

**Univerzita Hradec Králové**  
**Fakulta informatiky a managementu**  
**Katedra informačních technologií**

**Návrh datového skladu v prostředí zdravotnické instituce**  
**Diplomová práce**

Autor: Bc. Jakub Pohl  
Studijní obor: Informační management – kombinovaná forma

Vedoucí práce: Mgr. Josef Horálek, Ph.D.

Prohlášení:

Prohlašuji, že jsem diplomovou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne: 28. 4. 2017

Bc. Jakub Pohl

Poděkování:

Děkuji vedoucímu diplomové práce Mgr. Josefu Horálkovi, PhD., za metodické vedení práce a cenné rady během jejího vypracování a své manželce a dceři za podporu v době studia, kdy každý den děkuji za to, že je mám, přičemž nezáleží na čísle stránky, ani na počtu znaků.

## **Anotace**

Tato diplomová práce se na základě získaných teoretických poznatků zabývá návrhem datového skladu pro specifické prostředí zdravotnické instituce. V souvislosti s tím je v práci nejprve představena architektura vrstev datových skladů a následně jeho samotná architektura, u které jsou datové sklady blíže definovány z pohledu dvou nejvýznamnějších osobností v tomto oboru, Williama H. Inmona a Ralpa Kimballa. Teoretická část dále popisuje dimenzionální přístup k ukládání dat, multidimenzionální databázi, OLAP kostku a s ní spojenou technologii OLAP.

Praktická část této práce obsahuje návrh a implementaci funkčního modelu datového skladu pro společnost Městská nemocnice, a.s., který byl po naplnění daty použit jako zdroj pro vytvoření uživatelských reportů a analýz. Účelem praktické části je také demonstrace možností a výhod tohoto řešení pro podporu rozhodovacích procesů v dané společnosti.

## **Annotation**

### **Title: Design of a data warehouse in environment of health care institution**

Based on theoretical knowledge this diploma thesis is dealing with design of a data warehouse in specific environment of health care institution. In connection with that, in this work is at first architecture of data warehouse layers and then its own architecture introduced, in which the data warehouses are specifically defined from the perspective of the two most important personalities in this field, William H. Inmon and Ralph Kimball. Furthermore, in the theoretical part the dimensional approach to storing data, multidimensional databases, OLAP cube and its associated OLAP technology are described.

The practical part of this work includes the design and implementation of a functional model of a data warehouse for the company Městská nemocnice, Inc., which has been after filling of data used as a source for creation user reports and analysis. The purpose of the practical part is also demonstration of the possibilities and advantages of this solution for decision support processes in this company.

## Obsah:

Úvod.....	1
1 Datové sklady.....	2
1.1 Vrstvy architektury.....	4
1.1.1 Architektura s jednou vrstvou.....	4
1.1.2 Architektura se dvěma vrstvami .....	5
1.1.3 Pokročilá architektura se třemi vrstvami .....	6
1.2 Architektura datového skladu .....	6
1.2.1 Architektura samostatných datových tržišť .....	7
1.2.2 Architektura a definice datového skladu podle Williama H. Inmona.....	7
1.2.3 Architektura a charakteristika datového skladu podle Ralpa Kimballa..	12
1.2.4 Federativní architektura .....	17
1.3 Neúspěšné řešení datového skladu.....	18
2 Systémy OLTP.....	19
2.1 Normalizovaný přístup k ukládání dat .....	19
2.1.1 První normální forma (1NF).....	20
2.1.2 Druhá normální forma (2NF).....	20
2.1.3 Třetí normální forma (3NF).....	20
2.1.4 Boyce-Coddova normální forma .....	20
2.1.5 Čtvrtá normální forma (4NF).....	20
2.1.6 Pátá normální forma (5NF).....	20
3 Dimenzionální přístup k ukládání dat .....	21
3.1 Tabulka faktů .....	21
3.2 Tabulka dimenzí.....	22
3.2.1 STAR schéma .....	23
3.2.2 SNOWFLAKE schéma.....	24
3.3 OLAP kostka.....	25
3.3.1 Operace spojené s OLAP kostkou .....	27

4	OLAP .....	28
4.1	Uložení dat v OLAP.....	32
4.1.1	Multidimenzionální OLAP (MOLAP).....	32
4.1.2	Relační databázový OLAP (ROLAP).....	33
4.1.3	Hybridní OLAP (HOLAP).....	33
4.1.4	Desktopový OLAP (DOLAP).....	33
5	Business Intelligence.....	35
5.1	Nástroje a aplikace BI .....	36
6	Specifikace společnosti Městská nemocnice, a.s. ....	38
6.1	Zdrojové systémy .....	38
7	Výběr vhodného řešení datového skladu .....	41
7.1	Definování požadavků organizace .....	42
7.2	Návrh dimenzionálního modelu.....	42
7.3	Návrh fyzického modelu a technologické architektury .....	45
7.4	Šablony rutin ETL.....	47
7.4.1	ETL rutiny pro extrakci dat .....	48
7.4.2	ETL rutiny pro transformaci a nahrání dat .....	49
7.5	Implementace pilotního DW .....	54
8	Návrh uživatelských reportů .....	57
9	OLAP analýza .....	61
9.1	Vytvoření multidimenzionální databáze .....	61
9.2	Analýza .....	62
10	Testování navrženého řešení .....	64
11	Hodnocení navrženého řešení .....	66
11.1	Funkční přínosy.....	66
11.2	Ekonomické přínosy.....	67
12	Závěr .....	68
13	Seznam použité literatury.....	69

## Seznam obrázků

Obrázek č. 1 Základní komponenty systému DW (upraveno dle: Laberge, 2012, s. 37).	3
Obrázek č. 2 Architektura s jednou vrstvou (upraveno podle: Laberge, 2012, s. 221) ....	4
Obrázek č. 3 Architektura se dvěma vrstvami (upraveno podle: Laberge, 2012, s. 222).	5
Obrázek č. 4 Architektura se třemi vrstvami (upraveno podle: Laberge, 2012, s. 223)...	6
Obrázek č. 5 Architektura datových tržišť (upraveno podle: Laberge, 2012, s. 225) .....	7
Obrázek č. 6 Problematika integrace (zdroj: Inmon, c2005, s. 31). .....	9
Obrázek č. 7 Trvalost uložených dat (zdroj: Inmon, c2005, s. 32).....	10
Obrázek č. 8 Architektura centrálního datového skladu dle Williama H. Inmona (upraveno podle: <a href="https://kurtzahra.com/2015/06/20/inmon-or-kimball-the-debate-of-the-century/">https://kurtzahra.com/2015/06/20/inmon-or-kimball-the-debate-of-the-century/</a> ) .....	11
Obrázek č. 9 Základní elementy datového skladu (zdroj: Kimball a Ross, c2002, s. 7)	13
Obrázek č. 10 Sběrnice podnikového datového skladu (zdroj: Kimball a Ross, c2002, s. 78).....	15
Obrázek č. 11 Federativní architektura (upraveno podle: Laberge, 2012, s. 227).....	17
Obrázek č. 12 Schéma hvězdy (upraveno podle: <a href="https://en.wikipedia.org/wiki/Star_schema">https://en.wikipedia.org/wiki/Star_schema</a> ) .....	24
Obrázek č. 13 Schéma sněhové vločky (upraveno podle: <a href="https://en.wikipedia.org/wiki/Snowflake_schema">https://en.wikipedia.org/wiki/Snowflake_schema</a> ).....	25
Obrázek č. 14 OLAP kostka (upraveno podle: Lacko, 2003, s. 32) .....	26
Obrázek č. 15 Nástroje a komponenty Business Intelligence (upraveno podle: <a href="https://www.systemonline.cz/business-intelligence/data-uz-nestaci-manazerske-systemy-a-nastroje-pro-podporu-rozhodovani-ve-verejne-a-statni-sprave.htm">https://www.systemonline.cz/business-intelligence/data-uz-nestaci-manazerske-systemy-a-nastroje-pro-podporu-rozhodovani-ve-verejne-a-statni-sprave.htm</a> ).....	37
Obrázek č. 16 Znázornění systémů ve společnosti MNDK (zdroj: autor).....	40
Obrázek č. 17 Konceptuální model pilotního datového skladu (zdroj: autor).....	42
Obrázek č. 18 Logický model pilotního datového skladu (zdroj: autor) .....	43
Obrázek č. 19 Fyzický model pilotního datového skladu (zdroj: autor) .....	47
Obrázek č. 20 Vzor ETL rutin extrakce dat – control flow (zdroj: autor).....	48
Obrázek č. 21 Vzor ETL rutin extrakce dat – data flow (zdroj: autor).....	49
Obrázek č. 22 Vzor ETL rutin nahrání dat do pilotního DW – control flow (zdroj: autor) .....	50
Obrázek č. 23 Ukázka ETL rutin transformace a nahrání dat do pilotního DW – data flow (zdroj: autor).....	51

Obrázek č. 24 Doplnění atributu OPI do faktové tabulky (zdroj: autor) .....	51
Obrázek č. 25 Ukázka některých duplicit v atributu Zamestnani (zdroj: autor).....	52
Obrázek č. 26 Použití Fuzzy Grouping Transformation pro vyčištění atributu Zamestnani (zdroj: autor).....	53
Obrázek č. 27 Použití Fuzzy Lookup Transformation pro čištění atributu Mesto (zdroj: autor) .....	54
Obrázek č. 28 První uživatelský report (zdroj: autor) .....	58
Obrázek č. 29 Druhý uživatelský report (zdroj: autor).....	59
Obrázek č. 30 Report upravený pro potřebu primáře chirurgického oddělení (zdroj: autor).....	60

## **Seznam grafů**

Graf č. 1 První ukázka OLAP analýzy (zdroj: autor) .....	63
Graf č. 2 Druhá ukázka OLAP analýzy (zdroj: autor).....	63



## **Seznam zkratk**

DW – Data warehouse

BI – Business Intelligence

ETL – Extract Transform Load

ODS – Operational Data Store

OLTP – Online Transaction Processing

OLAP – Online Analytical Processing

MDD – Multidimensional Database

CRM – Customer Relationship Management

ERP – Enterprise Resource Planning

NIS – Nemocniční Informační Systém

LIS – Laboratorní Informační Systém

# Úvod

Konkurenceschopnost každé firmy je v dnešní době podmíněna správným rozhodováním v klíčových situacích, ve kterých jsou na management podniku kladeny čím dál větší nároky. Stále významnější podíl na tom má turbulentní a globalizované prostředí, které vlivem neustálých změn nutí firmy a podniky k mnohem rychlejšímu rozhodování, než tomu bylo dříve. Tato rozhodnutí již nevycházejí jen ze zkušeností a znalostí klíčových manažerů, ale stále častěji musí mít základ ve správných informačních podkladech. S vývojem informačních technologií však rapidním způsobem narůstá objem firemních dat obsahující cenné, ale běžným způsobem nedostupné informace, které lze využít pro podporu rozhodování. Pro potřeby sjednoceného přístupu k těmto datům, jejich správnému pochopení nebo použití při pokročilých analýzách byl na konci osmdesátých let vytvořen systém datových skladů.

Hlavním cílem této diplomové práce je návrh datového skladu ve specifickém prostředí zdravotnické instituce. V teoretické části je představen systém datových skladů, architektura jeho vrstev a následně jeho samotná architektura, ve které je DW blíže definován z pohledu nejvýznamnějších osobností tohoto oboru, Williama H. Inmona a Ralpha Kimballa. Dále tato práce vysvětluje přístup k dimenzionálnímu ukládání dat, multidimenzionální databázi a technologii OLAP, která pro své pokročilé analýzy využívá právě MDD.

V praktické části práce bude navržen model pilotního datového skladu se všemi jeho součástmi pro společnost Městská nemocnice, a.s. Jejím cílem je vytvoření a implementace fungujícího modelu DW, jeho naplnění daty navrženými ETL rutinami s využitím ODS, a poté s pomocí odpovídajících nástrojů demonstrovat možnosti a výhody jeho použití pro potřeby vedoucích pracovníků výše zmíněné organizace. Poslední kapitoly praktické části jsou zaměřeny na vytvoření uživatelských reportů, OLAP analýzu a testování a hodnocení navrženého řešení.

Tato práce vychází ze skutečnosti, že podstatná část softwarového vybavení této organizace tvoří produkty firmy Microsoft, a proto je také návrh řešení a jeho implementace založena na produktu Microsoft SQL Server 2008 Enterprise a většině jeho nástrojů.

# 1 Datové sklady

Jedním z nejdůležitějších a nejcennějších aktiv jakékoliv organizace jsou její data. Tato aktiva jsou téměř vždy používána ze dvou důvodů: archivování provozních dat a analytické rozhodování.

„V případě nevhodně udržovaných nebo nesprávných dat budou pravděpodobně nesprávná i podniková rozhodnutí založená na těchto informacích. Z tohoto důvodu je zásadně důležité zajistit patřičnou správu dat na vysoké úrovni transparentnosti, kvality a dostupnosti. Kvůli dosažení uvedeného cíle organizace budují systémy datových skladů, které umožňují používat analytické funkce business intelligence.“ (Laberge, 2012, s. 241)

Zjednodušeně řečeno, provozní systémy (operational systems) slouží k ukládání dat a datové sklady (DW) a systémy business intelligence (BI) pro získávání dat zpět. Uživatelé provozních systémů jsou tedy zodpovědní za samotný chod organizace, vyřizují objednávky, registrují nové zákazníky, sledují stav provozních činností, přijímají stížnosti atd. Uživatelé DW a systémů BI naopak chod organizace sledují a hodnotí výkon (produkci), sčítají nové objednávky a porovnávají je s objednávkami např. z minulého týdne, zjišťují, proč se noví zákazníci registrovali nebo na co si zákazníci stěžovali. (Kimball a Ross, 2013, s. 2)

Samotný počátek datových skladů je spojen s vývojem a rozvojem tzv. decision support systems (DSS), neboli systémů pro podporu rozhodování. Mnoho firem si uvědomilo důležitost historicky uložených dat a jejich významu při následných analýzách, které mohou být využívány pro podporu rozhodování. DSS byly odpovědí na otázku, jak využít nashromážděná firemní data, která obsahují cenné informace využitelné pro rozvoj dané firmy. Tato data po svém nasazení začaly produkovat provozní informační systémy v ohromném množství. Ačkoli existují velmi rozsáhlá řešení, jako je např. SAP R/3<sup>1</sup>, v žádné společnosti nebyl a není používán pouze jeden komplexní provozní systém.

Provozní informační systémy slouží obvykle velmi dobře k tomu, k čemu byly navrženy, vyvinuty a implementovány (umožňují evidovat zákazníky, vystavovat objednávky a faktury, párovat příchozí platby od zákazníků, rezervovat zboží na skladě, evidovat transakce na účtech zákazníků, plánovat výrobu, přenášet data do účetnictví nebo kontrolovat plnění objednávek). Pokud bychom ale chtěli za pomoci stejných systémů

---

<sup>1</sup> SAP R/3 je softwarovým produktem společnosti SAP, který slouží pro řízení podniku.

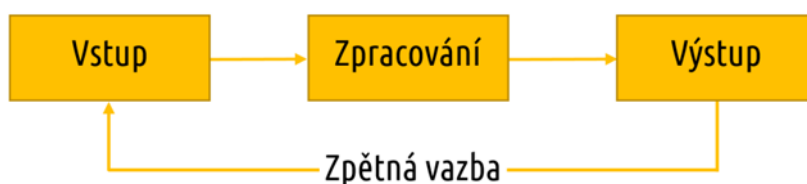
vzniklá data následně zpracovat takovým způsobem, aby byla použitelná pro analýzy nebo rozhodování (předpověď počtu zakázek na příští měsíc, eliminace sezónních výkyvů při posuzování trendu v příjmech nebo zjištění společné charakteristiky zákazníků kupující stejný výrobek), potýkali bychom se se značnými problémy. (Hlavní principy datových skladů a proces jejich vytváření, 2000)

„Společnosti proto financují projekty s cílem uspořádat, organizovat, vyčistit, popsat a centralizovat podniková data. Tyto komplexní systémy poskytují zásadní informace, které společností dávají výraznou konkurenční výhodu na trhu.“ (Laberge, 2012, s. 23)

Mezi tyto komplexní systémy a nástroje pro podporu rozhodování lze zařadit Business Intelligence (BI), datové sklady, data mining nebo OLAP.

„Termín „datový sklad“ často označuje systém datového skladu a v jiných případech se vztahuje na úložiště datového skladu.“ (Laberge, 2012, s. 35)

V systému datového skladu stejně jako v jiných systémech jsou hlavními součástmi vstup, zpracování, výstup a zpětná vazba. Ve fázi vstupu, který je v systému DW spojen s identifikací a záznamem dat, má hlavní význam kvalita dat. Chybná data vedou k nepřesným výstupům a následně ke špatným podnikovým analýzám a rozhodnutím. Fáze zpracování převádí, uspořádává, analyzuje či ukládá data strukturovaným nebo uspořádaným způsobem. Data mohou být uložena v jedné velké databázi, nebo v odlišných sadách databází, které se nacházejí na různých serverech nebo v různých lokalitách. Výstupní fáze zajišťuje prezentaci zpracovaných informací jejich přenos k uživatelům, kteří je vyžadují. (Laberge, 2012, s. 37)



Obrázek č. 1 Základní komponenty systému DW (upraveno dle: Laberge, 2012, s. 37)

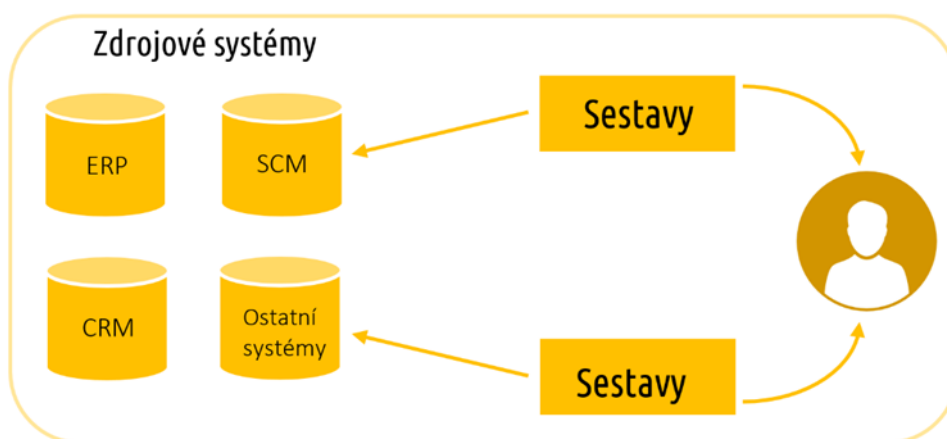
Zpětná vazba pak dle dosažených výstupů transformuje způsoby identifikace a záznamu dat na vstupu pro zpracování tak, aby následný výstup co nejvíce vyhovoval potřebám koncových uživatelů.

## 1.1 Vrstvy architektury

„Systém datového skladu lze vytvořit z několika propojených systémů, které se jednotlivě označují jako vrstvy. Díky izolaci těchto vrstev a znázornění jejich souvislostí je možné se namísto globální perspektivy zaměřit na jednotlivé systémy.“ (Laberge, 2012, s. 220)

### 1.1.1 Architektura s jednou vrstvou

Z důvodu nerozlišování architektury s jednou vrstvou mezi zdrojovými systémy a systémem datového skladu, a s tím souvisejícího přílišného zatěžování provozního systému, se architektura tohoto typu v oboru datových skladů vyskytuje už jen výjimečně.



Obrázek č. 2 Architektura s jednou vrstvou (upraveno podle: Laberge, 2012, s. 221)

„U každého analytického dotazu nebo rutiny vykazování dochází k přímému přístupu ke zdrojovému systému, což představuje dodatečné zatížení procesoru a vstupně-výstupního podsystému. Proto byl tento přístup převážně opuštěn již koncem 80. let 20. století.“ (Laberge, 2012, s. 221)

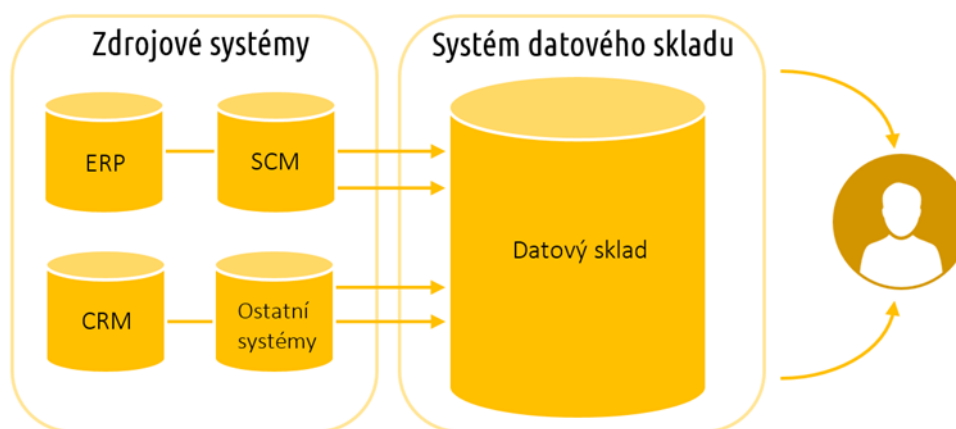
Nadměrné zatěžování procesoru a operační paměti serveru silně ovlivňuje jeho výkonnost a může narušit i základní funkce podniku. Architektura s jednou vrstvou nesplňuje ani klíčové požadavky na architekturu datových skladů, protože nerozlišuje analytické a transakční oblasti. Její jedinou výhodou může být v některých případech rychlá doba reakce a odezvy při dotazování.

### 1.1.2 Architektura se dvěma vrstvami

Klasická architektura se dvěma vrstvami zahrnuje zdrojové systémy a systém datového skladu, což představuje architekturu s centrálním úložištěm. Z provozní perspektivy je důležité rozlišit zdrojové systémy a systém datového skladu.

„Díky architektuře se dvěma vrstvami lze vytvořit nezávislý datový sklad, který je poté možné navrhnout s ohledem na optimalizované použití se zaměřením na jednotlivé vrstvy datového skladu.“ (Laberge, 2012, s. 223)

Izolace těchto systémů zajišťuje, aby provozní a transakční systémy pracovaly bez vzájemných závislostí mezi sebou a bez závislosti na systému datového skladu. Pomocí nových doplňků či změn v procesech plnění a ETL<sup>2</sup> rutin ale při dodržení definované terminologie systém datového skladu dovoluje připojení dalších systémů.



Obrázek č. 3 Architektura se dvěma vrstvami (upraveno podle: Laberge, 2012, s. 222)

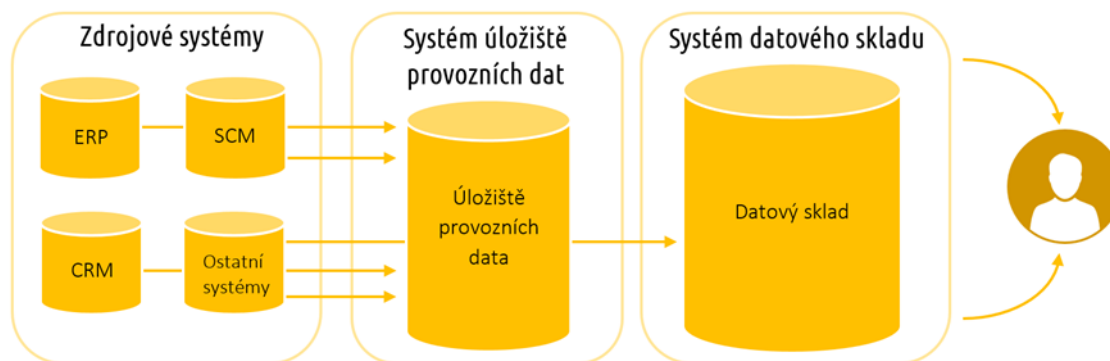
Tento přístup je vhodné použít v situacích, kdy není potřeba nebo není technologicky možné vytvořit celopodnikové řešení, které je založeno na třívrstvé architektuře, v případě nutnosti rychlého vybudování řešení pro několik mezi sebou nezávislých oddělení, když neočekáváme jejich budoucí potřebu sloučení do jednoho celku nebo když zadavatel není ochoten vynaložit nebo nevládní dostatečné finanční prostředky na počáteční činnosti integrace spojené s architekturou trvalého datového skladu. (Novotný, Pour a Slánský, s. 47)

<sup>2</sup> Zkratka z anglického Extract Transform Load. Podrobněji vysvětleno v kapitole Architektura a charakteristika datového skladu podle Ralpa Kimballa.

### 1.1.3 Pokročilá architektura se třemi vrstvami

Na rozdíl od dvouvrstvé architektury, architektura se třemi vrstvami zahrnuje další systém, kterým bývá zpravidla úložiště provozních dat (operational data store – ODS<sup>3</sup>).

„Hlavní princip spočívá v tom, že úložiště provozních dat představuje vlastní systém, který čerpá data ze zdrojových systému a následně je poskytuje systému datového skladu. Z toho tedy vyplývá označení architektury se třemi vrstvami.“ (Laberge, 2012, s. 224)



Obrázek č. 4 Architektura se třemi vrstvami (upraveno podle: Laberge, 2012, s. 223)

Na první pohled vypadá architektura se třemi vrstvami jako ideální kombinace rychlé doby odezvy architektury s jednou vrstvou a dvouvrstvé architektury se separací datového skladu jako přístupovým prvkem. Skutečnost je ale poněkud složitější, protože vytvoření datového skladu tohoto typu architektury trvá déle, je náročnější a finančně více nákladné.

## 1.2 Architektura datového skladu

Nejnáročnější aspekt datových skladů nespočívá v jejich technických obtížích, ale ve výběru nejlepšího přístupu k vybudování architektury datového skladu vhodné pro strukturu a kulturu společnosti a ve vyřešení organizačních otázek a otázek firemní politiky, které budou nevyhnutelně vznikat během její realizace.

