



ZADÁNÍ DIPLOMOVÉ PRÁCE

Název:	Zavád ní otev ených dat ve státní správ
Student:	Bc. Mat j Kunte
Vedoucí:	Ing. Josef Gattermayer
Studijní program:	Informatika
Studijní obor:	Webové a softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	do konce letního semestru 2015/16

Pokyny pro vypracování

- Nastudujte problematiku otev ených dat.
- Nastudujte možnosti strojového tení r zných formát dat.
- Na p íkladu vytipovaného m sta, které je sv tovým lídrem v otev ených datech, popište proces zve ej ování (s akcentem na použití API).
- Vytipujte na M Praha 8 datasey vhodné pro strojové zve ejn ní.
- Navrh te pro tyto datasey vhodné API.
- Implementujte do asné ešení, které umožní zve ejnit dané datasey pomocí API.
- Specifikujte zadání, dle kterého bude možné vytvo it trvalé ešení pro dané API respektující sou asné IS a procesy.

Seznam odborné literatury

Dodá vedoucí práce.

L.S.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

prof. Ing. Pavel Tvrdík, CSc.
d kan

V Praze dne 2. února 2015

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Diplomová práce

Zavádění otevřených dat ve státní správě

Bc. Matěj Kunte

Vedoucí práce: Ing. Josef Gattermayer

10. května 2016

Poděkování

Chtěl bych poděkovat zejména svým rodičům za podporu, kterou mi neustále poskytují. Také děkuji Ing. Michalovi Valentovi, Ph.D. za zásadní popostrčení správným směrem při psaní této práce. Na závěr děkuji svým přátelům, všem usměvavým lidem a tzv. „Půlnočníkům“.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické 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. § 46 odst. 6 tohoto zákona 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, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 10. května 2016

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2016 Matěj Kunte. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Kunte, Matěj. *Zavádění otevřených dat ve státní správě*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2016.

Abstrakt

Tato práce se zabývá otevřenými daty, jejich formáty, zpracováním, uchováváním a publikací. Nabízí krátké nahlédnutí na otevírání dat v zahraničí a v jednoduché webové aplikaci zobrazuje možnost praktického využití.

Klíčová slova Otevřená data, Linked data, sémantický web, RDF

Abstract

This master's thesis describes Open Data, its formats, processing, storing and publishing. It offers a brief review on publishing Open Data in foreign countries and with a simple web application it renders an example of practical use.

Keywords Open data, Linked data, semantic web, RDF

Obsah

Úvod	1
1 Otevřená data	3
1.1 Co?	3
1.2 Kdo?	5
1.3 Proč?	5
2 Open Government Partnership	7
2.1 Otevřená data ve Spojeném království[1]	9
3 Zvyšování otevřenosti dat	11
3.1 Použité nástroje	11
3.2 Sběrná místa pro odpad a umístění kontejnerů	19
3.3 Základní a mateřské školy	29
3.4 Implementace	42
4 Referenční aplikace	45
4.1 Použité nástroje	45
4.2 Návrh aplikace	47
4.3 Implementace	49
Závěr	53
Literatura	55
A Seznam použitých zkratk	57
B Obsah příloženého CD	59

Seznam obrázků

1.1	5-star scheme[2]	5
3.1	RDF graf[3]	12
3.2	Vzorový RDF model	14
3.3	Ontologie RDF Data Cube[4]	18
3.4	RDF model řešení svozu odpadu	25
3.5	RDF model řešení škol	35
4.1	Případy užití	48
4.2	Komunikace komponent aplikace	51

Seznam tabulek

3.1	Konstrukty RDF schématu[5]	13
3.2	RDF <i>subject</i> : Majitel	24
3.3	RDF <i>subject</i> : Provozovatel	26
3.4	RDF <i>subject</i> : Svoz odpadu	26
3.5	RDF <i>subject</i> : Otevírací doba	27
3.6	RDF <i>subject</i> : Komunikační kanál	27
3.7	RDF <i>subject</i> : Lokace	28
3.8	RDF <i>subject</i> : Typ odpadu	28
3.9	RDF <i>subject</i> : Poštovní adresa	28
3.10	RDF <i>subject</i> : GPS souřadnice	29
3.11	RDF <i>subject</i> : Škola	36
3.12	RDF <i>subject</i> : Adresa školy	36
3.13	RDF <i>subject</i> : Zaměstnanec školy	37
3.14	RDF <i>subject</i> : Kontakt mateřské školy	38
3.15	RDF <i>subject</i> : Lokace školy	38
3.16	RDF <i>subject</i> : GPS souřadnice	38
3.17	RDF <i>subject</i> : Ročník, Školní rok	39
3.18	RDF <i>subject</i> : Měření	39
4.1	Pokrytí požadavků případy užití	50

Úvod

Moderní svět je plný dat. Produkují a sbírají je běžní uživatelé, firmy, vzdělávací zařízení, neziskové organizace i vlády a veřejné správy všech zemí. V poslední době můžeme slyšet o fenoménu otevřených dat. Pokud se ovšem zeptáte běžného uživatele, o významu pojmu *otevřená data* většinou nemá tušení. O co tedy jde? Zde v úvodu své práce se omezím na velmi stručnou definici pojmu:

Otevřená data jsou taková, která je možno použít a redistribuovat bez omezení.[6]

Ve své práci se zabývám problematikou otevírání dat, tedy jejich zveřejňováním ve strojově čitelné formě. Hlavním těžištěm mé práce mělo být otevírání dat v městské části Praha 8. Otevírání dat je mezi politiky oblíbené téma, již při formování zadání jsem ale měl jisté pochybnosti o jeho bezproblémové proveditelnosti. Přestože politická poptávka po otevírání dat je vysoká, strávil můj vedoucí mnoho času v boji s úředníky. Ukázalo se totiž, že úředník naše nadšení pro otevřenost a svobodný přístup k informacím nesdílí. Je to tvor libující si ve starých pořádcích, neochotný vyjít vstříc, snažit se začne až ve chvíli, kdy mu to nařídí nějaká směrnice. Přestože získat přímý přístup k datům městské části, abych mohl data otevírat, se mému vedoucímu nepodařilo, netrvalo ani rok a úředník data otevřel sám. Pro veřejnost bohužel, pro mne bohudík, kvalita zveřejněných dat je bídná, a tak se ve své práci místo otevírání dat mohu věnovat zvyšování jejich kvality a použitelnosti.

Součástí práce je dále referenční aplikace založená na přístupu k otevřeným datům a informace o otevírání dat v zahraničí.

Otevřená data

V této kapitole shrnu nejdůležitější informace o tom, co to jsou otevřená data, kde a jak vznikají, k čemu jsou dobrá, a jak je udržovat a zvyšovat jejich kvalitu.

1.1 Co?

V úvodu jsem zmínil vcelku stručnou definici pojmu *otevřená data*, nezmínil jsem však několik podstatných vlastností dat, aby mohla být otevřená[6]:

- Data musí být úplná a uživateli dostupná při vynaložení minima nákladů, ideálně ke stažení přes internet.
- Data by měla být dostupná ve vyhovující a upravitelné podobě, používající standardy s volně dostupnou specifikací.
- Data musí být poskytována s takovými podmínkami, aby bylo možné jejich znovupoužití a redistribuce, a to včetně jejich rozšíření o další data. Omezení „jen pro nekomerční účely“ není povoleno.
- Data musí být poskytována každému bez rozdílu (pohlaví, rasy, věku apod.)

Tvůrce World Wide Webu a průkopník Linked data Tim Berners-Lee v roce 2010 navrhl systém hodnocení otevřených dat, tzv. „5-star schema“ [2]. Jak název napovídá, systém má pět úrovní, otevřenost dat se hodnotí počtem hvězd, více hvězd znamená lepší formát dat. Obvykle vyšší počet hvězd znamená pro vydavatele dat jejich náročnější tvorbu, ale zároveň i vyšší hodnotu pro případného uživatele dat.

1.1.1 Úroveň *

Pro vytvoření otevřených dat této úrovně jen to, že je nahrajeme v jakémkoli formátu (nemusí být strojově čitelná) na web s licencí umožňující volné využívání. Nejběžnějšími představiteli této úrovně jsou dokumenty ve formátu PDF a naskenované obrázky.

Uživatel může data v tomto formátu bez problémů prohlížet, tisknout, modifikovat, tvořit lokální kopie, používat v dalších systémech apod. V tomto formátu nalezneme nejčastěji zveřejněné smlouvy a dokumenty veřejných institucí. Vzhledem k tomu, že data nejsou strojově čitelná, jsou zajímavá spíše pro běžné uživatele, jejich další využití je problematické.

Výhodou pro vydavatele otevřených dat je jejich nenáročná publikace.

1.1.2 Úroveň **

Do této kategorie patří strukturovaná data publikovaná ve strojově čitelné formě. Nejčastěji jde o data vytvořená v proprietárních tabulkových procesorech, jako je Microsoft Excel.

Výhodou oproti předchozí úrovni je velmi snadná práce s daty, možnost provádění výpočtů a vizualizace výsledků. Přestože není triviální, je možné data zpracovávat strojově a využít je např. v aplikacích.

Publikace je stále velmi jednoduchá, většinou není třeba data nijak výrazně upravovat.

1.1.3 Úroveň ***

Tříhvězďová úroveň vyžaduje využití otevřeného formátu dat. Nejrozšířenější otevřeným formátem je CSV, tedy text oddělený čárkami.

Práce s daty této úrovně je pro běžného uživatele prakticky totožná s předchozí úrovní, naproti tomu strojové zpracování je mnohem jednodušší.

Mnoho vydavatelů otevřených dat svá data interně uchovává v Excelových tabulkách, publikace dat do CSV pro ně už není triviální, přesto ale přináší podstatné zvýšení použitelnosti dat za cenu jen malého úsilí.

1.1.4 Úroveň ****

Požadavkem na data čtyřhvězďové úrovně je použití jednoznačného identifikátoru. Díky tomu je možné odkazovat na data dalších uživatelů. K popisu dat této úrovně se nejčastěji používá formát RDF.

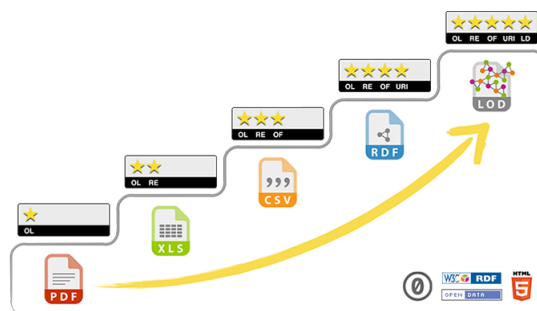
RDF je založen na XML a byl navržen, aby umožnil počítačům rozumět významu dat. Není primárně určen pro čtení a úpravu běžnými uživateli, naopak je velmi vhodný pro další využití v aplikacích a webových službách. Výhodou tohoto formátu je velmi snadná kombinace různých datasetů.

Publikace dat na této úrovni už je komplikovanější, je potřeba vybrat vhodné slovníky a navrhnout strukturu datasetu.

1.1.5 Úroveň ★★★★★

Pětihvězďová data jsou ve stejném formátu jako čtyřhvězďová, liší se tím, že jsou obohacena o odkazy na data v jiných datasetech. Díky vzájemné provázanosti datasetů je snazší zjistit o daném objektu mnohem více informací, na druhou stranu je potřeba počítat s problémy, jako jsou např. nefunkční odkazy.

Publikace dat na této úrovni je podobná jako na předchozí úrovni, navíc je ale potřeba hledat další datové zdroje kvůli prolínání.



Obrázek 1.1: 5-star scheme[2]

1.2 Kdo?

První otevřená data začala publikovat vláda USA v roce 2009. V září 2011 vznikla iniciativa Open Government Partnership, bojující za otevřenost veřejných správ po celém světě[7]. Od jejího založení se k ní připojilo celkem 69 zemí, mezi nimi i Německo, Francie, Česká republika a Spojené království. Poslední jmenovaná se stala světovým lídrem v publikaci otevřených dat.

Otevřená data poskytují nejen veřejné správy, ale i firmy a různé organizace. Mezi nejzajímavější poskytovatele patří:

- Katalog Spojeného království – data.gov.uk
- Katalog USA – data.gov
- Katalog České republiky – portal.gov.cz
- Katalog Hlavního města Prahy – opendata.praha.eu
- DBpedia – dbpedia.org

1.3 Proč?

Státní orgány, orgány územní samosprávy a veřejné instituce jsou podle zákona č. 106/1999 Sb., o svobodném přístupu k informacím povinny poskytovat

případným žadatelům informace vztahující se k jejich působnosti[8]. Příslušný úředník musí každou žádost individuálně posoudit a vyřídit. Vyřízení dotazu trvá dlouho, kapacity úřadů na vyřízení žádostí nebývají vysoké. Otevření dat, tedy jejich zveřejnění s otevřenou licencí se zde jeví jako ideální řešení, neboť úředník již nemusí vyhledávat požadovaná data, k vyřízení žádosti stačí poskytnutí odkazu na webovou stránku s otevřenými daty.

Někdy má zájem o zřízení aplikace pro občany politické vedení úřadu, úřad ale pro její vývoj a správu nemá vlastní kapacity. Otevřená data jsou opět řešením, o vývoj i správu aplikace se postará veřejnost, náklady pro úřad jsou minimální. V katalogu otevřených dat Spojeného království můžeme najít téměř 30 000 datasetů a skoro 400 aplikací, které jsou na přístupu k těmto datasetům postaveny. Mezi příklady takových aplikací patří wheredoesmymoneygo.org a roadworks.org.

Zveřejnění dat zvyšuje transparentnost úřadu, důvěru v úřad, pomáhá odhalovat korupci a jiné nepravosti. V Kanadě existuje případ, kdy zkoumání volně dostupných dat odhalilo daňové úniky ve výši 3,2 miliardy dolarů[9].

Kromě toho má otevírání nemalý ekonomický dopad, britské *The Centre for Economic and Business Research* odhaduje, že jen v oboru Big Data v letech 2012-2017 vznikne ve Spojeném království 58 000 pracovních míst[10].

Open Government Partnership

Iniciativa OGP vznikla, aby podporovala snahy vlád po celém světě zvyšovat jejich transparentnost, odpovědnost a spolupráci s občanskou společností[11].

Aby se mohla země stát členem OGP, musí nejprve splňovat následující kritéria:

- Fiskální transparentnost,
- svobodný přístup k informacím,
- povinnost volených zástupců a vysokých činitelů zveřejňovat majetková priznání,
- možnost zapojení občanské veřejnosti.

