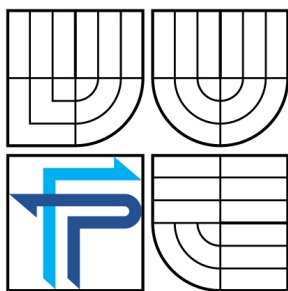


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA PODNIKATELSKÁ
ÚSTAV MANAGEMENTU

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUTE OF MANAGEMENT

SOFTWARE PRO DETEKCI, ANALÝZU A OPRAVU KOLIZNÍCH OBJEDNÁVEK V CRM SYSTÉMU

SOFTWARE FOR DETECTING, ANALYSING AND CORRECTING OF CONFLICT ORDERS IN
CRM

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. MILAN BAŽANT

VEDOUCÍ PRÁCE
SUPERVISOR

doc. Ing. MILOŠ KOCH, CSc.

BRNO 2009

Sem vložit zadání VŠKP

Abstrakt

Diplomová práce se zabývá vytvořením softwaru pro automatizaci oprav integračních chyb objednávkového systému. Na základě analýzy chyb, jejich příčin a možnostech opravy zajistit automatizaci opravy tak, aby nebyl potřeba ruční zásah a nedocházelo ke zpoždění realizace objednávek zákazníků.

Abstract

The master's thesis focuses on the creation of the automatic error cleansing software which clears the errors generated by the ordering system. Based on the error analysis (e.g. their causes and possibilities of their correction) the software enables the automatic error correction without the necessity of any manual action resulting in minimal delay of customers' orders.

Klíčová slova

Analýza, databázový link, datový sklad, dolování dat, integrace, nástroj, objednávka, primární klíč, oracle streams, procedura, tabulka, zákazník

Key words

Analysis, database link, data warehouse, data mining, integration, tool, order, primary key, oracle streams, procedure, table, customer.

Bibliografická citace :

BAŽANT, M. *Software pro detekci, analýzu a opravu kolizních objednávek v CRM systému*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2009. 97 s. Vedoucí diplomové práce doc. Ing. Miloš Koch, CSc.

P r o h l á š e n í

Prohlašuji, že diplomovou práci na téma
**„Software pro detekci, analýzu a opravu kolizních objednávek v CRM
systému“**

jsem vypracoval samostatně.

Použitou literaturu a podkladové materiály
uvádím v seznamu literatury.

V Brně dne: 30.4.2009

Milan Bažant

Poděkování

Děkuji doc. Ing. Miloši Kochovi, CSc. z Ústavu informatiky Fakulty podnikatelské Vysokého učení technického v Brně, za poskytnutí odborného vedení, potřebných rad a připomínek pro vypracování této práce. Zároveň mé poděkování patří panu Mgr. Marku Stojanovovi za přísný přístup a odborné rady při tvorbě diplomové práce a také všem ostatním lidem, kolegům, rodině, kteří mi pomohli zdárně dokončit tuto práci.

Obsah

Úvod	8
1. Seznámení se společností Telefónica O2 Czech Republic, a.s.....	9
2. Analýza současného stavu	13
2.1. Objednávkový systém CLAUDIA – integrated order managment.....	13
2.2. Objednávkový systém – technická architektura	15
2.3. Objednávkový systém – rozhraní.....	16
2.4. Objednávka	17
2.4.1. Struktura objednávky	20
2.5. Životní cyklus objednávky.....	23
2.6. Přehled stavů objednávek	25
2.7. Vývojový diagram zpracování chyb	27
2.7.1. Analýza vstupu pro ruční opravy.....	30
2.7.2. Identifikace objednávek s potenciální chybou a kalkulace zdrojů..	31
2.8. Integrační SW	33
2.8.1. Služba.....	33
2.8.2. Enterprise Service Bus (ESB).....	34
2.8.3. Složená služba (BPM) – Integrační workflow.....	35
2.8.4. Základní schéma	36
2.8.5. IBTUX	36
2.8.6. IBWLI.....	37
2.9. Provisioning – OMS	37
2.9.1. Architektura OMS.....	38
3. Teoretické podklady k řešení – dolování dat.....	40
3.1. Capture.....	47
3.2. Propagace	48
3.3. Consumption	48
4. Vlastní řešení	49
4.1. Zjištěné příčiny chyb.....	51
4.2. Vývojový diagram nového procesu	52
4.3. Analýza příčin vzniku chyb	54
4.3.1. Žlutý blok.....	54
4.3.2. Fialový blok	55
4.3.3. Modrý blok.....	56
4.3.4. Okrový blok	57
4.3.5. Červený blok.....	57
4.4. Analýza zdrojů	57
4.5. Vytvoření DB struktury	59
4.5.1. Entito – relační model.....	60
4.6. Opravný nástroj.....	61
4.6.1. Schéma toků dat.....	62
4.6.2. Tok číslo 1, automat → order sum.....	63
4.6.3. Tok číslo 2, siebel → order sum	63
4.6.4. Tok číslo 3, wli → order sum	64
4.6.5. Tok číslo 4, SPC → order sum	64

4.6.6.	Tok číslo 5, order sum → automat.....	64
4.6.7.	Tok číslo 6, automat → Siebel.....	64
4.7.	Další funkce nástroje.....	67
4.7.1.	Data mining.....	67
4.7.2.	Fixace.....	68
4.8.	Nástroj pro opravu transakcí.....	69
4.9.	Uživatelské rozhraní pro analýzu.....	70
4.10.	Přínosy řešení.....	72
4.11.	Ekonomické zhodnocení.....	73
Závěr		75
	Seznam použité literatury a všech informačních zdrojů.....	76
	A) Knihy.....	76
	B) Zákony a vládní vyhlášky.....	76
	C) Firemní materiály.....	76
	Seznam zkratk.....	77
	Seznam příloh.....	78
Přílohy		79
	Příloha č.1: Popis objektů technické architektury.....	79
	Příloha č.2: Přehled změn stavů objednávek.....	80
	Příloha č.3: Změny stavů objednávek v závislosti na jejich stavu.....	84
	Příloha č.4: Struktura tabulky ORDER_SUM.....	86
	Příloha č.5: Struktura tabulky ORDER_SUM_HISTORY.....	88
	Příloha č.6: Struktura tabulky ORDER_SUM_LOG.....	90
	Příloha č.7: Struktura tabulky ORDER_SUM_MESSAGE.....	90
	Příloha č.8: Struktura tabulky ORDER_SUM_TMP.....	90
	Příloha č.9: Opis spouštěcí části kódu.....	91
	Příloha č.10: Procedura Auto_send.....	92
	Příloha č.11: Procedura Update_tran.....	94
	Příloha č.12: Opis tabulky ORDER_SUM_MESSAGE.....	97

Úvod

Objednávkový systém společnosti O2 je součástí integrovaného systému obsluhy požadavků zákazníků. Jedná se o strategickou aplikaci s dopadem na hospodaření celé firmy. Od její bezchybné funkcionality se odvíjí jak výnos společnosti, tak spokojenost zákazníků. Základním strategickým cílem je zákaznická orientace, a proto je primárním úkolem eliminovat veškeré důvody mající vliv na spokojenost zákazníka.

Vedení společnosti každodenně sleduje vývoj objednávek zákazníků, od počtu přijatých dle produktů, po počet uzavřených. Tyto informace slouží ke strategickým rozhodnutím dalšího vývoje portfolia produktů, k určení poptávky zákazníků.

K tomu, aby měly tyto informace co nejvyšší vypovídající schopnost, je nutné zajistit 100% kvalitu zpracování objednávek. Na kvalitu má vliv řada faktorů (technické problémy, administrativní chyby, HW závady atd.), u kterých je nutné eliminovat možnost problémů. Jednou z oblastí, mající vliv na kvalitu realizace, je složitost integrace objednávkového systému do procesu realizace požadavku. Každý z integračních přechodů mezi aplikacemi nese s sebou riziko přerušení běhu realizace. Na jednotlivých rozhraních jsou nasazeny sondy průchodů, které sledují časové intervaly realizace v jednotlivých částech procesu.

Dalším faktorem jsou chyby způsobené SW. Na tuto oblast jsem se zaměřil, protože zde jsou obrovské možnosti zlepšení.

Cílem méj práce je zabezpečit monitorování, analýzu, opravu a reportování kolizních objednávek s co nejmenším zpožděním. Výsledkem by mělo být uspoření nákladů v oblasti lidských zdrojů a zvýšení spokojenosti zákazníků.

1. Seznámení se společností Telefónica O2 Czech Republic, a.s.

O společnosti Telefónica O2 Czech Republic, a.s

Telefónica O2 Czech Republic, a.s., je prvním integrovaným operátorem v České republice, který vznikl 1. července 2006 spojením nejvýznamnějšího provozovatele pevných linek, ČESKÉHO TELECOMU, a.s., a nejsilnějšího mobilního operátora, Eurotelu Praha, spol. s r.o., do jedné telekomunikační společnosti. Společnost dnes provozuje téměř osm miliónů mobilních a pevných linek, umožňuje využívání síťové infrastruktury pro provozovatele a poskytovatele veřejných i neveřejných sítí a služeb, což z ní činí jednoho z hlavních poskytovatelů plně konvergentních služeb na světě.

Telefónica O2 Czech Republic nabízí nejucelenější nabídku hlasových a datových služeb v České republice.

Telefónica O2 Czech Republic je držitelem několika ocenění kvality, např. ISO 9000:2001, ISO 14001:2004, OHSAS 18001:1999 a Recognised For Excellence.

V rámci mezinárodní skupiny Telefónica patří Telefónica O2 Czech Republic ke skupině Telefónica O2 Europe.

Prodej služeb je orientován na dva základní segmenty zákazníků: spotřebitelský segment a podnikatelský segment (včetně korporátní klientely a státní správy). Společnost poskytuje také velkoobchodní služby ostatním provozovatelům veřejných telekomunikačních sítí a poskytovatelům veřejných telekomunikačních služeb v České republice i v zahraničí.



Produktové portfólio firmy Telefónica O2 Czech Republic, a.s.

- **Datové služby**
 - o ATM
 - o Frame Relay
 - o Privátní síť
 - o VOICE VPN
 - o SLA a Dohled datových služeb
- **Pevné okruhy**
 - o Pronajaté okruhy
 - o Ethernet
- **Internetové služby**
 - o O2 Internet ADSL a IOL Broadband
 - o O2 Internet Komplet
 - o IOL DIAL-UP, FIXED, DIGITAL
 - o Webhosting, e-mail a domény
 - o Hostingové služby (Server Hosting a Dedicated Server)
 - o IOL IP VPN
 - o IOL Quick

- **Služby s přidanou hodnotou**

- o ***Barevné linky***

- 800 Zelená linka
 - 840, 841 Bílá linka
 - 844 Modrá linka
 - 976 Datarif - zprostředkování přístupu k veřejným datovým službám se zvláštním tarifem
 - 842 Linka 842
 - Audiotex – Linka 900, 906 a 909
 - TeamNet (Voice VPN)
 - Zákaznická hláška
 - Memobox
 - Karta X
 - O2 TELEKONTO
 - Jednotné číslo
 - CLIP - Identifikace čísla volajícího
 - InfoLimit
 - Audit příchozích hovorů
 - 16kHz, AOC

- **Produktová řešení**

- o Prodej PC
 - o O2 SMS v pevné síti
 - o Dohled 24 - pult centralizované ochrany (PCO)
 - o Digital Business
 - o Kompletní kancelář
 - o O2 TV
 - o Optimalizace ICT
 - o O2 PC Strážce

- **Ostatní**
 - o Dálnopis
 - o Telegraf

- **Telefonní služby**
 - o Vnitrostátní telefonní služby
 - o Mezinárodní telefonní služby
 - o Cenové tarify/ balíčky
 - o Doplnkové služby

- **Zákaznická koncová zařízení (ZKZ)**
 - o ZKZ – datová
 - o ZKZ – hlasová
 - o ZKZ - konferenční
 - o ZKZ - ADSL

- **Konvergentní služby**
 - o O2 TRIO
 - o O2 DUO MOBIL

- **Wholesale služby**

2. Analýza současného stavu

Cílem této kapitoly je provést seznámení s vlastním aplikačním prostředím, současným procesem zpracování chyb systému a vyjádření kalkulací nákladů na tuto činnost. V kapitolách 2.8. a 2.9. je informační seznámení s integračním a provisioningovým SW, který má zásadní vliv na zpracování objednávek.

2.1. Objednávkový systém CLAUDIA – integrated order management

Jak z (12) plyne, z uživatelského hlediska je řešení CLAUDIA založeno na použití zákaznického balíkového software Siebel eCommunications 7.8 – to znamená řešení, které je určené pro použití v prostředí telekomunikačního operátora. Pro podporu objednávkových funkcí se využívá v co největší možné míře standardní funkčnost, která je rozšířena podle potřeb a požadavků uživatelů a z hlediska integrace na další systémy zabezpečující realizaci a zpoplatnění zadaných objednávek.

Základní cíle integrovaného řízení objednávání je možné shrnout do třech bodů:

- zrychlení a optimalizace procesu realizace služby
- proces realizace objednávky na „jedno zastavení“ bez nutnosti opětovného kontaktování zákazníka
- kontrola nad celým životním cyklem objednávky a řešení výjimečných stavů

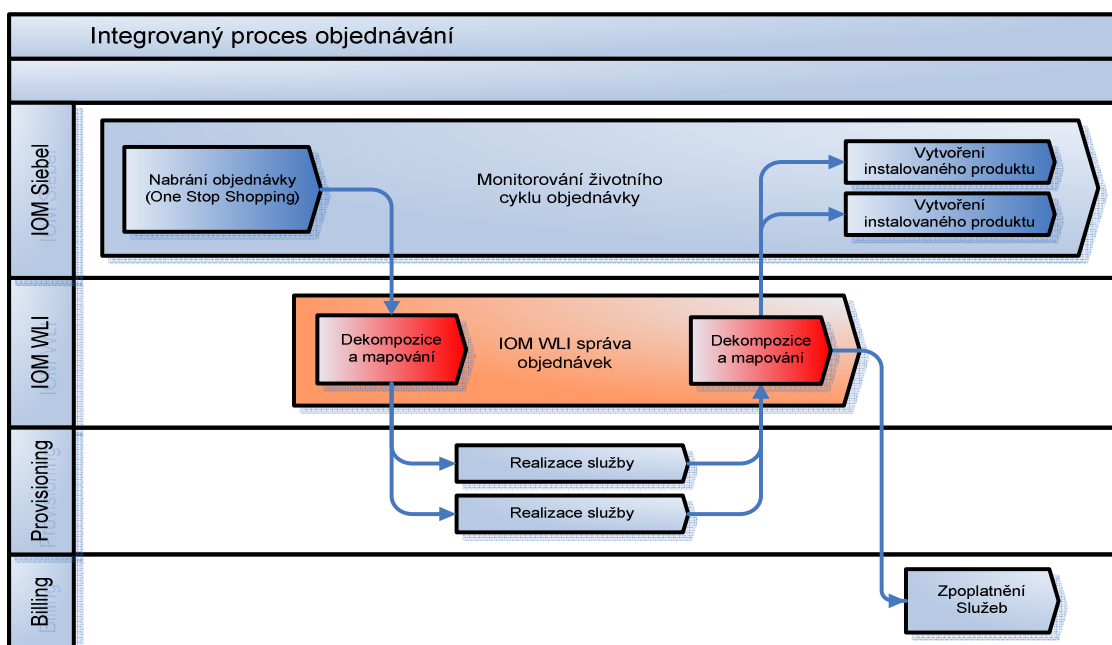
K naplnění uvedených cílů směřují následující koncepty CLAUDIA:

- systém CLAUDIA představuje z dlouhodobého hlediska jediný vstup pro zadávání objednávek, přičemž objednávky zadané do systému CLAUDIA nebude nutné opakovaně ručně zadávat do dalších systémů
- CLAUDIA jako zdrojový systém kontroluje celý životní cyklus objednávky od jejího zadání přes realizaci až po zpoplatnění. Řízení celého procesu objednávání pozůstává z monitorování životního cyklu objednávky prostřednictvím definovaných pozitivních a výjimečných stavů objednávky a z nástrojů na manipulaci s objednávkami po dobu celého jejího životního cyklu (odesílání objednávek, oprava chybových stavů, storno objednávek apod.)

- CLAUDIA podporuje objednání na „jedno zastavení“ (one stop shopping) prostřednictvím minimalizování integračních závislostí po dobu zadávání objednávky, nástrojů na technické ověření realizovatelnosti služby a dohodu o termínu návštěvy u zákazníka, přičemž tyto nástroje jsou optimalizované a umožňují vzájemné paralelní zpracování
- unifikovaný pohled na produktový katalog, který se skládá z dílčích produktových katalogů specifických podle jejich určení (sales, provisioning, billing) a pravidel pro automatické mapování těchto dílčích katalogů a dekompozici objednávek. Částečné katalogy jsou uloženy v jednotlivých systémech a mapovací pravidla jsou součástí integrace
- struktura objednávky v systému CLAUDIA umožňuje seskupování více služeb představujících jednu transakci tak, aby mohla být objednávka komunikována zřizovacím systémům jako celek, čímž se dosáhne optimalizace zřizovacího procesu

CLAUDIA unifikuje nástroje pro zadávání, monitorování a zpracování objednávek napříč všemi segmenty.

Na obr.č.1 jsou schématicky uvedeny jednotlivé kroky životního cyklu objednávky a v které oblasti CLAUDIA se odehrávají.



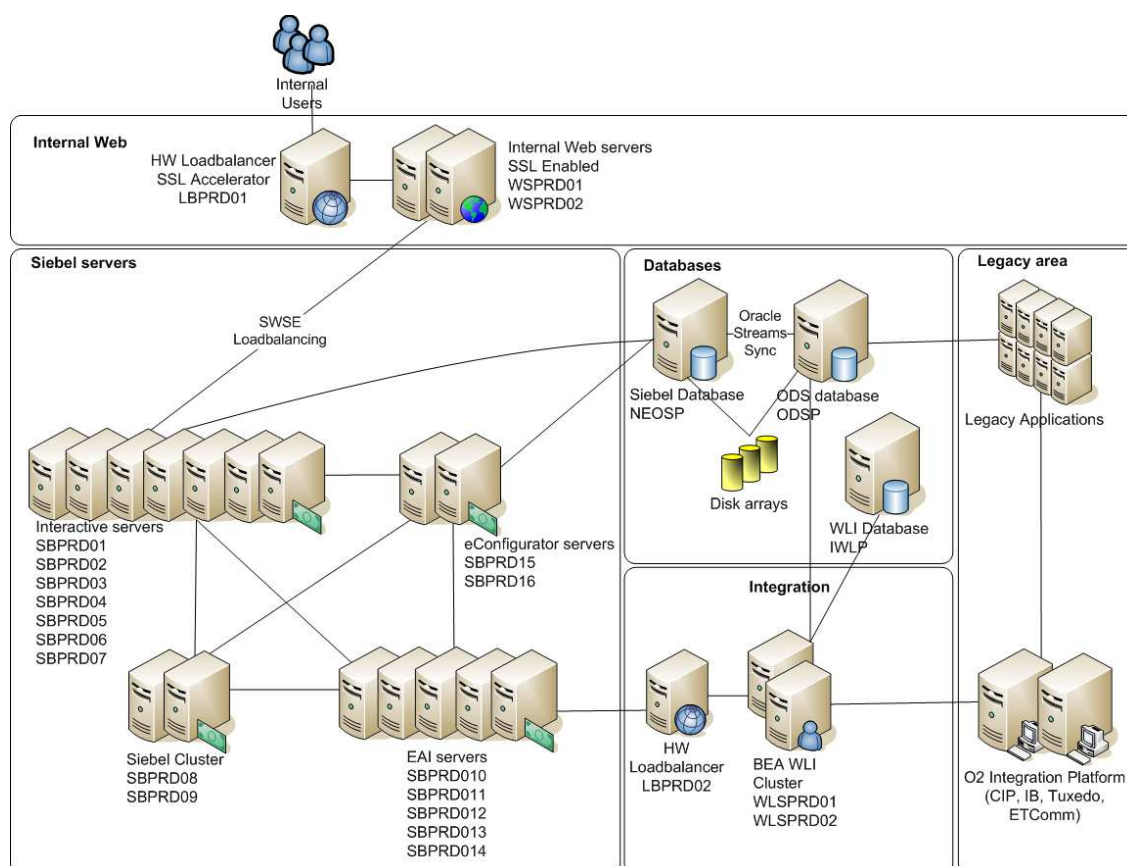
Obr.č. 1: Integrovaný proces objednávání

2.2. Objednávkový systém – technická architektura

Obrázek č.2 zobrazuje tři základní komponenty architektury systému CLAUDIA

- CLAUDIA Siebel (Webová farma, aplikační servery, report server a databázi)
- CLAUDIA WLI middleware
- ODSP databázi

Komponenty prostředí jsou na sobě kvazi nezávislé, pokud jedna z nich přestane fungovat, ostatní mohou pokračovat omezeně funkční. Pokud například nastane chyba WLI, je možné pokračovat v práci se CLAUDIA Siebel, ale specifické funkčnosti spojené s request-respons mechanismy a synchronizací přes WLI nebudou funkční. Komponenty systému jsou navrženy s ohledem na vysokou dostupnost systému a použité mechanismy omezují vliv výpadků jednotlivých HW komponent na celkovou funkci systému, pro SPOF (Single point of failure) komponenty je používáno loadbalancování případně clustering. Podrobná legenda k obrázku č.2 je přílohou č.1.



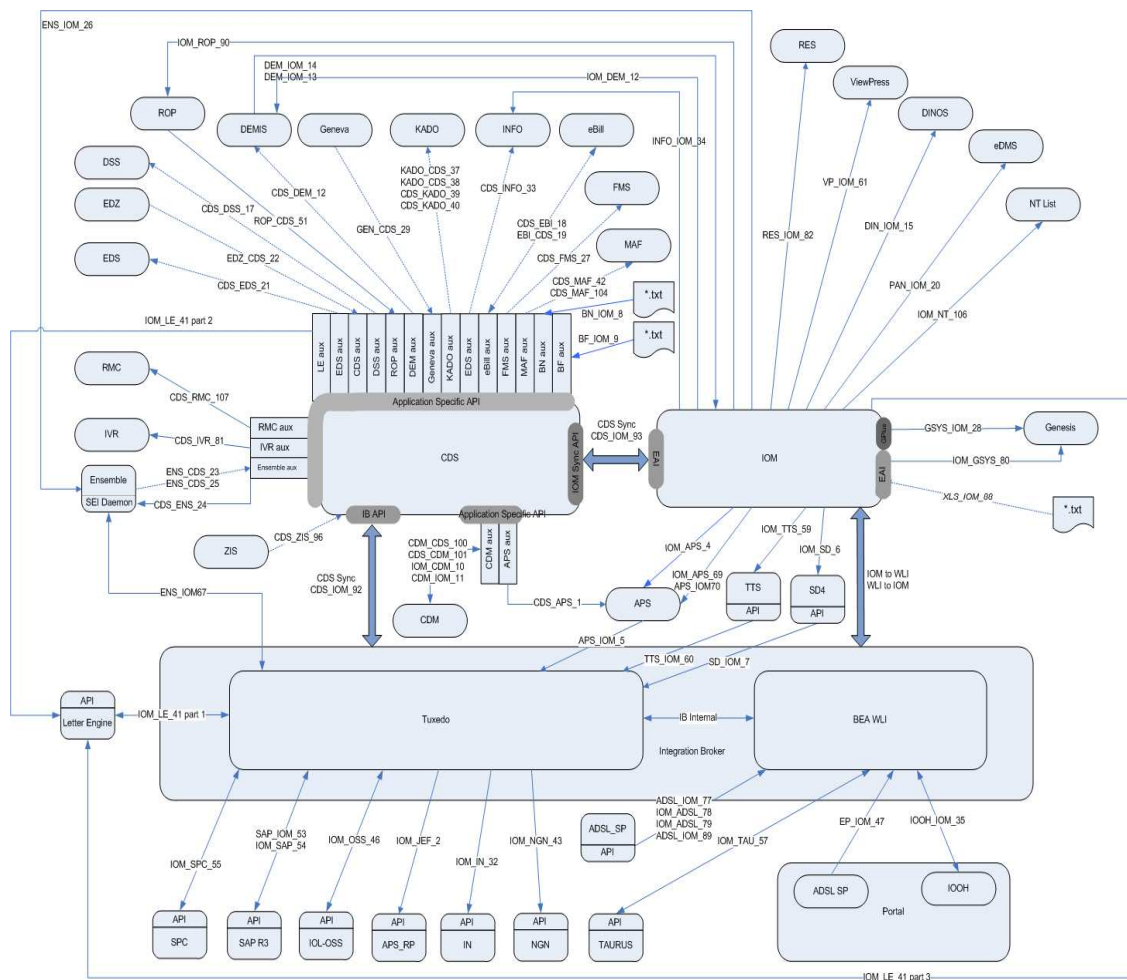
Obr. č.2: Technická architektura CLAUDIA

2.3. Objednávkový systém – rozhraní

Systém CLAUDIA Siebel je integrován s dalšími systémy O2. Použity jsou různé typy a technologie rozhraní :

- Online WLI interfacy (http volání a WLI procesy)
 - o inbound
 - o outbound
- EAI interfacy
 - o EIM
 - o COM Data Server (excel loadery)
 - o Embedded drivery (CTI, FileNet)
 - o Http volání
- ODSP dávkové interfacy (SQL, PL/SQL, JAVA vložené procedury)
- Http launch interfacy

Na obrázku č.3 je zobrazeno schéma aplikační integrace. V současné době je využíváno 93 interface. Rozdělení a popis funkcionality těchto rozhraní je nad rámec této práce.



