Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Diplomová práce

Datový model EEG/ERP portálu v prostředcích sémantického webu

Plzeň 2013 Filip Markvart

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 7. února 2013

Filip Markvart

Abstract

This thesis DODELAT

Obsah

1 Úvod

EEG/ERP portál je webová aplikace sloužící výzkumným pracovníkům ke shromažďování a organizaci dat získaných při neuroinformatických experimentech v EEG laboratoři. Jejím cílem je ukládání naměřených dat v kontextu prováděného experimentu, který lze popsat rozsáhlou množinou různorodých údajů. Tato aplikace již prošla mnohaletým vývojem v jehož průběhu postupně docházelo ke změnám datového modelu kvůli přibývajícím požadavkům na uchovávaná data. Relační databáze jež slouží jako persistentní úložiště tak postupně byla rozšiřována o další tabulky, jejichž počet se k datu tvorby této práce pohybuje v řádu desítek. Většina realizací požadavků na ukládání dalších dat tak přímo znamená zásah nejen do databáze portálu ale také to do datové vrstvy, která ji využívá. V současné době tak databáze obsahuje velké množství tabulek uchovávající různá data, která jsou ale ve smyslu sémantiky často příbuzná a existuje mezi nimi vazba, která je prostřednictvím relačního datového modelu velmi obtížně popsatelná. Zároveň lze očekávat, že budou přibývat požadavky na uchování dalších dat, která navíc nemusejí mít jen homogenní strukturu (ve smyslu relační databáze), ale může se jednat i o množiny sémanticky příbuzných údajů – tzv. metadata, které budou vázány pouze k některým datům. Možnost ukládání strukturně heterogenních, ale sémantický příbuzných metadat je tak dalším otevřeným problémem.

Cílem této práce je prozkoumání struktury a nalezení sémantiky dat v současném datovém modelu relační databáze portálu a následná úprava tohoto modelu do podoby, která by dovolovala uchovat jak sémantiku dat, kterou není možné relačním modelem vyjádřit tak dodávat dynamicky datům přídavná metadata, aniž by muselo docházet k větším zásahům do datového modelu portálu. Pro realizaci úpravy datového modelu budou v této práci využity prostředky tzv. sémantického webu, který poskytuje množství standardů a technologií pro uchovávání organizaci a správu dat. Tyto technologie a nástroje zde budou popsány a na základě jejich analýzy budou vybrány prostředky, které se využijí pro implementaci úpravy zmiňovaného datového modelu. Poslední část práce se věnuje testování modifikovaného modelu a to především z výkonnostního hlediska. Díky této části by mělo být možné posoudit jak užitečnost samotné úpravy tak i použitelnost a efektivnost získaného modelu pro potřeby EEG/ERP portálu.

2 Sémantický web

Dnešní podoba webu, tak jak je všeobecně známa, je tvořena značným množstvím informací, které mají řadu autorů, v podobě různých organizací či jednotlivců, jež se liší jak svým obsahem tak i podobou publikace. Tyto informace jsou poměrně snadno přístupné díky jejich jednoznačné identifikaci prostřednictvím URI identifikátoru (za předpokladu, že jej známe). K usnadnění získávání dalších (často příbuzných) informací napomáhají tzv. hypertextové odkazy, jež usnadňují přístup k dalším zdrojům informací odstraněním požadavku na uživatelovu znalost identifikátoru cílového zdroje. Samotné hypertextové odkazy tak sice zajišťují provázání jednoho informačního zdroje s jiným díky znalosti jeho URI identifikátoru, ale nenesou už žádné další informace, které by například uživateli poskytly další údaje o cílovém zdroji. Takováto podoba umožňuje získávání informací jak koncovým uživatelům webu, tak v omezené podobě i vyhledávacím strojům, ale má své limity, neboť se v nepřeberném množství dat lze snadno ztratit, či se jen dostat k irelevantním informacím [7]. Základním úkolem sémantického webu, jehož první myšlenky prezentoval v roce 2001 zakladatel konsorcia W3C Tim Berners-Lee, je umožnit aby informace dostupné prostřednictvím webu byly srozumitelné nejen uživatelům, ale také počítačům, jež tato data zpracovávají [3]. Hlavním cílem je tedy vývoj standardů a technologií, které by umožňovaly přesnější a podrobnější vyhledávání, integraci dat a také automatizaci častých úkonů. Sémantický web je založen na několika principech, které budou níže uvedeny.

