

# Aplikace s odvozováním nad ontologiemi

Doc. Ing. Vojtěch Svátek, Dr.

*Zimní semestr 2012*

<http://nb.vse.cz/~svatek/rzzw.html>

# Přehled

- Odvozování v medicíně
- Odvozování ve stavebnictví
- Odvozování v Linked Data

# Ontologie v medicíně

- Jako *terminologicky orientované* doménové ontologie lze chápat většinu z tradičních nomenklatur (ICD, SNOMED, MeSH...)
  - Např. <http://www.who.int/classifications/icd/en/>
- Více k pojetí *znalostní* ontologie směřuje sémantická síť UMLS nebo HL7 RIM
- Žádný z tradičních zdrojů se však neopíral o formálně-logickou sémantiku; pouze volné zmapování možných vztahů mezi entitami
- Široký okruh biomedicínských ontologií je přístupný na <http://bioportal.bioontology.org>

# Deskripční logika v medicíně

- V 90. letech první rozsáhlá iniciativa k zapojení formální logiky do terminologických systémů – projekt GALEN
  - Využití formálního jazyka GRAIL založeného na deskripční logice
  - Systematické definování medicínských pojmů logickými prostředky
  - Možnost odvozování – testování hierarchické podřazenosti, konzistence terminologické báze atd.
  - Hlavní slogan: „Making the impossible very difficult“
  - Šířen v rámci neziskové asociace OpenGalen
- Nástupcem GALENu je SNOMED-OWL

# SNOMED - OWL

- Projekt převodu SNOMED-CT do formátu OWL je „vlajkovou lodí“ ontologické interoperability v medicíně
  - Např. specializovaný workshop KR-MED 2008, „Representing and Sharing Knowledge Using SNOMED“ (květen 2008, Arizona) se z velké části věnoval aplikaci deskripční logiky na SNOMED
  - Analogické k projektu GALEN, ale využívá standardizovaný jazyk – možnost integrace s jinými ontologiemi a využití v obecných aplikacích

# SNOMED - OWL

- První verze převodu do OWL v r. 2007
- Cca 400 tisíc konceptů
- Definice konceptů využívá zvl. existenční omezení
  - `Appendicectomy equivalentTo Surgical_Procedure and (method some Excision) and (procedure-site some Appendix_structure)`
- Systém není uzavřený - je možno libovolně vytvářet nové koncepty, aniž by musely obdržet vyčleněné jméno a ID (post-koordinace)
  - `Excision and (procedure-site some (kidney and (laterality some left)))`
- To vede k nutnosti využití odvozovacích systémů

# Scénář využití SNOMED-OWL

- (Dle J. Daviese, British Telecom + projekt RIDE)
- Lékař vloží klinická data do zdravotnické aplikace a přiřadí je konceptu, který pro tento účel *nově vytvoří*, např. „alergie způsobená mandlemi“
- Aplikace informuje o novém konceptu *terminologickou službu*
- Terminologická služba kontaktuje *odvozovací nástroj* (nad DL), a ten automaticky zařadí nový koncept do *taxonomie* existujících
- Terminologická služba informuje zdravotnickou aplikaci o změně taxonomie, ve které je nyní nový koncept podřazený konceptu „alergie na ořechy“

# Nástroj Snow Owl

- Prohlížení (volná verze)
- Odvozování v DL (komerční verze)
  - <http://www.b2international.com/portal/snow-owl>



# Aktuální výzkum

- Tvorba rozsáhlých ontologií pomocí skriptů
- Extrakce modulů s ohledem na danou signaturu
  - Nezbytnost pro rozumné využívání velkých ontologií
  - Signatura: množina tříd a individuí
  - Modul: množina axiomů, které zachycují vše, co se týká dané signatury