Architekturu datového skladu lze dělit podle čtyř základních přístupů. V následujících kapitolách budou popsána samostatná datová tržiště, architektura centrálního datového úložiště Williama H. Inmona (přístup „shora dolů“), architektura orientovaná na datová tržiště Ralpa Kimballa (přístup „zdola nahoru“) a federativní architektura.

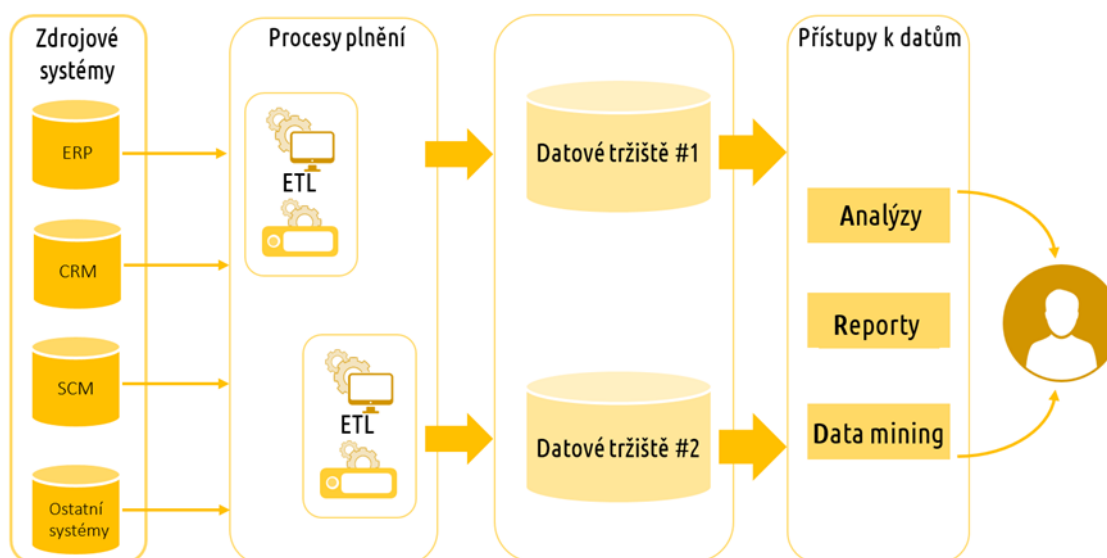
<sup>3</sup> Podrobněji vysvětleno v kapitole Architektura a charakteristika datového skladu podle Ralpa Kimballa.

## 1.2.1 Architektura samostatných datových tržišť

Samostatné datové tržiště nebo nezávislá datová tržiště, která nejsou nijak propojena, představují nejjednodušší formu datového skladu.

„Vytvoření této architektury datových tržišť začíná obvykle iniciativou na úrovni podnikového oddělení, jejímž cílem je kvůli vykazování centralizovat data do jedné databáze.“ (Laberge, 2012, s. 224)

První datová tržiště historicky vznikala převážně v aplikacích Microsoft Access a Excel. Na obrázku č. 5, který znázorňuje architekturu nezávislých datových tržišť, jsou zobrazeny i ETL rutiny, v mnoha případech jsou ale data nahrána přímo do datových tržišť a transformována nebo spravována až potom.



Obrázek č. 5 Architektura datových tržišť (upraveno podle: Laberge, 2012, s. 225)

„Dotazování probíhá přímo na základě datového tržiště s použitím kteréhokoliv nástroje, který je v dané chvíli k dispozici.“ (Laberge, 2012, s. 224)

## 1.2.2 Architektura a definice datového skladu podle Williama H. Inmona

William H. Inmon, který v roce 1991 poprvé definoval termín „data warehouse“, a který bývá často označován za „Otce datových skladů“ a významného odborníka a průkopníka této oblasti, ve své knize uvádí: „Datový sklad je subjektivně orientovaná, integrovaná, neměnná a trvale uložená kolekce dat sloužící pro podporu rozhodování. Datový sklad



obsahuje granulární korporátní data. Data v datovém skladu je možné použít pro mnoho různých účelů, včetně „sezení a čekání“ na budoucí požadavky, které jsou v současné době ještě neznámé.“ (Inmon, c2005, s. 29)

Subjektově orientovaná data jsou taková data, která se týkají vlastního předmětu podnikání. Jak zmiňuje Inmon (c2005, s. 29) pro výrobce mohou být hlavní oblasti zájmu produkt, objednávky, účet za materiál nebo hotové zboží. Pro prodejce pak např. produkt, SKU<sup>4</sup>, prodeje, prodejce apod. „Každý typ společnosti má svůj vlastní jedinečný soubor předmětu podnikání.“ (Inmon, c2005, s. 29)

Druhou a zároveň nejdůležitější charakteristikou datového skladu je jeho integrace. Dle Lacka (2003, s. 49) datový sklad musí být jednotný a integrovaný. Data, která se ukládají do datového skladu, mají různý počátek. Ve většině případů pocházejí z mnoha nejrůznějších podnikových provozních systémů, ale jejich původ může být i z externích zdrojů (např. kurzy měn z kapitálových trhů). Proto musejí být během plnění převedena, vyčištěna a sumarizována do jednotné podoby, např. všechny měny jsou převedeny na EUR. (Inmon, c2005 s. 30-31)

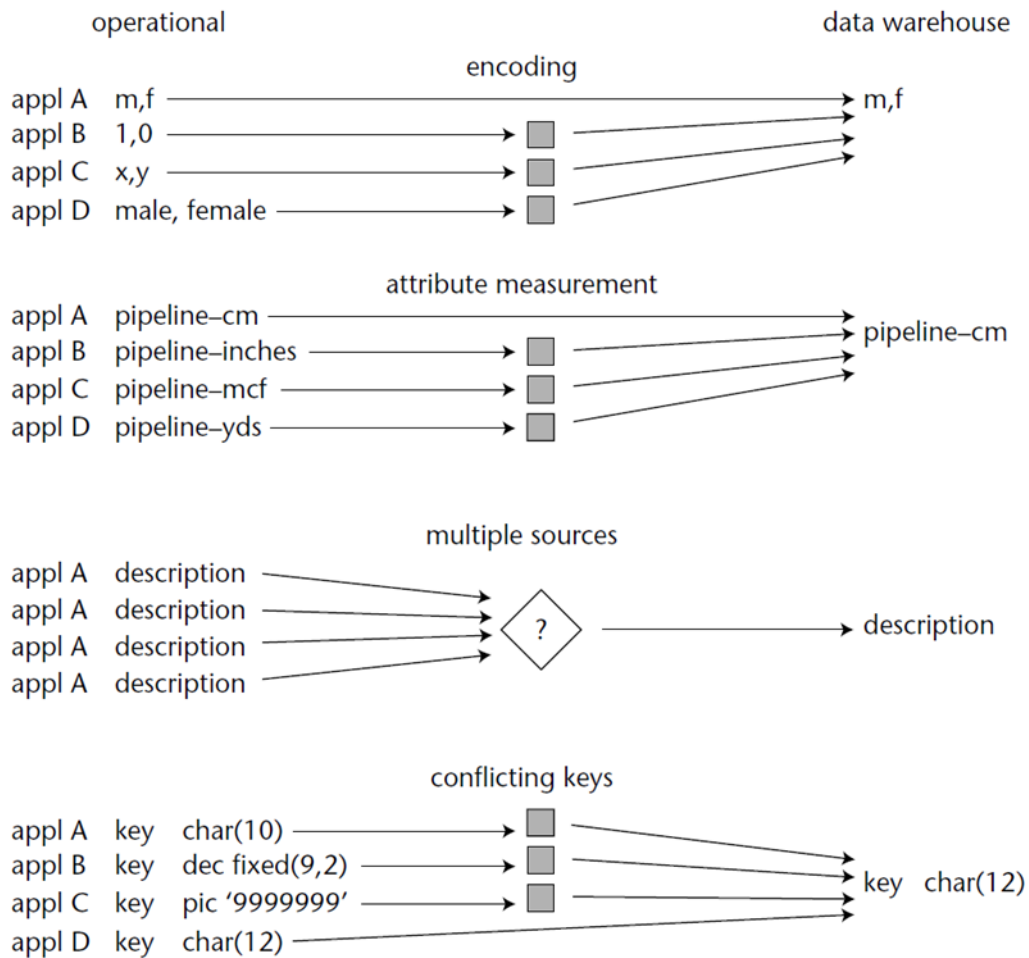
„Tato úvaha soudržnosti platí pro všechna navržená použití, jako jsou například konvence pojmenování, klíčové struktury, vlastnosti měření nebo fyzické vlastnosti dat.“ (Inmon, c2005 s. 30-31)

Proces převodu dat z provozních systémů do datového skladu znázorňuje obrázek č. 6.

---

<sup>4</sup> SKU (Stock Keeping Unit), je zkratka užívaná ve spojitosti se skladováním a prodejem.

## INTEGRATION

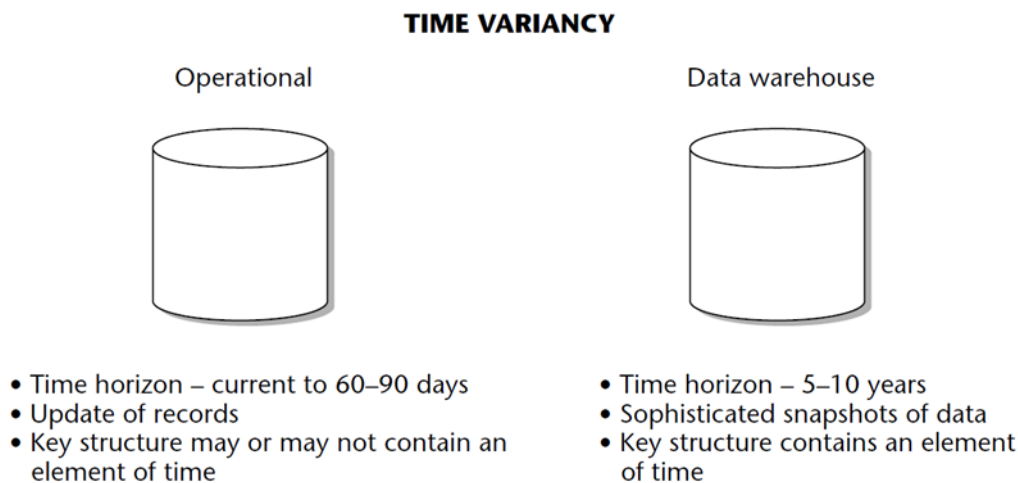


Obrázek č. 6 Problematika integrace (zdroj: Inmon, c2005, s. 31).

Na rozdíl od provozních databázových systémů, ve kterých lze uložená data následně editovat nebo mazat, data nahraná v datovém skladu nemohou být již nijak měněna. Data se do datového skladu neukládají v reálném čase, ale po dávkách jako série časových snímků v předem stanovených intervalech, které mohou být den, týden, měsíc apod. V případě nějaké změny, je namísto modifikace uložených dat, vytvořen a uložen zcela nový snímek. To zajišťuje jejich neměnnost. (Inmon, c2005 s. 32)

Poslední charakteristikou datového skladu je trvalost uložených dat. Znamená to, že každá jednotka dat v datovém skladu je přesná k určitému okamžiku v čase. V některých případech je vyznačen čas záznamu dat. V ostatních případech pak datum uložení záznamu transakce do datového skladu, ale vždy je zde uvedena nějaká časová značka, která určuje okamžik v čase, během něhož jsou data ve snímku přesná. (Inmon, c2005 s. 32)

Zatímco v provozních databázových systémech se data ukládají za kratší časové období (dny, měsíce), v datovém skladě jsou data ukládána za mnohem delší období, řádově několik let.



Obrázek č. 7 Trvalost uložených dat (zdroj: Inmon, c2005, s. 32).

Ve své definici Inmon zmiňuje pojem granulórní data. Granularitu (zrnitost) pak popisuje jako úroveň detailu nebo sumarizaci dat v datovém skladu a považuje ji za jeden z nejdůležitějších aspektů při navrhování datového skladu (Inmon, c2005 s. 41).

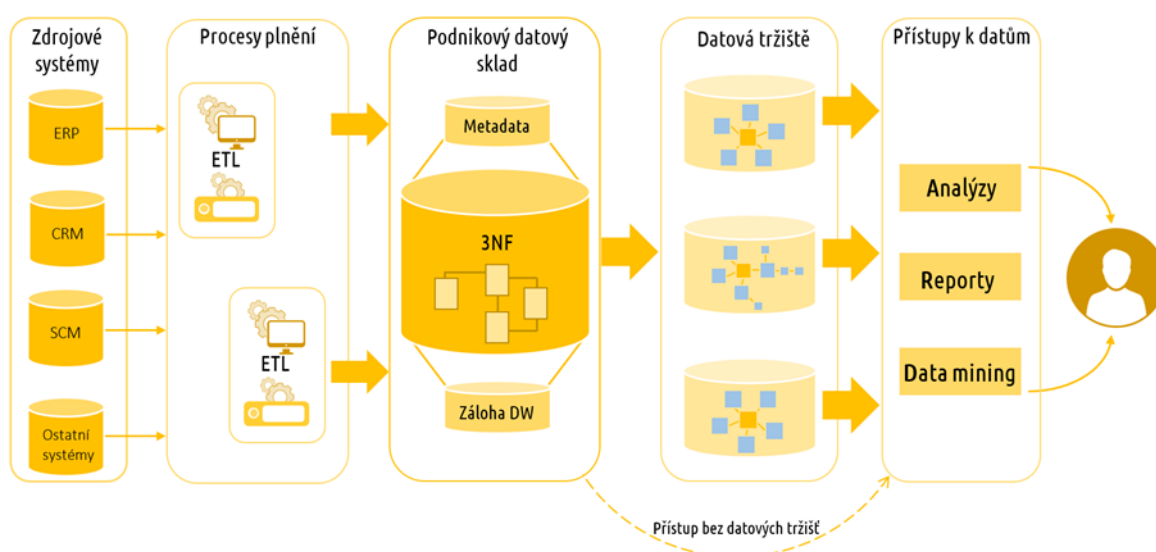
„Čím více je detailu, tím nižší je úroveň granularity. Čím méně je detailu, tím vyšší je úroveň granularity. Například, podrobný výpis každého telefonátu provedeného zákazníkem za jeden měsíc je na nízké úrovni granularity, souhrn telefonních hovorů uskutečněných zákazníkem po dobu jednoho měsíce je naopak na vysoké úrovni granularity.“ (Inmon, c2005 s. 41)

Inmon se domnívá, že obsah v datovém skladu musí být na co nejvyšší možné úrovni granularity, a musí obsahovat všechna možná historická data podniku. (Anupindi, 2005)

Přístup Williama H. Inmona k návrhu a následnému vývoji datového skladu je odlišný od přístupu Ralpa Kimballa. Inmon zastává názor k návrhu datového skladu metodou „shora dolů“. Tato metoda spočívá ve vytvoření nejprve centrálního datového úložiště, které uchovává data v čase spolu s jejich změnami ve strukturované normalizované formě pro celý podnik. (Anupindi, 2005)

„Je přitom nutné vytvořit takový formát struktury dat, který zajistí její pružnost a umožní růst, když do datového skladu budou další projekty přidávat nové položky.“ (Laberge, 2012, s. 232)

Přístup k datům pro libovolného podnikové uživatele je pak obvykle zajištěn pomocí satelitních databází, které slouží pro podporu rozhodovacích procesů jednotlivých útvarů podniku. Tyto databáze označujeme jako data marts nebo také datová tržiště. (Anupindi, 2005)



Obrázek č. 8 Architektura centrálního datového skladu dle Williama H. Inmona (upraveno podle: <https://kurtzahra.com/2015/06/20/inmon-or-kimball-the-debate-of-the-century/>)

Celkový princip je vcelku jednoduchý. Data jsou zachytávána z různých zdrojových systémů (ERP, CRM a další) nebo úložišť provozních dat, pomocí rutin ETL vrstvy jsou převedena do potřebného formátu a sloučena s úložištěm datového skladu, které je navrženo normalizovaným způsobem. Pro specifické podnikové použití jsou konkrétní data extrahována a distribuují se do požadovaných datových trhů. Datové trhy mohou být navrženy normalizovaným způsobem, ale nemusejí. Záleží to na tom, který způsob je pro podnikové použití vhodnější. (Laberge, 2012, s. 233)

Oba autoři se také neshodují v otázce pohledu na pojetí struktury uložených dat. Inmon navrhuje, aby vytvořený centrální datový sklad byl v normalizovaném datovém modelu a data v něm obsažená měla velmi vysokou úroveň granularity.

Z definice datového skladu a architektury podle Williama H. Inmona můžeme vyvodit jeho výhody i nevýhody. Pravděpodobně největší výhodou může být relativně rychlé a jednoduché vytvoření jednotlivých datových tržišť, a z důvodu jejich generování z centralizovaného datového skladu zajištění i jejich konzistentnosti. Další výhodou nahrávání dat do centrálního úložiště, z kterého jsou poté data distribuovány do jednotlivých datových tržišť, je zajištění vyšší kvality těchto dat nebo možnost přístupu k celopodnikovým datům. Díky menšímu počtu rozhraní mezi systémy zdrojů dat a cílovým úložištěm se zmenšuje prostor pro výskyt chyb a redundancí.

Nevýhoda tohoto přístupu spočívá v potřebě zajištění pružného a škálovatelného návrhu úložiště. V případě, kdy je nutné změnit návrh úložiště, kvůli jeho nepružnosti, tak aby vyhovovalo minulým i novým požadavkům, pak takováto aktivita ovlivní všechny předchozí návrhy, ETL rutiny, datová tržiště, distribuční metody a další. (Laberge, 2012, s. 234)

Nevýhodou může být také složitější, náročnější, finančně nákladnější a z časového hlediska déle trvající vytvoření centrálního datového skladu, které má za následek pozdější přínos pro uživatele. Dle Laberge (2012, s. 234) je optimální přístup u této metody založen na nákupu a přizpůsobení již existující logického a aplikačního datového modelu datového skladu, který je vyvíjen speciálně pro určité obory činnosti podniku.

### **1.2.3 Architektura a charakteristika datového skladu podle Ralpa Kimballa**

Druhá nejdůležitější osoba v oblasti datových skladů je bezesporu Ralph Kimball, který jako první popsal využití dimenzionálního modelování a konceptualizoval datová tržiště včetně „star“ a „snowflake“<sup>5</sup> datové struktury. (Anupindi, 2005)

Ralph Kimball ve své knize popsal datový sklad takto: „Datový sklad je místo, kde jsou provozní data specificky strukturovaná pro dotazování a analýzy výkonnosti a jsou snadno použitelná.“ (Kimball a Ross, c2002, s. 397)

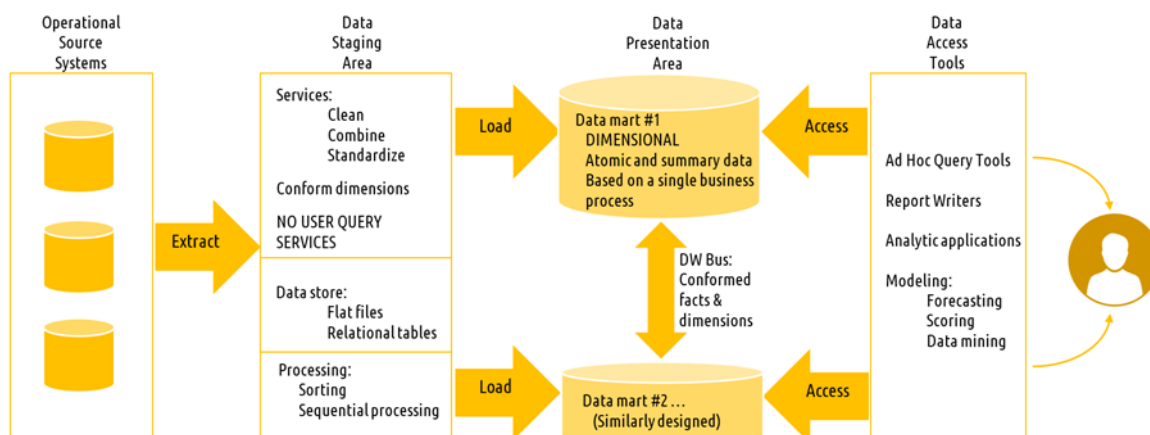
Přístup Ralpa Kimballa k implementaci datových skladů se zaměřuje na podnikovou analytiku a orientuje se na konečné výsledky, jinak se také nazývá jako přístup „orientovaný na datová tržiště“ nebo přístup „zdola nahoru“. (Laberge, 2012, s. 234)

---

<sup>5</sup> Obě struktury jsou blíže vysvětleny v dalších kapitolách. V českém jazyce mluvíme o schématech hvězdy a sněhové vločky.

I přesto, že definice od Ralpha Kimballa je daleko jednodušší a srozumitelnější než definice dle Inmona, vyžaduje podrobnější vysvětlení. Toho lze docílit analýzou základních součástí přístupu k implementaci datového skladu dle Kimballa.

Jednotlivé samostatné části datového skladu, které jsou zobrazeny na obrázku č. 9, by měly být alespoň čtyři. Jsou to provozní systémy, data staging area (pracovní oblast), data presentation area (oblast prezentace dat) a nástroje pro přístup k datům.



Obrázek č. 9 Základní elementy datového skladu (zdroj: Kimball a Ross, c2002, s. 7)

Provozní systémy zaznamenávají jednotlivé transakce a operace uživatelů. Je třeba mít na paměti, že tyto systémy stojí prakticky mimo datový sklad, protože obsah a formát dat v nich lze kontrolovat jen velmi málo nebo vůbec. Hlavní priority provozních systémů je jejich dostupnost a výkonnost. Z důvodu nízké optimalizace pro sdílení dat mezi provozními a operačními systémy, je vývoj datového skladu značně ztížen. (Kimball a Ross, c2002, s. 7)

Data staging area je prakticky vše mezi zdrojovými provozními systémy a data presentation areou. Kimball (c2002, s 8) přirovnává data staging areu ke kuchyni v restauraci, kde je ze syrových potravin připravováno chutné jídlo, kam mají přístup jen kvalifikovaní pracovníci a zákazníci nejsou zasvěceni do toho, co se v kuchyni děje. Podobně v data staging area hrubá provozní data podstoupí tzv. Extract – Transform – Load (ETL) procesy, aby byla k dispozici pro potřeby a dotazy uživatele, přistupovat sem mohou jen zkušení odborníci a uživatel nemusí vědět, jak tato oblast pracuje (Kimball a Ross, c2002, s. 8).