Pokud země splňuje stanovená kritéria a je přijata do OGP, dalším krokem je vytvoření tzv. Národního akčního plánu. Národní akční plán je soubor závazků, jejichž splněním se země snaží docílit pěti hlavních cílů OGP:

Zlepšení služeb veřejnosti Cílí na zlepšení služeb, jako je zdravotnictví, vzdělávání, vymahatelnost práva, dodávky vody a elektřiny, telekomunikace a podobné.

Zvýšení veřejné integrity Cílí na veřejnou etiku a boj proti korupci, přístup k informacím, reformu financování kampaní, svobodu médií a občanské společnosti.

Zvýšení efektivity využívání veřejných zdrojů Opatření týkající se veřejných rozpočtů, veřejných zakázek, přírodních zdrojů či dotačních programů.

Zajištění vyšší bezpečnosti společnosti Opatření týkající se bezpečnosti, řešení přírodních hrozeb, reakce na pohromy a krize.

Zvýšení odpovědnosti soukromého sektoru Opatření týkající se ochrany životního prostředí, protikorupční opatření, zapojení veřejnosti.

Každý závazek vůči OGP by měl být zdůvodněn, zhodnocen dopad jeho splnění a měl by mít stanoven termín jeho splnění. Zapojení občanské společnosti je klíčové nejen pro vytvoření Národního akčního plánu, ale i pro jeho následnou implementaci.

Mezi další povinnosti vůči OGP patří:

- Implementace Národního akčního plánu podle stanoveného časového plánu,
- tvorba pravidelných ročních sebehodnotících reportů,
- spolupráce s nezávislými hodnotiteli plnění plánu,
- zapojení do spolupráce s ostatními členy OGP.

Každý Národní akční plán je tvořen na omezené časové období, obvykle dva roky. Aby členové OGP dosáhli většího pokroku, měly by jejich závazky v nových akčních plánech být ambicióznější než v předchozích, neměli by vydávat staré závazky za nové a podobně. To bohužel není případ České republiky, která momentálně implementuje svůj druhý akční plán, který se ale prakticky neliší od prvního. Oba plány zahrnují pouze tři závazky, které navíc nejsou nové, neboť plynou již ze závazků vůči Evropské unii[12]:

1. Přijetí zákona o státní službě,
2. zefektivnění systému svobodného přístupu k informacím,
3. zpřístupnění dat a informací.

Zemí, ze které by si Česká republika mohla vzít příklad, je Spojené království. Aktuální Národní akční plán Spojeného království zahrnuje celkem 21 závazků, předchozí obsahoval dokonce 41 závazků. Oproti tomu českému je podstatně obsáhlejší, konkrétnější a ambicióznější. O ambicích Spojeného království svědčí následující závazek:

Cílem Spojeného království je stát se do roku 2015 nejtransparentnějším sociálním investičním trhem v rámci OGP a G20, a to v souladu se zásadami Charty otevřených dat (Open Data Charter)[12].

2.1 Otevřená data ve Spojeném království[1]

Otevřenost státní správy je jednou z priorit Spojeného království, i proto je vidět její pokrok v otevírání dat. Své otevřenosti dosáhla i díky závazkům vůči OGP, které spočívaly v hledání a analýze všech dostupných datových sad veřejné správy, či přijetí zákonů vyžadující zveřejňování konkrétních dat. Jedním z nich je tzv. *Local Authorities Data Transparency Code*.

Tento zákon všem místním správám Spojeného království ukládá zveřejňovat svá data. Kromě rozsahu dat zákon specifikuje i frekvenci aktualizace dat. Mezi data, zveřejňovaná čtvrtletně, patří zejména data týkající se výdajů:

- Výdaje přesahující £500 – faktury, granty, platby za zboží a služby, půjčky, dobropisy, transakce s ostatními veřejnými institucemi,
- transakce přes Government Procurement Card,
- veřejné zakázky – pozvánky do tendrů přesahujících £5 000, smlouvy, objednávky, rámcové dohody nebo jiné právně vymahatelné smlouvy s hodnotou přesahující £5 000.

Většina zbylých dat je zveřejňována ročně:

- Nemovitý majetek místní správy – prostory využívané místní správou, nemovitý majetek místní správy pronajmutý soukromému sektoru, garáže, nevyužitá pozemky,
- sociální bydlení – počty domovů, tržní hodnoty domů,
- granty dobrovolným a komunitním společnostem,
- organizační schema – nejvyšší tři úrovně organizace
- obchodní unie – počty zaměstnanců věnující se uniím, názvy unií, výdaje na obsluhu unií,
- parkování – příjmy a výdaje na parkovné, pokuty za špatné parkování, parkovací místa
- platy vysokých úředníků – platy a bonusy úředníků, které přesahují £50 000 ročně, úředníci s platem převyšujícím £150 000 ročně musí být uvedeni včetně jména,
- boj proti podvodům – počet vyšetřovatelů, akreditovaných specialistů na boj proti podvodům, vyšetřených případů.

2. OPEN GOVERNMENT PARTNERSHIP

Kromě toho, že zákon specifikuje minimální rozsah dat ke zveřejnění, dále také uvádí řadu doporučení, co publikovat nad rámec minimálního rozsahu. Mezi doporučení patří většinou častější zveřejňování, použití takových kritérií pro výběr dat, aby jich bylo zahrnuto více, či více podrobností o zveřejňovaných položkách. Pro publikaci zákon doporučuje využití alespoň tříhvězdkové úrovně otevřených dat, tedy formátu CSV.

Se zveřejňováním místním správám pomáhá i řada organizací, díky kterým mají správy k dispozici příručky, jak zveřejňovat data podle *Local Authorities Data Transparency Code*, či připravená schemata pro publikaci těchto dat.

Podle nezávislé hodnotící zprávy druhého Národního akčního plánu Spojeného království není tvorba procesů týkajících se zveřejňování dat ani identifikace nezveřejněných datasetů stále dokončena, i přesto se však Spojené království může pochlubit jednou z nejotevřenějších veřejných správ na světě.

Rozdíl mezi Spojeným královstvím a Českou republikou by se dal shrnout velmi stručně...

Když se chce, tak to jde.

Zvyšování otevřenosti dat

Původní název této kapitoly měl znít „Otevírání dat na MÚ Praha 8“. Tak by se stalo, kdyby šlo vše podle plánu, tedy že bych měl přístup přímo k datům na městském úřadě, sám bych mohl identifikovat data vhodná ke zveřejnění a řešil bych i způsob, jakým budou zveřejněna. Přístup jsem nedostal, po roce přišel úřad s otevřenými daty sám.

Mohlo by se zdát, že mou práci tedy udělal někdo jiný a že toho na mne moc nezbylo, ale není tomu tak. Domnívám se, že data, která byla zveřejněna, se prakticky neliší od dat, která úřad uchovává interně. Data jsou nekvalitní, obsahují chyby a spolu s nimi nebyla dodána žádná dokumentace, význam některých údajů tak lze spíše odhadovat.

Mým cílem je data zpracovat, popsat pomocí RDF a povýšit na pětihvězdo-ovou úroveň.

3.1 Použité nástroje

3.1.1 RDF[3, 5]

RDF je nástroj, který slouží k reprezentaci informací o zdrojích (resources) na WWW, včetně informacích o těch, které na WWW mohou být identifikovány, ale nemusí zde být dostupné. Zdrojem může být cokoli, dokumenty, lidé, fyzické objekty, události, abstrakce. Příkladem mohou být základní informace o nějaké firmě, ale také její nabídka produktů či statistické údaje o porodnosti v ČR.

RDF je určeno spíše pro strojové zpracování než prohlížení lidmi, je vhodným prostředkem pro výměnu informací mezi různými aplikacemi, a to i mezi aplikacemi, pro které data původně nebyla vytvořena.

RDF je založeno na rozlišování zdrojů jedinečnými identifikátory (IRI) a jejich popisování pomocí tvrzení (statements). Struktura každého tvrzení je následující:

<subject> <predicate> <object>

3. ZVYŠOVÁNÍ OTEVŘENOSTI DAT

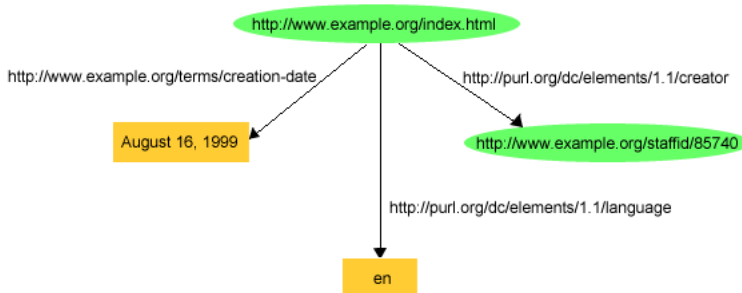
Každé tvrzení popisuje vztah mezi zdrojem *subject* a zdrojem *object*, povahu vztahu reprezentuje *predicate*. Protože každé tvrzení sestává ze tří prvků, nazýváme je *triples*.

Výpis 3.1: Příklad RDF tvrzení

```
<http://www.example.org/index.html>  
<http://www.example.org/terms/creation-date>  
"August 16, 1999" .
```

Tvrzení lze popsat následovně; webová stránka `http://www.example.org/index.html` (*subject*) byla vytvořena (*predicate*) 16. srpna 1999 (*object*).

Množinu tvrzení lze reprezentovat jako orientovaný graf, ve kterém *subject* a *object* tvoří uzly, hrany tvoří *predicate*. Orientace hran je vždy od *subjectu* k *objectu*.



Obrázek 3.1: RDF graf[3]

Pro každý zdroj obvykle existuje více tvrzení, zdroj se může vyskytovat ve všech rolích (*subject*, *predicate* i *object*). Toto neplatí pro hodnoty, které nemají vlastní IRI, tzv. literálů, které se mohou vyskytovat pouze v roli *object*. Každý literál má přiřazen datový typ, aby mohl být správně reprezentován, literál typu string může ještě mít tzv. language tag – informaci o jazyku hodnoty.

Používání IRI a literálů je pro tvorbu tvrzení postačující, někdy se ale může potřeba využívání jedinečných identifikátorů zdát nadbytečná. V takových situacích je možno využít tzv. *blank node*. Jde o uzel, který nemá jedinečný identifikátor, nelze na něj tedy odkazovat z vnějšku grafu. *Blank node* lze použít pouze v rolích *object* a *subject*.

Výpis 3.2: Příklad *Blank node*

```
<http://www.example.org/places/1> <http://schema.org/geo>  
_:blanknode1 .  
_:blanknode1 <http://schema.org/latitude> 4.99138864E1 .  
_:blanknode1 <http://schema.org/longitude> 1.73754947E1 .
```

Tento zápis pomocí plně kvalifikovaných IRI je poněkud dlouhý a nepřehledný, zjednodušit se dá oddělením společných částí IRI a zavedením prefixů:

Výpis 3.3: Použití prefixů

```
@prefix ex: <http://www.example.org/> .
@prefix schema: <http://schema.org/> .

ex:places/1 schema:geo _:blanknode1 .
_:blanknode1 schema:latitude 4.99138864E1 .
_:blanknode1 schema:longitude 1.73754947E1 .
```

3.1.1.1 RDF Schema language

RDF Schema language je jazyk, kterým je možno definovat RDF slovník. Pro rozdělení zdrojů podle typu používáme třídy (class), pro vztahy mezi třídami používáme atributy (property). Ze tříd a atributů lze tvořit hierarchie pomocí podtříd a subatributů. Typ *subjectu* a *objectu* lze omezit definicí domén (domain) a rozsahů (range).

Konstrukt	Syntaktická forma	Popis
Class (třída)	C rdf:type rdfs:Class	zdroj C je RDF třída
Property (třída)	P rdf:type rdf:Property	zdroj P je RDF atribut
type (atribut)	I rdf:type C	zdroj I je instancí třídy C
subClassOf (atribut)	C1 rdfs:subClassOf C2	třída C1 je podtřídou třídy C2
subPropertyOf (atribut)	P1 rdfs:subPropertyOf P2	atribut P1 je subatributem atributu P2
domain (atribut)	P rdfs:domain C	doménou atributu P je třída C
range (atribut)	P rdfs:range C	rozsah pro atribut P je třída C

Tabulka 3.1: Konstrukty RDF schématu[5]

Použitím těchto konstruktů lze vytvořit například následující model:

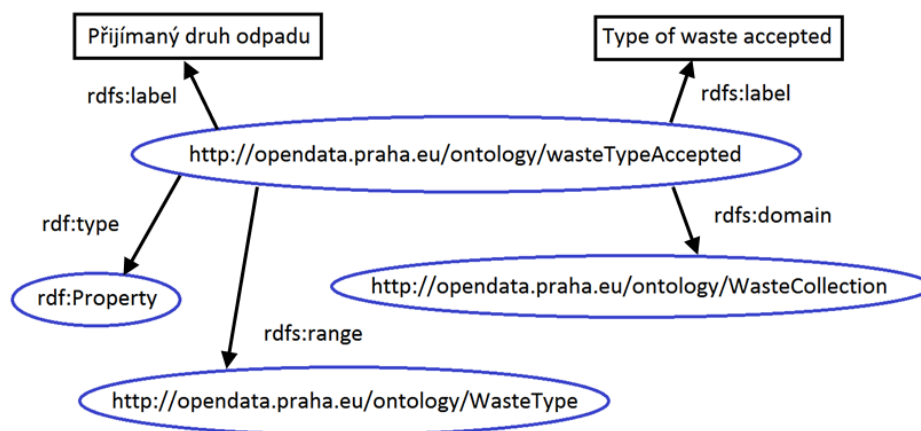
Výpis 3.4: Příklad RDF modelu

```
@prefix ex: <http://www.example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

ex:Osoba rdf:type rdfs:Class .
ex:jePribuzny rdf:type rdf:Property .
ex:Muz rdfs:subClassOf ex:Osoba .
ex:jeOtec rdfs:subPropertyOf ex:jePribuzny .
ex:jePribuzny rdfs:domain ex:Osoba .
ex:jeOtec rdfs:range ex:Muz .
```

3.1.1.2 Zápis RDF

RDF model je abstraktní, není závislý na žádné konkrétní syntaxi. Existuje několik konkrétní syntaxí, zápis stejného modelu v různých syntaxích vždy vytvoří stejný graf.



