

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

RACIONALIZACE PROCESŮ V PROSTŘEDÍ LOTUS NOTES

PROCESS RACIONALIZATION IN LOTUS NOTES SURROUNDINGS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. JAN KOJECKÝ

VEDOUCÍ PRÁCE
SUPERVISOR

doc. Ing. BRANISLAV LACKO, CSc.

BRNO 2013

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav automatizace a informatiky

Akademický rok: 2012/2013

ZADÁNÍ DIPLOMOVÉ PRÁCE

student(ka): Bc. Jan Kojecký

který/která studuje v **magisterském navazujícím studijním programu**

obor: **Aplikovaná informatika a řízení (3902T001)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Racionalizace procesů v prostředí LOTUS NOTES

v anglickém jazyce:

Process racionalization in LOTUS NOTES surroundings

Stručná charakteristika problematiky úkolu:

Navrhnout řešení, které zracionalizuje v prostředí používaného produktu LOTUS NOTES vybrané firemní procesy s cílem zvýšit efektivitu automatizovaných funkcí.

Cíle diplomové práce:

- 1)Analyzovat různé způsoby racionalizace procesů s využitím současných informačních technologií
- 2)Vybrat vhodné procesy podporující administrativní činnosti v LOTUS NOTES
- 3)Naprogramovat vybraná řešení ve skriptovacím jazyku LOTUS NOTES

Seznam odborné literatury:

Tvrdíková, M.: Zavádění a inovace informačních systémů ve firmách. Grada 2000 Praha

Molnár, Zd.: Efektivnost informačních systémů. Grada 2001 Praha

Voříšek, J.: Strategické řízení informačního systému a systémová integrace. Management Press 1997 Praha

Vedoucí diplomové práce: doc. Ing. Branislav Lacko, CSc.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2012/2013.

V Brně, dne 21.11.2012

L.S.

Ing. Jan Roupec, Ph.D.
Ředitel ústavu

prof. RNDr. Miroslav Doupovec, CSc., dr. h. c.
Děkan fakulty

ABSTRAKT

Tato práce popisuje racionalizaci procesu v prostředí Lotus Notes ve firmě, která zajišťuje kompletní podporu produktu Lotus Notes pro velkou mezinárodní firmu. Popisuje tříúrovňový systém podpory. Dále se zabývá racionalizací procesu mezi první úrovní podpory (zákaznický servis) a druhou úrovní podpory (Lotus notes administrace). V neposlední řadě popisuje vytváření – naprogramování příslušných agentů v Lotus Notes pro tým druhé úrovně, kteří pomohou ke zrychlení a zkvalitnění služeb, poskytovaného tímto týmem.

ABSTRACT

This work describes the rationalization process in Lotus Notes in a company that provides complete support for Lotus Notes Product for a large international company. It describes three level support and also deals with the rationalization process between the first level of support (customer service) and second level support (Lotus Notes administration).

Finally, describes how to create – programming the agents in the Lotus Notes team for the second level, which will help to speed up and improve the services provided by this team.

KLÍČOVÁ SLOVA

Zákaznické centrum, Lotus notes, LotusScript, Administrace, tlačítková lišta, migrace

KEYWORDS

Call Center, Lotus Notes, LotusScript, administration, Tool bar, migration

PROHLÁŠENÍ O ORIGINALITĚ

Prohlašuji, že jsem diplomovou práci vypracoval samostatně za použití uvedené literatury a odborných zdrojů pod vedením doc. Ing. Branislava Lacka, CSc.

V Brně dne 21.5.2013

.....

podpis

BIBLIOGRAFICKÁ CITACE

KOJECKÝ, J. *Racionalizace procesů v prostředí LOTUS NOTES*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2013. 64 s. Vedoucí diplomové práce doc. Ing. Branislav Lacko, CSc..

PODĚKOVÁNÍ

Chtěl bych touto formou poděkovat vedoucímu diplomové práce doc. Ing. Branislavu Lacku, CSc. za věcné připomínky a rady ohledně tvorby této práce.

