EKONOMICKÁ UNIVERZITA V BRATISLAVE
FAKULTA HOSPODÁRSKEJ INFORMATIKY

Evidenčné číslo: 103004/I/2022/421000353535

MultiDim pre konceptuálny model dátového skladu

Diplomová práca

2022 **Bc. Diana Trubirohová**

EKONOMICKÁ UNIVERZITA V BRATISLAVE
FAKULTA HOSPODÁRSKEJ INFORMATIKY

MultiDim pre konceptuálny model dátového skladu

Diplomová práca

Študijný program: Informačný manažment
Študijný odbor: Informačný manažment
Školiace pracovisko: Katedra aplikovanej informatiky FHI
Vedúci záverečnej práce: doc. Dr. Ing. Miroslav Hudec

Bratislava 2022 **Bc. Diana Trubirohová**

Ekonomická univerzita v Bratislave
Fakulta hospodárskej informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Diana Trubirohová
Študijný program: informačný manažment (Jednoodborové štúdium, inžiniersky II. st., denná forma)
Študijný odbor: ekonómia a manažment
Typ záverečnej práce: Inžinierska záverečná práca
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: MultiDim pre konceptuálny model dátového skladu

Anotácia: Cieľom práce je vytvoriť konceptuálny model dátového skladu, zachyteným rôznych typov hierarchií dimenzii, ktorý bude slúžiť ako manuál.

Pre dátové sklady nie je široko akceptovateľný štandard na modelovanie konceptuálneho modelu. Používajú sa modely pre relačné databázy alebo iné modely, prípadne sa vytvára logicky model bez zdokumentovania konceptov. Táto práca vyhodnotí výhody a limitujúce prvky modelovania pomocou MultiDim.

Vedúci: doc. Dr. Ing. Miroslav Hudec
Katedra: KAI FHI - Katedra aplikovanej informatiky FHI
Vedúci katedry: Ing. Mgr. Peter Schmidt, PhD.

Dátum zadania: 30.10.2020

Dátum schválenia: 30.10.2020

Ing. Mgr. Peter Schmidt, PhD.
vedúci katedry

Čestné vyhlásenie

Čestne vyhlasujem, že diplomovú prácu s názvom MultiDim pre konceptuálny model dátového skladu som vypracovala samostatne s využitím štúdia uvedených zdrojov.

Dátum:

.....
(podpis študenta)

Poďakovanie

Touto cestou by som rada poďakovala vedúcemu diplomovej práce doc. Dr. Ing. Miroslavovi Hudecovi za usmerňovanie a odborné konzultácie pri vypracovávaní na záverečnej práce.

Abstrakt

TRUBIROHOVÁ, Diana: *MultiDim pre konceptuálny model dátového skladu*. – Ekonomická univerzita v Bratislave. Fakulta hospodárskej informatiky; Katedra aplikovanej informatiky. – Vedúci záverečnej práce: doc. Dr. Ing. Miroslav Hudec. Bratislava: FHI, 2022, 66 s.

Cieľom záverečnej diplomovej práce je objasnenie konceptuálneho modelovania využitím MultiDim, ktorý je vhodný na modelovanie hierarchii a dimenzii.

Práca je rozdelená do troch hlavných kapitol. V prvej kapitole sa zameriame na súčasný stav problematiky, definíciu multidimenzionálnych dátových skladov a fázam modelovania so zameraním na konceptuálne modelovanie. Taktiež sa pozrieme na využitie MultiDim a grafickej notácii jednotlivých prvkov množstvom schém. V ďalšej kapitole „Ciele, metódy a metodika“ si definujeme cieľ práce a popíšeme postup, ktorý bol počas písania práce aplikovaný. Záverečná kapitola sa zaoberá výsledkami práce a aplikovaniu MultiDim na dve prípadové štúdie, pre ktoré vytvoríme konceptuálne modely.

Kľúčové slová: MultiDim, multidimenzionálne dátové sklady, konceptuálny model

Abstract

TRUBIROHOVÁ, Diana: *MultiDim for conceptual model of data warehouse*. - University of Economics in Bratislava. Faculty of Business Informatics; Department of Applied Informatics. - Thesis supervisor: doc. Dr. Ing. Miroslav Hudec. Bratislava: FHI, 2022, 66 p.

The objective of the final thesis is to clarify conceptual modeling using MultiDim, which is suitable for modeling hierarchies and dimensions in multidimensional models.

The thesis is divided into three main chapters. In the first chapter, we will focus on the current state of problematic, the definition of multidimensional data warehouses and modeling phases with a focus on conceptual modeling. We will also look at the use of MultiDim and the graphical notation of individual elements by a number of schemes. In the next chapter "Objectives, methods and methodology" we define the goal of the work and describe the procedure that was applied during the writing of the thesis. The final chapter deals with the results of the thesis and application of MultiDim on two case studies, for which we create conceptual models.

Keywords: MultiDim, multidimensional data warehouses, conceptual model

Obsah

Zoznam obrázkov

Zoznam tabuliek

Úvod

Príchodom vysokej konkurencie na trhu vznikol dopyt po dostupnom riešení, ktoré by podporovalo rozhodovanie, čím vznikla potreba analýzy multidimenzionálnych dát. Multidimenzionálna databáza tento dopyt uspokojila a umožnila jednoduchú a rýchlu analýzu multidimenzionálnych dát s nízkymi nákladmi. Proces dizajnu dátového skladu nie je značne odlišný od modelovania relačných databáz. Postupne prechádza všetkými fázami ako už široko známy a formalizovaný dizajn relačných databáz. Príchodom multidimenzionálnych skladov sa proces modelovania nezmenil, až na jednu výnimku.

Dizajnéri obchádzajú modelovanie konceptuálneho modelu a venujú sa priamo modelovaniu logického modelu. V tomto kroku by nemusel byť žiadny problém, v prípade, že modelovanie konceptuálneho modelu by nemalo svoje miesto v procese dizajnu databáz a dátových skladov vo všeobecnosti. Počas posledných rokov je široko rozoznávaná potreba konceptuálneho modelovania. Konceptuálny model slúži na komunikáciu so zadávateľom a na definovanie užívateľských požiadaviek, ktoré definujú doménu modelu. Logický model je priamo závislý na implementačnej platforme a hovorí už o implementácii dátového skladu. To však prináša problém.

V práci sa zameriame na definovanie multidimenzionálného skladu a jeho prvkov. Zachytíme aj proces modelovania a bližšie sa zameriame na konceptuálne modelovanie. Určíme dôvod, prečo je konceptuálny model vynechávaný. Priblížime konceptuálne modelovanie multidimenzionálnych dátových skladov využitím modelu MultiDim. Taktiež si povieme, prečo je MultiDim vhodným modelom modelovania konceptuálneho modelu pre dátové sklady.

Výsledkom práce bude vytvorenie konceptuálneho modelu použitím MultiDim, ktorý bude zachytávať rôzne typy hierarchií dimenzií. Nadobudnuté znalosti budú aplikované na dve vybrané prípadové štúdie, pre ktoré bude vytvorený konceptuálny model. Práca by mala slúžiť ako manuál modelovania konceptuálneho modelu pre multidimenzionálny dátový sklad.

1 Súčasný stav

1.1 Definícia dátového skladu

Husemann 2000, definuje dátový sklad nasledovne: „Dátový sklad je vo všeobecnosti chápaný ako integrovaný a časovo premenlivý zber údajov primárne využívaných pri strategickom rozhodovaní prostredníctvom techník online analytického spracovania (OLAP). Ide v podstate o databázu, ktorá uchováva integrované, často historické a súhrnné informácie extrahované z viacerých, heterogénnych, autonómnych, a distribuovaných informačných zdrojov."

Dátový sklad je špecifická štruktúra dát, ktorá je určená pre podporu rozhodovania. Skladá sa z dát, ktoré bežne nájdeme v relačných databázach, ale sú upravené takým spôsobom, aby zabezpečili podporu rozhodovania v biznis analýze. Databázové systémy sú vhodné pre každodenné operácie, ktoré zabezpečujú rýchly prístup k dátam. (Vaisman, 2014) Dáta sú transformované z viacerých zdrojov, interných a externých, tak aby ich štruktúra bola vhodná pre analýzu. Dáta sa integrujú z heterogénnych zdrojov. Dátový sklad je určený pre prácu so súčasnými ale aj historickými dátami. Nie sú však vhodné pre prácu s dátami, nad ktorými prebiehajú časté transakcie. Pre ukladanie dát využívame ETL alebo ELT procesy.

Dátový sklad je subjektovo orientované (orientované na konkrétnu doménu) úložisko dát skladajúce sa z atomických dát. Dáta v dátovom sklade sú nemenné, neodstraňujú sa a neupravujú. Vznikajú tým historické dáta, ktoré sú použité na podporu rozhodovania. Uchované historické dáta sú časovo premenlivé, pretože môžu pochádzať z viacerých časových úsekov. V rámci dátového skladu sa využíva jednotná terminológia, čím sú dáta integrované. (Dátové sklady a OLAP, 2021)

Dátové sklady sú stavané na multidimenzionálnych modeloch. Dáta tu sú reprezentované hyperkockami s rozmermi, ktoré sa zhodujú so sledovanými faktormi spoločnosti. Jednotlivé bunky reprezentujú konkrétne údaje, ktoré budú analyzované. (Vaisman, 2014)

1.2 Multidimenzionálny databázový model

Analýza dát dokáže spoločnostiam priniesť konkurenčnú výhodu na trhu. Tradičné databázové systémy (OLTP) však nie sú vhodné pre analýzu dát. Ich úlohou je zabezpečenie optimálneho prístupu k dátam transakčným spracovaním. „Dátové sklady a OLAP sú založené

na multidimenzionálnom modely, ktorý zobrazuje dáta n-rozmernom priestore, ktorý sa nazýva dátová kocka alebo hyperkocka.“. (Vaisman, 2014) Multidimenzionálny model je vhodný práve na analýzu dát, keďže poskytuje možnosť zložitejších dotazov, bez ich častého modifikovania.

Multidimenzionálna dátová kocka sa skladá zo zložitejších dátových štruktúr ako nájdeme v relačnej databáze. Kocka je definovaná dimenziami, hierarchiami a faktami. Tieto prvky rozoberieme podrobnejšie v nasledujúcej kapitole. Ďalej sa pozrieme práve na rozdiel medzi OLTP, ktoré sa spája s operačnými databázami a OLAP, ktoré nájdeme v multidimenzionálnych modeloch.

1.3 OLTP a OLAP

Rozdiel medzi transakčnými databázami a dátovými skladmi nájdeme v type dát, ktorými sa zaoberajú. Operačné databázy alebo inak **OLTP** (Online Transaction Processing) kladú dôraz na spracovanie transakcií jednoduchým a bezpečným spôsobom. Ide o ukladanie operatívnych údajov. OLTP poskytuje technológiu spracovania dát v reálnom čase. Transakčné databázy sú spravidla normalizované, aby bola zaistená jednoduchosť dotazovania a zamedzilo sa redundancii dát. Databáza musí splňovať aspoň tretiu normálovú formu. Ich výkon je však slabý v prípade dopytov, ktoré spájajú veľké množstvo tabuliek alebo je agregované veľké množstvo dát. (Vaisman, 2014) Výsledkom OLTP dotazov sú tabuľky získané agregačnými funkciami. OLTP je optimalizovaný na zápis dát do štruktúry databázy.

OLTP nám však nedokáže poskytnúť odpovede na všetky otázky, ktoré by sme pre efektívnu analýzu potrebovali poznať. Preto vznikol pojem **OLAP** (Online Analytical Processing). OLAP nám poskytuje možnosť využitia dopytov, ktorých podstata je analytická. OLAP vykonáva multidimenzionálnu analýzu business dát a poskytuje schopnosť komplexných výpočtov, analýzy trendov a sofistikovaného modelovania údajov. Bol navrhnutý a optimalizovaný na rozsiahle analýzy. (Vaisman, 2014)

OLAP umožňuje koncovým používateľom vykonávať analýzu údajov vo viacerých dimenziách, čím poskytuje prehľad a pochopenie, ktoré je potrebné pri rozhodovaní. (OLAP.com, 2021) Systémy online analytického spracovania umožňujú užívateľom vykonávať automatickú agregáciu mier pri prechádzaní hierarchiami: operácia súhrnu transformuje

podrobné merania na agregované hodnoty (napr. denné na mesačné alebo ročné tržby), zatiaľ čo operácia rozbalenia vykonáva opak. (Malinowski, 2008) Operáciám nad hierarchiami sa budeme zaoberať v práci neskôr.

Informácie sú veľmi drahé a ich znalosť vie smerovať organizáciu k lepšej konkurencieschopnosti. Schopnosť OLAP vytvárať veľmi rýchle agregácie a výpočty základných množín údajov, je užitočná pre spoločnosti pri robení lepších a rýchlejších rozhodnutí na základe informácií z analýzy, ktoré im multidimenzionálny OLAP poskytol. OLAP ako prostriedok dotazovania využíva agregáciu dát. Spracovanie takýchto dopytov vytvára veľkú záťaž na systém, tým, že sú spracovávané všetky záznamy v systéme, ktorých v prípade historických dát môže dosahovať veľké množstvo. (Vaisman, 2014) Obsahuje read-only dáta, ktoré sú vyhodnocované omnoho rýchlejšie.

Môžeme tak povedať, že jeden z najzakladanejších rozdielov medzi OLTP a OLAP vyplýva z možnosti ich použitia. Kým OLTP sú dáta priebežne, no ale aj často pridávané a modifikované, OLAP vykonáva zložité dopyty nad dátami, ktoré boli jednorázovo nahraté do dátového skladu a umožňuje nad nimi vykonávať zložité analýzy.

S príchodom OLAP vznikla potreba vytvorenia novej databázovej štruktúry – dátového skladu. Dátový sklad okrem podpory analytických dopytov, slúži aj na podporu data miningu, reportovania, štatistickej analýzy a ďalších analytických úloh. (Vaisman, 2014) Tie slúžia ako podklady pre rozhodovanie a riadenie procesov. Dáta sú uložené v optimalizovanej multidimenzionálnej databáze, zatiaľ čo pohľady na údaje sa vytvárajú podľa dopytu. Analytik si takto vie zobraziť iba pohľad, ktorý ho konkrétne zaujíma.

1.3.1 Typy OLAP

Dáta v OLAP sú ukladané rôznymi spôsobmi, ktoré sa rozlišujú na základe techniky používanej na usporiadanie a ukladanie údajov a súvisiacich položiek v databáze. Ide o usporiadanie relačné, viacrozmerné alebo hybridné (kombinácia relačných aj viacrozmerných), odkiaľ možno údaje získať a zapojiť do zložených analytických výpočtov. Následne rozoberieme rozdiely medzi tromi základnými typmi usporiadania údajov pomocou OLAP:

1.3.1.1 Relačný OLAP - ROLAP

Multidimenzionalita dát môže byť spracovaná pomocou relačných databáz. Relačný OLAP stojí na predpoklade, že dáta nemusia byť uložené viacrozmerne na to, aby mohli byť multidimenzionálne zobrazené. Dáta sú uložené v relačnej databáze a využívajú funkcie, ktoré relačná databáza ponúka. Relačný OLAP má možnosť priameho prístupu k dátam uloženým v relačným databázach. (Olap.com, 2022) Dáta sú poskytnuté užívateľovi v multidimenzionálnom pohľade. Nevzniká tu redundancia.

Výhodou je implementácia ROLAP, ak podnik už má integrovanú relačnú databázu v RDBMS systéme. Dáta sú uložené v relačnej databáze odkiaľ sú pomocou SQL príkazov vytvorené užívateľské dopyty. (Dátové sklady a OLAP, 2021) ROLAP je založený na **star** schéme.

1.3.1.2 Multidimenzionálny OLAP - MOLAP

Dáta sú modelované v multidimenzionálnom prostredí. Ich štruktúra nie sú tabuľky ako v relačnej databáze, ale štruktúra dát je reprezentovaná **kockou**. Kocka je daná viacrozmerným polom. Jeden záznam v tabuľke v relačnej databázy je prienikom dvoch dimenzií. V kocke môže byť jeden záznam prienikom nekonečného množstva dimenzií. (Olap.com, 2022)

Multidimenzionálna databáza je unikátna v tom, že umožňuje jej užívateľom definovať množstvo dimenzií bez nutnosti pridania ďalšej tabuľky, ako to je v relačnej databáze. V dimenzii prebieha množstvo medzivýpočtov, ktoré to umožňujú. Modelovanie dát a kalkulácia výsledkov je v multidimenzionálnom OLAP flexibilná a rýchla. Dôvodom je aj rýchle vyhľadanie dát, na základe priesečníku mena dimenzií. V relačnej databáze takéto vyhľadávanie je zložitejšie, pretože prebieha na základe indexu, alebo prehľadávania celého modelu SQL príkazom. (Olap.com, 2022)

Ako ďalšiu výhodu môžeme uviesť efektívne ukladanie údajov a ich spracovanie, ktoré v porovnaní k relačným databázam trvá omnoho kratšie. Užívateľ má tak rýchly prístup k dátam. Viacrozmerné modely poznajú techniky spracovania polí a algoritmy na správu dát a ich výpočtov. (Olap.com, 2022)

Nemôžeme zabudnúť aj na nevýhodu MOLAP. Migrácia relačného systému na multidimenzionálny model môže pre podnik znamenať vysoké náklady a redundanciu dát pri prenášaní dát do MOLAP. Dáta pred prenosom do MOLAP štruktúry kocky musia byť upravené.

1.3.1.3 Hybridný OLAP - HOLAP

Hybridný OLAP je kombináciou ROLAP a HOLAP. Dáta sú v relačnej databáze pričom agregáty sú vo viacrozmernej štruktúre. (Dátové sklady a OLAP, 2021) HOLAP umožňuje ukladanie veľkých objemov dát údajov. Na jednej strane HOLAP využíva väčšiu škálovateľnosť ako ROLAP. Na druhej strane, HOLAP využíva technológiu kocky na rýchlejší výkon a na informácie typu súhrnu. Kocky sú menšie ako MOLAP, pretože podrobné údaje sa uchovávajú v relačnej databáze.

