UNIVERZITA PARDUBICE

FAKULTA EKONOMICKO-SPRÁVNÍ

BAKALÁŘSKÁ PRÁCE

2009 Dominik Sýkora

Univerzita Pardubice
Fakulta Ekonomicko-správní

Datový sklad nad IS STAG

Dominik Sýkora

Bakalářská práce
2009

Univerzita Pardubice
Fakulta ekonomicko-správní
Ústav systémového inženýrství a informatiky
Akademický rok: **2008/2009**

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Dominik SÝKORA**
Studijní program:	**B6209 Systémové inženýrství a informatika**
Studijní obor:	**Informatika ve veřejné správě**
Název tématu:	**Datový sklad nad IS STAG**

Zásady pro vypracování:

Cílem bakalářské práce je na základě RMD IS STAG navrhnout datový sklad, resp. datové tržiště.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

Novotný, O.,Pour, J., Slánský, D.. Business Inteligence: Jak využít bohatství ve vašich datech. Praha: Grada, 2005. 254 s. ISBN 80-247-1094-3.

Gála, L., Pour, J., Toman, P.. Podniková informatika. Praha: Grada, 2006. 484 s. ISBN 80-1278-4.

Date, C.J.. An Introduction to Database Systems. Boston: Addison-Wesley, [2002?]. 352 s.

Humphries, M., Hawkins, M. W., Dy, M. C.. Data Warehousing: Návrh a Implementace. Praha: Computer Press, 2002. 257 s. ISBN 80-7226-560-1.

Šimonová

Vedoucí bakalářské práce: **Ing. Stanislava Šimonová, Ph.D.**
Ústav systémového inženýrství a informatiky

Datum zadání bakalářské práce: **6. října 2008**

Termín odevzdání bakalářské práce: **1. května 2009**

doc. Ing. Renáta Myšková, Ph.D.
děkanka

L.S.

doc. Ing. Jiří Křupka, Ph.D.
vedoucí ústavu

V Pardubicích dne 6. října 2008

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.
Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 30. 4. 2009

Dominik Sýkora

Poděkování

Rád bych poděkoval především vedoucí práce Ing. Stanislavě Šimonové, Ph.D. za všechny rady a připomínky při zpracování bakalářské práce a stejně tak i za výborné vedení a dohlížení na tvorbu práce. Dále bych také chtěl poděkovat Ing. Ondřeji Pruskovi, Ph.D. a Ing. Miroslavu Koblížkovi za odborné rady ohledně relačního modelu IS STAG, které mi při tvorbě práci velmi pomohly.

Anotace

Práce se zaměřuje na návrh datového skladu, resp. datového tržiště nad informačním systémem STAG. Dále se zabývá problematikou normalizovaného a dimenzionálního modelování. Ukazuje výhody datových skladů z analytického hlediska a k modelování návrhu využívá modelovací nástroj case/4/0.

Klíčová slova

datový sklad; dimenzionální modelování; normalizace; relační databáze

Title

Data warehouse upon IS STAG

Annotation

The work focuses on projecting of data warehouse of data mart upon information system STAG. It also deals with the problematic of normalized and above all dimensional modeling. It points advantages of data warehouses from analytical point of view and it uses a tool case/4/0 to model the projection.

Keywords

data warehouse; dimensional modeling; normalization; relational database

Obsah

Seznam obrázků

Seznam tabulek

1 Úvod

Informační technologie jsou součástí téměř všech sfér lidských činností. V souvislosti s tím roste počet sledovaných údajů každé ziskové i neziskové společnosti, a tak narůstají nároky na objemy dat a jejich úložiště.

Data do systémů jsou získávána z různých aplikací, a jsou tedy ukládána v různých formátech, proto dochází často k problémům s kompatibilitou a s konzistencí dat. V obrovském množství takto ukládaných dat, která jsou dosti nepřehledná, může docházet k vytvoření chybných rozhodnutí, jež mohou být životně důležitá pro danou organizaci.

Významnou informační podporou pro každou organizaci je předpovídání a sledování trendu ekonomických i neekonomických veličin. Proto je třeba získávaná data archivovat (později se z těchto stávají historická data), ale přitom by měla být neustále k dispozici a vhodně předpřipravena pro tvorbu dotazů, výstupních sestav aj. Právě pro umožnění práce s obrovskými objemy dat se budují tzv. datové sklady či datová tržiště.

Data se získávají z informačních systémů, kde informačním systémem chápeme spojení člověka, technologie a metod, které slouží ke sběru, údržbě, zpracování a uchování dat za účelem jejich použití dalšími uživateli. Jedním z takových systémů je i IS STAG.

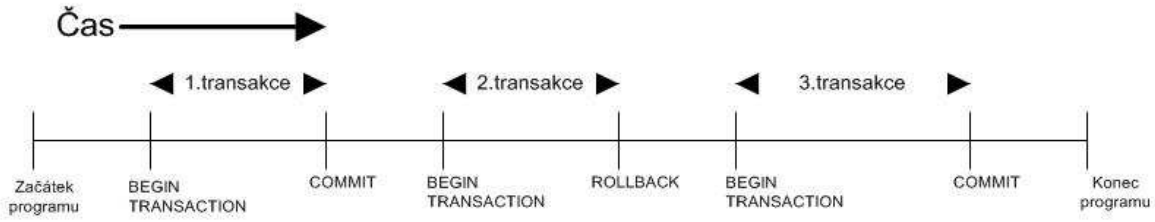
Cílem této práce tedy je na základě získaných poznatků provést analýzu RMD STAG a podle výsledku analýzy navrhnout datový sklad vyhovující tomuto informačnímu systému. Pro potřebu analýzy je třeba provést dimenzionální modelování, které bude charakterizováno i ve srovnání s modelováním normalizovaným.

2 Normalizované modelování pro transakční systémy

Databáze jsou v dnešní době realizovány především pomocí relačních modelů, ale pro analytické potřeby se stále více prosazují technologie business intelligence, resp. datových skladů. Je ovšem třeba si něco říci o normalizovaném modelování, aby byly lépe vidět výhody dimenzionálního modelování při využití v analytických úlohách.

2.1 Vymezení pojmů

Základním důvodem proč byly zavedeny databáze do praxe, je manipulace s transakcemi. Transakce je jednotkou práce. Skládá se z operací, které jsou vykonávány v sekvencích specifikovaných aplikacemi. Tyto sekvence začínají speciální operací BEGIN TRANSACTION a končí buď operací COMMIT a nebo operací ROLLBACK. COMIT se používá k označení úspěšného dokončení transakce a ROLLBACK naopak k označení neúspěšné transakce, jestliže se při běhu dostane transakce do nějakého výjimečného stavu (např. když nemůže být nalezen záznam). Slovo dokončení zde neznamená konec programu, ale pouze konec transakce, protože program může vykonávat několik transakcí za sebou. Schéma chodu programu znázorňuje obrázek 1.



Obrázek 1: Transakce a vykonání programu (zdroj: vlastní - zpracováno podle [2])

Dalším ze základních pojmů, především, v oblasti relačních databází a normalizovaného modelování je integrita. Termín integrita má v databázích význam jako přesnost, správnost nebo validita. Integrita zajišťuje, že data v databázích jsou správná, tedy že chrání databáze před nevalidními aktualizacemi. Nesprávnost dat může být způsobena chybami při vstupu dat, chybami na straně operátora či aplikačního programátora, selháními systému a dokonce i úmyslnou falzifikací dat. Falzifikace je ovšem spíše otázkou bezpečnosti. Když se mluví o integritě, předpokládá se, že je uživatel oprávněn vkládat data a že se snaží předcházet nesprávnosti dat. Pro zajištění integrity musí existovat tzv. *integrity subsystem* neboli podsystém

integrity, který má za úkol monitorovat transakce, detekovat porušení integrity a v případě porušení provést vhodné úkony (např. odmítnutí operace, nahlášení narušení integrity nebo i opravení chyby). Aby mohl podsystém integrity tyto funkce plnit, musí obsahovat sadu pravidel, která říkají, jakou chybu má hledat, kdy provést kontrolu a co dělat, jestliže je nalezena chyba. Existují různé druhy integrity, jako jsou doménová integrita, která znamená, že se na úrovni sloupců definují omezení na určitý datový typ nebo případně rozsah hodnot, referenční integrita, jež definuje vztah dvou tabulek pomocí cizích klíčů, či integrita entitní, která musí být definována v rámci každého relačního modelu, jelikož se jedná především o určení primárního klíče. [2]

2.2 Relační model dat

Relační model dat využívá matematickou teorii množin a predikátovou logiku, definuje způsob reprezentace dat, způsob jejich ochrany (integritní omezení) a možné operace nad daty. Pravidla relačního modelu publikoval Dr. E. F. Codd v r. 1970 v článku „A relations model of data for large shared databanks“. Hlavní principy jsou [12]:

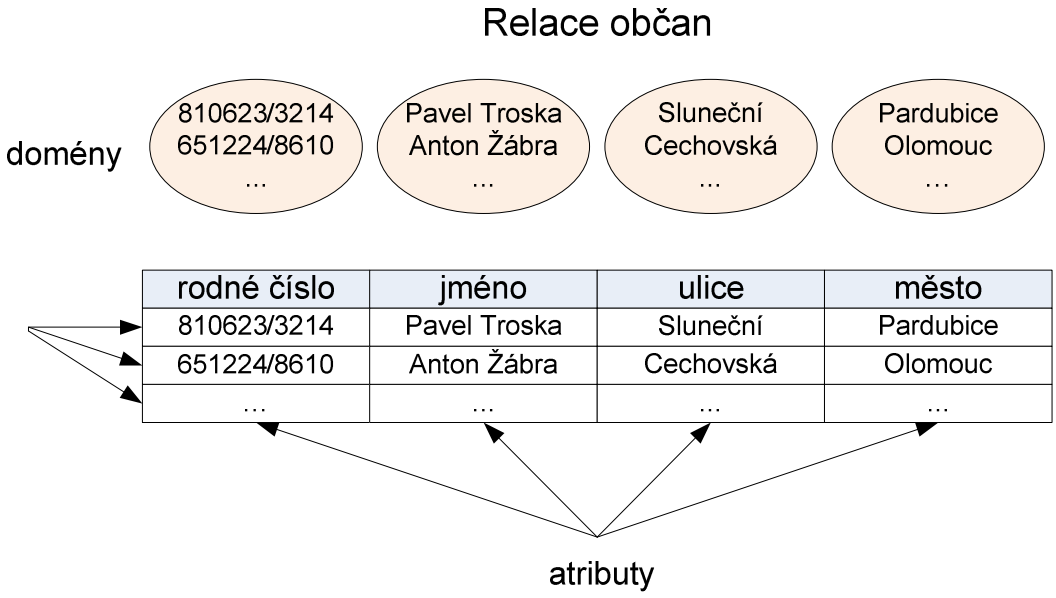
- RMD odděluje data od jejich implementace;
- při manipulacích s daty se uživatel nezajímá o přístup k datům;
- pro práci s daty slouží relační kalkul a algebra;
- pro omezení redundance dat je k dispozici normalizace relací.

Databázová relace je množinou obsahující data a je vybavena pomocnou strukturou tzv. schématem relace. Schéma relace obsahuje jméno relace, jména atributů (sloupců) a popisuje domény (integritní omezení). Relace lze obrazně chápat jako tabulku uspořádanou do sloupců (atributů) a řádků (n-tic). Jelikož je relace jakoby tabulka a může být výsledkem dotazu, může se pracovat s relací stejně jako s tabulkou. Vzhledem k tomu, že relace neobsahuje duplicitní hodnoty a je uspořádána do sloupců a řádků, nelze záznam v relaci adresovat podle čísla řádku, jelikož řádky nemají specifické pořadí. Adresovat jednotlivé záznamy je ovšem nutné a k tomu slouží konstrukce nazývaná primární klíč.