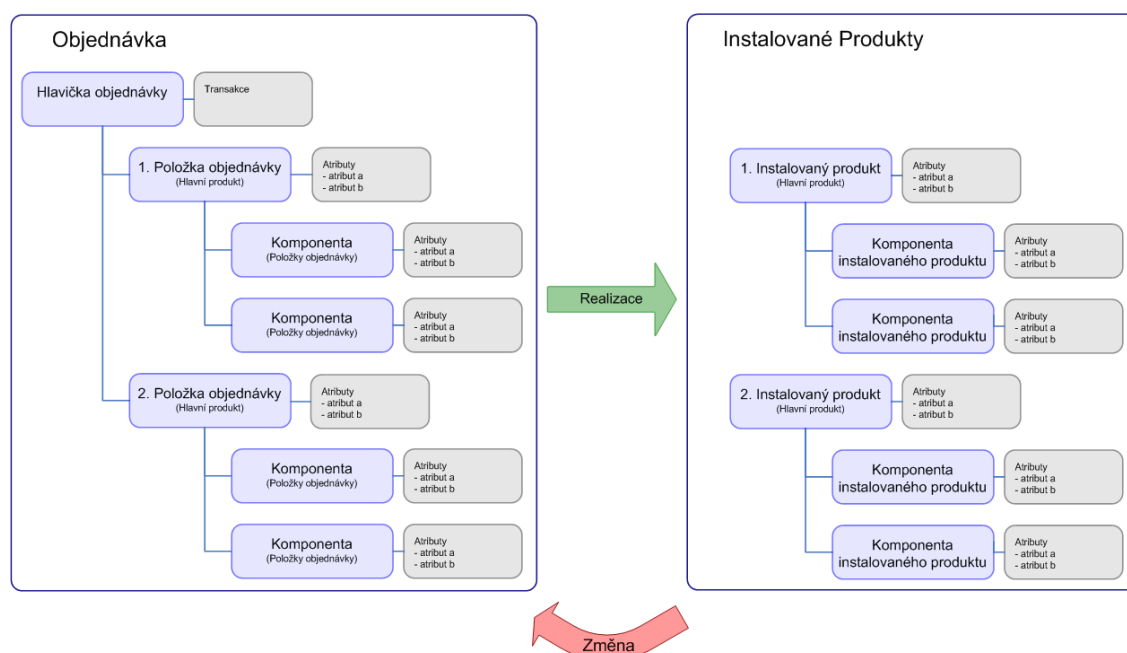
2.4. Objednávka

Objednávka viz. (15) reprezentuje požadavek zákazníka na změnu využívaných produktů (zřízení, změna, zrušení, atd.). Objednávku tvoří hlavička, položky objednávky a jejich atributy. Objednávka má svůj životní cyklus, za běžných okolností od stavu Založeno po stav Dokončeno.

Objednávka má omezenou životnost do doby realizace a vytvoření instalovaných produktů.

Instalované produkty odrážejí informáci o aktuálném stavu produktů, které zákazník používá. Je to výsledek dokončené objednávky – z 1 položky objednávky vznikne 1 instalovaný produkt.

Následující obrázek č.4 popisuje vztah mezi objednávkou a instalovaným produktem.



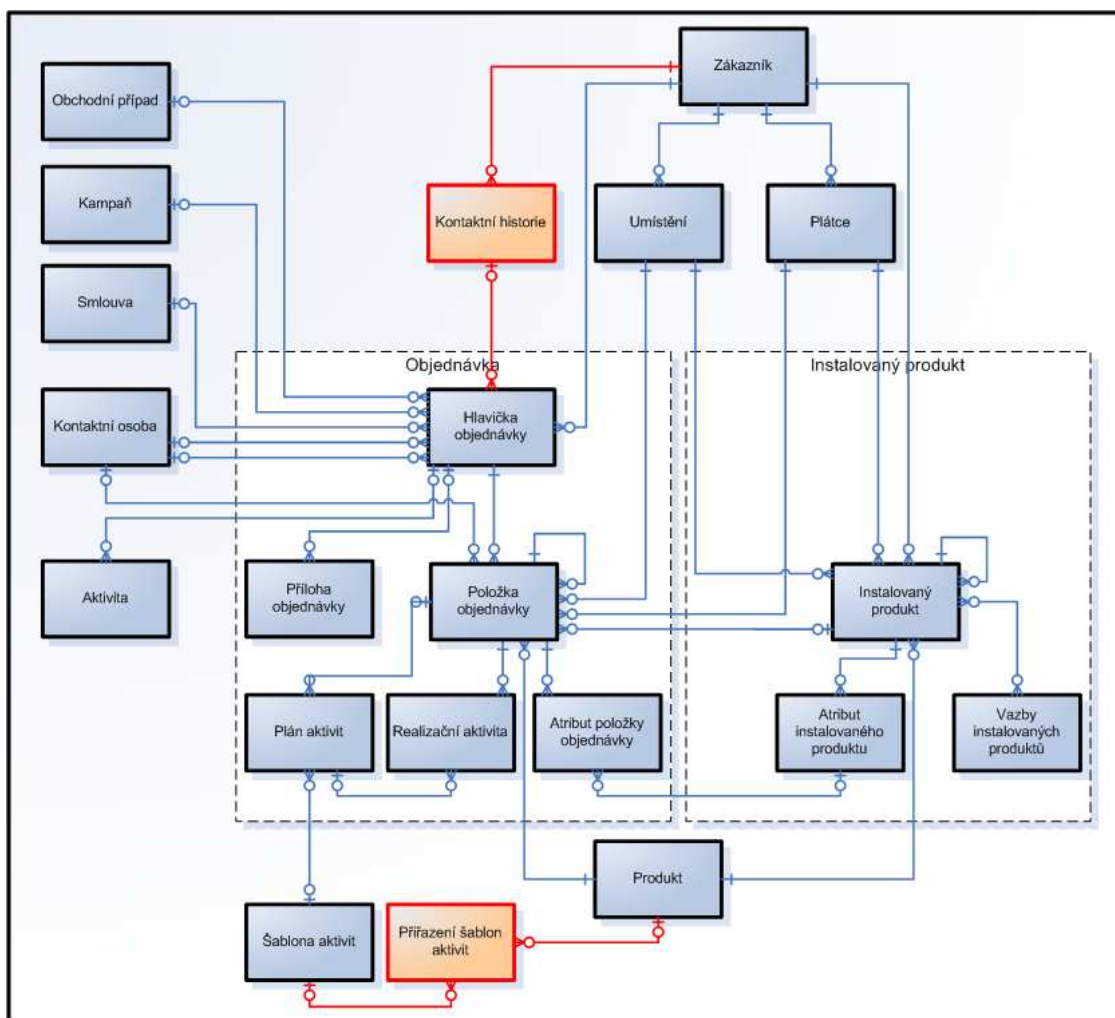
Obr. č.4: Vztah mezi objednávkou a produktem

Objednávku tvoří hlavička objednávky, položky objednávky, komponenty položek objednávky a jejich atributy. Na úrovni hlavičky objednávky je definována transakce (např. zřízení, změna, deaktivace), která platí pro všechny položky objednávky.

Jedna položka reprezentuje sadu produktů a služeb, prodáváných jako jeden celek - tvoří ji hierarchická struktura komponentů podle definice z produktového katalogu.

Například pro HTS je ve formě komponentů zachycena jak struktura cenových plánů, doplňkových služeb, údaje o tel. čísle, tak i další související produkty jako Internet ADSL, Telekonto atd. Pokud je v objednávce několik položek, hovoříme o tzv. komplexní objednávce.

Obr.č.5 ilustruje strukturu objednávky a její návaznosti na ostatní entity používané v procesu objednávání. Aplikace CLAUDIA Siebel obsahuje cca 3300 tabulek, a proto je schéma uvedeno po entitách.



Obr. č. 5: Seznam Siebel CLAUDIA entit a jejich vztahů

Struktura objednávky je jednotná pro všechny produkty, podporované v rámci projektu CLAUDIA. Specifika jednotlivých produktů jsou zachycena v jejich struktuře v produktovém katalogu.

Zadávání objednávky je výrazně usnadněno nástrojem eConfigurator, který ulehčuje konfiguraci komplexních produktů v objednávce. Kromě komplexních produktů se v objednávkách používají také jednoduché produkty, které se upravují přímo změnou položek objednávky bez použití nástroje eConfigurator.

Po realizaci objednávky se její jednotlivé položky změní na instalované produkty. Instalované produkty mají stejnou strukturu jako položky objednávky – Instalovaný

produkt, komponenta instalovaného produktu a jejich atributy. Instalované produkty zachycují vždy aktuální stav produktů zákazníka.

2.4.1. Struktura objednávky

Objednávka představuje informaci o transakci, kterou si zákazník přeje provést se svými instalovanými produkty (např. zřízení, změna, zrušení).

Následující obrázek č.6 str.22 ukazuje příklad objednávky. Popis vyznačených rámečků následuje za obrázkem.

Soubor Upravit Zobrazit Procházet Dotaz Nástroje Nápověda powered by SIBEL

Dotazy: All Orders this Week

Číslo objednávky:

Domovská stránka **Zákazník** **Plátce** **Smlouvy** **Umístění** **Kontaktní historie** **Objednávky** **Instalované produkty** **Nabídky (cenové)** **Servisní požadavek**

Domovská stránka objednávek Seznam prodejních objednávek Seznam položek objednávek Seznam NP objednávek Grafy prodejních objednávek

Prodejní objednávka | 1 ze 1+ |

Nabídka | Spustit TO | Zjistit stav TO | Odeslat | Dotaz | Zaparkovat | Odparkovat | Waiting List | Vybrat šablony | ZIJ | INFO | Stornovat | Zrušit | Zamítnout

OBECHNÉ INFORMACE	ZÁKAZNICKÉ ÚDAJE	VAZBY	ORGANIZAČNÍ ZAŘAZENÍ
*Číslo objednávky: 1-1526714	Zákazník: ABET HOLDIN	Požadované datum: 25.3.2005	Prodejce - Login: DSTRASSER
*Transakce: Zřízení	Přednastavený plátce: ABET HOLDIN	Datum podání: 24.3.2005 17:48	Prodejce - Jméno: Daniel
*Status: Založeno	Přednastavené umístění:	Typ Kontaktní historie:	Prodejce - Přijetí: Strasser
Důvod statusu:	*Zaslat korespondenci na: Korespondenční	Datum Kontaktní historie:	Prodejce - Prodejní útvar: Default Organization
Organizace objednávky: Default Organizat	Tisknout: ☒	Příležitost:	Prodejce - Prodejní Kanál: Default Organization
Důvod objednávky:	*Způsob tisku korespondence: Hromadně	Ceník: x_mj_test	Prodejce - Pozice: CT Developer
Číslo hlavní obj: 1-1526714	Jméno zákazníka:	Plánované datum instalace:	Prodejce - Tým:
Hlavní objednávka: Ano	IČO: 25431005	Smlouva:	Prodejce zákazníka - Login: TRAINER1
Ověřeno: ☐	DIČ: CZ25431005	Číslo smlouvy:	Prodejce zákazníka - Tým:
Způsob ověření:	RČ zákazníka:	Status smlouvy:	Prodejce zákazníka - Útvar: Default Organization
Zdroj objednávky:	Segment zákazníka: BSBA1	Kampaň:	Prodejce zákazníka - Kanál: Default Organization
Číslo externí objednávky:	Jméno obchodní KO:	Popis:	Dealer - Název:
*Měna: CZK	Tel. číslo obchodní KO:		Dealer - IČO:
Vytvořit: PCAPEK	Email obchodní KO:		Distributor - Název:
*Vytvořeno: 24.3.2005 17:48	Korespondenční kontakt:		Distributor - IČO:

Další informace | Hlavní - podrobnosti | **Položky** | Realizace | Obchodní kontaktní osoba | Korespondenční kontaktní osoba | Platby | Výběr instalačního termínu | Dodávka

Položky | Přidat položku | Konfigurovat | Tisk | Odstranit | Zrušit skupinu | Kopírovat | Inst. produkty | Dotaz | 1 - 1 ze 1

Nabídka | Přidat položky | Přecenit | Přecenit vše | NT List | Rozdělit | Stornovat | Zrušit | Zamítnout

Pořadí	Skupina	Produkt	Kód produktu	Číslo řádku	Množství	Akce	Status	Důvod statusu	Popis St
1		HTS	PR0001	1-XT38	1	Přidat	Založeno		

Instalační otázky k položce obj. | Zadaní identifikace | **Detaily položky** | Technická kontaktní osoba | Součty | Akce | Obchody | Plány aktivit | Podrobnosti řádku

Nabídka | Dotaz | 1 - 1 ze 1

Název atributu	Hodnota	Akce	Popis
Způsob zapojení	100 Samostatné vec	Přidat	

Podrobnosti řádku | Status: Založeno | 1 z 1

Nabídka | Nové | Odstranit | Dotaz

INFORMACE POLOŽKY	STAVOVÉ INFORMACE	KONTAKTNÍ INFORMACE	DETAILY UMÍSTĚNÍ
*Číslo řádku: 1-XT38	Status: Založeno	Plátce: ABET HOLDING, a.s.	Ulice:
Číslo inst. produktu:	Detailní Status:	Číslo plátce: 0401147004	Číslo umístění:
Produkt: HTS	Důvod statusu:	Typ plátce: Ensemble	Umístění:
Kód produktu: PR0001	Status TO:	Jméno technické kontaktní os.:	Popisné:
Akce: Přidat	Datum změny statusu:	Tel. číslo technické kontaktní os.:	Orientační:
*Množství: 1	Popis Statusu:	Email technické kontaktní os.:	Číslo bytu:
ID služby:	Datum ukončení služby:	Připojit RSM: ☐	Číslo poschodí:
Původní ID služby:	Zpoplatnit od:	Instalační cena:	Město:
Host ID:	Datum podpisu objednávky:	Měsíční poplatek: 0,00 Kč	Městská část:
Priorita položky:	Datum urgency objednávky:	Uzamčení cen:	Okres:
Provisioning ID:	Datum vrácení objednávky:	Status aktualizace cen: Ceny nezměněny	Kraj:
NT List:	Status navracení objednávky: Nedoručeno - obj.	Poznámky:	PSČ umístění:
RSU ID:	Datum skutečné realizace:		ROP ID:
			Popis umístění:

ND SandBox Environment | 1 z 1

Obr.č. 6: Struktura objednávky

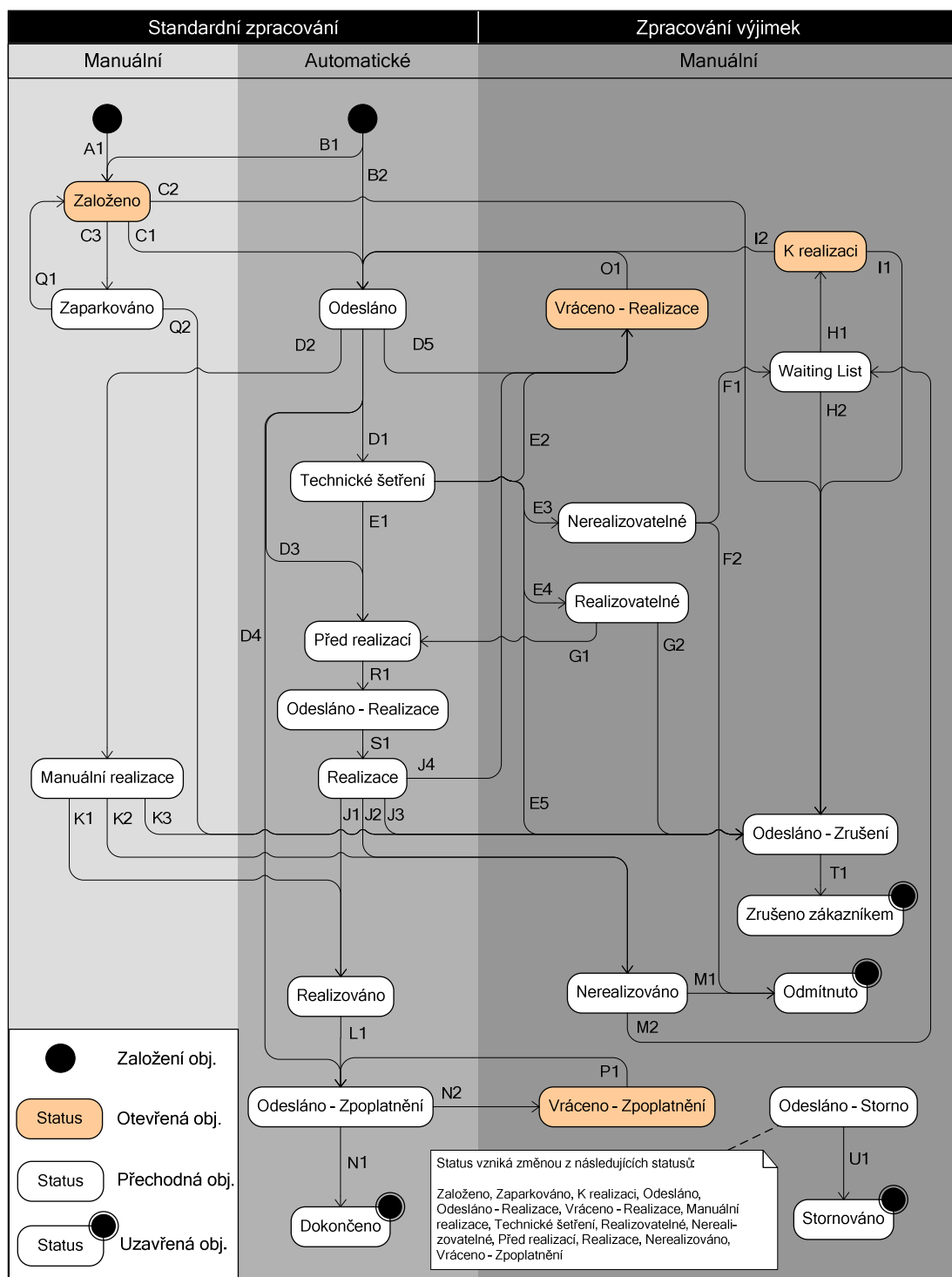
Objednávku tvoří následující entity:

- **Hlavička objednávky** (první rámeček na Obr.č. 6: Struktura objednávky)
 - může mít několik
 - Položek objednávky
 - Aktivit – slouží k předávání objednávek mezi útvary
 - je navázána na
 - Zákazníka
 - Kontaktní historii
 - Příležitost
 - Kampaň
 - Kontaktní osobu
- **Položky objednávky a jejich hierarchie** (druhý rámeček na Obr.č. 6: Struktura objednávky)
 - mohou mít několik
 - Atributů položek objednávky
 - Realizačních aktivit
 - jsou navázány na
 - Hlavičku objednávky
 - Instalovaný produkt, který vytvořila nebo mění
 - Produkt z produktového katalogu
 - Jinou Položku objednávky – tvoří hierarchii
 - Umístění
 - Plátce
- **Atribut položky objednávky** (třetí rámeček na Obr.č. 6: Struktura objednávky)
 - je navázán na
 - Položku objednávky
 - Atribut instalovaného produktu, který vytvořil nebo mění
- Realizační aktivita - je navázána na Položku objednávky
- Příloha objednávky - je navázána na Hlavičku objednávky

2.5. Životní cyklus objednávky

Na obr.č.7 str.25 dle (15) je popsán vývoj stavu objednávky v průběhu jejího životního cyklu. Vývoj stavu objednávek je rozdělen do následujících kategorií:

- **standardní ruční zpracování** – zahrnuje stavy, které vyžadují ruční akci uživatele. Jedná se o případy, kdy nenastaly žádné komplikace a není nutné řešit chyby či výjimky.
- **standardní automatické zpracování** – zahrnuje stavy, které vyplývají ze standardního automatického zpracování objednávek a nevyžadují ruční zásah uživatele. Není nutné řešit chyby či výjimky.
- **ruční zpracování výjimek** – zahrnuje stavy, které indikují stav mimo standardní zpracování objednávek. Tyto stavy vyžadují další ruční zpracování uživatelem.



Obr.č. 7: Životní cyklus objednávky

2.6. Přehled stavů objednávek

V této kapitole uvádím základní popis stavů objednávek. V přílohách 2 a 3 jsou uvedeny doplňující informace nezbytné k pochopení problematiky. Objednávka může nabýt těchto stavů viz tab.č.1 :

Stav Objednávky	Popis
Založeno	Počáteční stav – objednávka je založená ručně nebo automaticky. V případě ručního založení je možno pro každou ze základních položek objednávky spustit technické ověření. V průběhu technického ověření jsou vybrané atributy příslušné položky objednávky uzamčeny pro jakékoliv modifikace.
Zaparkováno	Stav pro založené objednávky, které nebylo možné odeslat k realizaci (např. kvůli chybějícímu ROP Id, zadání pokuty při existenci závazku atd.). Před dalším odesláním objednávky k realizaci musí být opětovně přepnuta do stavu Založeno.
Odesláno	Ručně kliknutím na tlačítko Odeslat je objednávka validována a odeslaná k: 1) technickému šetření, 2) realizaci, 3) zpoplatnění.
Technické šetření	Probíhá technické šetření včetně rezervací zdrojů. Výsledkem technického šetření je informace o možnosti realizace objednávky.
Před realizací	Některé položky objednávky mají nepotvrzené realizační aktivity typu „Před realizací“, tzn. před posunutím objednávky k realizaci je nutné ještě vykonat ruční realizaci, napr. v externím systému. Pro více informací o ruční realizaci obj. prostřednictvím aktivit viz školící modul Objednávkové Transakce I.
Odesláno – Realizace	Objednávka je odeslána k realizaci.
Realizace	Probíhá automatická realizace objednávky. Výsledkem realizace je realizovaná nebo nerealizovaná služba.
Ruční realizace	Probíhá ruční realizace celé objednávky (např. datové služby). Výsledkem realizace je realizovaná nebo nerealizovaná služba.

Stav Objednávky	Popis
Nerealizováno	Objednávku se nepodařilo zrealizovat. V případě univerzální služby ještě objednávka může být zařazena na Waiting List. Pozn. Waiting List představuje seznam nerealizovaných objednávek na zřízení služby, které se evidují za účelem plánování investiční výstavby sítě, po jejíž dokončení se objednávky mohou realizovat.
Realizováno	Objednávka byla úspěšně realizována. Objednávku již není možné stornovat.
Odmítnuto	Konečný stav – služba nemohla být realizována z technických důvodů na straně O2.
Nerealizovatelné	Z technického šetření vyplývá, že objednávku není možné zrealizovat nebo je realizovatelná přes FGSM. Zákazník má možnost potvrdit realizaci přes FGSM (zakládá se nová objednávka), být zařazen na waiting list nebo zrušit stávající objednávku.
Realizovatelné	Z technického šetření vyplývá, že objednávku je možné zrealizovat. Zákazník má možnost potvrdit realizaci služby nebo zrušit stávající objednávku.
Waiting List	Objednávka čeká na dokončení investiční výstavby.
Odesláno – Zrušení	Objednávka byla odeslána ke zrušení na žádost zákazníka. Čeká se na potvrzení zrušení z provisioning systému.
Zrušeno zákazníkem	Konečný stav – objednávka byla zrušena na žádost zákazníka.
K realizaci	Po dokončení investiční výstavby může být původní objednávka realizována. Objednávka podléhá standardnímu procesu včetně technického ověření a technického šetření.
Vráceno – Realizace	Objednávka byla odmítnuta po jejím odeslání. Důvod odmítnutí je specifikován na úrovni položek objednávky. Objednávka je otevřená pro modifikaci a po opravě může být opět odeslána.
Odesláno – Zpoplatnění	Zrealizována objednávka byla odeslaná ke zpoplatnění v billingu.
Vráceno – Zpoplatnění	Objednávka byla odmítnuta po jejím odeslání ke zpoplatnění. Důvod odmítnutí je specifikován na úrovni položek objednávky. Relevantní položky objednávky jsou otevřené k aktualizaci billingových dat.
Dokončeno	Konečný stav – objednávka byla úspěšně zrealizována i zpoplatněna v billingu.
Odesláno – Storno	Objednávka byla odeslána ke stornování. Čeká se na potvrzení storna z provisioning systému.

