

TECHNICKÁ UNIVERZITA V LIBERCI
Hospodářská fakulta

Studijní program: Systémové inženýrství a informatika
Studijní obor: B 6209 Manažerská informatika

**Návrh na optimalizaci informačního systému SQS ve
společnosti Škoda Auto pomocí statistických metod
s prvky umělé inteligence**

**Proposal for optimisation of information system SQS in
Škoda Auto company utilizing statistical methods with
elements of artificial intelligence**

DP-MI-KIN-2008-06
Jaroslav Klimeš

Vedoucí práce: Ing. Dana Nejedlová, Ph.D., katedra informatiky
Odborný konzultant: Ing. Miroslav Grepl, Škoda Auto a.s., GQA
Počet stran: 52
Počet příloh: 4
Datum odevzdání: 11.1.2008

PODKLAD PRO ZADÁNÍ DIPLOMOVÉ PRÁCE

PŘEDKLÁDÁ:	ADRESA:	OSOBNÍ ČÍSLO:
KLIMEŠ Jaroslav	Olbrachtova 611/2; Liberec	H02332000

NÁZEV TÉMATU ČESKY:

Návrhy na optimalizaci informačního systému SQS ve společnosti Škoda Auto pomocí statistických metod s prvky umělé inteligence

NÁZEV TÉMATU ANGLICKY:

VEDOUCÍ PRÁCE:

Ing. Dana Nejedlová, Ph.D.

ZÁSADY PRO VYPRACOVÁNÍ:

1. Přehled dat pro analýzu, procesů tato data generujících a prakticky využitelných výsledků analýzy
2. Teoretické základy vybraných metod analýzy
3. Analýza dostupných dat a interpretace výsledků

SEZNAM DOPORUČENÉ LITERATURY:

1. Vladimír MAŘÍK, Olga ŠTEPÁNKOVÁ, Jiří LAŽANSKÝ a kolektiv: Umělá inteligence, Academia Praha. ISBN 80-200-0502-1 (soubor). 1. díl vyšel roku 1993, 2. díl roku 1997, 3. díl roku 2001, 4. díl roku 2003.
2. Patrick Henry WINSTON: Artificial Intelligence, Addison-Wesley Publishing Company, Reading, Massachusetts, ISBN: 0-201-53377-4, March 1992, 1999.
3. Laurene FAUSETT: Fundamentals of Neural Networks, Architectures, Algorithms and Applications. Prentice Hall International, Inc., New Jersey, 1994, ISBN 0-13-334186-0.
4. Stuart RUSSELL, Peter NORVIG: Artificial Intelligence: A Modern Approach. Prentice Hall International, Inc., New Jersey, 1995, ISBN 0-13-360124-2.

PODPIS

STUDENTA: _____

DATUM: _____

PODPIS

**VEDOUCÍHO
PRÁCE:** _____

DATUM: _____

Byl(a) jsem seznámen(a) s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, zejména § 60 - školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

Datum: 11.1.2008

Podpis:

Resumé

Tato diplomová práce se zabývá návrhy na optimalizaci informačního systému SQS ve Škoda Auto a.s. První část práce je věnována úvodu do problematiky, tedy informačním systémům ve Škoda Auto a.s. a popisu stavby a fungování informačního systému SQS. Další část se pak krátce zabývá projektem pro statistické zpracování výstupů, který byl realizován v roce 2005 a nastínil další možnosti optimalizace SQS. Následuje popis cílů této optimalizace a teoretická analýza metod, které jsou při ní použity. Poslední část práce ukazuje konkrétní výsledky, které byly pomocí těchto metod získány, jejich hodnocení a význam pro optimalizaci informačního systému SQS.

Klíčová slova: informační systém, optimalizace, statistické metody, Škoda Auto, webové programovací jazyky, databáze, PHP

Abstract

The thesis deals with optimisation of SQS information system in Škoda Auto a.s. The first part of the work aims at the information systems used in Škoda Auto a.s and gives a thorough description of the structure and functions of SQS. The second part briefly tells of the project of statistical processing of outputs, which was performed in 2005 and formed a base for further optimisations. The outline of these optimisations' goal follows, as well as the theoretical analysis of the methods they require. The last part of the thesis shows the outcomes of these methods usage, together with the assessment of their significance for the optimisation of information system SQS.

Keywords: information system, optimisation, statistical methods, Škoda Auto, web programming languages, database, PHP

Obsah

1	Úvod	10
2	Informační systémy ve Škoda Auto	11
2.1	Architektura IS ve Škoda Auto	11
2.2	Správa a podpora IS	12
2.2.1	Systémy ve správě Škoda Auto	13
2.2.2	Systémy ve správě koncernu VW	13
2.3	Informační systémy kvality	14
3	Informační systém SQS	16
3.1	Funkce SQS	16
3.2	Architektura SQS	17
3.3	Vstupní část SQS	18
3.4	Výstupní část SQS – aplikace SQS Global II	21
3.5	Projekt statistického zpracování výstupů SQS Global II	26
4	Optimalizace informačního systému SQS	28
4.1	Stanovené cíle	28
4.2	Metody zvolené k dosažení cílů	29
4.3	Data a prostředky pro jejich zpracování	29
4.4	Testování nezávislosti v kontingenční tabulce	35
4.4.1	Testování statistických hypotéz	36
4.4.2	Neparametrické testy	38
4.4.3	χ^2 test nezávislosti v kontingenční tabulce	38
4.4.4	Měření závislosti kategoriálních veličin	40
4.5	Bayesův teorém podmíněné pravděpodobnosti	41
4.5.1	Bayesovské sítě	42
4.6	Analýza parametrizace sestav	43
4.6.1	Cíl analýzy	43
4.6.2	Konstrukce testů	44
4.6.3	Způsob provedení testů	45
4.6.4	Testy nezávislosti používání parametrů sestav na uživateli	46
4.6.5	Testy nezávislosti používání parametrů sestav na lince	51

4.7	Návrh úpravy parametrizací sestav SQS	55
4.8	Zachycení závislosti spouštění sestav na čase	57
5	Závěr	61
6	Seznam tabulek	62
7	Seznam obrázků	62
8	Použité zdroje	63
9	Seznam příloh	64

Seznam použitých zkratk

AGOS Audit	Informační systém kvality ve Škoda Auto
DBMS	Database management system, systém řízení báze dat
EO	Oddělení informačních technologií ve Škoda Auto
ER	Entity relationship, vztah entit
FIS	Informační systém řízení výroby ve Škoda Auto
Gedas	Firma zajišťující vývoj informačního systému SQS
GQA	Oddělení kvality ve Škoda Auto
HTML	Formát dokumentu pro webový prohlížeč
HTTP	Hypertext transfer protocol, protokol pro přenos hypertextu
IS	Informační systém
K-QS	Oddělení kvality v koncernu Volkswagen
KB	Kontrolní bod
KDNR	Kundendienstnummer – zákaznické číslo dílu
KKV	Kontrolní karta vozu
MySQL	Open-source databázový systém
PDF	Portable document format, formát dokumentu
PHP	Hypertext preprocessor, webový programovací jazyk
QM	Management kvality
QUASI-FI	Informační systém kvality ve Škoda Auto
QUASI-LIMS	Informační systém kvality ve Škoda Auto
SQL	Structured query language, strukturovaný dotazovací jazyk
SQS	Skoda Quality System, informační systém kvality ve Škoda Auto
SQS Global II	Část informačního systému SQS
SZV	Statistické zpracování výstupů
Tevon	Informační systém kvality ve Škoda Auto
VDS	Informační systém kvality ve Škoda Auto
VW	Volkswagen
Wissensportal	Informační systém kvality ve Škoda Auto
XLS	Formát dokumentu pro MS Excel

1 Úvod

V průběhu své dvousemestrální praxe v oddělení GQA ve Škoda Auto a.s. v období od září 2004 do června 2005 jsem měl možnost se seznámit s většinou z informačních systémů kvality, které se ve Škoda Auto používají. Pravděpodobně nejdůležitějším z nich pro každodenní fungování výroby vozů je SQS, který umožňuje on-line vyhodnocování a zobrazování informací o kvalitě vyráběných vozů ve společnosti Škoda Auto.

Během své praxe jsem byl odpovědný za projekt statistického zpracování výstupů. Cílem tohoto projektu bylo vyhodnocení využívanosti výstupů poskytovaných SQS z různých hledisek tak, aby měli správci systému dokonalý přehled o tom, co, kdy, jak a kým je používáno. Projekt byl úspěšně realizován pomocí přidání nového statistického modulu přímo do IS SQS. Ve své původní podobě projekt počítal pouze s využitím dat správci ve formě jednoduchých tabulek a grafů absolutních četností, později se začaly vytvářet i o něco složitější maticové přehledy. Jedním z důsledků řešení problému tímto způsobem je ovšem skutečnost, že veškerá data o činnosti uživatelů jsou nyní systematicky sbírána a ukládána na serverech správců SQS pro využití tímto modulem.

Tato diplomová práce na projekt statistického zpracování výstupů navazuje a snaží se jej posunout ještě o něco dále. S pomocí programovacího jazyka PHP jsou vytvořeny skripty, které spolupracují s relačními databázemi obsahujícími data o činnosti uživatelů a na tato data aplikují různé statistické metody. Cílem je nalezení užitečných souvislostí v datech týkajících se činnosti uživatelů – především pak způsobu, jakým uživatelé upřesňují své dotazy do SQS, a zjištění struktury druhů dotazů spouštěných v SQS v závislosti na čase. Cílem práce je využít takto nalezené souvislosti pro návrh optimalizace informačního systému SQS.

V této práci se nejprve věnuji koncepci informačních systémů ve Škoda Auto a pak specificky IS kvality. Dále charakterizuji samotný IS SQS a zejména jeho část pro zpracování výstupů pomocí tzv. sestav. Ve třetí části stručně popisuji projekt statistického zpracování výstupů a data, která se pro jeho potřeby sbírají a archivují. Jsou nastíněny možnosti dalšího využití dat pro optimalizaci, vybrány metody pro práci s nimi a ty potom

popsány. Další část je čistě praktická, ukazuje konkrétní způsob zpracování dat a jeho výsledky spolu s jejich interpretací a hodnocením. Na závěr práce uvádím přínosy, které mohou výsledky práce společnosti Škoda Auto přinést.

2 Informační systémy ve Škoda Auto

Společnost Škoda Auto používá pro své potřeby velké množství informačních a komunikačních systémů, podle dostupných údajů z listopadu 2004 jich bylo 191 [1]. Většinou jsou tyto systémy úzce specializované na jednu základní funkci, díky čemuž jsou pro uživatele přehledné a snižuje se tak potřeba proškolení uživatelů. Právě velké množství a rozmanitost IS ve Škodě ale způsobuje problémy při potřebě sdílení dat mezi různými IS.

Jediným informačním systémem, který používají všechny organizační jednotky Škoda Auto jsou intranetové stránky. Tam je kromě informací o společnosti, předpisů a směrnic možné najít také prezentace činností jednotlivých oddělení. Přímou z Intranetu je také možné pomocí odkazů přistupovat do některých informačních systémů, které využívají webové rozhraní.

2.1 Architektura IS ve Škoda Auto

Informační systémy ve Škoda Auto lze podle architektury rozdělit na tři druhy:

- Pracovní stanice / souborový či databázový server

Tyto IS fungují následujícím způsobem: Na pracovní stanici je nainstalována aplikace, která může vykonávat veškeré funkce daného systému, včetně zpracování dat. Komunikaci s jinými pracovními stanicemi zajišťuje souborový nebo databázový server, kam aplikace ukládá již zpracovaná data. Tato architektura systému je ve Škoda Auto poměrně rozšířená, ale v současné době se od ní ustupuje především kvůli vyšším nákladům na podporu těchto

systémů. Také nároky na bezpečnost uchovávaných dat se ve Škoda Auto zvyšují, a tyto IS jim ne vždy vyhovují, neboť často ukládají citlivá data na pevné disky lokálních stanic.

- **Klient / Server**

Tato koncepce umožňuje, aby byla hlavní část IS spuštěna pouze na serveru. Na serveru jsou uložena všechna data, ale server také zároveň zprostředkovává služby uživatelům, kteří k nim přistupují ze svých počítačů pomocí klientských aplikací, většinou jednodušších a méně náročných na hardwarové prostředky než serverová část aplikace. Výhodou této architektury IS je kromě nenáročnosti na hardware klientských stanic také oddělení klienta od serveru, tudíž klient a server mají možnost pracovat na rozdílných platformách. Další výhodou jsou menší nároky na podporu IS a vyšší bezpečnost uložených dat, neboť klientské aplikace neukládají data na lokální stanice. Konečně, jako klientskou část informačního systému lze použít také obyčejný webový prohlížeč, který bývá standardním vybavením každé stanice a klientskou aplikaci pak není třeba vůbec instalovat. Toto řešení má mnoho výhod, proto je mnoho nových IS ve Škoda Auto koncipována právě takto.

- **Terminál / terminálový server (Host)**

Toto řešení minimalizuje nároky na klientskou aplikaci, která tak může běžet téměř na jakémkoli hardware. Veškeré operace zde vykonává serverová část systému. Velkou výhodou je vysoká bezpečnost dat, na druhou stranu se mohou objevit problémy, pokud chce uživatel pracovat s lokálními zdroji (pevný disk, tiskárna). Tato architektura je používána například u systémů, které mají příliš vysoké nároky na výpočetní výkon nebo rychlost síťového připojení.

2.2 Správa a podpora IS

K objasnění správy a podpory IS ve Škoda Auto je nejprve nutné systémy rozlišit na IS Škody a IS koncernu VW.

2.2.1 Systémy ve správě Škoda Auto

Systémy ve správě Škoda Auto mají obvykle na starosti dvě organizační jednotky:

Tou první je oddělení EO, které se stará o hardware, na kterém IS běží (servery, stanice). Do jeho kompetencí spadají také instalace softwaru na pracovní stanice a řešení problémů technického rázu. Podporu uživatelů a vývoj systému pak má v rukou to oddělení, do jehož kompetencí daný IS spadá svojí funkcí (např. IS kvality bude spadat pod oddělení zabývající se kvalitou).

V některých případech je správa outsourcována na jinou společnost; příkladem může být například IS SQS, jehož správu zčásti (správa serverů, vývoj) zajišťuje firma Gedas, bývalá dceřiná společnost koncernu VW, od roku 2006 součást společnosti T-Systems..

2.2.2 Systémy ve správě koncernu VW

Pokud ve Škoda Auto sledujeme vývoj informačních systémů, lze vypožorovat rostoucí trend sdílení dat z IS na úrovni celého koncernu Volkswagen. Vzhledem k tomu, že starší IS Škody jsou obvykle jednoúčelové a jejich výstupy nekompatibilní s jinými IS, dochází často k jejich nahrazení jinými IS, jejichž administrace a vývoj jsou plně zajištěny koncernem VW. Takové informační systémy pak používají buďto jednu databázi pro celý koncern, nebo jsou databáze nacházející se fyzicky v jednotlivých automobilkách propojeny. Zálležitosti týkající se hardwaru a softwaru ve Škoda Auto spadají opět pod oddělení EO. Podporu uživatelů pak má na starost ta organizační jednotka Škoda Auto, do jejichž kompetencí systém spadá svojí funkcí.

Pro úplnost je třeba dodat, že ne vždy je kvůli potřebě sdílení dat třeba nahradit původní IS jiným z koncernu VW. Existuje zde také možnost rozšíření původního IS o možnost importu/exportu dat pomocí konverze. Tento způsob je výhodnější pro uživatele IS, kteří se tak nemusí učit pracovat s jiným systémem. Kontrola nad systémem tady také zůstává v rukou Škoda Auto. Nevýhodou je, že tento způsob není možné aplikovat vždy, neboť funkce, a tedy i výstupy IS Škoda Auto a koncernových IS jsou často odlišné a může být

problém je sladit. V některých případech tak může nastat nutnost používat pro jednu činnost dva IS, jeden pro potřeby Škoda Auto, druhý pak pro sdílení dat na úrovni koncernu VW. Tento způsob však přiděluje uživatelům mnoho práce především při zadávání dat. Příkladem mohou být například IS pro sledování závad u zákazníka – AGOS ORi a koncernový QUASI-FI. Oba informační systémy mají stejné druhy výstupů, nicméně metodika výpočtu těchto výstupů je rozdílná a proto jsou používány souběžně.

2.3 Informační systémy kvality

Informační systémy kvality jsou určené k záznamu, uchování, zpřístupnění a vyhodnocení dat týkajících se kvality vozů v různých fázích vývoje či výroby, jejich částí či dílů. O většinu z nich se ve Škoda Auto stará oddělení GQA, kde jsem během své praxe pracoval.

Oddělení GQA má název „Strategie QM a Audit kvality“. Je rozděleno na tři části. První se zabývá audity – konkrétně audity dílů, vozů či procesními audity. Druhá část se věnuje určení Strategie Quality Managementu. Konečně, třetí část se stará o IS kvality. Tato část zajišťuje veškerou podporu uživatelům IS kvality, přijímá jejich požadavky a komunikuje s dodavateli systémů. Aktualizace a inovace těchto systémů jsou zajišťovány vývojářskými firmami, případně koncernovým oddělením K-QS na základě požadavků a připomínek uživatelů.