Extrakce je prvním krokem v procesu získávání dat do prostředí datového skladu. V mnoha případech jsou extrahovaná data z několika různých provozních systémů nesourodá. Extrahováním je myšleno porozumění a čtení zdrojových dat a jejich kopírování do data staging area za účelem další manipulace. (Kimball a Ross, c2002, s. 8)

Jakmile jsou data extrahována do data staging area, prochází četnými transformacemi, jako je jejich očištění (oprava překlepů a pravopisných chyb, doplnění chybějících elementů, převod do standartního formátu), kombinování z různých zdrojů, de-duplikace a přidělení databázových klíčů. (Kimball a Ross, c2002, s. 8)

Poslední krokem ETL rutin je nahrání dat do data presentation area. Nahrání dat do datového skladu, popř. datového tržiště, má obvykle podobu dimenzionálních tabulek se zaručenou kvalitou prezentovaných dat. Cílové datové tržiště pak musí indexovat nově příchozí data pro budoucí použití v dotazech. Při každém novém nahrání dat datové tržiště informuje uživatele o zveřejnění těchto dat spolu s uvedením změn, ke kterým došlo. (Kimball a Ross, c2002, s. 9–10)

Data presentation area je místo, kde jsou data organizována, ukládána, a jsou k dispozici pro přímé dotazování, reporty a další analytické procesy ze strany uživatelů. Tato oblast se většinou skládá z řady integrovaných datových tržišť. (Kimball a Ross, c2002, s. 10)

Jednou ze základních podmínek této oblasti je, aby data byla prezentována, skladována a zpřístupněna v dimenzionálním modelu. Dimenzionální schéma obsahuje stejné informace jako normalizovaný model, ale v podobě, která je srozumitelnější pro uživatele, vhodnější pro dotazování a odolnější proti změnám. (Kimball a Ross, c2002, s. 10–12)

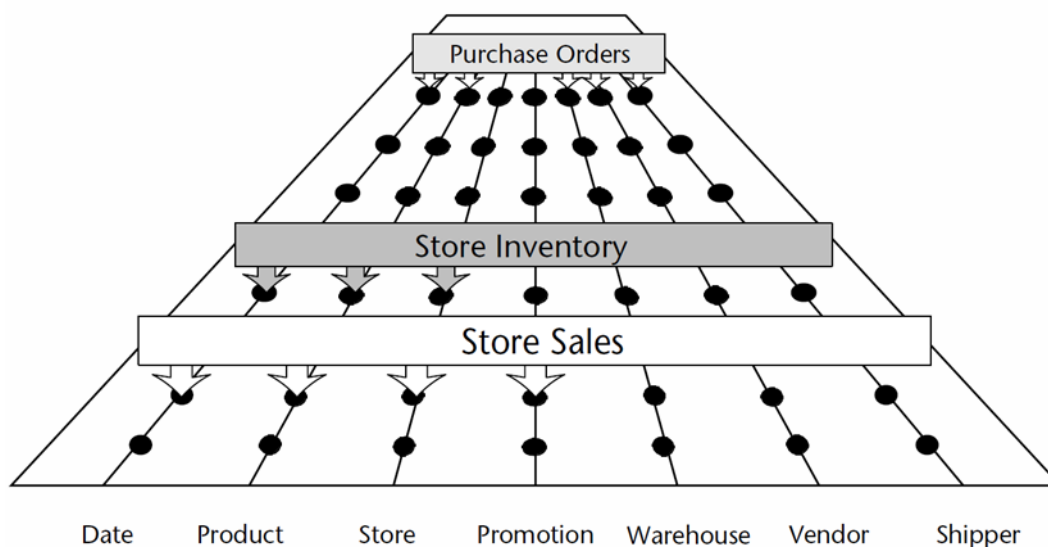
Další podmínkou data presentation area je nutnost, aby datová tržiště obsahovala detailní, tedy atomická data. Atomická data jsou zde nezbytná z důvodu odolnosti datového tržiště proti intenzitě nepředvídatelného uživatelského dotazování, ačkoli mohou také obsahovat, kvůli zvýšení výkonu, data souhrnná nebo agregovaná. (Kimball a Ross, c2002, s. 12)

Všechna datová tržiště musejí být vytvořena za použití společných dimenzí a faktů<sup>6</sup>. Kimball toto pravidlo nazývá „conformed“, díky kterému mezi sebou jednotlivá datová tržiště komunikují a sdílejí data, a které je i základem sběrnice datového skladu. Data presentation area ve velkém datovém skladu firmy se může skládat i z více než dvaceti velmi podobně vypadajících datových tržišť. Bez sdílení, přizpůsobení se dimenzím

---

<sup>6</sup> Oba pojmy jsou vysvětleny v kapitole zabývající se dimenzionálním přístupem

a faktům, se datové tržiště stává samostatnou aplikací. Izolovaná datová tržiště, která nelze dohromady propojit jsou zhoubou pro další rozvoj datového skladu. Z tohoto důvodu je integrace pomocí sběrnice datového skladu nezbytná a na stejném principu je i založen přístup Ralpha Kimballa k vývoji datového skladu. (Kimball a Ross, c2002, s. 12)



Obrázek č. 10 Sběrnice podnikového datového skladu (zdroj: Kimball a Ross, c2002, s. 78)

Definováním standartního rozhraní sběrnice pro prostředí datových skladů, mohou být separovaná datová tržiště spojena do různých skupin v různém čase. Jednotlivá datová tržiště tak lze připojit k sobě jako kousky puzzle a prospěšným způsobem pak mohou spolu koexistovat, pokud vyhovují definovanému standartu. (Kimball a Ross, c2002, s. 78)

„Kimball věří ve vytvoření několika menších datových tržišť pro dosažení analýz a reportů využitelných jednotlivými odděleními firmy.“ (Anupindi, 2005)

„Nechť všichni vytvoří, co chtějí a kdy to to chtějí, my to vše spojíme, když to budeme potřebovat a jestli to budeme potřebovat.“ (Reynolds, 2006)

Poslední z hlavních komponent datového skladu dle Kimballa jsou nástroje pro přístup k datům. Termín „nástroj“ zde odkazuje na paletu možností, které mohou být poskytnuty koncovým uživatelům k využití data presentation arey pro analytickou podporu rozhodování. Dle definice, jsou všechny přístupové nástroje pro dotazování na data datového skladu v data presentation area. Dotazování je samozřejmě hlavním důvodem využívání datového skladu. Nástroj pro přístup k datům tak může být například jednoduchý dotaz, stejně jako aplikace pro modelování nebo sofistikovaný data mining. (Kimball a Ross, c2002, s. 13)



Mimo hlavní komponenty, které jsou v datovém skladu podle Kimballa nutností, lze jako jeho další součásti označit i tzv. metadata a úložiště provozních dat.

Metadata obecně jsou data, která popisují data. Jako příklad můžeme uvést fotografie uložené v počítači, které obsahují také informace o tom, jakým fotoaparátem byly pořízeny, nastavení ISO objektivu, rozlišení fotografií nebo kde a kdy byly vytvořeny. Tyto informace jsou metadata.

V prostředí datových skladů metadata obsahují definici data (význam a zdroj každého sloupce), definici samotného datového skladu (jinými slovy, strukturu úložiště, ETL procesů a kvality dat), definice souvisejících systémů (například zdrojové systémy), informace pro audit (jaké procesy a kdy probíhaly) a využití (které reporty a OLAP kostky jsou používány a kdy). (Rainardi, c2008, s. 301)

Úložiště provozních dat (ODS) je provozní systém fungující téměř v reálném čase. V praxi se jedná o samostatný systém, který se nachází mimo hranice datového skladu. S datovým skladem je však ODS úzce spjato, protože data staging area poskytuje data úložišti provozních dat, které pak slouží jako přímý zdroj datového skladu. (Laberge, 2012, s. 251)

ODS je často aktualizované a integruje kopie provozních dat. Frekvence aktualizace a míra integrace ODS se liší v závislosti na konkrétních požadavcích. Nejčastěji se úložiště provozních dat využívá k provoznímu reportování, zvláště když starší ani novější a modernější transakční databázové systémy (OLTP<sup>7</sup>) toto neposkytují. V ostatních případech je ODS vytvořeno pro podporu interakce v reálném čase, a to zejména v oblasti řízení vztahu se zákazníkem (CRM<sup>8</sup>). (Kimball a Ross, c2002, s. 15)

„Úložiště pracovních dat obvykle uchovává data, která jsou téměř aktuální. To znamená, že není zachována historie, ačkoli určitá data o historii mohou existovat ve formě souhrnů nebo tabulek profilů.“ (Laberge, 2012, s. 251)

Pro systém datového skladu je úložiště provozních dat primárním zdrojem většiny dat, i když obvykle získává data i z jiných zdrojových systémů. ODS tak lze použít nezávisle na datovém skladu, ve většině případů se ale implementují společně. (Laberge, 2012, s. 224)

---

<sup>7</sup> Online Transaction Processing – blíže jsou tyto systémy popsány v kapitole Systémy OLTP

<sup>8</sup> zkratka z anglického Customer Relationship Management

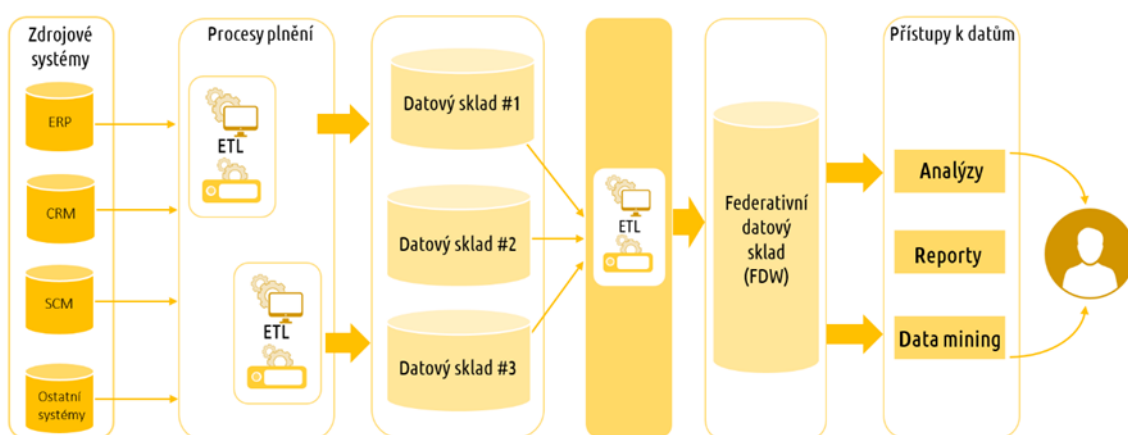
Implementace datového skladu podle Kimballa doporučuje nejprve vytvořit několik datových tržišť pro jednotlivá oddělení podniku a ty následně za pomoci již zmíněné sběrnice DW spojit. Případně lze k vytvoření jednoho datového skladu sloučit vybraná datová tržiště dohromady. Tato architektura tak nabízí dobrý kompromis mezi centralizovaným přístupem Billa Inmona a architekturou samostatných datových tržišť.

Za největší výhodu u tohoto přístupu je možno považovat rychlý a méně nákladný vývoj, díky kterému jsou požadované výsledky mnohem dříve k dispozici než u implementace centrálního datového skladu B. Inmona.

Největší nevýhodou pak je větší počet rozhraní mezi zdrojovými systémy a datovými trhy, složitější integrace datových tržišť nebo vyšší nároky na údržbu a správu datového skladu. Protože jsou data v jednotlivých datových tržištích přímo odvozena ze zdrojových systémů a nikoli z centrálního datového úložiště, může v nich také docházet ke vzniku redundance dat.

#### 1.2.4 Federativní architektura

Federativní architektura klade hlavní důraz na přístupy k datům (analýzy, dolování dat a další), jejichž zdrojem jsou již existující datové sklady nebo datová tržiště. Na obrázku č. 11 sice nejsou datová tržiště zobrazena, ale pokud by tam byla, byla by ve stejné oblasti jako DW.



Obrázek č. 11 Federativní architektura (upraveno podle: Laberge, 2012, s. 227)

Federovaný datový sklad získává data z existujících datových skladů nebo tržišť za použití ETL rutin a ukládá je do nového dimenzionálního úložiště dat. Nastavení frekvence spouštění ETL rutin je důležité z důvodu časově rozdílného nahrávání dat do zdrojových

datových skladů. Zároveň je třeba zjistit, zda data nahrávaná z jednoho zdrojového datového skladu nejsou duplikáty dat z jiného zdrojového datového skladu. V případě duplicity jsou tato data sloučena. (Rainardi, c2008, s. 40)

Granularita dat ve FDW má stejně vysokou úroveň jaká je nejvyšší úroveň v integrovaných zdrojových datových skladech (např. sklad #1 má granularitu G1, úroveň obsahující dny, sklad #2 granularitu G2, úroveň týdnu a sklad #3 granularitu G3, úroveň měsíců, pak výsledný federativní datový sklad bude mít granularitu G3, protože agregací lze ze dnů a týdnů vytvořit měsíce). (Rainardi, c2008, s. 39)

V případech používání samostatných systémů a následné potřeby integrace podniků nebo slučování dvou oddělení se za využití této architektury může přistupovat k federativnímu datovému skladu při minimálních nárocích na stávající provozní systémy. „Udržování federativní architektury více systémů však může komplikovat správu a zvyšovat náklady.“ (Laberge, 2012, s. 227)

### **1.3 Neúspěšné řešení datového skladu**

Virtual Case File byl aplikační software vyvíjený Federálním úřadem pro vyšetřování (FBI) ve Spojených státech mezi roky 2000 a 2005. Tento projekt byl oficiálně zrušen ještě ve fázi vývoje v lednu roku 2005 a federální vláda za něj do té doby utratila téměř 170 milionů dolarů. Eggen (Eggen a Witte, 2006) ve svém článku uvádí, že software byl neúplný, nedostatečný, a tak špatně navržený, že by bylo v podstatě nemožné ho použít v reálných podmínkách. Dokonce i v primitivních testech, systém nesplňoval základní požadavky.

Mezi zásadní nedostatky tohoto systému, které vedly k jeho selhání, patřila od začátku špatně plánovaná architektonická rozhodnutí, opakované změny ve specifikaci, průběžně přidávané požadavky do systému a tím narůstající zdrojový kód do extrémních rozměrů (přes 700 000 řádků) nebo dokonce nemožnost třídění dat. (Eggen a Witte, 2006)

V březnu roku 2005 FBI oznámila začátek nového, ambicióznějšího softwarového projektu s kódovým označením Sentinel, který měl nahradit Virtual Case File. Po několika zpožděních, změně vedení, navýšení základního 425 milionového rozpočtu, byl tento projekt dokončen a začal být agenturou používán od 1. července 2012. (Foley, 2012)

## 2 Systémy OLTP

Systémy OLTP (Online Transaction Processing), transakční databázové systémy mají skoro neomezenou oblast využití. Jejich primárním cílem je umožnit uživatelům databázového serveru provádění velkého množství transakcí online spolu se zautomatizováním každodenních činností podnikání, jako jsou mzdy, nákupy a prodeje nebo skladové hospodářství.

Pro svoje výhody, ale i z důvodu existence množství specialistů, jsou transakční systémy velmi oblíbené a často používané v podnikové praxi. „V případě, že transakční databázový systém pokrývá většinu firemních aktivit, nazýváme ho systém ERP (Enterprise Resource Planning). Ke zdroji údajů ve stejném čase přistupuje velké množství uživatelů, kteří z databáze čtou, jiní do něj zapisují, případně někteří vykonávají i jednodušší analýzy.“ (Lacko, 2003, s. 20)

Transakční databáze, někdy označované i jako databáze OLTP, jsou určeny především pro ukládání operačních údajů a z důvodu jednoduchého dotazování a vyloučení možné redundance zpravidla normalizovány, tzn., že transakční databáze vyhovují pravidlům tzv. normálních forem (Lacko, 2003, s. 22). To znamená mnoho atomických a relačně svázaných tabulek.

Ve většině případů je struktura operačních údajů v databázích OLTP vysoce strukturovaná, díky tomu tyto systémy dosahují při online transakcích vysokých výkonů. Při komplexních analýzách, vyžadující značné nároky na výpočetní kapacitu procesorů nebo při získání informací pro podporu rozhodování nejsou databáze OLTP příliš vhodné. To je zapříčiněno především tím, že pro dosažení výsledku dotazu musí být spojeno velké množství tabulek, což je méně efektivní než načítat data pouze z jedné i když mnohem obsáhlejší tabulky.

### 2.1 Normalizovaný přístup k ukládání dat

Otec databázové teorie E. F. Codd definoval v roce 1972 tři úrovně normalizace databází. 1NF až 3NF, které jsou založené na funkčních závislostech mezi atributy relace. Boyce-Coddovu normální formu, která byla vytvořena, aby odstranila některé anomálie vyskytující se v relacích ve 3NF, a 4NF a 5NF, založené na vícehodnotových závislostech a join závislostech. (Lacko, 2011, s. 89)

„Normalizace je přístup, který odstraňuje redundance v datech přesouváním opakujících se atributů do vlastních tabulek.“ (Mundy, Thornthwaite a Kimball, c2011, s. 34)

### **2.1.1 První normální forma (1NF)**

Tabulka splňuje podmínku první normální formy tehdy, když jsou všechny hodnoty v attributech (sloupcích) atomické, tedy hodnoty z pohledu databáze dále nedělitelné. Jeden sloupec nesmí obsahovat více druhů údajů, nebo řečeno matematickou terminologií, musí obsahovat skalární<sup>9</sup> hodnotu. (Lacko, 2011, s. 90)

### **2.1.2 Druhá normální forma (2NF)**

„Relace se nachází v druhé normální formě, jestliže je v první normální formě a každý neklíčový atribut je plně závislý na primárním klíči, a to na celém klíči, a nejen na nějaké jeho podmnožině.“ (Teorie relačních databází: Normalizace, 2007) Druhá normální forma tedy platí jen pro relace, které obsahují více primárních klíčů. Pokud relace obsahuje pouze jeden primární klíč, automaticky splňuje podmínku pro 2NF (Lacko, 2011, s. 92).

### **2.1.3 Třetí normální forma (3NF)**

Tabulka splňuje podmínky třetí normální formu tehdy, jestliže je v druhé normální formě a zároveň neexistují závislosti neklíčových sloupců relace. (Lacko, 2011, s. 92)

### **2.1.4 Boyce-Coddova normální forma**

Boyce-Coddova normální forma se nachází mezi 3NF a 4NF. Tabulka je v BCNF tehdy, pokud je ve 3NF a žádný samostatný atribut nezávisí na ničem jiném než na nadklíči, což jednoznačně identifikuje každý řádek (Rainardi, c2008, s. 506). Boyce-Coddova normální forma je o něco silnější než 3NF.

### **2.1.5 Čtvrtá normální forma (4NF)**

„Tabulka je ve čtvrté normální formě tehdy, je-li ve BCNF a popisuje jen jeden fakt nebo pouze příčinnou souvislost.“ (Lacko, 2011, s. 93)

### **2.1.6 Pátá normální forma (5NF)**

Tabulka je v páté normální formě tehdy, pokud je ve čtvrté normální formě a není možné do ní přidat nový sloupec, případně skupinu sloupců, aniž by se vlivem skrytých závislostí rozpadla na několik dílčích tabulek. (Lacko, 2011, s. 93)

---

<sup>9</sup> V matematice skalár (z lat. scala, stupnice) označuje zpravidla jediné reálné či komplexní číslo.

### 3 Dimenzionální přístup k ukládání dat

Oproti systémům OLTP, které používají normalizované datové struktury, ale nejsou primárně optimalizované pro podporu rozhodování, v mnoha případech vyžadují znalost jazyka SQL a pro koncového uživatele nejsou příliš pochopitelné, dimenzionální modelování databází klade důraz na srozumitelnost pro uživatele, je optimalizované pro vyhledávání dat, složité analýzy a pro podporu rozhodování. „Takto vytvořené databáze slouží jako podklad pro získání sumarizovaných a agregovaných údajů, tedy vlastně informací.“ (Lacko, 2003, s. 31)

„Dimenzionální databáze je denormalizovaná databáze obsahující tabulky faktů a tabulky dimenzí.“ (Rainardi, c2008, s. 30)

Tabulka nebo tabulky faktové jsou vždy centrální a k nim jsou přidružené tabulky dimenzí. Každá faktová tabulka by dle Kimballa (Kimball a Ross, c2002, s. 16-21) měla být v normalizovaném (zpravidla ve 3NF) stavu a tabulka dimenzí ve stavu denormalizovaném.

„Denormalizovaná databáze je databáze s některými redundantními daty, které ještě neprošly procesem normalizace.“ (Rainardi, c2008, s. 30)

Multidimenzionální databáze (MDD) se obvykle používají pro business intelligence, zejména pro Online Analytical Processing (OLAP) a data mining. Výhody použití multidimenzionálních databází pro OLAP a data mining namísto relačních databází jsou, že používají méně místa na disku a jejich výkon je vyšší. (Rainardi, c2008, s. 378)

#### 3.1 Tabulka faktů

Tabulka faktů je primární tabulkou v dimenzionálním modelu, v níž jsou uložena analyzovaná a sumarizovaná data podniku s cílem získat lepší pochopení svého podnikání. Jeden řádek tabulky vyjadřuje určitou míru nebo hodnotu, proto jsou tabulky faktů nejobtavnější tabulky v dimenzionální databázi a obsahují velké objemy dat. Se svým malým počtem sloupců, ale velkým množstvím řádků, tyto tabulky zabírají až 90 % z celkové velikosti dimenzionální databáze. Kromě numerických měrných jednotek musí tyto tabulky obsahovat minimálně dva cizí klíče, které je spojují s primárními klíči tabulek dimenzí. Tabulka faktů má sama svůj vlastní primární klíč, který tvoří podmnožinu cizích klíčů. Tento klíč se často nazývá „composite“ nebo „concatenated“. Každá tabulka faktů

v dimenzionálním modelu obsahuje composite klíč, a naopak, každá tabulka, která obsahuje composite klíč je tabulka faktů. (Kimball a Ross, c2002, s. 16-18)

„Měrné jednotky se mohou uložit v tabulce faktů, případně vyvolat, když je to nevyhnutelné, na účely vykazování.“ (Lacko, 2003, s. 113)

Velký význam má v dimenzionálním modelu granularita a aditivita faktů. Pokud je zvolena příliš nízká úroveň granularity, nebude možné pracovat s detailními daty, v opačném případě dochází k vysokému nárůstu dat, manipulace s daty bude časově náročnější a požadavky na dostatečný diskový prostor tedy budou mnohem vyšší. Tabulky faktů by měly být dle Kimballa (Mundy, Thornthwaite a Kimball, 2006, s. 42) navrženy na co nejnižší úrovni detailu (nejvyšší úroveň granularity) která je možná tzn. s použitím atomických dat.

Aditivita faktů je vlastnost, které určuje, zda je možné sečíst pole faktů napříč dimenzemi. Aditivní fakta je možné sumarizovat napříč všemi dimenzemi, poloaditivní nebo také semiaditivní lze agregovat pouze podle některých dimenzí a neaditivní, která sečíst nelze. (IBM Knowledge Center, 2015)

### **3.2 Tabulka dimenzí**

Tabulky dimenzí obsahují logicky nebo organizačně hierarchicky uspořádané popisy podniku. Oproti tabulkám faktů jsou tabulky dimenzí většinou menší a data v nich se nemění tak často. Atributy tabulek dimenzí mají v systému datového skladu nepostradatelnou roli, protože slouží jako hlavní zdroj dotazů a reportů a jsou i klíčovým prvkem pro vytvoření použitelného a srozumitelného datového skladu. V dobře navrženém dimenzionálním modelu mají jednotlivé tabulky dimenzí velký počet atributů. Není neobvyklé, když tyto tabulky obsahují až 100 sloupců. I přesto jsou ale poměrně malé a neobsazují více než 10 % celkové kapacity úložiště datového skladu. Dimenzionální atributy jsou používány aplikacemi business intelligence pro filtrování a seskupování faktů. (Kimball a Ross, c2002, s. 19)

„Síla datového skladu je přímo úměrná kvalitě a hloubce atributů tabulek dimenzí.“ (Kimball a Ross, c2002, s. 20)

„Tabulky dimenzí vysvětlují všechna „proč“ a „jak“, pokud jde o obchodování a transakce prvků. Dimenze obecně obsahují relativně stabilní data, dimenze zákazníku se aktualizují

častěji. Velmi často se používají časové, produktové a geografické dimenze.“ (Lacko, 2003, s. 113)

Jednoduchost dimenzionálního modelu přináší také výkonnostní benefity. Databáze mohou tato schémata procházet daleko efektivněji díky nižší potřebě spojovat jednotlivé tabulky. Uživatelské dotazy tak v porovnání s normalizovaným přístupem trvají daleko kratší dobu. (Kimball a Ross, c2002, s. 22)

Tabulky dimenzí obsahují obvykle stromovou strukturu, například dimenze vytvořená na základě času se člení na jednotlivé úrovně podle kalendářních nebo fiskálních zvyklostí (rok, kvartál, měsíc, týden, den apod.).

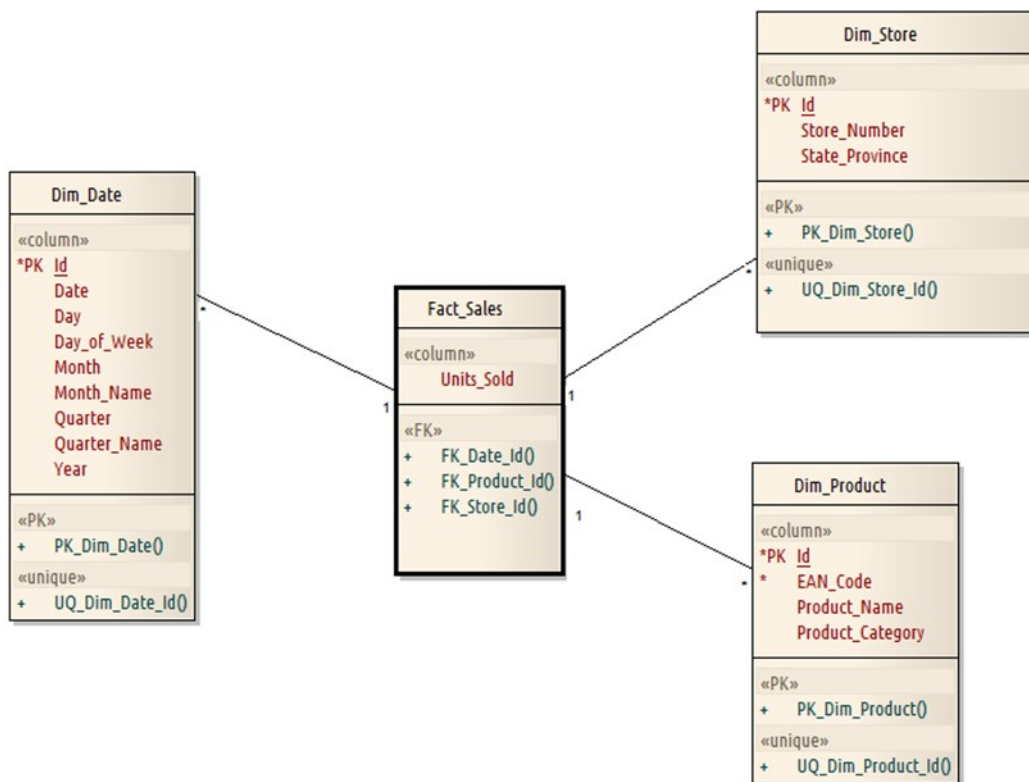
Tabulky faktů a dimenzí svým spojením v dimenzionálním modelu, který má dané topologické uspořádání, vytvářejí určitá schémata. Nejběžnější je schéma hvězdy (star schema) a schéma sněhové vločky (snowflake schema). V prvním schématu jsou v tabulce dimenzí zahrnuty i všechny další podřízené úrovně hierarchie, u druhého je pro každou úroveň hierarchie vytvořena samostatná tabulka dimenzí, která je provázána na ostatní tabulky. Mezi dvěma takto vytvořenými sousedícími tabulkami je vždy použita kardinalita 1:N.

### **3.2.1 STAR schéma**

Schéma hvězdy je tvořeno většinou jednou rozsáhlou centrální tabulkou faktů spojenou s tabulkami dimenzí. Velký důraz je kladen na to, aby každý cizí klíč v tabulce faktů měl svůj unikátní primárním klíč v příslušné tabulce dimenzí. Pro každou dimenzi je vytvořena právě jedna samostatná tabulka s několika atributy.