- **Jednoznačná identifikace entit prostřednictvím URI**

  Veškerá data, reprezentující obvykle objekty reálného světa publikovaná prostřednictvím webu je možné jednoznačně odkazovat prostřednictvím identifikátoru URI. Díky této skutečnosti je tak možné realizovat i nepřímé odkazy na objekty, například osobu Petr Novák s emailem petr.novak@w3.org je možné identifikovat jako osobou, jejíž email má URI mailto:petr.novak@w3.org.

- **Zdroje i odkazy mezi nimi je možné typovat**

  Současná podoba webu je tvořena zdroji a odkazy jež je vzájemně propojují. Zdroje, které jsou reprezentovány webovými dokumenty jsou publikovány za účelem poskytnutí informací lidskému uživateli, který dokáže ze samotného obsahu dokumentu získat i některá jeho metadata

(pokud jsou v určité formě součástí obsahu) a do jisté míry také vztah k ostatním dokumentům, na něž vedou případné odkazy. Stroje v podobě různých vyhledávačů či automatů pro shromažďování dat ale tuto schopnost nemají nebo je pro ně příliš náročná. Řešením sémantického webu je typování jak samotných zdrojů, tak i odkazů, které je provazují. Díky této skutečnosti je pak možné webovým dokumentům dodávat metadata jako např. autora, verzi či závislost na jiném dokumentu. Z hlediska typování odkazů je například možné jeden webový zdroj označit pouze jako odlišnou verzi jiného zdroje.

- **Tolerance neúplných informací**

  U současně podoby webu může nastat situace, kdy některý zdroj není dostupný. V takovém případě uživatel ztrácí přistup k danému dokumentu, ale díky koncepci webu není nikterak ohrožena dostupnost ostatních zdrojů. V případě sémantického webu se situace nemění, nedostupnost některého zdroje není žádnou překážkou, neboť nástroje sémantického webu zpracovávají pouze ty informace, které jsou dostupné a těch vytvářejí závěry. V důsledku je tak možné dojít při zpracovávání dat ke stejným výsledkům, jako v případě, když jsou zpracovávány jen některé vybrané informace, jejichž rozsah je explicitně definován.

- **Zpracování neověřených dat**

  Při zpracování informací pocházejících z neověřených zdrojů je možné dohledávat prostřednictvím typovaných odkazů důvěryhodná data, jejichž obsah a odkazy poslouží jako ověřovací prostředek. Tento princip je možné uvést na jednoduchém příkladu. Aplikace zpracovávající data sémantického webu vyhledá informace, přičemž je kladen požadavek na vysokou pravděpodobnost správnosti výsledku. Pokud část nalezených informací pochází z neověřeného zdroje, je možné vyhledávat například jejich autora v odkazech zdrojů, které jsou důvěryhodné. V případě úspěchu nalezení takového odkazu u více různých zdrojů je pak možné považovat zkoumaný zdroj s vysokou pravděpodobností rovněž za důvěryhodný a tím zajistit plnění požadavků na výsledek.

- **paralelního vývoje dat**

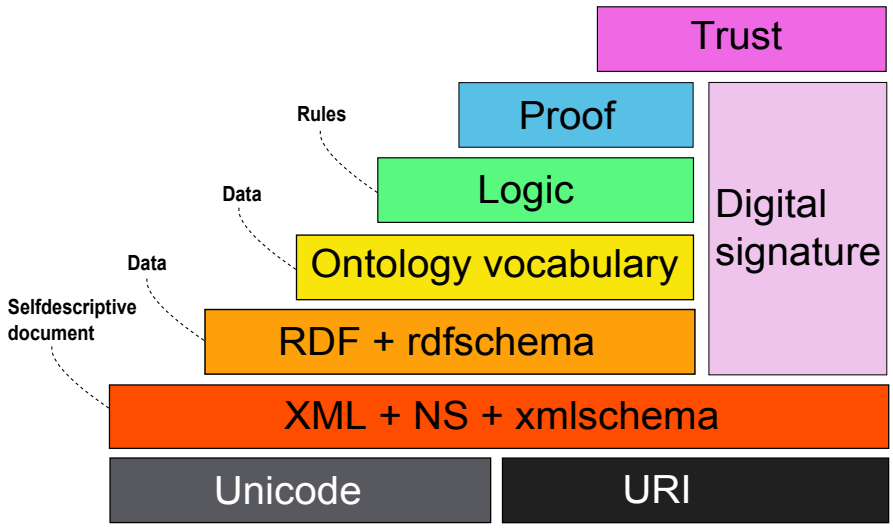
  V průběhu času nezřídka nastávají situace, kdy autoři, či skupiny autorů publikují obdobná data na různých místech nebo v odlišném čase. Obsah těchto dokumentů se může navíc lišit svým jazykem či použitou terminologií, ač význam bude shodný. S využitím prostředků sémantického webu je ale možné prostřednictvím typovaných odkazů zajistit provázanost významově obdobných či na sebe navazujících dat