Stav Objednávky	Popis
Stornováno	Konečný stav – objednávka byla zrušena ze strany O2 z důvodu chybného zadání nebo jiného důvodu.

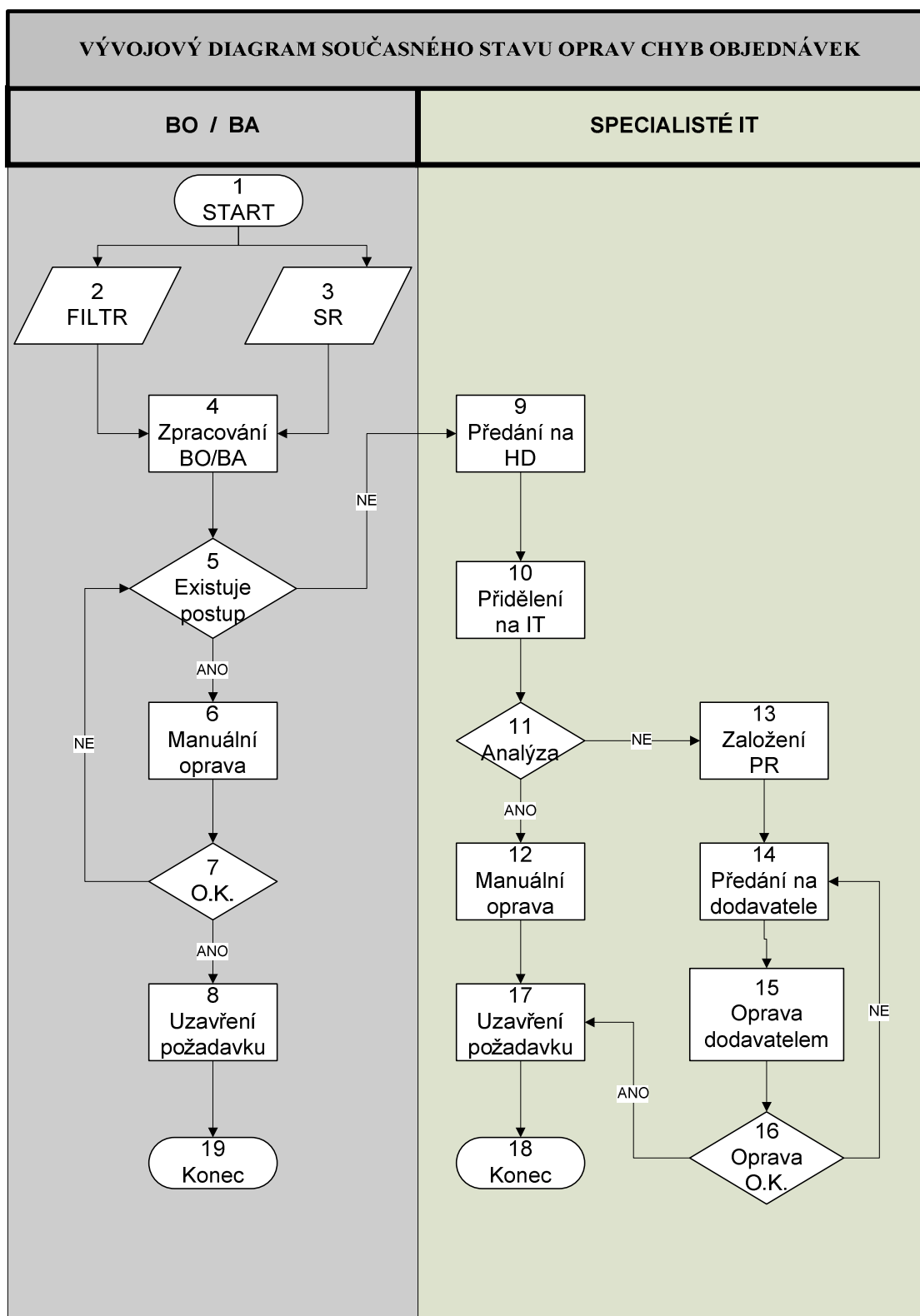
Tab. č. 1: Stavy objednávky

2.7. Vývojový diagram zpracování chyb

V současné době proces opravy chyb objednávkového systému spočívá v ručním procházení jednotlivých objednávek. Řešení se odehrává na bázi reaktivní se zapojením vysokého počtu zaměstnanců z různých oddělení. Oddělení jsou poplatná rozdělení zákazníků do segmentů.

- RS - residentní segment - fyzické osoby
- BS - business segment - právnické subjekty s výnosy do určité výše
- CS - corporate segment - právnické subjekty s výnosy od určité výše

Pro všechna tato oddělení platí procesní diagram, obr.č.8 s popisem činností v tabulce č.2 str.29 :



Obr. č. 8: Vývojový diagram současného stavu oprav

Aktivita	Popis
1 Start	Proces je startován každý pracovní den
2 Filtr	Jednotlivým pracovníkům je přidělen chybový stav. Zpracují si sestavu nad chybou, která jim přísluší. Sestavu tvoří přes aplikační rozhraní.
3 SR	Na speciální skupinu BO jsou přidělovány SR . Toto probíhá uvnitř aplikace. Zdrojem jsou urgencye a stížnosti zákazníků zachycené front kanály.
4 Zpracování BO/BA	Pracovníci BO/BA provedou kontrolu příslušnosti na BO/BA
5 Existuje postup	Pracovníci BO/BA zjistí, jestli na daný případ existuje postup řešení
6 Ruční oprava	Pokud existuje postup řešení, BA/BO provede opravu
7 O.K.	Kontrola, že se objednávka dostala do procesu zpracování. V případě, že se objednávka nepodařila opravit, návrat do bodu 5.
8 Uzavření požadavku	Pokud je kontrola v pořádku, SR se uzavírá a vrací zadavateli
9 Předání na HD	V případě, že neexistuje postup pro řešení, je požadavek na opravu objednávky předán na Help Desk. Help Desk provede zaevidování požadavku do systému Service Desk.
10 Přidělení na IT	Založený požadavek přidělí na příslušnou skupinu specialistů IT. Přidělení se řídí primární identifikací chyby a systému, kde chyba vznikla.
11 Analýza	Specialisté IT provedou analýzu požadavku. Po kontrole definují, jestli jsou schopni vyřešit na své úrovni nebo ne.
12 Ruční oprava	Pokud je oprava možná na této úrovni, specialisté IT ji provedou.
13 Založení PR	Pokud oprava není možná, specialisté IT zakládají problém, tj. vyšší forma servisního požadavku.
14 Předání na dodavatele	Problém specialisté IT předávají na dodavatelskou firmu. Vše se odehrává v aplikaci Service Desk.
15 Oprava dodavatelem	Dodavatel dodá opravný SW nebo postup opravy.
16 Oprava O.K.	Specialisté IT aplikují dodavatelské řešení, pokud není řešení v pořádku vrací problém dodavateli.
17 Uzavření požadavku	Pokud je oprava v pořádku uzavírají servisní požadavek
18 Konec	Proces končí
19 Konec	Proces končí

Tab. č. 2: Popis činností – původní

2.7.1. Analýza vstupu pro ruční opravy

V této kapitole provedu rozbor chyb vstupujících do procesu ručních oprav. V manažerském reportu jsou denně uváděny aktuální počty objednávek podle stavů. Pro ukázkou uvádím report ze dne 7.1.2009.

Stav	Category	7.1.2009
Dokončeno	Final stav	9 261 986
Stornováno	Final stav	1409128
Zrušeno zákazníkem	Final stav	136058
Odmítnuto	Final stav	149869
Sub total		10957041
K realizaci	Manual intervention of BO	118
Nerealizováno	Manual intervention of BO	282
Nerealizovatelné	Manual intervention of BO	949
Realizovatelné	Manual intervention of BO	427
Waiting list	Manual intervention of BO	11892
Založeno	Manual intervention of BO	856
Zaparkováno	Manual intervention of BO	17671
Vráceno - realizace	Manual intervention of BO	879
Vráceno - zpoplatnění	Manual intervention of BO	1439
Realizováno	Manual intervention of BO	178
Ruční realizace	Manual intervention of BO	2240
Sub total		36931
Realizace	Correct stav - future processing	12834
Technické šetření	Correct stav - future processing	33
Odesláno	Correct stav - future processing	4186
Odesláno - realizace	Correct stav - future processing	43
Odesláno - Storno	Correct stav - future processing	6
Odesláno - zrušení	Correct stav - future processing	0
Pozdrženo - Zpoplatnění	Correct stav - future processing	10
Vygenerováno - Zpoplatnění	Correct stav - future processing	2625
Sub total		19737
Odesláno - zpoplatnění	Detailed split in List4	457
Realizace	Potentially correct stav	12491
Technické šetření	Potentially correct stav	84
Sub total		12575
Odesláno	Error stav - processing in the past	5701
Odesláno - realizace	Error stav - processing in the past	701
Odesláno - Storno	Error stav - processing in the past	247
Odesláno - zrušení	Error stav - processing in the past	14
Sub total		6663
Total		11033404

Tab. č. 3: Manažerský report

V manažerském reportu viz. tab.č.3 jsou objednávky rozděleny do několika barevných bloků.

- **Zelený blok** – v něm jsou uvedeny počty a stavy tzv. objednávek v koncovém stavu. Popis, proč tomu tak je, je uveden v tab.č.1
- **Žlutý blok** – zde jsou uvedeny objednávky, které vyžadují ruční uživatelskou intervenci
- **Modrý blok** – obsahuje objednávky, které jsou v pořádku a zatím jim nevypršel datum realizace
- **Okrový blok** – realizované, nezpлатněné objednávky, které čekají na ruční kontrolu. Tato chyba se rozpadá na jednotlivé příčiny, jejich analýza a rozbor jsou nad rámec této práce
- **Fialový blok** – objednávky, které probíhají v provisioning systémech, mohou, ale nemusí být v pořádku, report je podrobněji neanalyzuje
- **Červený blok** – jsou objednávky, u kterých vypršel datum realizace (Odesláno) nebo jsou ve stavu (ostatní), ve kterém se dle tab.č.1 objednávka nesmí zastavit

2.7.2. *Identifikace objednávek s potenciální chybou a kalkulace zdrojů*

Z předchozí kapitoly vyplývá, že v životním cyklu objednávku je nutný ruční zásah a to na objednávkách ve žlutém, okrovém a červeném bloku. Ve žlutém bloku se jedná o objednávky ve stavu Vraceno – Zpлатnění a Vraceno – Realizace. Tyto jsou jednoznačně v chybě. Z ostatních bloků, vyjma zeleného, jsou potenciálně „zamrzlé“ i objednávky ve stavech Realizace, Odesláno, Vygenerováno – Zpлатnění, Odesláno – Zpлатnění a Pozdrženo – Zpлатnění. Identifikace těchto objednávek je za současného stavu nástrojů pro kontrolu velmi omezená. Zjištění o jejich poškození je detekováno až stížností zákazníka na nerealizování jím požadovaného úkonu.

Pro výpočet požadovaných lidských zdrojů tedy použijeme tyto počty objednávek :

- Žlutý blok : Vraceno – Zpлатnění (1439), Vraceno – Realizace (879)
- Fialový blok : Realizace (12491)
- Červený blok : všechny stavy (6663)
- Modrý blok : Vygenerováno – Zpлатnění (2625), Pozdrženo – Zpлатnění(10)
- Okrový blok : Odesláno – Zpлатnění (457)

Součet těchto hodnot je 24564. Stanovenou denní normou pro řešení na BO je 50 kusů objednávek. Pokud bychom požadovali kontrolu a opravu všech objednávek v jednom

dni, bylo by potřeba 490 pracovníků. Tím by se provedla jednorázová oprava. Systém ovšem není zafixován a každý den se stavy mění a je potřeba pracovat přírůstkovou metodou. Součástí manažerského reportu je tedy i evidence přírůstku:

Stav	Category	7.1.2009	Delta	8.1.2009
Dokončeno	Final stav	9 261 986	9 914	9 271 900
Stornováno	Final stav	1409128	1 012	1410140
Zrušeno zákazníkem	Final stav	136058	84	136142
Odmítnuto	Final stav	149869	19	149888
Sub total		10957041	11 029	10968070
K realizaci	Manual intervention of BO	118	-32	86
Nerealizováno	Manual intervention of BO	282	676	958
Nerealizovatelné	Manual intervention of BO	949	-664	285
Realizovatelné	Manual intervention of BO	427	-199	228
Waiting list	Manual intervention of BO	11892	13	11905
Založeno	Manual intervention of BO	856	-106	750
Zaparkováno	Manual intervention of BO	17671	280	17951
Vráceno - realizace	Manual intervention of BO	879	-15	864
Vráceno - zpoplatnění	Manual intervention of BO	1439	63	1502
Realizováno	Manual intervention of BO	178	248	426
Ruční realizace	Manual intervention of BO	2240	-8	2232
Sub total		36931	256	37187
Realizace	Correct stav - future processing	12834	462	13296
Technické šetření	Correct stav - future processing	33	-21	12
Odesláno	Correct stav - future processing	4186	-142	4044
Odesláno - realizace	Correct stav - future processing	43	-30	13
Odesláno - Storno	Correct stav - future processing	6	-1	5
Odesláno - zrušení	Correct stav - future processing	0	2	2
Pozdrženo - Zpoplatnění	Correct stav - future processing	10	2	12
Vygenerováno – Zpoplatnění	Correct stav - future processing	2625	591	3216
Sub total		19737	863	20600
Odesláno - zpoplatnění	Detailed split in List4	457	-58	399
Realizace	Potentially correct stav	12491	-1 130	11361
Technické šetření	Potentially correct stav	84	-68	16
Sub total		12575	-1 198	11377
Odesláno	Error stav - processing in the past	5701	-232	5469
Odesláno - realizace	Error stav - processing in the past	701	-572	129
Odesláno - Storno	Error stav - processing in the past	247	8	255
Odesláno - zrušení	Error stav - processing in the past	14	0	14
Sub total		6663	-796	5867
Total		11033404	10 096	11043500

Tab. č.4: Manažerský report – denní přírůstek

Z tab.č.4 jasně vyplývá, že v kritických stavech je poměrně vysoký úbytek. Toto je důkazem toho, že se s reportem pracuje a jsou delegovány prostředky na krizová místa. V současné době na BO jednotlivých segmentů pracuje cca 70 zaměstnanců, což je k výše uvedenému poměrně nízký stav. Vzhledem k nákladům je nereálné tento stav zvyšovat a je nutno hledat jiné, nekonvenční prostředky. Jedním z prostředků je i cíl této diplomové práce.

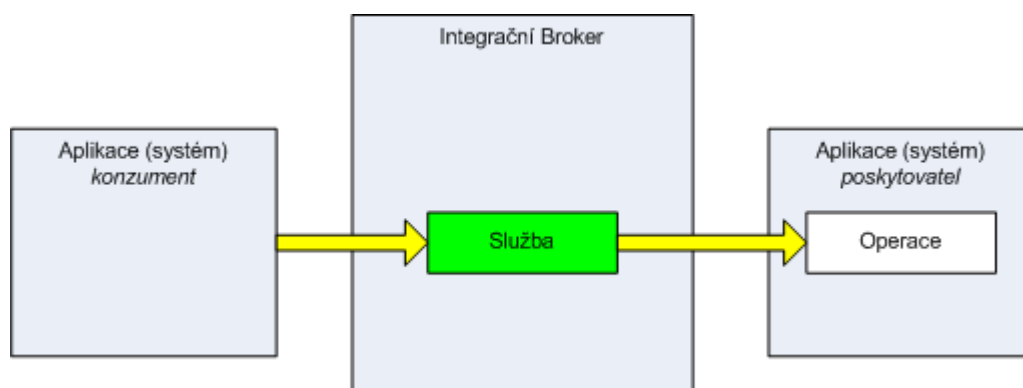
2.8. Integrační SW

V každé moderní technologické společnosti, dle (13), jakou je například O2, vyvstane časem potřeba pro *integraci* jejích aplikací. Pod pojmem *integrace* zde rozumíme schopnost jednotlivých aplikací nabídnout *službu* a schopnost jiných aplikací tuto službu *konzumovat*. Tento princip je založen na architektuře SOA (service oriented architecture) a ESB (enterprise service bus). Tyto pojmy budou osvětleny později. V O2 byl pro účely vyřešení integrace vytvořen projekt *Integrační Broker*.

Integrační Broker je oficiální integrační nástroj O2, který má za cíl podporovat a usnadňovat budování rozhraní, výměnu dat, vzájemnou komunikaci mezi aplikacemi a jejich zapojení do celofiremních procesů.

2.8.1. Služba

Pod pojmem *služba* zde rozumíme schopnost aplikace nabídnout operaci jako *službu*, kterou definovaným způsobem mohou konzumovat ostatní aplikace:

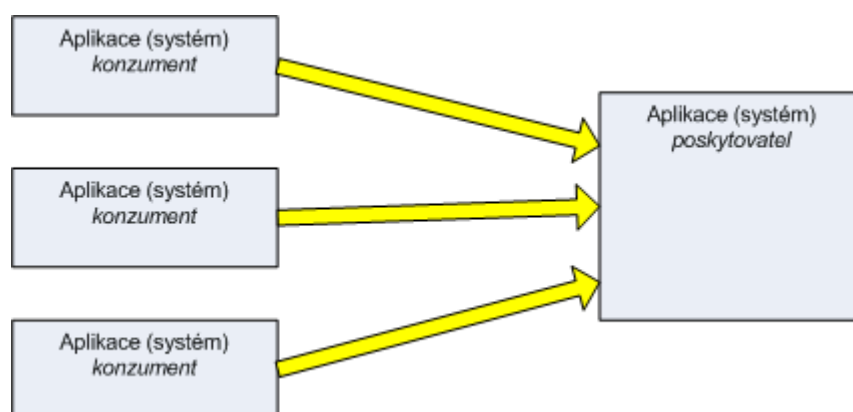


Obr. č. 9: Služba

Implementace služby je tudíž připojení aplikace k IB a poskytnutí operací jako obecných služeb dalším aplikacím. Připojením aplikace k IB se poskytovatel zavazuje dodržovat určitá pravidla, která potom umožní konzumentům využívat služeb jednotným a definovaným způsobem.

2.8.2. *Enterprise Service Bus (ESB)*

Klasická architektura SOA je založená na principu poskytování služeb jednotlivými aplikacemi, kdy konzument volá přímo poskytovatele.

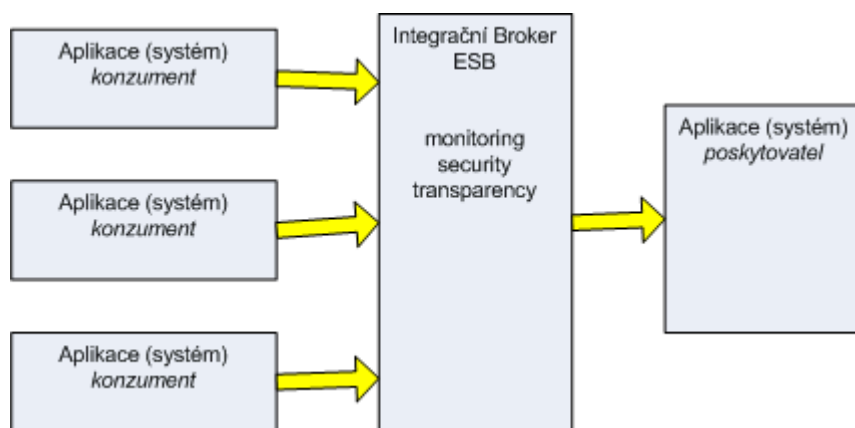


Obr. č. 10: SOA

Tento princip je užitečný, nicméně přináší některé nevýhody:

- Každá aplikace musí implementovat standardní technologii pro své konzumenty. Každý konzument musí implementovat technologii pro volání poskytovatele. Tento problém se většinou řeší použitím technologie WS. Některé *legacy* systémy ovšem nemají prostředky pro implementaci technologie WS.
- Nelze centrálně monitorovat komunikaci.
- Nelze centrálně řídit bezpečnost.
- Při změně poskytovatele se musí specificky změnit i všichni konzumenti.

Z těchto důvodů se do architektury SOA zavádí prvek ESB. ESB zaručuje řešení nevýhod čistého SOA.

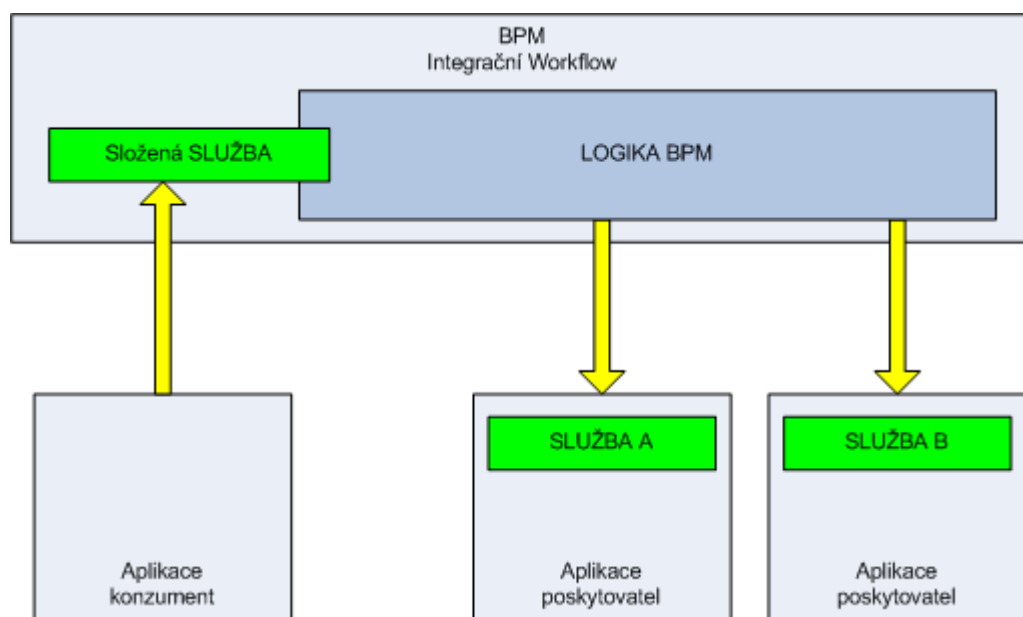


Obr.č. 11: ESB

Jednou z obrovských výhod řešení pomocí architektury ESB je transparentnost, tj. pokud změním poskytovatele, při zachování rozhraní služby není třeba provádět jakékoliv změny na straně konzumenta.

2.8.3. Složená služba (BPM) – Integrační workflow

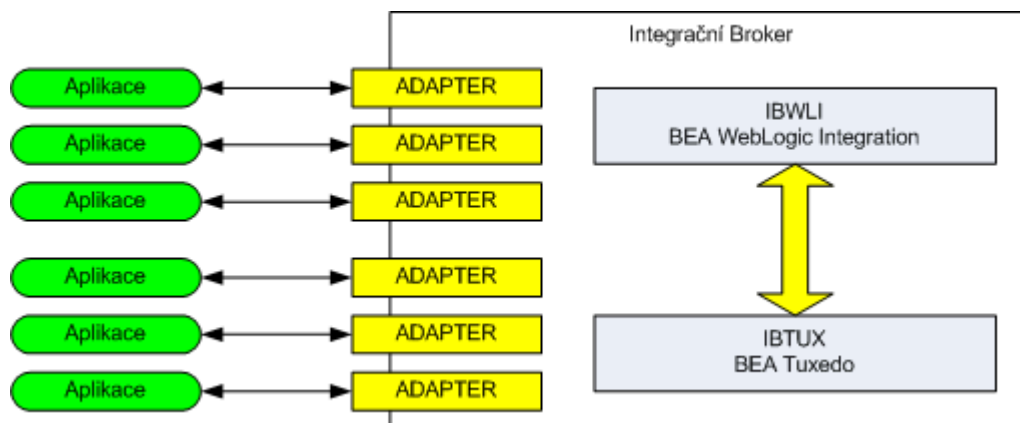
Nedílnou součástí IB je možnost tvorby složených služeb (composed services), tj. spojovat služby různých aplikací s předepsanou logikou a to jak v rozhodování, tak v čase, tak v datovém obsahu (datové transformace). Tento proces tvorby složených služeb se nazývá *Business Process Management*.