Rád bych se zde krátce zmínil o informačních a komunikačních systémech, jejichž provoz zajišťuje oddělení GQA. Jsou to:

SQS	Informační systém pro zadávání a vyhodnocování dat o kvalitě vyráběných vozů na všech výrobních linkách ve všech závodech Škoda Auto a. s. Jeho základem je serverová část s databázemi, která slouží dvěma druhům klientských aplikací: Klientu pro zadávání dat ve výrobě, a webovému prohlížeči, pomocí kterého si lze prohlédnout výstupy.
-----	--

AGOS Audit	IS kvality pro evidenci a analýzu závad při auditech částí vozu, součást systému AGOS ORi. V roce 2005 byl přidán modul umožňující zpracování procesních auditů. Jedná se o řešení pracovní stanice-fileserver, kde jsou na server ukládány zprávy z auditů.
VDS	Koncernový IS pro sledování problémů a komunikaci o problémech na vozech ve fázích vývoje, výroby a prodeje vozu. Obsahuje databázi problémů a jejich řešení, pokud již bylo nalezeno. Řešeno jako klient-server, o server se stará koncernové oddělení kvality K-QS.
QUASI-FI	Koncernový statistický IS. Informuje o počtu závad u zákazníka, na trzích Německa, Francie, Itálie, Švýcarska, Británie, Španělska aj. Opět řešení klient-server, jako klient funguje webový prohlížeč.
QUASI-LIMS	Koncernový informační systém, který umožňuje zadávat zakázky na analýzy v laboratořích a sledovat stav jejich zpracování zadavateli i příjemci. Poskytuje databázi zpracovaných zakázek (know-how). Funguje buďto jako klient-server, nebo je možné použít terminálovou verzi – to kvůli dlouhým prodlevám při práci s databází při použití klasického klienta.
TEVON	IS pro záznam a sledování dat o vzorkování dílů. Terminálové řešení.
WISSENSPORTAL	Informační portál koncernové kvality. Obsahuje inteligentní vyhledávač, který provádí rešerše v datech z koncernových IS kvality, dále Quasi moduly, Explorer, Diskusní fóra expertů, tiskový informační servis. Jako klient je použit webový prohlížeč, o serverovou část se stará oddělení K-QS.

3 Informační systém SQS

Informační systém SQS (Skoda Quality System) vyvinula společnost Škoda Auto ve spolupráci s firmou Gedas. Tato firma převzala technickou realizaci systému, nyní zajišťuje jeho vývoj a spravuje jeho serverovou část. Projekt vývoje SQS byl zahájen v roce 1994, a v polovině roku 1995 již fungoval na montážní lince M1 [1]. V dalších letech byl rozšířen také na montáž v závodě Kvasiny, svařovnu a montáž vozu Octavia. V roce 1998 byly sloučeny databáze ze všech provozů, kde byl tento IS nasazen, a přidána důležitá funkce – výstupy z SQS přístupné přes intranetovou síť. Do dnešní doby byl SQS rozšířen do všech provozů výroby vozů v českých závodech Škoda Auto, zaveden byl již i do některých zahraničních závodů (např. Indie, Rusko). Informační systém SQS byl také propojen s koncernovým IS FIS pro řízení výroby. V současné době používá SQS ve Škoda Auto přes 450 uživatelů.

3.1 Funkce SQS

IS SQS umožňuje kontrolu a hodnocení kvality vozů během celého procesu výroby. Na jeho funkce se můžeme podívat ze dvou pohledů. Takto by se na funkce systému dalo nahlížet z pohledu uživatelů:

- Přímá podpora výroby
 - prostřednictvím propojení s výrobním systémem FIS
- Podpora nižšího managementu
 - prostřednictvím monitoringových výstupů SQS Global II
- Podpora středního managementu
 - prostřednictvím statistických výstupů SQS Global II
- Archivace dat

Pokud bychom se na IS SQS naopak podívali ze strany správce či vývojáře, můžeme systém rozdělit na dvě hlavní části:

- Zadávání, zpracování a archivaci dat týkajících se výroby vozů

- Příprava a výstup těchto dat ze systému

První skupinu, tedy zadávání, zpracování a archivaci dat, popisují podrobněji v kapitole 3.3 (Vstupní část SQS). Přípravou a výstupy dat se zabývám v kapitole 3.4 (Výstupní část SQS – SQS Global II)

3.2 Architektura SQS

SQS se skládá z následujících částí:

- Kontrolní body

Na každé výrobní lince funguje několik stanic, kde se kontrolují operace na voze provedené. Tyto stanice se nazývají kontrolní body (KB) a vždy disponují zařízením, které umožňuje zadávat do SQS jak samotný průchod daného vozu kontrolním bodem, tak i nedostatky na voze zjištěné obsluhou KB či jinými pracovníky linky. Do SQS jsou pak zapisována i data o nápravách těchto nedostatků. Každý z těchto záznamů obsahuje také přesné údaje o času, místě a personálu spojeném s daným úkonem.

- Rozhraní se systémem FIS

Systém FIS je koncernový systém pro řízení výroby. Rozhraní mezi SQS a FIS umožňuje obousměrnou výměnu dat mezi systémy a odstraňuje tak nutnost zadávat některé údaje dvakrát.

- Centrální databáze

Hlavní databázový server spravovaný firmou Gedas ukládá data ze všech KB i dalších vstupních míst do databází Oracle.

- Webový server

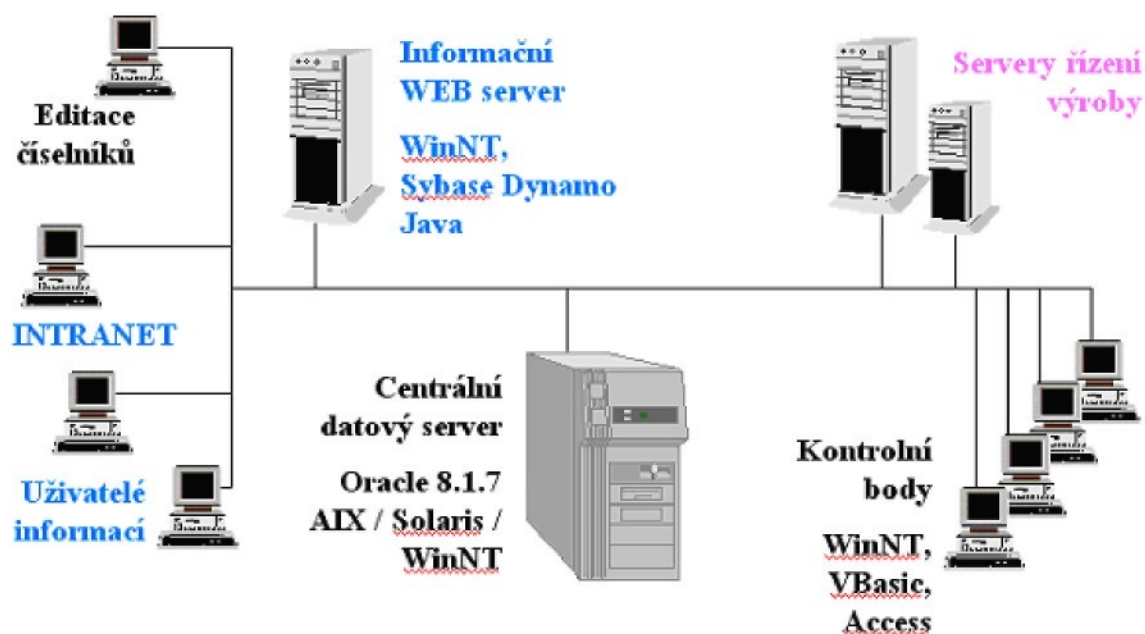
Na tomto serveru běží webové rozhraní systému SQS pro přípravu výstupů, aplikace SQS Global II. SQS Global II přijímá dotazy od uživatelů, zasílá je na hlavní databázový server a odpovědi (výstupy) upravuje do příslušného formátu. Pro tvorbu výstupů aplikace

používá tzv. sestavy, což jsou předdefinované dotazy do databáze SQS s předem stanovenou formou výstupu. Uživatel může u sestav ještě měnit některé parametry, aby výstup přizpůsobil svým požadavkům.

- Uživatelé výstupů z SQS

Uživatelé se nejprve musí pomocí svých webových prohlížečů do SQS Global II přihlásit, poté mohou vybírat z různých druhů výstupů. Aplikace jim tyto výstupy umí poskytnout buďto ve formátu HTML pro webový prohlížeč, nebo excelovském XLS.

SQS - stavba systému



Obrázek 3.1: Stavba IS SQS [1]

3.3 Vstupní část SQS

Každý vůz vyráběný ve Škoda Auto projde několika výrobními provozy: Nejprve je to lisovna, kde se vylisují plechové díly karosérie, dále pak svařovna, kde se tyto díly svaří. Tak vznikne karosérie vozu. Každé karosérii jsou přiděleny identifikační údaje; především

číslo vozu, číslo zakázky a PR-čísla komponent, z kterých má být vůz složen [2]. PR-čísla označují nejen druh komponenty, ale i její provedení – třeba barvu vozu, typ motoru, přítomnost střešního okénka apod. Už ve svařovně karosérií je tedy jasné, jak má který vůz vypadat – tudíž jaké operace na něm mají být provedeny.

Karosérii je ve svařovně přidělena kontrolní karta vozu (KKV), která je vložena dovnitř vozu a putuje s ním dále přes svařovnu, lakovnu a montáž až ke KB8, což je poslední kontrolní bod ve výrobním procesu vozu. KKV obsahuje identifikační údaje o vozu, a to jak v čitelné formě, tak i v podobě čárového kódu. Do KKV se dále na příslušných místech zaznamenávají závady na voze nalezené a průchody jednotlivými kontrolními body. KKV se skládá z několika stran, z nichž některé jsou určeny pro strojové čtení dat, jiné pak obsahují informace v čitelné formě. Stránky se strojově čitelnými údaji obsahují políčka, kde každé políčko odpovídá určitému druhu závady na určitém dílu. Zodpovědný pracovník na lince při rozpoznání závady tuto závadu vyznačí do KKV tužkou, přesněji řečeno začerní políčko odpovídající konkrétní závadě.

Vstup dat do IS SQS se děje na KB. KB je místo, kde se zkontrolují operace na voze provedené od průchodu předchozím KB, překontrolují se závady nalezené pracovníky linky a vozy se závadami se pošlou zpět na opravu. KB jsou vybaveny stanicí pro zadávání dat, která se skládá z PC vybaveného ručním skenerem čárových kódů, skenerem KKV a tiskárnou. PC je přes síť propojen s databázovým serverem SQS. Samotné zadávání dat probíhá pomocí klientské aplikace, která je na PC nainstalována.

Postup zadání dat do SQS je následující: Pracovník KB zkontroluje vůz a vyznačí případné závady do KKV. Pak se musí přihlásit do stanice pro zadávání dat, pokud tak neučinil již dříve. Pracovníci jsou pro rychlé přihlášení vybaveni osobním štítkem s čárovým kódem, který jednoduše naskenují čtečkou čárových kódů, a jsou přihlášení. Dále musí pracovník KB naskenovat číslo vozu, které je ve formě nálepky s čárovým kódem nalepené na KKV. Pak již může zadávat data o závadách. To udělá tím způsobem, že vloží KKV do skeneru, a údaje o závadách jsou přeneseny na databázový server SQS. Po načtení závad se ještě vytiskne protokol o průchodu KB, kde jsou čitelně vyznačeny všechny důležité údaje včetně závad. Tento protokol je nalepen na odpovídající místo v KKV.



Obrázek 3.2: Stanice pro zadávání dat na KB vybavená skenerem KKV [1]

Kromě standardního načítání údajů o závadách přes KKV a skener existuje ještě několik alternativních způsobů zadávání dat do SQS, které však nejsou příliš rozšířené:

- Načítání závad skenováním čárových kódů odpovídajících závadě
- Zadávání závad přes klávesnici
- Zadávání závad pomocí Pocket PC



Obrázek 3.3 (vlevo): Stanice pro zadávání dat na KB vybavená skenerem KKV [1]

Obrázek 3.4 (vpravo): Stanice pro zadávání závad přes klávesnici [1]

3.4 Výstupní část SQS – aplikace SQS Global II

SQS Global II je webová aplikace, která slouží k přípravě, analýze a výstupu dat z IS SQS. Tuto aplikaci může používat kterýkoli uživatel intranetu Škody, který má potřebná oprávnění. Samotný vstup do ní probíhá tak, že uživatel klikne na odkaz umístěný na stránkách oddělení GQA, což jej přesune na stránku, kde zadá své uživatelské jméno a heslo do SQS Global II.

Po přihlášení se uživateli zobrazí úvodní stránka aplikace. Ta může být, stejně jako celá aplikace v českém, německém anglickém a ruském jazyce – záleží na uživatelském nastavení. V její levé části můžeme najít menu, které obsahuje seznam závodů Škoda Auto. Po kliknutí na některý z těchto závodů se objeví seznam jeho výrobních provozů, pod nímž uživatel najde příslušné druhy výstupů pro daný provoz. Zde je nutno upozornit, že většina uživatelů si nemůže nechat zpracovat jakýkoli výstup - systém oprávnění SQS Global II je totiž poměrně pružný, a umožňuje administrátorům z GQA nadefinovat ke každému uživatelskému jménu, ke kterým položkám bude mít jím přihlášený uživatel přístup.

Výstupy jsou zajišťovány pomocí sestav. Sestava je v podstatě předdefinovaný dotaz do databáze SQS. Každá sestava je charakteristická parametry, které si uživatel může nastavit a dotaz do databáze tak upravit, a také rozvržením odpovědi z databáze do výstupu. Většina výstupů má podobu tabulky s předem stanovenými atributy, ale některé výstupy mohou obsahovat také graf.



Obrázek 3.5: Úvodní stránka grafického uživatelského rozhraní SQS Global II

Pokud například uživatel klikne na závod Mladá Boleslav, objeví se seznam linek (Montáž A04, Svařovna A5...). Uživatel vybere linku, a zobrazí se seznam možných druhů výstupů - tedy sestav. Pokud je např. vybrána Montáž A04, což je montážní linka pro Fabie, zobrazí se 15 možných sestav, mezi nimiž jsou například:

- Největší závadovost KDNR - ukazuje díly, na kterých se našlo nejvíce závad
- Kmenová data - k zadanému číslu vozu zobrazí detaily o vozu
- Monitoring výroby - ukazuje aktuální počty vyrobených vozů
- Monitoring závad - ukazuje aktuální počty závad
- Seznam závad - ukazuje detaily o závadách za období
- Seznam vozů - ukazuje detaily o vozech za období
- Uvolněné vozy - ukazuje seznam vozů uvolněných na KB8

Výstupy sestav se u každé linky mohou lišit. Například v lakovnách se používají jiné výstupy než na montážích nebo ve svařovnách. Jisté rozdíly ve výstupech jsou dokonce i mezi linkami stejného druhu – např. lakovnou v Mladé Boleslavi a lakovnou v Kvasinách. To je způsobeno tím, že management na těchto linkách používá pro svoji činnost různé

podklady. Některé výstupy (např. Kmenová data, Seznam vozů) jsou ale stejné u všech linek.

Na následujícím příkladu je vybrán výstup Největší závadovost KDNR na lince Montáž A04. Jako první se zobrazí stránka s výběrem základních parametrů – existuje totiž několik stran s parametry, mezi nimiž se dá přepínat pomocí záložek v horní části obrazovky:

Obrázek 3.6: Výběr základních parametrů výstupní sestavy Největší závadovost KDNR

Mezi základními parametry, které jsou víceméně stejné u každé sestavy, je především Sledované období. To omezuje dotaz pouze na vozy, které prošly daným KB v příslušném období. Z důležitých parametrů je tu dále výběr KB, a možnost zobrazit položky součtově. V tomto případě to znamená, že stejné druhy závad budou sloučeny do jednoho řádku výstupní tabulky, zatímco jinak by každé závadě připadl jeden řádek. Dále lze pomocí parametru Počet řádků omezit délku výstupní tabulky. Důležitý je také parametr Použít srovnávací období. Ten umožňuje porovnání dvou výběrů z databáze v jedné výstupní tabulce. Pokud je checkbox *Použít srovnávací období* zaškrtnut, je možné přes záložky na

horní straně obrazovky nastavit parametry pro srovnávací období. Srovnávat lze nejen různá období, ale třeba vozy s dvěma různými motory.

Na obrázku 3.6 jsou parametry sestavy nastavené tak, aby dotaz směřoval pouze na závady mezi 1.1.2006 a 2.6.2006 (22:00). V potaz budou dále brány pouze závady nalezené na KB8, a budou zobrazeny součtově. Ostatní parametry jsou ponechány na standardních hodnotách. Na obrázku 3.7 je pro ilustraci možností, které parametry této sestavy nabízejí, zobrazen obsah záložky Typ vozu. Možnosti jsou opravdu široké – výstup lze omezovat podle barvy, trhu, karoserie, výbavy, motoru a dalších kritérií. Na dalších záložkách jsou k dispozici další možnosti nastavení – ale pro účely této práce se s nimi není třeba podrobně zabývat. Vzhledem k složitosti parametrizace obsahuje SQS Global II možnost nastavení parametrů ukládat a načítat – to umožňují tlačítka v pravém spodním rohu stránky.

