

Mendelova univerzita v Brně
Provozně ekonomická fakulta

Business Intelligence v prostředí informačního systému QI

Diplomová práce práce

Vedoucí práce:
Ing. Jan Přichystal, Ph.D.

Bc. Tomáš Blažek

Brno 2015

Rád bych poděkoval vedoucímu práce panu Ing. Janu Přichystalovi, Ph.D. za odborné vedení práce, cenné rady a připomínky při tvorbě bakalářské práce.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Business Intelligence v prostředí informačního systému QI**

vypracoval samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací*.

Jsem si vědom, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona.

Dále se zavazuji, že před sepsáním licenční smlouvy o využití díla jinou osobou (subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmětná licenční smlouva není v rozporu s oprávněnými zájmy univerzity, a zavazuji se uhradit případný příspěvek na úhradu nákladů spojených se vznikem díla, a to až do jejich skutečné výše.

V Brně dne 22. 5. 2015

.....

Abstract

Blažek, T. Business Intelligence of the QI information system environment. Master thesis. Brno: MU in Brno, 2015.

This work provides a survey of creation of the Business Intelligence by using QI information system data. The work concisely describes Business intelligence functions and its user's environment. QI system and the way of data storage are also defined in this dissertation. The following chapters describe the design and implementation of Business Intelligence itself over the actions made by registered employees. The proposal describes the design of a data storage system, a method of obtaining the necessary data from the QI information system, ETL process design, OLAP cubes design, and how to interpret the information in the form of reports. The last chapter then describes the realisation of all the proposed districts in MS SQL Server.

Abstrakt

Blažek, T. Business Intelligence v prostředí Informačního systému QI. Diplomová práce. Brno: MU v Brně, 2015.

V této práci je popsána tvorba Business Intelligence nad daty informačního systému QI. Je zde stručný popis, co je to Business Intelligence a jak vypadá informační systém QI z pohledu uživatele. Dále je popsáno, jak je QI navrženo a způsob uchování dat. V následujících kapitolách je popsán návrh a samotná realizace Business Intelligence nad evidovanými akcemi zaměstnanců. V návrhu je popsán návrh datového skladu, způsob získání potřebných dat z informačního systému QI, návrh ETL procesu, návrh OLAP kostky a způsobu interpretace informací v podobě reportů. V poslední kapitole je pak popsána realizace všech navržených částí v prostředí MS SQL Serveru.

Obsah

1	Úvod a cíl práce	8
1.1	Úvod	8
1.2	Cíl práce	8
2	Metodika	9
3	Business Intelligence	11
3.1	Datový sklad	11
3.2	ETL proces	11
3.3	Datové kostky	13
3.4	Reporting	13
4	Informační systém QI	15
4.1	Moduly	15
4.1.1	Ekonomika	15
4.1.2	Lidské zdroje	16
4.1.3	Obchod a marketing	16
4.1.4	Řízení firmy	16
4.1.5	Výroba	17
4.1.6	E-moduly	17
4.1.7	Specializované moduly	17
4.1.8	Analýza	17
4.2	Elasticita systému	18
4.3	Struktura databáze	18
5	Návrh řešení	23
5.1	Požadavky	23
5.2	Nalezení potřebných dat	24
5.3	Potřebné datové řezy QI	25
5.4	Zdroje dat mimo QI	26
5.5	Návrh databáze datového skladu	26
5.6	Atributy tabulek	27
5.6.1	Dimenze Místo	27
5.6.2	Dimenze DatumHodina	27
5.6.3	Dimenze Uživatel	28
5.6.4	Dimenze Obchodní partner	28
5.6.5	Dimenze DruhTyp	28
5.6.6	Dimenze Stav	28
5.6.7	Faktová tabulka Akce	28
5.7	Návrh ETL procesu	29
5.8	Extrakce	29
5.8.1	Dimenze Místo	30

5.8.2	Dimenze DatumHodina	30
5.8.3	Dimenze Uživatel	30
5.8.4	Dimenze Obchodní partner	30
5.8.5	Dimenze DruhTyp	30
5.8.6	Dimenze Stav	31
5.8.7	Faktová tabulka Akce	31
5.9	Transformace	31
5.9.1	Dimenze Místo	31
5.9.2	Dimenze DatumHodina	31
5.9.3	Dimenze Uživatel	32
5.9.4	Dimenze Obchodní partner	32
5.9.5	Dimenze DruhTyp	32
5.9.6	Dimenze Stav	32
5.9.7	Faktová tabulka Akce	32
5.10	Nahrávání	33
5.11	Návrh tvorby OLAP kostky	33
5.12	Návrh způsobu predikce informací	34
5.13	Návrh interpretace informací	35
6	Implementace	37
6.1	Nalezení potřebných dat	37
6.2	Realizace databáze datového skladu	40
6.2.1	Tabulka Místo	40
6.2.2	Tabulka DatumHodina	41
6.2.3	Tabulka Uživatel	41
6.2.4	Tabulka Obchodní partner	42
6.2.5	Tabulka DruhTyp	43
6.2.6	Tabulka Stav	43
6.2.7	Tabulka Akce	43
6.2.8	Pomocné tabulky	44
6.3	Realizace ETL procesu	45
6.3.1	Dimenze Místo	45
6.3.2	Dimenze DatumHodina	46
6.3.3	Dimenze Uživatel	46
6.3.4	Dimenze Obchodní partner	49
6.3.5	Dimenze DruhTyp	51
6.3.6	Dimenze Stav	52
6.3.7	Faktová tabulka Akce	52
6.3.8	Vložení nulových hodnot	54
6.3.9	Nahrání tabulek	55
6.4	Realizace OLAP kostky	55
6.4.1	Nadefinování atributů a hierarchií	56
6.4.2	Nadefinování automaticky počítaných hodnot	57

6.4.3	Nadefinování KPI	59
6.5	Realizace predikce informací	59
6.6	Interpretace informací	61
6.6.1	Tvorba MDX dotazů	61
6.6.2	Tvorba reportů	62
7	Diskuze a zhodnocení	70
7.1	Nedostatky a přednosti implemenace	70
7.2	Zhodnocení	70
7.3	Další možná rozšíření	71
7.4	Ekonomické zhodnocení	71
8	Závěr	73
9	Reference	74

1 Úvod a cíl práce

1.1 Úvod

Informační systém QI vyvíjen českou společností *DC Concept* využívá dnes přes osm set firem na českém trhu (QI, 2012a). I přes jeho robustnost nabízí informační systém QI pouze několik málo možností analýzy nad ukládanými daty. V dnešní době, kdy jsou ukládány do databází denně velké objemy dat, je pro člověka čím dál tím více náročnější z těchto dat získat užitečné informace. Pro získání přehledných informací, které by pomohly firemnímu managementu v rozhodování a tvorbě dlouhodobých či krátkodobých plánů nebo v přehledu nad samotným chodem firmy, je více než nutné využití automatizovaných analytických nástrojů. *Business Intelligence* (BI) je obor, který se touto problematikou zabývá. Využitím těchto nástrojů a metodik lze účinným způsobem vytvořit reporty, které pomáhají odhalit klíčové znalosti, díky kterým může firma zlepšit svoje postavení na trhu.

Firma *SolidVision* poskytla pro tuto práci svoje data a informační systém QI. Společnost vznikla v roce 2002 a poskytuje komplexní 3D CAD řešení postaveného na platformě *SolidWorks*. Mimo jiné také nabízí konstrukční činnosti a 3D skenování. Dnes má firma více než 80 zaměstnanců a stala se předním distributorem systému *SolidWorks* pro český a slovenský trh (SOLIDVISION, 2010). Pořídit komplexní firemní informační systém QI se firma rozhodla roku 2010, aby podpořila firemní procesy a řízení celkového chodu firmy. Dnes je informační systém QI nedílnou součástí každodenní práce každého zaměstnance firmy.

1.2 Cíl práce

Cílem této práce je vytvořit na produkčním serveru firmy *SolidVision Business Intelligence* řešení, které umožní analýzu dat informačního systému QI. Přesněji se jedná o tvorbu BI nad akcemi zaměstnanců firmy. Požadavkem managementu bylo vytvořit reporty, které by umožnily vedení dohlížet nad plněním obchodních plánů svých obchodníků a celkový pohled nad produktivitou svých zaměstnanců a firmy. Před každým úspěšně uzavřeným obchodem musí obchodník vykonat určité akce (například jednání, telefonní hovory, emaily), které zabírají obchodníkovi čas. Vytvořené BI řešení nad akcemi zaměstnanců poskytne vedení firmy získat přehled nad tím, jak jsou obchodníci úspěšní vzhledem k vykonaným akcím. Výsledné reporty budou vedení firmy informovat o celkových počtech jednotlivých druhů akcí zaměstnance, denních průměrech akcí zaměstnance, o počtu akcí zobrazených v poměru vůči kolegům z oddělení, stavu obchodníka vzhledem k plnění obchodního plánu a průměrných počtů akcí pro prodej. Dále budou reporty také informovat o obchodních partnerech s největším počtem realizovaných akcí a rozmístění obchodních partnerů po ČR obchodního referenta a celé firmy. Nad akcemi bude také realizována predikce počtu obchodních případů, která bude vytvořena ve dvou různých nástrojích, aby bylo možné porovnat výsledky predikce.

2 Metodika

Jelikož produkční server společnosti *SolidVision* běží pod systémem *MS SQL Server 2008 R2* bude nutné seznámit se s problematikou tvorby BI v tomto prostředí. Výhodou je, že tento systém obsahuje nástroje určené pro tvorbu BI. Dále bude nutné seznámit se se strukturou databáze informačního systému QI a zjistit, jak informační systém s daty pracuje. K získání těchto informací bude sloužit vývojářské školení ve firmě *DC Concept* a znalostní báze v informačním systému QI. Na základě získaných znalostí bude vytvořen návrh tvorby BI řešení. Návrh bude popisovat požadavky managementu na výsledné BI řešení, způsob získání potřebných dat z informačního systému QI a externích zdrojů, návrh datového skladu, naplnění datového skladu ETL procesem, návrh tvorby OLAP kostky, způsob predikce obchodních případů a výslednou interpretaci informací koncovým uživatelům. Návrh bude v souladu se zažitými BI standardy a bude konzultován s panem Ing. Janem Přichystalem, Ph.D. Návrh způsobu interpretace informací bude také konzultován s managementem firmy *SolidVision*.

Po sepsání návrhu bude možné přejít k samotné realizaci. Pro nalezení potřebných dat bud zapotřebí použít samotný informační systém QI, protože tabulky a atributy databáze nenesou názvy, ale číselné identifikátory. Databáze je tedy pro člověka nečitelná. Jakmile budou stanovena potřebná data, bude následovat vytvoření datového skladu. Datový sklad bude vytvořen v nástroji *Microsoft SQL Server Management Studio 2008*, protože umožňuje pohodlnou správu nad databází serveru. V rámci datového skladu bude vytvořena databáze určená pro daný datový sklad a následně tabulky datového skladu. Názvy tabulek a atributů budou získány z návrhu.

Po vytvoření datového skladu bude nutné vytvořit SQL dotazy, které budou získávat potřebná data z databáze informačního systému QI. Dotazy budou vytvářeny také v *Management Studiu*, protože umožňuje jednoduchý přístup do databáze QI, kontrolu syntaxe příkazů a kontrolu existence tabulek a atributů. Pro přípravu externích dat bude využit nástroj *MS Excel*, který je vhodný pro úpravu strukturovaných dat uchovávaných mimo databáze. Po vytvoření dotazů budou realizovány ETL procesy. Procesy budou implementovány ve vývojovém prostředí *SQL Server Business Intelligence Development Studio*, který je součástí *MS SQL Serveru*. Tento nástroj je přímo určen pro tvorbu BI na serverech Microsoftu. Pro realizaci ETL slouží v tomto programu komponenta *Integration Services (SSIS)*, která umožní nadefinovat celý ETL proces od nadefinování zdroje extrakce po určení cíle nahrávání dat.

Jakmile bude datový sklad naplněn, vytvoří se ve stejném vývojovém prostředí OLAP kostka. Ke tvorbě OLAP kostky bude využita komponenta *Analysis Services (SSAS)*. Tato komponenta obsahuje všechny potřebné funkce pro vytvoření datové kostky a následnou analýzu dat. Zde bude určen zdroj dat (datový sklad), nadefinovány vazby mezi dimenzemi a faktovou tabulkou, hierarchie v dimenzích a počítané atributy. Tato komponenta bude využita i pro zadanou predikci, protože obsahuje již

předdefinovaný model pro predikování časových řad. Predikce bude také realizována v expertním modelu pana doc. Ing. Jiřího Trenze, Ph.D. z Mendelovy univerzity, aby bylo možné porovnat výsledky s nezávislým nástrojem, který nevytvářela společnost *Microsoft*. Tento nástroj byl vybrán z důvodu zkušenosti získané na několika seminářích, kde byly tímto nástrojem pokusně predikovány měnové kurzy. Výsledky těchto predikcí budou následně porovnány a zhodnoceny. Výsledná OLAP kostka bude nahrána na zmíněný produkční server, aby byla připravena pro tvorbu reportů.

Následně bude možné vytvořit jednotlivé dotazy na OLAP kostku, které zajistí získání potřebných dat pro zobrazení požadovaných informací. Pro tento účel bude použit nástroj *PowerPivot* společnosti *Microsoft*. *PowerPivot* je zdarma dodávaná komponenta k programu Excel profesionální edice. Jakmile Microsoft uvolnil *PowerPivot* pro *Excel 2010*, umožnil vytvářet i velké BI řešení přímo v nástroji *Excel* a tím zpřístupnil BI pro uživatele *Excelu* tzv. *self-service BI*. BI tedy již nebylo pouze pro profesionály ve svém oboru (Ferrari, 2013). *PowerPivot* v prostředí *Excelu 2013* bude připojen k OLAP kostce a dotazy budou realizovány jazykem MDX¹, který je přímo určený k dotazování se na OLAP kostku. MDX dotazy budou vytvořeny v MDX editoru nástroje *PowerPivot*. Nástroj *PowerPivot* byl zvolen pro jednoduchou možnost připojení OLAP kostky a také pro jeho MDX editor, kde se dotazy vytvářejí jednoduchým „drag and drop“ nástrojem.

Po vytvoření dotazů bude následovat tvorba samotných reportů (dashboardů). Tyto reporty budou tvořeny komponentou *PowerView* společnosti *Microsoft* dodávanou také zdarma do nástroje *Excel*. *PowerView* umožňuje vytvářet grafické analýzy dat a uživatelsky jednoduchou formou je upravovat (Ferrari, 2013). Důvod výběru tohoto nástroje pro tvorbu reportů je možnost vytváření interaktivních grafů s minimalistickým designem. Reporty budou vytvářeny dle návrhu a výsledek pak bude konzultován s managementem firmy.

Schválené reporty budou předány managementu a ten bude následně zaučen do práce s uživatelským prostředím a obeznámen s možnostmi *PowerView* nástroje *Excel 2013*.

¹MultiDimensional Expression

3 Business Intelligence

Business Intelligence (BI) je soubor nástrojů, které napomáhají získávat informace z podnikových informačních systémů. První zmínka o *Business Intelligence* byla v 90. letech 20. století, jako o nástroji reportů a analýz dat uložených v datovém skladu (Thorntwaite, 2008). Informace získané z BI aplikací by především měly sloužit managementu, jako podpora pro rozhodování podnikových strategií, popřípadě jako nástroj pro zhodnocení efektivnosti již realizovaných změn a plánů. Modelovým příkladem je případ, kdy firma změní vlastnost svého produktu a nyní by ráda zjistila dopad této změny na tržby. I na tuto otázku dokáže BI aplikace odpovědět, pokud je správně vytvořena. BI je možné rozdělit do čtyř základních částí.

3.1 Datový sklad

Datový sklad² je důležitou součástí BI. Jedná se o databázi, kam se přenášejí data z informačních systémů, která bude BI aplikace využívat. Datový sklad má odlišnou strukturu od běžných transakčních databází. Jedná se o strukturu analytických databází, ve kterých se data neudržují v normalizované formě. Návrh a tvorba datového skladu je nazývána dimenzionální modelování.

Hlavním cílem dimenzionálního modelování je vytvořit základní logiku uložení a uspořádání dat tak, aby vyhovovala požadavkům na analytické a plánovací aplikace podnikového řízení (Pour, 2012). Tato struktura je výhodná nejen pro tvorbu dotazů ale také pro její srozumitelnost koncovými uživateli. Hlavní myšlenkou dimenzionálního modelování je rozdělení tabulek databáze do dvou typů. Primární tabulkou je tabulka faktová, která obsahuje číselné atributy (například počet prodaných produktů a jejich peněžní výnos) (Kimball, 2002). Druhým typem tabulek jsou tzv. dimenzionální tabulky, které obsahují především textové atributy. Ty umožňují nahlížet na číselné atributy faktových tabulek z různých pohledů. Nejčastější dimenzí je dimenze časová. Jedná se o výčet dnů, popřípadě hodin (záleží na požadavcích zadavatele – to určuje tzv. granularitu faktové tabulky) a k nim atributy, které určují mimo jiné rok, název měsíce, název dne, dále pokud je den pracovní či svátek atd. Tyto atributy slouží ve výsledné aplikaci především k filtrování, nebo k výpočtu dalších hodnot.

Výsledný návrh datového skladu má vzhledem k rozdělení tabulek do dvou druhů podobu hvězdy (tzv. Star schéma). Faktová tabulka se nachází uprostřed a je napojena na všechny dimenze, které jsou rozprostřeny kolem ní.

3.2 ETL proces

ETL je proces získávání dat z podnikových databází do datového skladu. Často je ve firmách několik různých systémů, které ukládají data. ETL proces tyto data shromáždí a upraví do jednoho místa tak, aby měly jednotnou a srozumitelnou formu

²Datový sklad je uložení dat, které byly získány z podnikových systémů pro BI aplikaci.

pro koncového uživatele. Je důležité, aby BI aplikace zobrazovaly vždy aktuální výsledky a proto se ETL již vytvořený ETL proces provádí běžně každý den. Nejčastěji to bývá v pozdních nočních hodinách, kdy podnikové servery nejsou vytíženy velkým množstvím dotazů od pracujících zaměstnanců. ETL proces je základním kamenem celého BI. Tento proces je potřeba dobře navrhnout a otestovat. Pokud by došlo ke špatnému načítání či ztrátě informací, tak již nic nepomůže tomu, aby BI aplikace podávaly správné informace. Jak zmiňuje ve své knize (Kimball, 2004) ETL systém je tvůrcem, ale i zhoubou BI. ETL proces rozdělujeme do tří částí (extrakce, transformace a ukládání).

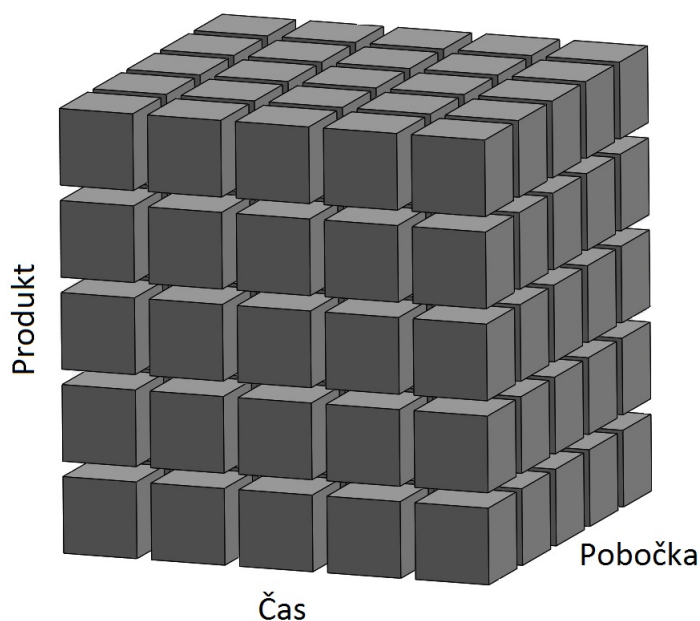
Extrakcí rozumíme nalezení zdrojů dat a jejich filtrací, aby byly získány pouze ta data, která budou potřeba pro získání požadovaných informací. Například pro získání informací o tržbách firmy nebudeme potřebovat data z docházkového systému. Častým problémem u propojení dat z více systémů bývá nekonzistence těchto dat. Zdroje dat mohou být i externí. Například data z hydrometeorologického ústavu mohou napomoci ke zjištění závislosti počtu objednávek na počasí. Rozdělujeme tři možnosti extrakce (iniciální, periodická a občasná). Iniciální se provádí především z archivních dat, periodická z interních zdrojů (IS) a občasná ze zdrojů externích.

Transformace dat je mezistupeň mezi prací se vstupní databází a výstupní. Tato část představuje mapování vstupních dat na data v datovém skladu. S tím je spojeno několik častých úprav, jako je například konverze datových typů. Například čas databáze ukládají různými způsoby. Tyto způsoby je potřeba sjednotit do jedné formy. U konverze datového typu je nutné dávat velký pozor na ztrátu dat způsobenou ořezem. Dále zdrojová data obsahují často prázdné hodnoty. Je vhodné tyto hodnoty buď doplnit, nebo je prezentovat uživateli vhodnou formou. Nejčastěji se nahrazují hodnotami „Unknown“ nebo „N/A“. Nulové hodnoty lze také ignorovat, ale je potřeba, aby při odstranění tohoto záznamu nedošlo k mylné hodnotě na BI výstupu. V rámci cílové databáze se také vytvářejí nové primární a cizí klíče, aby nedošlo k poškození dat při změně ve zdrojové databázi. Nové primární klíče také slouží k uchování historických údajů. Při změně barvy produktu se nemusí přepisovat záznam v cílové databázi, ale je možno založit nový záznam se stejným ID zdrojové databáze. Tento způsob se nazývá *Slowly Changing Dimension* a umožňuje náhled na data v různých časových řezech.

Ukládání dat se provádí většinou pomocí relačních databází, které představují datový sklad BI. Transformovaná data jsou uložena do datového skladu a jsou připravena pro datovou kostku ke zpracování. Existují tři způsoby ukládání dat (iniciální, inkrementální a přepis). Iniciální se provádí při prvním nahrání dat do datového skladu. Jedná se často o velké množství dat v řádech desítek tisíců či statisíců údajů. Tato náročná operace by měla být prováděna v době, kdy je server nejméně vytížen. Inkrementální nahrávání představuje promítnutí změn ve zdrojových databázích do datového skladu (provádí se periodicky). Posledním typem je přepis dat, díky kterému dochází ke kompletnímu smazání datového skladu a jejímu znovunaplňení aktuálními daty. Zde se ovšem ztrácí informace o změně dat, která bývá často žádaná, jak již bylo zmíněno výše.

3.3 Datové kostky

Nad multidimenzionálním datovým skladem se vytvářejí tzv. datové (OLAP³) kostky. Technologie OLAP umožňuje rychlé zpracování a efektivní analýzy multidimenzionálních dat. Principem technologie je několikadimenzionální tabulka umožňující velmi rychle a pružně měnit jednotlivé dimenze a přinášet tak uživateli různé pohledy na vlastní data (Pour, 2012). Jak je znázorněno na obrázku 1 osy datové kostky tvoří jednotlivé dimenze. Na průniku dimenzí se nacházejí předzpracovaná data.



Obrázek 1: OLAP kostka

Datové kostky umožňují seskupit dimenzionální atributy do hierarchií. Jedná se o důležitou funkcionalitu, kde je stanovena vazba mezi atributy. Pokud například produkt může být určitého druhu a tyto druhy se dělí do dalších poddruhů, je potřeba mezi nimi vytvořit hierarchickou vazbu, která určuje, že daný druh produktu může nabývat pouze určené poddruhy.

3.4 Reporting

Reporting je činnost spojená s dotazováním se na databáze pomocí standartních rozhraní těchto databází (Pour, 2012). Jedná se o samotné zobrazení získaných informací z datového skladu. K této části přistupují samotní uživatelé BI aplikací. Rozlišujeme různé druhy reportů. Jedním ze způsobů je umožnění uživateli přímo tvořit dotazy. Další způsob je vytvoření již předpřipravených dotazů zobrazených

³On Line Analytical Processing

ve formě grafů či tabulek. Tyto předpřipravené reporty je vhodné doplnit o alespoň minimální způsob interaktivity ve formě různých filtrů. Předpřipravené reporty je možné seskupit do tzv. dashboardů, které poskytují manažerům přehled nad různými oblastmi dat. Tuto část BI není dobré zanedbat, protože je to jediná část, kterou uživatelé vidí a jak je známo design je to, co aplikaci prodává.