„Hvězdicové schéma nemá normalizované dimenze ani relační propojení mezi tabulkami dimenzí, proto je velmi lehce pochopitelné a poskytuje vysoký „dotazovací výkon“. V důsledku nenormalizovaných dimenzí je však vytvoření takového modelu relativně pomalé.“ (Lacko, 2003, s. 114)





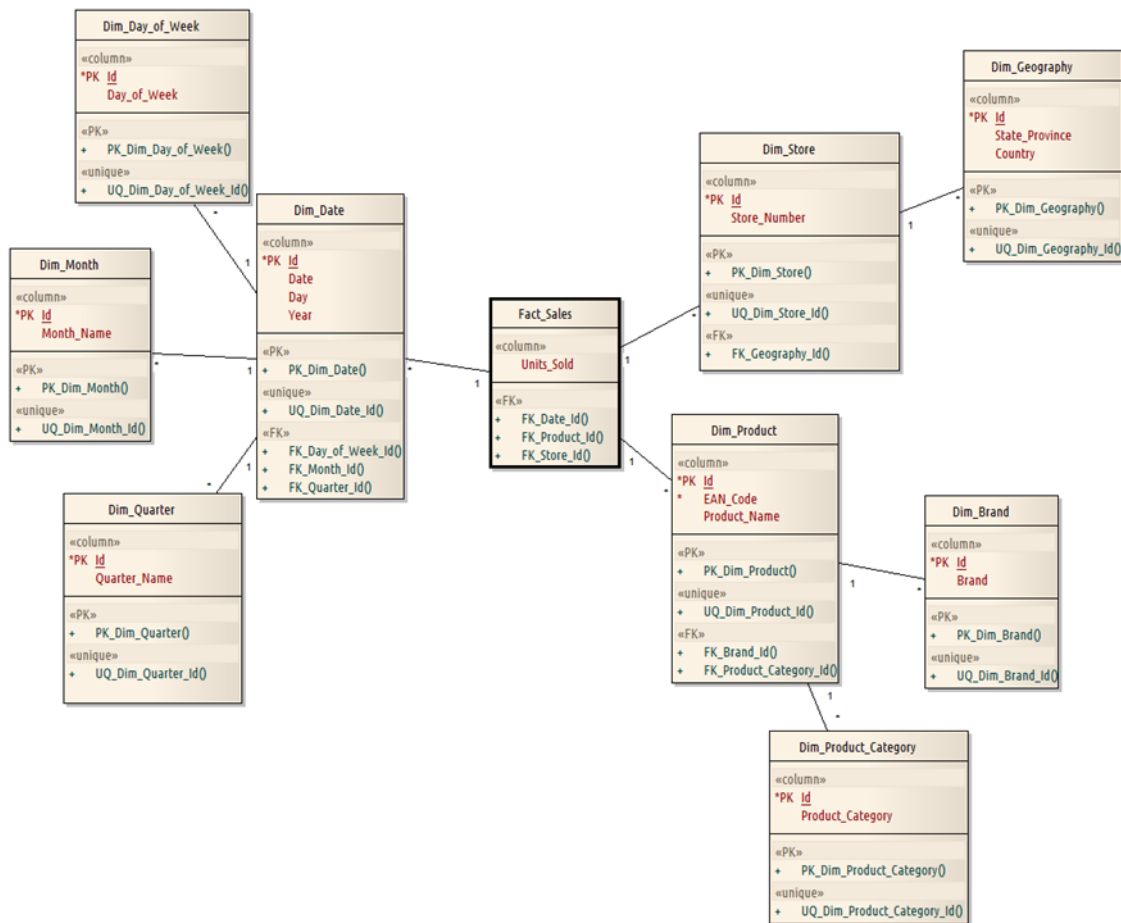
Obrázek č. 12 Schéma hvězdy (upraveno podle: [https://en.wikipedia.org/wiki/Star\\_schema](https://en.wikipedia.org/wiki/Star_schema))

Výhodou tohoto schématu je jeho lehké pochopení, rychlejší získání výsledků dotazování díky odpadnutí operací spojování tabulek (dostačující je spojení mezi tabulkou faktů a dimenzí) nebo jednodušší prohlížení dimenzí. Nevýhodou je jeho již zmíněné relativně pomalejší vytvoření, vyšší nárok na prostor a nemožnost agregace z důsledku denormalizace tabulek a atributů nebo nevhodnost použití při častých změnách v hierarchiích prvků dimenzí.

### 3.2.2 SNOWFLAKE schéma

Druhé nejběžnější je schéma sněhové vločky. Toto schéma tvoří centrální tabulka faktů a skupina tabulek dimenzí, které jsou normalizovány (rozděleny do dalších tabulek). Tím vznikají poddimenze již existujících dimenzí.

Ralph Kimball (Kimball a Ross, c2002, s. 21) uvádí, že vzhledem k tomu, že tabulky dimenzí jsou geometricky menší než tabulky faktů a z pohledu celkové velikosti dimenzionální databáze zabírají pouze malý zlomek prostoru, je v podstatě zbytečné přecházet na normalizované schéma. „Téměř vždy obětujeme prostor v dimenzionálním modelu pro jeho jednoduchost a uživatelskou dostupnost.“ (Kimball a Ross, c2002, s. 21)



Obrázek č. 13 Schéma sněhové vločky (upraveno podle:  
[https://en.wikipedia.org/wiki/Snowflake\\_schema](https://en.wikipedia.org/wiki/Snowflake_schema))

Schéma sněhové vločky je díky normalizovaným tabulkám dimenzí (3NF) vhodné pro časté změny v těchto tabulkách a v jejich hierarchické struktuře prvků. Výhodou je možnost vytvoření agregačních tabulek a menší nároky z pohledu diskového prostoru než schéma hvězdy. Nevýhoda tohoto schématu je nižší efektivnost analýz dat, jeho nepřehlednost a větší časová náročnost zapříčiněná potřebou provedení operací spojování jednotlivých tabulek dimenzí.

Na základě těchto dvou dimenzionálních modelů (schéma hvězdy a sněhové vločky) se vytváří vícerozměrná datová kostka tzv. OLAP kostka.

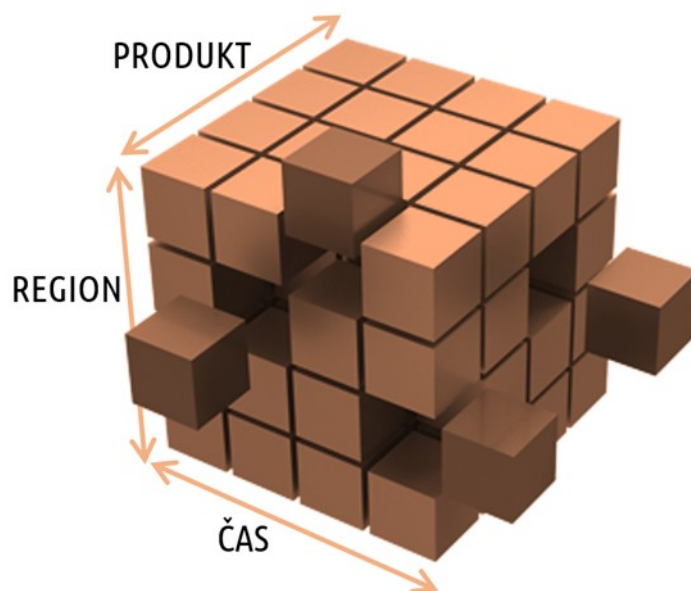
### 3.3 OLAP kostka

Součástí datového skladu může, ale ve všech případech to tak není, být i OLAP kostka, záleží pouze na zvolené architektuře a záměru datového skladu nebo na finančních možnostech. Datové sklady, které kostku neobsahují, mají menší nároky na údržbu a jejich

pořízení je levnější. OLAP kostka se skládá z jedné nebo více tabulek faktů a informací, které jsou uloženy v tabulkách dimenzí. Na rozdíl od transakčních databází, kde jsou data uložena v řádcích a sloupcích (ve dvojrozměrných tabulkách), jsou data v multidimenzionálním modelu uložena ve více dimenzích. Výsledkem je obvykle dimenzionální datová struktura tzn. kostka. „Zjednodušeně by se dalo říct, že krychle je v multidimenzionálním datovém modelu jakýmsi ekvivalentem tabulky v relační databázi.“ (Lacko, 2003, s. 32)

Každá OLAP kostka je tvořena několika dimenzemi. V reálných dimenzionálních databázích je její počet většinou vyšší (až desítky dimenzí) než když bychom si představili klasickou trojrozměrnou kostku tak, jak je zobrazena na obrázku č. 14, která kombinuje pouze časovou a zeměpisnou dimenzi a dimenzi typů produktu. OLAP kostka tak umožňuje datům, aby byla prohlížena nebo modelována v několika dimenzích. Jednotlivé záznamy se nacházejí na průsečících těchto dimenzí.

„Pro výpočet OLAP kostek je zapotřebí vykonat velké množství agregací a výpočtů v reálném čase.“ (Lacko, 2003, s. 32) Tyto operace zatěžují server datového skladu a z důvodu co největšího množství předkalkulovaných agregací v OLAP kostkách se zvyšují nároky na prostor úložiště (kompletní DW až desítky terabytů).



Obrázek č. 14 OLAP kostka (upraveno podle: Lacko, 2003, s. 32)

Výhodou OLAP kostek je pro koncového uživatele možnost provádět vlastní analytické postupy s využitím OLAP pravidel a velmi vysoký dotazovací výkon, který je i u velkého

objemu dat v řádech sekund. Pro potřebu nahlížení, analyzování a práce s agregovanými daty v OLAP kostkách existují speciálně navržené operace.

### 3.3.1 Operace spojené s OLAP kostkou

Důvodem pro vytvoření datových skladů je snadná možnost provádět přijatelným způsobem analýzy nashromážděných dat v multidimenzionálním modelu, resp. v OLAP kostce. Základní operace, které umožňují tyto analýzy, jsou Drill-down, Roll-up, Pivoting, Slicing a Dicing.

Operace Drill-down umožňuje zaměřit se (zjemnit) na podrobnější hodnoty, sestoupit na nižší agregační úroveň (např. v zeměpisné dimenzi z Evropy na jednotlivé státy nebo města v určitém státě). Po provedení této operace je do OLAP kostky přidána jedna nebo více dimenzí. Opakem této operace je Roll-up, která ve zvolených dimenzích nastavuje vyšší agregační úroveň, tzv. zobecňuje (např. výsledek hospodaření za kalendářní měsíce zobecňuje na kvartály či roky). V případě provedení Roll-up operace, je jedna nebo více dimenzí z OLAP kostky odstraněna. Pivoting umožňuje rotovat s datovou kostkou a tím měnit úhel pohledu na agregovaná data z pohledu dimenzí. Slicing nabízí výběr jedné konkrétní dimenze z dané kostky a vytvoří kostku novou (sub-kostku), tato operace zároveň snižuje dimenzionalitu kostek. Operace Dice je velmi podobná operaci Slicing, jen s tím rozdílem, že kromě jedné dimenze jsou zde vybrané dvě a více. Také zde vznikne nová sub-kostka a současně s tím se snižuje počet hodnot v jedné nebo více dimenzích. (Padre, c2010)

Další používané operace jsou Drill-across a Drill-through. Drill-across je přístup k více než jedné tabulce faktů, které jsou spojeny společnými tabulkami dimenzí a kombinace datových kostek, které sdílejí jednu nebo více dimenzí. Drill-through je operace Drill-down až na zdrojová data (spodní úroveň) OLAP kostky k relačním tabulkám. (Čech a Bureš, 2009, s. 105)

Termín multidimenzionální databáze (MDD) je často zaměňován s pojmem online analytical processing (OLAP), tyto pojmy mají ale odlišné významy. MDD je databáze a OLAP je aktivita používaná k analýze této databáze. Zmatek je způsoben slovem OLAP kostka, které má stejný význam jako MDD, to znamená že OLAP kostka je multidimenzionální databáze. (Rainardi, c2008, s. 379) O OLAP analýze pojednává další kapitola této práce.

## 4 OLAP

Zkratka OLAP (Online Analytical Processing) znamená okamžité zpracování dat. Poněkud výstižnější pojem je pružné nebo rychlé zpracování analýz a dotazů.

„OLAP je aktivita interaktivního analyzování podnikových dat uložených v multidimenzionálním datovém skladu pro účely taktického a strategického rozhodování.“ (Rainardi, c2008, s. 380)

„OLAP umožňuje provádět komplexní výpočty pro potřeby analýz v multidimenzionálních systémech, které by nebylo možné snadno napodobit čistě relačním řešením.“ (Stackowiak, Rayman a Greenwald, c2007, s. 116)

V roce 1993 E. F. Codd publikoval v článku „*Providing OLAP to User-Analysts: An IT Mandate*<sup>10</sup>“ sadu dvanácti pravidel, která by měl OLAP dodržovat, tato pravidla jsou do dnes dobře známá. Dalšíh šest pravidel, o nichž panuje už menší povědomí, bylo vytvořeno v roce 1995 a všech osmnáct jich bylo následně uspořádáno do čtyř skupin, které E. F. Codd nazval „rysy“.

*Základní rysy:*

Pravidlo č. 1 – Multidimenzionální konceptuální pohled (původně pravidlo č. 1): OLAP by měl uživateli poskytovat multidimenzionální model se kterým je snadnější a intuitivně jednodušší manipulace, než je tomu u jedno-dimenzionálních modelů (transakční systémy), a který odpovídá jeho podnikatelským potřebám. Toto pravidlo zahrnuje operace slice, dices nebo pivot jako součást požadavku na jeho splnění. (Lacko, 2003, s. 112; Pendse, c2001)

Pravidlo č. 2 – Intuitivní manipulace s daty (původně pravidlo č. 10): Pravidlo definuje konsolidované přeorientování cest na nejnižší (detailní) úroveň a zpátky, to odpovídá operacím drill-down a roll-up. Uživatelské rozhraní by mělo být jednoduché pro použití bez jeho vícenásobného procházení. (Lacko, 2003, s. 112; Pendse, c2001)

Pravidlo č. 3 – Dostupnost (původně pravidlo č. 3): Systém OLAP by měl přistupovat pouze k těm datům, které potřebuje k provedení analýzy. Nástroje OLAP (ne samotný uživatel) musí být schopny přistupovat ke všem potřebným údajům nezávisle na tom, ze kterého heterogenního zdrojového systému pocházejí a zároveň musejí provádět konverzi

---

<sup>10</sup> Dostupné z: [http://www.minet.uni-jena.de/dbis/lehre/ss2005/sem\\_dwh/lit/Cod93.pdf](http://www.minet.uni-jena.de/dbis/lehre/ss2005/sem_dwh/lit/Cod93.pdf)

těchto dat k vytvoření jednotného, koherentního a konzistentního uživatelského pohledu. (Lacko, 2003, s. 112; Pendse, c2001)

Pravidlo č. 4 – Dávková extrakce versus interpretace (nové pravidlo): Pravidlo požaduje, aby systémy nabízely vlastní data staging databázi pro OLAP stejně jako přímý přístup k externím datům. (Pendse, c2001)

Pravidlo č. 5 – Modely OLAP analýzy (nové pravidlo): OLAP produkty by měly podporovat všechny čtyři modely analýzy. Kategorické (parametrizované statické reporty), exegetické (operace slicing, dicing a dril-down), kontemplativní („co kdyby?“ analýzy) a stereotypní (modely hledající řešení). (Pendse, c2001)

Pravidlo č. 6 – Architektura klient-server (původně pravidlo č. 5): Severová část OLAP nástrojů musí být dostatečně inteligentní, aby mohli být různí klienti připojeni bez velkého úsilí, s přihlédnutím k maximálnímu výkonu, flexibilitě a cenové dostupnosti. (Lacko, 2003, s. 112; Pendse, c2001)

Pravidlo č. 7 – Transparentnost (původně pravidlo č. 2): OLAP by měl být součástí otevřené systémové architektury, která může být vložena kdekoli, kde je to uživatelem vyžadováno bez nepříznivého vlivu na funkčnost hostitelských nástrojů. Technologie OLAP, podřízená databáze a architektura výpočtů by měly být pro uživatele co nejvíce transparentní. (Lacko, 2003, s. 112; Pendse, c2001)

Pravidlo č. 8 – Podpora více uživatelů (původně pravidlo č. 8): Nástroje OLAP musí zajistit integritu, bezpečnost a zároveň také možnost souběžného vyhledávání a práce s konkrétním modelem. (Lacko, 2003, s. 112; Pendse, c2001)

#### *Speciální rysy*

Pravidlo č. 9 – Zacházení s nenormalizovanými daty (nové pravidlo): Žádným aktualizacím provedených v prostředí OLAP by nemělo být povoleno měnit uložená denormalizovaná data v systémech, které provádějí operace plnění MDD. (Pendse, c2001)

Pravidlo č. 10 – Ukládání OLAP výsledků a zachování jejich izolace od zdroje dat (nové pravidlo): Provádění operací „čtení-zápis“ OLAP aplikacemi by nemělo probíhat přímo na základních datech a změny OLAP dat by měly být odlišené od transakčních dat. (Pendse, c2001)

Pravidlo č. 11 – Extrakce chybějících hodnot (nové pravidlo): Všechny chybějící hodnoty je třeba odlišit od nulové hodnoty v jednotném vyobrazení definované relačním modelem. (Pendse, c2001)

Pravidlo č. 12 – Zacházení s chybějícími hodnotami (nové pravidlo): Všechny chybějící hodnoty jsou OLAP analyzátory ignorovány bez ohledu na zdroje, ze kterých pocházejí. (Pendse, c2001)

#### *Rysy vykazování*

Pravidlo č. 13 – Flexibilní vykazování (původně pravidlo č. 11): Pravidlo vyžaduje, aby všechny dimenze (řádky, sloupce a buňky OLAP kostky) mohly být ve všech případech vybrány takovým způsobem, který umožní analýzu a intuitivní vizuální prezentaci analytických sestav dle potřeb uživatele. (Pendse, c2001; Lacko, 2003, s. 112)

Pravidlo č. 14 – Konzistentní vykazování (původně pravidlo č. 4): Zvyšování počtu dimenzí nebo velikosti databáze vlivem času by nemělo žádným způsobem ovlivnit výkon vykazování. (Pendse, c2001)

Pravidlo č. 15 – Dynamické ošetření řídkých matic (původně pravidlo č. 7): Systém OLAP musí být schopen automaticky přizpůsobit své fyzické schéma konkrétního typu analytického modelu, objemu dat a řídkosti matic zároveň s dosažením a udržením požadované úrovně výkonu. (Pendse, c2001; Lacko, 2003, s. 112)

#### *Kontrola dimenzí*

Pravidlo č. 16 – Generická dimenzionalita (původně pravidlo č. 6): „Každá dimenze údajů musí být ekvivalentní ve své struktuře a operačních schopnostech.“ (Lacko, 2003, s. 112)

Pravidlo č. 17 – Neomezené dimenze a úrovně agregace (původně pravidlo č. 12): Systém OLAP by neměl zavádět žádná, ani umělá (např. z technických důvodů), omezení počtu dimenzí nebo úrovní agregace. (Lacko, 2003, s. 112)

Pravidlo č. 18 – Neomezené křížové dimenzionální operace (původně pravidlo č. 9): Systémy OLAP musí rozeznávat hierarchie dimenzí a automaticky vykonávat asociované kumulované výpočty skrz všechny dimenze i mezi nimi. (Lacko, 2003, s. 112)

Jen malý počet nástrojů pro OLAP splňuje všech osmnáct pravidel. S největší pravděpodobností je to z důvodu sponzorování E. F. Codda softwarovou firmou Arbor Software<sup>11</sup> při vydávání publikace, ve které jsou tato pravidla uveřejněna.

Tato pravidla jsou v současné době již zřídka citována, málo využívána a také se ukázalo, že je to nevhodný způsob, jak detekovat OLAP. Dvanáct nebo dokonce osmnáct pravidel je také velmi těžké k zapamatování, a tak v roce 1995 Nigel Pendse sumarizoval všech osmnáct pravidel OLAP do pouze pěti počátečních písmen slov „*Fast Analysis of Shared Multidimensional Information*“ neboli zkráceně FASMI. (Pendse, c2001)

*Fast* znamená, že systém doručí výsledky dotazování uživateli zpět v čase okolo pěti sekund s tím, že nejjednodušší analýzy netrvají déle než sekundu a jen málo jich trvá více než dvacet sekund. Doba odezvy by rozhodně neměla překročit dobu třiceti sekund. Tuto rychlost není snadné s velkým množstvím dat dosáhnout, a to zvláště v případech, když uživatel mění požadavky analýzy během jejího průběhu, tzn. bez zastavení dotazování nebo při použití ad-hoc<sup>12</sup>. Výrobci pro dosažení tohoto cíle používají širokou řadu způsobů, včetně rozsáhlé předkalkulace, specializovaných forem ukládání dat nebo konkrétními požadavky na používaný hardware. Pokud systém svou rychlostí nevyhovuje požadavkům uživatelů, má to logický dopad na kvalitu i četnost prováděných analýz. (Pendse, c2001)

*Analysis* znamená, že systém je schopný spojit statistické analýzy s jakoukoli obchodní logikou v dostatečně pochopitelném stavu pro koncového uživatele. Umožňuje uživateli definovat nové ad-hoc výpočty jako součásti analýz libovolným způsobem bez nutnosti programování a poskytuje všechny požadované funkce analýz intuitivním způsobem koncovému uživateli. To může zahrnovat specifické funkce jako je neprocedurální modelování, analýzy časových řad, data mining a další funkce v závislosti zaměření aplikací na cílové trhy. (Pendse, c2001)

*Shared* znamená, že systém musí umožňovat sdílení informací napříč firmou a implementuje veškeré bezpečnostní požadavky víceuživatelského přístupu k datům pro čtení a zápis. Tato oblast je největší slabinou mnoha OLAP systémů, jejichž autoři mají

---

<sup>11</sup> Výrobce systému pro správu multidimenzionálních databází Essbase, který poskytuje platformu, na niž lze vytvářet analytické aplikace.

<sup>12</sup> Zadávání dotazů v případech akutní potřeby získání názoru.



sklon se domnívat, že všechny OLAP aplikace by měly být pouze pro čtení spolu s jednoduchými bezpečnostními kontrolami. (Pendse, c2001; Příbylslavský, 2003)

*Multidimensional* je klíčovým požadavkem a jednoslovnou definicí systémů OLAP. Tento systém musí poskytovat multidimenzionální koncepční pohled na data, včetně možnosti používat vícenásobné hierarchie. „Právě tato oblast dává OLAP systémům jedinečnost a potřebnou analytickou sílu.“ (Příbylslavský, 2003)

*Information* jsou všechna data a z nich odvozené kvalitní, relevantní a správné informace. Zpracování objemu dat, ze kterých jsou tyto informace tvořeny, se u jednotlivých systémů liší. Největší OLAP produkty mohou obsahovat i tisíckrát více dat než ty nejmenší, proto bychom je neměli porovnávat podle velikosti úložného prostoru, ale podle množství vstupních dat, které jsou schopné kvalitně zpracovat. (Pendse, c2001; Příbylslavský, 2003; Sládeček, 2006)

Kromě FASMI testu, který by měl jednoduše a jasně definovat cíle, které by měly OLAP systémy splňovat, by se měly brát v úvahu také běžné požadavky kladené na informační systémy, jejich vytvoření a následný provoz, právní odpovědnost výrobce, bezpečnost, řízení přístupu a požadavků, uživatelská přívětivost, řešení na zakázku nebo výběr již hotového produktu a samozřejmě mnoho dalších. (Sládeček, 2006)

## **4.1 Uložení dat v OLAP**

Se zvyšujícím se počtem rozměrů v multidimenzionálních databázích současně velmi rychle rostou i nároky na kapacitu úložného prostoru. V praxi se u těchto databází používají různé technologie pro kompresi, ukládání a zpracování dat potřebných k analýzám. Mezi nejčastěji používané technologie patří MOLAP, ROLAP, HOLAP a DOLAP.

### **4.1.1 Multidimenzionální OLAP (MOLAP)**

Technologie pro multidimenzionální online analytické zpracování (MOLAP) získává data z datového skladu nebo ze zdrojových systémů a ukládá je v multidimenzionálních datových strukturách (OLAP kostkách). Během tohoto procesu je předem generováno tolik výpočtů, kolik je z technického a časového hlediska možné. Databáze je tak organizována pro potřebu rychlého získávání příslušných údajů z více dimenzí. Hlavní výhodou této technologie je maximální výkon při dotazování uživateli, nevýhodou pak redundance údajů z důvodu jejich uložení v relační i multidimenzionální databázi nebo jen omezené

množství dat, které lze tímto způsobem zpracovat. Z tohoto důvodu se většinou nehodí pro práci s velkými objemy dat. (Lacko, 2003, s. 116; Čech a Bureš, 2009, s. 107)

#### **4.1.2 Relační databázový OLAP (ROLAP)**

Relační online analytické zpracování údajů (ROLAP) získává data přímo z relačních databází datových skladů, datových tržišť nebo i z relačních databází. Tyto údaje jsou po zpracování předloženy uživateli jako multidimenzionální pohled na data. Metadata jsou v ROLAP uložena jako záznamy v relační databázi a jsou dynamicky používána OLAP serverem pro generování SQL příkazů, které jsou nezbytné při získání dat požadovaných uživatelem. Na rozdíl od způsobu ukládání dat v MOLAP, data v ROLAP zůstávají v relační databázi, a proto nevzniká problém s redundancí. Doba zpracování dotazů je obecně pomalejší, než při použití technologií MOLAP nebo HOLAP, na druhou stranu ROLAP umožňuje uživatelům přistupovat přímo k aktuálním a historickým datům do zdrojové databáze a neexistují zde žádná omezení týkající se velikosti použité databáze. (Lacko, 2003, s. 116; Čech a Bureš, 2009, s. 106)

#### **4.1.3 Hybridní OLAP (HOLAP)**

Hybridní OLAP je kombinací obou předcházejících způsobů ukládání dat, přičemž se snaží využít jejich výhody a jejich nevýhody co možná nejvíce eliminovat. Údaje pro detailní analýzy, pracující s velkým objemem dat, zůstávají v datových zdrojích a agregovaná data, u kterých je jejich získání časově mnohem náročnější, jsou uložena v multidimenzionální databázi. HOLAP tak poskytuje propojení mezi rozsáhlými objemy dat v relačních databázích a současně nabízí výhodu rychlejšího výkonu při dotazování multidimenzionálně uložených agregovaných dat. (Lacko, 2003, s. 117; Čech a Bureš, 2009, s. 107)

#### **4.1.4 Desktopový OLAP (DOLAP)**

Desktopová OLAP technologie online analytického zpracování údajů (DOLAP) umožňuje uživateli stáhnout si potřebnou podmnožinu OLAP kostky z centrálního úložiště OLAP dat do lokálního počítače a následně provádět veškeré analýzy nad touto lokální kostkou bez nutnosti připojení k OLAP serveru. DOLAP je tak výhodné využít zejména pro mobilní aplikace nebo podporu mobilních uživatelů. (Gála, Pour a Toman, 2006, s. 93; Čech a Bureš, 2009, s. 107)

OLAP je interaktivní nástroj využívaný obchodními analytiky k dotazování v datovém skladu pro potřeby provádění obchodních analýz a zároveň je i součástí Business Intelligence. Mezi tři široce používané činnosti BI patří vykazování, OLAP a data mining. (Rainardi, c2008, s. 380)

## 5 Business Intelligence

Existuje celá řada definicí, které více či méně vysvětlují termín Business Intelligence. Většina těchto definic se omezuje na to, že BI je komplex přístupů a aplikací, které podporují analytické a plánovací činnosti podniků a organizací. V některých případech je uveden princip multidimenzionality, kterým se rozumí možnost nahlížení na daná fakta z několika různých úhlů.