Obrázek 3.7: Parametry sestavy Největší závadovost KDNR, záložka Typ Vozu

Jakmile je uživatel s nastavením parametrů spokojen, zbývá mu ještě zvolit formát výstupu (HTML nebo XLS) na spodní straně obrazovky. Pak už jen klikne na jedno z tlačítek *Sestava* nebo *Sestava odložená*. Tlačítko *Sestava* slouží k okamžitému

vypracování výstupu, zatímco tlačítko *Sestava odložená* nechá výstup zpracovat až 15 minut po nejbližším konci směny – to jsou totiž servery SQS nejméně vytížené. Tento postup se používá u složitých výstupů.

Níže na obrázku 3.8 je vidět již hotový výstup z SQS Global II zpracovaný ve formátu HTML. Výstup obsahuje několik částí: Navrchu jsou informace o lince, uživateli, verzi systému a datu a času vypracování. V další části oddělené modrým pruhem jsou informace o výstupu – jaká sestava byla použita, a jsou vypsány také parametry. Tato část je důležitá především pro porovnávání zpráv z SQS. Konečně v poslední části je tabulka s požadovanými údaji.

	Sledované období	Srovnávací období
Počet prošlých vozů (karoserií)	2	
Počet vozů uvolněných	0	
Počet poprvé prošlých vozů	2	
Počet prošlých vozů (karoserií) včetně kolování	2	
Počet prošlých vozů se závadou	2	
Počet prošlých vozů se závadou (%)	100.00 %	%
Počet načtených závad	2	
Další údaje jsou závislé na zvolených parametrech závad a viníků		
Počet prošlých vozů (karoserií) se závadou - dle volby	2	
Počet prošlých vozů (karoserií) se závadou - dle volby (%)	100.00 %	%
Počet načtených závad - dle volby	2	
Prům. počet závad na vůz (karoserií) - dle volby	1.00	

Díl	Typ závady	Sledované období			Srovnávací období		
		Počet závad	[%]	Závad na 100 vozů	Počet závad	[%]	Závad na 100 vozů
5851-Dveře zadní	Deformace - svař.	1	50.00	50.00			
6329-Nárazník přední	Poškozené	1	50.00	50.00			

Kmencová data

Obrázek 3.8: Výstup ze sestavy „Největší závadovost KDNR“

Výstup zpracovaný na obrázku 3.8 je ze sestavy Největší závadovost KDNR. V tomto případě je velmi krátký, protože sledované období bylo po skončení vánočních svátků, kdy se výroba teprve rozbíhala. Normálně by výstup obsahoval mnoho položek, a byl dlouhý

několik stran. Z tohoto výstupu se dá pouze odvodit, že mezi 1.1.2006 22:00 a 2.1. 2006 22:00 prošly KB8 dva vozy, a byly nalezeny dvě chyby – poškozený nárazník a deformace předních dveří.

Na tomto příkladu byla dobře vidět hlavní funkce SQS Global II a principy, na kterých stojí. Další důležité funkce, které tato aplikace poskytuje jsou např. Sledování / Blokování vozů či Alarmová hlášení. U Sledování/Blokování vozu si uživatel vybere určitý vůz v procesu výroby, může pak sledovat jeho průchody KB a závady na něm. Informace o tom, že vůz je sledovaný se zobrazí také obsluze KB. Uživatel SQS Global II může také vůz blokovat – tedy dát příkaz k zákazu průchodu určitým KB. Tento postup se používá např. u vozů určených pro výstavy. Funkce Alarmová hlášení pak posílá uživateli (např. mistrovi) e-mail, pokud se někde objeví typ závady, který si předem nadefinoval, případně vyšší množství těchto závad.

3.5 Projekt statistického zpracování výstupů SQS Global II

Tato část práce se věnuje projektu statistického zpracování výstupů (SZV) z roku 2005. Cílem tohoto projektu bylo vyhodnocení využívanosti výstupů poskytovaných SQS z různých hledisek tak, aby měli správci systému dokonalý přehled o tom, co, kdy, jak a kým je používáno. Ačkoliv se jisté kroky v oblasti SZV provedly již dříve, nebyly dotaženy k viditelným výsledkům. Důvod byl ten, že se k problému přistupovalo poměrně neefektivním způsobem. Původní nápad byl totiž takový, že GQA si od firmy Gedas vyžádá údaje o zpracovaných výstupech za dobu existence aplikace SQS Global II, ty budou převedeny do MS Excelu a analyzovány. Nicméně proces převádění do MS Excelu z formy čistého textu (plain text), v jaké byly výstupy dodávány by byl poměrně problematický, analýza a vyhodnocení těchto dat by také přinášela značné obtíže, a především by získané informace o užívání systému nebyly aktuální. Spolu s Ing. Greplem jsme se proto dohodli na jiném přístupu: Funkce SZV budou poskytovány modulem implementovaným přímo do aplikace SQS Global II.

Projekt byl započat v dubnu 2005, kdy jsem začal s jednáními s firmou Gedas o jeho podrobnostech. V červnu 2005 byly Gedasu předány specifikace nástroje pro SZV, v září 2005 byl nástroj připraven k nasazení a s drobnými změnami uveden do provozu. V nynější době je nástroj úspěšně používán k tvorbě pravidelných zpráv o užívání SQS Global II.

Následuje popis někdejších cílů SZV a zhodnocení jejich splnění. Některé cíle nebyly naplněny zcela beze zbytku, a staly se proto součástí cílů této diplomové práce.

Mezi hlavní cíle SZV patřily tyto:

- Odstranění příčin přetěžování serveru

Při provozu SQS Global II se stává, že z důvodu přetížení serveru není zpracování výstupů funkční nebo se výstupy zpracovávají jen velmi pomalu. To může být zapříčiněno buď zpracováváním mnoha výstupů ve stejném okamžiku nebo jednotlivými uživateli, kteří např. špatně zadali časové rozpětí v parametrech sestavy a server je pak jejich požadavkem vytížen velmi dlouhou dobu. SZV je schopno odhalit uživatele, kteří zatěžují server nesprávně zadanými či příliš složitými výstupy. Těm pak může být poskytnuto školení v používání systému.

Problém přetěžování serveru byl SZV vyřešen zčásti – použité řešení sice umožňuje sledovat činnost uživatelů v minulosti a následně je například kontaktovat či proškolit, neumožňuje však sledovat vytíženost serverů on-line a podnikat kroky, které by vedly k okamžitému snížení vytížení – jako například okamžité stornování dotazů apod. Realizace těchto funkcí by však vyžadovala řešení náročná jak implementačně, tak i finančně, a proto od ní bylo upuštěno.

- Získání informací pro podporu systému

Správci SQS v oddělení GQA zajišťují v rámci podpory systému také telefonickou a e-mailovou podporu uživatelů. Při řešení problémů s SQS, které uživatelé ohlašují, je často třeba dohledat, které kroky daný uživatel v systému provedl. Tyto informace se zjišťovaly buďto rozhovorem s uživatelem, nebo telefonátem do Gedas, kde programátoři vyhledali

v databázi potřebné údaje. S nástrojem SZV mají správci systému možnost se kdykoli podívat jaké výstupy uživatel spouštěl v určitém období a zda byly zpracovány v pořádku.

V tomto bodě byly cíle splněny beze zbytku. Získané informace se ukázaly být velice užitečnými, a to jak pro telefonickou a e-mailovou podporu, tak i pro organizaci školení nebo komunikaci s hlavními uživateli SQS.

- **Optimalizace sestav pro různé linky**

Jak jsem již zmínil, SQS Global II nabízí zhruba stejné druhy výstupů (sestavy) pro všechny linky, které ale nemusí být využívány všemi linkami stejným způsobem a ve stejné míře. Pomocí SZV je možné zjistit, které sestavy na různých linkách uživatelé používají, a které jsou pro ně naopak zbytečné. Také odstraněním či úpravou parametrizací těchto sestav by bylo možno zvýšit přehlednost aplikace.

Toto je bod, který v původním projektu nebyl naplněn. Modul SZV umí poskytnout data týkající se využívání sestav na různých linkách, zatím tato data ale nebyla zpracována. Modul SZV ovšem již nedovede sbírat údaje o použitých parametrech, aby následně mohly být parametrizace sestav pro různé linky upraveny.

Problémem úpravy parametrizací sestav pro různé linky se proto zabývám v této diplomové práci.

4 Optimalizace informačního systému SQS

4.1 Stanovené cíle

Cíle, které jsem si v této práci stanovil, vycházejí z potřeb a přání správců informačního systému SQS v oddělení GQA ve Škoda Auto. První z nich jsou změny v parametrizacích sestav.

Každá sestava v SQS Global II umožňuje stanovit hodnoty parametrů, které upravují dotaz odesílaný na databázový server. Těchto parametrů nastavitelných dle přání uživatele však většina sestav obsahuje tolik, že se tím uživatelské rozhraní SQS Global II stává pro uživatele nepřehledné. Mnoho parametrů také není uživateli využito vůbec, nebo je využito pouze několika málo uživateli se specifickými funkcemi. Změna parametrizací sestav tak, aby byly správně reflektovány požadavky uživatelů a zároveň se zpřehlednilo uživatelské rozhraní, je proto logickým krokem v optimalizaci informačního systému SQS a také nejdůležitějším cílem této práce.

Druhým cílem této práce je zachycení závislosti užívání jednotlivých sestav na čase. To umožní správcům informačního systému SQS zjistit, jakým způsobem se struktura požadavků na SQS v průběhu pracovního dne či týdne mění. Tato informace by jim měla pomoci především s uživatelskou podporou SQS.

4.2 Metody zvolené k dosažení cílů

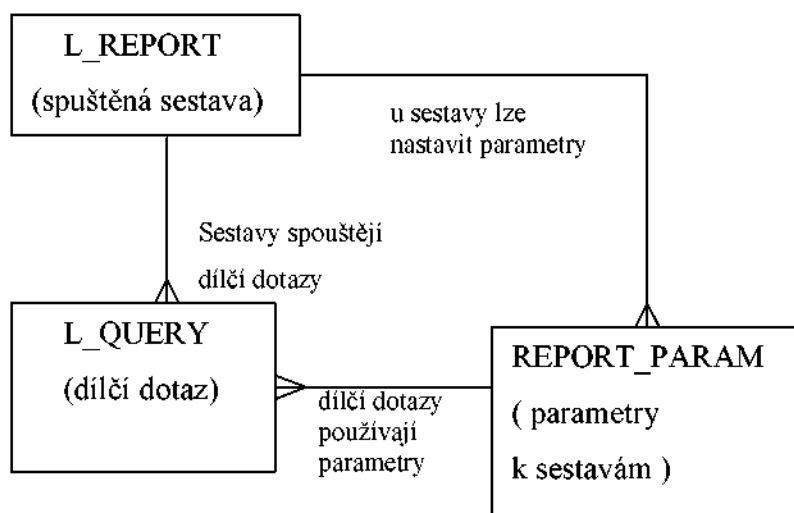
K dosažení obou stanovených cílů je třeba dvou kroků; prvním z nich je zjištění požadavků uživatelů na informační systém SQS, druhým krokem je pak tyto požadavky odpovídajícím způsobem zpracovat a vyhodnotit. V části 4.3 popisují způsob získání příslušných dat, jejich strukturu a programové prostředky, pomocí kterých jsem data zpracoval. V dalších částech jsou pak popsány metody, které jsou obsaženy v algoritmech pro zpracování dat a které také slouží k vyhodnocení výsledků. V části 4.4 je teoreticky popsána statistická metoda použitá pro analýzu parametrizace sestav, χ^2 test nezávislosti v kontingenční tabulce a vzorce pro výpočet použitých koeficientů kontingence. Část 4.5 se týká teorie pravděpodobnosti a podmíněných pravděpodobností, jež jsou použity pro zachycení závislosti užívání jednotlivých sestav na čase.

4.3 Data a prostředky pro jejich zpracování

Inspirován konstrukcí samotného SQS a funkcí jeho statistického modulu jsem se pro účely této práce rozhodl využít ke zjištění požadavků uživatelů na SQS stejný princip, na

kterém pracuje tento informační systém, tedy zpracování dat uložených v databázích pomocí dotazů SQL a webových programovacích jazyků. Klíčová výhoda tohoto řešení spočívá v tom, že data týkající se činnosti uživatelů již jsou SQS sbírána a ukládána do databází pro potřeby statistického modulu SQS a programátorů firmy Gedas. Sám statistický modul k dosažení cílů práce sice využít nelze, data v databázích je nicméně možné použít i pro zpracování jiným způsobem.

Pro potřeby této práce bylo třeba mít data týkající se spuštěných sestav. Nutné bylo znát především uživatele, čas spuštění a druh sestavy, dále parametry, které uživatel použil a také samozřejmě všechny parametry, které je u dané sestavy možné použít. Díky struktuře informačního systému SQS se mi všechna tato data podařilo získat z jeho databází, konkrétně z tabulek s názvy L_REPORT, L_QUERY a REPORT_PARAM. Celkově jde asi o dva miliony záznamů týkající se činnosti uživatelů za přibližně 5 měsíců fungování informačního systému SQS.



Obrázek 4.1: Jednoduchý ER model získaných dat

Tabulka L_REPORT je uchovávána primárně pro statistický modul SQS. Jeden záznam v ní odpovídá jednomu spuštění sestavy uživatelem. Obsahuje následující atributy:

Atribut *idlog* je primárním klíčem tabulky, slouží především k propojení tabulek.

Atribut *idreport* identifikuje spouštěnou sestavu.

Atribut *dstart* označuje datum a čas, kdy byl dotaz uživatele přijat serverem.

Atribut *cas* ukazuje, za jak dlouho byl výstup zpracován. Pokud je hodnota atributu rovna -1, pak bylo zpracování přerušeno.

Atribut *automat* ukazuje, zda se jedná o automaticky zpracovávaný výstup (SQS jej periodicky zpracovává sám) – A=Ano, N=Ne

Atribut *sqsuser* označuje uživatele, který výstup zadal.

Atribut *ip* ukazuje, na jaký server byl dotaz směřován.

Atribut *lang* specifikuje, ve kterém jazyce byl výsledek uživateli zaslán.

Atribut *linka_fyz* ukazuje, které linky se dotaz týká.

idlog	idreport	dstart	cas	automat	sqsuser	ip	lang	linka_fyz
3004810	SV	2007-08-14 20:45:00	.62	N	U01434	10.217.16.143		20
3004811	ZMET	2007-08-14 20:45:00	3.35	N	LARRYD	10.217.16.143		6
3004812	SV	2007-08-14 20:45:00	.6	N	U01434	10.217.16.143		20
3004813	logon	2007-08-14 20:47:00	121.25		U00250	10.217.16.143		0
3004814	KDNZ	2007-08-14 20:48:00	1.18	N	U00250	10.217.16.143		16
3004815	logon	2007-08-14 20:49:00	.117		U00896	10.217.16.143		0
3004816	SV	2007-08-14 20:50:00	.68	N	U01434	10.217.16.143		20
3004817	SV	2007-08-14 20:51:00	.72	N	U01434	10.217.16.143		20
3004818	KMENDATA	2007-08-14 20:51:00	1.71	N	U01434	10.217.16.143		20

tabulka 4-1: Výpis z tabulky L_REPORT z databáze SQS

Další tabulkou použitou v práci je L_QUERY, v které každý záznam odpovídá dílčímu dotazu (poddotazu), který se spustí po spuštění určité sestavy, přičemž jedna sestava může spustit více takovýchto dílčích dotazů. U některých poddotazů jsou také použity parametry sestavy, které byly zadány uživatelem.

Atribut *idlog* se odkazuje na sestavu, která danému dílčímu dotazu odpovídá.

Atribut *idstep* ukazuje, v kterém kroku zpracování byl dílčí dotaz proveden.

Atribut *qname* ukazuje název dílčího dotazu.

Atribut *err_pos* ukazuje případnou chybu ve zpracování.

Atribut *bindvars* obsahuje atributy a jejich hodnoty, které byly při zpracování dílčího dotazu použity.

idlog	idstep	qname	err_pos	bindvars
3007056	3	KD_BGRP		:IDENT:=G004331330;
3007056	3	KD_BGRP		:IDENT:=G004332377;
3007058	1	SYSDATE		
3007235	6	KD_POHYB		:IDENT:=G003760473;
3007235	7	AGOS_HL		:IDENT:=G003760473;
3007235	7	AGOS_ZAV		:IDENT:=G003760473; :LNGDECODE:=C;
3007237	1	SYSDATE		

tabulka 4-2: Výpis z tabulky L_REPORT z databáze SQS

Poslední použitou tabulkou je REPORT_PARAM, která ukazuje všechny parametry, které je u sestav možné vyplnit. Jeden záznam v této tabulce odpovídá jednomu parametru některé ze sestav. Jak již bylo zmíněno, parametrů sestavy bývá velké množství, proto je nelze zadat všechny na jedné „obrazovce“. U většiny sestav proto musí uživatelé pro vyplnění hodnoty určitého parametru nalistovat záložku, na které je parametr možné zadat.