4 Informační systém QI

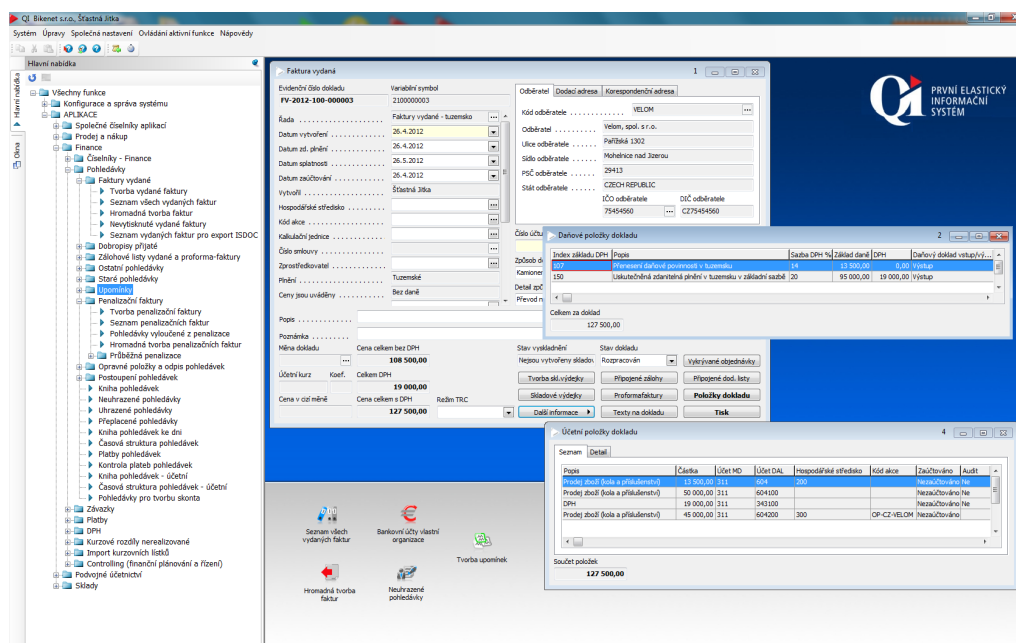
QI je informační systém vyvíjen českou společností *DC Concept* se sídlem v Brně. Jedná se o komplexní systém typu ERP⁴ pro střední a velké podniky. Umožňuje jednoduchou správu a automatizaci firemních procesů. Pomáhá hlídat termíny projektů, účetní deníky a stavy zásob či zdrojů. (QI, 2012b) Zkratka QI poskytne firmě všechny běžné výhody nasazení informačního systému.

4.1 Moduly

Informační systém QI má širokou paletu působnosti. Aby nemusely firmy z různého odvětví platit za funkcionalitu, kterou by ani nevyužily, tak *DC Concept* rozdělil QI do třiceti samostatně fungujících modulů. Tím také umožňují nasazení QI do firem menších, které tak nemusejí platit za robustní informační systém jako velké firmy, protože nemají tolik druhů procesů. Zmíněných třicet modulů je rozděleno do sedmi kategorií.

4.1.1 Ekonomika

Kategorie *Ekonomika* zahrnuje moduly *finance*, *účetnictví*, *majetek* a *insolvenční rejstřík*. Poskytuje komplexní informace pro účetní a vedoucí pracovníky. Umožňuje mimo jiné jednoduchou správu pohledávek a závazků.



Obrázek 2: Modul ekonomika (QI, 2012c)

⁴Enterprise resource planning je typ informačního systému, který spravuje všechny firemní procesy

4.1.2 Lidské zdroje

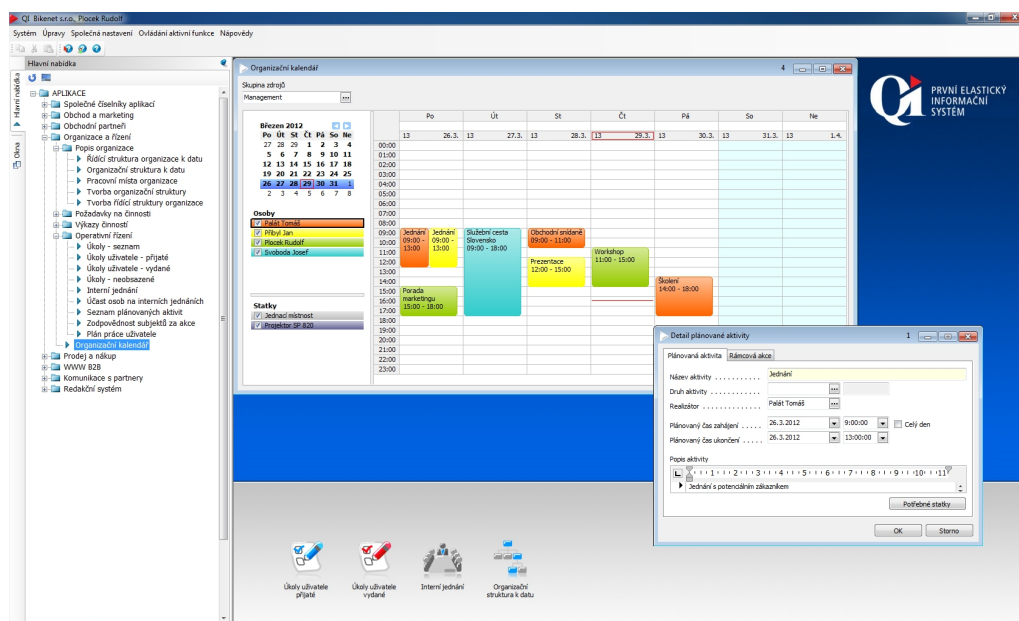
Kategorie *Lidské zdroje* obsahuje moduly *personalistika*, *mzdy* a *docházka*. Moduly umožňují podrobnou evidenci zaměstnanců a vedení docházky. Moduly využije také personální oddělení pro nábor nových zaměstnanců a evidenci osob, které projevíly zájem o zaměstnání.

4.1.3 Obchod a marketing

Další kategorií je *Obchod a marketing*, který obsahuje moduly *sklady*, *obchod*, CRM a *Marketing*. Tyto moduly umožňují jednoduchou správu dokladů spojených s pohybem zboží. Napomáhají k rychlému a spolehlivému vyřízení objednávek.

4.1.4 Řízení firmy

Důležitá kategorie, kterou ocení větší firmy, nese název *Řízení firmy*. Obsahuje moduly *organizace a řízení*, *procesy*, *projekty* a *workflow*. Tyto moduly poskytují přehled o projektech a jejich řízení (přehled je umožněn například pomocí přehledného Ganttova diagramu). Zde je také velmi užitečný organizační kalendář (viz obrázek 3), který zobrazuje vytížení zaměstnanců a statků. Tento modul jistě ocení i firmy s malým počtem zaměstnanců.



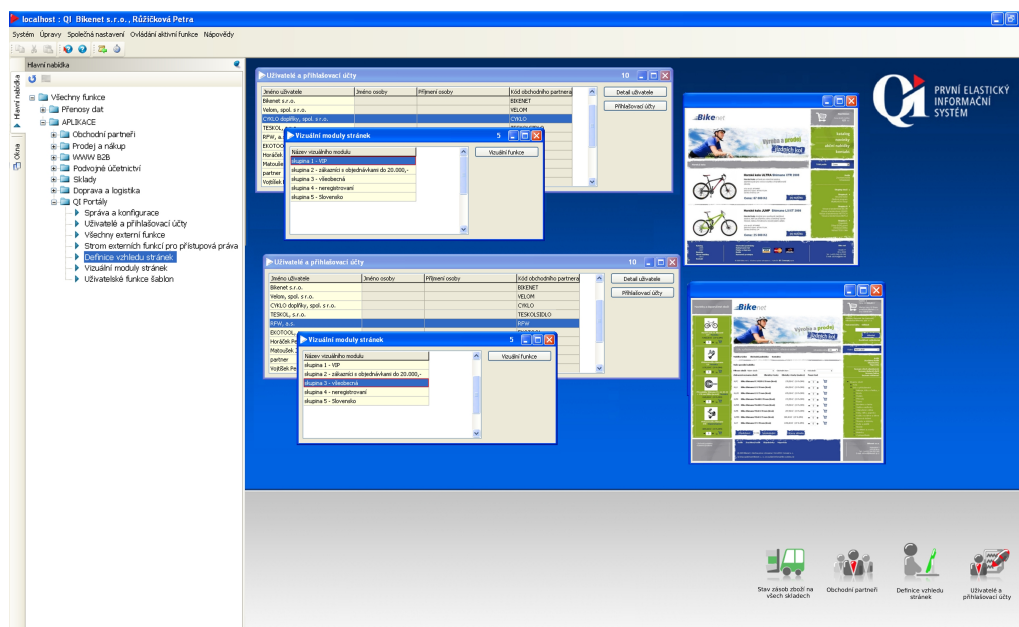
Obrázek 3: Řízení firmy (QI, 2012d)

4.1.5 Výroba

Výrobní podniky bude jistě zajímat kategorie *Výroba* s moduly *kapacity*, *tpv*, *jakost*, *plánování výroby* a *řízení výroby*. Umožňuje efektivně využít lidské zdroje a dohlížet na skladové zásoby. Výhodou je také jednoduchá organizace kapacit prostorů například pro plánování školení.

4.1.6 E-moduly

Zajímavou kategorií jsou *E-moduly*. Jsou to moduly *QI portál*, *QI shop*, *QI helpdesk*, *Redakční systém* a *QI mobile*. Tyto moduly nabízí pohodlné řešení například internetového obchodu firmy, který je přímo napojen na informační systém a tak umožňuje automatickou změnu zboží při editaci v QI (viz obrázek 4). Velkou výhodou je přímé napojení internetového obchodu na stav skladu.



Obrázek 4: QI web (QI, 2012e)

4.1.7 Specializované moduly

Poslední kategorií jsou tzv. *Specializované moduly*, které poskytují nadstandardní podporu specializovaných činností firmy. Jsou to moduly *správa prostor*, *vodárenství*, *servis a údržba*, *doprava a expedice* a *pokladní prodej* (QI, 2012f).

4.1.8 Analýza

I přes velký výběr modulů nenabízí informační systém QI, kromě tzv. manažerského přehledu (dashboardu), ve kterém se zobrazuje cash flow a tržby, žádný BI modul.

V samotném informačním systému je ovšem možnost naplnit daty několik předdefinovaných OLAP kostek. Jsou to *Obchodní partneři*, *Zboží a výrobky*, *Účetnictví*, *Poskytnutá plnění*, *Přijatá plnění*, *Výrobní zakázky*, *Reklamace a neshody* a *Vyhodnocení akcí*. OLAP kostka *Vyhodnocení akcí* poskytuje pohled na nákladové a výnosové vyhodnocení akcí. Nejedná se tedy o pohled z hlediska akcí zaměstnanců. Výslednou OLAP kostku je následně možné exportovat do souboru s příponou *cub*. Společnost *DC Concept* možnost vytváření OLAP kostek na svém portálu neprezentuje.

4.2 Elasticita systému

Informační systém QI, jak uvádí na svých internetových stránkách, je první elastický informační systém (qig). QI je schopno přizpůsobit se potřebám zákazníka a splnit tak všechny jeho požadavky od stylistických úprav formulářů až po tvorbu nových databázových struktur. Tato funkcionalita je jistě u zákazníků velice žádána. Firmy nerady upravují své zaběhlé standardy podle vzoru informačního systému.

Ovšem tato funkcionalita nese sebou i určitá rizika. Nadměrné přizpůsobování informačního systému pro každou firmu zvláště přináší do dlouhodobého vývoje systému problémy. Počet zakázkových úprav se tímto stále navyšuje a stává se, že po upgradu vyšší verze QI přestanou některé zakázkové úpravy fungovat. Bylo by tedy zapotřebí automatizovaných testů, které by zajistily otestování funkčnosti nové verze. To ovšem prodejci informačního systému QI nenabízí. Velký problém také nastává, pokud odejde z firmy implementátora zaměstnanec, který má daný informační systém na starosti. Zde dochází se zakázkovými makry k ještě větším problémům.

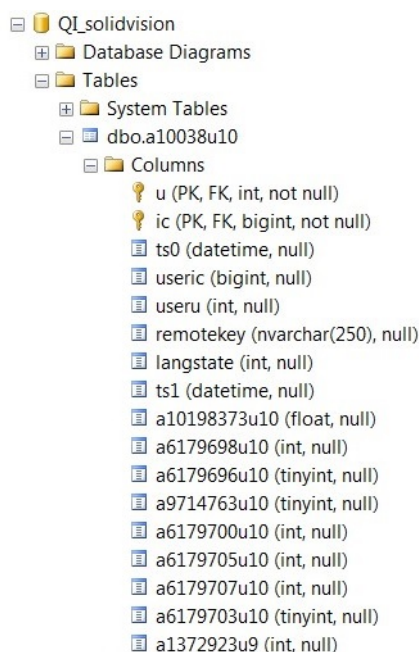
Zákazník má také možnost zasáhnout do svého informačního systému v případě, že má zakoupený tzv. *QI Builder*, který umožňuje vývoj datového modelu (datové třídy, objektové a relační vazby, datové řezy), vývoj programových funkcí (formuláře, tiskové výstupy, makra) a modifikaci vzhledu programových funkcí (formuláře a tiskové výstupy). Rozsah zásahu je omezen schopnostmi programátora a licencí. Licence zaručuje, aby nezasahoval do funkcí *DC Concept* popřípadě prodejce.

Nicméně firma *DC Concept* se snaží zakomponovat již do základní části co možná největší pokrytí běžných požadavků na informační systém, aby k modifikacím u zákazníka docházelo co nejméně.

4.3 Struktura databáze

Důležitou součástí informačního systému je ukládání dat. *DC Concept* zvolil pro ukládání dat kombinaci relační a objektové databáze. Vše, co QI ukládá do databáze, ukládá pod jednoznačným identifikátorem. Tento identifikátor je složen ze dvou částí (IC a U). Tímto identifikátorem jsou pojmenovány i názvy tabulek a atributů (viz obrázek 5). První část (IC) je obyčejný inkrementální generátor v aplikační vrstvě QI, který se navyšuje v případě jakéhokoli zapisování do databáze. Druhá část (U) pak jednoznačně identifikuje databázi, do které je zapisováno. Tímto je zajištěn jednoduchý přenos z různých vývojových databází do jediné ostré databáze

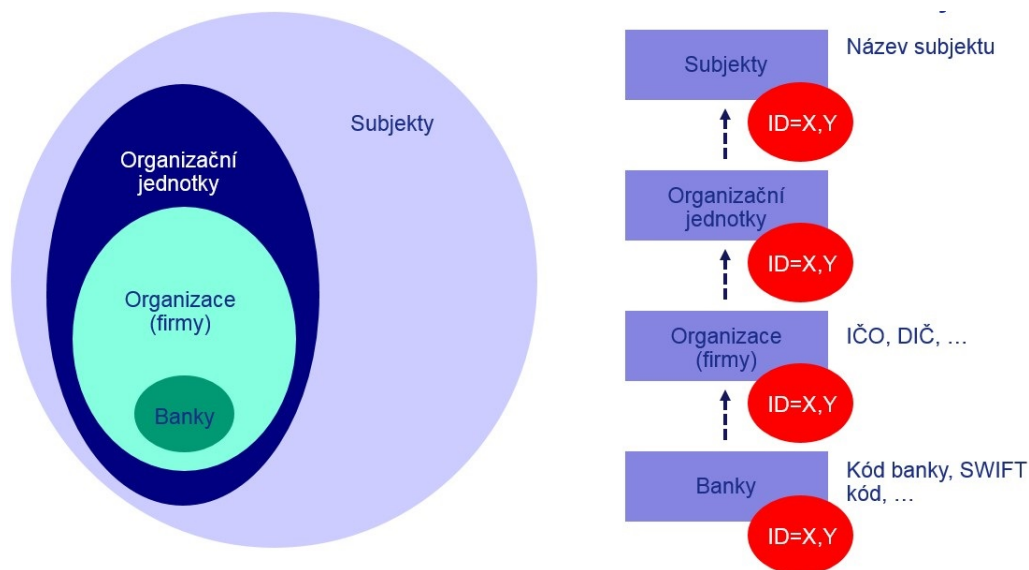
u zákazníka. V případě databáze 0 se jedná o prvotní databázi, ze které vzešlo celé QI. Databáze s číslem 10 je pak samotná databáze firmy *DC Concept*.



Obrázek 5: Ukázka názvů tabulek a atributů (identifikátory IC a U)

Objektové vazby v databázi představují dědičnost mezi třídami. Celá databáze QI dědí ze čtyř tzv. vrcholových tabulek (*subjekty*, *statky*, *doklady* a *akce*)⁵. Jak je vidět na obrázku 6 z vrcholové tabulky *subjekty* dědí tabulka *organizační jednotky* z té pak *organizace* a jako poslední je tabulka *banka*. Přítomnost objektu je dána přítomností záznamu se shodným id (IC) v dané třídě. Každá tabulka obsahuje atributy, které obsahuje navíc oproti tabulce předchůdce. Další větví, která dědí z vrcholové tabulky *subjekty*, jsou například *osoby*. V tabulce *subjekty* budou tedy všechny záznamy (objekty) *Organizačních jednotek*, *Organizací*, *Bank* a *Osob*, ale u těchto záznamů budou atributy příslušící dané tabulce. Ostatní atributy musíme hledat v tabulkách potomků.

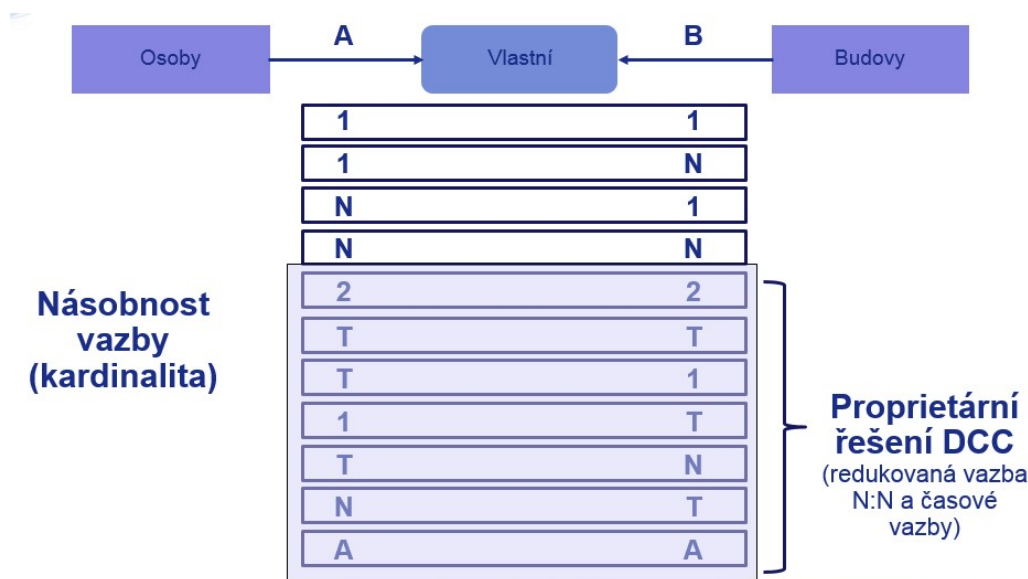
⁵Postupem času DC Concept zavedl tzv. „téměř vrcholové tabulky“, které logicky nezapadaly do hierarchie. Jsou to například položky dokladů nebo bankovní účty.



Obrázek 6: Objektové vazby

Druhým typem vazeb v databázi QI jsou vazby relační. Tyto vazby představují vztahy mezi objekty. Vazba je vždy realizována přes tzv. vazební třídu, která seskupuje identifikátory svázaných objektů obou tříd podobně, jako je tomu u relačních databází v případě kardinality typu M:N.

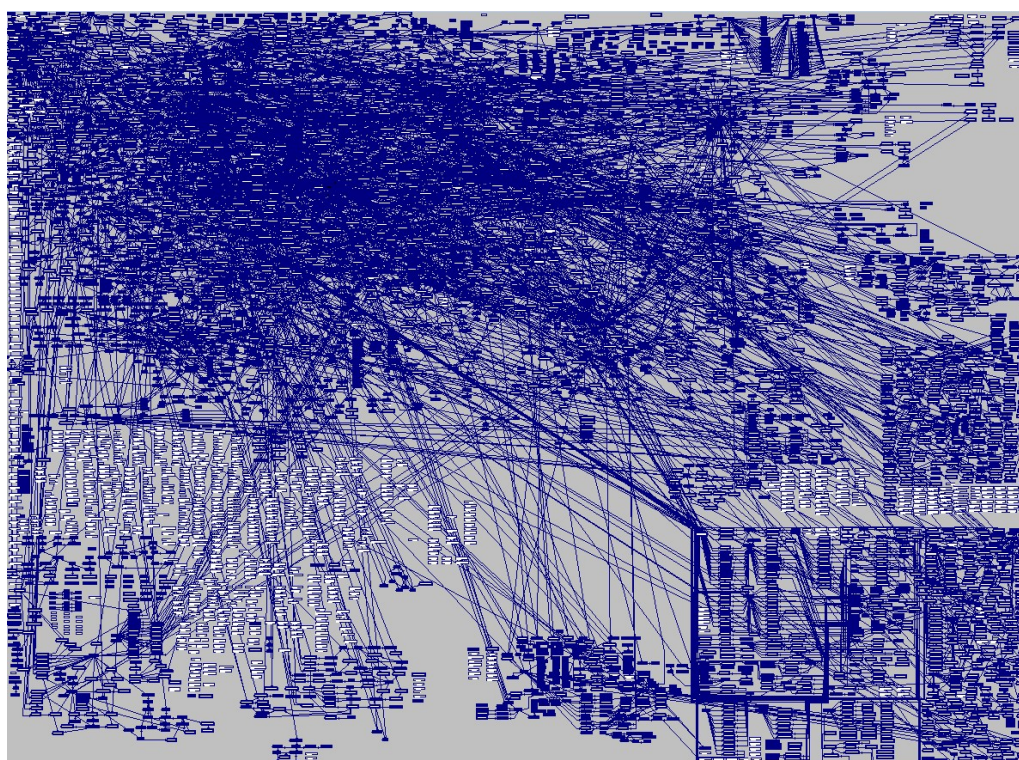
Kromě běžné kardinality vazeb QI zavádí do své databáze také vlastní typy. Jsou to vazby časové a redukovaná vazba M:N. Redukovaná vazba 2:2 zajišťuje, že dva stejné objekty nebudou propojeny více než jednou vazbou. Tato vazba je využita například u informací k dokladu. Daná informace se podle informačního systému QI nemůže vztahovat k více než k jednomu dokladu. Druhým typem jsou časové vazby, které přidávají do běžné kardinality časovou složku. Například vazba T:1 určuje vztah, ve kterém objekt jedné třídy může být v relaci s právě jedním objektem v daném časovém okamžiku. Historicky může být v relaci s více objekty. Tato vazba se například vyskytuje u vazební třídy časově platný *doklad ke statku*, kde statek může mít vždy právě jeden platný doklad v daném okamžiku. Je to tedy způsob řešení historie, protože tyto záznamy vždy doprovází časová značka.



