



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA PODNIKATELSKÁ
ÚSTAV INFORMATIKY**

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUT OF INFORMATICS

NÁVRH DATABÁZE PRO EFEKTIVNÍ CHOD STANICE MĚŘENÍ EMISÍ

PROJECT OF A DATABASE FOR EFFICIENT OPERATION OF THE EMISSION STATION

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MICHAL KNĚŽÍNEK

VEDOUCÍ PRÁCE
SUPERVISOR

ING. JIŘÍ KŘÍŽ, PH.D.

BRNO 2012

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Michal Kněžínek

Manažerská informatika (6209R021)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává bakalářskou práci s názvem:

Návrh databáze pro efektivní chod stanice měření emisí

v anglickém jazyce:

Project of a Database for Efficient Operation of the Emission Station

Pokyny pro vypracování:

Úvod
Vymezení problému a cíle práce
Teoretická východiska práce
Analýza problému a současné situace
Vlastní návrhy řešení, přínos návrhů řešení
Závěr
Seznam použité literatury
Přílohy

Seznam odborné literatury:

BERND, H. Access VBA: Velká kniha řešení. 1. vydání. Praha: Computer press, 2006. 639 s. ISBN 80-521-1112-1.

KOCH, M. Datové a funkční modelování. 2. vydání. Brno: Akademické nakladatelství CERM, 2006. 108 s. ISBN 80-214-3252-7.

MORKES, D. Visual Basic: Učebnice pro střední školy. 1. vydání. Praha: Computer Press, 2000. 165 s. ISBN 80-7226-312-9.


ŠEDOVÁ, H. Access 2007 nejen pro střední školy. 1. vydání. Kralice na Hané: Computer media, 2007. 192 s. ISBN 978-80-7402-038-4.

WALKENBACH, J. Excel 2010 Power programming with VBA. 1. vydání. Indiana: Wiley publishing, 2010. 1052 s. ISBN 978-0-470-47535-5.


Vedoucí bakalářské práce: Ing. Jiří Kříž, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2011/12.





Ing. Jiří Kříž, Ph.D.
Ředitel ústavu



doc. RNDr. Anna Putnová, Ph.D., MBA
Děkanka

V Brně, dne 29. 2. 2012

Abstrakt

Bakalářská práce pojednává o návrhu vytvoření databáze pro stanici měření emisí na STK Královo pole, s.r.o. pro zefektivnění administrativních činností na této stanici. Databáze je tvořena dle konkrétních požadavků firmy. Práce obsahuje vlastní návrh databáze a její vytvoření v programu Microsoft Access.

Abstract

This bachelor's thesis deals with the project to create a database for emission measurement station at STK Královo pole, s.r.o. to streamline administrative operations at this station. The database is made according to specific business requirements. The work contains own database design and creating by Microsoft Access.

Klíčová slova:

Databáze, Microsoft Access, Návrh databáze, Tvorba databáze, Visual Basic, VBA, Relační datový model, Data, Informační systém, Normalizace, Informace

Keywords:

Database, Microsoft Access, Database design, database creation, creation of a database, Visual Basic, VBA, Relational Data Model, Data, Information System, Normalization, Information

Bibliografická citace

KNĚŽÍNEK, M. *Návrh databáze pro efektivní chod stanice měření emisí*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2012. 86 s. Vedoucí bakalářské práce
Ing. Jiří Kříž, Ph.D.

Prohlášení

Prohlašuji, že předložená bakalářská práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná a neporušuje autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne

.....

podpis autora

Poděkování

Děkuji tímto vedoucímu bakalářské práce panu Ing. Jiřímu Křížovi, Ph.D. za cenné připomínky, ochotu a rady během práce na bakalářské práci.

Dále bych chtěl poděkovat panu Ing. Michalu Suchomelovi za pomoc při tvorbě návrhu databáze.

Obsah

Úvod.....	11
Vymezení problému.....	12
Cíl práce.....	13
1 Teoretická východiska	14
1.1 Data a informace	14
1.2 Pojem databáze.....	15
1.2.1 Databáze.....	15
1.2.2 Typy databází.....	16
1.2.3 Systémy pro správu databáze.....	17
1.3 Datové modely	19
1.3.1 Lineární datový model	19
1.3.2 Hierarchický datový model.....	19
1.3.3 Síťový datový model.....	20
1.3.4 Objektový datový model.....	21
1.3.5 Relační datový model	22
1.4 Relační databáze.....	22
1.4.1 Terminologie.....	23
1.4.2 Vlastnosti relačních tabulek.....	24
1.5 Integrita relačního modelu	24
1.5.1 Klíče.....	24
1.5.2 Integrita hodnot.....	25
1.5.3 Vztahy mezi relačními tabulkami	25
1.6 Datové sklady.....	26
1.6.1 OLAP vs. OLTP	26
1.6.2 Architektura datových skladů	28
1.6.3 Multidimenzionální databáze.....	30
1.6.4 Datová tržiště	31
2 Analýza problému a současné situace	32
2.1 Představení společnosti	32
2.2 Organizační struktura firmy	33
2.3 Zhodnocení podnikání firmy.....	33
2.3.1 Orientace na trhy.....	33
2.3.2 Proces financování firmy	33
2.3.3 Legislativní aspekty podnikání	34

2.3.4	Konkurence	34
2.3.5	Hlavní výrobní zdroje a majetek podniku.....	35
2.4	Marketingový mix	36
2.4.1	Produkt.....	36
2.4.2	Místo	37
2.4.3	Cena	38
2.4.4	Propagace.....	38
2.4.5	Personál.....	39
2.5	Silné a slabé stránky podniku.....	39
2.5.1	Porterův model pěti sil	39
2.5.2	SWOT analýza.....	41
2.6	Zhodnocení běžného provozu	41
2.7	Zhodnocení informačních technologií ve firmě.....	42
2.7.1	Hardware a software	42
2.7.2	PC síť	43
2.7.3	Záloha a archivace dat	43
2.7.4	Informační systémy.....	43
2.8	Zaměření na řešený problém	44
2.9	Shrnutí situace z ekonomického hlediska	45
2.10	Shrnutí nedostatků informačních systémů.....	45
2.11	Návrh kroků pro další rozvoj společnosti.....	46
3	Návrh řešení	48
3.1	Požadavky na databázi	48
3.1.1	Formulace cílů a úkolu	48
3.1.2	Pozorování provozu a dodatečná formulace cílů a úkolu	49
3.1.3	Požadavky na datový sklad.....	50
3.2	Postup tvorby relační databáze.....	50
3.2.1	Identifikace tabulek a jejich atributů	51
3.2.2	Návrh modelu	52
3.2.3	Klíče.....	54
3.2.4	Normalizace	55
3.2.5	Tvorba obslužných procedur databáze	57
3.2.6	Formuláře – uživatelské prostředí.....	66
3.3	Postup tvorby datového skladu	68
3.4	Předpokládané přínosy návrhu řešení	69

Závěr	70
Použitá literatura	71
Seznam ilustrací	72
Seznam použitých zkratk	73
Seznam příloh	74

Úvod

V dnešní době rozmachu informačních technologií se databázové systémy začínají stávat pomocníkem nejen velkých firem ale i drobných podnikatelů, kterým mohou sloužit k uchovávání informací v datech a následnému získávání poznatků z těchto informací. Doba, kdy se informace uchovávaly v papírové podobě v různých kartotékách, je již přežitkem, a proto se databázové systémy a jejich programovací jazyky dostávají stále více do povědomí i širší veřejnosti. Co je ale hlavním smyslem databáze a co slouží pro podporu rozhodování v podnikatelské činnosti, je schopnost použít tyto poznatky a informace získané z dat, která se nacházejí v různých databázových systémech.

Postupem času, jak se databázové systémy vyvíjejí, tak se snižuje i jejich finanční náročnost a nezůstává jen výsadou velkých firem, ale z praxe lze soudit, že dnes již každý drobný podnikatel využívá prostředků databázových systémů. Velký přínos poskytují pro mnoho obchodníků v podobě transakčních databází, kdy podnikatel potřebuje neustále aktualizovat informace o různých výrobcích a jejich stavu. Ale také v podobě datových skladů sloužících pro uchovávání rozsáhlého množství dat v dlouhém časovém horizontu a tvorbě složitých analýz, které jsou východiskem pro mnoho zkušených manažerů řídicích obrovské podniky.

Dnes už se však můžeme setkat i s případy, kdy před mnoha lety vytvořená databáze už přestává splňovat požadavky, které před pár lety plně postačovaly, ale dnes již vlivem důsledku modernizace a rozvíjení podniků nepostačují. Proto je zapotřebí tyto informační systémy neustále vyvíjet tak, aby držely krok s dobou a hlavně s požadavky jejich uživatelů. Tímto problémem se budu zabývat ve své bakalářské práci, kde se bude pojednávat o návrhu nové databáze pro stanici měření emisí tak, aby se zefektivnil chod firmy, jelikož stávající databáze je již zastaralá a namísto usnadňování práce začíná být spíše přítěží.

Vymezení problému

Problém, který nastal, mě přivedl na teorii relačních databází, které jsou v dnešní době velmi populární a rozšířené. V podstatě jsem byl postaven před problém, který se postupem času začal neustále více projevovat na efektivním chodu stanice technické kontroly (dále jen STK). Jinými slovy lze říci, že stávající zastaralý informační systém pro evidenci a pro potřeby vedení společnosti, ve smyslu registrace zákazníků, účetnictví, statistik a kontroly zaměstnanců, přestal vyhovovat. Do firmy jsem byl přijat v rámci praxe ve třetím ročníku s požadavkem vedení firmy, abych se pokusil navrhnout řešení, které by odbouralo nedostatky stávajícího systému. Problém se hlavně týkal pracovníků registračního oddělení pro stanici měření emisí. Na tomto pracovišti se nacházel již zmiňovaný zastaralý informační systém, který dle mého názoru velmi znesnadňoval práci zaměstnanců, kdy se do tohoto systému musel každý zákazník zdlouhavě registrovat. Navíc tento systém znemožňoval efektivní kontrolu, celkové souhrny a součty jednotlivých druhů měření a pracovníků, jež tato měření prováděli pro celkové měsíční a roční statistiky a výsledky hospodaření podniku. Jinými slovy musel sekretariát podniku všechny informace získávat zdlouhavými cestami a tyto statistiky pak provádět ručně.

Cíl práce

Cílem mé bakalářské práce je vytvořit návrh, který bude řešit nedostatky v podobě pohodlné registrace zákazníků a jejich vozidel, objednávek a jednoduchých dotazů. Hlavně se bude jednat o návrh transakční databáze v prostředí MS Access, neboť na počítačích v této firmě běží operační systém Microsoft Windows a je zde dostupná sada nástrojů MS Office. Prostor MS Access mi přijde relativně jednoduché na pochopení řadovými pracovníky evidence na STK. Navíc se dá velmi jednoduše propojit s aplikací Microsoft Excel, kde se pak dají vytvářet různé pohledy na data a statistiky, které jsou v dnešní době velmi důležité pro podporu rozhodování. Druhým bodem pak je vytvoření návrhu datového skladu v prostředí Microsoft SQL server 2008.

V práci budu postupovat dle mého názoru srozumitelným postupem tak, aby i člověk, který není zasvěcený do problematiky relačních databází, se zde mohl zorientovat a pochopit smysl tvorby návrhu řešení. Nejprve se zaměřím na teoretickou problematiku databází obecně, kde a jak vůbec vznikly a k čemu jsou dobré. V dalších podkapitolách se více zaměřím na relační databáze, neboť ty jsou pro můj návrh řešení stěžejní. Po tomto krátkém seznámení se pokusím o rozbor současné situace, kde bude společnost patřičně představena, a kde se také bude nacházet analýza současné databáze, která je velice důležitá, abych se mohl vyvarovat stávajících chyb. Na toto téma bude navazovat vlastní návrh řešení, který pak bude podrobně popsán.

1 Teoretická východiska

Pokud budeme chtít porozumět problematice datového modelování, je nezbytné poznat základní pojmy, jako jsou data, informace a znalost. Pokud toto pochopíme, tak zároveň porozumíme smyslu, proč se data uchovávají a k čemu nám slouží.

V této kapitole se tedy budu zabývat teoretickými východisky, které by měly objasnit problematiku datového modelování a databázových systémů, kdy nám tyto systémy poskytují možnost shromažďovat data a z těchto dat pak získávat informace potřebné pro rozhodování.

1.1 Data a informace

Porozumění informace nám dává určitou znalost, při vědomí, že čerpáme ze smysluplného zdroje. Informaci tedy můžeme chápat jako určitou zprávu či vjem, který splňuje tři základní požadavky. Prvním z nich je porozumění sdělení neboli tzv. syntaxe. To znamená, že subjekt, který zachytí takovéto informační sdělení, mu rozumí. Druhým požadavkem pak je porozumění obsahu tzv. sémantika. Subjekt je schopen vědět, jaký má zachycená zpráva význam, tedy co znamená, co vypovídá o něm a jeho okolí. Z této zprávy má subjekt nějaký dojem, tedy má pro něj určitý význam, tedy mluvíme o třetím požadavku a to sice relevanci (Koch, 2006).

O datech se dá říci, že představují jakýsi zdroj informací, které se vhodným způsobem dají z dat vyčíst. Pro naše potřeby se mohou data nacházet v různých formách datových uložišť. Pod tímto pojmem si můžeme představit různé databázové systémy, ale i psané dokumenty, které například zaznamenávají určité údaje o podniku, které pro daný podnik mají určitý význam. V jistém slova smyslu jsou data statická, tedy neměnná až do okamžiku, kdy je někdo nezmění. Na první pohled mohou být data nesmyslná. Je tedy důležité nejprve data vhodně zpracovat, než dostanou podobu smysluplné informace (Hernandez, 2006).

Výsledkem porozumění informacím je znalost, která je důležitá pro již zmiňovanou podporu rozhodování, avšak:

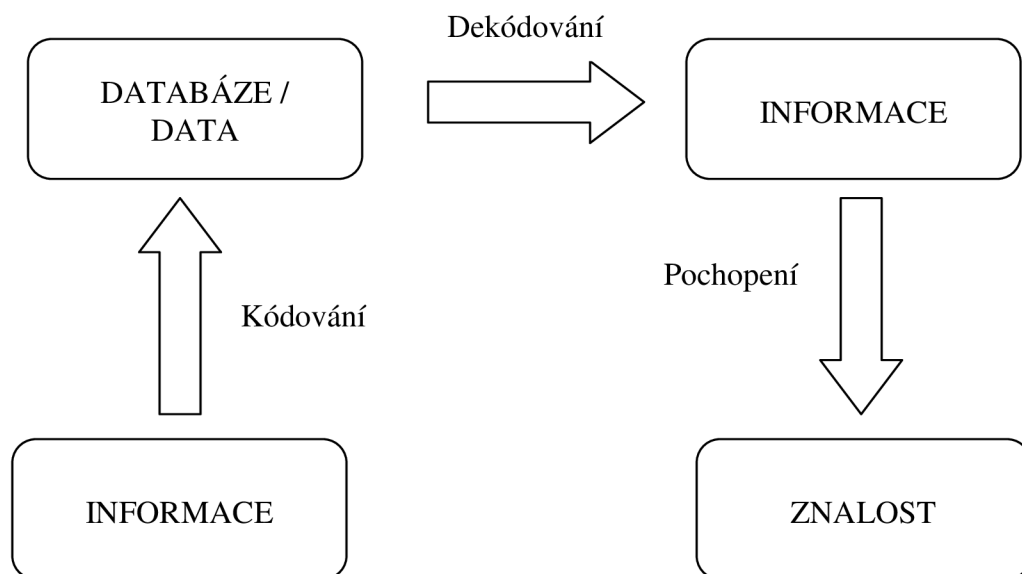
„Doba pro rozhodnutí nesmí překročit čas, který je vymezen existencí problému“
(Koch, 2006, s. 6).

1.2 Pojem databáze

Nyní, když už bylo vymezeno, co je to informace, data a lehce nastíněno, jak se tato data uchovávají, přešel bych k vysvětlení některých základních pojmů, které jsou spojeny s databázemi a jsou nezbytné k pochopení dané problematiky.

1.2.1 Databáze

Databáze jsou dnes neoddělitelnou součástí každodenního života velké většiny z nás. Za dobu, co jsou efektivně využívány, vytvořily průmysl s obratem v miliardách dolarů. Jsou využívány v takovém rozsahu, než by si kdokoli z nás mohl představit. Pro úplnost zde uvedu několik konkrétních příkladů, kdy se s nimi běžný uživatel může setkat. Tak například je používáme při naší obvyklé návštěvě obchodu nebo plánování dovolené v cestovní kanceláři, nebo objednání letenky. Výčet takovýchto situací by mohl pokračovat, ale bylo by to bezpředmětné. Co je však důležité je uvědomění si jejich významu a přínosu (Hernandez, 2006).



Obrázek 1: Schéma koloběhu informací
Upraveno dle: Koch, 2006, s. 5

Databázi tedy můžeme formulovat asi takto: „*Databáze je soubor dat, který je používaný k modelování některých typů organizačních struktur nebo organizačních procesů*“ (Hernandez, 2006, s. 42). V tomto obecném případě vůbec nezáleží na tom,

zda data uchováváme na papíře, nebo k tomu využíváme nějaký počítačový program. Pokud tato data uchováváme určitým organizovaným způsobem, pak můžeme hovořit o tom, že máme databázi (Hernandez, 2006).

Databáze nám slouží k uchovávání, sbírání, vyhledávání a zpracování dat. Pro zprávu dat nám v dnešní době slouží různé informační systémy, které v průběhu několika předchozích let vytlačily klasické psané záznamy, jako byli např. kartotéky, ale také odstraňují problémy, kdy se data uchovávala na různých místech v organizacích, na různých počítačích a v různých textových editorech (Hernandez, 2006). Tato chaotická data se velmi obtížně analyzují. Data jsou nekonzistentní, obtížně dosažitelná, izolovaná, nechráněná a hlavně je zde problém integrity dat. Abych tento výčet nedostatků trochu upřesnil, tak zde uvedu některé příklady z reálného světa (Šeda, 2002):

- **Problém integrity:** Problém může nastat například, pokud ukládáme informace o peněžní hodnotě, kdy se mohou pokaždé vyskytovat jiné měny
- **Nekonzistentnost:** Například se nám vdá pracovnice, a změnu příjmení nahlásí na účtárně, ale v jiných oblastech podniku bude vedena stále pod starým příjmením.
- **Nechráněnost:** Neoprávněná osoba se může dostat k firemním záznamům a pozměnit je. Firma pak z těchto dat získá chybné informace. Velký problém například v bankovníctví.
- **Izolovanost:** Data jsou roztroušena v různých souborech a na různých místech v podniku.

Vyřešení výše zmíněných problémů s daty vedlo postupem času k zavedení databázových systémů. Data jsou již organizovaná v centrální struktuře dat, zvané databáze. Centrální zpráva takovéto databáze je řízena takzvaným systémem řízení báze dat SŘBD, která spolu s datovým úložištěm tvoří databázový systém (Šeda, 2002).

1.2.2 Typy databází

Ve světě databází se můžeme setkat se dvěma typy databází. Každý z nich se liší strukturou uchovávání dat a správou přístupu k těmto datům. Jsou to operační databáze a analytické databáze.

Operační databáze jsou zatím nejvyužívanějším typem databází pro správu dat, jsou využívány hlavně malými a středně velkými podniky. Z anglického OLTP můžeme přeložit jako online transaction processing. Jejich typickým použitím jsou situace, kdy je potřeba neustále měnit a aktualizovat data. Příkladem z reálného světa mohou být maloobchody, nemocnice, průmyslové podniky, apod., neboť jejich data se neustále mění (Hernandez, 2006).

Jakýmsi pomyslným protikladem k tomu, co jsem právě zmínil, může sloužit analytická databáze. Z anglického OLAP pak zase můžeme přeložit jako online analytical processing, kde je potřeba ukládat data za co nejdelší časové období. Tato data se v tomto typu databáze považují za statická, což znamená, že se nemění. Přidávají se zde data pouze nová. Informace, které nám tyto databáze umožňují poskytnout, pak ukazují obraz dat v čase a slouží pro rozsáhlé analytické operace. Z reálného světa zde zase můžeme zmínit například geologické organizace nebo chemické laboratoře (Hernandez, 2006).

1.2.3 Systémy pro správu databáze

„Softwarový systém, který umožňuje definovat, vytvářet a udržovat databázi a poskytuje řízený přístup k této databázi“ (Conolly a kol., 2009, s. 38).

Jak je již patrné z definice, tak se jedná o software, někdy také přezdívaný zkráceně DBMS nebo česky systém řízení báze dat SŘBD. Tento software je navržený k tomu, aby uživatelům usnadnil přístup k datům. Komunikuje tedy s uživatelem a to oboustranně, tzv. interaktivně. Dále také musí komunikovat s databázovými aplikacemi a s databázemi tak, aby umožnil uživatelům vkládat nová data, aktualizovat tato data, mazat a vyvolávat data z databáze. Pokud pak máme naše data uloženy v centrální databázi, na kterou je možný centrální pohled na všechna data, můžeme pomocí DBMS použít takzvaného dotazovacího jazyka k vytváření pohledů na tato data. Jedním ze základních a nejpoužívanějších dotazovacích jazyků, který bych zde chtěl zmínit, je například SQL jazyk. Doslovně tato zkratka znamená Structured Query Language (Conolly a kol., 2009).

Důvod, proč zde zmiňuji jazyk SQL, je ten, že je hlavním dotazovacím jazykem pro relační DBMS, o kterém se ještě zmíním v dalších kapitolách. Mohou to být

například Microsoft SQL Server, DB2, Oracle, ale hlavně Microsoft Access, který budu používat pro tvorbu návrhu databáze (Conolly a kol., 2009).

SQL jazyk tedy vznikl pro práci s relacemi. Za jakýsi počátek vzniku tohoto jazyka lze považovat osmdesátá léta, kdy se datuje vznik relačních databází, pro jejichž potřebu byl jazyk vytvořen. Tento jazyk patří do kategorie tzv. deklarativních programovacích jazyků. To v praxi znamená, že se kód jazyka nepíše v žádném samostatném programu, ale je vkládán do jiného programovacího jazyka, který je již procedurální. S tímto se pak pracuje tak, že se naším klientským počítačem, někdy nazývaným terminálem, připojíme na databázový server, kde na příkazový řádek zadáváme přímo příkazy jazyka SQL (Koch, 2006).

Jazyk SQL se postupem času neustále vyvíjel a postupně byl přijat jako standard různými výrobci databázových systémů (Koch, 2006).

Naznačil jsem tady, že se zde sdílejí nějaká centrálně uložená data. Tato myšlenka se stala zřejmou během 80. a 90. let, kdy se databáze prudce vyvíjely a začínalo je používat stále více uživatelů. S tímto se umožnila také implementace bezpečnosti. Klientský přístup k takovému centrálnímu úložišti dat se pak nazývá klient/server. Zjednodušeně se data nacházejí na počítači, který slouží jako databázový server a uživatelé pak pracují s těmito daty prostřednictvím svých klientských počítačů a aplikací spuštěných na těchto počítačích. Toto je známo jako dvouvrstvá architektura klient- server (Conolly a kol., 2009).