1.4 Výhody a nevýhody využitia multidimenzionálného modelu

Vyššie sme popísali rozdiel medzi systémom OLTP a OLAP. Ako môžeme vidieť, podľa výberu systému, vieme dosiahnuť požadovaný výsledok. OLTP aj OLAP však majú svoje výhody a nevýhody.

Jednou z mnoho výhod multidimenzionálného modelovania je, že viacrozmerný model sa blíži k reálnemu zmýšľaniu analytikov, a preto pomáha užívateľom porozumieť údajom, ktoré sú v ňom uchovávané. Taktiež tento prístup pomáha predpovedať, čo chcú koneční užívatelia robiť, a tým zlepšuje výkon a napomáha pri rozhodovaní.

Multidimenzionálny model nám poskytuje možnosť využitia komplexných analýz a jednoduchý a rýchly prístup k veľkému množstvu údajov. Na druhej strane multidim je komplexnejší. Každá zmena štruktúry je náročnejšia a vzniká tu takisto aj veľký predpoklad redundancie dát. Nasledujúce tabuľky nám prehľadne popisujú výhody a nevýhody využitia OLAP teda multidimenzionálného modelu v porovnaní s relačným modelom OLTP.

1.4.1 Výhody

Tabuľka 1. popisuje výhody multidimenzionálného modelu s porovnaním výhod relačného modelu. (Dátové sklady a OLAP, 2021)

	Relačný model	**Multidimenzionálny model**
+	využitie v transakčných databázach, aj dátových skladoch	prístup k multidimenzionálnym aj relačným dátovým štruktúram
+	väčšie množstvo odborníkov so znalosťami práce s relačným modelom	možnosť komplexných analýz
+	normalizácia a odstránenie redundancie	schopnosti pre modelovanie a prognózy
+	jednoduchšia prezentácia dát	rýchly a komplexný prístup k veľkému množstvu údajov
+		rýchla analýza alternatívnych scenárov

Tab. 1 - Výhody multidimenzionálného modelu

1.4.2 Nevýhody

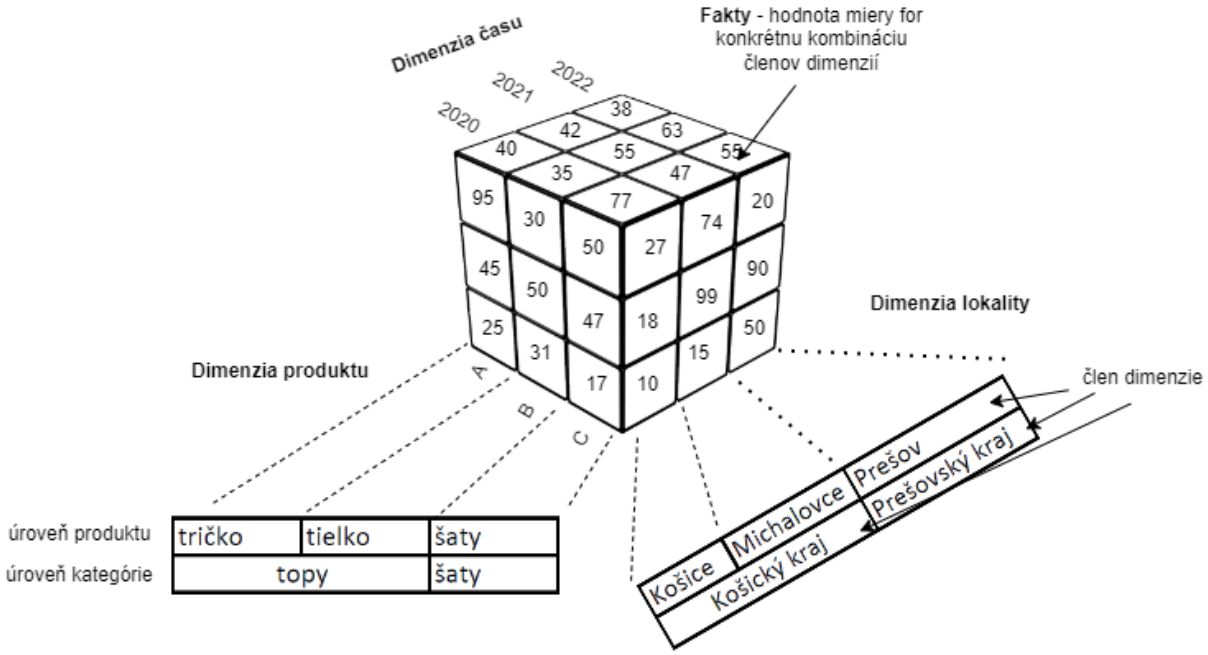
Tabuľka 2. popisuje nevýhody multidimenzionálného modelu s porovnaním nevýhody relačného modelu. (Dátové sklady a OLAP, 2021)

	Relačný model	**Multidimenzionálny model**
-	potenciálne kapacitné obmedzenia	menšia flexibilita pri zmenách dimenzii
-	optimalizácia prístupu k dátam v čase	vyššie kapacitné nároky
-	vyšší počet tabuliek	redundancia dát
-	absencia možnosti komplexnej analýzy	nutnosť organizovania dát do zložitých štruktúr (star, snowflake schéma)
-		nemožnosť prístupu k transakčným dátam

Tab. 2 - Nevýhody multidimenzionálného modelu

1.5 Multidimenzionálna dátová kocka

Dáta v OLAP sú reprezentované dátovou kockou alebo hyperkockou, v prípade ak kocka obsahuje viac ako tri dimenzie, Dátová kocka umožňuje modelovať a zobrazovať dáta na n-rozmernom priestore. Je to dátová štruktúra prevyšujúca obmedzenia, ktoré prinášajú relačné databázy. Príklad dátovej kocky môžeme vidieť na obrázku 1.



Obr. 1 – Príklad dátovej kocky s dimenziami a faktami

V tejto kapitole sa pozrieme bližšie na dátové prvky OLAP kocky. Dátová kocka je množina definovaná **faktami** a **dimenziami**. Multidimenzionálny dátový model sa skladá z dimenzionálnych a faktorových tabuliek.

1.5.1 Dimenzie

Dimenzia je abstraktný pojem zoskupujúci dáta, ktoré majú spoločný sémantický význam v rámci modelovanej domény. (Malinowski, 2008) Dimenzia hovorí o úrovniach, ktoré budú analyzované. V kocke je reprezentovaná hranou kocky. V obrázku 1. môžeme vidieť ako dimenzia produktu je zároveň aj hranou kocky. Každá dimenzia je zoskupením spoločných

alebo súvisiacich stĺpcov z jednej, alebo viacerých tabuliek do jednej entity. (Oracle, 2021) Dimenzia reprezentuje perspektívu alebo pohľad, z ktorého budú dáta analyzované.

Dimenzia konečná diskrétna množina skladajúca sa z logicky a hierarchicky usporiadaných súvisiacich objektov a atribútov, ktoré poskytujú informácie o faktoch z kocky. (Vaisman, 2014) Dáta v nich nie sú často aktualizované. Dimenzie sú obsiahnuté v množine, ktorú nazývame schéma. Minimálne jedna dimenzia v dátovej kocke by mala byť časová.

Dimenzia je uložená v tabuľke dimenzií. Každý stĺpec reprezentuje úroveň v hierarchii. V star schéme sú všetky stĺpce definované v rovnakej tabuľke. (Oracle, 2021) Dimenzia sa skladá z jednej úrovne, alebo jednej alebo viacerých hierarchií. (Malinowski, 2008)

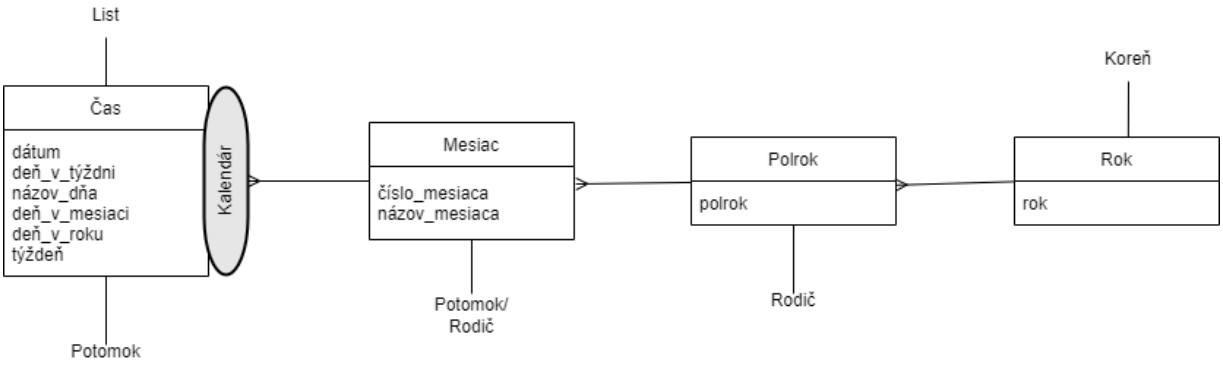
Príklad: Dátová kocka obsahuje dimenzie produkt, čas a lokalita. Dimenzia Produktu sa môže skladať zo stĺpcov ako sú kategórie, sub-kategórie a produkty.

1.5.2 Hierarchie

Hierarchie sú základným prvkom dimenzionálneho modelovania, ktoré popisujú hierarchické vzťahy medzi členmi dimenzií. Dáta sú v hierarchii reprezentované na rôznych úrovniach abstrakcie. (Vaisman, 2014) Hierarchia poskytuje poradie úrovniam v rámci dimenzie. (Oracle, 2021) „Hierarchia znázorňuje vzťahy medzi skupinami stĺpcov v tabuľke dimenzií.“ (Oracle, 2022) Je to množina úrovní, ktoré medzi sebou majú väzbu 1:n. Množina tak vytvára dimenziu, ktorá má stromovú štruktúru. Inštancie dimenzie nazývame **members** (členovia), ktoré sú usporiadané v hierarchickej štruktúre. (Oracle, 2022) Hierarchia definuje vzťahy medzi nadriadeným členom (rodič) a podradeným členom (potomkom). Nadriadený člen predstavuje usporiadanie členov, ktoré sú jeho potomkami. Nadradený člen môže byť takisto agregovaný ako podriadený člen inému rodičovi. (Dimensional Data Modeling, 2022)

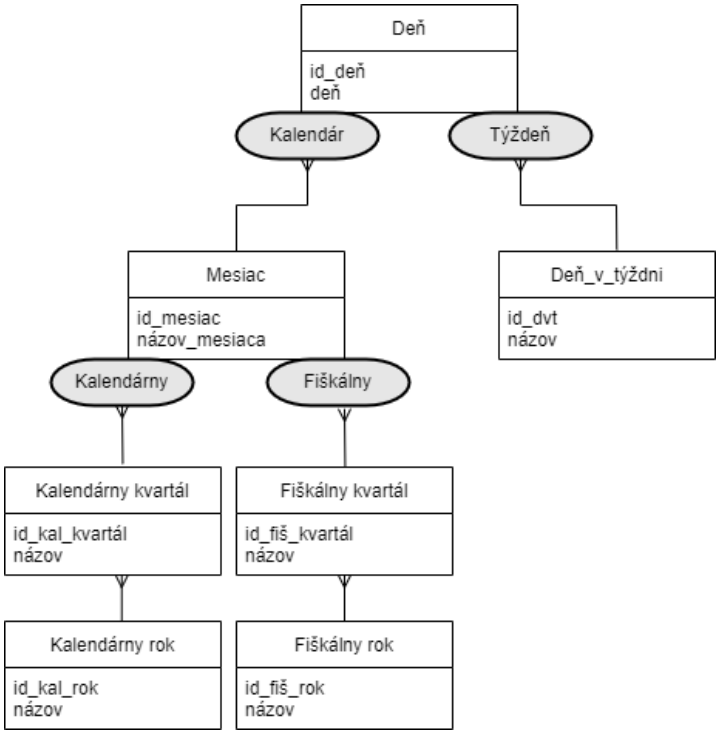
Každá os kocky môže mať na seba naviazané úrovne, ktoré zobrazujú dáta viac, či menej detailnejšie. Ako príklad uveďme zobrazenie predaja za jednotlivé mesiace alebo za jednotlivé roky predaja spoločnosti. Hierarchie dimenzie, dimenzionálna schéma, umožňuje možnosť zovšeobecnenia modelu, alebo naopak, zobrazenia jeho detailnejšej podoby. Tým sa zisťuje prechod medzi úrovňami. Vyššiu úroveň, v tomto prípade jednotlivé roky predaja, nazývame **parent** (rodič). Hierarchia jednotlivých mesiacov predaja, ktorá má nižšiu úroveň, sa nazýva **child** (potomok). Celková úroveň **root** (koreň), pod ktorou sú podradené všetky hie-

rarchie sa používa na agregovanie celej hierarchie. Agregácia zlúči ceny všetkých produktov do jedenej, čím získame tržbu za všetky produkty. (Vaisman, 2014) Hierarchie sa využívajú na vytvorenie zmysluplných agregačných ciest v rámci schémy. Hierarchia zahŕňa niekoľko súvisiacich úrovní, napr. úroveň mesiac a rok. Tieto vzťahy sa používajú na prechod z jednej úrovne do ďalšej. (Malinowski, 2008) „Všetky hierarchie pre danú dimenziu musia mať spoločnú najnižšiu úroveň“ (Oracle, 2022)



Obr. 2 – Vzťah medzi úrovňami dimenzie

Príklad (Obrázok 2): Úroveň čas je potomok úrovne mesiac. Úroveň mesiac je zároveň rodičom úrovne čas ale aj potomkom úrovne polrok. Úroveň polrok je rodičom úrovne mesiac. Úroveň rok je koreňová úroveň dimenzie.



Obr. 3 – Vzťah medzi úrovňami dimenzie (Hudec, 2022)

Príklad (Obrázok 3): Dimenzia čas bude obsahovať fiškálnu hierarchiu a hierarchiu kalendára. Ich spoločná najnižšia úroveň bude deň.

1.5.3 Typy hierarchii

Hierarchie môžu byť analyzované na základe rôznych kritérií a môžu takisto zdieľať rôzne úrovne hierarchií. (Malinowski, 2004) V tejto časti sa sústredíme na typy hierarchií, ktoré sa využívajú v multidimen-zionálnom modely na základe analyzovaných kritérií a zdieľaných úrovní hierarchií. Príklady k jednotlivým hierarchiám budú uvedené neskôr, v časti konceptuálneho modelovania a gra-fickej notácie prvkov MultiDim.

1.5.3.1 Jednoduchá hierarchia

Jednoduchá hierarchia reprezentuje väzbu medzi svojimi členmi stromovou štruktúrou. Jednoduchá hierarchia sa ďalej delí na symetrickú, asymetrickú a generalizovanú hierarchiu. (Malinowski, 2004) V prípade, že všetky úrovne sú v jednej tabuľke tak jedná sa o star sché-mu. Snowflake schéma popisuje prípad, kedy každá úroveň má vlastnú tabuľku.

Jednoduchá hierarchia sa delí na vyváženú a nevyváženú hierarchiu podľa toho, či sú jednotlivé úrovne v hierarchii povinné. V prípade nepovinnosti všetkých úrovní vznikajú možnosti viacerých ciest v hierarchii.

1.5.3.2 Generalizovaná hierarchia

Generalizovaná hierarchia popisuje situáciu, kedy členovia na jednej úrovni môžu byť rozdelení do dvoch kategórii na inej úrovni a majú rôznu granuralitu. V relačnom modely by bola daná situácia reprezentovaná generalizáciou. Miery budú v tomto prípade agregované podľa typu rozdielne. (Vaisman, 2014)

1.5.3.3 Alternatívna hierarchia

Alternatívna hierarchia obsahuje viac hierarchií, ktoré zároveň zdieľajú spoločnú úroveň, prinajmenšom najnižšiu úroveň (úroveň listu) hierarchie stromu. Hrany dimenzie majú rovnaký analytický význam. Star schéma hovorí o situácii, keď všetky úrovne sú v jednej tabuľke. Naopak v snowflake schéme má každý bod na hrane vlastnú tabuľku. (Vaisman, 2014)

Rozdielom medzi generalizovanou a alternatívnou hierarchiou spočíva v ceste, ktorá môže byť vybraná pre analýzu, pretože sú reprezentované rozdielne situácie. V generalizovanej hierarchii môže byť vybratá len jedna cesta. Naopak v alternatívnej hierarchii, ako aj jej názov naznačuje, podriadený člen súvisí v viacerými cestami a cesta pre analýzu môže byť vybratá podľa požiadavkou. (Vaisman, 2014)

1.5.3.4 Paralelná hierarchia

Paralelné hierarchie sú definované v situáciách, keď dimenzia spája niekoľko hierarchií, ktoré zodpovedajú odlišným kritériám analýzy. Paralelná hierarchia môže byť zložená z viacerých druhov hierarchií. Delí sa na základe toho, či hierarchia zdieľa s inými hierarchiami úroveň, alebo nezdieľa, na závislé a nezávislé. (Malinowski, 2004) V prípade paralelne závislej hierarchie, kde sú niektoré úrovne zdieľané. Naopak paralelne nezávislé hierarchie sú združené iba s jednou dimenziou.

Paralelná a alternatívna hierarchia môžu byť jednoducho rozlíšiteľné na úrovni konceptuálneho modelu. Tieto hierarchie je však ťažšie rozlíšiť na úrovni logického modelu, čo nás privádza k dôležitosti konceptuálneho modelovania pre dátové sklady. (Vaisman, 2014)

1.5.3.5 Nestriktná hierarchia

Nestriktná hierarchia už nepopisuje vzťah 1:n, kde nadriadený člen môže mať vzťah k viacerým podriadeným členom hierarchie ale podriadený člen len k jednému nadriadenému členovi. Nestriktná hierarchia popisuje situácie z reálneho sveta, kedy medzi nadriadeným a podriadeným členom existuje m:n väzba. Ak je hierarchia nestriktná, musí byť v hierarchii aspoň jedna m:n väzba.