Primární klíč je atribut nebo soustava atributů, jejichž hodnoty tvoří jednoznačnou identifikaci řádku relace. Každá relace musí obsahovat primární klíč, přičemž v nejhorším případě jsou primárním klíčem všechny atributy. Každý atribut, který je součástí primárního klíče, se nazývá klíčový, ostatní jsou neklíčové. Relace může obsahovat i cizí klíče, což jsou atributy, které jsou

primárními klíči v jiné relaci, nebo alternativní klíče, které by mohly být v dané relaci primárním klíčem, ale nebyly jako primární klíč zvoleny. [12]

Jako názorná ukázka relace slouží obrázek 2.



Obrázek 2: Příklad vzhledu relace (zdroj: vlastní)

V tomto příkladu je primárním klíčem atribut rodné číslo. Je přehledně znázorněn rozdíl mezi n-ticí, atributem a doménou. N-tice je tedy řádkem relace, atribut je sloupcem relace a doména je souhrn všech hodnot jednotlivých atributů.

2.3 Tvorba informačního systému v relační databázi

Navržený relační model dat je implementován v relačním databázovém systému. Relační model dat nicméně musí být nejdříve vytvořen, a to transformován z analytického neboli konceptuálního modelu. Pro zpracování analytického modelu lze využít doporučené postupy či přístupy, které lze vymezit jako přístup strukturovaný a objektově orientovaný. Objektový přístup se opírá o standard jazyka UML. Strukturovaný přístup je směrovaný cíleně pro relační databáze a opírá se o některý z rodiny diagramů ERD (Entity-Relationship diagram).

ER diagram

ERD je typ diagramu, který se řadí do konceptuální úrovně návrhu databází. V ER diagramu vystupují dva základní druhy prvků, což jsou entity, tedy objekty reálného světa, o kterých má, pro tvůrce databáze, smysl uchovávat informace. Entitou tedy mohou být lidé, hmotné věci i nehmotné objekty, události a pojmy. Pojem entita není jednoznačně vymezený, ale každá entita musí být jednoznačně identifikovatelná. Musí tedy obsahovat identifikátor, tedy minimální množinu atributů jednoznačně identifikujících danou entitu. Identifikátor odpovídá pojmu primární klíč na vyšších úrovních tvorby datového modelu.

Druhým ze základních prvků ER diagramu jsou vztahy. Ty určují v jakém vztahu je daná entita ke druhé entitě. Vztah tedy vyjadřuje informaci, kterou nelze odvodit z atributů entit. Vztahy také určují integritní omezení na konceptuální úrovni. Mezi tato omezení se řadí parcialita a kardinalita. Parcialita znamená volitelnost členství ve vztahu, čili zdali daná entita musí nebo může mít vztah s druhou entitou. Kardinalita vypovídá o počtu výskytů dané entity v určitém vztahu, tedy jestliže výskyty jednotlivých entit mohou být ve vztahu jen jednou, vůbec nebo neomezeně. [7]

Po vytvoření ER diagramu je nutné převést tento konceptuální model na vyšší tzv. technologickou úroveň. Toto převedení se provádí pomocí dvou postupů, a to transformace a normalizace.

Transformace a normalizace

Do RMD (viz kapitola 2.2) se transformuje většina entit, vytvořených v RMD a také velká část vztahů, do relací, přičemž to, jestli se z entity či vztahu vytvoří relace, záleží na integritních omezeních konceptuální úrovně. Jestliže jsou například dvě entity ve vztahu, kdy kardinalita obou entit je jedna, čili se ve vztahu mohou vyskytnout pouze 1:1, a členství obou entit je povinné, tedy parcialita je jedna, vznikne z těchto dvou entit a vztahu na technologické úrovni jedna relace, která v sobě bude obsahovat atributy všech členů vztahu. Naproti tomu, jestliže by parcialita byla nula, tedy členství entit ve vztahu by nebylo povinné, tak by v relačním modelu vznikly relace tři. Je tedy vidět, že vytvoření konceptuální úrovně je pro správnost konečného modelu velmi důležité. [7]

Transformace není ovšem zcela dostačující pro relační model, a proto je nutné ještě vytvořené relace normalizovat. Normalizace je odstranění redundantních dat, omezení složitosti a zabránění tzv. aktualizačním anomáliím (např., aby se smazáním všech knih neztratili údaje o autorovi). Normalizace tedy vede k větší přehlednosti, rozšiřitelnosti a výkonnosti databáze. Existuje

několik tzv. normálních forem, kterých je nutné při normalizaci dosáhnout. Nejdůležitějšími z nich jsou [11]:

- První normální forma (1.NF) - relace je v 1.NF, pokud každý její atribut obsahuje jen atomické, tedy dále nedělitelné, hodnoty.
- Druhá normální forma (2.NF) - relace je v 2.NF, jestliže je v první normální formě a každý neklíčový atribut je plně závislý a primárním klíči, a to na celém klíči a nejen na jeho podmnožině.
- Třetí normální forma (3.NF) - relace je v 3.NF, když splňuje předchozí dvě formy a žádný z jejích atributů není tranzitivně závislý na klíči, tudíž její neklíčové atributy jsou nezávislé.

Existují ještě Boyce/Coddova normální forma, čtvrtá normální forma a pátá normální forma, ale tyto již mohou být plněny pouze za velmi specifických podmínek, tudíž nebudou více rozebírány, protože normální formy nejsou náplní této práce.

Normalizace vede ke zvětšení množství tabulek při implementaci, protože jakýkoliv rozpor s normálními formami se řeší rozpadem relací na menší relace. Vzniká tak velké množství propojených relací. [11]

Výsledným modelem po odstranění anomálií je kolekce normalizovaných relací různých stupňů, které jsou posléze implementovány ve vybraném databázovém systému.

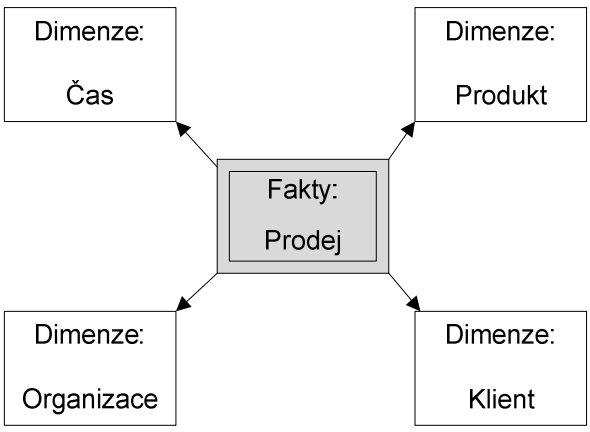
3 Dimenzionální modelování pro datový sklad

Analytická data, tedy data pro analýzy a predikci, mají jiné požadavky než data operativní v primárních zdrojích. Pro data analytického typu se nehodí, aby byla ukládána v relačních databázích pouze ve svých detailech a podle pravidel normalizace (typické pro transakční systémy). Aby analytické systémy mohly poskytovat různé analýzy a přehledy sloužící pro strategické rozhodování, je nutné, abychom se na jejich data mohli dívat z více hledisek současně. Mělo by tedy být možné vytvářet tzv. multidimenzionální pohledy, což je pro data uložená ve třetí normální formě velký problém. Navíc je nutné procházet velká množství dat, vypočítávat agregace, rychle měnit pohledy na data, rychle a automatizovaně je ukládat do přehledných tabulek a grafů. [9]

3.1 Dimenzionální modelování obecně

Dimenzionální modelování poskytuje množství technik a principů pro denormalizaci databází a tedy vytvoření schémat vhodných pro podporu rozhodování. Díky těmto technikám můžeme mluvit o multidimenzionálních databázích, které se uplatňují především k predikci trhu a analýze stávající firemní situace v podnicích. V rámci dimenzionálního modelování se rozlišují dva typy tabulek – tabulka faktů a dimenzí.

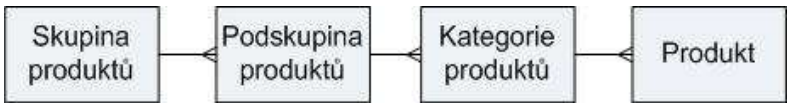
V tabulkách faktů se ukládají podnikatelsky zajímavé číselné datové položky, čili zejména aktuální obchodní fakta a měřítka. Tato fakta jsou pro různá odvětví odlišná. Například pro maloobchod jsou důležitými fakty hodnoty „Počtu prodaných kusů a Objem prodeje“, kdežto pro telekomunikace to je „Délka hovoru v minutách“ a „Průměrný počet hovorů“. Fakta jsou tedy hodnoty, které obchodní uživatelé zpracovávají s cílem získat lepší pochopení svého podnikaní.



Obrázek 3: Hvězdicové schéma tabulek dimenzí a tabulky faktů (zdroj: vlastní – zpracováno podle [4])

Tabulky dimenzí stanovují obsah faktů a obsahují atributy popisující fakta pro jednotlivá průmyslová odvětví. Jedním z hlavních principů dimenzionálního modelování je používání plně normalizovaných tabulek faktů spolu s denormalizovanými tabulkami dimenzí. Jak ukazuje obrázek 3, díky tomu, že jsou tabulky dimenzí denormalizované obsahuje zobrazené schéma pouze čtyři tabulky dimenzí.

Plně normalizovaná dimenze Produkt ale může obsahovat množství dalších tabulek, jak dokazuje obrázek 4.



Obrázek 4: Normalizovaná dimenze "Produkt" (zdroj: vlastní - zpracováno podle [4])

Denormalizace dimenzí způsobí, že každá dimenze má svoji vlastní hierarchii, která vede k seskupování a členění, což se uplatní při zobecňování výstupů. Uživatel tak může získat více či méně podrobné informace. Časová dimenze je vždy součástí multidimenzionální databáze - která se implementuje velice hojně v podobě datového skladu nebo datového tržiště - protože jedním z cílů těchto databází je zajistit historická data. Například dimenze „Čas“ může obsahovat hierarchii den-měsíc-čtvrtletí-rok. [4]

Dimenzionální schéma je velmi podobné hvězdě, proto se pro dimenzionální model vžilo označení STAR (hvězdicové schéma). Tabulky faktů se nacházejí ve středu schématu a odpovídající dimenze jsou obvykle umístěny okolo (viz. obrázek 3). Řešení podle hvězdicového schématu je někdy z různých důvodů nevýhodné. Proto se ve speciálních případech tabulky dimenzí upravují, resp. normalizují. Vznikne tak schéma SNOWFLAKE (sněhová vločka), ve kterém některé dimenze mohou být normalizované. [4], [9]

3.2 Podstata dimenzionálního modelování v příkladech

Podstatou dimenzionálního modelování a jeho hlavním úkolem je vytvořit základní logiku uložení nebo uspořádání dat tak, aby vyhovoval požadavkům na analytické a plánovací aplikace v rámci podnikového řízení.

Dimenzionální modelování vychází z poznání a zhodnocení potřeb řízení dané organizace a na základě toho [9]:

- definuje všechny dimenze, jejich obsah, včetně vnitřní hierarchie prvků, a dílčí charakteristiky jednotlivých dimenzí;
- určuje soustavu sledovaných ukazatelů a definuje jejich dílčí charakteristiky;
- specifikuje vazby mezi ukazateli a odpovídajícími dimenzemi.