OBSAH:

1. ÚVOD	11
2. CÍLE DIPLOMOVÉ PRÁCE	13
3. POTŘEBA A ÚČEL RACIONALIZACE	15
3.1 Reengineering	15
3.2 Tří úroňový systém podpory	16
3.2.1 První úroveň.....	17
3.2.2 Druhá úroveň	18
3.2.3 Třetí úroveň	18
3.3 Tíketovací systém	19
4. PROSTŘEDÍ LOTUS NOTES.....	21
4.1 Bezpečnost	22
4.2 Replikace.....	23
5. POUŽITÉ NÁSTROJE	25
5.1 LotusScript.....	25
5.2 Jazyk vzorců.....	25
5.3 Pole v Lotus Notes Dokumentu	26
5.3.1 Nástroj „Notes Document Viewer“.....	28
5.3.2 Tlačítko „Special Button“	28
5.3.3 Nástroj „ScanEZ“	30
5.4 Nástroj „Violet UML Editor“.....	30
6. ANALÝZA.....	31
6.1 Analýza dokumentace na zákaznickém centru.....	31
6.2 Analýza dokumentace na druhé úrovni	32
6.3 Analýza technické práce na druhé úrovni	33
6.3.1 Upozornění na smazání účtů VIP osob – současný stav	34
6.3.2 Analýza procesu při přejmenování uživatele a následná úprava profilu v databázi personálního oddělení	35
6.3.3 Analýza zjištění kdo poslední upravil dokument	35
6.3.4 Analýza přidávání uživatelů nebo úprava polí v jednotlivých dokumentech v jakékoliv databázi	36
6.3.5 Analýza nastavení tlačítek	36
6.3.6 Analýza migrace uživatele z lokální do globální domény a naopak	36
7. NÁVRH RACIONALIZOVANÉHO ŘEŠENÍ	39
7.1 Návrh dokumentace na zákaznickém centru	39
7.2 Návrh dokumentace na druhé úrovni	40

7.3	Návrh technické práce na druhé úrovni.....	40
7.3.1	Návrh upozornění na smazání účtů VIP osob.....	40
7.3.2	Návrh informovat personální oddělení při přejmenování uživatele.....	41
7.3.3	Návrh na zjištění, kdo upravoval dokument.....	41
7.3.4	Návrh na přidávání uživatelů nebo úprava polí v jednotlivých dokumentech v jakékoliv databázi.....	42
7.3.5	Návrh řešení automatické distribuce uživatelsky definované tlačítkové lišty.....	42
7.3.6	Návrh řešení na migraci uživatele.....	42
8.	REALIZACE ŘEŠENÍ.....	43
8.1	Vytvoření dokumentace na první úrovni.....	43
8.2	Vytvoření dokumentace na druhé úrovni.....	43
8.3	Realizace technické práce na druhé úrovni.....	44
8.3.1	Upozornění na smazání účtu VIP osoby.....	44
8.3.2	Vytvoření agenta „Informuj HR“.....	45
8.3.3	Vytvoření tlačítka „Modified by“.....	47
8.3.4	Vytvoření agenta „Add/Remove“.....	47
8.3.5	Realizace automatické distribuce uživatelsky definované tlačítkové lišty.....	49
8.3.6	Vytvoření migrační databáze.....	56
9.	TEST PROGRAMOVÉHO ŘEŠENÍ.....	59
10.	ZÁVĚR.....	61

1. ÚVOD

Tato práce popisuje racionalizaci procesu v prostředí Lotus Notes ve firmě, která zajišťuje kompletní podporu produktu Lotus Notes pro velkou mezinárodní společnost.

Obsah práce je zaměřen na problematiku racionalizace v oblasti firemních informačních systémů. Informační systémy dnes slouží pro podporu jak řízení na strategické úrovni, tak na úrovni středního managementu. Jak zdůraznil prof. Voříšek ve své publikaci [4] pro dobré řízení, firmy si musí vybudovat vedení firmy i velmi dobrý informační systém.

I když se práce zabývá racionalizací v prostředí Lotus Notes na druhé úrovni, tak předtím než začneme přemýšlet o různých zlepšovacích návrzích na zjednodušení řešení těchto problémů, tak je potřeba si položit otázku, zda není lepší nejprve zmenšit celkový počet některých problémů. To se pokusím důkladnou analýzou těchto problémů a pokusím se docílit toho, že v některých kategoriích se počet požadavků zmenší a některé kategorie se úplně eliminují.

