



Online mapová aplikace pro zobrazování dat o životním prostředí

Bakalářská práce

Studijní program: N2612 – Elektrotechnika a informatika

Studijní obor: 2612R011 – Elektronické informační a řídicí systémy

Autor práce: **Vojtěch Hájek**

Vedoucí práce: Mgr. Kamil Nešetřil, Ph.D.





TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies ■

Online map application for visualisation of environmental data

Bachelor thesis

Study programme: B2612 – Electrical Engineering and Informatics

Study branch: 2612R011 – Electronic Information and Control Systems

Author: **Vojtěch Hájek**

Supervisor: Mgr. Kamil Nešetřil, Ph.D.



ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Vojtěch Hájek**
Osobní číslo: **M15000091**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Elektronické informační a řídicí systémy**
Název tématu: **Online mapová aplikace pro zobrazování dat o životním prostředí**
Zadávající katedra: **Ústav mechatroniky a technické informatiky**

Z á s a d y p r o v y p r a c o v á n í :

1. Seznamte se s platformou Pentaho, konkrétně pak s komponentami Community Dashboard Editor a Community Dashboard Framework.
2. Navrhněte a implementujte online aplikaci pro zobrazování měření v mapě na platformě Pentaho s využitím Community Tools, případně i s využitím dalších javascriptových knihoven. Využijte abstraktní vrstvu pro přístup k databázi (Pentaho Metadata).
3. Vyhodnoňte vhodnost využití Community Tools pro tvorbu aplikace v prostředí Pentaho.

Rozsah grafických prací: dle potřeby dokumentace

Rozsah pracovní zprávy: 30–40 stran

Forma zpracování bakalářské práce: tištěná/elektronická

Seznam odborné literatury:

- [1] BOUMAN, Roland L. a Jos van DONGEN, 2009. Pentaho solutions: Business intelligence and data warehousing with Pentaho and MySQL. Indianapolis: Wiley. ISBN 978-0-470-48432-6.
- [2] GASPAR, Miguel, 2016. Learning Pentaho CTools?: Acquire finesse with CTools features and build rich and custom analytics solutions using Pentaho. Birmingham ? Mumbai: Packt Publishing. ISBN 978-1-78528-342-0.
- [3] ŽÁRA, Ondřej, 2015. JavaScript: programátorské techniky a webové technologie. COMPUTER PRESS. ISBN 978-80-251-4573-9. Dostupné z: <http://knihy.cpress.cz/javascript-d6.html>
- [4] www.dataearth.cz

Vedoucí bakalářské práce: **Mgr. Kamil Nešetřil, Ph.D.**


Ústav mechatroniky a technické informatiky

Datum zadání bakalářské práce: **10. října 2017**

Termín odevzdání bakalářské práce: **14. května 2018**


prof. Ing. Zdeněk Plíva, Ph.D.
děkan

L.S.


doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 10. října 2017

Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 11.5.2018

Podpis:



Abstrakt

Cílem této bakalářské práce je vytvořit webovou aplikaci, která zobrazuje údaje o geologických objektech uložené v databázi PostgreSQL, ke které se přistupuje přes abstraktní vrstvu konkrétně Pentaho Metadata. Součástí práce je zhodnotit využití Community Tools pro tvorbu webových aplikací. Dále je zde popsán samotný postup vypracování aplikace. Aplikace je vytvořena na platformě Pentaho (Community Edition).

Klíčová slova:

Pentaho, PostgreSQL, Webová aplikace, dashboard, OpenLayers, CDE, CDF, metadata, CanvasJS

Abstract

Purpose of this bachelor thesis is to create a web application, which displays data stored in PostgreSQL database accessed through the abstract layer in other words using Pentaho Metadata. Part of the thesis is an evaluation the use of Community Tools for creating web application. The thesis also shows process of development. Application was created in Pentaho Platform, (Community Edition).

Key words:

Pentaho, PostgreSQL, Web application, dashboard, OpenLayers, CDE, CDF, metadata, CanvasJS

Obsah

Obsah	7
Seznam obrázků.....	9
Seznam zdrojových kódů.....	9
Zkratky.....	10
1 Úvod.....	11
2 Pentaho Platform.....	12
2.1 Business Analytics	13
2.2 Administrace systému	14
2.3 Souborový systém	15
2.4 Marketplace.....	16
2.5 Pentaho Community Edition	16
2.5.1 Community Dashboard Editor (CDE)	17
2.5.2 Community Dashboard Framework (CDF)	18
2.6 Pentaho Metadata	19
2.6.1 Fyzická vrstva	19
2.6.2 Abstraktní Business vrstva.....	20
2.6.3 Business pohled	20
2.6.4 Pentaho Metadata Editor.....	21
3 Mapová knihovna	22
4 Cíl práce.....	23
5 Řešení.....	25
5.1 Rozvržení	26
5.2 Datové zdroje	26
5.3 Abstraktní vrstva	27
5.4 Community Data Access.....	27
5.5 Community Tools.....	28

5.6	Mapa.....	28
5.7	Výběr dat.....	29
5.8	Graf	30
5.9	Ladění.....	30
6	Výsledná aplikace	31
6.1	Ovládací panel.....	32
6.2	Zobrazený graf a porovnání	34
6.3	Mapa a legenda	35
7	Detaily implementace	36
7.1	Komponenty	36
7.2	Kód mapy	37
7.3	Kód grafu	38
7.4	Vytvoření dotazů.....	39
8	Řešené problémy.....	40
8.1	Nedostatek dostupných zdrojů	40
8.2	Předávání parametrů.....	40
8.3	Dotazy na metadata	40
8.4	Tlačítka pro zobrazení dat.....	41
8.5	Využití CSS.....	41
9	Shrnutí práce	42
	Použitá literatura	43
	Příloha na CD.....	44

Seznam obrázků

Obrázek 1: Struktura Business Analytics	13
Obrázek 2: Administrace systému	14
Obrázek 3: Souborivý systém.....	15
Obrázek 4: Ukázka tvorby dashboardu v CDE.....	17
Obrázek 5: Diagram metadata domain	19
Obrázek 6: Pentaho Metadata Editor	21
Obrázek 7: Návrh rozložení	26
Obrázek 8: Schéma posloupnosti komponent.....	29
Obrázek 9: Ukázka výsledné aplikace	31
Obrázek 10: Ukázka tabulky stratigrafie	32
Obrázek 11: Ovládací panel.....	33
Obrázek 12: Rozdíl grafů.....	34
Obrázek 13: Mapa s legendou zobrazující objekty v Mongolsku	35
Obrázek 14: Mapa s legendou zobrazující objekty v ČR	35

Seznam zdrojových kódů

Zdrojový kód 1: Ukázka objektu definující komponentu	36
Zdrojový kód 2: Ukázka vytvoření časové řady	38
Zdrojový kód 3: Ukázka vytvoření časové řady	39

Zkratky

CCC — Community Chart Components

CDA — Community Data Access

CDE — Community Dashboard Editor

CDF — Community Dashboard Framework

CSS — Cascading Style Sheet

CTE — Community Text Editor

HgIS — Hydrogeologický informační systém

MQL — Metadata Query Language

PBA — Pentaho Business Analytics

PCE — Pentaho Community Edition

PME — Pentaho Metadata Editor

S-JTSK — Systém jednotné trigonometrické sítě katastrální

SQL — Structured Query Language

WGS 84 — World Geodetic System

1 Úvod

V rámci informačního systému HgIS [1] jsou sledovány některé složky životního prostředí. Jedná se o veličiny, které se vyskytují ve vodě, jako například hořčík, vápník, železo atd. Data shromážděná z těchto měření jsou uložena v databázi PostgreSQL. Dále je zapotřebí tyto nasbírané údaje vizualizovat s možností porovnání určitých složek, které byly naměřeny na konkrétním místě.

Řešení je mnoho, avšak jsou bez zdrojových kódů, popřípadě i zpoplatněna a jejich řešení neodpovídá přesnému požadavku. Proto je nutné navrhnout individuální řešení, které bude stabilně zobrazovat veškerá požadovaná data.

Táto práce je rozšířením systému HgIS, který využívá platformu Pentaho pro import a export dat, následné uložení dat do datového skladu. Dále systém obsahuje tvorbu zpráv a analýzu dat. Z tohoto důvodu je nutné využít komponentu Pentaho Business Analytics (PBA).

Pro realizaci vizualizace dat je platforma PBA vhodná, jelikož obsahuje komponenty pro shromažďování dat a jejich následné upravování a zobrazování. Zároveň platforma Pentaho umožňuje prostřednictvím programu PME vytvořit abstraktní vrstvu pro přístup k databázi.

Byla vyvinuta funkčně podobná aplikace [2] doplňující systém HgIS, avšak ta je vytvořena v jazyce PHP a běží samostatně. Také je toto řešení nekompatibilní s platformou Pentaho a nelze tudíž využít jejích funkcí. Jelikož se tato aplikace nedá převzít a použít, je zapotřebí vytvořit novou. Výhodou je možnost vytvoření abstraktní vrstvy a popřípadě upravit či doplnit některé funkce aplikace. Pro vývoj této bakalářské práce jsou použity stejné ikony značek v mapě a dotazy v SQL jsou inspirací některých dotazů aplikace.

V rámci předkládané práce byla vyvinuta aplikace umožňující zobrazovat data v mapě, tabulce a grafu s využitím platformy Pentaho. Ve druhé kapitole je tato platforma obecně popsána a s využitím těchto informací je technicky zformulován cíl této práce, který je uveden v kapitole čtyři.

