Sem vložte zadání Vaší práce.

České vysoké učení technické v Praze
Fakulta informačních technologií
Katedra softwarového inženýrství

Bakalářská práce

Metadata pro datový sklad fakulty

Jakub Krejčí

Vedoucí práce: Ing. Stanislav Kuznetsov

5. května 2015

Poděkování

Chtěl bych poděkovat svému vedoucímu bakalářské práce Ing. Stanislavu Kuznetsovovi za odborné vedení, za pomoc a rady při zpracování této práce. Dále bych chtěl poděkovat všem blízkým a příbuzným za pochopení a podporu.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl 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 5. května 2015 .....

Odkaz na tuto práci

KREJČÍ, Jakub. *Metadata pro datový sklad fakulty.* Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2015.

Abstrakt

Tato bakalářská práce se zabývá návrhem a realizací metadatového řešení pro nový datový sklad fakulty informačních technologií ČVUT. V práci je popsaná teorie v oblasti metadat a podrobně rozebírá tři základní typy metadat (business, technická a procesní). V implementační části práce je popsán návrh komplexního metadatového řešení pro fakultní datový sklad. Je zde popsáno provedení pilotního nasazení procesních metadat pomocí nástroje Pentaho Data Integration a business metadat pomocí nástroje Pentaho Metadata Editor. Pilotní nasazení bylo úspěšně otestováno na databázi PostgreSQL a při tvorbě reportu v nástroji Pentaho Report Designer.

Klíčová slova Metadata, business metadata, technická metadata, procesní metadata, datový sklad, business model, ETL, Pentaho Metadata editor, Pentaho Data Integration, PostgreSQL, úložiště metadat.

Abstract

This bachelor‘s thesis describes design and implementation of metadata solutions for new data warehouse Faculty of Information Technology, CTU. In thesis is described theory of metadata and three basic types of metadata in data warehouse (business, technical, process execution). Implementation part

describes complex design of metadata solution for data warehouse of faculty. It is described pilot deployment of process metadata using Pentaho Data Integration and businesss metadata using Pentaho Metadata Editor. Pilot deployment was successfully tested on PostgreSQL database and on creation report in Pentaho Report Designer.

Keywords Metadata, business metadata, technical metadata, process execution metadata, data warehouse, business model, ETL, Pentaho Metadata editor, Pentaho Data Integration, PostgreSQL, metadata repository.

Obsah

Úvod **1**
Cíl práce 1

I Teoretická část **3**

1 Teoretický úvod do metadat **5**
1.1 Metadata v běžném životě 5
1.2 Co jsou metadata 6
1.3 Druhy metadat 7
1.4 Využití metadat 8

2 Metadata pro potřeby datových skladů **9**
2.1 Datové sklady dle Kimballa 9
2.2 Metadata v datových skladech 12
2.3 Back room vs. front room metadata 14
2.4 Business metadata 16
2.5 Technická metadata 19
2.6 Procesní metadata 22
2.7 Metadata v jednotlivých částech datového skladu 22
2.8 Shrnutí 25

II Implementační část **27**

3 Analýza **29**
3.1 Současný datový sklad fakulty 29
3.2 Analýza použitelných nástrojů 30
3.3 Požadavky pro metadatové řešení 31

4 Návrh metadatového řešení pro datový sklad fakulty **33**
4.1 Business metadata 33
4.2 Technická metadata 34
4.3 Procesní metadata 35
4.4 Jiná metadata 36

5 Realizace metadatového řešení **37**
5.1 Implementace business metadat 37
5.2 Implementace procesních metadat 39

6 Otestování implementovaného metadatového řešení **41**
6.1 Business metadata 41
6.2 Procesní metadata 42

Závěr **45**

Literatura **47**

A Seznam použitých zkratek **49**

B Relační model datového skladu fakulty **51**

C Úložiště procesních metadat **53**
C.1 Relační model databáze 53
C.2 SQL skript na vytvoření úložiště 55

D Výsledky provedených testů **59**
D.1 Obsah části úložiště procesních metadat 59
D.2 Report vygenerovaný pomocí PRD 61

E Obsah přiloženého CD **63**

Seznam obrázků

1.1 Katalogizační lístek 6
1.2 Výrobní štítek 7

2.1 Dimenzionální model 10
2.2 Architektura datového skladu 13
2.3 Vizualizace ETL transformace 21
2.4 Zdroje front room metadat 24

4.1 Návrh struktury procesních metadach 36

5.1 Datová kostka KOS 38
5.2 Business model KOSu 38
5.3 Nastavení business metadat v PME 39
5.4 Nastavení procesních metadat v PDI 40

6.1 Testování business metadat v PRD 41
6.2 ETL job pro testy 42
6.3 Obsah tabulky job_log po běhu testů 43

Seznam tabulek

2.1 Datový slovník 18
2.2 Logická datová mapa 19
2.3 ETL job metadata 20

4.1 Návrh logické datové mapy část I. 34
4.2 Návrh logické datové mapy část II. 34
4.3 Návrh logické datové mapy část III. 34
4.4 Hierarchie ETL procesů 35

Úvod

Tato práce je součástí projektu, který má vylepšit současný datový sklad fakulty informačních technologií vytvořený ing. Stanislavem Kuznetsovem [1] o nové prvky, které jsou v dnešní době standardem při tvorbě datových skladů.

Mezi hlavní požadavky tohoto projektu patří rozšíření datového skladu o historizaci, server realizující automaticky ETL procesy, metadatové řešení a webovou encyklopedii datového skladu. Tyto požadavky byly rozděleny na jednotlivé bakalářské práce a realizují je společně se mnou studenti Martin Čejka [2], Robert Kotlář [3] a Radim Lenger[4].

Má práce se zabývá částí, která je odpovědná za metadatové řešení. Metadatové řešení pro datový sklad pomáhá zejména koncovým uživatelům pochopit význam „surových“ dat získaných přímo z datového skladu nebo zanalyzovaných dat v rámci reportu, či dashboardu (business metadata). Také se zabývá ukládáním potřebných technických informací pro administrátory datového skladu a ETL tým (technická metadata) a informací o běhu ETL procesů (procesní metadata).

Svou bakalářskou práci jsem rozdělil do dvou hlavních částí. První část se zabývá teorií spojenou s metadaty. V těchto teoretických kapitolách analyzuji obecné definice metadat a metadata v datových skladech. Z metadat obsažených v datových skladech se zaměřuji především na business, technická a procesní metadata. Ve druhé části se zaměřuji na metadatové řešení pro fakultní datový sklad. Tato implementační část obsahuje kapitoly obsahující analýzu, návrh řešení, implementaci a otestování realizovaného řešení. Celý tento projekt je implementován pomocí technologie vytvořené společností Pentaho.

Cíl práce

Cílem této bakalářské práce je navrhnout metadatové řešení pro nový datový sklad fakulty. Součástí této práce je definovat jednotlivé typy metadat, které je důležité ukládat v rámci datového skladu fakulty. Vyzkoušet mož-

nosti současné technologie, kterou využívá fakultní datový sklad (technologie společnosti Pentaho) v oblasti metadat a provést zkušební implementaci částí metadatového řešení závislých na technologii současného datového skladu.

Část I

Teoretická část

Kapitola 1

Teoretický úvod do metadat

V této kapitole se budu věnovat vysvětlení a přínosu metadat. Pro zjednodušení pochopení tohoto pojmu, který je pro tuto práci zásadní, vynechám v této kapitole teorii metadat pro datové sklady a rozeberu ji až v kapitole následující. Tuto kapitolu jsem napsal především podle teorie popsané v knize Introducition to Metadata [5] a teorii popsané v disertační práci ing. Jakuba Novotného [6].

1.1 Metadata v běžném životě

S metadaty se setkává člověk běžně v rámci každodenních činností, pouze je bere jako samozřejmost a většinou ani neví co pojem metadata znamená a co si pod ním představit. Tento fakt demonstruji na následujících dvou příkladech.

1.1.1 Knihovna

Metadata spojená s katalogizačním lístkem budu ve své práci několikrát zmiňovat. Na Obrázku 1.1 můžeme vidět katalogizační lístek z Městské knihovny Tábor. Na lístku si můžeme všimnout dat, která nám pomáhají odlišit knihu v rámci všech knih v knihovně. Mezi tyto data patří signatura (oddělení v rámci knihovny), přírůstkové číslo knihy, počet stránek, obal knihy a status.

1.1.2 Strojírenství

Na každém výrobku můžeme nalézt výrobní štítek. Tyto štítky rozšiřují popis výrobků, zejména v oblasti výroby - výrobce, výrobní čísla, revize (o těchto informacích se koncový zákazník mnohdy ani nedozví).

Na Obrázku 1.2 vidíme výrobní štítek leteckého pístového motoru Mikron IIIB. Tento výrobní štítek je umístěn přímo na krytu motoru a je pilotovi nebo mechanikovi po otevření krytu motoru dostupný. Popisuje základní charakteristiky motoru - maximální výkon a otáčky (tyto údaje se jinak dají zjistit

Obrázek 1.1: Elektronický katalogizační lístek táborské knihovny

pouze v provozní příručce). Z výrobního štítku také můžeme vyčíst, že motor byl vyroben v roce 2001, ale byl vyroben v Československu. Z těchto dat můžeme odvodit následující informaci - jedná se o repasovaný motor po generální opravě [1].

1.2 Co jsou metadata

Metadata se velmi často popisují jako data o datech. A proto přichází otázka co si pod tímto pojmem, kterým jsou v mnoha případech definovány, vůbec představit.

„It is a construct that has been around for as long as humans have been organizing information.“ [5] Tento popis popisuje metadata velmi obecně, ale jeho myšlenka platí pro všechna metadata. Pokud se zamyslíme nad touto větou, tak z ní můžeme odvodit, že metadata jsou konstrukce, která popisuje data a vznikla primárně pro potřeby archivování, vytváření seznamů a zpřehledňování uložení informací. Tento způsob ukládání informací byl používán již před stovkami let a používá se doteď (díky informačním technologiím stále více).

[1] Skutečně se jedná původní motor Walter Mikron III, který se vyráběl v poválečném období a dnes je velmi rozšířen mezi ultralehkými replikami historických letadel. Tento motor byl modernizován při generální opravě do současné verze IIIB v roce 2001.

Obrázek 1.2: Výrobní štítek leteckého motoru Mikron

V současných informačních systémech, které jsou stále komplexnější, je velmi důležité rychle znát význam informací, které jsou v systému obsaženy, což popisují právě metadata. Protože historická definice by nebyla naprosto přesná pro dnešní metadata, kterým neustále vyrůstá využití, musela se doplnit.

1.2.1 Definice

Metadata jsou definována různými interpretacemi definice, kterou jsem nastínil v předešlé části textu. Mě nejvíce zaujala následující interpretace, kterou jsem vybral do své práce. Další definice najdete v disertační práci [6], kde se různým výkladům definic metadat autor rozsáhle věnuje.

V. Sklenák definuje ve své knize [7] metadata jako *„data sdružená s objekty, která zbavují jejich potencionálního uživatele nutnosti předběžné znalosti existence či charakteristik těchto objektů.“*

Tato definice dle mého názoru plně vystihuje využití metadat v současných informačních systémech a zároveň popisuje metadata i při jejich klasickém použití (objekt si můžeme na příklad představit jako knihu, kterou popisuje katalogizační lístek pro lepší vyhledání a zařazení v knihovně).

1.3 Druhy metadat

Metadata se dají rozdělit do těchto skupin, v této práci používám rozdělení dle knihy [5]. Jedná se o rozdělení podle kterého můžeme roztřídit libovolná metadata podle použití.

- **Administrativní** (administrative) se používají pro správu a administraci dat a zdrojů informací. Např. informace o poloze, dokumentace právních požadavků k přístupu, výběr kritérií k digitalizaci.
- **Popisné** (descriptive) se používají k identifikaci a popisu dat a souvisejících zdrojů informací. Např. záznamy seznamu, anotace tvůrce, pomoc při hledání, speciální indexy.
- **Ochranné** (preservation) se používají ke správě ochrany dat a zdrojů informací. Např. dokumentace o stavu zdrojů, dokumentace akcí k zachování fyzické a digitální verze zdrojů.
- **Technická** (technical) zobrazují, jak se systém nebo metadata chovají. Např. dokumentace hardwaru a softwaru, technické informace digitalizace, přihlašování a zabezpečení dat.
- **Pro užití dat** (use) popisují úroveň a druh použití dat a zdrojů informací. Např. oběhové záznamy, sledování používání systému a uživatelů, záznamy z vyhledávání.

1.4 Využití metadat

Metadata pomáhají k přeměně dat v informace tím, že data identifikují, definují, lokalizují, zdrojují a zpřístupňují [8].

Metadata používají uživatelé (lidi, stroje) k dalšímu vyhodnocování informací. Uživatel pomocí metadat může rychle pochopit význam bez znalosti detailního původu dat, tímto efektivně přeměňuje data v informace. Pod touto přeměnou si můžeme představit hledání knihy v knihovně dle katalogizačního lístku. Údaje v něm obsažené urychlují nalezení knihy a přidávají informace, které jsou s knihou spojeny (např. počet stran, žánr, anotace). Také si pod ní můžeme představit zpracovávání dat analytikem, kde provádí analytik analýzu na datech jejichž význam mu popisují pouze metadata (většinou nemá přístup do zdrojových databází a nezná architekturu systému).

V dnešních informačních systémech, které se většinou skládají z několika odlišných subsystémů jsou metadata nutnost. Bez metadat by nebylo možné dále pracovat s informacemi pro další analýzy, ale ani by nebylo možné systém efektivně spravovat.

Kapitola 2

Metadata pro potřeby datových skladů

V této kapitole se již věnuji metadatovým řešením pro současné datové sklady. V textu vycházím především z problematiky popsané dle jednoho z otců datových skladů Ralpha Kimballa[2]. Kimballovu architekturu datových skladů jsem zvolil především proto, že je na ni postavena současná verze fakultního datového skladu. Architekturu fakultního datového skladu popsal ing. Kuznetsov ve své diplomové práci [1].

2.1 Datové sklady dle Kimballa

V této části bych nerad rozebíral všechny pojmy používané v datových skladech. Chci zde pouze vysvětlit základní principy a pojmy používané v datových skladech. Tento základní pohled je vhodnější pro lepší pochopení metadat v datových skladech, která jsou úzce spojeny s některými pojmy definovanými v teorii datových skladů.

Datový sklad je *„the queryable source of data in the enterprise“* a technicky ho můžeme definovat jako *„the union of all the constituent data marts“* [9].

