

# Ontologie

## 2. část - ontol. jazyky



14. 5. 2010

**PŘEDMĚT: ORGANIZACE ZNALOSTÍ**

**PŘEDNÁŠEJÍCÍ: SILVIE KOŘÍNKOVÁ PRESOVÁ**

# Jazyky pro reprezentaci ontologií



- ontology languages
  - Formalita ontologie (tj. strojově čitelná) je založena na jazyku pro reprezentaci, který poskytuje potřebné strojově zpracovatelné kódování.
  - Současné výzkumné aktivity – standardizace
- tři kategorie jazyků (Slavic):
- logic based
  - frame based
  - web based
- } primárně určeny pro aplikace z oblasti umělé inteligence

# Jazyky pro reprezentaci ontologií



- Jazyky spjaté s oblastí umělé inteligence – příklady  
(Svátek)

## Ontolingua

<http://www.ksl.stanford.edu/software/ontolingua/> -

Základními konstrukty jazyka jsou definice tříd, relací a funkcí, „mezijazyk“ pro rámcové znalostní systémy.

(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))))

# Jazyky pro reprezentaci ontologií



- Jazyky spjaté s oblastí umělé inteligence – příklady  
CyCL (<http://www.opencyc.org/>) sloužící k tvorbě rozsáhlé všeobecné ontologie CyC  
<http://sw.opencyc.org/> - Usiluje o shromáždění všeobecných znalostí („common sense“), které by ve znalostních systémech fungovaly komplementárně ke znalostem expertním a zabraňovaly absurdnímu chování.

# Webové ontologické jazyky (Svátek)



- Vznikly za účelem přidání sémantiky k webovým stránkám.
- vývojově starší jazyky – SHOE, Ontobroker
- novější jazyky
  - RDF Schema
  - DAML, OIL
- sloučením vzniká DAML+OIL (opírá se o deskripční logiku – podřazení tříd je vyhodnocováno na základě jejich popisů – vznik taxonomie dynamicky X apriorní vymezení vztahů)
- OWL (Web Ontology Language)

# Web Ontology Language - OWL



- OWL - jazyk pro definování, publikování a sdílení webových ontologií.
- doporučení W3C (spolu s RDF)  
<http://www.w3.org/2004/OWL/>
- OWL je součástí W3C Semantic Web Activity.  
<http://www.w3.org/2001/sw/>
- Ontologie OWL - popis tříd, vlastností a jejich instancí.
- OWL - konstruovaný pro užití aplikacemi, které zpracovávají obsah webových zdrojů.

# Web Ontology Language - OWL



- OWL syntax a formální sémantika - součástí slovníku, který vychází z RDF.
- **OWL formální sémantika** umožňuje specifikovat, jak odvozovat logické důsledky dané danou ontologií, např. fakta písemně neprezentovaná v ontologii, ale vyplývající z daných významů.

# Druhy OWL

- OWL Lite - základní verze, tvorba hierarchických struktur, definování vlastností a základních omezení, vhodné při převodu tezaurů a klasifikačních systémů na ontologie (např. nelze použít operátory `intersectionOf`, `unionOf`, `complementOf`)
- OWL DL - maximální vyjadřovací síla a realizace odvozování, obsahuje všechny konstrukce jazyka, založen na deskripční logice
- OWL Full - maximální vyjadřovací síla, volná syntaxe (nezávislost na RDF), odvozování je obtížné (Důvodem jsou omezení současných DL odvozovačů (klasifikátorů))



# Jak číst jazyk ontologie v OWL



- Ukázky na ontologii

<http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#>

- Lze stáhnout a otevřít pomocí textového editoru.

# Jak číst jazyk ontologie v OWL - deklarace typu dokumentu

```
<!DOCTYPE rdf:RDF [
```

```
<!ENTITY vin "http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#" >
```

```
<!ENTITY food "http://www.w3.org/TR/2003/PR-owl-guide-20031209/food#" >
```

```
<!ENTITY owl "http://www.w3.org/2002/07/owl#" >
```

```
<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
```

- V rámci ontologie se odkazuje na jednotlivé hodnoty atributů pomocí URIs.
- **Deklarace (definice) entity v rámci deklarace typu dokumentu**  
- Pomocí ní lze nahradit identifikátor zkratkou.