Potom se teprve začnu soustředit na to jak si práci zjednodušit na samotné druhé úrovni podpory Lotus Notes.

Práce dále popisuje racionalizaci procesu mezi první úrovní podpory (Zákaznický servis) a druhou úrovní podpory (Lotus notes administrace). Racionalizace v mém konkrétním případě představuje specifické řešení inovace stávajících informačních systémů. Inovace současných informačních systémů je velmi důležitý proces, který reaguje jednak na neustálý rychlý vývoj informačních prostředků a informačních technologií, jednak na neustálé změny v oblasti požadavků na informační systémy. Podněty pro inovaci informačních systémů zpracovala ve své publikaci Doc. Trvdíková z VŠB Ostrava. [2]

Následně vytváření příslušné dokumentace pro procesy první a druhé úrovně podpory. V neposlední řadě vytváření – naprogramování příslušných agentů v Lotus Notes pro tým druhé úrovně, které pomohou ke zrychlení a zkvalitnění služeb, poskytovaného tímto týmem. Poskládání kvalitní dokumentace a propojení těchto jednotlivých komponentů podpory obou úrovní dochází ke zrychlení a zkvalitnění celkové podpory Lotus Notes pro koncového zákazníka.

Zákazník v kontaktu s firmou nabývá pocitu velmi profesionálního přístupu týmové práce s přesně definovanými procesy, které jsou v souladu se standardy podporované firmy.

Snahy o racionalizaci procesů celkové podpory servisních služeb jsem odvodil z potřeb zlepšování efektivity všech dotčených procesů, protože efektivita v informačních systémech hraje vždy významnou roli (viz. publikace prof. Molnára [3]).

2. CÍLE DIPLOMOVÉ PRÁCE

Cíle jsem odvodil ze zadání své diplomové práce.

Cílem je analyzovat různé způsoby racionalizace procesů s využitím současných informačních technologií. Dále vybrat vhodné procesy podporující administrativní činnosti v Lotus Notes. Navrhnout řešení, které zracionalizuje v prostředí používaného produktu Lotus Notes vybrané firemní procesy s cílem zvýšit efektivitu automatizovaných funkcí.

Pro konkretizaci východisek a závěrů pro návrhy ve své diplomové práci jsem vycházel z prostředí firmy, která se zabývá outsorsingem údržby počítačových aplikací ve velkých mezinárodních společnostech. Protože toto prostředí je mi dobře známo z mé současné i minulé praxe.

3. POTŘEBA A ÚČEL RACIONALIZACE

Aby jakákoliv firma obstála v dnešní obrovské konkurenci, je potřeba její procesy neustále zdokonalovat, zefektivňovat či zjednodušovat. Existuje mnoho různých způsobů, jak toho lze dosáhnout. Zatímco například „reengineering“ popisuje úplnou rekonstrukci firmy od samotných základů, já jsem se v práci více zaměřil na racionalizaci dílčích procesů v komplexním „Tříúrovňovém systému podpory“. [8]

3.1 Reengineering

„Existuje jen nemnoho podniků, jejichž vedení netvrdí, alespoň v rámci svého působení na veřejnost, že chce mít dostatečně pružnou organizaci, aby byla schopna se rychle přizpůsobovat měnícím se tržním podmínkám a obstát i v té nejostřejší cenové konkurenci.

Vedení by si přálo, aby jejich podniky byly štíhlé, čilé, pružné, schopné pohotové reakce, konkurenceschopné, inovační, výkonné, orientované na zákazníky a ziskové. Realita je ovšem jiná – podniky trpí otylostí, neohrabaností, strnulostí, pomalostí, neschopností konkurovat, nedostatkem tvořivosti, neohospodárností, pohrdáním potřebami zákazníků a klesající ziskovostí.

Odpověď je třeba hledat v tom, jakým způsobem a proč tyto podniky pracují.