Tato Kimballova definice popisuje datový sklad jako zdroj dat pro analýzu a reporting ve společnostech a říká, že datový sklad je sjednocení všech datových tržišť (*data mart* - segment datového skladu, který obsahuje informace pro analýzu určité části společnosti např. určitý úsek výroby, zároveň musí být reprezentován pomocí dimenzionálního modelu).

Datový sklad na rozdíl od běžných relačních databází, které se používají v informačních systémech, obsahuje i neaktuální data (časová dimenze rozli-

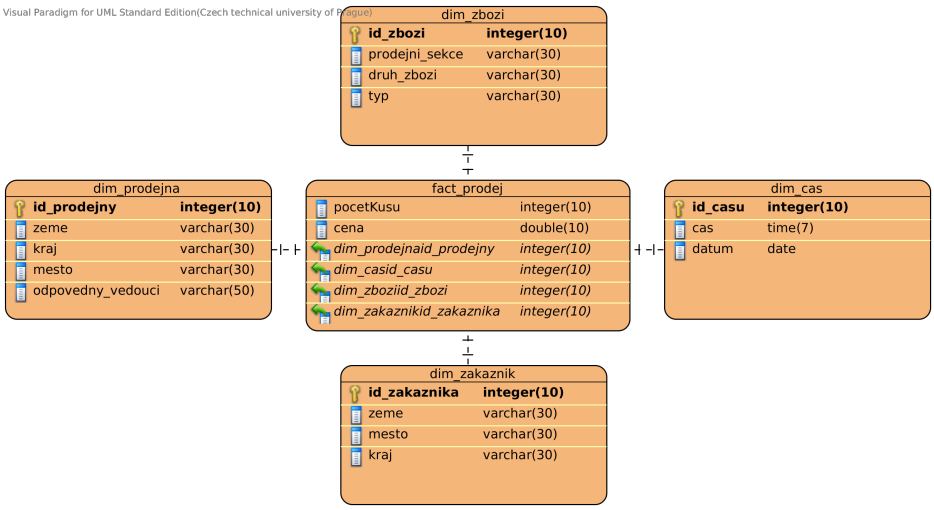
[2]Bill Inmon definoval pojem datový sklad první, ale používá odlišný pohled na vnitřní strukturu datového skladu než Kimball. Takže použití jeho poznatků z oblasti metadat by mohlo být v této práci matoucí.

šuje, kdy data vznikly). Tyto data uchovává především proto, aby bylo možné realizovat analýzy v časově širokém období.

Kimball také definuje základní pravidla, která by měl každý datový sklad splňovat [10].

- Informace jsou jednoduše dostupné a věrohodné.
- Přizpůsobivá architektura datového skladu, která je odolná na změnu zdrojových systémů.
- Datový sklad je dobře zabezpečen a je správně řízen přístup uživatelů do datového skladu.
- Musí obsahovat strategické informace, které pomáhají managementu v rozhodování.
- Uživatelé musí být správně proškoleni a využívat datový sklad jako primární zdroj informací.

Dimenzionální model je specifický způsob návrhu relační databáze, který se skládá z *faktových* a *dimenzionálních* tabulek. Tento návrh se používá pro práci s rozsáhlými soubory dat a využívá často technologii OLAP. Oproti klasickému návrhu používaném v informačních systémech - 3NF (třetí normální forma) je povolena v dimenzionálním modelu duplicita a celý model je optimalizován na minimální počet joinů (od toho se odvíjí rychlost zpracování dotazu).



Obrázek 2.1: Dimenzionální model obchodní společnosti

Faktová tabulka je tabulka, ve které jsou umístěny metriky, které se používají k analytickým výpočtů, dále se v této tabulce nachází i cizí klíče dimenzionálních tabulek. Tyto tabulky zabírají většinu paměťového prostoru používaného datovým skladem.

Dimenzionální tabulka rozšiřuje data obsažená ve faktových tabulkách o další informace. Na základě těchto informací můžeme při tvorbě analýz vybrat, které řádky faktové tabulky nás zajímají. Zjednodušeně tedy můžeme říct, že dimenzionální tabulka slouží jako filtr, který určuje, jaké řádky chceme vybrat z faktové tabulky.

Na Obrázku 2.1 je vidět, že celý dimenzionální model je tvořen především pro účely analýzy a strategického rozhodování. Informace nutné pro bezproblémový chod obchodního procesu a ani data nutná pro správný chod informačního systému nejsou v datovém skladu zahrnuta (jména zákazníků, přesné adresy, kontaktní údaje).

2.1.1 Architektura datového skladu dle Kimballa

Achitekturu datového skladu můžeme vidět na Obrázku 2.2. Tento obrázek popisuje prvky, které jsou nutné pro správný chod datového skladu. Tyto jednotlivé prvky datového skladu ještě Kimball rozdělil do dvou skupin dle jejich viditelnosti pro uživatele.

- **Back room** je skupina, ve které se nachází zdrojové systémy a ETL procesy. Je to tedy skupina, která obsahuje prvky datového skladu skryté pro jeho koncové uživatele a odehrávají se v ní vnitřní procesy (především procesy získávající data ze zdrojových systémů).

- **Front room** je skupina prvků, které jsou pro koncové uživatele viditelné. Tudíž do této vrstvy patří data presentation area a data acces tools (systémy zabývající se zpracováním a presentováním informací z datového skladu).

Operational Source Systems jsou informační systémy, které slouží pro provoz společnosti (např. ERP - Enterprise Resource Planning, CRM - řízení vztahu se zákazníky). Informace z těchto systémů slouží jako vstupní data do ETL procesů. Datové sklady většinou získávají data z více nezávislých systémů.

Extract-transformation-load (ETL) procesy probíhají v části *Data Staging Area*. ETL procesy zpracovávají data ze zdrojových systémů, které transformují do podoby použitelné pro datové sklady. V těchto procesech dochází především k čištění, standardizaci a integraci dat z nezávislých systémů. Výstupem ETL procesů musí být vždy naprosto validní data, z tohoto důvodu

bývají ETL procesy nejsložitější a nejdražší částí datového skladu. Těmito validními daty se plní jednotlivé data marty datového skladu.

Do těchto procesů patří i zpracování metadat, tomuto tématu se budu věnovat na následujících stránkách podrobněji a *historizace* - způsob který určuje, jak zacházet s již neaktuálními informacemi v dimenzích, aby mohly být používány v analytických dotazech, které se zaměřují na minulost a zároveň, aby nezpomalovaly a zbytečně nezahlcovaly datový sklad. Této problematice se věnuje Robert Kotlář ve své bakalářské práci [3].

Data Presentation Area je část datového skladu, ve které jsou fyzicky uloženy v data martech jednotlivé informace (relační databáze, kde jsou jednotlivé tabulky navrhnuty dle dimenzionálního modelu). Dle Kimballa by se tato část měla budovat na základě potřeb společnosti metodou bottom up (zdola nahorů).

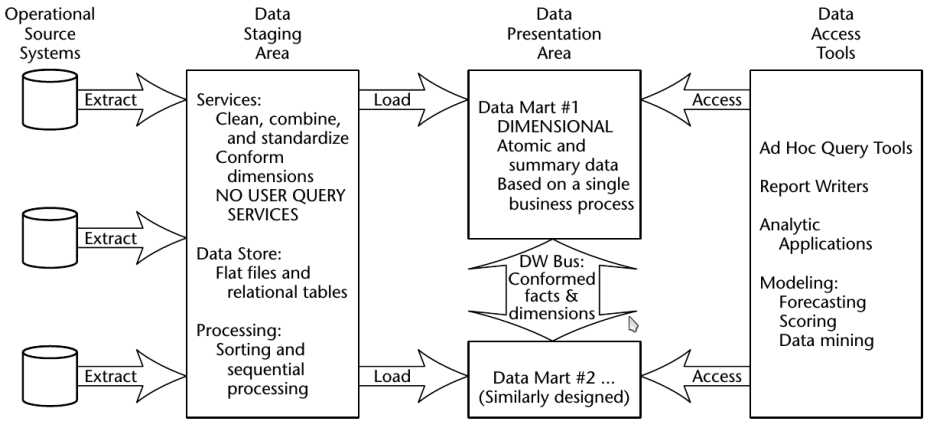
V praxi to znamená, že data marty se začnou tvořit odděleně nejprve pro analýzu jednotlivých odděleních (či jiného logického celku ve společnosti, který je potřeba odděleně analyzovat). Postupně se budou přidávat další data marty, které již nebudou popisovat jednotlivé oddělení, ale jednotlivé úseky a takto se pokračuje, dokud datový sklad neobsáhne celou organizační strukturu společnosti.

Data Acces Tools je část datového skladu, ve které se reprezentují informace obsažené v datovém skladu jeho koncovým uživatelům. Za tuto část by měl již zodpovídat analytik zodpovědný za vytváření reportů ve společnosti nebo dodavatelé BI systémů. Reprezentace informací z datového skladu může být vedena těmito způsoby.

- Business intelligence aplikace
- Předdefinovaný report, či dashboard
- Data mining

2.2 Metadata v datových skladech

Metadata jsou důležitou součástí datového skladu, protože data marty jsou tvořeny dimenzionálním modelem, který mění nejenom strukturu, ale i názvy atributů pod kterými jsou data uložena ve zdrojových systémech. Abychom správně chápali význam dat uložených v data martech je nutné tato data doplnit o data, které je popisují (použití definice metadat v praxi). Tímto doplněním se stávají z dat uložených v datovém skladu informace. V této sekci své bakalářské práce se věnuji především popisu jednotlivých typů metadat a jejich propojenost s jednotlivými prvky datového skladu.



Obrázek 2.2: Architektura datového skladu dle Kimballa [10]

Kimball obdobně jako u architektury datového skladu rozlišuje metadata do dvou skupin [11]. Opět zde používá dělení podle viditelnosti (v tomto případě spíše přístupu) pro koncové uživatele.

- **Back room metadata** jsou tvořena údaji, které potřebuje ETL tým pro správné vytváření a nastavení ETL procesů. Dále obsahují údaje nutné k ostatním vnitřním procesům datového skladu a jeho správě.

- **Front room metadata** tvoří údaje, které přibližují uživatelům celý datový sklad (nejen data obsažená v data martech). Tato část je velmi podstatná, protože na ní závisí budoucí využívání a správnost analýz informací získaných z datového skladu.

V dnešní době, zejména z důvodu rozšiřování pojmu *big data*[3], se do datových skladů a tedy i do metadat dostávají informace, které nemají předem přesně definovanou strukturu. Díky tomuto faktu se můžeme setkat i s dalším dělením metadat [12].

- **Struktorovaná metadata** - můžeme určit s vysokou pravděpodobností vnitřní strukturu těchto dat.

- **Nestruktorovaná metadata** - předem nelze odhadnout vnitřní strukturu dat a musíme je před uložením do datového skladu různými způsoby editovat.

[3]Big data je termín aplikovaný na soubory dat, jejichž velikost nám neumožňuje spravovat a zpracovávat tyto data běžnými nástroji v rozumném čase.

Uložení metadat a vizualizace Metadata pro datový sklad můžeme ukládat pomocí dvou základních způsobů. Tyto způsoby se liší především další využitelností a vizualizací uložených metadat.

Databázové úložiště se využívá pro metadata, která získáváme přímo ze zdrojových systémů nebo vznikají při ETL procesech. Tento způsob uložení má obrovskou výhodu v možnosti používání metadat v dalších systémech, které jsou napojeny na datový sklad (např. BI aplikacích). Protože ukládáme metadata do další databáze, tak se můžeme dostat ke strukturám, které budou „meta meta data data“ [11] - musíme popsat způsob uložení těchto dat, které popisují jiná data. Toto celé metadatové úložiště využíváme při vyhodnocování dat a jako popis při analýze dat v BI aplikacích (business metadata), ale i při odhalování chyb (technická, procesní metadata). Tyto metadata se vizualizují přímo v BI aplikacích nebo pomocí externího softwaru určeného pro správu metadat.

Encyklopedie datového skladu obsahuje metadata získaná jinou formou (např. dokumentace zdrojových systémů, definice reportů, manuály, procesy společnosti). Informace obsažené v této encyklopedii slouží i jako dokumentace celého datového skladu. Vizualizace tohoto způsobu uložení bývá nejčastěji internetová encyklopedie (např. Confluence[4]). Encyklopedii datového skladu řeší Martin Čejka ve své bakalářské práci [2].

Závěrem této úvodní části, než se začnu věnovat vysvětlování jednotlivých typů metadat, bych rád poznamenal, že tato metadatová dělení se používají pouze pro rozkouskování takto objemného pojmu. Jednotlivé kategorie metadat se mohou mezi sebou prolínat a mnohdy nejde přímo určit, do jaké kategorie spadají. Samozřejmě se také nedá vždy přesně určit, pro jakou skupinu uživatelů jsou určeny.

2.3 Back room vs. front room metadata

Metadata uložená v encyklopedii datového skladu můžeme do poměrně vysoké míry odlišit a tedy i rozdělit na back/front room. Tato metadata většinou obsahují položky, které jsou dobře rozdělitelné do skupin cílových uživatelů (např. modely zdrojových databází pro ETL tým, návod k použití pro uživatele).

Problém nastává s metadaty, které získáváme ze zdrojových systémů (vstupujícími s daty společně do ETL procesů datového skladu). Tato metadata již snadno rozlišit nejde, mnohdy se nedají odlišit ani nastavením přístupu uživatelů k těmto informacím. Často dochází ke změnám v řízení přístupu uživatelů k těmto informacím (např. analytik hledá chybu a žádá o přístup k technickým informacím o zdroji dat). Proto bylo zavedeno další rozdělení a to podle typu informace, kterou metadata obsahují (respektive co popisují).

[4]Internetová encyklopedie, webová stránka výrobce https://www.atlassian.com/

2.3.1 Rozdělení metadat dle typu informace

Metadata můžeme rozdělit následovně podle typu informace, kterou obsahují. [11].

- **Business metadata** - popisují význam dat a pravidla určená obchodním procesem společnosti.
- **Technická metadata** (technical) - popisují technické aspekty dat (např. délka, typ atributů) a technického řešení datového skladu.
- **Procesní metadata** (process execution) - prezentují statistiky spojené s průběhem ETL procesů.

Pro lepší rozdělení těchto skupin metadat můžeme rozdělit ještě jednotlivé skupiny na back/front room. Toto rozdělení se nejvíce vyplatí u business metadat, která jsou využívány nejširší skupinou uživatelů. Rozdělení jednotlivých skupin na back/front room nám pomůže rozlišit metadata určená pro jednotlivé druhy uživatelů, protože nemůžeme tvrdit, že business metadata jsou pouze pro koncové uživatele nebo technická a procesní metadata jsou pouze pro ETL tým.