„&vin;merlot“ za "<http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#merlot>"

Příklad užití: <owl:Class rdf:ID="Wine">

<rdfs:subClassOf rdf:resource="&food;PotableLiquid" /> místo

<rdfs:subClassOf rdf:resource="http://www.w3.org/TR/2003/PR-owl-guide-20031209/food#PotableLiquid" />

# Namespace - jmenný prostor



```
<rdf:RDF
```

```
  xmlns    = "http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#"
  xmlns:vin = "http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#"
  xml:base  = "http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#"
  xmlns:food= "http://www.w3.org/TR/2003/PR-owl-guide-20031209/food#"
  xmlns:owl = "http://www.w3.org/2002/07/owl#"
  xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs= "http://www.w3.org/2000/01/rdf-schema#"
  xmlns:xsd  = "http://www.w3.org/2001/XMLSchema#">
```

- Je potřeba přesně indikovat jaké specifické slovníky byly použity.
- Standardním počátečním prvkem ontologie je soubor deklarací jmenného prostoru XML (XML namespace) uzavřených tagem `rdf :RDF`.

# Záhlaví ontologie



```
<owl:Ontology rdf:about="">  
  <rdfs:comment>An example OWL ontology</rdfs:comment>  
  <owl:priorVersion>  
    <owl:Ontology rdf:about="http://www.w3.org/TR/2003/CR-owl-guide-20030818/wine"/>  
  </owl:priorVersion>  
  <owl:imports rdf:resource="http://www.w3.org/TR/2003/PR-owl-guide-20031209/food"/>  
</owl:Ontology>
```

- Metadata, informace o ontologii, např. informace o verzi ontologie, komentáře, převod z jiných formátů, zahrnutí jiných ontologií, vkládají se různá metadata vycházející se standardu Dublin Core.

# Třídy v OWL



- třídy v OWL - předdefinované, vytvořené uživatelem
- Předdefinovanou třídou je `owl:Thing`. Každé individuum v OWL je členem třídy `owl:Thing`. Tudíž každá třída definovaná uživatelem je její podtřídou.

# Popisy tříd



Typy popisů tříd :

1. identifikátor třídy (odkaz URI)
2. vyjmenování individuů (konstrukt oneOf)
3. omezení vlastností (property restriction)
4. průnik (intersection) dvou nebo více popisů tříd
5. sjednocení (union) dvou nebo více popisů tříd
6. komplement (complement) popisu třídy

# Popisy tříd - identifikátor třídy



```
<owl:Class rdf:ID="Wine">
```

```
....definice třídy
```

```
</owl:Class>
```

```
<owl:Class rdf:ID="Winery"/>
```

indikace jména třídy

V rámci daného dokumentu je na danou třídu, např. **Wine** odkazováno **#Wine**, např.

```
<owl:Class rdf:ID="WhiteWine">
```

```
  <owl:intersectionOf rdf:parseType="Collection">
```

```
    <owl:Class rdf:about="#Wine" />
```

Z vnějšku (jiné ontologie) pomocí celé formy URI,

```
"http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#Wine".
```

Odkaz na prvek z jiné ontologie (např. importované), pokud je deklarována v jmenném prostoru a v rámci definice entity např.

```
  rdf:resource="&food;PotableLiquid"
```

# Popisy tříd - omezení vlastností



```
<owl:Class rdf:ID="Wine">  
  <rdfs:subClassOf rdf:resource="&food;PotableLiquid" />  
  <rdfs:subClassOf>  
    <owl:Restriction>  
      <owl:onProperty rdf:resource="#hasMaker" />  
      <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>  
    </owl:Restriction>  
  </rdfs:subClassOf>
```

- Omezuje rozsah vlastnosti ve specifických kontextech různými způsoby.
- Typy omezení: `allValuesFrom`, `someValuesFrom`, `owl:cardinality`, `hasValue`



# Typy omezení vlastností (Property Restrictions)



- **allValuesFrom** (Všechny hodnoty vlastnosti musí pocházet z dané třídy.), **someValuesFrom** (Alespoň jedna hodnota vlastnosti musí pocházet z dané třídy.)
- **owl:cardinality** - (kardinalita viz předchozí přednáška)  
(OWL DL: owl:maxCardinality, owl:minCardinality)
- **hasValue** (nelze u OWL Lite) - Omezení umožňuje specifikovat, že daná třída má konkrétní hodnotu vlastnosti (existence třídy je dána konkrétní hodnotou vlastnosti), např. bílé víno.