Príklad: Kategória obsahuje viac produktov a produkt môže byť obsiahnutý vo viacerých kategóriách.

Väzba m:n pri nestriktnej hierarchii prináša komplikácie pri agregáciách a vzniká tu problém s dvojitým súčtom. Keďže člen má m:n väzbu s ďalším členom, pri operácii roll-up je výsledok skreslený nesprávnym započítaným miery. Operáciám nad hierarchiami sa budeme venovať v ďalšej kapitole, kde sa bližšie pozrieme aj na operáciu roll-up. Tento problém by sa dal vyriešiť transformovaným nestriktnej hierarchie do striktnej hierarchie. Pre každú m:n väzbu by bol vytvorený nový člen v oboch hierarchiách spojených m:n väzbou. (Vaisman, 2014)

Vaisman (2014), navrhuje aj ďalšie riešenia zaobchádzania s nestriktnou hierarchiou. Ďalej sa im venovať nebudeme, keďže to nie je predmetom práce. Výsledkom však je, že každé riešenie má svoje výhody a nevýhody a riešenie musí byť zvolené na základe používateľských požiadaviek.

1.5.4 Operácie nad hierarchiami

Multidimenzionálny model je založený na zobrazovaní dát z rôznych perspektív a rôznych úrovní podrobnosti. Operácie uskutočňované nad OLAP vytvárajú prostredie pre interaktívnu analýzu dát. Poznáme päť základných operácií vykonávaných nad OLAP hierarchiami:

1. **Roll-up** - Roll-up zmenšuje počet dimenzií tým, že v hierarchii stúpa smerom nahor. Zoskupuje dáta, takým spôsobom, že z menších a podrobnejších celkov vznikajú väčšie celky. Prebieha zovšeobecnenie. Počas tohto procesu je aspoň jedna dimenzia odstránená. Ide o vystupovanie do vyššej úrovne dimenzie. (Dátové sklady a OLAP, 2021)

2. **Drill-down** - Operácia, ktorá je opakom k roll-up. Dáta sú rozložené do menších celkov. Zo všeobecnejšieho modelu vzniká podrobnejší. Počet dimenzií sa zväčšuje, a teda dimenzie sú do schém pridávané. V hierarchii sa posúvame smerom dolu. Zostupujeme do nižšej úrovne dimenzie.
3. **Drill-across** - Operácia spojenia dvoch kociek do jednej. Každá bunka obsahuje miery oboch kociek, ktoré boli definované rovnakými schémami.
4. **Slice** - Operáciou slice dokážeme vytvoriť novú subkocku na základe vybranej dimenzie. Subkocka je vytvorená projekciou cez jednu dimenziu.
5. **Dice** - Operácie dice je podobná operácii slice. Sub-kocka je vytvorená dvoma a viac dimenziami projekciou cez viac dimenziií (Dátové sklady a OLAP, 2021)
6. **Pivot** - Osi kocky sú rotované, čím je vytvorená nová perspektíva zobrazenia dát.

1.5.5 Faktorová tabuľka

Druhým typom tabuľky je faktorová tabuľka. Faktorovú tabuľku môžeme definovať ako hlavný bod záujmu domény. (Trujillo, 2001) Fakty sú numerické miery určené hodnotami, podľa ktorých analyzujeme dáta. Každá úroveň môže mať jeden alebo viac identifikátorov, ktorý je unikátny a môže pozostávať z jedného alebo viacerých atribútov. (Vaisman, 2014) Faktorová tabuľka pozostáva z dvoch typov stĺpcov: stĺpce popisujúce atribúty a stĺpce popisujúce miery. Stĺpce s atribútom sú určené pre cudzie kľúče, ktoré vytvárajú vzťahy k dimenzionálnym tabuľkám ale aj ďalšie atribúty. Taktiež definujú skupiny atribútov, podľa, ktorých budú vytvárané miery. Stĺpce s mierami sú definované numerickou mernou jednotkou. Tie sú potom analyzované rôznymi dimenziami.

Miera je daná agregačnou funkciou, ktorá definuje hodnotu viacerých mier do jednej hodnoty. (Vaisman, 2014) Môžeme využívať rôzne pravidlá agregácie. Najbežnejším pravidlom je súčet ale môže ísť aj o priemer, medián, počet, počet odlišných hodnôt a štandardnú odchýlku, maximum či minimum a iné.

Aby mieram mohli byť zabezpečené konzistentné výsledky agregácie naprieč všetkými dimenziami, musia byť sumarizovateľné. Na zaistenie sumarizovateľnosti musí miera spĺňať nasledujúce podmienky: (Vaisman, 2014)

1. **Nesúvislosť inštancií** - Zoskupenie inštancií na úrovni musí viesť k nesúvislým podmnožinám. To sa musí diať vzhľadom na ich nadriadený člen ďalšej úrovne.
2. **Úplnosť** - Všetky inštancie miery musia byť zahrnuté v hierarchii. Inštancie musia súvisieť s jedným rodičom na ďalšej úrovni. V prípade nerešpektovania podmienky, výsledky agregácie nebudú správne.
3. **Korektnosť** - Miera je korektná ak je jej aplikovaná správna agregačná funkcia na základe jej typu

Tabuľka faktov je najväčšia tabuľka v databáze. Jej objem dát je veľký. Je teda väčšia ako tabuľky dimenzii. Miery faktov delíme na aditívne, semiaditívne, neaditívne a odvodené, podľa toho ako ich agregujeme.

- **Aditívne** - Najviac využívaný typ faktu, ktorý možno agregovať cez všetky dimenzie vo faktorovej tabuľke. *Príklad: operácia súčtu*
- **Semiaditívne** - Miera, ktorú je možno agregovať cez niektoré dimenzie faktorovej tabuľky ale nie všetky. *Príklad- cena za položku*
- **Neaditívne** - Mieru nie je možné zmysluplne sčítať naprieč dimenziami. *Príklad: vypočítanie výmenného kurzu meny*
- **Odvodené** - Miery vypočítané na základe iných mier alebo atribútov v schéme. *Príklad: cena celej objednávky*

1.6 Modelovanie dátových skladov

Modelovanie dátových skladov je vo väčšine vedeckej literatúry obsiahnuté iba z pohľadu fyzického alebo logického modelu. Konceptuálny model v literatúre často spomínaný nie je. Celkovo sa málo hovorí o tom ako realizovať konceptuálny model vychádzajúci z požiadaviek užívateľa aj napriek tomu, aký je dôležitý pre modelovanie dátových skladov. Pri budovaní informačného systému, ktoré práve často dátové sklady využívajú, je konceptuálny model jeho nevyhnutnou súčasťou. Ten ho plne dokumentuje jeho funkcionality a požiadavky užívateľov, ktoré spĺňa. (Golfarelli, 1998) V tejto kapitole sa zameriame na rozdiel medzi modelovaním relačných databáz a dátových skladov, prečo nemôžeme použiť relačný konceptuálny model. Taktiež objasníme dôvod vynechávania konceptuálneho modelu v modelovaní dátových skladov a prečo dizajnéri priamo siahajú po logickom modely a jeho schémach.

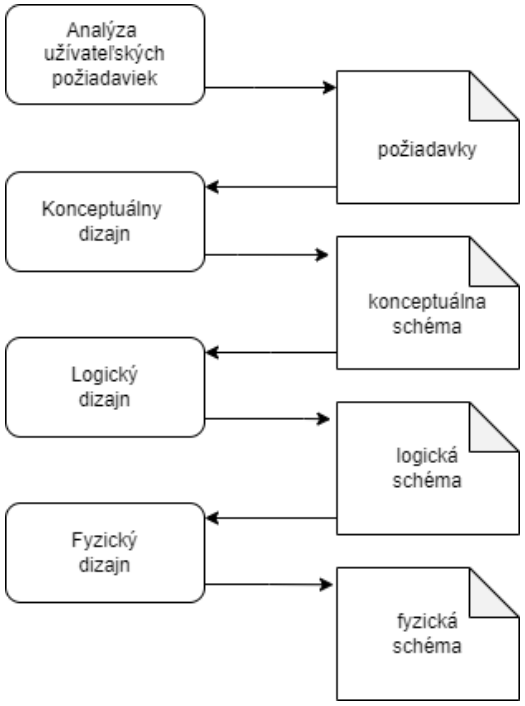
Využívanie entitno-relačného modelu (E/R) pri modelovaná dátových skladov podľa Kimballa 1996, nie je dobre pochopiteľný pre užívateľov a DBMS softvér s nim nedokáže pracovať. Dôvodom je absencia množstva prvkov, ktoré sú pre model dátového skladu nevyhnutné. Preto nie je vhodné využívanie entitno-relačného modelu ako stavebný prvok modelovania dátových skladov. (Kimbal, 1996)

Aj napriek tomuto poznatku, množstvo informačných systémov za posledné roky je vybudovaných práve na entitno-relačných modeloch. Častokrát E/R modelovanie je neúplne a nesprávne a jedinou dostupnou dokumentáciou systému sú logické relačné schémy. (Golfarelli, 1998)

Konceptuálne multidimenzionálne modelovanie sa zameriava na poskytnutie vysokej úrovne abstrakcie pre opis procesu dátového skladu a jeho architektúry, nezávisle od implementácie. Je široko akceptovaný ako jedna z hlavných častí celkového procesu vývoja dátového skladu. V posledných rokoch sa veľa práce venovalo práve konceptuálnemu multidimenzionálnemu modelovaniu, no aj napriek tomu žiadny formalizmus nebol zatiaľ zavedený. (Golfarelli, 1998)

Najväčší rozdiel medzi modelovaním transakčnej databázy a dátovým skladom je denormalizácia. Metódy, ktoré boli doposiaľ vytvorené, vidia potrebu v rozlišovaní konceptuál-

nej fázy modelovania a fázy logického modelovania. Konceptuálny model je zameraný na implementačne nezávislé požiadavky užívateľa a štruktúry zdrojových databáz. Logický model zoberie konceptuálny model a vytvorí logickú schému na vybranej platforme a jej požiadaviek. Tu sú zohľadnené aj obmedzenia a potreby vybranej platformy, či sa už jedná o indexovanie alebo alokáciu. (Rizzi, 2006) V obrázku 4. je popísaný vzťah medzi týmito fázami:



Obr. 4- Fázy procesu modelovania (Vaisman, 2014)

V praxi sa tieto fázy bežne prelínajú a je často krát potrebné sa vrátiť k predchádzajúcej fáze a doplniť model o poznatky nadobudnuté v ďalších fázach. Ďalej si opíšeme fázy modelovania a potrebu konceptuálneho modelovania v relačných databázach. Modelovanie relačných databáz a dátových skladov je úzko prepojené a podobné. Dátové sklady aj napriek tomu majú potrebuje rozšírenejších prvkov ako relačné databázy.

Návrh databázy sa skladá zo štyroch úrovní, ktoré je potrebné zohľadniť. Tieto úrovne využívame z toho dôvodu, aby bola zaistená **nezávislosť dát**. Aby v prípade zmeny v nižšej úrovni, bola vyššia úroveň ovplyvnená čo v najmenšej miere. Úrovne sú rozlíšené úrovňami ich abstrakcie. Všetky modely sú modelované pre určitý účel, ktorý potrebujeme poznať, aby

konkrétny model vedel reprezentovať ten aspekt, ktorý je dôležitý pre danú úroveň. V ďalších kapitolách sa pozrieme v skratke na jednotlivé úrovne modelovania.

1.6.1 Špecifikácia požiadaviek

Ako prvé je potrebné pozbierať a pochopiť, čo užívateľ od systému požaduje a aké má očakávania. Táto fáza je veľmi dôležitá a vyžaduje aktívnu účasť zadávateľa a budúcich užívateľov, pretože určí smerovanie návrhu a implementácie systému. Všetky vzniknuté nepresnosti a odchýlky nám spôsobia zvýšené náklady, ktorým sa môžeme v počiatočnej fáze ľahko vyhnúť a tým aj minimalizovať náklady na opravu v ďalších úrovniach. (Vaisman, 2014) V tomto momente je dôležité definovať cieľ, rozsah, analyzovať ukladané údaje a následne k čomu budú údaje v systéme používané. Tieto kroky nám môžu určiť a upresniť finálnu podobu návrhu. Na tejto úrovni nemáme explicitne zadané čo má model obsahovať, ale iba to, čo od neho očakávame.

1.6.2 Konceptuálny dátový model

Konceptuálny model je orientovaný na prezentáciu databázy užívateľovi, ktorému sprostredkováva vedomosti o modelovaných informáciách. Zameriava sa na pochopenie biznisu daného procesu a čo je jeho problematické úzke miesto. Konceptuálny model je hlavnou témou práce, preto sa mu budeme venovať v nasledujúcej kapitole viac dopodrobna.

1.6.3 Logický model

Keď je konceptuálny model dokončený, začína sa fáza logického modelovania. Logický model nabaľuje detailnejší pohľad na konceptuálny model. Jeho úlohou je pretvorenie konceptuálnej schémy do logickej schémy, ktorý bude ďalej optimalizovaný pre implementáciu vybraným systémom. (Rizzi, 2006)

Logický model špecifikuje, ako má byť model implementovaný nezávisle na DBMS, čim vymedzí technickú mapu pravidiel a dátové štruktúry. Vytvára základ pre fyzický model bez toho aby poskytoval detailnejšie informácie, stále zostáva všeobecný. Definuje štruktúru dát a vzťahy medzi nimi. V logickom modely nedefinujeme primárne a sekundárne kľúče a rozvádzame prepojenie entít, ktoré bolo definované na predošlej úrovni. Atribúty obsahujú dátové typy s ich presnou dĺžkou. V tomto kroku sú aj aplikované aj normované formy.

V multidimenzionálnom modelovaný môže byť vybratý relačný alebo multidimenzionálny databázový systém. V relačných implementáciách sú využívané schémy star, constellationa snowflake. Tieto schémy sú formalizované pre spravovanie dátových kociek. V multidimenzionálnych implementáciách doposiaľ nie je schválený globálne uznávaný formalizmus. Je však možné využívať štruktúry ako condensed cubes, dwarfs a QC-Trees. (Rizzi, 2006)

1.6.4 Modelovanie dátového skladu v logickom modely

Štruktúra dátových skladov je bežne v logickom modely reprezentovaná schémach popisujúcich usporiadanie faktorových tabuliek, dimenzií a hierarchií. Bežne sú používané star, snowflake a constellation schémy. V tejto časti popíšeme schémy dátových skladov využívané v multidimenzionálnom modely.

1.6.4.1 Star schéma

Star schéma je najčastejšie paradigma, podľa, ktorého sa ‚multidimenzionálny model modeluje. Obsahuje jednu faktorovú tabuľku, ktorá neobsahuje redundantné dáta a popisuje možné dimenzie v modely a jej atribúty. Pre každú dimenziu existuje jedna samostatná nenormalizovaná tabuľka, ktorá obsahuje jej primárny kľúč a množinu atribútov, ktorú dimenziu reprezentujú. V star schéme vzniká vysoká redundancia dát. Je jednoduchá na vizualizáciu ale integrita dát nie je kontrolovateľná. Star schéma sa využíva v prípade, keď všetky atribúty faktorovej tabuľky majú význam na prieniku zhodných dimenzií. (Hudec, 2022)

Príklad: Faktorová tabuľka Predaj, dimenzie čas, produkt, značku a lokalitu

1.6.4.2 Snowflake schéma

Snowflake schéma je variantom star schémy. Tabuľky dimenzií môžu byť normalizované a tým rozdelené do ďalších tabuliek. Rozdielom medzi star a snowflake schémou je redukovaná redundancia dát. Schéma obsahuje jednu faktorovú tabuľku a normalizované dimenzionálne tabuľky. Snowflake schéma je využívaná v prípade, keď je faktorová tabuľka k najnižšej úrovní dimenzií a atribúty faktorovej tabuľky majú význam na prieniku zhodných dimenzií. (Hudec, 2022)

Príklad: Faktorová tabuľka Predaj, dimenzia čas produkt a lokalita. Dimenzia lokalita bude obsahovať primárny kľúč mesto_id, na ktorý bude naviazaná ďalšia dimenzia mesto, čím sa zredukuje redundancia

Výhodou snowflake schémy sú menšie náklady na ukladanie dát a ľahšia údržba. Naopak nevýhodou je potreba joinov v dotaze a tým sa zníži efektívnosť prehľadávania dát.

1.6.4.3 Constellation schéma

Constellation schéma je komplexnejšia schéma, ktorá obsahuje viac faktorových tabuliek. Vzniká tak kolekcia prepojených star schém. Faktorové tabuľky majú na seba naviazané normalizované dimenzionálne tabuľky. Constellation schéma sa využíva v prípade, keď atribúty rozdelené do viacerých faktorových tabuliek majú význam na prieniku odlišných dimenzií alebo v rámci jednej dimenzie na odlišných hierarchiách. (Hudec, 2022)

Príklad: Faktorová tabuľka 1 Predaj, dimenzia čas produkt a lokalita. Faktorová tabuľka 2 Doprava, dimenzie, produkt, čas, lokalita, dopravca. Dimenzie produkt, čas a lokalita sú zdieľané dimenzionálne tabuľky oboma faktorovými tabuľkami.

1.6.5 Fyzický model

Fyzický model reprezentuje dáta fyzicky zaznamenané v databáze. Silne závislý na implementačnom prostredí. Definuje ako bude model implementovaný vzhľadom na konkrétny DBMS. Účel je konkrétna implementácia databázy. Dáta z logického modelu sú mapované do fyzickej štruktúry databázy. V tomto kroku prichádza na rad aj ELT alebo ELT procesy, ktoré určujú proces tvorby dátového skladu.

ETL sa skladá z troch základných fáz. Prvá fáza zahŕňa extrahovanie a získanie dát z rôznych zdrojov. Musí sa popasovať s heterogénnymi dátami, ktoré nie sú konzistentné a nemajú kompatibilné dátové štruktúry. Druhá fáza sa zaoberá transformáciou údajov z heterogénnych zdrojov, aby dáta v dátovom sklade boli konzistentné. V tretej fáze pri efektívnom prístupe k dátam je využitá optimalizácia komplexných dotazov a pokročilé techniky indexovania. Tie sú využité ako rozhranie pre OLAP. (Golfarelli, 1998) V ELT procese je druhá a tretia fáza zamenená.