2 Pentaho Platform

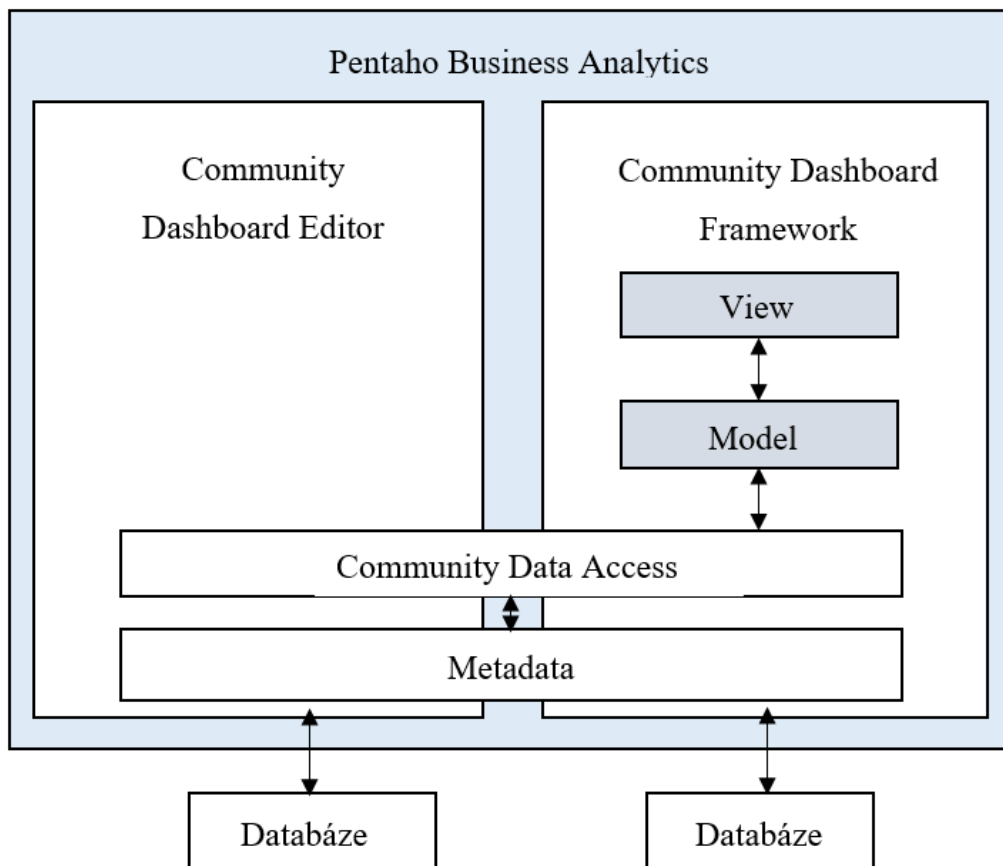
Aplikace je vyvinuta na platformě Pentaho Community Edition [3], jelikož je to opensource program, a to umožňuje bezplatné vytváření přehledových zobrazení (*dashboards*) a reportů.

Pentaho je platforma, určené ke vkládání, manipulaci a následného zobrazení a analyzování dat, která jsou uložena ve vlastním úložišti. Komponenty platformy jsou vhodné i pro uživatele, kteří nemají žádné zkušenosti s programováním, avšak pro efektivnější a komplexnější aplikace jsou zapotřebí znalosti SQL, JavaScriptu, apod. Pentaho se skládá ze tří základních komponent, Business Analytics, Data Integration a Reporting.

Pentaho Data Integration [10] – ELT – poskytuje transformování, načítání dat ve formátu, který je přístupný, pochopitelný a podstatný pro konečného uživatele. PDI je komponenta, která shromažďuje informace z vícero databází a ukládá je do datového skladu.

2.1 Business Analytics

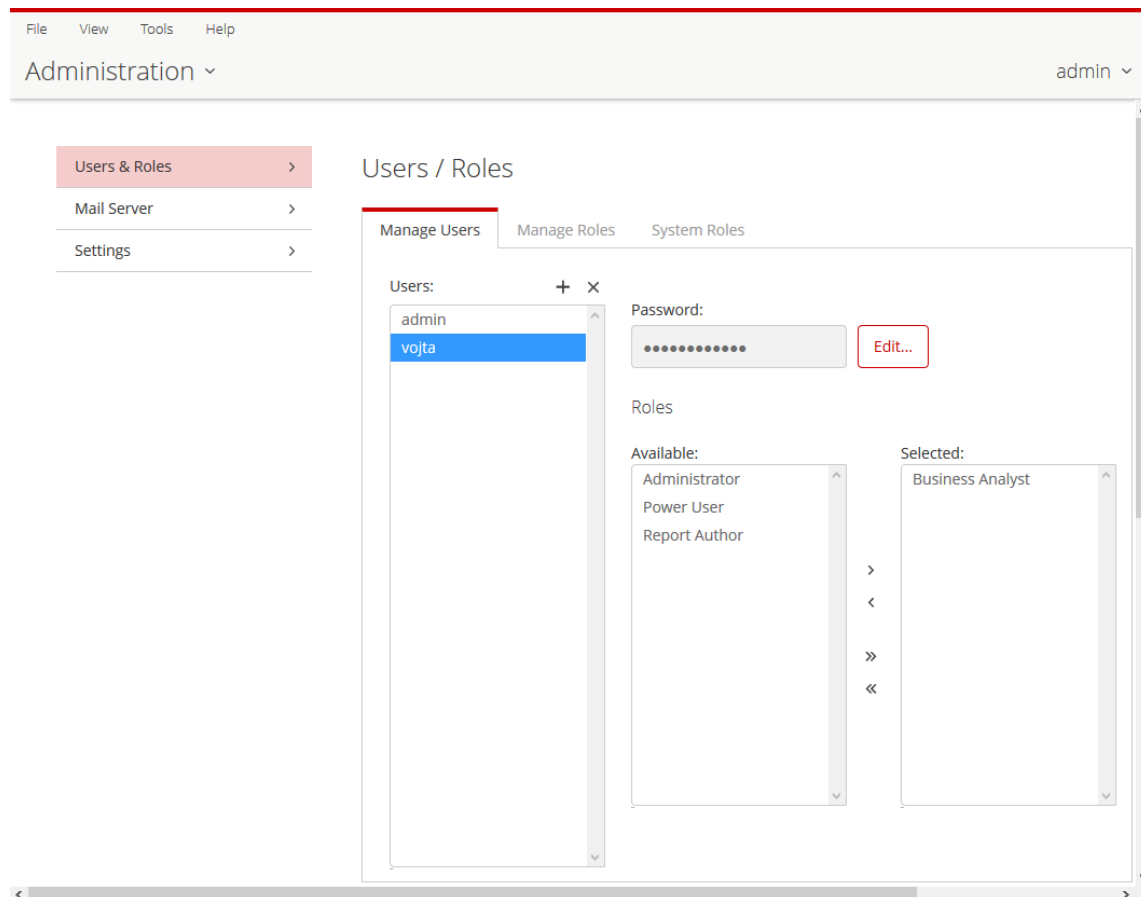
Pentaho Business Analytics, zkráceně PBA, je komponenta určená pro vytváření vizualizací, zpráv (*reports*) a dashboardů založených na navrženém datovém modelu. Na obrázku 1 je znázorněno blokové schéma PBA. Schéma zobrazuje komponenty PBA, které jsou významné pro tuto práci. Tato sada aplikací obsahuje mnoho dalších komponent umožňující práci s daty. Více o struktuře platformy Pentaho se lze dočíst na [1] webových stránkách.



Obrázek 1: Struktura Business Analytics

2.2 Administrace systému

Důležitou součástí PBA je možnost administrace systému. V ní může administrátor, či jiný, jím pověřený, uživatel přidělovat role oprávnění všem registrovaným. Navíc, každé roli může měnit přístupová práva obecně pro celý systém. V případě potřeby je možné měnit práva pro soubory či složky jednotlivě. Rozložení administrace je znázorněno na obrázku 2.

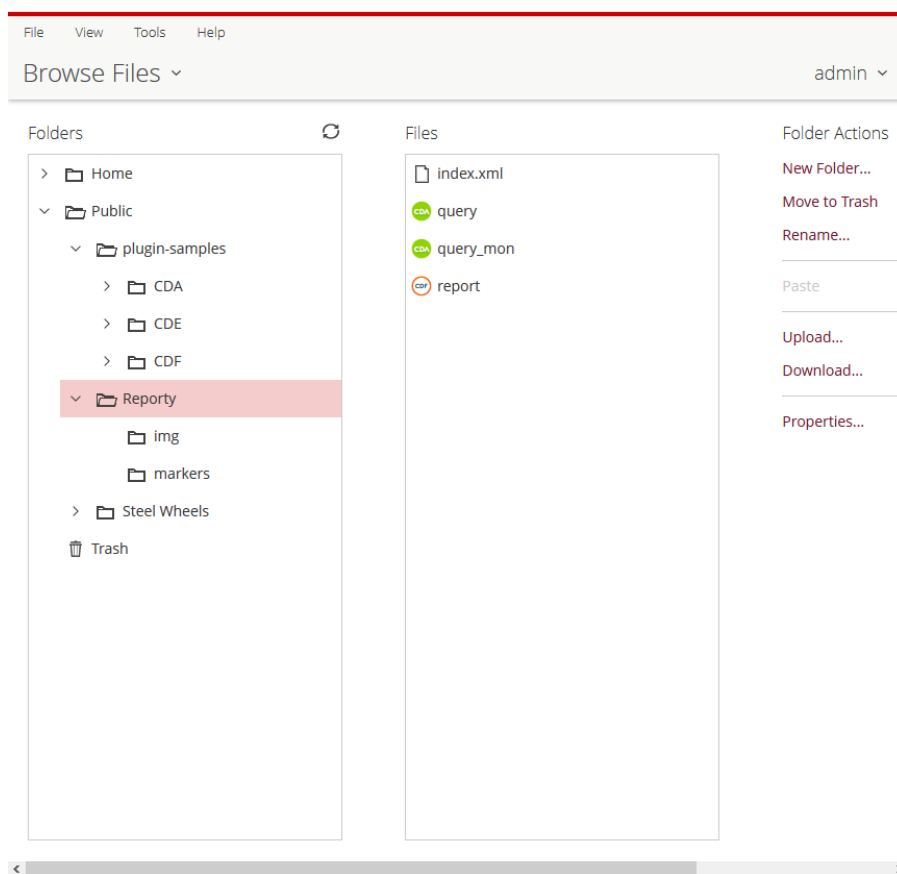


Obrázek 2: Administrace systému

2.3 Souborový systém

PBA obsahuje repozitář Jackrabbit, který umožňuje sdílet soubory a složky všech komponent platformy. Pentaho repozitář je prostředí pro spolupráci analýzy dat a vývoje ELT (*Extract, Transform and Load*) procesů. Jeho prostřednictvím je možné vytvářet složky, nahrávat či stahovat soubory, mazat je nebo jednoduše přejmenovat. V těchto složkách se nacházejí vytvořené dashboardy, transformace, reporty apod.

Jednotlivé soubory lze také kopírovat a následně vkládat do jiných složek. V možnosti vlastností daného souboru či složky je, mimo zobrazení základních údajů o daném objektu, dále možné nastavit sdílení přístupu pro každého uživatele zvlášť a přidat mu klasické oprávnění *Read, Write, Delete*. Souborový systém je ukázán na obrázku 3.



