FAKULTA INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

Zadání bakalářské práce

Název:	Využití ontologické analýzy pro zajištění sémantické interoperability heterogenních dat
Student:	Artěmij Danilov
Vedoucí:	doc. Ing. Robert Pergl, Ph.D.
Studijní program:	Informatika
Obor / specializace:	Informační systémy a management
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	do konce letního semestru 2022/2023

Pokyny pro vypracování

Téma přispívá k projektu Datového inkubátoru dat. Cílem práce je ontologická analýza klíčových domén a jejich propojení s datovými sadami tak, aby byla umožněna jejich sémantická interoperabilita.

1. Seznamte se s projektem Datového inkubátoru, problematikou sémantické interoperability, Unified Foundational Ontology, jazykem OntoUML a nástrojem OpenPonk.
2. Ve spolupráci s vedoucím vyberte několik klíčových domén a souvisejících datových sad.
3. Vytvořte ontologické konceptuální modely těchto domén.
4. Propojte ontologické konceptuální modely s datovými sadami a vytvořte pravidla pro jejich mapování.
5. Zdokumentujte své řešení a přínos pro zajištění sémantické interoperability.

Elektronicky schválil/a Ing. David Buchtela, Ph.D. dne 11. ledna 2022 v Praze.

Bakalářská práce

VYUŽITÍ ONTOLOGICKÉ ANALÝZY PRO ZAJIŠTĚNÍ SÉMANTICKÉ INTEROPERABILITY HETEROGENNÍCH DAT

Artěmij Danilov

Fakulta informačních technologií
Katedra softwarového inženýrství
Vedoucí: doc. Ing. Robert Pergl, Ph.D.
23. června 2022

Odkaz na tuto práci: Danilov Artěmij. *Využití ontologické analýzy pro zajištění sémantické interoperability heterogenních dat.* Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií, 2022.

Obsah

Seznam obrázků

Seznam tabulek

Seznam výpisů kódu

Rád bych poděkovat především mému vedoucímu doc. Ing. Robert Perglovi, Ph.D. za trpělivost a pochopení během vypracování bakalářské práce. Také bych rád poděkoval své rodině a přátelům za jejích nekonečnou podporu.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, avšak pouze k nevýdělečným účelům. Toto oprávnění je časově, teritoriálně i množstevně neomezené.

V Praze dne 23. června 2022 .....

Abstrakt

Tato bakalářská práce se zaměřuje na použití ontologické analýzy k zajištění sémantické propojitelnosti heterogenních dat v rámci různých klíčových domén.

Součástí práce je ontologická analýza domén a datových sad v rámci modelovacího týmu projektu Nest Big Data Arena, následná tvorba konceptuálních modelů těchto domén a pak propojení dat z uvedených datových sád s vytvořenými modely pomocí mapování. Pro tvorbu modelů byl využit modelovací jazyk OntoUML a platforma OpenPonk.

Celkem během práce byly vytvořeny dva konceptuální modely s dokončeným mapováním dat a jedna šablona sloužící ke zlepšení následné analýzy a práce se stejnou doménou. V rámci těchto modelů je kladen důraz na jasnou definici využitých pojmů, znázornění jejích vzájemných vztahů uvnitř domény a dosažení interoperability vzniklých modelů mezi všemi heteregonními datovými sadami v rámci projektů Nest Big Data Arena.

Klíčová slova konceptuální model, ontologická analýza, heterogenní data, Nest Big Data Arena, sémantická interoperabilita, Fair data, OpenPonk, OntoUML, UFO

Abstract

This bachelor thesis focuses on using ontological analysis to assure semantic connectivity of heterogeneous data within the scope of different key domains.

This thesis includes ontological analysis of domains and data sets within modeling team of Nest Big Data Arena project, creation of conceptual models of those domains and then connection of data from aforementioned data sets and the created models using the process called mapping. Modeling language OntoUML and OpenPonk platform were used in creation of the models.

During my work two conceptual models with finished mapping of the data and one template serving the purpose of improving subsequent analysis and work with a given domain were created altogether. In creation of those models a special emphasis was given to a clear definition of used concepts, representation of relationships between those concepts and achieving interoperability between created models and all of the heterogeneous data sets already within Nest Big Data Arena project.

Keywords conceptual model, ontological analysis, heterogeneous data, Nest Big Data Arena, semantic interoperability, Fair data, OpenPonk, OntoUML, UFO

Seznam zkratek

NBDA	Nest Big Data Arena
UFO	Unified Foundational Ontology
UML	Unified Modeling Language
IT	Information technology
DOLCE	Descriptive Ontology for Linguistic and Cognitive Engineering
GFO	General Formalized Ontology

Úvod

V dnešní době využití internetu jako zdroje informací se pro mnoho z nás stává největším a někdy i jediným zdrojem znalostí. A i přestože je tato praxe samozřejmostí, přichází s ní spousta problému. Velkým a často podceňovaným problémem je nejednoznačnost a nepřesnost používaného jazyka. Jelikož neexistuje jednotný způsob popisování konceptu velmi často se setkáváme s případem, kdy jednotlivé internetové zdroje používají různé termíny pro popis téhož pojmu (např. auto, automobil, vůz) nebo naopak používají stejné termíny pro různé pojmy (např. řidič může v různých kontextech znamenat různé věci). Tato práce se zaměřuje právě na eliminaci tohoto problému v rámci projektu Nest Big Data Arena společnosti Remmark, a.s.

Podstatou NBDA projektu je vytvoření datové platformy, na které bude shromážděné velké spektrum dat a výzkumu z oboru marketingu. Cílem je datové úložiště, které zajistí uživateli jednoduchý a jednoznačný přístup ke všem uloženým informacím jako celku. Je tedy velmi důležité, aby pro všechny datové sady a pro všechny užité pojmy uvnitř úložiště platila jasnost a jedinečnost jejích významů, aby nedocházelo k výše zmíněnému problému. A v tom právě spočívá zařízení sémantické interoperability dat a moje práce.

Sémantická interoperabilita je v tomto případě zařízená přes využití ontologické analýzy obsahu datových sad a porovnání vyskytnutých klíčových pojmů s pojmy z již zpracovaných sad. Za využití informací vytažených během analýzy se vytvoří konceptuální model důležitých entit pro danou datovou sadu obohacený o esenciální doménové entity a vztahy pro dosažení co nejpřesnější a nejjednoznačnější definice pojmu. Na konec po umístění modelu na datovou platformu se obsah datových sad přímo napojí na vzniklý model pomocí mapování atributu datové sady na vytvořené entity.

V teoretické částí práce nejdříve uvedu důležité pojmy a informace potřebné k pochopení celé problematiky sémantické interoperability. Následně v rámci praktické části využijí modely a šablony, které jsem v průběhu práce vytvořil abych přiblížil celý modelovací proces a poukázal na konkrétní problémy, se kterými se můžeme během tohoto procesu setkat.

Kapitola 1
Cíl práce

Jádrem práce a primárním cílem je analýza klíčových datových sad, dodaných Remmark, a.s., v rámci projektu NBDA, a následné vytvoření konceptuálních modelů s provedeným mapováním tak, aby byla zajištěna sémantická interoperabilita dat.

Cílem teoretické části je představení problematiky sémantické interoperability. Součástí je uvedení podstaty konceptuálního modelování a význam ontologie pro tento proces, přiblížení principu FAIR data a seznámení s jazyky UML a OntoUML se zmínkou platformy OpenPonk.

Cílem praktické části je vytvoření ontologických konceptuálních modelů a jejich následné propojení s datovými sadami. Na vzniklých konceptuálních modelech bude předveden postup jejich tvorby a problémy, se kterými jsem se při tvorbě modelů setkal. Taktéž je uveden smysl a důležitost tvorby šablon při vykonání ontologické analýzy a konceptuálním modelování.

Část I

Teoretická příprava

Kapitola 2

Interoperabilita a úvod do problematiky

Pojem interoperabilita označuje schopnost spolupráce. Ve sféře IT se vztahuje hlavně ke schopnosti různých komponent nebo systému vzájemné kooperovat i přes rozdíly v jazyce, rozhraní nebo platformě. Jedna se tedy o míru jejích vzájemné propojenosti. [1]

2.1 Sémantická interoperabilita

Interoperabilitu mužem rozdělit do více druhu podle toho k jakému typu propojenosti referují. Například se může jednat o syntaktickou interoperabilitu, která se vztahuje k možnosti zpracování sdílených dat, díky existenci a použití standardních syntaktických struktur. [2]

Pro nás je ale nejdůležitější interoperabilita sémantická. Tedy propojení systému a hlavně jejích obsahu na významové úrovni.Přesná definice podle [3] je obsahové vyjádření struktury metadat, které dovoluje sémanticky kombinovat datové prvky různých schémat, slovníků či modelů a umožňuje tak jednoduše vyhledávat informace napříč heterogenními distribuovanými databázemi. Mluvíme tedy o situací, kdy nejednoznačnosti významu použitých výrazu a konceptu v rámci systému nebrání spolupráci těchto systému čí části systému. Obecně tento problém můžeme rozdělit na případ kdy jeden použitý vyraz muže v rámci různých systému znamenat rozdílné věci (např. monokl jako typ brýlí a monokl jako slangově podlitina) a nebo když dva rozdílné pojmy reprezentují tu samou věc (např. obchod a prodejna).

2.2 Remmark a.s. a NBDA projekt

Projekt Nest Big Data Arena na problematiku interoperability přímo navazuje. Ze jeho realizaci stojí Remmark, a.s, což je česká full-service agentura, která se na trhu pohybuje již od roku 1998. Specializuje se primárně na tvorbu a realizaci informačních kampaní. [4]

Cílem projektu NBDA je tvorba a provoz datové platformy, která bude agregovat různé informace, statistiky a výzkumy z oborů marketingu a komunikace. Tyto data představují heterogenní sady, které nemají společný slovník a nejsou významově propojené. Dalším a primárním cílem je tedy u uložených datových sad zařídit interoperabilitu, tedy tyto sady propojit. Následně budou vzájemně propojena data poskytnuta v rámci platformy uživatelům, kteří s nimi budou mocí jednoduše pracovat jako s celkem(např. vyhledávat, filtrovat a uspořádávat data napříč všemi uloženými sadami). [4]

Táto skutečnost se stává obrovskou výhodou pro každého kdo plánuje nějaký druh marketingové aktivity. Data, která jsou za normálních okolnosti roztroušená, uložena na různých místech, v různých formátech a významově nepropojena jsou teď na jednom místě a přístupna uživateli s jasným rozhraním pro jejích manipulaci. To spolu se schopnosti platformu konstantně rozšiřovat a v budoucnu propojovat informace i z jiných sektorů může mít až nekonečný potenciál.

V roce 2020 navíc získala firma Remmark, a.s. podporu z Evropských strukturálních a investičních fondů, skrz Operační program Praha – pól růstu ČR a čerpanou přes 3. výzvu Pražský voucher na inovační projekty na podporu NBDA. [4]

2.3 Sémantická interoperabilita v rámci NBDA

Jak jíž bylo zmíněno, zařízeni propojeností datových sad v rámci platformy je základem celého projektu. Pro dosažení tohoto cíle musí být dosažena i sémantická interoperabilita. Jako řešení sémantické interoperability je v rámci projektů použita ontologická analýza a správný návrh sémantických konceptuálních modelů datových sad, který se o analýzu opírá. Nedodržení sémantické interoperability v rámci platformy muže vést k duplikaci ukládaných dát a nebo kolizemi objektu v modelech. To pák znamená zkomplikování nebo znemožnění jejích automatického zpracování systémem a tedy vymizení všech benefitu datové platformy.

Zařízeni sémantické interoperability je sice velkou součásti hlavně této práce, není ale dostačující pro zajištěni podstaty projektu NBDA a vybudovaní funkční datové platformy. Z tohoto důvodu se celý projekt ve velké míře opírá na všechny principy FAIR.

Kapitola 3
FAIR principy

Množství dát napříč internetem je stálé větší a větší. A s množstvím dát roste i jejích nejednotnost. Data jsou uložena v různých formátech a na různých místech, jejích původ často není dohledatelný a jejích význam nemusí být jednoznačný. To celé znemožňuje je strojově zpracovávat a výrazně zhoršuje jejích použitelnost. Jelikož je v moderním světě manuální zpracování dát ve velkým měřítku skoro nemožné, byly definovaný FAIR principy jako způsob minimalizovat v budoucnu tento problém. Jsou to principy a pravidla, která zajišťuji udržitelné a efektivní publikování dát. Samotný název FAIR je akronymem pro *Findable*, *Accessible*, *Interoperable*, *Reusable*. [5]

3.1 Vznik FAIR

Principy FAIR byly prvně definované v roce v 2016 v publikaci 'FAIR Guiding Principles for scientific data management and stewardship' v rámci časopisu Scientific Data. Tato publikace navazovala na konferenci Jointly designing a Data FAIRPORT, která byla zorganizovaná v roce 2014 v Lorentz Center v Nizozemsku. [6] A pravě v rámci této konference vznikly zárodky principu, které by zajistili dohledatelnost, dostupnost, propojenost a znovupoužitelnost dat.

Zajímavé je, že FAIR principy nedefinují přesný způsob implementace nebo potřebné technologie ale pouze ukazují postupy pro dosažení vyžadovaných vlastností. V dnešní době jsou FAIR principy zastupovaný a propagovaný v rámci iniciativy GOFAIR. GOFAIR se snaží o šíření osvěty ohledně FAIR dat a o pomoc s nasazením FAIR principu. [7]

3.2 Rozdělení FAIR

Kategorie do kterých se principy FAIR děli popisují jak by měla vypadat data pro jejích efektivní zpracování.