Obrázek 3.2: Vzorový RDF model

N-Triples Nejjednodušší používanou syntaxí RDF je syntaxe N-Triples. IRI v této syntaxi jsou plně kvalifikované, prefixy nejsou povoleny, což může vést k velmi dlouhým řádkům, každé tvrzení je na jednom řádku. Vzorový RDF model na obrázku 3.2 zapsaný v syntaxi N-Triples vypadá následovně:

Výpis 3.5: Vzorový model v syntaxi N-Triples

```

1 <http://opendata.praha.eu/ontology/wasteTypeAccepted> <http://www.
  w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.w3.org
  /1999/02/22-rdf-syntax-ns#Property> .
2 <http://opendata.praha.eu/ontology/wasteTypeAccepted> <http://www.
  w3.org/2000/01/rdf-schema#range> <http://opendata.praha.eu/
  ontology/WasteType> .
3 <http://opendata.praha.eu/ontology/wasteTypeAccepted> <http://www.
  w3.org/2000/01/rdf-schema#domain> <http://opendata.praha.eu/
  ontology/WasteCollection> .
4 <http://opendata.praha.eu/ontology/wasteTypeAccepted> <http://www.
  w3.org/2000/01/rdf-schema#label> "Přijímaný druh odpadu"@cs .
5 <http://opendata.praha.eu/ontology/wasteTypeAccepted> <http://www.
  w3.org/2000/01/rdf-schema#label> "Type of waste accepted"@en .

```

Snadno můžeme vidět, že počet řádků zápisu odpovídá počtu hran v grafu.

Turtle Tato syntaxe je rozšířením předchozí syntaxe N-Triples. IRI nemusí být plně kvalifikované, je možné používat prefixy. Zápis pomocí Turtle je stručnější díky využití řady syntaktických zkratk, je dobrým kompromisem co se týče náročnosti psaní, parsování a čitelnosti.

Vzorový RDF model na obrázku 3.2 zapsaný v syntaxi Turtle vypadá následovně:

Výpis 3.6: Vzorový model v syntaxi Turtle

```

1  @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2  @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
3
4  <http://opendata.praha.eu/ontology/wasteTypeAccepted> a rdf:
    Property ;
5  rdfs:range <http://opendata.praha.eu/ontology/WasteType> ;
6  rdfs:domain <http://opendata.praha.eu/ontology/
    WasteCollection> ;
7  rdfs:label "Přijímaný druh odpadu"@cs, "Type of waste
    accepted"@en .

```

První syntaktickou zkratku můžeme zaznamenat na řádce 4, kde je `rdf:type` nahrazen stručným `a`. Velkého zestručnění zápisu docílíme používáním středníku, díky kterému v dalších tvrzeních znovu používáme *subject*. Pokud chceme znovupoužít dvojici *subject* a *predicate*, hodnoty (*object*) od sebe oddělíme čárkou (řádek 7).

Více grafů Předchozí dvě syntaxe byly použitelné v případech, kdy bylo potřeba popsat pouze jediný graf. V případech, kdy je potřeba rozlišit více grafů, je třeba použít jinou syntaxi, například N-Quads, která je rozšířením N-Triples, nebo TriG, která je rozšířením Turtle. Obě syntaxe každému tvrzení přiřazují ještě název grafu. Protože ve své práci budu vždy pracovat pouze s datasey obsahujícími pouze jediný graf, nebudu zde zacházet do bližších podrobností.

JSON-LD Jak již název napovídá, je způsob zápisu RDF modelu, který pro využívá syntaxe JSON. JSON-LD definuje, jakým způsobem má vypadat vnitřní struktura JSONu.

Výpis 3.7: Vzorový model v syntaxi JSON-LD

```

1  {
2  "context": {
3  "rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#",
4  "rdfs": "http://www.w3.org/2000/01/rdf-schema#",
5  "xsd": "http://www.w3.org/2001/XMLSchema#"
6  },
7  "@id": "http://opendata.praha.eu/ontology/wasteTypeAccepted",
8  "@type": "rdf:Property",
9  "rdfs:domain": {
10 "id": "http://opendata.praha.eu/ontology/WasteCollection"
11 },
12 "rdfs:label": [
13 {
14 "@language": "cs",
15 "@value": "Přijímaný druh odpadu"
16 },
17 {
18 "@language": "en",

```

3. ZVYŠOVÁNÍ OTEVŘENOSTI DAT

```
19     "@value": "Type of waste accepted "  
20   }  
21 ],  
22 "rdfs:range": {  
23   "@id": "http://opendata.praha.eu/ontology/WasteType "  
24 }  
25 }
```

Parametr `@id` je jedinečný identifikátor, parametr `@context` definuje, jakým způsobem se má reprezentovat zbytek JSONu. JSON ve výpisu 3.7 je jednoduchý, ve větších datasetech (existuje více než jeden *subject*, hrany jdou z více uzlů) jsou ostatní parametry zabaleny v parametru `@graph`.

RDF/XML První existujícím způsobem zápisu RDF bylo RDF/XML, které využívá syntaxe XML. V době vzniku RDF to byl jediný možný způsob zápisu, někteří lidé ho proto stále nazývají „RDF“.

Výpis 3.8: Vzorový model v syntaxi RDF/XML

```
1 <?xml version="1.0" encoding="UTF-8"?>  
2 <rdf:RDF  
3   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"  
4   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"  
5 >  
6   <rdf:Description rdf:about="http://opendata.praha.eu/ontology/  
7     wasteTypeAccepted">  
8     <rdfs:label xml:lang="cs">Přijímaný druh odpadu</rdfs:label>  
9     <rdfs:label xml:lang="en">Type of waste accepted</rdfs:label>  
10    <rdfs:range rdf:resource="http://opendata.praha.eu/ontology/  
11      WasteType"/>  
12    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-  
13      syntax-ns#Property"/>  
14    <rdfs:domain rdf:resource="http://opendata.praha.eu/ontology/  
15      WasteCollection"/>  
16  </rdf:Description>  
17 </rdf:RDF>
```

Všechna tvrzení v datasetu jsou subelementy tagu `rdf:RDF`, na kterém jsou definovány i prefixy a jmenné prostory. Tvrzení je popsáno tagem `rdf:Description`, IRI *subjectu*, je definováno atributem `about`. Zbytek tvrzení, tedy *predicate* a *object* je definován v subelementu. *Predicate* je vždy název subelementu, pokud je *object* literál, je jeho hodnota obsah subelementu, pokud má *object* IRI, je obsah subelementu prázdný a hodnota je v atributu `rdf:resource`. Tag `rdf:Description` může mít více subelementů, v tom případě jde o více tvrzení se stejným *subjectem*.

3.1.2 Groovy

Groovy je objektově orientovaný jazyk vlastnostmi velmi blízký Javě, občas známý jako skriptovací jazyk pro tuto platformu. Stejně jako Java není překlá-

dán rovnou do strojového kódu, ale do tzv. bytekódu, který je následně interpretován JVM. Díky tomu je Groovy multiplatformní, pro běh kódu na dané platformě je potřeba pouze implementace interpreteru pro tuto platformu.

Syntaxe Groovy je podobná syntaxi Javy, velká část syntaxe jazyka Java je platná i pro syntaxi Groovy, neplatí to však stoprocentně. Groovy programátorovi nabízí řadu zjednodušení a rozšíření, kód je oproti Javě stručnější a čitelnější, využívá se více prvků funkcionálního programování. Na druhou stranu Groovy nenutí programátora deklarovat datové typy, což může vést k nižší přehlednosti a špatným návrhům a návykům při programování.

Groovy ve svém řešení využívám ve skriptu, který zdrojový dataset převede do RDF modelu a vytvoří výstup ve zvolené RDF syntaxi.

3.1.3 OpenCSV

OpenCSV je jednoduchá open source knihovna určená k parsování a zápisu formátu CSV.

3.1.4 OpenRDF Sesame

OpenRDF Sesame je rodina nástrojů, které slouží k práci s RDF daty. Mezi hlavní funkce patří parsování a uchovávání dat, dotazování nad daty pomocí jazyka *SPARQL*. RDF data je možné ukládat v paměti, na disku, nebo je možné k nim přistupovat vzdáleně přes SPARQL endpoint. Výhodou je, že všechny možnosti využívají shodné rozhraní, úložiště lze tedy snadno nahradit a to i implementacemi třetích stran.

Knihovny byly napsány v Javě a jsou dostupné jako open source[13].

3.1.5 RDF slovníky

Při vytváření RDF modelu datasetu jsem se snažil využít v maximální možné míře existující slovníky a nové třídy a atributy vytvářet jen v případech, kdy ty nebudou dostačovat.

RDF, RDFS Slovníky, které obsahují základní konstrukty RDF modelu, viz 3.1.1.1

XMLSchema Toto schema definuje datové typy používané v RDF.

OWL OWL je slovník používaný spolu se slovníky RDF a RDFS k definici nových ontologií.

Schema.org Cílem iniciativy schema.org je vytvářet, spravovat schemata pro strukturovaná data ve světě Internetu. Iniciativa je podporována předními

3. ZVYŠOVÁNÍ OTEVŘENOSTI DAT

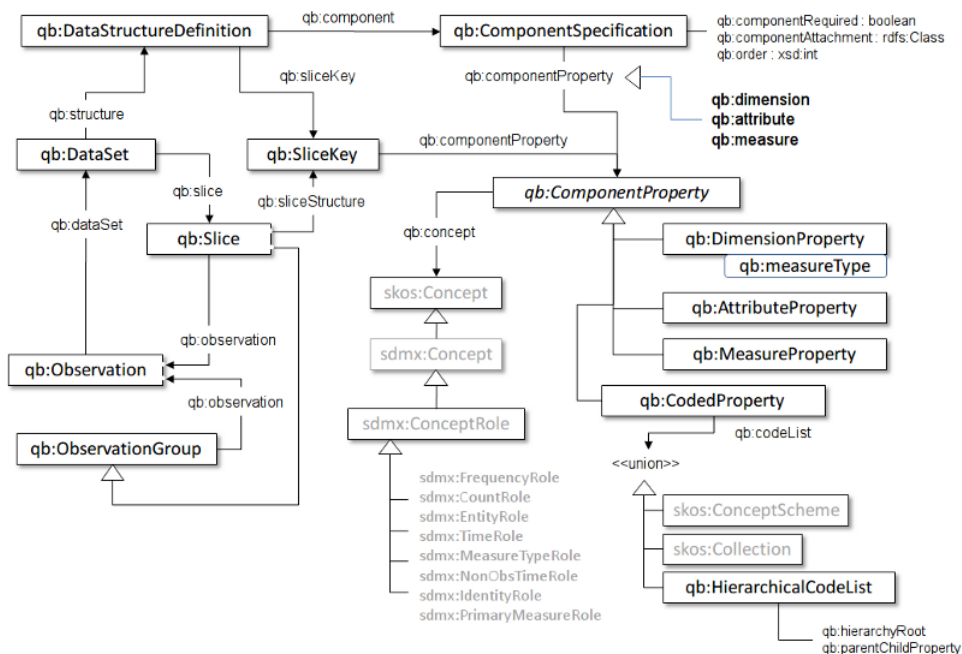
firmami v oboru, jako jsou Google, Microsoft, Yahoo a Yandex. Do diskuze o vývoji schemat se může zapojit každý[14].

Good relations Good relations je slovník, jehož hlavním účelem je popis komerčních služeb, produktů a organizací pro snazší zpracování obsahu webovými vyhledávači. Největší uplatnění nachází na webových stránkách eshopů.

DCMI Terms[15] Dublin Core Schema je menší slovník, který slouží k popisování objektů na webu. Svůj název získal podle názvu města, ve kterém byl navržen – v Dublinu v USA ve státě Ohio.

RDF Data Cube[4] RDF Data Cube je slovník, který slouží k publikaci statistických nebo jiných vícedimenzionálních dat ve formátu RDF.

Statistická data popsaná pomocí tohoto slovníku se v zásadě skládají ze dvou částí – definice a vlastních měřených hodnot. K vytvoření definice se využívají podtřídy `ComponentProperty`, pro dimenze (např. oblast nebo období) se používá `DimensionProperty`, pro typ pozorovaných dat `MeasureProperty`. Naměřené hodnoty jsou popsány pomocí třídy `Observation`, její instance sdružuje třída `DataSet`. `DataStructureDefinition` pak zajišťuje propojení těchto dvou částí. Celé schema je k dispozici na obrázku 3.3.



Obrázek 3.3: Ontologie RDF Data Cube[4]

SDMX Iniciativa SDMX byla založena v roce 2001 sedmi mezinárodními organizacemi (BIS, ECB, Eurostat, IMF, OECD, OSN, Světová banka). Jejím cílem bylo zajistit na poli statistických měření zvýšení efektivity. Výsledkem její práce bylo vytvoření několika standardů a technických doporučení, dostupné jsou na webu www.sdmx.org. Standardy přijala a používá ke sběru, zpracování a výměně statistických dat řada veřejných institucí, Organizace spojených národů je jako preferované využívá roku 2007[4].

3.2 Sběrná místa pro odpad a umístění kontejnerů

Městský úřad Praha 8 v současné době zveřejnila celkem 14 datasetů. Naprostá většina (12) je ve formátu CSV, jeden dataset je ve formě XLS, poslední je pouze odkaz webovou adresu.

Všechny datasety zveřejněné úřadem jsem prozkoumal, abych s ohledem na velikost, kvalitu dat a potenciál dalšího využití, vybral několik vhodných k převedení na vyšší úroveň otevřenosti. Vybral jsem dataset „Sběrná místa pro odpad a umístění kontejnerů“.

3.2.1 Popis datasetu

Dataset je ve formátu CSV, stejně jako u všech ostatních datasetů zveřejněných Městským úřadem Praha 8 k němu chybí dokumentace a je aktualizován jednou ročně.

3.2.1.1 ID

ID označuje jednoznačný identifikátor v rámci datasetu.

Formát dat Formát dat odpovídá regulárnímu výrazu `CZA\d{5}`.

3.2.1.2 IČ zřizovatele/majitele

Sloupec označuje identifikační číslo majitele či zřizovatele služby, která zajišťuje svoz odpadu. V záhlaví datasetu je název sloupce s překlepem (IČ zřizovatele/majitele).

Formát dat Formát dat odpovídá regulárnímu výrazu `\d{,8}`

Poznámka Identifikační číslo má přesně 8 cifer, pokud je číslo kratší, má být zleva doplněno nulami. Do datasetu ale byla data pravděpodobně exportována jako číslo, nuly u kratších IČ chybí.

