

**VYSOKÁ ŠKOLA EKONOMIE A MANAGEMENTU**

**BAKALÁŘSKÁ PRÁCE**

**2011**

**TOMÁŠ SVORA**

**VYSOKÁ ŠKOLA EKONOMIE A MANAGEMENTU**

**Nárožní 2600/9a, 158 00 Praha 5**

# **BAKALÁŘSKÁ PRÁCE**

**PODNIKOVÁ EKONOMIKA**

**Vysoká škola ekonomie a managementu**

+420 841 133 166 / [info@vsem.cz](mailto:info@vsem.cz) / [www.vsem.cz](http://www.vsem.cz)

# VYSOKÁ ŠKOLA EKONOMIE A MANAGEMENTU

Nárožní 2600/9a, 158 00 Praha 5

## NÁZEV BAKALÁŘSKÉ PRÁCE

Implementace Data Warehouse

## TERMÍN UKONČENÍ STUDIA A OBHAJOBA (MĚSÍC/ROK)

Říjen 2011

## JMÉNO A PŘÍJMENÍ / STUDIJNÍ SKUPINA

Tomáš Svora / PE 21

## JMÉNO VEDOUcíHO BAKALÁŘSKÉ PRÁCE

Doc. Ing. Jan Žižka, CSc.

## PROHLÁŠENÍ STUDENTA

Prohlašuji tímto, že jsem zadanou bakalářskou práci na uvedené téma vypracoval samostatně a že jsem ke zpracování této bakalářské práce použil pouze literární prameny v práci uvedené.

Datum a místo: V Praze dne 28.8.2011

\_\_\_\_\_  
podpis studenta

## PODĚKOVÁNÍ

Rád bych tímto poděkoval vedoucímu bakalářské práce za metodické vedení a odborné konzultace, které mi poskytl při zpracování mé bakalářské práce.

Vysoká škola ekonomie a managementu

+420 841 133 166 / info@vsem.cz / www.vsem.cz

**VYSOKÁ ŠKOLA EKONOMIE A MANAGEMENTU**

# **Implementace Data Warehouse**

DataWarehouse Implementation

Autor: Tomáš Svora

## **Souhrn**

V současné době, kdy je neustálý tlak na rychlost a přesnost rozhodování, je velmi důležité mít správná data k dispozici rychle a bez velkých nákladů na jejich zpracování. Tomuto problému plně vyhovuje implementace celopodnikového jednotného zdroje dat, která budou předpřipravena a uzpůsobena pro analytickou činnost. Všem výše uvedeným kritériím vyhovuje správně postavený datový sklad.

V rámci této práce jsou popsány základní principy, kterými by se měl projekt implementace datového skladu řídit. Jednotlivé fáze na sebe logicky navazují tak, jak se vyvíjí životní cyklus aplikace a u většiny kapitol je uvedeno doporučení z mé dosavadní praxe, na které oblasti se zaměřit, případně které způsobovaly při implementaci datového skladu nad účetním systémem Oracle Financial ve společnosti GE Money Bank, a.s. problémy.

Tato práce neobsahuje přímé technické detaily jednotlivých fází implementace, ale je spíše projektovým manuálem, jak k jednotlivým fázím přistupovat.

## **Summary**

Nowadays it is crucial for a business success to make perfect and exact decisions. For these reasons it is a must to have correct data as fast and inexpensive as possible. Good Data Warehousing is the way how it's possible to manage such requirements.

In this thesis there are described basic principles which should be considered by any Data Warehouse implementation. Single phases are logically arranged as they follow each other in their life cycle and most of them contain my best practice mainly in case of issues in Data Warehouse implementation over accounting system Oracle Financial at GE Money Bank, a.s.

This thesis does not contain direct technical details for any individual implementation phase but it can be used as a project manual how to manage them.

**Klíčová slova:**

Datový sklad, Datové tržiště, Implementace, Životní cyklus aplikace, Analýza, Oracle

**Keywords:**

Data Warehouse, Data Mart, Implementation, Application life cycle, Analysis, Oracle

**JEL Classification:**

M15 - IT Management

L86 - Information and Internet Services; Computer Software

C88 - Other Computer Software

M40 – General

# Obsah

|   |    |
|---|----|
| 1 Úvod .....  | 1  |
| 1.1 Cíle práce .....  | 2  |
| 1.2 Důvod pro výběr tématu .....  | 3  |
| 2 Teoreticko-metodologická část .....   | 4  |
| 2.1 Vymezení základních pojmů .....   | 4  |
| 2.1.1 Databáze .....  | 4  |
| 2.1.1.1 Tabulka v databázi .....  | 4  |
| 2.1.1.2 Views .....   | 5  |
| 2.1.2 Jazyk SQL .....   | 6  |
| 2.1.3 Metadata .....  | 6  |
| 2.1.4 Logický a fyzický datový model .....                                    | 7  |
| 2.1.5 Data Warehouse .....  | 9  |
| 2.1.5.1 Definice Data Warehouse dle W. Inmona .....                           | 9  |
| 2.1.5.1 Definice Data Warehouse dle R. Kimballa .....                         | 11 |
| 2.1.5.2 Rozdíly ve tvorbě a implementaci DW dle W. Inmona a R. Kimballa ..... | 12 |
| 2.1.6 OnLine Analytical Processing .....                                      | 14 |
| 2.1.7 Oracle Financials .....   | 14 |
| 2.1.8 GE Money Bank, a.s. ....  | 15 |
| 2.1.8.1 Divize Financí .....  | 16 |
| 2.2 Vývoj DW dle životního cyklu aplikace .....                               | 17 |
| 2.2.1 Plánování a příprava DW .....   | 17 |
| 2.2.1.1 Vstupní analýza .....   | 18 |
| 2.2.1.2 Plánování projektu DW .....   | 19 |
| 2.2.1.3 Výběr dodavatele DW .....   | 20 |
| 2.2.1.4 Úvodní studie .....   | 22 |
| 2.2.2 Analýza a návrh DW .....  | 22 |
| 2.2.2.1 Analýza podnikových procesů .....                                     | 22 |
| 2.2.2.2 Analýza vstupních datových zdrojů pro DW .....                        | 23 |
| 2.2.2.3 Analýza stávajících DM/DW a BI aplikací .....                         | 23 |

|         |  |    |
|---------|--|----|
| 2.2.2.4 | Návrh DW .....   | 24 |
| 2.2.3   | Implementace DW .....                                      | 26 |
| 2.2.3.1 | Detailní specifikace DW .....                              | 26 |
| 2.2.3.2 | Vývoj struktur DW .....                                    | 26 |
| 2.2.3.3 | Prototypy – zkušební ETL proces vstupních dat pro DW ..... | 26 |
| 2.2.3.4 | Dovývoj a customizace .....                                | 27 |
| 2.2.3.5 | Akceptační řízení .....                                    | 27 |
| 3       | Praktická část práce .....                                 | 28 |
| 3.1     | Vstupní analýza pro DW .....                               | 30 |
| 3.2     | Plánování projektu DW .....                                | 32 |
| 3.3     | Výběr dodavatele DW .....                                  | 32 |
| 3.4     | Studie proveditelnosti .....                               | 36 |
| 3.5     | Analýza vstupních datových zdrojů .....                    | 37 |
| 3.6     | Návrh DW .....   | 38 |
| 3.6.1   | Seznam primárních systémů .....                            | 38 |
| 3.6.2   | Logický model .....  | 39 |
| 3.6.3   | Fyzický datový model .....                                 | 40 |
| 3.6.4   | Tvorba metadat .....                                       | 41 |
| 3.6.5   | Způsob přenosů a definice vstupních dat - Extract .....    | 41 |
| 3.6.6   | Úpravy nad nahrávanými daty – Transform .....              | 42 |
| 3.6.7   | Nahrání dat – LOAD .....                                   | 43 |
| 3.6.8   | Odpovědnosti za jednotlivé procesy .....                   | 45 |
| 3.6.9   | Specifikace rolí a přístupových práv .....                 | 45 |
| 3.7     | Návrh BI vrstvy .....                                      | 46 |
| 3.8     | Implementace DW .....                                      | 48 |
| 4       | Závěr .....  | 49 |
| 4.1     | Shrnutí cílů .....   | 49 |
| 4.2     | Subjektivní zhodnocení očima autora .....                  | 50 |
| 5       | Literatura .....   | 52 |
| 6       | Přílohy .....  | 55 |



## Seznam zkratk

|        |  |
|--------|--|
| DW     | Data Warehouse (Datový sklad)                            |
| DM     | Data Mart (Datové tržiště)                               |
| OF     | Oracle Financial   |
| GE     | GE Money Bank, a.s.                                      |
| Oracle | Databáze Oracle 11g                                      |
| BI     | Business Intelligence                                    |
| BO     | Business Objects   |
| OLAP   | Online Analytical Processing                             |
| KPI    | Key Performance Indicator (Klíčové ukazatele výkonnosti) |
| EUL    | End User Layer   |
| ETL    | Extract Transfer Load                                    |
| ADS    | Analytical Data Storage                                  |
| MIS    | Management Information System                            |
| RFP    | Request for Proposal                                     |
| RFI    | Request for Information                                  |
| SLA    | Service Level Agreement                                  |

## **Seznam tabulek**

|   |    |
|---|----|
| Tabulka č. 1: Příklad detailu RFP – otázky k HRIS ..... | 33 |
| Tabulka č. 2: Příklad hodnotícího formuláře .....       | 35 |
| Tabulka č. 3: Příklad definice extraktů pro GL.....     | 38 |
| Tabulka č. 4: Příklad fyzického datového modelu.....    | 40 |

## **Seznam grafů**

Graf č. 1: Počet unikátních přístupů (za den) do webových reportů nad DW .....46

## Seznam obrázků

|  |    |
|--|----|
| Obrázek č. 1: Příklad struktury tabulky .....  | 5  |
| Obrázek č. 2: Příklad SQL skriptu – výsledky prodeje hypoték v GE .....                            | 6  |
| Obrázek č. 3: Logický datový model – příklad.....  | 7  |
| Obrázek č. 4: Fyzický datový model – příklad .....   | 8  |
| Obrázek č. 5: Architektura DW dle W.Inmona.....  | 11 |
| Obrázek č. 6: Architektura DW dle R. Kimballa .....  | 12 |
| Obrázek č. 7: Obrazovka z aplikace Oracle Financials – modul pro zadání pohledávkové faktury ..... | 15 |
| Obrázek č. 8: Logo GE Money Bank, a.s. ....  | 16 |
| Obrázek č. 9: Organizační struktura divize Financí v GE Money Bank, a.s. ....                      | 16 |
| Obrázek č. 10: Obrazovka z aplikace Oracle Discoverer .....  | 29 |
| Obrázek č. 11: Ilustrační příklad vrstvy EOL .....   | 29 |
| Obrázek č. 12: Ilustrační příklad formátu RFP v oblasti podmínek formy nabídky..<br>.....          | 34 |
| Obrázek č. 13: Architektura výsledného řešení uvedená v rámci studie proveditelnosti<br>.....      | 37 |
| Obrázek č. 14: Logický model pro GL Daily Balances .....   | 39 |
| Obrázek č. 15: Příklad chybějících metadat ve formě komentářů .....                                | 41 |
| Obrázek č. 16: Popis toku dat .....  | 42 |
| Obrázek č. 17: Příklad transformace a nahrání dat pomocí PL/SQL.....                               | 43 |
| Obrázek č. 18: Příklad PHP reportingu.....   | 48 |

# 1 Úvod

Historický vývoj poslední doby ukazuje, že chce-li management organizace či podnikatelského subjektu přežít konkurenční boj v současném světě, kdy se nepotýká pouze s konkurencí lokální, ale v rámci celosvětové globalizace také s konkurenty, o kterých jeho předchůdci nemuseli vůbec uvažovat, je naprosto nezbytné, aby mimo skvělou produktovou nabídku a fungující klíčové procesy, měl také k dispozici co nejaktuálnější správně strukturovaná a očištěná data o skutečném fungování jimi řízeného podniku. Nároky na rychlost rozhodování vzrostly za poslední dobu několikanásobně a reflektování skutečného stavu na trhu je minimálně stejně důležité jako mít správný úsudek a schopnost odhadnout budoucí vývoj. K tomu, aby byl management schopný těmto požadavkům dostát, je potřeba, aby měl k dispozici fungující a spolehlivé datové podklady.

Tyto potřeby jsou stále ve vyšší míře kladeny za cíl podnikové informatice, která je z pohledu umístění dat k nim nejbližší. Ovšem v průběhu času se ukázalo jako značně neefektivní vytahovat požadované informace přímo z primárních systémů, zejména z důvodu zvýšené zátěže na aplikace, jejichž účelem není reporting a pak také dochází velmi často k multiplikaci dat případně jinému zkreslení požadovaných údajů. Těmto požadavkům nejlépe odpovídá vytvoření systému pro podporu rozhodování a jednotného datového úložiště, kam budou směřovat veškeré podstatné informace z primárních systémů, bude probíhat jejich transformace, čištění a časové rozlišení a tento systém bude také přizpůsoben pro co nejsnadnější přístup k takto upraveným datům, ideálně ve formě „*Business Intelligence*“ nástrojů.

Samotná implementace datového skladu ovšem rozhodně není jednoduchá záležitost a její podcenění v kterékoli fázi se rozhodně projeví ve finálním řešení. V lepším případě určitou nepohodlností a složitostí při jeho užívání, v tom horším mohou být následky pro celý datový sklad natolik fatální, že data v něm obsažená nebudou mít správnou vypovídající hodnotu a jeho jakékoliv použití bude nemožné. Vzhledem k tomu, že proces implementace datového skladu je poměrně drahou záležitostí, ať už se jedná o řešení vlastními silami organizace nebo pomocí dodavatele, je nasnadě pokusit

se minimalizovat riziko neúspěchu při nasazení dodržováním určitých platných zásad a principů, které jsou víceméně shodné jako při vývoji jakékoli jiné aplikace.

Jedním z velkých rizik, které sice neohrožují proces implementace datového skladu přímo, ale pro správné fungování týmu zodpovědného za její realizaci je podpora vrcholového vedení organizace tomuto projektu. Nezřídka se stává, obzvláště u nadnárodních korporací, že projekt implementace datového skladu přinese první výsledky až v době, kdy jeho přínosy již nepocítí současné vedení. Lidé ve vrcholovém managementu by tedy měli být natolik zkušení, že i přes to, že jejich snaha přinese přidanou hodnotu jejich nástupcům, budou procesu implementace věnovat dostatečnou pozornost a bude na něj nahlíženo jako na jednu z priorit co se týká rozvoje podnikových aplikací.

## 1.1 Cíle práce

Cílem této práce je seznámit její čtenáře s principy vývoje datového skladu dle použití teorie „*životního cyklu aplikace podnikové informatiky*“, kdy primární důraz je kladen na oblasti

- Plánování a příprava datového skladu
- Analýza a návrh
- Implementace (pouze v teoreticko metodologické části)

Mimo teoretické poznatky je součástí a také hlavním cílem praktická ukázka aplikace těchto metod v praxi při implementaci datového skladu pro účely reportingu po nasazení nového účetního systému *Oracle Finance* ve společnosti GE Money Bank, a.s., ověření jejich účinnosti a upozornění na možné nedostatky, které se objevily při jeho vývoji. Součástí je také velmi stručné nastínění použitého BI řešení nad tímto datovým skladem. Použitá technologie při stavbě výše uvedeného řešení je databáze od firmy *Oracle* ve verzi *11g*, z tohoto důvodu mohou některé z ukázkových skriptů a obrázků obsahovat klíčová slova fungující jenom v tomto prostředí.

Z důvodů popsaných výše je tato práce rozdělena na dvě části, první část je teoreticko-metodologická, jejímž obsahem je také vymezení základních pojmů a teoretický popis procesů obsažených v jednotlivých oblastech cyklu, druhá část se pak zabývá praktickou implementací uvedených poznatků při zavádění datového skladu.

Cílem této práce není detailní popis jednotlivých technologických postupů, případně popis skriptů, kterými byl tento datový sklad vytvořen, jejich příklady jsou uvedeny v příloze, ovšem bez jakýchkoli komentářů.

## 1.2 Důvod pro výběr tématu

Jako autor této práce jsem strávil většinu svého dosavadního, byť poměrně krátkého, profesního života na hraně mezi klasickým IT a businesssem v divizi Financí ve společnosti GE Money Bank, a.s.. Tato pozice mě, svým způsobem, rozevřela oči, protože do této doby jsem nahlížel na implementace systémů čistě z technického hlediska a nedokázal jsem si představit dopady, jaké může mít jakákoliv změna na chod celé společnosti. V samotných začátcích jsem působil jako čistě datový analytik podporující činnost firmy dolováním dat z různých dílčích databází nebo přímo ze zdrojových systémů. Jedním z prvních setkání s problematikou tvorby datových skladů bylo spíše datové tržiště, a to nad starým účetním systémem *FEIS*, kdy jeho funkce byly značně zjednodušené, neboť nedocházelo prakticky k žádné transformaci dat a jednalo se pouze o čistý extrakt z tohoto zdrojového systému jedna ku jedné s tím, že přidaná hodnota vzniklého data martu byla spíše v usnadnění přístupu k datům, ulehčení zátěže zdrojového systému a časovému rozlišení dat. Z důvodů sjednocení účetního systému skrze celou korporaci GE, došlo v minulém roce k nasazení *Oracle Finance* a tím pádem také vyplynula potřeba vytvořit pro něj datové úložiště, jehož tvorby jsem se aktivně účastnil. V průběhu vstupní analýzy se posléze ukázalo, že data potřebná pro celou divizi Financí nejsou jenom účetní čísla a tím pádem vzniknul datový sklad místo datového tržiště, kdy se v jeho případě do budoucna počítá jako s datovou základnou pro veškerá potřebná data všech oddělení.

## 2 Teoreticko-metodologická část

Tato část práce se zabývá objasněním základních pojmů a metodologií řešení aplikací podnikové informatiky pomocí životního cyklu.

### 2.1 Vymezení základních pojmů

Primárním cílem této práce není detailní rozbor jednotlivých metodologií tvorby DataWarehouse, z tohoto důvodu nebude v rámci této kapitoly věnován prostor pro popis historie vzniku jednotlivých použitých pojmů, ale pouze jejich stručná definice a popis odpovídající posledním trendům v jejich použití.

#### 2.1.1 Databáze

Pojem databáze lze charakterizovat jako „*kolekce vzájemně souvisejících souborů dat*“<sup>1</sup> resp. „*určitá uspořádaná množina informací (dat) uložených na paměťovém médiu*“<sup>2</sup>, z čistě technického hlediska lze pak databázi charakterizovat jako systém, který má za úkol uchovávat požadovaná data tak, aby jejich dostupnost byla co nejrychlejší a velikost co nejmenší.

Mezi nejznámější dodavatele databázových řešení se řadí společnosti jako Oracle se stejnojmennou databází, Microsoft s řešením MS SQL Server, DB2 od IBM a další, mezi nejznámější volné distribuce (open source) pak patří MySQL a PostgreSQL, kdy zejména první jmenované je velmi oblíbené u webových řešení (jednoduché internetové obchody, prezentace firem...) pro jeho jednoduchou instalaci a relativně rychlé odezvy při dolování dat.