1. Findable (Dohledatelná) [7]

   Prvním důležitým krokem v použití dát je jejích nalezení. Tento proces by měl být jednoduchý jak pro lidi tak i pro stroje. Proto je zavedení správných identifikátoru a metadat esenciální.

   a. Datum a metadatům je přiřazen unikátní a trvalý identifikátor.
   b. Data jsou v dostatečné míře popsána pomocí metadat.
   c. Metadata mají jasně přiřazený identifikátor dat, které popisují.
   d. Data a metadata jsou registrovaná a indexovaná v rámci zdroje ve kterém můžeme vyhledávat.

2. Accessible (Dostupná) [7]

   Po nalezení dát dalším důležitým krokem pro uživatele, ať člověka nebo stroje, je otázka jak k ním dostát přistup.

   a. Data a metadata jsou získatelná pomocí jejích identifikátoru za použiti standardizovaného komunikačního protokolu. Tento protokol musí být otevřený, volně přístupny a univerzálně implementovatelný. Navíc je požadavkem aby protokol umožňoval autentizaci a autorizaci.

   b. Metadata musí být přístupna, i po té co data na která odkazují již nejsou k dispozici.

3. Interoperable (Propojená) [7]

   Popisují jak dosáhnout propojenosti dat.

   a. Data a metadata by měla využívat formálního, dostupného a široce použitelného jazyku pro svou reprezentaci.

   b. Slovníky vytvořené pro datové sady by měli taktéž splňovat FAIR principy.

   c. Data a metadata by měla obsahovat kvalifikované odkazy na jiná data a metadata

4. Reusable (Znovupoužitelná) [7]

   Hlavním cílem FAIR principu je optimalizace znovupoužitelnosti dat. Pro jeho dosažení data by měly být důkladně popsaný aby mohly být jednoduše replikovaný a kombinovaný s jinými daty v rámci jiných kontextu.

   a. Data a metadata jsou důkladně a přesně popsána pomocí velké množiny relevantních atributu.

   b. Data a metadata jsou publikována s jasnou a dostupnou licenci na použiti.

   c. Data a metadata by měla obsahovat jasný informace o tom odkud se vzaly.

   d. Data a metadata by měla splňovat doménově stanovené standarty

3.3 FAIRifikační proces

Nejdůležitějším komponentem iniciativy GOFAIR je definice FAIRifikačniho procesu. Je to proces, který popisuje jak zpracovat data, která FAIR principy nesplňují, a proměnit je v FAIR data.

Celý proces se dělí do sedmi různých kroků. [8]

1. **Zisk neFAIRifikovaných dát**

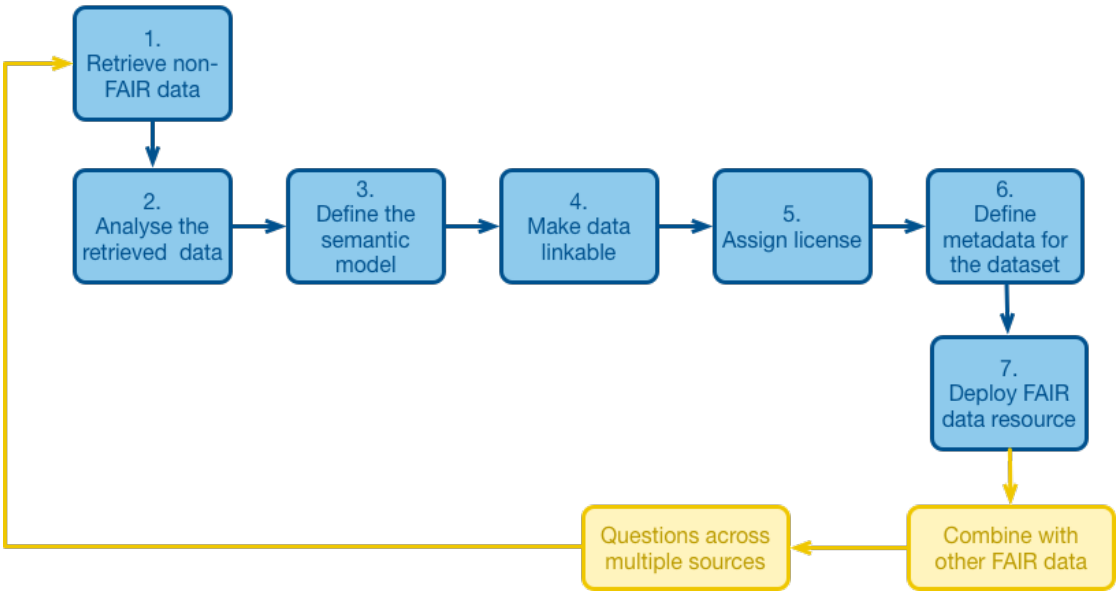
   Tento krok spočívá v zisku přístupu k datům, které chceme FAIRifikovát. [8]

2. **Analýza neFAIRifikovaných dát**

   Spočívá v analýze získaných dát za účelem vytěžit informace o nich. Zde se primárně zaměřujeme na to jakou mají data strukturu, z jaké domény pochází a jaké vztahy existují mezi datovými elementy. [8]

3. **Vytvoření sémantického modelu**

   Proces založený na výsledcích minulého kroku. Jedna se o vytvoření konceptuálního sémantického modelů, který popisuje získanou strukturu dát. Při tvorbě modelů se taký doporučuje opravdu důkladná ontologická analýza domény ze které data pochází, tedy již existujících ontologii, modelu a slovníku. [8]



Obrázek 3.1 Schema FAIRifikačního procesu [8]

4. **Zařízení linkovatelností dát**

   Po aplikaci vzniklého sémantického modelu na data je transformujeme na data linkovatelná.[8]

5. **Přidělení licence**

   Přidělení licence a způsobu použití nově vzniklých dát. Tento krok je obzvlášť důležitý protože absence explicitně uvedené licence může zabránit následnému využití dát jinou stanou i když to nebylo naším cílem. [8]

6. **Definice metadat pro datovou sadu**

   V tomto kroku by se k datum měli přidat relevantní metadata. Tento proces dramatický zjednoduší následné využití těchto dat. [8]

7. **Nasazení FAIR dat**

   Finální krok FAIRifikačniho procesu. Pojednává o správné publikaci vzniklých FAIR dát. Zde je kladen důraz na splněni předchozích kroků, a to hlavně doplnění dát o metadata a přidaní licence, před publikaci. [8]

Kapitola 4
Ontologie

Jak vychází z FAIR principu pro dosažení efektivního využití dat je důležité zařízeni jejich interoperability. Velkou součásti toho je i zařízení interoperability sémantické 2.1. Toho ale nelze dosáhnout bez aktivního využiti ontologie. [2]

4.1 Definice ontologie a její vznik

Ontologie (z řečtiny *ontos* jsoucí a *logos* výklad) je odvětví filozofie, které se soustředí na otazky toho co věci jsou, jakou mají strukturu a význam, jaké vztahy sdělují, co prožívají a jakých události se účastni ve všech částech našeho světa. Zkráceně, ontologie se zabývá bytím jako takovým a snahou ho popsat. Její kořeny vedou až k počátkům filozofie a Aristotelu, kdy se původním zkoumáním ontologického zaměření říkalo Metafyzika. [9]

4.2 Ontologie v IT

Dnešní význam ontologie v IT je trochu odlišný od toho filozofického. Bere z ní všechno potřebné, tedy snahu o poskytnutí úplné definice všech objektů a vztahu mezi nimi, a přidává potřebu pro explicitní vyjádřeni těchto definic v dostatečné míře pro vyřešení daného problému. Podle T. Gruebera je ontologie pak "explicitním vyjádřením konceptualizace". [10] Není tím pádem překvapivé, že pojmy část, systém, relace, událost, prostor, čas atd., které jsou definovány v rámci ontologie filozofické, se hojně využívají i v konceptuálním modelování, což je pravě proces vyjádřeni a zobrazení významu a vztahu námi zkoumaných entit.

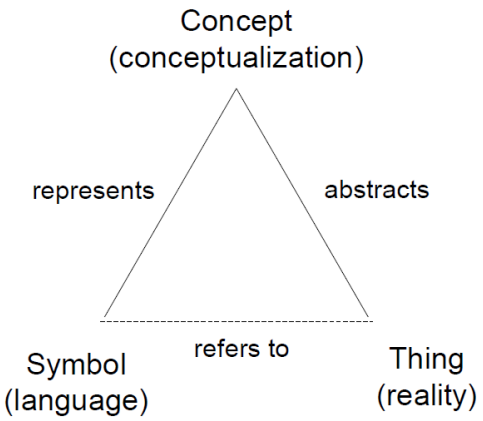
4.3 Rozdělení ontologie

Obecně můžeme ontologii rozdělit na velké množství kategorii podle různých vlastnosti. Mezi primární patři rozdělení podle historických paradigmat a podle předmětu formalizace. [11] [12]

1. Rozdělení podle historických paradigmat

   a. Terminologická ontologie

      Tento druh ontologie se vztahuje na obory, které se opírají na textové zdroje informaci, jako například knihovnictví. Lze jí přirovnávat k tezaurům a často znázorňují sémantické vztahy mezi slovy(synonyma, antonyma, holonyma atd.).



Obrázek 4.1 Ullmanův trojúhelník. Zobrazuje vztah mezi reálnou věcí, naší konceptualizaci této věci a symbolu pomocí, kterého naší konceptualizaci reprezentujeme [11]

b. Informační ontologie

Informační ontologie spočívá primárně v rozvinutí databázových schémat. Je zde kladen velký důraz na formální stránku použitých jazyku.

c. Znalostní ontologie

Znalostní ontologie se zabývá reprezentaci dat v rámci umělé inteligence.

2. Rozdělení podle předmětu formalizace

a. Top-level ontologie

Top-level ontologie popisuje obecné koncepty nezávisle na určité doméně nebo problematice. Je zde snaha zachytit nejobecnější pojmy a vztahy aby se následně mohli využít v rámci tvorby konkrétnějších ontologii.

b. Doménové ontologie

Charakterizují pojmy specifické pro určitou doménu. Důležité je, že tyto pojmy nedefinují ve vakuu ale opírají se na definice z top-level ontologii.

c. Úlohové ontologie

Úlohové ontologie jsou podobné doménovým. Zaměřují se na konkretní úlohu či problém. Na rozdíl od doménové ontologie, které se často fokusují na doménu v statickém smyslu(prodejna nářadí či nemocnice), úlohové ontologie se zaměřují na nějakou akci(prodej nebo diagnostika).

d. Aplikační ontologie

Aplikační ontologie kombinují vlastností a informace ontologii předchozích. Často se soustředí na vyřešení konkretní situace v rámci určité domény.

4.4 Konceptuální modelovaní

Konceptuální modelovaní je činnost, která je s ontologii úzce spojena. Jejím cílem je získat formální popis/model zkoumané domény, aktivit, které se v ní odehrávají a vztahu, které entity v

rámci této domény sdělují. Je to tedy proces, kterým můžou jednotlivé ontologické struktury vznikat. Jelikož podstatou konceptuálního modelování je zajištění porozumění popsaných konceptu, je důležité, aby výsledný popis používal vhodný jazyk, pochopitelný pro všechny zúčastněné, a měl dostatečnou úroveň abstrakce. [13] [14]

Abstrakce je v tomto případě důležitá hlavně pro identifikaci opravdu důležitých a podstatných pro tvořenou ontologii pojmu. Slouží k zjednodušení procesu modelování a zabraně vzniku zavádějících modelů. [15]

Další esenciální části konceptuálního modelování je jazyk používaný k uchování modelů. Je potřeba aby jazyk byl pochopitelný pro všechny strany a zároveň byl schopný reprezentovat konceptualizaci stručně, kompletně, jasně a jednoznačně. Existuje množství jazyku, které se ke konceptuálnímu modelování používají. Obecně je můžeme dělit na jazyky doménově závisle a jazyky doménově nezávisle. Doménově závisle jazyky, jako například DEMO, jsou ohraničené ve svém využití pouze na doménu čí domény pro které byly speciálně vytvořené. Toto často umožňuje použití specifických metodik a větší úroveň abstrakce, což vede k usnadňují vytvoření nebo pochopení modelu v rámci dané domény. Doménově nezávislé jazyky mají naopak větší flexibilitu, jelikož nejsou vázané na určitou sféru. [16] Příkladem takového jazyka je OntoUML, který byl pravě z tohoto důvodu využíván v rámci mé bakalářské práce a projektu NBDA.