3.2.1.3 Název zřizovatele/majitele

Sloupec označuje název majitele či zřizovatele služby, která zajišťuje svoz odpadu.

3.2.1.4 Url

Sloupec označuje webovou adresu, na které jsou dostupné webové stránky majitele či zřizovatele služby, která zajišťuje svoz odpadu.

Poznámka Zde došlo hned ke dvěma problémům, prvním bylo, že sloupec s názvem `url` je v datasetu dvakrát. Bylo by tedy vhodnější oba sloupce přejmenovat, aby bylo jasné, k čemu se hodnoty vztahují.

Druhým problémem jsou „prázdné“ hodnoty, které jsou reprezentovány symbolem `x`. Vhodnější by bylo použití prázdných řetězců.

3.2.1.5 IČ provozovatele zajišťující svoz

Sloupec označuje identifikační číslo provozovatele zajišťujícího svoz odpadu. Velmi často jsou hodnoty shodné s údaji ve sloupci *IČ zřizovatele/majitele*, zřizovatel a provozovatel se však mohou lišit.

Formát dat Formát dat odpovídá regulárnímu výrazu `\d{,8}`

Poznámka Formát dat je stejný jako ve sloupci *IČ zřizovatele/majitele* a data trpí stejnými problémy – ořezáváním nul.

3.2.1.6 Název provozovatele zajišťující svoz

Sloupec označuje název provozovatele zajišťujícího svoz odpadu.

3.2.1.7 Typ sběrného místa

Sloupec označuje typ sběrného místa, možnosti jsou stabilní, mobilní a IPPC.

3.2.1.8 Provozní doba

Ve sloupci je popsána doba, ve kterou je služba sběru odpadu dostupná.

Formát dat V datasetu se vyskytují 3 druhy hodnot:

1. Prázdná hodnota reprezentovaná symbolem `x`,
2. neomezená pracovní doba reprezentovaná hodnotou `nonstop`,
3. řetězec, ve kterém jsou popsány pracovní doby pro jednotlivé dny nebo rozsah dní.

3.2. Sběrná místa pro odpad a umístění kontejnerů

Hodnoty v tomto sloupci odpovídající posledně zmíněnému formátu vypadají například následovně:

- po -ne: 6:00 až 22: 00 hod.
- po: 7:30 až 18:00 hod. ut: 7:30 až 15:30 hod. stř: 7:30 až 18:00 hod. čt: 7:30 až 15.30 hod. pá: 7:30 - 15:30 hod.

Z těchto dat lze pozorovat následující pravidla:

- Kdekoli v řetězci se může vyskytovat jeden či více „bílých znaků“,
- hodiny a minuty jsou někdy odděleny dvojtečkou, někdy tečkou,
- čas *od* je od času *do* oddělen pomlčkou nebo slovem *až*,
- za každým časem *do* následuje text *hod.*,
- úterý jako jediný má zkratku bez diakritiky.

Poznámka Formát dat v tomto sloupci je poměrně komplikovaný, bylo by vhodné ho zjednodušit například:

- nahrazením symbolu *x* pro prázdné hodnoty prázdným řetězcem,
- používáním jediného oddělovače pro otevírací doby pro jednotlivé dny nebo rozsahy,
- odstraněním „bílých znaků“, odkud nejsou potřebné,
- používáním jednotného formátu času,
- používáním jediného oddělovače časů *od* a *do*.

3.2.1.9 Provoz povolen do

Ve sloupci je datum, do kterého má provozovatel povolení službu provozovat.

Formát dat V datasetu se vyskytují 3 druhy hodnot:

1. Prázdná hodnota reprezentovaná symbolem *x*,
2. povolení bez omezení data reprezentované textem *neomezeno*,
3. datum ve formátu *d.M.rrrr*.

3.2.1.10 Přijímaný odpad

Ve sloupci se vyskytují údaje o druzích odpadu, které sběrné místo odpadu přijímá.

Formát dat V datasetu se vyskytují 3 druhy hodnot:

1. Prázdná hodnota reprezentovaná symbolem **x**,
2. prázdná hodnota reprezentovaná prázdným řetězcem,
3. řetězec, ve kterém jsou jednotlivé durhy odpadu oddělené čárkou.

3.2.1.11 Přijímaný nebezpečný odpad

Sloupec obsahuje informaci o tom, zda sběrné místo přijímá i nebezpečný odpad.

Formát dat Dataset obsahuje 3 druhy hodnot:

1. Nepravda reprezentovaná symbolem **x**,
2. nepravda reprezentovaná prázdným řetězcem,
3. pravda reprezentovaná textem **ano**.

Poznámka Provozovatel může nebezpečný odpad buďto přijímat nebo nepřijímat, není logické, aby sloupec obsahoval více jak dva druhy hodnot. Vhodnější by bylo použít hodnoty `{true, false}` nebo `{1, 0}`.

3.2.1.12 Kontakt (mobil, telefon)

Sloupec obsahuje telefonní kontakt na provozovatele svozu odpadu.

Formát dat Dataset obsahuje 3 druhy hodnot:

1. Prázdná hodnota reprezentovaná symbolem **x**,
2. prázdná hodnota reprezentovaná prázdným řetězcem,
3. řetězec, ve kterém je jedno či více telefonních čísel oddělených čárkami.

3.2.1.13 E-mail

Sloupec obsahuje emailový kontakt na provozovatele svozu odpadu.

Formát dat Dataset obsahuje 3 druhy hodnot:

1. Prázdná hodnota reprezentovaná symbolem **x**,
2. prázdná hodnota reprezentovaná prázdným řetězcem,
3. řetězec, ve kterém je jeden či více emailových adres oddělených čárkami.

3.2.1.14 Url

Sloupec označuje webovou adresu, na které jsou dostupné webové stránky provozovatele svozu odpadu.

Poznámka Jde o druhý sloupec pojmenovaný „url“, viz 3.2.1.4.

3.2.1.15 Datová schránka

Sloupec obsahuje identifikátor datové schránky provozovatele svozu odpadu.

Formát dat Dataset obsahuje 3 druhy hodnot:

1. Prázdná hodnota reprezentovaná symbolem **x**,
2. prázdná hodnota reprezentovaná prázdným řetězcem,
3. alfanumerický identifikátor datové schránky.

3.2.1.16 Adresa

Sloupec obsahuje adresu provozovatele svozu odpadu.

Formát dat Většina hodnot tohoto sloupce obsahuje název městské části, název ulice a číslo domu, několik záznamů však název městské části postrádá. Název městské části je od názvu ulice oddělen čárkou.

Poznámka Za účelem snazšího parsování by bylo vhodnější sloupec rozdělit na dva nebo tři:

1. Obec,
2. název ulice (a číslo domu),
3. (číslo domu).

3.2.1.17 GPS umístění

Sloupce *GPS umístění (zeměpisná šířka)* a *GPS umístění (zeměpisná délka)* obsahují GPS souřadnice sběrných míst.

Formát dat Hodnota je kladné desetinné číslo následované písmenem značícím světovou stranu (N – sever, S – jih, W – západ, E – východ).

Poznámka Data v těchto sloupcích pravděpodobně obsahuje několik chyb, některé souřadnice se nachází mimo území Prahy, nejvzdálenější bod se nachází až v německém Národním parku Saské Švýcarsko.

Pro jednodušší zpracování hodnot by také bylo vhodnější nahradit písmeno světové zápornými hodnotami pro zeměpisnou délku západně od nultého poledníku a pro zeměpisnou šířku jižně od rovníku.

3.2.2 Model řešení

Povýšení dat na čtvrtou, případně pátou úroveň otevřenosti, spočívá v následujících krocích:

1. Analýza vzorových dat;
2. návrh transformace dat jednoho řádku na množinu tvrzení (RDF model);
3. aplikace RDF modelu na všechny řádky datasetu.

V této části se zabývám popisem druhého kroku, popsáním navrženého RDF modelu, který je také zachycen na obrázku 3.4.

3.2.2.1 Majitel

Majitel je firma, která vlastní nebo objednává službu svozu odpadu.

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{{IČ}}>, IČ, viz 3.2.1.2	
Třída	<http://schema.org/Organization>	
Predicate	Typ	Object
<http://schema.org/legalName>	literál	název majitele, viz 3.2.1.3
<http://schema.org/url>	literál	url majitele, viz 3.2.1.4
<http://www.w3.org/2002/07/owl#sameAs>	IRI	<http://linked.opendata.cz/resource/business-entity/CZ{{IČ}}>, IČ, viz 3.2.1.2

Tabulka 3.2: RDF *subject*: Majitel

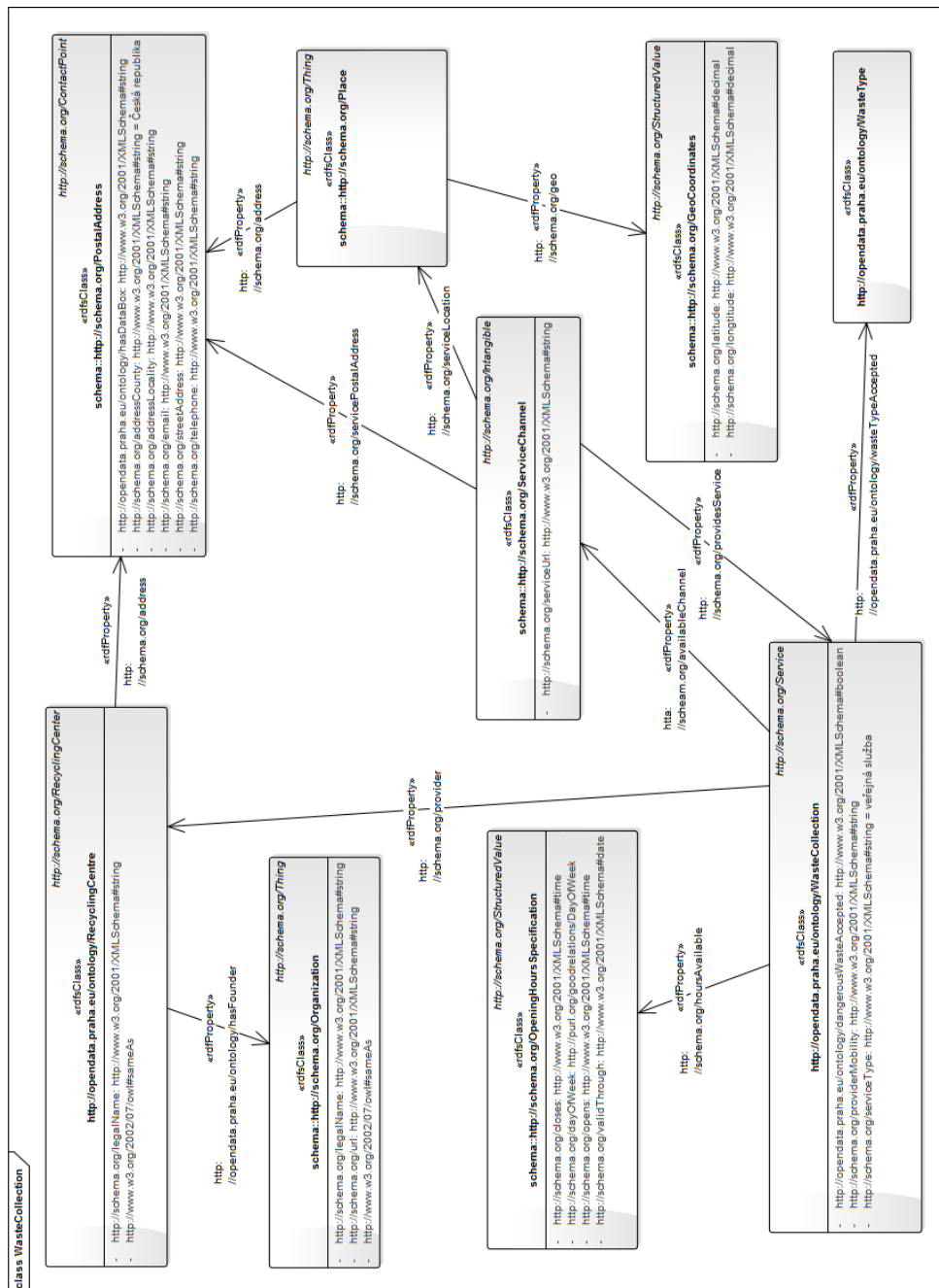
3.2.2.2 Provozovatel

Provozovatel je firma, která zajišťuje svoz odpadu. Majitel (3.2.2.1) a provozovatel je velmi často totožná firma.

3.2.2.3 Svoz odpadu

Svoz odpadu je služba poskytovaná provozovatelem (3.2.2.2).

3.2. Sběrná místa pro odpad a umístění kontejnerů



Obrázek 3.4: RDF model řešení svozu odpadu

3. ZVYŠOVÁNÍ OTEVŘENOSTI DAT

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{IČ}>, IČ, viz 3.2.1.5	
Třída	<http://opendata.praha.eu/ontology/RecyclingCentre>	
Je podřída	<http://schema.org/RecyclingCenter>	
Predicate	Typ	Object
<http://schema.org/address>	IRI	poštovní adresa, viz 3.2.2.8
<http://schema.org/legalName>	literál	název provozovatele, viz 3.2.1.6
<http://opendata.praha.eu/ontology/hasFounder>	IRI	majitel, viz 3.2.2.1
<http://www.w3.org/2002/07/owl#sameAs>	IRI	<http://linked.opendata.cz/resource/business-entity/CZ{IČ}>, IČ, viz 3.2.1.5

Tabulka 3.3: RDF *subject*: Provozovatel

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{IČ}/services/{ID}>, IČ – viz 3.2.1.5, ID – viz 3.2.1.1	
Třída	<http://opendata.praha.eu/ontology/WasteCollection>	
Podtřídou	<http://schema.org/Service>	
Predicate	Typ	Object
<http://schema.org/availableChannel>	IRI	komunikační kanál, viz 3.2.2.5
<http://schema.org/hoursAvailable>	blank node	otevírací doba, viz 3.2.2.4
<http://schema.org/providerMobility>	literál	typ sběrného místa, viz 3.2.1.7
<http://schema.org/serviceType>	literál	"veřejná služba"@cs
<http://opendata.praha.eu/ontology/dangerousWasteAccepted>	literál	true, pokud je přijímán nebezpečný odpad, viz 3.2.1.11
<http://opendata.praha.eu/ontology/wasteTypeAccepted>	IRI	typ odpadu, viz 3.2.2.7

Tabulka 3.4: RDF *subject*: Svoz odpadu

3.2.2.4 Otevírací doba

Otevírací doba popisuje dostupnost služby, v tomto případě dostupnost sběrného místa. Otevírací doba je definována časem od (<http://schema.org/opens>), časem do (<http://schema.org/closes>) a jedním nebo více dny v týdnu (<http://schema.org/dayOfWeek>). Otevírací doba dále může mít datum, kdy vyprší její platnost (<http://schema.org/validThrough>).