V devadesátých letech, kdy se aplikace začaly stávat složitějšími, tento systém začal vykazovat jisté nedostatky v podobě podstatného zatížení klienta administrací a dalšími požadavky na klientský počítač jako je výkon CPU, ale také zdroje diskového prostoru a RAM. Někdy se takto připojenému klientovi přezdívá také tlustý klient. Z těchto problémů vyšel nový, třívrstvý model neboli třívrstvá architektura klient-server. Každá z těchto vrstev se pak mohla nacházet na jiné platformě. Klient zde vystupuje jako takzvaný tenký klient, kdy jsou po něm požadovány mnohem menší nároky na hardware než ve dvouvrstvé architektuře. Hlavní výhodou pak ale je, že tato architektura celkem věrohodně odpovídá prostředí internetu s webovým prohlížečem, které simuluje tenkého klienta a webovým serverem, který vystupuje jako aplikační server. Celkem tedy můžeme shrnout, že vrstva uživatelského rozhraní se nachází na

klientském počítači, vrstva logiky provozu a zpracování dat se pak nachází na aplikačním serveru, který je navržen tak, že ho může obsluhovat více klientů. DBMS může ležet na odděleném serveru nazývaném databázový server (Conolly a kol., 2009).

1.3 Datové modely

„Posláním automatizovaného informačního systému je poskytovat informace o určité části reálného světa“ (Šeda, 2002, s. 7).

Jinými slovy bychom měli zvolit takový model dat pro tvorbu databázového systému, který bude co nejlépe vystihovat realitu. Než se tedy pustíme do samotného budování takového systému, je velmi důležité analyzovat realitu a vytipovat objekty, o nichž budeme chtít udržovat informace v databázi (Šeda, 2002).

V následujících podkapitolách představím některé základní modely dat, ze kterých projektant vybírá ten nejvhodnější, z nichž dva už nemají technickou podporu současných databázových systémů, neboť se už nepoužívají (Koch, 2006). Hlavním modelem, u kterého se zastavím, je relační model, který je v dnešní době nejpoužívanější a bude také předmětem mého návrhu databáze.

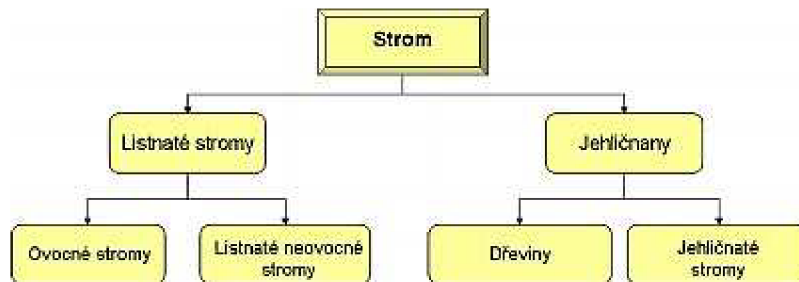
1.3.1 Lineární datový model

Jde o velmi jednoduchý model, ve kterém sledované objekty nemají mezi sebou žádný vztah. Nemůžeme tedy přímo stanovit, jaké jsou vazby mezi jednotlivými objekty. Pro příklad zde uvedu kartotéku pacientů u lékaře, kde každý pacient představuje jeden sledovaný objekt. Údaje o pacientovi jsou zaznamenávány na kartu pacienta a tyto karty jsou pak seřazeny v kartotéce. Každá karta pak představuje jeden objekt nebo chceme-li větu databázového systému (Koch, 2006).

1.3.2 Hierarchický datový model

Data v tomto modelu jsou reprezentována tabulkami, které mezi sebou mají určité vazby. Tyto vazby pak tvoří hierarchickou strukturu v podobě obráceného stromu. V takovéto stromové struktuře je pak jedna tabulka kořenem celého stromu a ostatní tabulky reprezentují větve tohoto stromu, tzv. děti a tyto se dále můžou větvit (Managed Dedicated Servers, 2009).

Vztahy v hierarchické databázi lze tedy jednoduše charakterizovat jako vztah rodiče a potomka. Takovéto tabulky jsou pak mezi sebou propojeny šipkami, nebo prostorovým rozvržením záznamů v tabulce. Přístup k datům se provádí ve směru hierarchie se začátkem u kořene, přičemž se postupuje dále přes stromovou strukturu až ke hledaným datům (Managed Dedicated Servery, 2009).



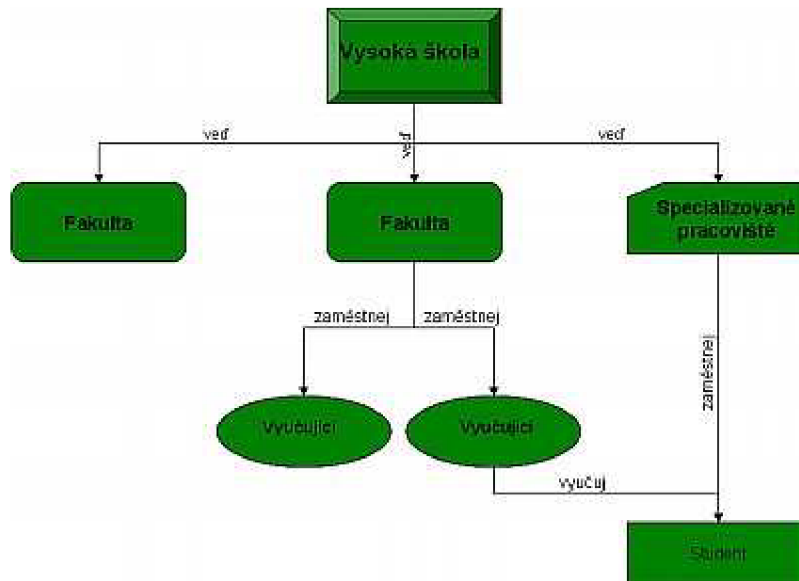
Obrázek 2: Hierarchický datový model

Zdroj: Managed Dedicated Servery, 2009

Výhodou tohoto modelu je velmi rychlý přístup k hledaným datům díky přehlednému propojení. Možnou výhodou pak ještě může být zabudování automatické referenční integrity. To znamená, že pokud smažeme záznam v tabulce rodiče, budou odstraněny všechny propojené záznamy v tabulkách potomků. Nevýhodou je pak redundance, což je způsobeno tím, že tento model nepodporuje tvorbu komplexních vztahů (Managed Dedicated Servery, 2009).

1.3.3 Síťový datový model

Síťový model byl vyvinut hlavně jako pokus o vyřešení problémů hierarchické databáze (Managed Dedicated Servery, 2009). Podle (Koch, 2006) je tedy obdobou hierarchického modelu, kde ale vazby nevedou pouze z rodičovské tabulky, ale mezi sebou obecně v různých směrech. Struktura databáze bývá vyjádřena v pojmech uzlů jako záznamy a množinových struktur (Managed Dedicated Servery, 2009). Na obrázku můžeme vidět schéma datového modelu, konkrétně schéma z prostředí vysoké školy.



Obrázek 3: Schéma síťového datového modelu

Zdroj: Managed Dedicated Servers, 2009

Je to snadno pochopitelná konstrukce, kde je vztah mezi uzly definován jako vlastník a prvek. V této množinové struktuře jeden ze záznamů z vlastníka může být ve vztahu k jednomu, nebo více záznamům v uzlu prvek. Naopak zase jeden ze záznamů z prvku může mít vztah pouze k jednomu ze záznamů v uzlu vlastníka (Managed Dedicated Servers, 2009).

Uživatel má možnost k datům přistupovat pomocí odpovídajících množinových struktur, což může být oproti hierarchické databázi výhodou, kde se naopak musí postupovat od kořenové tabulky. Tedy uživatel může k datům přistupovat z libovolného uzlu a procházet pak přidruženými množinami. V dnešní době se však tento model už vůbec nepoužívá. Uživatel musí pro práci s těmito daty velmi dobře znát strukturu této databáze, což může být značnou nevýhodou pro běžného uživatele (Managed Dedicated Servers, 2009).

1.3.4 Objektový datový model

Je to relativně nový přírůstek ve světě databází. Tento model je konstruován pro sofistikované aplikace například z oblasti multimédií, geografických systémů apod. Umožňuje takto v určité oblasti, kde relační databáze již nedokáže věrně poskytovat obraz reality, reprezentovat složité vztahy mezi reálnými daty (Managed Dedicated Servers, 2009).

Základním prvkem objektového modelu OODBMS z anglického object oriented database management system, jak již název vypovídá, je objekt. Tento objekt má navíc kromě svých atributů definované také metody, které určují jeho chování. Každý záznam nebo-li entita objektu i objekt samotný je jednoznačně identifikovatelný identifikátorem, který je generovaný systémem. Objekty jsou pak navíc charakterizovány pomocí tříd. Tato třída je jakýmsi abstraktním popisem objektu určující datové složky a operace, které lze provádět. Pod tímto si lze představit například kontrolu záznamu při vkládání (Švec, 2003).

Podobně jako je tomu u relačních databází i zde existuje jakýsi obecný popis schémat prostřednictvím jazyka ODL a také dotazovací jazyk OQL (Švec, 2003). V této práci se však více do hlubin tohoto modelování nebudu pouštět, jelikož by tato kapitola byla velice obsáhlá a hlavně není předmětem této práce. Snahou bylo pochopit, že tu něco takového je a že lze očekávat v budoucnu vývoj v tomto směru.

Objektově orientované databáze jsou tedy vhodné pro určité oblasti použití, které tuto strukturu vyžadují, avšak nelze očekávat, že by v nejbližší době nahradily relační databázové modely, které těží především ze své jednoduchosti a efektivity (Švec, 2003).

1.3.5 Relační datový model

Jak již bylo několikrát zmíněno, tak právě relační model dat je nejpoužívanějším a nejrozšířenějším datovým modelem v současnosti a je využíván mnohými organizacemi ke správě dat. Tento model si lze jednoduše představit jako spojení několika lineárních modelů dohromady s tím, že mezi jednotlivými sledovanými objekty existují vztahy. Objekty zájmu zde představují jednoduché dvourozměrné tabulky. Vztahy mezi záznamy těchto tabulek jsou pak zřízeny pomocí identifikátorů, takzvaných klíčů (Koch, 2006).

Jelikož se tímto modelem ve své práci budu více zabývat a vzhledem k jeho obsáhlosti bude k věci jej více představit v následující kapitole.

1.4 Relační databáze

Relační model dat jako první navrhl E. F Codd ve své práci „A relations model of data for large shared data banks“. Toto byl průlom ve světě informačních technologií

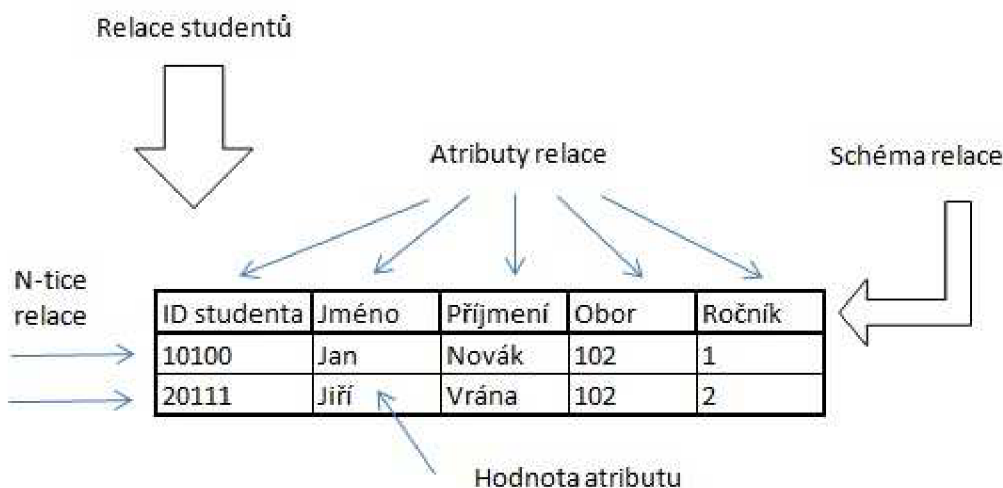
a je považován za zrod relačních databází. Nejvýznamnější výzkum lze pak také připsat firmě IBM, která vyvinula první prototyp relačního DBMS, který se vyvíjel od konce 70. let. Tento model se snaží věrohodně reprezentovat reálné prostředí reálného světa (Conolly a kol., 2009).

„Účelem modelu dat je reprezentovat data a učinit data srozumitelnými“ (Conolly a kol., 2009, s. 63).

1.4.1 Terminologie

Relační datové modely jsou založeny na matematickém konceptu relace a tato je fyzicky reprezentována tabulkou. Tato teorie dále zahrnuje teorii množin a teorii predikálové logiky (Conolly a kol., 2009).

Pro relační model dat existuje mnoho různých terminologií a tyto mohou uvádět nezasvěcené ve zmatek. Základem je, že relační model se skládá z pěti hlavních položek. Konkrétně to jsou relace, atribut, n-tice relace, hodnota atributu a relační schéma. Tyto pojmy reprezentují více známé pojmy jako je tabulka, řádek tabulky, sloupec apod. (Conolly a kol., 2009). Z obrázku č. 4 by měly být zřejmé významy těchto pojmů.



Obrázek 4: Terminologie z pohledu relací

Upraveno dle: Koch, 2006, s. 25

Existují však také jiné pohledy na tuto terminologii, jako je teorie množin a terminologie z pohledu aplikačního. V těchto případech se setkáme s n-ticí jako s entitou nebo větou, atributem jako položkou, hodnotou atributu jako údajem, nebo

relací jako tabulkou (Koch, 2006). Důležité je však vědět, že relace pro nás bude představovat tabulku, atribut pak sloupec a n-tice relace představuje záznam (Conolly a kol., 2009).

1.4.2 Vlastnosti relačních tabulek

Relační tabulky mají své charakteristické vlastnosti, kterými je lze identifikovat. Tyto vlastnosti pak vypovídají o celkové struktuře tabulky a o jejím chování. Tyto vlastnosti lze shrnout do následujících bodů (Conolly a kol., 2009):

- Každá tabulka má své jedinečné jméno v dané databázi, podle kterého ji lze identifikovat.
- Každá buňka tabulky obsahuje přesně jednu hodnotu. Jinak řečeno, data se neopakují.
- Každý sloupec tabulky je identifikovatelný svým jedinečným jménem
- Hodnoty v jednom sloupci jsou ze stejné domény
- Nezáleží na pořadí sloupců
- Neexistují duplicitní záznamy- každý záznam je jedinečný
- V teoretické úrovni nezáleží na pořadí záznamů

1.5 Integrita relačního modelu

Je třeba si uvědomit, že při modelování obrazu reálného světa nám přináší určitá teoretická omezení (Koch, 2006). „*Omezení je pravidlo, které definujeme nad jistým databázovým objektem (zpravidla nad tabulkou nebo sloupcem) a které nějakým způsobem omezuje přípustné datové hodnoty tohoto databázového objektu*“ (Oppel, 2006, s. 49).

1.5.1 Klíče

Jak již bylo zmíněno ve vlastnostech relační tabulky, tak každý řádek musí být jedinečně identifikovatelný. K tomuto slouží takzvaný klíč. V tomto případě se jedná o primární klíč, pomocí kterého se daný záznam identifikuje. Tento klíč musí být jedinečný, tedy jeho hodnota se nesmí opakovat. Dále pak rozlišujeme cizí klíč. Mějme

dvě tabulky A a B. Jednoduše lze říci, že se jedná o primární klíč z tabulky A umístěný v tabulce B (Skopal, 2010). V databázi se pak jedná o sloupec nebo skupinu sloupců v tabulce B, která odpovídá primárnímu klíči z tabulky A (Conolly a kol., 2009). Takto je pak tvořena relační vazba mezi záznamy z více tabulek (Skopal, 2010). Klíč, který splňuje vlastnosti primárního klíče, se pak nazývá kandidátní klíč. Tabulka může mít takovýchto kandidátních klíčů více, ale také nemusí mít žádný. V tomto případě je nutné vytvořit takzvaný umělý klíč, který může vystupovat například v podobě ID zákazníka apod. (Conolly a kol., 2009).

Tyto omezení pak představují referenční a entitní integritu. Pod pojmem referenční omezení lze pak chápat takové omezení, které zajišťuje relaci mezi tabulkami, tedy spojení tabulek pomocí cizích klíčů. Aby tohoto mohlo být dosaženo, je zapotřebí si uvědomit entitní integritu, která představuje omezení primárních klíčů (Oppel, 2006).

1.5.2 Integrita hodnot

Omezení integrity je omezení, které nám pomáhá v databázi zvyšovat přesnost dat. Zde definujeme zejména doménu jako množinu hodnot a specifikaci povolených hodnot pro daný atribut (Koch, 2006). Nejčastější dva typy integritního omezení jsou pak omezení typu NOT NULL a CHECK. Kde omezení NOT NULL určuje, že daný atribut nesmí zůstat prázdný, tedy hodnota tohoto atributu nesmí být NULL. Pod omezením CHECK si pak lze představit ověření platnosti hodnoty atributu. Příkladem může být jednoduchý výraz, kdy testujeme, zda zadaná hodnota je větší než nula (Oppel, 2006).

1.5.3 Vztahy mezi relačními tabulkami

Protože do databáze ukládáme údaje, které mezi sebou úzce souvisí, je třeba definovat vzájemné vztahy mezi jednotlivými entitami. Tímto zajistíme, že databáze bude správně fungovat a držet pohromadě. Rozlišujeme maximální kardinalitu, která určuje největší počet instancí jedné entity a minimální kardinalitu, která naopak určuje minimální počet těchto instancí a může být i rovna nule (Oppel, 2006). Integritní omezení pro vztahy pak určuje kardinalitu vztahu na poměry 1 : 1, 1 : N, N : 1, N : M (Koch, 2006).

Vztah 1 : 1 říká, že vždy jedné n-tici relace odpovídá jedna nebo žádná n-tice jiné relace. Příkladem takovéto relace může být člověk a občanský průkaz, kdy jeden člověk vlastní žádný nebo jeden průkaz (Koch, 2006).

Vztah 1 : N říká, že jedné n-tice relace odpovídá jedna nebo více n-tic jiné relace. Naopak n-tice druhé relace může odpovídat pouze jedné n-tici první relace. Příkladem pro tento typ vztahu pak může být člověk a jeho automobil, kdy jeden člověk může vlastnit žádný nebo více automobilů, ale jeden automobil v současnosti je vlastněn pouze jedním člověkem (Koch, 2006).

Vztah N : M říká, že jedné nebo více n-tic relace může odpovídat jedna nebo více n-tic jiné relace. Příkladem pak může být člověk a mobilní operátor, kdy jeden člověk může vlastnit účet u jednoho nebo více operátorů a jeden operátor pak poskytuje služby více uživatelům. U tohoto vztahu se vyskytuje jistá komplikace a to sice, že tato vazba nedává smysl. Řešením této vazby je tvorba nové tabulky, která bude obsahovat primární klíče z obou původních tabulek. Tato tabulka se také nazývá jako průniková entita (Koch, 2006).

1.6 Datové sklady

„Pod pojmem datový sklad (data warehouse) rozumíme předmětově orientovanou, integrovanou, časově variantní a neměnnou kolekci dat, jejichž úkolem je podpora rozhodování vedoucích pracovníků (managementu)“ (Oppel, 2006, s. 280).

1.6.1 OLAP vs. OLTP

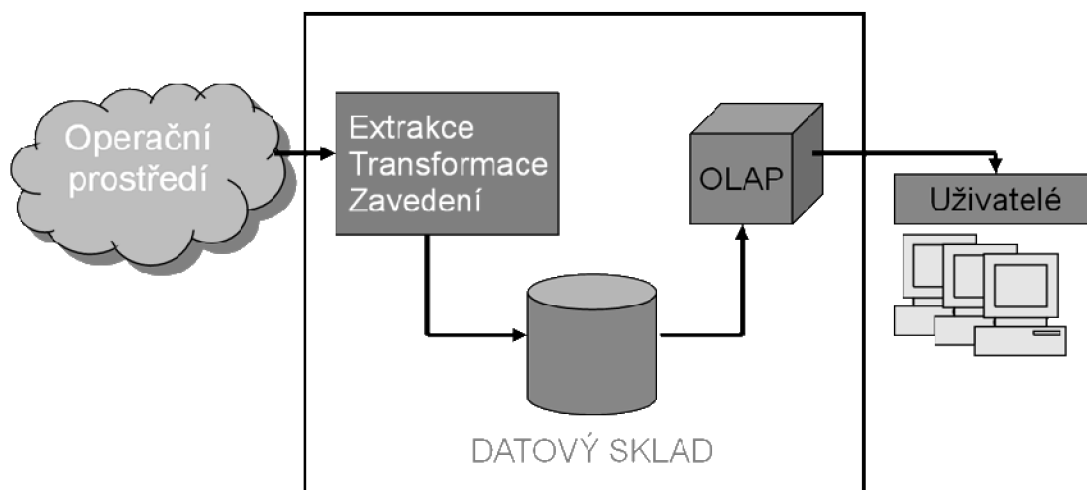
OLTP neboli Online Transaction Processing jsou systémy určené primárně pro zpracování velkého objemu transakcí. S těmito se setkáváme v běžném životě například při tvorbě objednávky zájezdu nebo zboží apod.

OLAP systémy jsou na rozdíl od transakčních databázových systémů určeny k uchování dat za co nejdelší časové období. V těchto systémech se data nemění, ale pouze se přidávají nová data a vzhledem k uspořádání těchto dat jsou vhodnější pro rychlejší a rozsáhlejší analýzy než OLTP. Vhodnější bude uvést zde tabulku, která porovnává oba zmíněné systémy.

Tabulka 1: Porovnání systému OLTP se systémy datových skladů

OLTP	OLAP
Obsahují aktuální data.	Obsahují historická data
Zaznamenávají pouze detailní údaje.	Zaznamenávají detailní údaje, ale i částečné a celkové souhrny dat.
Data jsou dynamická.	Data jsou statická, s výjimkou pravidelného rozšiřování o nová data.
Databázové dotazy je možné zpracovat za krátkou dobu a pracují s poměrně malým počtem řádků.	Databázové dotazy se zpracovávají dlouho a pracují s poměrně velkým počtem řádků dat.
V systému probíhá vysoký objem transakcí.	Počet transakcí v systému je nízký.
Opakované zpracování, předvídatelné vzorky využití systému.	Nahodilé (ad hoc) a nestrukturované zpracování dat, nepředvídatelné vzorky využití systému.
Je řízený transakcemi, podporuje každodenní činnost firmy.	Je řízený analýzou, slouží pro podporu strategického rozhodování.
Je procesně orientovaný.	Je předmětově orientovaný.
Slouží velkému počtu současně pracujících uživatelů.	Slouží poměrně malému počtu uživatelů z řad vedoucích pracovníků neboli rozhodovatelů.