1.6.6 Sémantická medzera

Dôvodom, prečo sme sa venovali všetkým fázam modelovania je práve problém vzniku sémantickej medzery. Prevod konceptuálneho modelu do relačných alebo multidimenzionálnych logických modelov nie je jednoznačný a vzniká tu sémantická medzera. Žiadne komerčné riešenia si napríklad nedokážu poradiť so vzťahmi zovšeobecňovania v hierarchiách OLAP. Problémov stále je aj reprezentácia obmedzení dimenzie alebo dokonca menej výrazné závislosti kontextu, ktoré vysvetľujú existenciu nulových hodnôt v dimenziách v logických implementáciách. (Rizzi, 2006)

Je potrebné aby sme sa zamerali na uzavretie sémantickej medzery medzi konceptuálnym modelom a logickým modelom, aby sme zachovali všetky potrebné informácie pri prevode modelov. (Rizzi, 2006) Práve aj problémový a nejasný prevod informácií medzi týmito dvoma modelmi, môže byť príčinou priameho modelovania logického modelu a vynechaním fázy konceptuálneho modelovania. Tu vzniká priama potreba jasne definovaného formalizmu pre konceptuálny model dátového skladu.

Kým táto problematika nebude vyriešená, budú dizajnéri stále siahať priamo modelovanie logického modelu. Logický model využíva jasne definované schémy pre multidimenzionálne modelovanie popísané vyššie (star, snowflake, constellation). Ich prehľadnosť a absencia jasne definovaného formalizmu pre konceptuálne modelovanie nabáda dizajnérov siahnuť priamo po logickom modely a obchádzať tak konceptuálny model. Modelovanie priamo logického modelu je takto jednoducho ľahšie, ako sa neskôr snažiť popasovať so sémantickou medzerou pri prevode modelov. V ďalšej kapitole o konceptuálnom modelovaní sa bližšie pozrieme na problematiku konceptuálneho modelovania pre dátové sklady a dôvody prečo dosiaľ nebol formovaný jasne definovaný návod pre konceptuálne modelovanie multidimenzionálnych dátových skladov.

1.7 Konceptuálne modelovanie dátového skladu

Ako sme už vyššie spomenuli, konceptuálny model sa zameriava na komunikovanie užívateľských požiadaviek. Užívateľovi tak prezentuje databázu a sprostredkováva vedomosti o modelovaných informáciách. V tomto bode modelovania nie sú potrebné technické detaily o implementácii a reprezentácia dát v dátovom sklade. (Vaisman, 2014) V tejto kapitole sa zameriame na konceptuálny model pre multidimenzionálny dátový sklad.

Ako už vieme, kroky, ktoré sú bežne súčasťou návrhu databázy, napomáhajú k dosiahnutiu správneho vytvorenia modelu dátového skladu. Tieto kroky procesu boli zhrnuté v predchádzajúcej kapitole. Konkrétne išlo o špecifikáciu požiadaviek užívateľa, konceptuálny model, logický model a fyzický model. Tento prístup sa využíva aj pri modelovaní relačných databáz. Môžeme povedať, že celý proces modelovania relačných databáz zastáva svoju úlohu aj pri návrhu dátových skladov. To zahŕňa modelovanie aj konceptuálneho modelu pre dátový sklad.

Univerzálna forma konceptuálneho modelu pre dátový sklad nie je štandardne formalizovaná, no jeho dôležitosť pri tvorbe dátového skladu je signifikantná, práve vďaka jeho zrozumiteľnosti pri definovaní užívateľských požiadaviek. Ďalším dôvod, prečo je konceptuálny model tak dôležitý, je aj závislosť logického modelu od implementačnej platformy. V prípade ak sa implementačná platforma zmení, je potrebná aj zmena logického modelu. Konceptuálny model zostáva nezmenený pretože nie je závislí na implementácii dátového skladu. (Vaisman, 2014) Každá komplexná zmena znamená vyššie náklady a zanesenie potencionálnych chýb do už schváleného modelu.

Konceptuálny model zdôrazňuje celkový obraz a nezameriava sa na detaily. Môžeme preto povedať, že konceptuálny model definuje veľmi abstraktný pohľad na celkovú štruktúru skladu. Konceptuálny model definuje „čo“ model obsahuje. Model bude neskôr v ďalších fázach modelovania definovaný viac do detailu. Konceptuálny model sa zameriava na hlavné entity a možné vzťahy medzi nimi, poprípade atribúty entít. Vzťahy nie sú explicitne definované ich kardinalitou alebo typom. Potrebujeme len vedieť, že ktoré entity sú prepojené. Model sa zameriava na identifikáciu potrebných údajov a nie na to ako sú spracovávané alebo ich fyzickú podobu, štruktúry ukladania dát.

Konceptuálne modelovanie poskytuje vysokú úroveň abstrakcie. Popisuje procesy dátového skladu a jeho architektúru vo všetkých jeho aspektoch. Model je zameraný na dosiahnutie nezávislosti v otázkach implementácie. (Rizzi, 2006) Konceptuálne modelovanie dát je jedným z najefektívnejších modelovacích techník, ktoré nám pomáhajú chápať a ujasniť si informácie. (Tupper, 2011) Pomáha nám pochopiť požiadavky užívateľa v štruktúrovanom tvare. Môže sa zdať, že na konceptuálne modelovanie DW je kladený značná pozornosť. Napriek tomu sa predpokladá, že tie najdôležitejšie otázky stále zostávajú otvorené. (Rizzi, 2006)

Ako sme už spomenuli, nemáme žiadny formalizmus, ktorý by popisoval ako by konceptuálny model dátového skladu mal byť správne vytvorený. Formalizmus nebol dosial štandardizovaný. Spoločnosti si vytvárali vlastné riešenia, ktoré spĺňali požiadavky konkrétnej spoločnosti. (Rizzi, 2006) Žiaľ tieto riešenia nie sú uplatniteľné globálne v kontexte iných domén.

Chýbajúci formalizmus konceptuálneho modelu spôsobuje začiatok dizajnu dátového skladu logickým modelom. Logický model však nemusí byť užívateľovi bez predchádzajúcich znalosti úplne jasný. Aby bol logický model vhodný na použitie pre dátové sklady, zakladá sa na schéme star alebo snowflake, čím rozširuje logický entitno-relačný model. Toto stále nie je vhodné riešenie, keďže neposkytuje plnú podporu konceptuálneho modelu pre dátový sklad. (Vaisman, 2014)

Entitno-relačné modely majú dobre definovaný formalizmus pre konceptuálne modelovanie. Sú dobre zdokumentované a široko využívané v praxi relačnými informačnými systémami. V minulosti bolo vyvíjané veľké úsilie, aby entitno-relačné schémy boli využívané aj v prípade modelovania štruktúr, ktoré nie sú založené na relačných databázach. (Golfarelli, 1998) Entitno relačné modely nemôžu používatelia pochopiť a ani im nevieme správne porozumieť softvérom DBMS. Nemožno ich použiť ako základ pre podnikové dátové sklady. (Kimball, 1996)

Rizzi (2006), uvádza dôvody prečo nebol doposiaľ schválený formalizmus pre tvorbu konceptuálneho modelu DW:

1. Vedecké komunity a prax sa nevedia dohodnúť na tých najdôležitejších multidimenzionálnych vlastnostiach, ktoré by bolo potrebné modelovať v konceptuálnom modely.

2. Konceptuálny model je sémanticky bohatý, čo spôsobuje problémy v prenesení niektorých modelovaných vlastností konceptuálneho modelu do logického modelu. To má za následok nekompletnosť logického modelu.
3. Komerčné nástroje umožňujú užívateľom grafické zakreslenie priamo až iba logického schéma. Konceptuálny model pre dátový sklad je vynechaný pri modelovaní v komerčných nástrojoch. Rozšírenie komerčných nástrojov o možnosť intuitívne modelovať konceptuálne modely skladu by prinieslo veľké výhody pre spoločnosti v praxi ale ja vedecké komunity. Tento nástroj musí splňovať množstvo požiadaviek a musí podporovať integrované modelovanie DW architektúry, zdroje, ETL, fakty, mapovanie a podobne.

Golfarelli (1998) sa takisto zaujímal o problematiku, prečo konceptuálnemu modelu nie je kladený dôraz. Ako svoje dôvody uvádza:

1. Dátové sklady boli pôvodne navrhnuté na použitie v industriálnom svete, kde užívatelia nekladú dôraz na zostavenie konceptuálneho modelu.
2. Hlavným dôvodom pre využitie dátových skladov je optimalizácia výkonnosti systému. Logický a fyzický model sa zameriavajú práve na systém a platformu, v ktorej sú modelované a teda je im kladený väčší dôraz ako na modelovanie konceptuálneho modelu.

Existujú rôzne prístupy k modelovaniu dátových skladov. Golfarelli (1998), uvádza, že konceptuálny a logický model sú častokrát zamenené. Autori pri modelovaní dátového skladu navrhujú model postaviť na základe biznis modelu podniku. Tým však dostaneme relačnú databázovú schému, čo reprezentuje logický model. Konceptuálny model pre multidimenzionálne dátové sklady musí riešiť otázky konceptuálneho modelovania, ktoré dosiaľ nie sú adresované. Týmito otázkami je napríklad štruktúra hierarchií atribútov alebo obmedzenia neaditivity. (Golfarelli, 1998)

V situáciách v reálnom svete je sa vyskytuje množstvo hierarchii. Logický model pre dátový sklad a systémy OLAP umožňujú modelovať len limitovaný počet hierarchií. Zložité hierarchie však vieme modelovať pomocou konceptuálneho modelu. Pri modelovaní dostupnými nástrojmi sú užívatelia častokrát obmedzovaný predefinovanými druhmi hierarchií. Základná sémantika multidimenzionality tak nedokáže byť zachytená. (Vaisman 2014)

Vaisman (2014), nám ponúka ďalší problém absencie konceptuálneho modelu a jeho potenciónálnu výhodu v prípade ak by bol predstavený jednotný formalizmus modelovanie konceptuálneho modelu pre multidimenzionálne dátové sklady.

Na konceptuálne modelovanie dátových skladov môžeme popísať z troch pohľadov. Prvým je prístup k modelovaniu z dostupných údajov a schém z prevádzkových databáz, ktorý nazývame **data driven**. Druhým, je naopak **user driven**, kde figurujú požiadavky užívateľov a model sa formuluje na základe požiadaviek. Treťou možnosťou je hybridný prístup, kde sa podľa dostupných dát zo schém vytvorí model, ktorý sa neskôr modifikuje na základe pripomienok užívateľa. (Hudec, 2022). V našej práci a pri vytváraní konceptuálneho modelu sa zameriame na user driven prístup.

Dôvodom, prečo sa v práci zameriavame na konceptuálny model práve pre multidimenzionálny model je, že multidimenzionálne modely majú všetky predpoklady na reprezentovanie všetkých požiadaviek potrebných pre dátový sklad a OLAP aplikácii na konceptuálnej úrovni. Tieto požiadavky sú prvky multidimenzionálnych modelov, čí sú dimenzie, hierarchie, fakty a miery, tak ako bolo vyššie spomenuté.

1.8 MultiDim pre konceptuálny model

Vývoj dizajnu dátových skladov nie je zdokumentovaný v mnohých publikáciách. Dokumentujú ho primárne ľudia z praxe na reálnych modelových situáciách. Tento spôsob sa môže zdať ako vhodný a dokonca lepší ako učebnicové príklady. Vedecká komunita prišla s viacerými možnými prístupmi k modelovaniu konceptuálneho modelu, no tieto prístupy sú v praxi prakticky nepoužiteľné, kvôli svojej komplexnosti. Chýbajú zdroje, ktoré nás vedia previesť všetkými fázami vývoja (Vaisman, 2014)

Súčasné modely dátového skladu zahŕňajú časovú dimenziu, ktorá sa používa rovnako ako ostatné typy dimenzií, na účely zoskupovania (použitie operácie roll-up). Časová dimenzia tiež udáva časový rámec mier. Napríklad v marci 2022 sa predalo 100 jednotiek produktu. Nemôžeme ho však použiť na sledovanie zmeny v iných rozmeroch, napríklad, keď výrobok zmení svoje ingrediencie. Preto je obvyklé, že viacrozmerné modely nie sú symetrické v spôsobe reprezentácie zmien pre miery a rozmery. V dôsledku toho, vlastnosti „časového variantu“ a „nevolatility“ sa vzťahujú len na miery, ktoré znázornenie zmien, ku ktorým dochádza v dimenziách ponechávajú na aplikácie pracujúcimi s OLAP modelmi. (Malinowski, 2008)

Pretože v mnohých prípadoch sú zmeny v dimenziách a čas, kedy k nim došlo dôležité pre účely analýzy, boli navrhnuté viaceré implementačné riešenia tohto problému v kontexte relačných databáz. Boli vytvorené takzvané pomaly sa meniace dimenzie. Tieto riešenia však nie sú uspokojivé, pretože buď nezachovávajú celú históriu údajov, alebo sa ťažko implementujú. (Malinowski, 2008)

Pomaly meniacim dimenziám sa v práci nebudeme hlbšie venovať, keďže nie sú predmetom práce ale spomenuli sme ich ako dôvod pre vznik modelu Multidim. Model MultiDim poskytuje podporu pre úrovne, atribúty, hierarchie a miery. Pre hierarchie diskutujeme o rôznych prípadoch v závislosti od toho, či sa menia v úrovniach alebo vo vzťahoch medzi nimi. Pre miery uvedieme rôzne scenáre, ktoré ukazujú užitočnosť rôznych typov dočasnosti. Ďalej, keďže miery možno pred vložením do dátových skladov agregovať, budeme diskutovať o problémoch týkajúcich sa rôznych časových podrobností medzi zdrojovými systémami a dátovými skladmi. (Malinowski, 2008)

Prvky modelovania MultiDim pre konceptuálny model rozoberieme v ďalšej kapitole. Vysvetlené budú na príklade predaja produktov, ktorý sa v literatúre spomína najčastejšie. Dôvodom je jeho jednoduchosť a jednoduchá predstava o jeho procesoch aj pre užívateľom, ktorý sú v danej doméne laici.

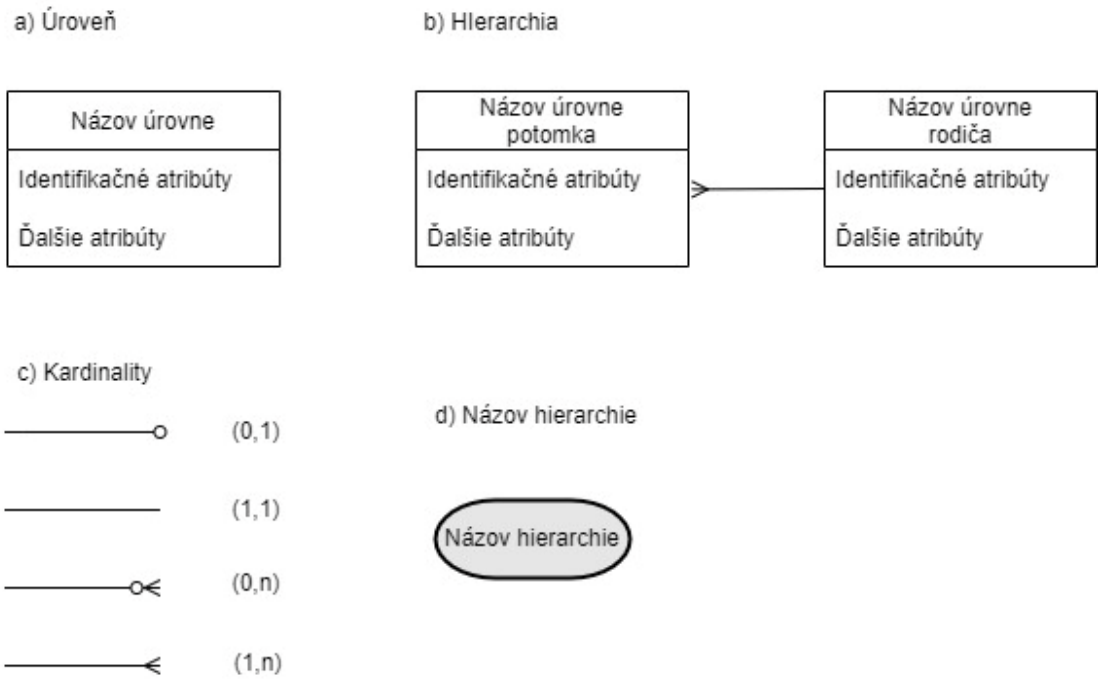
1.9 Modelovanie prvkov MultiDim pre konceptuálny model

Model MultiDim využíva grafický zápis podobný entitno-relačným modelom. Bol vytvorený tak, aby odpovedal požiadavkám pri analýze časových údajov, ktoré sa v dimenziách môžu aj meniť. V nasledujúcich častiach si rozoberieme grafický zápis jednotlivých prvkov a hierarchií.

1.9.1 Grafická notácia prvkov MultiDim

V nasledujúcej časti môžeme vidieť zápis prvkov konceptuálneho modelu pre Multidim. Schéma sa skladá z množiny úrovní organizovaných do dimenzionálnych tabuliek a faktorovej tabuľky. Úroveň mapuje entitu v entitno-relačnom modely, ktoré zodpovedajú konceptu reálneho sveta z pohľadu aplikácie. (Malinowski, 2008)

V obrázku 5. môžeme vidieť označenie úrovne, (a). V úrovni definujeme identifikátor a postupne aj ďalšie atribúty. Hierarchia (b) prepája hierarchie medzi sebou a určuje ich poradie v rámci dimenzie. Dve úrovne sú prepojené väzbou, ktorá hovorí o ich vzťahu medzi sebou a ich kardinalite (c) . Ich vzťah je daný ich podstatou a vyplýva z daných biznisových pravidiel v rámci domény. Hierarchia je v schéme jednoznačne rozlíšená jej názvom. Jej grafické vyjadrenie znázorňuje ovál pridružený k hierarchii (d). Hierarchia taktiež predstavuje kritériu, podľa ktorého budú dáta analyzované.