3.2. Sběrná místa pro odpad a umístění kontejnerů

Atribut	Hodnota	
IRI	blank node	
Třída	<http://schema.org/OpeningHoursSpecification>	
Predicate	Typ	Object
<http://schema.org/closes>	literál	čas <i>do</i> , viz 3.2.1.8
<http://schema.org/dayOfWeek>	IRI	<http://purl.org/goodrelations/v1#{{den}}>, den z {Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday} podle 3.2.1.8
<http://schema.org/opens>	literál	čas <i>od</i> , viz 3.2.1.8
<http://schema.org/validThrough>	literál	datum konce povolení provozu, viz 3.2.1.9

Tabulka 3.5: RDF *subject*: Otevírací doba

3.2.2.5 Komunikační kanál

Komunikační kanál obsahuje informace o službě, a to nejen webovou adresu a telefon, ale i místo, kde je služba k dispozici.

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{{IČ}}/services/{{ID}}/channel>, IČ – viz 3.2.1.5, ID – viz 3.2.1.1	
Třída	<http://schema.org/ServiceChannel>	
Predicate	Typ	Object
<http://schema.org/providesService>	IRI	svoz odpadu, viz 3.2.2.3
<http://schema.org/serviceLocation>	IRI	lokace, viz 3.2.2.6
<http://schema.org/servicePostalAddress>	IRI	poštovní adresa, viz 3.2.2.8
<http://schema.org/serviceUrl>	literál	url provozovatele svozu odpadu, viz 3.2.1.14

Tabulka 3.6: RDF *subject*: Komunikační kanál

3.2.2.6 Lokace

Lokace popisuje existující místo pomocí poštovní adresy a GPS souřadnic.

3.2.2.7 Typ odpadu

Sběrné místo může přijímat jen některé druhy odpadu. K jejich rozlišení slouží tato enumerace.

3. ZVYŠOVÁNÍ OTEVŘENOSTI DAT

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{IČ}/services/{ID}/location>, IČ – viz 3.2.1.5, ID – viz 3.2.1.1	
Třída	<http://schema.org/Place>	
Predicate	Typ	Object
<http://schema.org/address>	IRI	poštovní adresa, viz 3.2.2.8
<http://schema.org/geo>	blank node	GPS souřadnice, viz 3.2.2.9

Tabulka 3.7: RDF *subject*: Lokace

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/ontology/WasteType#{navez}>, navez – název typu odpadu bez diakritiky, viz 3.2.1.10	
Třída	<http://opendata.praha.eu/ontology/WasteType>	
Predicate	Typ	Object
<http://www.w3.org/2000/01/rdf-schema#label>	literál	"{navez}"@cs, navez – název typu odpadu, viz 3.2.1.10

Tabulka 3.8: RDF *subject*: Typ odpadu

3.2.2.8 Poštovní adresa

Poštovní adresa obsahuje informace o zemi, obci, ulici a číslu domu daného místa. Informace o poštovním směrovacím čísle zde není, neboť ji neobsahuje ani zdrojový dataset. Poštovní adresa jakožto podtřída kontaktního místa dále obsahuje informace o emailové adrese, telefonu a identifikátoru datové schránky.

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{IČ}/services/{ID}/address>, IČ – viz 3.2.1.5, ID – viz 3.2.1.1	
Třída	<http://schema.org/PostalAddress>	
Predicate	Typ	Object
<http://opendata.praha.eu/ontology/hasDataBox>	literál	identifikátor datové schránky, viz 3.2.1.15
<http://schema.org/addressCountry>	literál	"Česká republika"@cs
<http://schema.org/addressLocality>	literál	městská část, viz 3.2.1.16
<http://schema.org/email>	literál	emailová adresa, viz 3.2.1.13
<http://schema.org/telephone>	literál	telefonní číslo, viz 3.2.1.12
<http://schema.org/streetAddress>	literál	ulice a číslo domu, viz 3.2.1.16

Tabulka 3.9: RDF *subject*: Poštovní adresa

3.2.2.9 GPS souřadnice

GPS souřadnice jsou informace o zeměpisné šířce a délce daného místa.

Atribut	Hodnota	
IRI	blank node	
Třída	<http://schema.org/GeoCoordinates>	
Predicate	Typ	Object
<http://schema.org/latitude>	literál	zeměpisná šířka, viz 3.2.1.17
<http://schema.org/longitude>	literál	zeměpisná délka, viz 3.2.1.17

Tabulka 3.10: RDF *subject*: GPS souřadnice

3.3 Základní a mateřské školy

Jako další datasey vhodné ke zvýšení otevřenosti jsem vybral tyto tři:

- Seznam mateřských škol MČ Praha 8,
- Seznam základních škol MČ Praha 8,
- Počty tříd a žáků v základních školách zřízených MČ Praha 8.

Tři datasey jsem vybral z toho důvodu, že všechny popisují velmi podobné údaje. Řada základních škol je zároveň školou mateřskou, takže informace o nich lze najít ve více datasetech zároveň.

Oproti datasetu představenému v části 3.2 obsahují podstatně méně nedostatků, jsou snáze strojově zpracovatelné, na zjevné chyby jsem nenarazil.

3.3.1 Popis datasetu – Seznam základních škol MČ Praha 8

Dataset je ve formátu CSV, postrádá dokumentaci a je aktualizován jednou ročně.

3.3.1.1 IČ

Sloupec označuje identifikační číslo školy.

Formát dat Numerický řetězec o délce osmi znaků.

3.3.1.2 Název základní školy

Sloupec obsahuje název základní školy. Součástí názvu všech škol je i jejich adresa.

3.3.1.3 Typ

Tento sloupec popisuje, které typy vzdělávání škola poskytuje.

Formát dat Řetězec, ve kterém jsou typy vzdělávání oddělené čárkou a mezerou. Typy vzdělávání jsou ZŠ (základní škola) a MŠ (mateřská škola).

3.3.1.4 Adresa

Sloupec obsahuje adresu školy. K dispozici je název ulice, číslo domu, poštovní směrovací číslo a název městské čtvrti.

Formát dat Řetězec, obsahuje nejprve název ulice a číslo domu, následuje poštovní směrovací číslo a název čtvrti. Název ulice s číslem domu od poštovního čísla odděluje čárka, poštovní směrovací číslo a název čtvrti odděluje mezera.

Poznámka Přestože data v tomto sloupci jsou v podstatně lepším stavu než v případě adres v prvním datasetu (viz 3.2.1.16), neuškodilo by je podobným způsobem rozdělit do tří až čtyř sloupců.

3.3.1.5 Latitude

Sloupec obsahuje informace o zeměpisné šířce polohy školy.

Formát dat Desetinné číslo

3.3.1.6 Longitude

Sloupec obsahuje informace o zeměpisné délce polohy školy.

Formát dat Desetinné číslo

3.3.1.7 Ředitel / Ředitelka

Sloupec obsahuje celé jméno ředitele nebo ředitelky školy, a to včetně titulů před i za jménem nebo prostředního jména.

Formát dat Formát odpovídá běžnému zápisu, tedy tituly před jménem, jméno, prostřední jméno, příjmení, čárka a tituly za jménem.

3.3.1.8 Telefon

V tomto sloupci se nachází informace o telefonním kontaktu na ředitele nebo ředitelku školy.

Formát dat Řetězec, sloupec neobsahuje prázdné hodnoty.

Poznámka Z názvu sloupce není jasné, že jde o kontakt na ředitele nebo ředitelku, bylo by vhodné ho přejmenovat.

3.3.1.9 E-mail

Sloupec obsahuje emailovou adresu ředitele nebo ředitelky školy.

Formát dat Řetězec, sloupec neobsahuje prázdné hodnoty.

Poznámka Stejně jako u předešlého sloupce i zde není jasné, že jde o kontakt na ředitele nebo ředitelku a sloupec by bylo vhodné přejmenovat.

3.3.1.10 ID schránka

Ve sloupci se nachází identifikátor datové schránky školy.

Formát dat Sedmiznakový alfanumerický identifikátor datové schránky nebo prázdný řetězec reprezentující prázdnou hodnotu.

3.3.1.11 Telefonní ústředna

Sloupec obsahuje číslo telefonní ústředny školy.

Formát dat Řetězec, sloupec neobsahuje prázdné hodnoty.

3.3.1.12 E-mail školy

Sloupec obsahuje emailovou adresu školy.

Formát dat Řetězec, případně prázdné hodnoty reprezentované prázdným řetězcem.

3.3.1.13 Webová prezentace

Sloupec obsahuje adresu webových stránek školy.

Formát dat Řetězec, sloupec neobsahuje prázdné hodnoty.

3.3.1.14 Zástupce ředitele pro předškolní vzdělávání

Pokud je základní škola zároveň mateřskou školou, je v tomto sloupci uvedeno celé jméno zástupce ředitele pro předškolní vzdělávání.

Formát dat Shodný jako ve sloupci „Ředitel / Ředitelka“, viz 3.3.1.7.

3.3.1.15 Telefon MŠ

Sloupec obsahuje telefonní kontakt na část školy pro předškolní vzdělávání.

Formát dat Řetězec, případně prázdná hodnota reprezentovaná prázdným řetězcem.

3.3.1.16 E-mail MŠ

Sloupec obsahuje emailovou adresu části školy pro předškolní vzdělávání.

Formát dat Řetězec, případně prázdná hodnota reprezentovaná prázdným řetězcem.

3.3.2 Popis datasetu – Seznam mateřských škol MČ Praha 8

Dataset je ve formátu CSV, postrádá dokumentaci a je aktualizován jednou ročně.

3.3.2.1 Název mateřské školy

Sloupec obsahuje název mateřské školy včetně její adresy.

3.3.2.2 IČ

Sloupec označuje identifikační číslo školy.

Formát dat Numerický řetězec o délce osmi znaků.

3.3.2.3 Název

Sloupec obsahuje název mateřské školy včetně její adresy.

Poznámka Sloupec je víceméně kopií sloupce „Název mateřské školy“, jediný rozdíl spočívá ve zkrácení „Základní škola“ na ZŠ a „Mateřská škola“ na MŠ u škol, které poskytují jak základní, tak předškolní vzdělání. Existence tohoto sloupce v datasetu se mi zdá nadbytečná.

3.3.2.4 Typ, Adresa, Latitude, Longitude

Význam a formát dat sloupců odpovídá stejnojmenným sloupcům v předchozím datasetu:

Typ viz 3.3.1.3,

Adresa viz 3.3.1.4,

Latitude viz 3.3.1.5,

Longitude viz 3.3.1.6.

3.3.2.5 Statistická data

Dataset obsahuje tři sloupce, ve kterém se nachází statistické údaje.

Školní rok Ve sloupci je popsán školní rok, pro který jsou měření platná. Protože dataset obsahuje pouze data ze školního roku 2015/2016, nabízí se otázka, zda-li je to tím, že je dataset nový, nebo úřad uchovává data pouze z aktuálního školního roku.

Počet tříd Jde o jeden z měřených údajů, a to počet tříd mateřské školy, které v daném školním roce byly otevřeny.

Počet dětí Ve sloupci je druhý z měřených údajů, a to celkový počet dětí ve třídách mateřské školy, které v daném školním roce byly otevřeny.

3.3.3 Popis datasetu – Počty tříd a žáků v základních školách zřízených MČ Praha 8

Dataset je ve formátu CSV, postrádá dokumentaci a je aktualizován jednou ročně.

3.3.3.1 Název základní školy, IČ, Typ, Adresa

Význam a formát dat sloupců odpovídá stejnojmenným sloupcům v datasetu „Seznam základních škol MČ Praha 8“:

Název základní školy viz 3.3.1.2,

IČ viz 3.3.1.1

Typ viz 3.3.1.3,

Adresa viz 3.3.1.4.

3.3.3.2 Statistická data

Dataset obsahuje celkem čtyři sloupce, ve kterém se nachází statistická data.

Třída Sloupec je sice pojmenován „Třída“, přesnější by ale bylo použít název „Ročník“. Ve sloupci je popsán ročník, pro který jsou měřená data platná. Přestože některé ze škol v datasetu nabízejí i předškolní výuku, dataset obsahuje pouze údaje pro ročníky základních škol reprezentované římskými číslovkami (I.-IX), v několika případech i údaje o přípravných třídách.

Školní rok Ve sloupci je popsán školní rok, pro který jsou měření platná. Stejně jako v předchozím datasetu je zde pouze hodnota 2015/2016.

Počet tříd Sloupec vyjadřuje počet tříd příslušného ročníku, které základní škola v daném školním roce otevřela.

Počet žáků Sloupec vyjadřuje celkový počet žáků ve třídách příslušného ročníku, které základní škola v daném školním roce otevřela.

3.3.4 Model řešení

Na základě analýzy zmíněných datasetů jsem vytvořil RDF model. Jako základ modelu jsem použil data z datasetu „Seznam základních škol MČ Praha 8“ (viz 3.3.1), protože obsahuje největší množství informací. Pro propojení datasetů jsem použil sloupec IČ, neboť se nachází ve všech datasetech a ve stejném formátu. Některé údaje jsou ve více datasetech najednou, „duplicitní“ hodnoty používám pouze v případě, že daná škola se v předchozích datasetech nevyskytla.