Pro dimenzionální modelování je charakteristické, že úroveň detailu jeho řešení se může měnit, např. podle přístupu k projektu (uplatnění datových tržišť nebo jen datového skladu). Může se samozřejmě zpřesňovat a konkretizovat v průběhu projektu podle účelu a aktuálních potřeb řešení. Zde je uvedeno, co všechno je účelné v rámci dimenzionálního modelu určovat.

V případě návrhu dimenzí se určují následující charakteristiky [9]:

- identifikace dimenze – podle stanovených projektových standardů, např. d_zakaznik;
- plný název dimenze, např. Zákazníci a obchodní partneři podniku;
- hierarchická struktura dimenze, např. Dimenze Čas, Hierarchie Kalendářní rok – úrovně rok, měsíc, den; Hierarchie Fiskální rok – úrovně rok, čtvrtletí, měsíc, týden, den;
- prvky dimenze, vyjádřené většinou jejich základní hierarchickou strukturou, např.:
    - Všichni partneři,
        1. Klíčoví zákazníci,
            a) Klíčoví zahraniční zákazníci,
            b) Klíčoví tuzemští zákazníci, …;
- počty prvků v dimenzi na jednotlivých hierarchických úrovních, např. a) Klíčoví zahraniční zákazníci (10);
- Zdroj dat pro dimenzi, např. odpovídající databáze (p_zakaznici), databázová tabulka číselníku, textový soubor apod. s příslušnou standardní identifikací datového zdroje v rámci IS/IT;
- Definování kalkulovaných prvků v dimenzi, které se automaticky promítají do všech přiřazených ukazatelů k dimenzi a zajišťují požadované výpočty. Např. existuje-li dimenze *Plany* obsahující prvky *skutecnost* a *plan*, pak lze definovat kalkulovaný prvek *plneni_planu* = *skutecnost* /*plan* * 100, jehož přiřazením např. k ukazateli prodeje budou získány hodnoty plnění plánu prodeje, obdobně výroby atd., a to

v kombinaci se všemi ostatními dimenzemi. Podobně se obvykle definují nejrůznější bazické a řetězové indexy, odchylky od standardních hodnot apod.

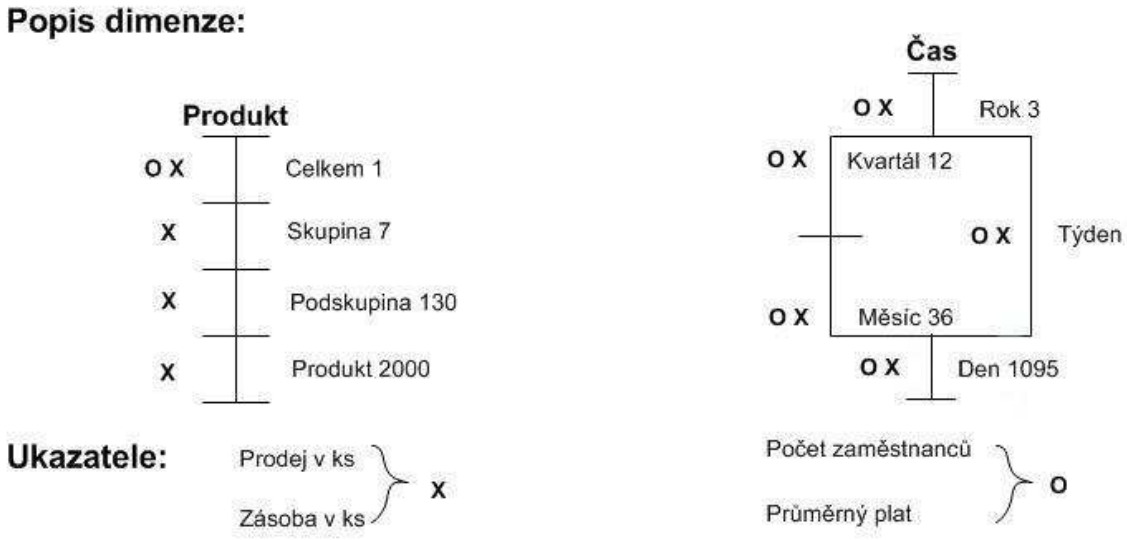
Obvyklými příklady dimenzí jsou čas, prognóza, útvary (obchodní, marketing, aj.), obchodní zastoupení a obchodní zástupci, zákazníci, teritoria, zakázky, produkty …

V případě návrhu ukazatelů se určují např. tyto charakteristiky [9]:

- symbolická identifikace ukazatele – rovněž podle stanovených projektových standardů, např. Prodej_objem;
- plný název ukazatele, např. Objem prodeje v Kč;
- jednotka vyjádření ukazatele – Kč, kusy, procenta;
- zdroj dat pro zdrojové ukazatele vstupující do datových skladů a OLAP databází. Např. databáze (p_prodej), obsahující hodnoty sledovaných ukazatelů, a to rovněž s příslušnou standardní identifikací datového zdroje v rámci IS/IT;
- výpočty ze zdrojových ukazatelů a příp. konstant pro ukazatele kalkulované – specifikace výpočetních předpisů, vzorců.
- určení tzv. analytických pravidel, tj. určení hodnot nebo vztahů, při jejichž překročení má dojít k signalizaci (např. barevným zvýrazněním) pro uživatele, že jde o problém nebo příležitost. Např. Problém: Prodej_objem < 250000. Tato pravidla se mohou vztahovat pouze k vybraným dimenzím nebo jejich prvkům. [9]

3.3 Přístupy k dimenzionálnímu modelování

Přístup k řešení a standardy zobrazení dimenzionální analýzy jsou v praxi různé a odpovídají i použitým základním nebo firemním metodikám. Obrázek 5 dokumentuje způsob řešení dimenzionální analýzy MDS (Multidimensional Domain Structure) Erika Thomsena, který je považován za jeden ze základních přístupů v této oblasti.



Obrázek 5: Erik Thomsen: MSD - Multidimensional Domain Structure (zdroj: vlastní - zpracováno podle [9])

Z obrázku je zřejmá struktura zvolených dimenzí (Produkt, Čas), počty prvků na jednotlivých úrovních dimenze a jejich přiřazení k ukazatelům *Prodej v ks, Zásoba v ks, Počet zaměstnanců, Průměrný plat.*

Další možnost, jak řešit a dokumentovat činnosti a výsledky dimenzionální analýzy, je založena na tabulkovém vyjádření dat. Návrh jednotlivých dimenzí může být vyjádřen, např. u dimenze *organizace* (viz. tabulka 1). [9]

Tabulka 1: Příklad popisu dimenze "Organizace" (zdroj: vlastní - zpracováno podle [9])

Dimenze: Organizace **Id**.: org				
	Struktura	**Prvky**	**Počty**	**Poznámka**
Struktura:		Podnik_celkem	1	
	1.	Výroba	7	
	1.1	Příprava	3	
	1.2	Konstrukce	2	

Dimenze: Organizace **Id**.: org				
	Struktura	**Prvky**	**Počty**	**Poznámka**
	1.3	Montáž	5	
	2.	Obchod	8	
	2.1	Obchodní skupina Praha	4	
	2.2	Obchodní skupina Brno	3	
	3.			
Výpočty:	Podíl_na_zakazce_mimo_Prahu = (Obchod – Praha)/ Obchod * 100			
Zdroje:	Utvary – číselník útvarů podniku			

V uvedeném příkladu je třeba uvážit, zda začlenit do této dimenze i strukturu poboček, nebo pro pobočky vyhradit zvláštní dimenzi. V tomto případě je použita druhá varianta. Je možné definovat i několik organizačních struktur, pokud je to účelné.

3.4 Postupy dimenzionálního modelování

Principy a postup dimenzionálního modelování jsou často ovlivněny konkrétní situací projektu. Přesto je možné vymezit následující standardní kroky [9]:

1) výběr a základní obsahové vymezení řešené oblasti podnikového řízení;
2) návrh všech relevantních dimenzí a jejich charakteristik;
3) návrh ukazatelů, jejich dílčích charakteristik a granularity;
4) řešení vazeb mezi dimenzemi a ukazateli;
5) promítnutí řešení do návrhu tabulek dimenzí, návrhu tabulek faktů, návrhu schémat (hvězda, vločka – viz dále), což je již obvykle součástí modelování datového skladu.

Podstatným aspektem řešení je určení náplně dimenzí a jejich prvků, to znamená např., jací konkrétní zákazníci budou naplňovat dimenzi *zákazník*, jaké konkrétní komodity dimenzi *komodita* apod. Je nutné zde pečlivě volit jak dimenze, tak prvky, neboť velký rozsah prvků v dimenzi často spíše znepřehledňuje práci uživatele a zvyšuje neúměrně provozní nároky. Je tedy nezbytné prvky v dimenzi racionálně strukturalizovat, a případně vybírat ty prvky, které jsou pro následné analytické aplikace nejvýznamnější. Problémem někdy je to, že taková

hierarchie nebo agregáty nemají vnitřní logiku (pokud jsou vytvořeny uměle) a uživatele mohou i mást.

Speciální místo v modelu má časová dimenze, tj. jaká bude struktura časových intervalů (roky, kvartály, měsíce), zda se bude k aktuálnímu datu nějakým způsobem měnit (např. na dekády, dny), zda se budou nějaké starší časové úseky přesouvat z provozního řešení do archivu (tzv. aging) apod. [9]

3.5 Výhody výsledku dimenzionálního modelování

Z předchozích charakteristik a možností dimenzionálního modelování vyplývá taková organizace dat, že ve svém výsledku a aplikacích nabízí tyto efekty [9]:

- lze je prezentovat na libovolné úrovni agregace (s využitím funkcí drill-down, drill-up, drill-across);
- dimenze lze v průběhu specifikace dotazu nebo požadavku na výstupní data libovolně kombinovat (na principu slice and dice[1], crosstabing[2], tedy identifikace dat pomocí dimenzí v různých tabulkách)
- nad dimenzionálně uspořádanými daty lze provádět nejrůznější aritmetické i množinové operace, lze využívat agregační a statistické funkce (např. SUM, MIN, MAX, COUNT, AVG, DEVIATION, VARIANCE), lze efektivně vyhledávat extrémní hodnoty podle dimenzí apod.

[1] slice and dice – technika, která redukuje obor hodnot jedné (slicing) nebo více (dicing) dimenzí podle zvoleného filtru

[2] crosstabing – možnost přetahování řádků a sloupců = transpozice matice

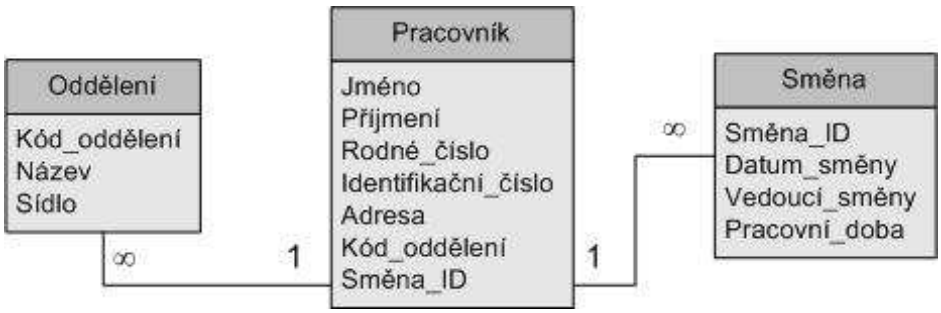
4 Technologie datového skladu

Technologie datového skladu zahrnuje více hledisek. Jedná se o vlastní datový model, ale také formu přístupu k datům a formu čerpání dat z primárních zdrojů.