# Popisy tříd - intersectionOf



```
<owl:Class rdf:ID="WhiteWine">  
  <owl:intersectionOf rdf:parseType="Collection">  
    <owl:Class rdf:about="#Wine" />  
    <owl:Restriction>  
      <owl:onProperty rdf:resource="#hasColor" />  
      <owl:hasValue rdf:resource="#White" />  
    </owl:Restriction>  
  </owl:intersectionOf>  
</owl:Class>
```

- Operátor pro tvorbu tzv. komplexních tříd (spolu s `unionOf`, `complementOf`). Slouží pro specifikaci členů třídy.
- logický součin (AND), průnik dvou nebo více popisů tříd
- Bílé víno = víno AND bílá barva
- Konstrukce tvrdí, že členem a současně individuem třídy je to, co je víno a současně bílé.

# Popisy tříd - unionOf



```
<owl:Class rdf:ID="WineDescriptor">  
  <rdfs:comment>Made WineDescriptor unionType of tastes and color</rdfs:comment>  
  <owl:unionOf rdf:parseType="Collection">  
    <owl:Class rdf:about="#WineTaste" />  
    <owl:Class rdf:about="#WineColor" />  
  </owl:unionOf>  
</owl:Class>
```

- Operátor pro tvorbu tzv. komplexních tříd.
- logický součet (OR), sjednocení (union) dvou nebo více popisů tříd
- WineDescriptor = WineTaste OR WineColor
- Třída WineDescriptor zahrnuje členy obou tříd - členy třídy WineTaste a WineColor

# Popisy tříd - complementOf



```
<owl:Class rdf:ID="NonConsumableThing">  
  <owl:complementOf rdf:resource="#ConsumableThing" />  
</owl:Class>
```

- Operátor pro tvorbu tzv. komplexních tříd.
- logická negace (NOT)
- Vymezení, že individua dané třídy nepatří k jiné třídě.

# Popisy tříd -konstrukt oneOf



```
<owl:Class rdf:ID="WineColor">  
  <owl:oneOf rdf:parseType="Collection">  
    <owl:Thing rdf:about="#White" />  
    <owl:Thing rdf:about="#Rose" />  
    <owl:Thing rdf:about="#Red" />  
  </owl:oneOf>  
</owl:Class>
```

Specifikace třídy prostřednictvím přímého vyjmenování všech možných individuí.

```
<owl:Class rdf:ID="WineColor">  
  <owl:oneOf rdf:parseType="Collection">  
    <WineColor rdf:about="#White" />  
    <WineColor rdf:about="#Rose" />  
    <WineColor rdf:about="#Red" />  
  </owl:oneOf>  
</owl:Class>
```



- **Konstrukt pro tvorbu taxonomie - `rdfs:subClassOf`**

```
<owl:Class rdf:ID="Wine">
```

```
<rdfs:subClassOf rdf:resource="&food;PotableLiquid" />
```

```
---
```

```
</owl:Class>
```

- **Disjunktivní třídy - `owl:disjointWith` - Uvádí disjunktivní třídy - nemohou mít stejné individuum.**

```
<owl:Class rdf:ID="LateHarvest">
```

```
<rdfs:subClassOf rdf:resource="#Wine" />
```

```
<owl:disjointWith rdf:resource="#EarlyHarvest" />
```

```
</owl:Class>
```



- `owl:equivalentClass` - Spojuje popis třídy s popisem jiné třídy s tím, že se vztahují na stejná individua.

```
<owl:Class rdf:ID="Wine">
```

```
<owl:equivalentClass rdf:resource="&vin;Wine"/>
```

```
</owl:Class>
```

- pojmenování třídy - `rdfs:label`, atribut `lang` - cizojazyčné varianty

```
<owl:Class rdf:ID="Wine">
```

```
<rdfs:label xml:lang="en">wine</rdfs:label>
```

```
<rdfs:label xml:lang="fr">vin</rdfs:label>
```

```
</owl:Class>
```

# Zápis individuí



```
<WineGrape rdf:ID="CabernetSauvignonGrape" />
```

- Uvedení individua deklarací členství ve třídě.  
nebo

```
<owl:Thing rdf:ID="CabernetSauvignonGrape" />
```

```
<owl:Thing rdf:about="#CabernetSauvignonGrape">
```

```
  <rdf:type rdf:resource="#WineGrape"/>
```

```
</owl:Thing>
```

rdf : type - Vlastnost RDF, která váže individuum ke třídě, jejímž je členem.



# Definování vlastností



Vlastnost - binární vztah.