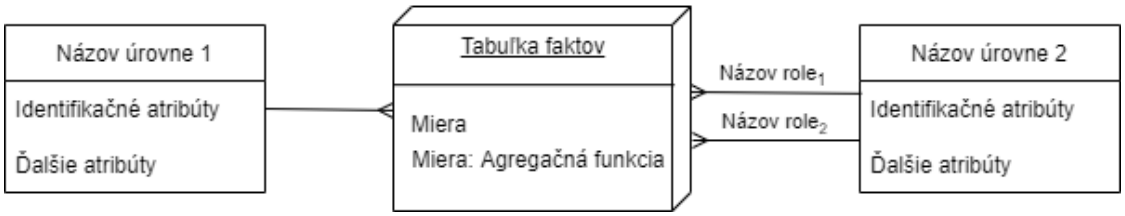


Obr. 5 - Zápis prvkov úroveň a hierarchia (Vaisman, 2014)

Obrázok 6. zobrazuje faktorovú tabuľku s mierami a pridruženými úrovňami. Ako sme už vyššie spomenuli, faktorová tabuľka reprezentuje predmet záujmu analýzy dátového skladu. Faktorová tabuľka obsahuje miery a ich pridružené agregačné funkcie.

Jedna úroveň sa môže vo vzťahu zúčastniť viac krát s odlišným významom. To znamená, že dve väzby medzi faktorovou tabuľkou a druhou úrovňou hovoria o rôznej sémantike vzťahu medzi týmito dvoma prvkami. V ďalšej časti práce sa budeme venovať prípadovej štúdii zaoberajúcej sa e-commerce predajom produktov, kde práve takáto dvojitá väzba vystane. Vznikne potreba analyzovať informácie o dátume vzniku objednávky, dátume, kedy bola objednávka odoslaná a dátume splatnosti objednávky. Každá rola je reprezentovaná samostatnou väzbou, kardinalitou a pomenovaním.

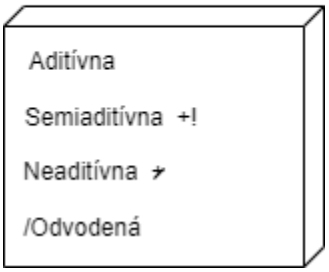
e) Faktorová tabuľka s mierami a pridruženými úrovňami



Obr. 6 - Zápis prvku Faktorová tabuľka (Vaisman, 2014)

V kapitole o faktorovej tabuľke sme popísali význam možných typov mier, ktoré sa vyskytujú vo faktorových tabuľkách. V tejto časti sa zaoberáme grafickým zobrazením jednotlivých prvkov v konceptuálnom modeli. V obrázku 7. môžeme vidieť označenie aditívnej, semiaditívnej, neaditívnej a odvodenej miery. Aditívne miery sú základným typom a sú sumarizovateľné naprieč všetkými dimenziami. Semiaditívne a neaditívne miery sú obe rozlíšené znakom za svojím názvom, ako je zobrazené v obrázku 7. Tieto miery možno sčítať iba naprieč niektorými dimenziami. Odvodené miery sú vypočítané alebo odvodené na základe iných mier alebo atribútov a sú odlíšené znakom „/“ pred svojím názvom.

f) Typy mier

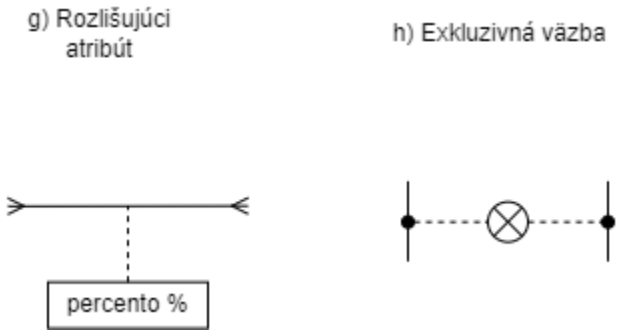


Obr. 7 - Zápis prvku - Typy mier (Vaisman, 2014)

Exkluzívna väzba (h) popisuje situáciu, kedy existujú dve alebo viac väzieb medzi nadriadenou a podriadenou dimenziou exkluzívne. (Conceptual Data Warehousing, 2022) Môžeme vybrať iba jednu väzbu z možných väzieb v jednej inštancii. V jednej inštancii môže exis-

tovať len jedna cesta. Exkluzívna väzba vyjadruje situáciu, keď z možných ciest medzi úrovňami môže byť vybraná iba jedna.

Rozlišujúci atribút sa využíva v prípade väzieb m:n. V prípade vzťahu m:n potrebujeme určiť percentuálny podiel patričnosti do viac záznamov na vyššej úrovní. (Hudec, 2022). Rozlišujúci atribút (g) určuje ako sú miery prerozdelené medzi nadriadenými členmi vo väzbe m:n. Používa sa na zoskupovanie rôznych členov počas operácie súhrnu z úrovní produktu po úrovne kategórie. (Vaisman, 2014)



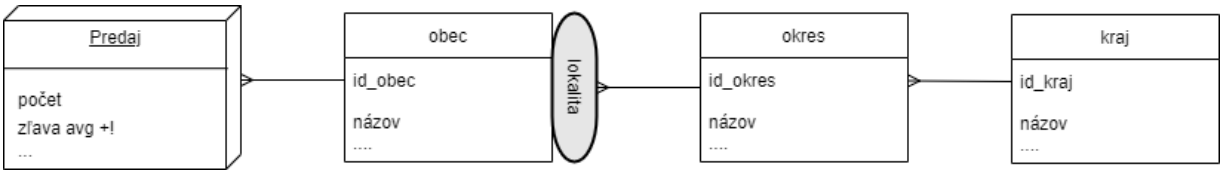
Obr. 8 - Zápis prvku rozlišujúci atribút a exkluzívna väzba (Vaisman, 2014)

1.9.2 Zakreslenie hierarchií MultiDim

V nasledujúcej časti sa pozrieme na možnosti zakreslenia jednotlivých hierarchií. Zameriame sa taktiež na dôvody zakreslenia jednotlivých hierarchií a na rozdiely medzi nimi.

1.9.2.1 Vyvážené hierarchie (Balanced hierarchies)

Vyvážená hierarchia popisuje typ hierarchie, kedy na úrovni schémy, existuje iba jedna cesta a každá z úrovní v hierarchii je povinná. (Vaisman, 2014) Nadriadený člen každého člena je na úrovni bezprostredne nad členom. Najbežnejším príkladom vyváženej hierarchie je dimenzia času, kedy hĺbka každej úrovne je konzistentná. Hierarchiu so stromovou štruktúrou, kde každý podriadený člen má iba jedného rodiča. Ako môžeme vidieť v príklade úroveň obec má jednu hierarchiu „územie“.



Obr. 9 - Zápis prvku - Vyvážená hierarchia (Hudec, 2022)

1.9.2.2 Nevyvážené hierarchie (Unbalanced hierarchies)

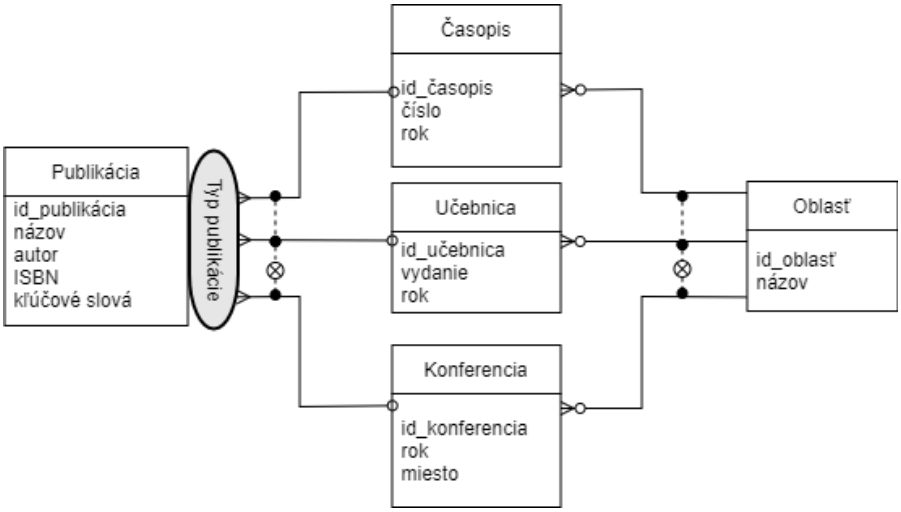
Nevyvážená hierarchia je podobná vyváženej hierarchii. Na úrovni schémy existuje takisto iba jedna cesta ale rozdielom je, že nie všetky úrovne sú povinné. (Vaisman, 2014) Nevyvážená hierarchia popisuje úrovne, kedy vzťah medzi nadriadeným členom a potomkom je konzistentný ale majú logicky nekonzistentné úrovne. Vetvy hierarchie môžu mať tiež nekonzistentnú hĺbku.

Príkladom nevyváženej hierarchie je organizačná schéma, ktorá zobrazuje vzťahy medzi zamestnancami organizácie. Úrovne v rámci organizačnej štruktúry sú nevyvážené, pričom niektoré vetvy v hierarchii majú viac úrovní ako iné.

Môžeme si všimnúť, že v predchádzajúcej kapitole vyvážená a nevyvážená hierarchia neboli definované. Na konceptuálnej úrovni delíme jednoduchú hierarchiu na vyváženú a nevyváženú.

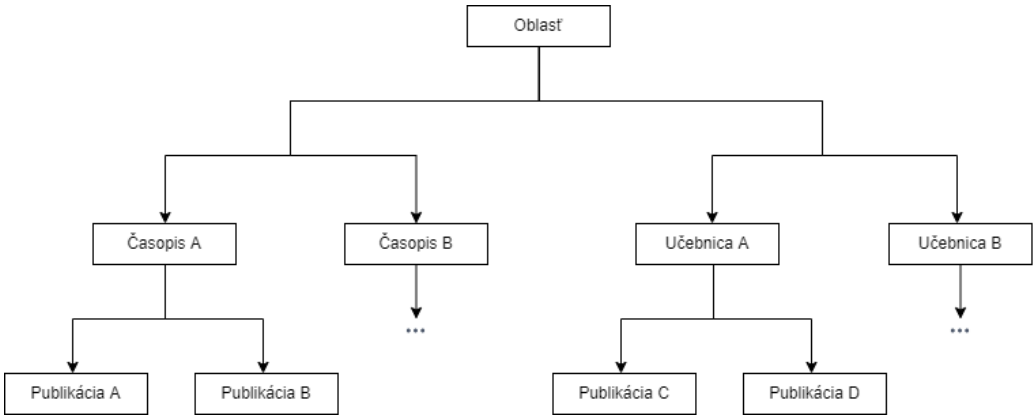
1.9.2.3 Generalizované hierarchie (Generalized hierarchies)

V generalizovanej hierarchii existuje viacero exkluzívnych ciest, ktoré zdieľajú aspoň úroveň listu. Môžu však zdieľať aj ďalšie úrovne. V obrázku 10. vidíme dve agregačné cesty, jedna pre každý typ zákazníka. Obe cesty patria do jednej hierarchie. Na označenie používame symbol „⊗“. že cesty sú exkluzívne pre každého člena. (Vaisman, 2014) Toto platí na úrovni schémy.



Obr. 10 - Zápis prvku - Generalizovaná hierarchia (Hudec, 2022)

Na úrovni inštancií pre každý člen je možná iba jedna cesta ako môžeme vidieť v obrázku 11.



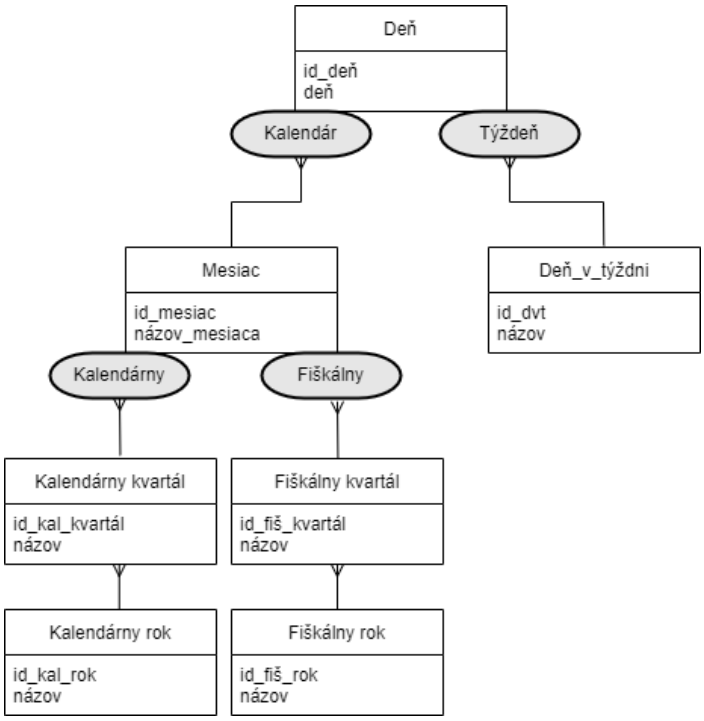
Obr. 11 - Príklad inštancie generalizovanej hierarchie

1.9.2.4 Alternatívne hierarchie (Alternative hierarchies)

Alternatívna hierarchia predstavujú situáciu, keď na úrovni schémy existuje niekoľko neexkluzívnych hierarchií, ktoré zdieľajú aspoň úroveň listu. Dimenzia čas zahŕňa dve ďalšie hierarchie – kalendárny rok a fiškálny rok. Alternatívne hierarchie využívame vtedy, keď po-

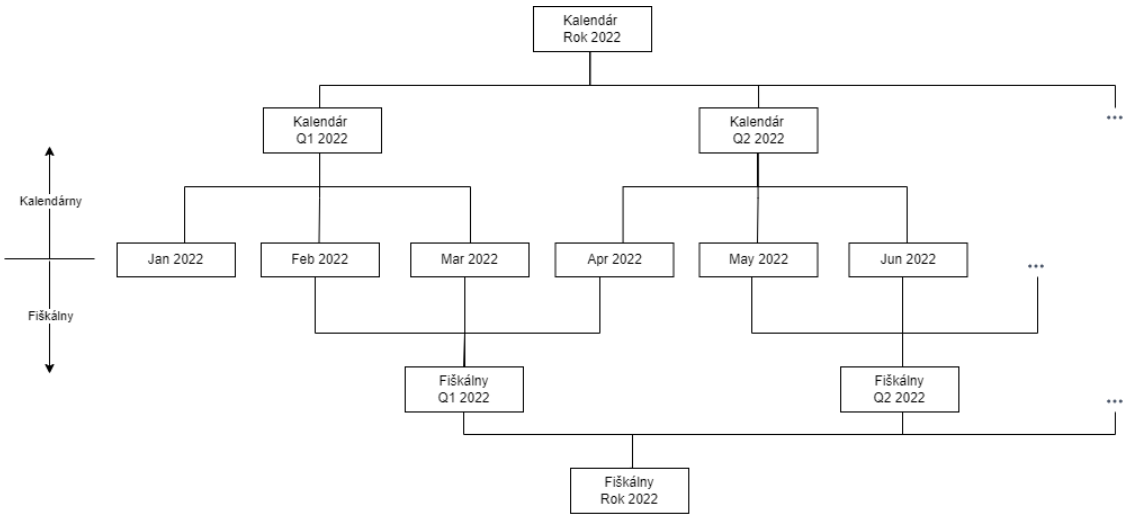
trebujeme miery analyzovať unikátneho pohľadu, akým môže byť aj napríklad čas alternatívnou agregáciou. (Vaisman, 2014)

Rozdiel medzi generalizovanou a alternatívnou hierarchiou je v tom, aké cesty vedú k úrovni potomka. V generalizovanej hierarchii to je iba jedna z ciest a naopak v alternatívnej hierarchii potomok súvisí so všetkými cestami a používateľ si musí vybrať na analýzu iba jednu z možností. (Vaisman, 2014)



Obr. 12 - Zápis prvku - Alternatívna hierarchia(Hudec, 2022)

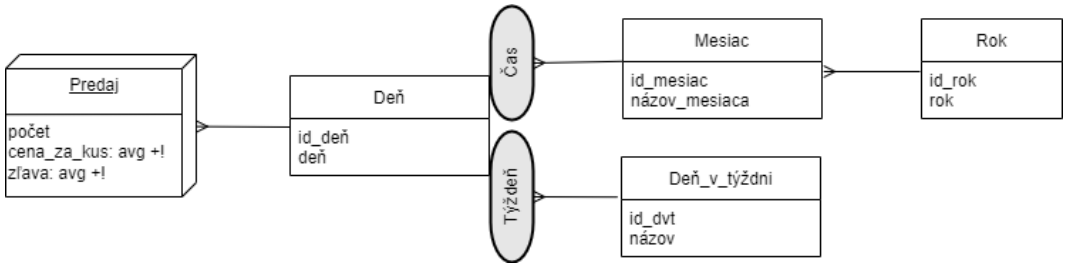
Na úrovní inštancií vidíme, že hierarchie sú prepojené. Potomok je spojený s viacerými nadriadenými členmi, ktorý patria do rôznych úrovní. (Vaisman, 2014)



Obr. 13 - Príklad inštancie alternatívnej hierarchie

1.9.2.5 Paralelné hierarchie (Parallel hierarchies)

Paralelná hierarchia je daná situáciou, keď jedna dimenzia je zložená z viacerých hierarchií, ktoré budú korešpondovať s rôznymi kritériami pre analýzu.