Informační systémy našly svou spolehlivou technologickou platformu v databázových systémech. Bylo třeba data shromažďovat, uchovávat, rychle k nim přistupovat a také pořizovat nová data ve stanoveném formátu. Došlo k oddělení dat a aplikací, operujících s těmito daty. Začaly vznikat hierarchické a síťové databáze, a pak dnes nejpoužívanější relační databáze (v praxi se ještě používají např. objektové databáze).

Relační databázové systémy mají svůj teoretický základ v relačním modelu dat (definovaný v roce 1969 Dr. E. Coddem). Model organizuje data do tzv. n-tit (tabulek), které tvoří základ relační databáze. Struktury dat jsou pevně definovány, některé položky tabulek mohou mezi sebou vytvářet vazby tzv. relace. Mezi další vlastnosti databází patří transakce, zamykání, multi-user přístup, strukturální indexování, secured access apod. [3]

Nevýhodou relačních databází bylo v jejich užití pro analytické účely to, že se v nich obtížně hledaly závislosti jednotlivých veličin, data byla nehomogenní, příprava údajů byla časově náročná, takže nasazování databází znamenalo sice možnost urychlení práce, větší bezpečnost při práci s daty a možnost vícenásobného přístupu uživatelů v reálném čase, ale i tohle mělo svá úskalí. Pro ilustraci je uveden příklad relačního modelu (viz. obrázek 6).



Obrázek 6: Schéma relačního databázového modelu (zdroj: vlastní - zpracováno podle [8])

Velkým problémem bylo i to, že operační aplikace organizace neuchovávají historické údaje a byl tedy omezen objem dat, ke kterým bylo možné v reálném čase přistoupit.

Prováděné analýzy zatěžují primární systémy, jsou náročné pro samotné tvůrce, navíc se požadavky uživatelů mohou měnit. Tedy samotné generování různých výstupních sestav a analýz bylo často náročnou činností, probíhající nezřídka i několik dní. Výstupy byly mnohdy

velmi rozsáhlé a ne příliš přehledné a většinou musely být dál zpracovávány, což si opět vyžádalo další časové náklady.[3], [5]

Na začátku 90. let došlo k oddělování dat pro analýzy a vzniku různých systému pro podporu rozhodování (DSS systémy, MIS apod.). Koncept datového skladu byl zpracován na začátku 90 let (1992 – William H. Inmon[3]) a plně uveden do praxe byl ve 2. polovině 90. let. Další boom přišel s nasazením různých e-business aplikací, u kterých datové sklady tvořily zdroj dat. Na datové sklady se začaly nabalovat další aplikace např. CRM, BI, E-commerce, Data Mining. [3]

4.1 Charakteristiky datového skladu

William H. Inmon definuje datový sklad jako „kolekci sjednocených, předmětově orientovaných databází navržených za účelem poskytovat informace požadované pro rozhodování“. Na rozdíl od toho provozní systémy mají zajišťovat záznam a dokončování různých typů transakcí a nevyhovují tedy požadavkům řídících pracovníků, kteří hledají souvislosti, trendy a specifika různých informací. [4]

Sjednocený

Datový sklad obsahuje data získaná z mnoha podnikových provozních systémů a může být plněn také externími daty. Každý z provozních systémů zaznamenává rozdílné typy obchodních transakcí a tyto nemusejí být integrovány (sjednoceny). Může tedy dojít k situaci, kdy se v případě zákazníka A v jednom provozním systému a zákazníka B v druhém provozním systému jedná o tu samou osobu. Tato skutečnost ovšem není nijak podchycena a zautomatizována. Datový sklad spojuje data z mnoha různorodých provozních systémů a poskytuje integrovaný pohled na zákazníka a na plnou šíři jeho vztahů s organizací. [4]

Předmětově orientovaný

Tradiční provozní systémy jsou zaměřeny na potřeby oddělení či divizí. S příchodem zpětného inženýrství obchodních procesů začaly podniky využívat procesně zaměřené týmy a pracovníky pro konkrétní případy. Moderní provozní systémy přenesly zaměření na provozní požadavky celého obchodního procesu a zaměřily se na podporu celého obchodního procesu od začátku do konce. Datový sklad přinesl tradiční informační pohledy na celopodnikové

[3] William H. Inmon je považován za otce data warehousingu, je expertem na databázové technologie a návrh datových skladů, autor knihy Building the Data Warehouse

subjekty, jako jsou zákazníci, prodeje či zisky. Tyto subjekty ohraničují jak hranice organizační, tak procesní a pro poskytnutí kompletního obrazu vyžadují informace z více zdrojů. [4]

Agregovaná data

Do datového skladu jsou ukládána agregovaná data ve formě „multidimenzionálních kostek“. Data jsou uložena denormalizovaná s různými stupni agregace. Toto uložení dat umožňuje získat odpovědi na otázky, které není možné předem připravit, ale které mohou podporovat průběžně vznikající statistická šetření. Datový sklad se může v průběhu svého životního cyklu optimalizovat, kdy optimalizace znamená hledání vhodného poměru mezi rychlostí získávání požadovaných analýz a objemem a dobou vytváření agregovaných dat. [10]

4.2 Podmínky pro vytvoření datového skladu

Návrh a implementace datového skladu vyžaduje sjednocení datových prvků v primárních zdrojích, resp. vyžaduje sjednocení datových přístupů v rámci celé organizace.

Jedna verze pravdy

Data v datových skladech jsou konzistentní a kvalitní dříve, než jsou zpřístupněna uživatelům. Oproti dobám, kdy byly používány běžné zdroje informací, jsou nyní požívány datové sklady, díky kterým jsou všechny diskuse o pravdomluvnosti použitých dat bezpředmětné. Datové sklady se stávají běžným zdrojem informací pro rozhodování v rámci celé organizace.

Jak nadpis říká, „jedna verze pravdy“ je často možná po mnoha diskusích a debatách o pojmech použitých v rámci organizace. Například, pojem zákazník může mít diametrálně odlišný význam – není výjimkou pro některé zaměstnance označovat možné klienty jako zákazníky, zatímco někteří jiní zaměstnanci té samé organizace mohou používat tento pojem pro označení pouze současných klientů. [4]

Přesně zaznamenaná minulost

Mnoho grafů a čísel, které získávají manažeři, má malý význam bez srovnání s historickými hodnotami. Například, zcela běžná je sestava porovnávající výkonnost podniku oproti výkonnosti předchozího roku. Velmi důležité jsou také sestavy zobrazující výkonnost podniku pro jeden a tentýž měsíc tři roky po sobě.

Datové sklady by měly být používány pro záznam minulosti, čímž transakční systémy mohou zůstat zaměřené na správný záznam aktuálních transakcí. Skutečné historické hodnoty nejsou

uloženy v transakčních systémech nebo nejsou odvoditelné přidáním či odebráním transakčních hodnot k poslednímu známému stavu. Spíše jsou (za účelem rychlého přístupu) tato historická data nahrána a sjednocena s dalšími daty do datových skladů. [4]

Porcování a krájení dat

Jak již bylo řečeno, dynamické sestavy umožňují uživatelům pohlížet na datové sklady z různých úhlů, na různých úrovních podrobností. Uživatelé mohou pomocí rozporcování a nakrájení dat v datových skladech aktivně získat informace, které potřebují.

Snadná dostupnost rozdílných pohledů na data také zdokonaluje obchodní analýzy snížením času a úsilí nutného na sběr, formátování a češtění informací ze surových dat. [4]

Oddělení analytického a transakčního zpracování

Proces rozhodování a proces transakčního zpracování mají velmi odlišné požadavky na architekturu. Pokusy sloučit jak rozhodovací, tak transakční informace vyžadující stejný systém či stejnou systémovou architekturu zvyšuje křehkost architektury IT a vede z hlediska správy systémů ke změně postupů.

Datové sklady oddělují analytické zpracování od transakčního, nabídnutím odlišné systémové architektury pro implementaci rozhodování. Toto činí celou architekturu IT podniku více přizpůsobivou změnám potřeb. [4]

4.3 Datové tržiště jako forma datového skladu

Na konci 80. a počátku 90. let byl datový sklad definován tak, aby obsahoval granulovaná, detailní, normalizovaná, historicky-orientovaná data. Poté začaly v 90. letech organizace budovat datová tržiště, která měla být obdobou datových skladů v jednotlivých odděleních organizace a obsahovat data pro specifický objekt zájmu. [1]

Datový sklad je náročný

Budování datového skladu bylo vnímáno jako finančně i aplikačně náročná činnost, protože byla potřebná integrace celopodnikových dat do centrálního úložiště, včetně detailních historických dat. V porovnání s datovým skladem byla datová tržiště chápána jako jednodušší a levnější zmenšená verze DW obsahující pouze data určitého úseku a pouze omezené množství historických dat. [1]

Slučitelnost datových tržišť

Díky těmto názorům začalo mnoho podniků implementovat datová tržiště. Po té, co byli firmy nuceny zbudovat druhá datová tržiště, zjistily, že jsou pro ně potřebná stejná data. To ještě nebylo považováno za problém, protože organizace se potýkaly s redundancí dat v OLTP[4] prostředích po léta. Jakmile však začaly vznikat třetí a další datová tržiště, byla v nich slučitelnost informací téměř nemožná a práce IT oddělení byla extrémně složitá. Přesto dostávali obchodní uživatelé z každého oddělení na stejnou otázku různé odpovědi. [1]

Výhody datového skladu

V datovém skladu je jednodušší vytvářet nové obchodní aplikace, protože obsahuje data o zákazníkovi ukládaná po celou dobu obchodního vztahu. Zatímco díky definici datových tržišť a jejich hranicím je téměř nemožné sledovat chování zákazníka.

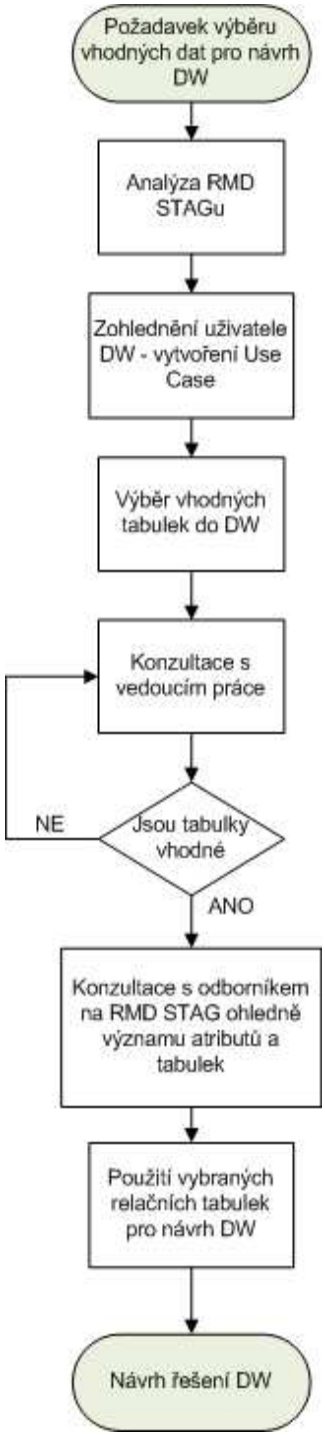
Datový sklad šetří. Datová tržiště mohou být implementována na mnoha platformách. Centralizovaný datový sklad vyžaduje pouze jednu. Z předchozího vyplývá, že datový sklad šetří při rozsáhlé implementaci místo na disku a také peněžní náklady díky zmírnění požadavků na správu dat.