2.3.2 Informace obsažené ve front room metadatech

Front room metadata datového skladu (encyklopedie + metadata uložená v databázi) by měly obsahovat informace zajišťující rychlé a bezproblémové seznámení koncových uživatelů s cílovým systémem [9].

Metadatové řešení datového skladu by mělo obsahovat tyto části v rámci front room metadat. Tyto části pomohou koncovým uživatelům, rychle pochopit využívání informací získaných z datového skladu. Tyto části, které by měly obsahovat front room metadata, jsou doporučeny pro veliké datové sklady.

- **Business** společnosti a všechny informace, které se týkají navázání dat z datového skladu na obchodní činnost (pochopení dat).
- **Reporting** ve společnosti, kde jsou rozebrány a definovány jednotlivé metriky obsažené v datovém skladu (pochopení faktů důležitých pro tvorbu reportů).
- **Dokumentace** všech systémů, které s datovým skladem souvisí pro koncové uživatele (zacházení s dostupnými nástroji).
- **Přístupy** k datům (práva jednotlivých uživatelů) a informace o zabezpečení datového skladu.
- **Změny** a vylepšení, které jsou nově realizovány v systémech datového skladu.

- **Kontakty** na tým, který zodpovídá za rozvoj a údržbu datového skladu. Případně na výrobce datového skladu.

2.4 Business metadata

Jsou metadata, která překládají data uložená v datovém skladu do obchodního modelu společnosti. Můžeme tedy říct, že business metadata jsou zobrazení, které jednoznačně přiřazuje k atributu obsaženému v datamartu datového skladu pojem v rámci obchodního procesu společnosti.

Tento typ metadat je v rámci datového skladu nejdůležitější, protože přibližuje informace obsažené v datovém skladu běžnému uživateli. Bez business metadat by musel pracovat s daty z datového skladu pouze člen týmu vývojářů datového skladu, který rozumí architektuře zdrojových systémů datového skladu a obchodnímu procesu společnosti (extrémní případ, protože všechny informace o datovém skladu zanikají výpovědí pracovníka). Toto je možné v rámci malých datových skladů, které spravuje člen IT týmu společnosti. Veliké projekty se takto řešit v žádném případě nedají.

Business metadata se dají rozdělit do dvou hlavních částí, které by měla každá implementace obsahovat.

- **Obchodní definice** - popis konkrétního atributu v rámci obchodní činnosti společnosti. Tato část business metadat je určena primárně pro koncové uživatele.

- **Informace o zdrojových systémech** - detailnější popis obchodní činnosti společnosti rozšířený o obchodní pravidla a větší specifikaci zdrojových systémů. Tuto část business metadat využívá primárně ETL tým.

Správci Tyto metadata by měl vytvářet a spravovat analytik zodpovědný za sběr požadavků v rámci obchodního procesu společnosti nebo designér datového modelu (dimenzionální model jednotlivých datamartů). Často se může stát, že odpovědnou osobou za část business metadat v rámci datového skladu může být i analytik zdrojových systémů (část business metadat se získává přímo ze zdrojových systémů). Není přesně určeno, kdo by měl tyto metadata vytvářet a spravovat. V praxi se určuje zodpovědná osoba za business metadata vždy v rámci konkrétního projektu.

„Do business metadat by neměl zasahovat ETL tým, který je součástí back room části datového skladu. Nicméně by tým měl rozumět významu dat se kterými pracuje - nastudovat obchodní termíny, které potřebuje.“[11] Pro členy ETL týmu by měli být business metadata tzv. *proxy metadaty* - metadata jsou získána z jednoho systému a vložena do jiného bez změn obsahu informací, které popisují.

2.4.1 Obchodní definice

Obchodní definice (business definitions) jsou typem metadat, které koncovým uživatelům překládá data obsažená v datovém skladu do informací, které jsou popsány v rámci obchodních termínů konkrétní společnosti.

Každá obchodní definice musí být v rámci společnosti zadefinována jednoznačně, jinak se mohou vyskytnout chyby v popisu jednotlivých atributů datového skladu. Protože v jednotlivých odděleních společností dochází často k odlišnému zadefinování klíčových obchodních pojmů (jinak na problém nahlíží technické oddělení a jinak finanční). Tuto možnou nesourodost v rámci definování jednotlivých obchodních pojmů musí vyřešit analytik. Je nutné se domluvit se všemi odděleními na zadefinování obchodních pojmů jednotně pro celou společnost (každý zaměstnanec si tuto definici vyloží stejně).

Metadata obsahující obchodní definice se skládají z relativně mála prvků (jsou to jedny z nejjednodušších souborů metadat) a dají se vyjádřit i běžnou tabulkou, která popisuje všechny atributy obsažené v datovém skladu. Toto řešení se příliš nevyužívá a je lepší ukládat tato metadata do databázové metadatového úložiště pro jejich lepší využití. Metadata popisující obchodní definice by měli obsahovat [11].

- **Název tabulky a sloupce** v relační databázi, ve které jsou umístěny data datového skladu (název může obsahovat různé technické prefixy a suffixy). Tyto fyzické názvy nemusí být prezentovány koncovým uživatelům.
- **Obchodní název sloupce** je název pod kterým se běžný koncový uživatel pracuje při využívání datového skladu. Pod tímto názvem je sloupec reprezentován v rámci BI aplikací. Z tohoto důvodu by měl být název jednoznačný pro danou společnost a lehce pochopitelný a nesmí obsahovat žádné technické zkratky.
- **Obchodní definice** stručně popisují význam konkrétního atributu v rámci obchodního procesu společnosti (zhruba dvě věty). Každý atribut datového skladu musí mít přesně danou obchodní definici. Pokud u atributu nelze přesnou definici určit, tak většinou tento atribut nemá žádný analytický přínos a neměl by být v datovém skladu obsažen.

Datový slovník (data dictionary) slouží k ukládání metadat obsahujících obchodní definice. Tato metadata mohou být rozšířena i o informace o zdrojových systémech, ale pro použití koncových uživatelů jsou tyto informace nepodstatné. Tato metadata, jak jsem již naznačil se mohou ukládat do databáze nebo se můžou formulovat pomocí statické tabulky. V Tabulce 2.1 můžeme vidět příklad informací obsažených v datovém slovníku.

Fyzický název	Obchodní název	Obchodní definice
k_p_username	uživatelské jméno	Unikátní uživatelské jméno v rámci celého ČVUT. Toto jméno zůstává osobě i po skončení působení na ČVUT.

Tabulka 2.1: Ukázka business metadat v datovém slovníku

2.4.2 Informace o zdrojových systémech

Informace ze zdrojových systémů (source system information) obsahují údaje nutné pro ETL tým. Tyto údaje musí poskytovat dostatečný obchodní popis zdrojových systémů a pravidel obchodního procesu společnosti, aby na základě těchto informací mohl ETL tým vytvořit kvalitní ETL procesy pro transformaci dat do datového skladu.

Kimball na základě svých zkušeností definuje následující kategorie informací nutných pro vývoj ETL procesů [11].

- **Identifikace zdroje** neboli název zdrojového systému ze kterého budeme získávat data. Může se jednat o název databáze nebo souboru (v případě exportů). Zároveň by se nemělo jednat o technický název, ale o jeho obchodní název (např. databáze tržeb).
- **Specifikace tabulky** obsahuje údaje nutné pro ETL tým, aby pochopil účel dané tabulky (velikost, primární klíč, alternativní klíč a seznam všech sloupců tabulky).
- **Pravidla pro výjimky** je část, která obsahuje informace o problémech s kvalitou dat a rady, jak tyto problémy řešit v ETL procesech.
- **Obchodní definice** jsou obdobné jako v předchozí části (Obchodní definice), důležité k rychlému pochopení smyslu dat.
- **Obchodní pravidla**, která obsahují omezení v rámci obchodního procesu společnosti. Každé pravidlo musí být testováno, zdokumentováno a vyřešeno ve zdrojovém systému nebo v ETL procesech. Pod těmito pravidly si můžeme představit vztah objednávka - zákazník, kde nastavíme pravidlo, že objednávka musí mít právě jednoho zákazníka.

2.4.3 Logical Data Maps

Informace obsažené v logických datových mapách se používají jako specifikace funkčních požadavků při vytváření konkrétních ETL jobů. Toto využití se váže k období vytváření analýzy a návrhu řešení konkrétního datového skladu. Tyto informace jsou důležité také při testování (uživatelské akceptační testy), řízení kvality a podpoře v případě nalezení chyb.

Tento dokument poskytuje zobrazení dat ze zdrojového systému do cílového uložení v datovém skladu (tzv. source-to-target mapping). Toto zobrazení je popsáno stručně a jednoduše (používají se pouze význačné informace - nejedná se o přesný a úplný popis transformací), takže tímto informacím by měli rozumět i koncoví uživatelé datového skladu.

Logické datové mapy by měli obsahovat následující tři části. Většinou tyto části bývají zobrazené v jedné tabulce.

- **Cíl** by měl obsahovat sloupce název tabulky v databázi, název sloupce v databázi, datový typ, typ tabulky (dělení pro datový sklad - dimenzionální/faktová), typ použité historizace (zkratka SCD - slowly changing dimension).
- **Zdroj** by měl obsahovat sloupce název zdrojové databáze, název tabulky, název sloupce, datový typ.
- **Transformace** tento sloupec stručně popisuje transformace, které nám zobrazují data ze zdroje do cíle. Veškeré hlavní operace s daty jsou zde popsány stručně a jednoduše v rámci několika vět.

Cíl					**Zdroj**				**Transformace**
Tabulka	Sloupec	Datový typ	Typ	SCD	Databáze	Tabulka	Sloupec	Typ	
d_student	id_student	integer	dim	0	excelovský soubor		id_student	číslo	Sloupec nastaven jako primární klíč dimenze.

Tabulka 2.2: Ukázka business metadat v logické datové mapě

v Tabulce 2.2 vidíme možnou realizaci datové mapy, která nám poskytuje základní přehled o transformaci dat. Z uvedené tabulky se můžeme dočíst, že atribut id_student je primárním klíčem dimenzionální tabulky d_student, která se historizuje dle typu 0 (atribut dimenze se nikdy nepřepisuje) a zdrojem těchto dat je excelovský soubor.

2.5 Technická metadata

Popisují celý datový sklad po technické stránce. Informace, které jsou obsažené v technických metadatech využívá především tým zodpovědný za implementaci a správu datového skladu a business aplikací s ním spojených. Technická metadata můžeme rozdělit do několika skupin na základě významu informací, které obsahují.

Soupis systému (system inventory) obsahuje informace o technických aspektech všech systémů, které jsou spojeny s datovým skladech. Do této skupiny patří i technické popisy jednotlivých částí datového skladu.

Datové modely jednotlivých tabulek, které jsou obsaženy v datovém skladu. Může se jednat o normalizované i dimenzionální schémata. Tyto schémata jsou potřebná pro rychlé zorientování ve vnitřní struktuře datového skladu (např. vztahy mezi dimenzemi a faktovými tabulkami).

Definice dat popisuje detailní technickou specifikaci jednotlivých sloupců obsažených v databázích. Tyto informace jsou nutné pro tvorbu ETL procesů, protože ETL tým na základě těchto informacích o zdrojových databázích vytváří ETL procesy. Takto popsány jsou i databáze spojené s datovým skladem, tento popis se využívá při vytváření BI řešení.

Obchodní pravidla mohou být někdy popisována i v rámci business metadat, ale musíme si uvědomit, že implementace těchto pravidel v rámci datového skladu je složitý technický proces (tato pravidla jsou implementovány v rámci ETL procesů). Proto se v rámci business metadat popisují pouze obchodní pravidla, ale v rámci technických metadat se popisuje jejich řešení v rámci datového skladu a omezení, které tyto pravidla určují.

Metadata spojené s ETL procesy se v rámci technických metadat dělí na tři hlavní skupiny. V některé literatuře jsou mezi ně ještě přidána procesní metadata, ale procesní metadata ukládají metriky pro vyhodnocení efektivnosti samotného datového skladu, proto je v rámci této práce uvádím samostatně.

2.5.1 ETL job metadata

Popisují jednotlivé ETL joby, které jsou součástí ETL procesů datového skladu. V Tabulce 2.3 vidíme nejjednodušší implementaci těchto metadat. Z této tabulky můžeme přehledně vyčíst nejdůležitější informace o konkrétním jobu.

Tyto základní údaje můžeme rozšířit o specifičtější informace o konkrétním průběhů jednotlivých jobů. ETL tým zde může dokumentovat jednotlivé fáze (extrakci, střední část, cíl). Pokud se rozhodneme popisovat metadata takto detailně, je dobré popsat i posloupnost jednotlivých transformací, které job obsahuje.

Název jobu	Zdrojová tabulka	Cílová tabulka	Účel jobu
Job_d_study_historization	d_study	d_study_after_scd	Historizace atributů v dimenzi d_study

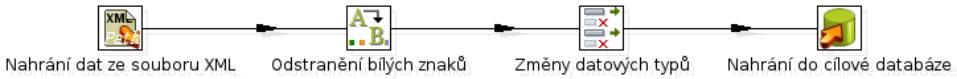
Tabulka 2.3: Nejjednoduší implementace metadat popisujících ETL joby

2.5.2 Transformační metadata

Obsahují údaje o nejdůležitějších a nejpodrobnějších prvcích ETL procesů tedy o transformacích. Měly by obsahovat konkrétní operace prováděné s daty (např. filtrování, sjednocení). Pod těmito metadaty si můžeme představit vizualizaci transformací. Nebo mohou být podrobnější a detailně popisovat každou součást transformace (toto řešení se používá výjimečně).

Současné ETL nástroje obsahují dostatečně přehlednou vizualizaci jednotlivých kroků transformace již při její tvorbě (viz. Obrázek 2.3). Pokud nevyužíváme žádný ETL nástroj a jednotlivé transformace píšeme ručně např. pomocí jazyka SQL, tak se dá použít jako transformační metadata i řádně okomentovaný zdrojový kód nebo skript.

Případně můžeme transformace popsat i detailnějšími popisy jednotlivých částí (jaké činnosti se v konkrétní části dějí). Ale tento popis není nutný, protože pokud používáme ETL nástroj, tak nám může nastavení konkrétní části transformace vysvětlit kterýkoliv člen ETL týmu (jednotné operace, které se liší jen nastavením).