##### 2.1.1.1 Tabulka v databázi

Tabulka je jedním ze základních objektů každé databáze sloužící k přímému uložení požadovaných dat. Tabulka se skládá z pevně vymezeného počtu sloupců, pro které platí, že každý z nich má přesně vymezený datový typ, a mimo technické možnosti

---













1 Citace: GÁLA L, POUR J, ŠEDIVÁ Z, (2009), Podniková informatika. Praha: Grada Publishing, a.s., str. 66

2 Citace: Wikipedia, dostupná online z <http://cs.wikipedia.org/wiki/Datab%C3%A1ze>, datum aktualizace 9.7.2011



jednotlivých řešení teoreticky neomezený počet řádků, kde se nacházejí samotné hodnoty odpovídající struktuře dle sloupců.<sup>3</sup>

Obrázek č. 1: Příklad struktury tabulky

| Column  | Type        | Nullable | Default | Comments |
|---|-------------|----------|---------|----------|
|  TIME_KEY    | NUMBER      | Y        |         |          |
|  DEN_YOY     | NUMBER(4)   | Y        |         |          |
|  DEN_V_TYDNU | VARCHAR2(3) | Y        |         |          |
|  SVATEK      | VARCHAR2(3) | Y        |         |          |
|  DATUM       | DATE        | Y        |         |          |
|  DEN         | VARCHAR2(2) | Y        |         |          |
|  MESIC       | VARCHAR2(2) | Y        |         |          |
|  ROK         | VARCHAR2(4) | Y        |         |          |
|  ID_PRODUKT  | VARCHAR2(7) | Y        |         |          |
|  ID_AKCE     | CHAR(3)     | Y        |         |          |
|  HODNOTA     | NUMBER      | Y        |         |          |
|  AKTUALIZACE | DATE        | Y        |         |          |

Zdroj: Vlastní tvorba

### 2.1.1.2 Views

Předem definované SQL příkazy, které se vzhledem k uživateli databáze chovají jako klasická tabulka (lze na nich vykonávat příkaz select) ovšem jakákoliv úprava dat je nemožná. Je nutno podotknout, že je za potřeby určitá obezřetnost při jejich vytváření, protože platí, že jak dlouho běží samotný SQL dotaz, stejně dlouho se bude zobrazovat i z něj vytvořené view. Na druhou stranu je ovšem toto výhodné, pokud je cílem dát uživateli plný přístup na určitá data online tak, jak jsou ve zdrojových tabulkách, a tento uživatel neovládá jazyk SQL. V případě vzniku datawarehouse se tohoto využívá zejména pro zobrazování dat pomocí uživatelských účtů, kdy koncovým uživatelům nejsou dána práva náhledu přímo na tabulky, ale jejich obraz jedna ku jedné právě ve formě views. Speciálním případem view jsou tzv. „*Materializovaná view*“, kdy tato slouží opět k zobrazení předem definovaného SQL dotazu, ovšem jeho spuštění není již online, avšak závisí na nastavení aktualizace daného skriptu a v případě zobrazení tohoto view dostane uživatel poslední instanci těchto dat. Jejich využití při tvorbě datawarehouse je nesmírné v případě rozsáhlých SQL skriptů, které mají dlouhý čas

<sup>3</sup> Zdroj: Wikipedia, dostupná online z [http://cs.wikipedia.org/wiki/Tabulka\\_\(datab%C3%A1ze\)](http://cs.wikipedia.org/wiki/Tabulka_(datab%C3%A1ze)), datum aktualizace 9.7.2011

běhu a jsou opakovaná, kdy současně platí, že data v tomto datovém skladu jsou aktualizována pouze v předem známých časových intervalech (nejčastěji jednou denně).

### 2.1.2 Jazyk SQL

Jedná se o standardizovaný dotazovací jazyk „*Structured Query Language*“, pomocí kterého přistupujeme k datům uloženým v databázích. Nejedná se o jazyk programovací, neboť v jeho standardní podobě není obsažena žádná možnost pro přímé využití hardwaru serveru (např. paměti), k těmto účelům slouží např. PL/SQL od společnosti Oracle, které v sobě má již zabudovanou podporu proměnných, rozhodovacích podmínek a cyklů a s jeho pomocí lze vytvářet procedury a funkce.

Základní příkazy jazyka SQL lze rozdělit dle jejich účelu na

- Příkazy pro manipulaci s daty (*select, update, insert, delete, ...*)
- Příkazy pro definici dat (*create, drop, truncate, ...*)
- Příkazy pro řízení dat (*grant, rollback, commit, ...*)
- Ostatní příkazy (*příkazy pro správu dtb.*)

**Obrázek č. 2:** Příklad SQL skriptu – výsledky prodeje hypoték v GE

```
select prod.produkt, prod.segment, prod.ID_PRODUKT, ID_AKCE, sum(HODNOTA)
  from dr_data_bo t, dr_cis_produkty prod
 where t.time_key >= to_number(to_char(to_date('1.5.2011'), 'J')) and
        t.id_akce = 'NVS' and
        t.id_produk in ('MORTG')
 and prod.id_produk = t.id_produk
 group by prod.produkt, prod.segment, prod.ID_PRODUKT, ID_AKCE
```

**Zdroj:** vlastní tvorba, GE Money Bank, a.s.

### 2.1.3 Metadata

Metadata lze charakterizovat jako data o datech, přesněji pak „*metadata jsou data popisující data jiná. Jejich prostřednictvím se lze na data dotázat, jsme schopni data doplňovat, konsolidovat je, vzájemně je synchronizovat a integrovat*“<sup>4</sup>.

Jak vyplývá z výše uvedeného je oblast metadat při tvorbě datawarehouse značně důležitá, protože popisuje, co přesně má uživatel ve kterém ze sloupečků

---

<sup>4</sup> Zdroj: GÁLA L, POUR J, ŠEDIVÁ Z, (2009), Podniková informatika. Praha: Grada Publishing, a.s., str. 62

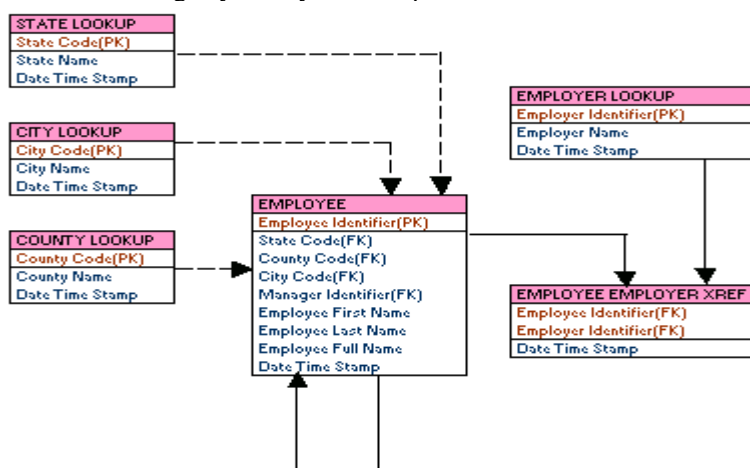
tabulky, popř. view, hledat. Opomenutí tvorby metadat, stejně jako absence logického modelu je poměrně častým jevem, který sice na chod samotného datawarehouse nemá přímý vliv, ovšem značně znesnadňuje jeho používání, případně budoucí rozšiřování. (tuto chybu obsahuje i obrázek č. 1, kdy při vzniku této tabulky není u jediného sloupečku komentář k tomu, co je jeho obsahem).

### 2.1.4 Logický a fyzický datový model

Logický datový model popisuje datové struktury v obecné rovině, nezávisle na konkrétním databázovém prostředí. Jeho součástí tedy nejsou informace o způsobu uložení dat a obvykle ani jejich datové typy, z tohoto důvodu je přímé převedení do příkazů pro definici dat pomocí SQL nemožné. Využití logického datového modelu je vhodné zejména v případech, kdy ještě není rozhodnuto o konkrétní technické platformě, na které datawarehouse poběží. Součástí logického návrhu jsou následující prvky<sup>5</sup>:

- Pojmenování datových entit
- Definování vazeb mezi těmito entitami
- Definici všech atributů
- Určení primárních i cizích klíčů

Obrázek č. 3: Logický datový model – příklad



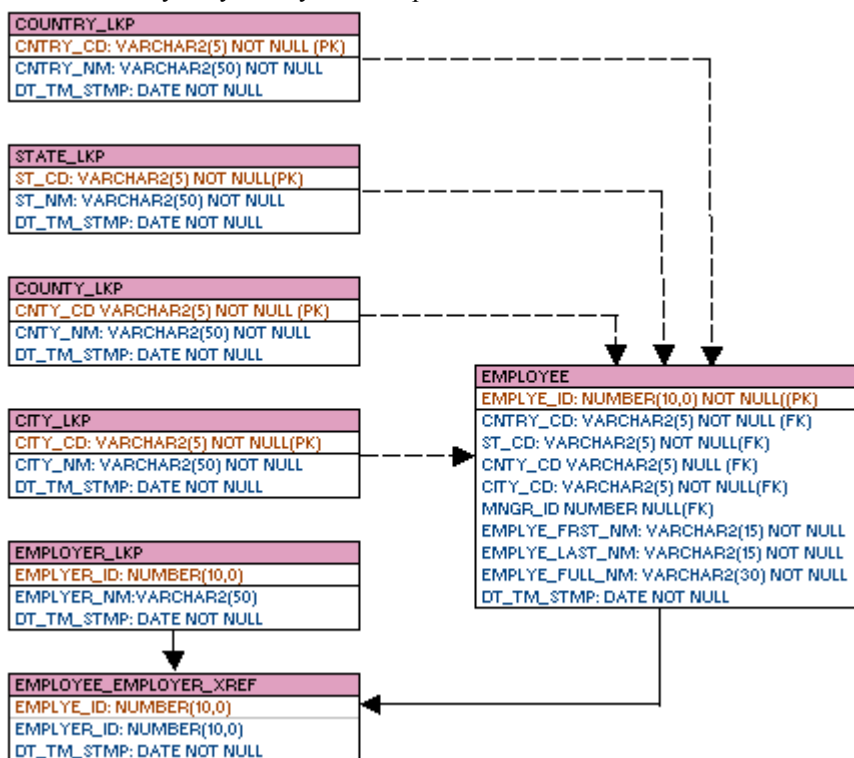
**Zdroj:** Learn data modeling, dostupné online z <http://www.learndatamodeling.com/ldm.htm>, aktualizováno 10.7.2011

<sup>5</sup> Řízení internetového projektu, dostupná online z <http://www.scribd.com/doc/53062183/18/Fyzick%C3%BD-datov%C3%BD-model>, aktualizováno 11.7.2011

Fyzický datový model oproti tomu reprezentuje fyzickou strukturu datového návrhu, jeho cílem je již přizpůsobení datového modelu konkrétní datatabázové technologii. Tento model tedy obsahuje specifická nastavení pro jednotlivé tabulky, např. datové typy jednotlivých sloupců. Převod logického modelu na fyzický obsahuje například následující kroky<sup>6</sup>:

- Převod entit na tabulky
- Převod atributů v těchto entitách na sloupce
- Konkretizaci jednotlivých datových typů těchto sloupců
- Zavedení integritních omezení
- Definování vztahů mezi tabulkami pomocí cizích klíčů

Obrázek č. 4: Fyzický datový model – příklad



**Zdroj:** Learn data modeling, dostupné online z <http://www.learndatamodeling.com/ldm.htm>, aktualizováno 10.7.2011

<sup>6</sup> Řízení internetového projektu, dostupná online z <http://www.scribd.com/doc/53062183/18/Fyzick%C3%BD-datov%C3%BD-model>, aktualizováno 11.7.2011

## 2.1.5 Data Warehouse

Při charakteristice pojmu datového skladu „Data Warehouse“, dále jen DW, jsou v rámci této práce použity materiály obou stěžejních představitelů, a to Williama Inmona a Ralpha Kimballa, kdy první jmenovaný je považovaný za „otce datových skladů“ a druhý za „otce business intelligence“. Tito dva hlavní představitelé se víceméně shodli na samotné definici DW, ovšem zastávají velký rozdíl v samotné tvorbě těchto struktur a jejich následné implementaci, kdy tento rozdíl je popsán níže v této práci.

### 2.1.5.1 Definice Data Warehouse dle W. Inmona

*„A data warehouse is a subject-oriented, integrated, nonvolatile, and time-variant collection of data in support of management's decisions. The data warehouse contains granular corporate data. (Datový sklad je subjektivně orientovaný, integrovaný, stálý a časově neměnný soubor dat sloužící pro podporu rozhodování managementu. Datový sklad obsahuje firemní data v určité granularitě.)“<sup>7</sup>*

Jednotlivé vlastnosti obsažené v této definici lze dále charakterizovat následujícím způsobem:

- **Subjektivně orientovaný**

definuje, že konkrétní DW obsahuje data rozdělená dle jejich typu a ne dle způsobu, kterým byla pořízena. Jako příklad lze uvést entitu zákazníka, kdy způsob jeho získání může být skrze různé produkty a tím pádem skrze různé zdrojové systémy, ovšem v DW by se měla tato informace objevit právě a pouze jednou.

- **Integrovaný**

definuje, že data by měla být ukládána v rámci celé organizace a ne jenom v rámci jednoho konkrétního oddělení. Pro tyto potřeby dochází v různých fázích zpracování DW k transformaci dat, taky aby odpovídala jednotnému zadání, například je sjednoceno jméno zákazníka, způsob interpretace jeho

---

<sup>7</sup> Citace: INMON, William Harvey. Building the Data Warehouse. 3rd ed. Wiley Publishing. Canada. 2002. str. 31

pohlaví, veškeré finanční operace jsou převedeny na Kč, atd. Z pohledu vzniku samotného DW se tato vlastnost jeví jako nejdůležitější.

- **Stálý**

Data obsažená v DW by se, na rozdíl od zdrojových systémů, neměla měnit, jejich přírůstky zajišťují externí zdroje a ona samotná by měla být obsažena v DW do konce jeho životnosti. Aktuálnost dat je řešena pomocí příznaku pro platný zápis a případné znevalidnění resp. výmaz záznamu by taktéž měl být realizován pomocí příznaku (obvykle v jiném sloupci). Tento způsob zpracovávání se nazývá „*historizace dat*“ viz následující položka seznamu.

- **Časově neměnný**

Zejména z důvodů potřeby zpracovávat i historické analýzy, případně analýzy za předem vymezené časové období, je nutné, aby veškerá data, která se v DW ocitnou, tam také zůstala a nesla s sebou informaci o tom, ve kterém čase byla pořízena a ke kterému časovému období se vztahují. Jako příklad lze použít stav objednávky, kdy ve zdrojovém systému bude uložena pouze aktuální informace, protože historie schvalování není z provozních důvodů až tak zajímavá, tak právě v DW lze jednoduše tuto sestavu získat, protože data v něm obsažená jsou přidávána a aktualizována v předem definovaném časovém okamžiku.

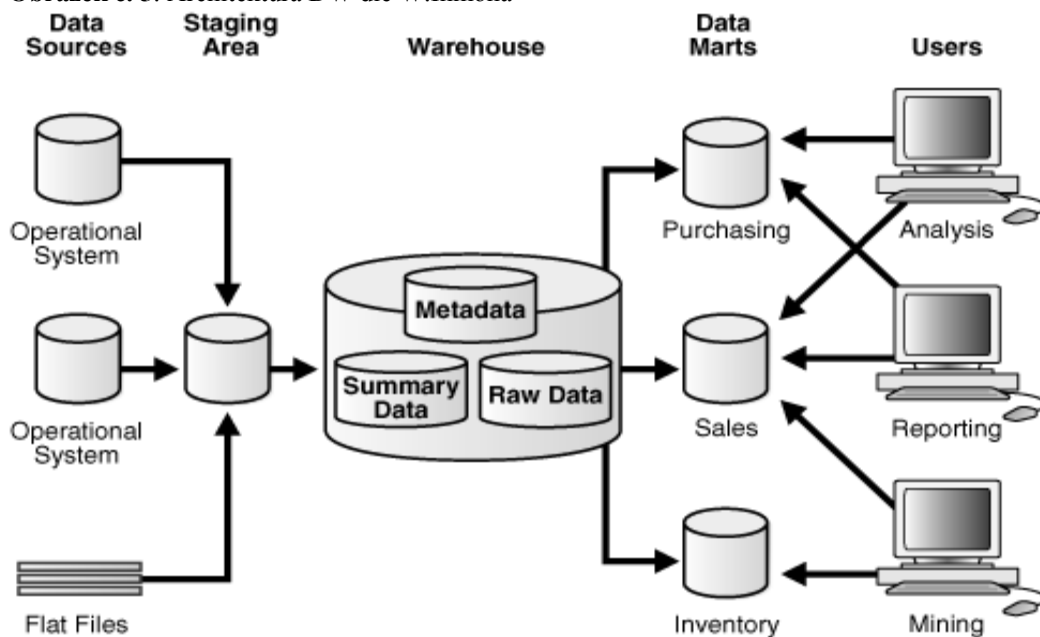
- **Granularita dat**

Tento pojem Inmon definuje „*Granularity refers to the level of detail or summarization of the units of data in the data warehouse. The more detail there is, the lower the level of granularity. The less detail there is, the higher the level of granularity. (Granularita vypovídá o úrovni detailu nebo souhrnech dat v DW. Čím je vyšší detail, tím je granularita menší a naopak)*“<sup>8</sup> a v jeho pojetí by granularita DW měla být co nejvyšší. V tomto případě je potřeba brát v potaz, k čemu má daný DW sloužit, pro manažerské rozhodování při řízení organizace určitě není až tak podstatný seznam jednotlivých transakcí u každého klienta zvláště, zejména pokud je jejich detail dle nařízení ČNB uchovávan v zdrojovém systému po celou dobu jeho životnosti, ale spíše jejich souhrn na denní bázi.

---

8 Citace: INMON, William Harvey. Building the Data Warehouse. 3rd ed. Wiley Publishing, Canada. 2002. str. 43

Obrázek č. 5: Architektura DW dle W. Inmona



**Zdroj:** Data Warehousing Concepts, dostupné online z [http://download.oracle.com/docs/cd/B28359\\_01/server.111/b28313/concept.htm](http://download.oracle.com/docs/cd/B28359_01/server.111/b28313/concept.htm), aktualizováno 10.7.2011

### 2.1.5.1 Definice Data Warehouse dle R. Kimballa

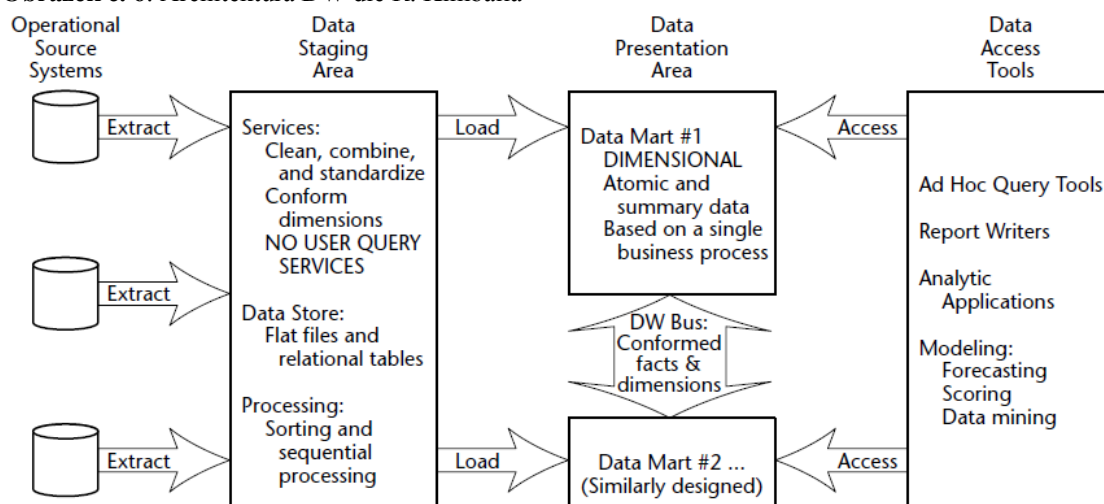
*„Data Warehouse is a copy of transaction data specifically structured for querying and reporting. (DW je kopie transakčních dat speciálně strukturovaných pro potřeby datového dotazování a reportingu)“<sup>9</sup>*

Na první pohled je o dost jednodušší než definice W. Inmana, ovšem je potřeba mít na paměti, že tato charakteristika vychází především z jeho přístupu ke tvorbě DW, jak bude popsáno v následující kapitole.

---

<sup>9</sup> Citace: The Data Warehousing Information Center, dostupné z <http://www.dwinfocenter.org/defined.html>, aktualizováno 11.7.2011

**Obrázek č. 6:** Architektura DW dle R. Kimballa



**Zdroj:** Kimball R., Ross M., The Data Warehouse Toolkit, 2. ed, Wiley Publishing. Canada. 2002, str. 7

### 2.1.5.2 Rozdíly ve tvorbě a implementaci DW dle W. Inmona a R. Kimballa

Ani jeden z obou přístupů nelze jednoznačně označit za správný či špatný oproti tomu druhému. Hlavní rozdíl spočívá ve způsobu tvorby DW.