Obrázek 7: Kardinalita

Databáze informačního systému QI má díky tomuto návrhu jednu nespornou výhodu. Touto výhodou je úspora času při vytváření zákaznických úprav databáze. V případě, že zákazník chce přidat nějakou funkcionalitu související s ukládáním nových hodnot do databáze, stačí implementátorovi QI pouze najít vhodné místo v hierarchii tabulek a nově vytvořenou tabulku tam „vložit“. Tato tabulka pak automaticky zdědí atributy a vazby nadřazených tříd. Například v případě, že zákazník bude chtít přidat do QI správu firemní knihovny, tak implementátor vytvoří novou tabulku publikace pod tabulku *evidované statky*, která dědí z vrcholové tabulky *statky*. Z tabulky *evidované statky* zdědí *evidenční číslo* a z vrcholové tabulky *statky* zdědí *název* a *popis* statku. Nemusíme je tedy znovu definovat. Dále se také nemusí opět vytvářet vazby, které již nadřazené tabulky obsahují, jako například vazby *vlastní* a *užívá*, které reprezentují vazby mezi *statky* a *subjekty*. Všechny tyto zděděné vlastnosti můžeme začít v QI ihned používat. Implementátorovi tedy stačí vytvořit pouze formulář, který bude sloužit k evidenci knih. Tímto si QI zajišťuje svoji elasticitu a rychlý vývoj.

Dnes má databáze informačního systému QI přes pět a půl tisíce tabulek a stále narůstá. Na obrázku 8 je znázorněný ERD diagram z informačního systému. V horní polovině obrázku je možné si všimnout přítomnosti zmíněných vrcholových tabulek (tmavě modré shluky čar). Tyto tabulky mají největší počty relací.



Obrázek 8: ERD diagram Informačního systému QI

5 Návrh řešení

Návrh řešení bude tvořen v souladu s pravidly pro tvorbu datových skladů a OLAP kostek. Návrh bude zkonzultován s firemním správcem serveru, aby nedošlo k omezení běžného provozu firmy. Následný návrh reportu bude konzultován s ředitelem a manažerem firmy a upravován dle jejich představ pro docílení co největší vypočítatelné schopnosti informací. Výsledný návrh bude poté sloužit, jako podklad pro realizaci.

5.1 Požadavky

Základním požadavkem pro toto BI řešení je zobrazení požadovaných informací vrcholovému managementu firmy. Výsledné reporty by měly manažerovi umožnit získat přehled nad akcemi obchodníků firmy a jejich stav v plnění obchodního plánu. Každý obchodník ve firmě má stanovený roční plán, který je součástí celkového plánu firmy. Plán prodeje je pro management velice důležitý ukazatel, protože prodej licencí je pro firmu nejen hlavním příjmem, ale také na základě jeho plnění jsou udělovány společností *Dassault Systemes* certifikace oficiálního prodejce softwaru *SolidWorks*. K uzavření úspěšného obchodu obchodníka je zapotřebí vykonat určité akce, jsou stanoveny firemními procesy (jsou to například telefonáty, e-maily, jednání, prezentace, atd.). Časové náklady, a tedy i finanční, na jednotlivé druhy akcí jsou stanoveny vedením firmy. Management by rád sledoval dva ukazatele. Efektivitu obchodníka spočívající v co nejmenším počtu jednotlivých akcí nezbytných k uzavření obchodu a tedy i prodeje licencí a stav obchodníka ve smyslu plnění obchodního plánu.

Managementu firmy by se tedy měly zobrazovat průměrné počty akcí obchodníka, které vedou k uzavření jednoho obchodu. Dále by měl být zobrazen výše uvedený stav obchodníka a odhad potřebného počtu akcí získaného z počtu chybějících licencí a průměrů akcí. Manažer na základě těchto informací následně zjistí, jak je v průběhu roku plněn plán obchodníka a zdali bude schopen tento plán naplnit na základě vlastní průměrné aktivity.

Mimo tohoto základního požadavku byly s managementem dohodnuty další požadavky, které by pomohly manažerovi získat celkový přehled nad zaměstnanci z pohledu akcí. Aplikace by tedy měla být schopna zobrazovat následující informace:

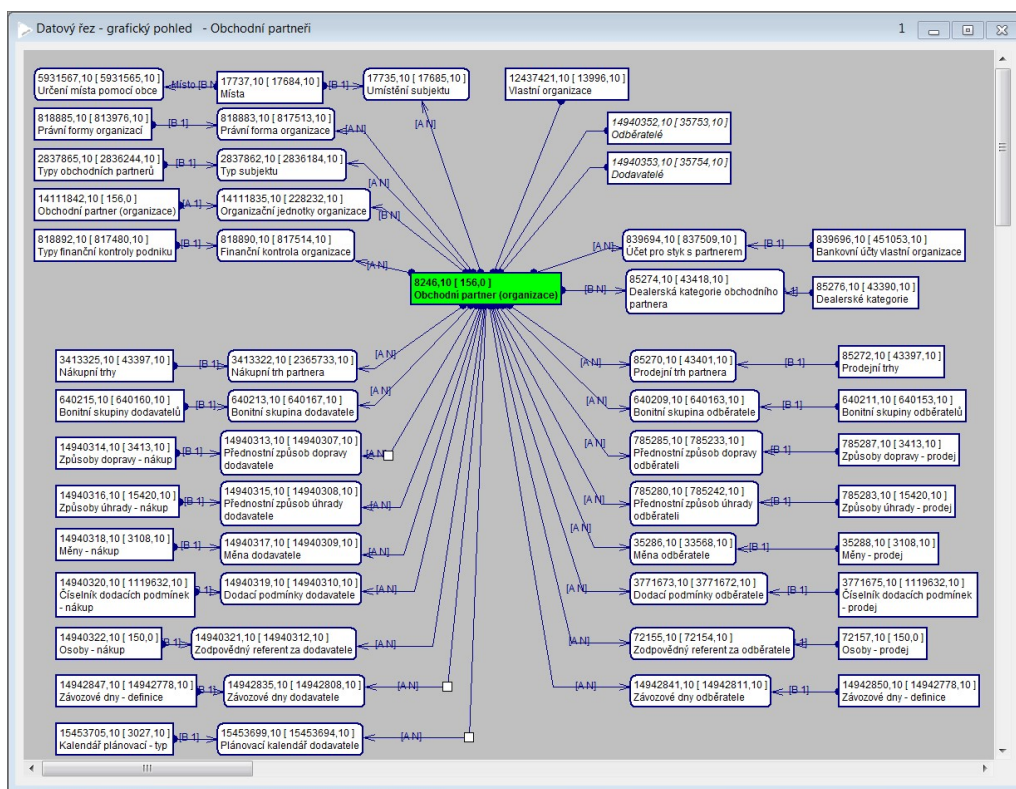
- Průměr potřebných akcí obchodníka na jednu prodanou licenci
- Plnění prodejního plánu obchodníka
- Průměrný denní počet akcí zaměstnance
- Celkový počet akcí zaměstnance
- Počet akcí realizovaných pro obchodního partnera
- Rozmístění obchodních partnerů po ČR

Informace by měly být interpretovány v různých časových řezech, aby mohl manažer jednoduše zjistit například vývoj aktivity zaměstnance, popřípadě porovnat hodnoty zaměstnance s ostatními. Informace by mělo být možné filtrovat, aby měl manažer možnost do jisté míry modifikovat výsledky.

Vzhledem k většímu množství informací, by měla výsledná BI aplikace mít podobu manažerského dashboardu, aby měl manažer všechny informace na jednom místě.

5.2 Nalezení potřebných dat

K nalezení množiny potřebných dat bude nutné využít informační systém QI. Bez informačního systému by nebylo možné se ve zdrojových datech vyznat, protože jak již bylo zmíněno názvy tabulek a atributů v databázi QI jsou číselné identifikátory. Informační systém proto umožňuje zobrazení tabulek použitých v datovém řezu. To je vhodné k nalezení souvisejících tabulek potřebných k získání celkového pohledu na data. Na obrázku 9 je pohled na datový řez obchodních partnerů, tedy tabulek, které nesou atributy obchodního partnera v QI popřípadě relací s nimi.



Obrázek 9: Datový řez obchodních partnerů v QI

5.3 Potřebné datové řezy QI

Z výše uvedených požadavků na systém vyplývá, že k realizaci aplikace budou potřeba data z různých datových řezů. Jedním z nich je výše uvedený datový řez obchodních partnerů pro získání společností, se kterou firma *SolidVision* spolupracuje. Pro dimenzi obchodních partnerů bude zapotřebí pouze tabulka *subjekty a místa*. Dále bude zapotřebí datový řez *osob* pro získání informací o uživateli systému. Pro získání zaměstnanců budou nutné tabulky *subjekt, obchodní partner, hmotné a nehmotné statky*. Datový řez *osoby* obsahuje i kontakty obchodních partnerů, které jsou pro naše potřeby zbytečné a proto budou tato data omezena na zaměstnance *SolidVision*. K zaměstnancům bude nutné připojit tabulky z datového řezu *objednávky přijaté* ke zjištění licencí obchodníka. Tabulky z tohoto řezu budou *položky objednávek přijatých, doklady, množství skladové*.

Pro získání samotných akcí bude zapotřebí tabulek z datového řezu *všechny akce*. Typ akce ovšem v QI není atribut akce. Informace, že je akce daného typu určuje příslušnost dané akce v podřízené tabulce nadřazené tabulky akce. Akce bude tedy nutné vzít z tabulek zvlášť, přidat název typu a následně opět spojit. Jedná se o tabulky *obchodní případ, interní jednání, jednání s partnery, úkoly, marketingové akce, vzdělávací akce, telefonní hovory, projekty, workflow a servisní zásah*. Mimo tyto tabulky bude zapotřebí i samotná tabulka *akce* pro zjištění dalších atributů *druh akce, subjekt a zadavatel*. Akce se budou omezovat datem vzniku po roce 2012, protože od tohoto roku byl informační systém zaveden do provozu firmy. Před tímto rokem se nacházel ve stavu implementace, kdy byly akce zakládány ojedinele a tyto akce by tvořily chyby ve výsledných hodnotách (zejména průměrných denních aktivitách zaměstnance).

Výše zmíněný problém se promítá do druhu akce. Druh akce určuje vazba do tabulky *druhy akcí*. Nicméně zde opět vyčnívá objektová stránka databáze QI. Aby bylo možné určit, které druhy patří danému typu akce, bude nutné načíst podřízené tabulky obecné tabulky *druhy akcí*. Názvy těchto tabulek jsou *druhy interního jednání, druhy obchodních případů*, atd. (tabulek je stejný počet, jako typů akce).

Dalším problematickým údajem je stav akce. Tento atribut je v QI ukládán, jako výčtový typ, což znamená, že hodnoty, které může tento atribut nabývat, jsou uloženy v tabulce *atributy tříd*, jako záznam, který obsahuje atribut hodnoty. V tomto atributu jsou povolené hodnoty oddělené čárkami. Tímto si QI ukládá hodnoty do takového primitivního pole. U samotné akce QI ukládá pouze číselné hodnoty, které představují pořadí údaje v poli.

K získání časových údajů je možné také využít informační systém QI. Přesněji se jedná o funkci *generování dnů pro QI organizační kalendář*. Funkce vygeneruje dny do uvedeného data a automaticky vyplní všechny potřebné údaje. Rozsah časové dimenze bude určovat počáteční datum nasazení systému do aktuálního roku navýšeného o deset let, aby byla zajištěna přítomnost údajů pro akce, které se plánují dlouhodobě.

5.4 Zdroje dat mimo QI

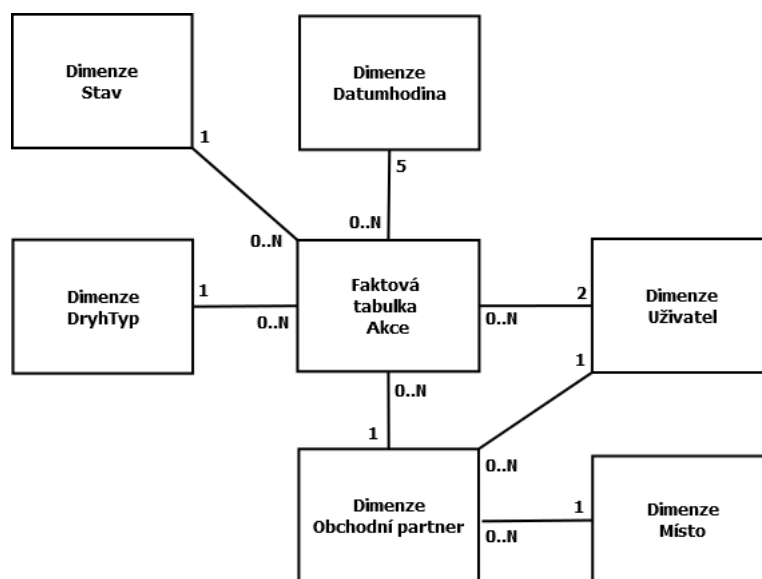
Jako externí zdroj dat bude nutné použít informace o místech České republiky z národního statistického úřadu. Informační systém QI neobsahuje žádné jiné údaje o místech než názvy obcí, ve kterých se nachází obchodní partner. Tento atribut je ukládán jako textový řetězec. Pro výslednou BI aplikaci bude zapotřebí mít dodatečné informace o těchto obcích, jako jsou například kraj a region.

Druhým zdrojem bude *xls* soubor obsahující plány prodeje licencí jednotlivých obchodníků. Tyto informace vytváří vedoucí obchodník společně s vedením firmy každý rok.

5.5 Návrh databáze datového skladu

Návrh databáze bude vycházet zejména z výše zmíněných požadavků. V centru datového skladu se bude nacházet jediná faktová tabulka obsahující informace o akcích. Samotné akce neobsahují žádné užitečné počítatelné atributy, a proto budou ve faktové tabulce pouze klíče do okolních dimenzí. Jediné, co bude důležité, je počet akcí.

Kolem faktové tabulky budou rozprostřeny tabulky dimenzí. Tyto tabulky budou s faktovou tabulkou v relaci. K zobrazení akcí zaměstnanců bude důležité vytvořit dimenzi uživatelů informačního systému. Do dimenze uživatelů budou přidány i stavy obchodníků (počty licencí). Dále se musí vytvořit dimenze obchodních partnerů pro možnost zjištění, které akce byly realizovány pro daného obchodního partnera. Na tuto dimenzi bude napojena dimenze míst, která nám zajistí informace o rozmístění obchodních partnerů. Dalšími důležitými dimenzemi budou dimenze *stav akce*, *druh akce* a *typ akce*. Dimenze *stav akce* bude postavena samostatně z důvodů nutnosti ručního naplnění, který byl popsán v kapitole potřebné datové řezy QI. Dimenze *druh akce* a *typ akce* budou spojeny do jedné, protože je to výhodné pro rychlost aplikace a lepší pochopení koncovým uživatelem. Jak již bylo zmíněno, nedílnou součástí každého datového skladu je dimenze časová. Tato dimenze zajistí prohlížení údajů pro různé časové úseky. Vzhledem k požadavkům bude plně dostačovat granularita časové dimenze v řádů hodin. Přesný čas akce se tedy bude zaokrouhlovat dolů na celé hodiny.



Obrázek 10: Návrh datového skladu

Toto schéma není typu hvězda, protože se zde nachází propojení mezi dimenzi-onálními tabulkami. Jedná se tedy o tzv. *Snow Flake* schéma, které připomíná tvar sněhové vločky.

5.6 Atributy tabulek

Všechny tabulky budou obsahovat automaticky generovaný primární klíč tzv. *Business Key*. Tento údaj slouží jako jednoznačný identifikátor záznamu tabulky pro BI aplikace.

5.6.1 Dimenze Místo

Dimenze místo bude evidovat *název obce, statut, okres, kraj a region*. Spolu s těmito atributy bude také ukládáno identifikační číslo obce. Atribut statut určuje tzv. postavení obce. Atribut může podle *Českého statistického úřadu* nabývat hodnot *hlavní město, město, obec, statutární město, městys a vojenský újezd* (Český statistický úřad, 2014).

5.6.2 Dimenze DatumHodina

U časové dimenze budou evidovány atributy *rok, pololetí, čtvrtletí, název měsíce, číslo měsíce, číslo týdne v roce, číslo dne v roce, číslo dne v měsíci, název dne, číslo dne v týdnu, hodina, datum a druh dne* pro určení dne, jestli je pracovní. Číselné atributy, jako je například *číslo dne v týdnu*, je nutné uchovávat pro řazení dle logické posloupnosti. Posledním atributem v tabulce bude atribut *datumhodina*, který bude sloužit, jako identifikátor záznamu. Jelikož zmíněná funkce QI kalendáře

pro získání časových údajů generuje data pouze na úrovni dní, bude nutné v ETL procesu doplnit data do úrovně hodin.

5.6.3 Dimenze Uživatel

U dimenze uživatelů bude evidováno mimo identifikátorů IC a U uživatele také jeho *jméno* nesoucí jméno a příjmení zaměstnance a *pozice* nesoucí název oddělení ve kterém zaměstnanec pracuje. Jak již bylo uvedeno dříve, do dimenze uživatelů budou vloženy informace o licencích obchodníků. Bude tedy nezbytné evidovat atributy celkového počtu prodaných licencí obchodníka za celou dobu působení ve firmě (*celkový prodej*), letošní počet prodaných licencí obchodníka (*letošní prodej*) a letošní plán prodeje licencí (*letošní plán prodeje*). Dále budou evidovány i dva atributy datového typu (*aktuální od* a *aktuální do*) určující aktuálnost záznamu. Důvod evidence těchto údajů bude popsán v následující kapitole.

5.6.4 Dimenze Obchodní partner

Atributy v dimenzi obchodních partnerů budou *název obchodního partnera* nesoucí celý název firmy, *kód* vytvořený z názvu firmy informačním systémem a identifikátory IC a U, které jednoznačně identifikují obchodního partnera. Dimenze obchodních partnerů bude obsahovat i klíče do dimenzí *místo* pro určení obce, kde má obchodní partner sídlo a *uživatel* pro určení zodpovědného referenta, tedy obchodníka, který má firmu na starosti.

5.6.5 Dimenze DruhTyp

Dimenze *druh-typ* bude evidovat údaje *název typu akce* určující typ akce (například jednání s partnery, telefonní hovor, atd.), *název druhu akce* určující druh daného typu akce (například pro jednání s partnery jsou to porada, prezentace, atd.), *kód druhu* obsahující zjednodušený zápis názvu druhu akce a identifikátory IC, U. Identifikátory IC a U patří v QI *druhu akce*, protože jak již bylo zmíněno, typ akce určuje přítomnost akce v podřízené tabulce.

5.6.6 Dimenze Stav

Nejjednodušší dimenzí bude dimenze *stav*, která bude obsahovat pouze atributy *názvu stavu* a *číslo*, jako identifikátor, který je ukládán v informačním systému QI. Stavů je celkem sedm (*Připravuje se*, *Zahájeno*, *Ukončeno*, *Zrušeno*, *Pozastaveno*, *Převzato* a *Připraveno*).

5.6.7 Faktová tabulka Akce

Jak již bylo zmíněno, faktová tabulka nebude obsahovat žádné počítatelné atributy, nicméně je zapotřebí ukládat klíče do dimenzionálních tabulek. Jedná se tedy o klíče

do dimenze *druh-tyt* určující typ akce, *obchodní partner* určující obchodního partnera pro kterého byla akce realizovaná a atribut *stav* určující stav ve kterém se akce nachází. Do ostatních dimenzí bude nutné ukládat více než jeden klíč. Do dimenze uživatel se budou odkazovat dva klíče a to *realizátor* určující osobu, která akci uskuteční a *zadavatel* akce, který akci založil v informačním systému QI. Do časové dimenze se bude odkazovat pět atributů, jsou to časy *zaevidování* určující den a hodinu, kdy byla akce zaevidovaná do QI, *zahájení plánované* určující naplánovaný den a hodinu, kdy se bude akce realizovat, *ukončení plánované* definující plánovaný den a hodinu ukončení akce, *zahájení* a *ukončení* určující dny a hodiny, kdy byla akce ve skutečnosti zahájena a ukončena. Všechny tyto časy QI u akcí eviduje. Pro výslednou aplikaci bude použit zejména čas zaevidování, který je plněn automaticky informačním systémem QI. Ostatní časové atributy jsou vyplňovány samotnými uživateli, a proto dochází často k prázdným hodnotám, ale případné využití těchto časů bude ponecháno na rozhodnutí managementu. Mimo klíče do dimenzí budou také evidovány identifikační atributy (IC, U) určující jednoznačnou identifikaci akce, *název* nesoucí název akce evidovaný v QI a *kód akce* nesoucí kód akce tvořený informačním systémem ze zkratky akce, roku vytvoření a pořadovým číslem. Tyto atributy budou evidovány pro případné potřeby dohledání záznamů v informačním systému QI či naopak v datovém skladu.

5.7 Návrh ETL procesu

Každá tabulka datového skladu bude mít vlastní ETL proces. Aby byla zajištěna aktuálnost dat, budou se spouštět ETL procesy automaticky každý den v brzkých ranních hodinách, aby nedocházelo k omezování výkonu SQL serveru, na kterém běží i samotné QI. Přesný čas stanoví IT technik firmy. Protože se některé dimenze nebudou měnit tak často (například časová dimenze), nebude ani nutné spouštět ETL proces u všech dimenzí každý den. Změny v dimenzích by měly být inkrementální, aby nedocházelo k opětovnému vkládání stejných dat.

5.8 Extrakce

Aby bylo možné získat data z informačního systému QI, bude nutné vytvoření komplexnější SQL dotazy do databáze QI. Základem bude vždy tzv. hlavní tabulka datového řezu, která obsahuje položky údajů. Následně se k této tabulce připojí tabulky, které nesou informace o položkách. Výjimkou bude faktová tabulka *akce* a *druh-tyt*, ve které se místo jedné hlavní tabulky datového řezu budou spojovat její podřízené tabulky, aby bylo možné získat informaci o typu (viz problém určení typu akce). Do SQL dotazu se tedy přidá atribut k akci, který ponese název typu. Časy získané z databáze budou již na úrovni SQL zaokrouhleny na hodiny.

5.8.1 Dimenze Místo

Pro extrakci dat pro tuto dimenzi bude připojen také externí soubor, ze kterého budou získány atributy *obec, statut, okres, kraj a region*.

5.8.2 Dimenze DatumHodina

Pro extrakci dat do časové dimenze bude připojen připravený externí soubor, ze kterého budou získány atributy *číslo dne v týdnu, datum, hodina, název dne, druh dne, číslo dne v měsíci, číslo dne v roce, číslo týdne v roce, číslo měsíce v roce, název měsíce, čtvrtletí, pololetí a rok*.

5.8.3 Dimenze Uživatel

Pro získání údajů do dimenze *uživatel* bude nadefinováno více SQL dotazů, aby byly odděleny zdroje uživatelů a licencí. SQL dotaz pro získání uživatelů bude spojovat hlavní tabulku *uživatel systému*, ze kterého budou získány identifikátory IC a U, s tabulkami *subjekt* pro získání *jméno uživatele, příslušnost k organizaci, organizace* pro omezení na organizace *SolidVision* a tabulky *hlavní profese a statky* pro získání atributu *pozice*.

Aby bylo možné získat celkové prodané licence a letošní prodané licence, bude vytvořen SQL dotaz spojující tabulky *položky objednávek přijatých* pro získání identifikátoru IC a U, *položky dokladů* pro získání *název zboží a kód produktu, položky patřící dokladu a doklady* pro získání atributu *vyřizuje a zaevidoval* a tabulku *množství skladové* pro získání *množství*. Kód zboží je omezen na licencované softwary, které se zahrnují do obchodních plánů.

Pro získání letošních prodejů licencí obchodníka bude použit stejný SQL dotaz s omezující podmínkou *zaevidování dokladu* na aktuální rok. Pro získání obchodních plánů bude připojen externí soubor od vedení podniku evidující plány aktuálního roku pro každého obchodníka, ze kterého budou získány atributy *uživatel a plán*.

5.8.4 Dimenze Obchodní partner

Z hlavní tabulky *obchodní partner* datového řezu *obchodní partneři* budou získány identifikátory IC a U. Dále budou napojeny tabulky *subjekty* pro získání atributu *název obchodního partnera a referent, umístění subjektu a místa* pro získání atributu *obec, zodpovědný referent a subjekt referent* pro získání atributu *referent*.