Atribut *idreport* identifikuje sestavu, ke které parametr patří.

Atribut *idbookmark* ukazuje, na které záložce lze hodnotu parametru zadat.

Atribut *idparam* označuje název parametru.

Atribut *o* ukazuje, zda je parametr pro sestavu povinný (ano/ne).

Atribut *m* říká, zda jestli parametr má pro sestavu povolen výběr více hodnot (ano/ne).

Atribut *s* říká, zda se má parametr zobrazovat v opisu parametrů (při zpracování dotazu).

Atribut *poradi* označuje pořadí v opisu parametrů (pokud je hodnota param. s ano).

idreport	idbookmark	idparam	o	m	s	poradí
AUDIT	1	AUDITOR	A	N	A	5
AUDIT	1	AU_TAB	A	A	N	0
AUDIT	1	AU_TABH	A	N	N	0
AUDIT	1	DATUM2	N	N	A	1
AUDIT	1	FORMAT_REPORT	N	N	N	0
AUDIT	1	KKOLEKTIV	A	N	A	4
AUDIT	1	SMENA	A	N	A	3
AUDIT	1	TYPCHECKU	A	N	A	2

tabulka 4-3: Výpis z tabulky REPORT_PARAM z databáze SQS

Výpisy z tabulek s údaji za posledních cca 5 měsíců mi byly se svolením zodpovědných osob ze společnosti Škoda Auto exportovány do textových souborů. Do výpisů jsem si nechal udělat také řídící znaky, které mi umožnily identifikovat jednotlivé záznamy a jejich atributy. Tyto znaky mi umožnily takto uložená data převést do syntaxe jazyka SQL – konkrétně příkazu *insert* ukládajícího data do záznamů v tabulce, a uložit je do relační databáze, kterou jsem si vytvořil v open-source databázovém serveru MySQL. K samotnému převedení jsem využil jednoduchý program Search and Replace, kterým jsem řídící znaky nahradil za klíčová slova a znaky příkazu *insert*, a funkci importu aplikace PHPMyAdmin pro MySQL. Ačkoliv by samozřejmě bylo možné s daty pracovat v původní formě přímo v textovém souboru, zvolený způsob opětovného převodu do formy relační databáze slibuje oproti nutnosti programovat veškeré procedury pro vyhledávání a porovnávání datových záznamů v programovacím jazyce jako třeba C nebo Pascal podstatně snadnější práci s daty pomocí jazyka SQL. Stejně tak je výhodou i mnohem vyšší rychlost zpracování dat. Zkoumání složitých vztahů mezi jednotlivými datovými záznamy by v případě uložení v textovém souboru, resp. ve více textových souborech bylo velmi náročnou procedurou, zatímco relační databáze jsou pro taková data a operace s nimi přímo určeny.

Server: localhost:3306 ▶ Databáze: diplomka ▶ Tabulka: L_report

Projít Struktura SQL Vyhledávání Vložit Export Import Úpravy

Vyprázdnit Odstranit

Sloupec	Typ	Porovnávání	Vlastnosti	Nulový	Výchozí	Extra	Akce
☐ idlog	int(11)			Ne			
☐ idreport	varchar(30)	cp1250_czech_cs		Ne			
☐ dstart	datetime			Ne			
☐ cas	varchar(10)	cp1250_czech_cs		Ne			
☐ automat	varchar(1)	cp1250_czech_cs		Ne			
☐ sqsuser	varchar(6)	cp1250_czech_cs		Ne			
☐ ip	varchar(15)	cp1250_czech_cs		Ne			
☐ lang	varchar(2)	cp1250_czech_cs		Ne			
☐ linka_fyz	int(11)			Ne			

↑ Zaskrtnout vše / Odškrtnout vše Zaskrtnuté:

Náhled k vytištění Navrhnout strukturu tabulky

Přidat 1 sloupců Na konci tabulky Na začátku tabulky Po idlog Proved

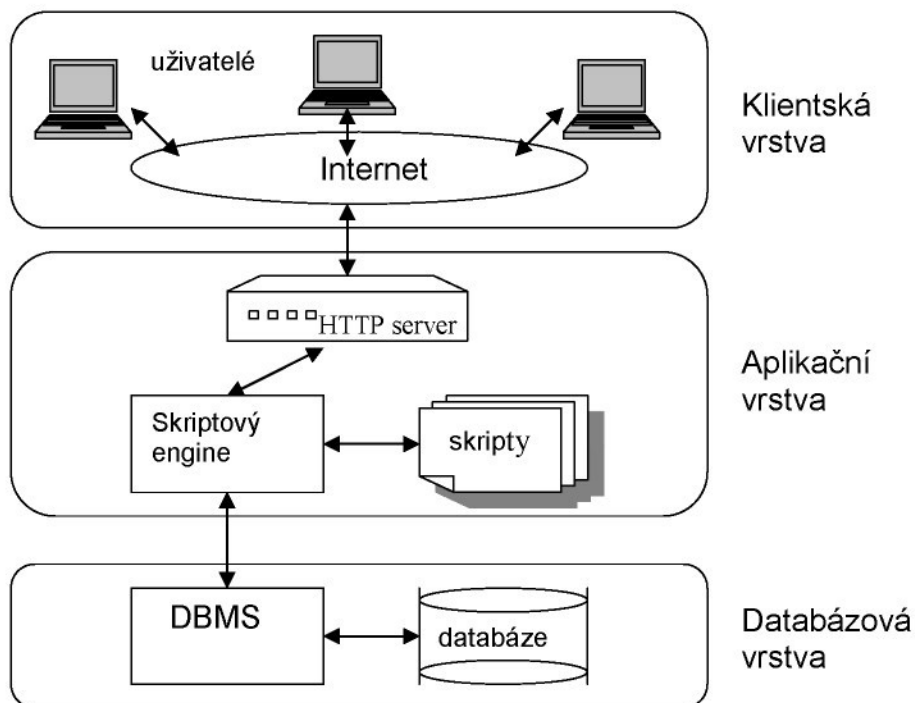
Indexy:					Využití místa	
Klíčový název	Typ	Mohutnost	Akce	Sloupec	Typ	Používá
PRIMARY	PRIMARY	348715		idlog	Data	17 865 KiB
dstart	INDEX	116238		dstart	Index	18 605 KiB

Obrázek 4.2: Importovaná data v MySQL, prostředí PHPMyAdmin

Pro zpracování dat v domácích podmínkách jsem zvolil programovací jazyk PHP, jehož interpreter je spolu již s výše zmíněným databázovým serverem MySQL a webovým HTTP serverem Apache obsažen ve freewarovém balíku pro Windows s názvem VertrigoServ. Nástroje obsažené v tomto balíku mi posloužily k uložení, zpracování i prezentaci dat, jedná se o tzv. třívrstvou architekturu používanou pro webové databázové aplikace.

Základem každé webové databázové aplikace je databázová vrstva tvořená databázovým systémem – DataBase Management System, česky Systém Řízení Báze Dat [3]. Ten spravuje samotnou databázi s daty a zajišťuje jejich vkládání, odstraňování, modifikaci a odpovědi na dotazy. Nad databázovou vrstvou je aplikační vrstva zajišťující aplikační logiku a komunikaci mezi vrstvami. Konečně nejvyšší klientská vrstva zajišťuje prezentační služby a prezentační logiku.

[3] DELISLE, M. *phpMyAdmin – efektivní správa SQL*, str. 2



Obrázek 4.3: Třívrstvá architektura webových aplikací [3]

V mnou použitém řešení tvoří databázovou vrstvu server MySQL, se kterým aplikační vrstva obsahující interpreter jazyka PHP a HTTP server Apache komunikuje pomocí jazyka SQL. Klientskou vrstvu pak zajišťuje běžný webový prohlížeč komunikující s Apache pomocí protokolu HTTP.

4.4 Testování nezávislosti v kontingenční tabulce

Kontingenční tabulka je dvourozměrná tabulka, kam jsou uspořádány údaje o dvou kategoriálních proměnných [4]. Obvykle se používá k tomu, aby ukázala závislost jednoho statistického znaku na druhém. K důkazu, že na sobě tyto statistické znaky závisí se používá χ^2 test o nezávislosti v kontingenční tabulce, který patří mezi neparametrické testy statistických hypotéz.

4.4.1 Testování statistických hypotéz

Statistická hypotéza je postup, pomocí kterého na základě náhodného výběru testujeme určité předpoklady o základním souboru – např. předpoklad o parametru či tvaru rozdělení zkoumaného znaku [5]. Takový předpoklad nazveme nulovou (testovanou) hypotézou, značíme ji H_0 . Proti nulové hypotéze se pak postaví alternativní hypotéza značená H_1 , která hypotézu H_0 nějakým způsobem popírá. Takto formulovaná hypotéza se nazývá dvoustranná hypotéza a test hypotézy dvoustranným testem.

Vzhledem k tomu, že při testování hypotézy jde o úsudek prováděný z údajů získaných náhodným výběrem, je možné se dopustit chybného závěru. V případě, že zamítneme testovanou hypotézu H_0 , ačkoliv ve skutečnosti platí, dopouštíme se chyby prvního druhu. Pravděpodobnost této chyby se značí α . Druhá možnost chybného závěru spočívá v tom, že je nulová hypotéza H_0 přijata, ačkoliv ve skutečnosti platí alternativní hypotéza H_1 . V tomto případě se jedná o chybu druhého druhu, jejíž pravděpodobnost je rovna β . Vztah mezi oběma druhy chyb zobrazuje následující tabulka:

Skutečnost > Úsudek o H_0 V	H_0 je pravdivá	H_0 je nepravdivá
	pravděp.	pravděp.
Nezamítá se	správné rozhodnutí $1-\alpha$	chyba I.druhu β
Zamítá se	chyba I.druhu α	správné rozhodnutí $1-\beta$
Celkem	1	1

tabulka 4-4: Chyby I. a II. Druhu a jejich pravděpodobnosti [5]

Pravděpodobnost $1-\beta$ se nazývá síla testu. Ta vyjadřuje, s jakou pravděpodobností je nulová hypotéza H_0 zamítnuta, pokud ve skutečnosti platí H_1 , to znamená s jakou pravděpodobností se nedopustíme chyby I. druhu. Běžně se při testování hypotéz předem zvolí pevná pravděpodobnost chyby I. druhu, tzv. hladina významnosti. Testovací postup je odvozen tak, aby při dané hladině významnosti zajišťoval minimální pravděpodobnost chyby II. druhu, a tedy maximální sílu testu.

[5] HINDLS, R., HRONOVÁ, S., SEGER, J., *Statistika pro ekonomy*, str. 133-138

U každého standardního testu je uvedeno, jaké použít testové kritérium. Množina hodnot, kterých může testové kritérium nabýt, se nazývá výběrový prostor a je značena S . Aby však bylo možné na základě hodnoty testového kritéria rozhodnout, zda bude přijata nulová hypotéza H_0 či alternativní hypotéza H_1 , je nutné rozdělit výběrový prostor S na dva podprostory - podprostor V nazývaný se obor přijetí, který obsahuje hodnoty svědčící ve prospěch H_0 , a podprostor W nazývaný se kritický obor, který obsahuje hodnoty svědčící ve prospěch H_1 . Platí, že průnik těchto podprostorů je prázdný, zatímco jejich sjednocení je rovno výběrovému prostoru S . Hranice, která podprostory V a W odděluje, se nazývá kritická hodnota.

Testovací postup začíná formulací hypotéz. Formulujeme dvojici hypotéz, H_0 a H_1 . Dále zvolíme testové kritérium, což je statistika – funkce výběru. Výpočet jeho hodnoty je při testování hypotéz cílem zpracování hodnot náhodným výběrem. Aby bylo možné stanovit kritický obor, je třeba znát rozdělení testového kritéria při platnosti hypotézy H_0 . Konečně je třeba sestavit kritický obor, který bude tak velký, aby bylo zajištěno, že chyby I. druhu se dopustíme jen ve $100 \times \alpha \%$ případů.

Při znalosti rozdělení testového kritéria při platnosti hypotézy H_0 je možné říci, že kritickými hodnotami budou kvantily rozdělení testového kritéria při platnosti H_0 . Pokud označíme toto rozdělení T , bude kritický obor představován následujícími intervaly:

- $W = (T_{\min} ; T_{\alpha} > v$ případě, že ve prospěch H_1 svědčí vysoké hodnoty testového kritéria
- $W = (T_{1-\alpha} ; T_{\max} > v$ případě, že ve prospěch H_1 svědčí nízké hodnoty testového kritéria
- $W = (T_{\min} ; T_{\alpha/2} > U < T_{1-\alpha/2} ; T_{\max})$ v případě, že ve prospěch H_1 svědčí jak extrémně nízké, tak extrémně vysoké hodnoty testového kritéria.

Dalším krokem testování statistické hypotézy je výpočet hodnoty testového kritéria. Podle toho, zda jeho hodnota patří do množiny V nebo W můžeme formulovat závěr testu:

- Pokud hodnota testového kritéria patří do množiny V , tedy oboru přijetí, nebyla hypotéza H_1 testem prokázána. Při prohlášení, že platí H_0 , je riziko omylu rovno β .
- Pokud hodnota testového kritéria patří do množiny W , tedy kritického oboru, byla hypotéza H_1 testem prokázána. Při prohlášení, že platí H_1 , je riziko omylu rovné α .

Velikost chyby I. druhu β nemůžeme znát, pokud neprostudujeme rozdělení testového kritéria při platnosti H_1 . Testové kritérium je sice u standardních testů zvolené tak, aby bylo riziko I. druhu co nejmenší, přesto je ale obvyklé vyhnout se výroku o platnosti H_0 a omezit se na závěr o platnosti hypotézy H_1 .

4.4.2 Neparametrické testy

Oproti testům parametrickým, kde je konstrukce odhadů charakteristik základního souboru a testů hypotéz vázána na předpoklad, že rozdělení základního souboru je určitého typu, u testů neparametrických rozdělení základního souboru neznáme. Mezi nejvýznamnější neparametrické testy patří testy dobré shody (χ^2 test, Kolmogorovův-Smirnovův test, testy nezávislosti v kontingenční tabulce (χ^2 test, test ve čtyřpolní tabulce), testy o shodě úrovně a další. Vlastností typickou pro všechny neparametrické testy je to, že na rozdíl od testů parametrických nevyžadují znalost rozdělení zkoumané veličiny. Tato vlastnost je i jejich hlavní předností, ačkoliv má v některých případech za následek snížení síly testu.

4.4.3 χ^2 test nezávislosti v kontingenční tabulce

Kontingenční tabulka se používá k přehlednému zobrazení vzájemného vztahu dvou statistických znaků. Dostaneme ji tak, že třídíme jednotky souboru podle variant dvou kvalitativních znaků [5]. Znaky označeny A a B, znak A má r variant a znak B s variant, které jsou uvedeny v hlavičce tabulky. Četnosti v jednotlivých políčkách jsou označeny jako n_{ij} . První index označuje i -tou variantu znaku A, druhý j -tou variantu znaku B.

[5] HINDLS, R., HRONOVÁ, S., SEGER, J., *Statistika pro ekonomy*, str. 160-162

Okrajové četnosti mají jeden index, tečkou je označeno sčítání. To znamená, že $n_{i.}$ je součet všech četností v i -té řádce, zatímco $n_{.j}$ součet četností v j -tém sloupci.

A\B	B1	B2	...	Bj	...	Bs	Σ_i
A ₁	n_{11}	n_{12}	...	n_{1j}	...	n_{1s}	$n_{1.}$
A ₂	n_{21}	n_{22}	...	n_{2j}	...	n_{2s}	$n_{2.}$
...	...	...	...	...	...	...	...
A _i	n_{i1}	n_{i2}	...	n_{ij}	...	n_{is}	$n_{i.}$
...	...	...	...	...	...	...	...
A _r	n_{r1}	n_{r2}	...	n_{rj}	...	n_{rs}	$n_{r.}$
Σ_j	$n_{.1}$	$n_{.2}$	...	$n_{.j}$	...	$n_{.s}$	n

tabulka 4-5: Kontingenční tabulka [5]

Test, který se používá k ověření nezávislosti v kontingenční tabulce, se nazývá χ^2 test nezávislosti. Spočívá v porovnání empirických četností s očekávanými v případě nezávislosti. Nezávislost v kontingenční tabulce znamená, že se oba znaky navzájem neovlivňují v tom, jakých hodnot nabývají. Hypotéza H_0 říká, že oba znaky jsou na sobě navzájem nezávislé, tedy že očekávané četnosti jsou rovny empirickým, zatímco hypotéza H_1 předpokládá závislost dvou znaků v kontingenční tabulce. Očekávané (teoretické) četnosti se označují jako n'_{ij} a pro $i = 1, 2, \dots, r$ a $j = 1, 2, \dots, s$ platí vztah:

$$n'_{ij} = \frac{n_{i.} \cdot n_{.j}}{n} \quad (4.1)$$

Testovým kritériem je pak statistika G :

$$G = \sum_{i=1}^r \sum_{j=1}^s \frac{(n_{ij} - n'_{ij})^2}{n'_{ij}} \quad (4.2)$$

Statistika G má v případě nezávislosti a v případě velkého počtu pozorování přibližně χ^2 rozdělení s $(r-1)(s-1)$ stupni volnosti. Jestliže pak hodnota testového kritéria G překročí kritickou hodnotu, kterou je kvantil rozdělení χ^2 s $(r-1)(s-1)$ stupni volnosti, zamítáme na

dané hladině významnosti hypotézu H_0 a považujeme závislost obou kvalitativních znaků za prokázanou.