W. Inmon je zastáncem tzv. „top-down“<sup>10</sup> principu kdy v rámci celého podniku by měl existovat pouze jeden datový sklad a jednotlivá datová tržiště by z něj měla vycházet jako z primárního zdroje. Data uložená v DW by měla být v normalizované podobě a jednotlivá datová tržiště vytvořena využitím dimenzionálního přístupu. Jeho přístup se dá charakterizovat pomocí tvrzení

*„Do not do anything until you have designed everything. (nedělejte nic, dokud nenaplánujete všechno)“<sup>11</sup>*

Oproti tomu R. Kimball stanovuje DW jako spojnicí různých datových tržišť v rámci podniku, tzv. „bottom-up“<sup>12</sup> přístup, a data v něm obsažená odpovídají dimenzionálnímu modelu.<sup>13</sup> Jeho přístup lze velmi dobře charakterizuje tvrzení

10 Wikipedia, dostupná online z <http://cs.wikipedia.org/wiki/Datab%C3%A1ze>, aktualizováno 9.7.2011

11 Citace: Kimball vs. Inmon...or, How to build a Data Warehouse, Toolbox, dostupné odkazem z <http://it.toolbox.com/blogs/confessions/kimball-vs-inmon-or-how-to-build-a-data-warehouse-10987>, aktualizováno 23.7.2011

12 Wikipedia, dostupná online z [http://en.wikipedia.org/wiki/Data\\_warehouse](http://en.wikipedia.org/wiki/Data_warehouse), aktualizováno 23.7.2011

13 Zdroj: Bill Inmon vs R. Kimball, dostupné online z <http://www.1keydata.com/datawarehousing/inmon-kimball.html>, aktualizováno 11.7.2011



*„Let everybody build what they want when they want it, we will integrate it all when and if we need to. (Nechte všechny stavět, co chtějí a kdy chtějí, vzájemnou integraci provedeme, až budeme potřebovat)“<sup>14</sup>*

Mezi hlavní výhody Inmanova přístupu patří zejména velmi snadná a rychlá tvorba datových tržišť, čehož je dosaženo právě díky tomu, že DM vycházejí ze samotného datového skladu a pak také snadnost importů dat z primárních systémů do tohoto DW, opět z toho důvodu, že je tento systém jediný v celé společnosti a integrovaný přímo do celkové architektury. Výhody Kimballova přístupu pak tvoří zejména samotná rychlost vývoje a změn v takto postavených DW včetně poměrně nízké kapitálové náročnosti.

Oproti tomu mezi hlavní nevýhody Inmanova přístupu lze jednoznačně zařadit časovou a kapitálovou náročnost implementace DW, kdy, zejména v době potřeby rychlých a relativně levných řešení, toto může být velkou překážkou pro rozhodnutí jít jeho cestou. Tohoto je si vědom i sám W. Inmon a jeho pojednání k tomuto problému lze shrnout, že přece není nutné vytvořit celý DW najednou se všemi extrakty, ale stačí nachystat vhodné struktury a data do něj ukádat průběžně, ovšem je otázkou, zda takto naplněný DW bude splňovat požadavky na něj kladené a zda nedojde k postrádání stěžejních oblastí dat. Nevýhodu Kimballova přístupu lze spatřit zejména v oblasti přenosu dat a jejich čištění, kdy je potřeba pro každý samotný DM nadefinovat vlastní sadu extraktů a provést samostatná čištění a taktéž následná integrace jednotlivých DM do výsledného datového skladu není jednoduchou záležitostí a je složitější docílit nulové redundance dat.

V současné době lze pozorovat trend ve spojování obou přístupů, kdy v rámci podnikové architektury je navrženo globální úložiště dat se všemi svými atributy, avšak jeho realizace a implementace následně probíhá velmi živelně a spíše připomíná implementaci skrze integraci jednotlivých datových tržišť. Toto je způsobeno zejména nižšími náklady pro jednotlivé suborganizační jednotky podniku, kdy se naplno projevuje efekt dělení nákladů, kdy celková částka za implementaci DW se rozdělí na větší počet menších a jejich obhajoba před jednotlivými členy vedení společnosti je

---

<sup>14</sup> Citace: Kimball vs. Inmon...or, How to build a Data Warehouse, Toolbox, dostupné odkazem z <http://it.toolbox.com/blogs/confessions/kimball-vs-inmon-or-how-to-build-a-data-warehouse-10987>, aktualizováno 23.7.2011

snažší, než by tomu bylo u souhrnu za celé řešení a také čas potřebný na jednotlivé implementace a tím pádem možnost efektivně využívat tento DW je značně kratší.

### **2.1.6 OnLine Analytical Processing**

*„Informační technologie založená především na koncepci multidimenzionálních databází. Jejím hlavním principem je několikadimenzionální tabulka umožňující rychle a pružně měnit jednotlivé dimenze a měnit tak pohledy uživatele na modelovanou ekonomickou realitu“<sup>15</sup>*

Jejich využití je možné již přímo při stavbě DW tím, že se faktové tabulky budou dělit podle více dimenzí. Samotné OLAP technologie se realizuje dle typu přístupu k multidimenzionalitě v následujících variantách

- MOLAP – charakterizována uložením dat v multidimenzionálních-binárních kostkách
- ROLAP – multidimenzionalita je řešena pomocí využití relační databáze
- HOLAP – kombinace dvou předchozích přístupů
- DOLAP – přístup, kdy je výsledná OLAP kostka exportována přímo na lokální počítač z centrálního úložiště

Všechny tyto typy mají společnou vlastnost, a to je uložení dat tak, aby jejich zpracování a zejména pak rychlost odezvy na dotazy byla co nejkratší.

### **2.1.7 Oracle Financials**

Jedná se o účetní systém od společnosti Oracle, který je postavený na bázi klient-server, využívající technologii *Java*<sup>16</sup> prostřednictvím internetového browseru. Tento produkt je pro potřeby České republiky lokalizován do češtiny a odpovídá legislativním požadavkům. Tento systém je také velmi otevřen pro různá podpůrná řešení, jako například systém pro workflow objednávky popřípadě software zajišťující vytěžení dat z příchozích dokumentů atd.

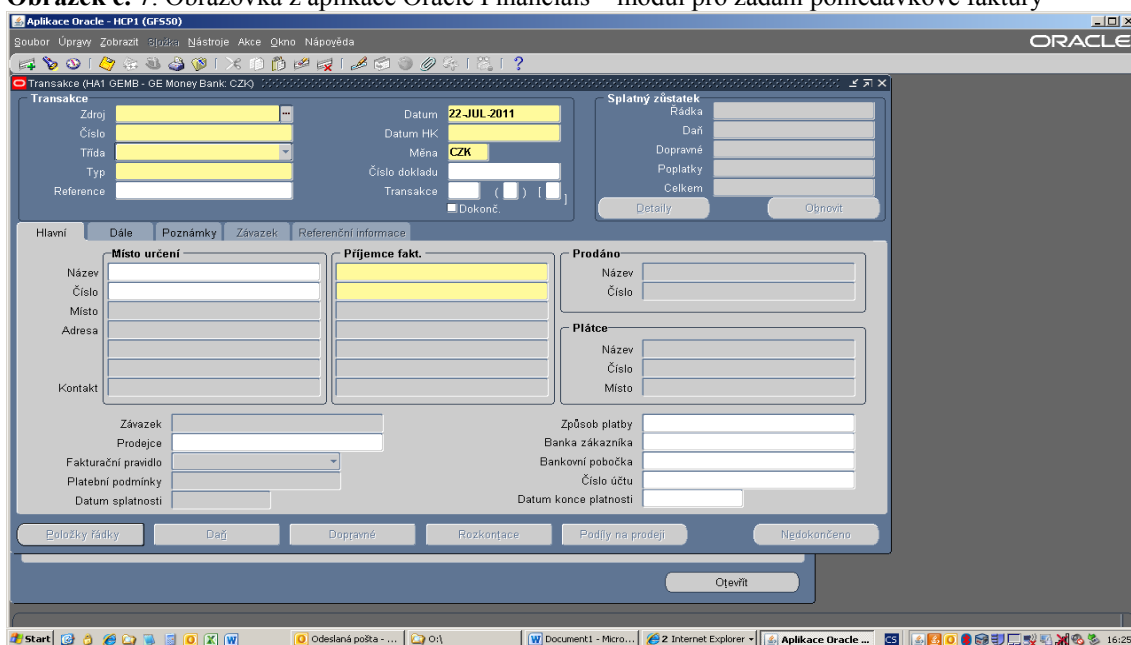
---

<sup>15</sup> Citace: GÁLA L, POUR J, ŠEDIVÁ Z, (2009), Podniková informatika. Praha: Grada Publishing, a.s., str. 223

<sup>16</sup> Objektově orientovaný programovací jazyk, známý především svojí přenositelností mezi různými platformami. Informace o výrobci dostupné online z <http://www.java.com/en/>, aktualizováno 23.7.2011, Zdroj: Dioné – ZČU, dostupná online z <http://v1.dione.zcu.cz/java/uvod.html>, aktualizováno 23.7.2011

Jeho provoz pro společnost GE v České republice je zajištěn pomocí outsourcingu z Velké Británie, kde jsou provozovány servery, a Číny, kde nalezneme podporu aplikace v první linii. Z tohoto důvodu není možný přímý přístup do jeho databází a tím pádem, mimo předdefinovaných tiskových sestav přímo z OF a omezenému využití nástroje Oracle Discoverer, je jedinou možností, jak získat data a provést nad nimi nějaké analýzy, stavba DW kdy dochází k denním přenosům a úpravě předdefinovaných snímků pomocí tzv. „*Extract-Transform-Load*“<sup>17</sup> ETL procesů.

**Obrázek č. 7:** Obrazovka z aplikace Oracle Financials – modul pro zadání pohledávkové faktury



**Zdroj:** vlastní tvorba, instance pro GE Money Bank, a.s.

### 2.1.8 GE Money Bank, a.s.

Jedná se o středně velkou univerzální banku působící na území České republiky, jejíž zaměření prošlo v posledních letech velkou změnou od původní, téměř výhradně retailové banky s divizí pro SME v oblasti zemědělství, což bylo dáno historicky jejím vznikem, ke kterému došlo odkoupením části Agrobanky, a.s. a společnosti zabývající se splátkovým prodejem Multiservis. V současné době je GE Money Bank, a.s. plně univerzální bankou poskytující produkty pro všechny skupiny klientů, kdy v oblasti splátkového prodeje již nepůsobí a naopak se snaží proniknout do oblastí komerčního a investičního bankovníctví. Jediným akcionářem GE Money Bank, a.s. je skupina GE

<sup>17</sup> Pomocí ETL jsou data do DW přenášena, očišťována a nahrávána, více info dostupné online z [http://en.wikipedia.org/wiki/Extract,\\_transform,\\_load](http://en.wikipedia.org/wiki/Extract,_transform,_load), aktualizováno 23.7.2011

Capital se sídlem v USA. V České republice v současné době zaměstnává okolo 4.500 lidí.

**Obrázek č. 8:** Logo GE Money Bank, a.s.

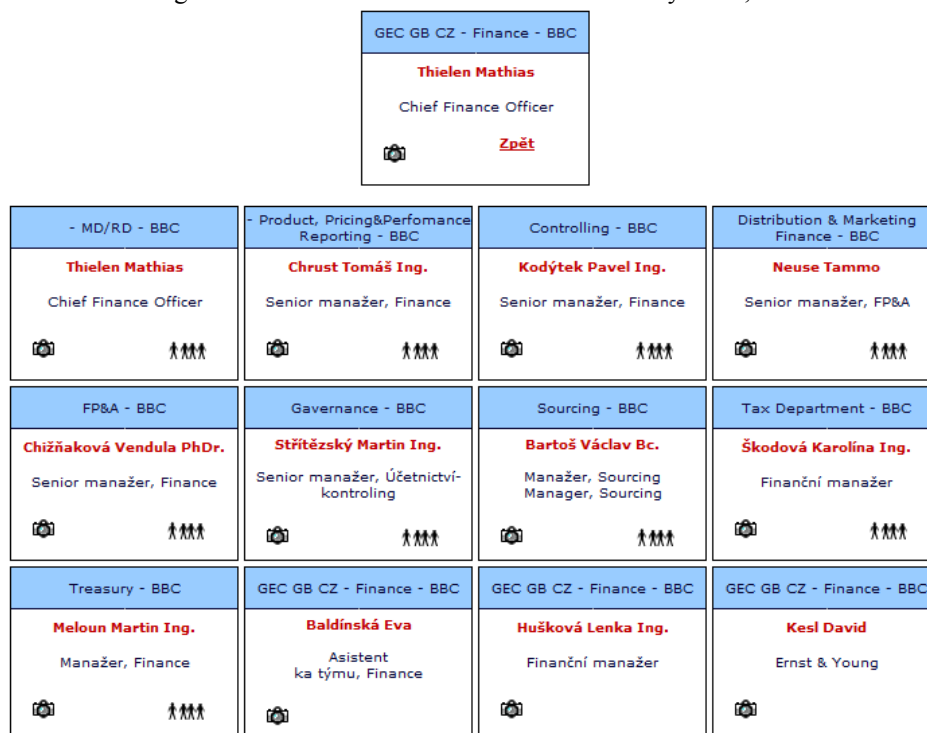


**Zdroj:** GE Money Bank, a.s., dostupné online z <http://www.gemoney.cz/ge/cz/1/nase-spolecnosti/pro-media/loga-spolecnosti>, aktualizováno 23.7.2011

### 2.1.8.1 Divize Financí

Tato divize má za úkol zajišťovat celkovou finanční stránku podniku, jejími stěžejními procesy jsou Controlling & Accounting, Finance Planning & Analysing, Treasury, Pricing, Sourcing a Finance IT, kdy poslední jmenované oddělení má na starost vývoj DW a DM jak pro čistě finanční procesy, tak také pro celý podnik, v případě, že se jedná o data, která mají jakýkoliv dopad na výpočty KPI a profitability.

**Obrázek č. 9:** Organizační struktura divize Financí v GE Money Bank, a.s.



**Zdroj:** vlastní tvorba, platnost organizační struktury k 23.7.2011

## 2.2 Vývoj DW dle životního cyklu aplikace

K samotnému vývoji a implementaci DW lze přistupovat mnoha metodami, ať už se bude jednat o metodu „*big-bang*“<sup>18</sup>, kdy živelně a nekordinovaně vznikne datová struktura, která se bude plnit poloautomaticky, kvalita dat bude značně nespolehlivá a jakýkoli další rozvoj bude otázkou podobně organizovaného dovývoje. Nutno přiznat, že ne zcela vždy je tato metoda neefektivní, zejména v případě potřeby rychlého, levného a alespoň nějakého řešení je její aplikace téměř jediným východiskem, nebo je přístup striktně projektově orientovaný, kdy jednotlivé fáze jsou shodné s klasickým pojetím řízení projektu jakékoliv podoby.

V rámci této práce je přikloněno ke kombinaci metody tvorby DW pomocí procesů popsaných v rámci „*životního cyklu aplikace*“<sup>19</sup>, která vychází ze standardu „*Information Technology Infrastructure Library - ITIL*“<sup>20</sup> a metody „*big-bang*“<sup>18</sup>, kdy finální řešení postupem času dospělo od Inmonova přístupu ke Kimballovým postupům. Hlavní zaměření je zde kladeno na fázi implementace DW a jí předcházejících.

### 2.2.1 Plánování a příprava DW

DW lze považovat za klasickou aplikaci podnikové informatiky a z tohoto důvodu je potřeba jej obsáhnout v rámci informační strategie podniku a začlenit jej do plánování celkového architektonického řešení. Nutno podotknout, že tato fáze bývá obvykle jednou z nejvíce podceňovaných a tato skutečnost posléze komplikuje veškeré následné kroky a značně ztěžuje jak implementaci, tak zejména následné používání DW. Cílem této fáze není pouze shrnout, že DW je potřebný, ale výstupem z ní musí být např.

- Jak (zda vůbec) bude DW realizován
- Jaké jsou očekávání funkcionality
- Cílový obsah dat

---

18 Teorie o vzniku vesmíru, zjednodušeně všechno vzniklo z ničeho velkým třeskem, přeneseně se používá pro živelný a prakticky nekoordinovaný vznik čehokoliv, více dostupné online z [http://en.wikipedia.org/wiki/Big\\_Bang](http://en.wikipedia.org/wiki/Big_Bang), aktualizováno 23.7.2011

19 Zdroj: GÁLA L, POUR J, ŠEDIVÁ Z, (2009), Podniková informatika. Praha: Grada Publishing, a.s., str. 265

20 „*Veřejně dostupný rámec, jenž popisuje nejlepší praktiky ve Správě služeb IT. Poskytuje rámec pro zvládnutí IT v organizaci, pojednává komplexně o službách a zaměřuje se na neustálé měření a zlepšování kvality dodávaných služeb IT, a to jak z pohledu businessu, tak z pohledu zákazníka*“ Citace: ITIL, dostupné online z <http://www.itil.cz/index.php?id=982>, aktualizováno 23.7.2011

### 2.2.1.1 Vstupní analýza

Jejím cílem by mělo být posouzení, jak zapadá DW z pohledu celkového řešení a plánování rozvoje informační strategie a zejména zjistit očekávání zákazníků resp. uživatelů tohoto datového zdroje. V rámci informační strategie by měly být výstupem odpovědi na následující otázky

- Do jaké míry jsou pomocí DW pokryty požadavky a cíle podniku
- Pozice DW v rámci celkové architektury IS
- Vazba na ostatní aplikace
- Zda se jeho implementace nepřekrývá s jinou aktivitou směřující ke stejnému cíli

Jak bylo popsáno výše, velmi důležitým procesem v rámci vstupní analýzy je zjištění uživatelských požadavků a jejich dokumentace tak, aby bylo transparentní, co bylo v původním zadání a co ne, a to zejména z důvodů závěrečného zhodnocení úspěšnosti projektu. Rozhodně není cílem vyhovět veškerým požadavkům, které vzniknou, pokud že tyto nejsou v souladu s celkovými cíly, nejsou smysluplné, anebo jejich poměr cena/výkon není pro podnik akceptovatelný. Tato fáze není otázkou jednoho interview s uživateli, jeho součástí by mělo být několik sezení, kdy bude v počátku dán cílové skupině uživatelů resp. jejím zástupcům prostor pro vyjádření všech jejich poznatků a přání na výstupy z DW, výsledek těchto jednání bude řádně a důkladně dokumentován a výsledek této dokumentace bude stejně skupině předložen v rámci dalšího setkání k oponentuře, zda skutečně reflektuje jejich požadavky z předchozího jednání. Důvodem pro tento postup je sjednocení všech požadavků a hlavně jejich odsouhlasení od uživatelů a zajištění vzájemného pochopení. Nežřídká totiž lidé zodpovědní za implementaci DW nemají prakticky nic společného s denní rutinou koncových uživatelů. V rámci těchto interview je velmi důležité správně sestavit skupinu, která bude vznášet požadavky. Jako velmi vhodné se ukázalo rozdělit úvodní schůzky na více částí, kdy v první dojde k setkání s managementem procesů, která budou využívat data obsažená v DW, protože tyto lidé mají ponětí o celkovém řešení všech stežných procesů a jejich výstupů a posléze se setkat s lidmi, kteří tuto rutinu

skutečně vykonávají, protože tito zase znají dílčí procesy do posledních detailů a jsou schopni vydefinovat skutečně potřebná jednotlivá data. Před samotným představením finálního dokumentu je také vhodné svolat ještě schůzku se všemi zástupci najednou a představit jim tento draft tak, aby bylo možno popřípadě rovnou odchytil vzniklé nesrovnalosti.

Závěrem vstupní analýzy je, mimo uvedené výše, také shrnutí efektů DW pro celý chod podniku a jeho případné ekonomické dopady.

### **2.2.1.2 Plánování projektu DW**

Projektový záměr implementace DW by měl obsahovat všechny nejdůležitější důvody, které vedou k rozhodnutí o jeho zřízení, cíle, pozitivní i negativní dopady na podnik, kdo uvnitř podnikové struktury je zákazníkem dat v něm obsažených. Tento dokument je velmi důležitý zejména ve fázi, kdy je nutno investici do stavby DW obhájit před managementem společnosti a ideálně by měl pomoci získat na svoji stranu všechny zainteresované strany tzv. „*stakeholders*“<sup>21</sup>. Pokud je seznam těchto osob předem znám, je vhodné jej taktéž obsáhnout do projektového záměru jako osoby zastřešující jednotlivé cílové skupiny zákazníků.

Posouzení navrhovaného plánu řešení zejména z pohledu finančních a personálních zdrojů, kdy součástí by mělo být taktéž stručné zhodnocení aktuálního stavu ICT trhu v oblasti vývoje DW tak, aby bylo možno plánovat, nakolik bude tato implementace nákladná např. srovnáním se stavbou DW v podniku o podobné velikosti a zaměření.