# Stavebnictví

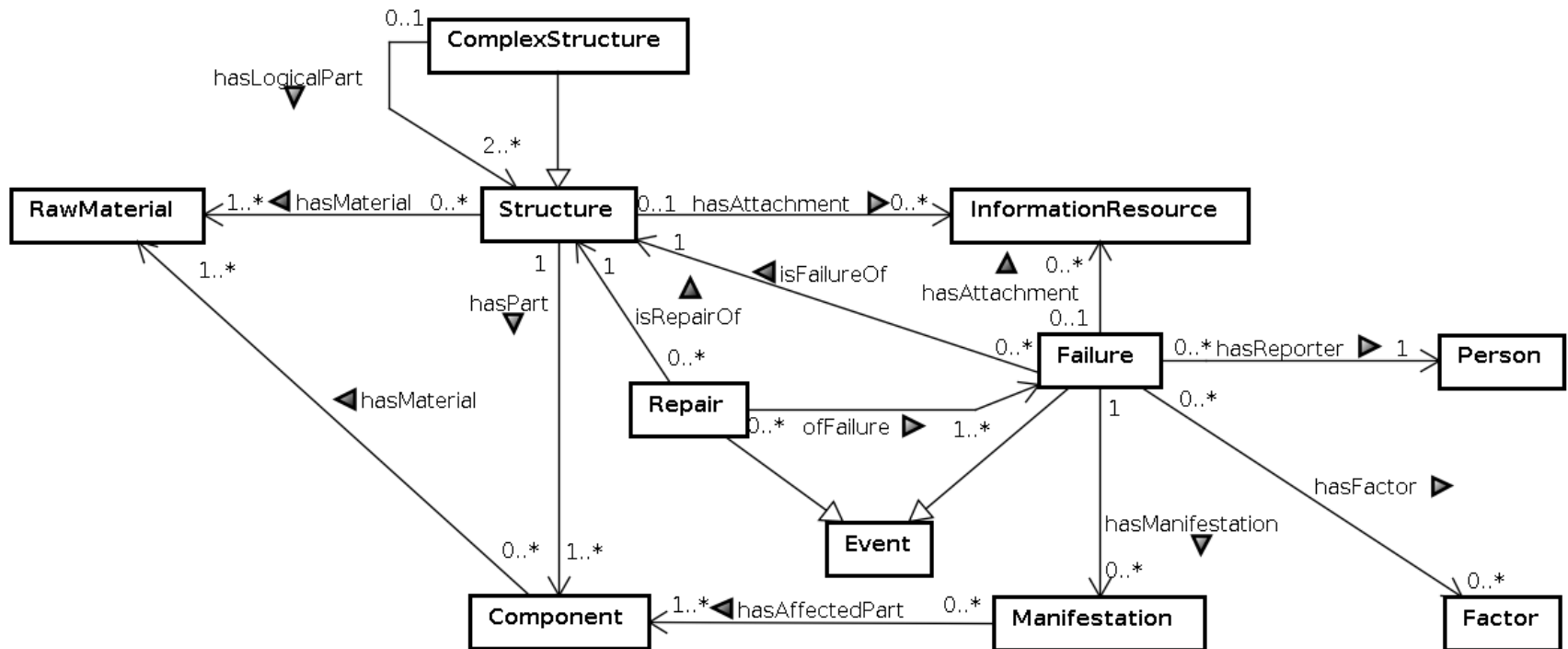
- Projekt StruFail – znalostní báze strukturálních poruch budov
  - Spolupráce ÚTAM AV ČR a KK FEL ČVUT
  - <http://krizik.felk.cvut.cz:8080/icefailures/>
  - Znalostní báze je postavena na dynamicky aktualizované ontologii (OWL)
  - Využívá architekturu JOPA, která kombinuje dva druhy odvozování
    - OWA – testování konzistence, dotazování do báze
    - CWA – validace dat (mj. kvůli stabilitě UI)

# JOPA

- Java OWL Persistence API
  - <http://krizik.felk.cvut.cz/km/jopa>
- Řeší problém aplikace běžící nad dynamicky upravovanou ontologií
- „Kontrakt“ mezi aplikací a ontologií (věcná logika aplikace) je reprezentovaný pomocí integritních omezení (IC) vyjádřených v dotazovacím jazyce SPARQL-DL<sup>NOT</sup>
- Nad IC se odvozuje pomocí CWA