Obrázek 2.3: Ukázka vizualizace transformace pomocí ETL nástroje Pentaho Data Integration

2.5.3 Metadata o jednotlivých dávkách

Neboli batch metadata obsahují informace o rozdělení jednotlivých jobů do dávek. Tato metadata by měli obsahovat i procesní metadata (přesněji časový plán spuštění konkrétní dávky). Měly by v ní být obsaženy informace o hierarchii jednotlivých jobů, přesný popis posloupnosti ve které se jednotlivé joby spouští a za jaké podmínky se spustí/nepustí.

Tato metadata můžeme vizualizovat okomentovaným obrázkem z ETL nástroje nebo vhodně formátovaným textem, který bude obsahovat všechny potřebné údaje.

Ukázku možné struktury informací o jednotlivých dávkách můžete najít v části 4.2.1 Hierarchie ETL procesů této bakalářské práce. Konkrétněji v Tabulce 4.4. V této tabulce je navržena možná struktura uložení základních informací o jednotlivých ETL dávkách pro fakultní datový sklad. Tato vizualizace pomocí tabulky je vhodná pro internetovou encyklopedii datového skladu. Proto se jedná o rozcestník, který slouží k navigaci ke konkrétní části ETL dávky.

2.6 Procesní metadata

Všechna procesní metadata jsou vygenerována při ETL procesech. Popisují totiž běh a výsledek těchto procesů. ETL procesy jsou složeny z mnoha jobů. Při každém spuštění jobu vznikají statistiky o jeho průběhu, které můžeme následně ukládat jako procesní metadata. Po sloučení těchto statistik vzniká vyhodnocení průběhu celého ETL procesu. Detailnost těchto informací závisí na konkrétní implementaci, protože při ETL procesech vzniká veliké množství procesních metadat, je nutné správně nastavit pravidla pro ukládání těchto metadat. Tyto pravidla nastavuje ETL tým, který hledá kompromisní řešení. Je nutné nalézt řešení, které ukládá dostatečně detailní informace a neukládá zbytečně moc informací, které se primárně nevyužívají při analýze běhu ETL procesů (může dojít k rychlému růstu velikosti databáze). Procesní metadata rozlišujeme na tři hlavní části a jsou hlavním zdrojem informací při měření kvality ETL procesů.

Run results obsahuje statistiky spojené s během ETL procesů a jejich výsledky. Tyto výsledky jsou velmi široký pojem a skládají se z několika částí.

- **Základní informace** obsahují základní údaje o jobu (jméno, s jakým datamartem je job spojen, zdroj a cíl).
- **Informace o zpracování** udávají kolik bylo řádek zpracováno (řádek na vstupu, zpracováno úspěšně, zpracováno chybně) a textový popis poslední chyby.
- **Čas a metriky** obsahují časové údaje o spuštění jobu (začátek, konec), rychlost čtení (metrika popisující kolik dat je job schopen načíst za sekundu) a rychlost ukládání (metrika popisující kolik dat je job schopen uložit za sekundu).

Exception handling můžeme přeložit jako řízení výjimek. V případě že v nějakém jobu nastane výjimka (většinou z důvodu špatné kvality dat) jsou vygenerovány informace o vyřešení této výjimky.

Batches schedules neboli časový plán spuštění ETL procesů. Obsahuje informace o nastavení automatického spouštění jednotlivých ETL procesů (např. čas a frekvence).

2.7 Metadata v jednotlivých částech datového skladu

Metadata se vyskytují, vytvářejí nebo získávají v následujících částech datového skladu [11]. Metadata, která zde budu uvádět nemusí být obsaženy ve

všech implementacích datového skladu. Vždy je nutné vymyslet optimální řešení na konkrétní datový sklad - u menších datových skladů není vynechání části metadat problém.

Metadata související s data access tools (práce s daty z datového skladu v externích systémech) nejsou v této kapitole řešena. Při jejich tvorbě vždy záleží na vývojářích konkrétního systému (neexistuje obecné doporučení, co by měla tato metadata obsahovat).

2.7.1 Operational Source System

Ze zdrojových systémů a jejich dokumentace můžeme získat informace, které použijeme jako prvky encyklopedie datového skladů. Tyto informace jsou z většiny užitečné pro vývojářský tým datového skladu a ke konečným uživatelům by se neměly dostat (pokud je nepotřebují k využívání).

- Relační schéma databází
- Architektura zdrojových systémů
- Změny v systémech a starý formát dat pro archivování
- Kopie manuálů

Na druhou stranu můžeme ze zdrojových systémů také získávat informace, která následně využíváme v ETL procesech a ukládáme je do databáze (toto ukládání probíhá až v ETL procesech), která obsahuje metadata datového skladu. Toto využití můžeme vidět na Obrázku 2.4, kde jsou informace z těchto zdrojových systémů zpracovávány a transformovány do podoby metadat pro datový sklad.

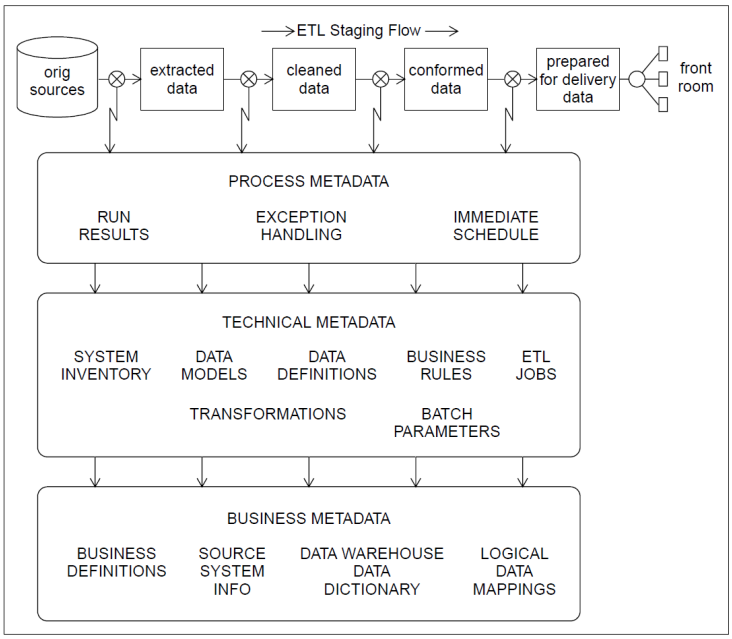
- **Deskriptivní informace** obsahují obchodní popisy dat, vlastníky zdrojového systému, právní omezení, řízení přístupu k datům apod.
- **Procesní informace** obsahují časové rozvrhy činností ve zdrojových systémech, nastavení pro automatické získávání dat apod.

2.7.2 Data Stagging Area

Metadata o ETL procesech již musí vytvářet ETL tým a designér dimenzionálního řešení datového skladu. Tato metadata slouží pro bug tracking, pro budoucí administrátory a členy ETL týmu datového skladu (technická část) a pro koncové uživatele (business část).

Do encyklopedie datového skladu by měly být přidány dokumenty, které popisují implementaci a architekturu ETL procesů a datového skladu.

- Dimenzionální model a vysvětlení jeho atributů



Obrázek 2.4: Zdroje jednotlivých typů metadat při ETL procesech [11]

- Detailní manuál k navrženému ETL procesu
- Popis použité historizace
- Informace nutné pro další využití (např. datamining)

Do metadatového úložiště bychom měli především ukládat informace, které nám přiřazují jednotlivé informace k transformacím použitým v ETL procesu (vhodné pro bug tracking) a business model (obchodní popis dat), který odpovídá atributům obsaženým v datovém skladu. Také bychom měli ukládat metriky, které nám pomůžou zefektivnit ETL procesy.

- Informace o konkrétním běhu ETL procesu (verze, datum, čas, kdo proces spustil)
- Zdrojová data použitá při ETL procesu (navázání na dimenzionální model)
- Původ atributů v dimenzionálním modelu a napojení na zdrojové systémy (využíváme deskriptivní informace obsažené ve zdrojových systémech)
- Metriky konkrétního běhu ETL procesu (doba běhu, počet úspěšných/-neúspěšných transformací)

2.7.3 Database management systems

Databázové systémy obsahují informace, které přímo souvisí s relační databází, ve které je datový sklad umístěn. V této části se setkáváme především s metadaty vhodnými pro administraci datového skladu. Informace obsažené v těchto metadatech mohou být následující.

- Využívané SQL skripty
- Nastavení relační databáze
- Nastavení zálohování
- Popis databáze

Do této části patří samozřejmě i informace vytvořené přímo konkrétním databázovým systémem. Tyto informace obsahují např. profilování, statistické informace o běhu dotazů, přístup jednotlivých uživatelů.

2.8 Shrnutí

V této kapitole jsme se setkali se všemi typy metadat, se kterými se můžeme v datových skladech setkat. Metadatové řešení by vždy mělo být postaveno na míru datového skladu. Neexistuje žádná konkrétní metodika krok za krokem pro metadata v datových skladech. Všichni autoři tuto problematiku popisují pouze jako „best practice“ z jejich profesního života.

Pokud budeme mít v metadatovém řešení obsažené všechny typy metadat, které jsem v této kapitole popisoval, tak to automaticky neznamená, že dané řešení je správné. Vždy je nutné zohlednit rozsah, účel, velikost implementačního týmu, počet a typ koncových uživatelů (např. informatik, manažer, úředník). Pokud tyto aspekty nezohledníme při návrhu může se stát, že budeme ukládat veliké množství zbytečných informací, ze kterých nebudeme schopni získávat ty užitečné.

Část II

Implementační část

Kapitola 3

Analýza

V současnosti nejsou v datovém skladu fakulty obsaženy žádné metadatové informace. Určité metadata můžeme najít pouze v dokumentaci fakultního datového skladu a jeho business intelligence aplikace.

V pilotním řešení datového skladu [1] nebylo nutné mít obsažena metadata. Protože toto řešení sloužilo především pro otestování vybrané technologie a pro demonstraci přínosu datového skladu pro fakultu informačních technologií. Potřeba metadatového řešení vznikla až po požadavku na vytvoření nového fakultního datového skladu, který již má obsahovat všechny části potřebné pro jeho zpřístupnění širokému spektru uživatelů. Současný datový sklad používá především jeho návrhář pro zkoušení analytických, databázových technologií, pro prezentaci technologie a možností využití vedení fakulty. Tento datový sklad zároveň používají studenti, kteří se snaží řešit základní analytické požadavky pomocí používané technologie v rámci předmětu Softwarový týmový projekt.

Implementaci svého metadatového řešení provedu na současném pilotním datovém skladu fakulty. Toto řešení jsem po konzultaci se svým vedoucím bakalářské práce zvolil, protože je nutné otestovat použitou technologii, aby bylo možné počítat s limity této technologie již při návrhu nového datového skladu. Proto budu v této kapitole analyzovat současný datový sklad fakulty.

3.1 Současný datový sklad fakulty

Současný datový sklad fakulty se neliší příliš od řešení popsaného v diplomové práci [1]. Toto řešení bylo následně rozšířeno o data o přihláškách studentů fakulty informačních technologií. Současné relační schéma datového skladu je přiloženo v Příloze B.

Informace ukládané do datového skladu se získávají z excelovských souborů, ve kterých jsou obsaženy exporty dat ze zdrojových systémů. Datový sklad také nemá nastaveno a implementováno žádné automatické spouštění ETL procesů nebo komplexní řešení historizace.

Fakultní datový sklad využívá databázový systém PostgreSQL. Technologie datového skladu je založena na open-source řešení společnosti Pentaho (ETL procesy jsou spravovány programem Pentaho Data Integration a analýzy jsou prováděny na BI aplikaci Pentaho BI Server).

3.1.1 Dostupná metadata pro řešení

V současné verzi datového skladu nejsou žádné metadata implementována nebo ukládána. Jelikož i data se zpracovávají na základě exportů pomocí excelovských tabulek, tak také nemůžeme využít ani metadata ze zdrojových systémů. V současnosti poskytují informace o přiblížení datového skladu následující informační zdroje.

- **Diplomová práce [1]**, kde autor popisuje celou architekturu fakultního datového skladu a detailně vysvětluje jednotlivá použitá řešení (např. popisuje činnost jednotlivých transformací). Diplomová práce obsahuje také stručný přehled business metadat (popis základních atributů obsažených v datovém skladu) a návod pro používání ETL nástroje společnosti Pentaho. Tato práce je v současnosti jediný komplexní zdroj technických metadat pro fakultní datový sklad.
- **Dokumentace projektů spojených s DW** na fakultním portálu Confluence. Hlavní část dokumentace obsažená na tomto portálu byla vytvořena týmem, kterého jsem byl součástí, v rámci týmových softwarových projektů spojených se studiem softwarového inženýrství na fakultě. V rámci těchto projektů jsme řešili především analýzy a způsoby vytváření reportů pro vedení fakulty. Takže tato dokumentace obsahuje potřebné změny ve struktuře datového skladu, schéma použitých datových kostek, návody na vytváření analýz a dashboardů.

Pro budoucí datový sklad fakulty bychom pravděpodobně mohli počítat se získáváním dat přímo ze zdrojových databází. Toto nám umožní lépe nakládat s metadaty (získávání především business metadat přímo ze zdrojových databází). Bohužel v současnosti toto není možné a přicházíme kvůli tomu o spoustu metadat (např. specifikace vstupních atributů, obchodní popisy).

3.2 Analýza použitelných nástrojů

Protože metadata jsou již velmi pokročilá funkce, tak v datových skladech nebývají v rámci open-source řešení volně přístupná (většinou se jedná o placené pluginy). Ale většinu metadat můžeme implementovat ručně bez pomoci metadatových nástrojů. K těmto implementacím můžeme využívat ETL nástroje, které nám budou ukládat námi vytvářená metadata do databázového úložiště (analýze ETL nástrojů se věnuje Radim Lenger v bakalářské práci [4]).

3.2.1 Technologie společnosti Pentaho

Celý sklad je postaven na open-source technologii společnosti Pentaho[5]. Open-source technologie Pentaho poskytuje velmi kvalitní základ pro tvorbu datových skladů i bez placených modulů. Placené nástroje jsou pro datové sklady velmi drahé (řádově tisíce dolarů) a pro fakultní nasazení nejsou na ně finance.

Data integration slouží jako nástroj pro tvorbu ETL procesů. Tento nástroj podporuje i kompletní správu a nastavení harmonogramu spouštění jednotlivých ETL dávek. Nástroj je volně dostupný z webových stránek společnosti.

Metadata editor vytváří business model na informacích obsažených v datovém skladu. Tento model je užitečný pro zobrazení informací analytikům pod jejich obchodními jmény. Bohužel tato technologie funguje pouze v externím vytváření reportů nebo při tvorbě CDE dashboardů[6]. Nástroj je opět volně přístupný z webových stránek společnosti.