4.4.4 Měření závislosti kategoriálních veličin

Pokud jsme pomocí χ^2 testu prokázali závislost znaků v kontingenční tabulce, je možné použít k měření síly této závislosti koeficienty kontingence. Pro tabulky s větším počtem řádků a sloupců než 2 je možné použít Cramerův koeficient kontingence, Spearmanův koeficient kontingence a Čuprovův koeficient kontingence.

Cramerův koeficient kontingence může nabývat hodnot od 0 (úplná nezávislost) po 1 (úplná závislost). Lze jej vypočítat podle následujícího vztahu:

$$C_c = \sqrt{\frac{G}{n \cdot \min(r-1, s-1)}} , C_c \in \langle 0, 1 \rangle \quad (4.3)$$

Pearsonův koeficient kontingence nabývá hodnot podobně jako Cramerův koeficient, až na to že nemůže nikdy dosáhnout hodnoty 1. Lze jej vyjádřit tímto vztahem:

$$C_p = \sqrt{\frac{G}{G+n}} , C_p \in \langle 0, 1 \rangle \quad (4.4)$$

Pro Čuprovův koeficient kontingence platí tento vztah:

$$K = \sqrt{\frac{G}{n \cdot \sqrt{(r-1)(s-1)}}} , K \in \langle -1, 1 \rangle \quad (4.5)$$

4.5 Bayesův teorém podmíněné pravděpodobnosti

Pravděpodobnost náhodného jevu je číslo, které je mírou očekávatelnosti výskytu jevu [6]. Náhodným jevem rozumíme opakovatelnou činnost prováděnou za stejných (nebo přibližně stejných) podmínek, jejíž výsledek je nejistý a závisí na náhodě. Pravděpodobnost události se obecně označuje reálným číslem od 0 do 1. Událost, která nemůže nastat, má pravděpodobnost 0, a naopak jistá událost má pravděpodobnost 1.

Bayesův teorém (někdy také Bayesovo pravidlo nebo Bayesův zákon) je část teorie pravděpodobnosti, která srovnává podmíněné a absolutní rozdělení pravděpodobnosti náhodných proměnných.

Náhodný jev se určuje vždy k určitým podmínkám. Pokud nejsou na výskyt daného jevu A kladeny žádné další podmínky, potom pravděpodobnost $P(A)$ jevu A označujeme jako nepodmíněnou pravděpodobnost. V případě, že se jev A může vyskytnout pouze tehdy, vyskytl-li se jev B, jehož pravděpodobnost je $P(B) > 0$, pak můžeme mluvit o podmíněné pravděpodobnosti jevu A a označujeme ji $P(A | B)$. Při $P(B) > 0$ je možné pravděpodobnost jevu A, která je podmíněna výskytem jevu B vyjádřit jako

$$p(A | B) = \frac{p(A \cap B)}{p(B)} = \left| \frac{A \cap B}{B} \right| \quad (4.6)$$

Pravděpodobnost události A za podmínky události B je obecně rozdílná od pravděpodobnosti události B za podmínky události A. Přesto však mezi nimi existuje vztah, a právě Bayesův teorém je vyjádřením tohoto vztahu.

Jako formální vztah je Bayesův teorém platný pro všechny interpretace pravděpodobnosti. Panují však rozpory týkající se objektů, jimž mohou být pravděpodobnosti přiřazeny: Frekventisté přiřazují pravděpodobnosti náhodným jevům v souladu s frekvencí jejich výskytu, zatímco Bayesiánci přiřazují pravděpodobnosti výrokům, které nejsou jisté.

Bayesův teorém srovnává podmíněné a absolutní pravděpodobnosti stochastických jevů A a B:

$$p(A|B) = \frac{p(B|A) \cdot p(A)}{p(B)} \quad (4.7)$$

kde $p(A)$ a $p(B)$ jsou absolutní (nepodmíněné) pravděpodobnosti jevů A a B, $p(A|B)$ je podmíněná pravděpodobnost jevu A za podmínky B, a $p(B|A)$ je podmíněná pravděpodobnost jevu B za podmínky A.

4.5.1 Bayesovské sítě

Bayesovská síť je acyklický orientovaný graf, jehož uzlům jsou přiřazeny veličiny [7]. Její název je odvozen ze skutečnosti, že pravděpodobnostní míra je reprezentována strukturou, jejíž podstatnou částí je graf. Aby se acyklický orientovaný graf stal bayesovskou sítí, je třeba ho doplnit systémem podmíněných pravděpodobnostních měr. Předpokládáme, že bayesovská síť pak vyjadřuje závislostní strukturu pravděpodobnostní míry popisující závislosti v oblasti našeho zájmu. To znamená, že pokud graf neobsahuje přímé spojení mezi dvěma uzly, pak jsou veličiny odpovídající těmto uzlům na sobě podmíněně nezávislé. Za tohoto předpokladu, který se nazývá také předpokladem úplnosti, odpovídá bayesovské síti jediná pravděpodobnostní míra, kterou dostaneme jako součin všech podmíněných měr přiřazených uzlům.

Bayesovské sítě jsou také často skloňovanou metodou v oblastech umělé inteligence a znalostních systémů, kde se používají pro modelování znalostí.

Aby bylo možné navrhnout bayesovskou síť, je nutné u každého uzlu specifikovat rozdělení pravděpodobnosti v závislosti na „rodičích“ tohoto uzlu [8]. Toto rozdělení může mít jakoukoli formu, nejčastěji se ovšem pracuje s Gaussovým rozdělením. process.) Tato

[7] MAŘÍK V., ŠTEPÁNKOVÁ O., LAŽANSKÝ J. a kolektiv: Umělá inteligence 4.díl, str. 98-101

rozdělení také zahrnují parametry, jež nejsou známy a musí být odhadnuty z dat pomocí různých algoritmů.

V nejjednodušších případech je bayesovská síť navržena expertem a následně používána pro odvozování závěrů. V mnoha případech je však navržení bayesovské sítě pro lidi příliš složitou úlohou. V takovém případě musí být struktura sítě a parametry rozdělení pravděpodobnosti získány přímo z dat strojovým učením.

4.6 Analýza parametrizace sestav

4.6.1 Cíl analýzy

V současné době obsahuje informační systém SQS Global II přes 60 různých sestav, tedy předdefinovaných dotazů, které jsou využívány více než 450 uživateli. Každá sestava je charakterizována možností nastavení parametrů, kterými může uživatel dotaz dále upřesnit. Parametrů je u většiny sestav velké množství, což sice umožňuje položit velmi specifický dotaz, nicméně uživatelské rozhraní systému se tak stává velice nepřehledné, neboť uživatelé musejí hledat parametry dotazu, jejichž hodnoty by rádi nastavili mezi mnoha jinými nepotřebnými. U většiny sestav je nutné parametry nastavovat na více záložkách, protože se nevejdou na jednu stránku. V některých případech jde až o 9 záložek, což zadávání dotazu značně komplikuje, a nezkušený uživatel systému pak může narazit na problémy s vyhledáním správných parametrů dotazu.

Lze předpokládat, že ne všechny parametry sestav jsou využívány všemi uživateli na všech linkách; proto by mohlo mít smysl zjednodušit parametrizace např. zavedením parametrizace specifické pro každého uživatele, nebo pro každou linku. Takové úpravy v informačním systému SQS nepředstavují velkou překážku, je však nutné nejprve zjistit, jakým způsobem by bylo nejvhodnější parametrizace změnit.

Pro zkoumání vhodnosti různých alternativ jsem vybral χ^2 test o nezávislosti v kontingenční tabulce spolu s Cramerovým koeficientem kontingence. Pomocí těchto nástrojů je možné zjistit, zda se dotazy různých uživatelů významným způsobem liší co se týče použitých parametrů, a mělo by tedy smysl zavést parametrizace specifické pro uživatele. Obdobným způsobem lze analyzovat případ linek, zde je možné zjišťovat, zda se významným způsobem liší dotazy z různých linek, co se použitých parametrů týče.

4.6.2 Konstrukce testů

Test je založen na skutečnosti, že pokud se soubor dotazů pocházejících z jedné sestavy rozdělí podle statistických znaků uživatel a parametr, získáme kontingenční tabulku, v jejích jednotlivých polích najdeme četnost použití parametru i uživatelem j . Provedením testu o nezávislosti lze pak zjistit, zda je použití parametrů pro sestavu na uživateli závislé či nikoliv, a pomocí Cramerova koeficientu kontingence určit sílu případné závislosti. V případě přijetí hypotézy o závislosti použití parametrů na uživateli a potvrzení tohoto výsledku vysokou hodnotou Cramerova koeficientu kontingence lze říci, že pro danou sestavu by bylo vhodné zavést parametrizaci specifickou pro každého uživatele. Pokud by se však takováto parametrizace měla zavést, byla by zavedena pro všechny sestavy. To znamená, že návrh zavedení parametrizace specifické pro uživatele by měl být podporován splněním výše zmíněných podmínek závislosti u většiny sestav.

Postup testování je následující:

1. Je sestavena kontingenční tabulka pro danou sestavu, rozděluje dotazy podle uživatele a použitých parametrů.
2. Sestavení hypotéz. Hypotéza H_0 předpokládá nezávislost používání parametrů na uživateli, hypotéza H_1 opak.
3. Stanovení kritického oboru. Nadále budu vždy používat standardní hladinu významnosti α ve výši 5 %. To znamená, že kritickou hodnotou bude 95% kvantil rozdělení χ^2 s $(r-1)(s-1)$ stupni volnosti, kritický obor bude pak takovýto:

$$W = < \chi^2_{0,95}(r-1)(s-1); \chi^2_{\max}(r-1)(s-1))$$

4. Vypočítání hodnoty testového kritéria podle vztahu (4.2)
5. Formulace závěru testu. Pokud G patří do množiny W , zamítáme hypotézu o nezávislosti používání parametrů na uživateli. Pokud G patří do množiny V , hypotézu nezamítáme.
6. V případě zjištění závislosti testem χ^2 stanovení hodnoty Cramerova koeficientu kontingence. V případě, že Cramerův koeficient přesáhne hodnotu 0,5, lze považovat závislost za silnou. Takováto závislost je pak argumentem k zavedení parametrizace specifické pro uživatele.

4.6.3 Způsob provedení testů

Ke kompletní analýze parametrizace je třeba provést χ^2 test o nezávislosti pro každou sestavu, a to ve dvou podobách: První test otestuje závislost užívání parametrů na uživateli, druhý pak na lince. Vzhledem k poměrně vysokému množství sestav (68), uživatelů (460) i parametrů (až cca 80 na sestavu) nepřichází v úvahu manuální sběr dat spolu s jejich zadáním do statistické aplikace, jakou je např. Statgraphics. Je zřejmé, že celé provedení testů musí být plně automatické – od sběru dat, vytvoření kontingenční tabulky a jejího naplnění daty, výpočtu testového kritéria a kritického oboru až po formulaci závěru testů a spočítání koeficientů kontingence.

Pro naprogramování všech těchto operací jsem si zvolil webový programovací jazyk PHP, který je, jak už jsem zmínil, součástí mnou použitého balíku VertrigoServ spolu s MySQL a Apache. Co se týče sběru dat, skripty PHP mohou velice jednoduše pracovat s databází pomocí dotazů v jazyce SQL – není potom problém vybrat z tabulek potřebná data, stejně tak je bezproblémové vytváření pomocných tabulek či pohledů a jejich zaplnění daty. Vytvoření kontingenční tabulky a spočítání testového kritéria či koeficientů kontingence v PHP také nepředstavuje velké obtíže. Jazyk PHP je opravdu mocným jazykem a operace s alfanumerickými a numerickými proměnnými či poli samozřejmě zvládá, o výpis údajů se pak starají pasáže v jazyce HTML, které mohou také být součástí PHP skriptů. Jistou překážku ovšem představuje sestavení kritického oboru pro test. Jazyk PHP ve své základní formě totiž neumí počítat hodnoty kvantilů statistických rozdělení. Uložení

hodnot kvantilů pro fixní hladinu významnosti 5 % a různé stupně volnosti do databáze (skripty by potom pro zjištění hodnot kvantilu jednoduše zjistily z tabulky) také není zrovna ideálním řešením. V případě testů u některých sestav jako je např. KDNZ nebo NZ jsou totiž kontingenční tabulky opravdu obrovské, což s sebou nese také velmi vysoké hodnoty stupňů volnosti – konkrétně až 7000 stupňů volnosti pro sestavu NZ a test závislosti parametrů na uživateli. Ruční zadávání hodnot 95% kvantilů rozdělení χ^2 do databáze až do hodnoty 7000 zcela jistě není nejlepší řešení problému.

K jazyku PHP existují i knihovny statistických funkcí, ale jak jsem zjistil, často bohužel není možné sehnat verze běžící na operačních systémech MS Windows. Sestrojit model rozdělení χ^2 pro PHP vlastními silami je pak dosti náročná věc, naštěstí se mi podařilo objevit článek *Take Web data analysis to the next level with PHP* od Paula Meaghera [9] na serveru www.ibm.com/developerworks/, který se týká právě testování dat χ^2 testem nezávislosti pomocí jazyka PHP. Pan Meagher zde nabízí i zdrojové kódy obsahující funkci pro výpočet hodnot kvantilů χ^2 v PHP produkující velice přesné hodnoty daných kvantilů.

4.6.4 Testy nezávislosti používání parametrů sestav na uživateli

Testy nezávislosti používání parametrů sestav na uživateli by měly ukázat, zda je výhodné zavést parametrizace sestav specifické pro uživatele. Pokud výsledek ukáže závislost a vysoký Cramerův koeficient kontingence u většiny sestav, bude závěrem testu doporučení parametrizace sestav specifické pro uživatele zavést. V případě, že z výsledku nebude zřejmá těsná závislost používání parametrů sestav na uživateli, bude nutné zkoumat závislost používání parametrů sestav na lince.

Pro test nezávislosti používání parametrů sestav na uživateli jsem vytvořil dva PHP skripty. První skript s názvem *chi_uziv_sestavy.php* (zdrojový kód viz příloha č.1) vytvoří

a vypíše pro zadanou sestavu kontingenční tabulku, u které pak provede test nezávislosti. Druhý skript přehledně uspořádává výsledky testů každé sestavy do tabulky.

Údaje níže představují výstup prvního skriptu pro sestavu ZZNV, která má pouze pět parametrů. Sloupce představují parametry, řádky uživatele. Každé pole obsahuje absolutní empirickou četnost použitých parametrů s názvem *i* uživatelem *j*. Na konci každého řádku *i* sloupce jsou zobrazeny marginální četnosti – na koncích řádků součet četností použitých parametrů za uživatele, na konci sloupců součet četností použití daného parametru.

Analýza závislosti užívání parametrů na uživateli - sestava ZZNV

Rozsah dat od 2007-07-02 do 2007-12-10

Uživatel\Parametr	DATUM1	DATUM2	ZOB_NZ	ZOB_SMENY	ZOB_SV	Celkem
KVM007	1	1	0	1	1	4
LAYPHV	1	1	0	1	1	4
SQS007	1	1	0	1	1	4
U00001	2	2	0	2	2	8
U00117	14	14	0	8	8	44
U00250	54	54	0	54	54	216
U00273	2	2	0	2	2	8
U00335	1	1	0	1	1	4
U00375	2	2	0	2	2	8
U00376	3	3	0	3	3	12
U00380	1	1	0	1	1	4
U00648	1	1	0	1	1	4
U00649	1	1	0	1	1	4
U00665	1	1	0	1	1	4
U00809	16	16	0	14	14	60
U00814	1	1	0	1	1	4
U00816	3	3	0	2	2	10
U00995	1	1	0	1	1	4
U01043	1	1	0	1	1	4
U01058	2	2	0	2	2	8
U01204	1	1	0	1	1	4
U01231	1	1	0	1	1	4
U01344	1	1	0	1	1	4
U01448	4	4	0	4	4	16
U01873	2	2	0	2	2	8
U01904	2	2	0	2	2	8
U01927	1	1	0	1	1	4

xxx136	143	143	0	0	0	286
Celkem	264	264	0	112	112	752

tabulka 4-6: Tabulka zobrazující vztah používání parametrů sestavy ZZNV uživateli

Výsledky chi-kvadrát testu o nezávislosti v kontingenční tabulce:

Testové kritérium $G = 199.68$

Hranice kritického oboru $X_{0.95}(108)=133.26$

Hypotéza o závislosti používání parametrů na uživateli je přijata.