4.5 UFO - Unified Foundational Ontology

UFO je ontologii, která v sobě kombinuje prvky dvou jiných ontologii - DOLCE a GFO. Byla vyvinuta Giancarlem Guizzardim a jeho týmem na počátku 21. století. Zajímavé na ní je to, že byla vytvořena se snahou vytvořit vyšší referenční ontologii, která bude plně splňovat potřeby pro konceptuální modelovaní [11]

UFO od svých předchůdců přebírá velmi důležité uznávaní čtyř kategorii objektu. Toto rozdělení původně pochází již z Aristotelovy doby a rozděluje věci mezi partikulary a univerzaly, a mezi podstatné a nepodstatné. Partikulary jsou konkretní věci a objekty a univerzaly jsou sdělitelné vlastností nebo charakteristiky těchto objektu(např. Artěmij Danilov je konkretní partikular a ”člověk”by byl jeden z univerzalu, který ho popisuje). [17] [18]

Podstatné věci jsou takové které podmět nutně potřebuje pro jeho existenci a pro to aby byl tím co je. Nepodstatné jsou přesným opakem. Jsou to věci, které jsou navíc a nejsou nutně potřebné k existenci podmětu. [17]

Toto rozdělení bylo následně doplněno o takzvané mikroteorie, které stavějí na odvětví filozofické ontologie, logiky a mnoha dalších. Tyto mikroteorie tvoří nástroj pro explicitní popis entit a vztahu mezi nimi, což je pro správný popis domény, a tedy i konceptuální modelování, esenciální. Tyto mikroteorie spolu s čtyř-kategorickým přístupem tvoří základ principu UFO. [19]

4.6 Rozdělení UFO

UFO ontologie se obecně dělí do tři různých části. [11] [13] [20]

1. UFO-A

   Ontologie endurantů. Zabývá se strukturálními aspekty reality. Pracuje s entity, jejími vlastnostmi a vztahy.

2. UFO-B

   Ontologie perdurantů. Zabývá se behaviorální aspekty reality. Pracuje s událostmi, chováním a kauzalitou.

3. UFO-C

   Ontologie pracující se sociálními aspekty reality. Je postavena na UFO-A a UFO-B a pracuje s organizacemi, agenty a jejích chováním či úmysly.

Kapitola 5
UML

UML neboli Unified modeling language je grafický modelovací jazyk, který se primárně využívá pro zobrazení architektury a procesu vývoje softwarových projektu. Je to velmi silný nástroj, který může pomoct jak během návrhu softwaru tak i během jeho implementace a údržby. Jeho velkou výhodou je jednoduchost použiti a přehlednost vytvořených modelů. Poskytnuta míra abstrakce totiž umožňuje prolomit komunikační bariery mezi programátory, kteří se starají o implementaci, návrháři, které jsou zodpovědné za architekturu a managementem, který kódu nemusí vždy rozumět. [21]

5.1 Kategorie modelu

UML poskytuje velké množství různých typu modelu, které se dělí do tři kategorii podle toho jakou skutečnost reprezentují. Jsou to strukturální modely, modely chování a modely interakci. Modely interakci přímo vycházejí z modelu chování a tak se často zařazují jako jejích poddruh.

1. Strukturální modely

   Strukturální modely ukazuji stav a strukturu systému v jeho statické podobě. Modely této kategorie se zaměřují na různé části systému a znázorňuji jak jsou spolu propojené. Můžou přibližovat konkretní fungování implementace těchto části, díky poskytnuti různých úrovní abstrakce. Obecně elementy strukturálních modelů jsou koncepty, které hrají důležitou roli v rámci systému. [21] [22]

   Mezi strukturální modely patři model tříd, model objektu, model komponent, model kompozitní struktury, model balíčku a model nasazení. [21]

2. Modely chování

   Modely chování ukazují dynamickou činnost objektu v rámci systému. Modely této kategorie se soustředí na znázorněni pohyblivých části systému(to zahrnuje i lidi) a znázorňuje jejích chování jako sérii změn v systému v určitým časovém úseku. [21] [22]

   Mezi modely chování patři model případu použiti, model aktivit a stavový automat. [21]

3. Modely interakci

   Modely interakci vychází z modelu chování a znázorňuji primárně interakce mezi jednotlivými části systému a to jak se navzájem ovlivňují.

   Mezi modely chování patři sekvenční model, komunikační model, časový model a model přehledu interakci. [21]

5.2 Univerzalita UML

Největší výhodou UML je jeho univerzalita. Není doménově vázaný a množství poskytovaných diagramu umožňuje použiti UML v nespočetném množství situaci. Je to nástroj, který může využít jak developer softwaru, tak i někdo kdo se pohybuje ve sféře byznysu. Jeho nejsilnější vlastnosti se stává ale rozšiřitelnost. Spousta firem v dnešní době adoptuje UML pro své projekty s implementaci změn, které umožňují aby vzniklý jazyk se hodil přesně na míru implementovanému projektu. Podobným případem je jazyk OntoUML, který adoptuje základy UML a kombinuje je s principy UFO, čímž vzniká perfektní stroj pro ontologické konceptuální modelování. [23] [24]

Kapitola 6
OntoUML

OntoUML je ontologický jazyk pro konceptualní modelování vytvořený Giancarlo Guizzardim. Vychází z UFO a používa notaci UML diagramu tříd. Vzniknůl jako pokus o vytvoření jednotného nastroje pro použíti během konceptualního modelování a tvorby doménových ontologii. [24]

6.1 Základní principy

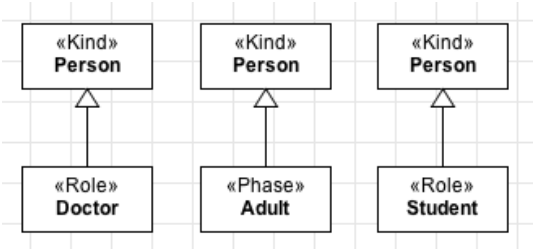
6.1.1 Typ a individuum

Jelikož se OntoUML zakládá na UFO, je pro jeho fungování důležité rozdělení věci mezi univerzaly a partikulary(viz. 4.5). V OntoUML ale pracujeme s trošku jinou terminologii. Rozdělujeme věci na typy a individua. Typy podobně univerzalum se týkají abstraktních věcí, které nám pomáhají chápat a budovat v hlavě odpovídající představu reality. Typy existují jako způsob klasifikace a popisu individuu. V OntoUML jsou typy hlavním stavebním kamenem modelování. V rámci modelu jsou typy znázorněny pomocí čtverečku a nazýváme je třídy. [25]

Konkretní příklady individuu, které jsou popsaný některým typem, nazýváme instanci toho typu. Pokud existuje typ, který popisuje podmnožinu instanci jiného typu(a jenom tu podmnožinu), pak ten typ můžeme nazvat podtypem většího typu. Vztah který sdělují pak názvem generalizaci. [25]

6.1.2 Generalizace

V rámci modelu znázorníme generalizaci pomocí šipky mezi dvěma třídami. Tuto skutečnost si mužem představit jako dědění v programovacích jazycích. Zajišťuje sdílení vlastností a principu identity 6.1.3 mezi různými typy.



Obrázek 6.1 Typy a podtypy spojené pomocí generalizace [25]

Jako příklad dejme tomu existenci dvou typu - typ Člověk a typ Student. Jelikož každá instance Studenta je taký instanci Člověka a vlastností patřičné pro Člověka patří i každému Studentu můžeme říct, že typ Student a typ Člověk v našem modelu můžou mít vztah generalizace. Student tedy bude podtypem Člověka.

6.1.3 Princip identity

Dalším důležitým konceptem pro OntoUML je princip identity. Princip identity si mužem představit jako něco co instanci dělá sebou a podle čeho jí můžeme odlišit od jiných instanci stejného typu. Princip identity je instancím poskytován od některého z typu, ke kterým patří. [26]

Princip identity musí být pro každou instanci stály a neměnitelný časem. Navíc musí platit pro každou instanci daného typu stejný, a pravě jeden, princip identity. To znamená, že každá instance muže být instanci pouze jednoho typu, který má schopnost poskytovat identitu. Typy ovšem můžou mít několik principu identity najednou, jelikož obecné koncepty nemají jedinou identitu. [26]

Typy s jednoznačným principem identity nazýváme Sortaly a ty, které jednoznačný princip identity nemají nazýváme non-Sortaly. [26]

6.1.4 Rigidita

Posledním důležitým pojmem, který stojí za zmínku je rigidita. Rigidita označuje schopnost instanci daného typu se proměňovat. Pokud individuum musí instancovát nějaký typ ve všech možných situacích dokud individuum existuje pak je tento typ Rigidní. V opačném případě bude typ Anti-Rigidní. [27]

Představme si dítě a pojmenujeme toho individua Josef. Josef definitivně bude patřit k typu Člověk a k jeho podtypu Ditě. Po uplynutí času Josef přestane být dítětem, ale stále zůstane člověkem. Tedy přestane být instanci typu Dítě. To vypovídá, že existuje scénář kdy individuum, který byl instanci Dítě již instanci Ditě není a tedy typ Ditě bude anti-rigidní.

6.2 Stereotypy

Jednou z nejdůležitější častí modelování v OntoUML jsou stereotypy. Během modelování přiřazujme třídám jejích stereotypy podle vlastností konceptu, které tyto třídy představují. Existence stereotypu umožňuje precizněji vyjádřit podstatu těchto konceptu a tedy vytvořit přesnější konceptualizaci reality.

6.2.1 Sortaly

1. Kind

   Stereotyp Kind je rigidním Sortalem. Slouží v modelech pro reprezentaci neměnných konceptu, které poskytují svým instancím identitu. Jelikož má Kind jednoznačný princip identity nemůže se nikdy stát podtypem jiného Sortalu, jeho instance by pak totiž měla dva principy identity. [28]

2. SubKind

   Stereotyp SubKind je taktéž rigidním Sortalem. Jeho rozdílem oproti Kind je to, že nemá vlastní princip identity. Svůj princip identity totiž dědí od některého typu, jehož je podtypem. Jeho instance, tedy stále splňují pravidlo jednoho principu identity, jelikož princip identity, který dostanou od SubKind a od jeho nadtypu poskytujícího princip identity bude stejným principem. [28]

3. Phase

   Stereotyp Phase je anti-rigidním Sortalem. Podobně SubKind svůj princip identity dostáva pomocí dědění. Tento stereotyp znázorňuje určité etapy, v rámci existence konceptu, které jsou založené na vnitřní vlastností nebo stavu tohoto konceptu. Pro pochopení Phase nám může pomoct již představený Josef. Josef je dítětem, který postupem času se stane dospělým. Dítě a Dospělý tak můžou být stereotypu Phase, jelikož popisují koncept, který je založený na vnitřní vlastností jejích instanci - věku. [28]

   Důležité je, že v rámci generalizace od jedné třídy, nemůže existovat pouze jedná Phase. Koncept tedy musí mít minimálně dva stavy aby byl reprezentován pomocí stereotypu Phase. [28]

4. Role

   Stereotyp Role je anti-rigidním Sortalem, který taký znázorňuje určitý stav entity. Rozdílem je, že tento stav se jíž nezakládá na vnitřní vlastností entity ale na vnějším vztahu s jinou entitou. Svůj princip identity dostává podobně Phase pomocí dědění od jiné třídy. [28]

6.2.2 Non-Sortaly

1. Category

   Stereotyp Category je rigidní non-Sortal. Obecně se používá pro popis společných vlastností entit, které mají různé principy identity. Třídy, kterých se dána vlastnost týka jí pak můžou získat pomocí generalizace. [29]

2. PhaseMixin

   Stereotyp PhaseMixin je anti-rigidní non-Sortal. Podobně Category slouží k popisu společných vlastností entit. PhaseMixin je ale vytvořený speciálně pro popis vlastností entit se stereotypem Phase, které ale mají rozdílné principy identity. [29]

3. RoleMixin

   Stereotyp RoleMixin je anti-rigidní non-Sortal, který má stejnou roli jako PhaseMixin, tedy popis společných vlastností entit s rozdílným principem identity, ale aplikuje se na entity se stereotypem Role. [29]

4. Mixin

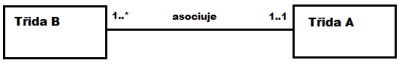
   Stereotyp Mixin je semi-rigidním non-Sortalem. To označuje, že je to non-Sortal, který může byt jak rigidní tak i anti-rigidní. Je to jediný stereotyp, který má tuto vlastnost. Tento stereotyp se používá v případech pokud potřebujeme vyjádřit některou vlastnost, kterou sděluje rigidní a anti-rigidní typ s různými principy identity. [29]

OntoUML nabízí i další stereotypy. Pro jejích porozumění je ale důležité představit ještě jeden druh vazby mezi třídy, který OntoUML definuje. Touto vazbou je Asociace.

6.3 Asociace

Asociace je druh relace v OntoUML. Vyjadřuje vztah mezi dvěma třídami a je reprezentována čarou tyto dvě třídy spojující. Velkým rozdílem oproti generalizaci je, že u asociace definujeme násobnost relace. Násobnost relace znázorňuje s kolika instancemi třídy na jedné straně asociace má nutně vztah instance třídy na druhé straně asociace. Násobnost je reprezentována ve tvaru X..Y na každé straně asociace, kde X a Y reprezentují spodní a horní mez. X a Z můžou být nahrazené za přirozená čísla nebo * reprezentující libovolné množství. [27]

Asociace může znázorňovat jakýkoliv relevantní vztah mezi dvěma objekty. Specifikujeme tento vztah pomocí pojmenování asociace.



Obrázek 6.2 Příklad použiti asociace spolu s násobností a pojmenováním

Název a multiplicita asociace vždy vychází z důkladné analýzy příslušné domény.

Asociace mužem podle typu vztahu, který reprezentují dělit na formální a materiální.