3.3 Požadavky pro metadatové řešení

Požadavky specifikující metadatové řešení můžeme rozdělit na funkční a nefunkční. Jedná se pouze o základní požadavky, které vymezují podobu řešení v rámci teorie popsané v teoretické části této bakalářské práce.

3.3.1 Funkční požadavky

- **Seznámení koncových uživatelů** datového skladu. Metadata musí obsahovat všechny důležité informace nutné pro rychlé seznámení a využívání datového skladu uživateli.
- **Specifikace atributů** obsažených v datovém skladu. Všechny atributy musí být popsány v rámci obchodních názvů společnosti. A musí být přesně definován jejich význam.
- **Technický popis** jednotlivých části datového skladu včetně ETL procesů. Metadatové řešení musí obsahovat všechny důležité technické aspekty datového skladu.
- **Získávání metrik efektivnosti** ETL procesů spouštěných v rámci datového skladu fakulty. Tedy součástí metadatového řešení musí být i implementace procesních metadat.

[5] Společnost Pentaho poskytuje i enterprise edici produktů spojených s datovými sklady a BI, které jsou placené.

[6] Vytváření analýz přímo na BI serveru, kde budou informace popsány pomocí metadat, je možné pouze při placené verzi BI serveru (enterprise edice).

3.3.2 Nefunkční požadavky

- **Způsob uložení** metadat na základě jejich dalšího využití. Správně rozdělit, co ukládat pouze do internetové encyklopedie a co do databáze.
- **Řízení přístupu** jednotlivých skupin uživatelů k informacím. Je nutné správně rozlišit v metadatovém řešení, jaká metadata jsou komu určena.

Kapitola 4

Návrh metadatového řešení pro datový sklad fakulty

Návrh metadatového řešení, které budu v této kapitole popisovat je určen pro novou verzi datového skladu fakulty. V této kapitole definuji strukturu a typy informací, které by metadatové řešení nové verze datového skladu již mělo obsahovat, aby splnilo základní požadavky, které jsou na toto řešení kladeny.

Proto v této kapitole definuji návrh metadatového řešení podrobně s popisem jednotlivých kategorií, protože na základě této šablony budeme implementovat metadatové řešení pro budoucí verzi fakultního datového skladu.

4.1 Business metadata

- **Specifikace zdrojových dat** a vysvětlení jednotlivých atributů v rámci obchodního procesu fakulty (např. definice slov předmět, student).
- **Specifikace obchodních pravidel**, které se vyskytují v obchodním modelu fakulty (např. student může mít zapsaný předmět nejvýše 2).
- **Problémy s kvalitou dat**, se kterými se na fakultě můžeme setkat. Jedná se především o definici možných nevyplněných polí (např. z historických důvodů daný atribut nebyl používán).
- **Řízení přístupu** k datům obsažených v datovým skladu na základě organizační struktury fakulty.

4.1.1 Návrh logické datové mapy

Navrhuji, aby v našem případě byla logická datová mapa tvořena ze tří částí. Z těchto tří částí bude část zdrojový (Tabulka 4.1) a cílový systém (Tabulka 4.2) povinná a část business popis (Tabulka 4.3) bude volitelná. Jedná se tedy o rozšířenější verzi logické mapy než popisuje Kimball ve své knize [11].

Zdrojový systém			
Databáze	Tabulka	Sloupec	Datový typ

Tabulka 4.1: Návrh části zdrojový systém v logické mapě

Cílový systém					**Transformace**	
Tabulka	Sloupec	Datový typ	Typ tabulky	Typ SCD	Název	Popis

Tabulka 4.2: Návrh části cílový systém v logické mapě

Business popis	
Obchodní název	Obchodní definice

Tabulka 4.3: Návrh části business popis v logické mapě

4.2 Technická metadata

Nový datový sklad by měl obsahovat dostatečně podrobné technické informace, aby bylo možné datový sklad vyvíjet bez ohledu na změny ve vývojářském týmu. S těmito změnami musíme počítat, protože sklad bude vyvíjen učiteli a studenty (ukončení pracovního poměru, studia, zahraniční stáže apod.).

Z tohoto důvodu doporučuji, aby technické metadata byly uloženy v encyklopedii datového skladu a obsahovaly následující informace, které pomohou novým členům vývojářského týmu s pochopením způsobu řešení jednotlivých části datového skladu.

Celý návrh řešení je tvořen dle teoretické části této bakalářské práce. Řešení je upraveno do podoby vhodné pro fakultní nasazení datového skladu.

- **Dokumentace systémů** spojených s datovým skladem. Definice zdrojových systémů a způsobu získávání dat. Informace o architektuře cílového systému (datového skladu).
- **Řešení obchodních pravidel** v rámci datového skladu. Neboli jaké obchodní pravidla jsou v datovém skladu implementována a kontrolována.
- **Řešení problémů s kvalitou dat** získaných ze zdrojových systémů, tedy popis chování systémů na známé problémy s kvalitou dat.
- **Datové modely** datového skladu a zdrojových systémů.
- **Dokumentace ETL řešení** musí obsahovat popis technologie, popis použité historizace, informace o nastavení spouštění jednotlivých dávek.

4.2.1 Hierarchie ETL procesů

V Tabulce 4.4 je můj návrh hierarchie do které by se měly ukládat informace o jednotlivých prvcích ETL procesů. Jedná se o přehledné zobrazení, které poskytuje základní informace a zároveň funguje jako rozcestník pro získání detailních informací o konkrétní položce tohoto seznamu.

Detailní informace o konkrétním prvku ETL procesu by měli obsahovat jeho vizualizaci pomocí použitého ETL nástroje, detailnější popis (co přesně tento prvek dělá). Přesně definovaný zdroj odkud získáváme v rámci tohoto prvku informace a přesně definovaný cílový systém (většinou konkrétní tabulku, která je součástí datového skladu). Dále by v těchto informacích měli být obsaženy prvky, které jsou s tímto prvkem spojeny (rodič, potomci).

Název dávky	**Hierarchie jobů**	**Transformace**	**Popis & účel**	**Odkaz**
Název dávky			Info o dávce	link
	Název jobu		Info o jobu	link
		Název transformace	Info o tran.	link
	Název jobu		Info o jobu	link
		Název transformace	Info o tran.	link

Tabulka 4.4: Návrh struktury hierarchie ETL procesů

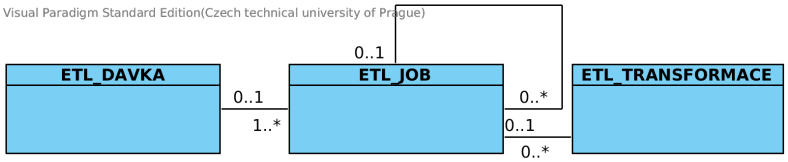
4.3 Procesní metadata

V rámci procesních metadat je nutné především vyřešit ukládaní statistik z běhu ETL procesů. Procesní informace, které by měli být v metadatovém řešení obsaženy můžeme rozdělit do dvou skupin.

- **Harmonogram** (scheduling) jednotlivých dávek. Musí obsahovat informace, které nám určují kdy a za jakých podmínek se konkrétní dávka spouští.

- **Výsledky běhu dávek** navrhuji ukládat do struktury zobrazené na Obrázku 4.1. V obrázku nejsou zahrnuty metriky a atributy, protože zde již bude záležet na konkrétní implementaci a nástroji ze kterýho spouštíme ETL procesy. Důležité je ukládat údaje o běhu dávek do hierarchie ve které lze zjistit jaký prvek co spustil a s jakým výsledkem.

  Tyto výsledky běhů jednotlivých dávek mohou být velmi paměťové náročné na ukládání (podle detailnosti informací, které ukládáme). Proto navrhuji implementovat i skript, který nám bude postupně čistit databázi od starších údajů (uchovávat detailní informace se vyplatí zhruba měsíc, kdy řeší ETL tým případné problémy).



Obrázek 4.1: Návrh struktury obsahující výsledky běhů dávek, která by měla být obsažena v procesních metadatech

4.4 Jiná metadata

Z metadat, která nepatří do žádné z již rozebíraných kategorií, navrhuji ukládat především údaje, které zlepší informovanost uživatelů a vývojářů datového skladu.

- **Kontaktní údaje** na členy vývojářského týmu datového skladu.
- **Návody** na používání systémů spojených přímo s datovým skladem. Vhodným příkladem tohoto systému je např. API pro datový sklad.

Kapitola 5

Realizace metadatového řešení

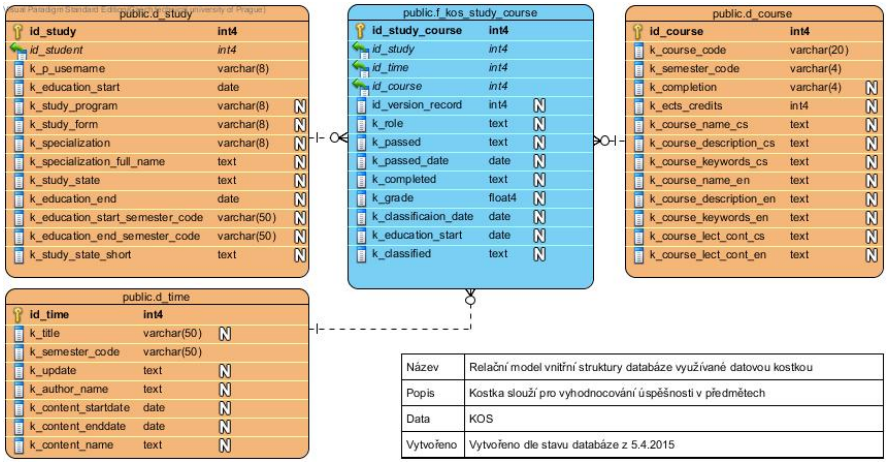
V této kapitole provedu zkušební realizaci procesních a business metadat zaměřenou na otestování technologie (u technických metadat není nutné technologii zkoušet, všechny informace se ukládají na webovou encyklopedii). Protože by bylo nutné implementovat metadatové řešení pro současný fakultní datový sklad zpětně, rozhodli jsme se s mým vedoucím bakalářské práce, že na současném datovém skladu fakulty otestujeme technologii nutnou pro tvorbu metadatového řešení a vyzkoušíme možnosti, které nám současná technologie dává.

Návrh, popsaný v předchozí kapitole a implementace s technologií kterou v této kapitole provedu, budou využity v novém datovém skladu fakulty, který se začne v létě vyvíjet. Tato zkušební implementace možných problémových částí (zatím nevyzkoušená technologie) nám umožní zareagovat na možné problémy s předstihem - tedy již při návrhu architektury.

5.1 Implementace business metadat

Při implementaci business metadat v rámci fakultního datového skladu využívám nástroj Pentaho Metadata Editor. Tento nástroj umožňuje vytvořit zobrazení, které překládá fyzické názvy sloupců a tabulek obsažených v datovém skladu do jejich obchodního významu. Soubor vytvořený v PME (formát typu XMI) nahrazuje standardní OLAP datovou kostku (formát XML). Oproti OLAP datové kostce nám soubory typu XMI umožňují nastavit přístupová práva jednotlivým skupinám uživatelů, přejmenovat atributy do obchodních názvů a přidávat k nim jejich obchodní popis.

Pro zkušební implementaci business metadat pomocí nástroji PME používám část datového skladu, na kterém je postavena datová kostka zkoumající hodnocení studentů v jednotlivých předmětech. Relační schéma této části datového skladu popisuje Obrázek 5.1.

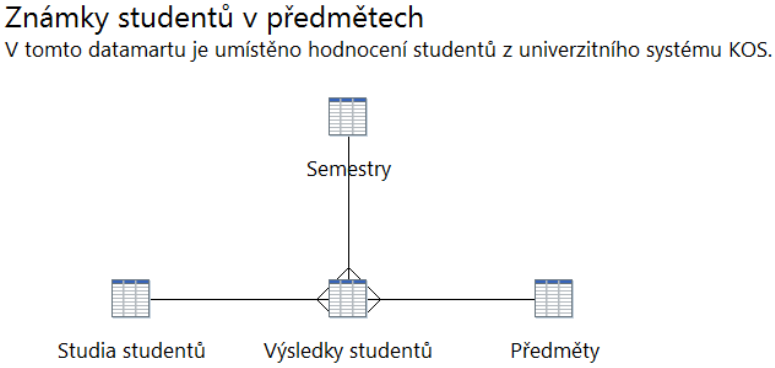


Obrázek 5.1: Fyzická podoba struktury databázových tabulek používaných datovou kostkou, která popisuje hodnocení studentů v systému KOS

5.1.1 Pentaho Metadata Editor

V PME musíme nejprve nastavit připojení databáze (v našem případě datového skladu). Po nastavení tohoto připojení proběhne načtení všech tabulek do nástroje. Tyto tabulky slouží jako zdroj pro tvorbu následující části - business modelu.

Business model zobrazuje data fyzicky uložená v datovém skladu do podoby informací z obchodního procesu společnosti. Tuto vlastnost můžeme vidět na Obrázku 5.2, kde je vytvořen business model datové kostky popisující systém KOS (Obrázek 5.1).

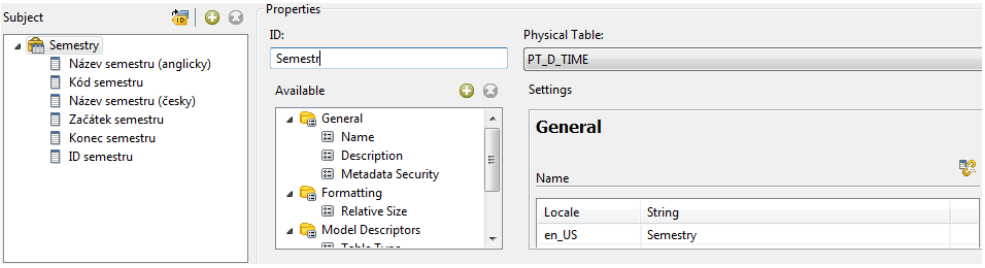


Obrázek 5.2: Podoba datové kostky z Obrázku 5.1 po aplikaci business metadat - vzniká business model (vizualizace pomocí nástroje PME)

Tvorba business modelu začíná výběrem tabulek z datového skladu, se

kterými chceme pracovat. Ve druhé fázi provedeme nastavení vztahu mezi tabulkami (vytváříme většinou strukturu dimenzionálního modelu, protože toto metadatové řešení nahrazuje OLAP datovou kostku). Po vytvoření tohoto základu začínáme vytvářet obchodní názvy a popisy atributů a tabulek. Toto nastavování obchodních popisů můžeme vidět na obrázku 5.3. Po dokončení business modelu provedeme export ve formátu XMI nebo nahrajeme business model přímo v aplikaci na BI server.