Zdroj: Oppel, 2006, s. 281



Obrázek 5: Základní schéma datového skladu

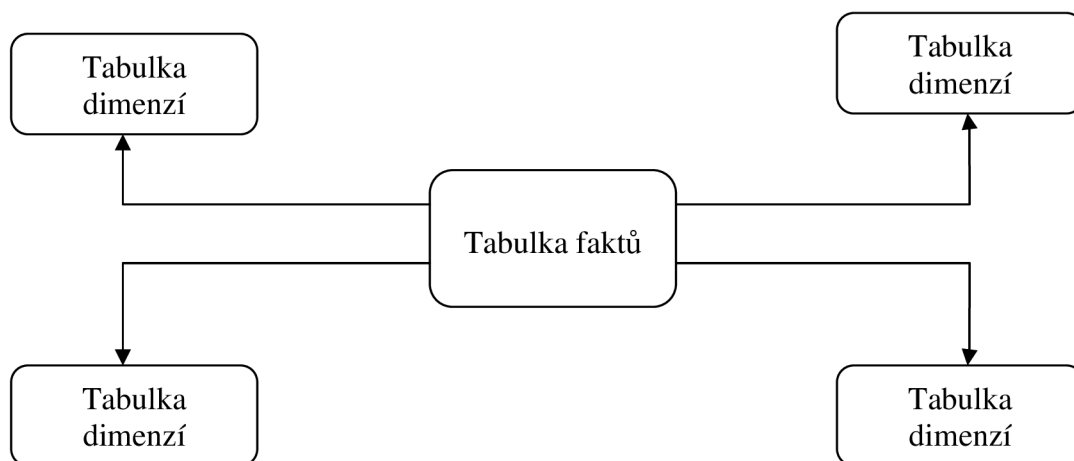
Zdroj: Bartík, 2008

Na obrázku č. 5 je uvedeno základní schéma datového skladu, které popisuje proces předávání dat. Data pocházejí většinou z různých nekonzistentních zdrojů a stejná data se často vyskytují v jiných formátech, apod. Tato data je před nahráním do datového skladu nutno upravit do vhodné podoby. K tomuto účelu zde slouží etapa ETL, neboli extrakce, transformace a loading, kdy se v jednotlivých částech této etapy provádějí potřebné korekce. Následně se tato upravená data ukládají do datového skladu. Nad těmito daty se pak provádí OLAP analýza a výsledky těchto analýz se předávají a interpretují koncovým uživatelům (Bártík, 2008).

1.6.2 Architektura datových skladů

Rozlišujeme celkem několik architektur, které se od sebe navzájem odlišují. Tyto lze rozdělit na metodu souhrnných tabulek a na metodu hvězdicového schématu (Oppel, 2006).

Architektura souhrnných tabulek pochází z původní myšlenky Billa Inmona. Tento postup využívá ukládání původních dat do datového skladu, avšak jsou dopředu vypočítány různě součty a souhrny, které by se opakovaly a je tak usnadněna a urychlena činnost analytických procesů. Klasická normalizace zde tedy ztrácí smysl. Data se do datového skladu ukládají jako série snímků a tyto se již pak nemění, ale pouze se přidávají (Oppel, 2006).



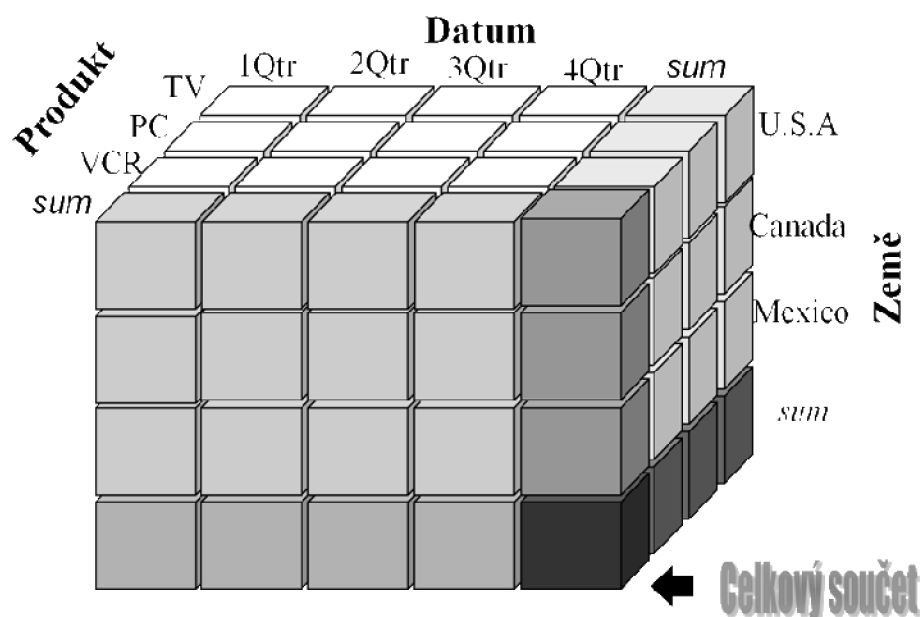
Obrázek 6: Architektura datového skladu s hvězdicovým schématem

Upraveno dle: Opperl, 2006, s. 283

Navazující architekturou na výše zmíněnou architekturu souhrnných tabulek je hvězdicové schéma, které vyvinul pro ukládání dat do datového skladu Ralph Kimball. V tomto schématu se nacházejí dva typy tabulek a to sice tabulka faktů a tabulky dimenzí. Kde do tabulky faktů se zaznamenávají převážně měrné a číselné údaje obchodování. Dále se v tabulce faktů vyskytují cizí klíče do tabulek dimenzí, které slouží jako popisy údajů v tabulce faktů (Opperl, 2006).

Další architektury, které vycházejí z hvězdicového schématu, je schéma sněhové vločky, kdy může být vzájemně provázáno více tabulek dimenzí, což u hvězdicového schématu nešlo a schéma souhvězdí, které se vyskytuje u specifických případů, kdy je potřeba více tabulek faktů (Opperl, 2006).

1.6.3 Multidimenzionální databáze



Obrázek 7: Datová kostka

Zdroj: Bártík, 2008

Tyto databáze si můžeme nejlépe představit jako krychli, tedy datovou krychli, kdy jednotlivé hrany datové krychle představují dimenze, které sledujeme (Oppel, 2006). V dnešní době existuje mnoho takovýchto systémů jako například MOLAP, ROLAP, DOLAP, HOLAP. Důležité je, aby nám tyto systémy poskytovali možnost detailního pohledu na data, sumarizační a agregační funkce, možnost pohledu na data z různých úhlů pohledu a mnoho dalších. Na obrázku je kostka, která zobrazuje tři dimenze. Kostky, které reprezentují více dimenzí, se nazývají hyperkostky. Práce s takovými kostkami není problém, ale problém se vyskytuje při zobrazování výsledků OLAP analýzy, kdy si nevystačíme s řádky, sloupci a stranami (Bártík, 2008).

Tak abychom mohli z datové kostky data číst a procházet v různých hierarchiích, potřebujeme k tomu některé základní operace. Pro posuny v hierarchii pro danou dimenzi směrem k detailní úrovni se používá operace Drill-Down a naopak vynoření k obecnější úrovni pak Roll-Up. Pokud potřebujeme přejít na jinou hierarchii definovanou nad stejnou dimenzí, využijeme operace Drill-Across. Pro přechod na úroveň záznamů a čtení konkrétních hodnot tabulky faktů slouží operace Drill-Through. Pro pohled na kostku pro jednu hodnotu jedné z dimenzí slouží operace Slice & Dice a

pro pohled na kostku podle jiné dimenze, tedy otočení kostky, slouží operace Rotation (Bártík, 2008).

1.6.4 Datová tržiště

Datové tržiště je vhodnou podmnožinou datového skladu, která podporuje specifické požadavky jednotlivých oddělení v podniku nebo činnosti firmy. Datové tržiště se tedy zaměřuje na jedno oddělení nebo proces ve firmě, neobsahuje normálně žádná data a obsahuje poměrně méně informací než datový sklad. Důvodů pro výstavbu datových tržišť je hned několik, počínaje potřebami konkrétního oddělení podniku přes omezení uživatelů až po celkové náklady na tvorbu datového tržiště (Oppel, 2006).

2 Analýza problému a současné situace

V této kapitole se zaměřím především na východiska, která budou sloužit pro podporu tvorby vlastního návrhu databáze. V úvodu představím společnost, pro kterou budu návrh tvořit. V dalších kapitolách pak budou popsány jednotlivé analýzy.

2.1 Představení společnosti

Společnost, kterou zde budu analyzovat, se zabývá technickými prohlídkami automobilů a motocyklů, mezi které spadá i měření emisí. Dále pak i podle obchodního rejstříku provozování a organizování sportovních akcí, pronájem a půjčování věcí movitých (Obchodnírejstřík.cz, 2000-2011).



Obrázek 8: Logo společnosti

Zdroj: stkkralovopole.cz, 2006

Jedná se o společnost s ručením omezeným se základním kapitálem, který činí 1 239 000 Kč. Jednatel společnosti je Ing. Jaroslav Ježek a prokuristkou Sylva Ježková (Obchodnírejstřík.cz, 2000-2011).

Stanice technické kontroly v Brně Králově poli byla založena v roce 1996. Firma STK Královo Pole, s.r.o., která se zabývá výkonem části státního odborného dozoru a podniká v oblasti poskytování služeb ve formě (stkkralovopole.cz, 1996):

- pravidelné technické prohlídky
- evidenční kontroly
- technické kontroly před schválením technické způsobilosti k provozu na pozemních komunikacích - dovozy a přestavby
- technické prohlídky na žádost zákazníka zástavby LPG

Společnost se nachází na Ulici Sladkovského 7 v Brně Králově Poli vedle Královopolské strojírně (stkkralovopole.cz, 1996).

2.2 Organizační struktura firmy

Firma jako taková je samostatná a nezávislá a nemá žádné nadřazené subjekty. V příloze č. 1 je uvedeno personální schéma firmy.

2.3 Zhodnocení podnikání firmy

V následujících podkapitolách se krátce zaměřím na podnikání firmy a její působení. Tyto poznatky budou východiskem pro tvorbu dalších analýz, které budou poskytovat dostatek potřebných informací z oblasti působení firmy pro tvorbu efektivní databáze, neboť dle teoretických základů databáze musí vystihovat co nejdříve reálné prostředí firmy a její chod.

2.3.1 Orientace na trhy

Společnost poskytuje služby ve formě technické prohlídky a s ní spojené měření emisí. Tento trh bychom si mohli představit jako lokální trh Brna města a jeho blízkého okolí, kde se tato společnost se svojí nabídkou střetává s jinými stanicemi STK v této lokalitě.

Další trh, na kterém se společnost pohybuje, je oblast poskytování služeb ve formě pořádání a organizování sportovních akcí. Tento trh již není pouze lokální, ale je celorepublikový. Podle mých zjištění se společnost v této oblasti nepohybuje na zahraničních trzích. Zde se střetává s ostatními společnostmi, které svoji činnost zaměřují na pořádání a organizování sportovních a jiných akcí.

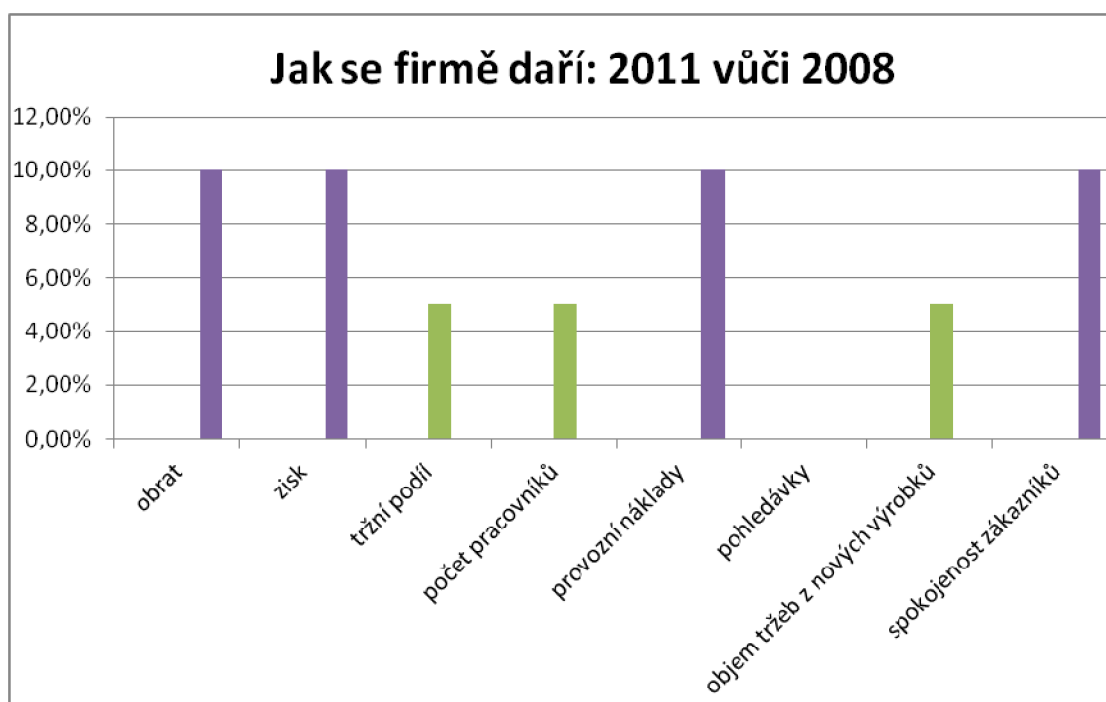
V minulosti tato společnost pořádala proslulé motokrosově akce v tehdejší brněnské hale Rondo. V současné době pořádá každoročně cyklistické závody v okolí Brna.

2.3.2 Proces financování firmy

Společnost získává finanční prostředky tak, že poskytuje služby v podobě technických prohlídek, evidenčních prohlídek, měření emisí a dalších služeb spojených s činností stanice technické kontroly. Takto získané finanční prostředky postačují na financování provozní činnosti. Společnost je v posledních pěti letech v mírném zisku a nemá žádné dluhy. Efektivně nakládá se svým majetkem.

Další finanční zdroje společnost získává svojí činností v podobě pořádání již zmiňovaných sportovních akcí. Tyto akce jsou pro společnost jistým finančním přínosem vzhledem k nákladům vynaloženým na konání těchto činností.

V grafu č. 1, který jsem sestavoval s pomocí účetní ve firmě, jsme se snažili vyjádřit, jak firmě v posledních letech prosperuje, kde na svislé ose je znázorněn nárůst v procentech, a na vodorovné ose jsou jednotlivé sledované ukazatele.



Graf 1: Jak se firmě daří v posledních letech

Zdroj: vlastní

2.3.3 Legislativní aspekty podnikání

Provedení technické prohlídky vozidla se řídí zákonem č.56/2001 Sb. o podmínkách provozu vozidel na pozemních komunikacích a o změně zákona č.168/1999 Sb., o pojištění odpovědnosti za škodu způsobenou provozem vozidla a o změně některých souvisejících zákonů (zákon o pojištění odpovědnosti z provozu vozidel), ve znění zákona č.307/1999Sb., který platí od 1.7.2001.

2.3.4 Konkurence

Společnost je proslulá svou dlouholetou tradicí a má své věrné zákazníky, kteří pravidelně využívají služeb této společnosti. Ovšem jako ve většině případů se i na

tomto trhu nachází mnoho konkurentů, kteří mezi sebou soupeří různou formou. Například to mohou být různé slevy pro věrné a nové zákazníky apod. Některé konkurenčně významné stanice zde vyjmenuji:

STK EMISE BRNO – SLATINA

AZ- STK BRNO

STK DEKRA BRNO S.R.O

STK 3756

V oblasti pořádání sportovních akcí je konkurence veliká, avšak toto není hlavní činností společnosti. Společnost přesto v posledních letech celkem stabilně organizuje každoroční cyklistické závody v okolí Brna zvané Starobrno-tour. Jak je již patrné z názvu akce, tak velkým sponzorem je pivovar Starobrno, s nímž společnost udržuje dobré vztahy.

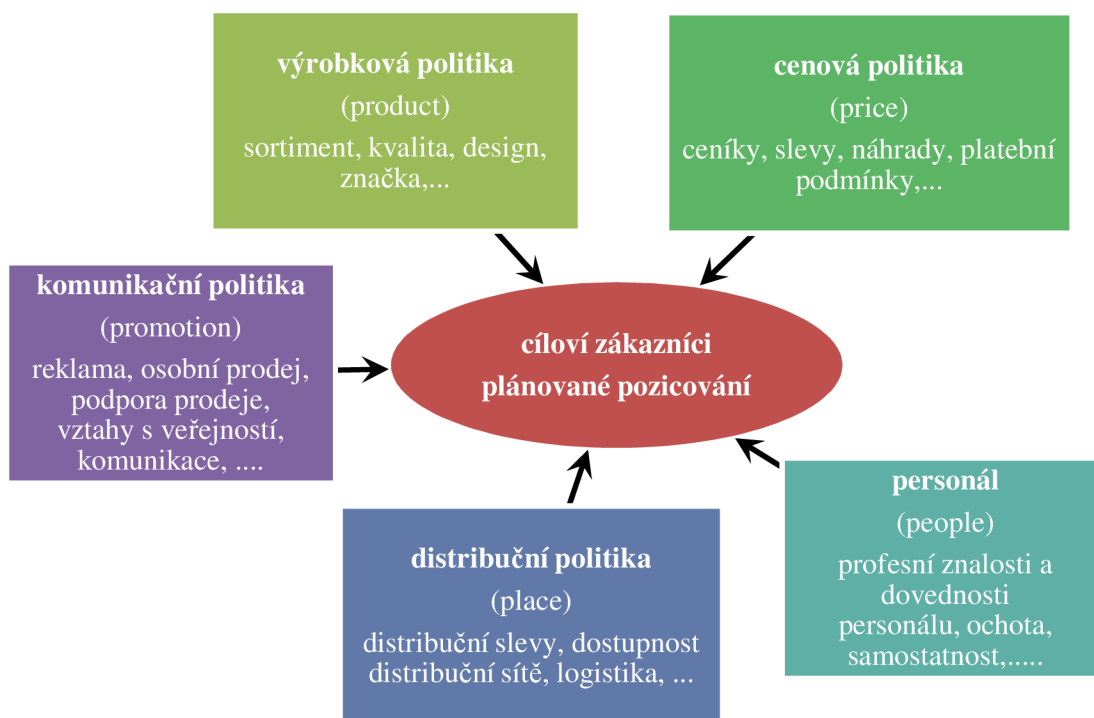
2.3.5 Hlavní výrobní zdroje a majetek podniku

Hlavními výrobními zdroji podniku jsou zaměstnanci vykonávající činnost v podobě technických prohlídek, měření emisí a služby s tímto spojené. Spíše vedlejším výrobním zdrojem jsou pak pronajatí pracovníci, většinou na dohody o provedení práce apod., pro příležitostně pořádané sportovní akce.

O společnosti se rozhodně nedá říci, že je kapitálově těžkou. Podle mých zjištění se jedná o společnost kapitálově lehkou. Většinu majetku společnosti tvoří vybavení stanice technické kontroly a stanice měření emisí. Jedná se především o různé měřicí přístroje a příslušenství s tímto spojené. Dalšími položkami v majetku společnosti jsou pak počítače, tiskárny, vybavení kancelářských prostor, pozůstalé vybavení po autoservisu, který společnost v minulosti provozovala a náhradní skladové díly, které se udržují jako spotřební materiál. Společnost dále vlastní služební automobil Škoda Felicia Pickup, který byl využíván především autoservisem a v současnosti je spíše neaktivní. Dalším automobilem je dodávka Peugeot Boxer, který je využíván především při příležitostech pořádání sportovních akcí. Prostory a budovy této stanice technické kontroly jsou v pronájmu od Automotoklubu a nejsou tedy majetkem analyzované společnosti STK Královo pole, s.r.o..

2.4 Marketingový mix

V této fázi rozeberu marketingový mix, který poslouží k bližší identifikaci prostředí firmy a jejích služeb. Tento rozbor může poskytnout cenné informace, které by mohly upřesnit některé nejasnosti při návrhu databáze. Z tohoto důvodu jsem se rozhodl pro analýzu marketingového mixu 5P se zaměřením na produkt, v našem případě toto bude představovat službu poskytování měření emisí.



Obrázek 9: Popis 5P marketingového mixu

Upraveno dle: Němec, 2001-2011

2.4.1 Produkt

V tomto případě bude hlavním produktem služba. Touto službou se rozumí provedení technické prohlídky, jejíž součástí je měření emisí. Toto měření emisí však může být provedeno samostatně bez další kontroly a kontrolu lze provést u konkurenčního subjektu nebo naopak. Nutnou podmínkou pro splnění požadavku technické způsobilosti vozidla pro provoz na pozemních komunikacích však je platná prohlídka technického stavu vozidla, pro kterou je nezbytné splnění emisních limitů, které se prověřuje měřením emisí. Pro tento případ se tedy zaměřím na tyto dva úkony jako na jednu službu, kde měření emisí pak bude rozšířením služby technické prohlídky.

Jádrem produktu je tedy samotná potřeba provozovatele automobilu dopravovat se na určitá místa dle svých potřeb automobilem, nebo převoz a doprava nákladu či zboží řidičem tohoto prostředku, motocyklem, či jiným prostředkem spadajícím do této kategorie, například nákladní automobil pro přepravu nákladů.

Formálním produktem je tedy provedení technické prohlídky na stanici technické kontroly bez měření emisí. Hlavním produktem je tedy služba v podobě technické prohlídky, kdy pokud má zkoumaný objekt splněnou prohlídku emisí, která nemusí nutně být prováděná ve stejném termínu jako je prováděná technická prohlídka a ani nemusí být prováděna u stejného subjektu, jako se provádí technická kontrola, obdrží oprávnění na další provoz vozidla. Ovšem pro provoz na pozemních komunikacích je vyžadována platnost jak technické kontroly, tak měření emisí.

Z výše uvedeného lze již vyvodit, že jakýmsi rozšířením naší formální služby, kterou je technická prohlídka, bude měření emisí. Jako další možnost formálního rozšíření služby je telefonické objednání zákazníka, kde je zákazníkovi zaručena přesná doba přijetí. Zákazník se tak nemusí potýkat se zbytečnými prostoji a čekáním.

Úplným produktem je pak služba, která zajišťuje jak technickou prohlídku, tak měření emisí, které jsou nutné pro způsobilost vozidla pro provoz.

Tento celkový pohled na tyto dvě služby jsem volil z toho důvodu, že ze statistik a z informací, které jsem se dozvěděl ve zkoumaném subjektu, vyplývá, že velká většina vozidel provádí technickou prohlídku a měření emisí současně. Tedy zanedbatelné procento nevyužije obou služeb při jedné návštěvě stanice technické kontroly.

2.4.2 Místo

Společnost se sídlem i výkonem v Brně Králově poli se svou činností provádění technické kontroly zaměřuje hlavně na Brno a jeho okolí.

Společnost má tedy strategicky výhodné místo pro výkon této činnosti i s ohledem na skutečnost, že v okolí společnosti je mnohem vyšší hustota obydlí, než je tomu u jiných lokalit (velkoměsto vs. maloměsto). Je totiž nutné, aby zákazník nebo pověřená osoba, která má potřebu využít služeb společnosti, osobně tuto společnost se svým vozidlem navštívil. Z těchto informací pak logicky můžeme usuzovat, že

potenciální zákazník z Jihlavy pravděpodobně nepojede na technickou prohlídku do Brna. Jediným problémem je, že v areálu stanice technické kontroly sídlí ještě firma Moto Forza a ta se tak podílí na obsazenosti parkoviště v areálu, které bývá často zaplněné.

2.4.3 Cena

Společnost se snaží udržovat cenovou hladinu služeb na úrovni spolu s konkurencí. Jak jsem již zmínil, tak je velice ovlivněna svým okolím, neboť v tomto okolí se nachází velké množství konkurence a potenciální zákazník z Brna či okolí pak určitě přihlédne i k cenové nabídce konkurence a tak je vhodné při tvorbě cen sledovat konkurenci.