Obrázek 3: Souborový systém

2.4 Marketplace

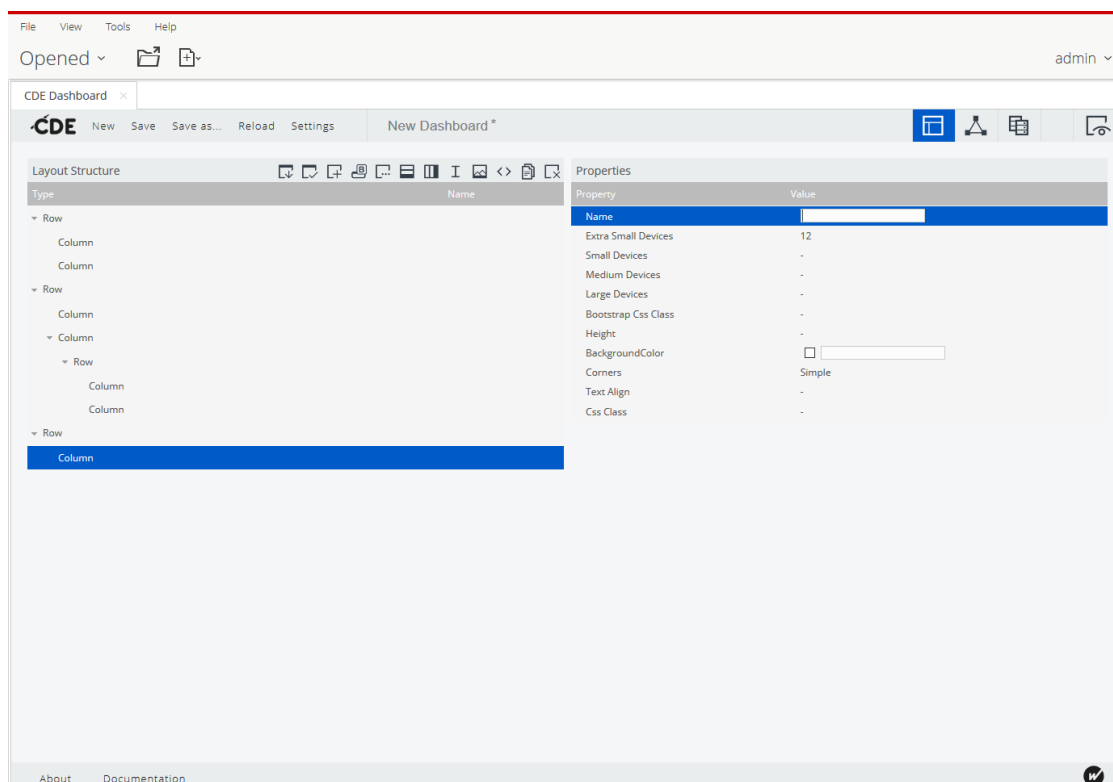
Pentaho Marketplace je plugin ve kterém si vývojář může najít potřebná rozšíření, která mu během vývoje aplikace ulehčí práci. Jedná se například o Community Text Editor (CTE), s jehož pomocí lze upravovat soubory přímo v prostředí Pentaho a nemusí se po každém rozšíření obsahu znovu nahrávat do souborového systému PBA.

2.5 Pentaho Community Edition

Pentaho Open Source BI Suite Community Edition (CE) je open source projekt, který obsahuje ETL, OLAP analýzy, metadata, data mining, reporty a dashboard. Rozdíl mezi CE a Pentaho Enterprise je samozřejmě v ceně, ale také ve funkcích, které poskytují. Dashboard je webová aplikace vytvořená určená pro práci s daty. Tento obsah se vytváří pomocí jedním ze dvou nástrojů (CDE a CDF), kde každý disponuje odlišnými výhodami.

2.5.1 Community Dashboard Editor (CDE)

CDE je grafický nástroj pro vytváření a editaci dashboardů. Tuto možnost může zvolit uživatel, či programátor, který nemá zkušenosti s programovacími jazyky. Veškeré komponenty, jako jsou tabulky, tlačítka, selectboxy, apod. spolu s připojování k databázi jsou vytvářeny pomocí formuláře. Uživatel vybere danou komponentu, pojmenuje ji, a přiřadí k ní spojení s databází. Zde je zapotřebí znalost SQL dotazů pro vybrání žádaných dat. Dále CDE nabízí možnost externího vložení skriptů, souborů HTML s textem či obrázky. Důležitou součástí je možnost náhledu, pro testování funkčnosti projektu. Na obrázku 4 je zobrazeno prostředí CDE umožňující vytvoření stejné aplikace.



Obrázek 4: Ukázka tvorby dashboardu v CDE

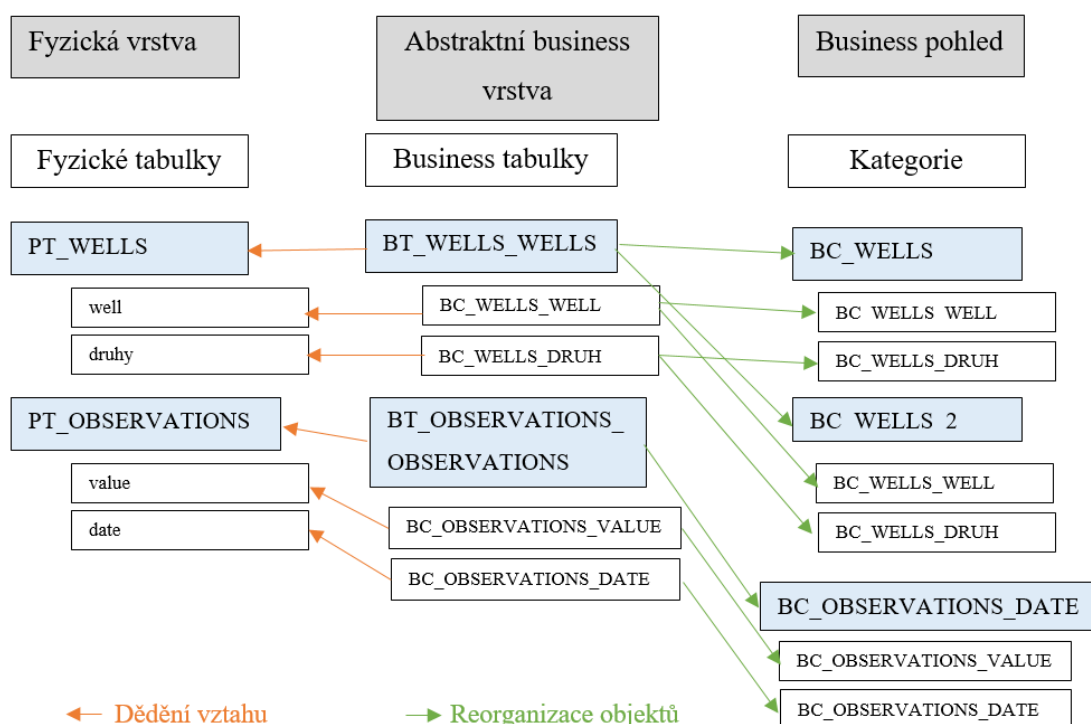
2.5.2 Community Dashboard Framework (CDF)

CDF poskytuje oproti CDE [5] mnohem komplexnější řešení tvorby přehledových zobrazení (*dashboards*). Programátor musí znát základy jazyků JavaScript, SQL, XML a CSS. Obsah, jako jsou komponenty, atd. se vytváří pomocí objektů definovaných v JavaScriptu. Pomocí atributů v těchto objektech lze definovat požadované vlastnosti. Výhodou oproti CDE jsou mnohonásobně větší možnosti, avšak je zapotřebí detailní seznámení s frameworkem a strukturou programu.

Projekt je rozdělen na několik souborů. Na soubor typu `.html`, což je hlavní stránka, kde je veškerý kód potřebný pro zobrazení webové stránky. Poté soubor `.xcdf` obsahující údaje ohledně kódování stránky, jméno autora, název projektu, popis a odkaz na soubor `.html`. Soubor `index.xml` definuje vzhled dashboardu. K této sadě souborů je možno přidat spojení a dotazy k databázi (`.cda`). Samozřejmostí jsou externí soubory vzhledu (CSS) a nebo JavaScriptu (JS). Všechny výše uvedené soubory jsou navzájem propojeny prostřednictvím `template.html`.

2.6 Pentaho Metadata

Pentaho Metadata Model [4] mapuje fyzickou strukturu databáze do logického business modelu (viz obrázek 5). Toto mapování je uloženo v metadatovém repozitáři a umožňuje administrátorovi nastavit parametry zabezpečení pro přístup uživatelů k datům, snížit počet potřebných změn při úpravě fyzické databáze, určit formátování textu, dat a čísel pro zjednodušení údržbu reportů a vytvářet definice pro databázové tabulky.



Obrázek 5: Diagram metadata domain

V diagramu jsou znázorněny dva typy vztahů. Vztah „dědění“ je mezi dvěma různými business objekty, kde jeden dědí metadata od druhého s možností přepsání některých zděděných vlastností. Příkladem toho je business tabulka, která dědí metadata z přidružené fyzické tabulky. Druhý stav „reorganizace“, je ten, kde dva business objekty jsou ve skutečnosti totéž, pouze zastoupené v jiném uspořádání nebo je objekt duplikován pro další použití.

2.6.1 Fyzická vrstva

Fyzická vrstva zahrnuje připojení, fyzické tabulky a sloupce. Tyto objekty představují databázi. Fyzická vrstva není považována za součást business modelu, jelikož v každém business modelu nejsou použita všechna připojení fyzické vrstvy. V každé metadata doméně může být pouze jedno fyzické připojení (databáze).

2.6.2 Abstraktní Business vrstva

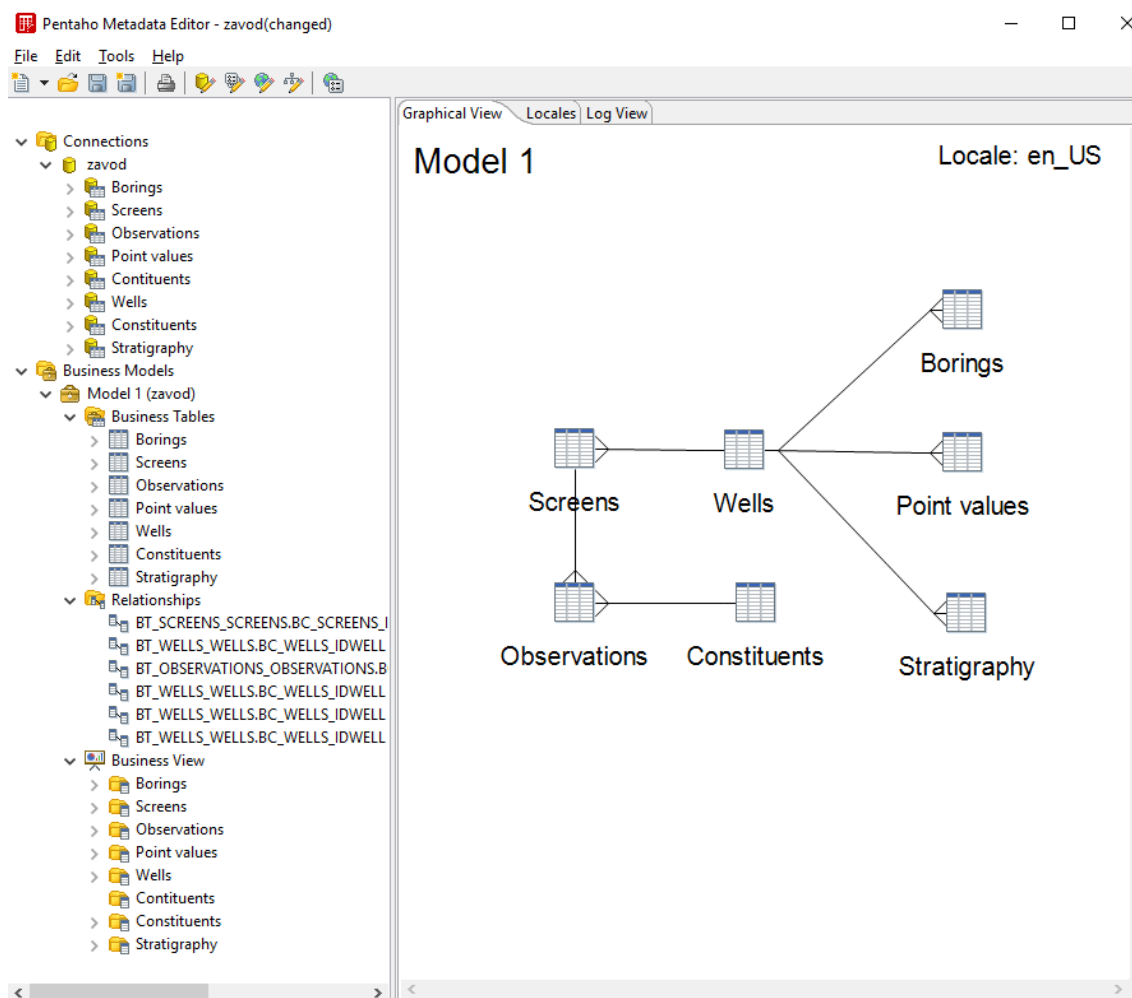
Tato vrstva obsahuje tabulky odkazující se na libovolné tabulky definované ve fyzické vrstvě. Zde lze vytvořit duplikované objekty, což se využívá z hlediska zabezpečení. To znamená, že nelze připojit jeden model na více připojení. Tento problém se dá vyřešit úpravou výsledného XML souboru.