Veškeré podrobné nastavení a informace spojené s PME jsou uvedeny v internetové encyklopedii společnosti Pentaho [13].



Obrázek 5.3: Nastavení obchodních názvů a popisů v Pentaho Metadata Editoru

5.2 Implementace procesních metadat

Při získávání procesních metadat z používaného ETL nástroje - Pentaho Data Integration (dále jen PDI) vznikly poměrně veliké problémy. Procesní metadata jsou implementována pouze v enterprise edici ETL nástroje, ale my používáme jeho open-source derivát, který nemá oficiálně tyto informace ukládat do úložiště.

Pentaho dodává k enterprise edici přímo úložiště pro procesní metadata. Toto úložiště funguje, jako malý datový sklad a informace jsou připraveny k OLAP analýzám (dimenzionální model). Bohužel toto řešení se v open-source verzi nevyskytuje, ale PDI obsahuje možnost nastavit vlastní databázi, kam se budou procesní metadata ukládat. Bohužel nás limituje strukturou databáze, která není popsaná v dokumentaci (společnost počítá, že tuto možnost budou využívat jen majitelé enterprise verze).

5.2.1 Návrh úložistě procesních metadat

Relační model úložiště procesních metadat a SQL skript pro vytvoření databáze najdete v Příloze C. Při tvorbě tohoto relačního modelu jsem byl limitován strukturou, se kterou je schopný nástroj PDI komunikovat. Postupně jsem se dopracoval databázi, kterou tvoří tento relační model plně kompatibilní s PDI.

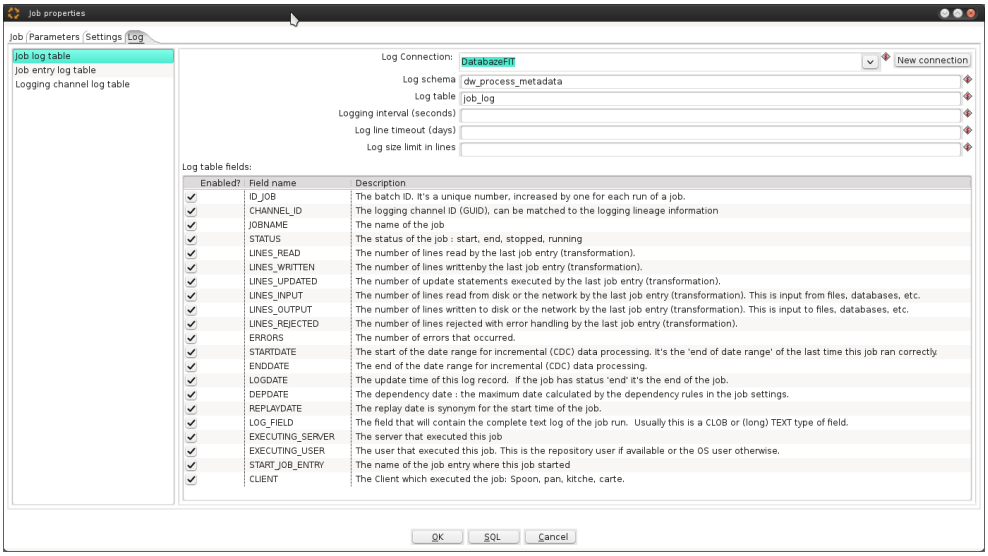
Databázový systém jsem zvolil PostgreSQL, protože i samotný fakultní datový sklad pracuje na tomto systému. V databázi, ve které jsou uloženy data datového skladu, je vytvořeno nové schéma dw_process_metadata. Tabulky obsažené v tomto schématu jsou připraveny ke komunikaci s PDI.

Pro vyhodnocování informací obsažených v úĺožišti doporučuji navrhnout malý datový sklad (návrh dle Kimballa je pro toto využití naprosto dostačující), protože úložiště je primárně modelováno pro podporu komunikace s PDI a není příliš optimalizované pro tvorbu analytických dotazů.

5.2.2 Nastavení nástroje Pentaho Data Integration

Ukládání procesních metadat je nutné nastavit přímo v ETL nástroji PDI. Nastavení ukládání procesních metadat nalezneme ve volbě job nebo transformation (podle toho s čím pracujeme) settings a po zobrazení formuláře vybereme položku Logging (tento formulář je dostupný i při stisknutí klavesové zkratky CTRL + J v případě jobu a CTRL + T v případě transformace).

V menu, které se nám zobrazí můžeme vybrat ukládání libovolných typů procesních metadat. Samozřejmě nejdříve musíme vyplnit, kam máme data ukládat (cílová databáze). Veškeré informace o možnostech procesních metadat v nástroji Pentaho Data Integration lze nalézt na internetové encyklopedii společnosti Pentaho [14].



Obrázek 5.4: Nastavení procesních metadat v PDI u konkrétního jobu

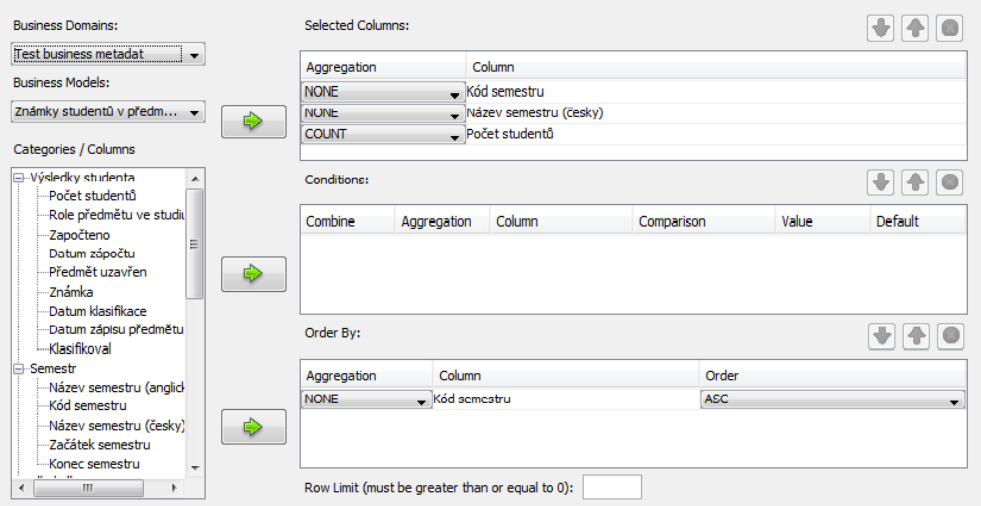
Kapitola 6

Otestování implementovaného metadatového řešení

Rozhodl jsem se otestovat implementované metadatové řešení na každodenních činnostech spojených s datovým skladem. Business metadata budou otestovány na základě tvorby zkušebního reportu a procesní metadata budou testována na běhu několika ETL jobů.

6.1 Business metadata

Business metadata testuji v nástroji Pentaho Report Designer, který umí využívat business metadata uložená v souborech ve formátu XMI.



Obrázek 6.1: Nastavení používaných atributů v PRD - lze vidět, že atributy jsou zde reprezentovány obchodními názvy

Nejprve je nutné nastavit zdroj dat jako metadata a použít vytvořený XMI soubor z Pentaho Metadata Editoru. Po tomto nastavení se již dostáváme do formuláře, který je vidět na Obrázku 6.1. Z tohoto formuláře je zřejmé, že nepracujeme s fyzickými názvy atributů a tabulek z datového skladu, ale s jejich obchodními ekvivalenty popsanými v obchodním procesu společnosti. Zkušební report vytvořený nástrojem PRD naleznete v Příloze D.2.

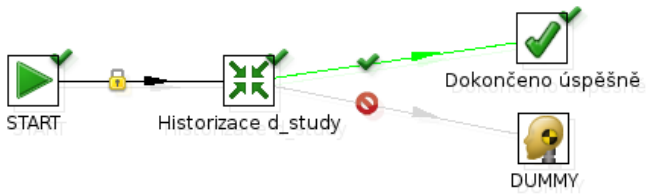
Po této implementaci business metadat nemusí znát analytik, který vyhodnocuje informace uložené v datovém skladu, vnitřní architekturu datového skladu a jednotlivé názvy atributů. Analytikovi při použití tohoto řešení stačí pouze znalost obchodního procesu společnosti - tímto je splněn princip business metadat v datových skladech.

Protože se podařilo nahradit běžnou OLAP datovou kostku souborem obsahujícím kromě vztahů mezi tabulkami i obchodní popis atributů a toto nahrazení bylo úspěšně otestováno v nástroji Pentaho Report Designer, tak považuji implementaci business metadat za úspěšnou.

6.2 Procesní metadata

Pro otestování úložiště procesních metadat jsem vytvořil v ETL nástroji PDI zkušební ETL job, jeho vizualizaci můžeme vidět na Obrázku 6.2. Základem tohoto jobu je transformace Historizace d_study vytvořená v rámci bakalářské práce Roberta Kotláře [3].

V nastavení nástroje PDI jsem zvolil ukládání všech typů metadat a jejich atributů v rámci jobu i v rámci transformací. Toto nastavení nám nejlépe otestuje komunikaci s úložištěm procesních metadat.



Obrázek 6.2: ETL job vytvořený v PDI pro otestování úložiště procesních metadat

Po spuštění několika běhů tohoto ETL jobu jsem zkontroloval log v PDI, který nehlásil žádnou chybu a provedl jsem analýzu dat uložených v úložišti procesních metadat pomocí nástroje pgAdmin3.

Na Obrázku 6.3 můžeme vidět výsledek běhu pěti ETL jobů a jejich základní procesní metriky. V této tabulce (job_log) se ukládají pouze metriky hlavního jobu. Na obrázku vidíme jen nejpodstatnější sloupce, protože tabulka má příliš sloupců a není možné je v rámci této kapitoly vizualizovat. Další ob-

sah databáze obsahující úložiště procesních metadat po běhu testů najdete v Příloze D.1.

jobname character varying(255)	status charac	lines_read bigint	lines_writt bigint	lines_upda bigint	lines_input bigint	lines_outp bigint	lines_rejec bigint	errors smallint	startdate timestamp without time zo
TestovaciHistorizace_	end	6647	6647	0	6647	0	0	0	1900-01-01 00:00:00
TestovaciHistorizace_	end	6647	6647	0	6647	0	0	0	2015-04-21 21:27:40.164
TestovaciHistorizace_	end	6647	6647	0	6647	0	0	0	2015-04-22 09:58:42.575
TestovaciHistorizace_	end	6647	6647	0	6647	0	0	0	2015-04-22 10:02:16.13
TestovaciHistorizace_	end	6647	6647	0	6647	0	0	0	2015-04-22 10:02:24.955
TestovaciHistorizace_	end	6647	6647	0	6647	0	0	0	2015-04-22 10:08:15.833

Obrázek 6.3: Vizualizace části dat tabulky job_log pomocí pgAdmin3 po běhu testů

Po zkontrolování obsahu všech tabulek ve schématu dw_process_metadata jsem zjistil, že se všechny procesní metadata ukládají bezproblémově a implementované řešení je tedy použitelné.

Závěr

V úvodu této bakalářské práce jsem definoval hlavní cíl - vytvoření komplexního metadatového řešení pro datový sklad fakulty. Tento cíl se mi povedl realizovat a navrhl jsem strukturu a pravidla pro tvorbu metadatového řešení použitelného v novém datovém skladu fakulty. Toto řešení by mělo pomoci při rozšiřování povědomí a využitelnosti datového skladu po celé fakultě.

V případě implementace business metadat, kterou jsem použil v této bakalářské práci, je nutné dořešit kompatibilitu XMI souborů a open-source verze Pentaho Business Intelligence Serveru, aby bylo možné vytvářet analýzy již přímo na BI serveru pomocí analytických nástrojů typu Saiko pod obchodními názvy atributů. Vývojáři CDE dashboardů deklarují, že jejich software by měl být s datovými zdroji typu XMI kompatibilní.

Při implementaci procesních metadat pomocí nástroje Pentaho Data Integration a databáze PostgreSQL jsem objevil zajímavé možnosti v rámci této open-source technologie, kterou používáme ve fakultním datovém skladu. Pokud vytvoříme i malý datový sklad pro ukládání nejdůležitějších informací získaných z procesních metadat, tak dostaneme řešení, které je srovnatelné s placenými nástroji (získáme kompletní přehled o běhu ETL procesů a můžeme i rychle vyhodnocovat metriky v rámci BI aplikací).

Doporučuji se držet tohoto metadatového řešení při vývoji nového fakultního datového skladu. Řešení, které jsem v této bakalářské práci navrhl, by mělo obsahovat všechny podstatné informace, které by měli být v rámci metadat popisujících datové sklady uloženy. Při tvorbě této bakalářské práce byly taky nalezeny zajímavé open-source nástroje pro tvorbu metadat v datových skladech, které doporučuji použít i v novém datovém skladu fakulty.

Výsledkem této bakalářské práce je použitelné nasazení procesních a business metadat, komplexní návrh metadatového řešení pro fakultní prostředí, kde jsou přesně definovány jednotlivé části, i ty které nejsou v této práci implementovány (metadata ukládané na internetovou encyklopedii). Zadání této bakalářské práce považuji tedy za splněné.

Literatura

[1] KUZNETSOV, S.: *Datový sklad fakulty.* Diplomová práce, České vysoké učení technické v Praze, Fakulta informačních technologií, Praha, 2013.

[2] ČEJKA, M.: *Encyklopedie datového skladu.* Bakalářská práce, České vysoké učení technické v Praze, Fakulta informačních technologií, Praha, 2015.

[3] KOTLÁŘ, R.: *Historizace dat pro potřeby datového skladu fakulty.* Bakalářská práce, České vysoké učení technické v Praze, Fakulta informačních technologií, Praha, 2015.

[4] LENGER, R.: *ETL server pro potřeby datového skladu fakulty.* Bakalářská práce, České vysoké učení technické v Praze, Fakulta informačních technologií, Praha, 2015.

[5] BACA, M.: *Introduction to metadata.* Los Angeles, CA: Getty Research Institute, 2008, ISBN 978-089236-896-9.

[6] NOVOTNÝ, J.: *Podniková metadata, jejich aplikační možnosti a využití při návrhu datového skladu.* Dizertační práce, VŠE-FIS, Praha, 2007.

[7] SKLENÁK, V.: *Data, informace, znalosti a Internet.* C.H. Beck pro praxi, C.H. Beck, 2001, ISBN 9788071794097.

[8] TANNENBAUM, A.: *Metadata Solutions: Using Metamodels, Repositories, Xml, and Enterprise Portals to Generate Information on Demand.* Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2001, ISBN 0201719762.