Cena se pak liší v závislosti na druhu, typu, palivu a rozsahu prováděné kontroly automobilu. Zákazník má však možnost si svou cenu předběžně zjistit na webových stránkách společnosti, kde je uveden ceník.

Možnou konkurenční výhodou jsou pak občasné slevové akce a různé slevy pro věrné zákazníky apod.

2.4.4 Propagace

Společnost nijak výrazně do reklamy neinvestuje. Spoléhá zejména na dobrou pověst a své stálé zákazníky. Svou propagaci si zajišťuje zejména pomocí svých internetových stránek, kde zákazník najde vše potřebné pro budoucí návštěvu stanice technické kontroly. Dále zde nalezne potřebné telefonní kontakty pro možnost objednávky. Na budově při sjezdu ze silnice na most k ulici Kociánka je zřetelný billboard, který oznamuje, že za rohem se nachází stanice technické kontroly s dlouholetou tradicí. Tento billboard má jakýsi podprahový efekt, neboť kolemjedoucí ho vnímá, ale jelikož v současné době nemá potřebu využít služeb stanice technické kontroly, tak si jej příliš nevšímá. Účelem je, aby si vzpomněl v době, kdy bude těchto služeb chtít využít. V okolí tohoto billboardu projede denně mnoho automobilů, a tak si myslím, že je na účelném místě. Reklamu společnosti můžeme pak ještě najít na motocyklu světového závodníka Ondřeje Ježka, který se účastní světového šampionátu třídy Supersport pořádaného v rámci Superbikového šampionátu, a tak je tato

společnost zřetelně propagována i na některých televizních záběrech na televizním kanálu Eurosport.

Pokud by měl zákazník potřebu se na služby podrobněji dotázat nebo více informovat, nalezne rovněž na stránkách společnosti jak celkový postup přípravy na technickou prohlídku, tak i telefonní kontakty na pracovníky, se kterými se může telefonicky kontaktovat a blíže domluvit a informovat, případně objednat.

2.4.5 Personál

Personál je nedílnou součástí celého podniku. Bez těchto fungujících lidí by podnik vůbec nemohl existovat. Setkáme se zde pouze se zaškolenými pracovníky oboru znalých, a tak má zákazník jistotu odborné práce.

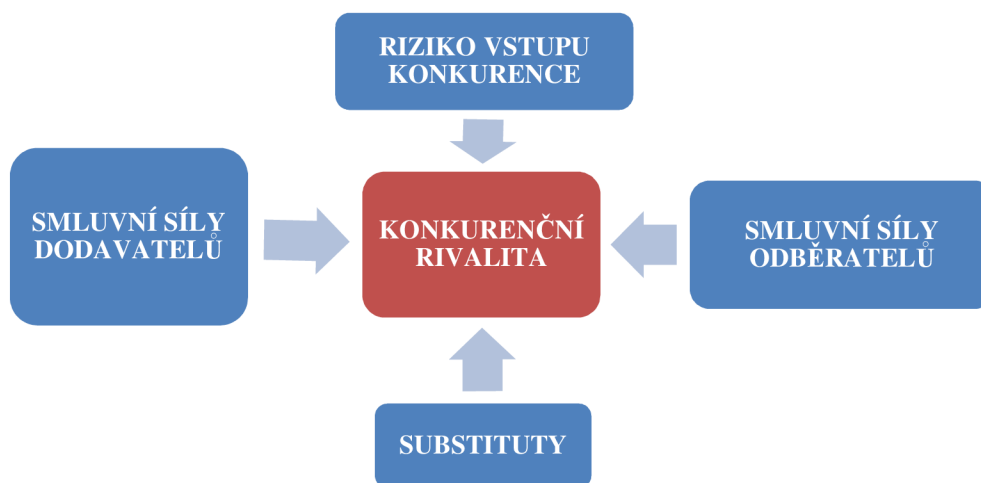
Pracovnice registračního oddělení mají velmi působivý a vstřícný přístup k řešení problémů. Jejich komunikace se zákazníkem z pohledu zákazníka je velmi přátelská a příjemná.

2.5 Silné a slabé stránky podniku

Pro analýzu silných a slabých stránek společnosti nejprve provedu analýzu Porterova modelu pěti sil, ze kterého pak budu následně vycházet při tvorbě SWOT analýzy, přičemž využiji informací z marketingového mixu. Hlavním produktem zde bude zase technická prohlídka a procesy a služby s tímto produktem spjaté.

2.5.1 Porterův model pěti sil

Obecně podíváme-li se na následující model, bude třeba si jej trochu poupravit pro potřeby analýzy této společnosti. V následujícím popisu pak uvedu hlavní aspekty této analýzy a její výsledky pro jednotlivé oblasti.



Obrázek 10: Porterův model pěti sil

Upraveno dle: vlastnicesta.cz, 2006-2009

Pod pojmem substitut, si můžeme představit službu, která by mohla současnou službu nahradit. V reálném měřítku žádná taková jiná služba, která by přesně mohla nahradit současný stav, zatím neexistuje. Jedině snad pokud chceme, tak služba konkurenční společnosti, která se zabývá technickými prohlídkami.

Tímto se dostávám ke konkurenci, kde je hrozba, že by se v okolí této stanice začala budovat stanice budoucí konkurenční společnosti. Avšak vstup na tento trh není nikterak jednoduchý a je podmíněn splněním několika různých právních předpisů a norem. Proto zde existuje určitá jistota, že jiný podnikatel se raději bude věnovat jiné činnosti. Už i z toho důvodu, že tento trh je celkem nasycen.

Ve společnosti jako takové se nevyskytují výrazné materiálové toky. V některých případech se jedná pouze o údržbu měřících a jiných zařízení souvisejících s provozem, o kterou se stará vedoucí technik, nebo technik pověřený vedoucím technikem. V jiných případech se pak může jednat o obnovu zařízení, ale vzhledem k jeho dlouhé životnosti není třeba se tímto problémem nikterak zatěžovat.

Pod pojmem odběratelé budou v našem případě vystupovat koncoví zákazníci, kteří využívají služby společnosti. Tito zákazníci jsou mnohdy velmi nároční a společnost se s tímto faktem musí vyrovnávat a částečně se přizpůsobovat a to ve smyslu objednávek a cen. Ve smyslu ovlivňování průběhu technické prohlídky musí být zaměstnanci společnosti nekompromisní, neboť toto podléhá přísnému doзору ze strany ministerstva dopravy a prohlídka se řídí zákonem (viz. legislativní aspekty podnikání).

2.5.2 SWOT analýza

Pomocí informací získaných z předešlých analýz jsem se pokusil vytvořit přehlednou SWOT analýzu, kde jsou uvedeny konkrétní vyzorované a analyzované výsledky.

Silné stránky:

- Stálá klientela
- Reklama, informovanost
- Dostupnost
- Dobrá pověst
- Možnost objednání s právem přednosti
- Aktuální ceny

Slabé stránky:

- Zastaralý hardware-možnost výpadku
- Malá kapacita parkoviště STK
- Konstrukce linky nedovoluje prohlídku větších nákladních automobilů

Příležitosti:

- Rozšíření klientely za pomoci reklamy
- Akční nabídky, slevy
- Nákup nového softwaru
- Rozšíření kapacity linky

Hrozby:

- Konkurence
- Závislost na automobilovém průmyslu

2.6 Zhodnocení běžného provozu

Po dobu pozorování, kdy jsem získával informace a podklady pro tvorbu budoucího informačního systému na podporu provozu, jsem se ve firmě neseťkal s žádnými výraznými provozními problémy. Prohlídka automobilů je velmi dobře koordinovaná. Ke každému novému zákazníkovi je přiřazen jeden technik a ten dovede celý proces do zdárného konce. Jediný potenciální problém je, že je zákonem stanovený časový limit na prohlídku automobilu a linka se tak potýká se zbytečnými prostoji.

Co se ovšem promítá do provozu je problém se zastaralým informačním systémem na evidenci zákazníků na stanici měření emisí. Tento systém neumožňuje efektivní chod v případech, kdy je zapotřebí registrovat více zákazníků během kratší doby, tedy znesnadňuje práci pracovním registračního oddělení. Následně znesnadňuje kontrolu pracovníků měření emisí seshora, tedy od vedení. Tímto problémem se dále budu zabývat v následující kapitole.

2.7 Zhodnocení informačních technologií ve firmě

V této kapitole se pokusím zhodnotit své poznatky, tak abych si připravil některé podklady pro budoucí tvorbu nového systému, který bude využíván především registračními pracovníky, kteří se budou starat o evidenci zákazníků a dalších informací potřebných pro efektivní chod a hlavně následné analýz, které bude provádět vedení firmy. Důležitým aspektem bude poznat stávající zařízení, která se nacházejí ve firmě a prostředky, se kterými se zde pracuje a pomocí kterých se komunikuje atp., tak, aby se budoucí systém vyvaroval chyb a nedostatků současného stavu.

Situace ohledně informačních technologií ve firmě by se dala zhodnotit dvěma slovy a to sice zastaralá a komplikovaná. Je také jedním z důvodů, proč jsem se společností spolupracoval, abych se pokusil se svými znalostmi získanými studiem navrhnout nějaký konkrétní postup, který by značně urychlil a usnadnil práci zaměstnancům firmy.

2.7.1 Hardware a software

Firma je osazena staršími počítači, které se zde postupně obnovovaly v letech 98 až 2011. Na většině počítačů běží operační systém Microsoft Windows XP s MS Office 2003. Na dvou počítačích, které se obnovily v posledním roce, je již operační systém MS Windows 7 s MS Office 2007.

Celkem se nacházejí dva počítače na příjmu, z nichž jeden je osazen Windows 7 a druhý Windows XP. Tyto počítače tedy slouží pro evidenci. Další tři počítače se nacházejí v kanceláři techniků, kde je také druhý z novějších počítačů. Tyto slouží technikům, kteří zde evidují technické prohlídky. Po jednom počítači je pak osazena každá ze dvou stanic měření emisí. Zbylé tři počítače jsou umístěny v kancelářských prostorách vedení společnosti.

Ve firmě se nachází hned několik tiskáren, které jsou zapojeny do firemní sítě a také tiskárny, které do sítě zapojeny nejsou. První ze síťových tiskáren se nachází na registračním pracovišti. Zde se tisknou například faktury. Druhá síťová tiskárna je umístěna v kanceláři techniků, kde se tisknou zejména protokoly o technické prohlídce. Třetí z těchto tiskáren je umístěna v kancelářských prostorách vedení společnosti. Zbývající dvě tiskárny se nacházejí po jedné na každém pracovišti měření emisí, kde se tisknou protokoly o měření emisí.

2.7.2 PC síť

Počítače v této firmě jsou propojeny klasickou LAN sítí, která je připojena k vysokorychlostnímu internetu přes router Cisco od společnosti O2, čímž je umožněna komunikace s centrální databází ministerstva dopravy. Síť jako taková je velmi pochybná, neboť je větvena od routeru do jednotlivých míst a tam dále větvena pomocí switchů, přičemž je použito kabeláže kategorie 5e, která není vedena ve zdech, ale pouze v lištách a to jen místy a není použito ani zásuvek.

2.7.3 Záloha a archivace dat

Záloha a archivace dat není dle mého názoru úplně ideální. Pro tyto účely je zde externí harddisk Zyxel s kapacitou 1 TR. Archivují se informace potřebné pro provoz firmy. Data ohledně technické prohlídky potřebná k evidenci ministerstva dopravy jsou odesílána přímo na centrální server ministerstva.

2.7.4 Informační systémy

Zde se pokusím osvětlit stav ohledně současných informačních systémů v této společnosti. Tato situace je velice složitá a netýká se pouze analyzované společnosti, ale i velké části jiných společností, které se zabývají technickými prohlídkami.

Stanice technické kontroly jako taková musí uchovávat údaje o technických prohlídkách a tyto údaje se odesílají do centrální databáze ministerstva dopravy, kde se shromažďují tyto údaje ze všech stanic technické kontroly v České republice. Pro tyto účely je využíván jediný možný systém CIS STK dodávaný ministerstvem dopravy. K tomu, aby bylo možné pro potřeby stanice technické kontroly evidovat prohlídky a vozidla a dále tvořit protokoly, existují na českém trhu další dva systémy, které slouží

jako rozšíření systému CIS STK. Tyto rozšiřující informační systémy v ČR jsou v současnosti nabízeny dvěma společnostmi a musí splňovat požadavky ministerstva dopravy, které vydalo oprávnění k užívání tohoto systému. Analyzovaná společnost užívá informační systém od společnosti Dekra s.r.o. Tento systém komunikuje s centrální databází ministerstva dopravy a v případě výpadku sítě umožňuje práci offline a údaje se aktualizují zpětně po obnovení sítě. Tento systém uchovává pouze informace o technické prohlídce a umožňuje tvorbu protokolu o technické prohlídce. Technici musejí při práci s tímto systémem evidovat kompletní údaje z technického průkazu vozidla.

Další dva informační systémy, které se zde užívají, slouží pro měření emisí. Fungují obdobně jako předchozí systém, ale s tím rozdílem, že pracují offline a protokoly o měření emisí se evidují čtvrtletně a odesílají se na ministerstvo dopravy. Spolu s tímto dochází k aktualizacím tohoto systému z důvodu obměňování vozidel. Společnost užívá na první stanici měření emisí systém od firmy Bosch s.r.o. a na druhé systém od firmy Atal spol. s r.o. Technik při práci s tímto systémem musí zadat stejné údaje o vozidle jako při technické kontrole a následně se provádí měření a výsledky se uloží do lokální databáze na měřícím počítači.

Aby byla situace ještě komplikovanější, používá společnost třetí informační systém, který má na starosti evidenci zákazníku, jejich vozidel a prováděných měření. Systém Duna od společnosti Till Consult a.s. je již staršího data a přestává vyhovovat. S tímto systémem pracují pracovníci na registračním oddělení. Poskytuje zejména cenné informace pro vedení společnosti jak z hlediska účetního, tak statistického.

2.8 Zaměření na řešený problém

Převážně jsem se zaměřil na pracovníky registračního oddělení pro stanici měření emisí. Na tomto pracovišti se nacházel zastaralý informační systém, který dle mého názoru velmi znesnadňoval práci zaměstnanců, kdy se do tohoto systému musel každý zákazník velmi zdlouhavě registrovat. Navíc tento systém neumožňoval efektivní kontrolování, celkové souhry a součty jednotlivých druhů měření a pracovníků, jež tyto měření prováděli, pro celkové měsíční a roční statistiky a výsledky hospodaření podniku, neboť je neevidoval. Jinými slovy musel sekretariát podniku všechny informace získávat zdlouhavými cestami a tyto statistiky pak provádět ručně. Ve

spolupráci s pracovníkem sekretariátu jsem tedy začal vytvářet návrh na nový databázový systém pro registrační oddělení stanice měření emisí tak, aby splňoval všechny požadavky vedení firmy. Tato práce je podrobněji popsána v kapitole č.3 Návrh řešení informačního systému firmy.

2.9 Shrnutí situace z ekonomického hlediska

I přes velkou snahu nebylo vedení firmy ochotné poskytovat důvěrnější informace a tak jsem rád, že se mi podařilo zjistit alespoň některé údaje, které jsem již uvedl. Z těchto informací je patrné, že po stránce ekonomické je na tom firma i v období krize ve stagnaci. V některých obdobích je zaznamenán nepatrný nárůst, který se však nepřehoupne přes pětiprocentní hranici. Tyto údaje jsem ještě konzultoval s účetní v podniku a lze konstatovat, že tato situace je v posledních deseti letech velmi podobná. Můžeme tedy soudit, že firmě by se i v období krize nemělo nějak značně přitížit tak, že by to ohrozilo chod firmy, neboť má sebejisté a cílevědomé vedení. Lidé budou jezdit automobily i v období krize, a to i kdyby kvůli např.: zdražení cen pohonných hmot jezdili o něco méně, frekvence technické prohlídky se tímto nijak nezmění. Dle druhu automobilu se musí na technickou prohlídku ze zákona pravidelně a to sice každé dva roky, jako je to například u osobního automobilu. Z hlediska péče o zákazníka bych se mezi pracovníky pokusil prosadit heslo "Na všechny platí stejný metr", neboť z praxe si asi každý přebere jak se věc má, když přijede na prohlídku dlouholetý kamarád nebo člověk, který komisaři nepadne takzvaně do oka. Z hlediska produktivnosti a motivace pracovníků bych neměl žádných výtek a přístup vedení se mi zdá relevantní k dané věci. Jedná se tedy o kapitálově lehkou menší společnost a nevyskytují se zde takové případy, kdy by bylo nutné nějak více zasahovat.

2.10 Shrnutí nedostatků informačních systémů

Pro účely uchovávání údajů o technických prohlídkách a odesílání těchto údajů do centrální databáze ministerstva dopravy je využíván jediný možný systém CIS STK dodávaný ministerstvem dopravy. Na tento systém existují pouze dvě možná rozšíření, která by měla usnadnit práci a umožnit společnosti provozující stanici technické kontroly uchovávat údaje. Tyto rozšiřující informační systémy musí splňovat požadavky ministerstva dopravy, které vydalo oprávnění k užívání tohoto systému a tak

není možno do těchto systémů zasahovat. Tento systém uchovává pouze informace o technické prohlídce a umožňuje tvorbu protokolu o technické prohlídce. Do tohoto systému se pokaždé opisují kompletní údaje z velkého technického průkazu vozidla. Pro potřeby stanice měření emisí musí být ve společnosti zaveden další informační systém, v našem případě dva systémy, které budou evidovat potřebné údaje. Důležitý je však rozdíl v systémech pro stanici měření emisí, tyto systémy jsou dodávány spolu s měřícím zařízením a zase se zde musí každé vozidlo kompletně registrovat dle velkého technického průkazu. Technik při práci s tímto systémem musí zadat stejné údaje o vozidle jako při technické kontrole a následně se provádí měření a výsledky se uloží do lokální databáze na měřícím počítači.

Aby byla situace ještě komplikovanější, používá společnost třetí informační systém, který má na starosti evidenci zákazníků, jejich vozidel a prováděných měření. Do tohoto systému se zase zadávají některé údaje z technického průkazu vozidla. S tímto systémem pracují pracovníci na registračním oddělení. Poskytuje zejména cenné informace pro vedení společnosti jak z hlediska účetního, tak statistického.

Situace, kterou jsem zde nastínil, se týká všech podobných subjektů v České republice a dle mého názoru je více než komplikovaná. Řešením tohoto problému by byl jeden informační systém zahrnující funkce všech těchto zmíněných komplikovaných systémů. Údaje týkající se jednoho vozidla by se tak nezadávaly do systému třikrát, jak je tomu v současnosti, ale pouze jednou. Proces by mohl probíhat následovně: Na evidenci by se zaregistroval zákazník a automobil, kdy by se vyplnily všechny povinné údaje o vozidle dle velkého technického průkazu. V jednotlivých fázích procesu prohlídky by se pouze dodaly do systému výsledky o měření emisí a výsledky z technické kontroly. Toto řešení by dle mého názoru mnohonásobně zvýšilo efektivitu a možné vyvarování se chyb. Ovšem tento problém není v mé kompetenci. Je zde třeba projít schvalovacím procesem na ministerstvu dopravy.

2.11 Návrh kroků pro další rozvoj společnosti

Pro tuto firmu je velmi obtížné navrhovat další kroky, neboť její výkonnost je omezena ze zákona (jedná se o stanovený čas na prohlídku jednoho automobilu a počet kontrolovaných vozidel v čase) a provoz již je po několika letech velmi dobře rozběhnutý. Jedině možné východisko je zaměřit se například na různé reklamní

kampaně a akce pro nové či stálé zákazníky. Dle mého názoru by bylo vhodné, aby se společnost pokusila zavést novou službu pro zákazníka. Touto službou je myšlen jakýsi předservis před prohlídkou, tak aby byl automobil pečlivě připraven na prohlídku a tím i na další dva roky spolehlivého provozu. Dalším krokem, který zde budu dále řešit je nový informační systém pro efektivnější chod stanice měření emisí, který by měl usnadnit práci, analýzy a statistiky, které jsou prováděny vedením, ale i například případné objednávky zákazníků. O tomto ale více v kapitole č.3 Návrh řešení. Určitým řešením celé situace by byl již zmiňovaný centrální informační systém, ovšem tímto by se měli zabývat pracovníci ministerstva dopravy, neboť toto není v mé kompetenci. S přihlédnutím k předchozím analýzám by se pravděpodobně vyplatilo investovat do nové počítačové sítě spolu s centrálním serverem, datovým rozvaděčem a záložním diskovým polem.

3 Návrh řešení

V této kapitole uvedu všechny potřebné informace pro vlastní návrh řešení. Bude se jednat zejména o výsledky požadavků vedení společnosti a následné konzultace a úpravy těchto údajů pro jejich možnou realizaci.

Následně se pokusím tyto požadavky realizovat vlastním návrhem řešení, který bude sestávat z databáze v prostředí MS Access 2007 a datového skladu v prostředí MS SQL Server 2008. Operační databáze v Accessu by měla sloužit k obvyklým transakcím a k tvorbě jednoduchých dotazů a sestav. Pro dlouhodobé a rozsáhlé analýzy by měl sloužit společnosti datový sklad.

3.1 Požadavky na databázi

Jak jsem již předeslal v teoretické části této práce, tak stěžejním krokem pro tvorbu efektivní databáze přesně podle požadavků zákazníka je sběr informací z prostředí analyzované firmy a konzultace požadavků na databázi se subjekty, které budou tuto databázi využívat.

3.1.1 Formulace cílů a úkolu

Zde se jedná o prvotní fázi návrhu databáze, která by měla objasnit, k čemu přesně databáze bude sloužit a poskytne tak směr pro další postup.

Po první schůzce s vedením firmy se mi dostalo několik prvních požadavků a představy firmy o tom, jak by měla tato databáze fungovat. Pro tuto část je velmi důležitá technika provádění rozhovorů. V některých případech je potřeba z účastníků informace přímo „dolovat“.

Od vedení firmy jsem získal tyto informace, ze kterých budu vycházet. Cílem databáze bude, aby sloužila k evidenci zákazníků, zaměstnanců a prováděných měření na jednotlivých vozidlech. Tato měření se pak liší podle typu a druhu vozidla, ale také podle druhu paliva. Dále by měla evidovat platby zákazníků tak, aby byla možná zpětná kontrola ze strany účetnictví, tedy kdo a kolik za co zaplatil. Dále by bylo dobré, aby databáze poskytovala informace o zaměstnancích a jejich aktivitě spojené s měřením emisí. Cílem datového skladu pak bude dlouhodobé uchování a shromažďování dat za účelem jejich archivace a provádění analýz.

Úkolem je tedy navržení databáze, která bude vyhovovat požadavkům vedení společnosti, ale bude také pohodlná na obsluhu ze strany registračních pracovníků, které s touto databází budou nejvíce pracovat.

Z těchto získaných informací je patrné, že bude potřeba udržovat informace o zaměstnancích, zákaznících a jejich vozidlech, o měření a o platbách. Tímto jsem získal základní představu o tom, co je potřeba evidovat. To ale není zdaleka vše. Pustil jsem se proto do sledování provozu. K tomu, abych mohl navrhnout fungující systém, jsem potřeboval také zjistit informace o fungování provozu jako takovém, a to jsem mohl získat pouze sledováním provozu v určitém čase.

3.1.2 Pozorování provozu a dodatečná formulace cílů a úkolu

Proto, abych zjistil podrobnější informace o údajích, které je potřeba uchovávat, jsem pozoroval po určitý čas práci registračního pracovníka při registraci vozidla.