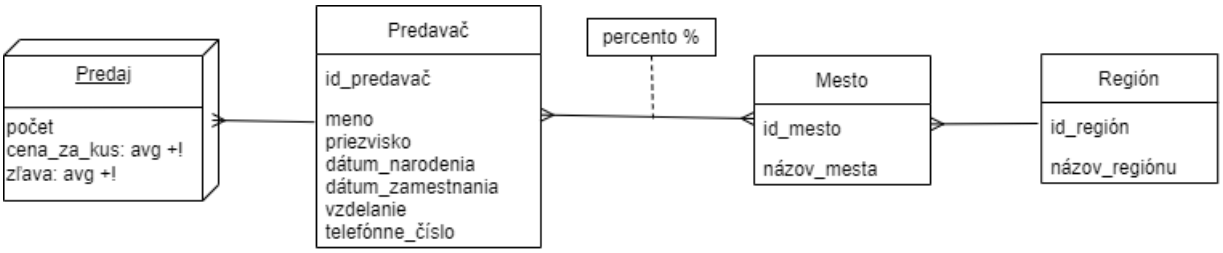
Obr. 14 - Zápis prvku - Paralelná hierarchia (Hudec, 2022)

Paralelné a alternatívne hierarchie sa môžu zdať veľmi podobné, keďže zdieľajú úrovne a môžu byť zostavené niekoľkých hierarchií. Na úrovni konceptuálneho modelu ich aj napriek tomu potrebujeme rozlišovať, pretože ich sémantika je rozličná. V prípade alternatívnych hierarchií nemá zmysel kombinovať úrovne z rôznych hierarchií, pričom pri paralelných hierarchiách to možno urobiť. Ďalším rozdielom týchto hierarchií je v alternatívnych hierar-

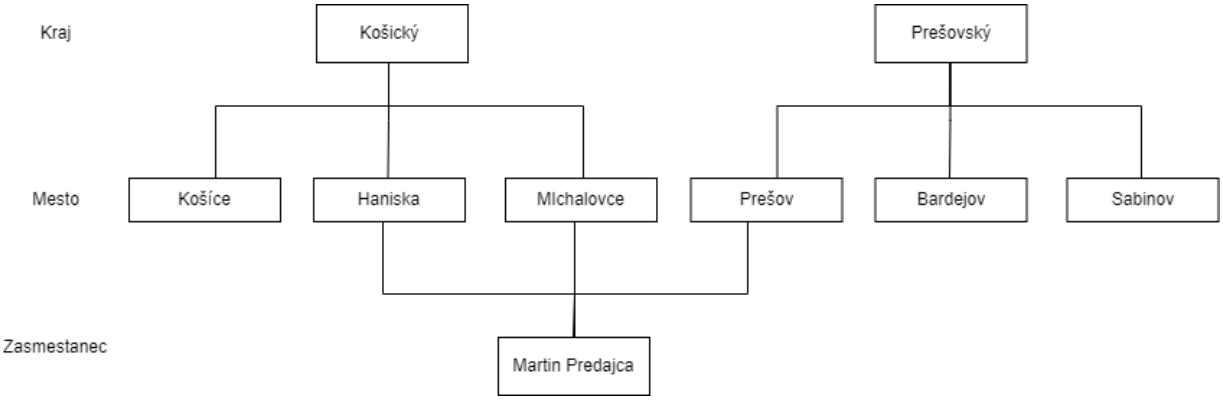
chiách listový člen nemôže byť spojený s ďalšími členmi na rovnakej úrovni. V prípade paralelných hierarchií to možné je. (Vaisman, 2014)

1.9.2.6 Nestriktné hierarchie (Non-strict hierarchies)

V prípade nestriktnej hierarchie vidíme situáciu, kedy vznikajú m:n väzby. Ako príklad nestriktnej hierarchie môžeme využiť prípad zamestnancov, ktorý môžu byť priradení vo viacerých oblastiach a prirodzene v oblasti pracuje množstvo zamestnancov.



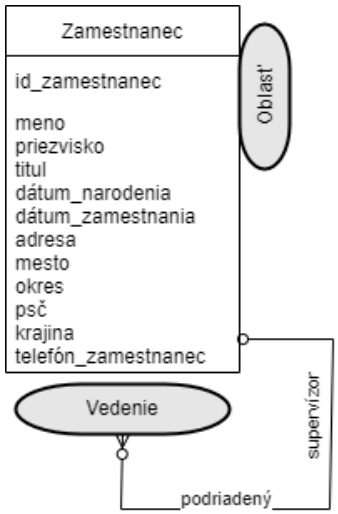
Obr. 15 - Zápis prvku - Nestriktná hierarchia(Hudec, 2022)



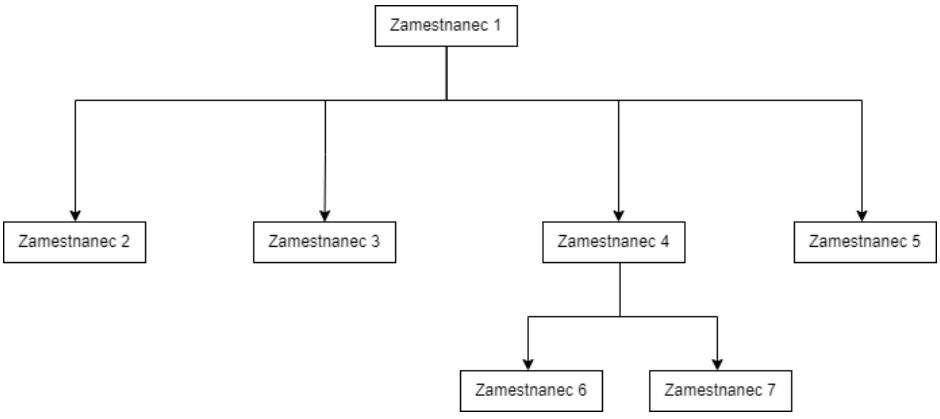
Obr. 16 - Príklad inštancie nestriktnej hierarchie

1.9.2.7 Rekurzívne hierarchie (Recursive hierarchies)

Rekurzívne hierarchie sú špeciálnym typom nevyváženej hierarchie. Tomuto typu sa budeme venovať z dôvodu, že sa bude neskôr vyskytovať v našej prípadovej štúdii. V rekurzívnych hierarchiách tá istá úroveň spadá pod dve role vzťahu. Úroveň je zároveň podriadeným a nadriadeným členom. Tieto role zároveň vyjadrujú rovnakú sémantiku. (Conceptual Data Warehousing, 2022) Dobrým príkladom je podriadenosť a nadriadenosť zamestnancov, ktorú využijeme aj neskôr v prípadovej štúdii.



Obr. 17 - Zápis prvku - Rekurzívna hierarchia (Vaisman, 2014)



Obr. 18 - Príklad inštancie rekurzívnej hierarchie (Vaisman, 2014)

2 Ciele, metódy a metodika

2.1 Cieľ práce

Cieľom práce je vytvoriť konceptuálny model multidimenzionálného dátového skladu, zachyteným rôznych typov hierarchií dimenzií, ktorý bude slúžiť ako manuál. V práci definujeme multidimenzionálny model a jeho prvky. Zameriame sa na modelovanie dátových skladov so zreteľom na konceptuálne modelovanie. Taktiež sa bližšie pozrieme na problém, prečo doposiaľ neexistuje jednotne definovaný formalizmus pre modelovanie konceptuálneho modelu dátového skladu, a prečo sa dátový sklad modeluje priamo logickým modelom.

Výsledkom práce bude aplikovanie nadobudnutých znalostí na dve konkrétne prípadové štúdie, pre ktoré budú vytvorené konceptuálne modely použitím MultiDimu.

2.2 Metódy a metodika

Prácu sme začali prieskumom dostupných zdrojov a definovaním súčasného stavu danej problematiky. Definovali sme pojem dátové sklady a zamerali sa na multidimenzionálny dátový sklad. Pozreli sme sa na rozdiel medzi OLAP a OLTP a na výhody a nevýhody multidimenzionálného modelovania. Špecifikovali sme prvky multidimenzionálnych dátových skladov – faktorovú tabuľku, miery, dimenzie, hierarchie a jednotlivé typy hierarchií.

Následne sme prešli procesom modelovania dátových skladov a popísali dôležitosť jednotlivých krokov. Hlbšie sme sa zamerali na modelovanie konceptuálneho modelu. Konceptuálne modelovanie nie je využívané pri modelovaní dátových skladov a pokúsili sme sa zamerať na dôvody, prečo tomu tak je.

MultiDim je jednou z možností modelovania konceptuálnych modelov dátových skladov. Uviedli sme výhody využitia MultiDim pre konceptuálne modelovanie. Po definovaní grafickej notácie základných prvkov sme tiež načrtli grafickú notáciu jednotlivých typov hierarchií. Existujú aj ďalšie typy hierarchií, ktoré vznikajú modifikovaním základných typov. Z tých sme uviedli iba tie, ktoré sme využili v prípadových štúdiách.

Postup modelovania konceptuálneho modelu pomocou MultiDim sme sa pokúsili vysvetliť na dvoch prípadových štúdiách, ktoré obsahovali rôzne typy hierarchií. Prvá prípadová

štúdia sa zamerala na e-commerce predaj produktov spoločnosti Northwind. Príklad bol použitý z primárneho zdroja práce (Vaisman, 2004), a mierne sme ho modifikovali. Aj ďalšie literatúry využívali rovnaký príklad spoločnosti Northwind. Všeobecne sa ako príklad modelovania dátových skladov využíva predaj produktov a skladové hospodárstvo naprieč dostupnou literatúrou.

Pred modelovaním konkrétnych prípadových štúdii sme špecifikovali jednotlivé kroky, ako vyskladať MultiDim pre konceptuálny model. Cieľom práce bolo, aby práca mohla slúžiť ako návod pre potencionálnych dizajnérov konceptuálnych modelov dátových skladov. Taktiež sa priblížiť k formalizovaniu štandardu modelovania konceptuálnych modelov pre dátové sklady.

Pri prvej prípadovej štúdii sme uviedli zadanie zákazníka. Toto zadanie nebolo v zdroji uvedené, no definovali sme ho na základe konceptuálneho modelu a informácií, ktoré sme sa zo zdroja dozvedeli o nimi využívanom príklade spoločnosti Northwind. Zadanie od zákazníka a jeho požiadavky sú dôležitou súčasťou modelovania na základe, ktorých sú jednotlivé schémy modelované, a preto sme to považovali za dôležitú súčasť prípadovej štúdie. Následne sme si zadanie rozobrali a určili, aké údaje budeme potrebovať v dátovom sklade uchovávať.

Určili sme postup, akým budeme postupovať pri modelovaní konceptuálneho modelu a jeho jednotlivé kroky, ktorými sme sa riadili. Špecifikovali sme dimenzie schémy, ich atribúty a zakreslili sme ich do modelu. Následne sme určili faktorovú tabuľku „Predaj“ a miery, na základe ktorých budú neskôr dáta v dátovom sklade analyzované. Faktorovú tabuľku sme takisto zakreslili do schémy.

Keď sme už definovali faktorovú tabuľku a dimenzionálne tabuľky mohli sme sa pustiť na definovanie väzieb medzi jednotlivými úrovňami. Vzťahy špecifikované v požiadavkách nám určili kardinalitu, ktorá upresňuje vzťah medzi faktorovou tabuľkou, dimenziami a úrovňami dimenzií. V tomto momente klasifikujeme hierarchie, určíme ich typ a správne ich zakreslíme. Zostalo nám iba pridať exkluzívne väzby a prípadne rozlišujúci atribút.

Rovnaké kroky sme aplikovali na prípadovú štúdiu 2., ktorá sa zaoberala predajov rôznych typov publikácii. Prípadová štúdia 2. nám uviedla príklad aj ďalších typov hierarchií, ktoré neboli uvedené v prípadovej štúdii 1. Ako príklad sme využili materiály z prednášok predmetu Business Inteligence, a tak ako v prípade prípadovej štúdie 1., sme ho mierne modi-

fikovali. Vytvorili sme zadanie, na základe informácií, ktoré sme zo schémy vyčítali. Definovali sme potrebné údaje, ktoré je potrebné v dátovom sklade uchovávať, určili dimenzie a faktorovú tabuľku. Následne sme definovali väzby medzi dimenziami a hierarchie, ktoré budú v schéme využité. Ako posledný krok boli aplikované exkluzívne väzby a rozlišujúce atribúty. Následne sme uviedli kompletnú schému konceptuálneho modelu pre prípadovú štúdiu 2. využitím MultiDim.

Na zakreslenie všetkých príkladov a schém prípadových štúdii sme využili online nástroj draw.io a zdroje, ktoré sú uvedené v zozname literatúry. V práci sme využili množstvo obrázkov a schém, ktoré nám pomohli pri vysvetľovaní problematiky.

3 Výsledky práce

V nasledujúcej časti práce sa zameriame na vymodelovanie konceptuálneho modelu MultiDim pre dve prípadové štúdie na základe určeného postupu. Prípadové štúdie sú zamerané na predaj produktov a predaj publikácii. V prípadových štúdia poukazujeme na rôzne typy hierarchií, Postup modelovania sme vytvorili na základe logickej postupnosti potrebných krokov a prvkov v MultiDim.

3.1 Postup pri tvorbe konceptuálneho modelu MultiDim

Pri tvorbe konceptuálneho modelu využitím MultiDim budeme postupovať nasledujúcimi krokmi:

1. Ujasníme si zadanie a štruktúru údajov, ktoré bude potrebné uchovávať v dátovom sklade na základe užívateľských požiadaviek
2. Určíme dimenzie, ktoré budú v schéme obsiahnuté z prvého kroku
3. Určíme miery faktorovej tabuľky, podľa ktorých sa budú dáta v dátovom sklade analyzovať
4. Určíme väzby medzi dimenziami a ich kardinalitu
5. Určíme figurujúce hierarchie v schéme
6. Určíme exkluzívne väzby a rozlišujúce atribúty na základe biznis požiadaviek
7. Spojením všetkých krokov nám vznikne konceptuálny model prípadovej štúdie

Tieto kroky postupne následnej rozoberieme a nakoniec ich spojením vymodelujeme konceptuálny model.

3.2 Prípadová štúdia 1. – E-commerce predaj produktov

Ako prvú prípadovú štúdiu si rozoberieme príklad, ktorý uvádza vo svojej publikácii Vaisman, 2014 a upravíme ju tak, aby vyhovovala naším požiadavkám. Prípadová štúdia 1. sa zaoberá predajom produktov. Predaj produktov je jeden z najviac využívaných príkladov pre modelovanie multidimenzionálnych dátových skladov v literatúre. Tento príklad bol uvedený v množstve ďalších zdrojoch, buď zhodný alebo čiastočne modifikovaný. Spoločnosť Northwind sa zaoberá exportom rôznych produktov. Na riadenie predaja a uchovávanie podnikových dát je potrebná implementácia riešenia, ktoré bude aj umožňovať ich následnú analýzu. To zákazníkovi umožní implementácia multidimenzionálneho skladu.

3.2.1 Zadanie od zákazníka:

Naša spoločnosť potrebuje uchovávať informácie predaji produktov a objednávkach. Produkty sú dodávané našimi dodávateľmi, ktorý sídlia v rôznych lokalitách. Sme medzinárodná spoločnosť. Produkty dodávame našim zákazníkom do celého sveta. Preto potrebujeme, aby informácie o lokalite mali celosvetovú štruktúru.

Dáta by sme potrebovali analyzovať z časového hľadiska mesiaca, kedy bol produkt predaný, polroku a roku. Produkt kategorizujeme do kategórii. Produkt môže byť zahrnutý iba v jednej kategórii. Objednávka obsahuje informácie, ktoré zákazníkovi, doručení, zamestnancovi, ktorý objednávku spracoval, cenu a doručovaciu adresu.

Zamestnanci, ktorý danú objednávku spracujú, majú nadriadených, ktorý za nich zodpovedajú. Vo firme teda nastavená štruktúra zamestnancov. Zamestnanec nemusí byť priradený iba do jednej lokality ale môže byť zodpovedný za viac lokalít naraz. Taktiež potrebujeme uchovávať informácie o dopravcoch, ktorý dopravujú objednávky na miesto doručenia uvedeného v objednávke.

Zhrňme si informácie, ktoré sme sa od zákazníka dozvedeli, ktoré potrebuje v dátovom sklade uchovávať. Informácie o schéme sme vyvodili z informácii, ktoré uvádza Vaisman, 2004. Zadanie v literatúre uvedené nebolo, no poznať požiadavky zadávateľa je dôležitou súčasťou modelovania konceptuálneho modelu. Údaje, ktoré by schéma mala obsahovať sú nasledovné:

- Údaje o zákazníkovi musia obsahovať *identifikátor, meno firmy, úplnú adresu, poštové smerovacie číslo a telefón*.
- Zamestnanec je definovaný *identifikátorom*. Ďalej si o zamestnancovi uchovávame informácie o *mene a priezvisku, titule, dátume narodenia a dátume zamestnania, adresu a telefónne číslo*.
- Geografické údaje, konkrétne územia, kde spoločnosť pôsobí. Tieto údaje sú usporiadané do regiónov. V súčasnosti je potrebné ponechať len popis územia a regiónu. Každý zamestnanec môže byť prepojený s viacerými územiami a každé územie môže byť prepojený s viacerými zamestnancami.
- Údaje o doručovateľovi, teda informácie o spoločnostiach, ktoré spoločnosť Northwind najíma na poskytovanie doručovateľských služieb. O dopravcovi si uchovávame iba *názov spoločnosti a telefónne číslo*.
- O dodávateľovi potrebujeme poznať *názov spoločnosti, meno kontaktnej osoby, adresu, telefón a webovú adresu*.
- Spoločnosť Northwind obchoduje s produktami, o ktorých potrebujeme poznať údaje, ako je *identifikátor produktu, názov, množstvo za jednotku, jednotkovú cenu a informáciu o tom, či predaj produktu bol pozastavený*.
- Produkty sú ďalej rozdelené do kategórií, o ktorých si evidujeme jej *názov* a *popis*. Každý produkt má vlastného dodávateľa.
- Informácie o uskutočnených objednávkach zahŕňajú identifikátor, informáciu o zákazníkovi, ktorý objednávku vytvoril, *dátum vytvorenia objednávky, dátum odoslania objednávky, požadované doručenie, skutočný dátum dodania objednávky, dopravca, ktorý objednávku doručil, informáciu o zamestnancovi, ktorý objednávku spracoval, náklady na dopravu a cena za objednávku, doručovacia adresa*
- Ďalej objednávka môže obsahovať minimálne jeden alebo aj viac produktov a pre každý z nich uchovávame *názov produktu, cenu za kus, jednotkovú cenu a jeho dostupnosť* (informáciu o tom, či produkt je dostupný na predaj)
- Dáta budú analyzované z časového pohľadu a to mesiaca, polroku a roku.