i přes překážku rozdílnosti jejich podoby zápisu. Navíc je také možné dodávat nové informace bez nutnosti úpravy původních dat, která tím pádem nezmění svoji strukturu [7].

2.1 Architektura sémantického webu

Architektura sémantického webu sestává z více oddělených vrstev, mezi nimiž je zajištěna zpětná i dopředná kompatibilita [2]. Nejnižší vrstva je tvořena dvěma technologickými standardy – URI identifikátory sloužící pro jednoznačné pojmenování zdrojů dat a Unicode kódování mezinárodní znakovou sadou. Druhou vrstvu architektury, jež je patrná z obrázku 2.1, reprezentuje značkovací jazyk XML (Extensible Markup Language), který umožňuje tvorbu strukturovaného dokumentu za užití vlastních značek. Tato vrstva zároveň zajišťuje definici XML schématu včetně jmenných prostorů. RDF + rdfschema jež následuje je klíčovou vrstvou sémantického webu neboť dovoluje tvorbu vazeb a vztahů mezi jednotlivými zdroji, které jsou typované spolu s odkazy. Je tak možné definovat libovolné vztahy mezi objekty či jejich kategoriemi bez nutnosti specifikace významu samotných vazeb či objektů. Díky RDF schématu je vytvářena základní sémantika datového modelu, která už definuje význam některých elementů jako třídy či podtřídy. Vrstva ontologického slovníku, zastoupená jazykem OWL, nabízí pokročilou reprezentaci znalostí na úrovni deskripční logiky a umožňuje tak vytvářet složitější struktury sloužící k popisu různých vlastností objektů [3]. Poslední vrstvou, která je jako všechny předchozí zmíněné konsorciem W3C standardizovaná jsou digitální podpisy. Ty poskytují možnosti například pro detekci různých verzí dokumentů. Zbylé výše znázorněné vrstvy slouží pro definice a vyhodnocování odvozovacích pravidel a v současnosti jsou ve fázi vývoje [7].

Vrstvení jazyků sémantického webu je podstatné pro úroveň expresivity znalostního modelu, neboť s rostoucí vyjadřovací možností jazyka také roste složitost dotazovacích operací nad modelem. Je tedy nutné před započetím tvorby modelu nejprve zjistit jeho požadovanou expresivitu a podle té zvolit pro zápis dat jazyk, který ji dovoluje obsáhnout. Využívá se tedy skutečnosti, že jazyk vyšší vrstvy zahrnuje vyjadřovací schopnosti vrstev nižších[3]. Další podkapitoly se budou zabývat podrobněji jednotlivý zmíněnými technologiemi.



Obrázek 2.1: Architektura sémantickém webu [7]

2.2 XML

XML (eXtensible Markup Language) je značkovací jazyk sloužící pro popis hierarchických struktur textových dokumentů prostřednictvím tzv. tagů. Tag je konstrukce, která slouží k počátečnímu a koncovému ohraničení společně definovaného elementu. Tag lze chápat jako prostředek pro dodání metadat ke textové struktuře, jež ohraničuje. Příkladem může být následující zápis *<prijmeni>Novák</prijmeni>*, kde elementu *Novák* je dodána meta informace, že se jedná o příjmení. Samotné XML ale nedefinuje žádný sémantický význam tagů, slouží pouze pro specifikaci syntaxe na úrovní XML dokumentu. Pro definici (zejména hierarchické) struktury XML dokumentu slouží XML Schema, které umožňuje zápis pravidel, jež musí cílový dokument dodržovat pro zachování své validity. Ze strany sémantického webu ale nemají pravidla XML Schema žádný sémantický význam a slouží tak pouze pro definici struktury a syntaxe. Zmíněné schéma také definuje základní datové typy (čísla, řetězce, čas a pod.), které nabývají významu v sémantických jazycích jako je RDF [2].