Datasety jsou skriptem zpracovávány v tomto pořadí:

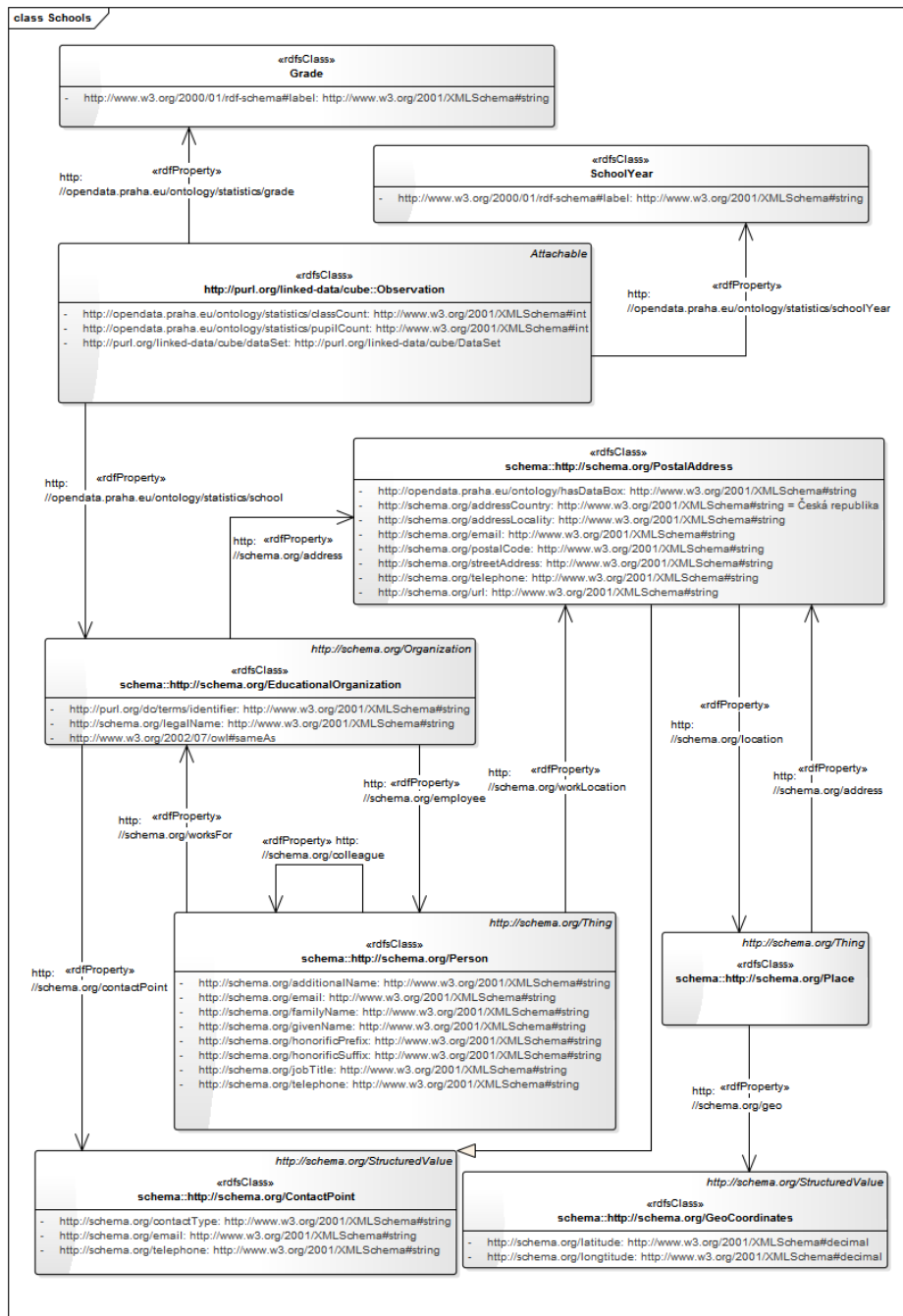
1. Seznam základních škol MČ Praha 8 (viz 3.3.1),
2. Seznam mateřských škol MČ Praha 8 (viz 3.3.2),
3. Počty tříd a žáků v základních školách zřízených MČ Praha 8 (viz 3.3.3).

RDF model, který popíšu v následující části práce, je oproti modelu týkajícímu se svozu odpadu mírně složitější, neboť obsahuje statistická data popsaná pomocí slovníku RDF Data Cube. Model bez definice datové kostky je zachycen na obrázku 3.5.

3.3.4.1 Škola

V této entitě jsou popsány vlastnosti školy, její název, identifikační číslo. Je typu `<http://schema.org/ElementarySchool>` (základní škola), `http://schema.org/Preschool` (mateřská škola), případně obojího podle toho, jaké úrovně vzdělávání poskytuje. Dále obsahuje link na zdroj v datasetu `linked.opendata.cz`, který také popisuje stejnou školu.

3.3. Základní a mateřské školy



Obrázek 3.5: RDF model řešení škol

3. ZVYŠOVÁNÍ OTEVŘENOSTI DAT

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{IČ}>, IČ – viz 3.3.1.1	
Třída	<http://schema.org/Preschool>, <http://schema.org/ElementarySchool>, případně obojí – viz 3.3.1.3	
Predicate	Typ	Object
<http://purl.org/dc/terms/identifier>	literál	IČ školy, viz 3.3.1.1
<http://schema.org/address>	IRI	adresa školy, viz 3.3.4.2
<http://schema.org/contactPoint>	IRI	kontakt mateřské školy, viz 3.3.4.4
<http://schema.org/employee>	IRI	zaměstnanec školy, viz 3.3.4.3
<http://schema.org/legalName>	literál	název školy, viz 3.3.1.2
<http://www.w3.org/2002/07/owl#sameAs>	IRI	<http://linked.opendata.cz/resource/business-entity/CZ-{IČ}>, IČ – viz 3.3.1.1

Tabulka 3.11: RDF *subject*: Škola

3.3.4.2 Adresa školy

Protože <http://schema.org/PostalAddress> je podtřídou třídy <http://schema.org/ContactPoint>, obsahuje adresa školy kromě názvu ulice, čísla domu, názvu městské části a poštovního směrovacího čísla i informace jako telefon, email a adresu webových stránek školy.

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{IČ}/address>, IČ – viz 3.3.1.1	
Třída	<http://schema.org/PostalAddress>	
Predicate	Typ	Object
<http://opendata.praha.eu/ontology/hasDataBox>	literál	identifikátor datové schránky, viz 3.3.1.10
<http://schema.org/addressCountry>	literál	"Česká republika"@cs
<http://schema.org/addressLocality>	literál	název městské čtvrti, viz 3.3.1.4
<http://schema.org/email>	literál	email základní školy, viz 3.3.1.12
<http://schema.org/location>	IRI	Lokace školy, viz 3.3.4.5
<http://schema.org/postalCode>	literál	poštovní směrovací číslo, viz 3.3.1.4
<http://schema.org/streetAddress>	literál	název ulice a číslo domu, viz 3.3.1.4
<http://schema.org/telephone>	literál	telefonní číslo ústředny školy, viz 3.3.1.11
<http://schema.org/url>	literál	adresa webových stránek školy, viz 3.3.1.13

Tabulka 3.12: RDF *subject*: Adresa školy

3.3.4.3 Zaměstnanec školy

V tomto případě je zaměstnancem školy ředitel nebo zástupce ředitele pro předškolní vzdělávání. Nacházejí se zde informace jako jméno či jména, příjmení, akademické tituly a název pracovní pozice. Pokud je zaměstnanec ředitel školy, má také telefon a emailovou adresu.

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{IČ}/staff/{Nazev}>, IČ – viz 3.3.1.1, Nazev – celé jméno zaměstnance bez diakritiky, viz 3.3.1.7	
Třída	<http://schema.org/Person>	
<i>Predicate</i>	Typ	<i>Object</i>
<http://schema.org/additionalName>	literál	prostřední jméno, viz 3.3.1.7
<http://schema.org/colleague>	IRI	zaměstnanec stejné školy (spolupracovník)
<http://schema.org/email>	literál	email, viz 3.3.1.9
<http://schema.org/familyName>	literál	příjmení, viz 3.3.1.7
<http://schema.org/givenName>	literál	křestní jméno, viz 3.3.1.7
<http://schema.org/honorificPrefix>	literál	akademické tituly před jménem, viz 3.3.1.7
<http://schema.org/honorificSuffix>	literál	akademické tituly za jménem, viz 3.3.1.7
<http://schema.org/jobTitle>	literál	"Ředitel školy"@cs resp. "Zástupce ředitele pro předškolní vzdělávání"@cs
<http://schema.org/telephone>	literál	telefonní číslo, viz 3.3.1.8
<http://schema.org/workLocation>	IRI	adresa, na které zaměstnanec pracuje, viz 3.3.4.2
<http://schema.org/worksFor>	IRI	škola, pro kterou zaměstnanec pracuje, viz 3.3.4.1

Tabulka 3.13: RDF *subject*: Zaměstnanec školy

3.3.4.4 Kontakt mateřské školy

Pokud je základní škola zároveň i mateřskou školou, zde jsou uchovány kontaktní údaje právě na část školy pro předškolní vzdělávání.

3.3.4.5 Lokace školy

Lokace školy popisuje její polohu, obsahuje odkazy na GPS souřadnice a adresu školy.

3.3.4.6 GPS souřadnice školy

GPS souřadnice jsou informace o zeměpisné šířce a délce daného místa.

3. ZVYŠOVÁNÍ OTEVŘENOSTI DAT

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{IČ}/kindergarten>, IČ – viz 3.3.1.1	
Třída	<http://schema.org/ContactPoint>	
Predicate	Typ	Object
<http://schema.org/contactType>	literál	"Kontakt mateřské školy"@cs
<http://schema.org/email>	literál	email, viz 3.3.1.16
<http://schema.org/telephone>	literál	telefonní číslo, viz 3.3.1.15

Tabulka 3.14: RDF *subject*: Kontakt mateřské školy

Atribut	Hodnota	
IRI	<http://opendata.praha.eu/resource/business-entity/{IČ}/location>, IČ – viz 3.3.1.1	
Třída	<http://schema.org/Place>	
Predicate	Typ	Object
<http://schema.org/address>	IRI	adresa školy, viz 3.3.4.2
<http://schema.org/geo>	blank node	GPS souřadnice, viz 3.3.4.6

Tabulka 3.15: RDF *subject*: Lokace školy

Atribut	Hodnota	
IRI	blank node	
Třída	<http://schema.org/GeoCoordinates>	
Predicate	Typ	Object
<http://schema.org/latitude>	literál	zeměpisná šířka, viz 3.3.1.5
<http://schema.org/longitude>	literál	zeměpisná délka, viz 3.3.1.6

Tabulka 3.16: RDF *subject*: GPS souřadnice

3.3.4.7 Ročník, Školní rok

Ročník a školní rok jsou enumerace používané jako dimenze datové kostky.

3.3.4.8 Měření

Měření obsahuje statistické hodnoty – počet tříd a žáků na dané škole, v daném ročníku a v daném školním roce.

3.3.4.9 Datová kostka

Pro publikaci otevřených dat ve formátu RDF jsem využil slovník RDF Data Cube[4]. Struktura slovníku je zachycena na obrázku 3.3, v následující části předvedu její použití.

Ve výpisu 3.9 uvádím definici prefixů a jmenných prostorů, díky kterým budou další výpisy kratší a přehlednější. Kromě dříve použitých se zde nově objevují slovníky rodiny SDMX.

3.3. Základní a mateřské školy

Atribut	Hodnota	
Ročník – IRI	<http://opendata.praha.eu/ontology/Grade#{nazev}>, nazev – název ročníku, viz 3.3.3.2	
Ročník – třída	<http://opendata.praha.eu/ontology/Grade>	
Školní rok – IRI	<http://opendata.praha.eu/ontology/SchoolYear#{nazev}>, nazev – název školního roku, viz 3.3.2.5	
Školní rok – třída	<http://opendata.praha.eu/ontology/SchoolYear>	
Predicate	Typ	Object
<http://www.w3.org/2000/01/rdf-schema#label>	literál	"#{název}"@cs, nazev – název ročníku resp. školního roku, viz 3.3.3.2 resp. 3.3.2.5

Tabulka 3.17: RDF *subject*: Ročník, Školní rok

Atribut	Hodnota	
IRI	blank node	
Třída	<http://purl.org/linked-data/cube/Observation>	
Predicate	Typ	Object
<http://purl.org/linked-data/cube/dataSet>	IRI	<http://opendata.praha.eu/resource/statistics/class-and-student-stats> (definice datové kostky, viz 3.3.4.9)
<http://opendata.praha.eu/ontology/statistics/school>	IRI	škola, viz 3.3.4.1
<http://opendata.praha.eu/ontology/statistics/grade>	IRI	ročník, viz 3.3.4.7
<http://opendata.praha.eu/ontology/statistics/schoolYear>	IRI	školní rok, viz 3.3.4.7
<http://opendata.praha.eu/ontology/statistics/classCount>	literál	počet tříd, viz 3.3.2.5 resp. 3.3.3.2
<http://opendata.praha.eu/ontology/statistics/pupilCount>	literál	počet žáků, viz 3.3.2.5 resp. 3.3.3.2

Tabulka 3.18: RDF *subject*: Měření

Výpis 3.9: Datová kostka – jmenné prostory

1	@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
2	@prefix sdmx-attribute: <http://purl.org/linked-data/sdmx/2009/attribute#> .
3	@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
4	@prefix sdmx-measure: <http://purl.org/linked-data/sdmx/2009/measure#> .
5	@prefix schema: <http://schema.org/> .
6	@prefix prg: <http://opendata.praha.eu/ontology/> .
7	@prefix qb: <http://purl.org/linked-data/cube/> .
8	@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
9	@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

Druhým krokem vytvoření datové kostky je určení jejích dimenzí. Dimenze jsou kritéria, která specifikují měření, jako například doba či lokalita, pro kte-

3. ZVYŠOVÁNÍ OTEVŘENOSTI DAT

rou je měření platné. Jak můžeme vidět ve výpisu 3.10, dimenze je definována pomocí třídy `qb:DimensionProperty`.

První dimenzí je škola (řádky 1-4), jde o specifikaci oblasti, proto je subatributem `sdmx-dimension:refArea`. *Object* je pomocí `rdfs:range` omezen na instance typu `schema:EducationalOrganization`, v datasetu se používají dvě její podtřídy (`schema:Preschool` a `schema:ElementarySchool`).

Druhou dimenzí je školní rok (řádky 6-9). Jde o specifikaci časového období, proto je subatributem `sdmx-dimension:refPeriod`. Strávil jsem mnoho času hledáním vhodných existujících webových zdrojů, které by popisovaly školní rok jako časové období, protože jsem ale žádný nenašel, definoval jsem za tím účelem vlastní třídu (`prg:SchoolYear`).

Poslední dimenzí je ročník (řádky 11-13). Stejně jako u školního roku i zde jsem pro ročník definoval vlastní třídu (`prg:Grade`).