Dnešní firmy ale mohou situaci vyřešit a provést svou obrodu. Techniky, které mohou k tomu použít, nazýváme podnikovým reengineeringem. Ten má pro nastupující revoluci v podnikání stejný význam, jaký dosud měla specializace práce. Světové podniky – dokonce i ty neúspěšnější a nejslibnější z nich – musí zvládnout a aplikovat principy podnikového reengineeringu. Jinak budou zastíněny úspěchem těch firem, které to dokáží.

Reengineering ale není rychlý a zaručený způsob nápravy, jež manažeři ve svých organizacích mohou použít. Není to nový trik, který slibuje skok v jakosti výrobků a služeb, jež tyto organizace dodávají či poskytují, nebo výrazné ořezání nákladů. Podnikový reengineering není programem zaměřeným na pozvednutí morálky dělníků nebo motivování prodejců. Nepřinutí starý počítačový systém, aby pracoval rychleji. Podnikový reengineering se prostě netýká nápravy něčeho.

Podnikový reengineering znamená začít vše znovu, začít od nuly. Znamená to odložit větší část moudrosti, která byla získána v průběhu dvousetletého vývoje průmyslového managementu. Je potřeba zapomenout, jak se pracovalo v době masových trhů, a rozhodně hledat ty nejlepší způsoby, jak pracovat dnes.

Reengineering staví na stejných vlastnostech, které tradičně charakterizovaly velké podnikové inovátory: individualismus, spoléhání na sebe sama, ochota přebírat riziko, přirozený sklon k iniciování změn. Reengineering zhodnocuje výhody existujících talentů a uvolňuje vynalézavost.

Firma, která nedokáže změnit způsob svého nazírání na informační technologii, nemůže provést reengineering. Firma, jež ztotožňuje informační technologii s automatizací, nemůže při reengineeringu uspět. Informační technologie má v podnikovém reengineeringu klíčovou úlohu, velice snadno však může dojít k tomu, že bude špatně obsazena. Moderní informační technologie je součástí každého reengineeringového úsilí, jeho významným spoluvůrcem, protože reengineering podnikových procesů umožňuje provést. Nestací ovšem pouze nasadit počítače na současné podnikové problémy, aby došlo k jejich automatickému reengineeringu.

Nesprávné užití informační technologie může reengineering zcela zablokovat tím, že posílí staré způsoby myšlení a stará schémata chování.

Aby si mohly uvědomit možnosti, jež jsou moderní informační technologii vlastní, a dokázali si představit její aplikace, je třeba, aby si podniky osvojili způsob myšlení, kterému se podnikatelé obvykle neučí a jenž jim ani příliš nesedí. Většina vedoucích či manažerů umí myslet deduktivně. Znamená to, že dokáží vymezit problém nebo problémy, hledat a vyhodnocovat jejich různá řešení. Ale uplatňování informační technologie k podnikovému reengineeringu vyžaduje induktivní myšlení – schopnost nejdříve rozpoznat, jaké významné možnosti informační technologie poskytuje, a teprve potom hledat problémy, které by mohla vyřešit, problémy, o nichž podnik pravděpodobně ani neví, že je má.

Zásadní chybou, které se většina podniků dopouští při pohledu na informační technologii, je, že ji nahlíží optikou svých dosavadních procesů. Ptají se: „Jak můžeme užít těchto nových možností k zlepšení, k zdokonalení či zefektivnění toho, co již děláme?“ Místo toho by se ovšem měly ptát: „Jak můžeme využít informační technologie k tomu, abychom mohli dělat věci, jež dosud neděláme?“ Reengineering se na rozdíl od automatizace týká inovace. Týká se využití nejnovějších možností technologie k dosažení naprosto nových cílů.“ [6][7]

3.2 Tři úroňový systém podpory

Při řešení jakýchkoliv problémů nebo požadavků od zákazníka nastává velmi často situace, že se některé problémy nebo požadavky opakují – každý člověk začne okamžitě intuitivně přemýšlet, jak se vyhnout jakékoliv monotónní práci a snaží si práci jakýmkoliv způsobem zjednodušit a zároveň tyto inovace pak nasdílet se zbytkem týmu, čímž se zvýší efektivita týmu a najde se prostor pro další inovace anebo pro další práci. S tímto spojené je i rychlejší řešení problémů či požadavků a na základě toho i spokojenost zákazníka, což je pro firmy, které poskytují zákaznickou podporu klíčová věc.