Dva typy vlastností

- **Objektová vlastnost** - vytváří spojení mezi objekty tříd. Vztahy mezi individui dvou tříd.  
`ObjectProperty`
- **Datotypová vlastnost** - spojuje jedince s hodnotou, která je určitého datového typu (RDF literals and XML Schema datatypes) . `DatatypeProperty`

# Definování objektové vlastnosti



```
<owl:ObjectProperty rdf:ID="hasColor">  
  <rdf:type rdf:resource="&owl;FunctionalProperty" />  
  <rdfs:subPropertyOf rdf:resource="#hasWineDescriptor" />  
  <rdfs:domain rdf:resource="#Wine" />  
  <rdfs:range rdf:resource="#WineColor" />  
</owl:ObjectProperty>
```

Wine       $\longrightarrow$     **hasColor**       $\longrightarrow$     WineColor (white, red...)  
                                    **hasSugar**                                    WineSugar (dry, sweet...)

na úrovni individua

**Rulandské bílé vinařství Krist má barvu bílou, je suché.**

# Definování datotypové vlastnosti



```
<owl:DatatypeProperty rdf:ID="yearValue">  
  <rdfs:domain rdf:resource="#VintageYear" />  
  <rdfs:range rdf:resource="&xsd;positiveInteger" />  
</owl:DatatypeProperty>
```

- Nejčastějším zdrojem datotypové vlastnosti je XML Schema datatypes.

# Vlastnosti individuí



```
<Sauvignon rdf:ID="Savignon1998">  
  <locatedIn rdf:resource="#ArroyoGrandeRegion" />  
  <hasVintageYear rdf:resource="#Year1998" />  
  <hasMaker rdf:resource="#SaucelitoCanyon" />  
  <hasSugar rdf:resource="#Dry" />  
  <hasFlavor rdf:resource="#Moderate" />  
  <hasBody rdf:resource="#Medium" />  
</Sauvignon>  
  
<VintageYear rdf:ID="Year1998">  
  <yearValue rdf:datatype="&xsd;positiveInteger">1998</yearValue>  
</VintageYear>
```

# Odvozování a deskripční logika

- OWL DL je jazyk založený na deskripční logice (DL).
- DL je teoretickým podkladem pro algoritmizované odvozování v oblasti ontologií.
- DL je logika pojmů/konceptů (množina individuí). Zachycuje vztahy mezi pojmy.
- Tvorba složených konceptů:
  - Muž  $\sqcap$  Rybář (průnik)
  - Muž  $\sqcup$  Rybář (sjednocení)
- Role - binární relace
  - $\exists$ obíháKolem.Hvězda (koncept planety (pro každou planetu existuje hvězda, kolem které obíhá)),
  - $\forall$ jeOtcem.Muž (Koncept zahrnující všechna individua, jejichž všechny děti jsou mužského pohlaví.)

# Odvozování a deskripční logika



Základní odvozovací úlohy v deskripční logice jsou následující:

- **Kontrola konzistence** znalostní báze zjišťuje, zda definice každého konceptu připouští náležení alespoň jednoho individua.
- **Kontrola individuí** zjišťuje, zda dané individuum spadá pod daný koncept.
- **Realizace** nalezne nejspecifičtější koncept, pod který individuum spadá.
- **Vyhledání** nalezne všechna individua spadající pod zadaný koncept.
- **Kontrola subsumpce** zjišťuje, zdali je jeden zadaný koncept podkonceptem druhého. Více viz Svátek, 2007

# Tezaurus vers. ontologie



## ● podobnost

- terminologické pokrytí určité domény, vymezení vztahů mezi jednotlivými termíny, seskupení termínů do kategorií a podkategorií
- obojí se používá pro popis a organizaci informací

## ● rozdílnost

- ontologie obsahuje mnohem více vztahů, které jsou formálně definovány a jsou jednoznačné, interpretovatelné strojově i lidmi
- tezaurus – vztahy mezi termíny, ontologie – definování pojmu a vztahu mezi nimi, logický a formální popis



Evropa

**eurovoc**  
THE SAURUS

cestina(cs)

[home](#)[navigation](#)[search](#)[maintenance](#)[download](#)[multilingual list](#)[translations](#)[info](#)[contact us](#)[helpdesk](#)[help](#)© European  
Communities

## víno

MT 6021 nápoje a cukr

BT1 alkoholický nápoj

BT2 nápoje

NT1 alkoholizované víno

NT1 aromatizované víno

NT1 bílé víno

NT1 červené víno

NT1 lahvé víno

NT1 místní víno

NT1 nešumivé víno

NT1 růžové víno

NT1 sekt

NT1 stolní víno

NT1 šumivé víno

NT1 výběrové víno

RT chaptalování (6036)

hroznové víno (6006)

vinné kvašení (6036)

vinohradnictví (5631)