Sjednocení dat umožňuje všem oddělením v organizaci odpovědět na otázky, které nebylo dříve zodpovědět. Datový sklad je nejlepší pro propojené společnosti, kde jednotlivá oddělení pracují společně. [1]

[4] OLTP (Online Transaction Processing) – OLTP systémy uchovávají záznamy o jednotlivých transakcích (viz. kapitola 2.1) pomocí relačních databázových technologií

5 Návrh datového skladu

Cílem této bakalářské práce je navržení datového skladu pro IS STAG. Nejdříve jsem si stanovil postup při tvorbě návrhu. Postup pro tuto konkrétní situaci znázorňuje obrázek 7.



Obrázek 7: Vývojový diagram postupu prací na projektu (zdroj: vlastní)

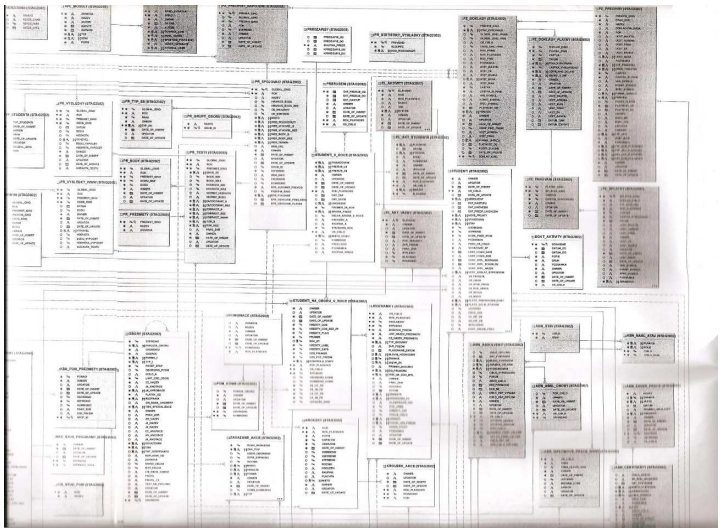
Prvním základním požadavkem je výběr vhodných dat pro návrh datového skladu. To se zdá jako banalita, ale je to právě výběr vhodných dat, který je nejdůležitější částí celého projektu,

protože kvalita datového skladu stojí a padá s kvalitou jeho zdrojových dat. Proto je třeba udělat důkladnou analýzu relačního modelu informačního systému STAG, což znamená hlavně získání relačního modelu IS STAG a jeho následné důkladné prostudování. Je třeba definovat, pro koho bude datový sklad určen, jelikož podle toho budou vybírány zdrojové tabulky, ze kterých datový sklad vznikne. Definice uživatelů a jejich požadavků je v tomto konkrétním případě provedena pomocí Use Case, což je diagram modelovacího jazyka UML.

Na základě těchto úvodních analýz je pak možné vytipovat tabulky vhodné pro vytvoření datového skladu. Samozřejmě je nutné konzultovat každý návrh s vedoucím práce, který usměrní další průběh celého projektu a musí schválit jednotlivé zdrojové tabulky. Následně je nutné dojednat schůzku s odborníkem na IS STAG. Ten vysvětlí, jaké atributy jsou vhodné k použití pro datový sklad a objasní význam jednotlivých atributů. Po této konzultaci již nic nebrání použití vybraných, schválených a objasněných tabulek k návrhu datového skladu. Návrh bude proveden v aplikacích case 4/0 a Microsoft Visio.

5.1 Analýza RMD STAG

Prvním základním krokem při analýze IS STAG bylo vlastní získání relačního modelu IS STAG. Ohledně tohoto problému byl osloven Ing. Ondřej Prusek, Ph.D., který pracuje na Univerzitě Pardubice jako jeden ze správců IS STAG. Ten tedy nechal vytisknout tabulky, které se nachází v relačním modelu STAG. Celý relační model se vešel na papír velikosti A2. Náhled na část celkového relačního modelu ukazuje obrázek 8.



Obrázek 8: Část relačního modelu IS STAG (zdroj: relační model IS STAG)

Obrázek je v tomto zobrazení nečitelný, ale čitelnost není jeho účelem, jde o nastínění velikosti a složitosti RMD systému STAG. Výsek na obrázku je pouhou čtvrtinou celkového

relačního modelu, takže je zřejmé, že relační model IS STAG má velmi komplikovanou strukturu, mnoho vazeb a mnoho relací. Důležité tedy je vybrat vhodné tabulky pro použití při návrhu datového skladu. V jednotlivých tabulkách je také mnoho atributů, které většinou nejsou napsány celým názvem. Jejich význam tak není zcela jasný, a proto byla provedena konzultace s odborníkem na IS STAG (Ing. O.Prusek, Ph.D.) ohledně významu tabulek a jejich atributů.

5.1.1 Identifikace kandidátů na fakty a dimenze

Po důkladném prostudování celého relačního modelu jsem zjistil překvapivou skutečnost, a to, že celý relační model IS STAG obsahuje minimum relací, které by poskytovaly hodnoty do tabulky faktů. Hledání vhodných tabulek faktů se tak hned od počátku práce stalo nejdůležitější činností úvodní analýzy relačního modelu. Naopak velmi snadné bylo nalezení relací vhodných pro tabulky dimenzí, jelikož relační model IS STAG obsahuje mnoho relací, které jsou popisného charakteru. Jedná se tedy o relace popisující studenty, učitele, osoby obecně, místnosti, studijní předměty, studijní obory, fakulty atd. Principem datového skladu je ovšem především uchovávání faktových hodnot, a proto bylo nutné vyřešit problém s nedostatkem těchto dat. Datové sklady se většinou používají ve velkých podnicích, a proto všechny knihy o datových skladech odkazují na tabulky typu objem prodeje v kusech nebo objem prodeje v korunách jako na univerzální příklady tabulek faktů, protože prodej je tím, co firmy samozřejmě nejvíce zajímá.

Ovšem prioritou Univerzity Pardubice prodej není. Pod záštitou Univerzity Pardubice se jistě prodávají skripta, vede se účet v menze a knihovně, ale tyto účty a informace nejsou součástí relačního modelu IS STAG. STAG je zkratka slov studijní agenda, z čehož vyplývá, že relační model IS STAG obsahuje především informace spojené se studiem. Proto musí být i datový sklad zainteresován právě studiem.

Po úvodním prostudování relačního modelu jsem došel k těmto závěrům:

- Relační model IS STAG obsahuje minimum faktových hodnot a bude třeba se na fakta zaměřit;
- existuje dostatek popisných dat, takže vytvoření tabulek dimenzí nebude problém;
- pro implementaci datového skladu by bylo vhodné, kdyby byly v relacích vytvořeny nové atributy pro podporu datového skladu.

5.1.2 Návrhy na tabulky faktů

Vyšel jsem z orientace STAGu na informace spojené se studiem a na tomto základě jsem vytvořil první návrh, a to použití relace Studenti, kterou znázorňuje obrázek 9.

Studenti	
PK	os_cislo
	owner
	updator
	date_of_insert
	date_of_update
	absolvent
	dat_nastupu
	dat_ukonceni
	dat_predp_ukonceni
	nove_prijaty
	vykazovan
	stav
	osobidno
	stpridno
	doba_pred_stud
	poznamka
	pred_os_cislo
	vykazovat_zp
	last_login_date
	last_login_adr
	dokt_stpl_sestaven
	dokt_stpl_schvalen
	dokt_stpl_nazev
	ucet_zasilat_stipendium
	vs_fakulta
	vs_obor
	vs_stud_prog
	vs_rok_absol
	vs_skola
	vs_misto
	platit_prekroceni_doby
	platit_dalsi_studium
	ucitidno
	prac_zkr
	cz_inf_ds
	an_inf_ds
	skolaidno
	dokt_druhe_prac

Obrázek 9: Relace Studenti (zdroj: vlastní - přepracováno podle relačního modelu IS STAG)

Studenti jsou z hlediska univerzity jistě důležitým pojmem, a proto by mohlo být zajímavé, kdyby datový sklad sledoval počet studentů na univerzitě podobně, jako se v podnicích sleduje

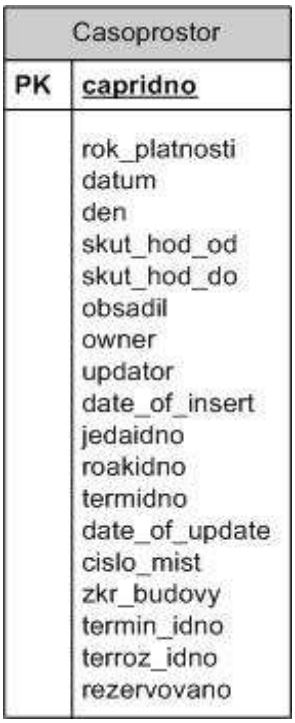
objem prodeje. O studentech je v relaci Studenti mnoho informací a tato relace je zároveň propojena s mnoha dalšími tabulkami, a proto se zdá být vhodnou pro hledané řešení.

Při analýze primárních zdrojů (v tomto případě systém STAG) pro potřeby potenciálního datového skladu jsou možné dva postupy:

- Buď na základě konkrétního požadavku (např. student v časovém vývoji a vzhledem k typu středoškolského vzdělání) hledat v primárních zdrojích vhodné atributy;
- nebo bez konkrétního požadavku hledat potenciální možnosti primárních zdrojů a snažit se vytipovat atributy vhodné pro případná datová tržiště.

Já jsem postupoval druhým způsobem, neboť takové bylo zadání bakalářské práce. Podle mého názoru, je to postup složitější, protože se nevyhází z konkrétního požadavku na přesně orientované datové tržiště. Proto jsem posuzoval vhodnost relace Studenti jako kandidáta na tabulku faktů, tzn. bral jsem v úvahu relaci Studenti jako celek a zároveň její jednotlivé atributy. Při bližším zkoumání relace Studenti byly nalezeny dva atributy odkazující na oblast placení a tudíž na nějaký obchodní fakt. Tyto atributy jsou v relaci zvýrazněny tučným písmem a jedná se o atributy platit_prekroceni_doby, platit_dalsi_studium (viz obrázek 9).

Z tohoto poznatku jsem usoudil, že v relačním modelu IS STAG se přece jen uchovávají informace obchodního charakteru a musí mít přiřazenu nějakou tabulku. Po opětovném prostudování relačního modelu se zaměřením na vazby relace Studenti byla nalezena velice strohá relace Doklady_platby. Do této relace patří mimo jiné i atribut Castka, který je tím, co bylo hledáno. Otázkou ovšem zůstávalo, k jakým platbám se tato relace vztahuje, jelikož patří mezi okrajové relace celého modelu. Na konzultaci s vedoucím práce byla tato relace navržena jako vhodná pro tabulku faktů, přičemž vedoucí práce tuto relaci akceptovala, s požadavkem domluvení konzultace s odborníkem na IS STAG. Byla tedy sjednána konzultace s Ing. M. Koblížkem, který byl navržen jako odborník pracovnicí finančního úseku Univerzity Pardubice. Na konzultaci bylo osvětleno, že relace Doklady_platby zahrnuje pouze platby školy za uznaná stipendia jakéhokoli typu studentům a naopak poplatky studentů spojené se studiem. Tato relace se tedy nezabývá vedením přehledu o platbách v menze, v knihovně, aj. Tyto běžné platby by byly pro tvorbu skladu mnohem zajímavější, a proto se od záměru využít tuto relaci jako tabulku faktů také upustilo.