Cramerův koeficient kontingence $C_c = 0.26$

Pearsonův koeficient kontingence $C_p = 0.46$

V tomto případě je zřejmé, že ačkoliv je na hladině významnosti $\alpha = 0,05$ přijata hypotéza o závislosti používání parametrů na uživateli, Cramerův koeficient kontingence dosahuje pouze hodnoty 0,26 – což znamená, že síla závislosti není vysoká. Dokonce ani Pearsonův koeficient kontingence, který skript také počítá, není vyšší než 0,5. Takovýto výsledek myšlenku zavést parametrizace specifické pro každého uživatele nepodporuje.

Na kontingenční tabulce je zajímavá ještě jedna skutečnost; totiž že parametr ZOB_NZ není využit žádným uživatelem. Skutečnost, že některý parametr žádný uživatel během pěti měsíců nevyužil napovídá, že tento parametr je velmi pravděpodobně zbytečný. Zbytečnými parametry se v této práci zabývám v kapitole 4.7.

K vytvoření závěru ohledně nasazení parametrizací specifických pro uživatele jsou ovšem potřeba výsledky testů pro každou sestavu. K tomuto účelu slouží druhý skript se názvem *chi_uziv_vse.php*. Tento skript přehledně uspořádává výsledky testů do tabulky, kterou vidíme níže:

Sestava	testové kritérium G	kritická hodnota	H_1 přijata	Cramerův koef.
AUDIT	92.73	160.91	ANO	0.15
AUDITW	0	146.57	NE	0
CETZAV	0	35.17	NE	0
CHKKB8	26.71	33.92	NE	0.26
KDNZ	50767.18	10295.34	ANO	0.06
KDNZ_LAK	302.66	940.77	NE	0.11
KDNZ_PDI	16.47	1007.17	NE	0.03

KDPRNR	0	0	NE	CHYBA
KDUPS	0	0	NE	CHYBA
KMENDATA	14472.25	4591.95	ANO	0.12
LAK_KB8	3.73	73.31	NE	0.01
LAK_MIST	0	0	NE	CHYBA
LAK_MIST_SM	0	0	NE	CHYBA
MON_KB8	77.89	179.58	NE	0.02
MON_KB8SV	181.42	160.91	ANO	0.02
MON_KB8SZ	26.77	133.26	NE	0.05
MON_VYR	210.12	156.51	ANO	0.27
MON_ZAV	25.87	85.96	NE	0.32
MON_ZAV_SV	2.3	15.51	NE	0.28
MON_ZAV_SZ	12.5	25	NE	0.24
NEPRIMEV	27949.2	1898.69	ANO	0.22
NEPRIMEV_SV	37.72	251.29	NE	0.19
NEPRIMEV_SVNEPR	1083.99	328.58	ANO	0.19
NEPRIMEV_SZ	1957.92	164.22	ANO	0.35
NZ	592763.46	23393.15	ANO	0.05
NZ_LAK	1823.03	2779.97	NE	0.07
NZ_PDI	13068.26	4407.76	ANO	0.05
NZ_SV	100604.18	10817.24	ANO	0.09
NZ_SV_LAK	780.3	904.38	NE	0.12
NZ_SV_PDI	2311.34	1610.69	ANO	0.07
NZ_SV_VC	7781.22	1329.14	ANO	0.1
NZ_SV_VRCH	4520.22	1772.26	ANO	0.07
NZ_VRCH	21730.24	4961.16	ANO	0.05
ORK	0	36.42	NE	0
ORK_SV	0.75	11.07	NE	0.25
ORK_TYDENNI	0	18.31	NE	0
PRAC	16.4	124.34	NE	0.1
PRAC_LAK	380.46	124.34	ANO	0.1
PRAC_SV	0	0	NE	CHYBA
SLEDBLOK	0	42.56	NE	CHYBA
STAT_CETNOST	39.07	47.4	NE	0.13
STAT_GRAF	6.15	14.07	NE	0.45
STAT_OPIS	7.93	46.19	NE	0.13
SV	52644.59	4075.83	ANO	0.14
SV_LAK	15690.42	1912.05	ANO	0.11
SV_PDI	289.36	229.66	ANO	0.17
SV_POCTY	2866.82	238.32	ANO	0.18
SV_RUS	0	0	NE	CHYBA

SV_SL	1622.18	1249.57	ANO	0.1
SV_SL_LAK	159.52	443.4	NE	0.16
SV_SL_PDI	12.85	79.08	NE	0.22
TREND	14367.66	2493.48	ANO	0.16
TREND_LAK	1073.8	500.46	ANO	0.27
TREND_PDI	1.89	85.96	NE	0.17
UPS_NZ	3.37	26.3	NE	0.1
UVOLNENEV	470.79	733.42	NE	0.27
VINICI_PDI	5.98	146.57	NE	0.03
VOZYPP	6.3	53.38	NE	0.42
VRACENEV	146.42	168.61	NE	0.1
WTM_CETNOST	0	0	NE	CHYBA
WTM_KMENDATA	0	15.51	NE	0
WTM_SEZ_TESTU	69.44	133.26	NE	0.13
WTM_STATISTIKA	0	0	NE	CHYBA
ZAV_ZEME	0.93	53.38	NE	0.19
ZMET	11905.89	532.08	ANO	0.19
ZMETKOLV	0	18.31	NE	0
ZZNV	199.68	133.26	ANO	0.26

tabulka 4-7: Výsledky testů nezávislosti používání parametrů na uživateli

Z těchto výsledků vidíme, že z celkových 68 sestav lze pouze u 25 z nich přijmout hypotézu o závislosti používání parametrů na uživateli. Z těchto 25 sestav ovšem u žádné nepřesahuje Cramerův koeficient kontingence hodnotu 0,5, kterou jsem stanovil jako hraniční pro důkaz významné závislosti, a proto je možné říci, že používání parametrů sestav není na uživateli významně závislé. Co se týče nápisu CHYBA v některých polích pro Cramerův koeficient, jedná se o sestavy, kde není možné Cramerův koeficient vypočítat, neboť jmenovatel zlomku pod odmocninou ve vztahu (4.3) je roven nule. Buďto to může být z důvodu, že hodnota testového kritéria G je rovna nule, nebo je kontingenční tabulka velmi malá (např. sestavu používá pouze jeden uživatel) a hodnota výrazu $(r-1)(s-1)$ je pak rovna nule.

Závěr z testů nezávislosti používání parametrů sestav na uživateli tedy zní, že neexistuje významná závislost používání parametrů sestav na uživateli. Z toho plyne, že není vhodné zavádět parametrizace specifické pro každého uživatele, neboť činnost uživatelů, co se týče parametrů, se od sebe příliš neliší.

4.6.5 Testy nezávislosti používání parametrů sestav na lince

Testy nezávislosti používání parametrů sestav na lince by měly ukázat, zda je výhodné změnit parametrizace sestav pro jednotlivé linky. U některých sestav mohou být jisté parametry používány např. svařovnou, nikoliv však lakovnou či montáží. V samotném uživatelském rozhraní SQS Global II uživatel nejprve vybere linku a až poté sestavu, která je pro linku relevantní (např. Mladá Boleslav/Montáž A05/Největší závadovost KDNR), sestavy jsou tedy pro linky zdánlivě diferencovány. Nabídka parametrů je ovšem pro všechny linky stejná, a pro výpočet je vlastně vždy použita jedna sestava, ať je linka kterákoliv. V databázi je možné sestavy pro různé linky identifikovat pomocí atributu *linka_fyz* v tabulce *l_report*.

Pokud výsledek ukáže závislost a vysoký Cramerův koeficient kontingence u většiny sestav, bude závěrem testu návrh na změny parametrizací sestav pro jednotlivé linky.

Pro zjištění závislosti používání parametrů sestav na lince jsem naprogramoval opět dva skripty, *chi_linky_sestavy.php* a *chi_linky_vse.php*. První z nich vytvoří a vypíše pro zadanou sestavu kontingenční tabulku, u které pak provede test nezávislosti. Druhý skript opět uspořádává výsledky testů každé sestavy do tabulky.

Údaje níže představují výstup skriptu *chi_linky_sestavy.php* pro sestavu MON_VYR se čtyřmi parametry. Sloupce představují parametry, řádky jednotlivé linky označené čísly. Každé pole obsahuje absolutní empirickou četnost použitých parametrů *i* linkou *j*. Na konci každého řádku *i* sloupce jsou zobrazeny marginální četnosti – na koncích řádků součet četností použitých parametrů za linku, na konci sloupců celkový součet četností použití daného parametru.

Analýza závislosti užívání parametrů na lince - sestava MON_VYR

Rozsah dat od 2007-07-02 do 2007-12-10

Linka\Parametr	DATUM1	DATUM2	FORMAT_REPORT	KKOLEKTIV	Celkem
1	1	1	0	0	2
2	3	3	0	0	6

4	2	2	0	0	4
7	48	48	0	0	96
8	4	4	0	3	11
10	10	10	0	0	20
13	4	4	0	0	8
18	9	9	0	0	18
19	12	12	0	0	24
20	356	356	0	8	720
21	6	6	0	0	12
23	7	7	0	0	14
Celkem	462	462	0	11	935

tabulka 4-8: Zobrazení závislosti užívání parametrů na lince u sestavy MON_VYR

Výsledky chi-kvadrát testu o nezávislosti v kontingenční tabulce:

Testové kritérium $G = 66.89$

Hranice kritického oboru $X_{0.95}(33)=47.4$

Hypotéza o závislosti používání parametrů na lince je přijata.

Cramerův koeficient kontingence $C_c = 0.15$

Pearsonův koeficient kontingence $C_p = 0.26$

V případě sestavy MON_VYR je vidět, že ačkoliv je na hladině významnosti $\alpha = 0,05$ přijata hypotéza o závislosti používání parametrů na lince, Cramerův koeficient kontingence dosahuje pouze hodnoty 0,15 – což znamená, že síla závislosti je poměrně nízká. Pearsonův koeficient kontingence pak dosahuje 0,26, což také indikuje nízkou sílu závislosti. Takovýto výsledek znamená, že pro sestavu MON_VYR není třeba diferencovat parametrizace pro různé linky.

Na kontingenční tabulce je opět vidět sloupec vyplněný nulovými hodnotami; parametr FORMAT_REPORT není využit na žádné lince. Skutečnost, že některý parametr nikdo během pěti měsíců nevyužil napovídá, že tento parametr je velmi pravděpodobně zbytečný.

Na rozdíl od analýzy závislosti používání parametrů na uživateli, kde bylo třeba učinit pouze jeden závěr pro všechny sestavy, u závislosti používání parametrů na lince můžeme činit závěry pro každou sestavu zvlášť. V tomto případě totiž mohou být změny v SQS provedeny pro každou sestavu jednotlivě. K výpisu výsledků testů pro všechny sestavy

slouží skript *chi_linky_vse.php*. Tento skript přehledně uspořádává výsledky testů do následující tabulky:

Sestava	testové kritérium G	kritická hodnota	H ₁ přijata	Cramerův koef.
AUDIT	0	0	NE	CHYBA
AUDITW	0	0	NE	CHYBA
CETZAV	0	62.83	NE	0
CHKKB8	0	0	NE	CHYBA
KDNZ	17517.95	1249.57	ANO	0.08
KDNZ_LAK	64.49	87.11	NE	0.18
KDNZ_PDI	0	0	NE	CHYBA
KDPRNR	0	0	NE	CHYBA
KDUPS	0	0	NE	CHYBA
KMENDATA	4378.02	338.19	ANO	0.07
LAK_KB8	1.7	18.31	NE	0.01
LAK_MIST	0	0	NE	CHYBA
LAK_MIST_SM	0	0	NE	CHYBA
MON_KB8	16.85	37.65	NE	0.01
MON_KB8SV	11.11	49.8	NE	0.01
MON_KB8SZ	2.25	40.11	NE	0.02
MON_VYR	66.89	47.4	ANO	0.15
MON_ZAV	2.51	32.67	NE	0.1
MON_ZAV_SV	2.3	15.51	NE	0.28
MON_ZAV_SZ	1.86	18.31	NE	0.11
NEPRIMEV	7619.88	255.6	ANO	0.15
NEPRIMEV_SV	20.26	133.26	NE	0.2
NEPRIMEV_SVNEPR	317.72	231.83	ANO	0.13
NEPRIMEV_SZ	515.13	107.52	ANO	0.23
NZ	75111.52	2104.03	ANO	0.03
NZ_LAK	85.48	90.53	NE	0.1
NZ_PDI	7.46	264.22	NE	0.01
NZ_SV	39355.7	1195.78	ANO	0.07
NZ_SV_LAK	21.55	53.38	NE	0.09
NZ_SV_PDI	642.37	55.76	ANO	0.22
NZ_SV_VC	1653.7	296.47	ANO	0.1
NZ_SV_VRCH	1537.86	156.51	ANO	0.15
NZ_VRCH	586.66	190.52	ANO	0.04
ORK	0	21.03	NE	0
ORK_SV	0.75	11.07	NE	0.25
ORK_TYDENNI	0	18.31	NE	0
PRAC	13.82	124.34	NE	0.09

PRAC_LAK	4.3	37.65	NE	0.02
PRAC_SV	0	28.87	NE	0
SLEDBLOK	0	42.56	NE	CHYBA
STAT_CETNOST	0	0	NE	CHYBA
STAT_GRAF	0	0	NE	CHYBA
STAT_OPIS	0	0	NE	CHYBA
SV	26461.17	783.53	ANO	0.11
SV_LAK	1100.11	95.08	ANO	0.12
SV_PDI	15.98	74.47	NE	0.08
SV_POCTY	12.92	88.25	NE	0.02
SV_RUS	0	1000	NE	CHYBA
SV_SL	740.8	563.64	ANO	0.09
SV_SL_LAK	30.94	51	NE	0.23
SV_SL_PDI	0	0	NE	CHYBA
TREND	10013.9	689.5	ANO	0.18
TREND_LAK	53.46	43.77	ANO	0.23
TREND_PDI	0	0	NE	CHYBA
UPS_NZ	1.22	21.03	NE	0.07
UVOLNENEV	188.98	190.52	NE	0.17
VINICI_PDI	0	0	NE	CHYBA
VOZYPP	4.11	30.14	NE	0.48
VRACENEV	0	0	NE	CHYBA
WTM_CETNOST	0	0	NE	CHYBA
WTM_KMENDATA	0	0	NE	CHYBA
WTM_SEZ_TESTU	0	0	NE	CHYBA
WTM_STATISTIKA	0	0	NE	CHYBA
ZAV_ZEME	0	0	NE	CHYBA
ZMET	4275.7	31.41	ANO	0.52
ZMETKOLV	0	0	NE	CHYBA
ZZNV	186.82	36.42	ANO	0.25

tabulka 4-9: Výsledky testů nezávislosti používání parametrů na lince

Z těchto výsledků vidíme, že z celkových 68 sestav lze pouze u 19 z nich přijmout hypotézu o závislosti používání parametrů na lince. Z těchto 19 sestav ale pouze u jediné sestavy ZMET (Zmetkovitost) přesahuje Cramerův koeficient kontingence hodnotu 0,5, kterou jsem stanovil jako hraniční pro důkaz významné závislosti.

Nápis CHYBA v některých polích pro Cramerův koeficient znamená, že se jedná o sestavy, u kterých není možné Cramerův koeficient vypočítat, neboť jmenovatel zlomku

pod odmocninou ve vztahu (4.3) je roven nule. To může nastat buďto z důvodu, že hodnota testového kritéria G je rovna nule, nebo je kontingenční tabulka velmi malá (např. sestava je používána pouze na jedné lince) a hodnota výrazu $(r-1)(s-1)$ je pak rovna nule.

Výsledek testů nezávislosti používání parametrů sestav na lince zní, že u všech sestav krom jedné neexistuje významná závislost používání parametrů sestav na lince. U této jedné sestavy (ZMET) je významná závislost předpokladem pro změnu parametrizace v závislosti na lince. Pro konkrétní návrh případných změn je ale ještě třeba prozkoumat činnost uživatelů této sestavy z různých linek. Použití skriptu *chi_linky_sestavy.php* ovšem ukáže, že diferenciací parametrizací sestavy pro různé linky není v tomto případě na místě, neboť v tomto případě neexistuje parametr, který by používala právě jedna ze dvou linek, které využívají tuto sestavu.

4.7 Návrh úpravy parametrizací sestav SQS

Z výsledků testů nezávislosti použití parametrů na uživateli a lince vyplývá, že nemá smysl provádět úpravy ve směru diferenciací nabídky parametrizace jednotlivým uživatelům či linkám. Z kontingenčních tabulek 4-6 a 4-8 je však zřejmé, že parametrizace přesto nejsou v pořádku: Mnoho parametrů nebylo totiž za celé pětiměsíční sledované období využito vůbec, některé jsou pak využívány velmi zřídka. Změna parametrizací má tudíž smysl ve směru odstranění nepoužívaných parametrů ze sestav a také umístění nejpoužívanějších parametrů na první záložku, přičemž málo používané parametry mohou být umístěny na záložky další. Uživatelé by pak měli pocítit zjednodušení práce se systémem v tom směru, že u mnoha z jimi používaných dotazů nebude třeba prohledávat záložky k nalezení potřebného parametru.