Jako úplně první popsal termín Business Intelligence Hans Petr Luhn v roce 1958, když ve svém článku *A Business Intelligence System* definoval termín *Business* jako: „Soubor činností prováděných k jakémukoli účelu, ať už je to věda, technologie, obchod, průmysl, právo, vláda, obrana a tak dále.“ a termín *Intelligence* jako: „Schopnost vnímat vzájemné vztahy uvedených skutečností takovým způsobem, aby prováděná činnost směřovala k požadovanému cíli.“ (Luhn, 1958, s. 314)

V té době se jednalo především o pochopení a porozumění podnikových skutečností a faktů takovým způsobem, aby se co nejefektivněji dosáhlo stanovených cílů.

Autoři různých odborných publikací definují pojem Business Intelligence několika způsoby:

„Pojem Business Intelligence můžeme definovat jako proces transformace údajů na informace a převod těchto informací na poznatky prostřednictvím objevování.“ (Lacko, 2003, s. 45)

„Business Intelligence (BI) představuje komplex přístupů a aplikací IS/ICT, které téměř výlučně podporují analytické a plánovací činnosti podniků a organizací a jsou postaveny na principu multidimenzionality, kterou zde rozumíme možnost pohlížet na realitu z několika možných úhlů.“ (Novotný, Pour a Slánský, 2005, s. 17)

„Business Intelligence je o způsobech, jak poskytovat informace koncovým uživatelům, aniž by museli být odborníky v operativním výzkumu.“ (Dresner, 2006 cit. podle Martens, 2006)

„Business Intelligence je souhrnný pojem pro procesy, technologie a nástroje potřebné k přetvoření dat do informací, informací do znalostí a znalostí do plánů, které umožní provést akce podporující splnění primárních cílů organizace.“ (Hroch a Cach, 2007)

„Business Intelligence lze definovat jako právo přístupu ke správným údajům a informacím potřebných k vykonání správného obchodního rozhodnutí ve správný čas.“ (Stackowiak, Rayman a Greenwald, c2007, s. 3)

„Business intelligence je soubor činností k porozumění obchodní situaci provedením různých typů analýz dat společnosti, jako i externích dat poskytnutých třetími stranami, pro podporu vytvoření strategického, taktického a operativního obchodního rozhodnutí a přijetí nezbytných opatření pro zlepšení výkonnosti podniku.“ (Rainardi, c2008, s. 12)

„Business Intelligence je sada procesů, know-how, aplikací a technologií, jejichž cílem je účinně a účelně podporovat řídicí aktivity ve firmě. Podporují analytické, plánovací a rozhodovací činnosti na všech úrovních a ve všech oblastech podnikového řízení, tj. prodeje, nákupu, marketingu, finančního řízení, controllingu, majetku, řízení lidských zdrojů, výroby a dalších.“ (Pour, Maryška a Novotný, 2012, s. 16)

„Business intelligence je zastřešující termín, který se vztahuje ke znalostem, procesům, technologiím, aplikacím a postupům, které usnadňují podnikové rozhodnutí.“ (Laberge, 2012, s. 33)

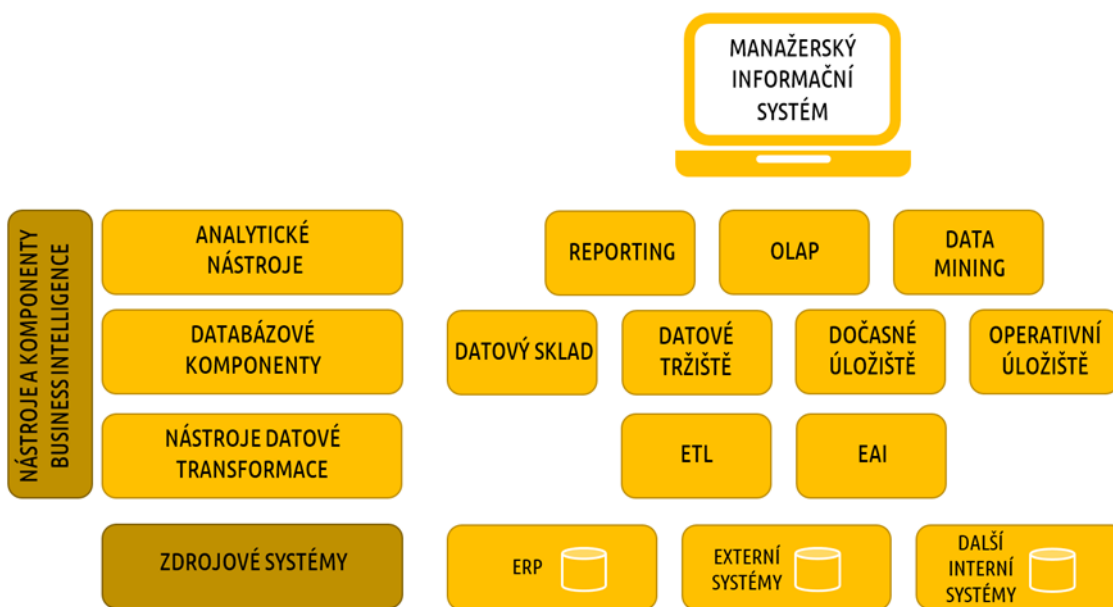
Pokud bychom o BI hovořili jako o výrazu, pokrýval by rozsáhlou řadu různých nástrojů, činností, procesů, aplikací a technologií, které podporují porozumění firemním datům, jejich vzájemným vazbám a možným vývojovým tendencím. V případě pohlížení na BI jako na oblast, by tato oblast poskytovala podnikům nástroje pro získávání, transformaci a analýzu operativních dat nezbytných k podpoře manažerských rozhodnutí.

## **5.1 Nástroje a aplikace BI**

Nástroje Business Intelligence v praxi přinášejí souhrnné a nové pohledy na stávající data a na klíčové ukazatele získané z těchto dat. Aplikace BI pokrývají analytické funkce a funkce plánování většiny oblastí řízení podniku (řízení lidských zdrojů, finanční řízení, prodej a nákup, marketing, výrobu a další). Mezi nástroje a aplikace Business Intelligence lze zařadit produkční (zdrojové) systémy, dočasná úložiště (Data Staging Area), transformační nástroje (ETL), operativní úložiště dat (ODS), datové sklady (DW), datová tržiště (DM), multidimenzionální databázi (OLAP kostka), OLAP, nástroje pro správu metadat, reporting, data mining a ostatní. (Novotný, Pour a Slánský, 2005, s. 19)

Přínos nástrojů a aplikací BI může být např. v lepším cílení marketingových kampaní, růstu tržeb nebo v poklesu zásob, s tím je ale spojena potřeba aktivního používání těchto nástrojů managementem podniku spolu s nutností práce s aktuálními a správnými daty.

Většina těchto nástrojů byla z hlediska své důležitosti architektury datových skladů blíže představena v předcházejících kapitolách této práce.



Obrázek č. 15 Nástroje a komponenty Business Intelligence (upraveno podle: <https://www.systemonline.cz/business-intelligence/data-uz-nestaci-manazerske-systemy-a-nastroje-pro-podporu-rozhodovani-ve-verejne-a-statni-sprave.htm>)

„Lze říci, že nástroje business intelligence integrují informační a podnikatelskou strategii a pomáhají podniku přežít v současném turbulentním prostředí.“ (Systémová integrace 2/2005, 2005, s. 106)

## 6 Specifikace společnosti Městská nemocnice, a.s.

Městská nemocnice a. s., Dvůr Králové nad Labem (dále jen „MNDK“) je zdravotnické zařízení ve stoprocentním vlastnictví Královéhradeckého kraje a plně podléhá správě Zdravotnického holdingu Královéhradeckého kraje (dále jen „ZH KHK“). Z odborného hlediska je MNDK poskytovatelem zdravotní léčebné a diagnostické péče na pracovištích akutní lůžkové péče (v oboru interní, chirurgické a urologické medicíny), oddělení následné péče, odborných ambulancích, a oddělení komplementu. (Pohl, 2014)

V MNDK je úspěšně zaveden integrovaný systém řízení kvality, environmentu a bezpečnosti a ochrany zdraví při práci. MNDK je držitelem certifikátů EN ISO 9001: 2008, EN ISO 14001: 2004 a OHSAS 18001. Na začátku roku 2017 MNDK po závěrečném auditu Spojené akreditační komise, o.p.s. obdržela akreditaci potvrzující kvalitu poskytované zdravotní péče a bezpečí pacientů. MNDK má 116 akutních, 50 ošetrovatelských lůžek a zhruba 200 zaměstnanců. (Pohl, 2014)

### 6.1 Zdrojové systémy

Ve společnosti MNDK je používáno několik různých systémů. Hlavním a patrně i nejpoužívanějším systémem je s pochopitelných důvodů NIS<sup>13</sup> FONS Akord od společnosti STAPRO, s.r.o., dále pak neméně důležitý LIS<sup>14</sup> od stejné společnosti FONS Openlims, ekonomický informační systém FEIS od společnosti ARBES Technologies, s.r.o., Microsoft Dynamics NAV – Navision od stejné společnosti využívaný všemi nemocnicemi ZH KHK pro základní pracovně-právní agendu a komunikaci s veřejnou správou (zejména ČSSZ<sup>15</sup>), objednávkový systém NeOS od společnosti MEDISYSTEMS a.s. a další externí systémy používané zejména pro aktualizaci číselníků v databázi NIS.

„FONS Akord je moderní klinický informační systém, který pokrývá provoz klinických pracovišť řadou navzájem propojených produktů umožňujících vedení zdravotní dokumentace a podporujících provozní činnosti na jednotlivých klinických pracovištích. Důležitou vlastností je jeho flexibilita v nastavení, které lze přizpůsobit potřebám daného pracoviště.“ (FONS Akord | STAPRO, c2017)

NIS je provozován na databázovém serveru Fujitsu PRIMERGY RX300 S5, na kterém je nainstalovaný relační databázový systém Microsoft SQL 2008 a operační systém Windows

<sup>13</sup> NIS – nemocniční informační systém

<sup>14</sup> LIS – laboratorní informační systém

<sup>15</sup> ČSSZ – česká správa sociálního zabezpečení

2008 R2 Server. Mezi hlavní role zastávající serverem RX300 patří celková správa databáze NIS, fileserver a Active Directory. V rámci NIS je nakládáno s osobními údaji a důvěrnými informacemi pacientů nemocnice. Základním legislativním předpisem, jehož požadavky musí NIS splňovat, je zákon č. 101/2000 Sb., o ochraně osobních údajů a o změně některých zákonů. Po schválení žádosti přístupu do databáze NIS za účelem extrakce dat potřebných pro vypracování praktické části této práce vedením společnosti MNDK, byly vytvořeny přihlašovací údaje k této databázi s právy pouze ke čtení.

„FONS Openlims je moderní laboratorní informační systém ověřený stovkami pracovišť v několika zemích střední a východní Evropy. Je určen pro všechny typy laboratoří. Podporuje specifické pracovní procesy většiny odborností. Produkt je rutinně používán od malých a středních laboratoří až po konsolidovaná pracoviště velkých fakultních nemocnic.“ (FONS Openlims | STAPRO, c2017)

LIS je provozován fyzicky odděleně na identickém serveru jako NIS (tedy Fujitsu PRIMERGY RX300 S5), i zde je nainstalovaný operační systém Windows 2008 R2 Server a relační databázový systém Microsoft SQL 2008. Dávkový přenos laboratorních výsledků spolu s jejich referenčními hodnotami do NIS je zajištěn v pravidelných intervalech každých deset minut. Žádost o přístup do této databáze byla vedením organizace schválena a následně byly vytvořeny i potřebné přihlašovací údaje s předem nastavenými právy.

„ARBES FEIS je technologicky vyspělý a uživatelsky přívětivý podnikový informační systém, který pokrývá správu ekonomické, účetní a daňové agendy.“ (ARBES Technologies | Zdravotnictví, 2016)

ARBES FEIS využívají všechny nemocnice ZH KHK pro běžné účetní a ekonomické operace. Tento systém je provozován na serverech v nemocnici Náchod a uživatelé, kteří ho využívají, k němu přistupují pomocí VPN<sup>16</sup> připojení. Žádost o přístup do tohoto systému byla zamítnuta s odůvodněním obavy zneužití nebo úniku citlivých dat.

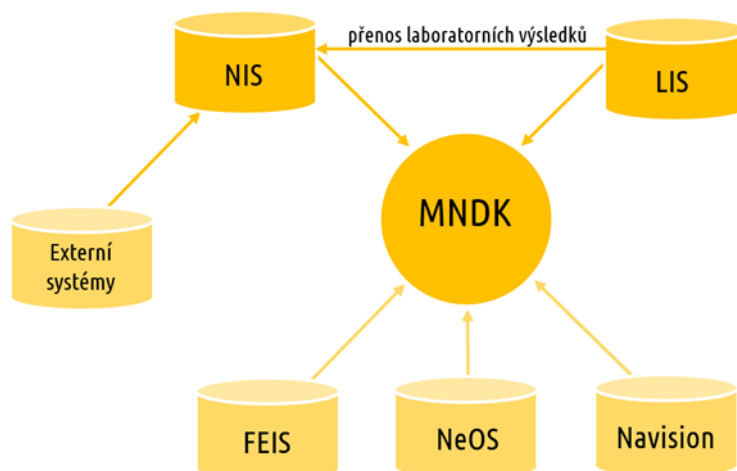
NeOS je komplexní informační systém pro řízení nákupu s ohledem na specifické vlastnosti nákupu ve zdravotnictví. Umožňuje vedení skladové evidence na jednotlivá nákladová střediska včetně vedení evidence pro konsignační sklady nemocnice. K tomuto systému uživatelé přistupují přes webové rozhraní a tento systém je provozován na serveru

---

<sup>16</sup> VPN – Virtual Private Network – prostředek k propojení několika počítačů prostřednictvím nedůvěryhodné (veřejné) počítačové sítě, typicky Internet.



v nemocnici Náchod, i k tomuto systému byla žádost o přístup ze stejných důvodů zamítnuta.



Obrázek č. 16 Znárodnění systémů ve společnosti MNDK (zdroj: autor)

Používané externí systémy slouží především pro aktualizaci číselníků léků, žadatelů o vyšetření, diagnóz, zdravotních výkonů, DRG<sup>17</sup> nebo číselníků PZT<sup>18</sup>, HVLP<sup>19</sup> a IVLP<sup>20</sup> v NIS. Tyto číselníky jsou volně ke stažení na webových stránkách Všeobecné zdravotní pojišťovny a Ministerstva zdravotnictví.

Realizace vlastní práce je založena na praktických zkušenostech z oboru zdravotnictví, zdravotnické informatiky a získaných teoretických poznatcích. Vývoj jednotlivých komponent byl realizován v nástrojích MS SQL 2008 Enterprise, SQL Server Management Studio a MS SQL Business Intelligence Development, který obsahuje několik dalších komponent potřebných pro vytvoření DW. Tato platforma byla zvolena především z důvodu jejího dlouhodobého využívání ve společnosti MNDK spolu s výhodou stejného výrobce těchto produktů, což zaručuje jejich vzájemnou kompatibilitu, její dostupnosti<sup>21</sup> a obsáhlé, volně dostupné dokumentace všech jejích vlastností.

<sup>17</sup> Diagnosis-Related Group – klasifikační systém klinických případů v lékařství

<sup>18</sup> PZT – prostředky zdravotní techniky

<sup>19</sup> HVLP – hromadně vyráběné léčivé přípravky

<sup>20</sup> IVLP – individuálně připravované léčivé prostředky

<sup>21</sup> Microsoft SQL Server 2008 Enterprise je k dispozici jako trial verze na 180 dní

## 7 Výběr vhodného řešení datového skladu

Vybudování datového skladu v jakékoli společnosti je velmi rozsáhlý, dlouhodobý a finančně náročný projekt, který by měl vést ke zlepšení schopnosti pružné reakce na požadavky vedoucích pracovníků jednotlivých organizačních jednotek a zároveň vyhovět i požadavkům top managerů na získání potřebných informací pro podporu rozhodování.

Zvolená metodika, která byla popsána v první části této práce, tedy doporučené praktiky realizace datového skladu na základě definovaných přístupů, je třívrstvá architektura s přítomností ODS (úložiště provozních dat) zaměřená na podnikovou analytiku.

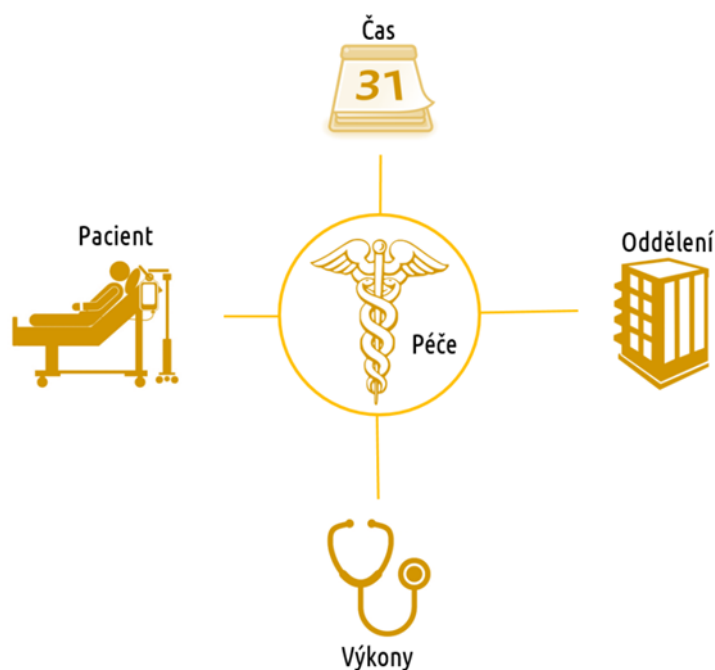
Důvodem pro výběr této metodiky je především potřeba nepřetržitého a neomezeného přístupu k databázím NIS a LIS během celého dne, kterou by mohly rutiny ETL v době svého spuštění vlivem velkého zatížení databázových serverů omezit. V tomto případě bude spouštěna pouze extrakce vybraných dat ze zdrojových databází do ODS v pravidelných intervalech v době od 1:00 do 4:00 hodin ráno. Existence dočasného úložiště dat je zde z důvodu přenesení dat s minimálním dopadem na výkonost produkčních (zdrojových) systémů a pro případnou možnost konverze dat z heterogenního prostředí do stejného formátu. Samotná transformace a nahrání dat do pilotního DW pak bude probíhat na fyzicky odděleném serveru, na kterém je ODS vytvořen a nedojde tak k omezení nebo narušení provozu NIS a LIS. Dalšími důvody je relativně rychlejší vytvoření DW a s tím spojené získání požadovaných výsledků.

Kvůli velkému objemu dat, které jsou v databázích NIS a LIS uloženy, není vhodné využít v multidimenzionální databázi pilotního DW technologii ukládání dat MOLAP, více by zde vyhovovala technologie HOLAP, která ponechává data potřebná pro detailní analýzy ve zdrojovém systému a data, u kterých je jejich získání časově velmi náročné, ukládá do multidimenzionální databáze. Tím by bylo do značné míry zajištěno optimální rozložení analýz a zajištění potřebného výkonu relačních databází NIS a LIS a multidimenzionální databáze pilotního DW.

Vzhledem k rozsahu projektu a na základě priorit podnikových požadavků je zapotřebí použít postupného (tzn. přírůstkového) vytváření DW. Detailní popisy jednotlivých přírůstků jsou nad rámec této práce, mají však společný základ v podobě jednotné koncepce a řešení dílčích prvků.

## 7.1 Definování požadavků organizace

Z popisu organizace byly pro vytvoření pilotního datového skladu určeny základní subjekty, které se účastní podnikových procesů. Hlavním subjektem je samozřejmě poskytovaná péče, dalšími faktory, které vstupují do tohoto procesu, jsou pacient, výkony, které byly provedeny, a také oddělení, které léčebnou péči poskytlo. Z důvodu potřeby analyzování podnikových dat v závislosti na čase je dalším subjektem čas. Základní podoba datového skladu je zobrazena na obrázku č. 17.



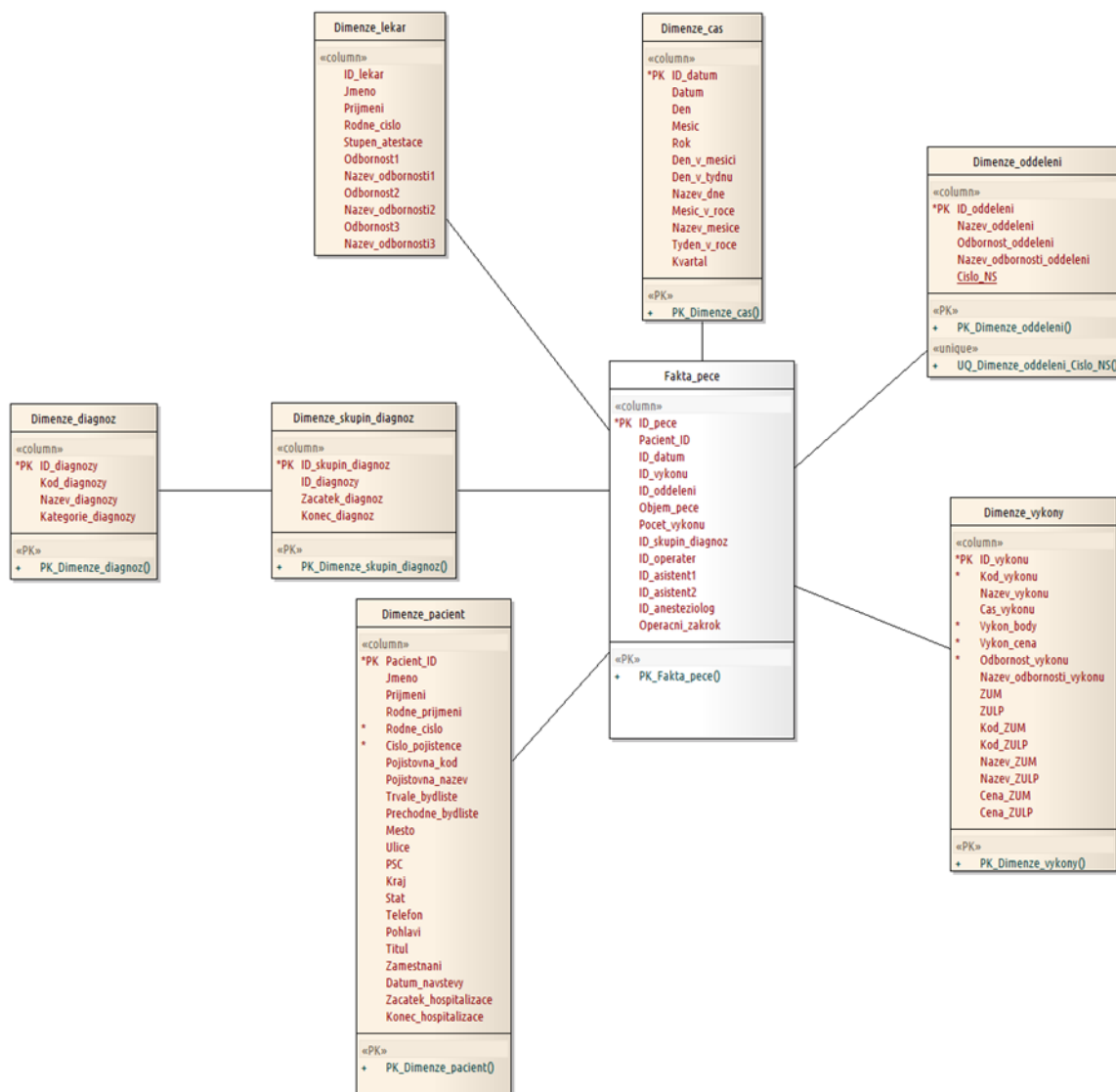
Obrázek č. 17 Konceptuální model pilotního datového skladu (zdroj: autor)

Kromě struktury je také třeba určit funkční požadavky na datový sklad. Datový sklad by měl umožnit analýzu požadovaných dat v závislosti na čase, na pacientovi, na oddělení, na kterém byla léčebná péče poskytnuta a na množství provedených operačních zákroků. Samozřejmostí je možnost vyhledávání pacientů podle objemu poskytnuté péče, diagnóz, věku, pohlaví, zaměstnání atd., analyzování dat na úrovni let, kvartálů, měsíců, týdnů a dnů a od prvního spuštění analyzovat nejméně čtyři roky stará data.