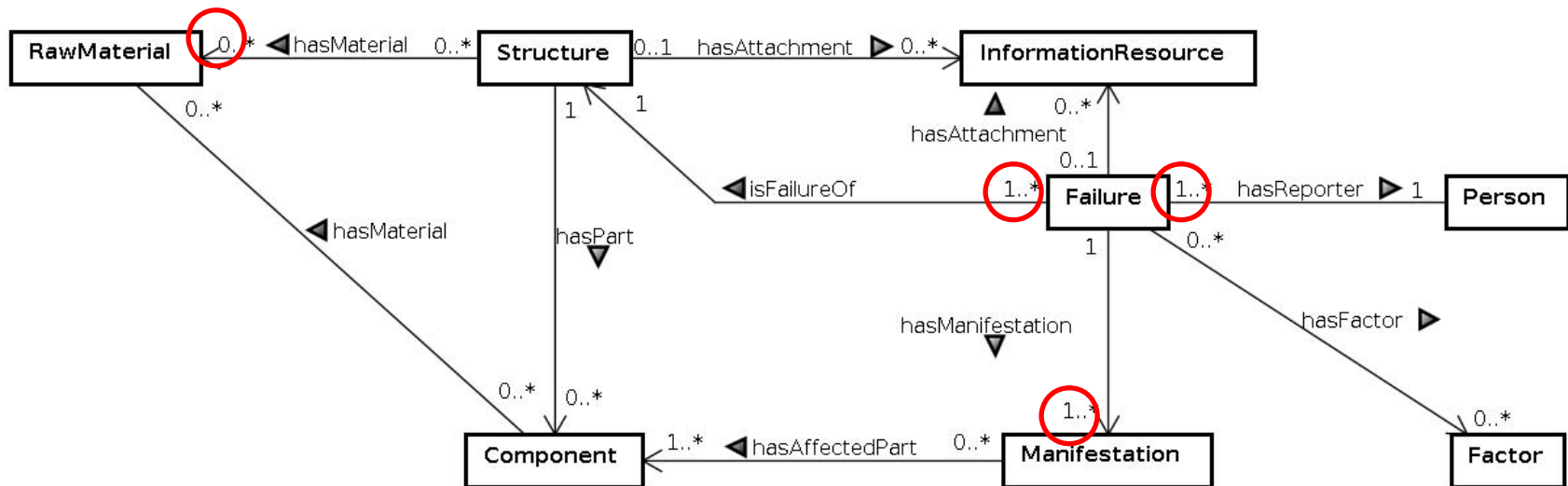
# Jádro ontologie

Co platí  
ve světě



# Struktura integritních omezení

Co musí platit  
v datech



# OWL2Query (v Protégé)

The screenshot displays the OWL2Query interface within Protégé. The main window shows a SPARQL query: `Q(y, x) :- SPO(?y, <http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#memberOf>), PV(?y, ?x, <http://www.Department0.University0.edu>), T(<http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#Person>, ?x).`

The graphical representation shows a variable `y` (red box) connected to `pre1:memberOf` (yellow box), which is linked to `pre1:Person` (yellow box) via a `<<SubPropertyOf>>` relationship. Another variable `x` (red box) is connected to `http://www.Department0.University0.edu` (yellow box) via a `<<TypeOf>>` relationship.

The results table is as follows:

| ?y            | ?x                          |
|---------------|-----------------------------|
| pre1:memberOf | pre0:UndergraduateStudent88 |
| pre1:memberOf | pre0:GraduateStudent100     |
| pre1:memberOf | pre0:GraduateStudent103     |
| pre1:memberOf | pre0:GraduateStudent102     |
| pre1:memberOf | pre0:GraduateStudent105     |

On the right, the Class hierarchy shows `Thing` with subclasses `Organization`, `Person`, and `Publication`. The `Publication` class has subclasses `Article`, `Book`, `Manual`, `Software`, `Specification`, and `UnofficialPublica`. The Individuals list shows `GraduateCourse8`, `GraduateCourse9`, `GraduateStudent0`, `GraduateStudent1`, `GraduateStudent10`, `GraduateStudent100`, `GraduateStudent101`, `GraduateStudent102`, and `GraduateStudent103`.

At the bottom, the Reasoner is active and Show Inferences is checked.

<http://krizik.felk.cvut.cz/km/owl2query>

# Odvozování a Linked Data

- Inverzně funkční vlastnosti – zjišťování identity
- Podvlastnosti, doplňování inverzních vlastností
- Často lze bez výpočtu deduktivního uzávěru teorie