V malých firmách může fungovat jednoduchý model zákaznické podpory, to znamená do určitého počtu uživatelů je možnost mít jednoho či více administrátorů, které pokrývají veškeré potřebné úkony k udržení takového informačního systému – chodu firmy. To znamená, že takový administrátor funguje i jako zákaznické centrum i jako administrátor.

V jakékoliv větší firmě je takovýto systém podpory absolutně nedostačující, prakticky nemožný.

Jednou z mnoha metod, které využívají velké firmy, je rozdělení celkové podpory na tři úrovně.

Na každé úrovni se nachází týmy – skupiny lidí, kteří se starají o jednotlivé úkoly popsané na obrázku č. 1



Obr. 1 Tříúrovňový systém [9]

3.2.1 První úroveň

Jedním z nejdůležitějších týmů nacházejícím se na první úrovni je zákaznické centrum. Pracovníci zákaznického centra mají na starosti přímý kontakt se zákazníkem. Ať už se jedná o komunikaci prostřednictvím telefonu, e-mailu, či vzdáleného připojení.

Aby se udržela nějaká úroveň podpory je potřeba mít práci dobře organizovanou a zároveň i tuto činnost nějak zlepšovat. Nabízí se různá řešení – například rozdělit tikety uživatelů na problémy (např. jestliže se někde něco pokazilo) a požadavky (např. nový počítač, změna jména atd.).

Je to z toho důvodu, že není potřeba na jednoduchý a jasný požadavek zatěžovat agenta zákaznického centra, ale tyto požadavky je možné posílat e-mailem prostřednictvím webové aplikace.

Přes nějaké webové rozhraní si může zákazník či uživatel zažádat například o přejmenování uživatele. Někdy manažer si zažádá o účet pro svého nového pracovníka, nebo o smazání účtu pro odchávajícího uživatele. Tím pádem dochází k odlehčení zatížení pracovníků zákaznického centra, kteří mohou více času věnovat jednotlivým problémům a dálkové podpoře. Dálková podpora je připojení k samotnému zákazníkovi za účelem přímého vyřešení problémů, nebo posbírání konkrétních a důležitých informací – chybových hlášek, které jsou následně nezbytné pro administrátory druhé úrovně k řešení takovýchto problémů.

Dále se na této první úrovni také nachází tým, který je zodpovědný za zpracování požadavků, ne ale problémů, to znamená požadavky typu – potřebujeme nový mobil pro našeho uživatele. Tomuto týmu přichází požadavky různou cestou, buďto e-mailem přímo od zákazníka nebo přes požadavek z webové databáze. Úkolem tohoto týmu je tyto požadavky přímo zpracovat nebo adresovat příslušným administrátorům.

Na první úrovni se také nachází tým, který je zodpovědný za monitorování serverů.

Lidé pracující v tomto týmu jsou zodpovědní za monitorování serverů, případně za monitorování různých aplikací nacházejících se na těchto serverech. Jedná se o kompletní monitoring jak po softwarové tak i po hardwarové stránce.

Na základě závažnosti daného problému se pracovník rozhoduje, zda bude stačit pouhý restart serveru, nebo se musí na řešení problému podílet i administrátor druhé úrovně.

V případě, že ano, potom administrátor zaznamená daný problém do tiketu, který následně pošle na administrátory druhé úrovně. V případě problémů vyšších priorit je nutné zároveň telefonicky informovat administrátora druhé úrovně, z důvodu bezprostřední reakce.

Dále se pracovník týmu také zabývá selektivním monitoringem různých aplikací, které nejsou hlídané automatickým systémem. V případě, že se tady vyskytne nějaký problém řekněme v SAPu, tak je schopen to vyřešit sám podle problém matice. Někakou základní problematiku je tým schopen vyřešit sám na základě dokumentace, kterou jim připravili administrátoři druhé úrovně.

Tento tým je také zodpovědný za zálohování a obnovu dat.

3.2.2 Druhá úroveň

Na druhé úrovni se nachází například následující týmy: SMTP, Network tým, podpora SAPU.

Nás bude samozřejmě nejvíce zajímat Lotus Notes Administrace.