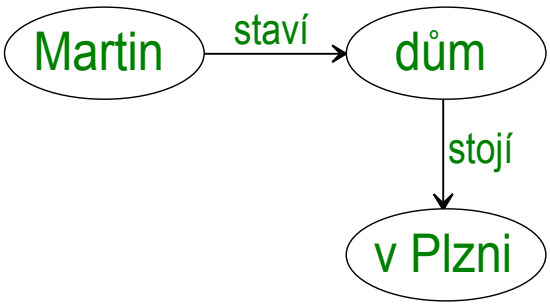
2.3 RDF

Technologický základ sémantického webu tvoří jazyk RDF (Resource Description Framework), jež slouží jako obecný rámec pro popis, výměnu a opětovné použití metadat [2]. Tento rámec poskytuje jednoduchý model sloužící pro popis zdrojů jež je nezávislý na jeho konkrétní implementaci [1]. Samotné informace o objektu jsou realizovány prostřednictvím tvrzení, jež se označují jako trojice (anglicky triple). Každou trojici tvoří spolu subjekt, predikát a objekt. Subjekt je libovolný objekt identifikovatelný pomocí URI, který se snažíme prostřednictvím trojice popisovat, zaznamenat nějakou jeho vlastnost. Tato vlastnost se popisuje prostřednictvím predikátu, který vede ve směru od subjektu ke objektu, přiřazuje tedy subjektu nějaký objekt prostřednictvím této vlastnosti. Cílový objekt pak představuje hodnotu, které předchozí objekt nabývá pro daný predikát. Tento princip je možné znázornit na jednoduchém tvrzení, zapsaném větou „Martin staví dům.“ Subjektem trojice potom bude Martin, predikátem staví a objektem dům, tak jak je znázorněno na obrázku 2.2.

Obrázek 2.2: Příklad RDF trojice

Dle definovaného standardu [5] je možné, aby objektem byl jiný subjekt. Díky této skutečnosti je možné jednotlivé trojice spojovat do většího celku, který ve výsledku tvoří strukturu orientovaného grafu, kterou lze označit jako model [2]. Jednoduchým model tvořený dvěma trojicemi lze vytvořit využitím dalšího tvrzení „Dům stojí v Plzni.“ V předchozí trojici pak bude dům subjektem namísto objektu a dojde tak ke spojení dvou trojic (za předpokladu že v obou tvrzeních je myšlen stejný fyzický dům), tak jak je znázorněno na obrázku 2.3.

Z hlediska implementace mohou být uzly orientovaného grafu datového modelu tvořeny URI identifikátorem, anonymním listem nebo literálem [4]. URI identifikátor obsahuje pouze adresu zdroje v textové podobě ve znakové sadě Unicode a zastupuje tak konkrétní jedinečný objekt. Anonymní list (anglicky blank node) je prvek nahrazující URI identifikátor při absenci unikátní adresy zdroje. Tato entita přestavuje zdroj, který je sice v rámci



Obrázek 2.3: Příklad jednoduchého RDF modelu

grafu popisován prostřednictvím trojic, ale není potřeba (často z hlediska významu), aby byl dostupný i vně grafu. S URI má společné to, že musí nést adresu zdroje (její syntaxe není implicitně definována), ale liší se skutečností, že tato adresa musí být unikátní pouze v rámci obalujícího modelu, nikoliv vně grafu. Může tak nastat situace, že dva odlišné modely budou obsahovat (ze syntaktického hlediska) dva stejné anonymní uzly, což v případě URI možné není (při respektování specifikace). Tento list tak může představovat anonymní objekt sloužící ke vytvoření vazby mezi jinými objekty, které jsou z hlediska sémantiky významné [5]. Jako příklad je možno uvést následující tvrzení. „Martinův přítel zná předpověď počasí. Počasí bude deštivé.„ Zjednodušeným převodem předchozích vět na trojice v podobě subjekt – predikát – objekt získáme: Martin – má – přítel, přítel – zná – počasí, počasí – bude – deštivé. Subjekt resp. objekt přítel zde může být reprezentován anonymním listem, v případě že jedinou signifikantní informací (z vnějšího pohledu na datový model) je Martinova získaná znalost počasí, nikoliv už jeho přítel, jež mu ji zprostředkoval. Poslední možnou reprezentací entity trojice je literál, jež nese Unicode textový řetězec, který slouží pro zápis koncové informace (užitečné pro lidského čtenáře). Literál může také navíc obsahovat položku, pro označení jazyka, ve kterém je textová informace zapsána [4].