K nalezení parametrů, které za sledované období nebyly ani jednou použity (sledované období je od 2.7.2007 do 10.12.2007 a bylo během něj provedeno cca 350 000 dotazů) jsem naprogramoval další skript *param_odstranit.php* (zdrojový kód viz příloha č.2), který ukazuje parametry s nulovou četností využití za celé sledované období. Výsledek práce

tohoto skriptu (viz příloha č. 4) bude předán správcům SQS, aby zvážili ponechání těchto parametrů v parametrizacích. Následuje ukázka výsledku skriptu pro sestavu AUDIT:

Parametr	Četnost	Číslo záložky
AU_TAB	0	1
AU_TABH	0	1
FORMAT_REPORT	0	1
KKOLEKTIV	0	1

tabulka 4-10: Parametry navržené k odstranění ze sestavy AUDIT

Jak je vidět z tabulky, u sestavy AUDIT je možné zvážit odstranění čtyř parametrů, AU_TAB, AU_TABH, FORMAT_REPORT a KKOLEKTIV. Jde o velmi významnou část celkové parametrizace, neboť sestava AUDIT má celkem pouze osm parametrů. Skript samozřejmě zjišťuje parametry vhodné k odstranění u všech sestav – z celkových 1568 parametrů bylo vybráno 654 vhodných k odstranění. Jedná se o dosti významný výsledek, na základě kterého by mohla být výrazným způsobem změněna tvář uživatelského rozhraní informačního systému SQS.

Druhou úpravou, kterou by bylo možné v parametrizacích provést je umístění nejvyužívanějších parametrů na první záložku za současného odsunutí málo používaných parametrů na záložky jiné. Pro navržení umístění parametrů do záložek jsem použil další skript *param_razeni.php*, který řadí parametry u sestav podle využívanosti. Nespecifikoval jsem přímo, které mají být umístěny na první záložku, neboť množství parametrů, které je možné zadat už na první záložce, velmi záleží na prostorovém uspořádání prvků pro zadávání hodnot parametrů. Zde je ukázka výsledku práce skriptu pro sestavu CHKKB8:

Parametr	Četnost	Číslo záložky
DATUM1	32	1
DATUM2	32	1
KB	32	1
SROVN_OK	32	1
SRDATUM2	32	3
SRDATUM1	32	3
CHECKLIST	1	1

SMENA	0	1
FORMAT_REPORT	0	1
SRS MENA	0	3

tabulka 4-11: Doporučené řazení parametrů na záložky

Tabulka 4-11 ukazuje, jakým způsobem by bylo vhodné uspořádat parametry sestavy CHKKB8. Každý řádek tabulky obsahuje název parametru, četnost jeho použití a záložku, na které se parametr v současné době nachází. V ideálním případě by však umístění parametrů na záložkách mělo respektovat jejich pořadí v tabulce. Výsledek práce skriptu bude opět předán správci systému SQS k posouzení případných úprav tímto směrem.

4.8 Zachycení závislosti spouštění sestav na čase

Jedním z cílů této práce je zachycení obecné závislosti spouštění jednotlivých sestav na čase. Správci informačního systému SQS to umožní zjistit, jakým způsobem se požadavky na SQS v průběhu pracovního dne či týdne mění. Tato informace by jim měla pomoci především s uživatelskou podporou SQS.

Pro zachycení závislosti spouštění sestav na čase jsem nezvolil absolutní počty, ale podíly pravděpodobností. Tímto způsobem se eliminuje vliv četnosti všech spouštěných sestav v jednotlivých dnech a bere se ohled pouze na strukturu spouštěných sestav. Podíl pravděpodobností zde tvoří index ukazující, zda je pravděpodobnost spuštění dané sestavy v daném časovém období podprůměrná či nadprůměrná. Čili hodnota 1,84 u sestavy AUDIT pro den pondělí znamená, že při spuštění sestavy je pravděpodobnost, že touto sestavou bude AUDIT silně nadprůměrná. Samotný vzorec pro výpočet indexu vypadá takto:

$$i = p(B|A) : p(B) \quad (4.6)$$

kde $p(B)$ je pravděpodobnost spuštění sestavy B a $p(B|A)$ je pravděpodobnost spuštění sestavy B za podmínky časového úseku A.

Stejně jako u testů χ^2 jsem k výpočtu těchto indexů vytvořil skripty v jazyce PHP, které potřebná data načítají z databáze. První skript *bayes_dny.php* (zdrojový kód viz příloha č.3) slouží k zachycení používání sestav v jednotlivých dnech v týdnu, druhý *bayes_hodiny.php* umí zachytit používání sestav v různých hodinách dne.

Zde je výstup prvního skriptu popisujícího spouštění sestav v jednotlivých dnech v týdnu, omezený na sedm nejpoužívanějších sestav:

Den v týdnu\Sestava	NZ	logon	KMENDATA	NZ_SV	SV	MON_KB8SV	MON_KB8
1	0	0	0	0	0.59	0	0
2	1.84	3.3	0	0	1.13	0.2	1.5
3	0.86	0	0	1.32	1.03	0.06	0.6
4	0.77	0	1.22	0	1.07	2.75	2
5	0.55	0.43	0	3.74	0.89	2.25	0.49
6	1.4	1.91	4.65	0.37	1.06	0.22	0.95
7	0.46	0	0	0	0.7	0	0

tabulka 4-12: Zachycení závislosti spouštění sestav na dni v týdnu

Z tabulky je na první pohled vidět, že zatímco některé sestavy jsou v průběhu týdne zastoupeny ve spouštěných sestavách ve zhruba stále stejné míře (např. SV), používání jiných sestav (KMENDATA) je zjevně závislé na dni v týdnu. Mezi nejmarkantnější extrémy, které můžeme ve výsledku skriptu nalézt patří:

Zvýšená používanost sestavy MON_ZAV_SV v neděli – podíl ve struktuře spouštěných sestav je v tento den 7,02 krát větší než je průměr.

Podíl sestavy WTM_STATISTIKA ve struktuře spouštěných sestav se v pátek zvýší oproti průměru 5,98 krát.

Podíl sestavy ORK_SV ve struktuře spouštěných sestav se ve čtvrtek zvýší oproti průměru 5,5 krát.

Podíl sestavy TREND_PDI ve struktuře spouštěných sestav se v pátek zvýší oproti průměru 4,89 krát.

Podíl sestavy MON_ZAV_SZ ve struktuře spouštěných sestav se v pátek zvýší oproti průměru 4,68 krát.

Co se týče extrémů ve snížení průměru, zde je minimem samozřejmě nula. Ta se objevuje v mnoha polích, protože mnoho sestav není určitý den vůbec využíváno.

Druhým pohledem na závislost spouštění sestav na čase je sledování struktury spouštěných sestav v průběhu dne. Tímto problémem se zabývá skript *bayes_hodiny.php*. Obdobně jako v případě dní v týdnu, i tento skript ukazuje podíly pravděpodobností ukazující, zda je pravděpodobnost spuštění dané sestavy v daném časovém období podprůměrná či nadprůměrná. Zde je ukázka výstupu skriptu pro sedm nejpoužívanějších sestav:

Hodina\Sestava	NZ	logon	KMENDATA	NZ_SV	SV	MON_KB8SV	MON_KB8
0:00 - 1:00	0	1.14	0	0	0.83	0.35	1.94
1:00 - 2:00	0	0	0	0	0.92	0	0.7
2:00 - 3:00	0.19	0	3.87	0	0.8	0	0
3:00 - 4:00	0	0	0	0	0.86	0.78	0
4:00 - 5:00	0	0	0	0	0.98	0	0
5:00 - 6:00	0.71	0	0	0	0.92	0	0.64
6:00 - 7:00	0.23	0	0	0	0.64	3.57	2.06
7:00 - 8:00	0	0	0	0.66	0.82	4.61	3.82
8:00 - 9:00	2.62	6.25	0	0.7	0.91	2.18	1.52
9:00 - 10:00	0.12	0	0	0	1.04	0.52	0.97
10:00 - 11:00	0.43	3.06	0	0	1.2	2.79	0
11:00 - 12:00	0.99	5.07	8.45	6.34	1.11	0.62	2.3
12:00 - 13:00	2.15	3.85	5.35	1.5	1.25	3.23	1.64
13:00 - 14:00	4.59	0	0	0	1.33	0	0.85
14:00 - 15:00	3.04	0	0	0	0.88	0	0
15:00 - 16:00	0	0	9.44	0	1.09	0	0
16:00 - 17:00	0.48	1.72	0	0	1.17	0	0
17:00 - 18:00	1.76	1.8	0	0	2.07	0	0
18:00 - 19:00	1.53	5.49	0	0	1.46	0	5.2
19:00 - 20:00	0	0	0	0	1.58	0	1.02
20:00 - 21:00	0	0	4.26	0	1.71	0	0
21:00 - 22:00	2.52	0	0	0	1.54	0	0
22:00 - 23:00	1.09	0	0	6.45	0.42	0	0.23
23:00 - 24:00	0	0	0	0	0.95	0	0

tabulka 4-13: Zachycení závislosti spouštění sestav na hodině

Z výstupu skriptu je opět možné vybrat nejzajímavější údaje:

Zvýšená používanost sestavy VOZY_PP mezi 14tou a 15tou hodinou – podíl ve struktuře spouštěných sestav je v tuto hodinu den 25.88 krát větší než je průměr.

Podíl sestavy MON_ZAV_SV ve struktuře spouštěných sestav sestavy se mezi 17h a 18h zvýší oproti průměru 22,4 krát.

Podíl sestavy STAT_GRAF ve struktuře spouštěných sestav sestavy se mezi 15tou a 16tou hodinou zvýší oproti průměru 16,99 krát.

Podíl sestavy SV_LAK ve struktuře spouštěných sestav sestavy se mezi 8:00 a 9:00 zvýší oproti průměru 14,88 krát.

Podíl sestavy MON_ZAV_SZ ve struktuře spouštěných sestav sestavy se mezi 14h a 15h zvýší oproti průměru 12,77 krát.

Minimální extrémy jsou opět nulové a to v mnoha polích tabulky – u většiny sestav existují hodiny, kdy není spouštěna žádná z nich.

Údaje z tabulek budou předány pracovníkům podpory informačního systému SQS.

5 Závěr

Výsledkem této práce jsou návrhy na zjednodušení parametrizací sestav SQS a informace o v čase se měnící struktuře spouštěných sestav. I přes skutečnost, že χ^2 testy nezávislosti v kontingenčních tabulkách ukázaly na nevhodnost změny parametrizací ve směru jejich diferenciací pro jednotlivé uživatele či linky, je návrh na změny parametrizací značně obsáhlý. Hlavním tématem návrhu je odstranění velkého množství nevyužívaných parametrů a také lepší seřazení parametrů z hlediska uživatelského komfortu. Myslím si, že cíle práce byly dobře splněny. Samotná realizace opatření na základě těchto návrhů je již nicméně na zvážení zodpovědných pracovníků společnosti Škoda Auto a.s.

Tato diplomová práce však poukazuje na ještě jednu důležitou skutečnost, a to vhodnost webových programovacích jazyků, jakým je PHP, které se běžně používají pro tvorbu dynamických webových stránek i pro složitější úlohy, jako například právě zpracování dat pomocí statistických či jiných metod. Skripty, které jsem pro účely této práce programoval prováděly bez jakýchkoli problémů a oproti mému očekávání i velice rychle výpočetně náročné operace s miliony záznamů z databáze. Myslím si, že kvalitní možnost práce s databázemi, rychlost zpracování a především velká flexibilita při výstupu dat předurčuje těmto jazykům slibnou budoucnost.

6 Seznam tabulek

tabulka 4-1: Výpis z tabulky L_REPORT z databáze SQS.....	31
tabulka 4-2: Výpis z tabulky L_REPORT z databáze SQS.....	32
tabulka 4-3: Výpis z tabulky REPORT_PARAM z databáze SQS	33
tabulka 4-4: Chyby I. a II. Druhu a jejich pravděpodobnosti [5]	36
tabulka 4-5: Kontingenční tabulka [5].....	39
tabulka 4-6: Tabulka zobrazující vztah používání parametrů sestavy ZZNV uživateli	48
tabulka 4-7: Výsledky testů nezávislosti používání parametrů na uživateli.....	50
tabulka 4-8: Zobrazení závislosti užívání parametrů na lince u sestavy MON_VYR.....	52
tabulka 4-9: Výsledky testů nezávislosti používání parametrů na lince.....	54
tabulka 4-10: Parametry navržené k odstranění ze sestavy AUDIT.....	56
tabulka 4-11: Doporučené řazení parametrů na záložky	57
tabulka 4-12: Zachycení závislosti spouštění sestav na dni v týdnu	58
tabulka 4-13: Zachycení závislosti spouštění sestav na hodině.....	59

7 Seznam obrázků

Obrázek 3.1: Stavba IS SQS [1]	18
Obrázek 3.2: Stanice pro zadávání dat na KB vybavená skenerem KKV [1].....	20
Obrázek 3.3 (vlevo): Stanice pro zadávání dat na KB vybavená skenerem KKV [1]	20
Obrázek 3.4 (vpravo): Stanice pro zadávání závad přes klávesnici [1].....	20
Obrázek 3.5: Úvodní stránka grafického uživatelského rozhraní SQS Global II.....	22
Obrázek 3.6: Výběr základních parametrů výstupní sestavy Největší závadovost KDNR. 23	
Obrázek 3.7: Parametry sestavy Největší závadovost KDNR, záložka Typ Vozu	24
Obrázek 3.8: Výstup ze sestavy „Největší závadovost KDNR“	25
Obrázek 4.1: Jednoduchý ER model získaných dat	30
Obrázek 4.2: Importovaná data v MySQL, prostředí PHPMyAdmin	34
Obrázek 4.3: Třívrstvá architektura webových aplikací [3]	35

8 Použité zdroje

- [1] Interní materiály Škoda Auto, a.s.
- [2] BOERNER, F., *Einsatz von Qualitätsregelkreisen in komplexen Montagelinien unter Verwendung von IT-Systemen* [diplomová práce]. Wolfsburg 2005. Fachhochschule Braunschweig/Wolfenbüttel, Fachbereich Fahrzeug-, Produktions- und Verfahrenstechnik
- [3] DELISLE, M. *phpMyAdmin – efektivní správa SQL* 1.vyd. Brno, Zoner Press 2004. ISBN 80-86815-09-9
- [4] WONNACOTT, T.H., WONNACOTT, R.J., *Introductory statistics for business and economics* 4th ed. New York, John Wiley & Sons 1990. ISBN 0-471-61517-X
- [5] HINDLS, R., HRONOVÁ, S., SEGER, J., *Statistika pro ekonomy* 6.vyd. Praha, Professional Publishing 2006. ISBN 80-86419-99-1
- [6] BARNARDO, J.M., SMITH, A.F.M. , *Bayesian Theory* 1st ed. Chichester, John Wiley & Sons 1994. ISBN 0-471-92416-4
- [7] MAŘÍK V., ŠTEPÁNKOVÁ O., LAŽANSKÝ J. a kolektiv: *Umělá inteligence* 4.díl, Academia Praha 2004. ISBN 80-200-0502-1.
- [8] WIKIPEDIA, *Bayesian network* [online] cit. 10.1.2008. Dostupné z http://en.wikipedia.org/wiki/Bayesian_network
- [9] MEAGHER, P. , *Take Web data analysis to the next level with PHP* [online] cit. 10.12.2007. Dostupné z <http://www.ibm.com/developerworks/web/library/wa-phpolla/>
- [10] KLIMEŠ, J., *Optimalizace informačního systému SQS na základě statistického zpracování výstupů* [ročníkový projekt] Liberec 2005, TU Liberec, Hospodářská fakulta
- [11] WILLIAMS, H.E., LANE D. *PHP a MySQL – Programujeme webové aplikace pomocí PHP a MySQL* 1.vyd. Praha, Computer Press 2002. ISBN 80-7226-760-4

9 Seznam příloh

Příloha č.1 – Zdrojový kód skriptu *chi_uziv_sestavy.php*

Příloha č.2 – Zdrojový kód skriptu *param_odstranit.php*

Příloha č.3 – Zdrojový kód skriptu *bayes_dny.php*

Příloha č.4 – Parametry navržené k odstranění

Veškeré zdrojové kódy, výstupy skriptů a balík VertrigoServ s celou použitou databází lze spolu se samotnou prací najít na přiloženém CD-ROM.