[9] KIMBALL, R.; REEVES, L.; THORNTHWAITE, W.; aj.: *The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing and Deploying Data Warehouses with CD Rom.* New York, NY, USA: John Wiley & Sons, Inc., první vydání, 1998, ISBN 0471255475.

[10] KIMBALL, R.; ROSS, M.: *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. New York, NY, USA: Wiley, 2002, ISBN 0471200247.

[11] KIMBALL, R.: *The data warehouse ETL toolkit practical techniques for extracting, cleaning, conforming, and delivering data*. Indianapolis, IN: Wiley, 2004, ISBN 0-764-57923-1.

[12] KRYSZCZUK, D.: *Návrh prezentace metadat reportů z IBM Cognos BI*. Diplomová práce, VŠE-FIS, Praha, 2014.

[13] PENTAHO CORPORATION: Pentaho Metadata Editor [online]. 2008, [cit. 2015-04-27]. Dostupné z: `http://wiki.pentaho.com/display/ServerDoc1x/Pentaho+Metadata+Editor`

[14] PENTAHO CORPORATION: Performance Monitoring and Logging [online]. 2013, [cit. 2015-04-27]. Dostupné z: `http://infocenter.pentaho.com/help/index.jsp?topic=%2Fpdi_user_guide%2Fconcept_pdi_usr_logging_transformations.html`

Příloha **A**

Seznam použitých zkratek

3NF Třetí normální forma

API Application Programming Interface

BI Business Intelligence

CDE Community Dashboard Editor

CRM Customer relationship management

DW Data warehouse

ERP Enterprise Resource Planning

ETL Extract, transform, load

OLAP Online Analytical Processing

PDI Pentaho Data Integration

PME Pentaho Metadata Editor

PRD Pentaho Report Designer

SCD Slowly changing dimension

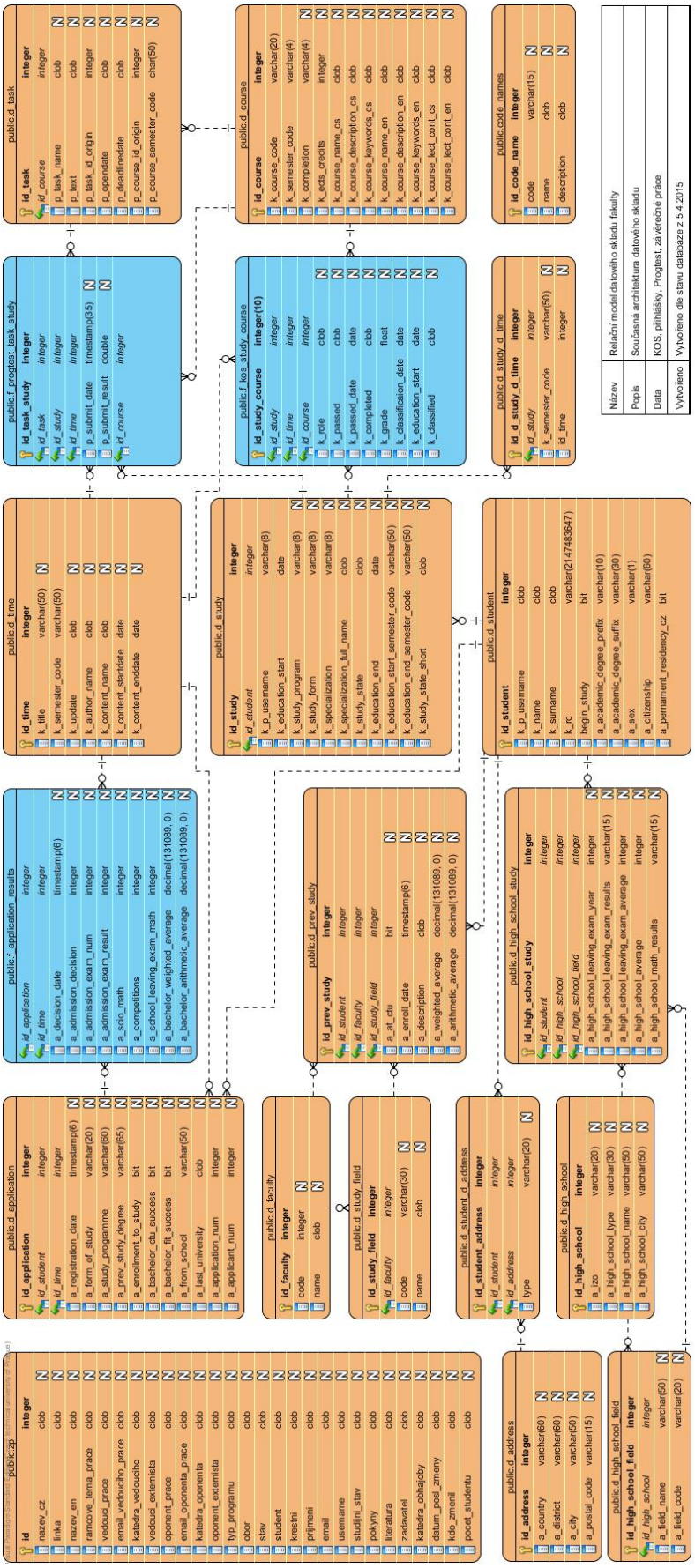
SQL Structured Query Language

XMI XML Metadata Interchange

XML Extensible Markup Language

Příloha B

Relační model datového skladu fakulty


Název
Relační model datového skladu fakulty
Popis
Současná architektura datového skladu
Data
KOS, přihlášky, Progtest, závěrečné práce
Vytvořeno
Vytvořeno dle stavu databáze z 5.4.2015

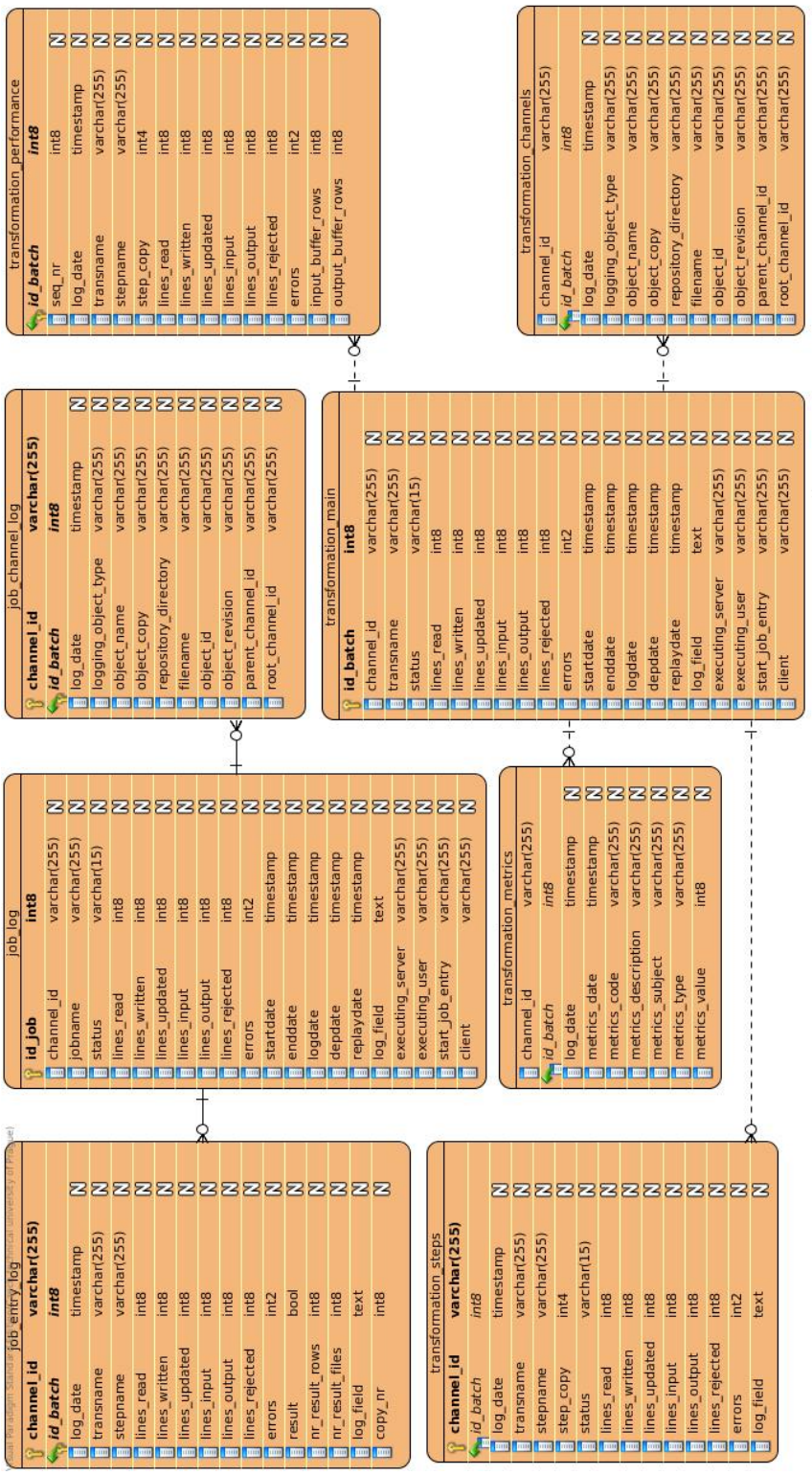
Příloha C

Úložiště procesních metadat

C.1 Relační model databáze

Jedině databáze s tímto relačním návrhem je schopna ukládat jakákoliv procesní metadata vytvořená v nástroji Pentaho Data Integration. Pokud bychom použili jiný návrh databáze, nemuselo by být možné ukládat vygenerovaná metadata. Tyto problémy jsou z důvodu, že se jedná se o open-source verzi nástroje, který společnost Pentaho poskytuje i v placené verzi - tato placená verze má procesní metadata lépe vyřešena.

Názvy jednotlivých tabulek odpovídají názvům jednotlivých typů procesních metadat, které se v rámci PDI dají ukládat.



Název	Úložiště procesních metadat
Popis	Pro ETL procesy vytvořené v Pentaho Data Integration
Technologie	PostgreSQL
Verze	1.1

C.2 SQL skript na vytvoření úložiště

```
CREATE SCHEMA dw_process_metadata;

CREATE TABLE dw_process_metadata.transformation_performance (
  id_batch              int8,
  seq_nr                int8,
  logdate               timestamp,
  transname             varchar(255),
  stepname              varchar(255),
  step_copy             int4,
  lines_read            int8,
  lines_written         int8,
  lines_updated         int8,
  lines_input           int8,
  lines_output          int8,
  lines_rejected        int8,
  errors                int2,
  input_buffer_rows     int8,
  output_buffer_rows    int8);

CREATE TABLE dw_process_metadata.transformation_steps (
  channel_id        varchar(255) NOT NULL,
  id_batch          int8 NOT NULL,
  log_date          timestamp,
  transname         varchar(255),
  stepname          varchar(255),
  step_copy         int4,
  status            varchar(15),
  lines_read        int8,
  lines_written     int8,
  lines_updated     int8,
  lines_input       int8,
  lines_output      int8,
  lines_rejected    int8,
  errors            int2,
  log_field         text,
  PRIMARY KEY (channel_id));

CREATE TABLE dw_process_metadata.transformation_main (
  id_batch          BIGSERIAL NOT NULL,
  channel_id        varchar(255),
  transname         varchar(255),
  status            varchar(15),
  lines_read        int8,
  lines_written     int8,
  lines_updated     int8,
  lines_input       int8,
  lines_output      int8,
```

```
  lines_rejected    int8,
  errors            int2,
  startdate         timestamp,
  enddate           timestamp,
  logdate           timestamp,
  depdate           timestamp,
  replaydate        timestamp,
  log_field         text,
  executing_server  varchar(255),
  executing_user    varchar(255),
  start_job_entry   varchar(255),
  client            varchar(255),
  PRIMARY KEY (id_batch));

CREATE TABLE dw_process_metadata.transformation_channels (
  channel_id              varchar(255) NOT NULL,
  id_batch                int8 NOT NULL,
  log_date                timestamp,
  logging_object_type     varchar(255),
  object_name             varchar(255),
  object_copy             varchar(255),
  repository_directory    varchar(255),
  filename                varchar(255),
  object_id               varchar(255),
  object_revision         varchar(255),
  parent_channel_id       varchar(255),
  root_channel_id         varchar(255),
  PRIMARY KEY (channel_id));

CREATE TABLE dw_process_metadata.job_entry_log (
  channel_id        varchar(255) NOT NULL,
  id_batch          int8 NOT NULL,
  log_date          timestamp,
  transname         varchar(255),
  stepname          varchar(255),
  lines_read        int8,
  lines_written     int8,
  lines_updated     int8,
  lines_input       int8,
  lines_output      int8,
  lines_rejected    int8,
  errors            int2,
  result            bool,
  nr_result_rows    int8,
  nr_result_files   int8,
  log_field         text,
  copy_nr           int8,
  PRIMARY KEY (channel_id,
  id_batch));
```

```
CREATE TABLE dw_process_metadata.transformation_metrics (
  channel_id          varchar(255),
  id_batch            int8,
  log_date            timestamp,
  metrics_date        timestamp,
  metrics_code        varchar(255),
  metrics_description varchar(255),
  metrics_subject     varchar(255),
  metrics_type        varchar(255),
  metrics_value       int8);


CREATE TABLE dw_process_metadata.job_log (
  id_job           BIGSERIAL NOT NULL,
  channel_id       varchar(255),
  jobname          varchar(255),
  status           varchar(15),
  lines_read       int8,
  lines_written    int8,
  lines_updated    int8,
  lines_input      int8,
  lines_output     int8,
  lines_rejected   int8,
  errors           int2,
  startdate        timestamp,
  enddate          timestamp,
  logdate          timestamp,
  depdate          timestamp,
  replaydate       timestamp,
  log_field        text,
  executing_server varchar(255),
  executing_user   varchar(255),
  start_job_entry  varchar(255),
  client           varchar(255),
  PRIMARY KEY (id_job));

CREATE TABLE dw_process_metadata.job_channel_log (
  channel_id          varchar(255) NOT NULL,
  id_batch            int8 NOT NULL,
  log_date            timestamp,
  logging_object_type varchar(255),
  object_name         varchar(255),
  object_copy         varchar(255),
  repository_directory varchar(255),
  filename            varchar(255),
  object_id           varchar(255),
  object_revision     varchar(255),
  parent_channel_id   varchar(255),
```

```
  root_channel_id         varchar(255),
  PRIMARY KEY (channel_id,
  id_batch));

ALTER TABLE dw_process_metadata.job_entry_log ADD CONSTRAINT
    FKjob_entry_608697 FOREIGN KEY (id_batch) REFERENCES
    dw_process_metadata.job_log (id_job);

ALTER TABLE dw_process_metadata.job_channel_log ADD
    CONSTRAINT FKjob_channe401903 FOREIGN KEY (id_batch)
    REFERENCES dw_process_metadata.job_log (id_job);

ALTER TABLE dw_process_metadata.transformation_performance
    ADD CONSTRAINT FKtransforma107322 FOREIGN KEY (id_batch)
    REFERENCES dw_process_metadata.transformation_main (
    id_batch);

ALTER TABLE dw_process_metadata.transformation_metrics ADD
    CONSTRAINT FKtransforma947932 FOREIGN KEY (id_batch)
    REFERENCES dw_process_metadata.transformation_main (
    id_batch);

ALTER TABLE dw_process_metadata.transformation_steps ADD
    CONSTRAINT FKtransforma472221 FOREIGN KEY (id_batch)
    REFERENCES dw_process_metadata.transformation_main (
    id_batch);

ALTER TABLE dw_process_metadata.transformation_channels ADD
    CONSTRAINT FKtransforma326193 FOREIGN KEY (id_batch)
    REFERENCES dw_process_metadata.transformation_main (
    id_batch);
```