5.8.5 Dimenze DruhTyp

Pro získání údajů do této dimenze bude nadefinováno celkem devět SQL dotazů pro každý typ akce zvlášť. Každá z SQL dotazů bude získávat údaje z tabulky stejnojmenného názvu jako je *typ akce (obchodní případ, interní jednání, jednání s partnery, úkoly, marketingové akce, vzdělávací akce, servisní zásah, projekty a workflow*. Tabulky budou vždy spojeny s nadřazenou tabulkou *druh akce*, ze které budou získány

identifikátory IC a U a atributy *kód akce* a *druh akce*. Do každého SQL dotazu byl přidán textový atribut, nesoucí název tabulky (tedy typu akce)

5.8.6 Dimenze Stav

Pro získání údajů do dimenze *stav* nebude zapotřebí SQL dotaz, postačí pouze připojení námi uměle vytvořené tabulky *stav*, ve které bude nutné vypsát všechny hodnoty stavů, kterých může akce nabývat.

5.8.7 Faktová tabulka Akce

SQL dotaz faktové tabulky *akce* bude složen z podřízených tabulce vrcholové tabulky *akce* datového řezu *všechny akce*. Z těchto tabulek budou získány identifikační atributy IC a U a uměle přidán textový atribut nesoucí název typu akce. Výsledná množina dat bude spojena s vrcholovou tabulkou *akce* pro získání atributů *kód akce*, *název akce*, *stav akce*, *datum zaevidování*, *plánovaný čas zahájení*, *plánovaný čas ukončení*, *čas zahájení* a *čas ukončení*. Následně budou připojeny tabulky *druh akce*, *akce se týká subjektu*, *subjekt zodpovědný za akci* a *zadavatel* pro připojení identifikátorů IC a U atributů *druh akce*, *subjekt*, *realizátor* a *zadavatel*. SQL dotaz bude obsahovat podmínku, aby všechny datumové atributy byly většího roku než 2012. Akce faktové tabulky bude nutno spojit také s daty o emailu, které QI nepovažuje za akce, ale za informaci.

V SQL dotazu emailu budou spojeny tabulky *elektronické zprávy odeslané* a *elektronická komunikace mimo IS*, ve kterých budou získány identifikátory IC a U a atribut *předmět*. Dále budou uměle vloženy textové atributy *druh emailu* a *typ akce*. Následně budou připojeny tabulky *záznamy o komunikaci* pro získání atributů *datum odeslání* a *odeslal*, *informace k subjektu*, *organizace* a *subjekt* pro získání atributu *obchodní partner*. Atributy, které informační systém QI u emailu neviduje (*kód*, *stav*, *zahájení plánované*, *ukončení plánované*, *zahájení* a *ukončení*), budou nastaveny na nulové hodnoty.

5.9 Transformace

5.9.1 Dimenze Místo

U této dimenze nebude zapotřebí žádných úprav. Z externího souboru mohou data téct přímo do dimenze.

5.9.2 Dimenze DatumHodina

Transformace v dimenzi *DatumHodina* bude pouze měnit datový typ z načteného externího souboru u textových atributů *název dne*, *druh dne* a *název měsíce*.

5.9.3 Dimenze Uživatel

Jelikož je zde potřeba sloučit data z vícero zdrojů, bude nutné je nejprve seřadit dle atributu, na základě kterého se daná data budou slučovat. Tímto atributem bude *uživatel*. Následně se tedy bude moci spojit uživatelé s celkovým počtem prodaných licencí, a poté tato množina dat s počtem letošních prodaných licencí. U dat z externího souboru plánů prodeje licencí bude nutné změnit datový typ u číselného atributu *plán* a seřadit podle atributu *uživatel*, aby je následně bylo možné spojit s uživateli. Ve výsledné množině dat bude posléze nutné nahradit nulové hodnoty. U atributu *pozice* bude nahrazena nulová hodnota za řetězec "N/A" a atributy *licence celkové*, *licence letošní* a *plán letošní* za číselnou hodnotu nula. U číselných hodnot se následně změní datový typ a na závěr se zajistí propis změny údajů do databáze, vložení údajů nových a v případě změny pozice, zapsání nového údaje s aktuálním datem platnosti.

5.9.4 Dimenze Obchodní partner

U transformace dat obchodních partnerů bude nejprve nutné, jako tomu bylo u faktové tabulky, vložit cizí klíče do atributů *referent* a *obec* a zajistit v případě nenalezení údaje zapsání klíče s nulovými hodnotami. V případě nenalezení atributu *kód obchodního partnera* bude vložen řetězec "N/A". jako poslední bude nutné zajistit opět propisování změn do všech změněných údajů v informačním systému QI a vložení údajů nových.

5.9.5 Dimenze DruhTyp

U této transformace bude nejprve nutné spojit všechny zdroje dat a následně změnit datový typ uměle vloženého textového atributu *typ akce*. Na závěr bude opět zajištěn propis změn údajů a vložení údajů nových.

5.9.6 Dimenze Stav

V této transformaci opět nebude zapotřebí žádných změn. Ze zdrojové tabulky lze rovnou vložit údaje do dimenze *stav*.

5.9.7 Faktová tabulka Akce

Nejvíce transformací proběhne u faktové tabulky. Nejprve bude nutné zajistit vkládání klíčů k atributům. První transformace bude vložení cizího klíče do atributu *realizátor*. V případě, že *realizátor* mezi uživateli nebude nalezen, je nutné vložit do tohoto údaje neznámého uživatele, tedy klíč odkazující se do dimenze *uživatel* s hodnotou "N/A". Následně se údaje, u kterých byl *realizátor* nalezen a u kterých nebyl nalezen, se spojí do jedné množiny. Tento postup je následně aplikován pro atribut *zadavatel*. Dále bude vložen nástroj pro nalezení cizích klíčů u atributů *obchodní partner* a *stav*. U těchto atributů není nutná kontrola nenalezených hodnot,

protože atributy *obchodní partner* a *stav* budou vždy nalezeny. Před nalezením cizích klíčů do atributu *typ akce* bude nutné změnit datový typ uměle vložených textových atributů na jednotný. Následně budou vkládány cizí klíče do atributů *datum zaevídování*, *zahájení*, *ukončení*, *zahájení plánované* a *ukončení plánované*. Na závěr bude nahrazen neznámý kód akce za hodnotu "N/A". Pro druhý vstupní proud emailu bude nejprve nutné vložit cizí klíče do atributu *odesláno*. Následně budou změněny datové typy uměle vložených textových atributů *typ*, *druh* a *kód*. Dále budou opět vkládány cizí klíče do atributů *typ akce*, *obchodní partner* a *realizátor*. V případě, že *realizátor* nebude nalezen, bude nutné vložit cizí klíč s nulovou hodnotou v dimenzi *uživatel*. Množiny akcí a emailů budou spojeny do jedné a na závěr bude zajištěno propisování změn atributů z databáze informačního systému QI pro již vložené akce a vložení akcí nových.

5.10 Nahrávání

Pořadí nahrávání dat má v případě tohoto návrhu datového skladu svoje omezení. Nejprve budou spuštěny procesy všech dimenzí a následně pak proces faktové tabulky. Pořadí procesů dimenzí je podřízeno dimenzí obchodních partnerů, které před jeho spuštěním musí předcházet nahrání dimenzí *Uživatel* a *Místo* kvůli referencím. Nahrání dimenzí bude spuštěno v pořadí dimenzí *Uživatel*, *DatumHodina*, *DruhTyp*, *Místo*, *Stav*, *Obchodní partner* a na závěr faktová tabulka *Akce*. Pro zajištění pořadí bude nutno vytvořit balíček procesů.

5.11 Návrh tvorby OLAP kostky

Pro BI aplikaci bude vytvořena jedna datová kostka, která by měla být umístěna na stejný server jako datový sklad. Datová kostka bude využívat všechny tabulky navrhnutého datového skladu.

Dimenze a faktová tabulka budou propojeny stejným způsobem jako v datovém skladu s tím rozdílem, že z faktové tabulky povede do dané dimenze vazba pro každý klíčový atribut. Tedy například bude pět vazeb do časové dimenze z důvodu uchovávání pěti časových údajů u každé akce. Výsledných dimenzí (řezů) bude tedy dvanáct.

U každé dimenze bude nutností určit potřebné atributy. Dále u časové dimenze budou skryty číselné hodnoty, které jsou určeny pouze k řazení skryty uživatelům a vytvořeny tři druhy hierarchií⁶. První hierarchie bude určovat rozpad rokměsíc, druhá rokden a poslední rok-hodina. Dimenze uživatelů bude obsahovat hierarchii přiřazující zaměstnance do jednotlivého oddělení podle pozice. Hierarchie *druh typu* bude obsažena v dimenzi *druh-typ* a bude zobrazovat druhy jednotlivých typů akcí. Dimenze *obchodní partner* bude obsahovat dvě hierarchie. První bude určovat obchodní partnery referenta a druhá obchodní partnery podle místa.

⁶Hierarchie umožňují propojení atributů z hlediska jejich logické příslušnosti (například čtvrtletí obsahuje určité tři měsíce v roce).

OLAP kostka bude mít předdefinované atributy, které budou automaticky počítány. Bude se jednat o obvyčejné počty údajů v tabulkách *akce* a *obchodní partner*. Pro určení počtu licencí více uživatelů zároveň budou vytvořeny sumy atributů v tabulce uživatel *celkový prodej licencí*, *letošní prodej licencí* a *letošní plán prodeje*. Mimo zmíněných jednoduchých součtů bude OLAP kostka počítat také složitější kalkulace. Prvním z nich bude potřebný *počet akcí na jednu prodanou licenci*, která bude počítána, jako *celkový prodej licencí* dělený počtem akcí s možností řezu podle jednotlivých druhů a typů akce a za jednotlivé obchodníky. Další údaj bude počítat zbývající počet licencí do splnění plánu, který se bude dát filtrovat podle zaměstnání. Bude se jednat o jednoduchý rozdíl prodaných licencí od stanoveného plánu. K tomuto rozdílu bude přidán ještě informativní kalkulace procentuálního naplnění plánu, která se bude počítat podílem letošního prodeje licencí a plánu. Poslední údaj, který bude informovat o licencích, bude odhadovaný počet akcí pro splnění plánu, který bude počítán, jako *počet zbývajících licencí do splnění plánu obchodníka* vynásobený prvním počítaným údajem, tedy *počet akcí na jednu prodanou licenci*. Tento údaj bude, jako takový odhad pro manažera, kolik bude muset obchodník vynaložit aktivit, aby splnil svůj plán. Jelikož manažer vlastní vedením stanovenou časovou náročnost jednotlivých aktivit lze na základě této informace zjistit, zdali je obchodník vůbec schopen svůj plán do konce roku splnit. Posledním počítaným údajem bude denní průměr zaevidovaných akcí. Tento údaj by měl být počítán pro celou časovou hierarchii roků. Kalkulace bude počítána, jako počet zaevidovaných akcí dělený počtem pracovních dní.

Kalkulace procentuálního naplnění plánu obchodníka bude sloužit také pro vytvoření KPI (Key Performance Indicator), který bude sloužit k určení naplnění cíle pomocí rozdělení hodnot do třech kategorií. První kategorie bude v rozmezí 0-49%, druhá kategorie 50-79% a třetí 80-100%. Manažerovi tento indikátor bude sloužit k rychlému přehledu nad obchodníky.

5.12 Návrh způsobu predikce informací

Jako predikovaná hodnota bude zvolen počet obchodních případů. Obchodní případ zakládá obchodník až v případě, že daná firma projeví o tento druh produktu zájem. Následně se k obchodnímu případu tvoří nabídky a objednávky. Tato aktivita je tedy zajímavá z pohledu počtu potenciálních uzavření obchodů. Vzhledem k tomu, že QI je ve firmě nasazeno poměrně krátkou dobu (od roku 2012) je predikce v řádů let bezpředmětná, protože by se hodnoty predikovaly pouze na základě tří hodnot. Proto by měly být počty obchodních případů predikovány v řádech měsíců. Tím bude získáno dostatečné množství vstupních hodnot pro naučení algoritmu.

Predikce bude vytvořena alespoň ve dvou různých systémech, aby bylo možné porovnat její výsledky.

5.13 Návrh interpretace informací

Informace by měly být uživateli srozumitelně podané a na jednom místě. Manažer by měl mít možnost sám si vytvářet a upravovat dotazy pomocí filtrů. Výsledný report by měl být celodenně dostupný.

Informace budou zobrazovány na sedmi dashboardech rozdělených dle druhu zobrazovaných informací. První dashboard bude zobrazovat průměrný denní počet akcí jednoho zaměstnance. Dotaz na OLAP kostku bude zahrnovat hierarchii rok-měsíc z časové dimenze zaevidováno, hierarchii oddělení z dimenze *uživatel*, hierarchii *druh-typ* a počítaný atribut *denní průměr akcí*. Jako omezující podmínky budou vybrány roky zaevidování 2012-2015 a realizátor nebude roven neznámým hodnotám (N/A). Výsledky budou zobrazeny pro jednotlivé roky a měsíce. Pro možnost srovnání hodnot bude dashboard obsahovat tytéž grafy, které zobrazují průměry celé firmy. Manažerovi bude umožněno filtrovat druhy a typy akcí, jména zaměstnanců, oddělení a rok zaevidování. Vzhledem k diskrétním hodnotám budou použity sloupcové grafy.

Další dashboard bude zobrazovat celkové počty akcí zaměstnance. Dotaz bude zahrnovat opět časovou dimenzi realizace, tentokrát ale s hierarchií *rok-hodina* pro možnost zobrazení počtu akcí v průběhu dne. K této hierarchii budou také připojeny číselné atributy určující řazení (*číslo měsíce* a *číslo dne v týdnu*). Opět jako u předchozího dotazu bude zahrnuta dimenze *druh-typ* a *realizátor*. Dimenze *stav* zde bude oproti minulému dotazu navíc. Jako počítaný atribut bude přiřazen *počet akcí*. Podmínky dotazu budou stejné, jako u předchozího dotazu. Počty akcí budou zobrazeny pro jednotlivé roky, měsíce, dny v týdnu a hodiny. Filtry k těmto grafům budou *druh dne*, *druh* a *typ akce*, *hodina*, *jméno*, *oddělení*, *rok*, *stav akce* a *realizátor*. Grafy budou opět zvoleny sloupcové.

Tento dotaz bude využit také při tvorbě dashboardu zobrazující podíl zaměstnance na celkovém počtu akcí firmy. Manažer zde jednoduše nalezne neaktivnějšího zaměstnance. Pro zobrazení informace bude použit výšečový graf, který bude obsahovat legendu zaměstnanců tak, aby byly poměry zastoupení zřetelné. Graf bude doplněn tabulkou s přesnými hodnotami. Možnost filtrace zůstane také stejná.

Nejdůležitější dashboard bude zobrazovat stav prodeje licencí obchodníka. Dotaz pro tento dashboard bude tvořen především z počítaných atributů *celkový prodej*, *letošní prodej*, *letošní plán prodeje*, *zbývá licencí*, *procentuální naplnění plánu*, *počet akcí na jednu prodanou licenci* a *odhadovaný počet akcí pro splnění plánu*. Tyto atributy doplní atribut *jméno* z dimenze *uživatel* a hierarchie *druh-typ*. Omezení dotazu bude opět v časové dimenzi zaevidováno a v dimenzi uživatel omezíme pozici na obchodníka, protože licence prodávají pouze obchodníci. Vzhledem k velkému počtu informací bude tento report tvořen formou tabulek. První tabulka bude obsahovat jednotlivé počty licencí ukončené KPI indikací naplnění plánu pro rychlý přehled manažera. Druhá tabulka pak bude obsahovat *počty akcí na jednu prodanou licenci* a *odhadované počty akcí ke splnění plánu* za jednotlivé druhy a typy akcí. Filtrace bude umožněna manažerovi pouze pro omezení druhů a typů akce a zaměstnance.

Další dashboard bude zobrazovat počty akcí realizovaných pro obchodního partnera. Dotaz pro tento dashboard bude zahrnovat hierarchie dimenze *místo*, *referent* obchodního partnera dimenze *obchodní partner*, *druh-typ*, *rok-měsíc*, *stav*, *realizátor* a číselný atribut *počet akcí*. Omezení bude, již zmíněné roky, pro časovou dimenzi. Filtrace bude umožněna na úrovni druhu a typu akce, obchodního partnera, realizátora stavu akce a roku. Informace budou zobrazeny ve sloupcovém grafu seřazeném podle počtu akcí sestupně, aby manažer měl přehled o firmách, kterým se firma nejvíce věnuje. Graf bude skládaný sloupcový s legendou realizátorů, aby bylo možné zjistit, který realizátor se zasloužil největší mírou o prvenství firmy.

Poslední skupina dvou dashboardů bude zobrazovat rozmístění obchodních partnerů firmy a zaměstnance. Dotaz bude tvořen z hierarchií dimenzí *místa* a *referent* obchodního partnera. Číselný atribut dotazu bude *počet obchodních partnerů*. Tento dotaz nemá žádná omezení. U prvního dashboardu, který bude zobrazovat obchodní partnery daného zaměstnance, bude manažer mít možnost filtrovat *kraj*, *oddělení* a *referent*. Počty obchodních partnerů budou zobrazeny na mapě České republiky s legendou krajů České republiky. Na mapě bude tedy zobrazené, pokrytí regionů daným obchodníkem, a zdali výrazně nezasahuje do regionu kolegy.

U druhého dashboardu, zobrazující počty obchodních partnerů pro celou firmu, budou data zobrazeny stejným způsobem jako u předchozího dashboardu. Pod mapou se bude nacházet sloupcový graf zobrazující stejná data seřazený sestupně podle počtu obchodních partnerů s legendou krajů. Manažer zde naleznе kraje České republiky, ve kterých nemá takové množství obchodních partnerů a na základě této skutečnosti se může vytvořit analýza, zdali daný kraj neobsahuje větší množství potenciálních obchodních partnerů, nežli firma eviduje.

6 Implementace

Implementace byla řešena v souladu s návrhem a metodikou. Prvním krokem dle metodiky bylo nalezení potřebných dat za pomoci informačního systému QI.

6.1 Nalezení potřebných dat

V hlavní nabídce informačního systému byly vždy nalezeny formuláře, které zobrazují hledané informace. Tyto formuláře jsou z databáze 10, tedy databáze firmy *DC Concept* a tak lze tyto formuláře nalézt ve všech zákaznických implementacích. V případě faktové tabulky se jednalo o formulář *Všechny akce*. U každého formuláře je možné si zobrazit jeho technické informace. Umožňuje to zkratka *Ctrl+Shift+Alt+T*. Formulář přehledně zobrazí informace o identifikačních údajích daného formuláře, datového řezu formuláře, filtry dat, hodnoty údajů datového řezu, které jsou ukládány do databáze, systémové hodnoty a časové značky (viz obrázek 11). Pro získání informace, ve které tabulce se údaj nachází je nutné rozkliknout identifikační číslo atributu. Takto byly nalezeny potřebné tabulky datového řezu.

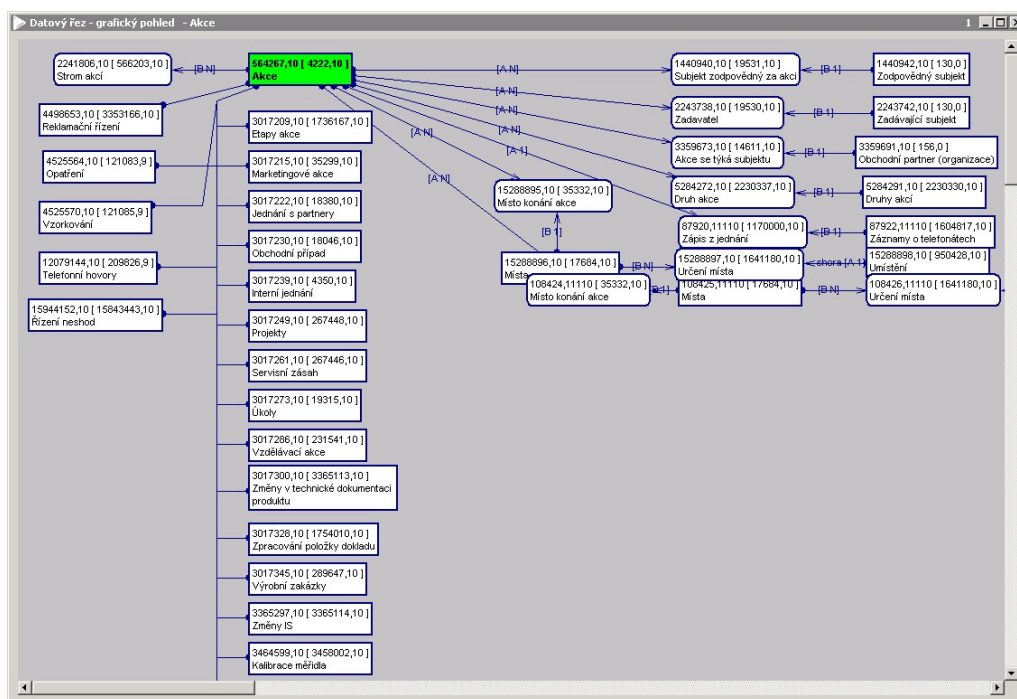
The screenshot shows a software interface with a window titled 'Všechny akce'. It has a menu bar with 'Seznam' and 'Detail'. Below the menu is a table with columns: 'Kód akce', 'Název akce', 'Datum zaevidování', 'Typ akce', and 'Stav akce'. The table contains two rows of data. Below the table is a 'Detail akce' button. A secondary window titled 'Informace o formuláři' is open, showing a 'Form' section with fields like 'PFID' and 'VariantID'. It also has a 'Dataset' section with fields like 'DS', 'DSPF', 'InnerDSDSID', and 'Timestamp'. A 'Settings' section includes a 'Filter' with a complex macro expression. At the bottom, there is a 'User fields' table with columns 'FieldName', 'FieldID', 'AttrID', and 'Value'.

FieldName	FieldID	AttrID	Value
Akce s potřebnými statky	1828319_9	1828318_9	1
Akce s účastníky	1828317_9	1828314_9	1
Čas ukončení	2242422_10	17265_10	
Čas zahájení	2242421_10	17262_10	
Číslo orientační	15288910_10	299757_10	
Číslo popisné	15288904_10	299759_10	
Číslovat dle druhu akce	5284295_10	5279352_10	

Obrázek 11: Formuláře Všechny akce a Informace o formuláři

Pro zobrazení relací a tabulek slouží u informací o datovém řezu odkaz *preview*. Z tohoto náhledu byly získány identifikační údaje tabulek potřebných atributů a tabulek, které je spojují s hlavní tabulkou. Obyčejné databázové tabulky jsou ve sché-

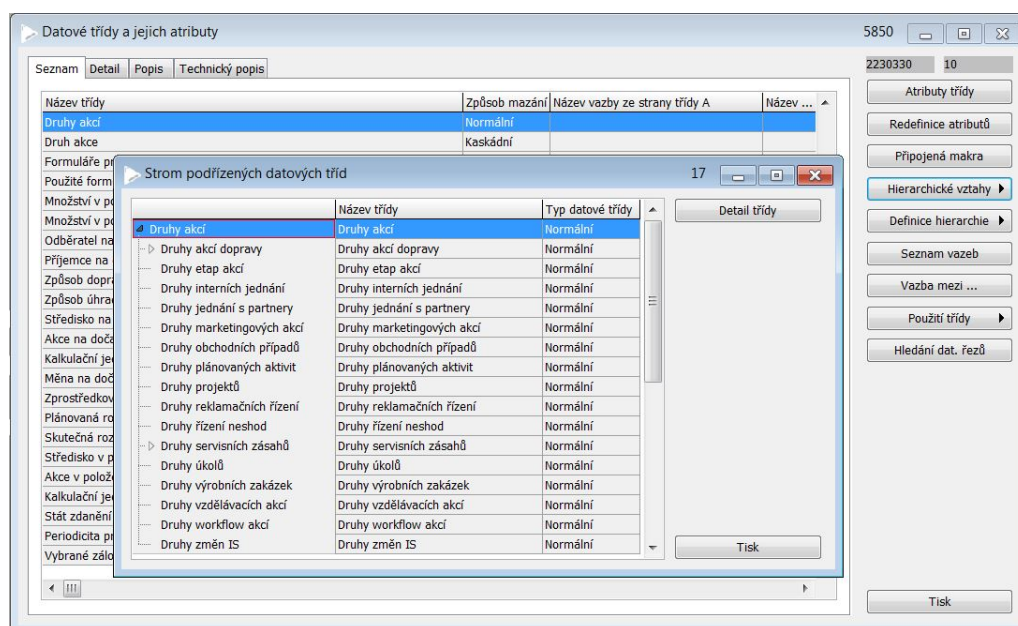
matu ohraničeny modrým obdélníkem s ostrými hranami, hlavní tabulka je navíc zvýrazněna zelenou barvou a vazební tabulky jsou ohraničeny modrým čtvercem se zaoblenými hranami. Text u vazeb určuje, jak již bylo zmíněno kardinalitu a stranu, ze které je tabulka připojena. Vazby bez textu znázorňují objektovou dědičnost, tedy že tabulka *Akce* je nadřazená tabulka všem tabulkám v levé části obrázku 12.