Jakmile přijde na řadu zákazník, předloží na výzvu pracovnice velký a malý technický průkaz vozidla. Dále pak kartičku měření emisí. Pracovnice se zeptá, jestli se bude měření provádět na jméno osoby uvedené v technickém průkazu nebo na jinou osobu (pouze v případě bezprostřední změny vlastníka) a podle toho začne registrovat údaje. O zákazníkovi se registruje jeho jméno a bydliště, případně název a adresa firmy, pokud je automobil registrovaný jako firemní. O vozidle pak registruje údaje o registrační značce tedy SPZ, druh vozidla (osobní, užitkové, nákladní, motocykl), značku (např. Škoda, Seat, Volvo, apod.), typ (např. Felicia, Fabia, apod.), druh měření a druh paliva. Poté se přistupuje k samotné platbě, která se uskuteční hotovostně nebo převodem na účet společnosti. Zákazníkovi je vystavena a vytisknuta faktura o platbě.

Po dobu, co jsem tento proces sledoval, jsem si povšiml, že aktuální systém nemá žádné číselníky a neuchovává informace o zákaznících, ani o automobilech. Pracovnice tedy musí pokaždé údaje o zákazníkovi i o vozidle vypisovat znovu, přičemž jim stěžují práci chybějící číselníky. Ty by umožnili rychlejší výběr druhu automobilu, jeho typu, nebo například paliva. V případě opětovného navštívení STK zákazníkem zde chybí možnost zobrazení dřívější návštěvy či jeho údajů registrovaného zákazníka a jeho případná editace.

Tato stanice technické kontroly umožňuje také telefonické objednání zákazníka. Za dobu, co jsem sledoval provoz, však k žádné telefonické objednávce nedošlo. Nicméně pracovníci registračního oddělení zpracovávají objednávky zastarale písemným záznamem do stolního kalendáře.

Všechny takto získané informace jsou velmi cenné a poslouží k tomu, abych se vyvaroval chyb a nedostatků stávajícího systému. Z informací získaných od vedení a od pracovníků registračního oddělení jsem vytvořil následující strukturovaný seznam údajů, které je potřeba uchovávat.

- Potřeba udržovat informace o zákaznících
- Potřeba evidovat informace o vozidlech zákazníků
- Potřeba udržovat informace o zaměstnancích
- Potřeba udržovat informace o prováděných měřeních, kdo a co měřil
- Potřeba evidovat platby za tato měření
- Potřeba evidovat objednávky

3.1.3 Požadavky na datový sklad

Požadavky na datový sklad jsem konzultoval hlavně s vedením společnosti a s účetní ve společnosti, která má za úkol provádět statistické a jiné analýzy a různé sestavy.

Jejím přáním je, aby mohla data za určité období pohodlně zpracovat na počítači a získat například odpověď na otázku, který technik provedl v daném období kolik měření. Dále by pak ráda evidovala údaje o platbách za jednotlivá měření v určitém období.

3.2 Postup tvorby relační databáze

V předchozích kapitolách jsem se dozvěděl, jaké informace bude společnost evidovat a jak s těmito daty bude dále nakládat. Teď bude namístě navrhnout tabulky, jejich atributy, integritní omezení a tabulky podrobit normalizaci.

3.2.1 Identifikace tabulek a jejich atributů

Z předchozího sledování jsem tedy zjistil, že bude potřeba evidovat údaje o zákazníkovi. Z toho důvodu jsem navrhl tabulku Zakaznik, která bude sestávat z několika atributů, které jsou podle vedení společnosti nutné pro fakturaci a pro identifikaci zákazníka. Dle návrhu by tabulka Zakaznik měla obsahovat jméno, příjmení, adresu a případně název firmy. Dále ze zadání úkolu plyne, že je nutné udržovat údaje o vozidlech zákazníků. První návrh tabulky Vozidla se skládá z atributů SPZ vozidla, značka vozu, druh vozu, typ vozu, rok výroby a palivo. Údaje týkající se zaměstnanců se budou ukládat do tabulky Zamestnanci. Tato tabulka bude obsahovat atributy jako jméno, příjmení a telefon. Údaje týkající se měření, která se budou na automobilech zaznamenávat, se budou ukládat do tabulky Mereni. Tabulka Mereni by měla obsahovat atributy jako jsou druh měření, jméno zaměstnance provádějícího měření, datum a identifikaci měřeného vozidla. Pro evidenci plateb bych pak budu navrhnout navrhl tabulku s názvem Faktury, jelikož tento termín je ve společnosti více používanější než termín platby. Zde v této tabulce bude potřeba vytvořit atributy jako druh platby, datum vystavení, datum splatnosti, datum uskutečnění zdanitelného plnění a cena s DPH a bez DPH. Poslední tabulkou, která je zřejmá ze zadání, pak bude tabulka Objednávky, která bude obsahovat údaje o objednaném zákazníkovi, respektive o jeho automobilu a také datum a čas objednání.

Další tabulky, které nejsou zřejmé ze zadání, ale jsou nutné ke správné funkčnosti celé databáze, pak budou pravděpodobně číselníky. Dle zjištěných informací se bude jednat zejména o tabulku Druhy_mereni, neboť těchto druhů je jen málo a neustále by se pak opakovaly. Po dalších úvahách jsem se rozhodl vytvořit tabulku Kontakty_zakazniku, která by měla uchovávat informace o e-mailové adrese. Tato tabulka by měla eliminovat problém se zákazníky, kteří mají například více e-mailů. Tímto by se také vyřešil problém s prázdným polem v tabulce zákazník, pokud by daný zákazník neměl zřízenou e-mailovou schránku. U tabulky Zamestnanec toto podle vedení společnosti není nutné, avšak vzhledem k možnému budoucímu využití této databáze jsem se rozhodl vytvořit analogickou tabulku k tabulce Kontakty_zakazniku a to sice Telefon_y_zamestnancu.

Toto by měl být pouze předběžný návrh, který jsem dále konzultoval s vedením společnosti, abych upřesnil všechny požadavky a případně tento návrh ještě upravil.

3.2.2 Návrh modelu

Při návrhu tohoto modelu jsem začal tak, že jsem si na papír nakreslil tabulky, které jsem si v předchozích bodech identifikoval. Následně jsem začal určovat vztah mezi jednotlivými tabulkami a začal tak tvořit prvotní diagram. Postupoval jsem tabulku po tabulce a určoval jsem integritní omezení.

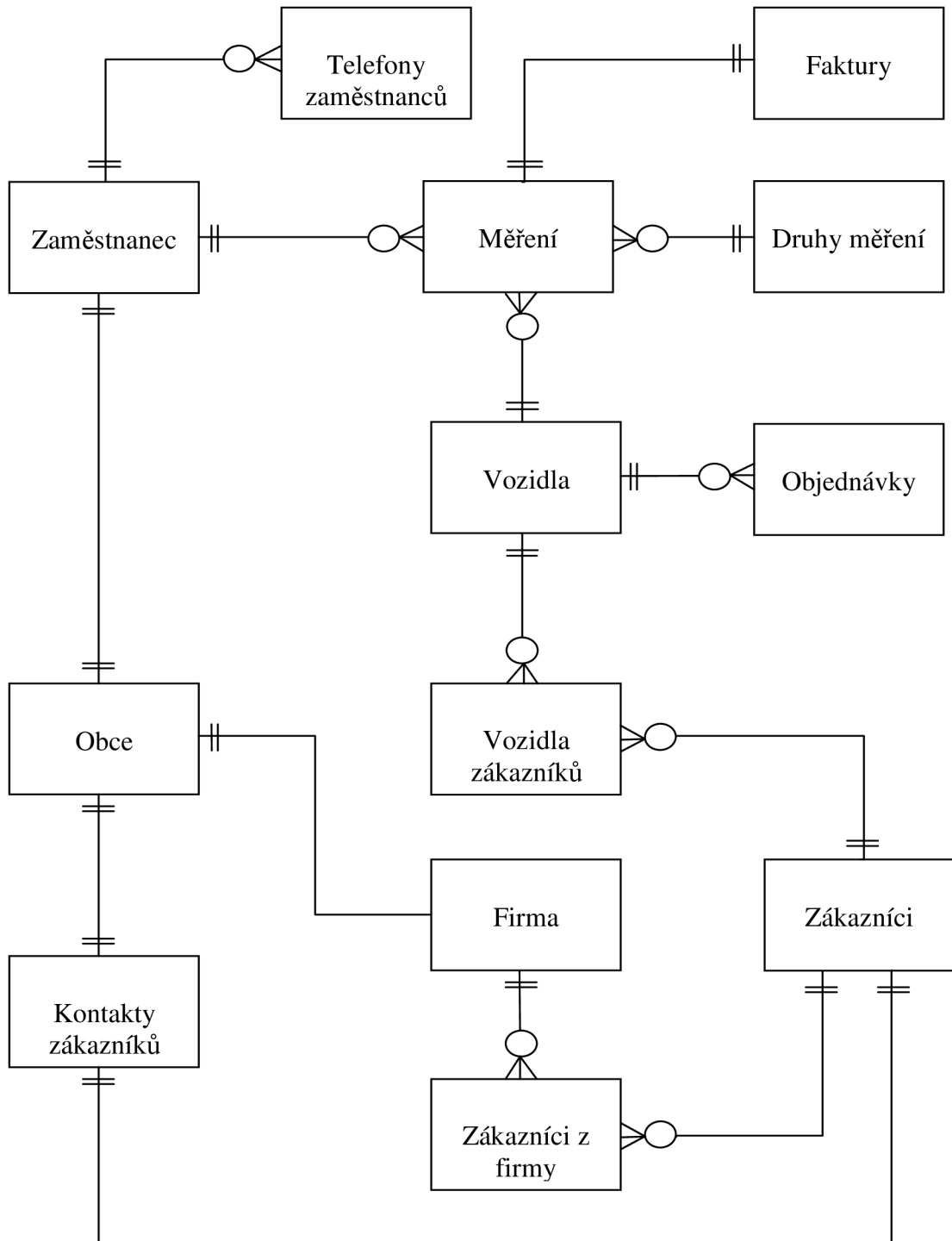
Začal jsem u zákazníka a položil si otázku: kolik může mít zákazník vozidel? Odpověď na tuto otázku je, že nemusí mít žádné vozidlo, nebo může mít vozidel několik. Situace, kdy zákazník nebude mít žádný automobil, může nastat v případě, kdy zákazník automobil prodá jiné osobě. Z těchto důvodů bude dobré si ponechat data o zákazníkovi, i když nemá žádný automobil, neboť v případě, kdy si pořídí nový automobil a následně navštíví společnost s jiným automobilem, bude proces zadávání informací kratší a rychlejší o již uchovaná data. Naopak tomu pak může být u vozidla. Druhou otázkou si pak položím ze strany vozidla a ta bude znít takto: kolik může mít vozidlo majitelů, respektive osob, na jejichž účet se provádí měření? Odpověď na tuto otázku je, že několik. Tímto mi vznikla vazba M:N, kterou dekomponuji pomocí umělé tabulky a to sice tabulky Vozidla_zakazniku, která bude obsahovat cizí klíče do tabulek Vozidla a Zakaznici. Primárním klíčem této tabulky pak bude složený klíč z těchto cizích klíčů.

Dále jsem postupoval z vozidel na měření a analogickým způsobem jsem určil vazbu 1:N. Kde jedno určité měření lze provádět na jednom a právě jednom vozidle. Naopak jsem vyvodil, že na jednom vozidle může být prováděno žádné nebo více měření.

Tabulka Mereni pak bude s číselníkem Druhy_mereni tvořit vazbu 1:N, neboť jedno měření se rovná právě jednomu druhu měření. Jeden druh měření pak může být použit u více měření.

V případě zaměstnance a měření, které zaměstnanec provádí, jsem určil vztah 1:N. Jeden zaměstnanec v současnosti může provádět žádné nebo více měření, ale jedno měření smí být prováděno pouze jedním a právě jedním zaměstnancem.

Faktury, tedy údaje týkající se plateb, se pak budou vztahovat k tabulce Merení, neboť nás zajímá, zda toto měření bylo zapláceno. Zde se bude jednat o vztah 1:1, jelikož jedna platba se týká pouze jednoho měření a jedno měření se týká pouze jedné platby.



Obrázek 11: Entito-relační diagram transakční databáze

Zdroj: Vlastní

Po důsledném prozkoumání návrhu jsem zjistil, že je zapotřebí přidat tabulku Firma tak, aby model splňoval normalizační podmínky. Tuto tabulku jsem propojil s tabulkou Zakaznici pomocí umělé tabulky Zakaznici_z_firmy, tak abych dekomponoval vzniklou vazbu M:N.

3.2.3 Klíče

Zde rozeberu určení jedinečných identifikátorů, pomocí nichž bude možné identifikovat záznam, takzvaných primárních klíčů. Ale také zde určím cizí klíče, pomocí nichž bude možné spojit tabulky relací.

U tabulky zaměstnanci bude potřeba určit umělý klíč, neboť žádná položka z této tabulky nemá vlastnosti kandidátního klíče. Jako klíč u tabulky Zamestnanci určím tedy ID_zamestnanec. Obdobně jsem takto musel vytvořit umělý klíč u dalších tabulek a to sice těchto: Zákazníci (ID_zakaznik), Měření (ID_mereni), Druhy_mereni (ID_druh_mereni), Objednavky (ID_objednavky). Specifickým pro mě byla tvorba klíče u tabulky Vozidla, kde by se dalo uvažovat a možných kandidátech na primární klíč a to položkách SPZ, Cislo_motoru nebo Cislo_karoserie. S přihlédnutím na okolnosti, které mě přesvědčily o tom, že ne každý automobil musí mít registrační značku (toto se vyskytuje u specifických případů například u dovozu a podobně), ale také nemusí být vždy určitá shoda u čísla motoru, jako je tomu například při výměně motoru nebo výmazu jeho čísla na úřadě, nebo možnost, že by se vyskytlo duplicitní číslo karoserie, což se nedá vyloučit, jsem se rozhodl i zde vytvořit umělý primární klíč ID_vozidla. V tabulce Vozidla_zakazniku pak bude tvořit složený primární klíč ID_vozidla, zároveň jako cizí klíč z tabulky Vozidla a ID_zakaznik jako cizí klíč z tabulky Zakaznici. Obdobně tomu pak bude u tabulek Kontakty_zakazniku (ID_zakaznik, Email, Ulice, PSC) a Kontakty_zamestnancu (ID_zamestnanec, Telefon). U tabulky Firma jsem určil jako primární klíč atribut IČO firmy, jelikož tento atribut je pro každou firmu jedinečný. Tabulka vzniklá dekompozicí vazby mezi zákazníkem a firmou pak bude obsahovat jako primární klíč své dva atributy (ID_zakaznik a ICO). Určitým způsobem složitý případ nastal u tvorby primárního klíče ID_faktury. Takzvané číslo faktury se řídí určitým řádem, který je již ve společnosti zaběhnutý a jeho strukturu již nelze měnit. Tento identifikátor se skládá z písmene označujícího, že se jedná o fakturu, číselné řady a data vystavení faktury, přičemž všechny tyto položky

jsou oddělené mezerou. Tuto posloupnost znaků tedy použiji jako primární klíč a jedinečný identifikátor tabulky Faktury. Poslední tabulkou, u které určím primární klíč, je tabulka Obce, kde poslouží jako primární klíč PSC.

3.2.4 Normalizace

Abych splnil podmínky první normální formy, musel jsem odstranit vícehodnotové a složené atributy. Toto se týkalo především položky Adresa, kterou jsem rozložil na položky Ulice, Obec a poštovní směrovací číslo (dále jen PSČ). Ale také se tato normalizace vztahovala například na kontakty zákazníků a zaměstnanců, tyto jsem rozdělil do samostatných tabulek.

Druhá normální forma říká, že všechny položky tabulky musí být závislé na celém primárním klíči. Tato situace by nastala v případě, kdybych měl například v tabulce Mereni položky z tabulky Druhy_mereni. Této situaci jsem však již předešel v samotném počátečním návrhu, nicméně jsem překontroloval všechny tabulky, zda-li splňují požadavky druhé normální formy.

Jakmile jsem měl splněny požadavky druhé normální formy, začal jsem zkoumat normu třetí. Tato norma říká, že všechny neklíčové atributy musí být vzájemně nezávislé. Číselník, který není tak zřejmý, ale využiji ho, bude číselník týkající se PSČ a obcí. Takovýto číselník podpoří tuto normalizaci, a to sice tím, že vyloučí takzvanou tranzitivní závislost, kdy na sobě závisí PSČ a obec. Tady poslouží jako primární klíč poštovní směrovací číslo atribut PSC. Do konfliktu jsem se dostal ještě v tabulce Zakaznici, kde v případě, že automobil je psaný na firmu, bude adresa firmy závislá na názvu firmy. Tento problém jsem řešil dekompozicí této tabulky tak, že jsem vytvořil novou tabulku Firma, která bude obsahovat položky týkající se kontaktní adresy, které jsou následně důležité pro fakturaci. V této tabulce bude tvořit primární klíč IČO firmy a s tabulkou Zakaznici bude propojená pomocí cizího klíče v tabulce Zakaznici_z_firmy a to sice ICO_firma a ID_zakaznik, která vznikla dekompozicí vazby mezi zmíněnými tabulkami. Tato tabulka pak bude obsahovat jako primární klíč své dva atributy (ID_zakaznik a ICO). U tabulky Firma jsem tedy určil jako primární klíč atribut IČO firmy, jelikož tento atribut je pro každou firmu jedinečný.

V souvislosti s tímto postupem došlo k odstranění primárního klíče ID_druh_mereni z tabulky Druhy_mereni. Jako nový primární klíč jsem zvolil položku

Nazev_mereni, která, jak jsem zjistil, splňuje specifické podmínky kandidátního klíče. Spolu s tímto však došlo ještě k úpravám tabulky Zakaznici a Kontakty_zakazniku z důvodu výskytu dvou případů, a to, že automobil je psaný pouze na firmu, nebo pouze na zákazníka.

Variaci třetí normální formy tzv. Boyce-Coddovy normální formy jsem prozkoumával, avšak ne všechny tabulky splňují požadavky, jako jsou minimálně dva kandidátní klíče v relaci, nejméně dva z kandidátních klíčů jsou složené a to, že se kandidátní klíče musí v některých attributech překrývat. Tuto formu tak nebylo možné realizovat také z důvodu nevhodnosti pro tento model. Výsledný návrh tabulek transakční databáze, který už obsahuje primární klíče a podléhá normalizaci, uvádím na obrázku č. 12.

Zamestnanci	Telefony_zamestnancu	Faktury
ID_zamestnanec Jmeno Prijmeni Ulice PSC	ID_zamestnanec Telefony	ID_faktury ID_mereni Druh_platby Datum_vystaveni Sazba_DPH
Obce	Mereni	Druhy_mereni
PSC Obec	ID_mereni Nazev_mereni ID_zamestnanec Datum_mereni ID_vozidla	Nazev_mereni Cena_mereni
Kontakty_zakazniku	Vozidla_zakazniku	Vozidla
ID_zakaznik Email Ulice PSC	ID_vozidla ID_zakaznik	ID_vozidla SPZ Druh Znacka Typ Palivo Cislo_karoserie Cislo_motoru Typ_motoru Rok_vyroby
Firma	Zakaznici	Objednavky
ICO Ulice PSC Email Název	ID_zakaznik Jmeno Prijmeni	ID_objednavky Datum Cas ID_vozidla
	Zakaznici_z_firmy	
	ID_zakaznik ICO	