# Ukázka definice pojmu v ontologii

The screenshot shows the Protégé 3.4 interface with the 'CLASS EDITOR' for the class 'Vína'. The 'CLASS BROWSER' on the left shows a class hierarchy where 'Vína' is selected. The 'CLASS EDITOR' displays the following information:

- For Class:** Vína (instance of :STANDARD-CLASS)
- Name:** Vína
- Role:** Concrete
- Template Slots:**

Name	Cardinality	Type	Other Facets
barva	single	Symbol	allowed-values={bílá,červená}
jakost	single	Symbol	allowed-values={stolní,jakostní-odrůdové,kabine...}
jídlo	multiple	Class with superclass Jí...	
name	single	String	
obsah cukru	single	Symbol	allowed-values={suché, polosuché, sladké}
plnost	single	Symbol	allowed-values={plné, střední, lehké}
synonyma	multiple	String	
výrobce	multiple	Instance of Vinaři	inverse-slot=produkce
vůně	multiple	Symbol	allowed-values={jemná, plná, diskrétní, delikátní, šv...}

At the bottom, the 'Superclasses' section shows ':THING'.

# Co byste měli znát



- strukturu dokumentu obsahující ontologii (OWL Web Ontology Language Guide, 2004a, kap. 2 The Structure of Ontologies)
- význam následujících prvků:

`owl:Class`, `rdfs:subClassOf`, `owl:Restriction`,  
`owl:hasValue`, `owl:intersectionOf`, `owl:unionOf`,  
`owl:oneOf`, `complementOf`, `rdfs:label`, `ObjectProperty`,  
`DatatypeProperty`, `rdfs:subPropertyOf`, `rdfs:domain`,  
`rdfs:range`

- zápis
  - individuí
  - hodnot vlastností individuí
- Lze využít zdroje z doporučené literatury: OWL Web Ontology Language Guide, 2004a, Graphical Representations of RDF/RDFS/OWL entity types, 2010

# Použitá literatura



- Arano, S. 2005. *Thesauruses and ontologies* [online]. *Hipertext.net*. 2005, č. 3, [cit. 2009-02-20]. Dostupné z WWW: <http://www.hipertext.net/english/pag1009.htm> ISSN 1695-5498
- *Graphical Representations of RDF/RDFS/OWL entity types* [online]. 2010. [cit. 2010-04-27]. Dostupné z WWW: <http://www.infowebml.ws/website/graphical-representations.htm#rdfsClass>
- *Lékárnička Znalostní technologie I. Doplnění předmětu Znalostní technologie I. vyučovaného na Fakultě informatiky a managementu UHK.* [online]. Datum poslední aktualizace: 27. 10. 2009 [cit. 2010-04-27]. Dostupné z WWW: [http://lide.uhk.cz/fim/ucitel/fshusam2/lekarnicky/zt1/zt1\\_index.html](http://lide.uhk.cz/fim/ucitel/fshusam2/lekarnicky/zt1/zt1_index.html)
- *OWL Web Ontology Language Guide* [online]. 2004a. W3C Recommendation 10 February 2004, Copyright © 2004 W3C [cit. 2010-04-27]. Dostupné z WWW: <http://www.w3.org/TR/owl-guide/>

# Použitá literatura



- *OWL Web Ontology Language : Use Cases and Requirements* [online]. 2004b. W3C Recommendation 10 February 2004, Copyright © 2004 W3C [cit. 2010-04-27]. Dostupné z WWW: <http://www.w3.org/TR/2004/REC-webont-req-20040210/>
- Sklenák, V. *Sémantický web* [online]. In *Sborník konference Inforum*. © 1995-2010 AiP [cit. 2010-04-27]. Dostupné z WWW: [http://www.inforum.cz/archiv/inforum2003/prispevky/Sklenak\\_Vilem.pdf](http://www.inforum.cz/archiv/inforum2003/prispevky/Sklenak_Vilem.pdf)
- Svátek, V., Vacura, M. 2007a. *Ontologické inženýrství* [online]. In *DATAKON 2007*, Brno, 20.-23. 10. 2007, 32 s. [cit. 2009-02-20]. Dostupné z WWW: <http://nb.vse.cz/~svatek/dkono7final.pdf>
- Svátek, V. 2002. *Ontologie a WWW* [online]. In *DATAKON 2002*, Brno, 19.-22. 10. 2002, 35 s. Dostupné z WWW: <http://nb.vse.cz/~svatek/onto-www.pdf>