Obr. č. 12: BPM

2.8.4. Základní schéma

Integrační Broker se skládá ze dvou relativně samostatných částí IBTUX a IBWLI, které jsou vzájemně propojeny do kompaktního funkčního celku.



Obr. č. 13: Integrační broker

2.8.5. IBTUX

IBTUX je vybudován na platformě BEA Tuxedo (verze 8.0), která slouží jako základní komunikační prostředí pro jeho součásti.

Součásti IBTUX jsou

- centrální komponenty, které řídí vyřizování přijatých požadavků, provádějí autorizaci požadavků, datové konverze, směrují požadavky, zaznamenávají chybové stavy a protokolují a řídí činnost IB.
- aplikační adaptory, které připojují k integrační platformě jednotlivé aplikace, autentizují tyto aplikace a umožňují přijímání požadavků na služby (zpráv) z aplikací, resp. doručování požadavků (zpráv) aplikacím. Adaptory komunikují směrem k Tuxedo jednotně protokolem ATMI, směrem k aplikacím komunikují protokolem (technologií), který je pro aplikaci nejpřijatelnější. Adaptory slouží jako "převodníky" mezi aplikací a platformou. V současné době jsou k dispozici adaptory pro ORACLE PL/SQL, COM, RPC, Tuxedo ATMI, SAP RFC, FORTEL.

IBTUX funguje jako "message broker", to znamená, že přijatý požadavek (zprávu) po případné datové konverzi směřuje na obsluhující aplikaci a případnou odpověď aplikace vrací zpět.

2.8.6. IBWLI

IBWLI je vybudován na platformě BEA WebLogic Platform (Verze 8.1 SP3) a to konkrétně z těchto komponent:

- BEA WebLogic Server (WLS) – J2EE aplikační server
- BEA WebLogic Integration (WLI) – Integrovaná platforma pro realizaci BPM
- IBWLI Foundation – Implementace ESB

IBWLI umožňuje volání služeb

- **synchronní** - kdy volající aplikace čeká na dokončení komunikace se vzdálenou aplikací či integračním workflow na obdržení výsledku zpracování
- **asynchronní** - kdy volající aplikace nečeká na dokončení komunikace se vzdálenou aplikací či integračním workflow a neočekává návrat výsledku zpracování, volání služby se vrátí okamžitě po převzetí požadavku IBWLI. V případě nemožnosti předání zprávy do IBWLI se vrátí komunikační chyba.

2.9. Provisioning – OMS

OMS je aplikace pro řízení procesu, viz.(14). Objednávky se mohou vkládat buď přes GUI, nebo přicházet přes upstream interface z orderingového systému, což je případ O2 (s výjimkou procesu Univerzál, kde se zakládají ručně).

Po založení objednávky aplikace směřuje objednávku do jednotlivých úloh (task), z nichž je sestaven Proces. Úlohy jsou automatické (automatické transakce s jinou aplikací), ruční (provádějí je uživatelé přes grafické rozhraní) a dále rule (větvi proces podle určitých kritérií) a delay (čeká na splnění určité podmínky), přičemž poslední dva typy provádí interně OMS pomocí tzv. Rule Engine.

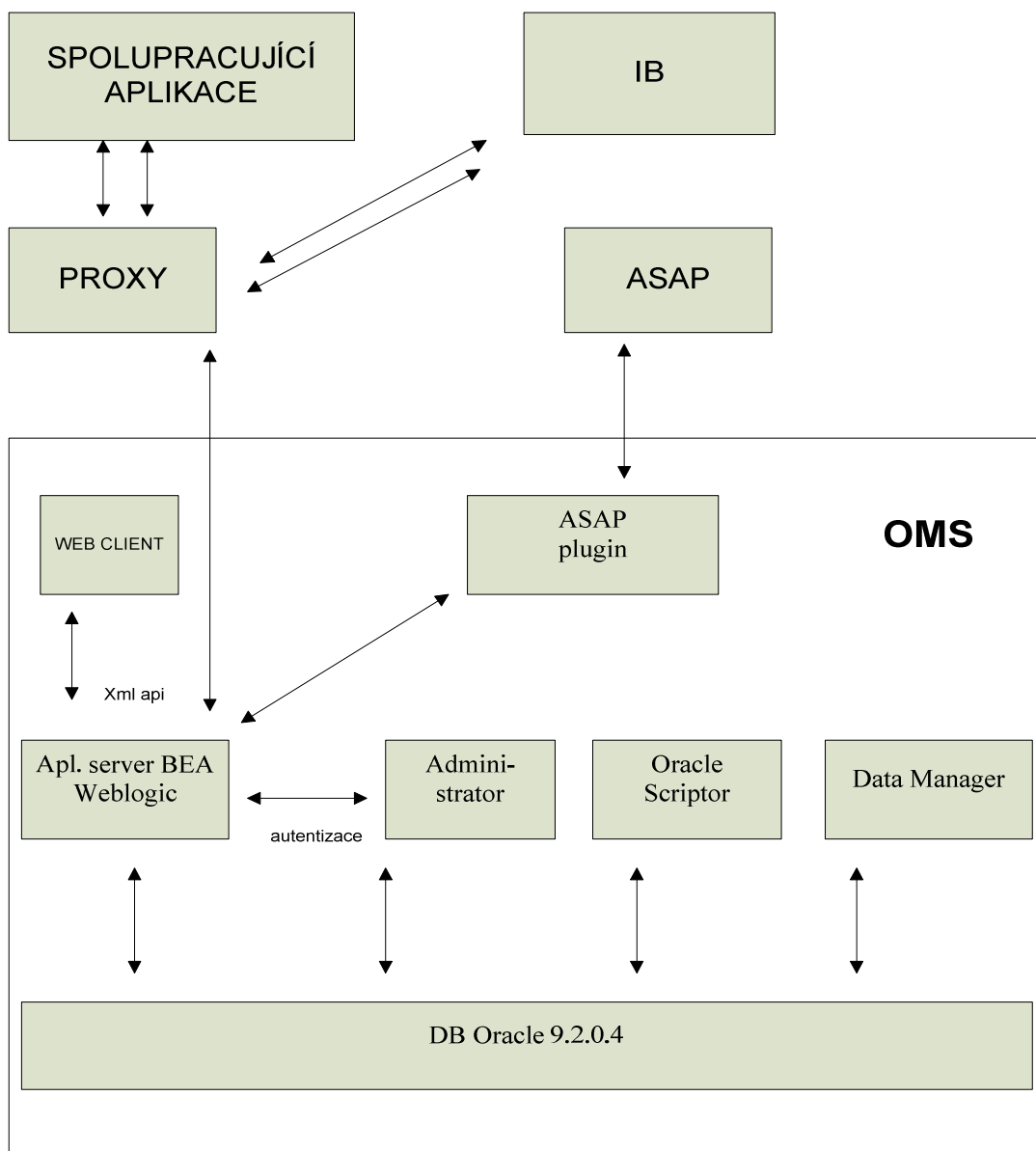
2.9.1. Architektura OMS

OMS má třívrstvou architekturu. Jako úložiště dat se používá databáze Oracle. S ní komunikuje vlastní aplikace běžící na aplikačním serveru BEA Weblogic. Přes BEA komunikují uživatelé pomocí webového klienta i spolupracující aplikace (konkrétně tzv. proxy komponenty) přes xml API.

Toto základní schéma dále doplňují tři klientské aplikace s tlustým klientem pro Windows, jež využívají správci aplikace (v našem případě ISD a ISO):

- Administrator – slouží k úpravě procesů a jiných objektů, které jsou součástí uživatelského datového modelu
- Oracle Scriptor – slouží např. k vytvoření aplikačního schématu v db, při upgradech na vyšší verzi se používá pro spouštění migračních skriptů
- Data Manager – slouží především k přenosu uživatelského datového modelu z vývojového prostředí na test a dále na provoz

Názorně je architektura aplikace OMS i její vazby na okolní prostředí zachyceny na následujícím schématu:



Obr.č. 14: Schéma OMS

3. Teoretické podklady k řešení – dolování dat

O2 má, jako každá moderní firma se zaměřením na zákazníka, svůj vlastní datový sklad. V něm dochází ke spojení dat ze všech aplikačních databází, které poskytují obrovská množství dat. Jedná se o zákaznický orientovaná data. Obsahem ale nejsou potřebné informace pro řešení problému chyb objednávek, a proto musím vytvořit svůj vlastní, rozsahem drobný datový sklad, obsahující specifická data použitelná pro opravu chyb. Vlastní opravný SW je vlastně vyvrcholením vysoce sofistikované analýzy nad shromážděnými daty.

Rychle se rozvíjející informační a komunikační technologie (ICT) umožňují ukládat a komunikovat velké množství dat nejrůznějšího charakteru. Přesto často nejsou schopny splnit všechny požadavky, se kterými se na ně obracíme. Rozvoj oblastí **dobývání znalostí** v databázích nebo úžeji **dolování dat** svědčí o tom, že situace se začíná měnit. Tato teoretická část práce si proto klade za cíl podat vysvětlení některých pojmů, s nimiž se v těchto a souvisejících oblastech setkáváme.

Dobývání znalostí z databází

O problematice dobývání znalostí z databází (**Knowledge Discovery in Databases**, KDD) se začalo výrazněji diskutovat počátkem 90. let 20. století, přičemž první impulsy přišly z konferencí o umělé inteligenci, které se konaly v USA. Podle (3) byla fráze knowledge discovery in databases vytvořena na prvním workshopu KDD již v roce 1989.

Znalosti se však nezačaly dobývat na zelené louce, ale byly využity již existující metody, postupy či technologie, zejména pak :

- metody strojového učení (součást umělé inteligence)
- databázové technologie (prostředek uchování rozsáhlých dat a vyhledávání v nich)
- statistické postupy (prostředek modelování a analýzy závislostí v datech)

Tyto disciplíny se předtím vyvíjely nezávisle. S nárůstem objemu zpracovávaných dat a se vznikem potřeby shromážděná data využívat pro podporu strategického rozhodování ve firmách se vytvořilo příhodné prostředí pro jejich propojení.

KDD můžeme podle definice (3) z roku 1996 i podle následných prací definovat jako netriviální získávání implicitních, dříve neznámých a potencionálně užitečných informací z dat. KDD bývá chápáno také jako interaktivní a iterativní proces tvořený kroky selekce, předzpracování, transformace, vlastního dolování (data mining) a interpretace.

V (1) si sice můžeme v předmluvě přečíst následující motto :

„Máte – li špatné údaje, ale dokonalou logiku, pak jsou vaše závěry zcela jistě mylné. Dopřejete- li si tudíž sem tam nějakou trhlinu v logickém uvažování, můžete díky náhodě dospět ke správnému závěru“.

Nepochybně však platí, že čím lépe jsou data v počáteční fázi připravena, tím lepších konečných výsledků dosahujeme. Na čištění dat dnes existují nejrůznější profesionální nástroje. Samotný proces čištění dat probíhá podle (8) v těchto fázích :

- analýza (investigation) – zkoumání charakteru vstupních dat (formáty a typy záznamů, identifikace obecných údajů v záznamech, frekvenční analýza atd.)
- standardizace – jednotná reprezentace informací vhodná pro další zpracování (identifikace jednotlivých položek v záznamu, jejich transformace, konverze apod.)
- obohacení (enrichment) – kompletace dat (včetně kontroly správnosti) a jejich rozšíření z jiných interních/externích zdrojů (např. číselníků)
- hledání souvislostí (linking) – identifikace vazeb mezi individuálními záznamy, seskupování (agregace) údajů, odstraňování duplicit
- integrace – umožnění implementace jednotného procesu kontroly a zušlechťování kvality dat

Impulsem pro zahájení procesu dobývání znalostí je nějaký reálný problém. Cílem procesu dobývání znalostí je získat co nejvíce relevantních informací vhodných k řešení problému.

Řešení problému probíhá postupně v několika etapách, jimiž jsou :

- vytvoření řešitelského týmu, mezi jehož členy by se měli objevit tyto osoby
 - expert na řešení problematiku
 - expert na data
 - expert na metody KDD
- specifikace problému
- získávání veškerých dostupných dat (i externích), která mohou být použita pro řešení problému
- výběr metod analýzy dat, například
 - klasifikační metody
 - různé klasické metody explorační analýzy dat
 - metody pro získávání asociačních pravidel
 - rozhodovací stromy
 - genetické algoritmy
 - bayesovské sítě
 - neuronové sítě
 - hrubé množiny (rough sets)
 - metody vizualizace
- předzpracování dat (příprava dat do formy vyžadované pro aplikaci vybraných metod, odstranění odlehlých a doplnění chybějících hodnot)
- dolování dat (data mining, modeling, analysis; aplikace vybraných analytických metod pro vyhledávání zajímavých vztahů v datech, přičemž metody data miningu jsou obvykle aplikovány vícekrát)
- interpretace (zpracování množství výsledků jednotlivých metod – některé z výsledků jsou z hlediska uživatele nezajímavé nebo samozřejmé)

Mezi úlohy KDD patří :

- klasifikace nebo predikce – cílem je nalézt znalosti použitelné pro klasifikaci nových případů, přednost má pokrytí na úkor jednoduchostí; rozdíl mezi klasifikací a predikcí je v tom, že u predikce hraje důležitou roli čas
- deskripce – cílem je nalézt dominantní struktura nebo vazby, které jsou skryté v daných datech
- hledání nuggetů – požadujeme nové, překvapivé znalosti, které nemusí plně pokrývat daný koncept

K nejpoužívanějším metodikám patří metodika 5A a CRISP- DM.

Metodika 5A zahrnuje těchto pět kroků :

- **Asses** – posouzení potřeb projektu
- **Access** – shromáždění potřebných dat
- **Analyze** – provedení analýz
- **Act** – přeměna znalostí na akční znalosti
- **Automate** – převedení výsledků analýzy do praxe

Metodika CRISP-DM (Gross- Industry Standard Process for Data Mining) se snaží nalézt univerzálně použitelný postup pro dolování dat. Z použité literatury lze také usuzovat, že terminologie uváděná v rámci této metodiky tvoří v oboru již standard, alespoň ve své anglické jazykové mutaci. Metodika CRISP – DM uvádí etapy :

- porozumění problematice (Business Understanding)
- porozumění datům (Data Understanding)
- příprava dat (Data preparation)
- modelování (Modeling)
- vyhodnocení výsledků (Evaluation)
- využití výsledků (Deployment)

Hovoříme-li o dobývání znalostí v databázích a o dolování dat, je vhodné připomenout si ještě existenci následujících dvou oblastí:

- knowledge discovery in text (resp. text mining), což není nic jiného, než speciální typ úlohy dobývání znalostí z databází (hlavním problémem je vhodná reprezentace původního nestrukturovaného textového dokumentu)
- web mining, který bývá dále členěn na web content mining, web structure mining a web usage mining

Datový sklad

Bez datového skladu (data warehouse) se při dolování dat a návazně dobývání znalostí v databázích neobejdeme, proto si jeho charakteristické rysy nyní detailněji představme.

Datový sklad představuje ucelené řešení, které poskytuje jednak prostředky pro ukládání dat, jednak sadu nástrojů pro jejich analýzu. Klíčovou roli v datovém skladu hrají relační databáze. Pro datový sklad je dále typické, že je neustále rozšiřován bez redukce obsahu (toto tvrzení však neplatí ve všech případech, viz. (4)). V datových skladech se setkáváme také s tzv. datovými pumpami neboli ETL (extrakt, transform & load) nástroji, které umožňují plnění databáze daty (nejprve získávají data ze vzájemně nekompatibilních zdrojů, poté je transformují do nových struktur a na závěr je ukládají do datového skladu).

Datový sklad je orientován na subjekty, kterými se firma zabývá (zákazník, dodavatel, produkt, aktivita) a uchovává data pro podporu rozhodování na manažerské úrovni. Lze jej také označit za integrovaný (jedná se o integraci a sjednocení dat) a časově proměnný (všechna data v datovém skladu představují časový snímek dat z produkčních databází v určitém okamžiku). Datový sklad je aktualizován v určitých časových intervalech (např. měsíčně, čtvrtletně, ročně) a analyzován odděleně od produkčních bází (ty uchovávají data potřebná pro operativní řízení), takže případný problém ve funkčnosti databázového skladu neovlivní operativní řízení firmy. Datový sklad je rovněž read – only, což znamená, že činnosti, které do datového skladu směřují uživatelé – analytici, nezpůsobují změnu uložených dat. Data uložená v datovém skladu

často představují neutrální datový prostor, který není vytvářen s myšlenkou konkrétní analýz. Z tohoto důvodu se doporučuje vytvářet v návaznosti na datový sklad řadu specializovanějších datových tržišť (data marts), kam se z datového skladu přesunou data relevantní pro určitý typ analýzy (resp. pro určité oddělení firmy).

V (5) nalezneme osm základních doporučení, kterých bychom se při tvorbě datového skladu měli držet :

- 1) začněte v malém (např. data mart pro jedno oddělení, výhodou je mj. rychlost implementace) ...
- 2) ... ale myslíte ve velkém
- 3) stanovte cíle a vyčíslete přínosy
- 4) zapojte top management
- 5) přehnaný perfekcionalismus není přínosem
- 6) vyberte systém, který se přizpůsobí potřebám uživatelům
- 7) ujistěte se, že komponenty datových skladů skutečně spolupracují (pro datové sklady zatím neexistují standardy v podobném rozsahu jako v ostatních oblastech IT : zejména TCP/IP pro sítě, SMTP pro elektronickou poštu a HTML a Java pro WEB)
- 8) váha mezilidských vztahů – k tomuto bodu dodejme, že s datovými sklady je spjato sdílení informací, což pro mnoho uživatelů může znamenat ztrátu kontroly. S tímto fenoménem, který autor v (5) výstižně označuje jako datový provincialismus, je potřeba se vyrovnat hned zpočátku

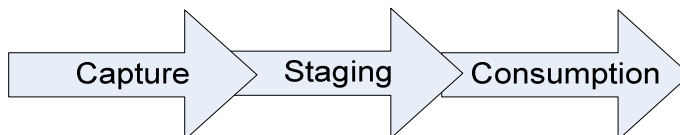
Z praxe a provedených studií zároveň vyplývá, že jestliže uživatelé získají přístup k informacím z datového skladu, zanedlouho zjistí, že jich potřebují mnohem více a na větší úrovni podrobnosti.

Z (9) je patrné, že nejde především o samotnou implementaci datového skladu jako takového, ale zejména o to, že by jím organizace měla řešit skutečné potřeby, a to nejenom své, ale především svých zákazníků. Proto je nezbytné, aby datové sklady poskytovaly využitelné informace, tedy informace, které přinesou v krátké době

měřitelné finanční výsledky (zvýšení obrátu a zisku, snížení nákladů, udržení zákazníků apod.).

V aplikaci CLAUDIA je datový sklad s názvem ODSP, který se obohacuje pomocí technologie Oracle streams. Jedná se o proces zabezpečující přenos obrovského množství dat online, takže datový sklad je prakticky neustále věrnou kopií produkční databáze a je připraven okamžitě poskytovat důležité informace, např. o prodeji, ale i pro níže řešený SW. Přenosem dat do ODSP pro jiné, než prodejní činnosti, je zabezpečena absolutní přednost prodejních kanálů v dostupnosti HW prostředků, které potřebují pro základní firemní činnost, prodej produktů a obsluhu zákazníků.

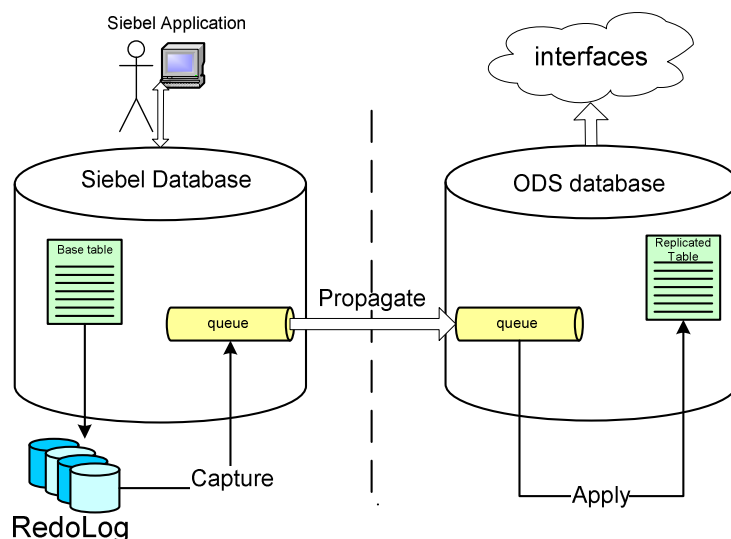
Dle (16), technologie Oracle Streams slouží k online datovému přenosu mezi databázemi pomocí transakcí. Celý proces přenosu se skládá ze tří částí: Capture (z redologu nasbírá data a přefiltruje), Staging (nebo také Propagation, přenesení data z fronty na zdrojové databázi do fronty na cílové databázi) a poslední část je Consumption (nebo také Apply, který zapisuje přenesené datové změny na cílové databázi). Schématické znázornění postupnosti jednotlivých kroků je znázorněn na obrázku č. 15.



Obr.č. 15: Základní části procesu replikace

Capture proces čte změny v redologu, nebo v archiveologu a nechá je projít filtrem, který odfiltruje jen změny, které jsou relevantní pro sadu pravidel, které jsou nastaveny při konfiguraci (filtrují se pouze celé tabulky, které jsou třeba jako zdroj dat na straně ODSP). Zachycené data přemění na LCR (logical change record), který obsahuje informace o změnách, které se mají provést, což znamená, že při update obsahuje staré hodnoty a nové hodnoty. Streams zachytává DML (insert, update a delete) a DDL (alter, drop a rename). Výsledné LCR zapíše do fronty na zdrojové databázi, kde si je vyzvedne staging proces a přenesení je do fronty na cílové databázi. Zde je vybírání apply

proces, který je zapíše do cílových tabulek. Před vlastním uložením (commitem) se spustí trigger navěšené na cílových tabulkách. Trigger zapisují data do front, které jsou použity pro další transformace. Schématický obrázek je znázorněn na obr.č. 16 Popis jednotlivých procesu capture, propagate a apply je popsán v následujících kapitolách.

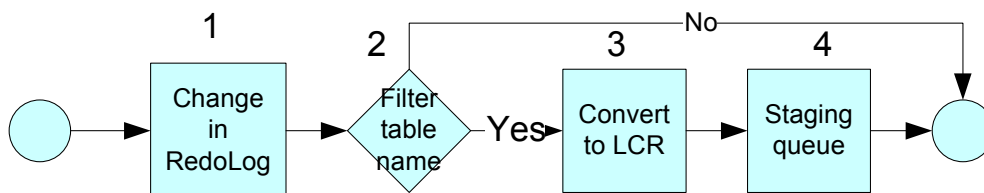


Obr.č. 16: Process capture, propagate a apply

3.1. Capture

Capture process lze rozdělit do několika kroků, jak ukazuje obr.č. 17.