3.2.2 Určenie dimenzií

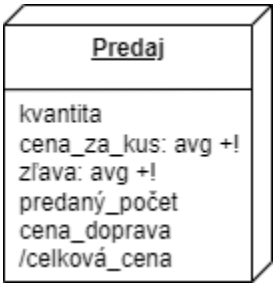
Podľa užívateľských požiadaviek si definujeme dimenzionálne tabuľky. „**Zamestnanec“**, „**Zákazník“**, „**Dodávateľ“** a „**Dopravca“** sú osoby, ktoré v našej schéme budú figurovať. O týchto osobách si potrebujeme uchovávať a analyzovať informácie, teda nám vytvoria dimenzie. „**Produkty“** potrebujeme klasifikovať do „**kategórii“**, touto požiadavkou nám vznikajú ďalšie dve dimenzie. Zákazníci budú vytvárať „**objednávky“**. Objednávka je ďalšia dimenzia. Ďalej potrebujeme uchovávať informácie o lokalite. Tu definujeme následné dimenzie – „**mesto“**, „**štát“**, „**okres“**, „**krajina“**, „**kontinent“**. Následne ešte potrebujeme dimenzie, ktoré nám budú hovoriť o čase v akom boli produkty predané. Tú nám vznikajú dimenzie „**čas“**, „**mesiac“**, „**polrok“** a „**rok“**. Dokopy nám týmto spôsobom vzniklo 16 dimenzií.

Dimenziám následne priradíme potrebné atribúty, ktoré boli popísané v užívateľských požiadavkách a dimenzie zakreslíme do schémy, ktorú môžeme vidieť v obrázku 22.

3.2.3 Definovanie faktorovej tabuľky

Vo faktorovej tabuľke definujeme miery, na základe ktorých budeme dáta analyzovať. Faktorová tabuľka súvisí s dimenziami, ktoré sme definovali vyššie.

Miera „kvantita“ splňuje pravidlo aditivity. Miera bude agregovaná cez všetky dimenzie využitím agregačnej funkcie sčítania. „Cena za kus“ je semiaditívna miera, pretože sa nesčíta sa naprieč všetkými dimenziami. Sčítanie tejto miery cez všetky úrovne nedáva zmysel. Napríklad zákazník nemá cenu. To isté platí pri miera „zľava“. Miery „cena za kus“ a „zľava“ využijú agregačnú funkciu avarage pre výpočet priemernej ceny alebo zľavy na dané časové obdobie pre daný produkt. Miery „predaný počet“ a „cena za dopravu“ sú aditívne miery. Hodnota miery celková cena sa vypočítava na základe miery „cena dopravy“, „cena za kus“, „predaný počet“ a „zľava“. Tým túto mieru definujeme ako odvodenú, keďže nemôže byť zmysluplne sčítaná v akejkoľvek dimenzii. Miery zapíšeme do faktorovej tabuľky a priradíme im príslušné symboly.



Obr. 19 - Faktorová tabuľka prípadovej štúdie 1. (Vaisman, 2014)

Podľa požiadaviek zákazníka v schéme našej prípadovej štúdie, definujeme faktorovú tabuľku „**Predaj“**, a šestnásť dimenzionálnych tabuliek.

3.2.4 Určenie väzieb medzi dimenziami

Väzby medzi dimenziami definujeme na základe užívateľských požiadaviek a toho, čo z požiadaviek vyplynulo. Faktorová tabuľka predaj je priamo spojená s dimenziami: „Produkt“, „Čas“, „Zamestnanec“, „Zákazník“, „Dodávateľ“, „Dopravca“, „Objednávka“.

Medzi faktorovou tabuľkou „Predaj“ a dimenziou „Produkt“ je kardinalita 1:n, čo znamená, že jeden predaj je spojený s produktom len jeden krát, pričom produkt môže byť predaný viac krát. Dimenzia „Čas“ sa zúčastňuje vo faktorovej tabuľke vo viacerých rolách. Vo faktorovej tabuľke je dimenzia zúčastnená s rolou ako dátum objednania, dátum splatnosti a dátum odoslania využitím kardinality 1:n.

Každý produkt môže byť obsiahnutý v jednej kategórii. Táto požiadavka nám definuje väzbu medzi dimenziami produkt a kategória ako 1:n. Dodávateľ pôsobí v jednom meste, ale v jednom meste môže pôsobiť viac dodávateľom, čo nám dá väzbu 1:n. Podobný princíp platí aj na väzbu medzi dimenziami zákazník a mesto, kde bude taktiež väzba 1:n.

Časové dimenzie budú mať medzi sebou väzbu 1:n. Dôvodom je, že v roku máme dva polroky. V polroku je viac mesiacov. Väzba medzi dimenziou čas a mesiac je 1:n, pretože k jednému mesiacu môže byť priradených viac inštancií definujúci čas, podľa ktorého budú dáta analyzované.

Zamestnanci majú svojho podriadeného. Na vrchu pomyselného stromu hierarchií zamestnancov však bude minimálne jeden zamestnanec, ktorý bude mať svojich podriadených.

Dimenzii „zamestnanec“ definujeme slučku, ktorá bude ukazovať „sama na seba“ a väzba tu bude 0:n. Zamestnanec môže ma viacerých podriadených zamestnancov alebo nemusí mať žiadneho podriadeného zamestnanca. Zamestnanec môže pôsobiť vo viacerých mestách, a zároveň v každom meste pracuje viac zamestnancov. Požiadavka o pôsobení zamestnancov nám definuje väzbu m:n.

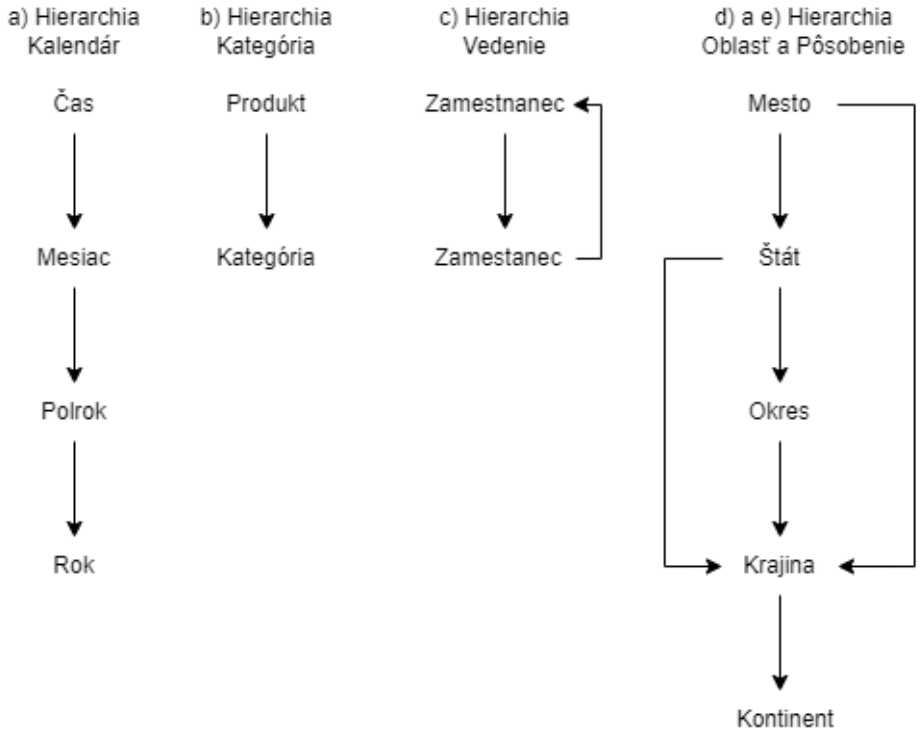
Pri dimenziách definujúcich polohu začneme od koreňu dimenzie, úrovňou „kontinent“. V kontinente máme prirodzene viac krajín a štátov spadajúcich do jedného kontinentu, teda tu bude väzba 1:n. Krajina môže mať okresy ale aj nemusí. Keď už je krajina rozdelená na okresy tak ich má viac. Preto medzi dimenziami „krajina“ a „okres“ bude väzba 0:n. Krajina ako napríklad USA pozostáva z viacerých štátov. To ale nie je prípad Slovenska, ktoré je samo o sebe štátom aj krajinou. Preto medzi dimenziami definujeme väzbu 0:n. Väzba medzi úrovňami „krajina“ a „mesto“ je 0:n, pretože ak si vezmeme ako príklad Vatikán, krajina je zároveň aj mestom.

3.2.5 Klasifikácia hierarchií

V modely môžeme vidieť, že budeme potrebovať päť hierarchií – „Kalendár“, „Kategória“, „Pôsobenie“, „Vedenie“ a „Oblasť“. V obrázku 21. môžeme vidieť usporiadanie jednotlivých hierarchií. Napríklad hierarchia „**Kalendár**“ (a) má rodičovskú úroveň „čas“. Následne potomkovia úrovne „čas“ sú úrovne „mesiac“, „polrok“ a „rok“. Hierarchia „**Kategória**“ (b) obsahuje iba jedného potomka rodičovskej úrovne „Produkt“, a tým je „kategória“. Následne hierarchia „**Vedenie**“ (c) obsahuje slučku, kde rodičovskou a potomkovou úrovňou je rovnaká dimenzia „zamestnanec“. Vyjadruje tým, že v dimenzia môže mať nadriadené a podriadené inštancie.

Hierarchie „**Oblasť**“ (d) a „**Pôsobenie**“ (e) majú rovnakú štruktúru, ale iný sémantický význam. Kým hierarchia „Oblasť“ hovorí o tom, že jeden zamestnanec môže pracovať vo viacerých oblastiach a v oblasti pracuje viac zamestnancov, hierarchia „Pôsobenie“ hovorí o úplne inej kardinalite. Jeden zákazník/dodávateľ pôsobí v jednej lokalite, ale v lokalite pôsobí viacero zákazníkov/dodávateľov. Z tohto dôvodu potrebujeme tieto hierarchie rozdeliť, aj napriek tomu, že štruktúra hierarchie je rovnaká. Rodičovská úroveň „mesto“ má podriadené úrovne „štát“, „okres“, „krajina“ a kontinent. Táto štruktúra je zostrojená tak, aby zodpovedala

celosvetovej štruktúre krajín, nie iba slovenskej, keďže spoločnosť nepôsobí na Slovensku a tiež vyváža do celého sveta. Práve z rôznych štruktúr krajín vo svete sú úrovne „mesto“ a úroveň „krajina“ prepojené a to isté je aplikované aj na úrovne „štát“ a „krajina“.



Obr. 20 - Klasifikácia hierarchií – Prípadová štúdia 1.

Následne definuje typ hierarchií figurujúcich v schéme:

Názov	**Typ hierarchie**	**Dôvod priradenia typu**
Kalendár	Vyvážená	Medzi jednotlivými úrovňami je možná iba jedna cesta, pričom sú všetky úrovne povinné
Kategória	Vyvážená	Medzi jednotlivými úrovňami je možná iba jedna cesta, pričom všetky úrovne sú povinné
Vedenie	Rekurzívna	Špeciálny prípad nevyváženej hierarchie
Oblasť, Pôsobenie	Roztrhaná	Špeciálny prípad generalizovanej hierarchie

Tab. 3 - Typy hierarchii - Prípadová štúdia 1.

Hierarchie „Oblasť“ (c) a „Pôsobenie“ (e) vyjadrujú špeciálny typ generalizovanej hierarchie, takzvaná roztrhaná alebo hierarchia preskakujúca úrovne. V prípade takejto hierarchie, alternatívne cesty vznikajú preskočením jednej alebo viacerých hierarchií. Pre každú inštanciu môže byť dĺžka cesty od listu k nadriadenému členovi odlišná. (Conceptual Data Warehousing, 2022)

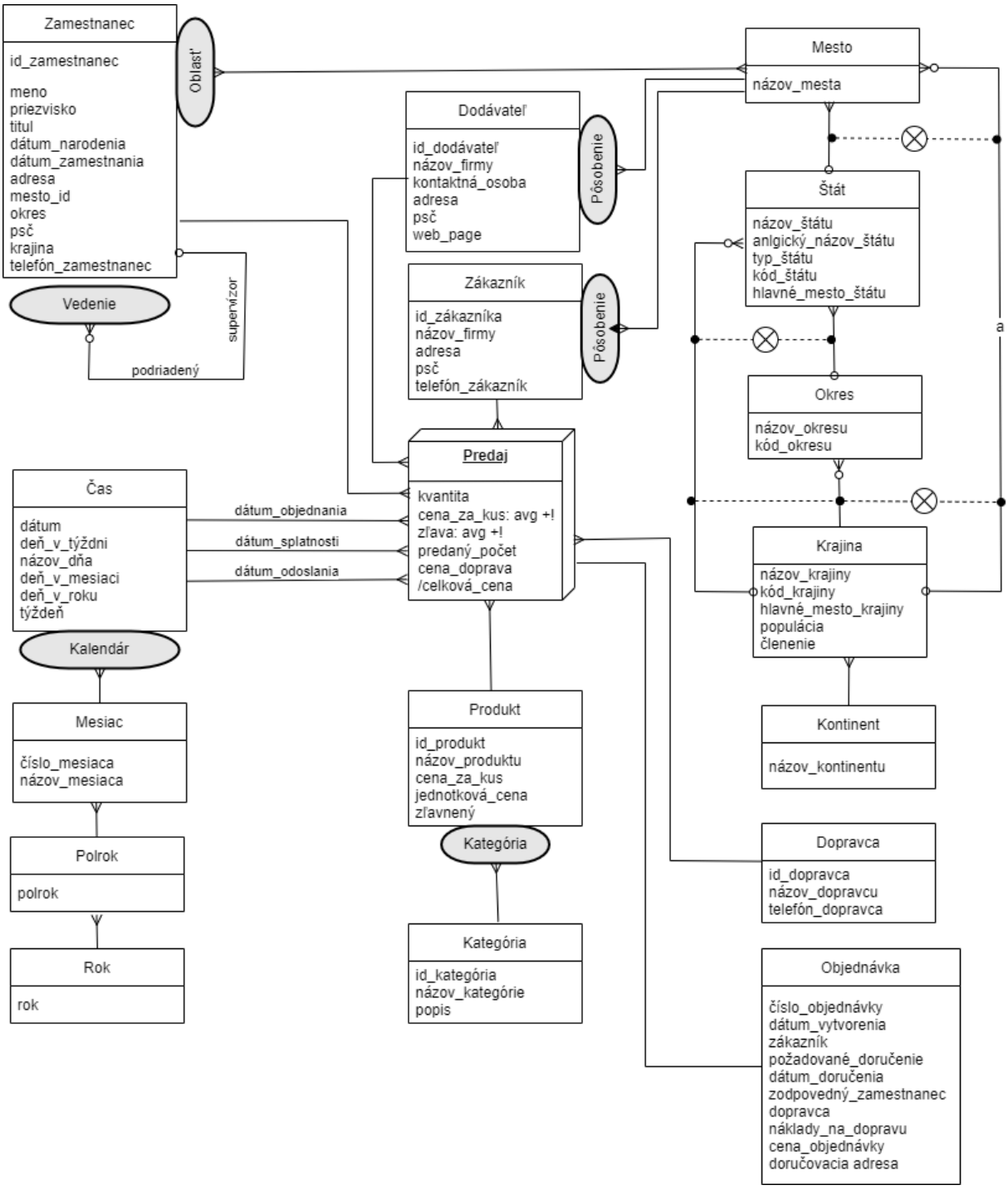
3.2.6 Určenie exkluzívnych väzieb a priradenie rozlišujúcich atribútov

Môže sa zdať, že rozlišujúci atribút bude aplikovaný na väzbu medzi dimenziami zamestnanec a mesto, kde je väzba m:n. Avšak dimenzie „Zamestnanec a „Mesto“ sú nezávislé, teda nemajú nič spoločné a preto tu rozlišujúci atribút nebude aplikovaný.

Exkluzívnu väzbu aplikujeme v hierarchii „Oblasť“/„Pôsobenie“, kde môžeme vybrať iba jednu väzbu z možných väzieb v jednej inštancii. Do úrovne „krajina“ môže vstupovať väzba z úrovní „štát“, „okres“ a mesto.

3.2.7 Konceptuálny model prípadovej štúdie 1. – E-commerce predaj

V obrázku 22. môžeme vidieť konečnú podobu MultiDimu pre konceptuálny model prípadovej štúdie 1. o e-commerce predaji spoločnosti Northwind.



Obr. 21 - Konceptuálny model prípadovej štúdie 1. (Vaisman, 2014)

3.3 Prípadová štúdia 2. – Publikácie

Ako druhú prípadovú štúdiu si uvedieme predaj publikácii. Užívateľské požiadavky toho, čo by malo byť implementované, sú nasledovné.

3.3.1 Zadanie od zákazníka

V predajni s odbornou literatúrou sa predávajú rôzne typy publikácii. Predajňa predáva časopisy, učebnice a zápisy z konferencií. Tieto publikácie sú z rôznych oblastí, ktorým sa venujú a tieto oblasti potrebujeme rozlišovať. Pobočky predajní sú rozmiestnené vo viacerých lokalitách. Predaj publikácii sledujeme z pohľadu kalendárneho a fiškálneho roku, ktoré sú rozdelené na kvartály.