Poslední fází plánování projektu DW je finální rozhodnutí o realizaci či zamítnutí předloženého projektu. V případě přijetí projektu a následné realizace je součástí této fáze také rozhodnutí o formě, jakou bude DW postaven, za pomoci dodavatelského řešení, nebo vlastními silami. V případě DW nad celopodnikovým řešením je také možné uvažovat o realizaci prostřednictvím tzv. „*krabicového řešení*“<sup>22</sup>

---

21 Lidé, kteří mají velký vliv v rámci organizace, či na konkrétní projekt, obvykle mají velké rozhodovací pravomoci a jsou schopni ovlivnit jeho průběh. Zdroj: ZIKMUND M., Kdo jsou to vlastně stakeholders a proč a jak se o ně zajímat, dostupné online z <http://www.businessvize.cz/strategie/kdo-jsou-to-vlastne-stakeholders-a-proc-a-jak-se-o-ne-zajimat>, aktualizováno 23.7.2011

22 Již vyvinutý SW víceméně bez možnosti customizace a jakéhokoliv přizpůsobení potřebám zákazníka

resp. typového DW, který je v některých případech dodáván rovnou s primárním systémem. (např. SAP, Oracle Financial). Toto řešení ovšem téměř nikdy neobsahuje veškeré požadavky, které podnik potřebuje a také jeho případné rozšiřování a customizování není zrovna levnou záležitostí.

### **2.2.1.3 Výběr dodavatele DW**

Správné rozhodnutí a výběr finálního dodavatele DW v případě, že jej nebude podnik stavět svépomocí, je klíčové pro jeho úspěšnou implementaci a zejména následné použití. V současné době vrcholový management má tendenci co nejvíce tlačit na náklady a tím pádem se spousta projektů dostává do fáze, kdy se snaží, obrazně řečeno, postavit dálnici s rozpočtem na příjezdovou cestu. Vzhledem k tomu, že efekt po implementaci DW již většinou přímo nepocítí, viz důvody popsány výše, je toto celkem pochopitelné, ovšem rozhodně ne šťastné řešení, tudíž by výsledná cena řešení neměla být podmínkou s největší vahou při hodnocení nabídek.

Samotný proces výběru dodavatele je složen z následujících procesů

- Příprava poptávkového dokumentu
- Zpracování nabídek od dodavatelů, případné konzultace s nimi při jejich vytváření
- Prezentace nabídek, ideálně včetně ukázky referenčního DW
- Hodnocení nabídek
- Výběr dodavatele
- Zadání přípravy úvodní studie

První fází při dodavatelském řešení je vypracování poptávkového dokumentu, kdy tento vychází zejména z projektového záměru. Tento dokument má za cíl co nejpřesněji specifikovat požadované řešení tak, aby oslovený dodavatel co nejlépe mohl zpracovat nabídku a také aby nebyla opoměna žádná z podstatných částí, které má DW obsahovat. Současně je obsahem tohoto dokumentu také informace pro dodavatele o způsobu výběrového řízení a způsobu předání nabídky, včetně jejího požadovaného obsahu.



U fáze, kdy dodavatel zpracovává nabídky, je velmi vhodné již posuzovat komunikaci a způsob, jakým probíhají případné konzultace nad požadovaným řešením, protože je velmi pravděpodobné, že podobná firemní kultura, včetně úrovně komunikace, bude fungovat skrze celou dodavatelskou společnost a je velmi vhodné toto zahrnout také do hodnotících kritérií zejména z důvodu co nejhladšího průběhu následných fází stavby DW.

U prezentace jednotlivých řešení je velmi důležité nenechat se ohromit prezentačními schopnostmi prodejce a skutečně se zaměřit na prezentovaný produkt, zda odpovídá všem požadavkům, které vyplynuly z projektového záměru, stejně jako zda prezentované řešení zapadá do celkové architektury ICT, úroveň poskytnutých záruk, servisu atd. Pro každou z hodnocených oblastí je dobré si předem stanovit jak „*k.o.*“<sup>23</sup> kritéria, tj. podmínky, které splněny být musí, tak pro ty zbylé hodnotící škály, kde se zaznamenává, do jaké míry to které řešení danou oblast splňuje. Každá z oblastí má také svou váhu při finálním rozhodování, kdy, jak bylo popsáno výše, cena by rozhodně neměla mít vyšší váhu než například flexibilita výsledného DW směrem k jeho případnému rozšiřování.

Hodnocení nabídek velmi ovlivňuje průběh předchozí fáze, z toho důvodu je velmi důležité rozdělit si na počátku hodnotící formulář do oblastí, stanovit každé z nich jak dílčí podmínky, tak hlavně celkovou váhu, kterou budou mít pro celkové hodnocení nabídky. Případné pod či přecenění některé z oblastí, jako již výše uvedená cena řešení, v rámci celkového hodnocení bude mít dopad zejména ve fázi implementace a následných. Při implementaci DW je potřeba věnovat dostatečnou pozornost zejména oblastem a cenám při customizaci řešení a následných rozšiřování. V případě že je DW poskytován jako služba od dodavatele, tj. není provozován jako služba dodavatelem, je velmi důležité správně ohodnotit rozsah poskytovaných servisních služeb a záruk za provoz daného systému.

---

23 Jedná se o zkratku Knock Out, která značí, že jakýkoliv jev vyskytující se v tomto stavu skončil. V případě byznysu se jedná o kritérium, které vyjadřuje konec jednání.

#### 2.2.1.4 Úvodní studie

Úvodní studie, známá též pod pojmem „*feasibility study*“<sup>24</sup>, lze taktéž charakterizovat jako studii proveditelnosti. Cílem tohoto dokumentu je zmapovat a stanovit celkovou koncepci řešení, včetně toho, jak zapadá do celkové architektury ICT. Jeho obsahem je přesné personální obsazení, resp. minimálně přesně stanovené nároky na personální zdroje, stejně jako technologické požadavky a finanční náklady. Rovněž vytvoření organizační struktury projektu včetně stanovení odpovědností jednotlivých členů patří do této studie. Z pohledu uzavření finální smlouvy o dodání DW, v případě řešení jeho implementace pomocí dodavatelského řešení, se ukázalo posečkat až na výslednou podobu tohoto dokumentu tak, aby byla možná jeho oponentura ze strany nejvyššího vedení, případně jiných zainteresovaných stran. V rámci zpracovávání úvodní studie je rovněž velmi důležitá vzájemná kooperace jak dodavatele, tak všech vnitropodnikových zdrojů, zákazníků i IT techniků tak, aby jeho výsledná podoba co nejvíce odpovídala skutečné realitě.

#### 2.2.2 Analýza a návrh DW

Tato fáze životního cyklu se svojí náplní částečně podobá fázi plánování, ovšem v rámci analýzy a návrhu je definován daleko přesnější detail jednotlivých datových struktur, způsobu jejich získávání, transformací, definice případných „*interface*“<sup>25</sup> do jiných datových struktur a ostatní vlastnosti a funkce, které má zaváděný DW obsahovat. Samotná analýza a aplikace mohou nabývat různého rozsahu, dle příslušných metod vedení konkrétního projektu. V rámci této práce je rozsah uzpůsoben potřebám pro stavbu DW.

##### 2.2.2.1 Analýza podnikových procesů

Pro úspěšnou stavbu DW je zapotřebí velmi důkladně zmapovat veškeré procesy, které využívají kterýkoliv z primárních systémů, jejichž data mají být součástí tohoto

---

24 „*Studie proveditelnosti by měla být v zásadě co nejkompaktnějším a koherentním popisem projektu*“, Citace: Sieber Uchytíl, s.r.o., dostupné online z <http://sieber-uchytil.cz/studie-proveditelnosti-feasibility.html>, aktualizováno 23.7.2011

25 Rozhraní mezi více systémy, více dostupné online z [http://en.wikipedia.org/wiki/Interface\\_\(computing\)](http://en.wikipedia.org/wiki/Interface_(computing)), aktualizováno 23.7.2011

DW. Důvodem je, že tímto způsobem nedojde k opomenutí žádného možného budoucího zákazníka a stejně tak dostaneme odpověď na to, co vlastně za data budou v DW obsažena resp. tato analýza zjednoduší do jisté míry tvorbu logického datového modelu a následně také tvorbu metadat. Stejně tak v případě že bude možné do transformace dat přidat jakoukoliv výpočetní logiku, je na místě zmapovat také všechny procesy, které sice do DW daty nepřispívají, ovšem jsou jeho zákazníkem, a proto se může stát, že jejich následnou úpravou při implementaci DW dojde k jejich zrušení či optimalizaci. Analýza procesů se svým rozsahem může značně lišit, a u stavby DW to platí obzvláště, ale z pohledu zjednodušení-si následné práce je velmi přínosná, tudíž její tvorbě by měl být věnován dostatečný prostor. V této fázi vývoje není na místě jakkoliv procesy měnit, součástí by měl být pouze jejich současný stav a případné návrhy na změny do budoucna.

#### **2.2.2.2 Analýza vstupních datových zdrojů pro DW**

Účelem této analýzy je zmapovat a posoudit stav vstupních datových zdrojů z primárních systémů pro DW. Tato oblast je důležitá zejména pro pozdější správné nastavení časových očekávání na aktualizaci dat v DW a také pro implementaci *ETL - Load* procesů, která budou tento sklad plnit. Časová očekávání na datový sklad, tj. kdy dojde k aktualizaci dat, je vhodné řídit podle poslední doběhnuté aplikace s nejmenším časovým rozlišením (např. pokud je nejmenší rolišovací jednotkou entity v DW jeden den a máme dva primární systémy, z nichž jeden je uzavřen o hodinu dříve, je vhodné posečkat s aktualizací výsledné vrstvy na uzavírku toho druhého), zamezí se tím následná nekonzistence výsledků různých SQL dotazů spuštěných v mezidobí. Obsahem analýzy vstupních dat je taktéž oblast kvality dat a nároků na *ETL - Transfer*.

#### **2.2.2.3 Analýza stávajících DM/DW a BI aplikací**

Pokud již v podniku existují DM a cílem implementace DW je dle Kimballova přístupu jejich propojení do společného DW, anebo je cílem stavby DW rozšíření či nahrazení nějakého stávajícího řešení, je namíste provést důkladnou analýzu těchto

datových zdrojů, která pomáhá ve stejné oblasti jako bylo oddílů v oblasti výše. V případě, že mají být tyto DM plněny z implementovaného DW, je potřeba také analyzovat, co za data, v jaké granularitě a jaké kvalitě jsou požadována, a zda je výhodnější je transformovat až na úrovni zákazníka tj. DM, anebo by bylo vhodnější tato data upravit již na úrovni DW a předávat pouze zpracovaný obraz. V případě, že v podniku je implementována BI vrstva a DW bude její součástí, je na místě také analyzovat technické a funkční požadavky této vrstvy na bezproblémovou integraci DW.

#### 2.2.2.4 Návrh DW

Obsahem návrhu DW by měl být jak logický, tak fyzický datový model a veškerá specifikace, která bude závazná pro další fázi a to samotnou implementaci. V případě logického modelu by tento měl být po odsouhlasení oběma stranami definitivní a nemělo by v něm docházet k žádným změnám, jenom tak bude zaručena neměnnost podmínek vyplývajících z úvodní studie. Fyzický model je obvyklé doupravovat během samotné implementace dle aktuálních problémů (např. datové typy jednotlivých sloupců). Součástí návrhu jsou následující oblasti:

- Seznam primárních systému
- Způsob přenosů a definice vstupních dat - *Extract*
- Definice úprav nad nahrávanými daty – *Transform*
- Specifikace způsobu nahrávání dat – *Load*
- Specifikace odpovědností za jednotlivé provozní procesy včetně servisní smlouvy
- Návrh jednotlivých datových entit, včetně jejich specifikace ve formě tabulek a jejich popisů
- Metadata k jednotlivých entitám
- Specifikace vazeb na ostatní datové zdroje
- Specifikace rolí a přístupových práv
- Security omezení
- Seznam předdefinovaných views
- Hardware nároky pro datový sklad

- Technická specifikace údržby

Fáze návrhu je klíčová pro budoucí vyjednávání s dodavatelem, jak interním tak externím, na téma „*in scope – out of scope*“<sup>26</sup> v případě, že finální řešení nebude zcela odpovídat požadavkům *stakeholderů*. Zejména u této fáze je velmi důležitá vzájemná spolupráce jak podniku s dodavatelem, tak vnitropodnikových složek ve smyslu zákazníků řešení a IT odborníků tak, aby tento dokument neobsahoval žádné nejasnosti a jeho specifikace plně odpovídala požadovanému stavu. Zejména oblast extraktů je velmi důležitou, protože jakékoliv opomění na straně zdrojových dat může ohrozit a značně prodražit implementaci DW. V rámci budoucího použití je také velmi důležité správně ošetřit oblast nahrávání dat. Zda budou nahrávána z primárního systému přímo, nebo skrze třetí stranu, např. „*DataHub*“<sup>27</sup>, kdy druhé řešení je sice technologicky čistější, protože prostředí pro samotné načítání dat je již optimalizováno pro tento účel a není tedy nutné customizovat ETL procesy např. pro nahrávání textových extraktů, ovšem jeho zapojením vzniká další článek v tomto procesu, na kterém se mohou vyskytnout chyby. Taktéž nastavení „*Service Level Agreement*“<sup>28</sup>, dále jen SLA, podmínek je z pohledu budoucího bezproblémového chodu DW důležité, obzvláště v případě, že provoz tohoto DW bude outsourcován do jiné země, kdy je potřeba počítat jak s časovým posunem pracovní doby, tak možností vzniku jazykových bariér. Problém outsourcingu a jeho servisu bude více popsán v praktické části, ovšem u klíčových systémů se v poslední době ukazuje toto řešení jako ne zrovna ideální, zejména z důvodu absence možnosti přímého kontaktu servisních techniků s lidmi z byznysu.

---

26 Jedná se o určení rozsahu konkrétní aktivity (co je její součástí a co už ne), více dostupné z [http://en.wikipedia.org/wiki/Scope\\_\(project\\_management\)](http://en.wikipedia.org/wiki/Scope_(project_management)), aktualizováno 23.7.2011

27 Jedná se o vrstvu jejímž cílem je shromažďovat veškeré výměny dat mezi systémy a která je optimalizována pro příjem, zpracování a následné poskytnutí přichozích extraktů v různých formách a z různých systémů, více dostupné online z [http://en.wikipedia.org/wiki/Master\\_data\\_management](http://en.wikipedia.org/wiki/Master_data_management), aktualizováno 23.7.2011

28 Dohoda o úrovni poskytovaných služeb mezi zákazníkem a dodavatelem, více dostupné online z <http://www.systemonline.cz/outsourcing-ict/tajemstvi-zkratky-sla-1.htm>, aktualizováno 23.7.2011

### **2.2.3 Implementace DW**

Implementace DW by se dala zjednodušeně charakterizovat jako převod návrhu DW do technické podoby a přípravu na uvedení do provozu. Nejobsáhlejší a časově nejnáročnější v této fázi je zejména převod logického modelu do jeho fyzické podoby, včetně tvorby metadat o jednotlivých sloupcích, a nastavení transformace vstupních dat včetně dalších customizací.

#### **2.2.3.1 Detailní specifikace DW**

Tato část vychází z návrhu DW a v ideálním případě by měla být jeho kopií, alespoň v technických oblastech. V rámci implementace jsou dopracovány detaily, které se vyskytnou při samotné tvorbě a obsahuje rovněž seznam všech customizací a požadovaných úprav nad rámec standardního resp. původně sjednaného řešení. Součástí detailní specifikace jsou rovněž logický a fyzický datový model, definice interfacových tabulek, definice extraktů a způsob jejich načítání, časová platnost jednotlivých údajů.

#### **2.2.3.2 Vývoj struktur DW**

Jedná se o samotný převod fyzického modelu a veškerých ETL procesů do vybrané technologie. Tento vývoj kopíruje detailní specifikaci a případné problémy nebo odchylky od definice je potřeba zaznamenat do projektové dokumentace, a v případě jejich akceptace ze strany zákazníka je potřeba je také zapracovat do detailní specifikace. Tento proces je časově velmi náročný a pro zákazníka se odehrává poněkud mimo jeho dohled, ovšem je velmi důležité pamatovat na ochotu kdykoliv prodiskutovat případné obtíže, protože ve fázi vývoje lze obvykle narazit na neshody mezi detailní specifikací a možností dané technologie.

#### **2.2.3.3 Prototypy – zkušební ETL proces vstupních dat pro DW**

Jedná se o prakticky nejspolehlivější test vlastností implementovaného DW. V tomto procesu jsou zkusmo natažena ideálně všechna zdrojová data, avšak lze zpracovat i jenom jejich určitý výsek, což je v praxi častější a na tomto loadu je provedena oponentura jak ze strany stakeholders, tak zejména ze strany cílových zákazníků. Jedná se prakticky o UAT testování a proto je velmi důležité, aby tato fáze

nebyla ze strany cílových uživatelů podceněna, a je vhodné získat pro toto testování podporu managementu.

Následně po oponentuře je provedeno zapracování případných připomínek a zhotovení dokumentu, který bude obsahovat jak její výsledek, tak navržená řešení a to včetně nároků na zdroje, finanční, personální i časové. Všechny tyto úpravy, pokud budou schváleny, je potřeba následně zapracovat do celkové projektové dokumentace.

#### **2.2.3.4 Dovývoj a customizace**

Svojí náplní se jedná o shodný proces jako v případně vývoje. Součástí je také důkladné protestování implementovaných změn a taktéž jejich oponentura před zákazníky.

#### **2.2.3.5 Akceptační řízení**

Správné sestavení skupiny, která bude rozhodovat o akceptování či zamítnutí implementovaného DW, je velmi důležité pro následné uvedení aplikace do provozu a jeho používání. V tomto případě již není potřeba oponentura před koncovými uživateli, ale je potřeba, aby se na představeném řešení shodli všichni zástupci všech oddělení podniku (jak technických, tak byznysových). Akceptační řízení v sobě zahrnuje taktéž důkladné testování všech procesů, kdy velmi záleží na správném nastavení scénářů v rámci akceptačních testů, kdy tyto by měly vycházet z projektové dokumentace. Taktéž data nahraná v DW při akceptačním řízení by měla odpovídat reálnému dění v podniku tak, aby bylo možné odchytil případné byznysové nedostatky, zejména ve fázi transformace dat, případně jejich konzistence.

### 3 Praktická část práce

Cílem praktické části této práce je popis reálné implementace datového skladu pomocí příkladů, jak probíhala ve společnosti GE Money Bank, a.s. na divizi Financí a to zejména ve fázích předcházejících samotné implementaci. Jednotlivé fáze jsou poskládány tak, jak skutečně následovaly při plnění projektu a některé z nich, ač byly součástí metodologické části, jsou záměrně vynechány, protože k jejich naplnění nedošlo.

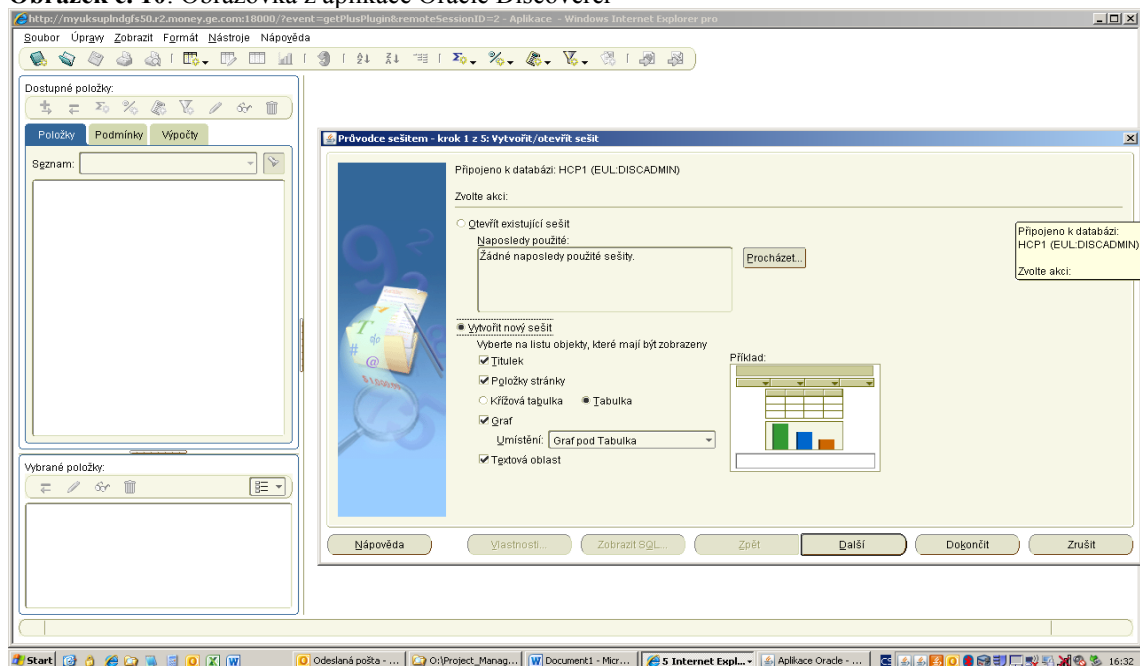
K rozhodnutí o stavbě DW vedla poměrně dlouhá cesta, zejména z důvodů neznalosti samotné aplikace Oracle Financials a jejích možností. V původní dokumentaci k tomuto účetnímu systému byl totiž popsán jako součást dodávky také modul Oracle Discoverer, který umožňuje základní funkčnosti pro reporting dat, ovšem ve velmi omezené míře a je nutno taktéž udržovat pod ním tzv. „*end-user-layers*“<sup>29</sup> dále jen EUL, což je vrstva definujících, která data může uživatel vytáhnout a jaké jsou mezi nimi vazby tak, aby mohl uživatel vytáhnout report bez znalostí jakéhokoliv přístupu k datům. Ovšem tento modul se ukázal jako zcela nadostatečný pro náročné reporty, což bylo způsobeno jak náročností na datové přenosy online mezi Velkou Británií, kde jsou provozovány aplikační servery OF, a Českou republikou, kde jsou uživatelé, tak tím, že EOL tvorba je pro změnu outsourcována do Číny, a tím pádem její jakékoliv rozšiřování či změna je vždy náročnější na čas a komunikaci, kdy ne vždy je zcela jasná interpretace mezi zadáním a doručeným dílem.