Obrázek 12: Grafický náhled na datový řez.

Tímto způsobem byly získány také potřebné údaje a tabulky pro dimenze *obchodní partner* a *uživatel*, pro které byly využity formuláře *Obchodní partneři* a *Osoby*.

V případě dimenze *druh-typ* byl postup odlišný. Ke zjištění druhů akcí byl použit formulář *Datové třídy* a jejich atributy zobrazující všechny datové tabulky. Zde byla vyhledána tabulka *Druhy akcí*, která určuje druh akce, jak je možné vidět v datovém řezu akcí (viz obrázek 12). Zde byly zobrazeny její podřazené třídy za pomoci formuláře *Strom podřazených datových tříd* zobrazovaného funkcí *Hierarchické vztahy*. Tento formulář poskytl seznam potřebných tabulek (viz obrázek 13).



Obrázek 13: Hierarchie podřízených tabulek druhů akcí.

Pro zjištění stavů akcí bylo zapotřebí zobrazit *Detail atributu*. Tento formulář se zobrazí mimo jiné po kliknutí na identifikátor atributu ve výše zmíněném formuláři Informace o formuláři. Formulář obsahuje veškeré informace o atributu. Pokud je atribut výčtového typu, tak se ve spodní části formuláře nachází výčet hodnot systémový a interpretovaný pro uživatele (viz obrázek 14). Z tohoto pole bylo nutné hodnoty vpsat ručně.

Detail atributu

Název atributu: Stav akce Název třídy: Akce 17240 10

Detail Makra Popis Technický popis

Název atributu Stav akce

Datový typ atributu Výčtový

Volby []

Způsob ochrany Chráněný

Způsob instalace Přenášet vždy

Způsob přenosu Přenášet vždy

Způsob indexace

Velikost znakového atributu

Zobrazovací šířka znakového atributu

Zarovnání atributu

Ořezávání mezer u znakového atributu

Převod na velká písmena u znakového atributu ...

Výčet hodnot - vnitřní uložení
0,6,1,2,5,3,4

Výčet hodnot - uživatelská interpretace
Připravuje se, Připraveno, Zahájeno, Ukončeno, Převzato, Zrušeno, Pozastaveno

Třída

Redefinice atributů

Použití atributu

Atribut značek ▶

Obrázek 14: Formulář detail atributu.

Data míst byla získána z databáze *Národního statistického úřadu* ve formátu *xls*. V tomto formátu jsou také evidovány prodejní plány licencí obchodníků. Oba soubory byly umístěny na server do místa určeného správcem IT.

6.2 Realizace databáze datového skladu

Datový sklad byl tvořen v *Microsoft SQL Server Management Studio*. Na základě struktury zdrojových dat byly zachovány datové typy, aby nedocházelo ke ztrátě informací. Byla vytvořena nová databáze s výchozím nastavením a názvem *QI_BI*. Všem tabulkám byl nastaven business klíč, jako primární klíč tabulky.

6.2.1 Tabulka Místo

Tabulka *dim.Misto* byla vytvořena pro evidenci všech obcí v České republice.

Tabulka 1: Atributy tabulky dimMisto

Atribut	Typ	NULL	Popis
MistaKey	bigint	ne	business klíč
MistaID	bigint	ano	identifikátor ze zdrojového souboru
Obec	nvarchar(60)	ano	název obce
Statut	nvarchar(60)	ne	typ obce (město, obec, atd.)
Okres	nvarchar(60)	ne	okres obce
Kraj	nvarchar(60)	ne	kraj obce
Region	nvarchar(60)	ne	region obce

6.2.2 Tabulka DatumHodina

Tabulka *dimDatumHodina* byla vytvořena za účelem evidence všech hodin v letech 2010–2025.

Tabulka 2: Atributy tabulky dimDatumHodina

Atribut	Typ	NULL	Popis
DatumHodinaKey	bigint	ne	business klíč
DatumHodina	datetime	ano	jednoznačné určení hodiny a dne
Datum	date	ne	datum
Hodina	tinyint	ne	hodina
CisloDneTyden	tinyint	ne	pořadí dne v týdnu
NazevDne	nvarchar(50)	ne	název dne
DruhDne	nvarchar(50)	ne	druh (pracovní, víkend, svátek)
CisloDneMesic	tinyint	ne	číslo dne v měsíci
CisloDneRok	int	ne	číslo dne v roce
CisloTydneRok	tinyint	ne	číslo týdne v roce
CisloMesiceRok	tinyint	ne	číslo měsíce v roce
NazevMesice	nvarchar(50)	ne	název měsíce
Cvrtleti	tinyint	ne	číslo čtvrtletí v roce
Pololeti	tinyint	ne	číslo pololetí v roce
Rok	smallint	ne	roku

6.2.3 Tabulka Uživatel

Tabulka *dimUzivatel* byla vytvořena pro evidenci všech zaměstnanců firmy.

Tabulka 3: Atributy tabulky dimUzivatel

Atribut	Typ	NULL	Popis
UzivatelKey	int	ne	business klíč
UzivatelIC	int	ano	identifikace v systému QI
UzivatelU	int	ano	identifikace v systému QI
Jmeno	nvarchar(120)	ne	jméno a příjmení uživatele
Pozice	nvarchar(250)	ano	pozice zaměstnance
CelkovyProdej	int	ne	celkový počet prodaných licencí obchodníka
LetosniProdej	int	ne	počet prodaných licencí obchodníka pro aktuální rok
LetosniPlanProdeje	int	ne	plánovaný počet prodání licencí obchodníka
AktualniOd	date	ano	datum, kdy byl zaměstnanec přijat na danou pozici
AktualniDo	date	ano	datum ukončení působnosti zaměstnance na dané pozici

6.2.4 Tabulka Obchodní partner

Dále byla vytvořena tabulka *dimObchodniPartner* pro evidenci všech obchodních partnerů.

Tabulka 4: Atributy tabulky dimObchodniPartner

Atribut	Typ	NULL	Popis
ObchodniPartnerKey	bigint	ne	business klíč
ObchodniPartnerIC	bigint	ano	identifikace v systému QI
ObchodniPartnerU	int	ano	identifikace v systému QI
KodOP	nvarchar(50)	ano	textový kód partnera
ObchodniPartner	nvarchar(120)	ne	název partnera
MistaKey	bigint	ano	obec, kde se partner nachází
ReferentKey	int	ano	zaměstnanec, který má partnera na starost

6.2.5 Tabulka DruhTyp

Tabulka *dimDruhTyp* eviduje všechny kombinace druhů a typů akce.

Tabulka 5: Atributy tabulky dimDruhTyp

Atribut	Typ	NULL	Popis
DruhKey	int	ne	business klíč
DuhIC	bigint	ano	identifikace druhu akce v systému QI
DruhU	int	ano	identifikace druhu akce v systému QI
DruhKod	nvarchar(50)	ano	textový kód druhu akce
Druh	nvarchar(100)	ano	název druhu akce
Typ	nvarchar(60)	ano	název typu akce

6.2.6 Tabulka Stav

Nejjednodušší tabulkou je tabulka *dimStav*, která byla vytvořena pro evidenci všech stavů, ve kterých se akce může nacházet.

Tabulka 6: Atributy tabulky dimStav

Atribut	Typ	NULL	Popis
StavKey	int	ne	business klíč
StavID	int	ano	identifikace stavu
Stav	nvarchar(50)	ano	název stavu

6.2.7 Tabulka Akce

Faktová tabulka *factAkce* byla vytvořena za účelem evidování všech akcí.

Tabulka 7: Atributy tabulky factAkce

Atribut	Typ	NULL	Popis
AkceKey	bigint	ne	business klíč
AkceIC	bigint	ne	identifikace v systému QI
AkceU	int	ne	identifikace v systému QI
Kod	nvarchar(50)	ano	smíšený kód akce
Nazev	nvarchar(250)	ano	název akce
StavKey	int	ano	stav akce
ZaevidovanoKey	bigint	ano	datum zaevidování akce
ZahajeniPlanovaneKey	bigint	ano	datum plánovaného zahájení
UkonceniPlanovaneKey	bigint	ano	datum plánovaného ukončení
ZahajeniKey	bigint	ano	datum reálného zahájení
UkonceniKey	bigint	ano	datum reálného ukončení
DruhTypKey	int	ano	druh a typ akce
ObchodniPartnerKey	bigint	ano	obchodní partner pro kterého byla akce realizována
UzivatelRealizatorKey	int	ano	realizátor akce
UzivatelZadavatelKey	int	ano	zadavatel akce

6.2.8 Pomocné tabulky

Pro pomocné účely byly vytvořeny tři tabulky. První tabulkou je tabulka *Hodiny* obsahující čísla od nuly do 23. Tabulka byla vytvořena za účelem kartézského součinu s daty určenými do dimenzionální tabulky *dimDatumHodina*. Jak již bylo zmíněno, informační systém QI poskytl data pouze na úrovni dní.

Tabulka 8: Atributy tabulky Hodiny

Atribut	Typ	NULL	Popis
id	int	ne	identifikátor
hodina	int	ne	číselné označení hodiny (0-23)

Druhou tabulkou je tabulka *Stav*, která sloužila k ručnímu vyplnění stavů akcí z QI pro jejich nahrání do dimenze *dimStav* kvůli způsobu ukládání stavů informačním systémem QI.

Tabulka 9: Atributy tabulky Stav

Atribut	Typ	NULL	Popis
id	int	ne	identifikace stavu v systému QI
Stav	nvarchar(50)	ne	název stavu

Poslední tabulkou je tabulka *Predikce* určená k predikování informací obsahující spojení atributů *rok* a *měsíc*, jakožto jednoznačného identifikátoru údaje a počtu obchodních případů v tomto měsíci. Tato tabulka sloužila k naučení predikčního algoritmu.

Tabulka 10: Atributy tabulky Predikce

Atribut	Typ	NULL	Popis
MesicRok	int	ne	číselný identifikátor vytvořený spojením roku a měsíce
ObchodniPripadPocet	int	ne	počet obchodních případů v měsíci

6.3 Realizace ETL procesu

Pro realizaci ETL procesu byla použita komponenta SSIS (*Integration Services*) nástroje *SQL Server Business Intelligence Development Studio*. V rámci Extrakce byl také využit nástroj *Microsoft SQL Server Management Studio* pro tvorbu SQL dotazů definující zdroje dat. Pro každou tabulku byl vytvořen vlastní balíček (*Package*), který definuje celý ETL proces. V každém balíčku bylo nutné nadefinovat v *Connection Managers* zdrojovou a cílovou databázi. Jako cílová databáze byl připojen datový sklad *QI_BI* pomocí provideru *Native OLE DB SQL Server Native Client 11.0*, kde se nastavuje název serveru, autentikace a název databáze. Zdrojovou databází byla nastavena databáze informačního systému QI stejným způsobem. U tabulek, kde byl zdrojem dat *CSV* soubor (*dimDatumHodina*, *dimMista* a *dimUzivatele*) bylo zapotřebí nadefinovat tento zdroj, jako *Flat File Connection*, kde se uvedla cesta k souboru, znaková sada a oddělovače sloupců a řádků.

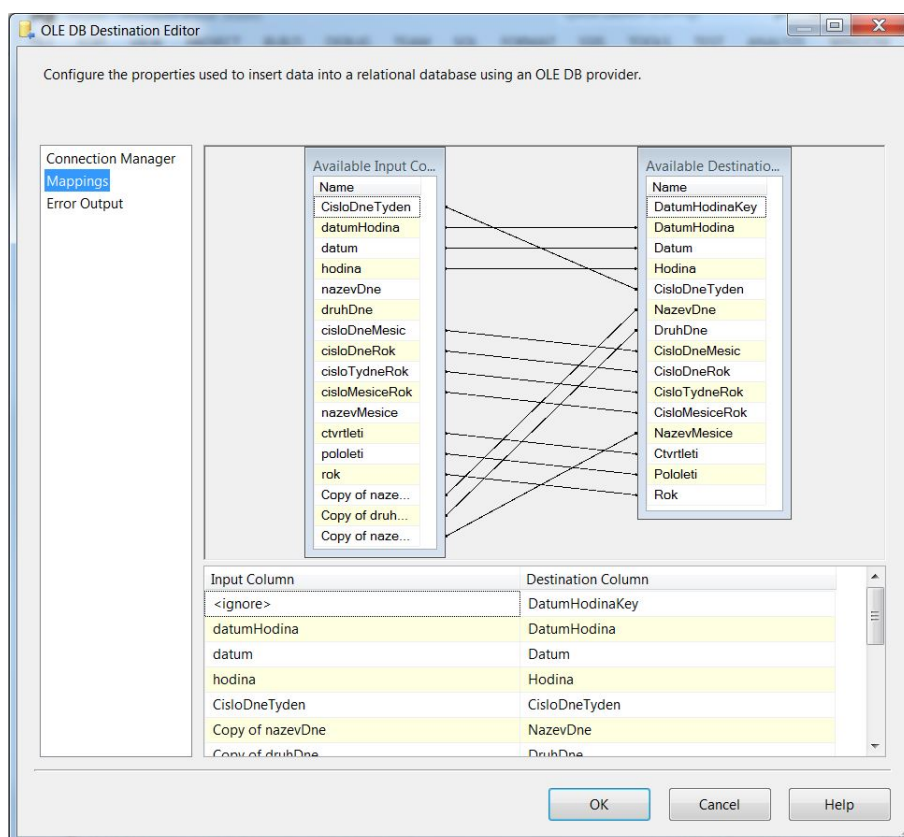
Do každého balíčku tabulek byla vložena komponenta *Data Flow Task*, která umožňuje nadefinovat tok dat ze zdroje až po cíl.

6.3.1 Dimenze Místo

U tabulek *dimMista* byla zdrojová data již v požadované formě, tudíž nebylo zapotřebí žádných změn.

6.3.2 Dimenze DatumHodina

U tabulky *dimDatumHodina* bylo zapotřebí datové konverze textových atributů. Pro datovou konverzi slouží nástroj *Data Converter*, který vytvoří kopii atributu nového typu. Obě tabulky mají nadefinovaný zdroj dat pomocí nástroje *Flat File Source*. Tento nástroj slouží k výběru zdrojových dat z *Flat File* Spojení v *Connection Manager*. Jako poslední nástroj byl do těchto balíčků přidán *OLE DB Destination* pro určení cílové tabulky datového skladu a mapování upravených atributů na atributy cílové tabulky (viz obrázek 15).



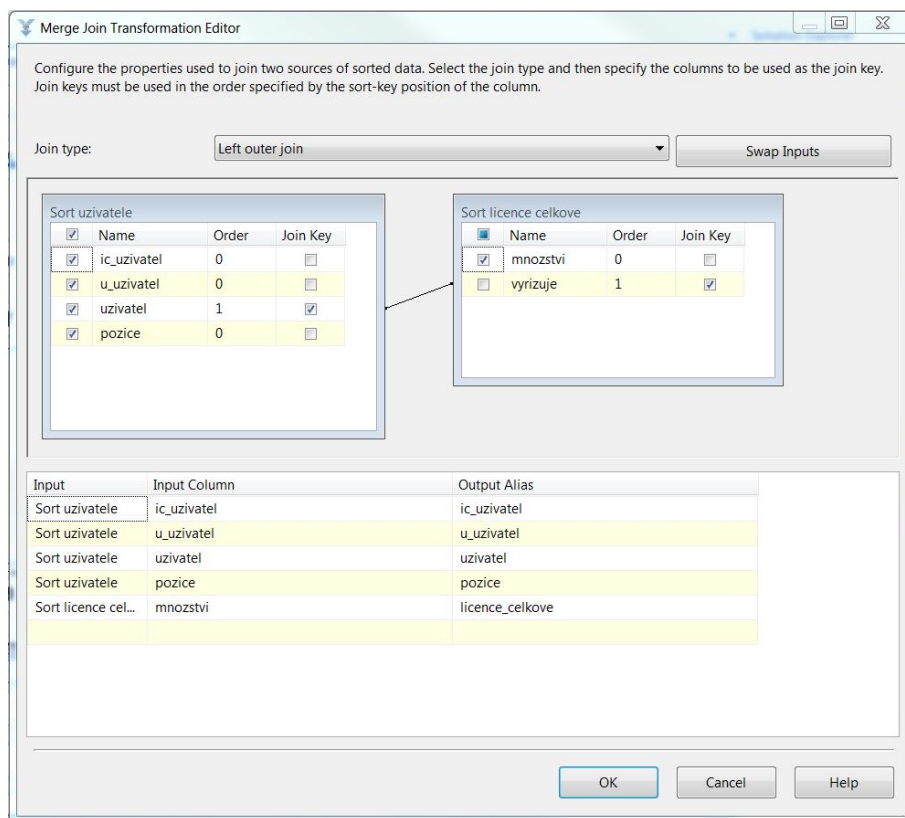
Obrázek 15: Mapování atributů časové dimenze.

Nástroj *OLE DB Source* umožňuje nahrát zdrojová data přes *OLE DB Connection*. Je zde možnost ručně nadefinovat tabulku nebo pohled databáze anebo je možné zdroj dat určit pomocí SQL dotazu. Vzhledem k nutnosti propojení velkého množství tabulek byla zvolena možnost definice zdrojových dat pomocí SQL dotazu.

6.3.3 Dimenze Uživatel

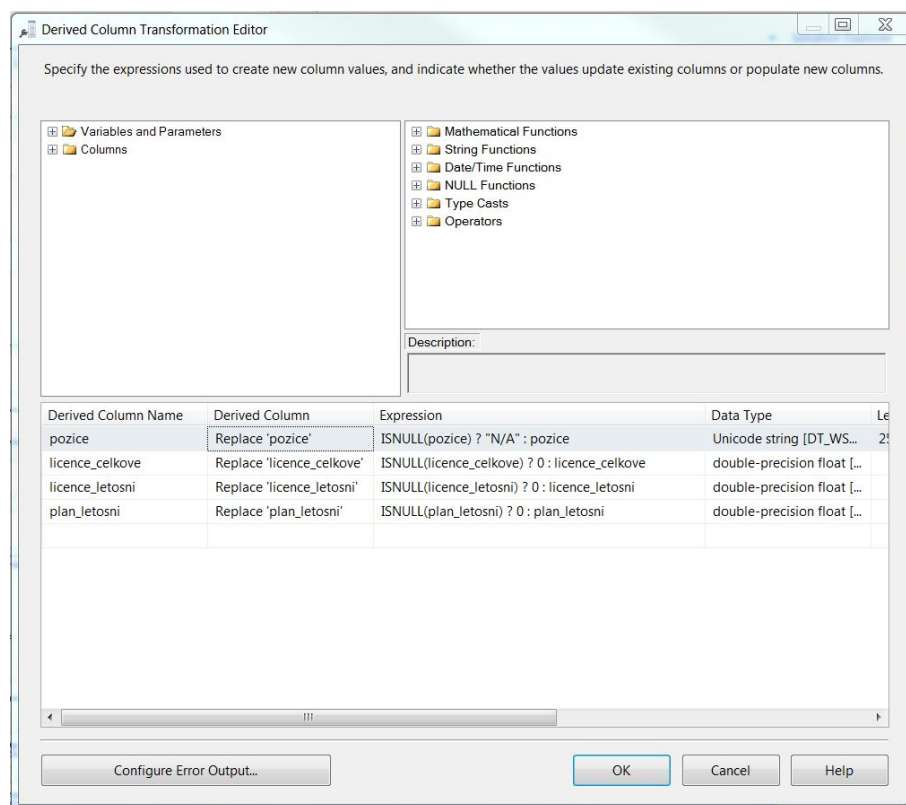
Pro tabulku *dimUzivatel* byly nadefinovány zdroje dat *OLE DB Source* databáze informačního systému QI a *Flat File* soubor plánů prodeje obchodníků. Zdroje dat z QI byly rozděleny do tří proudů (*uživatelé*, *celkový prodej licencí* a *letošní prodej*

licenci). V SQL dotazu Uživatelé byly propojeny tabulky uživatelé systému, subjekt, příslušnost k organizaci, organizace, hlavní profese a statky. Data byla omezena podmínkou kódu obchodního partnera na firmu *SolidVision*. V dotazu, který získává licence obchodníka, byly spojeny tabulky položky objednávek přijatých, položky dokladu, položky patřící dokladu, doklady a množství skladové. Položky jsou filtrovány dle kódu zboží, které se počítají do prodejního plánu. Výsledkem je suma licencí seskupená podle jména uživatele. Stejným způsobem byl získán i SQL dotaz pro letošní prodej licencí. Byla pouze přidána podmínka, ve které rok získaný z data prodeje je roven aktuálnímu roku. Jednotlivé proudy byly spojeny do jednotu toku pomocí nástroje *Merge Join*. Prerekvizita užití tohoto nástroje je, že data musí být seřazena podle atributu na základě kterého se budou data spojovat. Proudly tedy byly seřazeny podle jména uživatele a následně spojeny (viz obrázek 16).



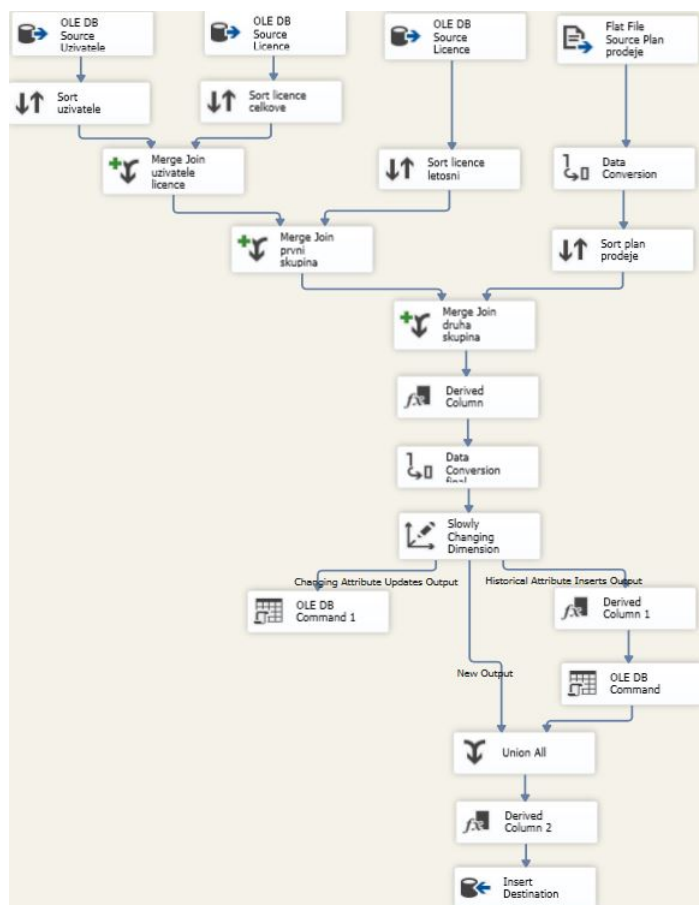
Obrázek 16: Nástroj Merge Join.

Jelikož uživatelé, kteří nejsou obchodníci, neobdrží hodnotu licencí, byl využit nástroj *Derived Column* pro doplnění nul do těchto polí. Zde byly také prázdné hodnoty atributu *pozice* nahrazeny za řetězec „N/A“. Nástroj *Derived Column* umožňuje na základě definované podmínky nahradit hodnotu atributu za jiný. Využívá se především pro nahrazení prázdných hodnot atributů.



Obrázek 17: Nástroj Derived Column.