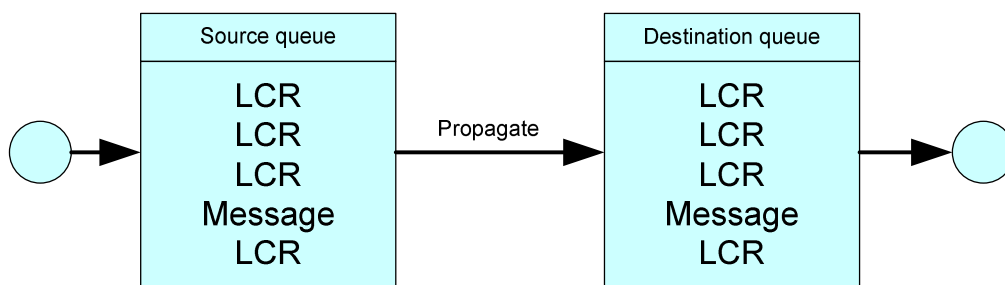
- 1) Capture process čeká na změnu v redologu.
- 2) Vyhodnotí jestli je zachycená změna pro capture process relevantní. V našem případě, jestli je změna provedena na definovaných tabulkách, které chceme přenést. Všechny ostatní změny jsou ignorovány.
- 3) Záznam z redologu přetvoří na LCR.
- 4) LCR zapíše do fronty, která je společná s propagation procesem.



Obr.č. 17: Diagram capture procesu

3.2. Propagace

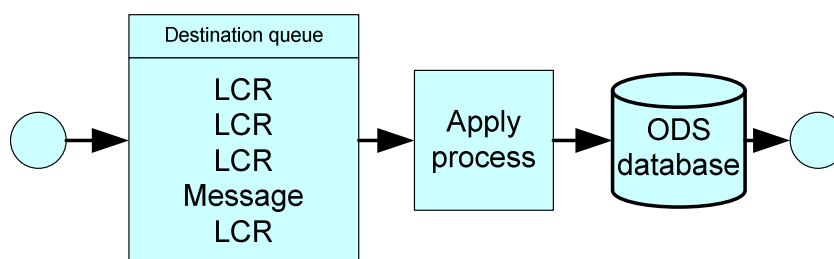
Propagation má za úkol přenos jednotlivých změn mezi frontami na zdrojové a cílové databázi. Pro proces propagace je nastaveno jen jedno globální pravidlo, proto se přenáší vše co přijde do fronty na zdrojové databázi. Schématicky na obr.č. 18.



Obr.č. 18: Schéma procesu propagace

3.3. Consumption

Apply proces je konečný proces, který zapisuje všechny data z fronty na cílové straně do databáze. Vykonává všechny DML změny, v případě, že nesedí staré hodnoty v LCR se stávajícími, zapíše transakci do dba_apply_error. V případě, že přijde DDL zpráva, zalogue ji do tabulky strmadmin.HISTORY_DDL a vykoná pouze ALTER TABLE, ostatní DDL příkazy pouze loguje.



Obr.č. 19: Schéma procesu apply

4. Vlastní řešení

Jak je z výše uvedeného zřejmé, na zpracování objednávky se podílí řada elementů. Přes řadu výkonných aplikací, integrační platformu, po lidský faktor. Veškerá snaha optimalizovat výkonnost HW platformy, postupné doplňování validačních pravidel zamezujících uživatelům generovat chyby a dokonalejší testování upgrade SW se vzhledem k rozsáhlosti řešení promítá do zkvalitnění procesu vyřizování objednávek zákazníků pomalu. Prioritou firmy je zákazník a co nejrychlejší a nejkvalitnější uspokojení jeho potřeb.

Dalším problémem s tímto spojeným je bezpečnost aplikací. Pro některé z oprav jsou potřeba vysoká oprávnění a jejich rozšíření mezi početnou skupinu sebou nese nemalá bezpečnostní rizika.

Zautomatizování oprav sníží dobu opravy, sníží počet drahé pracovní síly a zabezpečí jasnou transparentnost množství oprav. Dále výrazně sníží zátěž produkčních prostředí, protože bude vykonáváno v nočních hodinách.

Cílem je vytvořit nástroj, který bude, po shromáždění relevantních dat automaticky opravovat chyby objednávek a tím vrátit objednávku do procesu bez nutnosti ručního zásahu s minimálním zpožděním.

Ne všechny chyby lze opravit automatizovaně. Než bude chyba zařazena do opravného nástroje, musí být provedena analýza, stanovení postupu opravy a rozhodnutí, zda lze nebo nelze automatizovat.

Co by měl opravný nástroj vykonávat :

- Provést výběr všech teoreticky poškozených objednávek
- Do jednoho DB schématu doplnit všechny relevantní informace z okolních systémů
- Na základě kombinace těchto informací provést opravu objednávky
- Po opravě zajistit návrat objednávky do procesu
- Vést evidenci co bylo zpracováno

- Evidovat, kterým s modulů byla objednávka opravena
- Evidovat, kolikrát byla objednávka opravována
- Pokud je to více než třikrát, přesunout ji do speciální tabulky pro ruční kontrolu
- Nasbírané informace budou sloužit k podrobnému rozboru poškozených objednávek
- Podle detekovaných informací automaticky doplňovat tyto informace pro reportování
- Musí být uživatelsky příjemný
- Konfigurovatelný bez nutnosti zásahu do kódu programu
- Auditovat svůj běh

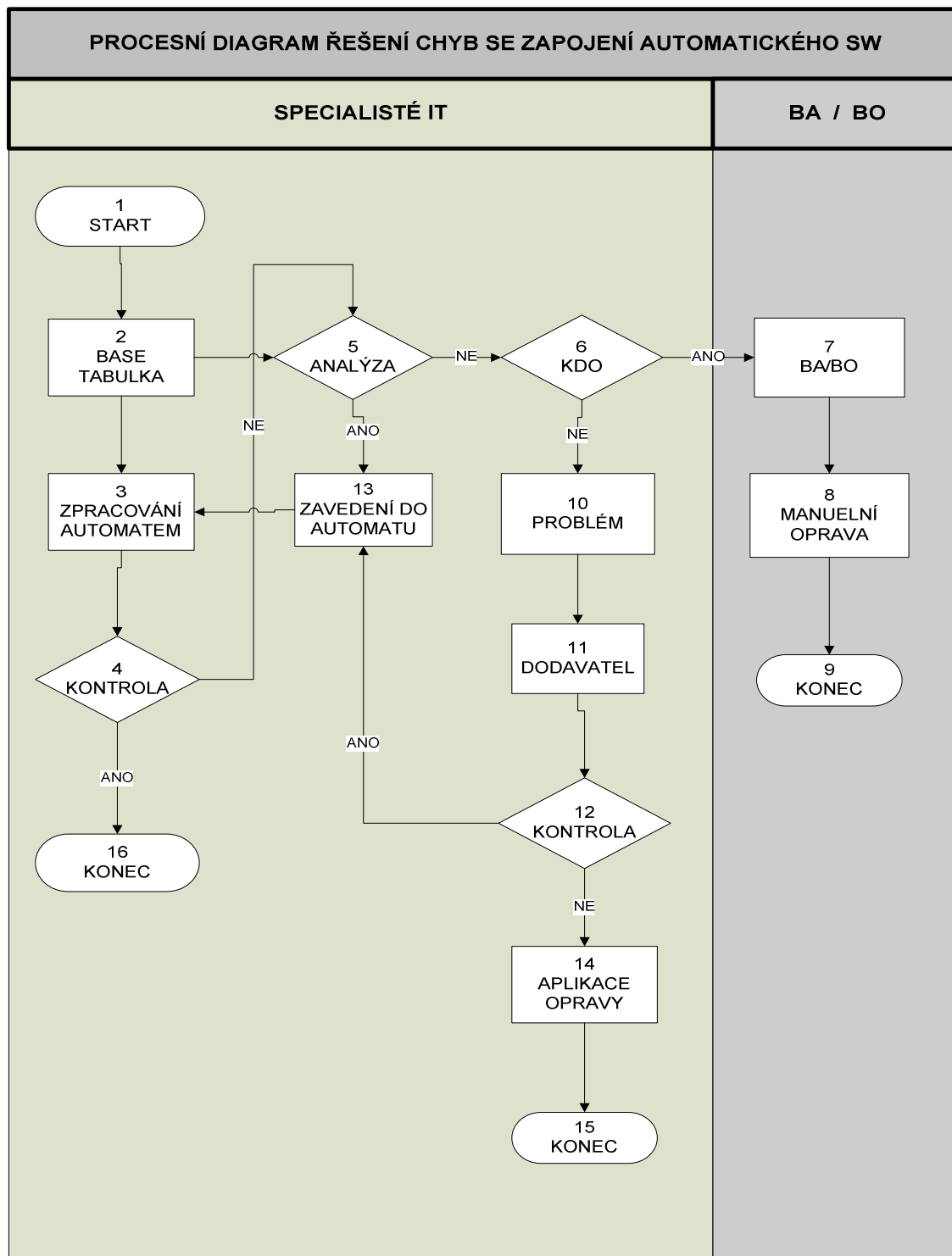
4.1. Zjištěné příčiny chyb

Aplikace CLAUDIA a její integrace do SW prostředí O2 sebou nese řadu potenciálně problémových oblastí. Na základě předchozího studia současného stavu se mi podařilo detekovat tyto příčiny možného přerušení životního cyklu objednávek :

- **datová nekonzistence** - vzhledem k obrovskému množství uložených dat, kde je potřeba zajistit konzistentnost mezi různými databázemi a nejednotností datových modelů a struktur, je tato příčina odstranitelná pouze za cenu obrovského nasazení lidských kapacit. Opravu je nutné provést nad různými aplikačními platformami, což sebou nese zvýšené nároky na vyškolení pracovníků provádějících opravu. Toto vše znamená vysoké náklady.
- **chyby aplikačního SW** - rozsah implementovaného SW je svým rozsahem prakticky nekontrolovatelný. Přestože je použito standardní jádro dodávané přímo výrobcem a zákaznické řešení pro O2 implementuje renomovaný dodavatel, je velmi problematické udržet kvalitu na 100% úrovni. Proto, aby dodávaný SW byl v pořádku, je nutné vytvořit odpovídající testovací prostředí a vyškolit specialisty testování na vysokou odbornou znalost. Toto je prakticky vzhledem k obrovským nákladům neřešitelné.
- **uživatelské chyby** - aplikace CLAUDIA byla implementována s poměrně nízkou kontrolou dodržování uživatelských procesů. Validace jsou prakticky nasazeny na několik základních položek, převážně takových, aby byla zajištěna integrita dat v databázi a naplnění parametrů pro okolní systémy. Nízká úroveň kontroly je implementována hlavně z výkonnostních důvodů. Výkonnost aplikace je nahrazována požadavky na vyškolení personálu tak, aby chyby nedělal. Vzhledem k fluktuaci na frontových kanálech je to naprosto nereálné.
- **chyby HW** - jak je zřejmé z technické architektury viz.kap. 2.2., je aplikace implementována na vysokém počtu HW zařízení. Okolní systémy, které mají vliv na životní cyklus objednávek taktéž disponují nemalým počtem HW, takže výskyt potíží se „železem“ lze očekávat. Řada strategických prvků infrastruktury je duplikována, ale vzhledem k vysokým finančním nákladům tomu tak není na všech prvcích. Nedostatky architektonického řešení jsou postupně, za cenu vysokých nákladů, nahrazovány SW úpravami procesu životního cyklu objednávky.

4.2. Vývojový diagram nového procesu

V této kapitole definuji nový proces při využití opravného nástroje, viz. obr.č.20 s příslušnými činnostmi, tab.č.5 str. 47.



Obr. č.20: Procesní diagram řešení chyb po implementaci automatu

1 Start	Proces je zahájen každý den v nočních hodinách, v době nízké zátěže aplikace
2 Base tabulka	Naplnění vstupní tabulky automatizovaným SW
3 Zpracování automatem	Automatické zpracování objednávek, které jsou po analýze vhodné pro hromadné zpracování
4 Kontrola	Kontrola zpracovaných objednávek. Pokud se oprava nezdařila, jsou objednávky předány k analýze specialistům IT
5 Analýza	Specialisté IT provedou analýzu potenciálně chybných objednávek
6 Kdo	Na základě analýzy specialisté rozhodnou, v čí kompetenci je provedení opravy
7 BA / BO	Pokud se oprava týká obchodních dat, nebo na ni lze aplikovat již definované opravné postupy, jsou objednávky předány na BO / BA
8 Ruční oprava	BO / BA provedou ruční opravu dle definovaných postupů
9 Konec	Zpracování končí
10 Problém	Pokud na opravu neexistují postupy a oprava je mimo rozsah kompetencí specialistů IT je založen v aplikaci Service Desk problém
11 Dodavatel	Dodavatel provede opravu postižených objednávek a dodá opravený SW, který byl příčinou chyby
12 Kontrola	Specialisté IT analýzu řešení a rozhodnou, jestli lze zařadit do automatizované opravy nebo ne
13 Zavedení do automatu	Specialisté IT provedou konfiguraci automatického SW
14 Aplikace opravy	Specialisté IT ručně aplikují poskytnuté řešení od dodavatele
15 Konec	Zpracování končí
16 Konec	Zpracování končí

Tab. č. 5: Popis činností - nově

4.3. Analýza příčin vzniku chyb

Příčiny vzniku chyb, viz. kap. 4.1., se dají rozdělit asi takto :

- Obrovské množství integračních průchodů (WLI), týdně cca 932 000
- Jiné požadavky při vstupu a jiné na výstupy z WLI
- Nedostatečné mapování jednotlivých služeb v návaznosti ve všech systémech
- Uživatelské chyby
- Chyby aplikačního SW CLAUDIA
- Výpadky HW

Všechny výše uvedené příčiny mohou mít za důsledek poškození běhu objednávky. Současný proces oprav, popsáný v kapitole 2.7. je příliš závislý jak na lidských kapacitách, tak i na jejich kvalitě. S tím jsou spojeny náklady na mzdy a školení vysokého počtu specialistů a pomalost řešení. Nyní provedme analýzu chyb v jednotlivých blocích manažerského reportu, tab.č.4.

4.3.1. Žlutý blok

Z teorie uvedené v kap.2.7.1. víme, že chybovými stavy jsou Vraceno – Realizace a Vraceno – Zpoplatnění. Ostatní stavy v tomto bloku sice vyžadují ruční zásah pracovníků BO, ale ten je procesně správně a kapacity jsou na tuto činnost připraveny.

Vraceno – Realizace

Pro opravu tohoto stavu potřebujeme znát stav objednávky v systémech CLAUDIA WLI a OMS. Přes tzv. *kukátka* (pomocné uživatelské nástroje pro nahlížení do těchto systémů) vybereme stavy objednávky v těchto systémech a provedeme opravu. Např. objednávka s chybou : *SPC 14 datum je příliš brzo*. Příčinou chyby je hodnota požadovaného datumu realizace, která je nižší než aktuální den. Systém OMS nemůže např. zapojit linku do provozu s historickým datem. Toto by způsobilo chybné naúčtování hovorného a stálých poplatků za užívání linky. Důvodem této chyby je opožděné odeslání objednávky, ať již z důvodu uživatelské chyby nebo systémového (performance HW) do realizace. Oprava spočívá v nastavení aktuálního datumu do pole Požadované datum, obr.č.6 a přeposlání objednávky z aplikace CLAUDIA. Na této jednoduché chybě je jasně zřejmé, že uživatelský zásah zabere poměrně dost času a není

nijak sofistikovaný. Dle procesního diagramu, obr.č.8, je tuto činnost nutné vykonat kus po kusu nad jednotlivými objednávkami v tomto stavu.

Vráceno – Zpoplatnění

Tento stav je oproti předchozímu jednodušší v tom, že je nutná kontrola pouze v systému CLAUDIA WLI. Požadovaná činnost na objednávce je již fyzicky realizována a problém je pouze při zpoplatnění. Nejčastější příčinou těchto chyb je nekonzistence mandatorních dat na vstupu (CLAUDIA) s požadovanými daty na výstupu (ODSP) pro zpoplatnění. Billingový systém má předepsány, jak zákonnými normami, viz. (10) a (11), tak interními požadavky tvar a obsah vyúčtování za telekomunikační služby a pokud tyto data neobdrží, nelze vystavit zákazníkovi účet. Opravu si opět předvedeme na jednoduchém příkladu. Objednávka je s chybou *Mandatory effective date is null* Tato chyba značí, že na objednávce není vyplněno datum Zpoplatnit od, což je mandatorní pole pro začátek účtování poplatků zákazníkovi. Příčinou je chybný ruční zásah uživatelů. Oprava spočívá v kontrole skutečné realizace požadovaného objednávkou (OMS), doplnění datumů na objednávce a její přeposlání. Vzhledem k dennímu přírůstku průměrně 100 objednávek s touto chybou je nutná lidská kapacita dle kap. 2.7.2. dva pracovníci. Opět je nutno tuto činnost vykonat, dle obr.č.8, ručně nad všemi objednávkami.

4.3.2. Fialový blok

Stav Realizace značí, že na objednávce nejsou ukončeny požadované akce. Problém je ten, že přestože není překročeno *požadované datum*, v řadě případů je objednávka poškozena a potřebuje ruční zásah. Pro výběr takto postižených objednávek je nutné provést každodenní analýzu. Ta spočívá v tom, že se objednávky rozdělí podle produktů a kontrolují se v relevantních systémech. Jedná se o velmi časově náročnou aktivitu, jednak z důvodu provádění sestav ze zdrojové aplikace CLAUDIA (nese sebou i zátěž systému v době nejvyššího využití aplikace uživateli), tak i následné kontrole jednotlivých objednávek ve všech relevantních systémech. Shromáždění informací ze systémů CLAUDIA Siebel, CLAUDIA WLI, OMS a jejich roztřídění v nočních hodinách přinese zásadní úsporu kapacit a zajistí jejich alokování na objednávky, které jsou postiženy.

4.3.3. *Modrý blok*

Vygenerováno – Zpoplatnění

Je stavem, kdy již bylo vše realizováno a objednávka čeká na promítnutí do billingu.

Důvody, proč se tento stav nezmění na konečný, *Dokončeno*, jsou tyto:

- Transakce jsou v chybě - nutná ruční oprava
- Transakce se neprocesují – běží billing, po skončení billingu budou zprocesovány
- Nepřišlo potvrzení o zprocesování transakcí - nutná ruční oprava

Transakce v tomto případě značí datový záznam, uložený v tabulce ODSP.SIBEL_CSM_PROCESSED, sloužící k přenosu dat do billingového systému. V případě tohoto stavu oprava a přeposlání objednávky již není možné, viz. teorie kap.2.4. V současné době je tento stav významně sledovaní oddělení prodejců. Důvodem je změna způsobu vyplácení provizí za prodané služby. Firma přešla od metody výplaty peněz za prodej po uzavření smlouvy se zákazníkem a založení objednávky k metodě výplaty peněz za prodej po uvedení stavu objednávky do stavu *Dokončeno*. Vzhledem k dopadu na mzdy stovek zaměstnanců a malým kapacitám na opravu transakcí, bylo potřeba vyvinout automatizovaný SW. Na opravu jsem tedy vyvinul opravný nástroj, který je uveden v kapitole 4.8.

Pozdrženo – Zpoplatnění

Tento stav je velmi specifický. Objednávky musí mít dodrženu posloupnost zpoplatnění. Jestliže objednávka se starším datem založení není v konečném stavu, obr.č.7, je následující pozastavena v tomto stavu a čeká na dořešení předchozí. Poté dořešení bude automatickým work flow posunuta dále do zpracování. Objednávky se musí ručně zkontrolovat a vyřešit příčina na primární objednávce, nedochází ke zpoplatnění služeb zákazníkovi. Tím, že se část objednávek vyřeší opravným nástrojem, automaticky se i opraví objednávky v tomto stavu, není tedy nutné alokovat na řešení tolik kapacit.

V tomto bloku je taktéž stav Realizace, práce s ním je již popsána v kap. 4.3.2.

4.3.4. Okrový blok

Odesláno – Zpoplatnění je stav, kde by objednávka neměla zůstat déle, než několik minut. Z důvodu chybných ručních zásahů se ale naruší automatické workflow řídící životní cyklus objednávky a je proto nutný zásah na odstranění chyby. Pro opravu je nutné provést kontrolu v systémech CLAUDIA Siebel, CLAUDIA WLI, OMS a ODSP. Z tohoto důvodu je zjišťování všech potřebných informací jednotlivě časově náročné, neefektivní a alokuje zbytečné kapacity.

4.3.5. Červený blok

Objednávky v tomto bloku jsou jednoznačně chybné a je na ně zaměřena primární aktivita oprav specialisty IT. Pro řešení těchto stavů je nutné shromáždit informace ze systému CLAUDIA Siebel, CLAUDIA WLI a OMS. Na základě porovnání stavu objednávky v těchto systémech je provedena ruční oprava. Dohledávání informací jednotlivě a nemožnost slučování chyb do celků sebou nese vysokou časovou náročnost a zvýšenou alokaci kapacit.

4.4. Analýza zdrojů

Z předchozí analýzy 4.2. nám jasně vyplývá nutnost shromáždit všechna relevantní data o vybraných objednávkách do jedné tabulky. Pro řešení jsem použil analýzu potenciálně *zamrzlých* objednávek ze zdrojové systému CLAUDIA SIEBEL, popis systému viz. kap. 2.1. Objednávky a jejich stav jsou primárně ukládány do tabulky Siebel.S_ORDER. Zde jsou uloženy všechny informace, na základě kterých lze provést prvotní, hrubou analýzu a identifikaci objednávek, které je potřeba dále zpracovávat a ke kterým budou doplňovány informace ze systémů CLAUDIA WLI, OMS a ODSP.

Z hodnoty pole STAV tabulky Siebel.S_ORDER lze vybrat dvě hodnoty, Vraceno – Realizace a Vraceno – Zpoplatnění, které, jak je uvedeno v teorii kap.2.5, jsou chybovými stavy a je potřeba je řešit. Pro další analýzu jsou důležité hodnoty pole STAV_CD, kde je podrobněji uveden důvod přerušení standardního cyklu objednávky.

Kombinace hodnot těchto dvou polí umožňuje rozdělit objednávky do skupin podle způsobů řešení :

- ruční zásah pracovníků BO - chybně založená objednávka, kterou lze opravit standardními aplikačními nástroji na úrovni normálních aplikačních uživatelů a nelze použít automatickou opravu
- ruční zásah specialistů první úrovně podpory - oprava vyžaduje již sofistikovanější nástroje aplikace a kontrolu v dalších systémech a nelze použít automatickou opravu
- ruční zásah specialistů druhé úrovně podpory – závažnější opravy s nutností zásahu do okolních systémů a nelze použít automatickou opravu

Aplikace CLAUDIA SIEBEL již další relevantní informace nemá a musíme je zjišťovat z dalších navazujících systémů. Dalším systémem obsahujícím informace pro řešení problémů je aplikace CLAUDIA WLI. Identifikace objednávky v tomto systému je možná podle hodnoty sloupce ORDER_ID z tabulky Siebel.S_ORDER zdrojového systému. Proces pro objednávku je evidován v tabulce CLAUDIA_PROCESS_INSTANCE, kde je pro potřeby automatického SW relevantní hodnota sloupce STAV. Ta může nabývat hodnot :

- null – proces neexistuje
- in progress – proces běží
- initial – proces se inicializuje
- completed – proces byl ukončen

Tato hodnota má zásadní vliv na zařazení objednávky do automatizované opravy. Do oprav nesmí vstoupit objednávky s hodnotou *completed*. V tomto případě je nutná vizuální kontrola objednávky ve všech systémech a prověření, jaké aktivity byly systémy provedeny, protože tento stav značí, že vše požadované v objednávce bylo vykonáno. Pro řešení ale není nutné vyřazovat tyto objednávky dále z procesu zjišťování dalších informací z ostatních systémů.