---

<sup>29</sup> Vrstva mezi klasickou databází a uživatelským rozhraním, více dostupné online z [http://download.oracle.com/docs/html/A86730\\_01/managing.htm](http://download.oracle.com/docs/html/A86730_01/managing.htm), aktualizováno 23.7.2011

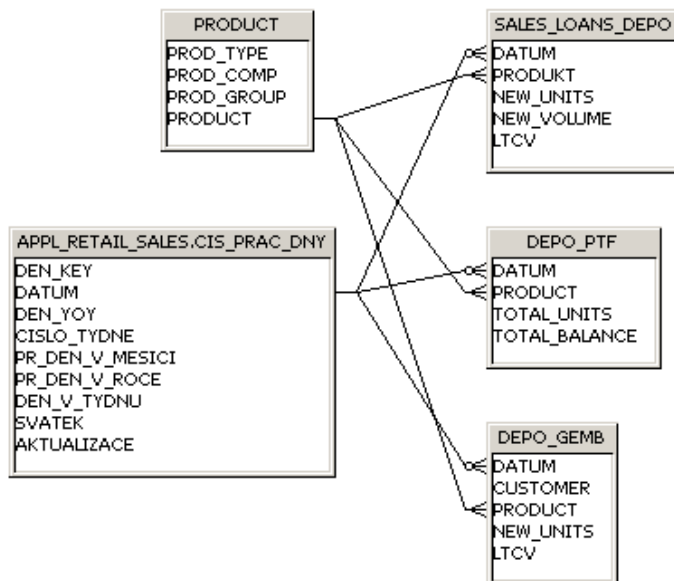


**Obrázek č. 10:** Obrazovka z aplikace Oracle Discoverer



**Zdroj:** vlastní tvorba, instance pro GE Money Bank, a.s.

**Obrázek č. 11:** Ilustrační příklad vrstvy EOL (v tomto případě použito universum pro BO)



**Zdroj:** vlastní tvorba, instance pro GE Money Bank, a.s.

Z výše uvedeného vyplynula potřeba stavby původně jednoho DM nad oblastí dat týkajících se hlavní knihy, jejíž reporty trvaly nejdéle a v mnoha případech ani nedoběhly, ovšem další analýza požadavků již jednoznačně ukázala, že toto řešení rozhodně nebude konečné a tím pádem začala vznikat myšlenka zřídit celý DW, do

kterého budou postupně integrována všechna data všech procesů, které jakýmkoliv způsobem zajímají finance.

Cílem tohoto řešení rozhodně nebyla ambice nahradit MIS, resp. ADS, které jsou hlavním DW pro potřeby GE, ale vytvořit samostaně stojící datový zdroj, který bude posléze na tyto napojen, případně jim bude posílat extrakty dat.

Při vstupních analýzách se také ukázalo jako značně ekonomicky neefektivní zapojit do tvorby tohoto DW třetí stranu v pozici dodavatele, a proto bylo rozhodnuto o tvorbě vlastními silami. Detailněji jsou důvody tohoto rozhodnutí popsány v samostatné kapitole níže.

### **3.1 Vstupní analýza pro DW**

Z pohledu celkového plánování rozvoje ICT/IS jednoznačně vyplynulo, že budovaný DW by měl být součástí ADS, ovšem tento záměr byl rovnou v rámci této analýzy označen za téměř nemožný z důvodu časové a finanční náročnosti a tudíž jeho implementace jako samostatného datového zdroje byla jako druhá varianta zvolena za jedinou dále diskutovanou. Výskyt tohoto typu problému lze označit za poměrně častý, kdy nezdědka dochází k ukončení projektu již v této fázi pro nedostatek kapacit. Z pohledu jiných implementací DW toto rozhodnutí nelze v této fázi v žádném případě doporučit a skutečně je vhodné jej nechat až po naplánování samotného projektu a zvážení všech kladů a záporů, které jednotlivé varianty přináší. Stejně tak se v průběhu této analýzy odhalila poměrně závažná skutečnost, a to, že při plánování velkých projektů není přizván nikdo z datové infrastruktury a tudíž existuje více projektů, které budou pravděpodobně potřebovat vlastní DM/DW, ale jejich implementace rovněž není počítána v rámci zdrojů na ADS.

Vazba na ostatní aplikace byla řešena zejména v oblasti možných extraktů do stávajících datových skladů a na vrstvu BI, reprezentovanou zejména intranetovou aplikací pro zobrazování pevně definovaných reportů s možností filtrace a BO nádstavbou, která umožňuje jistou flexibilitu v možnosti definice potřebných dat širšímu spektru uživatelů.

Získání očekávání cílových zákazníků proběhlo formou interviews se zástupci jednotlivých oddělení, kde jim vůbec nebyla kladena žádná omezení, ale vyloženě si mohli nerušeně specifikovat, co by potřebovali. Velkou výhodou bylo správné sestavení referenční skupiny zákazníků, kdy se podařilo dostat jak vedoucí procesů, tak samotné účetní, které se potýkají s denní rutinou a byly schopny velmi detailně popsat zejména požadovanou granularitu dat a taktéž analytiky z ostatních oddělení financí, zejména FP&A (*Finance Planning & Analysing*), kteří naopak vznesli požadavky na atributy, dle kterých potřebují data dále agregovat. Manažeři jednotlivých týmů byli pozváni v první části samostatně, kdy tímto bylo dosaženo „*high-level*“ požadavků, zato ale v celkovém kontextu a díky tomu byly určeny vzájemné vazby dat.

Výsledkem prvního interview byl draft dokumentace, který byl následně předán k oponentuře stejné skupině a v rámci něž došlo k upřesnění nejasností a oprav chybně interpretovaných požadavků. V rámci této oponentury je důležité předložit oponentům dokumentaci tak, aby pro ně byla pochopitelná a její provedení mělo smysl, což zejména u řadových pozic není zrovna jednoduchým úkolem a osobou zodpovědnou za vedení a prezentaci této oponentury by měl být člověk schopný jednat se širokou škálou různých typů lidí. V případě GE se velmi osvědčilo nechat tuto oponenturu provést člověka, který do této doby neměl s tímto projektem nic společného, a tudíž nebyl ovlivněn průběhem předchozích interview.

Na závěr vstupní analýzy bylo potřeba přesvědčit CFO (*Chief Financial Officer*) o skutečné potřebě implementace DW a toho bylo dosaženo pomocí představení pozitivních efektů, zejména v oblasti řízení divize, kdy stěžejní prezentované oblasti zlepšení byly následující:

- Snadné sledování výkonnosti účtárny na bázi aktuální datum - den
- Možnost kontroly budoucích nákladů z vytvořených objednávek
- Ušetření „*head count*“, lidských zdrojů, na analytickou podporu
- Menší počet potřebných licencí pro Oracle Discoverer

## 3.2 Plánování projektu DW

Tato fáze byla v rámci zavádění DW nad OF poněkud zkrácena z důvodů popasných výše, taktéž již nebylo potřeba získávat zainteresované strany, neboť jich bylo poměrně málo a hlavní stakeholder byl jednoznačně pro implementaci již od samotné vstupní analýzy. V rámci jiných DW je ovšem důležitá pro celkové shrnutí všech aspektů ovlivňujících tuto implementaci a taktéž až v této fázi dochází k rozhodnutí, zda bude projekt vůbec řešen a pokud ano, zda si je podnik schopen poradit vlastními silami, nebo je potřeba rozhodnout o dodavatelském řešení prostřednictvím některé z firem, které se touto problematikou přímo zabývají. Pro většinu manažerů jsou bohužel klíčové náklady, a to zejména finanční a tím pádem mnohdy dochází k situacím, že je vybráno řešení pomocí vlastních sil, aniž by vůbec byly vzaty do úvahy zdroje personální, a to zejména jejich kvalitativní stránka (to že někdo umí spustit SQL skript, ještě nutně neznamená, že je schopen také postavit DW).

V případě GE bylo rozhodování taktéž velmi ovlivněno finančními možnostmi, zejména proto, že plánování projektu a úvodní analýzy se konaly v průběhu roku 2009, kdy bylo relativně složité makroekonomické prostředí z důvodů finanční krize a nejvyšší vedení podniku se snažilo ušetřit, takže každá investovaná koruna byla velmi těžko prosaditelná. Z tohoto důvodu byl od začátku značný tlak na implementaci vlastními silami a samotné výběrové řízení na dodavatele bylo bráno spíše jako srovnávací analýza potřebných zdrojů, zejména časových, mezi odhady lokálního týmu a oslovené firmy.

## 3.3 Výběr dodavatele DW

V praxi je velmi těžké doporučit ideální počet oslovených firem o tzv. „*Request for Informations*“ dále jen RFI, což je dokument, který je zaslán různým dodavatelům a obsahuje velmi zjednodušený projektový záměr, ve kterém jsou obsaženy jenom skutečně klíčové funkcionality řešení, jehož cílem je zmapovat vůbec situaci na trhu a předpřipravít si a vytrídít vhodné kandidáty na další jednání. Tento vzorek dodavatelů je samozřejmě ideálně co největší, ovšem v rámci České republiky je potřeba být velmi obezřetný, neboť trh je zde poměrně malý a velmi často se stává, že se mezi sebou oslovené firmy dohodnou a některé buď rovnou odstoupí a požadovaný dokument ani

nepošlou, anebo se dodané informace velmi podobají jak svým rozsahem, tak odhadovanou cenou. V případě velkého množství oslovených dodavatelů je potřeba také počítat se zvýšenou časovou náročností na zpracování informačních dokumentů a tím pádem na prodloužení celkové doby trvání celého projektu. V případě implementace v GE tento proces vůbec nenastal a vybranému dodavateli byl rovnou poslán „Request for Proposal“ dále jen RFP.

**Tabulka č. 1: Příklad detailu RFP – otázky k HRIS**

| Requirements   |   | Vendor A                    |                                       |   |         |
|--|---|-----------------------------|---------------------------------------|---|---------|
| Requirements   | Importance  | meets the requirement 100%? | Meets %, development in offered price | Meets %, development out of offered price | Comment |
| <b>D Requirements for HR processes</b>                                       |   |                             |                                       |   |         |
| <b>D 0 General</b>   |   |                             |                                       |   |         |
| D 0 1  | Maximum coverage of existing HRbusiness processes for payroll   | A                           |                                       |   |         |
| <b>D 6 Compensation, wages</b>   |   |                             |                                       |   |         |
| <b>D 6 1 Planning and administration of basic wages, variable components</b> |   |                             |                                       |   |         |
| D 6 1 1  | Part of variable components see incentives.   | A                           |                                       |   |         |
| D 6 1 2  | Other part is VC bonuses - field for calculation of target (10 or 15 percent of annual wages) – take into account from when to when the employee was on VCT MBO and from when to when he/she was on 10% or 15% etc., the same as targets and payouts for incentives   | A                           |                                       |   |         |
| D 6 1 3  | Basic wages – it should be possible to export/import Oracle HRdata from the system  | A                           |                                       |   |         |
| D 6 1 4  | Ability to calculate anticipated and actual expenses for wages "total cash", i.e. the sum of basic wages for the year (it is anticipated that current wages will remain unchanged to the end of the year, but previous increases should be taken into account) + the sum of targets for variable components (incentive or bonus), actual TC – the sum of basic wages for the year (same) + the sum of variable components paid out (incentive or bonus) | A                           |                                       |   |         |
| D 6 1 5  | Ability to upload all EMU, extraordinary bonuses and rewards from the F&Rsystem, etc – include in Total cash – Yes/No option (i.e. sometimes we need TC with and sometimes without this), referral program  | A                           |                                       |   |         |

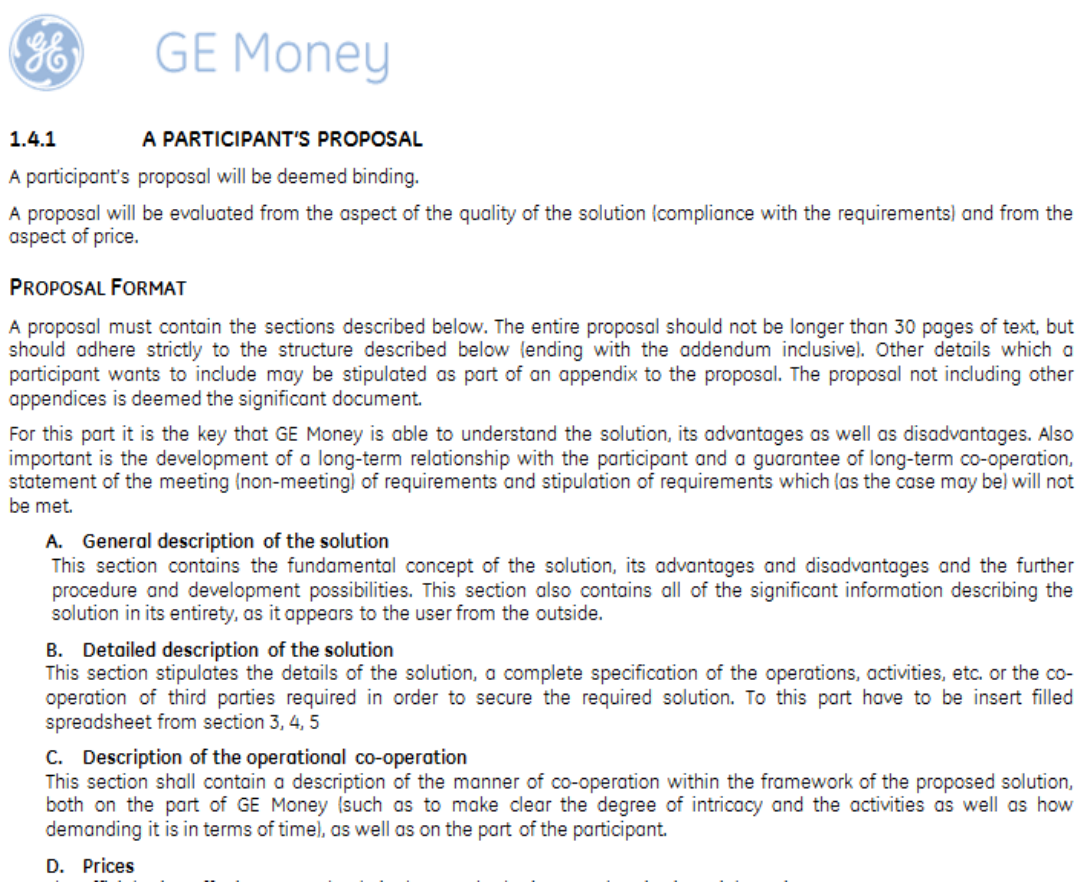
**Zdroj:** GE Money Bank, a.s., součást RFP pro implementaci HRIS

Příprava poptávkového dokumentu je klíčovou fází pro další pokračování ve výběru dodavatele a kvalitně zpracovaný RFP dokáže velmi zefektivnit a zejména zrychlit celý tento proces. V praxi se lze často setkat s tím, že RFP je prakticky čistou kopií projektového plánu, ovšem toto řešení opět nelze označit za šťastné a minimálně po technické stránce je velmi vhodné detailněji specifikovat požadovaná kritéria na poptávaný systém. Také se velmi osvědčilo být otevřený pro případné otázky osloveného dodavatele a ideálně s ním navázat otevřenou komunikaci, čímž lze získat poměrně jasnou představu o tom, jaká je jeho firemní kultura v oblasti vztahů a komunikace se zákazníkem, včetně nejdůležitější vlastnosti a to způsobu řešení případných konfliktů a přání zákazníka, což je klíčové v pozdějších fázích vývoje projektu.

V rámci korektních vztahů a vyhnutí se případným následným diskuzím, v horším případě soudním sporům, po ukončení výběrového řízení, je dobré jako

součástí poptávkového dokumentu jasně a konkrétně vymežit podmínky, za jakých bude nabídka akceptována a jakým způsobem bude o vítězi rozhodnuto, včetně způsobu předání nabídkového dokumentu a následných prezentací. Pro potřeby implementace DW v GE bylo rovnou v rámci RFP uvedeno, že není požadována žádná obecná prezentace jimi nabízeného DW, ale rovnou zajištění referenční ukázky DW nad libovolným účetním systémem.

**Obrázek č. 12:** Ilustrační příklad formátu RFP v oblasti podmínek formy nabídky



**Zdroj:** GE Money Bank, a.s., součást RFP pro implementaci HRIS

Hodnocení nabídek a finální výběr dodavatele DW po prezentacích, případně referenčních ukázkách, je na první pohled relativně jednoduchý proces, ovšem toto zdání je velmi klamné a je dobré mít na paměti následující zásady:

- Vyvarovat se „halo“, případně „horn“ efektu, kdy hodnocená nabídka je výrazně lepší či horší pouze v některé ze všech hodnocených kritérií
- Vyvarovat se předsudků, jak bylo popsáno v teoretické části, to, že dodavatel disponuje výborným prezentátorem rozhodně nemusí znamenat, že stejných kvalit dosahuje také celkové jím nabízené řešení.
- Stanovit si definitivní a shodná „k.o.“ kritéria
- Správné rozdělení vah u jednotlivých hodnotících oblastí, pokud možno nedávat ceně hlavní váhu

Tabulka č. 2: Příklad hodnotícího formuláře

| Evaluation table for Payroll - GE Money Bank, a.s. - RFP |             |             |             |             |            |
|--|-------------|-------------|-------------|-------------|------------|
| Supplier's name  | Weight      | XXX         | YYY         | ZZZ         | BBB        |
| Price + Conditions                                       | 25%         | 0,6         | 0,73        | 1,2         | 0,48       |
| IT REQUIREMENTS  | 10%         | 0,46        | 0,47        | 0,54        | 0,54       |
| IT Security  | 10%         | 0,39        | 0,39        | 0,5         | 0,41       |
| GENERAL BUSINESS REQUIREMENTS                            | 20%         | 0,88        | 1,08        | 1,09        | 1,06       |
| REQUIREMENTS FOR HR PROCESSES                            | 20%         | 0,73        | 1,07        | 0,99        | 0,96       |
| GENERAL VIEW OF THE RESPONSE                             | 10%         | 0,3         | 0,5         | 0,47        | 0,57       |
| REFERENCE  | 5%          | 0,14        | 0,23        | 0,25        | 0,28       |
| <b>Total</b>   | <b>100%</b> | <b>3,49</b> | <b>4,46</b> | <b>5,04</b> | <b>4,3</b> |
| <b>Rank</b>  |             | <b>4</b>    | <b>2</b>    | <b>1</b>    | <b>3</b>   |

Zdroj: Vlastní tvorba, GE Money Bank, a.s., implementace HRIS

Z tabulky č. 2 je patrné, že i přes znalost těchto zásad u výběrových řízení, zejména ve velkých korporacích, není dodržení zásady o ceně a její váze v rámci hodnotících kritérií vždy realizovatelné. V těchto případech se nezdá setkáme s případem, že některá z firem projde přes „k.o.“ kritéria, v ostatních oblastech je hodně za ostatními, ale vzhledem k cenové nabídce, která nezdá bývá značně podhodnocena, např. z důvodů, že dodavatel nemá dosud zkušenosti s takto rozsáhlou implementací, nebo se snaží proniknout na trh velkých korporací a na konkrétním projektu získat reference, je jeho řešení vítězné, což ale značně komplikuje následné fáze. Rozhodně je velmi vhodné se tomuto přístupu vyvarovat. V případě implementace DW nad OF byla tato situace značně podobná a dodavatel se srovnával cenovou nabídkou s interně vyčíslenými náklady.