Výpis 3.10: Datová kostka – dimenze

```
1 <http://opendata.praha.eu/ontology/statistics/school> a rdf:
   Property , qb:DimensionProperty ;
2   rdfs:subPropertyOf sdmx-dimension:refArea ;
3   rdfs:label "Škola"@cs ;
4   rdfs:range schema:EducationalOrganization .
5
6 <http://opendata.praha.eu/ontology/statistics/schoolYear> a rdf:
   Property , qb:DimensionProperty ;
7   rdfs:subPropertyOf sdmx-dimension:refPeriod ;
8   rdfs:label "Školní rok"@cs ;
9   rdfs:range prg:SchoolYear .
10
11 <http://opendata.praha.eu/ontology/statistics/grade> a rdf:
   Property , qb:DimensionProperty ;
12   rdfs:label "Ročník"@cs ;
13   rdfs:range prg:Grade .
```

Ve výpisu 3.11 je zobrazen třetí krok k vytvoření datové kostky – popis pozorovaných hodnot. V datasetech jsou dva typy měření, a to počet tříd (řádky 1-4) a počet žáků (řádky 6-9). Stejně jako u dimenzí i zde jde o RDF atributy, tentokrát typu `qb:MeasureProperty`. Počet tříd i počet žáků je subatributem `sdmx-measure:obsValue`. Je zde také omezen datový typ *objectu*, hodnoty obou měření musí být celá nezáporná čísla.

Výpis 3.11: Datová kostka – popis hodnot

```
1 <http://opendata.praha.eu/ontology/statistics/classCount> a rdf:
   Property , qb:MeasureProperty ;
2   rdfs:label "Počet tříd"@cs ;
3   rdfs:subPropertyOf sdmx-measure:obsValue ;
4   rdfs:range xsd:nonNegativeInteger .
5
6 <http://opendata.praha.eu/ontology/statistics/pupilCount> a rdf:
   Property , qb:MeasureProperty ;
7   rdfs:label "Počet žáků"@cs ;
```



```

8 | rdfs:subPropertyOf sdmx-measure:obsValue ;
9 | rdfs:range xsd:nonNegativeInteger .

```

V předchozích krocích jsem definoval dimenze a popis hodnot, v následujícím kroku je čas je propojit. K tomu slouží `qb:DataStructureDefinition`, do které se připojí dimenze a popisy hodnot jako komponenty (`qb:component`). Použití je vidět ve výpisu 3.12. Je zde také vidět využití stručnější syntaxe pro zápis prázdných uzlů (*blank node*) pomocí hranatých závorek.

Ve výpisu se kromě dimenzí (řádky 2-14) a popisu hodnot (řádky 14-20) nachází ještě jedna komponenta, která říká, že dataset datové kostky (`qb:DataSet`) musí povinně mít atribut `sdmx-attribute:unitMeasure`.

Výpis 3.12: Datová kostka – struktura

```

1 | <http://opendata.praha.eu/resource/statistics/class-and-student-
  | stats/definition> a qb:DataStructureDefinition ;
2 |   qb:component [
3 |     a qb:ComponentSpecification ;
4 |     qb:dimension <http://opendata.praha.eu/ontology/statistics/
  | school> ;
5 |     qb:order "1"^^xsd:int
6 |   ], [
7 |     a qb:ComponentSpecification ;
8 |     qb:dimension <http://opendata.praha.eu/ontology/statistics/
  | grade> ;
9 |     qb:order "2"^^xsd:int
10 |   ], [
11 |    a qb:ComponentSpecification ;
12 |    qb:dimension <http://opendata.praha.eu/ontology/statistics/
  | schoolYear> ;
13 |    qb:order "3"^^xsd:int
14 |   ], [
15 |    a qb:ComponentSpecification ;
16 |    qb:measure <http://opendata.praha.eu/ontology/statistics/
  | classCount>
17 |   ], [
18 |    a qb:ComponentSpecification ;
19 |    qb:measure <http://opendata.praha.eu/ontology/statistics/
  | pupilCount>
20 |   ], [
21 |    a qb:ComponentSpecification ;
22 |    qb:attribute sdmx-attribute:unitMeasure ;
23 |    qb:componentRequired true ;
24 |    qb:componentAttachment qb:DataSet>
25 |   ] .

```

Předposledním krokem je definice datasetu datové kostky, kterou zobrazuje výpis 3.13. K tomuto účelu slouží `qb:DataSet`. Najdeme zde stručný popis, odkaz na strukturu datové kostky definovanou v předchozích krocích a parametr, který struktura vyžaduje (řádek 4).

V datasetu je možné dále definovat tzv. *Slice*. Jde o výběr některých pozorování, většinou podle vybraného kritéria, např. podle konkrétního období.

3. ZVYŠOVÁNÍ OTEVŘENOSTI DAT

V datové kostce zachycující počty tříd a žáků na školách v Praze 8 jsem ale tuto možnost nevyužil.

Výpis 3.13: Datová kostka – dataset

```
1 <http://opendata.praha.eu/resource/statistics/class-and-student-
  stats> a qb:DataSet ;
2   rdfs:label "Počty tříd a žáků na MŠ a ZŠ v Praze 8"@cs ;
3   qb:structure <http://opendata.praha.eu/resource/statistics/class
  -and-student-stats/definition> ;
4   sdmx-attribute:unitMeasure <http://www.w3.org/2007/ont/unit#
  Quantity> .
```

Tím byla datová kostka definována, zbývá ji jen naplnit pozorováními (**qb:Observation**). Vzorové pozorování se nachází ve výpisu 3.14.

Výpis obsahuje odkaz na dataset (řádek 2), všechny dimenze kostky (řádky 3-5) a naměřené hodnoty (řádky 6 a 7).

Výpis 3.14: Datová kostka – pozorování

```
1 _:node1ahu73at1x65 a qb:Observation ;
2   qb:DataSet <http://opendata.praha.eu/resource/statistics/class-
  and-student-stats> ;
3   <http://opendata.praha.eu/ontology/statistics/school> <http://
  opendata.praha.eu/resource/business-entity/60433230> ;
4   <http://opendata.praha.eu/ontology/statistics/grade> <http://
  opendata.praha.eu/ontology/Grade#I.> ;
5   <http://opendata.praha.eu/ontology/statistics/schoolYear> <http
  ://opendata.praha.eu/ontology/SchoolYear#2015%2F2016> ;
6   <http://opendata.praha.eu/ontology/statistics/classCount> "4"^^
  xsd:int ;
7   <http://opendata.praha.eu/ontology/statistics/pupilCount> "92"^^
  xsd:int .
```

3.4 Implementace

Pro zpracování zdrojových datasetů a vytvoření otevřených dat páté úrovně jsem vytvořil dva skripty v jazyce Groovy.

Skripty pro svůj běh vyžadují instalaci JRE a Groovy. Skript byl testován s Groovy verze 2.4.6, ale s vysokou pravděpodobností by měl fungovat i při použití verze 2.3.0 a vyšší. Minimální verze JRE potřebná ke spuštění skriptu je 8. Dále je vyžadováno stabilní připojení k Internetu, neboť skript knihovny třetích stran stahuje teprve při běhu.

Skripty při spuštění vyžadují jeden či více prepínačů nebo parametrů. Použití skriptů je zobrazeno ve výpisu 3.15 (svoz odpadu) resp. 3.16 (školy). Oběma skriptům lze nastavit výstupní formát, k dispozici jsou syntaxe JSON-LD, N-Triples, RDF/JSON, RDF/XML a výchozí syntaxe Turtle. Stejně tak je u obou skriptů možné nastavit cestu a název výstupního souboru, přípona bude k názvu připojena automaticky podle požadovaného formátu.

Svoz odpadu Skript pro svoz odpadu vyžaduje pouze jeden zdrojový dataset, pro jeho výběr proto není potřeba žádný přepínač.

Výpis 3.15: Parametry skriptu – svoz odpadu

```

1 usage: WasteCollection.groovy -[fhou] [input]
2 -f,--format <format>          RDF output format defined by <
   format >.
3
4                               Supported formats are JSON-LD, N-
   Triples,
5                               RDF/JSON, RDF/XML, Turtle.
6 -h,--help                      Show usage information
7 -o,--output-filename <path>   Path of the output file (extension
   will be
8                               appended according to the format)
   -u,--source-url <url>       URL of input file to be downloaded

```

Školy Ke spuštění skriptu pro školy je potřeba zadat všechny tři datasety pomocí přepínačů, „Seznam základních škol MČ Praha 8“ pomocí přepínače `-e`, „Seznam mateřských škol MČ Praha 8“ pomocí přepínače `-k` a „Počty tříd a žáků v základních školách zřízených MČ Praha 8“ pomocí přepínače `-s`.

Výpis 3.16: Parametry skriptu – školy

```

1 usage: Schools.groovy -[efhknopst]
2 -e,--elementary <path>       Path of the elementary school
   dataset
3 -f,--format <format>          RDF output format defined by <
   format >.
4
5                               Supported formats are JSON-LD, N-
   Triples,
6                               RDF/JSON, RDF/XML, Turtle.
7 -h,--help                      Show usage information
8 -k,--kindergarten <path>    Path of the kindergarten dataset
9 -n,--url-kindergarten <url> URL of the kindergarten school
   dataset
10 -o,--output-filename <path>  Path of the output file (extension
   will be
11                               appended according to the format)
   -p,--url-elementary <url>   URL of the elementary school
   dataset
12 -s,--statistics <path>       Path of the statistics dataset
13 -t,--url-statistics <url>    URL of the statistics dataset

```

Automatické stažení datasetů Oba skripty mají pro zdrojové skripty alternativní možnost specifikace pomocí URL, odkud si je skripty samy stáhnou, tato funkce bohužel z neznámých důvodů nefunguje stoprocentně správně a je doporučeno používat již stažené datasety.

Pro jednodušší vyzkoušení jsem pro oba skripty vytvořil ještě dávkové soubory pro spuštění bez jakýchkoli parametrů.

Referenční aplikace

Abych demonstroval využití otevřených dat v praxi, vytvořil jsem jednoduchou webovou aplikaci, která využívá otevřených dat, jež jsem popsal v kapitole 3. Aplikace zobrazuje na mapě polohu sběrných míst odpadu a škol na území městské části Praha 8, uživatel může zobrazit jejich detailní informace a využít jednoduchých filtrů.

4.1 Použité nástroje

Aplikaci jsem implementoval s ohledem na minimální nároky, samotná webová prezentace využívá pouze HTML a Javascript.

4.1.1 OpenRDF Sesame

Webová aplikace svá data pochopitelně neuchovává otevřená data u sebe v souboru, ale vzdáleně si o ně žádá nějakého poskytovatele dat. Za tímto účelem jsem použil OpenRDF Sesame Server, jeden z produktů z rodiny OpenRDF Sesame zmíněné v části 3.1.4. Jde o repositář, ve kterém jsou nahrány data-sety ve formátu RDF, aplikace k němu přistupuje prostřednictvím protokolu HTTP a dotazovacího jazyka *SPARQL*.

OpenRDF Sesame Server ke svému běhu vyžaduje JRE minimálně verze 8 a Java Servlet Container podporující Java Servlet API 2.5 a Java Server Pages 2.0 nebo novější. Ideálním řešením je Apache Tomcat.

SPARQL SPARQL je (mimo jiné) dotazovací jazyk určený k dotazování RDF repositářů, jeho syntaxe je blízká jazyku SQL.

Příklad dotazu ve výpisu 4.1 vrátí seznam název všech společností typu `prg:RecyclingCentre`.

Výpis 4.1: Příklad dotazu v jazyce SPARQL

```
1 PREFIX prg: <http://opendata.praha.eu/ontology/>
2 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
3 PREFIX schema: <http://schema.org/>
4 SELECT $name WHERE {
5     $rc rdf:type prg:RecyclingCentre.
6     $rc schema:legalName $name.
7 }
```

4.1.2 AngularJS

AngularJS je komplexní webový framework pro Javascript vyvinutý společností Google. Vývojáři nabízí řadu funkcí a vzorů, které vedou k větší čistotě, stručnosti, čitelnosti a celkové udržitelnosti kódu. Mezi další výhody patří vysoká znovupoužitelnost komponent, dependency injection nebo snadné (i oboustranné) provázání HTML a Javascriptu[16].

4.1.3 angular-xml

Angular-xml je drobný modul pro framework AngularJS. Poskytuje interceptor, který odpovědi ve formátu XML na požadavky automaticky převádí na struktury JSON. Tuto funkci využívám, protože odpovědi na dotazy v jazyce SPARQL přicházejí ve formátu *SPARQL Query Results XML Format*[17], tedy speciálně strukturovaném XML dokumentu.

Výpis 4.2: Příklad SPARQL Query Results XML Format

```
1 <?xml version='1.0' encoding='UTF-8'?>
2 <?xml-stylesheet type='text/xsl' href='/openrdf-workbench/
   transformations/tuple.xsl'?>
3 <sparql xmlns='http://www.w3.org/2005/sparql-results#' ... xmlns:
   xsd='http://www.w3.org/2001/XMLSchema#'>
4   <head>
5     <variable name='name'/>
6     <link href='info'/>
7   </head>
8   <results>
9     <result>
10      <binding name='name'>
11        <literal>Atlantik servis, s.r.o.</literal>
12      </binding>
13    </result>
14    ...
15    <result>
16      <binding name='name'>
17        <literal>Skládka Uhy, spol. s r.o.</literal>
18      </binding>
19    </result>
20  </results>
21 </sparql>
```

Výpis 4.2 odpovídá odpovědi na SPARQL dotaz ve výpisu 4.1, výpis je pochopitelně zkrácený. Odpovědi v tomto formátu jsou uzavřeny v elementu `sparql`, odpovědi na dotaz typu `SELECT` obsahují element `head`, ve kterém je seznam proměnných (sloupců), výsledky dotazu se nacházejí v elementu `results`. Typ hodnoty se rozlišuje typem elementu:

- `<literal>` – literál,
- `<uri>` – URI/IRI,
- `<bnode>` – blank node.

4.1.4 ng-map

Ng-map je knihovna pro framework AngularJS, je určena k integraci Google Maps do aplikace.