Obrázek 10: Relace Casoprostor (zdroj: vlastní - přepracováno podle relačního modelu IS STAG)

Při konzultaci ohledně významu relací s Ing. Koblížkem jsem zjistil, že nejvíce využívanými tabulkami při reálném provozu IS STAG jsou tabulky zabývající se osobami, jako jsou tabulky Studenti, Uchazeči, Osoby atd. Další důležitou tabulkou při reálném provozu je tabulka Casoprostor (viz obrázek 10). Po důkladném prozkoumání relace Casoprostor a jejích vazeb s ostatními relacemi bylo zjištěno, že tato relace slouží k monitorování obsazenosti místností na Univerzitě Pardubice a je blízce propojena s mnoha popisnými relacemi, tudíž by po vytvoření datového skladu s použitím této relace nebyl problém s tabulkami dimenzí. Opět je ale třeba řešit situaci, kdy není zcela jasné, co by mělo být sledovaným faktem. Jak již bylo řečeno v reálných situacích je datový sklad zaváděn především ve firmách a sleduje se v něm objem prodeje. Z pohledu na Univerzitu Pardubice jako na firmu by objemem prodeje mohla být služba „učení“ a relace Casoprostor vypovídá o tom, kolik hodin bylo v daných místnostech odučeno. Navíc je tato relace propojena s relací Jedakce, která uchovává přehled o jednorázových akcích, jako jsou dlouhodobě neplánované změny v rozvrhu, externí přednášky atd. Jako sledovaný fakt byl tedy navržen počet odučených hodin, který byl následně na konzultaci s vedoucí práce schválen jako vhodný.

5.2 Analýza datového skladu

Po pracném hledání dostatečně smysluplné tabulky faktů bylo nutné zvolit další důležitou podmínku pro tvorbu datového skladu, a to pro koho bude datový sklad určen. Podle toho se potom bude orientovat celý další návrh skladu, protože kdyby byl určen například studentům, musel by obsahovat jiné tabulky dimenzí, než kdyby byl navržen pro učitele.

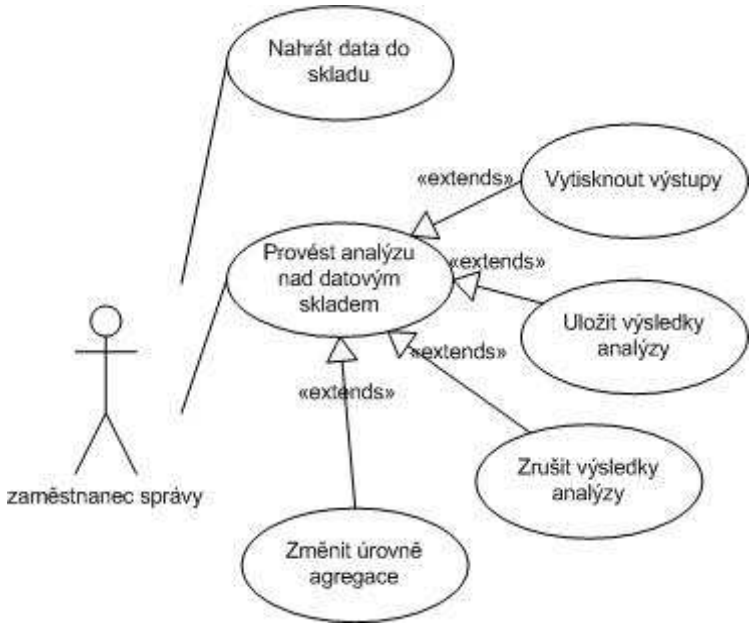
Stanovil jsem si následující zadání:

- datové tržiště má sloužit pro pracovníky rektorátu Univerzity Pardubice, resp. pracovníky v oddělení pro správu Univerzity Pardubice,
- datové tržiště má zároveň sloužit pro zaměstnance v oblasti správy jednotlivých fakult Univerzity Pardubice.

5.2.1 Use case neboli případ užití

Pomocí případů užití se snaží vývojáři modelovat typické interakce uživatelů se systémy. Cílem tohoto snažení je vymezit jednoznačně rozsah budované aplikace a skutečné požadavky uživatelů na budoucí systém. Věnování pozornosti vytvoření případu užití je potřebné, jelikož jen to, co popisují případy užití, se bude programovat. Všechny akce v systému jsou vyvolávány aktéry, což jsou externí objekty, které si vyměňují informace se systémem. Případ užití je sadou scénářů, které sledují společný cíl, a je vždy iniciován aktérem, přičemž případ užití vyjadřuje co, ale nikoliv jak, budoucí systém nabídne uživateli. Existuje několik druhů vazeb mezi případy užití. První z nich je vazba *include*, která se objevuje tam, kde existuje podobná nebo stejná část sekvence scénáře případu užití opakující se ve více případech užití. Další, neméně důležitou, je vazba *extend*, která přidává doplňkové chování do základního scénáře. Poslední je užití *zobecnění,* které umožňuje převést chování společné pro více případů užití do rodičovského případu užití. Zobecnění se ovšem v praxi téměř nepoužívá. [6]

Předpokladem pro navrhovaný datový sklad je, že všechny dotazy na datový sklad nebo nástroje reportingu budou používány výhradně zaměstnanci správy, a proto byl vytvořen diagram případů užití pro aktéra zaměstnanec správy (viz obrázek 11).



Obrázek 11: Případy užití pro zaměstnance správy (zdroj: vlastní)

Při implementaci velkých podnikových informačních systémů musí být případy užití vytvořeny pro několik aktérů, kteří mají rozdílné požadavky na systém. Tím, že je používání dat v budovaném datovém skladu omezeno na jeden typ pracovníka, je dostačující vytvoření pouze jednoho případu užití. Z pohledu zaměstnance správy by měl mít datový sklad tedy následující funkce.

5.2.2 Funkce navrhovaného datového skladu

Základní funkcí, kterou musí datový sklad zvládat, je nahrávání dat do skladu. Bez dat by samozřejmě datový sklad neměl vůbec smysl. Tuto funkci by mohl zaměstnanec provádět v pravidelných intervalech, ať už jednou za týden nebo jednou za měsíc. Rozhodně není třeba nahrávat data do skladu denně, protože rozvrh místností je stejný po celý semestr až na již dříve zmíněné jednorázové akce a využívání místností ve zkouškovém období.

Hlavní funkcí datového skladu je samozřejmě tvorba analýz nad ukládanými daty. K tomuto případu užití je několik rozšiřujících případů užití. Rozšiřujícími jsou proto, že mohou být „spuštěny“ pouze volitelně z případu užití *Provést analýzu nad datovým skladem.* Je tedy důležité, aby byly výsledky provedených analýz nějakým způsobem uloženy, ať už vytisknutím na papír, nebo uložením do textového formátu. Největší význam má ale rozšiřující případ užití pro změny úrovní agregace, jelikož jen tak je zajištěno, že se budou moci analýzy při jednom přihlášení pružně měnit.

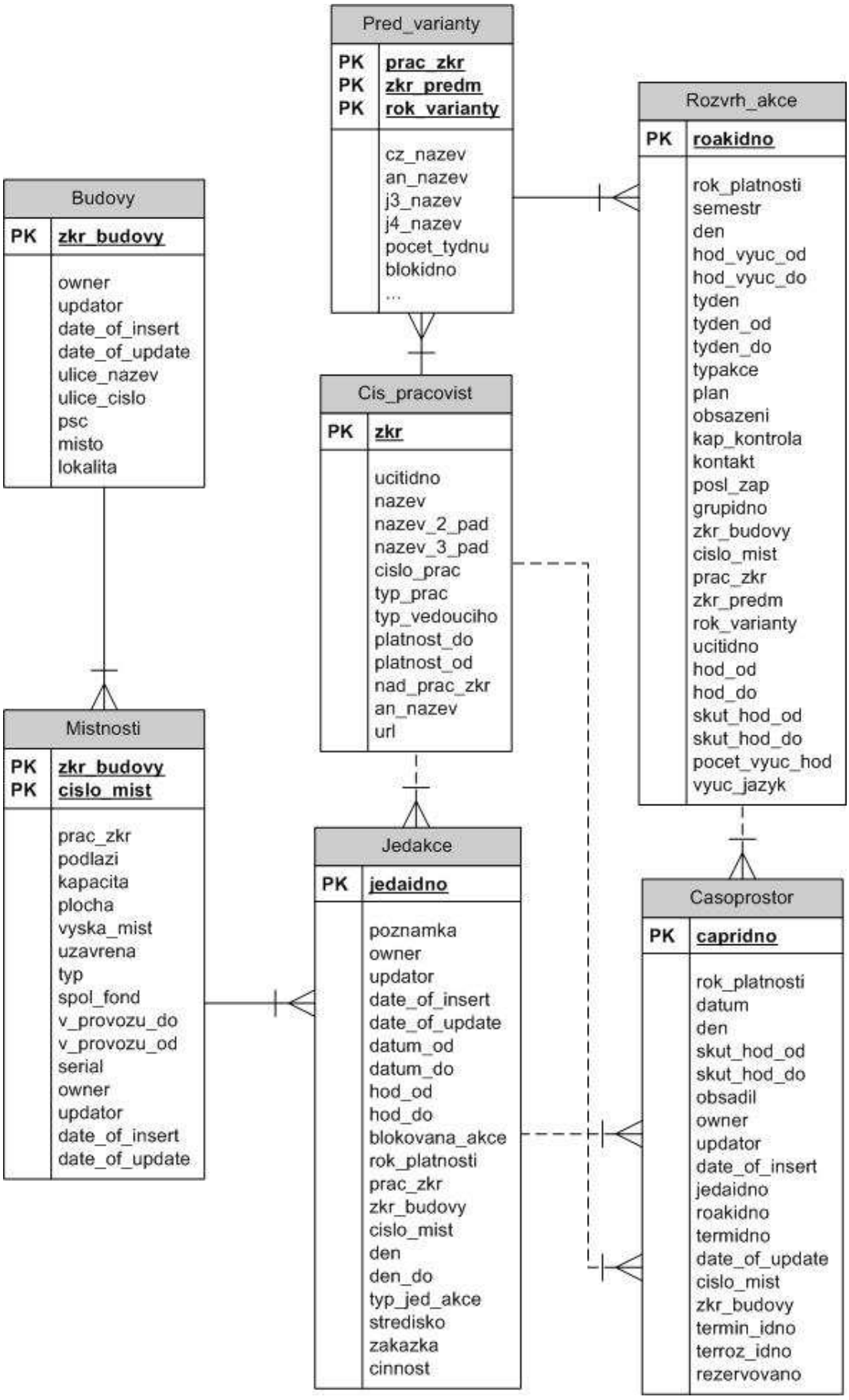
Řešení implementace funkčností navrhovaného datového skladu není však náplní této bakalářské práce, jelikož účelem této práce je pouze návrh datového skladu, a proto není potřeba dělat scénář případů užití. Je ovšem důležité případy užití vytvořit, kvůli ujasnění funkčnosti datového skladu a také kvůli vhodnému zvolení tabulek dimenzí. Zároveň jsou případy užití také nedílnou součástí každého návrhu, protože nejdříve je nutné ujasnit si požadavky uživatelů na systém a teprve potom začít systém budovat.

5.2.3 Zdrojové tabulky

Vytvořením diagramu případů užití se otevřela cesta pro vytipování potřebných zdrojových relací pro tvorbu datového skladu. Jak již bylo uvedeno v úvodní analýze RMD IS STAG, středem schématu zdrojových relací se stala relace Casoprostor (viz obrázek 12).