### 3.4 Studie proveditelnosti

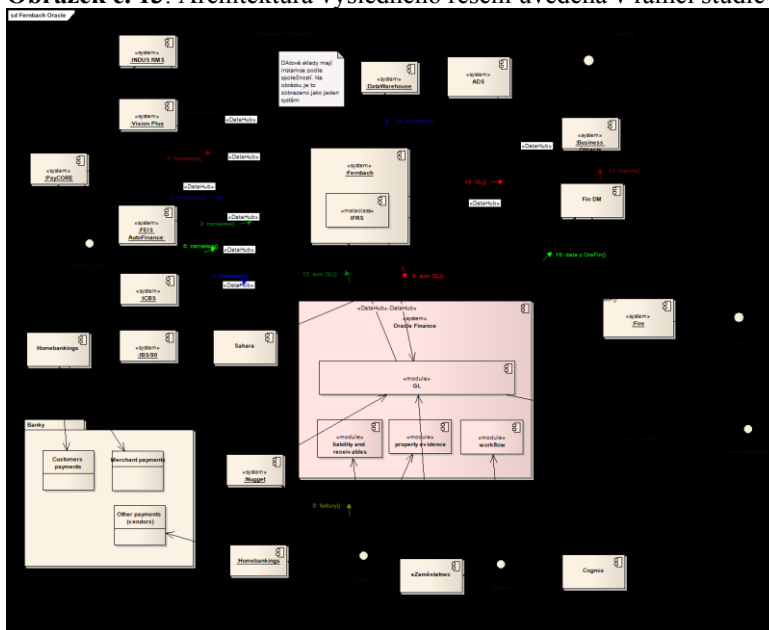
Tento dokument by měl být postaven z větší části na implementaci nabídky od finálního řešitele do struktur podniku a to nejenom v rámci architektury ICT/IT, ale také do celkové procesní mapy, pokud že systém nějaké procesy ovlivní. V případě implementace DW nad OF byla tato fáze značně zjednodušena, zejména kvůli způsobu provedení stavby DW vlastními silami a tudíž se předpokládalo, že projekt jako takový proveditelný bude, protože všechny záležitosti kolem byl podnik schopen přímo ovlivňovat. Opět je nutno konstatovat, že toto řešení situace není vyhovující a zejména v dalších fázích nastaly komplikace, jako např. časový posun mezi Čínou a Českou republikou a tím pádem hůře dostupný support, které mohly být odhaleny již v rámci této studie. Pro potřeby GE byly do studie proveditelnosti zahrnuty následující skutečnosti:

- Personální obsazení – nutno podotknout, že celá fáze návrhu a implementace byla finálně ve své podstatě záležitostí jednoho člověka, i když na vstupu se počítalo se zdroji daleko většími
- Technologické požadavky – přidělena byla samostatná instance Oraclu
- Časové požadavky – schválen byl postupný vývoj různých DM s prioritizací výstupů z GL
- Architektura výsledného řešení
- Definitivní funkční požadavky a jejich skutečný obraz
- Analýza rizik a jejich případných dopadů

Výsledek studie proveditelnosti byl dle očekávání kladný ve všech směrech s tím, že hlavní rizika, která ohrožují proces implementace, jsou v oblasti personálních kapacit.



**Obrázek č. 13:** Architektura výsledného řešení uvedená v rámci studie proveditelnosti



**Zdroj:** GE Money Bank, a.s., součást Solution Concept OF – část FS

### 3.5 Analýza vstupních datových zdrojů

Jedná se o časově a vědomostně velmi náročný proces, kdy je potřeba velké trpělivosti na straně řešitele této oblasti. V rámci tohoto procesu dochází k definici a následné dokumentaci potřebných extraktů ze zdrojových systémů, což vyžaduje velkou kooperaci mezi provozovateli těchto zdrojů a řešiteli DW. Největším nebezpečím této fáze je podcenění detailní dokumentace požadovaných údajů, kdy při definici extraktů je nutno postupovat velmi pečlivě a každý z požadovaných sloupců je potřeba popsat minimálně pomocí:

- Názvu sloupce
- Metadat
- Datového typu
- Zdroje, způsobu jak jej lze získat ze zdrojového systému

Z výše uvedeného jednoznačně vyplývá, že ve většině případů nelze tuto analýzu sestavit jednostranně na straně dodavatele.

**Tabulka č. 3: Příklad definice extraktů pro GL**

| #  | Column                        | Description  | Data type    | Domain | Mandatory | Unique Key | Source  | Notice |
|----|-------------------------------|--|--------------|--------|-----------|------------|---|--------|
| 1  | PER_CODE                      | Period code  | VARCHAR2(35) | CODE   | Yes       | Yes        | per.PERIOD_NAME   |        |
| 2  | PER_GROUP_CODE                | Period group code (ADJ, MINGT, STAT)                               | VARCHAR2(35) | CODE   | Yes       |            | case when per. ENTERED_PERIOD_NAME like '%-STAT' then 'STAT' when per. ENTERED_PERIOD_NAME like '%-MINGT' then 'MINGT' else 'ADJ' end                             |        |
| 3  | PER_MONTH_CODE                | Period month code  | VARCHAR2(35) | CODE   | Yes       |            | substr(per.PERIOD_NAME, 1,3)  |        |
| 4  | PER_MONTH_ID                  | Period month id  | number       | NUMBER | Yes       |            | decode(substr(per.PERIOD_NAME, 1,3) 'JAN', 1, 'FEB', 2, 'MAR', 3, 'APR', 4, 'MAY', 5, 'JUN', 6, 'JUL', 7, 'AUG', 8, 'SEP', 9, 'OCT', 10, 'NOV', 11, 'DEC', 12, 0) |        |
| 5  | PER_SET_CODE                  | Accounting calendar Name   | VARCHAR2(35) | CODE   | Yes       |            | per.PERIOD_SET_NAME   |        |
| 6  | PER_TYPE_CODE                 | Accounting period type code  | VARCHAR2(35) | CODE   | Yes       |            | USER_PERIOD_TYPE  |        |
| 7  | PER_NAME                      | accounting period name   | VARCHAR2(60) | NAME   | Yes       |            | per. ENTERED_PERIOD_NAME  |        |
| 8  | PER_SET_NAME                  | Accounting calendar description                                    | VARCHAR2(60) | NAME   | Yes       |            | pers.DESCRPTION   |        |
| 9  | PER_YEAR_TYPE_CODE            | Year type (Calendar or Fiscal)                                     | VARCHAR2(35) | CODE   | Yes       |            | decode(YEAR_TYPE_IN_NAME 'C', 'CALENDAR', 'F', 'FISCAL')  |        |
| 10 | PER_TYPE_NAME                 | Accounting period type name  | VARCHAR2(60) | CODE   | Yes       |            | per.description   |        |
| 11 | PER_PERIODS_FISCAL_YEAR_COUNT | Number of periods per fiscal year                                  | NUMBER       | COUNT  | Yes       |            | per.NUMBER_PER_FISCAL_YEAR  |        |
| 12 | PER_START_DATE                | Date on which accounting period begins                             | DATE         | DATE   | Yes       |            | per.START_DATE  |        |
| 13 | PER_END_DATE                  | Date on which accounting period ends                               | DATE         | DATE   | Yes       |            | per.END_DATE  |        |
| 14 | PER_YEAR_START_DATE           | Date on which the year containing this accounting period starts    | DATE         | DATE   | Yes       |            | per.YEAR_START_DATE   |        |
| 15 | PER_QUARTER_START_DATE        | Date on which the quarter containing this accounting period starts | DATE         | DATE   | Yes       |            | per.QUARTER_START_DATE  |        |
| 16 | PER_YEAR_NUMBER               | Accounting period year   | NUMBER       | NUMBER | Yes       |            | per.PERIOD_YEAR   |        |
| 17 | PER_PERIOD_NUMBER             | Accounting period number   | NUMBER       | NUMBER | Yes       |            | per.PERIOD_NUM  |        |

**Zdroj:** Vlastní tvorba, GE Money Bank, a.s., implementace DW

Tento proces se překrývá se samotným návrhem DW a je víceméně jeho nedílnou součástí, v případě implementace v GE byla v rámci analýzy datových zdrojů zpracována vždy první verze extraktové dokumentace a její finalizace byla již součástí návrhu.

### 3.6 Návrh DW

Při projektu, který se potýká s nedostatkem časových a personálních zdrojů, případně jeho konečná podoba je nejistá, lze jednoznačně doporučit pojetí řešení pomocí přístupu W. Kimballa, detailněji popsaného výše, a tudíž začít s rozdělením DW na dílčí DM, které se budou ve finále spojovat do jednoho řešení, tak jak budou postupně přibývat. Stejná situace nastala v případě DW nad OF a jeho rozdělení na menší DM znamenalo značnou časovou úsporu mezi zadáním a prvními výsledky pro uživatele.

#### 3.6.1 Seznam primárních systémů

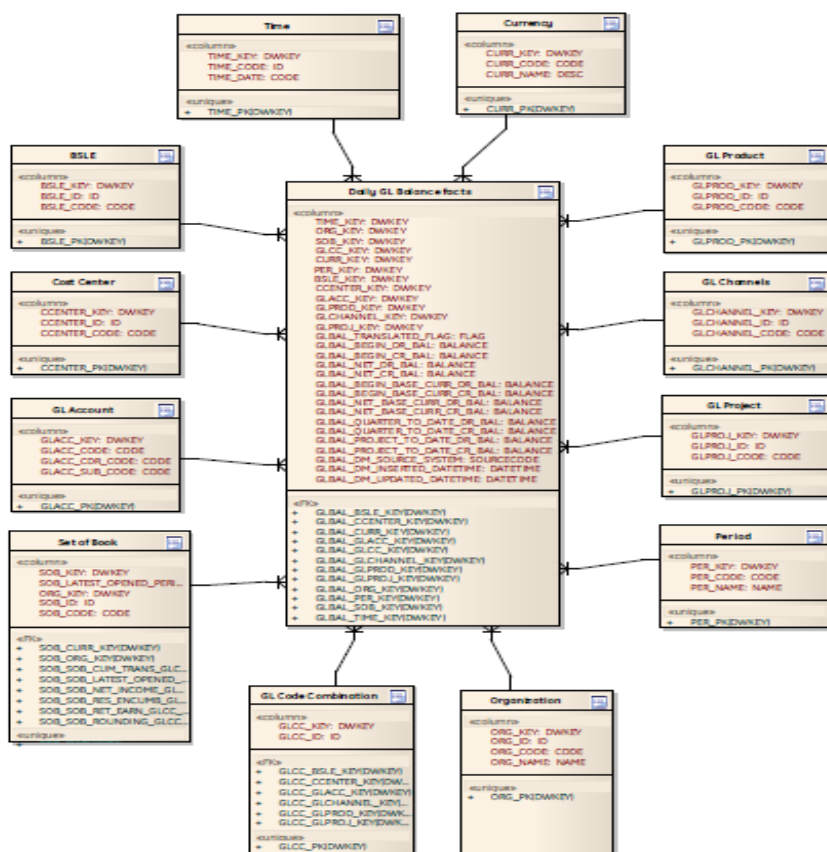
Cílem tohoto seznamu je finalizovat zdroje pro DW, v případě implementace v GE obsahoval seznam primárních systémů, pro finální řešení, následující aplikace:

- Oracle Financial
- Systém pro vytěžování dat z faktur
- Mzdový a personální systém
- Zaměstnanecký portál (drobná vydání, služební cesty...)

### 3.6.2 Logický model

Tvorbu logického datového modelu lze označit za relativně příjemnou a tvůrčí práci, kdy je velmi důležité zejména před jejím převodem do fyzické podoby provést oponenturu s technicky zdatnějšími zákazníky tak, aby bylo případně možno dopravit vzniklé odlišnosti od zadání. V této fázi je velmi vhodné opravdu důkladně nakreslit všechny entity, které má DW obsahovat zejména z důvodů vizualizace jejich vzájemného propojení a z toho plynoucí budoucí složitost reportingu nad těmito daty.

Obrázek č. 14: Logický model pro GL Daily Balances



Zdroj: GE Money Bank, a.s., Datové modely DW pro OF

### 3.6.3 Fyzický datový model

Na rozdíl od předchozí fáze, převod logického modelu do jeho fyzické podoby již tak kreativní není, ovšem jeho provedení je velmi důležité pro pozdější snadnou implementaci DW do provozu. Naprosto ideální je provádět tento převod pomocí některého z nástrojů pro to určených, jako je např. Enterprise Architect nebo Power Designer tak, aby jej bylo kdykoliv možno snadno upravit, a jeho stavba byla do jisté míry také nezávislá na konkrétní technologii. Ovšem v praxi je naprosto běžné převádět logický datový model pomocí tabulkového editoru, jako např. MS Excel. V případě implementace DW nad OF bylo využito právě tohoto nástroje. Tento postup je ovšem velmi náročný na jakékoliv případné změny, ale jeho výhoda je, že se dá spustit skutečně prakticky kdekoli a odpadají náklady za pořízení licence některého z výše popsaných nástrojů.

**Tabulka č. 4:** Příklad fyzického datového modelu

| Sloupec                       | Popis                               | Datový typ    | Doména                        | POV | Součást unikátního klíče | Zdroj                             |
|-------------------------------|-------------------------------------|---------------|-------------------------------|-----|--------------------------|-----------------------------------|
| INT_JOB_ID                    | Job Id                              | NUMBER        | JOBID                         | Yes |                          |                                   |
| INT_CURRENT_DATE              | Current Date                        | DATE          | DATE                          | Yes | X                        |                                   |
| INT_STATUS                    | Status                              | VARCHAR2(20)  | INTSTATUS                     | Yes |                          |                                   |
| INT_ERROR                     | Error                               | VARCHAR2(500) | INTERPROR                     | No  |                          |                                   |
| PER_CODE                      | No                                  | VARCHAR2(35)  | PER_CODE                      | Yes | X                        | per.PERIOD_NAME                   |
| PER_GROUP_CODE                | Period group code (ADJ, MNGT, STAT) | VARCHAR2(35)  | PER_SET_CODE                  | Yes |                          | case when per.ENTERED_PERIOD_NAME |
| PER_MONTH_CODE                | Period month code                   | VARCHAR2(35)  | PER_TYPE_CODE                 | Yes |                          | substr(per.PERIOD_NAME            |
| PER_MONTH_ID                  | Period month id                     | number        | NUMBER                        | Yes |                          | decode(substr(per.PERIOD_NAME     |
| PER_SET_CODE                  | No                                  | VARCHAR2(35)  | PER_SET_CODE                  | Yes |                          | per.PERIOD_SET_NAME               |
| PER_TYPE_CODE                 | No                                  | VARCHAR2(35)  | PER_TYPE_CODE                 | Yes |                          | USER_PERIOD_TYPE                  |
| PER_NAME                      | No                                  | VARCHAR2(60)  | PER_NAME                      | Yes |                          | per.ENTERED_PERIOD_NAME           |
| PER_SET_NAME                  | No                                  | VARCHAR2(60)  | PER_SET_NAME                  | Yes |                          | pers.DESCRPTION                   |
| PER_YEAR_TYPE_CODE            | No                                  | VARCHAR2(35)  | PER_YEAR_TYPE_CODE            | Yes |                          | decode(YEAR_TYPE_CODE             |
| PER_TYPE_NAME                 | No                                  | VARCHAR2(60)  | PER_TYPE_NAME                 | Yes |                          | per.description                   |
| PER_PERIODS_FISCAL_YEAR_COUNT | No                                  | NUMBER        | PER_PERIODS_FISCAL_YEAR_COUNT | Yes |                          | per.NUMBER_PER_FISCAL_YEAR_COUNT  |
| PER_START_DATE                | No                                  | DATE          | PER_START_DATE                | Yes |                          | per.START_DATE                    |
| PER_END_DATE                  | No                                  | DATE          | PER_END_DATE                  | Yes |                          | per.END_DATE                      |
| PER_YEAR_START_DATE           | No                                  | DATE          | PER_YEAR_START_DATE           | Yes |                          | per.YEAR_START_DATE               |
| PER_QUARTER_START_DATE        | No                                  | DATE          | PER_QUARTER_START_DATE        | Yes |                          | per.QUARTER_START_DATE            |
| PER_YEAR_NUMBER               | No                                  | NUMBER        | PER_YEAR_NUMBER               | Yes |                          | per.PERIOD_YEAR                   |
| PER_PERIOD_NUMBER             | No                                  | NUMBER        | PER_PERIOD_NUMBER             | Yes |                          | per.PERIOD_NUM                    |
| PER_QUARTER_NUMBER            | No                                  | NUMBER        | PER_QUARTER_NUMBER            | Yes |                          | per.QUARTER_NUM                   |

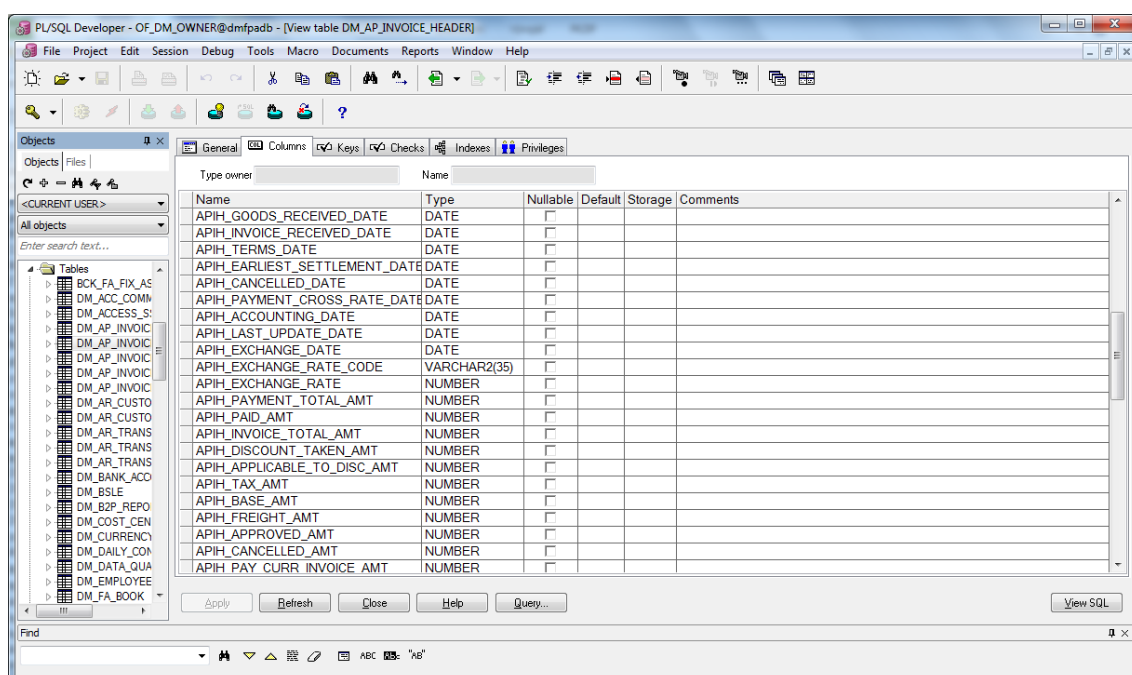
**Zdroj:** Vlastní tvorba, GE Money Bank, a.s., implementace DW

### 3.6.4 Tvorba metadat

Tento proces je velmi často opomíjený a bohužel jeho absence se projeví až v pozdějších fázích používání DW a to zejména při personálních změnách, kdy odejdou lidé, kteří byli u implementace DW a jsou tedy nositeli tacitních znalostí o jednotlivých datových strukturách. V praxi je také vhodné jít do co největšího detailu metadat v tom smyslu, že jsou popsány nejenom jednotlivé entity, ale také sloupce ve finálním fyzickém modelu tak, aby byznysový uživatel mohl pohodlně vyčíst, co za data jsou kde obsažena.