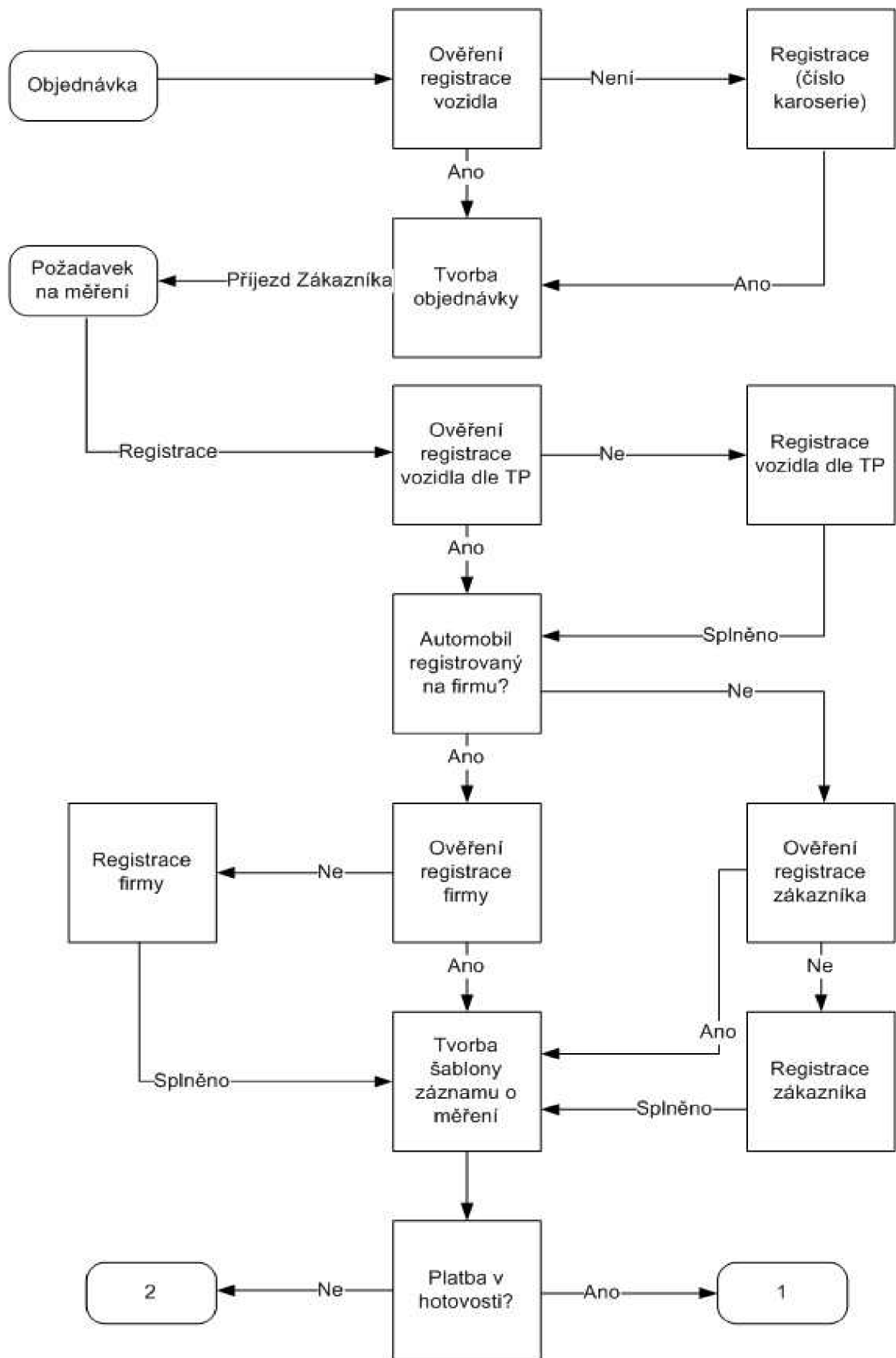
Obrázek 12: Tabulky transakční databáze

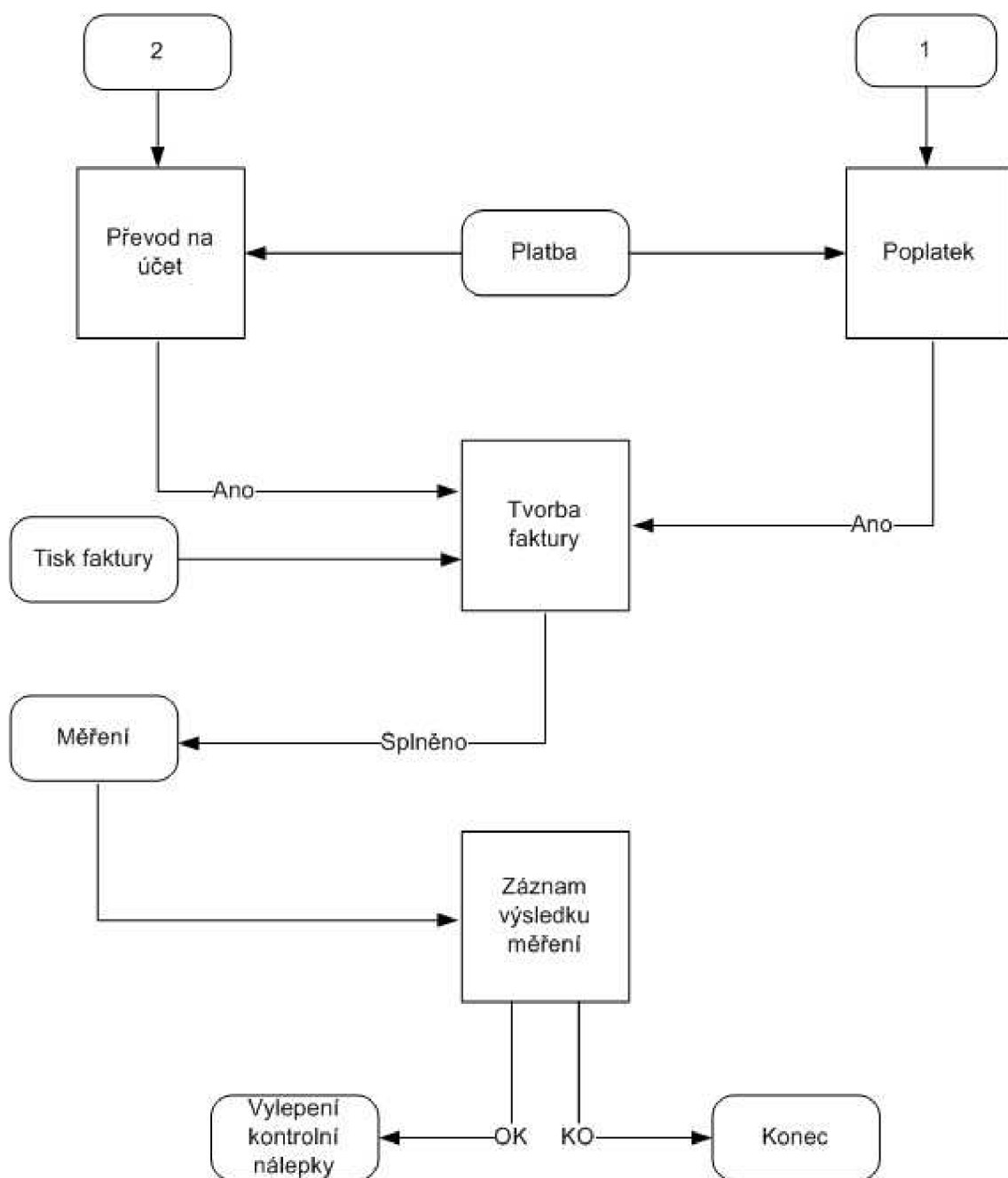
Zdroj: Vlastní

3.2.5 Tvorba obslužných procedur databáze

Pro jednoduchou obsluhu zejména ze strany registračních pracovníků, ale i tvorbu jednoduchých dotazů a sestav jsem se rozhodl využít nástrojů programovacího jazyka VBA, s jehož pomocí jsem pohodlně naprogramoval kompletní obsluhu databáze i s kontrolou integrity. Využívání kontroly Accessu jsem musel pro využití VBA částečně vyřadit tak, abych vše mohl ovládat pomocí programového kódu, který jsem tvořil ve vývojovém prostředí VBA v Accessu.

Tabulky, jejich atributy a vztahy jsem tvořil přímo v Accessu a to i přes to, že VBA touto možností disponuje, tato volba mi přišla příznivější a rychlejší. Jakmile byla databáze přichystaná, pustil jsem se do tvorby zadávacích formulářů a jejich obslužných procedur, kterých nebylo málo. Ze všeho nejdůležitější je si uvědomit, jakým směrem se budou vyvíjet jednotlivé procesy. Pro snadnou orientaci jsem vytvořil procesní diagram, který zde uvádím na obrázku č. 13. Z tohoto diagramu by mělo již být patrné, že se bude muset vytvořit nějaké uživatelské prostředí pro pohodlné zadávání a spravování údajů.

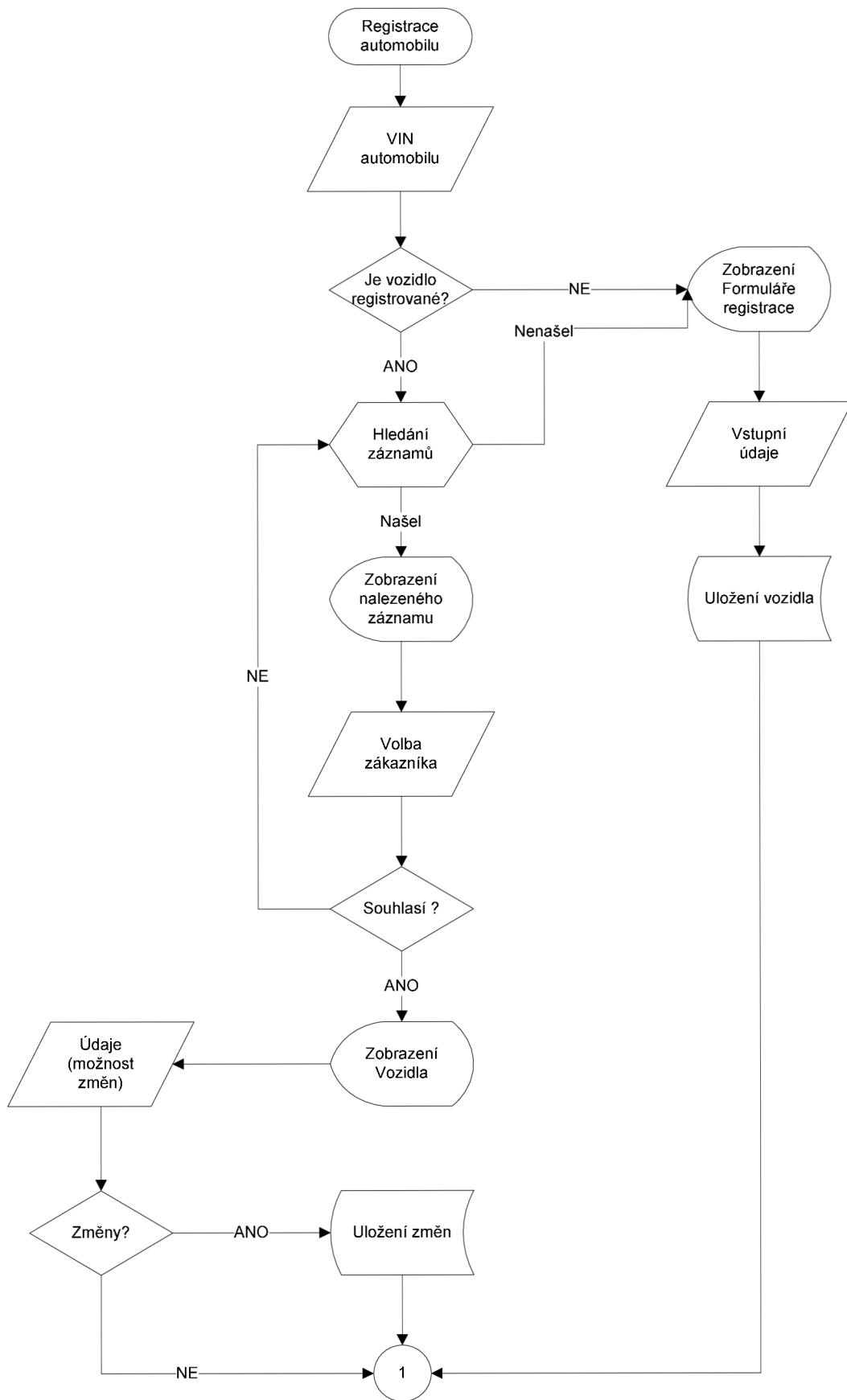


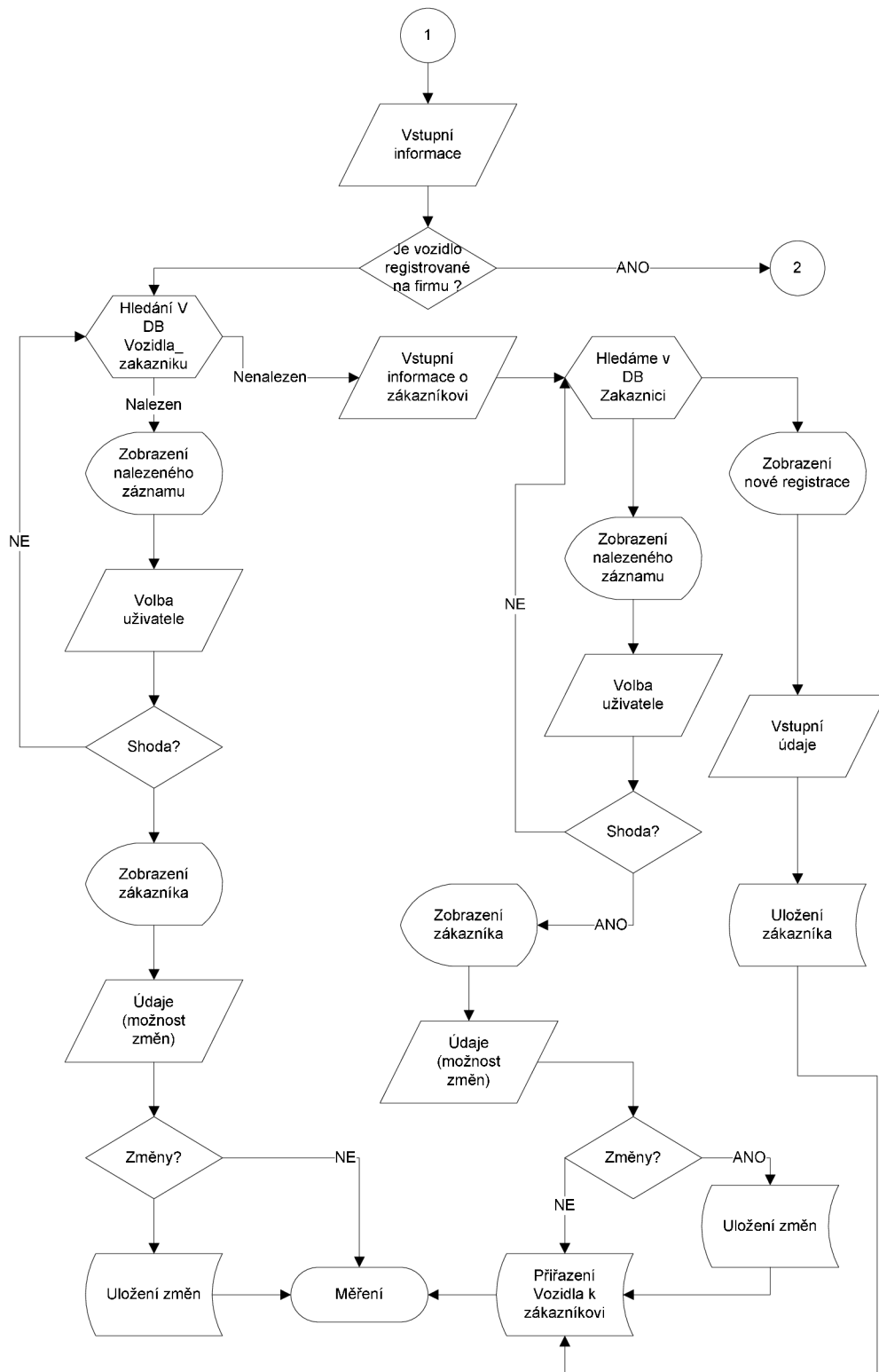


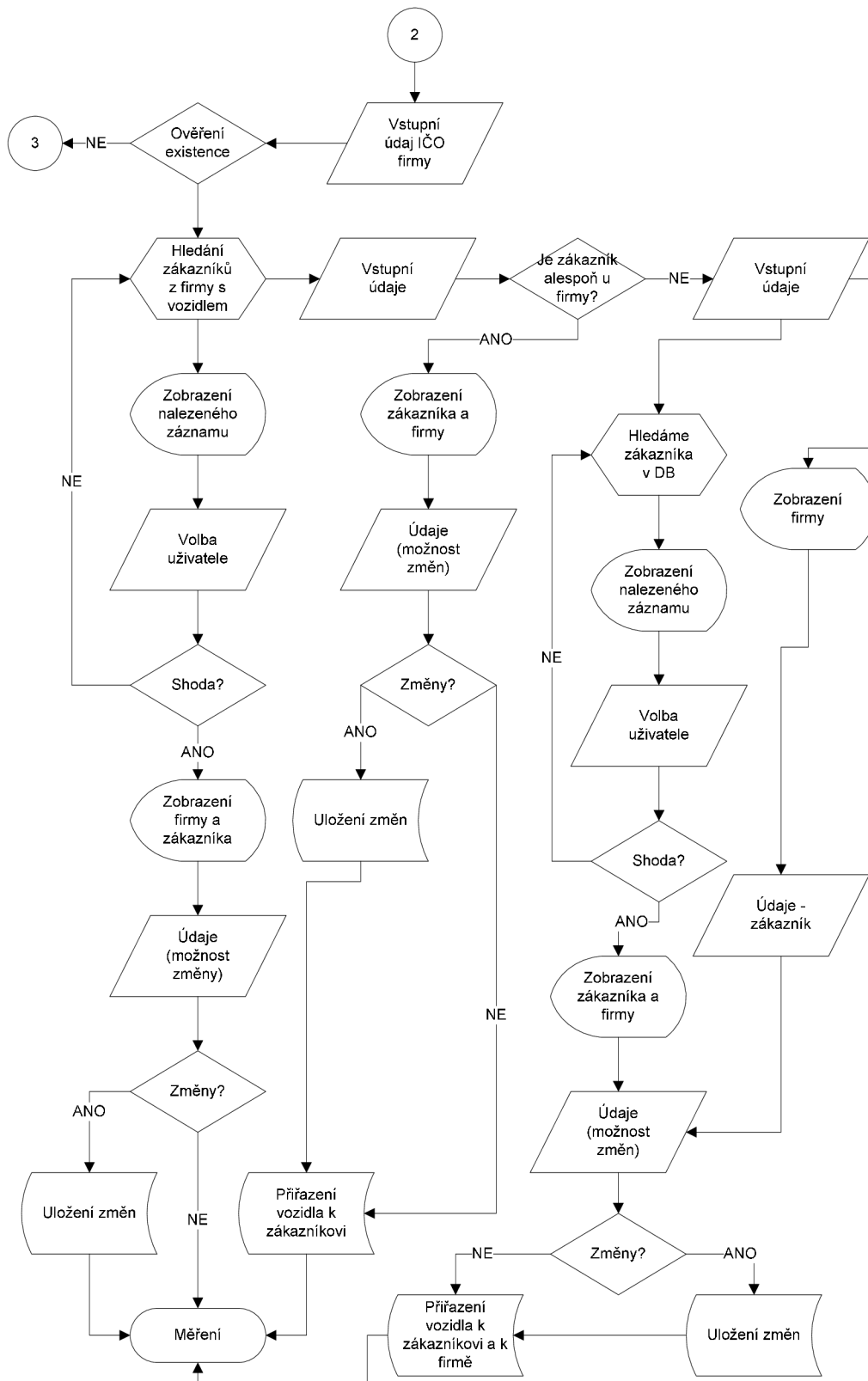
Obrázek 13: Procesní diagram transakční databáze

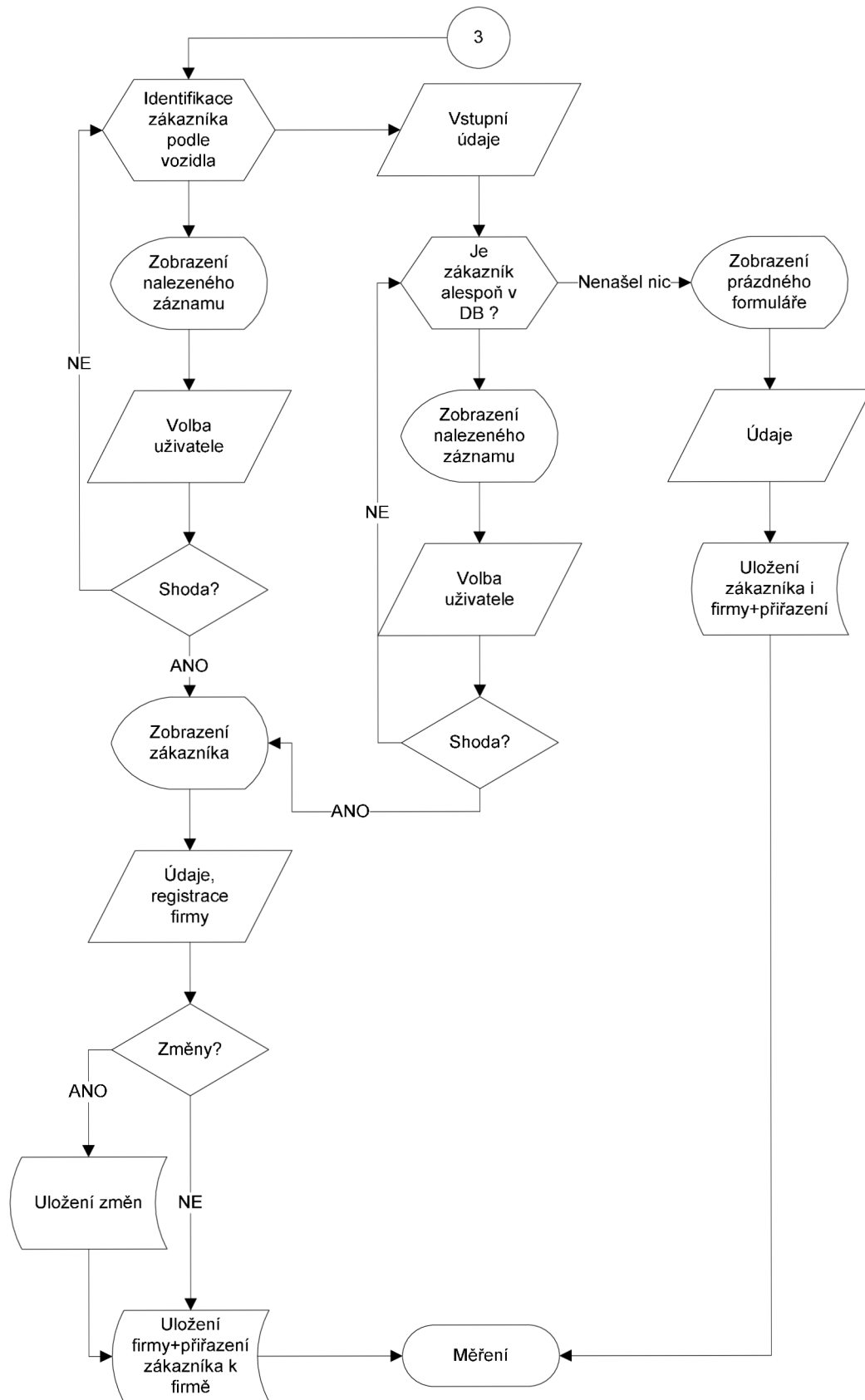
Zdroj: vlastní

Tento model je však poměrně hodně zjednodušený, a tak došlo na tvorbu vývojového diagramu, který se týká registrace vozidla a zákazníka, ze kterého již bude zřejmé větvení programu. Z vývojového diagramu je pak patrné, že může nastat několik možností při registraci jak zákazníka, tak i vozidla, které je nutno prozkoumat, abychom se vyvarovali například duplicitních nebo chybějících hodnot.









Obrázek 14: Vývojový diagram transakční databáze

Zdroj: vlastní

V následujících odstavcích popíši některé vybrané a důležité procesy.

Proces tvorby objednávky se uskutečňuje na formuláři Objednávky vozidla, kde je hlavní vstupní informací Číslo karoserie vozidla tedy VIN (dále jen VIN). Po jeho zadání systém zkontroluje, zda je vozidlo v databázi registrované nebo ne. V prvním případě se vozidlo zobrazí a dále se zadává objednávací doba pomocí jednoduchého kalendáře, který se neustále aktualizuje podle kapacity termínů. Na jeden termín pak lze objednat maximálně dvě vozidla, což je dáno kapacitou linky. Ve druhém případě se postupuje obdobně, ale pouze se vygeneruje identifikátor vozidla a uloží se pouze VIN a tento identifikátor, ostatní údaje se doplní až při klasické registraci na oddělení po příjezdu zákazníka.

Registrace Vozidla a zákazníka je nejdůležitějším procesem celého systému. V zásadě se teď budu opírat o procesní a vývojový diagram. Z diagramů je patrná situace objednaného a neobjednaného vozidla, ale to zde dále rozebírat nebudu. Samotná registrace začíná zadáním VIN karoserie do systému, kde se zjišťuje existence tohoto vozidla v databázi. Jestliže vozidlo existuje, dojde k jeho zobrazení na samostatném formuláři. Nyní má uživatel možnost některé údaje změnit, například dojde-li ke změně vlastníka apod. Následně se zkoumá, zda byly provedené změny a pokud ano, systém uživatele informuje o těchto změnách a s jeho souhlasem je uloží. Ve druhém případě uživatel zadá kompletní požadované údaje o vozidle a dále se v obou případech přistoupí k registraci zákazníka. V této fázi se systém větví na dvě větve, a to sice případ, kdy je automobil registrovaný na firmu, a nebo je registrovaný pouze na osobu. Specificky se pak celá procedura liší pokud jde o případ registrace nového vozidla a nového zákazníka a tyto možné kombinace. Na koho je vozidlo registrované se zjistí z informací zadaných do systému a v případě, že se jedná o vozidlo registrované na firmu, tak se zadává údaj IČO firmy. S pomocí tohoto údaje se dále v systému pracuje při hledání firmy. Nejprve se zjišťuje, zda je v databázi firma, u které je registrovaný nějaký zákazník, který je zaregistrovaný k uvažovanému vozidlu, tedy pokud vozidlo v databázi existuje, v opačném případě se postupuje odlišně. Pokud je záznam nalezen, tak se uživateli zobrazí a ten má možnost identifikace jestli se doopravdy jedná o hledaného zákazníka z dané firmy a pokud jej neidentifikuje, tak se hledá další záznam, který by odpovídal požadovaným parametrům až do doby, kdy jsou všechny záznamy vyčerpány. Jestliže uživatel identifikuje a označí hledaný záznam,

dostane se do další fáze, kdy se na dalším formuláři zobrazí firma a zákazník, kde má opět možnost editace. Dále se prozkoumá, zda nedošlo ke změnám, a pokud ano, tak se tyto změny uloží, jinak se pokračuje dále na proces měření. Tímto jsem teď naznačil, jak se zmíněný proces vyvíjí a jak se program chová v různých případech zmíněné situace, která je k nalezení ve vývojovém diagramu na jeho třetí straně jako první levá větev na stránce. Ostatní větve ani některé jiné procesy zde již nebudu rozepisovat, ale z předešlého popisu by měly být další větve vývojového diagramu snadno pochopitelné.

