožnilo např. obě varianty vlastnosti "typ"

Příklad 2: Mapování katalogů

- Když využijeme (dvě nezávislé!) ontologie s např. ručně porovnanými nejobecnějšími třídami:
 - položky v katalogu A jsou přiřazeny k třídám “1+1”, “2+1” apod., v ontologii A; tyto třídy jsou spojeny s třídami např. “kuchyň” či “místnost”
 - položky v katalogu B jsou přiřazeny k třídám “prodej”, “pronájem” apod., v ontologii B; tyto třídy jsou spojeny s třídami např. “prodávající” či “vlastník”
- Vzhledem ke *odlišnému kontextu* tříd v ontologii

Aktuální směry výzkumu

- *Automatické získávání* logických faktů (RDF trojic) z běžných WWW stránek
- *Pravidlové odvozování*: umožňuje rozsáhlejší inference, než samotné ontologie
- Práce s *přibližnou a vágní* informací
- ... a mnoho dalších

Odezva z praxe

- Zatím není rozsáhlá, nedostatek aplikací
- Velké firmy pozorně sledují, jejich výzkumné laboratoře se podílejí na vývoji standardů
- Často spojováno s webovými službami a automatizací business procesů (BPEL...)
- Očekávána slučitelnost s objektovými přístupy (?)

Podobnosti OM a ontologií

[OM chápaný jako UML, zvl. diagram tříd

Ontologie reprezentovány zejména jazykem OWL]

- Snaha účelově zachytit strukturu reálného světa
- Třídy, instance (\rightarrow individua)
- Dědění v hierarchii
- Asociace (\rightarrow objektové vlastnosti)
- Atributy (\rightarrow datotypové vlastnosti)
- Omezení na kardinalitu

Odlišnosti ontologií oproti OM

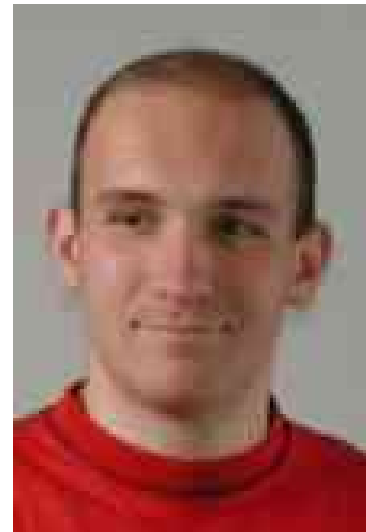
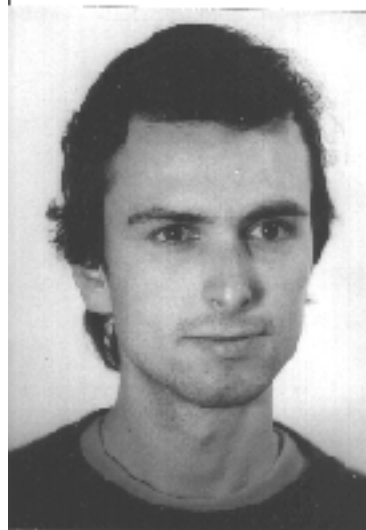
- Vlastnosti a individua mají samostatné postavení
- Třídy mohou být intenzionálně definovány
- Nejen pojmenované, ale i anonymní třídy

- Obtížně se definují omezení na datotypových vlastnostech
- Statické (oproti např. stavovým diagramům)

Možnost spolupráce?

- Přínosy OM pro ontologie
 - Transformace knihoven OM na ontologie
 - Tvorba ontologií pomocí OO nástrojů
 - Sdílení zkušeností s výukou modelování
 - Inspirace v oblasti podpory spolehlivosti
- Přínosy ontologií pro OM
 - „Znalostní“ vyhledávání v knihovnách OM
 - Slučování nezávisle vzniklých OM

Děkujeme za pozornost



Dotazy?