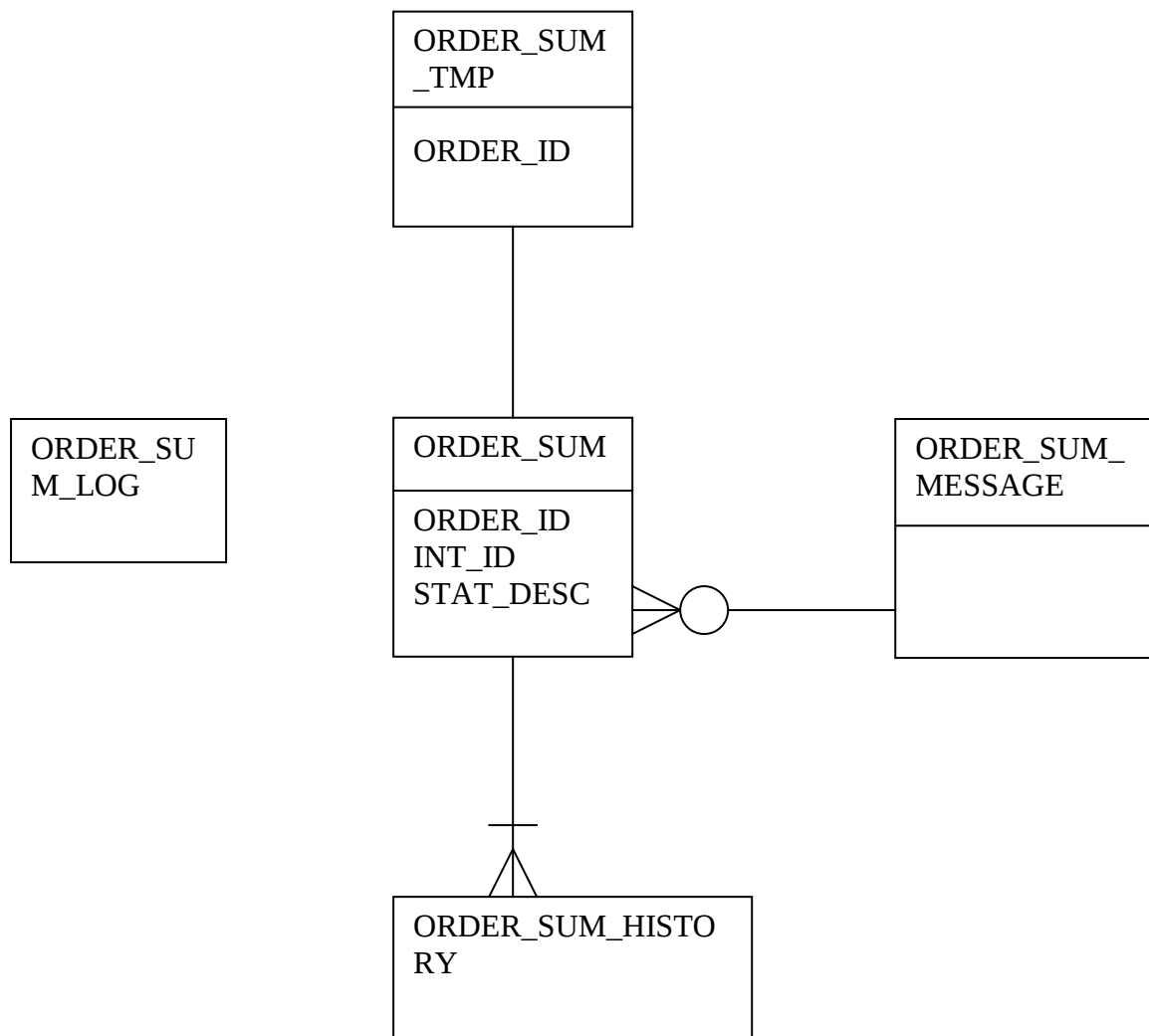
Dalším zdrojem informací nutných k opravám je aplikace OMS. Vazební položkou je hodnota sloupce INTEGRATION_ID z tabulky S_ORDER_ITEM. Dotazy dle zadaných požadavků dodali specialisté aplikace OMS.

4.5. Vytvoření DB struktury

Pro prvotní analýzu chyb je potřeba shromáždit a spojit data z různých databází. Z důvodu nutnosti bezpečnosti zdrojového informačního systému a oddělení databázové struktury od zdrojové je nutné vytvořit nové DB schéma viz. (6), pod kterým bude navedena jak tabulková struktura, tak vlastní SW pro obsluhu. Pro plnění daty budou použity standardní DB linky aplikace CLAUDIA do zdrojových databází. Data budou plněna do base tabulky, která bude sloužit pro všechny ostatní činnosti, analýzu chyb, opravu objednávek a reporting. Jedná se o specializovaný datový sklad, jež obsahuje pouze jednu tabulku faktů s několika pomocnými tabulkami, sloužícími k obsluze SW na automatizaci oprav chyb.

4.5.1. Entito – relační model

Ve schématu je uveden název tabulky a primární nebo cizí klíče. Celá struktura tabulek je popsána v přílohách č.4 až 8.



Obr. č.21: ER diagram

Schéma se skládá z těchto tabulek :

ORDER_SUM

Základní tabulka, do které jsou uložena všechna potřebná data pro požadovanou funkcionalitu opravného nástroje dle úvodu kapitoly 4.

ORDER_SUM_HISTORY

Tabulka, do které se přesouvají každodenně data s tabulky ORDER_SUM. Důvodem je nezaplňovat tabulku faktů historickými daty. Dále slouží k zjištění, jestli byla objednávka již zpracována nebo ne.

ORDER_SUM_LOG

Tabulka, do které se audituje běh opravného SW.

ORDER_SUM_MESSAGE

Tabulka řídící vstupy pro procedury , které přeposílají objednávky. Sloupec MESSAGE je primárním klíčem pro tabulku ORDER_SUM, sloupec STAT_DESC.

ORDER_SUM_TMP

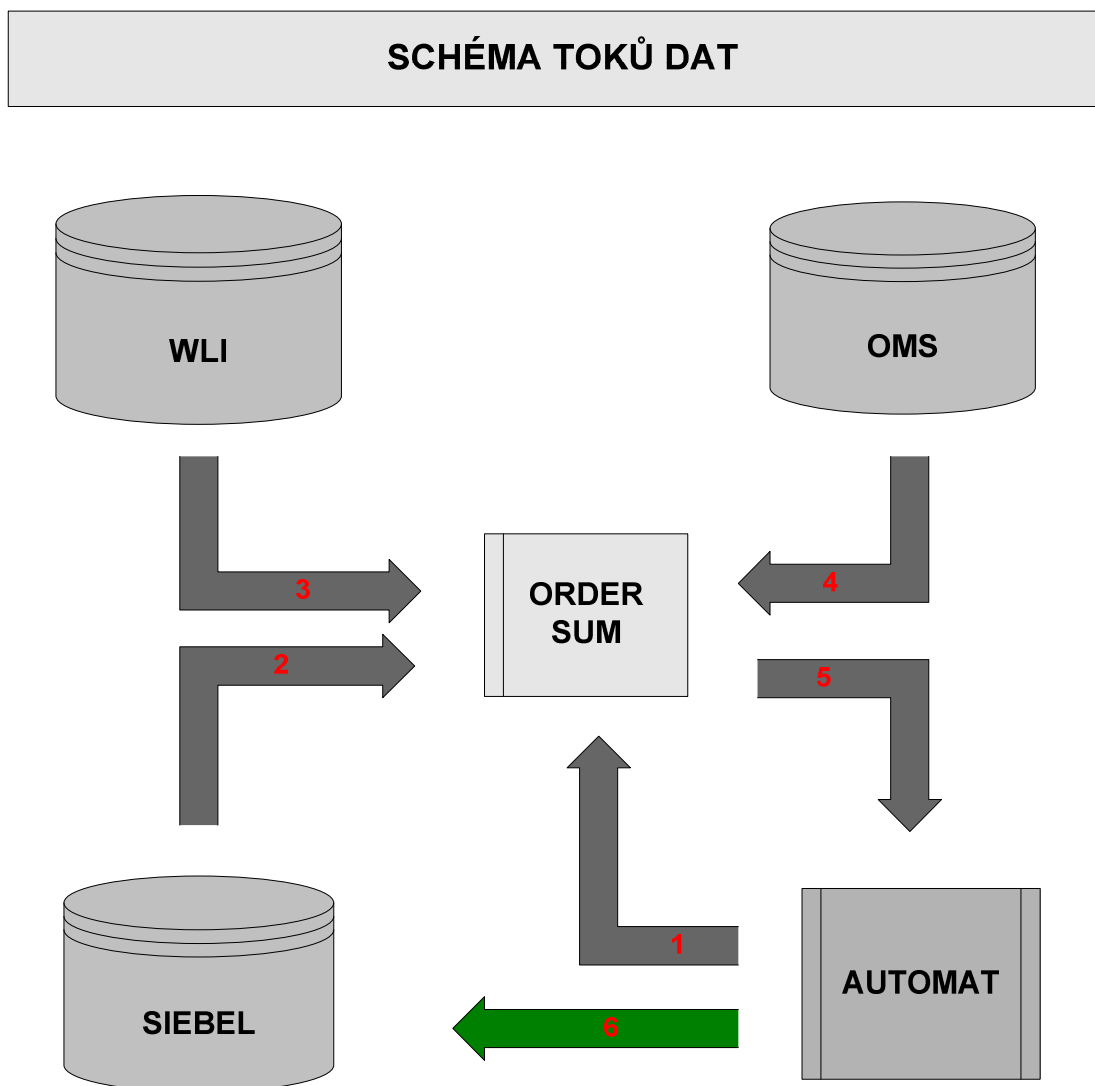
Do tabulky jsou ukládány objednávky, které prošly třikrát cyklem přeposlání a chyba se neodstranila.

4.6. Opravný nástroj

V této chvíli máme jak teoretické znalosti, viz kap.2 a 4.3., tak připravené databázové schéma. Vzhledem k délce kódu opravného SW, budu u jednotlivých částí uvádět analytické informace, celý kód programu je na přiloženém CD. Datový sklad ORDER_SUM je nakonfigurován i pro další činnosti, které jsou nad rámec této práce. Technologie je prakticky použitelná při jakémkoli rozšíření produktů a infrastruktury a není možné v rámci této práce, vše postihnout.

4.6.1. Schéma toků dat

Pro přehlednost nejdříve grafické znázornění :



Obr.č.22 : Schéma toků dat

Data nejsou nahrávána ETL nástroji, ale klasickým SQL insertem a to v několika krocích. Popíšme si nyní jeden tok za druhým.

4.6.2. Tok číslo 1, automat → order sum

Pomocí Oracle DB jobu, viz (7) je každý den, ve 2:30 spuštěn kompletní opravný SW (automat). V první etapě se provede přesun dat z minulého dne z tabulky ORDER_SUM do tabulky ORDER_SUM_HISTORY. Opis části kódu je uveden v příloze č.9.

4.6.3. Tok číslo 2, siebel → order sum

Jedná se o základní naplnění daty z produkčního prostředí aplikace Siebel do pomocného schématu. Provádí se pomocí procedur **Fill_order_sum** a **Fill_order_sum_VZ**. Do tabulky ORDER_SUM jsou vybrány objednávky v těch stavech, které indikují možnost přerušení standardního běhu objednávky, viz teorie kap.2 a 4.3. Těmi jsou :

- Odesláno – tehdy, kdy požadované datum zřízení služby je < sysdate-1. Podmínka je takto stanovena z důvodu nedodržení požadovaného data zřízení služby zákazníkem a objednávka by neměla v tomto stavu s vyexpirovaným datem existovat.
- Realizace – tehdy, kdy požadované datum zřízení služby je < sysdate-3. Tři dny jsou firmou tolerovány, protože může dojít ke zpoždění na straně zákazníka nebo opožděnému zavedení provedení práce do systémů O2, ale zákazník službu již využívá.

U níže uvedených stavů se data aktualizují již pouze s omezením dotazu na hodnotu stavu objednávky, protože jsou všechny ve stavu, kdy je potřeba je řešit bezprostředně:

- Vraceno – Realizace
- Vraceno – Zpoplatnění
- Odesláno – Storno
- Odesláno – Rušení
- Odesláno – Realizace

Tabulka je plněna i stavy objednávek, které nevstupují do automatické opravy. Důvodem je zvýšení efektivity při analýze chyb, popsané v kap. 4.3.

4.6.4. Tok číslo 3, wli → order sum

V tomto okamžiku už je naplněna tabulka ORDER_SUM relevantními objednávkami a začíná doplňování potřebnými údaji z ostatních systémů, nutnými pro analýzu a opravu. Použita je procedura **Update_WLI**, kterou se dle primárního klíče ORDER_ID aktualizují příslušné sloupce tabulky hodnotami z tabulek systému WLI.

4.6.5. Tok číslo 4, SPC → order sum

Za použití procedury **Update_SPC** se pro určité stavy objednávek provede update příslušných sloupců tabulky ORDER_SUM hodnotami z aplikace OMS. Protože tento výběr značně zatěžuje databázi OMS, je prováděn pouze pro stavy, kde je tato informace relevantní pro řešení. Z předchozí teorie, kap. 4.3. víme, že se jedná o stavy:

- Odesláno
- Realizace
- Vraceno – realizace
- Odesláno – storno
- Odesláno – rušení
- Odesláno – realizace

4.6.6. Tok číslo 5, order sum → automat

V této chvíli je již tabulka ORDER_SUM naplněna všemi důležitými informacemi pro spuštění vlastní automatické opravy objednávek. Relevantní objednávky jsou postupně automatem zpracovány, za použití procedur **Auto_send**, **Send_Msgs**, **Send_SPC14**, **Send_Attrib47**, **Auto_send_initial**. Procedury provádí zápis do databáze siebel, takže si je popíšeme v následujícím bodě.

4.6.7. Tok číslo 6, automat → Siebel

V této kapitole si popíšeme jednotlivé procedury. Všechny jsou vytvořeny na základě provedené analýzy a je otestováno, že všechny objednávky spadající do opravy těmito procedurami bez problému proběhnou svým životním cyklem.

Auto_send

Na základě analýzy objednávek popsané v kap. 4.3. ve stav Vráceno – Realizace byla identifikována skupina objednávek se specifickými chybami aplikace OMS. Objednávky jsou vybrány z tabulky ORDER_SUM a nad nimi jsou prováděny aktivity v produkční databázi Siebel. Přeposlání objednávky je komplexní úkon, který se obecně skládá z těchto částí :

- Update položek objednávky v tabulce Siebel.S_ORDER_ITEM
- Update hlavičky objednávky v tabulce Siebel.S_ORDER
- Update inbound sekvence v tabulce Siebel.S_ORDER_X

Tato procedura zpracovává takové objednávky, kde je potřeba doplňujících a upřesňujících podmínek z aplikace OMS. Opis procedury je v příloze č. 10.

Send_Msgs

Tato procedura ve spojení s následující je jádrem automatických oprav. Slouží jako vstup pro proceduru **Send_attrib47**. Po provedení analýzy dle kap. 4.3. umožňuje opakované přeposlání objednávek v nočních hodinách dle seznamu chybových hlášení uložených v tabulce ORDER_SUM_MESSAGE, popis viz příloha č.7.

Specialista provede analýzu a identifikuje chybové hlášení z pole STAT_DESC tabulky ORDER_SUM jako patřící do automatické opravy. Provede zápis hodnoty tohoto pole do sloupce MESSAGE tabulky ORDER_SUM_MESSAGE, nastaví hodnotu pole FLAG_ACTIVE na :

- Y – pokud má procedura aktuálně chybu zpracovávat
- N – pokud se chyba dočasně nevyskytuje a tudíž není relevantní, aby tyto hodnoty vstupovaly do zpracování

Následně nastaví hodnotu pole FLAG_RZ na :

- Z - pokud se jedná o objednávky ve stavu Vráceno – Zpoplatnění
- R – pokud se jedná o objednávky ve stavu Vráceno – Realizace

Send_attrib47

Vstupem jsou objednávky odpovídající stavu v tabulce ORDER_SUM_MESSAGE. Podle hodnoty pole FLAG_RZ jsou nastaveny stavy objednávky v tabulkách Siebel.S_ORDER a Siebel.S_ORDER_ITEM , tj. Odesláno v případě hodnoty pole R a Odesláno – Zpoplatnění v případě hodnoty pole Z. Toto odpovídá životnímu cyklu objednávky uvedenému na obr. č. 7.

Send_SPC14

Procedura je řešením vzorové analýzy uvedené v kap. 4.3.1. Není zařazena do procedury **Send_Msgs**, protože musí pracovat v podmínce výběru se specifickými stavy v aplikaci OMS.

Auto_send_initial

Speciální procedura pro objednávky ve stavu v aplikaci CLAUDIA Siebel Odesláno a v aplikaci CLAUDIA WLI initial. V kap. 4.3. je uvedeno, že tento stav značí, že se objednávka inicializuje v aplikaci CLAUDIA WLI . Vlastní inicializace (spuštění procesu dle kap. 2.7.) musí proběhnout během minuty. Pokud tomu tak není, objednávka se neodeslala do provisioning systémů a je nutno proces inicializace opakovat. Postup je následující :

- Vymazat záznam z tabulky IOM_WLI.CLAUDIA_PROCESS_INSTANCE
- Změnit stav objednávky v CLAUDIA SIEBEL z Odesláno na Vrácono – Realizace, to z toho důvodu, že přeposlání objednávky lze pouze z tohoto stavu
- Následně již standardní přeposlání

Na této proceduře je názorně vidět, že plnění tabulky ORDER_SUM na první pohled korektním stavem Odesláno je relevantní. Jedná se o skrytou chybu neodhalitelnou na uživatelské úrovni, k její identifikaci jsou nezbytně nutné informace ze systému CLAUDIA WLI.

4.7. Další funkce nástroje

Opravný nástroj nejenom zabezpečuje přeposlání poškozených objednávek, ale jak bylo požadováno v úvodu kap.4, slouží jako datový sklad pro efektivnější analýzu objednávek a možnost jejich přehledného reportování v podrobnějším rozsahu, než je manažerský výstup, viz. tab.č.3. Nástroj je dále využíván ke spouštění procedur, které přímo nesouvisí s opravou objednávek, ale je potřeba je vykonávat v nočních hodinách a je zbytečné vytvářet kopii master procedury.

4.7.1. Data mining

Do této kapitoly spadají procedury, které v tabulce ORDER_SUM aktualizují příslušná pole tak, aby třídění objednávek bylo efektivní.

Mark_term

Procedura aktualizuje pole RUN_COMMENT hodnotou „termín“ na základě zjištění, že v tabulce Siebel.S_EVT_ACT (obrázek č.5, entita Realizační aktivita) se vykazuje aktivita „termín instalace“ v neukončeném stavu. Toto značí, že je potřeba kontaktovat zákazníka a není tedy nutno s objednávkou pracovat jako v chybě.

Mark_ebill

Procedura aktualizuje pole RUN_COMMENT hodnotou „ebill“ na základě identifikace tohoto produktu v tabulce ORDER_SUM v poli PROD_CODE hodnotou „PR0019“. Pro produkt Ebill nelze vycházet při řešení z hodnoty v poli „požadované datum“, protože tento produkt je aktivován prvním přihlášením zákazníka. A to může být i s velkým zpožděním, takže se objednávky s tímto produktem neanalyzují.

Mark_RP

Procedura aktualizuje pole RUN_COMMENT hodnotou „RP“ na základě vyplnění pole RP_APS_DATE v tabulce ORDER_SUM. Pole RP_APS_DATE je plněno z jedné z aplikací OMS, RP (řízení prací). Hodnota značí, že je se zákazníkem nasmlouván

termín realizace a ten opět může být zpožděn oproti „požadovanému datu“. Objednávky jsou tudíž pro analýzu chyb nerelevantní, ale bez tohoto označení by se musely zpracovávat.

Update_TK_LLU

Procedura aktualizuje pole run_comment hodnotou „TK“ pokud je na objednávce produkt Telekonto a hodnotou „LLU“, pokud se jedná o objednávku na zřízení sdíleného přístupu k vedení v tabulce ORDER_SUM. U těchto produktů je problém v tom, že na hlavičce objednávky tyto produkty nejsou vidět, a tak je potřeba provést kontrolu přes položky objednávky viz. teorie kap.2.4.1. Podrobnější rozbor je nad rámec této práce.

4.7.2. Fixace

V této kapitole jsou uvedeny procedury, které slouží k hromadné opravě dat v tabulkách aplikace CLAUDIA Siebel. Všechny níže uvedené procedury slouží k odstranění administrativních chyb a tím snížení počtu nutných kontrol uživateli v aplikaci CLAUDIA Siebel. Jedná se důležitou součástí nástroje, ale protože se zde nepracuje přímo s opravou chyb objednávek, uvedu u každé procedury pouze krátké vysvětlení. V případě nejasností mohu podat vysvětlení při obhajobě diplomové práce.

Fix_S_Evt_Act

Procedura nastavuje stav realizační aktivity na Hotovo u objednávek v aplikaci CLAUDIA Siebel, které jsou v koncových stavech, obr.č.7. Jedná se o kosmetickou úpravu v datech, která způsobuje pouze administrativní problémy koncovým uživatelům. Nicméně má dopad na efektivitu jejich práce a je tedy žádoucí tyto nesrovnalosti opravit.

Fix_S_Order

Procedura slouží k automatickému stornování objednávek, které jsou chybně založeny. Provádí se zde kontrola, jestli má objednávka ve stav Založeno položky, což je podmínka korektnosti, viz. 2.4.1.

Park_Order

Tato procedura kontroluje objednávky ve stavu Založeno a pokud existují položky objednávky, nastaví, po čtyřech dnech od data vytvoření, objednávky do stavu Zaparkováno.

4.8. Nástroj pro opravu transakcí

V této podkapitole je uvedena procedura, která kontroluje, porovnává a opravuje objednávky v aplikaci CLAUDIA a billingové transakce v DB ODSP v návaznosti na účtování, viz kapitola 4.3.3. Pojem transakce v tomto případě znamená datový tok mezi aplikacemi ve formě tabulkového řádku s přesně stanovenou datovou strukturou. Oprava spočívá v provedení kontroly transakcí vázaných na jednotlivé objednávky a následnému nastavení konečného stavu pro objednávky, které splňují dané podmínky. Tím dojde ke snížení počtu objednávek pro ruční kontrolu a opravu.

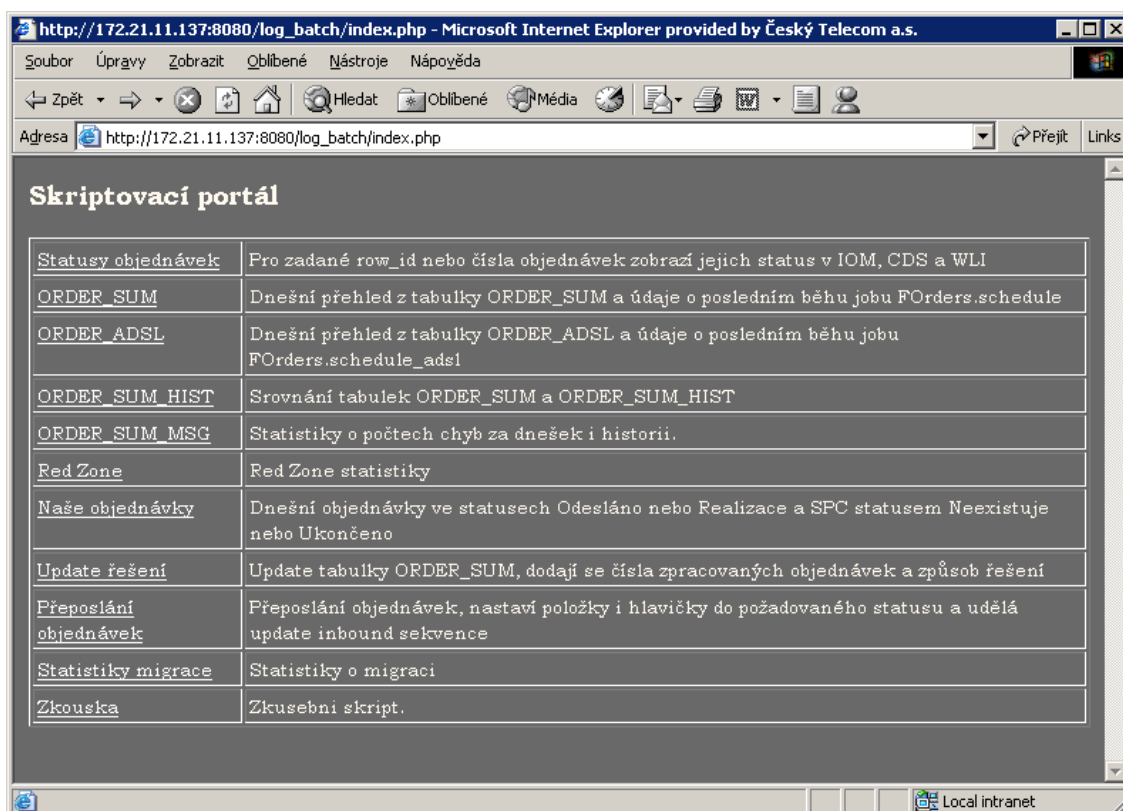
Update_trans

Procedura nejdříve vytváří dočasné tabulky, následně dle v nich uvedených informací provádí update relevantních transakcí v DB ODSP a objednávek v DN CLAUDIA. Výběr objednávek je omezen na starší než deset dní. Důvodem je možnost jetí billingového programu, při kterém je procesování transakcí zastaveno. Vlastní oprava se provádí pouze pro objednávky, které mají transakce výhradně v množině stavů {S,C,X}. Objednávky s transakcemi v jiném stavu musí být zkontrolovány ručně. Procedura je spouštěna denně v noci pomocí oracle jobu. Její opis je uveden v příloze č. 11.