Při implementaci v GE byl tento krok rovněž prakticky vynechán, zejména z časových důvodů a jeho absence se projevuje zejména při zaučování nových analytiků, kteří mají za úkol z něj získávat data.

Obrázek č. 15: Příklad chybějících metadat ve formě komentářů



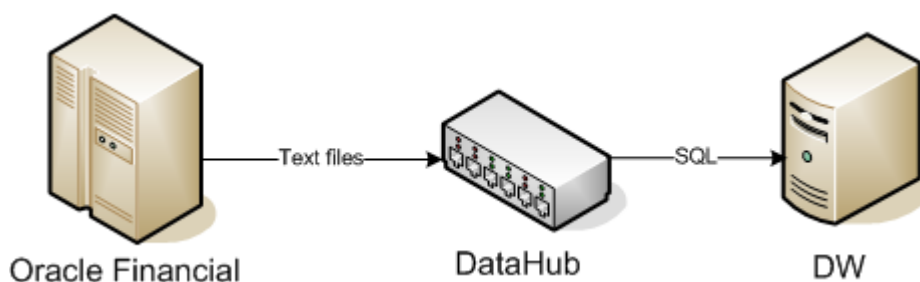
Zdroj: Vlastní tvorba, GE Money Bank, a.s.

### 3.6.5 Způsob přenosů a definice vstupních dat - Extract

Pokud je v podniku implementováno rozhraní pro nahrávání dat, jako např. DataHub, je část *Extract* značně jednodušší a to o oblast specifikace jakým způsobem dostat data například z textového řetězce, případně jak připravit přenos mezi MS SQL a

Oracle databází. Pokud tento nástroj chybí, je nezbytné co nejpřesněji odhadnout nároky na vývoj procesů, které jeho funkčnosti nahradí pro konkrétní primární systémy a jejich schopnosti exportovat data v určitém formátu. V rámci této specifikace je zejména u zpracovávání textových souborů velmi důležité zmapovat, v jakém kódování budou výstupní data. U „mezinárodních“ znaků většinou není problém, avšak kódování do češtiny bývá problematické. V případě citlivých dat je také nezbytné definovat, jakým způsobem bude docházet k jejich šifrování a následně jakým způsobem se bude předávat klíč k jejich dekódování do čitelné podoby. Oblast citlivých dat, např. mzdy zaměstnanců, je nutno více řešit zejména u dodavatelského řešení, pokud že není žádoucí, aby do nich vůbec kdy viděl kdokoliv ze třetí strany. V rámci implementace v GE tyto problémy naštěstí do jisté míry odpadly právě díky přítomnosti DataHubu, který patří mezi velmi univerzální a spolehlivé systémy.

**Obrázek č. 16:** Popis toku dat



**Zdroj:** Vlastní tvorba, GE Money Bank, a.s.

### ***3.6.6 Úpravy nad nahrávanými daty – Transform***

Z pohledu ETL celku se jedná většinou o nejnáročnější proces, protože málokdy je dosaženo stavu, kdy vstupní data není potřeba nijak upravovat, přidávat k nim klíče nebo je rošiřovat o potřebné dimenze, provádět agregace atd. V této fázi je znovu velmi důležitá kooperace mezi tvůrci a zákazníkem, protože dochází k tvorbě výsledné podoby dat a je tudíž nezbytné, aby bylo dosaženo požadovaného efektu.

Také je nezbytné správně stanovit, ve kterém kroku budou data upravována, zda částečná transformace proběhne již na straně zpracovatele extraktů, anebo dojde k jejich nahrání do DW 1:1 a veškeré úpravy budou na něm. Jako vhodnější se v praxi

jeví varianta rozdělit DW na více vrstev, kdy skutečně do té nejnižší, označované často jako vrstva nula dojde k nahrání přesné kopie vstupních dat, v první vrstvě dojde k jejich rozšíření o základní dimenze, vyčištění duplicit, přidání umělých klíčů a případné agregace dle požadované granularity. V druhé a každé další vrstvě pak dochází k úpravám dle konkrétnějších požadavků jejich uživatelů. Tato oblast se do jisté míry překrývá s fází nahrávání dat do DW, kde taktéž dochází k transformacím vstupních dat.

Při implementaci DW nad OF bylo postupováno naprosto shodně, kdy data z DataHubu jsou přebírána 1:1 do vrstvy nazvané jako „*EXT\_OWNER*“ kdy nad ní jsou pomocí PL/SQL provedeny finální transformace tak, aby do vrstvy „*INT\_OWNER*“ přišla již data v takové podobě, aby jejich nahrání do samotných koncových struktur DW proběhlo plným loadem a došlo pouze k vygenerování primárních klíčů, případně doplnění časových údajů o datu nahrání.

**Obrázek č. 17:** Příklad transformace a nahrání dat pomocí PL/SQL

```

49      commit;
50      pck_load_dim.checkDataQuality;
51      commit;
52      PCK_LOAD_DIM.SYNCHRONYZEWFIN;
53      COMMIT;
54      PCK_ASYNC_REPORTS.DROPUNUSEDREPORTTABLES;
55      COMMIT;
56  END RUNSTANDARD;
57
58  procedure GEN_SQL_CREATE_STG
59  is
60      n number;
61  begin
62      SQL_CREATE_STG := 'create table STG_'||rTableDef.ITD_SHORT_NAME||'('|| chr(10)
63      n := 0;
64      FOR i IN rCols.FIRST .. rCols.LAST LOOP
65          if rCols(i).ICD_IS_INSERTABLE_FLAG = 'Y' then
66              if n > 0 then
67                  SQL_CREATE_STG := SQL_CREATE_STG||', ';
68              end if;
69              SQL_CREATE_STG := SQL_CREATE_STG||rCols(i).ICD_COLUMN_NAME||' '||rCols(i)
70              n := n +1;
71          end if;
72      END LOOP;
73      SQL_CREATE_STG := SQL_CREATE_STG||')';
74      add_stmt SQL_CREATE_STG;

```

**Zdroj:** kód VADOVIČ M., GE Money Bank, a.s.

### 3.6.7 Nahrání dat – LOAD

Jak bylo nastíněno v předchozí kapitole, jedná se o proces, ve kterém jsou data nahrána do finálních struktur tak, aby jejich využití koncovým uživatelem bylo co nejjednodušší a nejeftivnější a současně bylo možné stavět OLAP kostky nad

obsaženými daty. Z toho důvodu jsou např. faktové tabulky rozšířeny v procesu transformace o co největší detail dimenzí, samozřejmě tak, aby byla splněna byznysová logika věci a agregace dat podléhala požadavkům plynoucím z projektového záměru. Při nahrávání dat do finálních struktur dochází k tvorbě „*partitions*“, které značně zjednodušují a zrychlují následný proces dolování dat a taktéž případné úpravy konkrétního loadu. Z praxe se velmi osvědčila metoda prvně nahrávaná data přenést do, velmi zjednodušeně řečeno, univerzálních struktur, které neřeší např. datové typy, tím zabezpečíme skutečné nahrání všech dat, které protekli DataHubem, a až v rámci dalšího jejich zpracovávání do finálních struktur je nastavena transformace do jednotlivých datových typů a jejich následná kontrola, případně oprava nalezených chyb. Z pohledu zachování kvality dat je nutné správně nastavit proces vyřazování záznamů či rovnou celého loadu pomocí rozdělení chyb na dílčí, které vyřadí pouze některé nahrávané záznamy do logovací tabulky a upozorní administrátora na jejich výskyt, kdy je vhodné počkat na jeho rozhodnutí o tom, zda vyřazené záznamy významně narušují chod celého DW a jeho aktualizace musí posečkat na odstranění chyb, nebo se jedná jenom o dílčí problém a jeho odstranění nemá vliv na zbytek obsažených dat, např. jeden chybějící zaměstnanec vyřazený díky maximální délce sloupce pro příjmení, který je nově nastupující do organizace a tudíž jeho výskyt ve faktových datech není reálný, a DW lze bez potíží aktualizovat nahrávanými daty. V případě vyřazení celého loadu je samozřejmě jakákoliv aktualizace nemožná a je nutné mít správně nastavenou komunikační a eskalační matici, dle které postupovat v případě jeho výskytu.

Při implementaci DW nad OF byly prakticky sloučeny fáze nahrávání dat a jejich úpravy, kdy druhé uvedené se děje prakticky v momentě, kdy jsou data nahrávána do svých koncových struktur. Tento přístup se osvědčil zejména v době, kdy vznikaly jednotlivé DM, které nevyužívali tolik agregací dat a jejich rozšiřování o potřebné dimenze nebylo náročné, např. majetkový modul, ovšem s postupujícím časem se ukázalo toto řešení jako vývojově a časově náročné a taktéž dochází mnohem častěji k pozastavení aktualizace celého DW z důvodu chybějících dat, kdy nastavený proces kontroly chyb je postaven na principu „buď je všechno nebo nic“. Eskalační matice v tomto případě obsahuje jak povinnost informovat byznysové uživatele, tak support na straně korporace a to ji dělá poměrně složitou z důvodu podpory umístěné mimo území



České republiky. Standardně má DW zpoždění aktuální datum mínus jeden den, ovšem v případě závažných problémů, způsobených zejména dodáním špatných vstupních souborů, z pohledu kvality dat v nich obsažených, je toto zpoždění mnohdy delší než 3 pracovní dny.

### ***3.6.8 Odpovědnosti za jednotlivé procesy***

Tato aktivita sama o sobě neohrožuje technické provedení DW, ovšem na jeho používání má zásadní vliv, zejména pokud se vyskytnou nestandardní události a je potřeba řešit jejich nápravu. Přirozeností lidí je nehlásit se o problémy, kterých řešení by jim způsobilo nepohodlí, v tomto konkrétním případě nutnost něco řešit případně spravovat. Z toho důvodu je nutné správně a přesně specifikovat, komu patří které procesy a kdo je za ně zodpovědný, včetně SLA podmínek. Z praxe je velmi dobré pojmenovávat zodpovědnou osobu nikoliv jménem, ale buď názvem pozice, anebo konkretizováním pracovní náplně, zejména v podnicích, kde je dlouhodobě vysoká fluktuace zaměstnanců.

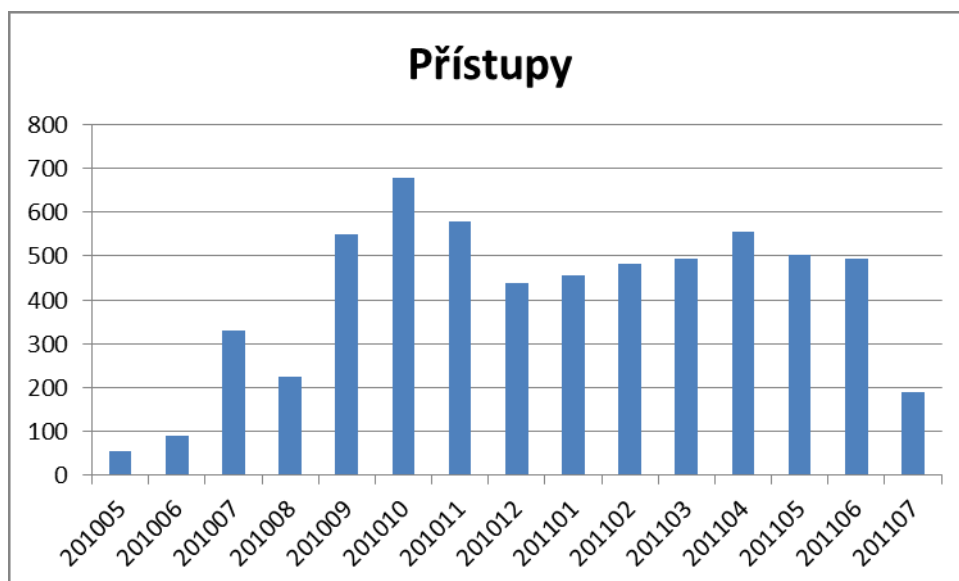
Jak vyplynulo z kapitol výše, největším problémem v GE je zejména rozdělení SLA mezi lokální IT, kde jeho plnění je na velmi vysoké úrovni, a globální podporu umístěnou ve zmiňované Číně, kdy tam plnění SLA lze označit jako ne zcela uspokojivé. Zejména v případech chybějících vstupních dat na DataHubu dochází k velkým diskuzím ohledně strany, která je zodpovědná za proces těchto přesunů

### ***3.6.9 Specifikace rolí a přístupových práv***

Pokud není implementovaný DW veřejně dostupným zdrojem pro širokou veřejnost, tak obvykle data v něm obsažená budou obsahovat citlivé údaje a je potřeba rozlišit, který typ uživatele má mít dostupná kterou oblast, či konkrétně který typ dat. Jako příklad lze označit dimenzi zákazníků, kdy zejména osobní údaje podléhají přímo legislativním požadavkům na jejich zveřejňování a uchovávání.

V případě DW implementovaného v GE nejsou přístupová práva na jednotlivá data v něm obsažená vůbec řešena na úrovni databáze a to z důvodu velmi malého počtu uživatelů, kteří do DW přistupují přímo pomocí databáze a ne prostřednictvím některého z nástrojů BI vrstvy, která je detailněji popsána níže.

**Graf č. 1:** Počet unikátních přístupů (za den) do webových reportů nad DW



**Zdroj:** vlastní tvorba GE Money Bank, a.s.

### 3.7 Návrh BI vrstvy

Pro koncové uživatele je správně navržená BI vrstva klíčová pro pohodlné a rychlé používání dat z DW. Její návrh by neměl být příliš složitý a taktéž by neměl obsahovat přílišnou volnost, co se týká způsobu přístupu a použitých aplikací, skrze které dochází k jejich dolování, případně reportování dále. Z uživatelského hlediska je vždy velmi pohodlné mít možnost výstupu do tabulkového editoru, nejvíce rozšířeným produktem je v této oblasti MS Excel, ovšem je velmi vhodné neřešit tento požadavek pomocí nastavení ODBC a vysvětlení, jakým způsobem se dají v Excelu data importovat. V současné době lze na trhu najít nepřeberné množství nástrojů pro BI řešení, kdy prakticky každá větší firma má svou vlastní aplikaci, ovšem jejich napojení na ostatní datové zdroje je v dnešní době naprostým standardem a ve většině případů nečiní žádné potíže. Při samotné implementaci této vrstvy je vhodné rovněž pamatovat na přístupová práva a role, kdy je nutné si uvědomit, že do DW se BI hlásí pod aplikačním účtem a jeho oprávnění jsou velmi často značně benevolentní z důvodu udržování jedné vrstvy, kde jsou interpretovány vazby mezi datovými entitami a jejich způsob napojení na samotný DW. Samotné řešení BI má v sobě obsaženo taktéž řízení uživatelských oprávnění, ovšem značně závisí do jaké míry a jak skutečně dochází

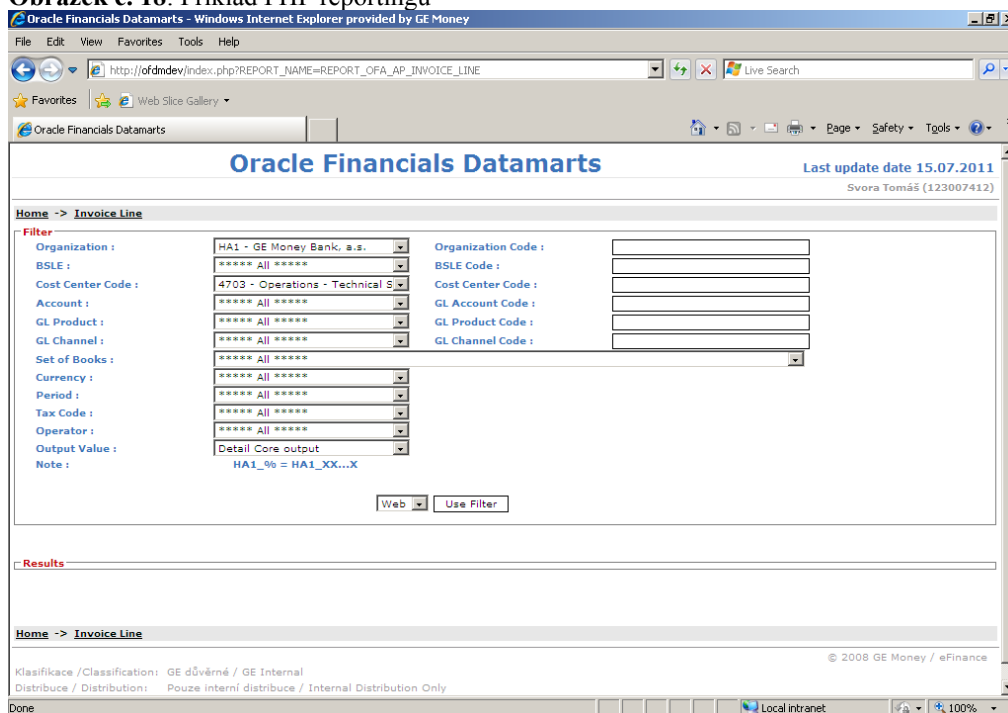
k jejich uplatňování, či na úrovni skupin reportů, resp. složek reportů, či na samotných zobrazovaných datech v jednotlivých reportech. V rámci návrhu BI je nutné také zahrnout správné rozdělení uživatelských rolí z pohledu znalostí daného prostředí a to na klasického uživatele, kterým je typicky člověk používající jenom předdefinované reporty, tvůrce reportů, což jsou lidé schopní sestavit tyto reporty, ovšem stále bez nutnosti znalostí o fungování DTB a poté správce prostředí mezi DW a BI, který musí znát jak způsob propojení dat v DW, tak byznysové požadavky, aby mohl správně navrhnout univerza.

V reálném prostředí GE je BI vrstva nad DW reprezentována dvěma hlavními směry a to pomocí *PHP*<sup>30</sup>, kde odpadá nutnost zřizování univerza a počet administrátorů je snížen prakticky na členy oddělení vývoje aplikací, kteří současně figurují jako správci celého DW řešení a nastavení jakýchkoliv restrikcí je tím pádem beze smyslu. Samotné práva na jednotlivé reporty jsou nadále rozdělena ještě na oprávnění na jednotlivé organizační složky podniku a dále na nákladová střediska, kdy tyto dvě oblasti jsou vzájemně odděleny tak, že nelze u dvou různých reportů zobrazit stejnému uživateli jiný výčet nákladových středisek. Toto řešení bylo zvoleno zejména z důvodu ušetření administrativní náročnosti a také protože skupina uživatelů má ve velké většině případů oprávnění na jakýkoliv nižší detail, než je právě kombinace podnikové organizace a nákladového střediska. PHP se rovněž osvědčily z pohledu rychlosti a schopnosti adaptace požadovaných funkcí na reporting a tím, že je jejich vývoj sjednocen do jednoho týmu, nedochází k duplicitě vytvářených reportů a taktéž každý z těchto reportů má snadno dohledatelného vlastníka, resp. tvůrce.

---

30 Skriptovací jazyk pro dynamické generování WWW stránek, více dostupné online z [www.php.net](http://www.php.net), datum poslední aktualizace dne 31.7.2011

Obrázek č. 18: Příklad PHP reportingu



Zdroj: Vlastní tvorba, GE Money Bank, a.s.

Druhou stěžejní aplikací pro reporting z DW jsou integrovaná Business Objects od společnosti SAP, která jsou primárním reportovacím systémem v GE a jejichž rozšíření také o univerzum obsahující data z implementovaného DW je otázka nejbližších dvou kvartálů. Řešení pomocí BO lze označit za více „*user friendly*“, kdy koncový uživatel má jak možnosti lépe ovlivnit data zobrazovaná v konkrétním reportu, tak zejména jejich filtrování a výstupy na lokální úložiště nabízejí více možností než jenom CSV nebo MS Excel.

### 3.8 Implementace DW

Při dodržení zásad uvedených v kapitolách výše se stává samotná implementace poměrně snadnou záležitostí převodu předcházejících analýz a specifikací do technického provedení. Samotná praktická ukázka jeho kroků již není obsahem této práce a v jejích zdrojích lze nalézt odkazy na materiály, kde je tato problematika velmi pečlivě řešena.