Ta obsahuje mnoho atributů cizích klíčů, které jsou v ostatních relacích primárními klíči. Jedná se především o atributy jedaidno a roakidno, patřící do relací Jedakce resp. Rozvrh_akce. Nejdůležitější částí relace Casoprostor jsou ovšem atributy skut_hod_od a skut_hod_do, které indikují, v jakém časovém rozmezí jsou dané místnosti obsazeny jednotlivými akcemi. Z těchto dvou atributů lze pak jednoduchou úpravou vytvořit faktovou hodnotu počet hodin, jelikož nebude potřebné znát přesný čas, ale dobu za určitou periodu. Faktová hodnota se tak může vytvořit prostým odečtením atributu skut_hod_od od atributu skut_hod_do. S relací Casoprostor je přímo propojeno mnoho tabulek. Pro potřebu tvorby návrhu datového skladu, tak jak je zamýšlen, se ovšem hodí pouze relace Rozvrh_akce a Jedakce.

Relace Rozvrh_akce obsahuje data popisující vyučované hodiny, které jsou zakotveny v semestrálním rozvrhu. Jedněmi z nejdůležitějších atributů této relace jsou typakce, vztahující se k tomu, zdali se jedná o seminář, přednášku či cvičení, dále zkr_predm, prac_zkr a cislo_mist, které jsou cizími klíči dalších tabulek potřebných pro návrh datového skladu. Relace Rozvrh_akce je přímo propojena s relací Pred_varianty, jež je jednou z nejrozsáhlejších tabulek celého RMD IS STAG, ale jelikož většina jejích atributů není pro tvorbu skladu relevantní, byla zkrácena, tak jak ukazuje obrázek 12. Z této relace jsou potřebné atributy prac_zkr, jenž udává pracoviště, které vyučuje daný předmět, a cz_nazev, obsahující doslovný název daného předmětu v češtině.



Obrázek 12: Zdrojové relace datového skladu (zdroj: vlastní)

Relace Jedakce navazuje na relaci Casoprostor a nachází se v ní informace o jednorázových akcích. Obsahuje také mnohé důležité atributy především typ_jed_akce, který říká, jaká akce se koná, zdali odborná přednáška či seminář atd., stredisko a zakazka, vypovídající o pořadateli akce, a cizí klíče cislo_mist a zkr_budovy, díky nimž je relace Jedakce propojena s relací Mistnosti. Relace Mistnosti monitoruje data spojená s popisem místností, jako jsou číslo místnosti, podlaží, kapacita, výška, plocha atd. Pro potřebu datového skladu nebudou potřeba všechny tyto atributy. Přes atribut zkr_budovy je tato relace propojena s relací Budovy, ve které se sledují informace o budovách, jako jsou PSČ, lokalita, ulice atd.

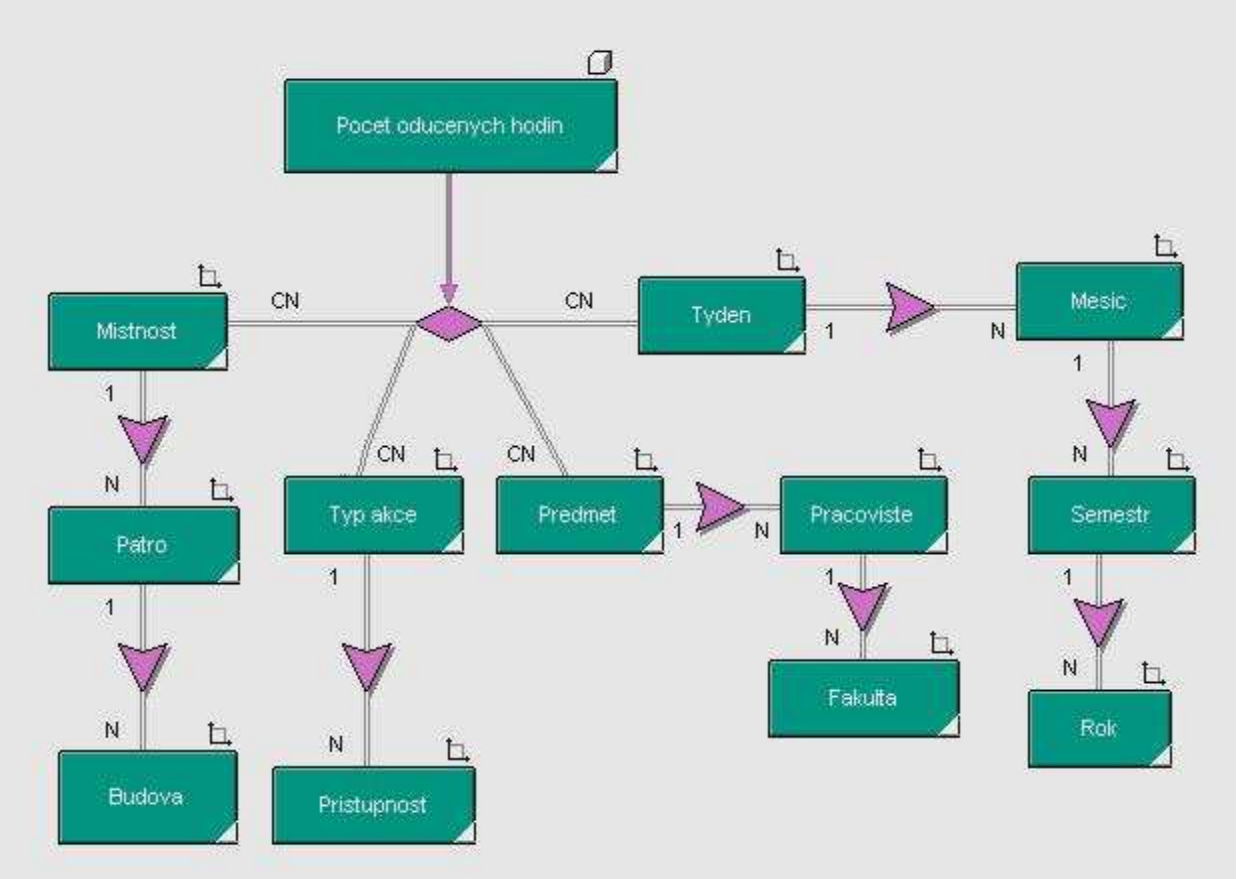
Poslední použitou relací je Cis_pracovist, která doslovně znamená číselník pracovišť. Zde je důležitý především atribut nazev, který určuje přesný název pracoviště, který můžeme určit pomocí atributu zkr resp. predm_zkr, jak je tento atribut uváděn v ostatních relacích. Po úvodní analýze a vytvoření diagramu případů užití byly pro použití při návrhu datového skladu vytipovány tyto relace, ale samozřejmě by mohly být ještě nějaké přidány nebo odebrány.

5.3 Návrh datového skladu nad IS STAG

Z vytipovaných relací bylo již jen třeba vytvořit datový sklad. Ujasnil jsem si, že tabulkou faktů se stane počet hodin, které byly odučeny či odpřednášeny na Univerzitě Pardubice. Ale na co byl tento fakt použit, nemusí být až tak zřejmé. Jestliže by například byly v rozpočtu Univerzity Pardubice nebo i jednotlivých fakult nějaké prostředky na rekonstrukci místností, pater či celých budov, podle čeho bude příslušný orgán přidělovat finance jednotlivým místnostem nebo budovám? Řešením na tuto otázku se měl stát právě navrhovaný datový sklad. Sledováním počtu odučených hodin v jednotlivých místnostech bude vytvořen prioritní seznam místností, protože je zřejmé, že pro místnosti, které nejsou tak využívány jako jiné, není třeba tolik prostředků pro renovaci. Samozřejmě není možné se rozhodovat jenom podle počtu odučených hodin, ale také podle toho, zdali se v dané místnosti konají především přednášky či cvičení, zdali se v dané místnosti pořádají sezení pro veřejnost či jen akademickou obec. Tyto charakteristiky mají také vliv na potřeby vybavení dané místnosti.

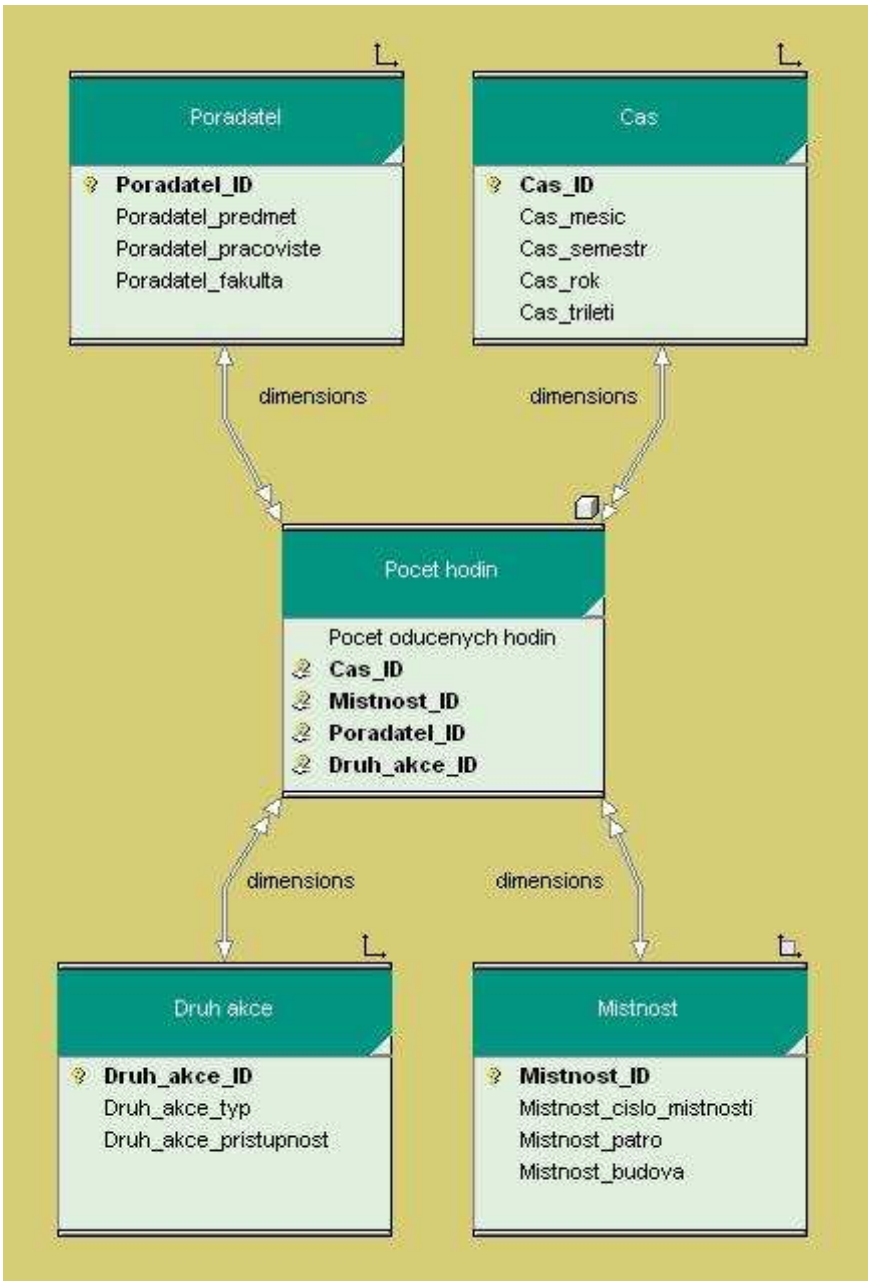
Jako základní podklad pro návrh datového skladu jsem vytvořil jeho konceptuální model (viz obrázek 13). Tento model byl vytvořen v modelovacím SW nástroji case/4/0 (produkt firmy Microtool). Model reprezentuje představu, že budoucí faktová tabulka Počet odučených hodin je provázána se všemi entitami, které jsou zamýšleny jako budoucí základní úrovně hierarchie dimenzí. Tyto entity jsou pak propojeny s entitami vyšších úrovní hierarchie dimenzí. Je tedy

použita vazba 1:N, protože například místnost může být pouze na jednom patře, ale na jednom patře může být více místností. Výsledek tohoto modelu může být ještě upraven při převodu na technologickou úroveň, tedy úroveň samotného návrhu.