Formální relace vznikají na základě vnitřních vlastnosti instanci a na základě vztahu celku a části. Existují mezi dvěma entitami nepřímo, bez intervence některé další entity. U formálních relaci je důležitá jejích dohledatelnost původu. Tedy musíme být vždy v rámci entit schopný najít na základě čeho tato asociace vznikla. [27]

Materiální relace jsou takové asociace, které existují mezi dvěma entitami přes některou spojovací třetí entitu. Tuto třetí entitu nazýváme v OntoUML Relator a je to jedním z dalších užitečných stereotypu. [27]

- Relator

  Relator je rigidním sortalém se schopností poskytovat princip identity. Umožňuje propojení dvou entit za pomocí materiální relace. Relator obvykle znázorňuje objekt nebo entitu, která je pečetidlem vztahu. Tedy něco co daný vztah symbolizuje(např. manželská smlouva pro vztah mezi mužem a ženou). Slouží často k dekompozici vztahu s násobnosti M:N. Jedna se o vztahy kdy na obou stranách asociace se může vyskytovat libovolné množství instanci. Takové vztahy často védou k nejednoznačnosti v modelech. Existence stereotypu Relator nám pak umožňuje takové relace rozložit na dva vztahy s násobnosti 1:M a tím model zpřesnit. [27]

6.4 Aspekty

Aspekty jsou takové entity jejichž existence přímo závisí na existenci jiné entity. Tuto entitu označujeme jako nositelé aspektu. Aspekty často slouží k popsání nebo zdůraznění inherentních vlastností jejích nositelů. Závislost aspektu na jejích nositelích značí, že v případe zániku nositele přestává existovat i s ním spojený aspekt. Aspekt může být nositelem dalších aspektu. [30]

Rozlisujeme dva typy aspektu.

1. Mode

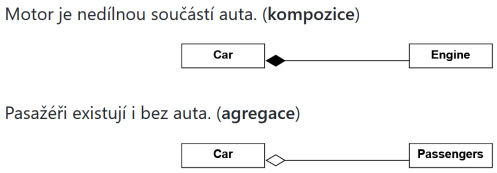
   Stereotyp Mode je rigidní sortál, který má schopnost poskytovat identitu. Slouží pro reprezentací vnitřních vlastností nositelé, které nejsou objektivně měřitelné(např. nálada). [30]

2. Quality

   Stereotyp Quality je rigidní sortál, který má schopnost poskytovat identitu. Na rozdíl od Mode slouží pro reprezentací měřitelných vlastností nositelé(např. věk, barva, velikost). [30]

6.5 Vztahy celku a jeho části

Vztahy celku a jeho částí jsou klíčová pro konceptuální modelování založené na ontologii, jelikož v reálném světě jejich porozumění hojně využíváme během kognitivního poznání světa. Umožňuje nám v rámci OntoUML vyjadřovat entity, které se skládají z více části. OntoUML přebírá a



Obrázek 6.3 Agregace a kompozice v UML [30]

rozšiřuje definici těchto vztahu z UML. UML definuje pouze dva vztahy vyjadřující celek a jeho část, jsou to agregace a kompozice. Kde kompozice znázorňuje nedílnost části od jejích celku a agregace vystihuje existenční nezávislost části na celku. [30]

OntoUML tyto definice pak rozšiřuje pro potřeby přesného ontologického modelování. Plný diamant, který v UML označuje kompozici, v OntoUML znamená, že část tohoto celku není sdělitelná, tedy nemůže ve stejnou chvíli být součásti více celku tohoto typu. Prázdny diamant pak znamená, že část je sdělitelná a může být součásti více celku stejného typu. Navíc přidává důraz na míru vzájemné závislosti mezi části a celky a i na směr takové závislosti. [30]

6.5.1 Povinnost části z hlediska celku

Povinnost části z hlediska celku můžeme rozdělit v OntoUML do tři kategorii. [30]

1. Volitelná část

   Označuje situaci kdy existence celku nezávisí na existenci části. Tedy může existovat celek jehož součásti v jeden okamžik jeho existence není žádná část. V OntoUML se to zobrazuje pomocí multiplicity nejméně 0 na straně části. [30]

2. Povinná část

   Vyjadřuje případ kdy existence části je podmínkou pro existenci celku. Celek muže v průběhu své existence instanci části měnit. Nesmí ale nastát situace kdy by celek zůstal bez přítomnosti nějaké instance části. Toto se reprezentuje pomocí multiplicity alespoň 1 na straně části. [30]

3. Esenciální část

   Označuje situaci kdy existence celku je přímo závislá na určité instanci části. To znamená, že po celou dobu existence celku je tato určitá část jeho součásti. Toto se zobrazuje v OntoUML pomocí klíčového slova Essential a multiplicity alespoň jedna na straně části. Problémem je, že Esencialní část můžeme použit pouze u relaci mezi rigidnimi typy. V případě anti-rigidních typu totiž muže nastát situace kdy instance přestane být instanci daného typu a jíž nebude vyžadovat její část k existenci. Pro vyjádřeni esenciality části u rigidních typu tak využijeme jiné klíčové slovo - Immutable part. [30]

6.5.2 Povinnost celku z hlediska části

V OntoUML rozlišujeme 3 typy povinnosti celku z hlediska části. [30]

1. Volitelný celek

   Vystihuje situaci kdy část může existovat bez přítomnosti celku. Navíc může během své existence celky libovolně měnit. V modelu se to značí multiplicitou nejméně 0 na straně celku. [30]

2. Povinný celek

   Vyjadřuje případ kdy část vyžaduje existenci celku pro svou existenci. Podobně jako u celku u 2, část může v tomto případe měnit celky během své existence ale nesmí nikdy existovat o samotě. Toto se reprezentuje pomocí multiplicity alespoň 1 na straně celku. [30]

3. Neoddělitelný celek

   Vyjadřuje situaci kdy část vyžaduje existenci určité instance celku pro svou existenci. Během existence části se tento celek nemůže měnit a je s ním propojena až do svého zániku. Označuje se klíčovým slovem Inseperable a multiplicitou alespoň 1 na straně celku. Podobně jako u Esencialní části je zde problém s použitím v relacích anti-rigidních typu. Pro takové relace tak používáme jiné klíčové označení - Immutable whole. [30]

6.5.3 Konstrukty celku a části

Pro úplný popis celek-část vztahu potřebujeme kromě popisu povinnosti také zavést i nové konstrukty v modelu a nové stereotypy.

1. Quantity

   Stereotyp Quantity je rigidní Sortal, který představuje celek jehož části jsou stejného typu jako on sám. Quantity se používá u nekonečně dělitelných objektu, které jsou udržovaný v určité formě nebo nádobě. Symbolizuje pak maximální spojitý objekt, který se s těchto nekonečně mnoha části skládá. Jako příklad si můžeme představit bazén napuštěný vodou. Maximálním spojitým objektem bude v tomto případe pravě celý bazén a část, ze který se skládá je voda. Quantity může poskytovat svým podtypům princip identity. [30]

   K tomuto stereotypu se vážou ještě dva typy relace

   - Containment
     Vztah mezi Quantity a formou či nádobou ve které je Quantity uchovávaná. [30]
   - SubQuantityOf
     Propojuje mezi sebou dva objekty typu Quantity a značí, že jedna se skládá z druhé. Zde je důležité podotknout, že jelikož v případe Quantity se vždy jedna o maximální spojitý objekt musí být multiplicita na straně části 1. [30]

2. Collective

   Stereotyp Collective je rigidní Sortal, který označuje situaci kdy celek je složen z části, které v rámci celku hrají stejnou rolí. Tento Sortal má schopnost poskytovat identitu svým podtypům. [30]

   Podobně jako v předchozím případě se k stereotypu Collective pojí dvě vazby.

   - MemberOf
     Relace vyjadřující vztah mezi Collective a jeho části. Pozor, v tomto případě násobnost na straně části musí být alespoň dva, jinak by se totiž nejednalo o kolektiv a tedy ani o stereotyp Collective. [30]
   - CollectionOf
     Relace mezi dvěma entity se stereotypem Collective. Vyjadřuje situaci kdy jeden Collective je tvořen z menších celku, které v jeho rámci splňují stejnou roli. [30]

3. Funkční celek

   Funkční celek není stereotypem v OntoUML ale spíše konstruktem. Je to nejběžnějším případem vztahu mezi celkem a části. Jedná se o situaci kdy instance různých stereotypu tvoří jednotný celek, ve kterým každá instance zastává jinou rolí. V závislosti na situaci může být reprezentován mnoha různými stereotypy, nejběžnější je ale reprezentace pomocí Kind. [30]

- ComponentOf
  Relace vystihující vztah mezi Funkčním celkem a jeho častmi. [30]

Kapitola 7
OpenPonk

OpenPonk je meta-modelovací platformou, která je vybudována na prostředí Pharo. Snaží se o podporu různých aktivit spojených s tvorbou softwaru. Je to velice univerzální platforma, která umožňuje své využíti v mnoha oblastech jako například konceptualní modelování, simulace a generování zdrojových kódu atd. Velkou výhodou platformy je její status jako open-source, což umožňuje jeho modifikaci pro vlastní projekty. OpenPonk je vyvíjen v Centru pro konceptuální modelování a implementace. [31]

Jeho hlavní výhodou pro daný projekt je podpora jazyku OntoUML. Poskytuje možnost tvorby všech standardních OntoUML konstruktu. Navíc umožňuje kontrolu vytvořených modelů, která automatický nachází chyby v špatně využitých stereotypech nebo nesprávné namodelovaných relacích. Navíc detekuje výskyt takzvaných antipatternů, což jsou konstrukty v OntoUML, které sice splňují obecná pravidla modelu ale často symbolizují, že je ve finálním modelů něco špatně z ontologického hlediska. Finální výhodou, která se během tohoto projektu využívala je možnost exportu vytvořených modelu ve formátu PNG. Toto umožňuje rychlý a jednoduchý způsob jak model sdílet někomu kdo platformu nevlastní. Například v případe práce s doménovým expertem bude pro něj často jednoduší pracovat pouze s obrázkem, než s celým modelovacím prostředím, se kterým nemusí být seznámen.

Jednoduchost použíti, možnost rozšíření a spousta užitečných funkcí jsou pravě důvodem k tomu proč se OpenPonk využívá jak v rámci této bakalářské práce tak i v rámci celého projektu NBDA.

Část II

Praktická část

Kapitola 8
Shrnutí

Praktická část této bakalářské práce byla vykonávaná v rámci meta-modelovacího týmu projektu NBDA pod vedením doc. Ing. Roberta Pergla, Ph.D. a dohledem Bc. Jany Martínkové a Bc. Terezy Machačové. Cílem této práce bylo ujmout se datových sad z klíčových domén a zařídit jejích sémantickou interoperabilitu v rámci datové platformy projektu NBDA. Datové sady pro práci byly poskytnuté přímo společnosti Remmark, a.s., ale pocházeli od různých zprostředkovatelů (samotný Remmark, a.s., Český statistický úřad atd.).

Zařízeni sémantické interoperability probíhá pomocí tvorby ontologických konceptuálních modelu a jejích následné propojení s poskytnutou datovou sadou. Tento proces pomůže přidat k datum důležité kontextové a doménové informace, sjednotit sémantiku všech modelu a tedy realizovat situaci kde bude umožněno propojení údajů z různých domén a datových sad uvnitř datové platformy.

Postup práce, který vede k zařízení sémantické interoperability v tomto projektu, můžeme rozdělit na dvě primární fáze:

1. Ontologická analýza domény a tvorba konceptuálního modelu

   První část tohoto kroku spočívá v důkladné a precizní analýze domény, ze které pochází datová sada. Její cílem je odhalení důležitých entit v rámci domény a vztahu mezi nimi. Umožňuje také určit přesnou úroveň abstrakce pro dostatečné vyjádřeni smyslu dat z odpovídající datové sady. Navíc pomáhá určit základní koncepty, na kterých je postavená cela datová sada, což tvoří bázi pro další krok této fáze.

   Konceptuální modelování je postaveno na informacích získaných analýzou. Cílem je vytvořit odpovídající model zpracované domény. Pro tento krok je obzvlášť důležité dodržení stejné terminologie a stereo-typizace jako u modelu jíž obsažených v datové platformě, jelikož pravě to umožní vzájemné propojení všech modelu a tedy i dát se kterými jsou modely spojené.

2. Transformace konceptuálního modelu na datový model

   V tomto kroku se zařídí propojení datové sady a konceptuálního modelu. Do modelu se přidají tak zvané datové entity a propojí se s již existující konceptualizaci. Následně se zformuji mapovací pravidla, které umožňují přesnější propojení jednotlivých atributu datové sady a ontologickými entity konceptuálního modelu.

Pro potřeby této práce byl využit jazyk OntoUML, který díky nabízeným konstruktům poskytuje možnost opravdu přesné konceptualizace pojmu z příslušných domén. Modely byly vytvářené v platformě OpenPonk. Tento nástroj byl vybrán z důvodu jeho plné podpory OntoUML, flexibility z hlediska možností rozšíření této platformy a nástrojů, které proces modelování značně zlehčují(např. automatické vyhledávaní chybných OntoUML konstruktu a antipatternu).

Celkem byly v rámci této práce zpracované dvě datové sady, ze kterých vznikly dva ontologické datové modely a jedna šablona.

Kapitola 9

Ontologická analýza domény a tvorba konceptuálního modelu

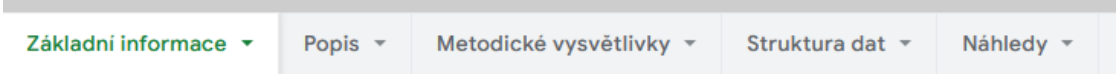
Ontologická analýza v rámci tohoto projektu začínala zkoumáním datové sady a identifikaci domény, do které dána datová sada patři. Během tohoto procesu dostáváme nejdůležitější informace pro následné konceptuální modelování. Umožňuje přesně určit oblast reality, kterou se budeme zabývat a odhaluje základní pro konkretní datovou sadu koncepty. Tyto koncepty se pák stávají základem konceptuálního modelů a umožňují určit jak do hloubky budeme muset danou doménu popsat pro přesné vyjádřeni smyslu dát. Ovšem to celé by bylo velmi obtížné bez důkladné popsané datové sady. Proto bych se rád na začátek zaměřil na to jak datové sady vypadají a jak se využívají pro tvorbu ontologický správných konceptuálních modelu.