4.9. Uživatelské rozhraní pro analýzu

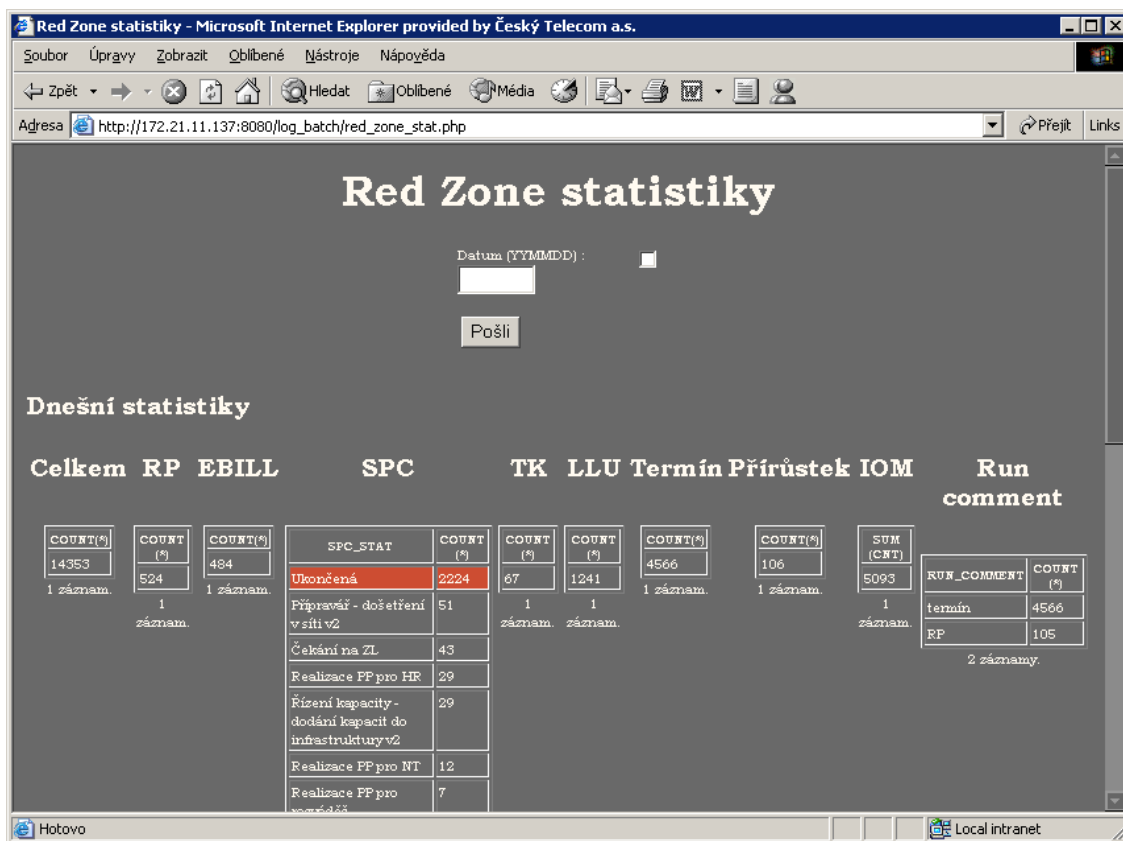
Přestože se předpokládá, že s nástrojem pracují dostatečně vyškolení pracovníci, pro rychlost reportování je vytvořeno uživatelské webové rozhraní, za použití (2). Přes toto rozhraní lze i jednorázově využít přeposílací proceduru **auto_send**. Pokud je z analýzy zjištěno, že se některé objednávky dají touto procedurou přeposlat a nelze, např. z důvodu urychlení směrem k zákazníkovi, čekat na noční zpracování, vybrané objednávky z ORDER_SUM se vloží do webového okna a jsou automaticky zpracovány ihned. Pro kontrolu zpracování těchto objednávek již ale nelze použít stavy v tabulce ORDER_SUM, ty se aktualizují v noci a přeposláním se mění. V kontrolní části portálu jsou tak použity výběry z nočního plnění, ale přímo směřované do příslušných aplikací (CLAUDIA WLI, OMS, ODSP).

Na obr.č.23 je úvodní stránka portálu. Portál obsahuje všechny potřebné aktivity pro analýzu, řešení a reportování.



Obr.č.23: Skriptovací portal

Na obr.č. 24 je ukázka obrazovky, kde je podrobně analyzována tzv. Red zóna, ta odpovídá červenému bloku z kap. 2.7.1.



Obr.č.24 : Analýza červeného bloku

Na uvedeném obrázku jsou např. vidět výsledky jednotlivých markovacích procedur z kap. 4.7.1. a to v polích TK, LLU, Termín atd. Vytvoření tohoto portálu výrazně zefektivnilo práci specialistů IT. Podrobný popis tohoto portálu je bohužel již nad rámec této diplomové práce.

4.10. Přínosy řešení

V této kapitole vyhodnotím uplatnění navrženého řešení nad jednotlivými oblastmi příčin chyb.

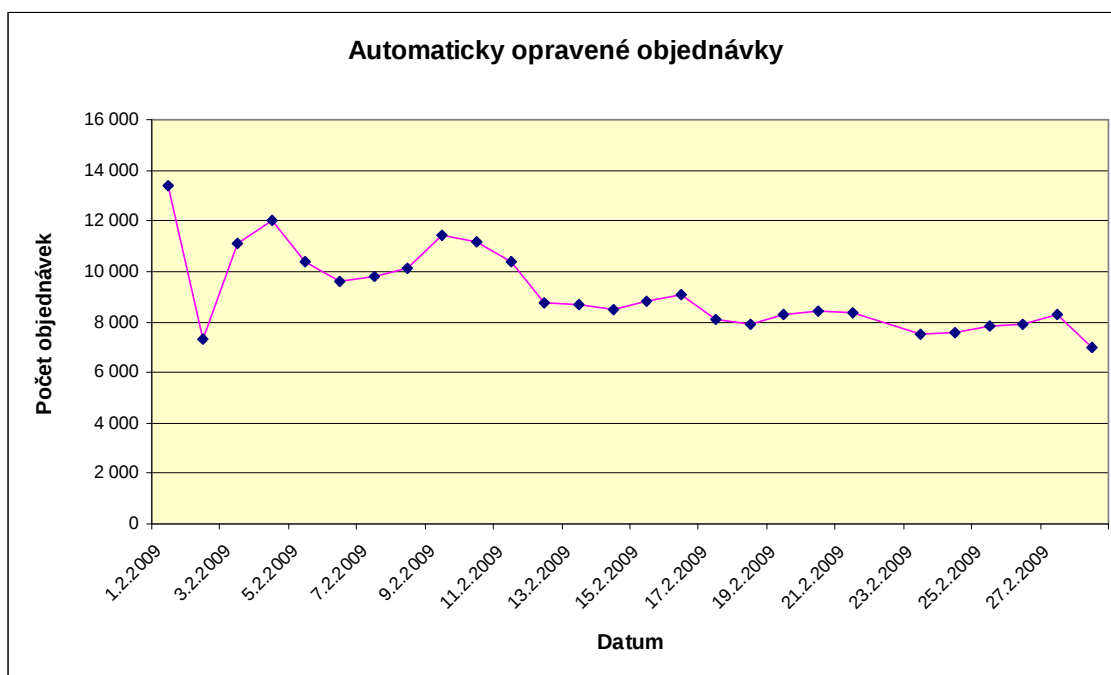
- **datové nekonzistence** – v této oblasti má opravný nástroj vysoké uplatnění. Problémy v datech jsou převážně detekovány reaktivně, dle chybných stavů objednávek. Ke kompletní rychlé analýze výborně slouží tabulka ORDER_SUM, která následně poslouží i pro proaktivní vyhledání všech postihnutých dat a jejich následné zpracování, buď jednorázově za použití procedury **auto_send** přes webové rozhraní kap. 4.9, nebo při trvalejším výskytu chyby, do jejího odstranění dodavatelem, zařazením do automatického každodenního nočního zpracování, kap. 4.6.
- **chyby aplikačního SW** – v případě chyb z tohoto důvodu je využití v nižší míře. Chyby jsou poměrně brzo identifikovány a opraveny, takže využití je většinou nad malou množinou poškozených objednávek. Jsou ovšem také případy, kdy se jedná o chybu nad velmi využívaným produktem a zde může SW pomoci i stovkám objednávek týdně. Jako v předchozím případě se jedná o využití jak ručního, tak automatizovaného přeposlání objednávek, kap. 4.6. a 4.9.
- **uživatelské chyby** – tato příčina je prakticky navrženým řešením nepostižitelná. Každý uživatel dělá různé individuální chyby převážně datového charakteru nebo nedodržením metodických postupů. Následná oprava se musí provádět dle typu chyby nad jednotlivými objednávkami a automatizace je tudíž v tomto případě nepoužitelná. Lze využít tabulku ORDER_SUM pro analýzu stavu objednávky.
- **chyby HW** – tato oblast je přímo určená pro automatizaci oprav. Z logiky řešení aplikace vyplývá, že se vždy jedná o skupiny objednávek stejného typu. Zde lze aplikovat hromadnou analýzu a následné navedení dle chybové hlášení do automatického zpracování. Vzhledem k tomu, že se jedná o typizované chyby, lze je přeposílat dříve, než vůbec kdokoli zjistí jejich poškození a to i v budoucnosti. Navržené řešení je v tomto případě jedinečnou formou zabezpečení zrychlení oprav.

4.11. Ekonomické zhodnocení

Základem ekonomické zhodnocení je stanovení nákladů na zpracování jedné poškozené objednávky. Po složitém pátrání se mi podařilo zjistit, že jednotková kalkulace čítá 50 Kč/ objednávka. Nepodařilo se mi zjistit, jak byla tato hodnota dosažena, nicméně lze konstatovat, že se na ní bezpochybně podílí tyto složky:

- Zpoždění zahájení vyúčtování
- Kapacity pro zpracování
- Nespokojenost zákazníka

První z oblastí ekonomického přínosu je část automatického přeposílání pomocí navrženého SW. Připomeňme si, že probíhá každou noc dle analyzovaných chyb uvedených v číselníku. Na obrázku č. 25 je uvedeno grafické znázornění zpracování objednávek za měsíc únor letošního roku. Na vysokém počtu denně opravených objednávek se vysokou měrou podepsal upgrade systému OMS (proběhl 31.1.2009), který vykazuje obrovskou chybovost a nestálost. V takovýchto případech je SW z této diplomové práce neocenitelný.



Obr.č.25 : Automaticky opravené objednávky

Druhým, neméně významným ekonomickým přínosem SW, je příprava informací o poškozených objednávkách pro další analýzu. Na základě těchto informací se velice efektivně dařilo a daří detekovat nové typy chyb vzniklých upgradem okolních systémů. Po detekci jsou poškozené objednávky s danou chybou nejdříve ručně přeposlány pomocí portálu popsaného v kapitole 4.8 a následně, po ověření správnosti opravy, je chyba zařazena do čísleníku, viz. příloha č. 12. Za měsíc únor tak bylo zpracováno, jak vyplývá z výkazu aktivit specialistů, 9 309 objednávek.

Jednoduchým výpočtem, kdy sečteme objednávky opravené automaticky, s objednávkami opravenými ručně, a to vše vynásobíme náklady na jednici, se dopracujeme k výslednému číslu, tj. 12 853 250 Kč ušetřených nákladů za měsíc únor. Na první pohled se může zdát, že je to obrovská suma, nicméně je nutné si uvědomit, že je např. vyčíslena kalkulace nedostupnosti systému CRM na 1 milion Kč/ hodinu odstávky. Nebo že denní průměrný počet objednávek je okolo 30 000.

Dále může být udivující vysoké procento neúspěšných objednávek. To je způsobeno upgrade systému OMS a jeho problémy. V době, kdy jsou systémy stabilizovány a neprovádí se SW upgrade je denní počet objednávek možných automaticky zpracovat tímto nástrojem okolo 500. Systém Claudia je výrazně ovlivňován problémy všech okolních systému. Chyby v procesu objednávek v těchto systémech se přenáší do Claudie. Tímto je demonstrována důležitost tohoto nástroje a jeho vynikající vlastnosti.

Závěr

Cíl této diplomové práce byl jednoznačně splněn. Vyvinutý SW výrazně zkrátil dobu realizace řady objednávek a umožňuje efektivní analýzu problémů. Je aplikován do velice složitého integračního prostředí s rozsáhlým množstvím zákaznických dat a široké škály kombinací produktů. Přesto jeho funkcionalita založená na, v tomto okamžiku si dovoluji vyzdvihnout způsob řešení, parametrizaci přes pomocnou tabulku, je výborným a jednoduchým řešením, které je neustále využíváno. V současné době se jedná o 57 aktivních chybových hlášení uvedených společně s dočasně neaktivními v příloze č.12. V kapitole 4.11. je uvedeno ekonomické zhodnocení za měsíc únor tohoto roku. Jednoznačně z něho vyplývá obrovský přínos pro firmu, která by musela absenci tohoto SW řešit buď navýšením lidských kapacit nebo vytvořením SW dodavatelskou firmou. Vždy by se jednalo o zbytečné, vysoké náklady v řádech milionů měsíčně, které nástroj tímto firmě ušetřil. Využití **auto_send** procedury přes webové rozhraní je účelnou pomůckou pro každodenní řešení problémů s objednávkami a je důkazem komplexnosti řešení. Ve neposlední řadě se v současné době ukazuje nadčasovost řešení v tom, že v případě problémů okolních systémů je tento SW schopen absorbovat a hlavně opravit tisíce poškozených objednávek s minimem nákladů a s minimálním dopadem na poskytování služeb zákazníkům.

Analytické řešení uvedeného SW slouží jako podklad pro řešení problémů u dalších produktů z nabídkového portfolia. Důležitým přínosem je taktéž vytvoření datového skladu, které umožňuje v jednom pohledu zjistit vše potřebné k analýze problému a šetří to nejdražší, čas.

Seznam použité literatury a všech informačních zdrojů

A) Knihy

- 1) BERKA, Petr. *Dobývání znalostí z databází*. Vyd. 1. Praha: Academia, 2003. 366 s. ISBN 80-200-1062-9.
- 2) CASTAGNETTO Jesus a RAWAT Harish a SCHUMANN Sasha aj. *Programujeme PHP profesionálně*
2.vydání Praha: Computer Press 2002 656 str. ISBN 80-7226310-2
- 3) FAYYAD, Usama a PIATETSKY-SHAPIRO, Gregory a SMYTH, Padhraic. *From data mining to knowledge discovery in databases*. American Association of Artificial Intelligence, 1996
- 4) KUČERA, Milan. Monitorování data warehouse. *Data security management : časopis o bezpečnosti, správě a řízení rizik informačních systémů*. 2001. Roč. 5, č. 4, s 44-45
- 5) KYJONKA, Vladimír. Jak začít s prvním datovým skladem. *IT systém*. 2004, č.3, s 36 – 37
- 6) KYTE, Thomas. *Oracle, návrh a tvorba aplikací* . 1.vyd Brno: CP Books a.s. 2005. 694s. ISBN 80-251-0569-5
- 7) LONEY, Kevin a THERIAULT, Marlene. *Mistrovství v Oracle, Kompletní průvodce tvorbou, správou a údržbou databází*.
1. vydání Praha : Computer press 2002 860 str. ISBN 80-7226-635-7
- 8) SCHILLER, Martin. Kvalitní data znamenají kvalitní rozhodování. *IT systém*. 2004, č. 3, s. 38-39
- 9) SKOPAL, Vít. Neimplementujte datový sklad. *IT systém*. 2004, č.3., s. 47.

B) Zákony a vládní vyhlášky

- 10) Zákon č. 127/2005 Sb. o elektronických komunikacích, ve znění pozdějších předpisů
- 11) Zákon č. 151/2000 Sb., o telekomunikacích, ve znění pozdějších předpisů

C) Firemní materiály

- 12) AP350_Functional_Design_Order_Magmt.doc
- 13) Integration_CookBook_v2.2.doc

- 14) Provozní dokumentace OMS v1.0.doc
- 15) TR435_Skolici_Prirucka_Zaklady_objednavky.doc
- 16) TA433 – ODSP Oracle streams

Seznam zkratek

BA/BO	Business Administrátor/Back Office
ODSP	Ordering Data Store Production
DB	Databáze
EAI	Enterprise Application Integration
ESB	Enterprise Service Bus
GUI	Graphical User Interface
HTS	Hlavní Telefonní Stanice
IB	Integrační Broker
IBTUX	Integrační Broker – část Tuxedo
IBWLI	Integrační Broker – část WebLogic Integration
CLAUDIA	Integrated Order Managment
ITAC	IT Architecture Committee
J2EE	Java 2 Platform, Enterprise Edition
JMS	Java Message Service
KDD	Knowledge discovery in databases
OMS	Order Management Systém
O2	Telefónica O2 Czech Republic, a.s.
RMI	Remote Method Invocation
RPC	Remote Procedure Call
SOA	Service Oriented Architecture
SW	Software
VPN	Virtual Private Network
WLI	BEA WebLogic Integration
WLS	BEA WebLogic Server

WS	Web Service
WTC	WebLogic Tuxedo Connector
XML	eXtensible Markup Language

Seznam příloh

Příloha č.1: Popis objektů technické architektury

Příloha č.2: Přehled změn stavů objednávek

Příloha č.3: Změny stavů objednávek v závislosti na jejich stavu

Příloha č.4: Struktura tabulky ORDER_SUM

Příloha č.5: Struktura tabulky ORDER_SUM_HISTORY

Příloha č.6: Struktura tabulky ORDER_SUM_LOG

Příloha č.7: Struktura tabulky ORDER_SUM_MESSAGE

Příloha č.8: Struktura tabulky ORDER_SUM_TMP

Příloha č.9: Opis spouštěcí části kódu

Příloha č.10: Procedura Auto_send

Příloha č.11: Procedura Update_trans

Příloha č.12: Opis tabulky ORDER_SUM_MESSAGE

Přílohy

Příloha č.1: Popis objektů technické architektury

ID	HW specification	Function/Component
WSPRD01	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Internal Web server
WSPRD02	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Internal Web server
Siebel Database	HP SD64 IA64 16 CPU 1.6Ghz, 64 GB RAM	Siebel database server
ODS Database	HP SD64 IA64 16 CPU 1.6Ghz, 64 GB RAM	ODS database server
WLI Database	HP SD64 IA64 4 CPU 1.6Ghz, 16 GB RAM	WLI database server
SBPRD01	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD02	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD03	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD04	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD05	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD06	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD07	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD08	IBM p550 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD09	IBM p550 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD10	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD11	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD12	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD13	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD14	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD15	IBM p520 server 4 CPU/4 core POWER6 4.2Ghz, 32 GB RAM	Siebel application server
SBPRD16	IBM p520 server 4 CPU/4 core	Siebel application

	POWER6 4.2Ghz, 32 GB RAM	server
LDPSRD01	Linux based box	LDAP server
LBPRD01	Cisco Content Switch	Web load balancer
LBPRD02	Cisco Content Switch	WLI load balancer
WLSPRD01	IBM p520 server 2 CPU/2 core POWER6 4.2Ghz, 32 GB RAM	BEA WLI application server
WLSPRD02	IBM p520 server 2 CPU/2 core POWER6 4.2Ghz, 32 GB RAM	BEA WLI application server

Příloha č.2: Přehled změn stavů objednávek

Aktuální stav	Id změny	Popis změny
N/A	A1	Objednávka je ručně založena uživatelem v stavu ' Založeno '. Objednávka je otevřena pro modifikace. Pro všechny položky objednávky je možné spustit technické ověření. V průběhu technického ověření jsou vybrané atributy relevantních položek objednávky uzamčené pro jakékoliv změny.
N/A	B1	Objednávka (např. na dočasné vypojení z důvodu neplacení) je automaticky založena v stavu ' Založeno '. Hlavička i jednotlivé položky objednávky jsou vyplněny automaticky. Objednávka čeká na ruční odeslání k realizaci.
N/A	B2	Objednávka (např. zřízení ADSL z portálu) je automaticky založena v stavu ' Odesláno '. Hlavička i jednotlivé položky objednávky jsou vyplněny automaticky. Objednávka pokračuje ve své realizaci bez nutnosti ručního zásahu uživatele.
Založeno	C1	Stisknutím tlačítka Odeslat je objednávka jako celek je validována podle předdefinovaných pravidel. V případě úspěšné validace se stav objednávky změní na ' Odesláno ', jinak zůstává objednávka ve stavu ' Založeno ' a objeví se chybová hláška.
Založeno	C2	Pokud zákazník požádá o zrušení objednávky ještě před jejím odesláním k realizaci, změní se ručním zrušením objednávky její stav na stav ' Odesláno – Zrušení '.
Založeno	C3	V případě, že není možné odeslat objednávku k realizaci, je tlačítkem Zaparkovat zařazena do stavu ' Zaparkováno '. Stav slouží k „odložení“ objednávek do chvíle, než se např. vytvoří umístění, vypořádá závazek na rušené stanici atd.

Aktuální stav	Id změny	Popis změny
Odesláno	D1	Objednávka je odeslána na technické šetření (včetně rezervace zdrojů) – po obdržení potvrzení z provisioningu je stav objednávky automaticky změněn na ' Technické šetření '.
Odesláno	D2	V případě objednávek na služby, pro které není relevantní automatická realizace, se objednávka dostává automaticky přímo do stavu ' Ruční realizace '.
Odesláno	D3	V případech kdy technické šetření není relevantní (např. objednávky doplňkových služeb apod.) se objednávka po odeslání dostává automaticky přímo do stavu ' Před realizací '.
Odesláno	D4	V případech kdy provisioning není relevantní (např. objednávky na změnu cenových plánů) se objednávka po odeslání dostává automaticky přímo do stavu ' Odesláno – Zpoplatnění '.
Odesláno	D5	V případě chyby při zpracování odeslané objednávky je objednávka vrácena na opravu, stav objednávky je automaticky změněn na ' Vráceno – Realizace '.
Technické šetření	E1	V případě obdržení pozitivního výsledku technického šetření bez nutnosti zastavovat objednávku v stavu ' Realizovatelné ' je objednávka automaticky odeslána k realizaci – stav objednávky je automaticky změněn na ' Před realizací '.
Technické šetření	E2	V případě chyby při zpracování odeslané objednávky v průběhu technického šetření v navazujících provisioning systémech je objednávka vrácena na opravu, stav objednávky je automaticky změněn na ' Vráceno – Realizace '.
Technické šetření	E3	Po obdržení negativního výsledku technického šetření je stav objednávky automaticky změněn na ' Nerealizovatelné '. Do stavu Nerealizovatelné se dostávají i objednávky realizovatelné přes FGSM.
Technické šetření	E4	Po obdržení pozitivního výsledku technického šetření a současně zaevidovaného negativního nebo žádného výsledku technického ověření je stav objednávky automaticky změněn na ' Realizovatelné '. V tomto případě je třeba kontaktovat zákazníka, který buď potvrdí svou objednávku a vybere si termín instalace anebo svou objednávku zruší.
Technické šetření	E5	Když zákazník požádá v průběhu technického šetření o zrušení objednávky, změní se stav ručním zrušením objednávky na ' Odesláno – Zrušení '.