U číselných hodnot licencí byl změněn datový typ. Pro zajištění vložení nového atributu v případě, že se změní pracovní pozice uživatele, byl využit nástroj *Slowly Changign Dimension*. Tento nástroj se používá pro nadefinování akcí v případě změny dat v databázi. V nástroji se u každého atributu určí, jestli je to business klíč, či není. Při příštím načtení dat se tedy tato data spárují s daty v datovém skladu na základě shodnosti business klíčů a porovná se shodnost zbylých nadefinovaných atributů. U atributů, které bude nástroj porovnávat, se nadefinuje akce, která se provede v případě změny údaje. Je možné nadefinovat tři druhy akcí. První akcí je *Fixed Attribute*. V tomto případě se změna neprojeví v datovém skladu. Druhá akce je *Changign Attribute*, kde v případě změny údaje dojde k jeho nenávratnému přepsání novým údajem nebo zapsáním údaje do jiného atributu. Poslední akce je *Historical Attribute*. Zde je možné zajistit, aby v případě změny údaje byl zapsán do datového skladu celý nový záznam. V tomto případě je nutné mít v dané tabulce atribut indikující aktuálnost záznamu nebo časové atributy určující odkdy do kdy byl daný záznam aktuální. Nástroj rozdělí datový tok na dva až tři proudy (záleží na výběru typu akce). Dva proudy představují druhy akce (*Changing Attributes* a *Historical Attributes*). Poslední proud je určen pro nově vložená data. Po provedení změn jsou proudy *Historical Attributes* a nově vložená data spojeny do jednoho, který vede do databáze (viz obrázek 18).

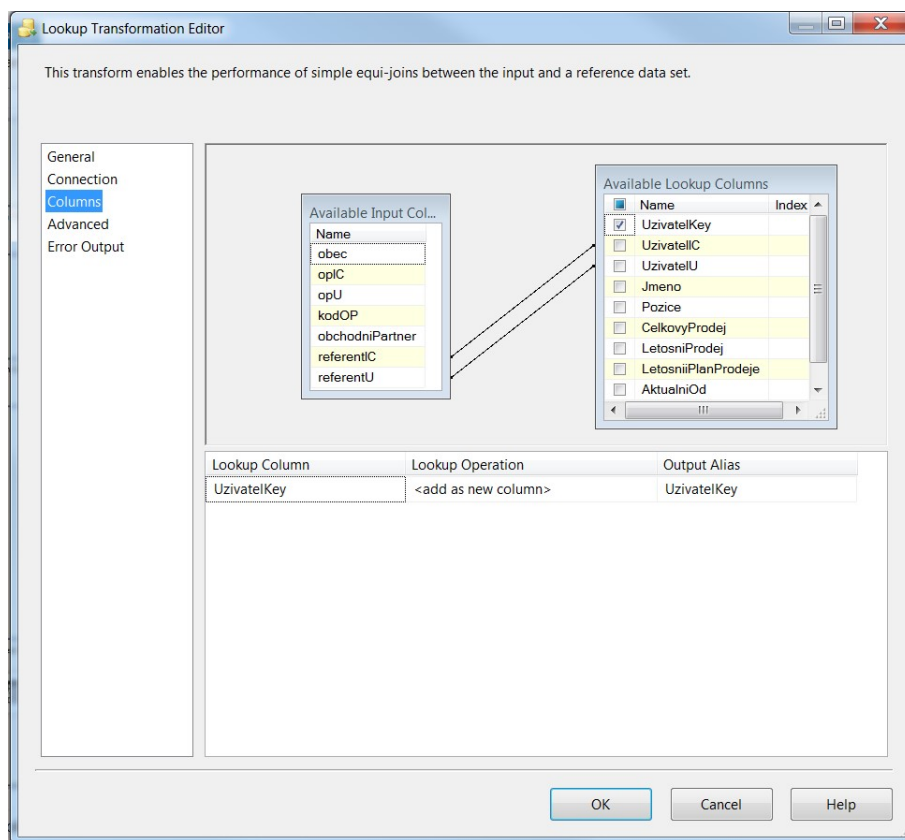


Obrázek 18: Balíček ETL procesu dimUzivatele.

6.3.4 Dimenze Obchodní partner

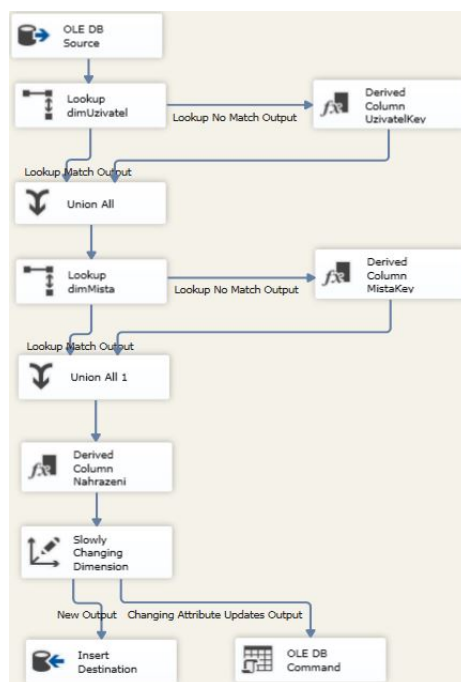
Pro tabulku *dimObchodniPartner* byl definován jediný *OLE DB source*. SQL dotaz tohoto zdroje propojuje hlavní tabulku *obchodni partner* s *tabulkami subjekty, umístění subjektu, místa a zodpovědný referent*. Tabulka *Subjekty* se v SQL dotazu vyskytuje dvakrát. Jednou pro napojení jejích atributů pro obchodní partnery a podruhé pro napojení atributů zodpovědného referenta.

Dále bylo zapotřebí získat business klíče referentů a míst obchodního partnera. Za *OLE DB Source* byl tedy napojen nástroj *LookUp*, který umožňuje získat atributy ostatních tabulek na základě shody klíčových údajů. V tomto nástroji se nejprve nadefinuje tabulka, ze které se bude atribut získávat a následně je nutné propojit atributy, které se mají shodovat a vybrat atribut, který požadujeme. Tento atribut je možné vložit do datového proudu, jako nový či jej nahradit za stávající (viz obrázek 19).



Obrázek 19: Nástroj LookUp.

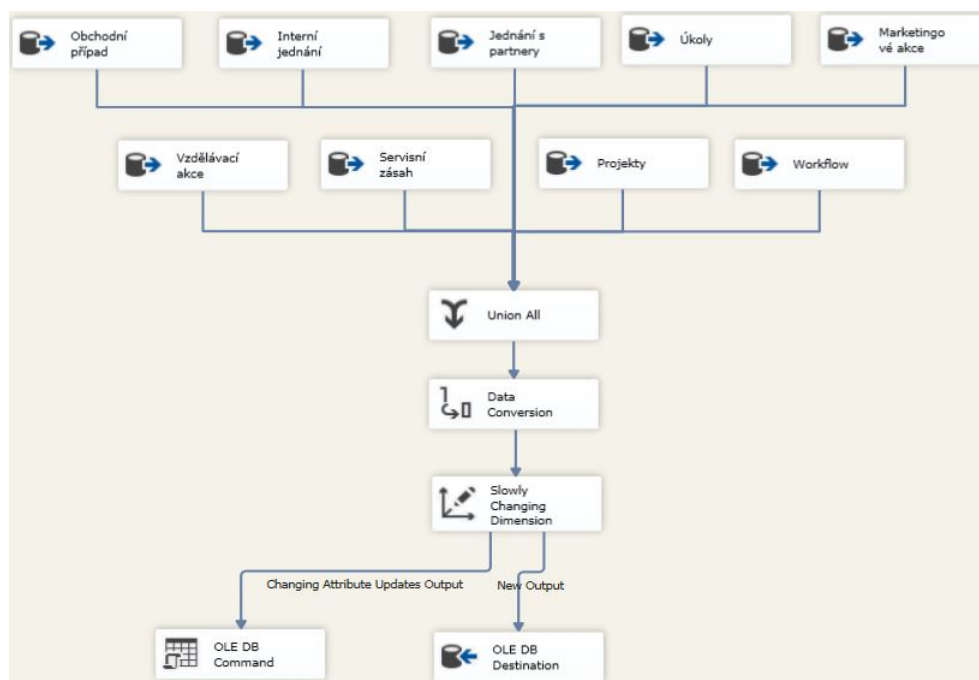
Nástroj *LookUp* umožňuje rozdělit datový tok na dva proudy. Prvním z nich je *Match outputs*, kam se posílají data, u kterých bylo nalezeno spojení. Druhým je *No Match Outputs*, který představuje pravý opak, tedy data, která spojení nenalezla. Za tento nástroj byl do proudu vložen nástroj *Derived Column*, do kterého byl poslán proud *No Match Outputs*. Jedná se o obchodní partnery, kteří nemají zodpovědného referenta, nebo tento referent u firmy již nepracuje. V *Derived Column* tedy nastavíme, aby se v tomto případě do údaje vložil klíč jedna. Údaj s business klíčem jedna bude rezervován v tabulce *dimUzivatel* pro neznámého uživatele. Oba proudy jsou následně spojeny nástrojem *Union All*, který pouze sloučí proudy do jednoho. Stejným způsobem bylo vyřešeno vkládání business klíče tabulky *dimMisto*. Dále bylo nutné nahradit nástrojem *Derived Column* prázdné hodnoty v atributu *Kod* za řetězec „N/A“. Jako poslední nástroj byl vložen *Slowly Changing Dimension* zajišťující změny údajů v databázi. Zde se pouze staré hodnoty nahrazují za nové a proto nástroj *Slowly Changing Dimension* rozdělil datový tok pouze na dva proudy a nikoli na tři, jako tomu bylo u ETL procesu *dimUzivatele*.



Obrázek 20: Balíček ETL procesu dimObchodniPartner.

6.3.5 Dimenze DruhTyp

Pro tabulku *dimDruhTyp* bylo nadefinováno celkem devět nástrojů *OLE DB Source*, pro každý typ akce jeden, aby byl zajištěn přehled nad všemi typy, které se do této dimenze načítají. Každý *OLE DB Source* byl definován SQL dotazem, kde byla spojena tabulka Druh Akce vždy s tabulkou obsahující druhy daného typu akce. V každém SQL dotazu byl vytvořen textový atribut *Typ akce* nesoucí název daného typu akce. Datové proudy byly spojeny nástrojem *Union All* do jediného toku. U atributů vytvořených v SQL dotazích bylo nutné změnit typ na *Unicode string* za pomoci nástroje *Data Conversion*. Jako poslední nástroj byl opět vložen *Slowly Changing Dimension* zajišťující přepis změněných atributů a vložení nových stejně, jako tomu bylo u ETL procesu tabulky *ETLdimObchodniPartner*.



Obrázek 21: Balíček ETL procesu dimDruhTyp.

6.3.6 Dimenze Stav

Nejjednodušší ETL proces je balíček dimenzionální tabulky *dimStav*. Zdrojová data je pomocná tabulka vytvořená v datovém skladu. U nástroje OLE DB Source tedy bylo pouze nadefinováno připojení do databáze a jméno tabulky. Tato data jsou následně rovnou vložena bez jakékoli transformace do tabulky *dimStav* nástrojem *OLE DB destination*.

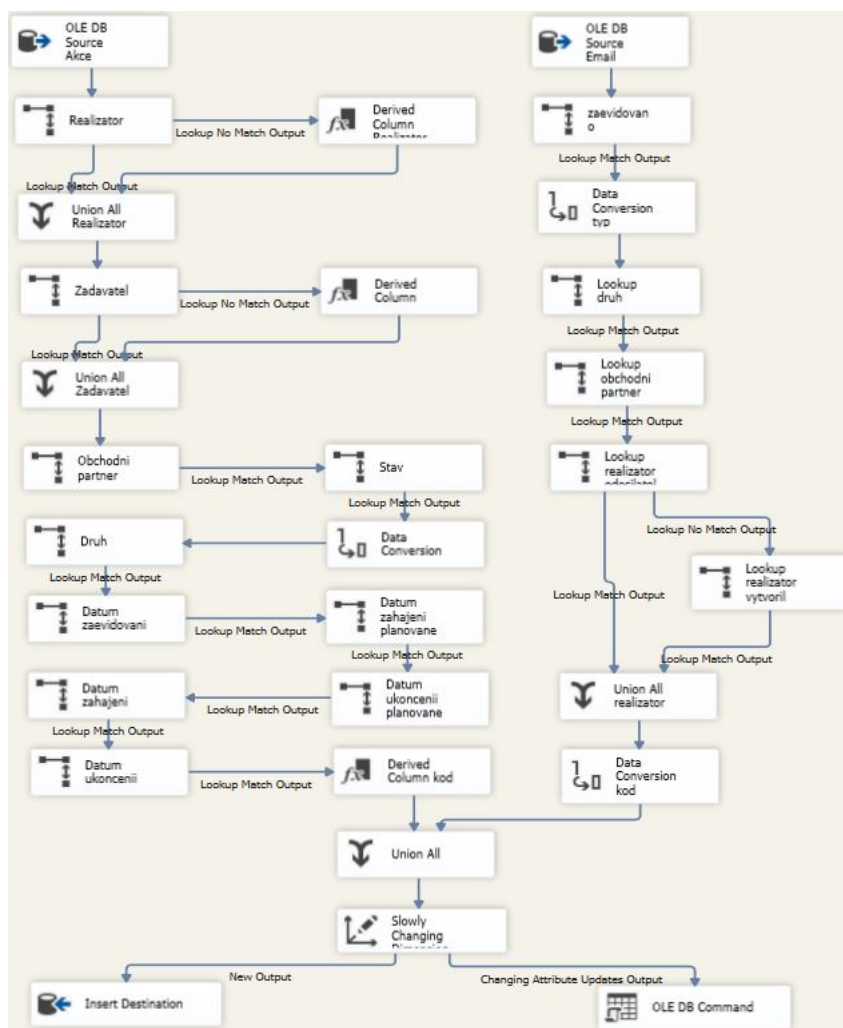
6.3.7 Faktová tabulka Akce

Naopak nejnáročnější ETL proces obsahuje balíček určený tabulce faktové, tedy *factAkce*. Z důvodů rozdílnosti zpracování emailů od zbytku akcí byly vloženy do datového toku dva zdrojové proudy, které se spojují až na závěr ETL procesu. Prvním *OLE DB Source* je tedy datový zdroj akcí. Byl nadefinován pomocí SQL dotazu. Místo hlavní tabulky datového řezu *Všechny Akce* byly spojeny její podřízené tabulky s vytvořeným textovým atributem *Typ akce*, která identifikuje typ akce. K této tabulce byly připojeny tabulky *Všechny akce*, *Druh akce*, *Akce se týká subjektu*, *Subjekt zodpovědný za akci* a *Zadavatel*. Časové atributy byly v SQL dotazu zaokrouhleny dolů na celé hodiny funkcí *DATEADD* a *DATEDIFF*, kde do hodin byl přidán rozdíl mezi nulou a hodinou daného času. Dále byly přidány postupně za sebou nástroje *LookUp* pro zjištění business klíče *realizátora*, *zadavatele*, *obchodního partnera*, *stav*, *druh-typ*, *data zaevidování*, *zahájení plánované*, *ukončení plánované*, *zahájení*, *ukončení*. U *LookUp* nástrojů *realizátora* a *zadavatele* byl přidán nástroj

Derived Colum pro *No Match Output* údaje. Do těchto údajů byl vložen business klíč jedna, který definuje neznámého uživatele. U kódu akce byl změně datový typ na *Unicode String* a nahrazeny prázdné hodnoty nástrojem *Derived Column* za „N/A“ řetězec.

Druhý *OLE DB Source* je určen pro extrakci emailů. Byl nadefinován SQL dotazem, kde byly spojeny dvě tabulky. V jedné tabulce QI eviduje emaily zaslané z informačního systému QI a ve druhé emaily, které byly importovány z jiných emailových klientů (například Outlook). K těmto tabulkám byly připojeny tabulky *záznamy o komunikaci, obchodní partner, organizace a subjekt*. V podmínce byly odfiltrovány emaily zaslané mezi kolegy firmy *SolidVision*. V tomto datovém proudu byly dále vloženy nástroje *LookUp* pro zjištění cizích klíčů atributů *zaevidováno, druh-typ, obchodní partner a realizátor*. Pro zjištění realizátora byly nadefinovány dva nástroje *LookUp*, protože pokud by byl email importován z emailového klienta, tak QI zapisuje do tohoto atributu údaj „Communicator“. Konkrétní uživatel se nachází až u atributu odesílatel. Dále byly použity nástroje *Data conversion* pro změnu typu u časových atributů a atributů *kod a stav*.

Tyto dva proudy byly následně spojeny do jednoho datového toku nástrojem *Union All*. Za tento nástroj byl vložen nástroj *Slowly Changing Dimension* pro zajištění propisu změn u atributů. Kromě atributu *realizátor*, který je ponechán bez změny. *Referent* je ponechán bez změny, protože v případě změny pozice zaměstnance by se přepsaly staré akce, které realizoval na jiné pozici na uživatele s novou pozicí. Uživatel s novou pozicí obsahuje jiný business klíč, protože to bylo nastaveno v nástroji *Slowly Changing Dimension* v balíčku *dimUzivatel*.



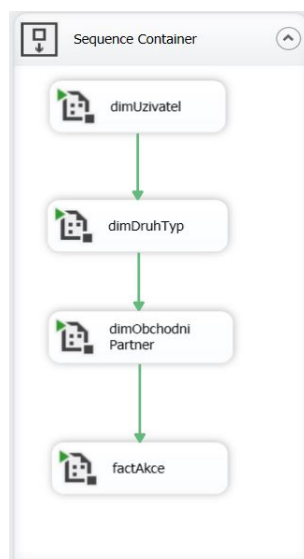
Obrázek 22: Balíček ETL procesu factAkce.

6.3.8 Vložení nulových hodnot

Po nadefinování všech balíčků byly vloženy do dimenzionálních tabulek datového skladu údaje pro nulové hodnoty. Tyto údaje byly vloženy před naplněním tabulek samotnými údaji, aby obdržely business klíč jedna. Tím bylo umožněno v nástrojích *Derived Column* u všech balíčků nahrazovat prázdné hodnoty business klíčem jedna. Tyto údaje byly vloženy SQL insertem v programu *Microsoft SQL Server Management Studio*. Údaje byly vytvořeny tak, aby všechny textové údaje obsahovaly „N/A“ řetězec a číselné údaje obsahovaly nulu. Atributy, na základě kterých byl realizován *LookUp* jsou hodnoty NULL.

6.3.9 Nahrání tabulek

Po vložení těchto údajů bylo realizováno nahrání údajů do tabulek pomocí výše popsaných balíčků. Balíčky byly spuštěny v pořadí *dimUzivatel*, *dimDatumHodina*, *dimDruhTyp*, *dimMisto*, *dimStav*, *dimObchodniPartner* a *factAkce*. Pro zajištění aktuálnosti dat bylo nutné zajistit u měnících se tabulek automatické spuštění ETL procesu. Za tímto účelem byl vytvořen balíček *Load*. Do tohoto balíčku byl vložen nástroj *Sequence Container*, sekvenční spuštění nástrojů SSIS. Pro spuštění konkrétního balíčku byl do *Sequence Container* vložen nástroj *Execute Package Task* pro každý balíček zvlášť. V *Connection Managers* byly nadefinovány balíčky tabulek, které se pravidelně mění, tedy balíčky tabulek *dimDruhTyp*, *dimObchodniPartner*, *dimUzivatel*, a *factAkce*.



Obrázek 23: Balíček Load.

Aby bylo zajištěno automatické spuštění balíčku *Load* na serveru, byl nadefinován v programu *Microsoft SQL Server Management Studio* nový tzv. *Job*. V této komponentě byla nastavena spolu s IT administrátorem cesta k SSIS balíčku *Load* a čas jeho spuštění na čtvrtou hodinu ranní. Report o výsledku provedení Jobu byl nastaven do textového souboru na serveru.

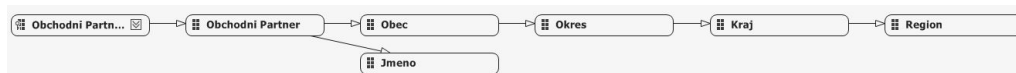
6.4 Realizace OLAP kostky

Pro realizaci OLAP kostky byla použita komponenta SSAS (*Analysis Services*) nástroje *SQL Server Business Intelligence Development Studio*. Nejprve byl nadefinována v *Data Source* databáze, ve které se nachází datový sklad. Dále byly vybrány tabulky datového skladu v *Data Source Views*. Tabulky byly následně propojeny dle návrhu. Každý atribut tabulky odkazující se do druhé tabulky má nadefinovanou svoji vlastní vazbu. Po vytvoření všech vazeb byla vytvořena samotná OLAP kostka

v Cubes. V průvodci tvorby OLAP kostky byla nejdříve vybrána faktová tabulka *factAkce* a následně všechny dimenzionální tabulky. Průvodce automaticky vytvořil údaj počtu akcí ve faktové tabulce a do záložky dimenzí vložil všechny nadefinované dimenze. Tímto krokem se vytvořila základní kostra OLAP kostky.

6.4.1 Nadefinování atributů a hierarchií

Pro jednotlivé dimenze byly následně definovány atributy a jejich hierarchie, které budou v OLAP kostce zpřístupněny. Pro dimenzi stavu jsou to atributy *StavKey* a *Stav*. V této dimenzi nebyla nadefinována žádná hierarchie. V dimenzi *Obchodní partner* jsou k dispozici tři tabulky, protože tabulka *dimObchodniPartner* je provázána s tabulkou *dimUzivatel* a *dimMisto*. Zpřístupněné atributy pro OLAP kostku byly *Jmeno* z tabulky *dimUzivatel*, *Kraj*, *MistaKey*, *Obec*, *Okres*, *Region*, *Statut* z tabulky *dimMista* a *ObchodniPartnerKey*, *ObchodniPartner*, *ReferentKey*, *MistaKey* z tabulky *dimObchodniPartner*. Pro tuto dimenzi byly nadefinovány dvě hierarchie. Hierarchie *Referent*, kde se rozpadá jméno referenta na obchodní partnery. Druhá hierarchie *Místo* zajišťuje rozpad atributů *Region*, *Kraj*, *Okres*, *Obec*, *ObchodniPartner*. Aby byly hierarchie funkční, je nutné nastavit vztahy mezi atributy. V záložce *Attribute Relationships* se definuje vztah vazbami. Podle hierarchie se postupuje tak, že se najde nejspodnější atribut a ten se napojí na business klíč. Následně se vymaže vazba mezi dalším atributem v pořadí hierarchie směrem nahoru a tento atribut se připojí za naposledy přidaný atribut hierarchie (viz obrázek 24).



Obrázek 24: Vztahy atributů *dimObchodniPartner*.

Do dimenze *DruhTyp* byly přidány atributy *DruhKey*, *Druh* a *Typ* tabulky *dimDruhTyp*. Pro tuto dimenzi byla vytvořena hierarchie *Druh Typu*, která definuje pro určitý typ akce její druhy. Stejným způsobem, jako to bylo popsáno u dimenze *Obchodní partner* byly vytvořeny vazby mezi atributy.

Dimenze uživatelů obsahuje atributy *Celkový prodej*, *Jmeno*, *LetosniProdej*, *LetosniPlanProdeje*, *Pozice* a *UzivatelKey*. Hierarchie této dimenze nese název *Oddělení* a určuje příslušnost zaměstnance do oddělení. Atributy této hierarchie jsou *Pozice* a *Jmeno*. Opět bylo nutné nastavit vztahy mezi těmito atributy.

Nejvíce atributů a hierarchií je nadefinováno v časové dimenzi. Zde jsou v OLAP kostce využity všechny atributy tabulky *dimDatumHodina*. Dimenze obsahuje tři hierarchie. První definuje rozpad atributů *Rok*, *Pololetí*, *Ctvrtletí*, *Nazev Mesice*, Druhá hierarchie rozšiřuje první o atributy *Cislo Tydne Rok* a *Nazev Dne*. Poslední hierarchie rozšiřuje předchozí o atribut *Hodina*. Opět byly nadefinovány vazby dle pořadí atributů v hierarchiích. Tentokrát byly ale vazby změněny na typ *Rigid*. Tento typ vazby uvádí, že se již hodnoty atributů nebudou v budoucnu měnit.