2.6.3 Business pohled

Business pohled je část modelu, se kterou komunikuje aplikace a vidí ji uživatel. Lze zde vytvořit kategorie, které dědí své vlastnosti z business tabulek. Slouží k reorganizaci jednotlivých sloupců tak, aby uživateli dávaly smysl. Jediné omezení je neschopnost přidání dvou stejných sloupců do jedné kategorie.

2.6.4 Pentaho Metadata Editor

Jedná se o program určený k tvorbě metadata domén (*domains*) a data modelů a zjednodušení práce s reporty. Umožňuje vytvářet nové sloupce abstraktních tabulek a definovat agregace libovolných sloupců. V PME verze 8 lze nastavit více agregačních funkcí najednou pro jeden daný sloupec. Po nastavení agregací lze v Community Charts Components (CCC) nastavovat zobrazení dat dle vybraných seskupení. Dále disponuje funkcemi, které omezí výběr dat z dotazů přehledového zobrazení. Schéma databáze včetně vztahů je zobrazeno na obrázku 6.



Obrázek 6: Pentaho Metadata Editor

3 Mapová knihovna

Jednotlivé pozorované objekty jsou v databázi uloženy spolu se souřadnicemi, které je zapotřebí zobrazit na mapě. Každý typ objektu může obsahovat jiný typ měřených údajů, tudíž je zapotřebí tyto objekty od sebe odlišit.

Existuje mnoho řešení jak zobrazit tyto údaje. Všechna jsou definována jako javascriptová knihovna, což umožňuje následné úpravy pro individuální nastavení. Všechna řešení poskytují stejné funkce, avšak různé provedení. Obecně platí, že jakákoliv funkce se dá zhotovit s využitím jakékoli mapové knihovny. Liší se pouze rozsah (v kódu), který je pro tuto funkci zapotřebí.

Při vyžití Community Tools (CTools) je k dispozici mapová komponenta, která využívá knihovnu OpenLayers [6]. Komponenta využívá verzi 2 javascriptové knihovny a vývojáři nemají v plánu aktualizaci na nejnovější verzi.

4 Cíl práce

Pro programování přehledových zobrazení pomocí CDF je zapotřebí ovládat programovací jazyk JavaScript, a znát jeho využití ve frameworku, s jeho pomocí se vytvářejí veškeré komponenty a propojují se jednotlivé objekty.

Hlavním cílem je zhotovit webovou aplikaci pomocí CDF, která bude sledovat naměřená data veličin jednotlivých sledovaných objektů, která jsou uložena v databázi. Hlavní částí zobrazení bude mapa vykreslující umístění vrtů pomocí souřadnic, které mohou být ve dvou různých formátech. Je třeba, aby uživatel měl možnost zvolit konkrétní pozorovaný objekt přímo z mapy, která mu ukáže základní informace nebo ze selectboxu, kde je vypsán seznam všech objektů právě vybraných druhů, které jsou na mapě zobrazeny. Tento seznam by měl být schopen filtrovat pomocí komponenty `multiselect`, která obsahuje seznam těchto typů. Mapa by při prvním spuštění aplikace měla zobrazit všechny vrty přehledně, což znamená přizpůsobit přiblížení a vycentrování. Poté umožní uživateli přiblížení, oddálení či posun dle libosti, jelikož zobrazených objektů může být hodně na jednom místě a uživatel si musí být jistý, že vybírá správně.

Pro účely této aplikace existuje mnoho mapových knihoven, avšak do Ctools je integrována pouze mapa OpenLayers 2, která je v současnosti zastaralá a neobsahuje tolik funkcí jako nejnovější knihovna OpenLayers 4. Z tohoto důvodu je výhodnější vložit knihovnu nejnovější verze mapy externě a nepoužívat tak CTools [7] komponentu se starší verzí. Dalším důvodem volby knihovny OpenLayers je obsáhlá dokumentace s množstvím příkladů veškerých funkcí a také možnost využívat různé podkladové vrstvy.

K dispozici jsou dvě samostatné databáze. První s názvem MARE obsahuje geologické objekty, které se nacházejí na území České republiky. Objekty uložené v druhé databázi se nacházejí v Mongolsku.

Objekty uložené v databázi MARE mohou mít své souřadnice popsány pomocí souřadnicového systému S-JTSK kde osa X směřuje na východ a osa Y směřuje na sever nebo v systému WGS 84, kde se poloha určuje pomocí zeměpisné šířky a délky. Pro zobrazení objektů v Mongolsku je zapotřebí použít WGS 84 / UTM zone 48N, což je projekce zobrazující severní polokouli. Objekty musí být vykresleny na topografickém podkladu, který odpovídá danému souřadnicovému systému.

Poté, co bude vybrán pozorovaný vrt, načtou se informace ohledně naměřených veličin z databáze a uživateli se zobrazí další možnosti jak nakládat s daty. Jelikož PBA je určen pro práci s dashboardy, jsou veškeré dotazy a výpisy prováděny bez potřeby opětovného načtení webové stránky, pouze dashboard při získávání dat zobrazí načítací ikonu a znemožní tak uživateli zasahovat, dokud nebudou veškeré informace načteny. Tento proces eliminuje možné chyby, kdy uživatel v průběhu dotazu na databázi změní své požadavky a způsobí tak chybu výpisu.

Následně si uživatel může vybrat určitou veličinu, která byla měřena a časové rozmezí, ve kterém se mají vypsat hodnoty uložené v databázi. Z těchto informací má být aplikace schopna sestavit graf, na kterém bude přehledně zobrazeno, jak se daná veličina měnila v závislosti na čase. Tento graf bude schopen zobrazit dvě veličiny a umožnit tak jejich porovnání.

V případě, že v databázi nejsou uložena žádná data pro právě vybraný geologický objekt, bude část aplikace pro vybírání veličin skryta.

5 Řešení

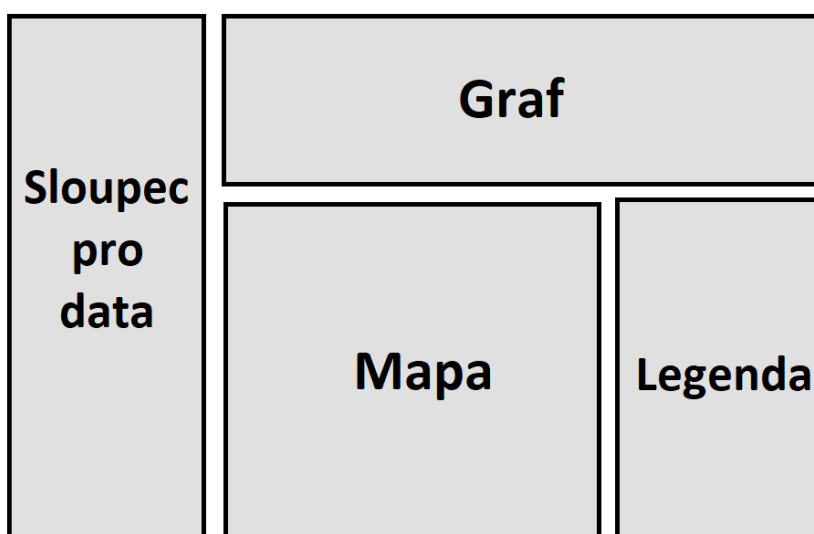
Nejprve je potřeba vytvořit veškeré požadované soubory pro funkčnost aplikace v CDF. Ty musí být vytvořeny externě v prostředí operačního systému a poté importovány na server Pentaho v komprimovaném stavu, který souborový systém rozpozná jako samotný dashboard.

Veškeré programování funkcí aplikace je uskutečněno v souboru `template.html`, který obsahuje strukturu webové stránky a program v jazyce JavaScript. Jelikož tyto soubory jsou uloženy v souborovém systému Pentaho a není možno je upravovat, je nutné po každém rozšíření kódu znovu vše nahrát do prostředí. Tento způsob tvorby by byl velice zdoluhavý a nepříjemný, proto je vhodné prostřednictvím Marketplace nainstalovat doplněk Community Text Editor (CTE), pomocí něhož lze soubory upravovat přímo v úložišti bez nutnosti stálého nahrávání aktualizací.

Pro využití abstraktní vrstvy v aplikaci je zapotřebí propojit soubor CDA obsahující dotazy na databázi s datovým zdrojem vytvořeným pomocí PME. Využité databáze jsou na serveru univerzity, avšak při programování z jiné, než místní adresy se k těmto zdrojům nelze připojit. V těchto případech se využívá VPN, avšak připojení k databázovému serveru na univerzitě je možné pouze přímo z interní sítě. Z tohoto důvodu jsou databáze překopírované na lokální server, který běží na domácím počítači s operačním systémem Windows 10. Databáze MARE zobrazující objekty v České republice je uložena v MySQL databázi, zatímco druhá databáze s objekty v Mongolsku je uložena v PostgreSQL databázi.