9.1 Datové sady

Pro členy modelovacího týmu práce s datovou sadou začíná výběrem datové sady ke zpracování. Úložiště, ke kterému mají přistup všechny členy modelovacího týmu, obsahuje množinu datových sad, které jsou připravené k analýze. U každé datové sady je taktéž uvedena informace o očekávaném datu zpracování a obtížnosti datové sady, podle které se určuje priorita výběru.

Jak jsem zmiňoval dříve práce s nepopsanou datovou sadou, která by obsahovala pouze ”raw”data, by vedla ke značnému stižení analýzy a modelovacího procesu a znemožnění vytvoření přesné konceptualizace. Proto práce modelovacího týmu navazuje na výsledky jiného týmu v rámci NBDA, který je zodpovědný za zařízeni syntaktické interoperability poskytnutých dát. Jejích práce spočívá v přesném popisu datových sad. Tyto popisy obsahují základní informace o doméně, popis obsahu sady, metodické vysvětlivky, strukturu dát a náhled toho jak data vypadají. Navíc každý popis obsahuje jedinečný identifikátor sady, který bude v rámci datové platformy tuto sadu označovat.

Popis sady obvykle sloužil k obecnému určení domény, o kterou se bude jednat. Na tuto informaci přímo navazují metodické vysvětlivky, které často obsahují víc vědomosti o konkretní problematice, poskytují kontext pro datovou sadu a přibližují smysl a důležitost atributu datové sady. To dává dobrý podklad pro určení případných problematických konceptu, které budou



Obrázek 9.1 Kategorie obsažené v popisu datových sád

Název	Datový typ	Popis	Formát
idhod	unikátní identifikátor údaje Veřejné databáze ČSÚ	využije se v případě dotazu ke konkrétnímu údaji	numerický
hodnota	zjištěná hodnota	v numerickém formátu	numerický
stapro_kod	kód statistické proměnné	ze systému SMS UKAZ	numerický
mj_cis	kód číselníku pro měřicí jednotky	v této DS číselník 78	numerický

Obrázek 9.2 Příklad struktury dat v popisu sady

vyžadovat důkladnější analýzu. Primárním zdrojem dát popisu datové sady pro tvorbu konceptuálního modelu je část popisující strukturu dat. Táto část obsahuje názvy atributu datové sady, jejích datový typ, význam a formát ve kterém jsou uložené. Z této části jsem během práce získával jak potřebnou míru abstrakce konceptualizace pro dostatečné vyjádřeni smyslu dát, tak i identifikaci klíčových entit, které budou tvořit základ modelu. Informace o názvech atributu a jejích formátu se navíc využívají během tvorby datového modelu a to hlavně při jejich mapování na hotový konceptuální model.

I když popis datové sady a datová sada samotná je důležitým prvkem pro vytvoření dobré konceptualizace, nesmíme spoléhat během analýzy pouze na ní. Neposkytuje totiž obecnější informace o doméně, nezabývá se reálnými aspekty, které se netykají přímo datové sady a často neobsahuje úplné údaje potřebné pro porozumění konceptu v rámci domény.

9.2 Analýza mimo popis datové sady

Analýza datové sady slouží primárně k vytvoření základního přehledu o zpracovávaných doménách, identifikaci klíčových konceptu a určení potřebné úrovně abstrakce. Pro vytvoření ontologický správného konceptuálního modelu potřebujeme ale víc informaci. U jednotlivých pojmu jak v rámci datové sady tak i v rámci domény často potřebujeme znát jejích přesnou definic a princip fungováni. U některých prvků reality musíme znát jejích vymezení z hlediska zákonu a u jiných potřebujeme vědět jak na tyto prvky nahlíží experty příslušných domén. To vše je potřebné aby jsme měli nejdetailnější pochopení používaných konceptu a tedy zabránily jejích nesprávnému či nejednoznačnému zobrazeni v rámci modelu.

V tomto kroků se pracuje s opravdu věrohodnými zdroji nebo doménovými experty. Veškerá získaná informace musí být kritický zhodnocena, zda je opravdu validní. Není dobré během analýzy spoléhat pouze na vlastní zkušenosti a znalosti. Nedostatečná důkladnost během tohoto kroku může mít za následky špatné navržení konceptuálního modelu, které se nemusí hned objevit. Taková chyba se pák může propagovat do dalších modelu z dané domény, ve kterých z důvodu sémantické interoperability byla dodržena stejná sémantika. Takový problém může vyvrcholit nutností předělaní všech vytvořených modelu, které se zakládali na původním chybném modelu.

9.3 Tvorba konceptuálního modelu

Je důležité říct, že všechny tři zmíněné fáze jsou úzce spjaté a tedy probíhají najednou. Po zahájení práce nad modelem můžeme objevit segmenty modelu, o kterých potřebujeme zjistit víc pro důkladné vystižení jejích sémantiky nebo během analýzy mužem průběžné znázorňovat získané informace do modelu. Tedy by jsme se neměli koukat na tvorbu konceptuálního modelů a analýzu jako na rozdělené fáze, ale jako na něco co na sebe cyklicky navazuje.

Konceptuální modelování datové sady a její domény začíná identifikaci základních entit, které budou tvořit kotvu pro zbytek modelu. Přistup určení těchto entit můžeme rozdělit do dvou skupin - přistup založený na doménovém zkoumání a přistup založený na atributech datové sady. Přistup založený na doménovém zkoumání se zaměřuje na odhalení globálně významných konceptu v rámci zkoumané domény. Jsou to nejčastěji koncepty, které dodávají smysl a kontext

všemu co v dané doméně existuje. Často se jedna o obecné koncepty jako Osoba, Území nebo, v případe domén tykajících se terciárního sektoru, Služba či Zboží.

Přistup založený na atributech datové sady se vyznačuje důkladnějším zkoumáním konkretní problematiky, které se datová sada tyká, místo zaměření na celou doménu. Spočívá v analýze atributu obsažených v datové sadě a modelování entit, ke kterým tyto atributy patři. Například pokud datová sada obsahuje atribut Pohlaví, tak náš model zajisté bude obsahovat entity Muž a Žena nebo pokud existují atributy tykající se formy a druhu studia, tak Forma studia a Druh studia se stávají dobrými kandidáty pro to stát se entitou našeho modelu.

Osobně jsem používal oba dva přístupy najednou, protože to pomáhá maximalizovat počáteční počet entit a připadne množství menších domén, které mužů hned začít prozkoumávat. Názvy entit totiž můžou sloužit jako klíčová slova o kterých budeme vyhledávat více informaci během analýzy. Díky existenci velkého množství potřebných entit, tento postup nám umožní relativně rychlé v rámci modelováni určit úroveň potřebné abstrakce modelu. Během modelování vždy chceme zachytit dostatečnou část domény pro vyjádřeni smyslu dat ale nechceme do modelů přidávat zbytečné pojmy, které sice figurují v doméně ale nepřidávají modelu přesnost. Špatné určení míry abstrakce může vést k velkému nabobtnání modelu a ztráty přehledu o tom co je v dané konceptualizaci opravdu důležité.

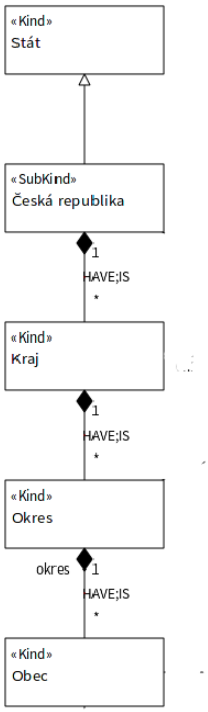
Dalším krokem po získáni klíčových entit se musíme zaměřit na zpřesnění jejích významu a vztahu v rámci modelu. Snažíme se získat víc informaci o klíčových entitách a následně na základě těchto informaci definovat další entity, které budou s původní entitou spojené. Jelikož během tohoto kroku modelování je velká pravděpodobnost, že narazíme na entity, které jsou jíž obsažené v jiných modelech, vitální rolí pro úspěšné dokončení této části konceptualizace hraje tak zvaný Index. Index je dokument udržovaný modelovacím týmem, který obsahuje názvy a stereotypy všech entit, které existují v datové platformě. Navíc u každé entity je udržována informace o tom z jakého modelu pochází. Tento dokument umožňuje se opírat na již vytvořené modely během práce a slouží taktéž pro kontrolu kolizi mezi modelem, který vytváříme a ostatními modely v rámci platformy. Existence a dostupnost těchto informaci umožňuje zabránit tvorbě kolizi, zjednodušit proces modelování a zkontrolovat splnění sémantické interoperability ještě během tvorby modelu.

V rámci tohoto kroku se navíc snažíme přiřadit stereotypy pro vytvořené shluky entit. Postupujeme zde podle doporučení ohledně modelování v OntoUML. Dle těchto doporučení se nejdřív identifikují rigidní Sortaly. Obvykle se jedná o entity, které jsme odhalili v rámci prozkoumání domény. Následně se identifikují anti-rigidní Sortaly, jako je Role nebo Phase, které jsou často napojené na jíž odhalené rigidní Sortaly. Posledními na řadu jsou non-Sortaly, které popisují vlastností sílené skupinou objektu a aspekty, které se zaměřují na interní vlastností jednotlivých entit.

Tímto nám vzniknou popsané a propojené shluky entit, které pro vytvoření korektního modelu stačí propojit mezi sebou. Obvykle během tohoto kroku jíž máme dobrou představu o tom jaký význam mají jednotlivé vzniklé shluky. Pomocí těchto znalostí se snažíme vymyslet sémantickou cestu tvořenou entity a relaci, která jednotlivé shluky propojí. Po propojení entit musíme zkontrolovat zda opravdu máme dostatečně popsanou doménu aby jsme v ní reprezentovali všechny atributy datové sady a zdá všechny koncepty v rámci modelů jsou dostatečně sémanticky vysvětlené. Pokud tomu tak je, máme konceptuální model, který je připravený k jeho proměně do modelu datového.

9.4 Praktický příklad

Popsaný postup analýzy a problému s ní spojených předvedu na datové sadě Školy a školská zařízení se speciálním identifikátorem skoly-a-skolska-zarizeni. Táto datová sada v sobě obsahuje informace o počtu školských zařízení v předškolním, základním a středním vzdělávání (mimo vysokoškolského), počtu žáků, studentů a tříd, a to za kraje České republiky od roku 2006.

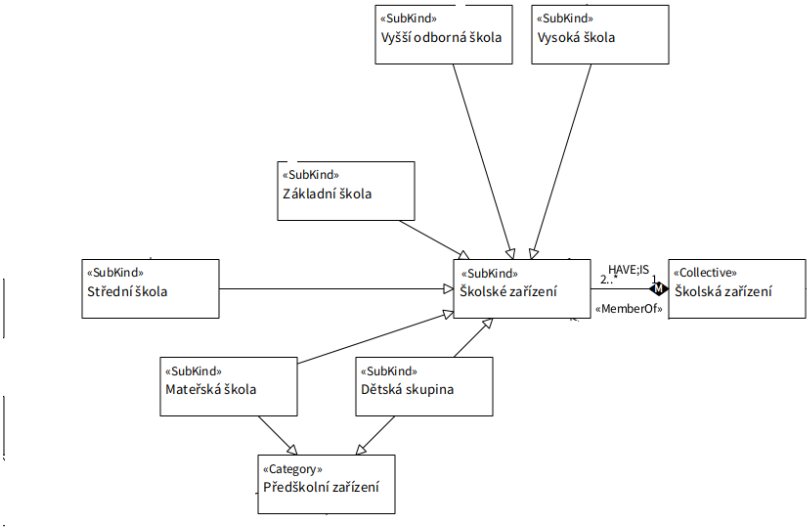


Obrázek 9.3 Vznikla územní kostra

Proces začínal analýzou datové sady a její domény, ze které se získaly entity tvořící základ modelu. Jíž z uvedeného popisu se dá poznat, že nás bude zajímat entita Školské zařízení, Studium, Žák a Student. Navíc přítomnost potřeby identifikace kraje a času vede na entitu Kraj a Referenční období. Na základě těchto entit jsem vykonal hloubější analýzu, která měla za cíl zjistit podstatu těchto pojmu a vyjádřit je v rámci modelu. Nejdříve jsem se soustředil na vytvoření kontextu pro entitu kraj. Jelikož doména území je hojně využívaná v rámci ostatních modelu měl jsem možnost se opírat na předchozí provedené analýzy. Hojnost využíti této domény ale taktéž znamenala, že jsem musel být obzvlášť opatrný s definováním konceptu abych se vyhnul kolizím. Kraj je z definice vyšší územní samosprávný celek. Kraj je po celou dobu své existence krajem a tedy ho mužem klasifikovat jako rigidní. Navíc z definice jasně poskytuje svým instancím princip identity, tedy se nejspíše jedná o Kind. Kraj je tvořen z okresu a v daném kontextu kráje tvoří Českou Republiku, tedy vytvoříme nové entity a propojíme je pomocí vztahu celek-část bez možnosti sdělitelností. Tímto rozšiřováním a navázáním dalších entit získáme územní kostru, která popisuje uzemní konstrukty a tvoří první shluk entit, v našem modelu.