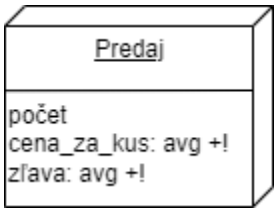
3.3.2 Určenie dimenzií

Ako prvú dimenziu definujeme dimenziu „**Predavač**“, o ktorom si budeme uchovávať základné informácie o predavačoch, ako sú *meno, priezvisko, dátum narodenia, dátum zamestnania a telefónne číslo*. Zamestnanec môže pôsobiť vo viacerých mestách. Touto požiadavkou definujeme dimenziu „**Mesto**“, ktorá bude obsahovať dva atribúty a to *identifikátor* a *názov mesta*. Predajňa predáva tri typy publikácii, ktoré budú mať spoločné atribúty, ktoré môže uchovávať dimenzionálna tabuľka „**Publikácie**“. Obsiahnuté atribúty z dimenzie sú *autor, názov, ISBN* a *kľúčové slová.* Teraz definujeme typy publikácií, ktoré budú dediť informácie z dimenzie „publikácie“. Prvým typom publikácie nám vzniká dimenzia „**Časopis**“ s atribútmi *názov, číslo* a *rok vydania*. Druhý typ nám dá dimenziu „**Učebnica**“ s atribútmi *vydanie* a *rok vydania*. Nakoniec tretím typom bude dimenzia „**Konferencia**“ s atribútmi *názov, rok vydania* a *miesto*.

V schéme nám tým vznikne štrnásť dimenzionálnych tabuliek.

3.3.3 *Definovanie faktorovej tabuľky*

Vo faktorovej tabuľke definujeme potrebné miery. Miera „počet“ je aditívna. Miery „cena za kus“ a „zľava“ sú semiaditívne miery a využijú agregačnú funkciu average pre výpočet priemernej ceny alebo zľavy na dané časové obdobie na daný produkt. „cena za kus“. Miery zakreslime do faktorovej tabuľky.



Obr. 22 - Faktorová tabuľka prípadovej štúdie 2.

3.3.4 *Určenie väzieb medzi dimenziami*

Tak ako v prípadovej štúdii 1., aj tu definujeme väzby medzi dimenziami na základe užívateľských požiadaviek. Zamestnanec môže byť priradený k predajniam vo viacerých mestách a prirodzene v jednom meste na predajni je viac predavačov. Môžeme tak povedať, že medzi dimenziami „Predavač“ a „Mesto“ definujeme väzbu m:n.

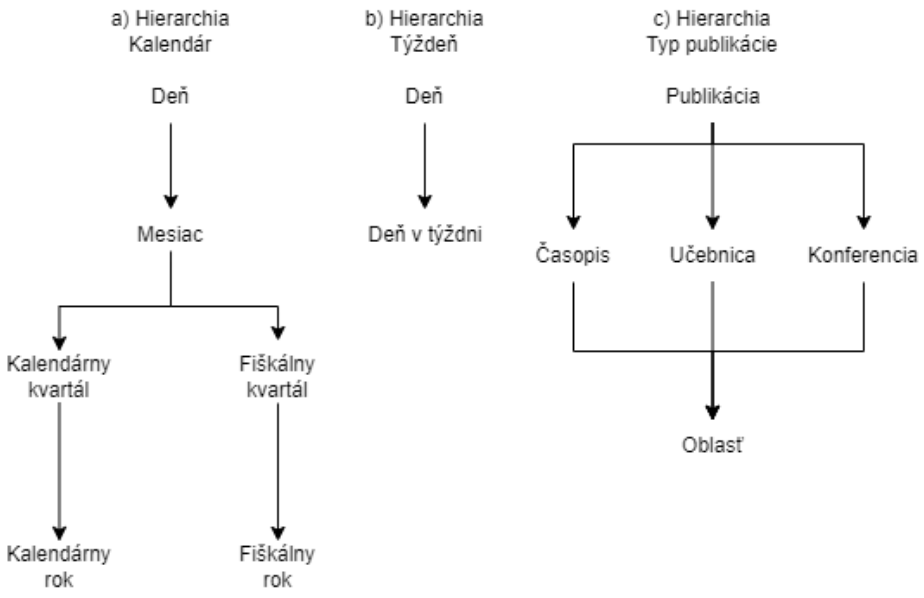
Pozrime sa ďalej na väzby medzi časovými dimenziami. Deň môže byť obsiahnutý len v jednom mesiaci. Mesiac môže patriť iba do jedného konkrétneho kvartálu, a kvartál do jedného roku. Medzi časovými dimenziami všade definujeme väzbu n:1.

Pri typoch publikácii sa musíme zamyslieť nad štruktúrou konkrétnych inštancií. Typ publikácie môže byť využitý vo viacerých publikáciách. Publikácia, na druhej strane, môže byť daný typ ale nemusí, pretože môže byť iba jedným z nich. Z toho dôvodu tu definujeme väzbu 0:n. Väzba medzi dimenziou „Oblasť“ a typom publikácie bude takisto n:0, pretože v konkrétnej oblasti, môže byť viac publikácii, ale zároveň oblasť nemusí mať priradenú ani jednu publikáciu.

3.3.5 *Klasifikácia hierarchií*

Na základe požiadaviek si určíme hierarchie, ktoré nám budú figurovať v schéme. V dátovom sklade bude potrebné rozlišovať informáciu o fiškálnom roku a kalendárnom roku. Tieto roky sa viažu na mesiace, pre ktoré je definované iné zloženie, a to na základe typu roku/kvartálu. Potrebujeme vybrať jednu z možných alternatív. Hierarchii „Kalendár“ priradíme typ alternatívna hierarchia. Táto hierarchia sa skladá z dvoch hierarchii „Kvartálny rok“ a „Kalendárny rok“. Tie zodpovedajú za zoskupenie mesiacov do kalendárnych rokov a fiškálnych rokov. Hierarchia „Týždeň“ je vyváženou hierarchiou, ktorá sa skladá z koreňovej úrovne „Deň v týždni“ a úrovne listu „Deň“.

Následne hierarchia „Typ publikácie“ je generalizovaná hierarchia, ktorá nám definuje typ publikácie, aký bude v inštancii definovaný. Hierarchia sa skladá z koreňovej úrovne „Oblasť“, ktorá definuje oblasť, ktorou sa publikácia zaoberá. Následne potomkovia tejto hierarchie sú typy jednotlivých publikácii. Dostaneme tak úrovne „Časopis“, „Učebnica“ a „Konferencia“. Úroveň „Publikácie“ bude definovať dodatočné informácie typov publikácii.



Obr. 23 - Klasifikácia hierarchií - Prípadová štúdia 2.

Následne definuje typ hierarchií figurujúcich v schéme:

Názov	Typ hierarchie	Dôvod priradenia typu
Kalendár	Alternatívna	Vyberá sa iba jedna z možných alternatív
Týždeň	Vyvážená	Medzi jednotlivými úrovňami je možná iba jedna cesta, pričom všetky úrovne sú povinné
Typ publikácie	Generalizovaná	V hierarchii existuje viac agregačných ciest, jedna pre každý typ publikácie. Všetky tri cesty patria do jednej hierarchie.

Tab. 4 - Typy hierarchii - Prípadová štúdia 2.

3.3.6 Určenie exkluzívnych väzieb a priradenie rozlišujúcich atribútov

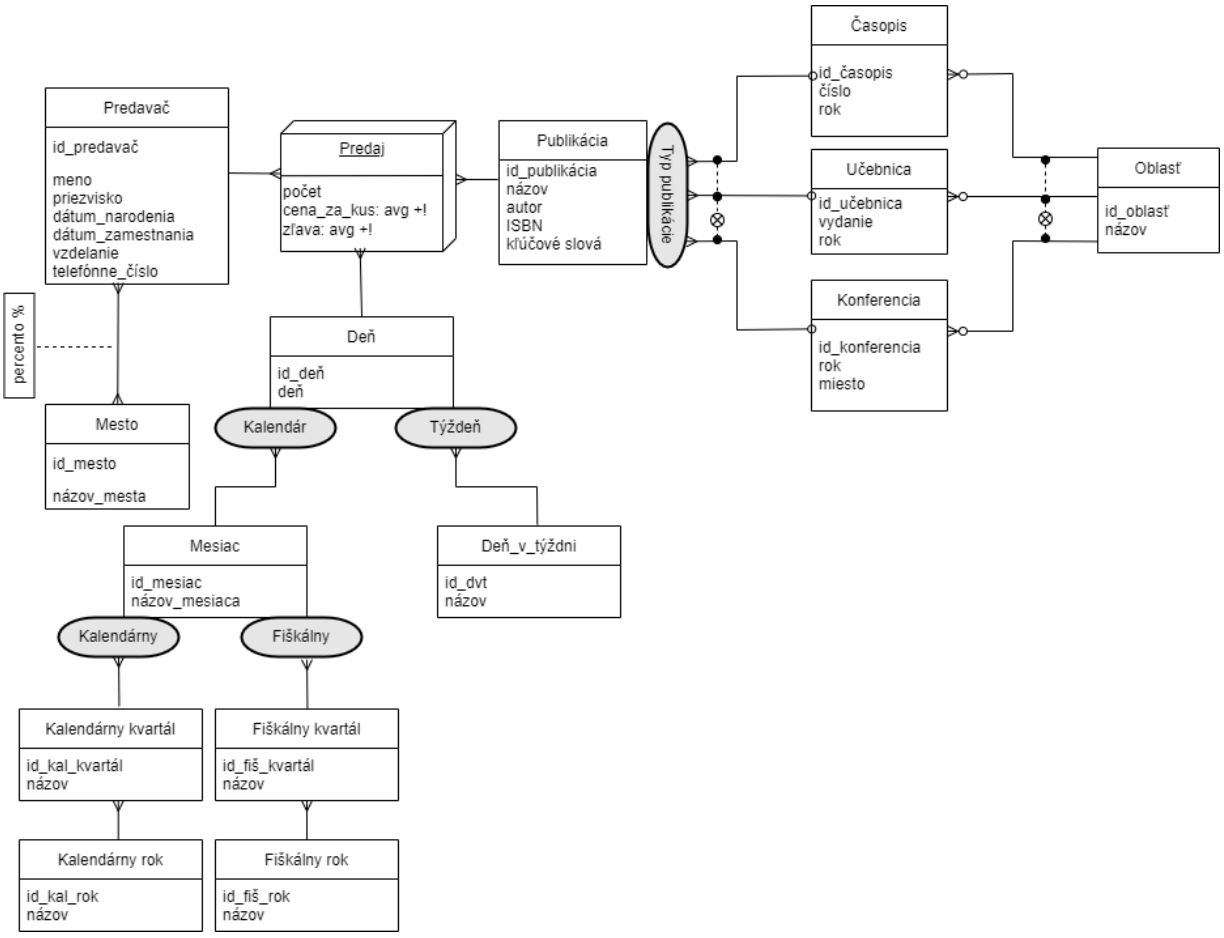
Pri výbere typu publikácie máme na výber z troch možností. Môžeme si vybrať buď, časopis, učebnicu, alebo zápis z konferencie. Problém však je, že jedna inštancia môže obsahovať iba jeden typ publikácie. Z tohto dôvodu zavedieme do tejto hierarchie exkluzívnu väzbu. V schéme sa budú nachádzať dve exkluzívne väzby, pretože sú na tieto typy publikácii naviazané dve dimenzie, „oblasť“ a publikácie. Je potrebné rozlíšiť vznik jednej cesty v inštancii.

Podobne ako v prípadovej štúdii 1., aj tu máme m:n väzbu medzi dimenziami „zamestnanec“ a „mesto“. Na rozdiel od predchádzajúceho typu sú dimenzie tentokrát závislé a preto tu potrebujeme zaviesť rozlišujúci atribút.

V tomto momente máme zavedené všetky prvky MultiDim a konceptuálny model môže byť zostavený.

3.3.7 Konceptuálny model prípadovej štúdie 2. – Publikácie

Nasledujúci obrázok 25. zobrazuje kompletný konceptuálny model pre prípadovú štúdiu 2. pre predaj publikácii.



Obr. 24 - Konceptuálny model prípadovej štúdie 2. (Hudec, 2022)

Záver

Náplňou práce bolo vytvorenie MultiDim pre konceptuálny model multidimenzionálného dátového skladu. Dosiaľ neexistoval formalizmus pre konceptuálny model multidimenzionálnych dátových skladov. Z tohto dôvodu, dizajnéri pri modelovaní pristupujú priamo k modelovaní logického modelu. Formalizmus logického modelu je jasne definovaný a jednoducho aplikovateľný. Problémom je závislosť logického modelu na implementácii dátového skladu. Štruktúra dátového skladu je zvyčajne reprezentovaná na logickej úrovni pomocou schémy star, snowflake alebo constellation. Tieto schémy poskytujú viacrozmerný pohľad na údaje tam, kde sú miery (napríklad množstvo predaných produktov), analyzované z rôznych uhlov pohľadu alebo dimenzií (napríklad podľa produktu) a na rôznych úrovniach detailov pomocou hierarchií.

V práci sme popísali fázy modelovania dátových skladov so zameraním na konceptuálne modelovanie a jeho dôležitosť. Definovali sme dôvod prečo konceptuálne modelovanie je častokrát pri návrhu dátových skladov vynechávané. Jedným z možných formalizmov, ktoré by sa pri konceptuálnom modelovaní mohli využívať je MultiDim. Po definovaní grafickej notácie prvkov MultiDim, sme sa zamerali na konkrétne príklady.

Výsledkom práce je aplikovanie nadobudnutých znalostí na dve prípadové štúdie. Vytvorili sme postup krokov, ktoré je potrebné pri modelovaní MultiDim dodržať a následne sme tieto kroky aplikovali na dve prípadové štúdie. Postupnosť definovaných krokov sa prelína a nie je fixná. Modelovanie každého konceptuálneho modelu je iteratívne a zmeny pri návrhu sú očakávané. Je teda možné, že bude potrebné sa k niektorým krokom neskôr vrátiť.

Zoznam použitej literatúry

(1) Conceptual Data Warehousing, [elektronický zdroj]. [cit. 2022-04-12]. Dostupné na: https://documents.uow.edu.au/~jrg/312sim/lectures/10conceptdwdesign/10conceptdwdesign.html#2

(2) Databázové systémy, [elektronický zdroj]. 2006, [cit. 2022-02-12]. Dostupné na: https://spseke.sk/tutor/prednasky/databazove_systemy.htm

(3) Dátové sklady a OLAP, [elektronický zdroj].[cit. 2021-11-2]. Dostupné na: http://matlab.fei.tuke.sk/subjects/mis/subory/podklady/prednaska_8.pdf

(4) Dimensional Data Modeling, [elektronický zdroj].[cit. 2022-01-20]. Dostupne na: https://datacadamia.com/data/type/cube/modeling/hierarchy

(5) GOLFARELLI, Matteo - MAIO, Dario - RIZZI, Stefano. *The dimensional fact model: a conceptual model for data warehouses*. [elektronický zdroj]. Bologna : International Journal of Cooperative Information Systems, 1998. 32 s. Dostupné na: http://www-db.deis.unibo.it/~srizzi/PDF/ijcis98.pdf

(6) HUDEC, Miroslav. *Business Inteligence _2021/22_Hudec, Horniakova _novy*, [elektronický zdroj]. [cit. 2022-04-12]. Dostupné na: https://moodle.euba.sk/course/view.php?id=358

(7) HÜSEMANN, Bodo - LECHTENBÖRGER, Jens - VOSSEN, Gottfried. *Conceptual data warehouse design*. [elektronický zdroj]. In: Workshop on Design and Management of Data Warehouses (DMDW), Stockholm, Sweden, 2000. 11 s. Dostupné na: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.33.670&rep=rep1&type=pdf

(8) KIMBALL, Ralph – ROSS, Margy. *The data warehouse toolkit : The definitive guide to dimensional modeling*. 3. vyd. Kanada : John Wiley & Sons, 1996. 520 s. ISBN 978-1-118-53080-1.

(9) MALINOWSKI, Elzbieta - ZIMANYI, Esteban. *A conceptual model for temporal data warehouses and its transformation to the ER and the object-relational models*. Brusel: Université Libre de Bruxelles. 2008. 32 s. ISBN 978-3-540-74405-4.

(10) MALINOWSKI, Elzbieta - ZIMÁNYI, Esteban. *OLAP Hierarchies: A Conceptual Perspective*. [elektronický zdroj]. In: Persson, A., Stirna, J. (eds) Advanced Information

Systems Engineering. CAiSE. 2004. 477-491 s. Dostupné na: https://doi.org/10.1007/978-3-540-25975-6_34

(11) Olap.com [elektronický zdroj]. [cit. 2022-03-10]. Dostupné na: https://olap.com

(12) Oracle. *Benefits of a Multi-Dimensional Model*, [elektronický zdroj]. [cit. 2021-11-22]. Dostupné na: https://www.oracle.com/technetwork/developer-tools/warehouse/benefits.pdf

(13) Oracle. *Modelovanie podnikových dát v službe Oracle Analytics Cloud*, [elektronický zdroj]. [cit. 2022-02-12]. Dostupné na: https://docs.oracle.com/cloud/help/sk/analytics-cloud/ACSMD/GUID-21532266-BAA8-4312-A0DF-3AD103B59FC1.htm#BILUG431

(14) RIVEST, S. a kol. S*olap: a new type of user interface to support Spatio-temporal multidimensional data exploration and analysis*. Quebec : Centre for Research in Geomatics, 2003. 8 s.

(15) RIZZI, Stefano a kol. *Research in data warehouse modeling and design: dead or alive*. [elektronický zdroj]. Arlington: In: Proceedings of the 9th ACM International Workshop on Data Warehousing and OLAP, 2006. 8 s. Dostupné na: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.79.801&rep=rep1&type=pdf

(16) TRUJILLO, Juan. *Designing Data Warehouses with OO Conceptual Models*. [elektronický zdroj]. 2001. 66-75 s. Dostupné na: http://www.ischool.drexel.edu/faculty/song/publications/p_IEEE-OODW-2001.pdf

(17) TUPPER, Charles. *Data Architecture.* Burlington : Elsever, 2011. ISBN: 978-0-12-385126-0

(18) TURBAN, Efarim a kol. *Business Intelligence*. 2. vyd. Boston : Prentice Hall, 2010. 312 s. ISBN 978-0136100669

(19) VAISMAN, Alejandro – ZIMÁNYI, Esteban. *Data Warehouse Systems : design and implementaion*. 1. vyd. Berlin : Springer-Verlag Berlin Heidelberg, 2014. 625 s. ISBN 978-3-642-54654-9.