5.1 Rozvržení

Hlavní částí je mapa, roztažená přes celou plochu webové stránky. Výběr sledovaného objektu ze selectboxu a následný výběr veličiny či rozmezí dat je umístěn pohromadě v panelu, který je možno zasouvat, vysouvat na levé straně plochy tak, že mapa může být stále přes celou obrazovku. Tento panel také obsahuje výpis naměřených hodnot v tabulkách. Druhým panelem je prostor grafu, který je vytvořen rovněž stylem zasouvání, a to v horní části stránky. Pro zpřehlednění je mapa doplněna legendou umístěnou v pravé části. Přehled rozmístění bloků znázorňuje obrázek 7.



Obrázek 7: Návrh rozložení

5.2 Datové zdroje

Oba datové zdroje jsou si velice podobné, avšak datové modely se nepatrně liší spolu s konvencí pro názvy tabulek a sloupců. Problém této rozdílnosti řeší Pentaho Metadata. Tento rozdíl znamená potřebu vytvořit další dotazy pomocí CDA, které mají tyto názvy upraveny a zároveň v celém kódu programu měnit volání jednotlivých dotazů. Pro vytvoření mapové aplikace není třeba všech tabulek uložených v databázích. Zapotřebí jsou pouze tabulky *wells*, která obsahuje informace o objektech, jako je druh, název, a cizí klíče k tabulce *observations*. Ta obsahuje data k jednotlivým měřením. Dále pak *screens*, určující jednotlivé intervaly měření. Nakonec jsou zapotřebí tabulky *borings*, *fill*, *stratigraphy*, *point_values* a *well_construction*, které obsahují data pro tabulky, které se zobrazí po stisku příslušných tlačítek.

5.3 Abstraktní vrstva

Abstraktní vrstvu je zapotřebí vytvořit v programu PME, kde se nejprve nadefinuje připojení ke skutečné databázi a vyberou se příslušné sloupce a tabulky, které webová aplikace využívá. Při vytvoření nových metadat je důležitou částí určit identifikační jméno daného modelu, které se využívá v jednotlivých dotazech na databázi. Abstraktní vrstva obsahuje identické tabulky, které se dají upravit přidáním dalších sloupců, využitím agregačních funkcí a podobně. V této aplikaci však není třeba definovat žádné další sloupce. Poslední částí vytvoření metadat je určit kategorie s využitím Business pohledu. Výsledný celek se pak pošle na Pentaho server, pomocí funkce zveřejnit na server. Zde se metadata zařadí mezi datové zdroje a přistupuje se k němu podobně jako k databázi.

Na obrázku Obrázek 5 je znázorněné rozdělení tabulky *wells* do dvou kategorií. Toto rozdělení ve výsledné aplikaci není použito, avšak je to další možnost, jak přistoupit k řešení práce. Každá kategorie je rozdělena podle typu souřadnicového systému, tudíž jsou vytvořeny dvě až tři tabulky, které obsahují jednotný formát dat. Toto řešení není použito, jelikož při dalším souřadnicovém systému by bylo zapotřebí upravit nejen kód programu, ale i celý datový zdroj. Výhodou však je rychlejší přístup k datům, jelikož by nebylo třeba filtrovat data podle použitých souřadnic těsně před vyobrazením v mapě.

5.4 Community Data Access

Pro propojení abstraktní vrstvy s aplikací lze využít CDA. Tento soubor obsahuje veškeré dotazy v MQL na metadata, které se programují ve struktuře XML. Rozdíl oproti SQL dotazům je ve struktuře každého dotazu, který je v MQL rozdělen na sekce zobrazení dat v komponentě, omezení výběru a seřazení výsledných řádků v určitém pořadí. Tento typ dotazů je sice přehlednější, ale nedisponuje tolika funkcemi, kterými by se daly omezovat vybíraná data. Tudíž je zapotřebí složitější dotazy řešit například dodatečným výběrem za pomoci JavaScriptu přímo v kódu aplikace nebo při tvorbě abstraktní vrstvy v PME.

5.5 Community Tools

Veškeré komponenty aplikace, jako jsou tlačítka tabulky apod., jsou řešeny za pomoci Community Tools. Tyto objekty jsou vytvořeny v souboru `template.html` a tvoří hlavní část programu. Každá komponenta má nadefinovány parametry, které určují, o jakou komponentu se jedná, v jakém HTML tagu na stránce se vytvoří, který dotaz v CDA je tomuto objektu přiřazen, o jaký datový typ se jedná a mnoho dalších. Každý objekt může obsahovat funkce, které se vykonají před tím a poté, než se komponenta inicializuje nebo, pokud se jedná o ovládací prvek, změní svou hodnotu. V těchto funkcích lze provádět dodatečné úpravy dat.

5.6 Mapa

Knihovna OpenLayers je do projektu vložena externě pomocí odkazů a vytvořeným skriptem pro získání souřadnic z databázové tabulky. Komponenta QueryComponent při prvním načtení dashboardu pošle dotaz na metadata, která vrátí veškeré informace o každém pozorovaném objektu, který je v tabulce uložen. Následně jsou tyto údaje filtrovány a vybírají se pouze ty, které patří objektům odpovídajícím požadavkům uživatele.

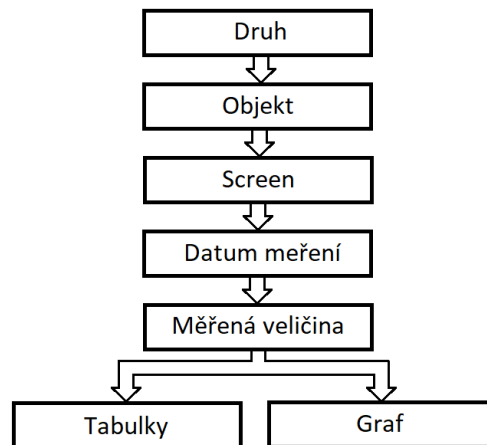
V tabulce se vyskytují dva typy souřadnic. Jeden obsahuje informace o zeměpisné šířce a délce a druhý záporné souřadnice systému S-JTSK/Krovak East North [9]. V případě druhé databáze, jsou souřadnice v jiném systému. Knihovna OpenLayers pracuje s kartézským souřadnicovým systémem, avšak umožňuje transformaci využitím knihovny Proj4js [11] a definovat projekci pomocí řetězce. Tato reprojekce je založena na trojúhelnících, kdy se cílový rastr rozdělí na limitovaný počet trojúhelníků s vrcholy transformovanými pomocí knihovny `o.proj`. Pixely uvnitř trojúhelníků jsou aproximovány lineární transformací mezi vektorovými prostory.

Jelikož je souřadných systémů více, musí se filtrovat tak, aby při uživateli zvoleném souřadnicovém systému se vyobrazily pouze ty, které danou hodnotu mají.

Získané souřadnice jsou poté přidělovány každému zobrazenému bodu na mapě spolu s ikonou značky a základními informacemi o daném objektu. Těmito bod jsou nakonec přiřazeny malé popup tabulky s informacemi o vrtu získanými z databáze. Po najetí kurzoru na značku se tato tabulka zobrazí.

5.7 Výběr dat

Po spuštění aplikace se inicializují veškeré komponenty, ty jsou na sebe navázány pomocí parametru *listeners*, který definuje nadřazený objekt, po jehož změně se znovu dotáže na metadata a opětovně se načte. Jak jsou jednotlivé komponenty na sebe vázány, zobrazuje obrázek 8. Při změně vybraných druhů objektů se přepíše selectbox, který je vypisuje. Zároveň se musí znovu načíst obsah mapy.



Když už je vybrán konkrétní pozorovaný objekt, vyšle se dotaz na databázi pro získání časového rozmezí, ve kterém jsou veškerá data naměřena, což je triviální příkaz, který vrátí nejmenší a největší datum. Ty se zobrazí v selectboxech. Tento způsob výběru data byl zvolen pro svou jednoduchost a rychlost. Jelikož se veličiny nemusely měřit každý den ve stejnou dobu, databáze obsahuje hodnoty k veličinám pouze v určitých dnech. Pokud by byl zvolil výběr dat pomocí kalendáře, funkčnost by zůstala stejná, ale uživatel by neviděl, ve kterých dnech byly hodnoty naměřeny a ve kterých ne. Zároveň si uživatel vybere veličinu, jejíž vývoj chce pozorovat a podle vybraných dat se následně zobrazí graf.

Tabulka *observations* databáze s objekty nacházejícími se v Mongolku obsahuje naměřená data, která však nejsou časově závislá. Ve sloupci pro datum je uložena hodnota „1. 1. 1900“, což je výchozí hodnota. Aplikace data s touto hodnotou v databázi v grafu nezobrazí, ale pouze je vypíše do tabulky.

5.8 Graf

Graf lze vytvořit využitím Community Chart Component (CCC), což je knihovna všech typů grafů. Pomocí CCC lze v CDF vytvořit libovolný graf v závislosti na informacích definovaných v parametrech nastavení komponenty grafu. Vzhledem k naměřeným hodnotám je pro zobrazení ideální spojnicový graf, kde na ose x je čas, jehož maximální a minimální hodnota se mění v závislosti na vybraném rozmezí a na ose y je zvolená veličina respektive veličiny.

Vytvoření CCC grafu je sice jednoduché a vykreslení rychlé, avšak nenabízí uživateli pokročilejší práci se zobrazenými daty, jako je například přiblížení. Z tohoto důvodu je v aplikaci využita externí javascriptová knihovna CanvasJS, která je zaměřena na tvorbu interaktivních grafů. Vývojáři poskytují tuto knihovnu pouze po zakoupení ke komerčnímu využití. Pro studijní účely je zapotřebí vývojáře požádat o nekomerční verzi, kterou poskytnou po podrobném popsání využití knihovny v aplikaci.

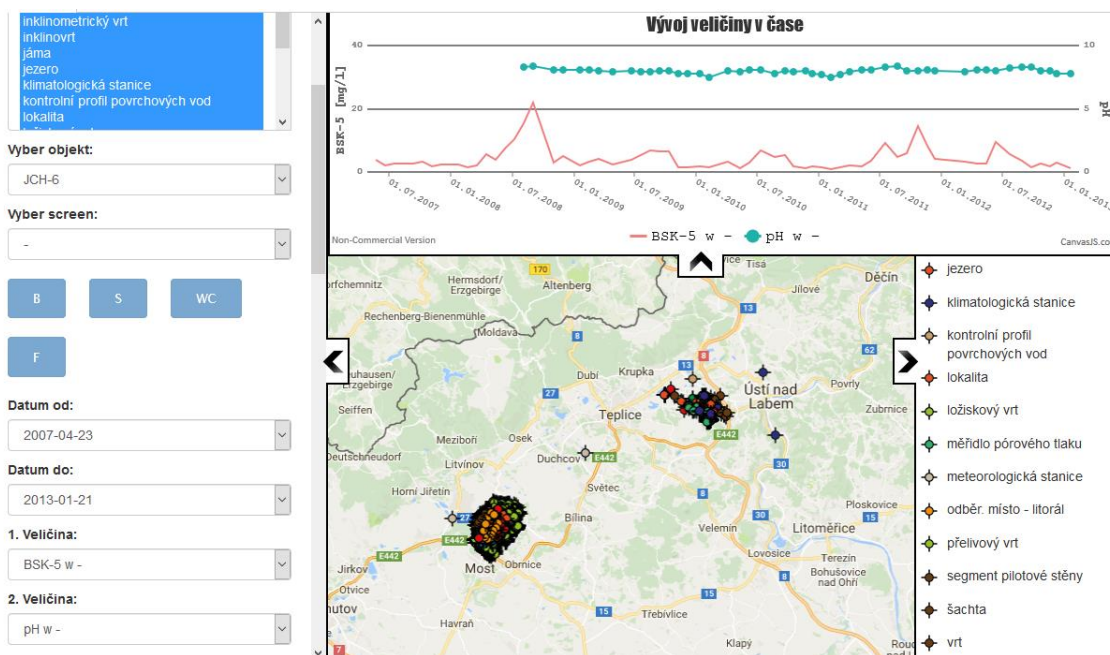
V případě, že by aplikace byla v budoucnu použita komerčně, jsou aktivní obě realizace grafů a výběr mezi nimi lze uskutečnit přepisem jediné proměnné v kódu.