Podobnou analýzou se vytvoří další shluky kolem mnou získaných počátečních entit. V tuto chvíli následuje další krok a to tedy propojení těchto shluku. Zde jsem plně spoléhal na získané znalosti o doméně aby jsem identifikoval entity, které sdílí nějaký vztah v rámci shluku. V případe, že nastane situace kdy entity v rámci shluku nemají nic společného a tedy nemůžou být propojené pomocí asociace či generalizace, vracel jsem se znovu k analýze, protože je to známkou toho, že mi v modelu ještě nějaká entita chyběla. Jako příklad toho můžu uvést shluk, který vznikl kolem entity Školské zařízeni a výše zmíněnou územní kostrou, kde problém jejích propojení byl vyřešen dalším štěpením jednotlivých územních konceptu než jsem se dostál k vytvoření entity Budova. Táto entita se pák dala propojit pomocí generalizace s Vzdělávacím zařízením a tedy i celým shlukem do kterého patři.

Pomocí postupné aplikace podobných úvah a konstantní analýzy se vytvoří finální kon-



Obrázek 9.4 Shluk entit vzniklý kolem entity Školská zařízeni

ceptuální model.

9.5 Ontologické problémy během konceptuálního modelováni a finalizace modelu

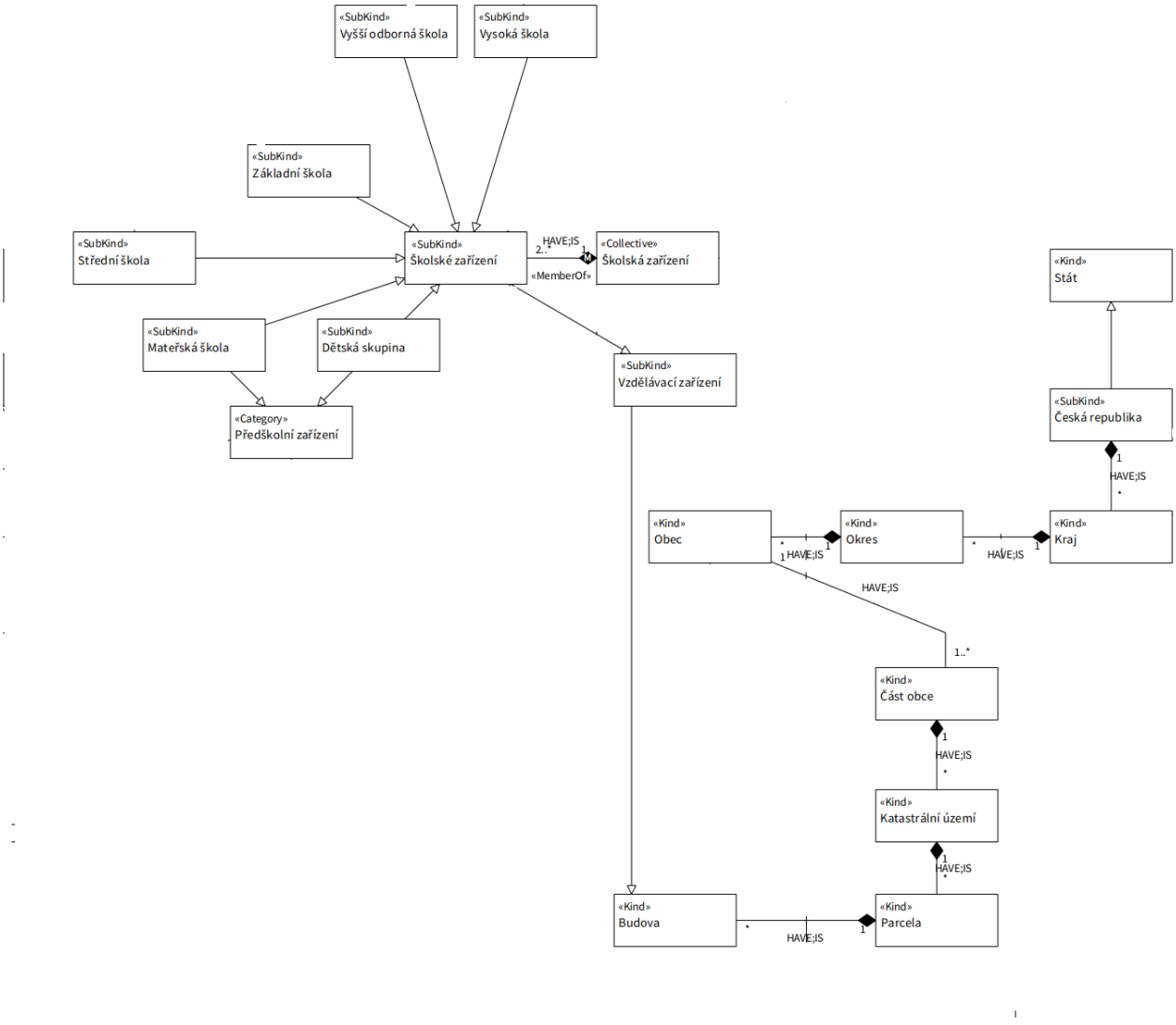
Proces tvorby konceptuálního modelu je ve své podstatě velmi přímočarý. V závislosti na zpracovávané doméně či datové sade ale může skrývat spoustu nečekaných problému. Tyto problémy ústí jak z nedostatečné analýzy tak i z nepřesností popisu datových sát. Zaměřím se tedy na popis nejdůležitějších z těch, se kterými jsem se v rámci modelováni potkal.

9.5.1 Kolize mezi modely

Prvním velkým problémem se kterým jsem se během v rámci tvorby konceptuálních modelů pro daný projekt potkal bylo objevení kolizí s předchozím modelem. Během analýzy datové sady skoly-a-skolska-zarizeni jednou z prvních základních entit, která vznikla byla entita Student. Během následného použití Indexu pro kontrolu kolizi, byla odhalena kolize s modelem MML-vzdelani. Kolize spočívala v nedostatečné přesném popsání domény vzdělaní. Entita Student v rámci MML-vzdelani modelu měla stereotyp Role, což odpovídá realitě jelikož koncept studenta je rigidní a vzniká na základě vnější změny(zahájeni studia). Problém spočíval v tom, že Student byl pomocí stereotypu Relator propojen s entitou Škola, což vede k znepřesnění konceptu studenta. Podle vyobrazené definici by stačilo zahájeni studia na libovolné škole pro získaní statusu studenta. Kolizi jde opravit upřesněním definice studenta.

9.5.2 Šablony a vyjádřeni rozdílu mezi žákem a studentem

Pro pochopení dané problematiky je důležité představit koncept šablon a jejích využití.



Obrázek 9.5 Rozšířena uzemní kostra následně propojena se shlukem okolo Školského zařízeni

9.5.2.1 Šablona

Šablona je konceptuální model, který nepatří ke konkretní datové sadě a tedy neobsahuje datové entity ani mapování. Je v nich stanovené řešení konkretní problematiky, u které víme, že se bude opakovat v rámci více modelu. Z hlediska OntoUML je šablona obyčejným konceptuálním modelem.

V projektu NBDA šablony slouží pro následné ulehčeni modelování a analýzy pří práci s doménou, ze které šablona pochází. Poskytují jasné řešení problému, který zobrazují. Tato vlastnost zabraňuje vzniku kolizi během opakované práce s daným problémem.

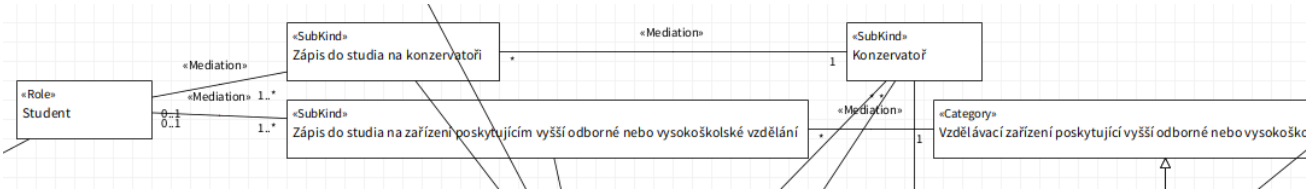
Šablony navíc slouží k propojení datových sad. Často je potřeba propojit dvě nebo více datových sad, které se zabývají příbuznými koncepty. Entity, které je potřeba vytvořit pro jejích propojeni jsou ale nad rámec konceptualizace každé sady a při jejích přidaní vzniklý model ztrácí přehlednost. Šablona v tomto případe může znázornit pouze část potřebnou k propojení modelu a pomocí funkcionality datové platformy, která spojuje modely na základě jmen entit a jejích stereotypu dostaneme modely, které již budou propojené. Další výhodou je to, že další modely, které budou využívat entity obsažené v šabloně budou taktéž přes šablonu propojené se zbytkem modelu.

9.5.2.2 Sémanticky rozdíl mezi žákem a studentem

V rámci objevení minulé kolize a diskuze o důležitosti domény vzdělání,vznikla potřeba propojit již namodelované typy vzdělaní s mnou namodelovanými druhy škol. Pro tento krok se vybralo řešení pomocí šablony. Tato šablona by měla obsahovat korektní definici žáka, studenta a škol spolu s vzděláním, které poskytují. Tato šablona byla mnou vytvořena na základě informaci získaných během modelovaní datové sady Škol a školských zařízení. Pro vzniklý problém je důležité přiblížit použitou definici studenta a žáka. V rámci této šablony a modelu skoly-a-skolska-zarizení student byl definován svým vztahem mediace s zařízením poskytujícím vysokoškolské nebo vyšší odborné vzdělaní. Žák byl naopak definován svou mediaci s zařízením poskytujícím střední a základní vzdělaní. Tedy člověk studující na vysoké či vyšší odborné škole je studentem a člověk studující na základní či střední škole je žákem. Navržené řešení prošlo kontrolou a bylo připraveno k přidaní do datové platformy. Jelikož pojmy student a žák nemají konkretní definici, která by se hodila pro danou konceptualizaci, byl tento rozdíl mnou vydedukován z nalezených zdrojů. Pravě proto jsem se rozhodl se ještě jednou ujistit, zda moje řešení je opravdu korektní. Rozhodl jsem se pro použíti nejmocnějšího v podobném případě zdroje informaci - doménového experta. Při diskuzi s ním jsme našli koncept, který nezapadá do mnou znázorněné definice studenta. Konzervatoř totiž poskytuje středoškolské vzdělaní ale lidí jenž studují na konzervatoři mají status studenta. Tento objev vedl k předělání šablony znázorněni studenta jako člověka, který studuje zařízeni poskytující vysokoškolské či vyšší odborné vzdělaní a nebo studuje na konzervatoří. Důležité zmínit, že uvedené řešení jak zobrazit tuto definici nesplňuje zcela pravidla OntoUML. Vznikla entita Student se stereotypem Role má totiž spojení se dvěma entitami Relator najednou, což je podle pravidel OntoUML antipatternem. V rámci této práce se ale existence nezávažných anti-patternu ignoruje. Tato skutečnost umožňuje prioritizovát přesné vyjádřeni reality proti potenciální neshody s existujícími pravidly. Navíc řešení antipatternu může způsobit vytvoření nového antipatternu, což povede k velkému nabobtnání modelu.

9.5.3 Sjednocení modelů

S finálním problémem, který popíšu, jsem se setkal během práce na modelu Naděje dožití v okresech a správních obvodech ORP. Tento problém je úzce spjatý s problémem heterogenity datových sad v rámci projektu NBDA. Poskytnuté datové sady mají jiné zprostředkovatelé, které nemají sjednocenou terminologii v rámci zpřístupněných datových sad. Táto skutečnost vedla k



Obrázek 9.6 Definice studenta podle nově vzniklé šablony

duplikaci informaci. Datová platforma totiž jíž měla obsažený model Střední délka života. Obě datové sady, které tyto modely reprezentovali, pocházeli od jiných poskytovatelů ale obsahovali stejná data. Rozdíl použité terminologie ztížil identifikaci duplikace informaci pomoci Indexu. Na schodu obsažených informaci se přišlo až během diskuze ohledně modelu. Jelikož model Naděje dožíti v okresech a správních obvodech ORP byl rozšířením již uloženého modelu v rámci platformy, tento problém se tedy vyřešil sjednocením dvou modelu.

Kapitola 10
Datový model

Datový model je rozšířením konceptuálního modelu o tak zvané datové entity a mapovací pravidla. Toto rozšířeni slouží pro provázaní obsahu datové sady a námi vytvořeného konceptuálního modelu. Toho se dosáhne pomocí provázaní atributu datové sady s jednotlivými ontologickými entity v rámci modelu. Provázaní můžeme rozdělit do dvou fázi - vytvoření data entit a využití mapovacích pravidel.