Po úspěšném dokončení procesu registrace se do nového formuláře vyplní všechny potřebné údaje a tyto se uloží, zde se již nic neaktualizuje. Údaje týkající se měření se do tabulky měření pouze přidávají. Dále se na formuláři nacházejí platební údaje, které se ukládají do tabulky faktury. Po zadání druhu měření se dopočítají hodnoty všech cen. Pokud je již vše vyplněno, údaje se uloží a postupuje se dále na tisk faktury, načež následuje ukončení celého procesu.

3.2.6 Formuláře – uživatelské prostředí

Pro tvorbu formulářů jsem využíval prostředků aplikace MS Access, jako jsou textová pole a jejich popisky, pole se seznamem, předdefinovaná pole, zaškrtačací políčka, přepínače, tlačítka atd. Některá textová pole dynamicky reagují na události, jako je tomu například u objednávek, kdy se při postupném výběru dne objednávky automaticky zobrazí v seznamu pouze volná místa. Jako příklad zde uvádím na obrázku č. 15 formulář Vozidla, kde jsou vidět některá předdefinovaná pole, ve kterých lze jednoduše vyhledávat pomocí zadání počátečního písmene na klávesnici po aktivaci daného pole. Nezbytností je pak tabulátorové přepínání mezi jednotlivými zadávacími poli. Některé další ukázky formulářů uvádím v příloze č. 2 Formulář Zákazníci a příloze č. 3 Formulář Mereni.

Vozidla

ID vozidla	1329668103	Číslo motoru	W894L5V1B45697BE
SPZ	BOC 63-29	Typ motoru	2.0 T D
Druh	<div style="border: 1px solid black; padding: 2px;"> Osobní Užitkové Nákladní </div>	Palivo	<div style="border: 1px solid black; padding: 2px;"> Benzín Nafta LPG CNG Ethanol </div>
Značka	<div style="border: 1px solid black; padding: 2px;"> Aero Alfa Romeo Audi BMW Citroen Dacia Daewoo Daihatsu </div>	Číslo karoserie	4455
Typ	A3	Rok výroby	2009

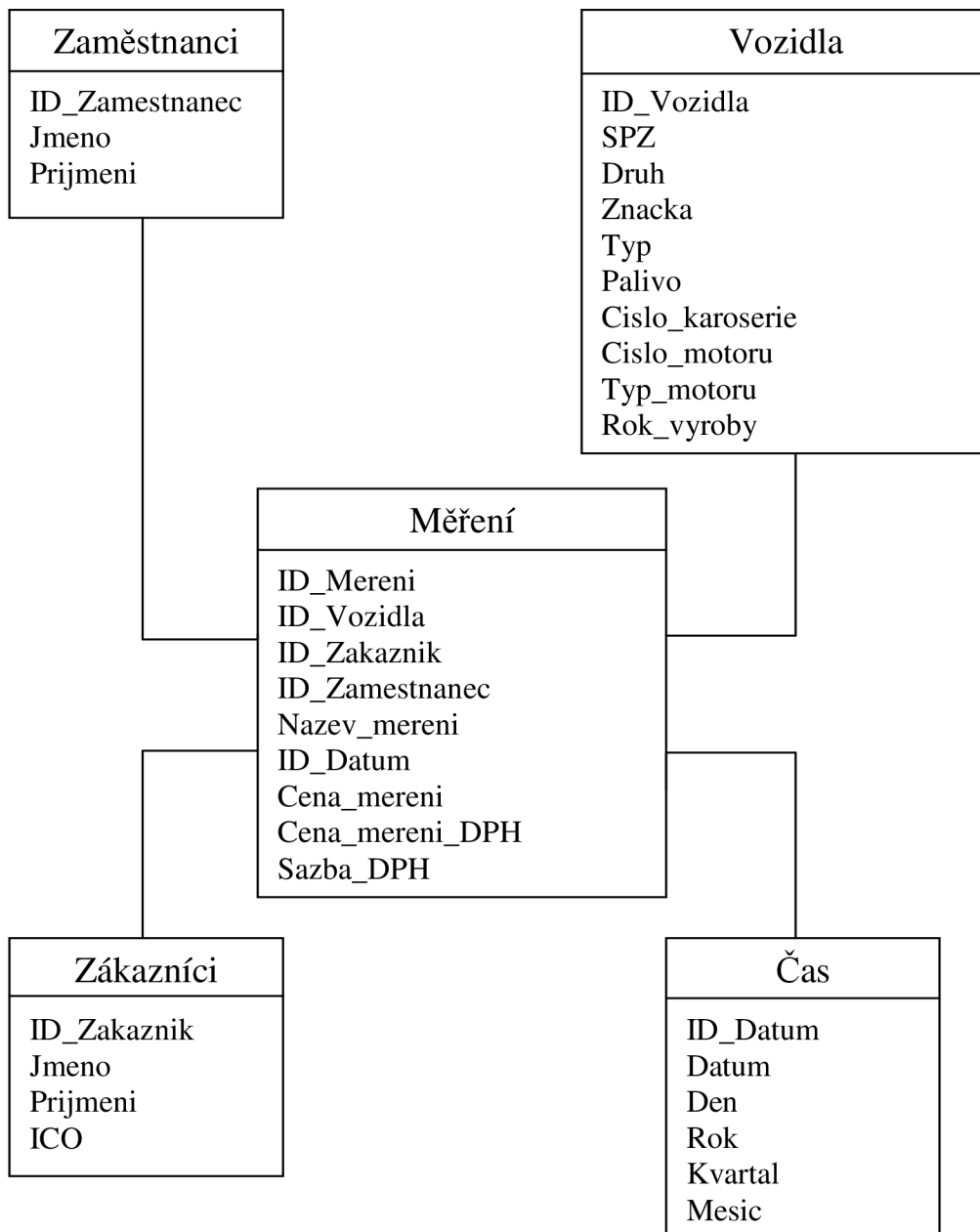
Nový záznam
Pokračuj
Konec

Obrázek 15: Formulář Vozidla

Zdroj: vlastní

3.3 Postup tvorby datového skladu

Datový sklad vychází z požadavků vedení společnosti na uchovávaná data a provádění analýz s těmito daty. To se bude hlavně týkat zobrazování dat z pohledu vozidel, zaměstnanců a času. Datový sklad se bude skládat z jedné tabulky faktů, kde budou uložena číselná data týkající se měření a ze čtyř tabulek dimenzí. Tabulky datového skladu tvoří schéma hvězdy, které uvádím na obrázku č. 15.



Obrázek 16: Schéma datového skladu

Zdroj: vlastní

3.4 Předpokládané přínosy návrhu řešení

Na přínosy mého návrhu řešení lze nahlížet z několika různých úhlů pohledu. Avšak pro přehlednost zde uvedu strukturovaně nejdůležitější skutečnosti, které by při správném využití mého řešení měli být splněny.

- Uchovávání důležitých údajů v datovém skladu a jejich možné analytické využití.
- Efektivní a rychlá registrace vozidla a zákazníka (úspora pracovního času)
- Přehledné grafické uživatelské rozhraní
- Evidence pohledávek
- Snadná obsluha transakční databáze (komplexní průvodce)
- Efektivní zpráva objednávek

Závěr

Tato bakalářská práce pojednává o návrhu transakční databáze, jejímž hlavním cílem je usnadnit a zefektivnit práci zaměstnancům a vedení společnosti, pro stanici měření emisí. V mé práci se mi podařilo takovou databázi vytvořit v prostředí MS Access za použití programovacího jazyka VBA. Výsledkem této databáze je snadná registrace zákazníků, jejich vozidel a měření, která se na těchto vozidlech provádí. Pro dlouhodobé účely jsem nastínil návrh datového skladu v MS SQL Server 2008, který bude uchovávat data z transakční databáze v dlouhodobém časovém horizontu.

Jelikož jsem při tvorbě transakční databáze úzce spolupracoval s vedením společnosti, pro kterou jsem databázi navrhoval a s jejími zaměstnanci, myslím si, že databáze přesně vystihuje všechny jejich požadavky. Databáze je v současné době nachystaná k implementaci a případné drobné nedostatky se ošetří ve zkušebním provozu.

Tato databáze je dělaná přímo pro společnost STK Královo Pole, s.r.o., a tak mi není známo, zda by mohla mít univerzálnější použití. Důležitý je fakt, že je pro společnost potenciálním přínosem, avšak jak jsem již zmínil v kapitole č. 2.10 Shrnutí nedostatků informačních systémů, neřeší nedostatky, které se vyskytují ve všech podobných společnostech. Myslím si, že hlavní síla databáze spočívá v její snadné obsluze a rychlé registraci zákazníků a jejich vozidel, což ve výsledku zefektivňuje chod celé stanice měření emisí.

Použitá literatura

CONOLLY, T. BEGG, C. HOLOWCZAK, R. *Mistrovství databáze*. 1. vydání. Brno: Computer Press, 2009. 584 s. ISBN 978-80-251-2328-7.

HERNANDEZ, M, J. *Návrh databází*. 1. vydání. Praha : Grada, 2006. 408 s. ISBN 80-247-0900-7.

KOCH, M. 2006. *Datové a funkční modelování*. 2. vydání. Brno : Vysoké učení technické v Brně, 2006. 108 s. ISBN 80-214-3252-7.

MANAGED DEDICATED SERVERY. Hierarchický databázový model. *Managed dedicated servery.cz* [Online]. ©2009 [cit. 2011-12-14]. Dostupné z: <http://www.managed-dedicated-servery.net/hierarchicky-databazovy-model.html>.

NĚMEC, R. Marketingový mix - jeho rozbor, možnosti využití a problémy. *RobertNemec.com* [Online]. ©2001-2011 [cit. 2011-1-10]. Dostupné z: <http://marketing.robertnemec.com/marketingovy-mix-rozbor/>.

OBCHODNÍ REJSTRÍK. STK Královo pole. *Obchodní rejstřík.cz* [Online]. ©2000-2011. [cit. 2011-1-10]. Dostupné z: <http://obchodnirejstrik.cz/stk-kralovo-pole-s-r-o-64508501/>

SKOPAL, T. 2010. Databázové systémy. *Cuni.cz*. [online prezentace] ©2010. [cit. 2011-12-30]. Dostupné z: http://siret.ms.mff.cuni.cz/skopal/databaze/slidy/DBI025_02.pdf.

STK KRÁLOVO POLE. *stkkralovopole.cz* [Online]. ©1996 [cit. 2011-1-10]. Dostupné z: <http://www.stkkralovopole.cz/index.htm>.

ŠEDA, M. Databázové systémy. *Doplňující text ke konzultacím v 3. ročníku kombinovaného bakalářského studia oboru Aplikovaná informatika a řízení* [online]. Brno: VUT FSI v Brně, 2002 [cit. 2011-12-03]. Dostupné z: http://www.uai.fme.vutbr.cz/~mseda/DBS02_BS.pdf

ŠVEC, M. Objektové databáze. *Fit.vutbr.cz* [online]. 2003. [cit. 2011-12-14]. Dostupné z: <http://www.fit.vutbr.cz/study/courses/VPD/public/0203VPD-Svec.pdf>.

Seznam ilustrací

Obrázek 1: Schéma koloběhu informací.....	15
Obrázek 2: Hierarchický datový model	20
Obrázek 3: Schéma síťového datového modelu	21
Obrázek 4: Terminologie z pohledu relací	23
Obrázek 5: Základní schéma datového skladu	27
Obrázek 6: Architektura datového skladu s hvězdicovým schématem	29
Obrázek 7: Datová kostka.....	30
Obrázek 8: Logo společnosti	32
Obrázek 9: Popis 5P marketingového mixu	36
Obrázek 10: Porterův model pěti sil	40
Obrázek 11: Entito-relační diagram transakční databáze	53
Obrázek 12: Tabulky transakční databáze	57
Obrázek 13: Procesní diagram transakční databáze	59 - 60
Obrázek 14: Vývojový diagram transakční databáze	61 - 64
Obrázek 15: Formulář Vozidla	68
Obrázek 16: Schéma datového skladu	68
Tabulka 2: Porovnání systému OLTP se systémy datových skladů.....	27
Graf 3: Jak se firmě daří v posledních letech.....	34

Seznam použitých zkratk

CPU – Central processing unit

DOLAP – Database On-Line Analytical Processing

ETL – Ectraction-Transformation-Loading

HOLAP – Hybrid On-Line Analytical Processing

IČO - Identifikační číslo organizace / osoby

LAN – Local Network Area

LPG – Liquefied Petroleum Gas

MOLAP – Multidimensional On-Line Analytical Processing

ODL – Object Data Language

OLAP – On-Line Analytical Processing

OLTP – On-Line Transaction Processing

OODBMS – Object-Oriented Database Management System

OQL – Object Query Language

PSČ – Poštovní směrovací číslo

RAM – Random Access Memory

ROLAP - Relational Online Analytical Processin

SPZ – Státní poznávací značka

SQL – Structured Query Language

SŘBD (DBMS) – Systém řízení báze dat (DataBase Management System)

STK – Stanice technické kontroly

VBA – Visual Basic for Applications

VIN – Identifikační číslo vozidla

Seznam příloh

Příloha č. 1: Personální schéma

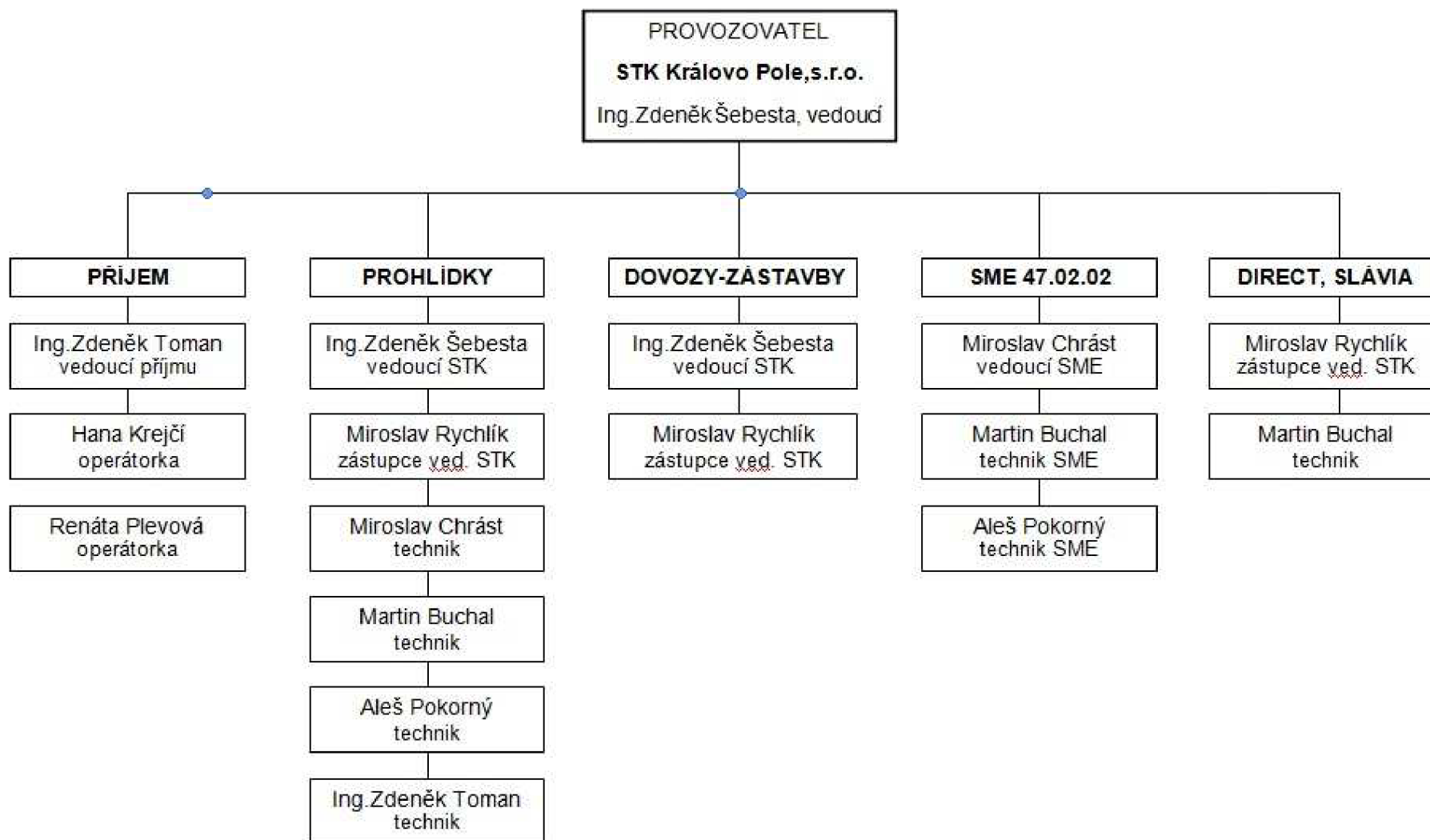
Příloha č. 2: Formulář Zakazníci

Příloha č. 3: Formulář Mereni

Příloha č. 4: Zdrojový kód – procedura registrace

Příloha č. 5: Zdrojový kód – procedura uložení vozidla

Příloha č. 1: Personální schéma



Příloha č. 2: Formulář Zakaznici

Zakaznici

ID zákazník	<input type="text" value="-1749360631"/>
Jméno	<input type="text" value="Pavel"/>
Příjmení	<input type="text" value="Pozler"/>
IČO	<input type="text" value="Neuvedeno"/>
E-mail:	<input type="text" value="pozler@seznam.cz"/>
Ulice:	<input type="text" value="Spodní 24"/>
PSC	<input type="text" value="625 00"/>
Obec	<input type="text" value="Brno 25 (Bohunice)"/>
Název	<input type="text" value="Neuvedeno"/>

Příloha č. 3: Formulář Mereni

Mereni

ID měření	<input type="text" value="-440657103"/>	<input type="button" value="Uložit"/>						
ID vozidla	<input type="text" value="1329668103"/>	<input type="button" value="Nové"/>						
Datum měření	<input type="text" value="21.5.2012"/>							
ID_zamestnanec	<input type="text"/>							
Nazev_mereni	<table border="1"><tr><td>6</td><td>Petr</td><td>Petrzela</td></tr><tr><td>7</td><td>Karel</td><td>Zápotočný</td></tr></table>	6	Petr	Petrzela	7	Karel	Zápotočný	
6	Petr	Petrzela						
7	Karel	Zápotočný						

Hotovost Převodem DPH [%]

ID_faktury	<input type="text" value="F-000 3-21-5-2012"/>
Datum vystavení	<input type="text" value="21.5.2012"/>
Datum splatnosti	<input type="text" value="21.5.2012"/>
Datum zdan. pln.	<input type="text" value="21.5.2012"/>
Celkem bez DPH	<input type="text" value="414,20 Kč"/>
Celkem s DPH	<input type="text" value="497,04 Kč"/>

Příloha č. 4: Zdrojový kód – procedura registrace

Private Sub cmb_registrace_Click()

Dim VIN As String
Dim dbs As Database
Dim rst As Recordset

Nove_vozidlo = False
novy_zakaznik = False

VIN = InputBox("Zadejte VIN vozidla", "Vstupní informace")

Set dbs = CurrentDb 'prirazení k součástce otevřené databázi
Set rst = dbs.OpenRecordset("SELECT * FROM Vozidla WHERE & _
Vozidla.cislo_karoserie = " & "" & VIN & "" ") 'otevře vybraný záznam

If rst.EOF <> False Then 'nenalezen záznam
MsgBox "Vozidlo zde není registrováno", vbOKOnly, "Registrace"
DoCmd.OpenForm "Vozidla", acNormal
Call Funkce.novy_zaznam_vozidlo
Nove_vozidlo = True
GoTo konec
Else
Do While rst.EOF = False
volba = MsgBox(rst![Cislo_karoserie] & " " & rst![SPZ] & " " & rst![Rok_vyroby] & _
, vbYesNo, "Souhlasí")
If volba = 6 Then
GoTo pokračuj
Else
rst.MoveNext
End If
Loop
End If

If volba = 7 Then
DoCmd.OpenForm "Vozidla", acNormal
Call Funkce.novy_zaznam_vozidlo
Nove_vozidlo = True
GoTo konec
Else
GoTo pokračuj
End If

pokracuj:
DoCmd.OpenForm "Vozidla", acNormal
Form_Vozidla.ID_vozidla.Value = rst![ID_vozidla]
Form_Vozidla.SPZ.Value = rst![SPZ]
Form_Vozidla.Druh.Value = rst![Druh]
Form_Vozidla.Znacka.Value = rst![Znacka]
Form_Vozidla.Typ.Value = rst![Typ]
Form_Vozidla.Cislo_motoru.Value = rst![Cislo_motoru]
Form_Vozidla.Palivo.Value = rst![Palivo]
Form_Vozidla.Typ_motoru.Value = rst![Typ_motoru]
Form_Vozidla.Rok_vyroby.Value = rst![Rok_vyroby]
konec:
Form_Vozidla.Cislo_karoserie.Value = VIN
rst.Close
dbs.Close

End Sub

Příloha č. 5: Zdrojový kód – procedura uložení vozidla

```
Private Sub cmb_ulozit_vozidlo_Click()
```

```
Dim Ulozto As Variant
```

```
Dim ID_vozidla As Long
```

```
Dim tb As Control
```

```
Dim SPZ, Druh, Znacka, Typ, Cislo_motoru, Palivo, Cislo_karoserie, Typ_motoru, & _
```

```
Rok_vyroby As String
```

```
Dim dbs As Database
```

```
Dim rst, a, b, x, xy, Zmena_vozidla, rsf, frm, rs As Recordset
```

```
Z1 = False
```

```
Z2 = False
```

```
F1 = False
```

```
F2 = False
```

```
F3 = False
```

```
F4 = False
```

```
F5 = False
```

```
F6 = False
```

```
F7 = False
```

```
'kontrola prázdných polí
```

```
For Each tb In Me.Controls
```

```
If TypeName(tb) = "textbox" Then
```

```
    If tb.Value <> "" Then
```

```
        GoTo juchu
```

```
    Else
```

```
        MsgBox "Nezadali jste " & tb.Name & ", doplňte prosím tento údaj, vbcritical"
```

```
        Exit Sub
```

```
    End If
```

```
juchu:
```

```
    End If
```

```
Next tb
```

```
ID_vozidla = CLng(Form_Vozidla.ID_vozidla.Value)
```

```
SPZ = UCase(Form_Vozidla.SPZ.Value)
```

```
Druh = Form_Vozidla.Druh.Value
```

```
Znacka = Form_Vozidla.Znacka.Value
```

```
Typ = Form_Vozidla.Typ.Value
```

```
Cislo_motoru = Form_Vozidla.Cislo_motoru.Value
```

```
Palivo = Form_Vozidla.Palivo.Value
```

```
Cislo_karoserie = Form_Vozidla.Cislo_karoserie.Value
```

```
Typ_motoru = Form_Vozidla.Typ_motoru.Value
```

```
Rok_vyroby = Form_Vozidla.Rok_vyroby.Value
```

```
Set dbs = CurrentDb 'prirazení k současně otevřené databázi
```

```
'Jedná se o nové vozidlo
```

```
If Nove_vozidlo = True Then
```

```
zpet_uloz:
```

```
    Ulozto = "INSERT INTO Vozidla" & _
```

```
    " VALUES (" & "" & ID_vozidla & "," & "" & SPZ & "," & "" & Druh & "," & _
```

```
        " & "" & Znacka & "," & "" & Typ & "," & "" & Palivo & "," & _
```

```
        " & "" & Cislo_karoserie & "," & "" & Cislo_motoru & "," & "" & Typ_motoru & _
```

```
        " & "" & Rok_vyroby & "" )"
```

```
    DoCmd.RunSQL Ulozto
```

```
    GoTo Pokracuj_bezezmen
```

```
End If
```