## 7.2 Návrh dimenzionálního modelu

Cílem této kapitoly je zformovat základní požadavky do podoby fyzického modelu datového skladu odvozením příslušných dimenzí z konceptuálního modelu. Tyto dimenze jsou pacient, čas, oddělení, výkony, lékař a diagnózy. Požadované sledované hodnoty,

kteře tvořĩ jãdra faktovũ tabulek, lze odvodit z funkčních požadavků DW. Jsou jimi objem poskytnutě péče a počet provedenũ operativních a neoperativních výkonũ. Spojením jednotlivũ tabulek dimenzĩ s tabulkou faktũ pomocí cizích klíčũ vznikne požadovanũ dimenzionální model, kterũ se vztahuje k podnikově problematice a jehož schéma odpovídã definici multidimenzionální databãze uvedeně v teoretické částĩ této práce.



Obrãzek č. 18 Logický model pilotního datového skladu (zdroj: autor)

*Fakta\_pece* – faktovã tabulka péče obsahuje unikãtnĩ primãrnĩ klíč *ID\_pece*, cizĩ klíče, kterẽ ji spojujĩ s jednotlivými dimenzemi, počty provedenũ výkonũ, operativní zãkroky a objem poskytnutě péče. Tato pološka je vypočítãvána databãzĩ jako součet bodũ za vsechny vỹkony, přepočítãné na koruny, a ZUM a ZULP. Lze předpokládat, že tento

atribut by mohl být jedním z nejčastějších předmětů při dotazování. U faktových tabulek, jak je uvedeno v teoretické části práce, má velký význam stanovení granularity. V tomto případě jeden řádek tabulky odpovídá jednomu ošetřenému pacientovi, tzn. opakovaná návštěva zdravotnického zařízení (převazy, pravidelné kontroly atd.) tím samým pacientem je v tabulce vždy uvedena na samostatném řádku.

*Dimenze\_pacient* – tato dimenze shromažďuje veškeré možné informace o pacientovi. Kromě primárního klíče dimenze *ID\_pacient* obsahuje další popisné atributy jako je rodné číslo, číslo pojištění, pojišťovna, pohlaví, věk, zaměstnání nebo datum návštěvy. Jedním ze základních projevů dimenzionální databáze je její denormalizace. Tato dimenze je toho jasným příkladem, protože obsahuje atributy jako trvalá adresa nebo PSČ, které by v případě normalizované databáze byly umístěny v samostatných tabulkách, a ne v jedné, jako je tomu zde.

*Dimenze\_cas* – dimenze jsou spojeny s faktovou tabulkou pomocí cizích klíčů, obsahují tedy svůj unikátní primární klíč. V případě dimenze času je to atribut *ID\_datum*. Dimenze času je zároveň vhodnou ukázkou možných redundantních informací ukrytých v dimenzionálních modelech. Kromě data samotného je zde také zvlášť definován měsíc, týden a den, a to v některých případech jak v číselné, tak i slovní podobě. Souvisí to s vytvářením na sebe navazujících atributů, tzv. hierarchií. Díky tomu bude možné procházet a zkoumat data uložená v datovém skladu na různých úrovních a získat tak daleko více užitečných informací.

*Dimenze\_oddeleni* – u této dimenze, která obsahuje atributy *Nazev\_oddeleni*, *Odbornost\_oddeleni* nebo *Cislo\_NS*, je její nejdůležitější vlastností možnost odlišit poskytnutou péči ve vazbě na jednotlivá oddělení a jejich dílčí střediska. Tím je zajištěna kontrola plnění objemu a struktury poskytované péče v návaznosti na úhradovou vyhlášku Ministerstva zdravotnictví a smlouvy o poskytování a úhradě zdravotní péče s jednotlivými pojišťovnami, ve kterých je stanovena i její maximální výše.

*Dimenze\_vykony* – v této dimenzi jsou obsaženy údaje veškerých léčebných výkonů spolu s jejich bodovým ohodnocením, cenou, odborností, časovou délkou výkonu a případným vykázaním ZUM či ZULP. Stejně jako u *Dimenze\_pacient* i zde je patrná denormalizace databáze, důvodem je zahrnutí atributů odbornosti, jejího názvu nebo ZUM a ZULP spolu s jejich kódem, názvy a cenou, které by byly procesem normalizace přesunuty do samostatných tabulek.

*Dimenze\_skupin\_diagnoz* a *Dimenze\_diagnoz* – tyto dvě dimenze byly vytvořeny z potřeby zbytečně nezvyšovat počet sloupců ve faktové tabulce. Běžně mívá pacient dvě a více diagnózy (u starších lidí není neobvyklý ani počet dvaceti a více diagnóz), které vytvářejí jednotlivé skupiny diagnóz. Z tohoto důvodu byla do modelu zahrnuta dimenze skupin diagnóz, která potřebné diagnózy pacienta sdružuje do jednotlivých skupin. Pokud bychom predikovali a předem do modelu zahrnuli možnost použití např. až třiceti diagnóz, rapidním způsobem by docházelo k nechtěnému růstu velikosti a fragmentaci dat v databázi. Protože diagnózy se během života člověka mění, jsou v této tabulce obsaženy také atributy pro začátek a konec určité skupiny diagnóz u pacienta.

*Dimenze\_lekar* – tato poslední dimenze obsahuje informace o lékařích. Kromě unikátního primárního klíče *ID\_lekar* jsou zde atributy jméno, příjmení, odbornost lékaře nebo stupeň atestace. I tato tabulka je v denormalizovaném stavu, protože atributy odbornost a název odbornosti by v normalizované databázi byly z této tabulky přemístěny do tabulky nové.

Výsledný logický model, znázorněný na obrázku č. 18, odpovídá schématu hvězdy i přes zahrnutí pomocné dimenze *Dimenze\_skupin\_diagnoz*. V případě normalizace tabulek dimenzí, bychom úplně ztratili výhody multidimenzionální databáze spolu s funkčními požadavky na analýzy a dotazování.

### **7.3 Návrh fyzického modelu a technologické architektury**

Jak bylo uvedeno výše, celý projekt pilotního datového skladu byl realizován především s využitím aplikací a softwaru od společnosti Microsoft. Pro správu pilotního DW byl použit Microsoft SQL Server 2008 Enterprise spolu s programem pro správu databázových tabulek SQL Server Management Studio a MS SQL Business Intelligence Development, který má v sobě zahrnutý další potřebné aplikace. Jde o aplikaci Integration Services, která umožňuje načtení dat do DW a aplikace Reporting Services a Analysis Services, které byly využity při vytvoření analytických reportů a dotazů. Pro analýzu takto vytvořených dat byl použit tabulkový editor Microsoft Excel 2013. Tabulky faktů, dimenzí a jejich vazby byly vytvořeny v programu Enterprise Architect.

*SQL Server Management Studio* – je komplexní sada nástrojů, určených pro správu databázového serveru, umožňující přístup k OLTP i OLAP databázím a programování nových databází v jazyku SQL.

*Integration Services* – tato aplikace je základem pro ETL rutiny používané k extrakci dat ze zdrojových systémů, jejich transformaci a nahrání do datového skladu nebo jiného cílového systému.

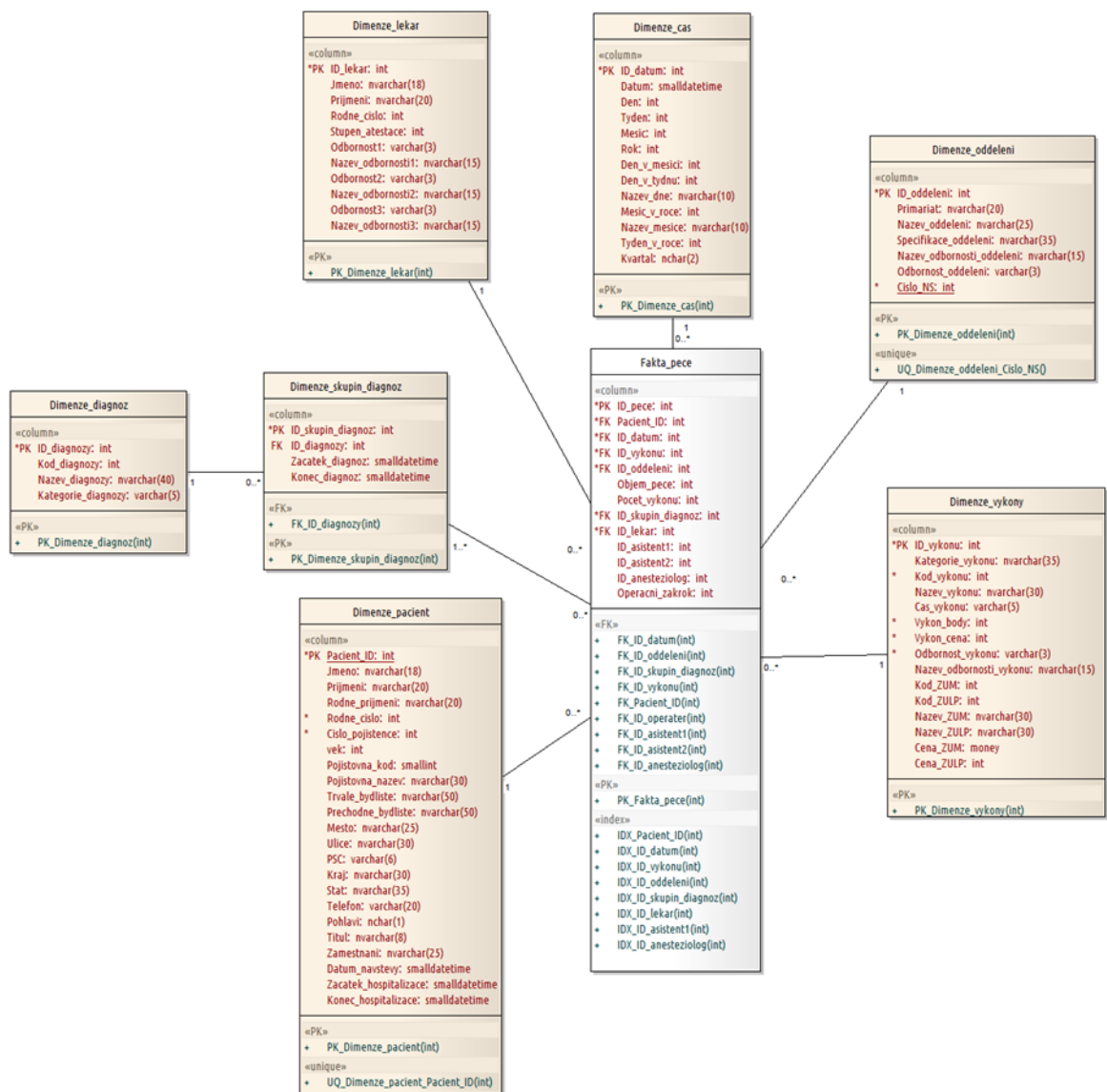
*Analysis Services* – je součástí Microsoft SQL Serveru, ale přesto jde o samostatný, na MS SQL Serveru nezávislý, OLAP server, který podporuje rychlou a pokročilou analýzu velkých a složitých datových sad s využitím vícedimenzionálních úložišť.

*Reporting Services* – tato aplikace obsahuje tvorbu základních i pokročilých reportů obohacených o nejrůznější grafické prvky, automatizaci reportingu nebo integraci s OLAP databázemi.

Základním softwarem pro všechny výše zmíněné technologie byl operační systém Windows 2008 Server v 64bitové verzi nainstalovaný na záložním serveru Fujitsu PRIMERGY RX100 S6 s rozšířenou kapacitou operační paměti (32 GB) a diskového prostoru (4 TB).

Pro převedení logického modelu pilotního DW na model fyzický je potřeba přizpůsobit jednotlivé datové typy databázi SQL Serveru. V případě většiny číselných hodnot byly datové typy nastaveny na *int*, u textu na *varchar* a *nvarchar* a u datumů na *smalldatetime*. Datový typ *nvarchar* byl zvolen z důvodu obsažení *unicode* znaků, které se mohou vyskytnout např. u jména ulic v adresách, v názvu pojišťovny nebo u příjmení pacienta či lékaře. Tento datový typ zabírá v databázi dva byty pro každý *unicode* i *non-unicode* znak, ale i přesto by konečná velikost pilotního DW neměla přesáhnout celkovou kapacitu diskového prostoru. Na rozdíl od klasického *date* se datový typ *smalldatetime* liší v rozmezí hodnot, které v nich mohou být obsaženy, u datového typu *smalldatetime* je to od roku 1900 do roku 2079, což je pro potřeby tohoto projektu více než dostačující.

Po přizpůsobení datových typů databázi je nutné správně zvolit cizí klíče k vytvoření vazeb mezi tabulkami dimenzí a faktovou tabulkou. Faktová tabulka tedy bude obsahovat informace o primárních klíčích každé dimenze, kterými jsou *Pacient\_ID*, *ID\_datum*, *ID\_vykonu*, *ID\_oddeleni*, *ID\_skupin\_diagnoz*, *ID\_lekare* atd. Již v této fázi převodu je rozumné myslet na případnou optimalizaci systému a navrhnout také základní indexy, které napomáhají vyššímu výkonu databáze (rychlejší vyhledávání nebo třídění dat). Indexy byly umístěny na cizí klíče ve faktové tabulce z důvodu častého spojování všech dimenzí s touto tabulkou při provádění reportů a analýz.



Obrázek č. 19 Fyzický model pilotního datového skladu (zdroj: autor)

## 7.4 Šablony rutin ETL

Před samotným vytvořením DW, ale i během něho, obvykle nelze přesně specifikovat kolik bude mít datový sklad zdrojových systémů, tabulek nebo transformací. Je zcela běžné, že datový sklad je neustále upravován a rozšiřován na základě měnících se požadavků, a proto je jeho konfigurovatelnost jedním ze základních klíčů k úspěchu kvalitního projektu.

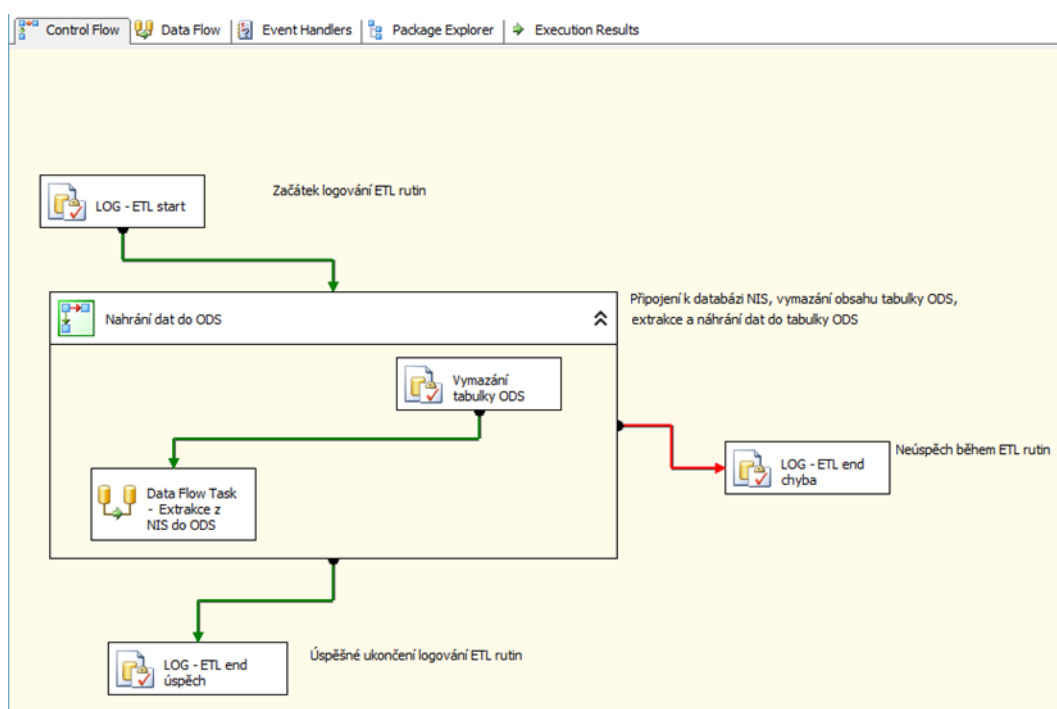
Jednou z nejdůležitějších a současně nejvíce upravovaných součástí DW jsou ETL rutiny. Tyto rutiny by proto měly být co nejvíce konfigurovatelné tak, aby byla zajištěna možnost rozšíření jejich dalšího nastavení (např. extrakce dat z nového zdrojového systému).



Mezi nejrozšířenější technologie a nástroje pro realizaci ETL rutin patří i SQL Server 2008, ve kterém je tato realizace zajištěna prostřednictvím komponenty *Integration Services*. ETL rutiny lze z funkčního hlediska rozdělit na rutiny pro extrakci dat ze zdrojového systému do dočasného úložiště dat a rutiny pro transformaci, očištění, deduplikaci a nahrání dat z ODS do datového skladu. Pro obě tyto verze ETL rutin byly vytvořeny samostatné návrhy.

#### 7.4.1 ETL rutiny pro extrakci dat

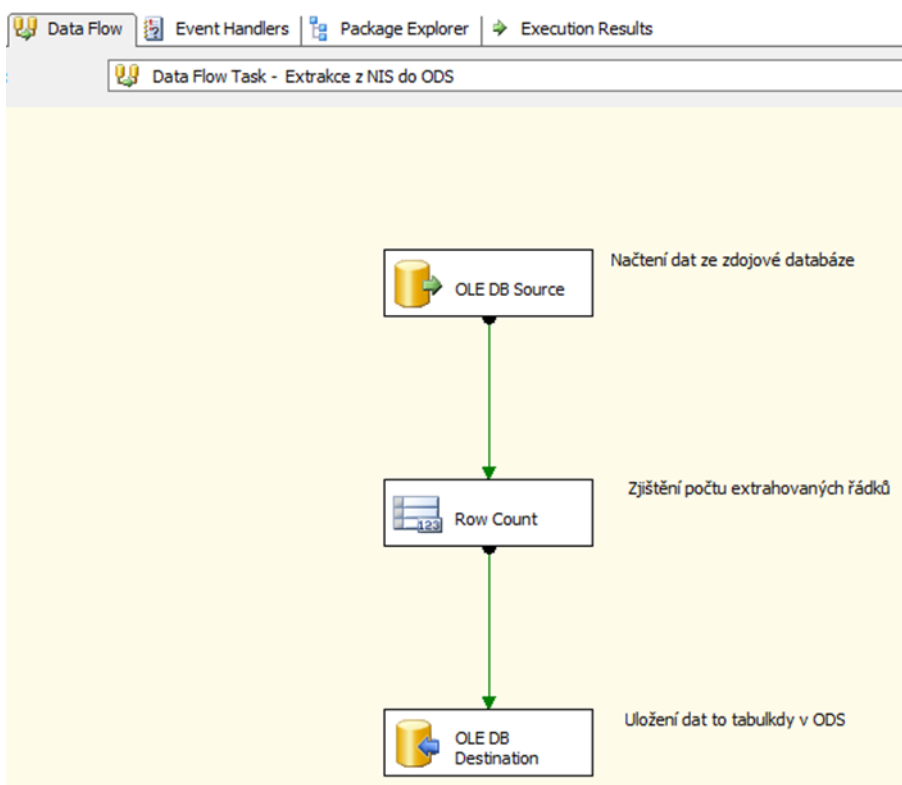
První návrh ETL rutin zajišťuje pouze extrakci dat ze zdrojových databází do ODS. Neprobíhají zde tedy žádné úpravy a zdrojová data jsou do dočasného úložiště dat kopírována ve stejné struktuře, stejném datovém typu a v poměru 1:1. Ve většině případů se vytváří tolik „balíčků“ ETL rutin, kolik je vybraných tabulek použitých k extrakci dat ve všech zdrojových systémech. Obvykle jsou to desítky a více tabulek a tedy i „balíčků“ ETL rutin. Proto je velmi vhodné vytvořit vzor zahrnující již nastavené logování průběhu tohoto procesu do souboru a přihlášení do jednotlivých zdrojových databází, které mají všechny rutiny extrakce shodné pro budoucí usnadnění vytváření dalších ETL rutin.



Obrázek č. 20 Vzor ETL rutin extrakce dat – control flow (zdroj: autor)

Na obrázku č. 20 je zobrazen kontrolní tok (control flow) jednotlivých kroků ETL rutiny pro extrakci dat, jejichž výsledky se ukládají do log souboru. Tento soubor je kontrolován administrátorem datového skladu a v případě chyby je o ní i automaticky informován.

Data Flow Task, zobrazený jako čtvrtý prvek na obrázku č. 20, je dále dekomponován na několik dalších kroků, kterými jsou extrakce (načtení) dat ze zdrojové databáze, získání počtu extrahovaných záznamů a uložení dat do dočasného úložiště dat. Všechny tyto kroky jsou zobrazeny na obrázku č. 21.



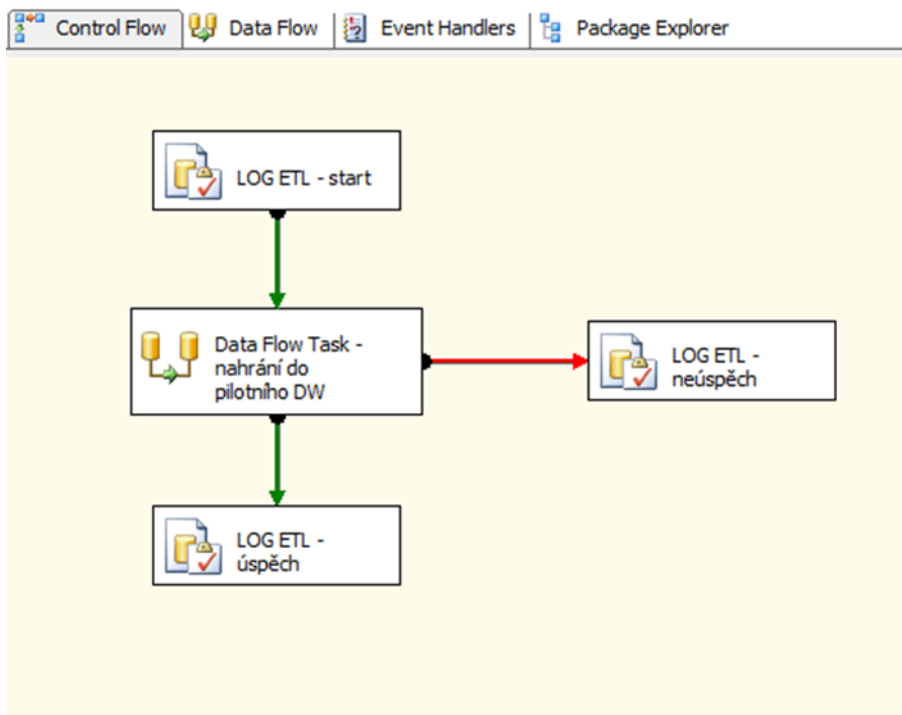
Obrázek č. 21 Vzor ETL rutin extrakce dat – data flow (zdroj: autor)

Stejným způsobem byly vytvořeny i zbývající ETL rutiny pro extrakci dat ze všech tabulek zdrojových databází NIS a LIS, které obsahovaly záznamy potřebné pro naplnění tabulek v dočasném úložišti dat, a tedy i v pilotním datovém skladu.

#### 7.4.2 ETL rutiny pro transformaci a nahrání dat

Mezi dočasným úložištěm dat a pilotním datovým skladem již probíhají určité transformace, které jsou samozřejmě mnohem náročnější než samotná extrakce dat. Uskutečňují se zde také výpočty pro některé agregace (např. objem péče), aby byla zajištěna dostatečně rychlá doba odezvy při dotazování a vytváření reportů, čištění dat

použitím *fuzzy lookup* (kontrola měst v adresách) nebo *fuzzy grouping* (čištění atributu *Zamestnani*). Logovací mechanismus je velmi podobný jako u extrakce dat, jedním z rozdílů je získávání počtu upravených a nových údajů v tabulkách pilotního datového skladu namísto počtu extrahovaných záznamů nebo neexistence potřeby přihlašovacích údajů do zdrojových databází z důvodu získávání dat z ODS.