10.1 Data entity

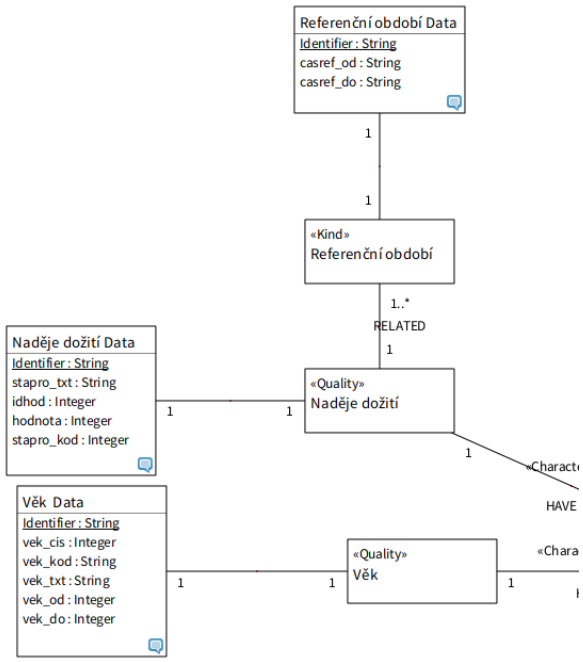
Data entita vypadá jako obyčejná třída v rámci modelu. Vyznačuje se absenci stereotypu a klíčovým slovem Data v názvu. Tato entita slouží jako kontejner pro atributy z datové sady. Atributy se dosadí do Data entity podle jejích významové provázanosti a pak se samotná Data entita propojí pomocí vazby asociace s jednou či více ontologickými entity. Určení se kterou entitou se Data entita propojí se děla na základe sémantické spojitosti atributu Data entity a konceptuálního modelu. Naší snahou je zde určit takové propojení aby sémantická spojitost byla co největší. U každého atributu obsaženého v Data entitě uvádíme jeho přesný název z datové sady a jeho datový typ. Dalším pravidlem je, že pro správné propojení datové sady žádný atribut se nesmí v rámci všech Data entit modelu opakovat.

Tvorbu Data entit předvedu na konceptuálním modelu nadějí dožití.

10.1.1 Praktický příklad

Nejdříve v rámci zpracování této datové sady jsem seskupil atributy, které jasně odkazují na již existující entity v rámci modelu. Mezi tyto uskupení patři atributy, které se přímo týkají konkretních hodnot nadějí dožití, atributy, které reprezentují věkovou skupinu člověka a atributy, které vypovídají o referenčním období, ke kterému informace v dané sadě patří. Tyto uskupení jsem hned vložil do Data entit a propojil s odpovídajícími ontologickými entity.

Zajímavější je situace s atributy pohlaví_kod, pohlaví_cis a pohlaví_text. Tyto atributy jednoznačně patří sémanticky k sobě, takže jsem vytvořil odpovídající pro ně Data entitu. Problém nastává s propojením této datové entity a modelu, jelikož v tomto případě již nemáme jednoznačnou ontologickou entitu Pohlaví. Musíme tedy se rozhodnout ke které z entit obsažených v modelu jsou dané atributy sémanticky nejblíž. Zde se musíme rozhodnout zda napojíme Data entitu na entitu Osoba nebo na entity Muž a Žena. Atributy dané Data entity sice mají ontologickou blízkost s entitou Osoba ale jsou o dost blíže k entitami Muž a Žena, jelikož přímo mluví o konkretních pohlavích. Zároveň ale nemůžeme vytvořit dvě Data entity, které by jsme odděleně propojili s entitou Muž a entitou Žena, jelikož by vznikli dvě entity obsahující stejné atributy.



Obrázek 10.1 Vazby Data entit na jednu ontologickou entitu

Data entitu tedy propojíme zároveň s entitou Muž a Žena. Podobně se provede spojení i Data entity obsahující atributy týkající se území a entity Okres a Obec s rozšířenou působností. Navíc v tomto kroků přidat vytvořených Data entit tak zvaný Identifier. Jedná se o atribut, který popisuje z jaké sady dané atributy pochází a musí být obsažen v každé Data entitě. Tímto máme hotovou tvorbu data entit. V tuto chvíli, díky posledním dvou propojením, ale máme zavedenou nejednoznačnost do provázanosti atributu a ontologických entit. Táto nejednoznačnost se řeší definici mapovacích pravidel.

10.2 Mapovací pravidla

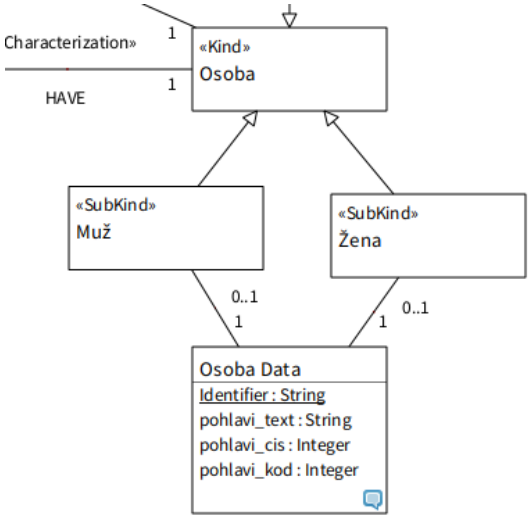
Pro vyřešení nejednoznačností při použiti Data entit a vytvoření přesnější provázanosti mezi modely a datovými sady byly zavedené mapovací pravidla. Mapovací pravidla využívají jednoduchý jazyk navržený modelovacím týmem pro účely projektu NBDA. Umožňují vytvořit přesnou definic pro vazby mezi atributy Data entit a ontologickými entity modelu. Pravidla se zapisují přímo do popisu jednotlivých Data entit.

Pravidla se rozdělují na dvě části. Zleva je část, která vyjadřuje atributy a zprava je část, která vyjadřuje entity modelu. Obě části jsou od sebe oddělené pomocí šípky.

```
"Data" -> "Cíl"
```

Takovéto jednoduché pravidlo vyjadřuje, že obsah atributu Data z datové sady se bude přímo mapovat na entitu modelu Cíl.

Pro potřeby korektního propojení dát a modelu potřebujeme definovat i složitější pravidla



Obrázek 10.2 Vazba jedné Data entity na dvě ontologické entity

10.2.1 Pravidla s podmínkou

Jedna se o rozšíření základního pravidla. Umožňuje se rozhodnout kam se data přiřadí na základě jejích obsahu.

```
"Data" = "X" -> "Cíl"
```

Toto pravidlo znamená, že pokud atribut data nabývá hodnoty X tak se prováže s entitou Cíl. X v tomto případě může nabývat jak číselné hodnoty, tak i textového řetězce a regulárního výrazu.

Pravidla navíc můžou obsahovat logické spojky AND a OR pro formulací složitějších pravidel.

```
"Data" = "X" AND "Data2" = "Y" -> "Cíl"
```

10.2.2 Podmíněna pravidla

Dalším rozšířením základního pravidla. Jedna se o pravidla, která umožňují podle hodnoty jednoho atributu určit kam se bude mapovat Jíny atribut.

```
("Data" = "X") -> ("Y" -> "Cíl")
```

Toto pravidlo znamená, že pokud hodnota atributu Data se rovná X, pák atribut Y se bude propojovat s entitou Cíl.

10.2.3 Praktický příklad

Vraťme se k modelu nadějí dožití, který jsme doplnili o Data entity. V tuto chvíli mužů přidáním mapovacích pravidel ukončit proces přeměny konceptuálního modelu na model datový. Nejdřív jsem doplnil pravidla pro Data entity propojené pouze s jednou entitou. V takovém případě se

totiž využívají jednoduchá mapovací pravidla, která mapují všechny atributy z Data entity na s ní spojený element ontologického modelu.

Výpis kódu 10.1 Příklad jednoduchého mapování mezi atributy Data entity a entitou Referenční období

```
   "casref_do" -> "Referencni obdobi"
   "casref_od" -> "Referencni obdobi"
```

Dále je na řadě mapování u Data entity spojené s entitou Muž a entitou Žena. Zde jsem využil pravidel s podmínkou,znalosti struktury dát daných atributu a podmíněných pravidel. Jelikož pohlaví_kód obsahuje hodnotu 1 pro může a hodnotu 2 pro ženu, můžeme toho využit v podmínce aby jsme vytvořili jednoznačné mapovaní. Pro tuto Data entitu bude mapování vypadat takto.

Výpis kódu 10.2 Mapování Data entity Osoba na entity Muž a Žena

```
"pohlavi_kod" = 1 -> "Muz";
("pohlavi_kod" = 1) -> ("pohlavi_cis" -> "Muz");
("pohlavi_kod" = 1) -> ("pohlavi_text" -> "Muz");
"pohlavi_kod" = 2 -> "Zena";
("pohlavi_kod" = 2) -> ("pohlavi_cis" -> "Zena");
("pohlavi_kod" = 2) -> ("pohlavi_text" -> "Zena");
```

Finálně podobným způsobem se vytvoří mapování u Data entity, která obsahuje atributy tykající se území. Zde atribut vuzemi_cis jednoznačně určuje svou hodnotou jestli se zbylý atributy týkají Okresu nebo Obce s rozšířenou působností. Jejích mapování tedy bude vypadat následovné.

Výpis kódu 10.3 Mapování Data entity Obec Okres Data na entity Obec s rozšířenou pusobnosti a Okres

```
"vuzemi_cis" = 101 -> "Okres";
("vuzemi_cis" = 101) -> ("vuzemi_kod" -> "Okres");
("vuzemi_cis" = 101) -> ("vuzemi_txt" -> "Okres");
"vuzemi_cis" = 65 -> "Obec s rozsirenou pusobnosti";
("vuzemi_cis" = 65) -> ("vuzemi_kod" -> "Obec s rozsirenou pusobnosti");
("vuzemi_cis" = 65) -> ("vuzemi_txt" -> "Obec s rozsirenou pusobnosti");
```

Kapitola 11

Zařízeni sémantické interoperability

Cílem této bakalářské práce bylo zařízení sémantické interoperability mezi datovými sady, které jsou ze své podstaty heterogenní. Sémantická interoperabilita byla v rámci tohoto projektu zařízena pomocí důkladné ontologické analýzy a tvorby ontologických konceptuálních modelu. Díky tomu je umožněné propojení vytvořených během této práce modelů a modelů jíž obsažených v rámci datové platformy projektu NBDA.

Toto propojení probíhá přes společné entity mezi modely. Tedy entity, které mají stejný stereotyp a název. Vzniklý model nadějí dožíti se například propojuje s modelem obyvatelstva podle věkových skupin a modelem zemřelých podle příčin smrti přes hojně se vyskytující entitu Bydliště(viz obrázky 13.113.213.3).

Elementy modelu škol a školských zařízení se například hojně využívají v rámci modelu radio(viz obrázky 13.413.5).

Existence obvyklých a častých entit jako je Osoba či Referenční období v rámci vytvořených modelů zařizuje ještě větší míru jejích propojenosti s ostatními modely.

Navíc, k zařízeni interoperability napomáhá vytvořena šablona, která slouží k propojení domény vzděláni a škol. Jelikož táto šablona mění definicí pojmu student, její prvky se propropagují do všech modelu jíž využívajících tohoto pojmu tak i do všech budoucích modelu z domény školství.

To vše funguje díky udržení jednotného pojmenování a konzistentního určení stereotypu pro stejné koncepty reality, které po jejích objeveni se v modelech umožňují dosáhnout jejích okamžitého propojení.

Kapitola 12
Závěr

Cílem bakalářské práce bylo ontologicky zanalyzovat klíčové domény a propojit je s datovými sadami tak, aby byla umožněna jejich sémantická interoperabilita. Datové sady byly poskytnuté společnosti Remmark, a.s., a zpracované pro využíti v projektu NBDA. Práce probíhala v rámci modelovacího týmu pod vedením doc. Ing. Roberta Pergla, Ph.D. a dohledém Bc. Terezy Machačové a Bc. Jany Martínkové. Její cílem bylo důkladně zanalyzovat patřičné datové sady a s nimi spojené domény a vytvořit konceptuální modely, tak aby po následném propojení dát a vzniklých modelů byla dodržená sémantická interoperabilita v rámci celé datové platformy projektu NBDA.

Během práce byly vytvořené ontologické konceptuální modely klíčových domén, které byly následně propojeny s odpovídajícími sady dat. Pro korektní vytvoření těchto modelu byla použita ontologická analýza příslušných domén, určení klíčových pojmu v rámci těchto domén a odhalení důležitých vztahu mezi nimi. Tyto modely byly vytvořeny v jazyce OntoUML s použitím metaplatformy OpenPonk.

Propojení datových sad a konceptuálních modelu bylo zařízeno pomoci tak zvaných Data entit a mapovacích pravidel, které byly vytvořené modelovacím týmem pro potřeby projektu NBDA. Takto zařízené propojení umožnilo opravdu přesné a precizní propojení mezi daty a koncepty v rámci modelu.

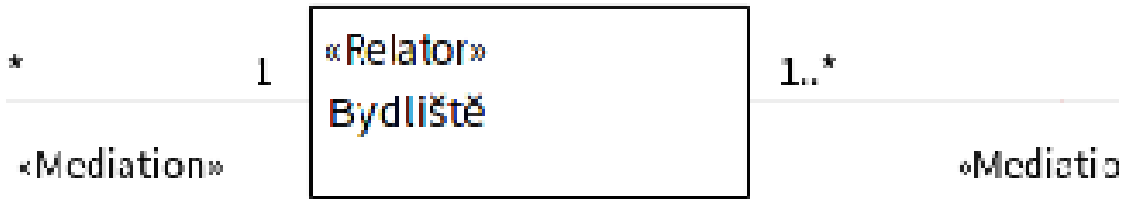
Navíc během práce byla vytvořena modelovací šablona. Šablona je konceptuální model, který nepatří ke konkretní datové sadě a tedy neobsahuje mapování. Slouží k popisu a stanovení řešení určitého problému. Slouží pro následné ulehčeni modelování a analýzy pří práci s doménou, ze které šablona pochází. Navíc poskytuje jasnou příručku na to jak s reprezentovaným problémem vypořádat, čímž zabraňuje vzniku kolizi časté nejednoznačnosti těchto řešení.