Příloha D

Výsledky provedených testů

D.1 Obsah části úložiště procesních metadat

Části dat z tabulky job_log

id_job	channel_id	jobname	status	lines_read	lines_writte	lines_upda	lines_input	lines_outp	lines_rejec	errors	startdate	enddate	logdate	depdate	replaydate
0	97c289e2-8646-	TestovaciHistorizace_procesniMetadata	end	6647	6647	0	6647	0	0	0	1.1.1900 0:00	21.4.2015 21:27	21.4.2015 21:27	21.4.2015 21:27	21.4.2015 21:27
1	099d2c48-ef9d-4	TestovaciHistorizace_procesniMetadata	end	6647	6647	0	6647	0	0	0	21.4.2015 21:27	22.4.2015 9:58	22.4.2015 9:58	22.4.2015 9:58	22.4.2015 9:58
2	8843549d-d393-	TestovaciHistorizace_procesniMetadata	end	6647	6647	0	6647	0	0	0	22.4.2015 9:58	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02
3	e45b962e-8dc1-	TestovaciHistorizace_procesniMetadata	end	6647	6647	0	6647	0	0	0	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02
4	5d32c976-19fd-4	TestovaciHistorizace_procesniMetadata	end	6647	6647	0	6647	0	0	0	22.4.2015 10:02	22.4.2015 10:08	22.4.2015 10:08	22.4.2015 10:08	22.4.2015 10:08
5	fba2604d-0aab-4	TestovaciHistorizace_procesniMetadata	end	6647	6647	0	6647	0	0	0	22.4.2015 10:08	22.4.2015 10:09	22.4.2015 10:09	22.4.2015 10:09	22.4.2015 10:09
6	adca375d-1496-	TestovaciHistorizace_procesniMetadata	end	6647	6647	0	6647	0	0	0	22.4.2015 10:09	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26
7	aa3cc8b4-d374-4	TestovaciHistorizace_procesniMetadata	end	6647	6647	0	6647	0	0	0	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26
8	61798b5f-95d4-4	TestovaciHistorizace_procesniMetadata	end	6647	6647	0	6647	0	0	0	22.4.2015 14:26	22.4.2015 14:35	2015-04-22 14:3!	22.4.2015 14:35	22.4.2015 14:35

Části dat z tabulky transformation_main

id_bat	channel_id	transname	status	lines_read	lines_writte	lines_upda	lines_input	lines_outp	lines_rejec	errors	startdate	enddate	logdate	depdate	replaydate
0	98b8badd-5d05-	d_study_historization__11	end	6647	6647	0	6647	0	0	0	1.1.1900 0:00	21.4.2015 21:27	2015-04-21 21:2	21.4.2015 21:27	21.4.2015 21:27
1	ab597d09-43a7-	d_study_historization__11	end	6647	6647	0	6647	0	0	0	21.4.2015 21:27	21.4.2015 21:27	21.4.2015 21:27	21.4.2015 21:27	21.4.2015 21:27
2	cd2c1421-61c3-4	d_study_historization__11	end	6647	6647	0	6647	0	0	0	21.4.2015 21:27	22.4.2015 9:58	22.4.2015 9:58	22.4.2015 9:58	22.4.2015 9:58
3	e301008c-d3d6-	d_study_historization__11	end	6647	6647	0	6647	0	0	0	22.4.2015 9:58	22.4.2015 9:59	22.4.2015 9:59	22.4.2015 9:59	22.4.2015 9:59
4	84fbf88d-98c3-4	d_study_historization__11	end	6647	6647	0	6647	0	0	0	22.4.2015 9:59	22.4.2015 10:01	22.4.2015 10:01	22.4.2015 10:01	22.4.2015 10:01
5	30cd4e0a-9794-	d_study_historization__11	end	6647	6647	0	6647	0	0	0	22.4.2015 10:01	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02
6	bb8ec0d1-1468-	d_study_historization__11	end	6647	6647	0	6647	0	0	0	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02	22.4.2015 10:02
7	4c256070-5a78-	d_study_historization__11	end	6647	6647	0	6647	0	0	0	22.4.2015 10:02	2015-04-22 10:08	22.4.2015 10:08	2015-04-22 10:0	22.4.2015 10:08
8	d98549e4-80f0-4	d_study_historization__11	end	6647	6647	0	6647	0	0	0	2015-04-22 10:08:1	22.4.2015 10:09	2015-04-22 10:0	22.4.2015 10:09	22.4.2015 10:09
9	5c6d97d5-9400-	d_study_historization__11	end	6647	6647	0	6647	0	0	0	22.4.2015 10:09	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26
10	0bc79374-c4d0-	d_study_historization__11	end	6647	6647	0	6647	0	0	0	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26	22.4.2015 14:26
11	85843e97-b3cd-	d_study_historization__11	end	6647	6647	0	6647	0	0	0	22.4.2015 14:26	22.4.2015 14:35	22.4.2015 14:35	22.4.2015 14:35	22.4.2015 14:35

Obsah části dat z tabulky job_entry_log

channel_id	id_batch	log_date	transname	stepname	lines_read	lines_writt	lines_upda	lines_input	lines_outpu	lines_rejec	errors	resul
a11f752e-0ed9-4	0	21.4.2015 21:27	TestovaciHistorizace_p	START	0	0	0	0	0	0	0	t
6dd7f5a2-103c-4	0	21.4.2015 21:27	TestovaciHistorizace_p	Dokončeno úspěšně	6647	6647	0	6647	0	0	0	t
706ff54e-d337-4	0	21.4.2015 21:27	TestovaciHistorizace_p	Historizace d_study	6647	6647	0	6647	0	0	0	t
8370da24-9a5e-4	0	21.4.2015 21:27	TestovaciHistorizace_p	DUMMY	0	0	0	0	0	0	0	f
40a27244-979f-4	1	2015-04-22 09:58:45.039	TestovaciHistorizace_p	START	0	0	0	0	0	0	0	t
9eef95db-6cc1-4	1	2015-04-22 09:58:45.042	TestovaciHistorizace_p	Dokončeno úspěšně	6647	6647	0	6647	0	0	0	t
fb118d5e-5bb6-4	1	2015-04-22 09:58:45.043	TestovaciHistorizace_p	Historizace d_study	6647	6647	0	6647	0	0	0	t
fa1e195a-357b-4	1	2015-04-22 09:58:45.044	TestovaciHistorizace_p	DUMMY	0	0	0	0	0	0	0	f
cbba1c15-7748-4	2	22.4.2015 10:02	TestovaciHistorizace_p	START	0	0	0	0	0	0	0	t
5b3e148b-02c9-4	2	22.4.2015 10:02	TestovaciHistorizace_p	Dokončeno úspěšně	6647	6647	0	6647	0	0	0	t
72f5576d-64cb-4	2	22.4.2015 10:02	TestovaciHistorizace_p	Historizace d_study	6647	6647	0	6647	0	0	0	t
d5a3ed4d-387e-4	2	22.4.2015 10:02	TestovaciHistorizace_p	DUMMY	0	0	0	0	0	0	0	f
4b294fff-9102-42	3	22.4.2015 10:02	TestovaciHistorizace_p	START	0	0	0	0	0	0	0	t
5d021ef7-e1b4-4	3	22.4.2015 10:02	TestovaciHistorizace_p	Dokončeno úspěšně	6647	6647	0	6647	0	0	0	t
bf3cccb2-4626-41	3	22.4.2015 10:02	TestovaciHistorizace_p	Historizace d_study	6647	6647	0	6647	0	0	0	t
fa8e4808-896b-4	3	22.4.2015 10:02	TestovaciHistorizace_p	DUMMY	0	0	0	0	0	0	0	f
c12d01e2-ade8-4	4	22.4.2015 10:08	TestovaciHistorizace_p	START	0	0	0	0	0	0	0	t
92ade535-4795-4	4	22.4.2015 10:08	TestovaciHistorizace_p	Dokončeno úspěšně	6647	6647	0	6647	0	0	0	t
8beb0241-a7b0-4	4	22.4.2015 10:08	TestovaciHistorizace_p	Historizace d_study	6647	6647	0	6647	0	0	0	t
5f541064-72b6-4	4	22.4.2015 10:08	TestovaciHistorizace_p	DUMMY	0	0	0	0	0	0	0	f
fcc93aa0-9efd-47	5	22.4.2015 10:09	TestovaciHistorizace_p	START	0	0	0	0	0	0	0	t
b530f52e-1313-4	5	22.4.2015 10:09	TestovaciHistorizace_p	Dokončeno úspěšně	6647	6647	0	6647	0	0	0	t
3afb83d3-3560-4	5	22.4.2015 10:09	TestovaciHistorizace_p	Historizace d_study	6647	6647	0	6647	0	0	0	t
1f6a7ebc-419e-4	5	22.4.2015 10:09	TestovaciHistorizace_p	DUMMY	0	0	0	0	0	0	0	f
ed4a2773-93a5-4	6	22.4.2015 14:26	TestovaciHistorizace_p	START	0	0	0	0	0	0	0	t
ab6b5248-a27f-4	6	22.4.2015 14:26	TestovaciHistorizace_p	Dokončeno úspěšně	6647	6647	0	6647	0	0	0	t
0b8db8d0-e7f9-4	6	22.4.2015 14:26	TestovaciHistorizace_p	Historizace d_study	6647	6647	0	6647	0	0	0	t
69f93b71-577b-4	6	22.4.2015 14:26	TestovaciHistorizace_p	DUMMY	0	0	0	0	0	0	0	f
de3cc77b-f850-4	7	22.4.2015 14:26	TestovaciHistorizace_p	START	0	0	0	0	0	0	0	t
e2527df9-0904-4	7	22.4.2015 14:26	TestovaciHistorizace_p	Dokončeno úspěšně	6647	6647	0	6647	0	0	0	t
9ed48746-c0e1-4	7	22.4.2015 14:26	TestovaciHistorizace_p	Historizace d_study	6647	6647	0	6647	0	0	0	t
340426e9-7322-4	7	22.4.2015 14:26	TestovaciHistorizace_p	DUMMY	0	0	0	0	0	0	0	f
056efdc1-0875-4	8	2015-04-22 14:35:47.054	TestovaciHistorizace_p	START	0	0	0	0	0	0	0	t
17f7acd9-1987-4	8	2015-04-22 14:35:47.056	TestovaciHistorizace_p	Dokončeno úspěšně	6647	6647	0	6647	0	0	0	t
af8ab389-aec5-4	8	2015-04-22 14:35:47.057	TestovaciHistorizace_p	Historizace d_study	6647	6647	0	6647	0	0	0	t
dfead4fb-9838-4	8	2015-04-22 14:35:47.058	TestovaciHistorizace_p	DUMMY	0	0	0	0	0	0	0	f

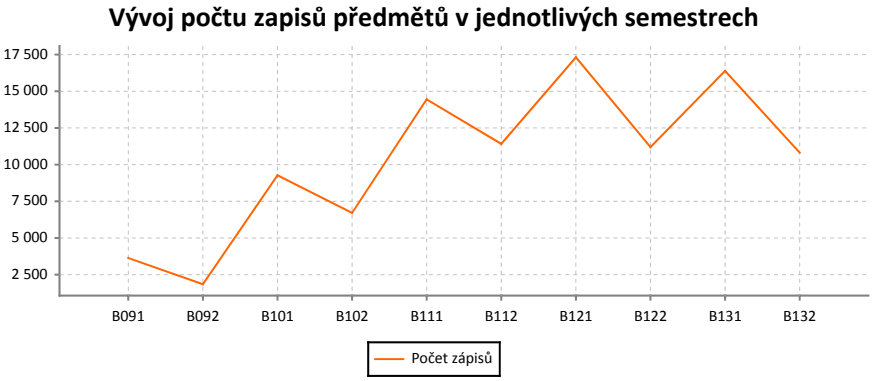
D.2 Report vygenerovaný pomocí PRD

České vysoké učení technické v Praze
Fakulta informačních technologií

Report

Vygenerováno: Tue Apr 28 15:24:45 CEST 2015

Vývoj zápisů do předmětů v rámci semestrů



Detailní informace

Kód semestru	Název semestru	Počet zápisů
B091	Zimní 2009/2010	3 636
B092	Letní 2009/2010	1 848
B101	Zimní 2010/2011	9 274
B102	Letní 2010/2011	6 714
B111	Zimní 2011/2012	14 450
B112	Letní 2011/2012	11 415
B121	Zimní 2012/2013	17 314
B122	Letní 2012/2013	11 197
B131	Zimní 2013/2014	16 389
B132	Letní 2013/2014	10 813

Report vytvořen za účelem otestování implementace business metadat v bakalářské práci zaměřené na metadatové řešení pro fakultní datový sklad

Příloha **E**

Obsah přiloženého CD

- `readme.txt` ..... stručný popis obsahu CD
- `src`
  - `impl` ..... zdrojové kódy implementace
    - `testy`
      - `testovacireport.pdf` ..... testovací report
      - `testovacireport.prpt` .. zdrojový soubor testovacího reportu
    - `businessMetadata.xmi` ..... soubor XMI s business metadaty
    - `ulozisteProcMeta.txt` .. create skript úložiště procesních metadat
  - `thesis` ..... zdrojová forma práce ve formátu LaTeX
- `text` ..... text práce
  - `thesis.pdf` ..... text práce ve formátu PDF