5.9 Ladění

Poslední částí vývoje je vytvoření a následné připojení externích souborů CSS, které přizpůsobí vzhled stránky návrhu a souboru JS. Ty vytvoří dojem dynamičnosti webové aplikace. Vesměs obsahují skripty pro vysunování panelů, či jiné efekty zpříjemňující uživateli práci s dashboardem. Také je vhodné optimalizovat konečný program, jelikož z důvodu některých nedostatků CDF a CTools, se určité operace s daty musí provádět v cyklech, které zabírají paměť a zvyšují dobu zobrazování dat aplikace. Výsledná aplikace zobrazená na obrázku Obrázek 9 se skládá z bloků zmíněných dříve.

6 Výsledná aplikace

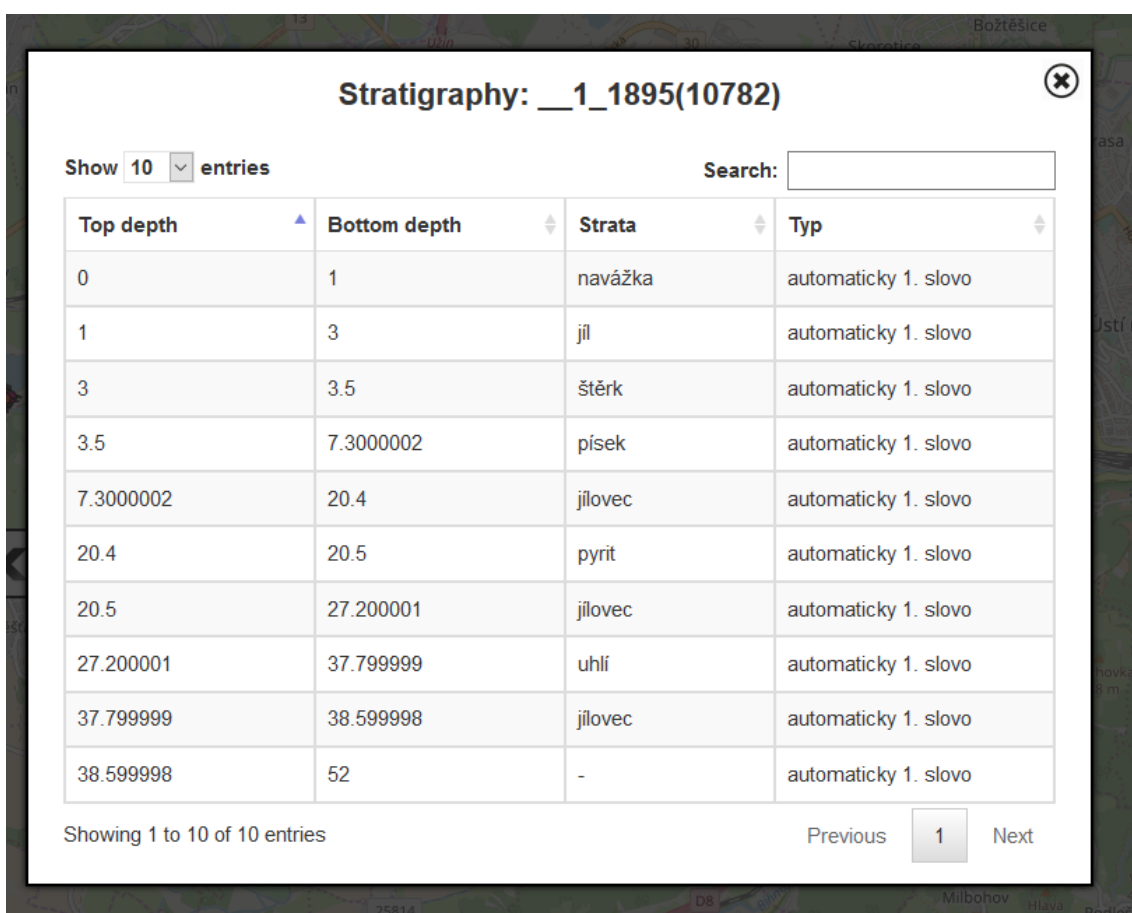
Webová aplikace je naprogramována podle struktury znázorněné na obrázku 1. Data jsou získána z databází, které jsou navázány na abstraktní vrstvu, kde se data definují vztahy jednotlivých tabulek. V závislosti na těchto vztazích pak aplikace posílá dotazy v MQL prostřednictvím CDA, kde jsou definovány. Vybraná data jsou vložena do jednotlivých CTools komponent zobrazující uživateli požadované časové úseky. Vzhled výsledné aplikace lze vidět na obrázku 9.



Obrázek 9: Ukázka výsledné aplikace

6.1 Ovládací panel

Tento panel umožňuje vybrat si, jaký datový zdroj bude použit a pro jaký souřadnicový systém bude dotaz tvořen. Dále obsahuje komponentu `multipleselectbox` pro výběr více druhů objektů zároveň. Tento výběr ovlivňuje `selectbox` níže, ve kterém jsou vypsané objekty, které odpovídají vybraným druhům. Po vybrání konkrétního objektu se zobrazí ovládací část, která obsahuje tlačítka pro výpis tabulek popisu geologických vrstev (tlačítko B), vymezení geologických vrstev (tlačítko S), obsyp a těsnění vrtu (tlačítko F) a výstroj (tlačítko WC). Tyto tlačítka se aktivují, pokud existují nějaká data pro výpis. Po kliknutí na dané tlačítko se zobrazí tabulka v popup okně. Vzhled těchto tabulek znázorňuje obrázek 10.



Stratigraphy: __1_1895(10782)

Show 10 entries Search:

Top depth	Bottom depth	Strata	Typ
0	1	navážka	automaticky 1. slovo
1	3	jíl	automaticky 1. slovo
3	3.5	štěrk	automaticky 1. slovo
3.5	7.3000002	písek	automaticky 1. slovo
7.3000002	20.4	jilovec	automaticky 1. slovo
20.4	20.5	pyrit	automaticky 1. slovo
20.5	27.200001	jilovec	automaticky 1. slovo
27.200001	37.799999	uhlí	automaticky 1. slovo
37.799999	38.599998	jilovec	automaticky 1. slovo
38.599998	52	-	automaticky 1. slovo

Showing 1 to 10 of 10 entries Previous 1 Next

Obrázek 10: Ukázka tabulky stratigrafie

V případě, že vybraný objekt má uložena měřená data, je k dispozici výběr časového rozmezí pomocí komponent selectbox pro výběr dat počátku a konce intervalu výběru informací. Následně je zapotřebí vybrat konkrétní sledovanou veličinu, popřípadě dvě. Vytvořený dotaz následně zobrazí dvě tabulky, které obsahují výpis dat z databáze. Výsledná vzhled panelu se všemi ovládacími prvky je vidět na obrázku 11.

Česko Mongolsko

Vyber druhy objektů:

-
- čerpací stanice důlních vod
- geodetický bod
- inklinometrický vrt
- inklinovrt
- jáma
- jezero
- klimatologická stanice
- kontrolní profil povrchových vod
- lokality

Vyber objekt:

JCH-6

Vyber screen:

-

Datum od:

2007-04-23

Datum do:

2013-01-21

1. Veličina:

BSK-5 w -

2. Veličina:

pH w -

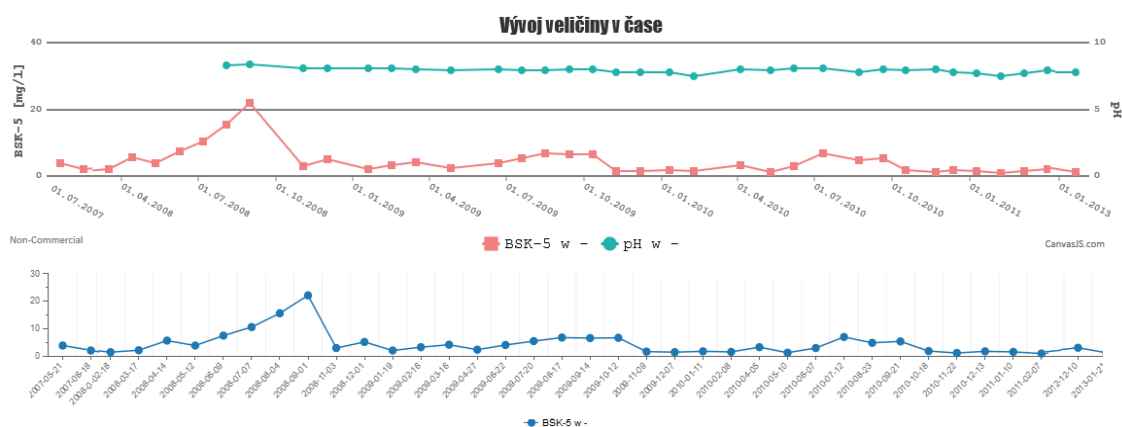
Show entries Search:

Datum	BSK-5 [mg/l]	Poznámka
2007-05-21	3.8	

Obrázek 11: Ovládací panel

6.2 Zobrazený graf a porovnání

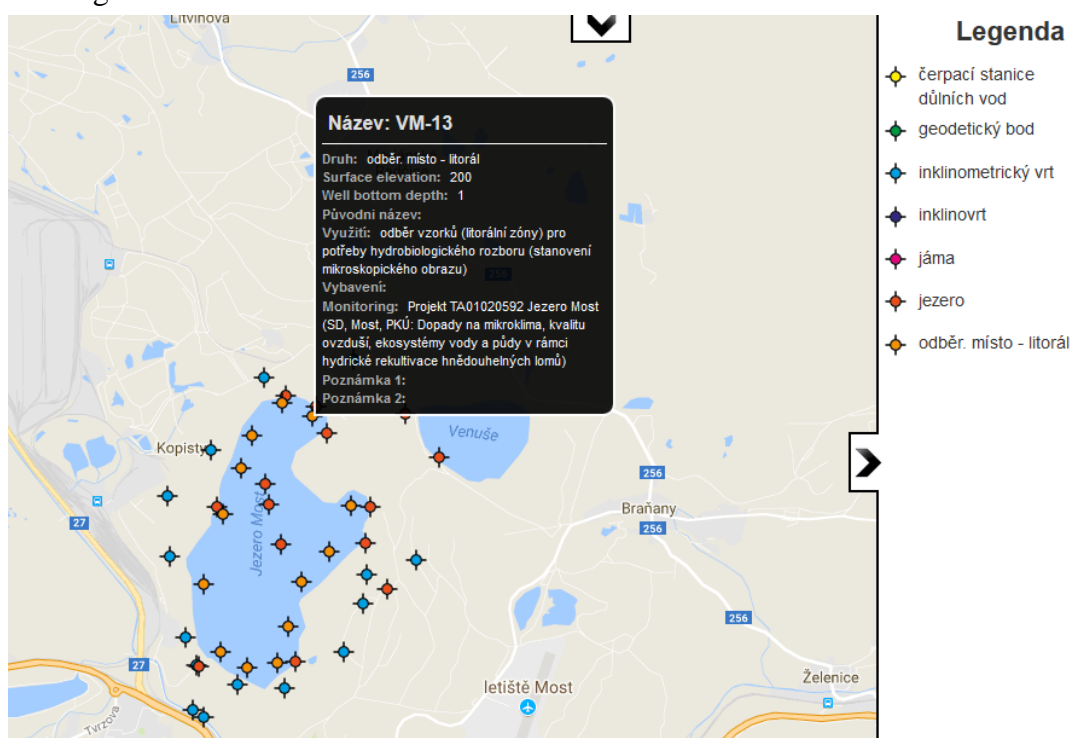
Po drobném zásahu do kódu je možné si zvolit zobrazující se graf naměřených dat. Lze si vybrat mezi grafem CCC a externě vloženým CanvasJS, jehož jediná nevýhoda je využití pouze pro nekomerční účely. Nevýhodou CCC grafu je mimo již zmíněné i absence druhé osy, kterou nelze využít z důvodu absence funkcí MQL ačkoli tato komponenta touto funkcí disponuje. Na obrázku 12 je vidět rozdíl těchto dvou grafů, kde horní část vyobrazuje data knihovna CanvasJS a dolní graf je vytvořen využitím CCC. Časové řady jsou vykresleny po zvolení dvou měřených veličin.



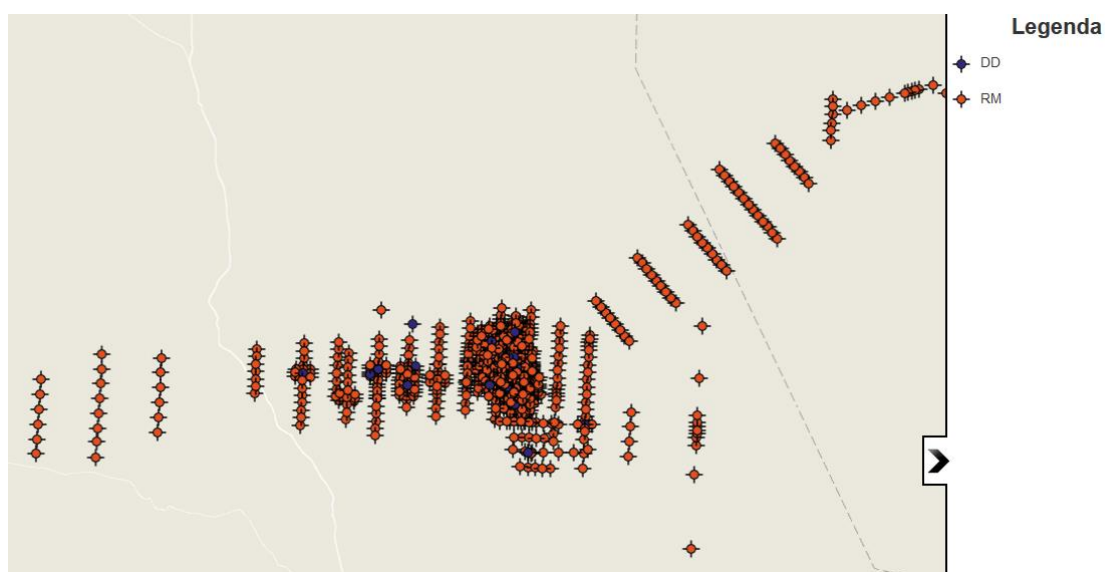
Obrázek 12: Rozdíl grafů

6.3 Mapa a legenda

Výsledná mapa je vytvořena externě s využitím OpenLayers 4. Na mapovém podkladu OpenStreetMap se vykreslí jednotlivé body znázorňující objekty, které byly vyfiltrovány pomocí kontrolního panelu. Legenda na pravé straně aplikace reaguje na změny výběru druhů. Po najetí myši na konkrétní objekt se zobrazí tabulka vypisující informace z databáze za pomoci funkce *Popup*. Obrázek 14 zobrazuje, jak vypadá tento výpis na mapě s legendou. Poté na obrázku 13 je lze vidět vykreslení objektů v Mongolsku z databáze.



Obrázek 14: Mapa s legendou zobrazující objekty v ČR



Obrázek 13: Mapa s legendou zobrazující objekty v Mongolsku

7 Detaily implementace

Kód této aplikace je rozdělen na tři základní části, které jsou reprezentovány metodami. Ty vytvářejí hlavní funkce jako je inicializace všech komponent, mapa a vykreslení grafu CanvasJS. Hlavní tři metody jsou doplňovány funkcemi, které jsou zprostředkovávány pomocnými metodami. Program pak sdílí globální proměnné se všemi částmi kódu.

7.1 Komponenty

Každá komponenta vytváří novou proměnnou, která je využívána k předávání hodnot pro další komponenty a následné dotazy. Například komponenta zobrazující jednotlivé hloubkové intervaly v selectboxu s názvem „screen“ obsahuje několik důležitých parametrů (viz Zdrojový kód 1).

```
screen =
{
  name: "screen_s",
  type: "select",
  parameter: "screen",
  parameters: [['studna', 'studna']],
  listeners: ['studna', 'datasource'],
  priority: 4,
  tooltip: "Vyber screen",
  valueAsId: false,
  htmlObject: "screen_select",
  queryDefinition: {
    path: cesta_cda,
    dataAccessId: dataId[4]
  },
  executeAtStart: false,
  postExecution: function() {
    $('#' + this.htmlObject + ' select:first-
    child').addClass("form-control");
  }
}
```

Zdrojový kód 1: Ukázka objektu definující komponentu

Parametr `type` určuje druh komponenty. Community Tools umožňuje takto vytvořit tabulky, tlačítka, mapa, textové pole apod. Významnou částí je parametr `parameter`, který do dané proměnné vloží hodnotu, která je v dané komponentě vybrána. Tento parametr úzce souvisí s `listeners`. Ten reaguje na změnu hodnoty proměnné, která je v něm nadefinována z ostatních komponent. Jednou z nejdůležitějších parametrů je `parameters`, který předává hodnoty proměnných dotazů v MQL. První index tohoto pole je název parametru v kódu a druhý je název parametru v souboru s dotazy. Pro propojení komponenty s CDA je zapotřebí definovat parametr `queryDefinition`, který obsahuje informaci o absolutní cestě k souboru s dotazy a identifikační jméno

dotazu. Po získání dat se volá funkce `postExecution`, která se vykoná okamžitě po inicializaci komponenty a získání dat. V této funkci je možné manipulovat s výsledným `selectboxem`. V tomto případě stačí pouze přidat elementu parametr `class` pro přidání vzhledu knihovny Bootstrap.

7.2 Kód mapy

Funkce mapy je rozdělena na několik částí. K inicializaci mapy a definici jednotlivých vrstev je využita metoda `Mapa`. Ta obsahuje funkce `OpenLayers` pro vložení správného podkladu a pro správné zobrazení bodů v souřadném systému. Při inicializaci mapy se pomocí podmínky kontroluje, jaký je zvolen souřadnicový systém a dle toho je využita funkce `ol.proj`. Dále je v této metodě definována popup tabulka zobrazující se při najetí kurzorem na objekt a nakonec vycentrování mapy tak, aby mapa vždy zobrazovala všechny značky.

Při prvním načtení mapy a zároveň při každé změně zobrazovaných dat se volá metoda `Body`. Ke správnému fungování této metody je zapotřebí globální proměnné `markers` a jedné komponenty `CTools`, která do této proměnné nahrává všechny objekty uložené v databázi z daného datového zdroje. Z těchto dat jsou ve `for` cyklu vybírány pouze údaje, které odpovídají filtru uživatele. V tomto cyklu se také definuje vzhled každého objektu hlavně tedy ikona, jejíž barva závisí na jeho druhu. Pro volbu správné barvy je vytvořena metoda `spravnyMarker`, která obsahuje `switch case` podmínku s pevně danými barvami ikon.

Pro správnou funkčnost mapy je zapotřebí dvou pomocných metod. Jedná se o metodu `VybranyObjekt`, která při kliknutí na ikonu uloží do proměnné, využívané jako parametr komponenty `well`, identifikační číslo z databáze, což umožňuje přepsání komponent ovládacího panelu tak, aby odpovídal vybranému objektu. Další funkcí této metody je vycentrování mapy pro lepší přehled, kde se daný objekt nachází. K tomu je zapotřebí nejdříve získat souřadnice daného bodu pomocí funkce `getGeometry().getCoordinates`. Druhá metoda s názvem `Legenda` vrací řetězec obsahující ikony s popisky legendy. Obsah této proměnné závisí na vybraných druzích objektů v komponentě `multiselect`. Poté se hodnota pouze vypíše do panelu vpravo.

7.3 Kód grafu

Inicializace grafu probíhá okamžitě při prvním načtení dashboardu, avšak bez dat. Je nutné pouze nadefinovat parametry určující styl grafu (spojnicový), počet os, formát času apod. K vytvoření grafu CanvasJS jsou použity dvě komponenty CTools pro získání dat k zobrazení jednotlivých časových os. Po uložení hodnot do pole `pole_graf` je nutné zavolat metodu `PrepisGrafu` pro nahrání hodnot k zobrazení a přepsání názvu os (viz Zdrojový kód 2).

```
var i = 0;
chart2.options.data[0].dataPoints = [];
if(pole_graf.length > 0)
{
    $(pole_graf).each(function() {
        var from = this[0].split("-");
        var f = new Date(from[0], from[1]-1, from[2]);
        chart2.options.data[0].dataPoints[i] = {x: f, y: this[1]};
        i++;
    });
    chart2.options.data[0].name = $("#velicina select
option:selected").text();
    chart2.options.axisY.title = ($('#table table thead tr
.column1').text());
}

chart2.render();
```

Zdrojový kód 2: Ukázka vytvoření časové řady

Nejprve je zapotřebí smazat předešlé hodnoty a následně vložit nové. Data pro jednotlivé osy jsou uložena v poli, proto je nutné toto pole procházet pomocí cyklu `each` a rozdělit informace příkazem `split` v každém indexu na část s časem a příslušnou hodnotou. Nakonec je zapotřebí změnit název os a zavolat funkci `render` pro vykreslení grafu.