4.2 Návrh aplikace

Demonstrativní aplikace je jednoduchá, existuje pouze jedna uživatelská role. Funkční požadavky jsou dva:

1. Získání informací o organizacích.
2. Možnost filtrování.

Požadavek 1 Uživatel pomocí aplikace získá následující informace:

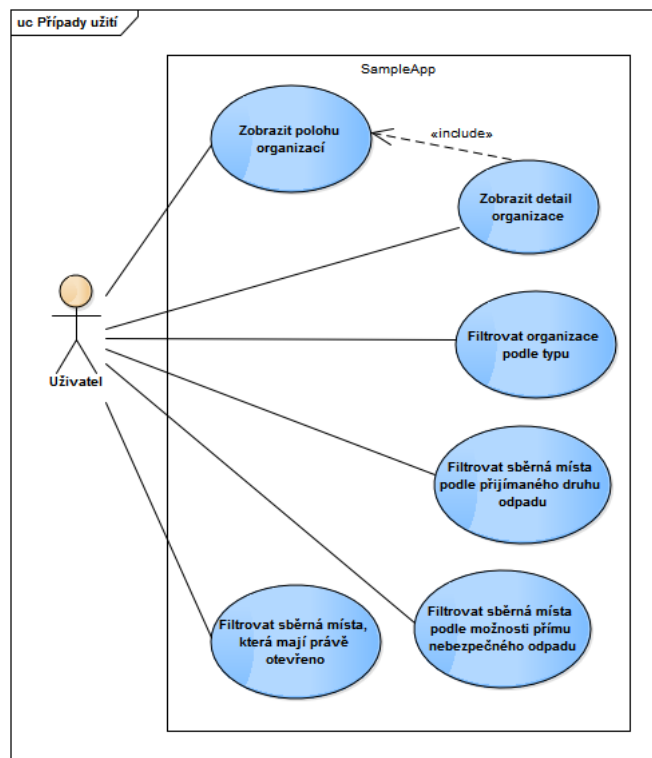
- Sběrné místo – název a IČ subjektu, adresa, poloha, kontakt (telefon, email, webová adresa, identifikátor datové schránky), typ sběrného místa, přijímaný odpad, možnost příjmu nebezpečného odpadu, otevírací doba.
- Mateřská škola – název a IČ subjektu, adresa, poloha.
- Základní škola – název a IČ subjektu, adresa, poloha, kontakt školy, kontakt na ředitele.
- Kombinovaná MŠ a ZŠ – název a IČ subjektu, adresa, poloha, kontakt školy, kontakt na ředitele, kontakt na mateřskou školu.

Požadavek 2 Uživatel může filtrovat zobrazené organizace podle jejich typu, sběrná místa může dále filtrovat podle následujících kritérií:

- Typ přijímaného odpadu,
- možnost příjmu nebezpečného odpadu,
- místa, která mají právě teď otevřeno.

4.2.1 Případy užití

Případy užití aplikace jsou zobrazeny na obrázku 4.1 a popsány v následujících částech.



Obrázek 4.1: Případy užití

4.2.1.1 Zobrazit polohu organizací

Případ užití začíná samotným navštívením stránky aplikace, po jejím načtení se zobrazí mapa, na které jsou vyznačeny všechny organizace pomocí barevně rozlišených špendlíků. Současně se zobrazí všechny ovládací prvky (pro filtrování).

Jde o výchozí stav aplikace, od uživatele není vyžadována žádná interakce, případ užití končí.

4.2.1.2 Zobrazit detail organizace

1. Uživatel na mapě klikne na špendlík organizace, jejíž detailní informace chce zobrazit.
2. Aplikace zavře informační okna jiné organizace, pokud byla otevřená.

3. Aplikace načte informace o organizaci.
4. Aplikace zobrazí načtené informace v informačním okně.

4.2.1.3 Filtrovat organizace podle typu

1. Uživatel v seznamu typů organizací pomocí zaškrtačacích políček vybere ty, které mají být zobrazeny.
2. Aplikace podle zvolených kritérií zobrazí odpovídající organizace.
3. Aplikace aktualizuje volby ostatních filtrů.

4.2.1.4 Filtrovat sběrná místa podle přijímaného druhu odpadu

1. Uživatel v seznamu typů odpadů pomocí zaškrtačacích políček vybere požadované typy odpadu.
2. Aplikace zobrazí sběrná místa, která přijímají všechny zvolené druhy odpadu.
3. Aplikace aktualizuje volby ostatních filtrů.

4.2.1.5 Filtrovat sběrná místa podle možnosti příjmu nebezpečného odpadu

1. Uživatel zaškrtně, zda-li vyžaduje příjem nebezpečného odpadu.
2. Aplikace aktualizuje seznam zobrazených sběrných míst a volby ostatních filtrů.

4.2.1.6 Filtrovat sběrná místa, která mají právě otevřeno

1. Uživatel zaškrtně, zda-li chce sběrná místa omezit pouze na ta, která mají právě otevřeno.
2. Aplikace podle momentálního času a otvíracích dob aktualizuje seznam zobrazených sběrných míst a volby ostatních filtrů.
3. Pokud je zaškrtačací pole zaškrtnuté, aplikace aktualizuje seznam podle momentálního času v minutových intervalech.

Jak je zobrazeno v tabulce 4.1, případy užití pokrývají oba funkční požadavky.

4.3 Implementace

Aplikace se skládá z několika jednoduchých komponent, které jsou popsány v následujících sekcích.

	Požadavek 1	Požadavek 2
Případ užití 4.2.1.1	✓	
Případ užití 4.2.1.2	✓	
Případ užití 4.2.1.3		✓
Případ užití 4.2.1.4		✓
Případ užití 4.2.1.5		✓
Případ užití 4.2.1.6		✓

Tabulka 4.1: Pokrytí požadavků případy užití

4.3.1 SparqlClient

`SparqlClient` je komponenta, která pomocí protokolu HTTP komunikuje se vzdáleným SPARQL endpointem.

Má jedinou metodu `query(sparql)`, která jako vstupní parametr očekává řetězec, ve kterém je dotaz v jazyce SPARQL. Před odesláním dotazu na SPARQL endpoint se ještě předpojí napevno definované jmenné prostory s prefixy, díky čemuž je možné definovat kratší dotazy (s využitím prefixů). Díky knihovně *angular-xml* je odpověď z formátu *SPARQL Query Results XML Format*[17] automaticky převedena na JSON, ta se pro další použití aplikací transformuje a díky asynchronní formě dotazu je vrácena v podobě *promise*.

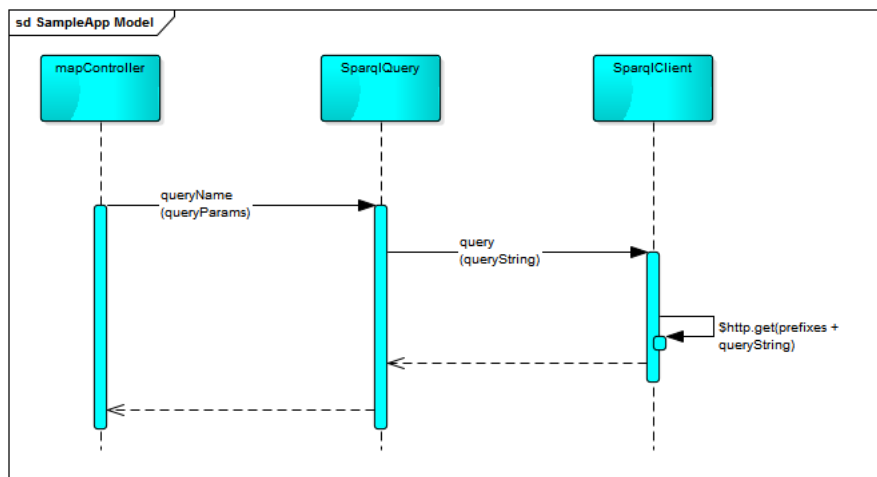
4.3.2 SparqlQuery

`SparqlQuery` je komponenta, ve které jsou definovány konkrétní SPARQL dotazy. Každá metoda komponenty odpovídá jednomu dotazu, který je rovnou proveden pomocí komponenty `SparqlClient`. Návrátová hodnota (*promise*) je bez úprav vrácena dále.

4.3.3 MapController

`MapController` je controller, komponenta, která zprostředkovává propojení mezi HTML a Javascriptem, zpracovává podněty od uživatele, ovládá prvky na stránce.

Komponenta při inicializaci využívá `SparqlQuery` k získání seznamu organizací, při výběru některé z nich volá další dotazy pro zjištění detailních vlastností. Na starosti má také veškeré filtrování. Mechanismus dotazování je zobrazen v sekvenčním diagramu na obrázku 4.2.



Obrázek 4.2: Komunikace komponent aplikace

Závěr

Při psaní této práce jsem se ponořil do problematiky otevřených dat a praxí ověřil předchozí znalosti z předmětu MI-SWE a dalších.

Čtyři různé datasety ve formátu CSV jsem úspěšně transformoval do dvou ve formátu RDF, oba nahrál do RDF repozitáře a zpřístupnil pomocí SPARQL endpointu. Jako praktický příklad využití otevřených dat jsem vytvořil aplikaci schopnou vizualizace dat v repozitáři. Přestože aplikace ukazuje většinu informací v datasetech, jejich plný potenciál jsem nevyužil.

Protože jsem neměl přístup k IT infrastruktuře městského úřadu, nerealizoval jsem trvalé řešení publikace dat a dočasné řešení si zjednodušil použitím Groovy ve skriptech. V opačném případě bych zvolil použití zkompilevaného Groovy kódu, čímž by pro spuštění nebyla vyžadována instalace Groovy a stálé připojení k Internetu.

Jasným ponaučením pro mne je, že (alespoň v České republice) technické problémy jsou v tomto oboru ty nejmenší. I přes to, že jsem neměl přístup k IT infrastruktuře a datům městského úřadu, podařilo se mi překážky obejít a odvést většinu práce (pravděpodobně) tak, jako bych přístup měl.

Literatura

- [1] Department for Communities and Local Government: Local Government Transparency Code 2015. únor 2015, [Online; přístup 4. května 2016]. Dostupné z: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/408386/150227_PUBLICATION_Final_LGTC_2015.pdf
- [2] Hausenblas, M.: 5 ★ Open Data. leden 2012, [Online; přístup 19. dubna 2016]. Dostupné z: <http://5stardata.info/en>
- [3] W3C: RDF Primer. únor 2004, [Online; přístup 22. dubna 2016]. Dostupné z: <https://www.w3.org/TR/2004/REC-rdf-primer-20040210>
- [4] W3C: The RDF Data Cube Vocabulary. leden 2014, [Online; přístup 2. května 2016]. Dostupné z: <https://www.w3.org/TR/vocab-data-cube>
- [5] W3C: RDF 1.1 Primer. červen 2014, [Online; přístup 22. dubna 2016]. Dostupné z: <https://www.w3.org/TR/rdf11-primer>
- [6] Open Knowledge: Open Data Handbook: What is Open Data? [Online; přístup 19. dubna 2016]. Dostupné z: <http://opendatahandbook.org/guide/en/what-is-open-data>
- [7] Opendata.cz: Proč otevřená data? 2016, [Online; přístup 19. dubna 2016]. Dostupné z: <http://opendata.cz/cs/node/29>
- [8] ČESKO: Zákon č.106/1999 Sb. ze dne 11.května 1999 o svobodném přístupu k informacím. In *Sbírka zákonů České republiky*, 1999, s. 2578–2582, [Online; přístup 20. dubna 2016]. Dostupné z: <https://portal.gov.cz/app/zakony/download?idBiblio=47807&nr=106~2F1999~20Sb.&ft=pdf>
- [9] eaves.ca: Case Study: How Open data saved Canada \$3.2 Billion. duben 2010, [Online; přístup 20. dubna 2016]. Dostupné z: <https://eaves.ca/2010/04/14/case-study-open-data-and-the-public-purse>

- [10] London Datastore: Data for London – A City Data Strategy for London. 2016, [Online; přístup 20. dubna 2016]. Dostupné z: <http://data.london.gov.uk/data-for-london>
- [11] Cabinet Office: Open Government Partnership UK National Action Plan 2013 to 2015. červen 2013, [Online; přístup 3. května 2016]. Dostupné z: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/255901/ogp_uknationalactionplan.pdf
- [12] Oddělení boje s korupcí Úřadu vlády České republiky: Akční plán České republiky Partnerství pro otevřené vládnutí na období let 2014 až 2016. říjen 2014, [Online; přístup 3. května 2016]. Dostupné z: <http://www.opengovpartnership.org/sites/default/files/Czech%20Republic%20-%20Second%20Action%20Plan%2C%202014%20-%202016.pdf>
- [13] rdf4j.org: OpenRDF Sesame homepage. 2016, [Online; přístup 24. dubna 2016]. Dostupné z: <http://rdf4j.org>
- [14] schema.org: About schema.org. 2016, [Online; přístup 24. dubna 2016]. Dostupné z: <http://schema.org/docs/about.html>
- [15] Dublin Core Metadata Initiative: Dublin Core Metadata Initiative homepage. 2016, [Online; přístup 24. dubna 2016]. Dostupné z: <http://dublincore.org>
- [16] AngularJS contributors: AngularJS on GitHub. 2016, [Online; přístup 8. května 2016]. Dostupné z: <https://github.com/angular/angular.js>
- [17] W3C: SPARQL Query Results XML Format. březen 2013, [Online; přístup 7. května 2016]. Dostupné z: <https://www.w3.org/TR/rdf-sparql-XMLres>

Seznam použitých zkratk

- BIS** Bank for International Settlements
- DCMI** Dublin Core Metadata Initiative
- ECB** European Central Bank
- GUI** Graphical user interface
- IČ** Identifikační číslo
- IMF** International Monetary Fund
- IRI** International Resource Identifier
- JRE** Java Runtime Environment
- JVM** Java Virtual Machine
- MČ** Městská část
- OECD** Organisation for Economic Co-Operation and Development
- OGP** Open Government Partnership
- OSN** Organizace spojených národů
- OWL** Web Ontology Language
- RDF** Resource Description Framework
- SDMX** Statistical Data and Metadata Exchange
- SPARQL** SPARQL Protocol and RDF Query Language
- URI** Unique Resource Identifier
- XML** Extensible markup language

Obsah přiloženého CD

/	
└─ dist	adresář se spustitelnou formou implementace
└─ apache-tomcat.....	Apache Tomcat s OpenRDF Sesame Serverem
└─ input	adresář se vstupními datasety ve formátu CSV
└─ output.....	adresář s výstupními datasety ve formátu RDF (Turtle)
└─ readme.txt	stručný popis obsahu CD
└─ thesis	
└─ images.....	obrázky použité v této práci
└─ DP_Kunte_Matej_2016.bib.....	bibliografie
└─ DP_Kunte_Matej_2016.pdf	text práce ve formátu PDF
└─ DP_Kunte_Matej_2016.tex	zdrojová forma práce ve formátu L ^A T _E X