Pro komponenty každé trojice grafu platí následující pravidla [4].

- **subjekt** - může být tvořen URI identifikátorem nebo anonymním listem
- **predikát** - může být tvořen pouze URI identifikátorem
- **objekt** - může být tvořen URI, anonymním listem nebo literálem

Pro zápis trojic RDF grafu je sice možné využít grafické podoby, ale pro vyjádření sémantiky webových zdrojů je nejvhodnější využít syntaxe jazyka XML. Níže uvedená ukázka kódu reprezentuje XML zápis trojice z obrázku 2.2.

```
<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 rdf:about="http://www.w3.org/Person/Martin">
    <dc:stavi>dům</dc:stavi>
  </rdf:Description>
</rdf:RDF>
```

Samotné RDF nedefinuje trojicím ani jejím částem sémantiku, ale dovoluje vyjádřit základní vztahy náležitosti prvků do kategorie prostřednictvím kontejnerů a kolekcí (např. rdf:Bag, rdf:List) [3]. Pro dodání základní sémantiky slouží schéma popsané v následující podkapitole.

2.4 RDFS

RDF Schema (Resource Description Framework Schema) funguje jako základní jazyk pro tvorbu ontologií s velmi jednoduchou sémantikou. Toto schéma rozšiřuje jazyk RDF o možnosti vyjádření vlastností objektů, konstrukce tříd objektů a popis jejich hierarchie [3]. Prostředky jazyka RDFS umožňují především vyjádření vztahů mezi zdroji a lze je rozdělit na dvě skupiny – třídy a vlastnosti.

2.4.1 Třídy

Skupiny zdrojů je možné rozčleňovat do skupin označovaných jako třídy (classes), jejichž členové se nazývají instance třídy. Na úrovni RDFS se rozlišují třídy od svých instancí a každá třída jich může mít neomezený počet. Dvě třídy mohou mít navíc shodnou množinu instancí tříd a zároveň tyto třídy mohou být navzájem různé. Bude-li se tedy například definovat třída A jako osoby pracující v kanceláři *1* a třída B jako osoby žijící ve městě X. Potom je možné aby různé třídy A a B měly stejné množiny instancí, za předpokladu,

že každá osoba pracující v kanceláři *1* bydlí ve městě *X*. Pro třídy platí také dědičnost – bude li třída *B* podtřídou *A*, pak všechny instance *B* jsou zároveň instancí třídy *A*. Koncept tříd RDFS definuje následující konstrukce [6]:

- **rdfs:Resource** představuje RDF zdroj, který je obalující třídou všech prvků – je tedy nejvýše postavenou rodičovskou třídou, *rdfs:Resource* je zároveň instancí *rdfs:Class*
- **rdfs:Class** reprezentuje rodičovskou třídu všech RDF tříd zdrojů, čímž je *rdfs:Class* instancí *rdfs:Class* (sebe sama)
- **rdfs:Literal** třída je instancí *rdfs:Class*, která slouží pro reprezentaci RDF literálů a je podtřídou *rdfs:Resource*
- **rdfs:Datatype** je obalující třídou pro datové typy, které jsou její instancí, *rdfs:Datatype* je zároveň podtřídou i instancí *rdfs:Class* a každá její instance je podtřídou *rdfs:Literal*
- **rdf:XMLLiteral** je třída XML literálů, která je podtřídou *rdfs:Literal* a instancí *rdfs:Datatype*
- **rdf:Property** představuje rodičovskou třídu všech definovaných vlastností a je instancí *rdfs:Class*