Tento tým má na starosti následující povinnosti:

- řešení klientských problémů,
- řešení problémů s kalendářem,
- řešení problémů s designem,
- nastavení limitu kvóty e-mailové schránky,
- nastavení hierarchie přístupů
- problémy s replikací,
- problémy s doručováním e-mailů,
- spravování serverů,
- administrace uživatelů,
- administrace databází,
- podpora blackberry.

3.2.3 Třetí úroveň

Na třetí úrovni se nacházejí návrháři, programátoři nebo developři. Ať už tito lidé pracují přímo v mateřské firmě nebo jsou situováni v nějaké externí firmě, která má na starosti vývoj specifických aplikací. Pracovníci první a druhé úrovně jsou situováni na různých lokacích firmy. Buďto jsou umístěni na jiném patře, či jiné budově, nebo v jiném městě či dokonce v jiném státě. Pracovníci druhé a třetí úrovně pracující v mateřské firmě jsou situováni společně, jelikož je potřeba vzájemného propojení povinností obou úrovní. Developer potřebuje totiž neustále aktualizované informace o stavu v produkci. Dále pokud přichází například s novou verzí designu nějaké databáze tak je potřeba tuto verzi nejdříve otestovat v testovacím prostředí a následně detailně monitorovat důsledky v produkčním prostředí.

3.3 Tiketovací systém

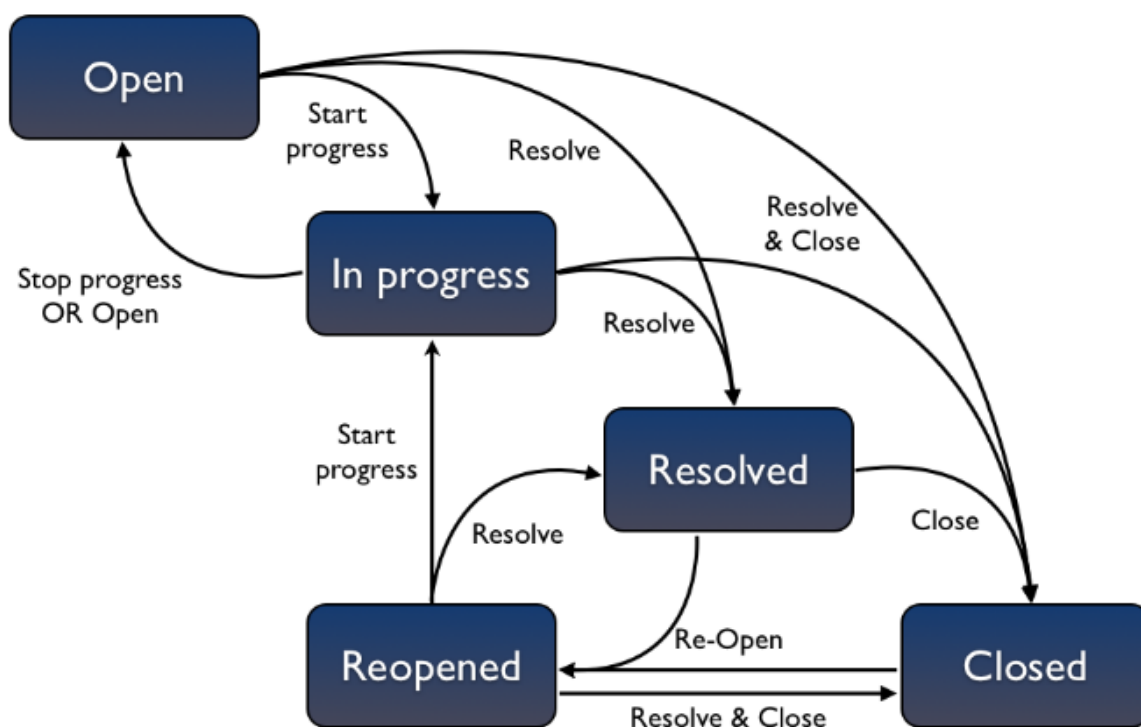
Aby mohla komunikace mezi týmy jednotlivých úrovní nějak fungovat, používá se takzvaný „Tiketovací systém“. Výhodou, oproti například posílání e-mailu mezi jednotlivými úrovní, je, že jakákoliv aktualizace v tiketu je hned viditelná pro všechny týmy. Jak takový systém funguje?