'zjištění zda-li došlo ke změnám již uloženého vozidla
'pokud dojde ke změně některého textboxu ve formuláři, uloží se změna

```
Set Zmena_vozidla = dbs.OpenRecordset("SELECT * FROM Vozidla WHERE & _  
cIng(Vozidla.ID_vozidla) = " & "" & ID_vozidla & "" ") ' otevře vybrany zaznam
```

'kontrola zda li je záznam opravdu uložený, v případě, že tomu tak není vracíme se zpět a ukládáme

```
If Zmena_vozidla.EOF = True Then  
    GoTo zpet_uloz  
End If
```

' došlo ke změně SPZ

```
If SPZ <> Zmena_vozidla![SPZ] Then  
    If MsgBox("Došlo ke změně SPZ, chcete uložit změny?", vbYesNo) = vbYes Then  
        Ulozto = "UPDATE Vozidla " & "SET Vozidla.SPZ = " & "" & SPZ & "" WHERE & _  
            cIng(Vozidla.ID_vozidla) = " & "" & ID_vozidla & "" "  
        DoCmd.RunSQL Ulozto  
    End If  
End If
```

```
If Zmena_vozidla![Druh] <> Druh Then  
    If MsgBox("Došlo ke změně Druhu, chcete uložit změny?", vbYesNo) = vbYes Then  
        Ulozto = "UPDATE Vozidla " & "SET Vozidla.Druh = " & "" & Druh & "" WHERE & _  
            cIng(Vozidla.ID_vozidla) = " & "" & ID_vozidla & "" "  
        DoCmd.RunSQL Ulozto  
    End If  
End If
```

```
If Zmena_vozidla![Znacka] <> Znacka Then  
    If MsgBox("Došlo ke změně Znacky, chcete uložit změny?", vbYesNo) = vbYes Then  
        Ulozto = "UPDATE Vozidla " & "SET Vozidla.Znacka = " & "" & Znacka & "" WHERE & _  
            cIng(Vozidla.ID_vozidla) = " & "" & ID_vozidla & "" "  
        DoCmd.RunSQL Ulozto  
    End If  
End If
```

```
If Zmena_vozidla![Typ] <> Typ Then  
    If MsgBox("Došlo ke změně Typ, chcete uložit změny?", vbYesNo) = vbYes Then  
        Ulozto = "UPDATE Vozidla " & "SET Vozidla.Typ = " & "" & Typ & "" WHERE & _  
            cIng(Vozidla.ID_vozidla) = " & "" & ID_vozidla & "" "  
        DoCmd.RunSQL Ulozto  
    End If  
End If
```

```
If Zmena_vozidla![Cislo_motoru] <> Cislo_motoru Then  
    If MsgBox("Došlo ke změně Cislo_motoru, chcete uložit změny?", vbYesNo) = vbYes Then  
        Ulozto = "UPDATE Vozidla " & "SET Vozidla.Cislo_motoru = " & "" & Cislo_motoru & _  
            & "" WHERE cIng(Vozidla.ID_vozidla) = " & "" & ID_vozidla & "" "  
        DoCmd.RunSQL Ulozto  
    End If  
End If
```

```
If Zmena_vozidla![Typ_motoru] <> Typ_motoru Then  
    If MsgBox("Došlo ke změně Typ_motoru, chcete uložit změny?", vbYesNo) = vbYes Then  
        Ulozto = "UPDATE Vozidla " & "SET Vozidla.Typ_motoru = " & "" & Typ_motoru & "" & _  
            WHERE cIng(Vozidla.ID_vozidla) = " & "" & ID_vozidla & "" "
```



```

DoCmd.RunSQL Ulozto
End If
End If

If Zmena_vozidla![Palivo] <> Palivo Then
If MsgBox("Došlo ke změně Palivo, chcete uložit změny?", vbYesNo) = vbYes Then
Ulozto = "UPDATE Vozidla " & "SET Vozidla.Palivo = " & "" & Palivo & "" WHERE & _
    clng(Vozidla.ID_vozidla) = " & "" & ID_vozidla & "" "
DoCmd.RunSQL Ulozto
End If
End If

If Zmena_vozidla![Rok_vyroby] <> Rok_vyroby Then
If MsgBox("Došlo ke změně Roku výroby, chcete uložit změny?", vbYesNo) = vbYes Then
Ulozto = "UPDATE Vozidla " & "SET Vozidla.Rok_vyroby = " & "" & Rok_vyroby & "" & _
    WHERE clng(Vozidla.ID_vozidla) = " & "" & ID_vozidla & "" "
DoCmd.RunSQL Ulozto
End If
End If

If Zmena_vozidla![Cislo_karoserie] <> Cislo_karoserie Then
MsgBox "Došlo ke změně Cislo_karoserie, tato změna je neproveditelná?", vbCritical
Exit Sub
End If

Zmena_vozidla.Close
Pokracuj_bezezmen:

'Zjištění, zda-li je automobil registrovaný na firmu
volba_zakaznik = MsgBox("je automobil registrovaný na firmu?", vbYesNo, "Pokračuj")

' Zjištění, zda li je na automobil registrovaný nějaký zákazník
Set x = dbs.OpenRecordset("SELECT * FROM Vozidla_zakazniku WHERE
clng(Vozidla_zakazniku.ID_vozidla) = " & "" & ID_vozidla & "" ") 'otevre vybrany zaznam

'Větev, automobil je registrovaný na osobu
If volba_zakaznik = 7 Then

If x.EOF = False Then 'nasel nějaký záznam + identifikace zákazníka uživatelem
Do While x.EOF <> True
Set rst = dbs.OpenRecordset("SELECT Z.ID_zakaznik, Z.Jmeno, Z.Prijmeni ,KZ.Ulice & _
,KZ.PSC ,KZ.Email FROM Zakaznici Z, Kontakty_zakazniku KZ WHERE & _
clng(Z.ID_zakaznik) = " & "" & x![ID_zakaznik] & "" AND clng(Z.ID_zakaznik) = & _
clng(KZ.ID_zakaznik) ") 'otevre vybrany zaznam
volba = MsgBox(rst![Jmeno] & " " & rst![Prijmeni] & " " & rst![Ulice] & " " & & _
rst![PSC], vbYesNo, "Souhlasí")
If volba = 6 Then 'uživatel identifikoval zákazníka
Set b = dbs.OpenRecordset("SELECT * FROM Obce WHERE Obce.PSC = " & "" & & _
rst![PSC] & "" ") 'otevre vybrany zaznam
DoCmd.OpenForm "Zakaznici", acNormal
Form_Zakaznici.ID_zakaznik.Value = rst![ID_zakaznik]
Form_Zakaznici.Jmeno.Value = rst![Jmeno]
Form_Zakaznici.Prijmeni.Value = rst![Prijmeni]
Form_Zakaznici.Email.Value = rst![Email]
Form_Zakaznici.Ulice.Value = rst![Ulice]
Form_Zakaznici.PSC.Value = rst![PSC]
Form_Zakaznici.Obec.Value = b![Obec]
Z2 = True
b.Close
GoTo konec

```

```

Else
  x.MoveNext
End If
Loop
'uživatel neidentifikova žádného zákazníka - zkoumáme dále jestli existuje v databázi
GoTo pokračuj_v_hledani_zakaznika

Else 'nenasel zaznam, hledáme zda-li zákazník existuje v databázi
pokracuj_v_hledani_zakaznika:
  jmenoaprijmeni = InputBox("Zadejte jméno a příjmení", Hledání)
  Jmeno = Left(jmenoaprijmeni, InStr(jmenoaprijmeni, " ") - 1)
  Prijmeni = Right(jmenoaprijmeni, Len(jmenoaprijmeni) - (InStr(jmenoaprijmeni, " ")))
  PSC = InputBox("Zadejte PSC hledané osoby", Hledání)
  If Len(PSC) = 5 Then
    PSC = ((Left(PSC, 3)) & (String(1, " ")) & (Right(PSC, 2)))
  End If

  Set rst = dbs.OpenRecordset("SELECT Z.ID_zakaznik, Z.Jmeno, Z.Prijmeni, KZ.Ulice, & _
KZ.PSC, KZ.Email FROM Zakaznici Z, Kontakty_zakazniku KZ WHERE KZ.PSC = & _
" & "" & PSC & "" AND clng(KZ.ID_zakaznik) = clng(Z.ID_zakaznik) AND Z.Prijmeni =& _
" & "" & Prijmeni & "" AND Z.Jmeno = " & "" & Jmeno & """) 'otevre vybrany zaznam
  Do While rst.EOF <> True
    volba = MsgBox(rst![Jmeno] & " " & rst![Prijmeni] & " " & rst![PSC] & " " & & _
rst![Ulice], vbYesNo, "Souhlasí")
    If volba = 6 Then
      podminka = True
      GoTo dalej
    Else
      rst.MoveNext
    End If
  Loop
  podminka = False
dalej:
  If podminka = False Then 'nenasel zaznam, který by odpovídal vstupním parametřům,
registrujeme nového zákazníka
    Z1 = True
    DoCmd.OpenForm "Zakaznici", acNormal
    Call novy_zaznam_zakaznik
    Form_Zakaznici.Jmeno.Value = Jmeno
    Form_Zakaznici.Prijmeni.Value = Prijmeni
    Form_Zakaznici.PSC.Value = PSC
    Set kontakt = dbs.OpenRecordset("SELECT * FROM Obce WHERE Obce.PSC = & _
" & "" & PSC & """) 'otevre vybrany zaznam
    Form_Zakaznici.Obec.Value = kontakt![Obec]
    kontakt.Close
    GoTo konec
  Else 'nasel zaznam, uživatel identifikuje zákazníka
    If volba = 6 Then 'uživatel identifikoval zákazníka
      Set b = dbs.OpenRecordset("SELECT * FROM Obce WHERE Obce.PSC = " & "" & & _
rst![PSC] & """) 'otevre vybrany zaznam
      DoCmd.OpenForm "Zakaznici", acNormal
      Form_Zakaznici.ID_zakaznik.Value = rst![ID_zakaznik]
      Form_Zakaznici.Jmeno.Value = rst![Jmeno]
      Form_Zakaznici.Prijmeni.Value = rst![Prijmeni]
      Form_Zakaznici.Email.Value = rst![Email]
      Form_Zakaznici.Ulice.Value = rst![Ulice]
      Form_Zakaznici.PSC.Value = rst![PSC]
      Form_Zakaznici.Obec.Value = b![Obec]
      Z3 = True
      b.Close
    End If
  End If
End If

```

```

        GoTo konec
    Else
        'uživatel neidentifikova žádného zákazníka - nutno registrovat nového zákazníka
    End If
End If

End If
End If

'Zákazník je registrovaný na firmu, Větev firmy
If volba_zakaznik = 6 Then
    ICO = InputBox("Zadejte IČO Firmy uvedené v technickém průkazu", "Dotaz")
    'ověření existence firmy
    Set frm = dbs.OpenRecordset("SELECT * FROM Firma WHERE Firma.ICO = " & "" & & _
        ICO & "" ") 'otevře vybrany zaznam
    If frm.EOF = False Then 'našel firmu v databázi ZT2
        Set rst = dbs.OpenRecordset("SELECT VZ.ID_Vozidla, VZ.ID_zakaznik, Z.Jmeno, & _
            Z.prijmeni, ZF.ICO FROM Zakaznici_z_firmy ZF, Zakaznici Z, Vozidla_zakazniku VZ & _
            WHERE clng(VZ.ID_vozidla) = " & "" & ID_vozidla & "" AND clng(VZ.ID_zakaznik) =
            Clng(Z.ID_zakaznik) AND clng(Z.ID_zakaznik) = clng(ZF.ID_zakaznik) AND ZF.ICO = & _
            " & "" & ICO & "" ") 'otevře vybrany zaznam
        If rst.EOF = False Then 'našel zákazníka, který je registrovaný u firmy a zároveň má
            registrované i měřené vozidlo, následuje identifikace zákazníka
            Do While rst.EOF <> True
                volba = MsgBox("Souhlasí? " & rst![Jmeno] & " " & rst![Prijmeni], vbYesNo, & _
                    "Souhlasi")
                If volba = 6 Then
                    F1 = True
                    Set b = dbs.OpenRecordset("SELECT * FROM Obce WHERE Obce.PSC = & _
                        " & "" & frm![PSC] & "" ") 'otevře vybrany zaznam
                    DoCmd.OpenForm "Zakaznici", acNormal
                    Form_Zakaznici.ID_zakaznik.Value = rst![ID_zakaznik]
                    Form_Zakaznici.Jmeno.Value = rst![Jmeno]
                    Form_Zakaznici.Prijmeni.Value = rst![Prijmeni]
                    Form_Zakaznici.ICO.Value = ICO
                    Form_Zakaznici.Email.Value = frm![Email]
                    Form_Zakaznici.Ulice.Value = frm![Ulice]
                    Form_Zakaznici.PSC.Value = frm![PSC]
                    Form_Zakaznici.Obec.Value = b![Obec]
                    b.Close
                    frm.Close
                    GoTo konec
                Else
                    rst.MoveNext
                End If
            Loop
            'Neshoda v hledání zákazníka, hledáme zda li je zákazník registrovaný u firmy
            GoTo hledej_firmu
        Else
            'nenašel zákazníka, který je registrovaný u firmy a zároveň má registrované i měřené vozidlo,
            následuje hledání, jestli je zákazník registrovaný alespoň u firmy
            hledej_firmu:
            jmenoaprijmeni = InputBox("Zadejte jméno a příjmení", Hledání)
            Jmeno = Left(jmenoaprijmeni, InStr(jmenoaprijmeni, " "))
            Prijmeni = Right(jmenoaprijmeni, Len(jmenoaprijmeni) - (InStr(jmenoaprijmeni, " ")))
            Set rst = dbs.OpenRecordset("SELECT ZF.ICO, Z.ID_zakaznik, Z.Jmeno, Z.Prijmeni & _
                FROM Zakaznici Z, Zakaznici_z_firmy ZF WHERE ZF.ICO = " & "" & ICO & "" AND & _
                clng(ZF.ID_zakaznik) = clng(Z.ID_zakaznik) AND Z.Jmeno = " & "" & Jmeno & "" AND & _
                Z.Prijmeni = " & "" & Prijmeni & "" ") 'otevře vybrany zaznam

```

```

If rst.EOF = False Then 'zobrazení zákazníka + firmy
    F2 = True
    Set b = dbs.OpenRecordset("SELECT * FROM Obce WHERE Obce.PSC = & _
        " & "" & frm![PSC] & "" ") 'otevře vybrany zaznam
    DoCmd.OpenForm "Zakaznici", acNormal
    Form_Zakaznici.ID_zakaznik.Value = rst![ID_zakaznik]
    Form_Zakaznici.Jmeno.Value = rst![Jmeno]
    Form_Zakaznici.Prijmeni.Value = rst![Prijmeni]
    Form_Zakaznici.ICO.Value = ICO
    Form_Zakaznici.Email.Value = frm![Email]
    Form_Zakaznici.Ulice.Value = frm![Ulice]
    Form_Zakaznici.PSC.Value = frm![PSC]
    Form_Zakaznici.Obec.Value = b![Obec]
    b.Close
    frm.Close
    GoTo konec
Else
'neshoda, zákazník není registrovaný ani k vozidlu ani k firmě, hledáme zda li je zákazník vůbec v
databázi, to je případ kdy by zde byl s automobilem, který není registrovaný na firmu, proto
hledáme v kontaktech zákazníků
    PSC = InputBox("Zadejte PSC hledané osoby", Hledání)
    Set rst = dbs.OpenRecordset("SELECT Z.* , KZ.Email, KZ.Ulice, KZ.PSC & _
FROM Zakaznici Z, Kontakty_zakazniku KZ WHERE KZ.PSC = " & "" & PSC & "" AND & _
clng(KZ.ID_zakaznik) = clng(Z.ID_zakaznik) AND Z.Jmeno = " & "" & Jmeno & "" AND & _
Z.Prijmeni = " & "" & Prijmeni & "" ") 'otevře vybrany zaznam
    Do While rst.EOF <> True
        volba = MsgBox(rst![Jmeno] & " " & rst![Prijmeni] & " " & rst![PSC] & _
& " " & rst![Ulice], vbYesNo, "Souhlasí")
        If volba = 6 Then
            vysledek = True
            GoTo dal
        Else
            rst.MoveNext
        End If
    Loop
    vysledek = False
dal:
If vysledek = True Then 'našel zákazníka/shoda
    F4 = True
    Set b = dbs.OpenRecordset("SELECT * FROM Obce WHERE Obce.PSC & _
        = " & "" & frm![PSC] & "" ") 'otevře vybrany zaznam
    DoCmd.OpenForm "Zakaznici", acNormal
    Form_Zakaznici.ID_zakaznik.Value = rst![ID_zakaznik]
    Form_Zakaznici.Jmeno.Value = rst![Jmeno]
    Form_Zakaznici.Prijmeni.Value = rst![Prijmeni]
    Form_Zakaznici.ICO.Value = ICO
    Form_Zakaznici.Email.Value = frm![Email]
    Form_Zakaznici.Ulice.Value = frm![Ulice]
    Form_Zakaznici.PSC.Value = frm![PSC]
    Form_Zakaznici.Obec.Value = b![Obec]
    b.Close
    frm.Close
    GoTo konec
Else 'nenašel zákazníka/neshoda
    F5 = True
    Set b = dbs.OpenRecordset("SELECT * FROM Obce WHERE Obce.PSC & _
        = " & "" & frm![PSC] & "" ") 'otevře vybrany zaznam
    DoCmd.OpenForm "Zakaznici", acNormal
    Call novy_zaznam_zakaznik
    Form_Zakaznici.ICO.Value = ICO

```

```

        Form_Zakaznici.Email.Value = frm![Email]
        Form_Zakaznici.Ulice.Value = frm![Ulice]
        Form_Zakaznici.PSC.Value = frm![PSC]
        Form_Zakaznici.Obec.Value = b![Obec]
        b.Close
        frm.Close
        GoTo konec
    End If
End If
End If
Else 'nenal firmu v databázi ZT1, zkoumáme zda li je zde alepoň zákazník registrovaný
If x.EOF = False Then 'nasel nějaký záznam + identifikace zákazníka uživatelem
    Do While x.EOF <> True
        Set rst = dbs.OpenRecordset("SELECT Z.* ,KZ.Email ,KZ.Ulice ,KZ.PSC & _
FROM Zakaznici Z, Kontakty_zakazniku KZ WHERE cLng(Z.ID_zakaznik) = " & "" & & _
x![ID_zakaznik] & "" AND cLng(Z.ID_zakaznik) = cLng(KZ.ID_zakaznik) ") 'otevre vybrany
zaznam
        volba = MsgBox(rst![Jmeno] & " " & rst![Prijmeni] & " " & rst![PSC] & " " & & _
rst![Ulice], vbYesNo, "Souhlasí")
        If volba = 6 Then 'uživatel identifikoval zákazníka
            DoCmd.OpenForm "Zakaznici", acNormal
            Form_Zakaznici.ID_zakaznik.Value = rst![ID_zakaznik]
            Form_Zakaznici.Jmeno.Value = rst![Jmeno]
            Form_Zakaznici.Prijmeni.Value = rst![Prijmeni]
            Form_Zakaznici.ICO.Value = ICO
            Form_Zakaznici.Email.Value = ""
            Form_Zakaznici.Ulice.Value = ""
            Form_Zakaznici.PSC.Value = ""
            Form_Zakaznici.Obec.Value = ""
            F6 = True
            frm.Close
            GoTo konec
        Else
            x.MoveNext
        End If
    Loop
    ' uživatel neidentifikoval zákazníka
    GoTo nenasel_nic
Else 'nenasel záznam o zákazníkovi, hledáme zda li je zákazník alespon v databázi
    jmenoaprijmeni = InputBox("Zadejte jméno a příjmení", "Ověření existence zákazníka")
    PSC = InputBox("Zadejte PSC hledané odoby", "Ověření existence zákazníka")
    Jmeno = Left(jmenoaprijmeni, InStr(jmenoaprijmeni, " "))
    Prijmeni = Right(jmenoaprijmeni, Len(jmenoaprijmeni) - (InStr(jmenoaprijmeni, " ")))
    If Len(PSC) = 5 Then
        PSC = (Left(PSC, 3)) & (String(1, " ")) & (Right(PSC, 2))
    End If
    Set rst = dbs.OpenRecordset("SELECT KZ.Email, KZ.Ulice, KZ.PSC, Z.Jmeno, & _
Z.Prijmeni FROM Zakaznici Z, Kontakty_zakazniku KZ WHERE Z.Jmeno = " & "" & & _
Jmeno & "" AND Z.Prijmeni = " & "" & Prijmeni & "" AND cLng(Z.ID_zakaznik) = & _
cLng(KZ.ID_zakaznik) AND KZ.PSC = " & "" & PSC & "" ") 'otevre vybrany zaznam
    Do While rst.EOF <> True
        volba = MsgBox(rst![Jmeno] & " " & rst![Prijmeni] & " " & rst![PSC] & " " & & _
rst![Ulice], vbYesNo, "Souhlasí")
        If volba = 6 Then 'uživatel identifikoval zákazníka
            DoCmd.OpenForm "Zakaznici", acNormal
            Form_Zakaznici.ID_zakaznik.Value = rst![ID_zakaznik]
            Form_Zakaznici.Jmeno.Value = rst![Jmeno]
            Form_Zakaznici.Prijmeni.Value = rst![Prijmeni]
            Form_Zakaznici.ICO.Value = ICO
            Form_Zakaznici.Email.Value = ""

```

```

        Form_Zakaznici.Ulice.Value = ""
        Form_Zakaznici.PSC.Value = ""
        Form_Zakaznici.Obec.Value = ""
        F6 = True
        frm.Close
        GoTo konec
    Else
        rst.MoveNext
    End If
Loop
' uživatel neidentifikoval zákazníka

nenasel_nic:
    F7 = True
    DoCmd.OpenForm "Zakaznici", acNormal
    Call novy_zaznam_zakaznik
    Form_Zakaznici.ICO.Value = ICO
    frm.Close
    GoTo konec
End If
End If

End If

konec:

    rst.Close
    x.Close
    dbs.Close

    If volba_zakaznik = 7 Then
        Form_Zakaznici.ICO.Visible = False
        Form_Zakaznici.Nazev_firmy.Visible = False
    Else
        Form_Zakaznici.ICO.Visible = True
        Form_Zakaznici.Nazev_firmy.Visible = True
    End If

End Sub

```