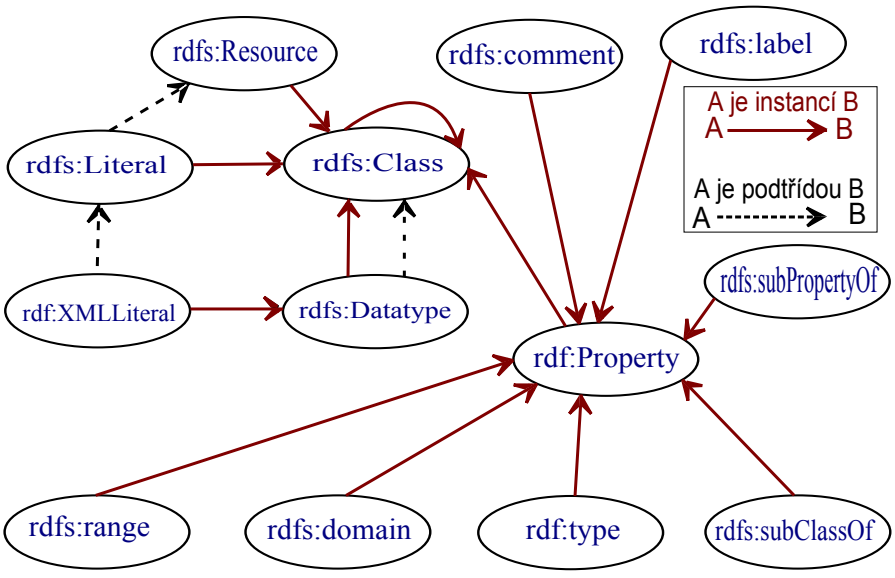
2.4.2 Vlastnosti

Vlastnosti (properties) slouží k vyjádření vztahu mezi dvěma zdroji – na úrovni trojice mezi subjektem a objektem. Koncept RDFS definuje následující vlastnosti:

- **rdfs:range** je instancí *rdf:Property*, která slouží ke vyjádření, že objekt jehož predikát má definovaný *rdfs:Range X* bude zároveň instancí *X*, například z následujících trojic *A – rdfs:Range B a X – A – C* bude vyplývat, že *C* je instancí *B*, zároveň platí, že objekt s definovaným predikátem může být instancí více tříd (pokud má predikát definováno více *rdfs:Range*).
- **rdfs:domain** představuje instanci *rdf:Property*, která vyjadřuje, že zdroj (subjekt) jehož predikát má definovaný *rdfs:Domain X* bude také instancí *X*, tento zdroj může být jako v předchozím případě instancí více tříd při definování více *rdfs:Domain*

- **rdf:type** slouží k definování, že zdroj (subjekt) je instancí třídy definované objektem, pokud tedy platí *A rdf:type B*, pak *A* je instancí *B*
- **rdfs:subClassOf** je vlastnost sloužící k vyjádření náležitosti instancí jedné třídy jako instancí jiné třídy, pokud platí *A rdfs:subClassOf B*, pak *A* je podtřídou *B* a všechny instance třídy *A* jsou zároveň instancí třídy *B*, tato vlastnost je navíc tranzitivní, takže popsanou dědičnost je možné řetězit do libovolné délky
- **rdfs:subPropertyOf** slouží k vyjádření dědičnosti vlastností, pokud platí *P1 subproperty P2*, pak pro trojici *A P1 B* platí, že subjekt *A* má pro vlastnost *P1* i *P2* pro objekt *B*
- **rdfs:label** umožňuje zdroji (subjektu) přidat textovou informaci jako lidsky čitelnou náhradu pro označení zdroje
- **rdfs:comment** dovoluje přidat zdroji popisek, který usnadňuje lidskému uživateli pochopit význam zdroje [6]

Vyjádření vztahů mezi jednotlivými vlastnostmi a třídami je patrné z obrázku 2.4. Rámec RDFS dále ještě definuje vlastnosti a třídy pro kontejnery a kolekce, které je možné naleznout ve [6].



Obrázek 2.4: Vztahy mezi vlastnosmi a třídami RDFS

Literatura

[1] Pitner Tomáš Matulík Petr. *Sémantický web a jeho technologie*. Masarykova univerzita, 2004. http://www.ics.muni.cz/zpravodaj/articles/296.html, (přístup 6.2.2013).

[2] Štencek Jiří. *Užití sémantických technologií ve značkovacích jazycích*. Vysoká škola ekonomická v Praze, 2009. http://vse.stencek.com/semanticky-web/, (přístup 4.2.2013).

[3] Vitvar Tomáš. *Sémantický web*. ČVUT, Praha, 2011, http://www.cvut.cz/pracoviste/odbor-rozvoje/stranky/habilitace-a-inaugurace/habilitacni-pr

[4] W3C. *RDFConcepts and Abstract Syntax*, 2004. http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/, (přístup 7.2.2013).

[5] W3C. *RDF/XML Syntax Specification*, 2004. http://www.w3.org/TR/REC-rdf-syntax, (přístup 6.2.2013).

[6] W3C. *RDF Vocabulary Description Language 1.0*, 2004. http://www.w3.org/TR/rdf-schema/, (přístup 8.2.2013).

[7] W3C. *W3C Semantic Web Activity*, 2006. http://www.w3.org/2001/12/semweb-fin/w3csw, (přístup 8.1.2013).