Aby byl zajištěn správný rozpad hierarchie, je nutné také definovat pro opakující se hodnoty atributů klíčové sloupce, které slouží k jednoznačné identifikaci výskytu údaje. Například v tabulce *dimDatumHodina* se vyskytuje první pololetí u všech evidovaných let. Aby bylo možné určit, že se jedná o první pololetí daného roku, je nezbytné nadefinovat atribut pololetí další klíčový sloupec a to sloupec *Rok*. Pro atribut *Ctvrletí* jsou to sloupce *Pololetí* a *Rok*, atd. Klíčové sloupce byly tedy nadefinovány ve vlastnostech pro všechny atributy vstupující do hierarchie.

U všech atributů je primárně nastaveno řazení podle abecedního pořadí názvů. Pro nastavení správného řazení názvů měsíců a dnů v týdnu, bylo nejprve zapotřebí u jejich řadících atributů (*CisloMesiceRok* a *CisloDneTyden*) nastavit vlastnosti *AttributeHierarchyEnabled*, *AttributeHierarchyOrder* na hodnotu *False* a atribut *AttributeHierarchyOptimizedState* na hodnotu *NotOptimized*, aby atribut nevstupoval do hierarchie a byl pro uživatele „neviditelný“. Dále bylo nutné vytvořit vazbu mezi řazeným a řadícím atributem. Po vytvoření vazby se ve vlastnostech řazeného atributu nastaví vlastnost *OrderBy* na hodnotu *AttributeKey* a vlastnost *OrderByAttribute* na námi zvolený řadící atribut.



Obrázek 25: Vztah atributů pro nastavení řazení.

6.4.2 Nadefinování automaticky počítaných hodnot

Do pole *Measures*, kde se nachází automaticky vytvořená hodnota z faktové tabulky, která představuje počet akcí, byly přidány další hodnoty. Z tabulky *dimUzivatel* byly přidány hodnoty *Celkovy Prodej Suma*, *Letosni Prodej Suma* a *Letosni Plan Prodeje Suma*, které sčítají hodnoty licencí uživatelů. Z tabulky *dimObchodniPartner* byla vytvořena hodnota *Obchodni Partner Pocet* pro získání počtu obchodních partnerů.

Další hodnoty potřebné pro reporty již bylo nutné vytvořit v záložce *Calculations*, kde se definují automaticky počítané hodnoty jazykem MDX. Byly zde vytvořeny hodnoty *Pocet Akci Na Prodanou Licenci*, *Zbyva Licenci*, *Odhadovany Pocet Akci Pro Splneni Planu*, *Splneny Plan*, *Pocet Pracovnich Dni* a *Prumer Akci Za Den*.

Hodnota *Pocet akci Na Prodanou Licenci* je počítána jako *Pocet Akci* dělený *Celkovy Prodej Suma*. MDX funkcí IIF bylo zajištěno, že pokud je *Celkovy Prodej Suma* roven nule, tak je výsledek roven také nule, aby nevstupovala nula do výpočtu jako dělitel. Výsledek je následně zaokrouhlen funkcí ROUND na dvě desetinná místa. Hodnota *Zbyva Licenci* je počítána jako rozdíl *Letosni Plan Prodeje* a *Letosni Prodej*. Pokud je *Letosni Prodej* větší než *Letosni Plan Prodeje*, tak je výsledek vždy nula, aby nenabývala záporných hodnot. Podmínka byla zajištěna funkcí IIF. Hodnota *Odhadovany Pocet Akci Pro Splneni Planu* je počítána, jako násobek dvou

předchozích hodnot, tedy *Pocet Akci Na Prodanou Licenci* a *Zbyva Licenci*. Výsledek je opět zaokrouhlen na dvě desetinná místa funkcí ROUND. Hodnota *Splneny Plan* je počítána jako podíl *Letosni Prodej Suma* a *Letosni Plan Prodeje*. Podmínkou IIF byla opět zajištěna nulová hodnota ve výsledku v případě nulové hodnoty *Letosni Plan Prodeje*. Hodnota *Pocet Pracovnich dni* je počítána třemi vnořenými funkcemi. Funkce DESCENDANTS vrací všechny potomky (názvy dnů) v aktuálně zvolených letech. Funkce FILTR omezuje tuto množinu na pracovní dny a funkce COUNT vrací jejich počet. Hodnota *Prumer Akci Za Den* je počítána, jako podíl počtu akcí za určité období a *Pocet Pracovnich Dni*. Hodnota počtu akcí za určité období je suma hodnot *Akce Pocet* pro všechny úrovně hierarchie *RokDen* získané funkcí CHILDREN. Byly zahrnuty i prázdné hodnoty funkcí *CoalesceEmpty*, které byly nahrazeny za nulu. Výsledný podíl je zaokrouhlen na dvě desetinná místa funkcí ROUND. V případě, že počet pracovních dní je roven nule, výsledná hodnota je také nula.

```
//Pocet Akci Na Prodanou Licenci
ROUND(
  IIF([Measures].[Celkovy Prodej Suma]=0
    ,0
    ,[Measures].[Akce Pocet]/[Measures].[Celkovy Prodej Suma]
  )
  ,2);

//Zbyva Licenci
IIF([Measures].[Letosni Prodej] > [Measures].[Letosni Plan Prodeje]
  ,0
  ,[Measures].[Letosni Plan Prodeje] - [Measures].[Letosni Prodej]
);

//Odhadovany Pocet Akci Pro Splneni Planu
ROUND([Measures].[Pocet Akci Na Prodanou Licenci] * [Measures].[Zbyva Licenci],2);

//Splneny Plan
IIF([Measures].[Letosni Plan Prodeje Suma] = 0
  ,0
  ,[Measures].[Letosni Prodej Suma]/[Measures].[Letosni Plan Prodeje Suma]
);

//Pocet Pracovnich Dni
COUNT(
  FILTER(
    {DESCENDANTS([Zaeviodovano].[RokDen].CURRENTMEMBER, [Zaeviodovano].[RokDen].[Nazev Dne])}
    ,[Zaeviodovano].[Nazev Dne].Properties( "Druh Dne" ) = "Pracovní"
  )
);

//Prumer Akci Za Den
IIF([Measures].[Pocet Pracovnich Dni]=0
  ,0|
  ,ROUND(
    SUM([Zaeviodovano].[RokDen].CHILDREN,CoalesceEmpty([Measures].[Akce Pocet],0))
    /
    [Measures].[Pocet Pracovnich Dni]
    ,2)
);
```

Obrázek 26: MDX dotazy Calculations Members.

6.4.3 Nadefinování KPI

Jako poslední úprava v OLAP kostce byl nadefinován KPI pro splnění plánu obchodníků. Vstupní hodnota KPI je procentuální hodnota *Splněny Plan*. Cílová hodnota v *Goal expression* je 0,8. Ve *Status expression* byly nadefinovány výrazy KPI. Pokud tedy bude *Splněny Plan* větší nebo roven osmdesáti procentům, tak je plán splněn a výsledná hodnota bude jedna. V případě, že je *Splněny Plan* mezi padesáti a osmdesáti procenty, tak je výsledná hodnota nula. Pokud bude *Splněny Plan* pod padesáti procenty, tak hodnota bude 1. Ve *Status indicator* byly těmto hodnotám přiděleny *Shapes*, které barvou definují stav. Zelená pro splněný plán, žlutá pro blížící se splněný plán a červená pro méně než polovinu splněného plánu.

Name:

Associated measure group:

Value Expression:

Goal Expression:

Status indicator:

Status expression:

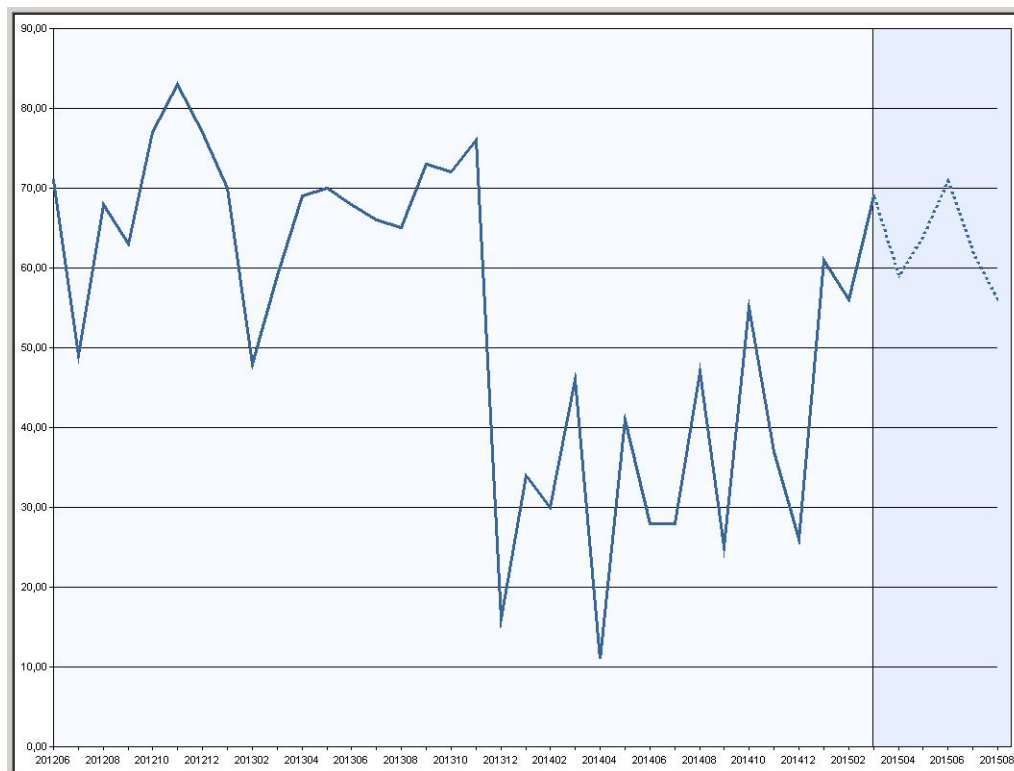
```
Case
When KpiValue("Prodejní Plan") >= KpiGoal("Prodejní Plan")
Then 1
When KpiValue("Prodejní Plan") < KpiGoal("Prodejní Plan")
And KpiValue("Prodejní Plan") > .5
Then 0
Else -1
End
```

Obrázek 27: KPI Prodejní plán obchodníka.

6.5 Realizace predikce informací

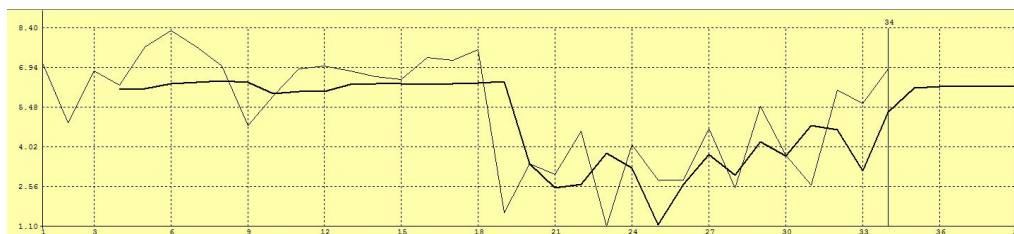
V komponentě SSAS nástroje SQL Server *Business Intelligence Development Studio* byla také vytvořena predikce počtu obchodních případů firmy. Pro Predikci byla vytvořena vlastní tabulka, která obsahuje dva atributy. První atribut představuje jednoznačný identifikátor měsíce složený z roku a čísla měsíce. Druhý atribut obsahuje počty realizovaných obchodních případů. Pomocí SQL dotazu byla tabulka naplněna daty z faktové tabulky akce a v nástroji SSAS byla připojena jako další *Data Source View*. Následně byla vytvořena predikce v záložce *Mining Structures*, která slouží pro tvorbu struktur dolování znalostí z dat. Pro predikci byla využita struktura Microsoftu *Time Series*. Tato technika je přímo určená pro predikci hodnot na časové ose. Byl určen klíčový atribut a atribut určený pro predikci. Jakmile

OLAP kostka zpracuje všechny údaje, je možné v záložce *Mining model Viewer* zobrazit výslednou křivku.



Obrázek 28: Predikce obchodních případů pomocí SSAS.

Jako druhý nástroj pro tvorbu stejné predikce byl zvolen expertní model n-vrstvé neuronové sítě *Neuturbo* pana docenta Trenze. Vstupní hodnoty, tedy počty obchodních případů byly nejdříve poděleny desítkou, aby dosahovaly hodnot mezi nulou a jedničkou. Dále byly nastaveny vrstvy neuronové sítě a aktivační funkce. Následně bylo spuštěno učení algoritmu. Ve výsledném grafu je tučnou čarou zvýrazněná predikce.



Obrázek 29: Predikce obchodních případů pomocí Neuturbo.

6.6 Interpretace informací

Pro tvorbu reportů byl využit nástroj Microsoft Excel. Pro výsledné dashboardy byly vytvořeny v komponentě *PowerPivot* čtyři MDX dotazy, které se dotazují na vytvořenou OLAP kostku a vrátí potřebnou množinu dat pro tvorbu reportů. *PowerPivot* se otevírá v Excelu tlačítkem Spravovat v záložce POWERPPIVOT. Zde byla pomocí tlačítka Načíst externí data ze služby *Analysis Services* připojená OLAP kostka a postupně nadefinovány MDX dotazy, které se vytváří jednoduchým „drag and drop“ nástrojem.

6.6.1 Tvorba MDX dotazů

MDX dotaz Denní průměr akcí zahrnuje hierarchii *Oddělení* dimenze *Realizator*, hierarchii *Druh Typu* dimenze *Druh Typ*, hierarchii *RokMesic* dimenze *Zaevidovano*, atribut *Cislo Mesice Rok* a počítaný atribut *Prumer Akci Za Den*. V dimenzi *Zaevidovano* byl omezen rok na hodnoty 2012-2015 a v dimenzi *Realizator* byly odebrány jména „N/A“ a „Unknown“.

Dotaz *Akce zaměstance* obsahuje hierarchii *Oddělení* dimenze *Realizator*, hierarchii *Druh Typu* dimenze *Druh Typ*, atribut *Stav* dimenze *Stav*, hierarchii *Rok Hodina* a atributy *Cislo Mesice Rok*, *Cislo Dne Tyden* a *Druh Dne* dimenze *Zaevidovano* a počítaný atribut *Akce Počet*. Omezení jsou stejná, jako u předchozího dotazu, tedy rok zaevidování 2012-2015 a jméno realizátora není „N/A“ a „Unknown“.

MDX dotaz *Stav obchodníka* zahrnuje atribut *Jmeno* dimenze *Uzivatel*, hierarchii *Druh Typu* dimenze *Druh Typ* a počítané atributy *Celkovy Prodej Suma*, *Letosni Prodej Suma*, *Letosni Plan Prodeje Suma*, *Zbyva Licenci*, *Pocet Akci Na Prodanou Licenci* a *Odhadovany Pocet Akci Pro Splneni Planu*. Časové omezení je zde stejné jako u předchozích dotazů, omezení uživatelů je zde pouze na pozici obchodník.

Dotaz *Akce pro obchodního partnera* obsahuje hierarchii *Místo* dimenze *Obchodni Partner*, atribut *Jmeno* z hierarchie *Referent* dimenze *Obchodni Partner*, hierarchii *Druh Typu* dimenze *Druh Typ*, atribut *Stav* dimenze *Stav*, hierarchii *Rok Mesic* dimenze *Zaevidovano*, hierarchii *Oddělení* dimenze *Uzivatel* a počítaný atribut *Pocet Akcí*. Omezení je nastaveno pouze u dimenze *zaevidovano* stejně jako u předchozích dotazů.

Dotaz *Obchodni Partneři* obsahuje hierarchii *Místo* dimenze *Obchodni Partner*, atribut *Jmeno* hierarchie *Referent* dimenze *Obchodni Partner*, atribut *Pozice* dimenze *Uzivatel* a počítaný atribut *Obchodni Partner Pocet*.

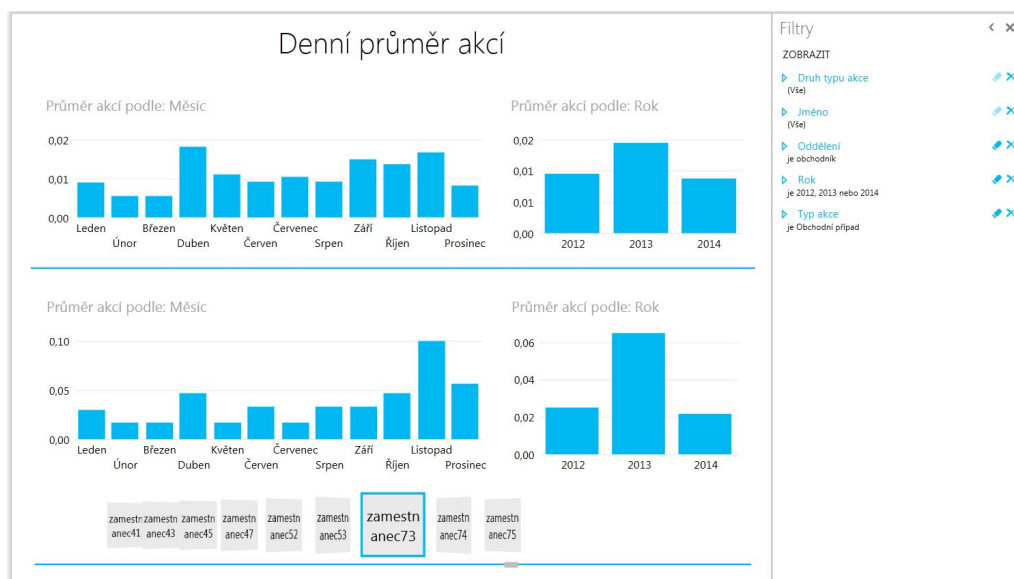
Úprava atributů pro uživatele Získaná data bylo zapotřebí upravit. Názvy všech atributů byly přepsány na uživatelsky srozumitelné názvy. Například atribut *Název Mesice* byl změněn na „Měsíc“. Číselné atributy desetinného typu byly nainportovány s desetinnou tečkou. Česká verze Excelu má nastavený oddělovač, jako desetinnou čárku. Desetinné tečky bylo tedy nutné nahradit pomocí Excel funkce *SUBSTITUTE* za desetinné čárky. Importovaná data bohužel neobsahují nadefinované vlastnosti, jako je například řazení dnů v týdnu. Z tohoto důvodu byly importovány i řadící atributy. Řazení atributu podle jiného bylo nastaveno v *PowerPivot*

pomocí funkce Seřadit podle sloupce, ve kterém byly vybrány řazené a řadící atributy. Nepotřebné atributy pro samotné reporty, jako jsou například zmíněné řadící atributy, byly pro uživatele skryty pomocí funkce Skrýt v nástrojích klienta. Do dat získaná pomocí dotazu *Denní průměr akcí* bylo nutné doplnit několik počítaných údajů. Přesněji se jedná o podíly atributu *Průměr akcí*. Jelikož *PowerPivot* načte všechna data jako jedinou tabulku do jednoho „sešitu“ a nad daty provádí pouze filtraci, docházelo k chybným výpočtům výsledného průměru. Aby byl výsledný průměr vždy jen jedna hodnota, je třeba hodnoty průměrů dělit počtem jedinečných hodnot ostatních atributů.

6.6.2 Tvorba reportů

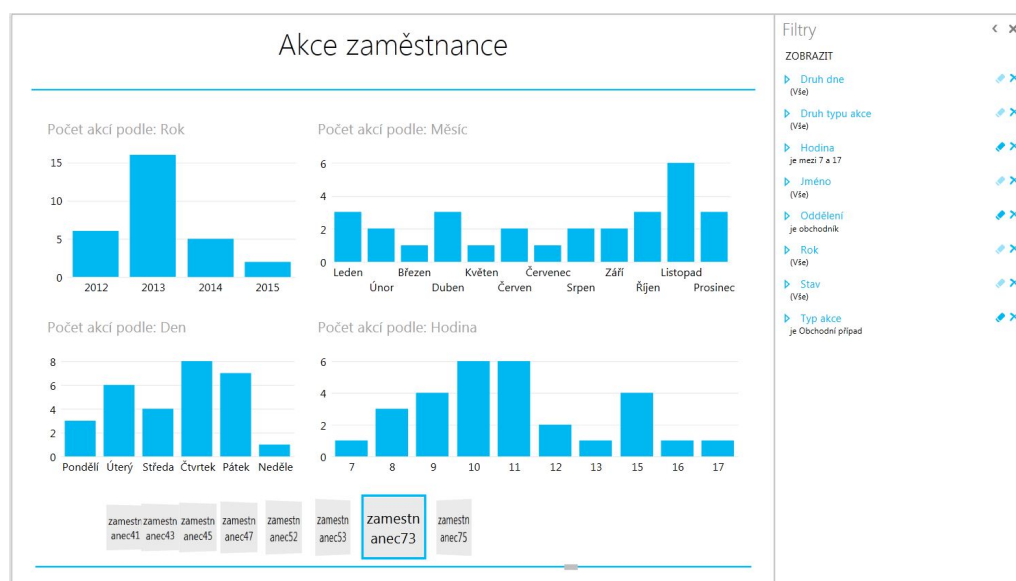
S upravenými daty bylo možné vytvořit reporty pro koncové uživatele. Dashboardy byly do *Excelu* vloženy tlačítkem *PowerView*, které se nachází v záložce Vložení. V nástroji *PowerView* jsou zobrazeny všechny vytvořené dotazy. Tyto dotazy obsahují všechny atributy, které nebyly pro uživatele skryty. Vkládání atributů do reportu je v nástroji *PowerView* realizováno jednoduchým „drag and drop“ nástrojem.

První dashboard *Denní průměr akcí* obsahuje atributy ze stejno jmenného dotazu. Atribut *Jméno* byl vložen do pole *Uspořádat dlaždice podle*, aby se grafy automaticky filtrovaly podle vybraného jména zaměstnance. Tato funkce vloží pod grafy pole čtverců, které nesou hodnoty atributu. Grafy se následně podle těchto hodnot filtrují. Filtr obsahuje dva grafy ve spodní polovině dashboardu. První graf byl vytvořen přesunutím atributu *Průměr akcí* do pole *Hodnoty* a atributu *Měsíc* do pole *Osa*. U každého grafu lze určit na horní liště nástroje *PowerView* typ zobrazování dat. Typ tohoto grafu byl zvolený *skládáný sloupcový*. Druhý graf byl vytvořen stejným způsobem jen s tím rozdílem, že se do pole *Osa* vložil atribut *Rok*. Typ grafu byl určený opět jako *skládáný sloupcový*. Stejně grafy byly vloženy do horní poloviny dashboardu mimo filtrovací oblast dlaždic. Tyto grafy tedy zobrazují stejné hodnoty, ale pro celou firmu. Po pravé straně dashboardu je možné nadefinovat filtry pro celý dashboard. Zde byly opět „drag and drop“ vloženy atributy *Jméno*, *Oddělení*, *Rok*, *Typ* a *Druh Typu akce*.



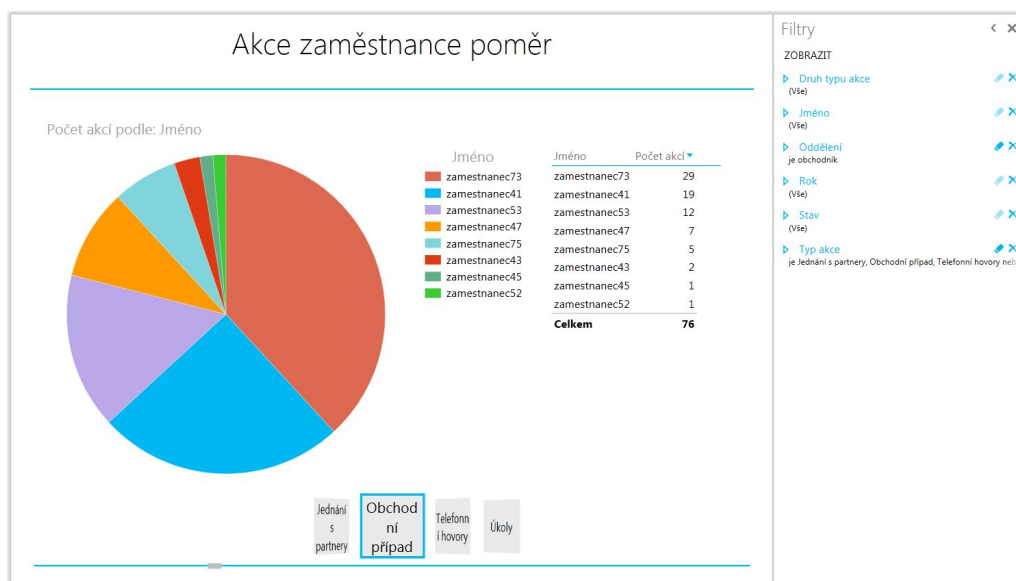
Obrázek 30: Dashboard Denní průměr.