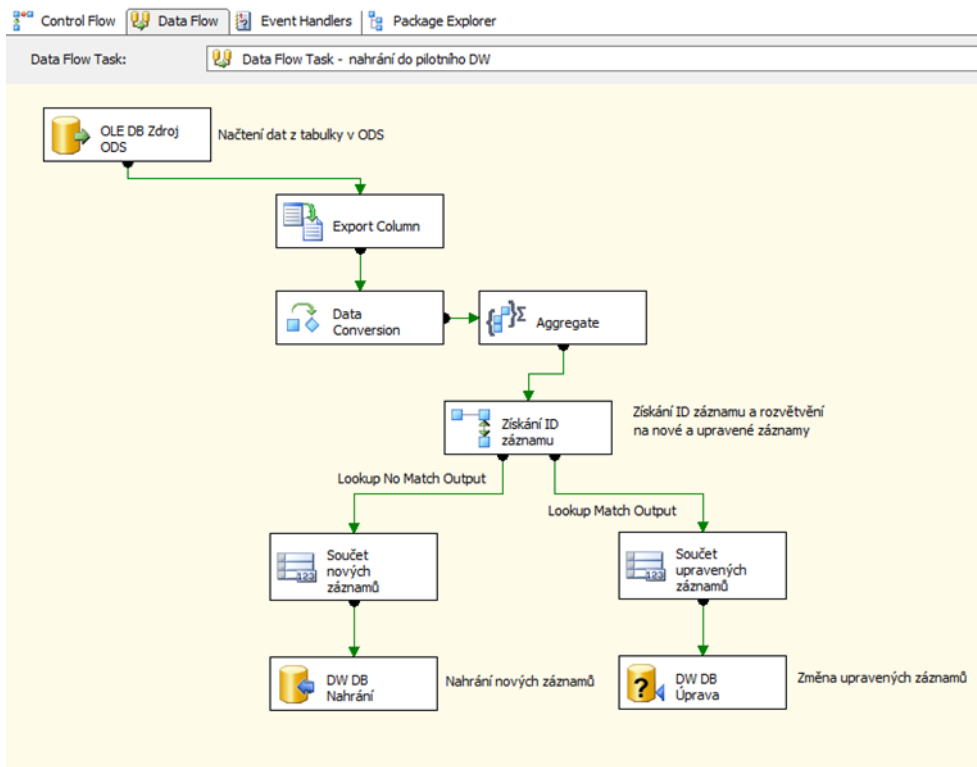


Obrázek č. 22 Vzor ETL rutiny nahrání dat do pilotního DW – control flow (zdroj: autor)

I zde je Data Flow Task dále dekomponován na další kroky, které obsahují načtení dat z tabulek ODS, jejich transformace pro následné uložení do tabulky dimenzí pilotního DW, získání ID záznamu pro uložení informace o tom, jaké ETL rutiny realizovaly vložení konkrétního záznamu do pilotního DW, rozvětvení jeho toku na nové a upravené záznamy, získání počtu nových a upravených záznamů a uložení dat do pilotního datového skladu.

Ve většině případů z praxe je schéma datového toku v závislosti na specifických potřebách transformace mnohem složitější, proto po načtení dat z ODS může být data flow task rozšířen o libovolný počet dalších prvků, které provádějí pokročilejší transformace a čištění dat.

Datový tok jednotlivých kroků ETL pro transformaci a nahrání dat do jedné z tabulek dimenzí pilotního DW je zobrazen na obrázku č. 23.



Obrázek č. 23 Ukázka ETL rutin transformace a nahrání dat do pilotního DW – data flow (zdroj: autor)

Ve fázi nahrávání dat do tabulek velmi záleží na tom, které tabulky budou naplněny dříve a které později. Obecně platí, že tabulky dimenzí se plní nejdříve a teprve pak tabulky faktů. Dále bylo nutné pro procentuální výpočet aktuálně poskytnuté péče do faktové tabulky pilotního datového skladu ručně doplnit další řádek, který obsahuje informaci týkající se maximální úhrady poskytnuté péče ze strany pojišťoven za minulá referenční období, tj. zpětně za čtyři roky.

Fakta_pece	
«column»	
*PK	ID_pece: int
*FK	Pacient_ID: int
*FK	ID_datum: int
*FK	ID_vykonu: int
*FK	ID_oddeleni: int
	Objem_pece: int
	OPI: int
	Pocet_vykonu: int

Obrázek č. 24 Doplnění atributu OPI do faktové tabulky (zdroj: autor)

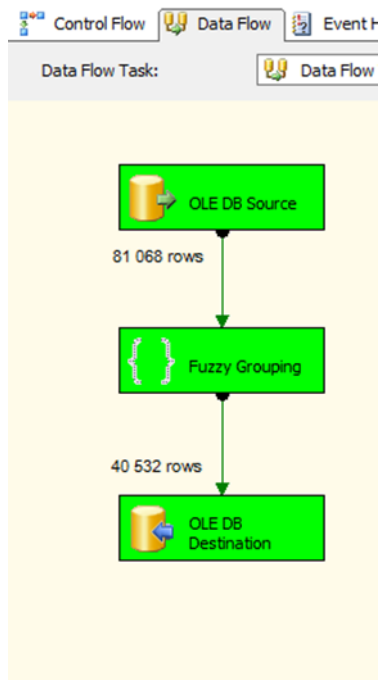
Pro čištění atributu *Zamestnani* v tabulce dimenzí *Pacient* byl použit proces s použitím funkcionality *Fuzzy Grouping Transformation* v *Integration Services*. Tento proces se

využívá k seskupení podobných řádků ve zdrojovém datovém souboru a zároveň i identifikuje řádky dat, která mohou být duplicitní.

219	část. ID	35
220	část. ID+ OSVČ	1
221	část. inv. důchod	1
222	část. inv. důchod	2
223	část. inv. důchodce	1
224	část. inv.důchodce	1
225	část. inval. důchod	3
226	část. inval. důchodc	1
227	část. inval.důchod	1
228	část. inval.důchod	1
229	část. invalid. důcho	1
230	část.ID	2
231	část.ID	3
232	část.inv. důchod	1
233	část.inv.důch.	3
234	část.inv.důchod	2
235	část.inv.důchodce	1
236	část.inv.důchodce	1
237	část.inval. důchodce	1
238	část.inval.důchod	1
239	část.invalid.důchod	1

*Obrázek č. 25 Ukázka některých duplicit v atributu Zamestnani (zdroj: autor)*

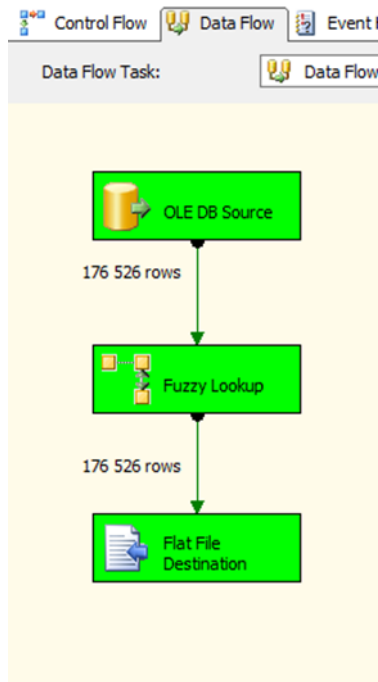
Atribut *Zamestnani* je do tabulky ve zdrojové databázi NIS zadáván ručně a tím při různém uvedení názvu zaměstnání vzniká velké množství chyb, které je potřeba pro budoucí analýzy a reporty co nejvíce eliminovat. Nejdříve byl pro kontrolu výstupu z tohoto procesu nastaven jako cíl výsledku externí flat soubor a teprve poté proběhla změna v databázi pilotního DW tak jak je zobrazeno na obrázku č. 26.



Obrázek č. 26 Použití Fuzzy Grouping Transformation pro vyčištění atributu *Zamestnani*  
(zdroj: autor)

Po ukončení tohoto procesu byl snížen počet různě uvedených hodnot atributu *Zamestnani* o 50 % a po jeho opakovaném provedení byl konečný celkový počet různých zaměstnání 5 815. Pro potřeby podrobnějších analýz by bylo nutné tento atribut ještě ručně zkontrolovat a sjednotit uvedené názvy zaměstnání se stejným významem (podnikatel = OSVČ = osoba samostatně výdělečně činná).

V *Integration Services* byla použita i další funkcionality pro vyčištění názvu měst u adresy v tabulce dimenzí *Pacient*. *Fuzzy Lookup Transformation* porovnává záznamy mezi zdrojovou a referenční tabulkou a výsledek podobnosti ukládá do externího souboru. Pro porovnání názvů měst a obcí byl použit veřejně dostupný číselník z databáze Ministerstva vnitra České republiky, který byl před samotným použitím mírně upraven (odstraněny řádky s názvy krajů a zastupitelských úřadů České republiky tzn. velvyslanectví v zahraničí a duplicit).



Obrázek č. 27 Použití Fuzzy Lookup Transformation pro čištění atributu Mesto  
(zdroj: autor)

Výsledkem tohoto čištění byla 100% shoda téměř 99 % záznamů, zbývající 1 % záznamů muselo být podrobena ruční revizi. Toto 1 % bylo do zdrojové databáze NIS importováno z ambulantního programu PC DOKTOR, který byl v minulosti využíván na detašovaných pracovištích společnosti MNDK a ve kterém byly, na rozdíl od databáze NIS názvy obcí a měst zadávány ručně lékařem nebo sestrou.

## 7.5 Implementace pilotního DW

Poslední fází při vývoji pilotního datového skladu je jeho implementace, která by se měla skládat z vytvoření struktur pilotního DW, jeho naplnění daty a případného doladění zjištěných chyb.

Nezbytné struktury pro provoz datového skladu jsou struktury ODS a pilotního DW (tabulky, funkce atd.), ETL rutiny, struktury OLAP (kostky, databáze, analýzy apod.) a sady reportů. K zajištění vytvoření potřebných struktur pilotního DW a OLAP databází lze využít XMLA<sup>22</sup> a SQL skripty. Výhodou programu Enterprise architekt je možnost

<sup>22</sup> Skripty vytvořené v aplikaci Analysis Services.

generování skriptů SQL pro vytvoření jednotlivých tabulek databáze. Příklad takto vygenerovaného scriptu pro vytvoření tabulky *Dimenze\_pacient*<sup>23</sup> je uveden níže.

```
CREATE TABLE Dimenze_pacient (
    Pacient_ID int NOT NULL,
    Jmeno nvarchar(18),
    Prijmeni nvarchar(20),
    Rodne_prijmeni nvarchar(20),
    Rodne_cislo int NOT NULL,
    Cislo_pojistence int NOT NULL,
    Pojistovna_kod smallint,
    Pojistovna_nazev nvarchar(30),
    Mesto nvarchar(25),
    ...
    Zamestnani nvarchar(25),
    Datum_navstevy smalldatetime,
    Zacatek_hospitalizace smalldatetime,
    Konec_hospitalizace smalldatetime
)
;
ALTER TABLE Dimenze_pacient
    ADD CONSTRAINT UQ_Dimenze_pacient_Pacient_ID UNIQUE
(Pacient_ID)
;
ALTER TABLE Dimenze_pacient ADD CONSTRAINT PK_Dimenze_pacient
    PRIMARY KEY CLUSTERED (Pacient_ID)
;
```

Stejným způsobem byly vygenerovány scripty a vytvořeny i ostatní tabulky ODS a pilotního DW spolu s jednotlivými vazbami mezi nimi. ETL „balíčky“ rutin jsou ve většině případů nakopírovány přímo na server, který obsahuje pilotní DW a následně je prostřednictvím aplikace *SQL Server Agent* vytvořen *job* a zajištěno pravidelné spouštění těchto ETL rutin a aktualizování údajů k OLAP analýzám.

Před načtením dat ze zdrojových systémů do ODS, provedením jejich transformace, očištění a následného nahrání do pilotního DW bylo ještě nutné upravit atribut *Objem\_pece* ve faktové tabulce, protože tento atribut není získáván ze zdrojových systémů, ale je dopočítán přímo databází pilotního datového skladu. S využitím funkčnosti SQL serveru (konkrétně s *computed column*), která umožňuje využít pro výpočet zvoleného atributu jakýkoli jiný atribut z příslušné tabulky, byl atribut *Objem\_pece* upraven následujícím způsobem:

```
[Objem_pece] AS (([OPI2017]/([OPI2016] * 1,092)) *100)
```

---

<sup>23</sup> Pro svou délku byl tento script zkrácen.



Atribut *Objem\_pece* je tedy fyzicky uložený ve faktové tabulce a je aktualizován vždy, když jsou aktualizované také ostatní atributy, na kterých tento vypočítaný atribut závisí.

Tímto posledním krokem bylo vytvoření pilotního datového skladu ve společnosti MNDK ukončeno a data v něm obsažená mohla být využita pro potřeby vytvoření uživatelských reportů a podrobnějších analýz (OLAP).

## 8 Návrh uživatelských reportů

Pro praktické ukázky uživatelských reportů ve spojení s pilotním datovým skladem byla použita aplikace *Reporting Services*, pomocí které byly všechny reporty vytvořeny.

Nástroje a aplikace pro vytváření reportů, se kterými ve většině případů pracují běžní uživatelé, by měly mezi ostatními nástroji Business Intelligence patřit k těm jednodušším. Aplikace *Reporting Services* s jejím přehledným GUI<sup>24</sup> vychází těmto uživatelům vstříc a vytvoření různých reportů je tak relativně snadné. Samozřejmě i zde se předpokládá alespoň základní uživatelská znalost práce s aplikacemi typu *drag and drop*, vytvářením grafů nebo kontingenčních tabulek.

Jako ukázka konečných reportů byly vybrány dva příklady, jejichž návrhy byly vzneseny vedoucími pracovníky organizačních jednotek a top managerem společnosti MNDK na poradě vedení dne 24. 3. 2017.

Prvním reportem byl návrh na zjištění vytiženosti lékařů – operatérů a asistentů jednotlivých oddělení z důvodu potřeby optimalizace práce na ambulantních pracovištích a lůžkové části oddělení. V případě nízkého počtu vykázaných výkonů, a tedy i času stráveného na operačním sále, spolu se zjištěním krátké pracovní doby na ambulanci daného oddělení, by mohla být u daného lékaře podstatně změněna doba ambulantní služby se zachováním operativní činnosti. Naopak při nadměrném vytižení lékaře – operatéra a existenci vhodného zástupu, by bylo z důvodu zachování co nejvyšší úrovně péče a bezpečnosti pacienta více než vhodné rozdělit operativu mezi tyto dva lékaře nebo přesunout „jednodušší“ operace na lékaře s nižší atestací, ale s potřebnou kvalifikací a oprávněním.

Na obrázku č. 28 je zobrazen první uživatelský report, který zobrazuje lékaře – operatéry se stejnou úrovní atestace, kteří mají nejmenší a největší počet operací od začátku roku 2013 do konce roku 2016. Z tohoto reportu jasně vyplývá, že jeden z lékařů zajišťuje jako hlavní operatér téměř čtvrtinu všech operačních zákroků vykázaných v odbornostech 501 a 5H1, zatímco druhý jen 13 %. Zajímavá je také položka maximálního a minimálního počtu operací provedených těmito lékaři za jeden měsíc v celém sledovaném období.

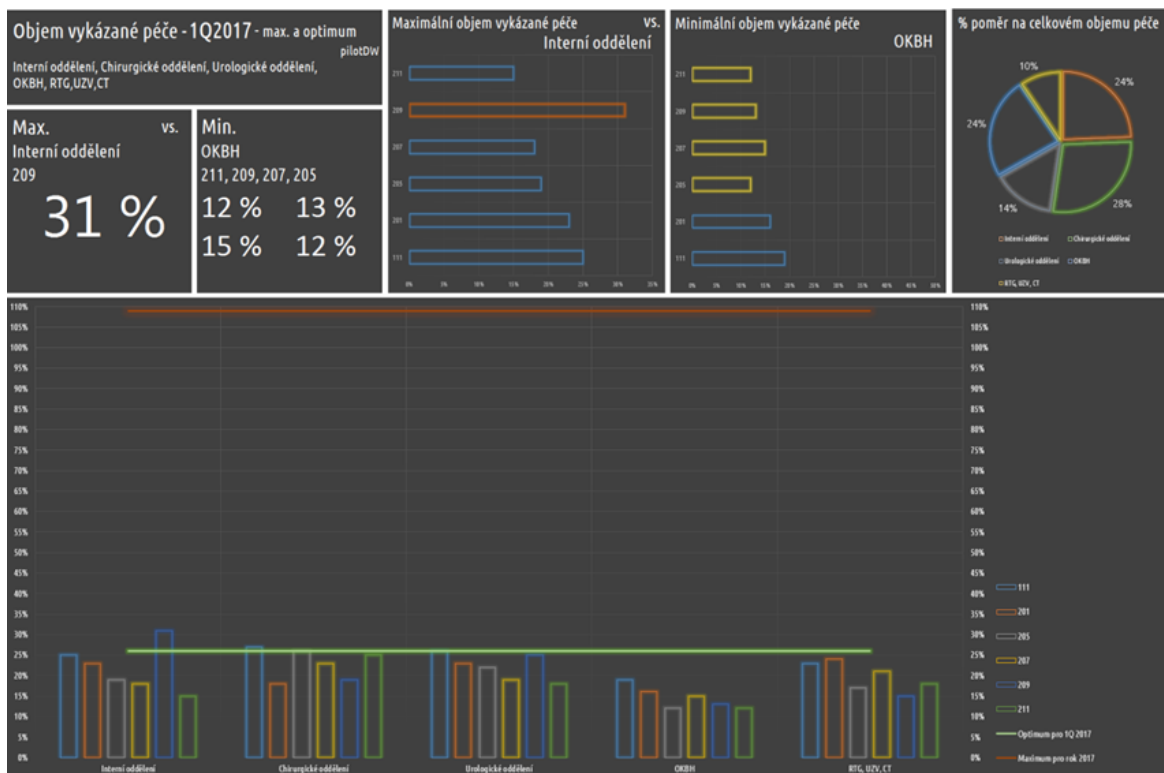
Pro znázornění kolísání počtu provedených operací tento report obsahuje také graf, na kterém je patrný nízký počet operačních zákroků provedených v sedmém měsíci každého

---

<sup>24</sup> GUI – Graphical User Interface (grafické uživatelské rozhraní)

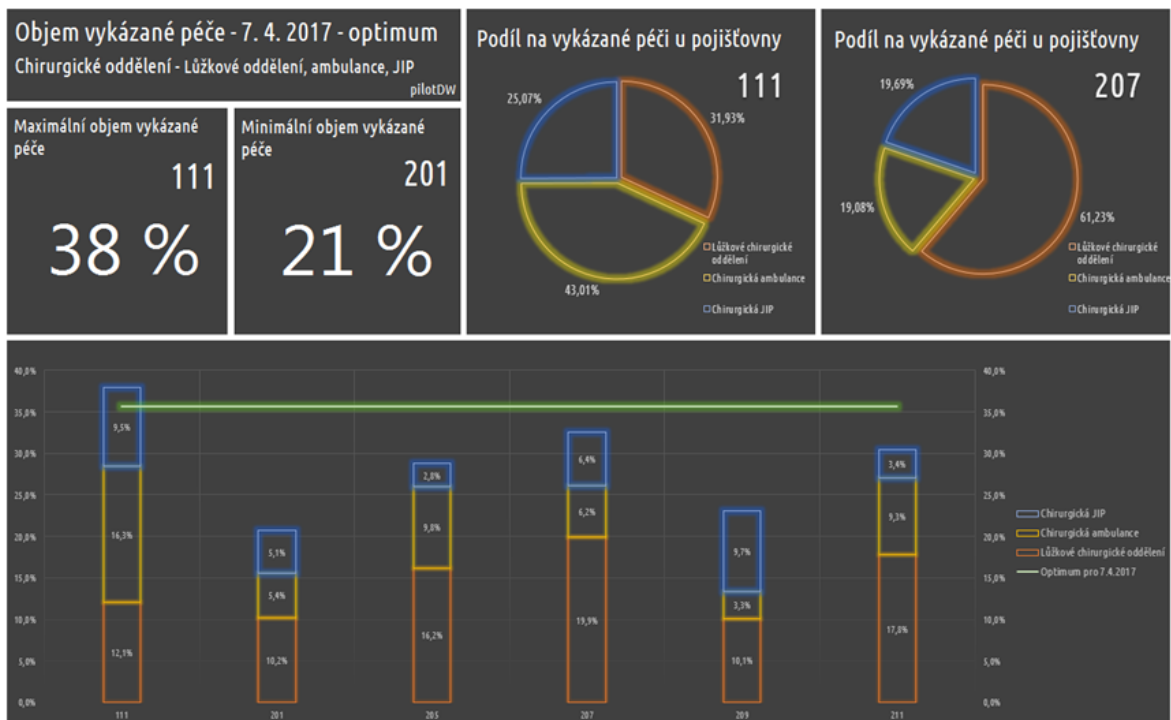


hospitalizace) pracovištích. Takto upravený uživatelský report pro potřebu primáře chirurgického oddělení je zobrazen na obrázku č. 30.



Obrázek č. 29 Druhý uživatelský report (zdroj: autor)

Možnost exportu takto vytvořených reportů do jiných programů (např. MS Excel nebo MS Word) ve formě kontingenční tabulky napojené na server pilotního DW a další práce s nimi je velkou výhodou při přípravě prezentací zjištěných údajů pro potřeby vedení společnosti MNDK. Další možností je umístění a publikování těchto reportů na webovém serveru nebo zasílání jednoho předem definovaného typu reportu založeného na denní bázi určité skupině uživatelů, kteří ho vyžadují.



Obrázek č. 30 Report upravený pro potřebu primáře chirurgického oddělení (zdroj: autor)

Případné další reporty nebo analýzy by mohly obsahovat např. maximální, minimální a průměrnou dobu hospitalizace jednotlivých skupin diagnóz v tomto roce, počet pacientů s určitou diagnózou nebo kombinací diagnóz za sledované období, pacienty s nejvyšším objemem poskytnuté péče nebo procentuální podíl jednotlivých lékařů na celkovém objemu vykázané péče na příslušném oddělení.

## 9 OLAP analýza

Za účelem praktického využití pilotního datového skladu ve spojení s analytickými nástroji byla v této fázi práce vytvořena multidimenzionální databáze sloužící jako zdroj pro OLAP analýzu. V první části této kapitoly je popsáno vytvoření MDD (OLAP kostky) a ve druhé jsou znázorněny možnosti OLAP nástrojů uvedením ukázky analýz.

### 9.1 Vytvoření multidimenzionální databáze

Pro vytvoření OLAP kostky (MDD), která se skládá z několika částí, byla použita aplikace *Analysis Services*. První částí vytvoření OLAP kostky je volba datového zdroje, který slouží jako základ pro její vytvoření. V tomto případě se samozřejmě jednalo o již dříve vytvořený pilotní datový sklad. Druhou částí je výběr struktury MDD na základě požadavků na analyzování dat podle místa vykázaní výkonů (oddělení MNDK) a věku pacienta v závislosti na čase. Jako součást struktury datové kostky, včetně faktové tabulky *Fakta\_pece*, byly vybrány dimenze *Dimenze\_vykony*, *Dimenze\_oddeleni*, *Dimenze\_pacient* a *Dimenze\_cas*. Dimenze obsahující informace o diagnózách a lékařích byly v tomto případě vynechány.

Další částí vytvoření MDD je návrh multidimenzionální databáze, který se skládá z výběru hodnot faktové tabulky pro analyzování daných parametrů a z výběru vhodných atributů a hierarchií dimenzionálních tabulek, kterými se bude s OLAP kostkou manipulovat. Z faktové tabulky byly vybrány atributy *Objem\_pece* a *Pocet\_vykonu*, jako hierarchie byla v tabulce *Dimenze\_cas* zvolena posloupnost *Rok*, *Kvartal* a *Mesic*. U tabulky *Dimenze\_oddeleni* jsou navazující atributy *Primariat*, *Nazev\_oddeleni* a *Specifikace\_oddeleni* a u tabulek *Dimenze\_pacient* a *Dimenze\_vykony* byly vybrány atributy *Vek*, *Kategorie\_vykonu*, *Kod\_vykonu* a *Nazev\_vykonu*.

Tímto způsobem navržená OLAP kostka je pomocí aplikace *Analysis Services* fyzicky vytvořena postupným spuštěním procesů *Process* (kontrola, zda je kostka správně navržená) a *Deploy* (vytvoření a odeslání kostky pod správu analytického serveru) v záložce *Build*.

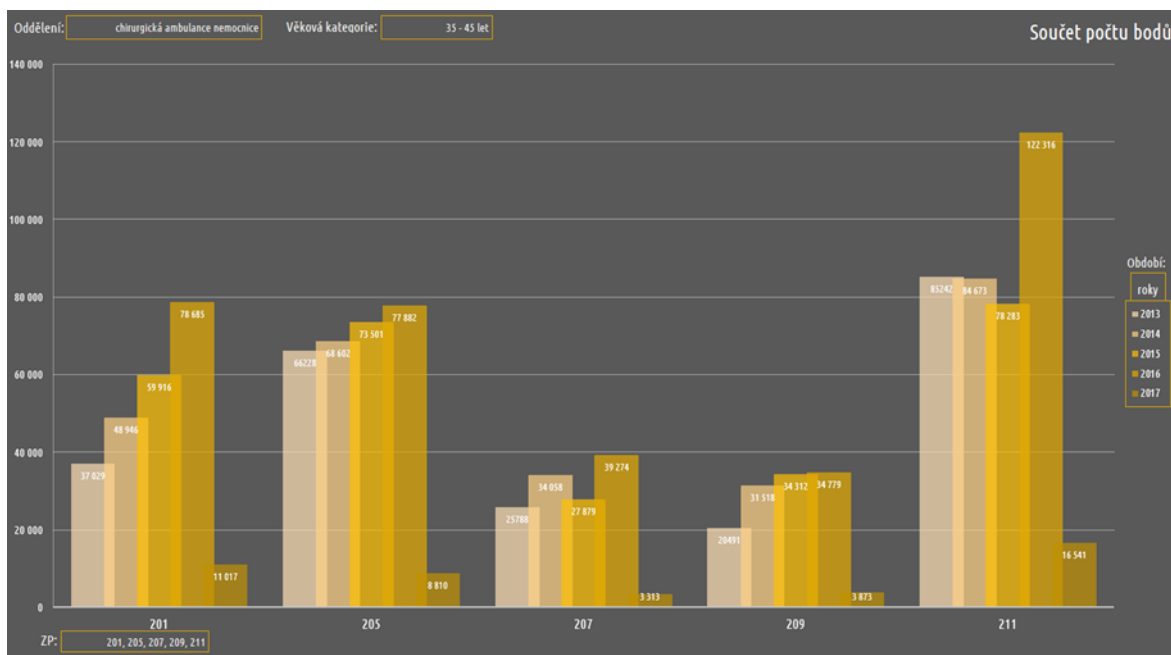
## 9.2 Analýza

S vytvořenou OLAP kostkou je možné pracovat přímo v prostředí *Analysis Services*, ale všeobecně vhodnějším programem pro OLAP analýzu je často označován *Microsoft Excel*, který je z tohoto důvodu použit pro její prezentaci i zde.