## 4 Závěr

Implementace DW se na první pohled může jevit jako poměrně snadná záležitost. Je to systém, do kterého stačí lít nějaká data, on prakticky sám o sobě vůbec nic nedělá a jeho případná nefunkčnost přímo podnik neohrožuje. Tato představa ovšem padá ihned s počátečními fázemi projektu, který jej má pořídit. Pokud má DW splňovat všechny požadavky na něj kladené, zejména co se týká rychlosti odezvy a zejména správnosti obsažených dat, je nezbytné, aby všechny fáze jeho implementace byly provedeny důkladně a v souladu s požadavky zákazníků, které většinou tvoří zaměstnanci společnosti, kteří mají za úkol připravovat pro vedení podniku analýzy o aktuálním stavu jejich společnosti.

Na oblast Data Warehousingu budou kladeny čím dál větší nároky na integraci všech možných datových vstupů, souhrn dat ze všech primárních i podpůrných systémů, a taktéž na rychlost zpracování dat tak, aby data v něm obsažená byla pokud možno co nejaktuálnější. Těmto požadavkům lze vyhovět jenom v případě, že DW je správně integrován v celkovém řešení ICT/IS a jeho rozšiřování a customizace jsou velmi otevřeny potřebám konkrétního podniku.

### 4.1 Shrnutí cílů

Cílem této práce bylo seznámit její čtenáře jednak s celkovou problematikou životního cyklu DW a jeho implementace do reálného prostředí v GE Money Bank, a.s. Shrnutí problematiky životního cyklu je projednáno v rámci teoreticko-metodologické části s tím, že snaha byla nevypisovat a srovnávat již popsání řešení, ale přispět k tomuto tématu vlastními zkušenostmi a názory. Dodržení tohoto cíle bylo složité, protože člověk má většinou tendenci zjednodušovat si život a v souvislosti s touto prací se to projevovalo zejména v touze nadužívat citací a cizích zdrojů. V praktické části bylo cílem této práce zejména popsat reálnou implementaci DW v GE Money Bank, a.s., z pohledu řízení tohoto projektu a upozornění na oblasti, ve kterých docházelo k problémům.

V teoretické části této práce byl popsán počátek životního cyklu aplikace v obecné rovině vlastními slovy takovou formou, aby byla snadno uchopitelná pro

netechnického čtenáře a pravdivost popsaných procesů byla snadno ověřitelná v následné praktické části. Její členění je koncipováno tak, jak na sebe jednotlivé procesy logicky navazují a vychází z primární literatury s tím, že jednotlivé popisy jsou uzpůsobeny přímo pro implementaci DW, která je specifickým případem aplikace a z toho důvodu jsou fáze jako např. analýza podnikových systémů a požadavků na funkčnost odlišné od průběhu stejné fáze při implementaci kteréhokoliv primárního systému. Mimo to obsahuje teoretická část také definice klíčových pojmů a přístupů pro tvorbu DW, kdy lze pozorovat dva stěžejní a nejznámější přístupy a to pánů Inmona a Kimballa, kdy v rámci této práce jsou stručně popsány jak ony samotné, tak zejména rozdíly, kterými se navzájem odlišují. Cíl této oblasti nednauživat cizích zdrojů a spíše teoretizovat poznatky z praxe se nepodařilo splnit dle očekávání autora a to zejména v oblasti vymezení základních pojmů. Oblast životního cyklu aplikace je již popsána vlastními slovy, samozřejmě s využitím zdrojů, které sloužily zejména jako nástroj pro utřídění si vlastních myšlenek.

V praktické části jsou rozebrány jednotlivé fáze tak, jak byly prováděny u reálné implementace DW v GE, kdy byl kladen důraz zejména na oblasti, které činily při této činnosti potíže a kterých je vhodné se při všech dalších podobných projektech vyvarovat. Praktická část je opět opřena zejména o praktické zkušenosti autora a je z ní patrná jeho snaha o vlastní přínos této práce spíše než o průzkum jednotlivých metod popsaných jinde.

## **4.2 Subjektivní zhodnocení očima autora**

Jako autor této práce jsem podléhal při implementacích různých databází pocitům, že tato oblast je vlastně hrozně jednoduchá a jedná se v ní pouze o znalosti PL/SQL, ovšem s odstupem času musím velmi poděkovat zkušenějším kolegům, kteří měli tu trpělivost a správně brzdili moje nadšení, tendenci podceňovat jednotlivé kroky a snahu rovnou se vrhnout na psaní skriptů pro přenosy dat, protože jenom díky jejich zkušenostem a ochotě předat je dál nedošlo k závažnějším ztrátám jak časovým tak finančním. Osobně se snažím tento přístup aplikovat i na členy mého týmu a musím

přiznat, že není vždy úplně jednoduché nenechat se svést myšlenkou na to, nechat dotyčného analytika na vlastní kůži vyzkoušet, co způsobí jeho zbrkllost.

Při samotném psaní této práce jsem měl velmi často tendence uchýlit se jak k nadužívání cizích zdrojů ve formě citací, tak na druhé straně zaběhnout do velkého detailu jednotlivých fází, kdy v případě některých z nich, jako např. fáze návrhu DW si dokážu představit jako samostatné téma pro práci o rozsahu 20 a více stran.

## 5 Literatura

### Primární zdroje

GÁLA L, POUR J, ŠEDIVÁ Z, Podniková informatika. Praha: Grada Publishing, a.s., 2009, ISBN 978-80-247-2615-1

INMON William Harvey. Building the Data Warehouse. 3rd ed. Wiley Publishing. Canada., 2002, ISBN 0-471-08130-2

KIMBALL R., ROSS M., The Data Warehouse Toolkit, 2th ed. Wiley Publishing. Canada., 2002, ISBN 0-471-20024-7

### Zdroje nedostupné online

KIMBALL R., CASERTA J. The Data Warehouse ETL Toolkit. 1th ed.. Wiley Publishing. Canada.. 2004. eISBN 0-764-57923-1

POUR J., Informační systémy a technologie. 1. vyd. Vysoká škola ekonomie a managementu. Praha. 2006. ISBN: 80-86730-03-4

LACKO L., Datové Sklady, analýza OLAP a dolování dat. 1. vyd. Computer Press. Brno. 2003. ISBN 8-07226-969-0

RAINARDI V., Building a Data Warehouse: With Examples in SQL Server. Apress. USA. 2008. ISBN 978-1-59059-0

KRÁL L., Využití datového skladu jako zdroje pro Business Intelligence. Bakalářská práce – Unicorn College. Praha. 2010

KHUN J., Návrh metodiky implementace Data Warehouse. Diplomová práce – Vysoká škola ekonomická v Praze. Praha. 2005



## **Internetové zdroje**

WIKIPEDIA: Tabulka databáze. Dostupná online z WWW:

<[http://cs.wikipedia.org/wiki/Tabulka\\_\(datab%C3%A1ze\)](http://cs.wikipedia.org/wiki/Tabulka_(datab%C3%A1ze))>, datum poslední revize dostupnosti odkazu dne 9.7.2011

ŘÍZENÍ INTERNETOVÉHO PROJEKTU. Dostupná online z WWW:

<<http://www.scribd.com/doc/53062183/18/Fyzick%C3%BD-datov%C3%BD-model>>, datum poslední revize dostupnosti odkazu dne 11.7.2011

LEARN DATA MODELING. Dostupná online z WWW:

<<http://www.learn-datamodeling.com/ldm.htm>>, datum poslední revize dostupnosti odkazu dne 10.7.2011

WIKIPEDIA: Databáze. Dostupná online z WWW:

<<http://cs.wikipedia.org/wiki/Datab%C3%A1ze>>, datum poslední revize dostupnosti odkazu dne 9.7.2011

THE DATA WAREHOUSING INFORMATION CENTER. Dostupná online z WWW:

<<http://www.dwinfocenter.org/defined.html>>, datum poslední revize dostupnosti odkazu dne 11.7.2011

WIKIPEDIA: Data Warehouse. Dostupná online z WWW:

<[http://en.wikipedia.org/wiki/Data\\_warehouse](http://en.wikipedia.org/wiki/Data_warehouse)>, datum poslední revize dostupnosti odkazu dne 23.7.2011

DIONÉ – ZČU. Dostupná online z WWW: <<http://v1.dione.zcu.cz/java/uvod.html>>, datum poslední revize dostupnosti odkazu dne 23.7.2011

WIKIPEDIA: ETL. Dostupná online z WWW:

<[http://en.wikipedia.org/wiki/Extract,\\_transform,\\_load](http://en.wikipedia.org/wiki/Extract,_transform,_load)>, datum poslední revize dostupnosti odkazu dne 23.7.2011

ITIL. Dostupná online z WWW: <<http://www.ital.cz/index.php?id=982>>, datum poslední revize dostupnosti odkazu dne 23.7.2011

ZIKMUND M., Kdo jsou to vlastně stakeholders a proč a jak se o ně zajímat. Dostupná online z WWW: <<http://www.businessvize.cz/strategie/kdo-jsou-to-vlastne-stakeholders-a-proc-a-jak-se-o-ne-zajimat>>, datum poslední revize dostupnosti odkazu dne 23.7.2011

SIEBER UCYHTIL, s.r.o, Dostupná online z WWW: <<http://sieber-uchytil.cz/studie-proveditelnosti-feasibility.html>>, datum poslední revize dostupnosti odkazu dne 23.7.2011

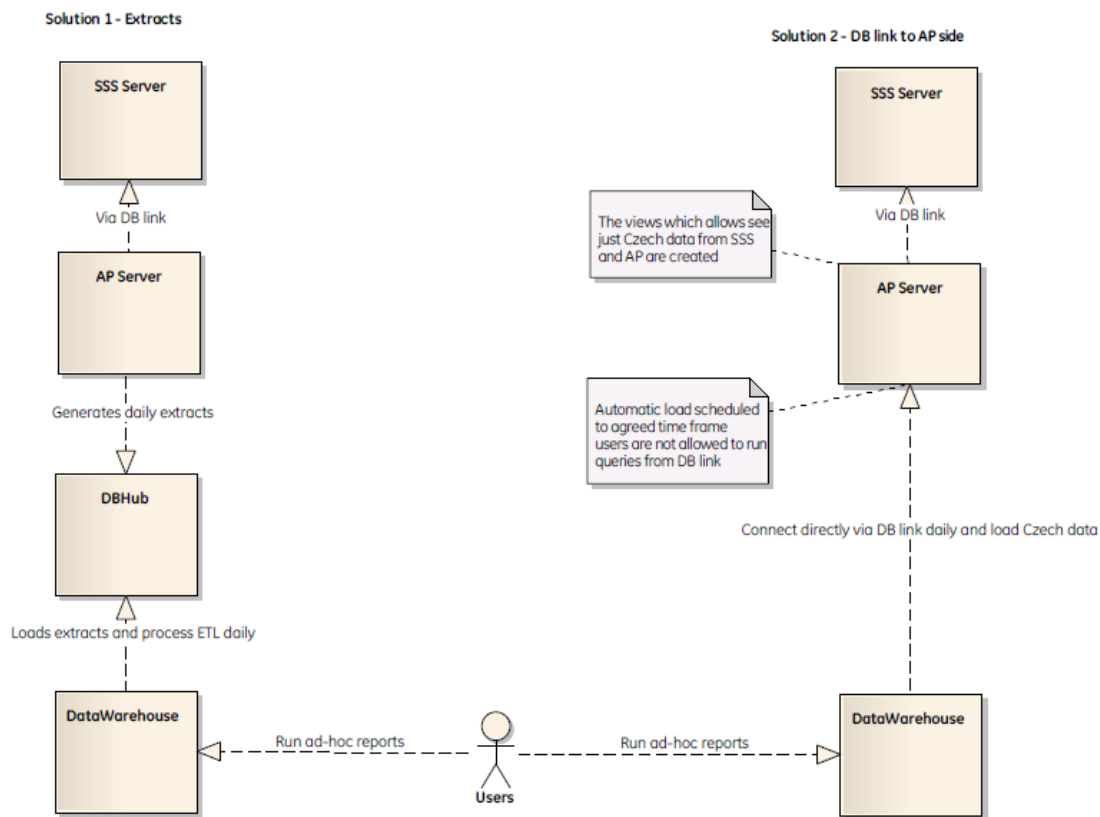
WIKIPEDIA: Scope. Dostupná online z WWW:  
<[http://en.wikipedia.org/wiki/Interface\\_\(computing\)](http://en.wikipedia.org/wiki/Interface_(computing))>, datum poslední revize dostupnosti odkazu dne 23.7.2011

WIKIPEDIA: Master Data Management. Dostupná online z WWW:  
<[http://en.wikipedia.org/wiki/Master\\_data\\_management](http://en.wikipedia.org/wiki/Master_data_management)>, datum poslední revize dostupnosti odkazu dne 23.7.2011

ORACLE: The End User Layer. Dostupná online z WWW:  
<[http://download.oracle.com/docs/html/A86730\\_01/managing.htm#996910](http://download.oracle.com/docs/html/A86730_01/managing.htm#996910)>, datum poslední revize dostupnosti odkazu dne 23.7.2011

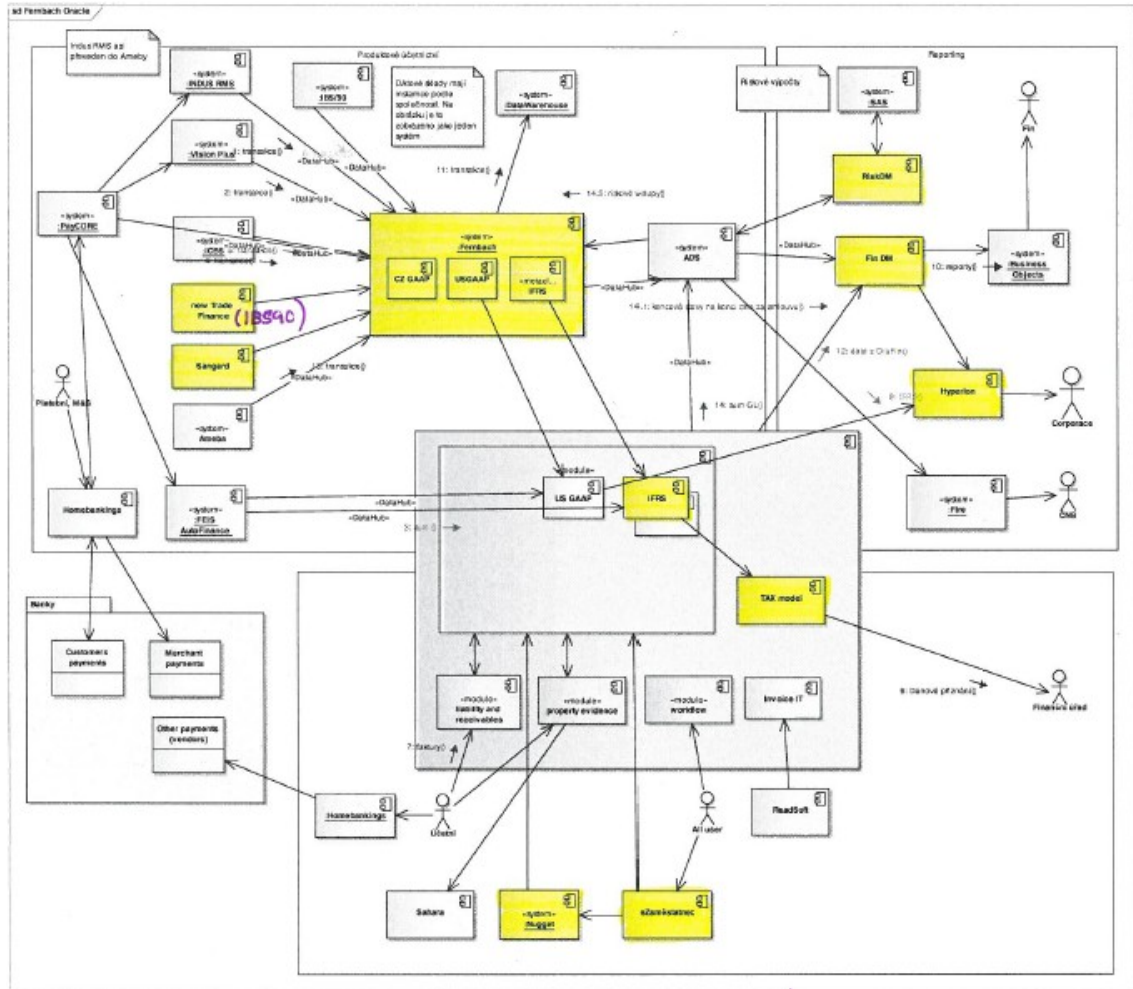
## 6 Přílohy

Příloha č. 1: Ukázka datových toků do DW v úvodní fázi analýzy vstupních zdrojů



**Zdroj:** vlastní tvorba, GE Money Bank, a.s.

Příloha č. 2: Celkové řešení architektury po implementaci Oracle Financial



Zdroj: vlastní tvorba, GE Money Bank, a.s.

### Příloha č. 3: Ukázka extraktu ze zdrojového systému

```
Lister - [x:\My Documents\vssem_bakalarka\bakalarka_new\zdroje\GL Extracts\U00220091006.TXT-4532783_4532784]
Soubor Upravit Možnosti Kódování Nápověda 16 %
ADJ-2-09|ADJ|ADJ|0|G01_26Period_01|26_Period|ADJ-2|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|12/31/2009
00:00:00|12/31/2009 00:00:00|01/01/2009 00:00:00|10/01/2009
00:00:00|2009|26|4|Y|12/31/3000 00:00:00
ADJ-1-09|ADJ|ADJ|0|G01_26Period_01|26_Period|ADJ-1|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|12/31/2009
00:00:00|12/31/2009 00:00:00|01/01/2009 00:00:00|10/01/2009
00:00:00|2009|25|4|Y|12/31/3000 00:00:00
DEC-STAT-09|STAT|DEC|12|G01_26Period_01|26_Period|DEC-STAT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|12/27/2009
00:00:00|12/31/2009 00:00:00|01/01/2009 00:00:00|10/01/2009
00:00:00|2009|24|4|N|12/31/3000 00:00:00
DEC-MNGT-09|MNGT|DEC|12|G01_26Period_01|26_Period|DEC-MNGT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|12/01/2009
00:00:00|12/26/2009 00:00:00|01/01/2009 00:00:00|10/01/2009
00:00:00|2009|23|4|N|12/31/3000 00:00:00
NOV-STAT-09|STAT|NOV|11|G01_26Period_01|26_Period|NOV-STAT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|11/22/2009
00:00:00|11/30/2009 00:00:00|01/01/2009 00:00:00|10/01/2009
00:00:00|2009|22|4|N|12/31/3000 00:00:00
NOV-MNGT-09|MNGT|NOV|11|G01_26Period_01|26_Period|NOV-MNGT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|11/01/2009
00:00:00|11/21/2009 00:00:00|01/01/2009 00:00:00|10/01/2009
00:00:00|2009|21|4|N|12/31/3000 00:00:00
OCT-STAT-09|STAT|OCT|10|G01_26Period_01|26_Period|OCT-STAT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|10/25/2009
00:00:00|10/31/2009 00:00:00|01/01/2009 00:00:00|10/01/2009
00:00:00|2009|20|4|N|12/31/3000 00:00:00
OCT-MNGT-09|MNGT|OCT|10|G01_26Period_01|26_Period|OCT-MNGT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|10/01/2009
00:00:00|10/24/2009 00:00:00|01/01/2009 00:00:00|10/01/2009
00:00:00|2009|19|4|N|12/31/3000 00:00:00
SEP-STAT-09|STAT|SEP|9|G01_26Period_01|26_Period|SEP-STAT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|09/20/2009
00:00:00|09/28/2009 00:00:00|01/01/2009 00:00:00|07/01/2009
00:00:00|2009|18|3|N|12/31/3000 00:00:00
SEP-MNGT-09|MNGT|SEP|9|G01_26Period_01|26_Period|SEP-MNGT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|09/01/2009
00:00:00|09/19/2009 00:00:00|01/01/2009 00:00:00|07/01/2009
00:00:00|2009|17|3|N|12/31/3000 00:00:00
AUG-STAT-09|STAT|AUG|8|G01_26Period_01|26_Period|AUG-STAT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|08/23/2009
00:00:00|08/31/2009 00:00:00|01/01/2009 00:00:00|07/01/2009
00:00:00|2009|16|3|N|12/31/3000 00:00:00
AUG-MNGT-09|MNGT|AUG|8|G01_26Period_01|26_Period|AUG-MNGT|CEE 26 Period
Calendar|CALENDAR|24 Months plus two Adjusting Periods|26|08/01/2009
00:00:00|08/22/2009 00:00:00|01/01/2009 00:00:00|07/01/2009
00:00:00|2009|15|3|N|12/31/3000 00:00:00
```

Zdroj: vlastní tvorba, GE Money Bank, a.s.