„Tiketovací systém“ – zaznamenání požadavků do formuláře – webové aplikace.

Každá skupina má specifické jméno, a každá skupina může dodat do tiketu nějakou informaci, pokud je tiket přiřazen právě této skupině. Tiket zaznamená pracovník první úrovně podpory a na základě první analýzy posílá patřičné skupině na vyřešení.

V podstatě máme dva druhy tiketů. První se používá na řešení problémů, to znamená, že pracovník zákaznického centra pošle tiket například na tým SMTP. Tento tým do tiketu napíše, co bylo vyšetřeno z jejich strany, buďto problém přímo vyřeší nebo pouze doplní informace ve formuláři a pošlou na další příslušný tým, aby věc vyšetřil ze svého pohledu. Tiket se může i vrátit zpátky na zákaznické centrum a to zejména v případě, že je potřeba kontaktovat zákazníka nebo uplatnit nějaké lokální řešení. Druhý druh tiketu se používá jakékoliv žádosti – například vytvoření účtu pro nového uživatele, smazání účtu uživatele apod. V tomto tiketu existuje více úkolů.

Tiket se v průběhu své cesty může dostat do různých statusů. Děje se to nezávisle na tom na jaké úrovni se nachází, to znamená, že požadavek může vyřešit buďto administrátor nebo pracovník zákaznického centra. Na obrázku č. 2 vidíme cestu takového tiketu dle statusu.



Obr. 2 Cesta tiketu z hlediska statusu. [10]

4. PROSTŘEDÍ LOTUS NOTES

IBM Lotus Notes/Domino je softwarový produkt společnosti IBM, který se orientuje do oblasti groupware. Marketingová barva, neboli takzvaný brand, je žlutá. Spíše než výsledný produkt, by se IBM Lotus Notes/Domino dal označit jako platforma pro vývoj groupware aplikací.

IBM Lotus Notes/Domino je technologie typu klient server, kde aplikace je obvykle umístěna na serveru a klient komunikuje s touto aplikací. Přístup k aplikaci je nativním protokolem nebo internetovým protokolem. Nativní komunikační protokol NRPC (Notes Remote Procedure Call) je použit pro komunikaci mezi serverem Domino a klientem Notes a vzájemně mezi Domino servery. Serverová část se nazývá *IBM Lotus Domino* a klientská část *IBM Lotus Notes*.

IBM Lotus Domino server je robustní aplikační/databázový server. Na tomto serveru jsou umístěny aplikace, jejichž služeb využívají uživatelé prostřednictvím IBM Lotus Notes klienta (popř. internetového klienta). Aplikace na IBM Lotus Domino serveru jsou aktivními databázemi. Prostředí IBM Lotus Domino je distribuovaným systémem, tzn., že aplikace může být umístěna na více než jednom serveru. Synchronizace mezi servery je zabezpečována službou zvanou replikace. Domino je zároveň serverem pro celou řadu dalších internetových služeb, např. POP3, IMAP, LDAP, MAPI, HTTP, NNTP a podporuje mnoho otevřených standardů, např. HTML/XHTML, XML, Java.[5]

Základem IBM Lotus Notes/Domino je dokumentová databáze, tedy na rozdíl od relačních databází je základní datovou jednotkou databáze takzvaný dokument. Dokument má specifické vlastnosti a variabilní počet polí. Dva dokumenty stejného druhu mohou mít tedy různá pole. [11]

Když jsem se poprvé seznámil s Lotus Notes, byla k dispozici verze 6. Já nyní používám verzi 8.5.3. [18] Tato verze se už může pochlubit např. využitím Lotus Notes Traveler 8.5.3. Lotus Notes Traveler synchronizuje e-maily, kalendář, úkoly, adresář a údaje v deníku v reálném čase. Poslední verzí z řady 8 je 8.5.4, ale v současné době existuje na trhu už verze 9 (IBM Notes 9 Social Edition). Tato verze už má mnohem modernější design, klade důraz na implementaci nových sociálních a webových funkcí a přidává i velký počet systémových rozšíření. Např. se jedná o možnost volby replikace databází na pozadí během kompaktu databáze, nebo možnost použití bezpečného úložiště klíčů v speciální databázi "credstore.nsf." "[14]

Funkce a Klienta Lotus Notes:

- e-mail,
- adresář,
- plánovač,
- rezervace zdrojů.