Další dashboard nese název *Akce zaměstnance* a využívá stejnojmenný zdroj dat z *PowerPivot*. Zde byl opět vložen atribut *Jméno* do pole *Uspořádat dlaždice podle* pro možnost přepínání zobrazovaných hodnot podle zaměstnanců. Do tohoto pole byly vloženy čtyři grafy. Všechny grafy obsahují v poli *HODNOTY* číselný atribut *Počet Akcí*. Liší se pouze v attributech umístěných v poli *OSA*. První graf má v tomto poli atribut *Rok*, druhý *Měsíc*, třetí *Den* a čtvrtý *Hodina*. Všechny grafy jsou typu *skládaný sloupcový*. Do pole filtrů byly vloženy atributy *Jméno*, *Oddělení*, *Rok*, *Stav*, *Typ* a *Druh typu akce*, *Druh dne* a *Hodina*.



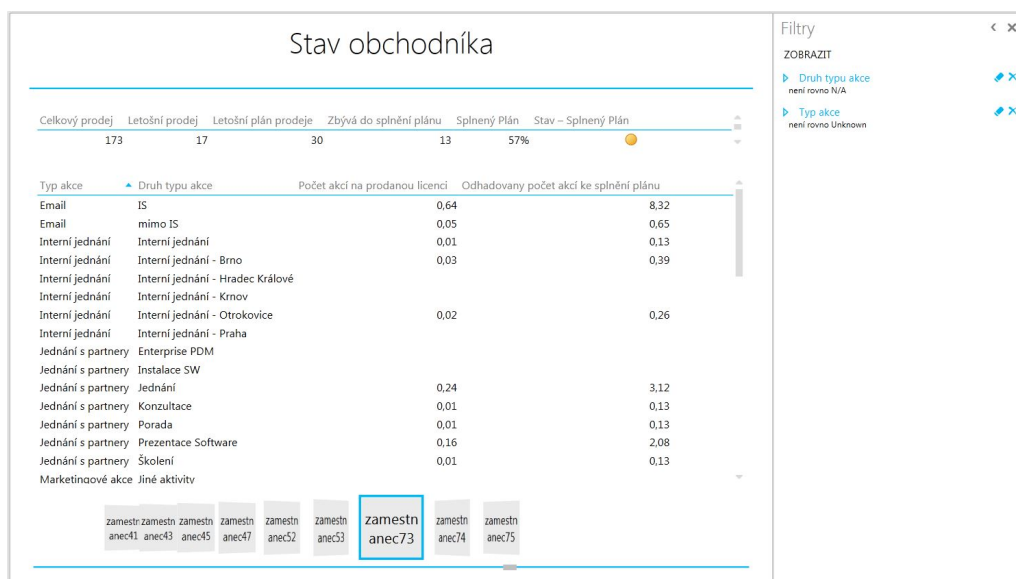
Obrázek 31: Dashboard Akce zaměstnance.

Dashboard s názvem *Akce zaměstnance poměr* využívá stejný datový zdroj jako předchozí dashboard *Akce zaměstnance*. Do pole *Uspořádat dlaždice podle* byl vložen atribut *Typ Akce*. Dále byl vložen graf a tabulka, které zobrazují stejné atributy. Graf je typu *výšečový*. Do pole *Velikost* byl vložen číselný atribut *Počet Akcí* a do pole *Barva* atribut *Jméno*. Graf je seřazen sestupně podle číselných hodnot. Tabulka zde slouží pro číselné zobrazení hodnot, které jsou graficky zobrazeny ve výšečovém grafu. Do pole filtrů zde byly vloženy atributy *Jméno*, *Oddělení*, *Rok*, *Stav*, *Typ* a *Druh typu Akce*.



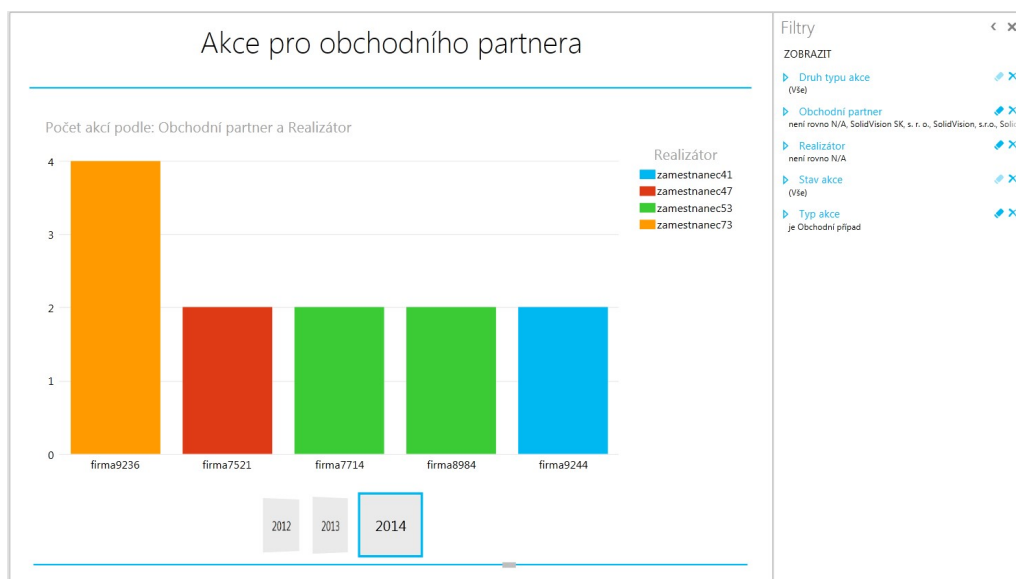
Obrázek 32: Dashboard Akce zaměstnance poměr.

Dashboard *Stav obchodníka* využívá stejnojmenný zdroj dat z nástroje *Power-Pivot*. Do pole *Uspořádat dlaždice podle* byl v tomto dashboardu vložen také atribut *Jméno*. Do Dashboardu byly vloženy dvě tabulky. První tabulka byla umístěna do horní čtvrtiny dashboardu a zobrazuje atributy *Celkový prodej*, *Letošní prodej*, *Letošní plán prodeje*, *Zbývá do splnění plánu* a KPI indikátor *Plnění plánu*. Druhá tabulka zabírá zbytek dashboardu a zobrazuje atributy *Typ Akce*, *Druh typu akce*, *Počet akcí na prodanou licenci* a *Odhadovaný počet akcí ke splnění plánu*. Do pole filtrů byly vloženy atributy *Typ* a *Druh typu akce*.



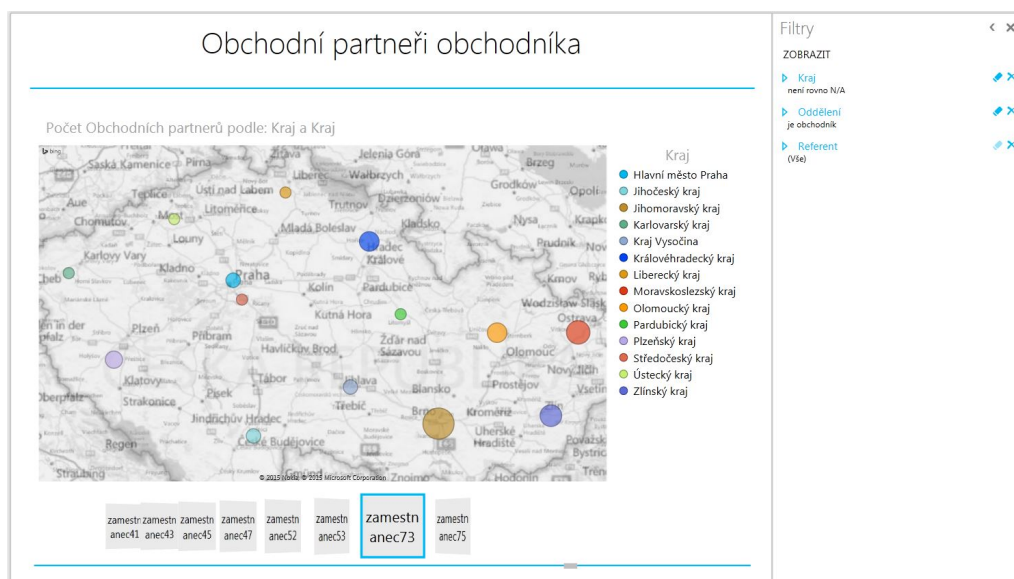
Obrázek 33: Dashboard Stav obchodníka.

Dále byl vložen dashboard *Akce pro obchodního partnera* využívající datový zdroj *Akce pro obchodního partnera*. Do pole *Uspořádat dlaždice podle* byl vložen atribut *Rok*. Celý dashboard obsahuje jediný graf vytvořený přesunutím atributu *Počet akcí* do pole *Hodnoty*, atributu *Obchodní partner* do pole *Osa* a atribut *Realizátor* do pole *Legenda*. Graf je nastaven na typ *sloupcový skládaný* a seřazen sestupně podle číselných hodnot. Do pole filtrů byly vloženy atributy *Realizátor*, *Obchodní partner*, *Stav akce*, *Typ* a *Druh typu akce*.



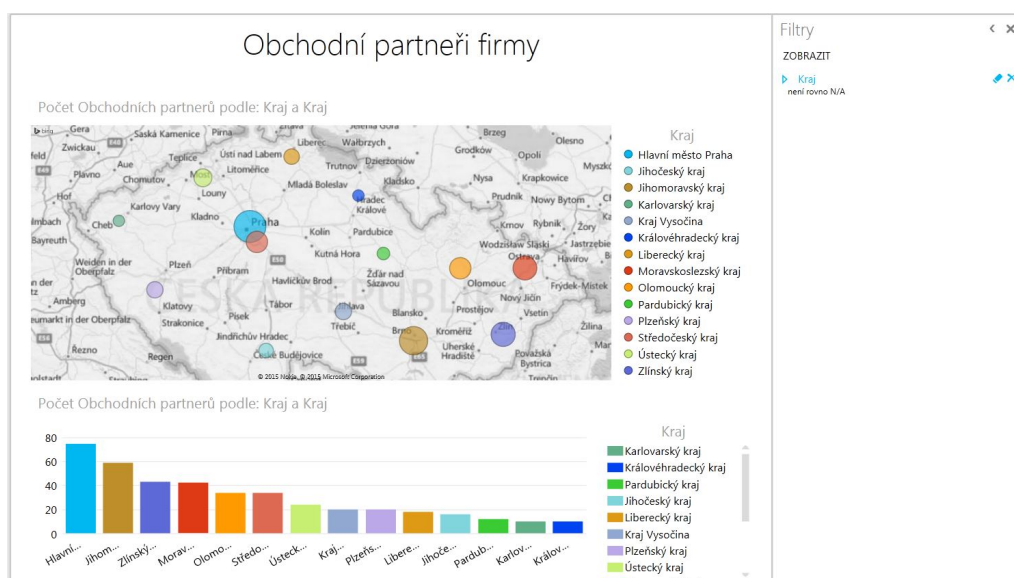
Obrázek 34: Dashboard Akce pro obchodního partnera.

Dashboard *Obchodní partneři obchodníka* využívá datový zdroj *Obchodní partneři*. Do pole uspořádat dlaždice podle byl vložen atribut *Referent*. Dashboard obsahuje speciální graf nástroje *PowerPivot*. Tento graf je typu *Mapa* a umožňuje rozmístit hodnoty na mapě světa díky propojení *Excelu* s mapami Bing. Byl vytvořen přemístěním číselného atributu *Počet obchodních partnerů* do pole *Velikost*, atribut *Kraj* do polí *Umístění* a *Barva*. Do pole filtrů byly vloženy atributy *Kraj*, *Oddělení* a *Referent*.



Obrázek 35: Dashboard Obchodní partneři obchodníka.

Poslední Dashboard nese název *Obchodní partneři firmy* a využívá stejný datový zdroj, jako předchozí dashboard. Byl zde vložen i stejný graf, jako u předchozího dashboardu, ovšem není zde atribut *Referent* v poli *Uspořádat dlaždice podle*. Graf tedy není omezen *Referenty*. Do spodní poloviny dashboardu byl vložen graf zobrazující stejné hodnoty. Graf je ale rozdílného typu, tedy typu *skládaný sloupcový*. Řazení grafu bylo nastaveno sestupně podle číselného atributu. Do pole filtrů byl vložen jediný atribut *Kraj*.



Obrázek 36: Dashboard Obchodní partneri firmy.

7 Diskuze a zhodnocení

7.1 Nedostatky a přednosti implementace

Velkým nedostatkem řešení je fakt, kterým ovšem trpí všechny analýzy prováděné nad databázemi systémů, do kterých vkládají data samotní uživatelé popřípadě zaměstnanci. Jedná se především o nevyplněná nebo chybně vyplněná pole formulářů, zaevidování faktu do systému s delší časovou prodlevou či jeho úplné vynechání. V tomto případě se jedná zejména o problém časových údajů akcí. Zde je možné použít jedině čas zaevidování, který vyplňuje informační systém QI sám a čas plánovaného zahájení. Ostatní časové údaje se pak pohybují pod polovinou vyplněných hodnot. Tento defekt se velice těžko odstraňuje. Zaměstnanci mnohdy vidí informační systém jako přítěž, která jim přidává práci navíc. Je tedy zapotřebí tyto zaměstnance nějakým způsobem motivovat či alespoň oznámit, že se ve firmě zavedl tento systém, díky kterému vedení může nahlížet na jejich aktivity.

Diskutabilní fakt je také zvolení nástroje pro tvorbu reportů v nástroji Microsoft Excel. Bezespornou výhodou je líbivý minimalistický design, který u koncových uživatelů mnohdy vítězí nad samotnou funkcionalitou a interaktivní grafy, které umožňují manažerovi intuitivní práci s grafy. Druhou stranou mince může být způsob, jakým *Excel* pracuje s daty. Komponenta *Power Pivot* získává data z OLAP kostky jednorázovým nahráním do samotného Excelu. Data se tedy musí manuálně aktualizovat. To by nebyl zas až takový problém, protože aktualizace se provádí jedním tlačítkem. Ale problém nastává, jak již bylo zmíněno, v implementaci u počítaných údajů. Pokud se jedná o průměry, tak vzniká problém u souhrnných pohledů (viz. Interpretace informací), protože Excel nežadá o přepočítání údajů samotnou OLAP kostku. Výhodou zde může být nezávislost na přístupu k serveru, kde se OLAP kostka nachází.

7.2 Zhodnocení

Ve výsledné realizaci bylo implementováno vše z nadefinováno návrhu. Reporty jsou přehledné a podávají pravdivé informace z pohledu informačního systému QI. Management je na základě reportů schopen dohlížet nejen na zaměstnance z obchodního oddělení, ale i celé firmy. Je schopen odhadnout potřebný čas obchodníka ke splnění obchodního plánu a jeho úspěšnost z hlediska prodeje licencí. Také má možnost srovnat jeho produktivitu s ostatními zaměstnanci v oddělení. Z hlediska obchodních partnerů může nalézt nevyužitá místa na českém trhu a obchodní partnery, kterým se firma nejvíce věnuje. Implementované řešení tedy naplnilo cíl této práce a také veškeré požadavky managementu. I přes veškeré zmíněné nedostatky byly výsledné reporty managementem firmy úspěšně přijaty a nyní jsou plně využívány. Tento fakt je nejdůležitějším důkazem, že práce splňuje zadané cíle.

Při pohledu na dvě realizované predikce počtu obchodních případů je možné si všimnout, že neuronové sítě spíše vyrovnaly „zubatý“ průběh a predikované měsíce se

pohybovaly kolem jedné hodnoty. Zatímco *Time Series* od Microsoftu napodoboval průběh dosavadní křivky. Predikce jsou ovšem pouze orientační. V reálném životě působí na počet založených obchodních případů mnoho externích a interních vlivů, které lze pouze stěží zahrnout do odhadu. V ostrých datech firmy se predikce více méně shodovaly v rostoucí tendenci.

S ohledem na konkurenční řešení je výsledná práce pro firmu plně dostačující. Na českém trhu již existuje mnoho firem, které vytvářejí BI. Mezi největší hráče na trhu patří zahraniční společnost *GoodData*. Výsledné report této práce by se jistě nedokázaly vyrovnat řešením od této společnosti, která se zabývá tvorbou BI řadu let. Svoje řešení nabízí formou on-line reportů. Ovšem tomuto řešení také odpovídá cena, kterou si nemohou všechny firmy dovolit. Daleko úspornější variantou by bylo zažádat o řešení samotného implementátora informačního systému QI. Výsledné reporty informační systém QI exportuje také do nástroje *Excel*. Ovšem nevyužívá zmíněný moderní nástroj *PowerView*, který dodává reportům tolik důležitou přímou interakci a líbivý vzhled, který může rozhodnout o nákupu daného řešení.

7.3 Další možná rozšíření

Realizované BI by se do budoucna mohla doplnit o faktovou tabulku a k ní přidružené dimenze, které by mapovaly prodané licence. Díky tomuto rozšíření by se mohl sledovat podrobněji plán prodeje obchodníka a celé firmy. Ve faktové tabulce licencí by se mohly evidovat ceny, za které byl software prodán. To by napomohlo přesněji určit schopnosti obchodníka. Bylo by zřejmé, který obchodník prodává velké množství licencí, ale na úkor různých slev a akcí, které firmě snižují výsledný zisk z prodeje.

Dále by bylo také samozřejmě možné rozšířit BI řešení o další oblasti podnikových procesů, které informační systém QI spravuje. Například v oblasti účetnictví, shodnocení objednávek přijatých a nabídek vydaných, atd. Některé důležité informace ovšem informační systém QI již zobrazuje v *Manažerském dashboardu*.

7.4 Ekonomické zhodnocení

Z ekonomického hlediska by mohly vytvořené reporty sloužit pro rozdělování odměn zaměstnancům. Z reportů lze lehce určit produktivitu zaměstnance a porovnat jí s celofiremními hodnotami. U obchodníků by také mohla sloužit, jako podklad při rozdělování odměn hodnota plnění plánu. Reporty lze ovšem využít i v případě nabírání nových zaměstnanců, kde by se mohl sledovat jeho vývoj ve zkušební době zaměstnaneckého poměru. Vzhledem k možnosti sledování plnění obchodních plánů firmy, která je primární složkou příjmů této firmy, jsou výsledné reporty velmi cenné a vedení firmy je začalo ihned po implementaci využívat.

Náklady daného BI řešení jsou jednorázové. Zahrnují návrh a samotnou implementaci. Z hlediska hardwaru a softwaru byly náklady nulové, protože vše potřebné firma vlastnila. Jedná se o server, na kterém běží systém *MS SQL Server 2008 R2*,

kteřý musí každá firma vlastnit v případě, že implementuje informační systém QI. Výkon serveru není nutné zvyšovat pro toto řešení BI, protože veškeré operace načítání dat, které by mohly ovlivnit výkon probíhají v nočních hodinách, kdy je server nejméně zatěžován. Jediným softwarovým nákladem by mohl být pro firmu nákup profesionální edice nástroje *Excel*. Lze jej pořídit v balíčku *Microsoft Office 2013 Professional*, který se dnes na trhu pohybuje kolem ceny třinácti tisíc.

Celkové náklady pro toto řešení tedy stanovuje čas, který implementátor stráví nad touto prací. Odhad časové náročnosti byl stanoven na dvanáct pracovních dní. Výsledná náklady byly odhadnuty na 20 000 Kč. Tyto náklady není jednoduché srovnat s náklady konkurancí. Konkurence často nezveřejňuje ceny svých služeb, aby neodradila potenciální zákazníky. Implementátor informačního systému QI si účtuje 1 250 Kč za hodinu práce. Ovšem je těžké zde odhadnout časovou náročnost. Informační systém QI má jistě již předpřipravené běžné dimenze, jako je například časová. Jednalo by se tedy nejspíše pouze o nějaké úpravy a zahrnutí obchodních plánů obchodníka. Výsledná částka by tedy nejspíše odpovídala výše zmíněným nákladům. U firmy *GoodData* jsou náklady v podobě měsíčních plateb, které probíhají po celou dobu užívání jejich BI řešení. Necelé dva roky zpět byla tato měsíční částka okolo 600 \$. Tyto náklady by převýšily výše zmíněné již za dva měsíce. Ovšem je nutné podotknout, že jejich řešení je vždy vysoce nadstandartní.

8 Závěr

Tato práce měla za cíl vytvořit Business Intelligence nad akcemi zaměstnanců evidovanými v informačním systému QI, které by umožnil vedení firmy přehled nad aktivitou svých zaměstnanců a plnění prodejního plánu obchodníků. Výsledné reporty splňují všechny požadavky managementu firmy shrnuté v kapitole návrh řešení. Implementace datového skladu a OLAP kostky byla realizována na podnikovém serveru *MS SQL Server 2008 R2*. Reporty byly vytvořeny v nástroji *Microsoft Excel*. Celý návrh a implementace byl konzultován s vedoucím diplomové práce panem Ing. Janem Přichystalem, Ph.D. a výsledné reporty s managementem firmy. Ze závěrečných sezení s managementem firmy vzešly možnosti rozšíření nasazeného BI.

V aktuální době je implementované řešení nasazeno na firemním serveru a reporty jsou aktivně využívány vedením firmy. S IT administrátorem firmy se jedná o nasazení Microsoft produktu Sharepoint, pro jednodušší distribuci reportů mezi jednotlivými manažery.

9 Reference

- ČESKÝ STATISTICKÝ ÚŘAD. *Číselník obcí (CISOB)*. [online]. [cit. 2015-05-03]. Dostupné z: https://www.czso.cz/csu/czso/ciselnik_obci_cisob-.
- FERRARI, A. – RUSSO, M. *Microsoft Excel 2013: building data models with PowerPivot*. 1. vyd. Sebastopol: O'Reilly, 2013. 487 s. ISBN 978-0-73-567634-3.
- KIMBALL, R. *The Data Warehouse ETL Toolkit*. 1. vyd. Wiley, 2004. 528 s. ISBN 978-07-645-6757-5.
- KIMBALL, R. – ROSS, M. *The data warehouse toolkit : the complete guide to dimensional modeling*. 2. vyd. New York: Wiley, 2002. 436 s. ISBN 978-0-471-20024-6.
- POUR, J. – MARYŠKA, M. – NOVOTNÝ, O. *Business intelligence v podnikové praxi*. 1. vyd. Praha: Professional Publishing, 2012. 276 s. ISBN 978-80-7431-065-2.
- QI. [online]. 2012a [cit. 2015-05-03]. Dostupné z: <http://www.qi.cz/>.
- QI. [online]. 2012b [cit. 2015-05-03]. Dostupné z: <http://www.qi.cz/o-nas/>.
- QI. [online]. 2012c [cit. 2015-05-03]. Dostupné z: http://www.qi.cz/fileadmin/_temp_/Finance_1.png.
- QI. [online]. 2012d [cit. 2015-05-03]. Dostupné z: http://www.qi.cz/fileadmin/_temp_/Organizace_a_rizeni.jpg.
- QI. [online]. 2012e [cit. 2015-05-03]. Dostupné z: http://www.qi.cz/fileadmin/_temp_/QI_shop_II.jpg.
- QI. *Moduly*. [online]. 2012f [cit. 2015-05-03]. Dostupné z: <http://www.qi.cz/moduly/>.
- QI. [online]. 2012g [cit. 2015-05-03]. Dostupné z: <http://www.informacni-system-qi.cz/>.
- QI. *Naše společnost*. [online]. 2010 [cit. 2015-05-03]. Dostupné z: <http://www.solidvision.cz/solidvision/>.
- THORNTHWAITE, W. – ROSS, M. – KIMBALL, R. *The Data Warehouse Lifecycle Toolkit*. New York, NY. USA: Wiley, 2008. 672 s. ISBN 978-0-47-0149-77-5.