Příloha č. 4: Parametry navržené k odstranění

Sestava AUDIT:			PRNR2	0	1	SRKKOLEKTIV	0	6
			PRNR3	0	1	SRSMENA	0	6
			PRNR4	0	1	SRVIN_KOLEKTIV	0	6
Parametr	Četnost	Číslo záložky	PRNR_COND	0	1	SRVIN_TYM	0	6
AU_TAB	0	1	SMENA	0	1	SRVIN_USEK	0	6
AU_TABH	0	1	SMER	0	1	SRBARVA	0	7
FORMAT_REPORT	0	1	TRHZEME	0	1	SRMOTOR	0	7
KKOLEKTIV	0	1	TYPVOZU	0	1	SRPREVODOVKA	0	7
			URCENI	0	1	SRPRNR1	0	7
			VETEV	0	1	SRPRNR2	0	7
			VYBAVA	0	1	SRPRNR3	0	7
						SRPRNR4	0	7
Sestava AUDITW:						SRPRNR_COND	0	7
Parametr	Četnost	Číslo záložky				SRTRHZEME	0	7
FORMAT_REPORT	0	1	Sestava CHKKB8:			SRVYBAVA	0	7
PRIPOMINKA1	0	1	Parametr	Četnost	Číslo záložky	SRKDNR	0	8
SPOJEA1	0	1	FORMAT_REPORT	0	1	SRKDNR_GROUP	0	8
SPOJEB	0	1	SMENA	0	1	SROKRUH	0	8
SPOJECELK	0	1	SRSMENA	0	3	SRVINIK	0	8
						SRZAVAZNOST	0	8
						SRTYP_ZAVADY	0	9
Sestava CETZAV:			Sestava KDNZ:			Sestava KDNZ_LAK:		
Parametr	Četnost	Číslo záložky	Parametr	Četnost	Číslo záložky	Parametr	Četnost	Číslo záložky
BARVA	0	1	FORMAT_REPORT	0	1	FORMAT_REPORT	0	1
FORMAT_REPORT	0	1	LINKAF2	0	1	KKOLEKTIV	0	1
KARA_PROV	0	1	VIN_KOLEKTIV	0	1	KARA_PROV	0	2
KARA_RADA	0	1	PRNR1	0	2	PRNR1	0	2
KARA_TYP	0	1	PRNR2	0	2	PRNR2	0	2
KAROSERIE	0	1	PRNR3	0	2	PRNR3	0	2
MOTOR	0	1	PRNR4	0	2	PRNR4	0	2
PREVODOVKA	0	1	PRNR_COND	0	2	PRNR_COND	0	2
PRNR1	0	1						

KDNR_GROUP	0	3	PRNR_COND	0	2
MOZNE_IO	0	3	VYBAVA	0	2
SMER	0	3	KDNR_GROUP	0	3
SRKKOLEKTIV	0	6	MOZNE_IO	0	3
SRSMENA	0	6	OKRUH	0	3
SRBARVA	0	7	SRKKOLEKTIV	0	6
SRKARA_PROV	0	7	SRSMENA	0	6
SRKARA_RADA	0	7	SRBARVA	0	7
SRKARA_TYP	0	7	SRKAROSERIE	0	7
SRPRNR1	0	7	SRMOTOR	0	7
SRPRNR2	0	7	SRPREVODOVKA	0	7
SRPRNR3	0	7	SRPRNR1	0	7
SRPRNR4	0	7	SRPRNR2	0	7
SRPRNR_COND	0	7	SRPRNR3	0	7
SRTRHZEME	0	7	SRPRNR4	0	7
SRKDNR_GROUP	0	8	SRPRNR_COND	0	7
SRMOZNE_IO	0	8	SRTRHZEME	0	7
SROKRUH	0	8	SRTYPVOZU	0	7
SRSMER	0	8	SRVYBAVA	0	7
SRVINIK	0	8	SRKDNR_GROUP	0	8
SRKDNR	0	9	SRMOZNE_IO	0	8
SRTYP_ZAVADY	0	9	SROKRUH	0	8
			SRVINIK	0	8
			SRZAVAZNOST	0	8
			SRKDNR	0	9
			SRTYP_ZAVADY	0	9

Sestava KDNZ_PDI:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
KKOLEKTIV	0	1
POCET_RADKU	0	1
SMENA	0	1
KAROSERIE	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2

Sestava KDPRNR:

Parametr Četnost Číslo záložky

Sestava KDUPS:

Parametr Četnost Číslo záložky

Sestava KMENDATA:

Parametr	Četnost	Číslo záložky
ARCH	0	1
FORMAT_REPORT	0	1
POUZEPOSL	0	1
TYPIDENT	0	1
ZOB_AGOS	0	1
ZOB_POHYB_VOZU	0	1
ZOB_POPIS_VOZU	0	1
ZOB_TYMY	0	1

Sestava LAK_KB8:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1

Sestava LAK_MIST:

Parametr Četnost Číslo záložky

Sestava LAK_MIST_SM:

Parametr Četnost Číslo záložky

Sestava MON_KB8:

Parametr Četnost Číslo

	záložky
FORMAT_REPORT 0	1

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	

PRNR_COND	0	1
TRHZEME	0	1
VYBAVA	0	1

Sestava MON_KB8SV:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	

Sestava MON_ZAV_SZ:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	

Sestava NEPRIMEV_SVNEPR:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	
MOTOR	0	1
PREVODOVKA	0	1
PRNR1	0	1
PRNR2	0	1
PRNR3	0	1
PRNR4	0	1
PRNR_COND	0	1
SMENA	0	1
VINIK	0	1

Sestava MON_KB8SZ:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	
KKOLEKTIV	0	1

Sestava NEPRIMEV:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2

Sestava MON_VYR:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	

Sestava NEPRIMEV_SZ:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	
PRNR1	0	1
PRNR2	0	1
PRNR3	0	1
PRNR4	0	1
PRNR_COND	0	1
TRHZEME	0	1

Sestava MON_ZAV:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	

Sestava NEPRIMEV_SV:

Parametr	Četnost	Číslo záložky
BARVA	0	1
FORMAT_REPORT 0	1	
KAROSERIE	0	1
MOTOR	0	1
PREVODOVKA	0	1
PRNR1	0	1
PRNR2	0	1
PRNR3	0	1
PRNR4	0	1

Sestava MON_ZAV_SV:

Sestava NZ:

Parametr	Četnost	Číslo
----------	---------	-------

		záložky	PRNR_COND	0	2	MOZNE_IO	0	3
FORMAT_REPORT	0	1	KDNR_GROUP	0	3	SRKKOLEKTIV	0	6
PRNR1	0	2	SRKKOLEKTIV	0	6	SRSMENA	0	6
PRNR2	0	2	SRSMENA	0	6	SRBARVA	0	7
PRNR3	0	2	SRVETEV	0	6	SRKAROSERIE	0	7
PRNR4	0	2	SRKARA_PROV	0	7	SRMOTOR	0	7
PRNR_COND	0	2	SRKARA_TYP	0	7	SRPREVODOVKA	0	7
SRKKOLEKTIV	0	6	SRPRNR1	0	7	SRPRNR1	0	7
SRSMENA	0	6	SRPRNR2	0	7	SRPRNR2	0	7
SRVIN_KOLEKTIV	0	6	SRPRNR3	0	7	SRPRNR3	0	7
SRVIN_TYM	0	6	SRPRNR4	0	7	SRPRNR4	0	7
SRVIN_USEK	0	6	SRPRNR_COND	0	7	SRPRNR_COND	0	7
SRBARVA	0	7	SRTRHZEME	0	7	SRTRHZEME	0	7
SRMOTOR	0	7	SRKDNR	0	8	SRVYBAVA	0	7
SRPREVODOVKA	0	7	SRKDNR_GROUP	0	8	SRKDNR	0	8
SRPRNR1	0	7	SRMOZNE_IO	0	8	SRKDNR_GROUP	0	8
SRPRNR2	0	7	SROKRUH	0	8	SRMOZNE_IO	0	8
SRPRNR3	0	7	SRSMER	0	8	SRMISTO	0	9
SRPRNR4	0	7	SRVINIK	0	8	SRTYP_ZAVADY	0	9
SRPRNR_COND	0	7	SRMISTO	0	9			
SRTRHZEME	0	7	SRTYP_ZAVADY	0	9			
SRVYBAVA	0	7						
SRKDNR	0	8						
SRKDNR_GROUP	0	8						
SROKRUH	0	8						
SRZAVAZNOST	0	8						

Sestava NZ_LAK:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
SUM_TYPCHECKU	0	1
KARA_PROV	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2

Sestava NZ_PDI:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
KKOLEKTIV	0	1
LINKAF2	0	1
SMENA	0	1
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
VYBAVA	0	2

Sestava NZ_SV:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
VIN_KOLEKTIV	0	1
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2

Sestava NZ_SV_LAK:

Parametr	Četnost	Číslo záložky
----------	---------	------------------

FORMAT_REPORT	0	1
KARA_PROV	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
KDNR_GROUP	0	3

VIN_USEK	0	1
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
VYBAVA	0	2
KDNR	0	3
KDNR_GROUP	0	3
OKRUH	0	3

VIN_KOLEKTIV	0	1
VIN_TYM	0	1
KAROSERIE	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
VYBAVA	0	2
KDNR	0	3

Sestava NZ_SV_PDI:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
KKOLEKTIV	0	1
LINKAF2	0	1
SMENA	0	1
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
VYBAVA	0	2
KDNR_GROUP	0	3
MOZNE_IO	0	3

Sestava NZ_SV_VRCH:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
TYP_VINIKA	0	1
VIN_KOLEKTIV	0	1
VIN_TYM	0	1
VIN_USEK	0	1
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
VYBAVA	0	2
KDNR	0	3
KDNR_GROUP	0	3

SRKKOLEKTIV	0	6
SRSMENA	0	6
SRVIN_KOLEKTIV	0	6
SRVIN_TYM	0	6
SRVIN_USEK	0	6
SRBARVA	0	7
SRKAROSERIE	0	7
SRMOTOR	0	7
SRPREVODOVKA	0	7
SRPRNR1	0	7
SRPRNR2	0	7
SRPRNR3	0	7
SRPRNR4	0	7
SRPRNR_COND	0	7
SRTRHZEME	0	7
SRVYBAVA	0	7
SRKDNR	0	8
SRKDNR_GROUP	0	8
SRZAVAZNOST	0	8
SRMISTO	0	9
SRTYP_ZAVADY	0	9

Sestava NZ_SV_VC:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
KKOLEKTIV	0	1
VIN_KOLEKTIV	0	1
VIN_TYM	0	1

Sestava NZ_VRCH:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1

Sestava ORK:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
KB	0	1

Sestava ORK_SV:

Parametr	Četnost	Číslo záložky
MOZNE_IO	0	1

Sestava ORK_TYDENNI:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1

Sestava PRAC:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
KKOLEKTIV	0	1
SRS MENA	0	1
BARVA	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2
TYPVOZU	0	2
VYBAVA	0	2

Sestava PRAC_LAK:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
KKOLEKTIV	0	1
SMENA	0	1
SRS MENA	0	1
KARA_PROV	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2

Sestava PRAC_SV:

Parametr	Četnost	Číslo záložky
BARVA	0	1
FORMAT_REPORT	0	1
KAROSERIE	0	1
MOTOR	0	1
PREVODOVKA	0	1
PRNR1	0	1
PRNR2	0	1
PRNR3	0	1
PRNR4	0	1
PRNR_COND	0	1
SMENA	0	1
TRH_ZEME	0	1
TYP_VOZU	0	1
VYBAVA	0	1

Sestava SLEDBLOK:

Parametr	Četnost	Číslo záložky
IDENT	0	1
IDENTSB	0	1
TYPIDENT	0	1
TYPIDENTSB	0	1
BARVA	0	2
KARA_PROV	0	2
KARA_RADA	0	2
KARA_TYP	0	2
KAROSERIE	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2
TYPVOZU	0	2
VYBAVA	0	2
DATUM1	0	3
DATUM2	0	3
EMAIL	0	3
HESLO	0	3
KB	0	3
LINKA	0	3
MIMVOZY	0	3
TEXT	0	3
TEXTKBKKV	0	3
TEXTKBMAIL	0	3
TEXTKBMONITOR	0	3

Sestava STAT_CETNOST:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	

Sestava STAT_GRAF:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	
INTERVAL 0	1	
POCETOBDOSI 0	1	

Sestava STAT_OPIS:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	
VYB_SESTAVA 0	1	

Sestava SV:

Parametr	Četnost	Číslo záložky
FILTR 0	1	
FORMAT_REPORT 0	1	
UVOLP 0	1	
UZIV 0	1	
ZOB_SV 0	1	
PRNR1 0	2	
PRNR2 0	2	
PRNR3 0	2	
PRNR4 0	2	

PRNR_COND 0 2

Sestava SV_LAK:

Parametr	Četnost	Číslo záložky
FILTR 0	1	
FORMAT_REPORT 0	1	
SROT 0	1	
SROTP 0	1	
URCENI 0	1	
ZOB_SV 0	1	
PRNR1 0	2	
PRNR2 0	2	
PRNR3 0	2	
PRNR4 0	2	
PRNR_COND 0	2	

Sestava SV_PDI:

Parametr	Četnost	Číslo záložky
BEZZAVAD 0	1	
FILTR 0	1	
FORMAT_REPORT 0	1	
LINKAF1 0	1	
LINKAF2 0	1	
SMENA 0	1	
SROT 0	1	
ZOB_SV 0	1	
MOTOR 0	2	
PREVODOVKA 0	2	
PRNR1 0	2	
PRNR2 0	2	
PRNR3 0	2	
PRNR4 0	2	

PRNR_COND 0 2

VYBAVA 0 2

Sestava SV_POCTY:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT 0	1	
MOTOR 0	2	
PREVODOVKA 0	2	
PRNR1 0	2	
PRNR2 0	2	
PRNR3 0	2	
PRNR4 0	2	
PRNR_COND 0	2	
TRHZEME 0	2	
TYPVOZU 0	2	
VYBAVA 0	2	

Sestava SV_RUS:

Parametr Četnost Číslo záložky

Sestava SV_SL:

Parametr	Četnost	Číslo záložky
FILTR 0	1	
FORMAT_REPORT 0	1	
KOLOVANI 0	1	
SMENA 0	1	
SROT 0	1	
PRNR1 0	2	
PRNR2 0	2	

PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2

Sestava SV_SL_LAK:

Parametr	Četnost	Číslo záložky
CISLO_STRIKU	0	1
FILTR	0	1
FORMAT_REPORT	0	1
KKOLEKTIV	0	1
MIMVOZY	0	1
MOZNE_IO	0	1
SROT	0	1
URCENI	0	1
UVOLL	0	1
VETEV	0	1
KARA_PROV	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2

Sestava SV_SL_PDI:

Parametr	Četnost	Číslo záložky
BEZZAVAD	0	1
FILTR	0	1
FORMAT_REPORT	0	1
KOLOVANI	0	1
SMENA	0	1
SROT	0	1

KAROSERIE	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2
VYBAVA	0	2

Sestava TREND:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
INTERVAL	0	1
POCETOBDABI	0	1
VIN_KOLEKTIV	0	1
VIN_TYM	0	1
VIN_USEK	0	1
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2
VYBAVA	0	2
KDNR	0	3
KDNR_GROUP	0	3
OKRUH	0	3
IDENTZAV	0	4

Sestava TREND_LAK:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
INTERVAL	0	1
POCETOBDABI	0	1
KARA_PROV	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2
KDNR_GROUP	0	3
OKRUH	0	3
IDENTZAV	0	4

Sestava TREND_PDI:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
INTERVAL	0	1
KKOLEKTIV	0	1
POCETOBDABI	0	1
SMENA	0	1
BARVA	0	2
KAROSERIE	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2
VYBAVA	0	2

KDNR	0	3
KDNR_GROUP	0	3
MOZNE_IO	0	3
OKRUH	0	3
VINIK	0	3
ZAVAZNOST	0	3
IDENTZAV	0	4

PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
VYBAVA	0	2

PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
VYBAVA	0	2

Sestava UPS_NZ:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
UPSSTAV	0	1

Sestava UVOLNENEV:

Parametr	Četnost	Číslo záložky
DEFINICE	0	1
FORMAT_REPORT	0	1

Sestava VINICI_PDI:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
SMENA	0	1
SRS MENA	0	1
BARVA	0	2
KAROSERIE	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2

Sestava VOZYPP:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
BARVA	0	2
KAROSERIE	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2
TYPVOZU	0	2
VYBAVA	0	2

Sestava VRACENEV:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
SMENA	0	1
BARVA	0	2
KAROSERIE	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2

Sestava WTM_CETNOST:

Parametr Četnost Číslo záložky

Sestava WTM_KMENDATA:

Parametr	Četnost	Číslo záložky
DIL_CISLO	0	1
FORMAT_REPORT	0	1

Sestava WTM_SEZ_TESTU:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
SMENA	0	1
SRS MENA	0	3

Sestava WTM_STATISTIKA:

Parametr Četnost Číslo záložky

Sestava ZAV_ZEME:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
TYPCHECKU	0	1
BARVA	0	2
KAROSERIE	0	2
MOTOR	0	2
PREVODOVKA	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2
TYPVOZU	0	2
VYBAVA	0	2

Sestava ZZNV:

Parametr	Četnost	Číslo záložky
ZOB_NZ	0	1

Sestava ZMET:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1
ZOB_ZAV	0	1
KARA_PROV	0	2
PRNR1	0	2
PRNR2	0	2
PRNR3	0	2
PRNR4	0	2
PRNR_COND	0	2
TRHZEME	0	2

Sestava ZMETKOLV:

Parametr	Četnost	Číslo záložky
FORMAT_REPORT	0	1