Dodržením konvencí v pojmenování entit a důkladnou verifikaci kolizi mezi vytvořenými modely a modely již existujícími v rámci datové platformy bylo dosaženo sémantické interoperability mezi nimi. Díky tomu datová platforma dokáže zakomponovat do sebe nově vzniklé modely a pracovat s nimi jako s jednotným celkem.

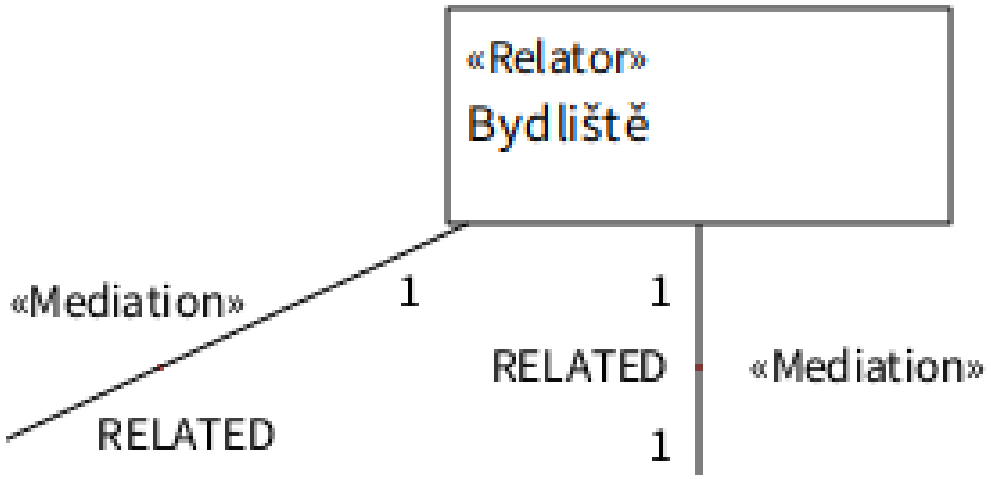
Kapitola 13

Příklady sémantické interoperability

Společné entity, přes které je možné jednotlivé datové sady významově propojit.

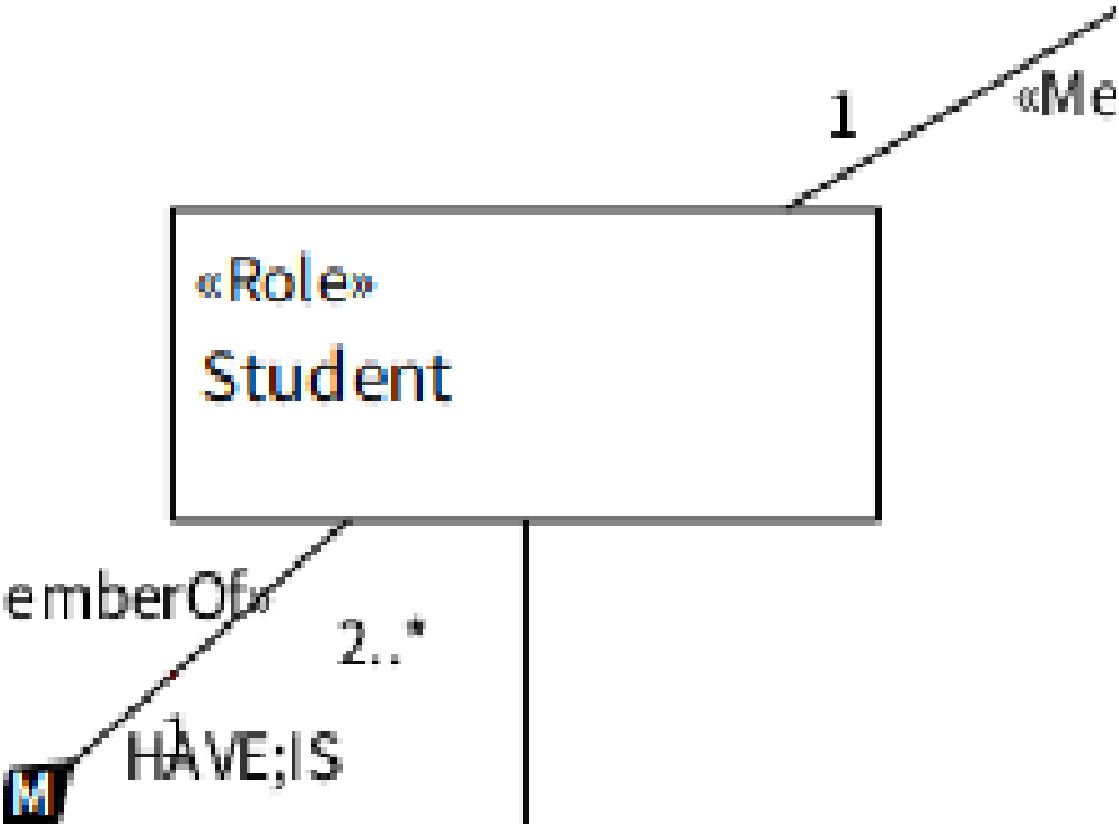


Obrázek 13.1 Zobrazeni entity Bydliště v modelu obyvatelstvo-podle-petiletych-vekovych-skupin-a-pohlavi-v-kraji

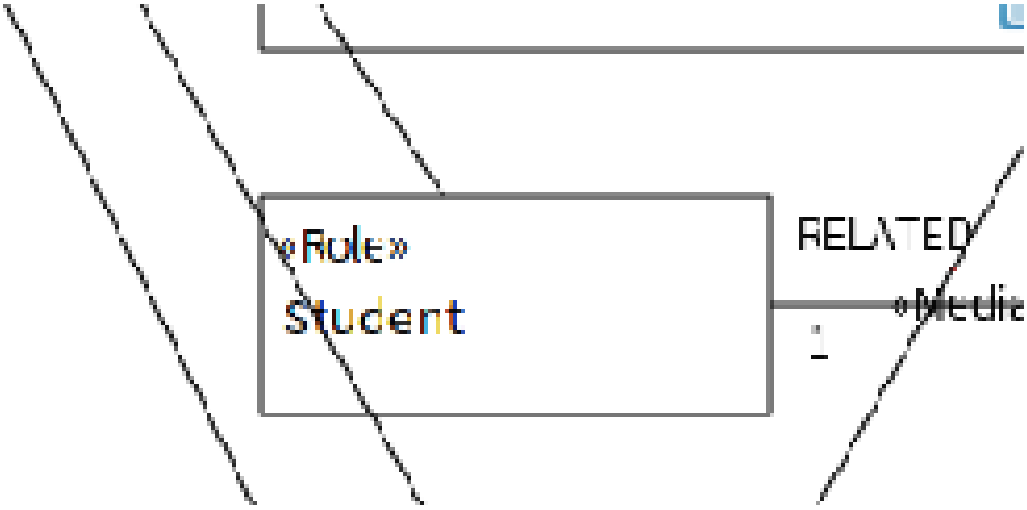


Obrázek 13.2 Zobrazeni entity Bydliště v modelu nadeje-doziti-v-okresech-a-spravnich-obvodech-orp

Obrázek 13.3 Zobrazeni entity Bydliště v modelu zemreli-podle-pricin-smrti



Obrázek 13.4 Zobrazení entity Student v modelu skoly-a-skolska-zarizeni



Obrázek 13.5 Zobrazeni entity Student v modelu radio

Bibliografie

1. WEGNER, Peter. Interoperability. *ACM Computing Surveys (CSUR)*. 1996, roč. 28, č. 1, s. 285–287.
2. GUIZZARDI, Giancarlo. Ontology, ontologies and the "I" of FAIR. *Data Intelligence*. 2020, roč. 2, č. 1-2, s. 181–191.
3. CELBOVÁ, Ludmila. Sémantická interoperabilita. *Information*. 2003. ISSN 2078-2489. Dostupné také z: `https://aleph.nkp.cz/F/?func=direct&doc_number=000000555&local_base=KTD`.
4. REMMARK, a.s. *Remmark* [online] [cit. 2022-06-15]. Dostupné z: `http://www.remmark.cz/`.
5. DATAFAIRPORT. *Data FAIRPort* [online] [cit. 2022-06-15]. Dostupné z: `https://www.datafairport.org/`.
6. LORENTZCENTER. *Jointly designing a data FAIRPORT* [online] [cit. 2022-06-15]. Dostupné z: `https://www.lorentzcenter.nl/jointly-designing-a-data-fairport.html`.
7. GOFAIR. *FAIR Principles* [online] [cit. 2022-06-15]. Dostupné z: `https://www.go-fair.org/fair-principles/`.
8. GOFAIR. *FAIRification Process* [online] [cit. 2022-06-15]. Dostupné z: `https://www.go-fair.org/fair-principles/fairification-process/`.
9. SMITH, Barry. Ontology. In: *The furniture of the world*. Brill, 2012, s. 47–68.
10. GRUBER, Tom. *Ontology*. [Online]. 2018 [cit. 2022-06-17]. Dostupné z: `https://tomgruber.org/writing/ontology-in-encyclopedia-of-dbs.pdf`.
11. GUIZZARDI, Giancarlo. Ontological foundations for structural conceptual models [online]. 2005 [cit. 2022-06-15]. Dostupné z: `https://www.researchgate.net/publication/215697579_Ontological_Foundations_for_Structural_Conceptual_Models`.
12. ČERBA, Otakar. *Ontologie* [online] [cit. 2021-06-16]. Dostupné z: `http://geomatika.kma.zcu.cz/studium/sgg/Materialy/Ontologie.pdf`.
13. PERGL, Robert. *Motivace a úvod do konceptuálního modelování* [online] [cit. 2021-06-17]. Dostupné z: `https://courses.fit.cvut.cz/BI-KOM/slides/html/lectures-czech/01-intro.htm`.
14. OLIVÉ, Antoni. *Conceptual modeling of information systems*. Springer Science & Business Media, 2007.
15. HUŇKA, František; MÁCHA, Ferdinand. Datové modelováni a typováni. [B.r.].

16. VISSER, Eelco. WebDSL: A case study in domain-specific language engineering. In: *International summer school on generative and transformational techniques in software engineering*. 2007, s. 291–373.
17. LOWE, E Jonathan. *The four-category ontology: A metaphysical foundation for natural science*. Clarendon Press, 2005.
18. BACHTER, Nikolai. *Universals and Particulars* [online] [cit. 2021-06-17]. Dostupné z: `https://philnotesblog.wordpress.com/2017/04/02/universals-and-particulars/`.
19. GUIZZARDI, Giancarlo; BENEVIDES, Alessander; FONSECA, Claudenir; PORELLO, Daniele; ALMEIDA, João; PRINCE SALES, Tiago. UFO: Unified Foundational Ontology. *Applied Ontology*. 2022. Dostupné z DOI: `10.3233/AO-210256`.
20. GUIZZARDI, Giancarlo; FALBO, Ricardo; GUIZZARDI, Renata. Grounding Software Domain Ontologies in the Unified Foundational Ontology (UFO): The case of the ODE Software Process Ontology. In: 2008, s. 127–140.
21. OBJECT MANAGEMENT GROUP®,INC. *Introduction To OMG's Unified Modeling Language* [online] [cit. 2022-06-17]. Dostupné z: `https://www.uml.org/what-is-uml.htm`.
22. PILONE, Dan; PITMAN, Neil. *UML 2.0 in a Nutshell*. "O'Reilly Media, Inc.", 2005.
23. "OBJECT MANAGEMENT GROUP®,INC. *Unified Modeling Language* [online]. 2017 [cit. 2022-06-18]. Dostupné z: `https://www.omg.org/spec/UML/2.5.1/PDF`.
24. SUCHÁNEK, Marek. *OntoUML* [online]. 2018 [cit. 2022-06-18]. Dostupné z: `https://ontouml.readthedocs.io/en/latest/intro/ontouml.html`.
25. SUCHÁNEK, Marek. *OntoUML* [online]. 2018 [cit. 2022-06-18]. Dostupné z: `https://ontouml.readthedocs.io/en/latest/theory/individuals.html`.
26. SUCHÁNEK, Marek. *OntoUML* [online]. 2018 [cit. 2022-06-18]. Dostupné z: `https://ontouml.readthedocs.io/en/latest/theory/identity.html`.
27. SUCHÁNEK, Marek. *OntoUML* [online]. 2018 [cit. 2022-06-18]. Dostupné z: `https://ontouml.readthedocs.io/en/latest/theory/rigidity.html`.
28. PERGL, Robert. *OntoUML: Základy, rigidní a anti-rigidní sortály*.
29. PERGL, Robert. *OntoUML: Non-sortály, relace, relátor*.
30. PERGL, Robert. *OntoUML: Vztahy Celek-Část, typy agregací; Aspekty* [online] [cit. 2021-06-19]. Dostupné z: `https://courses.fit.cvut.cz/BI-KOM/slides/html/lectures-czech/04-ontouml-aggr-aspects.htm`.
31. FIT CVUT. *OpenPonk modeling platform* [online]. 2022 [cit. 2022-06-19]. Dostupné z: `https://openponk.org/`.

Obsah přiloženého média

- `readme.txt` ..... stručný popis obsahu média
- `src`
  - `modely` ..... Konceptuální modely
    - `klicove-domeny` ..... modely klíčových domén
    - `sablona` ..... model šablony
  - `text` ..... zdrojová forma práce ve formátu LaTeX
- `bp-Artemij-Danilov-2022.pdf` ..... text práce ve formátu PDF