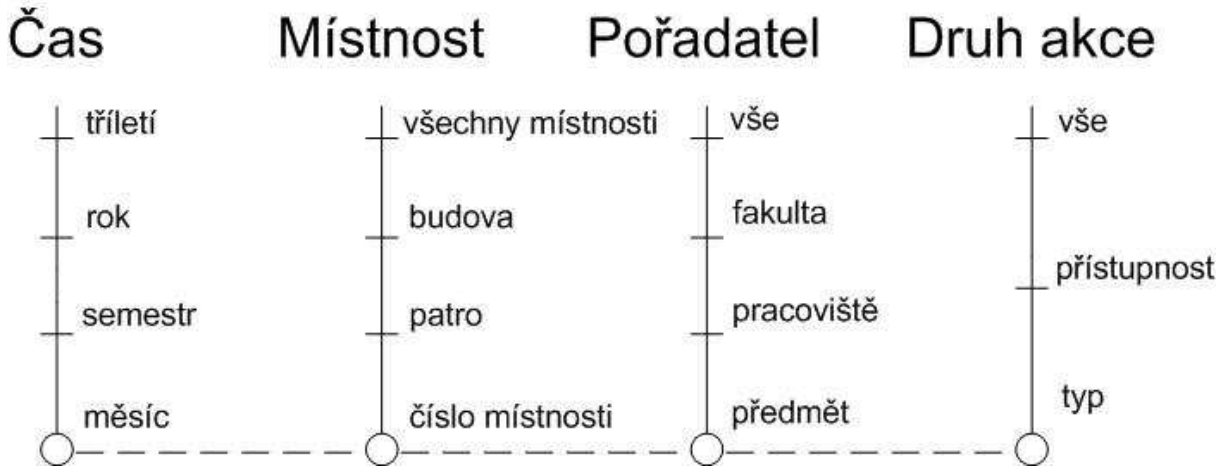
Obrázek 13: Konceptuální model pro datový sklad (zdroj: vlastní)

Z hlediska návrhu datového skladu jsem musel dále rozhodnout, zdali bude vytvořen schématem STAR či SNOWFLAKE. Pro tento konkrétní návrh jsem zvolil schéma STAR, jelikož je obecně rychlejší při poskytování výstupů a navrhovaný sklad nebude obsahovat nadměrný počet agregací ani dat, protože IS STAG nemá mnoho vhodných faktových hodnot, a proto výsek vybraných tabulek není příliš veliký. Předpokladem je, že nebude třeba téměř vůbec měnit prvky v hierarchiích dimenzí, a proto je také vhodnější schéma STAR. Model datového skladu, resp. datového tržiště vytvořený v aplikaci Case 4/0 znázorňuje obrázek 14.



Obrázek 14: Schéma datového skladu - Case 4/0 (zdroj: vlastní)

Na tomto obrázku je vidět, jak je tabulka faktů centrem schématu STAR a tabulky dimenzí jsou rozloženy okolo. Primární klíče dimenzionálních tabulek jsou obsaženy v tabulce faktů jako cizí klíče a všechny dohromady tvoří primární klíč tabulky faktů. Data se do datového skladu ukládají v základním tvaru počet odučených hodin za měsíc pro daný typ akce v rámci určitého předmětu (viz obrázek 15). Jednotlivé tabulky dimenzí nesou tedy název Čas, Místnost, Pořadatel a Druh akce.



Obrázek 15: Základní úroveň hierarchie dimenzí datového skladu (zdroj: vlastní)

Dimenze Čas je nezbytnou součástí datového skladu a má za základní úroveň své hierarchie zvolen měsíc. Samozřejmě, že by bylo možné data ukládat třeba i každý den, ale v důsledku by byla tato činnost zcela zbytečná. První věcí je, že ukládání dat každý den stojí definitivně určité náklady na režii a správu systému. Věcí druhou je, že sledovaný fakt počet odučených hodin není zase až tak proměnlivou hodnotou, jelikož například rozvrhované akce jsou stejné po celý semestr, čili mění se pouze jednorázové akce a těch se uspořádá maximálně několik za měsíc. Proto nebyly zvoleny za základní úroveň ani den, ani týden, ale měsíc. Dalšími úrovněmi dimenze Čas jsou semestr, tedy základní časová jednotka na vysoké škole, rok a tříletí. Takto rozsáhlé časové horizonty byly zvoleny, protože systém navrhovaného datového skladu klade jako prioritu ukládání dat o využívanosti místností za účelem vytipování nejvyužívanějších místností pro potřeby rozhodování ohledně investic do renovací, nového vybavení atd. Nemůže se počítat s tím, že investice budou poskytovány každý měsíc nebo semestr, a proto musí být tato data sledována i v dlouhodobém horizontu.

Dimenze místnost je určena hierarchií číslo místnosti, patro místnosti a budova (viz obrázek 15). Základní úrovní je číslo místnosti, které přesně určuje danou místnost, kdyby bylo potřeba renovovat pouze jednotlivé místnosti. Naproti tomu tato dimenze obsahuje i obecnější atributy patro a budova, které mohou posloužit při rozhodování ohledně řádově vyšších investic. Uživatel tedy bude moci provést výstup v podobě počtu odučených hodin pro určité patro, kdy patra pro jednotlivé budovy musí nést speciální označení kvůli jejich odlišení, nebo i pro celou budovu.

Dimenze Druh akce a Pořadatel byly problémem, protože bylo třeba od sebe rozlišit akce pořádané pouze pro studenty Univerzity Pardubice a na druhé straně akce veřejně přístupné. Nastal tak problém, jak s pojmenováním atributu, tak také s jeho umístěním. Nakonec byl tento

problém vyřešen vytvořením atributu Přístupnost, jenž se stal agregací typů pořádaných akcí. Atribut Přístupnost má dvě hodnoty, a to veřejně přístupné a Univerzita, kdy jednotlivé typy akcí se dělí na přednášky, veřejné přednášky, semináře, veřejné semináře atd. Musela být vytvořena ještě tabulka dimenze Pořadatel. Jelikož se datový sklad zabývá pouze učebními akcemi různých typů, každá z těchto akcí spadá do problematiky určitého předmětu, který se realizuje v rámci určitého pracoviště a to existuje pod správou příslušné fakulty. Díky této dimenzi může být implementována podmínka, že zaměstnanci správy jednotlivých fakult budou mít přístup pouze k datům týkajícím se dané fakulty.

Navrhl jsem tedy datový sklad, resp. datové tržiště. Mluvím zde o tržišti, jelikož se jedná o návrh malých rozměrů (tzn., není zde velký počet dimenzí). Oproti konceptuálnímu modelu byla v samotném návrhu změněna hierarchická struktura dimenze Čas. Dimenze Čas má za základní úroveň zvolen měsíc namísto týdne, protože se předpokládá, že investice či renovace místností jsou otázkou dlouhodobějšího charakteru a tudíž je nepotřebné ukládat data v týdenních intervalech. Navrhnutý model by mohl po implementaci pomoci v rozhodování o tom, jak efektivně vynakládat prostředky Univerzity Pardubice.

6 Závěr

Cílem bakalářské práce bylo na základě RMD IS STAG navrhnout datový sklad, resp. datové tržiště. Práce shrnuje základní principy normalizovaného dimenzionálního modelování včetně charakterizace rozdílů těchto postupů. Dimenzionální modelování je bráno jako základ pro vytvoření dimenzionálního modelu při tvorbě datového skladu. Dále je charakterizován význam technologie datového skladu pro informační podporu v organizacích a jsou uvedeny podmínky, které musí být splněny pro jeho vytvoření.

Bakalářská práce má následující výstupy:

- Na základě analýzy relačního modelu IS STAG byly vytipovány vhodné tabulky pro tvorbu datového skladu pro zaměstnance správy.
- Byl vytvořen model datového tržiště. Vzhledem ke specifičnosti IS STAG není tvorba datového skladu jednoduchou záležitostí z důvodu nedostatku faktových tabulek.
- Implementovaný model by mohl pomáhat při rozhodování ohledně investic do vybavení místností či jejich rekonstrukcích.

Vytvořený návrh datového skladu je jasně pochopitelný a implementovatelný, přičemž má zřejmou funkci, a to ukládání dat o využívání jednotlivých učeben pro pozdější analýzy. Proto konstatuji, že cíl práce byl s přehledem naplněn.

7 Zdroje

[1] *A Centralized Enterprise Data Warehouse Works Best* [online]. 2000 [cit. 2008-12-10]. Dostupný z WWW: <http://www.tdwi.org/research/display.aspx?ID=5313>.

[2] DATE, C.J. An Introduction to Database Systems [s.l.]: Adison-Wesley Publishing Company, Inc., 1985. 385 s. ISBN 0-201-14474-3.

[3] HINCA, P.. *Použití datových skladů v pojistné matematice* [online]. 2003 [cit. 2008-11-17]. Dostupný z WWW: <www.actuaria.cz/upload/prednDW.doc>.

[4] HUMPHRIES, Mark, HAWKINS, Michael W., DY, Michelle C. *Data warehousing : návrh a implementace*. 1. vyd. Praha : Computer Press, 2002. 258 s. ISBN 80-7226-560-1.

[5] HYNKOVÁ, Šárka. *Nasazení Business Intelligence v lékařských knihovnách*. [s.l.], 2006. 70 s. Masarykova Univerzita. Vedoucí bakalářské práce Šimral. Dostupný z WWW: <http://is.muni.cz/th/110973/ff_b/bc_BI_cs.doc>.

[6] KANISOVÁ, Hana, MÜLLER, Miroslav. *UML srozumitelně.* 2. aktualit. vyd. Brno: Computer Press, a.s., 2006. 176 s. ISBN 80-251-1083-4.

[7] KAULER. *Databázové technologie* [online]. 2005 [cit. 2009-04-19]. Dostupný z WWW: <http://webzam.fbmi.cvut.cz/szabozol/ISZ/Kauler/511.doc>.

[8] KŮS, Václav. *Datový sklad pro okresní úřad*. [s.l.], 2001. 80 s. Univerzita Pardubice. Diplomová práce.

[9] NOVOTNÝ, Ota, POUR, Jan, SLÁNSKÝ, David. *Business Intelligence : Jak využít bohatství ve vašich datech*. 1. vyd. Praha : Grada Publishing, a.s, 2005. 256 s. ISBN 80-247-1094-3.

[10] STUPKA, Petr. *Časopis IT Systems: Vytvoření datového skladu pro ČSSZ* [online]. 2002 [cit. 2009-04-23]. Dostupný z WWW: <http://www.systemonline.cz/clanky/vytvoreni-datoveho-skladu-pro-cssz.htm>.

[11] *Teorie relačních databází: Normalizace* [online]. 2007 [cit. 2009-04-19]. Dostupný z WWW: <http://www.manualy.net/article.php?articleID=9>.

[12] *Teorie relačních databází: Relační model dat* [online]. 2006 [cit. 2009-04-17]. Dostupný z WWW: <http://www.manualy.net/article.php?articleID=9>.