Aktuální stav	Id změny	Popis změny
Nerealizovatelné	F1	Jestliže zákazník trvá na zřízení služby, kterou v daném okamžiku není možné realizovat (stav objednávky ' Nerealizovatelné '), může být jeho objednávka zařazena na waiting list – stav objednávky je ručně změněn na ' Waiting list '.
Nerealizovatelné	F2	Jestliže nebyla zákazníkem objednávana univerzální služba (tj. služba, kterou musí ze zákona ČTc zřídit), může být jeho objednávka odmítnuta – stav objednávky je změněn na konečný stav ' Odmítnuto '.
Realizovatelné	G1	Jestliže zákazník potvrdí nabízené zřízení služby (stav objednávky ' Realizovatelné '), je jeho objednávka ručně odeslána k realizaci a její stav je automaticky změněn na ' Odesláno – Realizace '.
Realizovatelné	G2	Jestliže zákazník nepotvrdí nabízené zřízení služby a svou objednávku zruší – stisknutím tlačítka Zrušeno zákazníkem je stav objednávky změněn na stav ' Odesláno – Zrušení '.
Waiting List	H1	V průběhu investiční výstavby jsou pravidelně aktualizované objednávky v stavu ' Waiting List '. Stav objednávek, které je možné realizovat, je automaticky změněn na ' K realizaci '.
Waiting List	H2	Pokud zákazník požádá o zrušení objednávky zařazené do waiting listu, změní se ručním zrušením objednávky její stav na stav ' Odesláno – Zrušení '.
K realizaci	I1	Pokud zákazník již nemá zájem o zřízení služby, která po dokončení investiční výstavby může být realizována, může být objednávka ručně zrušena – stav objednávky změněn na stav ' Odesláno – Zrušení '.
K realizaci	I2	Objednávky ve stavu ' K realizaci ' jsou zpracovávány stejně jako nově založené objednávky, tzn. mohou být modifikovány podle aktuálních požadavků zákazníka, jsou předmětem technického ověření a musí být ručně odeslané k realizaci stisknutím tlačítka Odeslat (stav objednávky je změněn na ' Odesláno ', viz také bod C1).
Realizace	J1	Po obdržení informace o úspěšné realizaci objednávky je stav objednávky automaticky změněn na stav ' Realizováno '.
Realizace	J2	V případě, že se navzdory pozitivnímu výsledku technického šetření nepodaří úspěšně realizovat objednávku, je její stav automaticky změněn na stav ' Nerealizováno '.
Realizace	J3	Pokud zákazník požádá o zrušení objednávky v realizaci, změní se ručním zrušením objednávky její stav na stav ' Odesláno – Zrušení '.

Aktuální stav	Id změny	Popis změny
Realizace	J4	V případě chyby při zpracování odeslané objednávky v průběhu realizace v navazujících provisioning systémech je objednávka vrácena k opravě, stav objednávky je automaticky změněn na ' Vráceno – Realizace '.
Ruční realizace	K1	Po dokončení ruční realizace objednávky je její stav změněn na stav ' Realizováno '. Pro více informací o ruční realizaci objednávek prostřednictvím aktivit viz školící modul Objednávkové Transakce I.
Ruční realizace	K2	V případě, že se nepodaří úspěšně realizovat objednávku v ruční realizaci, je její stav prostřednictvím zrušení ruční realizační aktivity změněn na stav ' Nerealizováno '.
Ruční realizace	K3	Pokud zákazník požádá o zrušení objednávky ve stavu ruční realizace, změní se ručním zrušením objednávky její stav na stav ' Odesláno – Zrušení '.
Realizováno	L1	Po úspěšné realizaci služby je objednávka automaticky odeslána ke zpoplatnění do billingu. Stav objednávky je automaticky změněn na ' Odesláno – Zpoplatnění '.
Nerealizováno	M1	Jestliže není možné realizovat objednávku zákazníka a nejedná se o univerzální službu (např. jednoduchá HTS), může být objednávka ručně zrušena – její stav je změněn na konečný stav ' Odmítnuto '.
Nerealizováno	M2	Jestliže zákazník trvá na zřízení služby, kterou v daném čase nebylo možné realizovat (stav objednávky ' Nerealizováno '), může být jeho objednávka zařazena na waiting list – stav objednávky je ručně změněn na ' Waiting List '.
Odesláno – Zpoplatnění	N1	Po úspěšném zpoplatnění služby v billingu je stav objednávky automaticky změněn na konečný stav ' Dokončeno '.
Odesláno – Zpoplatnění	N2	V případě chyby při zpoplatnění objednávky v billingu je stav objednávky automaticky změněn na ' Vráceno – Zpoplatnění '. Relevantní položky objednávky jsou otevřené k změnám chybných dat relevantních pro billing.
Vráceno – Realizace	O1	Ručně opravená objednávka může být opětovně odeslána k realizaci stisknutím tlačítka Odeslat – stav objednávky je změněn na ' Odesláno '.
Vráceno – Zpoplatnění	P1	Po opravě chybných položek objednávky je možné objednávku ručně odeslat do billingu – stav objednávky je automaticky změněn na ' Odesláno – Zpoplatnění '.
Zaparkováno	Q1	Pro dokončení zaparkované objednávky je nutno stisknutím relevantního tlačítka zrušit její zaparkování. Ve stavu ' Založeno ' je pak možné provést dodatečné modifikace objednávky a odeslat ji k realizaci.
Zaparkováno	Q2	Pokud zákazník požádá o zrušení objednávky ještě před jejím odesláním k realizaci, změní se ručním zrušením objednávky její stav na stav ' Odesláno – Zrušení '.

Příloha č.3: Změny stavů objednávek v závislosti na jejich stavu

	Skupina objednávek			
	Skupina 1	Skupina 2	Skupina 3	Skupina 4
Stav objednávky	Založeno, Vráceno – Realizace, K realizaci	Realizovatelné, Nerealizovatelné	Zaparkováno, Odesláno, Před realizací, Odesláno – Realizace, Technické šetření, Realizace, Ruční realizace, Waiting List, Nerealizováno	Odesláno – Zrušení, Zrušeno zákazníkem, Odmítnuto, Realizováno, Odesláno – Storno, Stornováno, Odesláno – Zpoplatnění, Vráceno – Zpoplatnění, Dokončeno
Změny obecně	Objednávka je otevřená pro jakékoliv modifikace.	Objednávka je otevřená pro určité modifikace.	Objednávka je uzavřena pro jakékoliv modifikace.	Objednávka je uzavřena pro jakékoliv modifikace.
Smazání objednávky	Zakázáno	Zakázáno	Zakázáno	Zakázáno
Zrušení (Storno) objednávky a. ze strany O2 b. zákazníkem	Povoleno	Povoleno	Povoleno Výjimka: Objednávka ve stavu 'Realizace' může být zrušena do doby její fyzické realizace na ústředně. Při zrušení objednávky se provádí dotaz, zda je možné objednávku zrušit nebo nikoliv.	Zakázáno
Změna termínu instalace	Povoleno	Povoleno	Povoleno	Zakázáno

	Skupina objednávek			
	Skupina 1	Skupina 2	Skupina 3	Skupina 4
Stav objednávky	Založeno, Vráceno – Realizace, K realizaci	Realizovatelné, Nerealizovatelné	Zaparkováno, Odesláno, Před realizací, Odesláno – Realizace, Technické šetření, Realizace, Ruční realizace, Waiting List, Nerealizováno	Odesláno – Zrušení, Zrušeno zákazníkem, Odmítnuto, Realizováno, Odesláno – Storno, Stornováno, Odesláno – Zpoplatnění, Vráceno – Zpoplatnění, Dokončeno
Změna billingových dat	Povoleno	Povoleno	Povoleno	Zakázáno Výjimka: Na objednávce ve stavu 'Vráceno – Zpoplatnění' je možné měnit chybná billingová data.
Změna doplňkových služeb	Povoleno	Zakázáno	Zakázáno	Zakázáno
Změna data vrácení smlouvy	Povoleno	Povoleno	Povoleno	'Realizováno', 'Odesláno – Zpoplatnění' – Povoleno 'Dokončeno' – Změna data je možná pomocí speciální funkce (jen v případě že datum ještě nebylo vyplněno)
Tisk korespondence – vytváření tiskových aktivit	Povoleno	Povoleno	Povoleno	Povoleno

Příloha č.4: Struktura tabulky ORDER_SUM

ORDER_SUM				
Název sloupce	Datový typ	Klíč	Zdrojový systém	Popis
Order_id	varchar2 (60)	PK	Siebel	jednoznačný identifikátor objednávky
Order_num	varchar2 (120)		Siebel	číslo objednávky
Trans	varchar2 (200)		Siebel	typ transakce objednávky
Created	date		Siebel	vytvořeno
Requested	date		Siebel	požadované datum realizace
Rp_aps_date	date		Siebel	datum, kdy bylo provedeno v systému APS
Stav	varchar2 (120)		Siebel	stav objednávky
Stav_reason	varchar2 (400)		Siebel	podrobný popis stav hlavičky
Item_id	varchar2 (60)		Siebel	identifikátor položky objednávky
Int_id	varchar2 (120)	FK	Siebel	integrační ID pro interface
Int_flag	char (4)		Siebel	příznak integrace
Prod_code	varchar2 (400)		Siebel	kód produktu
Prod_name	varchar2 (400)		Siebel	název produktu
Action	varchar2 (120)		Siebel	co se má vykonat
Item_stav	varchar2 (120)		Siebel	stav položky objednávky
Stat_date	date		Siebel	datum změny stavu položky
Stat_detail	varchar2 (1600)		Siebel	opis uživatelské poznámky
Stat_desc	varchar2 (1020)	FK	Siebel	chybové hlášení
ODSP_stat	varchar2 (120)		ODSP	popis stavu

				objednávky v ODSP
ODSP_date	date		ODSP	datum zpracování v ODSP
Wli_stat	varchar2 (120)		Wli	popis stav obj ve WLI
Wli_date	date		Wli	datum zpracování ve WLI
Spc_stat	varchar2 (1016)		Spc	popis stav obj v SPC
Tk	number (1)		Automat	flag, určující produkt telekonto
Llu	number (1)		Automat	flag, určující produkt OLO
Dbg	number (1)		Automat	určuje počet zpracování automatem
Run_date	varchar2 (24)		Automat	datum zpracování automatem
Run_comment	varchar2 (2000)		Automat	pole pro ruční poznámky
Run_person	number (1))		Automat	pole, pro identifikaci opravného modulu

Příloha č.5: Struktura tabulky ORDER_SUM_HISTORY

ORDER_SUM_HIST				
Název sloupce	Datový typ	Klíč	Zdrojový systém	Popis
Order_id	varchar2 (60)		Siebel	jednoznačný identifikátor objednávky
Order_num	varchar2 (120)		Siebel	číslo objednávky
Trans	varchar2 (200)		Siebel	typ transakce objednávky
Created	date		Siebel	vytvořeno
Requested	date		Siebel	požadované datum realizace
Rp_aps_date	date		Siebel	datum, kdy bylo provedeno v systému APS
Stav	varchar2 (120)		Siebel	stav objednávky
Stav_reason	varchar2 (400)		Siebel	podrobný popis stav hlavičky
Item_id	varchar2 (60)		Siebel	identifikátor položky objednávky
Int_id	varchar2 (120)		Siebel	integrační ID pro interface
Int_flag	char (4)		Siebel	příznak integrace s ostatními systémy
Prod_code	varchar2 (400)		Siebel	kód produktu
Prod_name	varchar2 (400)		Siebel	název produktu
Action	varchar2 (120)		Siebel	co se má vykonat
Item_stav	varchar2 (120)		Siebel	stav položky objednávky
Stat_date	date		Siebel	datum změny stavu položky
Stat_detail	varchar2 (1600)		Siebel	opis uživatelské poznámky
Stat_desc	varchar2		Siebel	chybové hlášení

	(1020)			
ODSP_stat	varchar2 (120)		ODSP	popis stavu objednávky v ODSP
ODSP_date	date		ODSP	datum zpracování v ODSP
Wli_stat	varchar2 (120)		Wli	popis stav obj ve WLI
Wli_date	date		Wli	datum zpracování ve WLI
Spc_stat	varchar2 (1016)		Spc	popis stav obj v SPC
Tk	number (1)		Automat	flag, určující produkt telekonto
Llu	number (1)		Automat	flag, určující produkt OLO
Dbg	number (1)		Automat	určuje počet zpracování automatem
Run_date	varchar2 (24)		Automat	datum zpracování automatem
Run_comment	varchar2 (2000)		Automat	pole pro ruční poznámky
Run_person	number (1))		Automat	pole, pro identifikaci opravného modulu

Příloha č.6: Struktura tabulky ORDER_SUM_LOG

ORDER_SUM_LOG		
Název sloupce	Datový typ	Popis
Popis	varchar2 (1000)	Názvy jednotlivých částí opravné package
Datum	date	Kdy proběhly

Příloha č.7: Struktura tabulky ORDER_SUM_MESSAGE

ORDER_SUM_MESSAGE			
Název sloupce	Datový typ	Klíč	Popis
Message	varchar2 (255)	PK	Text chybového hlášení, určující vstup do procesu automatického přeposlání objednávky
Flag_Active	varchar2 (1)		Flag, určující jestli je chyba aktivní nebo ne
Flag_RZ	varchar2 (1)		Flag určující, která s procedur přeposlání objednávek se použije

Příloha č.8: Struktura tabulky ORDER_SUM_TMP

ORDER_SUM_TMP			
Název Sloupce	Datový typ	Klíč	popis
Order_id	varchar2 (15)	FK	Row_id objednávky z order_sum
Order_num	varchar2 (30)		Číslo objednávky z order_sum

Příloha č.9: Opis spouštěcí části kódu

*/*Tato procedura je spouštena kazdy den v 2:30 na SIEBEL_CLAUDIA a zahrnuje naplneni*

tabulky ORDER_SUM potencialne zamrzlymi objednavkami a spousti dalsi "cistici" procedury/*

procedure Schedule

IS

Begin

Log('Start Schedule');

Fix_S_Evt_Act;

Fix_S_Order;

auto_send_initial;

commit;

Fill_Order_Sum ('Realizace#Vráceno - Realizace#Odesláno - Storno#Odesláno - Zrušení#Odesláno - Realizace');

Fill_Order_Sum_VZ;

Update_TK_LLU;

Update_SPC('Odesláno#Realizace#Vráceno - Realizace#Odesláno - Storno#Odesláno - Zrušení#Odesláno - Realizace');

Update_WLI('Odesláno');

commit;

Auto_Send;

Send_Msgs;

Send_SPC14;

Mark_term;

Mark_ebill;

Mark_RP;

Park_Order;

Log('End Schedule');

End Schedule;

Příloha č.10: Procedura Auto_send

```
/******  
*****/  
/*Automaticke znovu odeslani objednavek ve stavu Vraceno - Realizace a s chybou z  
SPC nebo RP*/  
procedure Auto_Send  
IS  
Begin  
Log('Start Auto_Send');  
for rec in (   
    select distinct order_num from order_sum where stav='Vraceno - Realizace'  
and run_date=to_char(sysdate,'YYMMDD') and  
    ((stat_desc like '%SPC 5: Chyba není definována%' and spc_stat in  
( 'Neexistuje','Čekání na ZL') ) or  
    (stat_desc like '%SPC 3: Chybý povinný parametr%' and spc_stat='Neexistuje'  
and trans<>'Přeměna') or  
    (stat_desc like '%SPC 16: Formát parametru je chybný%' and spc_stat ='Čekání  
na ZL')  
    )  
    ) loop  
    --3. Update polozek  
    update s_order_item  
        set last_upd = SYSDATE, stav_cd = 'Odesláno'  
        where order_id in ( SELECT row_id  
FROM s_order  
WHERE order_num = rec.order_num )  
        and stav_cd = 'Vraceno - Realizace';  
  
    --4. Update hlavicky  
    update s_order  
        set last_upd = SYSDATE, x_order_int_stav = '1',  
        x_CLAUDIA_out_seq = x_CLAUDIA_out_seq + 1,  
stav_cd = 'Odesláno'
```

```

                                where row_id in ( SELECT  row_id
FROM s_order
WHERE order_num = rec.order_num )
and stav_cd = 'Vráceno - Realizace';

--5. Update inbound sekvence
update s_order_x
      set ATTRIB_48 = '0',last_upd = sysdate
      where row_id in ( SELECT  row_id
FROM s_order
WHERE order_num = rec.order_num );

BEGIN
      update      order_sum      set      RUN_PERSON='8'      where
order_num=rec.order_num;
      EXCEPTION
      WHEN OTHERS THEN
      null;
      END;
end loop;
Log('End Auto_Send');
End Auto_Send;

```

Příloha č.11: Procedura Update_tran

```
-- čistící sekce
drop table pomoc1;
drop table VG_WLI_CDS;
drop table VG_WLI_CDS_TRAN;
drop table VG_WLI_CDS_BEZ_TRAN;
drop table VG_WLI_CDS_BEZ_TRAN_s_PROC;
drop table vG_WLI_CDS_BEZ_TRAN_bez_PROC;
drop table pom2;

-- vytvoření base tabulky pro další datamining
create table pomoc1 as select * from siebel.s_order
where stav_cd='Vygenerováno - Zpoplatnění' and last_upd < sysdate - 10;

-- tab obohacena o údaje z WLI a ODSP
create table VG_WLI_CDS as select turbo.row_id iom_id, wli.stav wli_stav,
cds.order_id cds_id
from          pomoc1          turbo,iom8.iom_process_instance@lnk_wli
wli,cds.order_header@lnk_cds cds
where turbo.row_id=wli.public_id and
      turbo.order_num=cds.order_number;

-- obj s transakcemi
create table VG_WLI_CDS_TRAN as select turbo.*, b.id central_id,b.stav central_stav,
c.iom_code          code_proceseed          from          cds.siebel_csm_central@lnk_cds
b,cds.siebel_csm_processed@lnk_cds c,VG_WLI_CDS turbo
where
turbo.cds_id=b.siebel_order_id and
turbo.iom_id=c.iom_code;

-- obj bez transakcí
```

```

create table VG_WLI_CDS_BEZ_TRAN as select turbo.* from VG_WLI_CDS turbo
where
turbo.iom_id not in (select distinct iom_id from VG_WLI_CDS_TRAN);

```

-- obj bez transakcí, ale s procesorem

```

create table VG_WLI_CDS_BEZ_TRAN_s_PROC as select turbo.* ,c.iom_code
code_proceed
from VG_WLI_CDS_BEZ_TRAN turbo,cds.siebel_csm_processed@lnk_cds c
where
turbo.iom_id=c.iom_code;

```

-- obj bez transakcí a bez procesoru !!!!!!!

```

create table VG_WLI_CDS_BEZ_TRAN_bez_PROC as select turbo.*
from VG_WLI_CDS_BEZ_TRAN turbo
where
turbo.iom_id not in (select iom_id from VG_WLI_CDS_BEZ_TRAN_s_PROC);

```

-- vytvoření pomocné tabulky s objednávkami s transakcemi v jiném než koncovém stav, na základě čehož, proběhne vlastní automatické ukončení

```

create table pom2 as select * from VG_WLI_CDS_TRAN;
begin
for rec in (select iom_id from pom2 where central_stav not in ('S','C','X'))
loop
delete pom2 where pom2.iom_id=rec.iom_id;
commit;
end loop;
for rec in (select iom_id from pom2 where wli_stav='In Progress')
loop
delete pom2 where pom2.iom_id=rec.iom_id;
commit;
end loop;

```

```

end;
-- update transakcí na konečný stav
update cds.siebel_csm_central@lnk_cds set stav='C' where id in (select central_id from
pom2);

-- update item objednávek na konečný stav
update siebel.s_order_item set last_upd = SYSDATE, stav_cd = 'Dokončeno'
where order_id in (select iom_id from pom2);

--update hlavičky objednávek na konečný stav + znemožnění další modifikace
objednávky
update siebel.s_order set last_upd = SYSDATE, stav_cd = 'Dokončeno',active_flg='N'
where row_id in (select iom_id from pom2);

```

Příloha č.12: Opis tabulky ORDER_SUM_MESSAGE

MESSAGE	FLAG_ ACTIVE	FLAG_ RZ
SPC 20: Chybí podmíněný parametr	A	R
SPC: Nelze odeslat do SPC	A	R
Timeout occured in ADSL check process	A	R
SendIntOrder general exception	A	R
cz.telecom.ibwli.core.IB8_ServiceProcessingException: Service error in processing : osasdl: Aplikace SDL vratila ORA:%	A	R
weblogic.wtc.jatmi.TP%	A	R
Systém SdlCis vrátil chybu%	A	R
Nastal problém při odesílání dat do IOL	A	R
Chybová zpráva z IOL: Msg: ERR_0000%	A	R
Cannot send message to Recharging system.%	A	R
Chybová zpráva z IOL: Msg: Exception caught:cz.hpi.db.exceptions.UpdateDaoException: %	A	R
cz.telecom.ibwli.core.IB%	A	R
RP: Nelze odeslat do RP%	N	R
Extra materiál:%	A	R
Chybová zpráva z IOL: NOSB-0002 : OSS unreachable%	A	R
Chybová zpráva z IOL: Msg: null, Node: Check Migration-Free Campaign Conditions%	A	R
%Msg: There is no running add-on product with service attribute: HasModem, Node: changeAddOnProductStart%	A	R
com.accenture.iom.exceptions.mapping.MappingException: Error in mapMigrateLinesInfo;Exception:null%	A	R
Error in mapContactPerson;Exception:null%	A	R
Neplatná odpověď z IOL%	A	R
%Cannot stop billing sooner%	A	R
%TransactionRolledbackLocalException:%	A	R
%Msg: ERR_1779: Invalid modem code%	A	R
%ORA-02049: timeout: distributed transaction waiting for lock%	A	R
NOSB-1009 : There is already running order in NOSB%	N	R
Already running order%	N	R
NOSB-0001 : Unexpected exception.Msg: ERR_1813: Can't start provisioning for accId%	A	R
SendIntOrderResponse returned error%	A	R
Neplatná odpověď z NOSB (Prázdné XML)	A	R
TerminationTime is already updated.	A	R
TPETIME - timeout occured, nested throwable (Exception=weblogic.wtc.jatmi.TPException)	A	R
Error code: 6; Description: NOSB internal error	A	R
Error code: 6; Description: Error while calling getCustomerData:%	A	R
TPESVCERR - server error while handling request, nested throwable (Exception=weblogic.wtc.jatmi.TPReplyException)	A	R

NOSB-1001 : Synchronization of customer failed%	A	R
Error while sending to RP: too many resend tries	N	R
3P/2P billing switch IOM-NOSB in-progress.	A	R
%Error code: 6; Description: empty order %	A	R
SPC 3: Chybí povinný parametr%	N	R
Error code: 6; Description: There is an already running order	N	R
SPC 22: Pro toto číslo již existuje jiná objednávka	N	R
cz.telecom.ibwli.core.IB%	A	R
SPC 5: Chyba není definována%	A	R
Nelze provést,probiha storno%	A	R
ORA-20225: #\$/;-2291;integritni omezeni (CDS.FK%	A	Z
Integration failure: SQLERRM: ORA-20225%	N	Z
ORA-20225: #\$/;-1422;pA?esnA© naĀ?tenA vracA vAce%	A	Z
ORA-20225: #\$/;-1422;presne nacteni vraci více nez pozadovany%	A	Z
CDS_DB: Exception occured on statement number %	A	Z
ORA-20225: #\$/;-2291;integrity constraint (CDS.FK_ORDER_HE%	A	Z
ORA-20225: #\$/;-2291;integrity constraint (CDS.FK_ORDER_IT_ORDER_ITE_ORDER_%	A	Z
ORA-20225: #\$/;-2291;integrity constraint (CDS.FK_ORD_ITEM_CONN_MODE_ID%	A	Z
ORA-20225: #\$/;-2291;integrity constraint (CDS.FK_ORDER_IT_ORD_ITEM_PRODUCT%	A	Z
%ORA-20225: #\$/;100;no data found;P_GET_SUBSCRIBER_STAV;F_FILL_PARAMS;P_MA IN;P_FINISH_HIERARCHIC_ENTITY;P_SET_OBJECT%	A	Z
ORA-20225: #\$/;-2291;integritn omezen (CDS.FK_ORDER_ITEM_PARTY_2) poruāeno - nenalezen %	A	Z
ORA-20225: #\$/;-4063; has errors ORA-04063: package body%	A	Z
ORA-20225: #\$/;-20225;Parameter Refc was not found within string Split_ref_no%	A	Z
%unable to extend%	A	Z
ORA-20225: #\$/;-20000;ORU-10027: buffer overflow, limit of 500000 bytes;F_TRX_ORDER_%	A	Z
ORA-20225: #\$/;100;nenalezena zadna data;F_TRX_ORDER_MAIN;P_MAIN;P_FINISH_HIERARCH	A	Z
ORA-20225: #\$/;-20225;Parameter p_iom_date is NULL or has an incorrect format;P_GET_INPUT_PARAMETER;%	A	Z
%Could not create pool connection. The DBMS driver exception was: Listener refused the connection with the following error:%	A	Z
%942;tabulka nebo pohled neexistuje;%	A	Z
Integration failure: SQLERRM: ORA-0000: normal, successful completion, Trace: , COMPARE ERROR%	A	Z
ORA-20225: #\$/;-20225;Input parameter p_install_price, type N%	A	Z