7.4 Vytvoření dotazů

Veškeré dotazy jsou vytvořeny v jazyce MQL pomocí CDA (viz Zdrojový kód 3). K vytvoření každého dotazu je nutné definovat nejprve jazyk dotazu a jedinečný název. Poté v tagu `Parameters` jsou definovány parametry, které jsou předány z jednotlivých komponent do dotazu. Prozatím je k dispozici pouze datový typ `numeric`, `string` a `date`. Název datového zdroje je určen tagem `domain_id`. Dalším tagem je identifikační jméno vytvořeného modelu `model_id`. Následuje blok `selections`, který obsahuje vždy název tabulky `view` a název sloupce `column`. V bloku `constraint` jsou omezení výběru dat. V SQL je tato část reprezentována klíčovým slovem `WHERE`. Poslední je blok `orders`, který umožňuje seřadit informace v sestupném či vzestupném pořadí.

```
<DataAccess access="public" connection="mqlQuery" id="datum"
type="mql">
  <Cache duration="3600" enabled="true"/>
  <Parameters>
    <Parameter name="screen" type="Numeric" default="4158"/>
  </Parameters>
  <Query><![CDATA[<?xml version="1.0" encoding="UTF-8"?>
  <mql>
    <domain_id>hgis</domain_id>
    <model_id>HgIS</model_id>
    <selections>
      <selection>
        <view>BC_OBSERVATIONS</view>
        <column>BC_OBSERVATIONS_DATE_</column>
      </selection>
      <selection>
        <view>BC_OBSERVATIONS</view>
        <column>BC_OBSERVATIONS_DATE_</column>
      </selection>
    </selections>
    <constraints>
      <constraint>
        <operator>AND</operator>
        <condition>[BC_OBSERVATIONS.BC_OBSERVATIONS_SCREEN_IDSCREEN]
= [param:screen]</condition>
      </constraint>
    </constraints>
    <orders>
      <order>
        <direction>DESC</direction>
        <view_id>BC_OBSERVATIONS</view_id>
        <column_id>BC_OBSERVATIONS_DATE_</column_id>
      </order>
    </orders>
  </mql>]]>
  </Query>
</DataAccess>
```

Zdrojový kód 3: Ukázka vytvoření časové řady

8 Řešené problémy

8.1 Nedostatek dostupných zdrojů

Během tvorby práce je vhodné čerpat z informace z příručky [8], ze které se lze naučit základy tvorby dashboardu pomocí CDF, avšak tento zdroj informací není ani zdaleka dostačující pro porozumění potřebným funkcím pro vytvoření aplikace. Bohužel většina online materiálu a podpor se zabývá dashboardy v CDE, což je z většiny pro tuto práci nepodstatné. Lze zakoupit tutoriál pro tvorbu komponent, který obsahuje detailní popis možností každé komponenty, tedy seznam parametrů, jejich využití, propojení apod. Jedná se však o základní vlastnosti, které k vytvoření komplexnějších aplikací nejsou dostačující. V případě potřeby lze využít komunitního fóra, které je spravováno vývojáři PCE, kteří mohou pomoci porozumět funkčnosti platformy a jejích komponent.

8.2 Předávání parametrů

V případě obdržení dat z databáze v závislosti na více než jednom údaji, který je předem známý, vyskytuje se v CDF problém s jejich získáním. Jednotlivé komponenty umožňují předání pouze jednoho parametru pro jeden dotaz. Z tohoto důvodu, při získání hodnot větších a menších než je určitá mez, je zapotřebí tyto údaje z databáze použít všechny a teprve poté je ve skriptu filtrovat dle potřeby. Tento způsob je časově náročný a zahrnuje přesun velkého množství dat. Naštěstí komponenty CTools jsou výkonově optimalizovány, takže i velké objemy dat se z databáze přenášejí rychle, bez dlouhého čekání.

8.3 Dotazy na metadata

Při vytváření dotazů v MQL pro získání dat je v některých případech zapotřebí složitých dotazů. Tento jazyk neobsahuje také množství funkcí pro selekci dat jako například SQL. Z tohoto důvodu je dodatečná úprava získaných dat prováděna až po zobrazení v komponentách pomocí JavaScriptu.

8.4 Tlačítka pro zobrazení dat

Pokud jsou v tabulce *borings*, *stratigraphy* či *fill* data pro zvolený objekt, jsou tlačítka pro výpis těchto dat aktivní. Tato kontrola probíhá při dotazu na danou tabulku. V některých případech tato se tato kontrola provede úspěšně a při vrácení prázdného řetězce z dotazu jsou tyto tlačítka deaktivována, a někdy jsou tlačítka aktivní, i v případě, že se tato podmínka vyhodnotí jako nesplněná a data neexistují. Důvodem tohoto úkazu může být špatná návaznost získání dat přes abstraktní vrstvu na funkce prováděné po inicializaci komponent.

8.5 Využití CSS

Pro přizpůsobení vzhledu požadavkům je nutné využít CSS, avšak komponenty, které se vykreslují, jsou vytvořeny pomocí objektu, který nenabízí možnost definování identifikátoru či třídy. Tudíž u každé komponenty je zapotřebí využít funkce, která se volá po vykreslení do dashboardů a v této funkci využitím jQuery přiřadit dané komponentě třídu z CSS.

9 Shrnutí práce

Zadání bakalářské práce bylo splněno. Seznámil jsem se s platformou Pentaho a s jejími komponentami, především s CDF a PME. (kapitola druhá). Navrhnul jsem strukturu aplikace (kapitola čtvrtá). Vytvořil jsem a abstraktní vrstvu a naprogramoval aplikaci (kapitola pátá). Nakonec jsem zhodnotil vhodnost komponenty CDF a CTools pro tvorbu aplikací v prostředí Pentaho (kapitola osmá). Webová aplikace disponuje všemi funkcemi, které byly zadány. Návrh pohledového zobrazení je založen na snaze vytvořit pohodlné a intuitivní řešení pro uživatele. Databázové dotazy jsou optimalizovány pro rychlost v rovnováze s prací s daty. Data jsou z databáze vybrána v určitých případech všechna a jejich selekce je prováděna pomocí skriptu tak, aby nezabírala zbytečně mnoho času. Uživatel čeká minimální dobu a to v řádu jednotek sekund v případě filtrování výpisu seznamu pozorovaných objektů podle jejich druhu.

Po vypracování aplikace v CDF je zřejmé, že tato komponenta není vhodná pro tvorbu aplikací. Sice lze docílit požadované funkce programu, ale absence některých funkcí nedovoluje vytvoření optimálního řešení. Některé funkce, které je jednoduché vytvořit za pomoci JavaScriptu je těžké vytvořit ve frameworku s využitím CTools.

Community Tools jsou na druhou stranu velikým přínosem, jelikož umožňují jednoduše naprogramovat prvky webové stránky a nahrát do nich data z datového zdroje využitím CDA.

Výhodou využití abstraktní vrstvy je v možnosti manipulace s daty, jelikož umožňují tabulky přeorganizovat či odvozovat nové, které zjednoduší výsledný dashboard nebo data zabezpečí omezením přístupu k nim. Oproti přístupu k datům pouze využitím CDA je doba přístupu mnohem větší, jelikož skutečná data musí být nejdříve zpracována touto abstraktní vrstvou, což podstatně zpomaluje konečnou aplikaci.

PBA je vhodný prostředek pro manipulaci s velkým množstvím dat, které je třeba přehledně zobrazit například na webové stránce. Disponuje mnoha komponenty ulehčující programátorovi i uživateli práci, avšak není vhodný pro podrobnější analýzu nasbíraných dat.

Použitá literatura

- [1] *HgIS: Hydrogeologický informační systém* [online]. Liberec: Kamil Nešetřil, [cit. 2018-04-27]. Dostupné z: <https://dataearth.cz>
- [2] JODAS, Tomáš. *Webová prezentace prostorových dat*. Liberec, 2014. Bakalářská práce. Technická univerzita v Liberci. Vedoucí práce Kamil Nešetřil.
- [3] *Pentaho Community* [online], 2017 [cit. 2017-05-12]. Dostupné z: <http://community.pentaho.com/>
- [4] A Conceptual Overview of the Pentaho Metadata Editor. *Pentaho: Documentation* [online]. 2005 [cit. 2018-04-27]. Dostupné z: <https://help.pentaho.com/Documentation/7.0/ON0/110/020>
- [5] Community Dashboard Framework. *Pentaho Community* [online]. 2017 [cit. 2017-05-12]. Dostupné z: <http://community.pentaho.com/ctools/cdf/>
- [6] *OpenLayers* [online]. 2017 [cit. 2017-05-13]. Dostupné z: <http://openlayers.org/>
- [7] Community Tools. *Pentaho Community* [online]. 2017 [cit. 2017-05-12]. Dostupné z: <http://community.pentaho.com/ctools>
- [8] *Learning Pentaho CTools*. Livery Place 35 Livery Street Birmingham B3 2PB, UK.: Packt Publishing, 2016. ISBN 978-1-78528-342-0.
- [9] Souřadnicové systémy. ČÚZK. *Geoportál ČÚZK* [online]. 2010 [cit. 2017-05-18]. Dostupné z: [http://geoportal.cuzk.cz/\(S\(0fm4bdol1kbfm4n5t0jlgwm\)\)/Default.aspx?lng=CZ&mode=TextMeta&side=sit.trans&text=souradsystemy](http://geoportal.cuzk.cz/(S(0fm4bdol1kbfm4n5t0jlgwm))/Default.aspx?lng=CZ&mode=TextMeta&side=sit.trans&text=souradsystemy)
- [10] What is Data Integration? | Pentaho. *Data Integration, Business Analytics and Big Data / Pentaho* [online]. 2005 [cit. 2017-05-21]. Dostupné z: <http://www.pentaho.com/what-is-data-integration>
- [11] Raster Reprojection. OpenLayers [online]. 2017 [cit. 2017-05-22]. Dostupné z: <https://openlayers.org/en/latest/doc/tutorials/raster-reprojection.html>

Příloha na CD

- Text bakalářské práce
 - bakalarska_prace_2018_Vojtech_Hajek.pdf
 - bakalarska_prace_2018_Vojtech_hajek.docx
- Zdrojový kód programu
 - vzhled webové aplikace (v jazyce CSS)
 - dotazy na metadata (v jazyce MQL)
 - definice webové stránky (ve struktuře XML)
 - hlavní zobrazující část (v jazyce HTML)
 - doplňující skripty (v jazyce JavaScript)
 - markers (ikony bodů užitých v mapě)
 - img (doplňující obrázky)