Sešit ve formátu Excel 2007 (xlsx), který obsahuje několik listů, je konečným výstupem této části práce. Každý list obsahuje analýzu zaměřující se na různé cíle spolu s odlišnými možnostmi prezentace výsledků dotazování. Hlavním záměrem bylo představit některé z možností OLAP analýzy v multidimenzionální databázi, jako je nabídka prohlížení dat na základní úrovni nebo naopak na vysoké úrovni agregovaných dat, spolu s operacemi spojenými s OLAP kostkou (*drill – up, slice*).

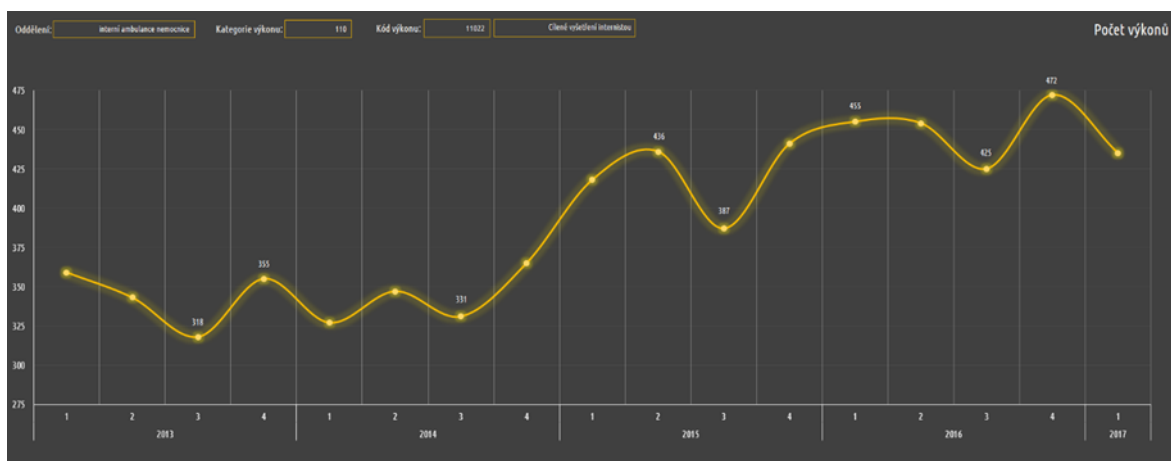
První list obsahoval analýzu součtu počtu bodů za vykázané výkony v závislosti na věku a pojišťovně pacienta, období (ukázka této analýzy z důvodu přehlednosti zobrazuje jen agregace za jednotlivé roky) a oddělení vykázaní výkonů. Tato analýza umožňovala operaci *drill down/up* na základě prohlížení počtu bodů a objemu poskytnuté péče na úrovni primariátů a jejich dílčích oddělení, stejně jako možnost změny úrovně období vykázaní, až na jednotlivé měsíce nebo volbu jiné věkové kategorie pacienta.

Na druhém listu byla výše zmíněná analýza graficky zobrazena v podobě grafu, který se dynamicky měnil v závislosti na parametrech vybraných uživatelem. Graf č. 1 zobrazuje analýzu součtu bodů za vykázané výkony na oddělení chirurgické ambulance v MNDK u vybraného rozsahu věku pacienta (35–45 let) s rozdělením na jednotlivé pojišťovny v roční posloupnosti. Důvodem nízkých hodnot u roku 2017 je skutečnost, že obsahují data pouze za první čtvrtletí tohoto roku.



Graf č. 1 První ukázka OLAP analýzy (zdroj: autor)

Třetí list této analýzy obsahoval informace o kategorii vykázaného výkonu s možností určení, kde byl daný výkon vykázan. Jako parametr pro operaci *slice* je zde zvoleno oddělení, které je následně zobrazeno v kontextu s kategorií, kódem a názvem výkonu a také obdobím vykázaní (zde nastaveny kvartály jednotlivých let). Metoda, při které je zvolen jeden parametr na nejnižší úrovni (v tomto případě oddělení interní ambulance umístěné v areálu nemocnice) a následně je analyzován na základě zbývajících dimenzí, tak zcela odpovídá operaci *slice*. Na čtvrtém listu, stejně jako na druhém, byla tato analýza zobrazena v uživatelsky přívětivější formě, tedy jako graf.



Graf č. 2 Druhá ukázka OLAP analýzy (zdroj: autor)



## 10 Testování navrženého řešení

Pro testování OLAP v multidimenzionální databázi pilotního datového skladu byl použit v teoretické části zmíněný FASMI test. V některých případech byly pro porovnání výkonosti OLAP a transakční databáze vytvořeny SQL scripty, které se ve zdrojových transakčních databázích NIS a LIS dotazovaly na stejné parametry jako OLAP v MDD pilotního DW.

*Fast* – výsledky dotazování byly uživateli systémem doručeny prakticky po kliknutí myši nebo ihned po nastavení vybraných ukazatelů. Jen v případě zahrnutí všech agregovaných parametrů (např. součet počtu bodů), každého oddělení a vysoké úrovně granularity trvalo doručení výsledku dvacet pět sekund. Pro porovnání OLAP a transakční databáze SQL byla zvolena analýza agregace vybraného výkonu na urologických ambulancích společnosti MNDK za poslední čtyři roky s úrovní granularity jednotlivých dnů. Tato analýza u OLAP trvala přibližně patnáct sekund, zatímco dotaz v transakční databázi byl po deseti minutách, nadměrném zatížení serveru, vyřízením šesti telefonátů od nespokojených uživatelů a nemožnosti běžné práce se systémem NIS zastaven a znovu spuštěn v nočních hodinách pomocí naplánovaného *jobu*. Samotný výsledek SQL dotazu byl získán po téměř dvou hodinách dotazování se stejným výsledkem jako u OLAP systému. Systém pilotního datového skladu tak úspěšně prošel testem rychlosti s výbornými výsledky oproti transakční databázi.

*Analysis* – výstupy OLAP analýz a také samotný soubor s těmito analýzami, který byl určený pro předem vybrané uživatele, jsou v dostatečně pochopitelném stavu a tito uživatelé neměli z definováním nových výpočtů v analýzách problémy. Uživatel nepracující s programy Microsoft Office, a zvláště s kontingenčními tabulkami by měl ze začátku s pochopením a používáním OLAP analýz značné potíže, které ale lze praktickým zaškolením spolu s vysvětlením jednotlivých parametrů těchto analýz do značné míry eliminovat.

*Shared* – systém pilotního datového skladu umožňuje sdílení získaných informací napříč celým vedením společnosti MNDK v podobě vytvořených uživatelských reportů a OLAP analýzy. V případě rozšíření celého systému pilotního DW na další oblasti a jeho provázání s některými výše představenými externími systémy by bylo nutné nastavit práva víceuživatelského přístupu k zápisu, která jsou v pilotním DW nastavena tak, aby údaje mohl měnit pouze vedoucí pracovník, a to pouze u jemu příslušících organizačních

jednotek. Při změně každého údaje je v databázi pilotního DW zaznamenáno kým, kdy a u jakého údaje byla provedena změna.

*Multidimensional* – klíčový požadavek systémů OLAP a tedy i poskytování multidimenzionálního koncepčního pohledu na data a možnost používat vícenásobné hierarchie (čas, výkony, oddělení) byl u systému pilotního DW splněn. Systém pilotního datového skladu tedy naplňuje požadavky multidimenzionální databáze.

*Information* – všechny informace získané z pilotního DW za použití OLAP analýz, uživatelských reportů a dalších náhodně vybraných parametrů (např. součet bodů za výkon 51023<sup>25</sup> a jeho počet na detašovaném pracovišti chirurgické ambulance v období od 9. do 20. ledna roku 2016 a 2017 u mužů nad 40 let) byly porovnány s daty, která byla výsledkem SQL dotazování v transakčních databázích NIS a LIS. Základními testy byl také počet záznamů v tabulkách pilotního DW a agregovaných hodnot u zvolených metrik v MDD. Oba testy byly realizovány porovnáním získaných hodnot s hodnotami ve zdrojových databázích. Takto získané údaje se v každém z případů porovnání shodovaly, a proto mohou být údaje obsažené v databázi pilotního DW prohlášena za kvalitní, relevantní a správná.

Testováním navrženého řešení bylo zjištěno, že pilotní datový sklad vytvořený pro potřebu společnosti Městská nemocnice, a.s. naplňuje veškeré výše zmíněné požadavky FASMI testu, a proto může být použit jako dobrý základ v případě vytváření plnohodnotného datového skladu pro tuto společnost.

---

<sup>25</sup> 51023 – Kontrolní vyšetření všeobecným chirurgem

## 11 Hodnocení navrženého řešení

Hodnocení řešení pilotního datového skladu je v této kapitole rozděleno na dvě části. První je zaměřena na zhodnocení funkčních přínosů vytvořeného pilotního DW. Druhá část se věnuje ekonomickým přínosům pro společnost MNDK.

### 11.1 Funkční přínosy

Pomocí sledování aktivity v *SQL Server Management Studiu* a určením obvyklého objemu a typů činnosti systému bylo zjištěno, že nasazení systému pilotního datového skladu a z něho vycházejících uživatelských reportů a analýz MDD (OLAP kostky), napomohlo k snížení dotazování pomocí SQL skriptů zdrojové databáze NIS o více než 60 %. Tím byl také zajištěn vyšší výkon tohoto systému, a to zejména jeho plynulejší chod při větším zatížení (např. pravidelná reindexace databáze nebo příprava podkladů pro fakturaci vykázané péče pojišťovně).

Důvodem tohoto snížení byl především fakt, že téměř všichni vedoucí pracovníci jednotlivých organizačních jednotek prakticky přestali používat do té doby pro ně vytvořené procedury, které získávaly potřebná data přímo ze zdrojové databáze, a místo toho začali používat vytvořené uživatelské reporty a pracovat s OLAP kostkou v programu MS Excel.

Za další funkční přínosy řešení pilotního DW může být považováno centrální uložení dat ze všech zdrojových systémů v jedné pevné a logické struktuře, kterou je zajištěna snadná interní i externí dostupnost uložených dat. Pravidelná aktualizace dat v předem stanovených intervalech zaručuje zvýšení jejich vypovídající hodnoty a kvality. Představuje tedy další funkční přínos vytvořeného pilotního DW. Stejně tak jako spojování dat provozních systémů (např. pro získání najednou všech dat o pacientovi, i když jsou rozmístěna mezi různými systémy), tvorba tzv. křížových dotazů nebo rychlá on-line dosažitelnost dat. Podstatným přínosem tohoto řešení je rovněž poměrné odstranění souběžného ručního zpracování dat pracovníky, dostupnost dat „on demand“<sup>26</sup> kdykoli a kdekoli nebo v této práci již několikrát zmíněná eliminace duplicit dat.

---

<sup>26</sup> On demand odkazuje na služby, které jsou zaměřeny na okamžité uspokojení potřeb uživatele a bezprostřednost jejich použití.

## 11.2 Ekonomické přínosy

Jedním z největších ekonomických přínosů vytvoření pilotního DW je možnost sledování vývoje vykázaného objemu péče u jednotlivých pojišťoven na všech dílčích pracovištích po celý rok. To umožňuje zajistit optimální plnění její výše tak aby nedošlo k jejímu překročení (složitá dokazování a časté neuhrazení této péče ze strany zdravotních pojišťoven) nebo naopak k nedostatečnému plnění (vrácení části obdržených záloh zpět pojišťovnám). Tím by bylo docíleno zlepšení hospodářského výsledku bez snížení kvality poskytované péče a bezpečnosti pacienta v dalších obdobích.

Analýzou vykázaných HVLP, IVLP a PZT lze odhalit nedostatky ve vykazování, kterými mohou být např. rozdíly jejich cen od nejnižší možné ceny na trhu, alternativy prostředků nebo nepřiměřené množství vykázaných přípravků. Náprava těchto chyb, tj. omezení plýtvání nebo používání levnějších, ale stejně kvalitních přípravků, zajistí snížení variabilních nákladů. V případě snížení těchto nákladů třeba jen o 2 % by na konci roku byla ušetřena částka přesahující půl milionu korun, která může být použita ke zvýšení úrovně poskytované péče.

Podrobná znalost produktivity práce lékařů může sloužit jako podklad pro následnou optimalizaci jejich pracovní doby (rozdělení hodin strávených na jednotlivých ambulantních a lůžkových pracovištích). Očekávaným přínosem je také zkvalitnění plánování operačních programů (rozšíření nebo zkrácení operativy jednotlivých lékařů-operatérů, koordinace využívání operačních sálů). Všechny tyto aktivity pak směřují k celkovému zvýšení efektivity činnosti zdravotnického zařízení.

Dalším ekonomickým přínosem je především ušetření času vedoucích pracovníků u vytváření potřebných přehledů. Dříve byli tito pracovníci nuceni čekat na výsledek dotazování pomocí SQL procedury a následně museli takto získaná data v programu MS Excel zdoluhavě upravovat pro své specifické potřeby. Poskytnutím přístupu k OLAP kostce a vytvořením uživatelských reportů byl čas strávený vytvářením původních přehledů podstatně zkrácen a vedoucí lékaři se tak mohli více věnovat své primární činnosti, kterou je léčebná (terapeutická) a preventivní péče.

S ukončením používání původních vytvořených procedur klesl současně počet žádostí vedoucích pracovníků na IT oddělení o pomoc při úpravě získaných dat nebo o změnu využívaných procedur. Tím byl ušetřen také čas pracovníků IT oddělení strávený při řešení těchto žádostí změřený na dva člověkodny měsíčně.

## 12 Závěr

Současně s tím, jak se mění ekonomická situace a podle způsobu zvětšování objemu firemních dat stále více společností potřebuje z těchto dat extrahovat potenciale využitelné informace pro podporu rozhodování. Jedním ze systémů, který je nezbytnou součástí Business Intelligence, a který toto umožňuje je také datový sklad.

V první kapitole této diplomové práce byly nejprve popsány možné přístupy k vytvoření datových skladů s uvedením jejich možných výhod i nevýhod. Hlavními postavami, od kterých byly čerpány teoretické poznatky z oblasti datových skladů, jsou W. Inmon a R. Kimball. Dále byly porovnány způsoby ukládání dat v transakčních a multidimenzionálních databázích, u kterých bylo zjištěno, že jedinečnou strukturou DW zaručuje jednoduchost jeho pochopení pro uživatele při vytváření všech reportů, a proto je pro využití DW nezbytná. Datový sklad je také jediný možný zdroj k vytvoření plnohodnotné MDD, která je využívána pro OLAP analýzu.

Na základě praktických zkušeností z oboru zdravotnictví a získaných znalostí byla navržena metodika pro praktickou část této práce, která popisuje jednotlivé nástroje a kroky k dosažení stanoveného cíle návrhu datového skladu v prostředí zdravotnické instituce.

V praktické části práce byl navržen a implementován model pilotního datového skladu se všemi jeho součástmi (ODS, ELT rutiny atd.) pro společnost MNDK. Tento pilotní DW byl následně použit pro vytvoření různých druhů uživatelských reportů a multidimenzionální databáze, ve které byly provedeny série analytických dotazů. Současně zde byl také kladen důraz na použití moderních aplikací i běžně rozšířených kancelářských programů jako je např. Microsoft Excel.

Lze konstatovat, že tato práce splnila výše stanovený cíl a dále bude záležet jen na uživateli a úzkém vedení společnosti, zda bude tento systém, který přináší velké množství výhod a možností, nadále používán a případně rozšířen o nové funkční oblasti.

## 13 Seznam použité literatury

1. ANUPINDI, Nagesh V., 2005. Inmon vs. Kimball - An Analysis. *Nagesh V. Anupindi* [online]. Denver: Nagesh [cit. 2017-03-07]. Dostupné z: <http://www.nagesh.com/publications/technology/173-inmon-vs-kimball-an-analysis.html>
2. ARBES Technologies | Zdravotnictví, 2016. *ARBES Technologies | Softwarová řešení pro finanční instituce, obchodní společnosti a zdravotnické subjekty* [online]. Praha: ARBES Technologies [cit. 2017-03-30]. Dostupné z: <http://www.arbes.com/zdravotnictvi/>
3. ČECH, Pavel a Vladimír BUREŠ, 2009. *Podniková informatika*. Hradec Králové: Gaudeamus. ISBN 978-80-7041-479-8.
4. DRESNER, Howard, 2006 cit. podle MARTENS, China, 2006. BI at age 17. *IT news, careers, business technology, reviews* [online]. Framingham: Computerworld [cit. 2017-03-15]. Dostupné z: <http://www.computerworld.com/article/2554088/business-intelligence/bi-at-age-17.html>
5. EGGEN, Dan a Griff WITTE, 2006. The FBI's Upgrade That Wasn't. *Washington Post: Breaking News, World, US, DC News & Analysis - The Washington Post* [online]. Washington: The Washington Post Company [cit. 2017-04-03]. Dostupné z: <http://www.washingtonpost.com/wp-dyn/content/article/2006/08/17/AR2006081701485.html>
6. FOLEY, John, 2012. FBI's Sentinel Project: 5 Lessons Learned. *InformationWeek News Connects The Business Technology Community* [online]. New York: UBM [cit. 2017-04-03]. Dostupné z: <http://www.informationweek.com/applications/fbis-sentinel-project-5-lessons-learned/d/d-id/1105637?>
7. FONS Akord | STAPRO, c2017. *STAPRO: Informace v ceně života* [online]. Pardubice: STAPRO [cit. 2017-03-30]. Dostupné z: <http://www.stapro.cz/produkty-fons/fons-akord/>
8. FONS Openlims | STAPRO, c2017. *STAPRO* [online]. Pardubice: STAPRO [cit. 2017-03-30]. Dostupné z: <http://www.stapro.cz/produkty-fons/fons-openlims/>

9. GÁLA, Libor, Jan POUR a Prokop TOMAN, 2006. *Podniková informatika: počítačové aplikace v podnikové a mezipodnikové praxi, technologie informačních systémů, řízení a rozvoj podnikové informatiky*. Praha: Grada. Management v informační společnosti. ISBN 80-247-1278-4.
10. Hlavní principy datových skladů a proces jejich vytváření, 2000. *SystemOnLine.cz* [online]. Brno: CCB [cit. 2017-02-28]. Dostupné z: <https://www.systemonline.cz/clanky/hlavni-principy-datovych-skladu-a-proces-jejich-vytvareni.htm>
11. HROCH, Michal a Pavel CACH, 2007. Business intelligence staví na datovém skladu. *SystemOnLine.cz - ekonomické a informační systémy v praxi* [online]. Brno: CCB [cit. 2017-03-14]. Dostupné z: <https://www.systemonline.cz/business-intelligence/business-intelligence-stavi-na-datovem-skladu.htm>
12. IBM Knowledge Center: Tabulky faktů, 2015. *IBM - United States* [online]. Weschester (New York): IBM [cit. 2017-03-23]. Dostupné z: [https://www.ibm.com/support/knowledgecenter/cs/SSHEB3\\_3.5.2/com.ibm.tap.doc/abp\\_data/r\\_fact\\_tables.html](https://www.ibm.com/support/knowledgecenter/cs/SSHEB3_3.5.2/com.ibm.tap.doc/abp_data/r_fact_tables.html)
13. INMON, William H., c2005. *Building the data warehouse*. 4th ed. Indianapolis, Ind.: Wiley. ISBN 07-645-9944-5.
14. KIMBALL, Ralph a Margy ROSS, c2002. *The data warehouse toolkit: the complete guide to dimensional modeling*. 2nd ed. New York: Wiley. ISBN 04-712-0024-7.
15. KIMBALL, Ralph a Margy ROSS, 2013. *The data warehouse toolkit: the definitive guide to dimensional modeling*. Third edition. Indianapolis, Ind.: Wiley Pub. ISBN 978-1-118-53080-1.
16. LABERGE, Robert, 2012. *Datové sklady: agilní metody a business intelligence*. Brno: Computer Press, 350 s. ISBN 978-80-251-3729-1.
17. LACKO, Ľuboslav, 2011. *1001 tipů a triků pro SQL*. Brno: Computer Press, 416 s. ISBN 978-80-251-3010-0.
18. LACKO, Ľuboslav, 2003. *Databáze: datové sklady, OLAP a dolování dat s příklady v Microsoft SQL Serveru a Oracle*. Brno: Computer Press, 486 s. ISBN 80-722-6969-0.

19. LUHN, H. P., 1958. A Business Intelligence System. *IBM Journal of Research and Development*. **2**(4), 314-319. DOI: 10.1147/rd.24.0314. ISSN 0018-8646. Dostupné také z: <http://ieeexplore.ieee.org/document/5392644/>
20. MUNDY, Joy, Warren THORNTHWAITE a Ralph. KIMBALL, c2011. *The Microsoft data warehouse toolkit: with SQL server 2008 R2 and the Microsoft Business intelligence toolset*. 2nd ed. Indianapolis, IN: Wiley Pub. ISBN 978-1-118-06793-2.
21. NOVOTNÝ, Ota, Jan POUR a David SLÁNSKÝ, 2005. *Business intelligence: jak využít bohatství ve vašich datech*. Praha: Grada. Management v informační společnosti. ISBN 80-247-1094-3.
22. PADRE, Andrei, c2010. OLAP Cubes: Data Visualization. *Online Analytical Processing* [online]. Massachusetts: Padre [cit. 2017-03-23]. Dostupné z: <https://apandre.wordpress.com/data/datacube/>
23. PENDSE, Nigel, c2001. What is OLAP? *Decision Support Systems Resources - DSSResources.COM* [online]. London: Pendse [cit. 2017-03-23]. Dostupné z: <http://dssresources.com/subscriber/password/papers/features/pendse04072002.htm>
24. POHL, Jakub, 2014. Městská nemocnice a.s. | Městská nemocnice a. s., Dvůr Králové. ŠTÁSTEK, Petr. *Nemocnice Dvůr Králové nad Labem | Městská nemocnice a. s., Dvůr Králové* [online]. Dvůr Králové nad Labem: Městská nemocnice [cit. 2017-03-30]. Dostupné z: <http://www.mndk.cz/spolecnosti-holdingu/mestska-nemocnice>
25. POUR, Jan, Miloš MARYŠKA a Ota NOVOTNÝ, 2012. *Business intelligence v podnikové praxi*. Praha: Professional Publishing. ISBN 978-80-7431-065-2.
26. PŘIBYSLAVSKÝ, Jiří, 2003. Máte kvalitní manažerský informační systém? *SystemOnLine.cz* [online]. Brno: CCB [cit. 2017-03-26]. Dostupné z: <https://www.systemonline.cz/clanky/mate-kvalitni-manazersky-informacni-system.htm>
27. RAINARDI, Vincent., c2008. *Building a data warehouse with examples in SQL Server*. Berkeley, CA: Apress ; Distributed to the book trade worldwide by Springer-Verlag New York. ISBN 978-1-59059-931-0



28. REYNOLDS, D. G., 2006. Kimball vs. Inmon..or, How to build a Data Warehouse. *IT Communities - Share Knowledge at Toolbox.com* [online]. Miami: Toolbox.com [cit. 2017-03-16]. Dostupné z: <http://it.toolbox.com/blogs/confessions/kimball-vs-inmon-or-how-to-build-a-data-warehouse-10987>
29. SLÁDEČEK, Martin, 2006. Manažerské informační systémy. *Computerworld.cz | Deník pro IT profesionály* [online]. Praha: IDG Czech Republic, a.s. [cit. 2017-03-26]. Dostupné z: <http://computerworld.cz/archiv/manazerske-informacni-systemy-24689>
30. STACKOWIAK, Robert., Joseph. RAYMAN a Rick. GREENWALD, c2007. *Oracle data warehousing and business intelligence solutions*. Hoboken, N.J.: Wiley Pub. ISBN 04-719-1921-7.
31. *Systémová integrace 2/2005: Nástroje business inteligence - - struktura a integrační charakter*, 2005. **12**(2). ISSN 1210-9479
32. Teorie relačních databází: Normalizace, 2007. *Manualy.net - tvorba webu, tutoriály, PHP, MySQL* [online]. Praha: Vebloud [cit. 2017-03-01]. Dostupné z: <http://www.manualy.net/article.php?articleID=13>

**Podklad pro zadání DIPLOMOVÉ práce studenta**

<b>PŘEDKLÁDÁ:</b>	<b>ADRESA</b>	<b>OSOBNÍ ČÍSLO</b>
Pohl Jakub	Kotkova 858, Dvůr Králové nad Labem	I1500768

**TÉMA ČESKY:**

Návrh datového skladu v prostředí zdravotnické instituce

**TÉMA ANGLICKY:**

Design of a data warehouse in environment of health care institution

**VEDOUcí PRÁCE:**

Mgr. Josef Horálek, Ph.D. - KIT

**ZÁSADY PRO VYPRACOVÁNÍ:**

Cílem práce je zmapovat aktuální vývoj a využitelnost datových skladů a na základě získaných informací navrhnout datový sklad pro specifické prostředí zdravotnické instituce.  
V teoretické části autor představí principy DWH, jejich dělí a přístupy k návrhu.  
V praktické části autor zpracuje případovou studii pro vytvoření a implementaci DWH v prostředí zdravotnické organizace s přihlédnutím ke specifickým potřebám a citlivosti dat.

**SEZNAM DOPORUČENÉ LITERATURY:**

KIMBALL, Ralph a Margy ROSS, c2002. The data warehouse toolkit: the complete guide to dimensional modeling. 2nd ed. New York: Wiley. ISBN 04-712-0024-7.  
INMON, William H., c2005. Building the data warehouse. 4th ed. Indianapolis, Ind.: Wiley. ISBN 07-645-9944-5.

**Podpis studenta:** .....

**Datum:** .....

**Podpis vedoucího práce:** .....

**Datum:** .....