Výhody platformy IBM Lotus Notes/Domino:

- hlavní výhodou je možnost rozšíření základní množiny databází o další aplikace,
- vysoká míra zabezpečení dat,
- replikace,

- snadná rozšiřitelnost,
- rychlý vývoj a snadná aplikace business procesů formou workflow.

Pro vývoj databázových aplikací slouží třetí klient – *Lotus Designer*.

Administrace se provádí pomocí čtvrtého klienta – *Lotus Administrátor*.

Pomocí Administrátora můžeme spravovat:

- uživatele nebo celé skupiny uživatelů,
- jednotlivé servery, serverové clustery,
- opravovat poškozené databáze,
- přidávat, či odebírat práva. [1]

4.1 Bezpečnost

Bezpečnosti každé aplikace je potřeba věnovat v současné době velkou pozornost, protože se jedná vždy o velmi citlivé firemní informace. A v tomto směru je potřeba dodržovat doporučení ISO 27001. [12]

Bezpečnost Lotus Notes je založena na existenci tzv. ID souborů. V těchto souborech jsou uloženy certifikáty, digitální podpisy, šifrovací a dešifrovací klíče a další citlivé informace.

Databáze mohou obsahovat důvěrné informace, tedy vyžadují určitý stupeň zabezpečení. LN nabízejí velice pružný systém ochrany dat, vyhovující různým pracovním prostředím a procesům. Administrátor zodpovídá za bezpečnost serverů Notes, bezpečnost jednotlivých aplikací na těchto serverech mají na starosti manažeři databází. Tedy, přístup k datům je věcí návrhových prvků databáze, nikoliv fyzického umístění souborů. Hlavním bezpečnostním prvkem LN je identifikační ID soubor, který LN vygenerují automaticky každému uživateli při založení jeho účtu. LN používají tento ID soubor k přístupu ke svým serverům a souborům na nich uložených. Ve skutečnosti tento ID soubor obsahuje pár klíčů, veřejný a privátní klíč. Veřejný klíč je prostřednictvím serveru LN publikován všem uživatelům Notes, privátní klíč by měl vlastnit pouze oprávněný uživatel. [1]

Dalším bezpečnostním prvkem LN jsou přístupová práva (ACL), určují úroveň přístupu, kterou má k databázi uživatel, skupina uživatelů, nebo serverů.

Existuje sedm přístupových rolí:

- Manager (manažer) – má právo číst, psát nebo upravovat dokumenty. Může upravovat ACL, nastavovat replikace, spouštět a zastavovat záznam aktivity uživatelů, ale také databázi smazat;
- Designer (návrhář) – má právo provádět stejné operace jako manager s výjimkou úprav ACL, nastavování replikací a záznamu aktivity uživatelů a smazání databáze;
- Editor (editor) – může číst, psát, upravovat a mazat všechny dokumenty;
- Author (autor) – může číst všechny dokumenty, zakládat nové, ale upravovat nebo mazat může pouze dokumenty, jichž je vlastníkem;
- Reader (čtenář) – má právo dokumenty pouze číst;

- Depositor (zapisovatel) – může pouze vytvářet nové dokumenty, ale nesmí žádné dokumenty číst, dokonce ani ty, které sám vytvořil;
- No access (bez přístupu) Nesmí databázi otevírat ani do databáze posílat dokumenty poštou. [11]

4.2 Replikace

Replikace dat je datová výměna mezi databázemi a jejich replikami. Proces replikace umožňuje uložení replik téhož souboru NSF na mnoho serverů a pracovních stanic. Z tohoto důvodu systém Notes a Domino potřebuje mít nějakou možnost zjistit, že dva soubory NSF jsou ve skutečnosti replikami sobě navzájem. Na toto zjištění se používá identifikační číslo databáze, tzv. „Replica ID“. ID repliky je řetězec, sestávající z osmi bajtů. Těchto osm bajtů je obvykle zapisováno hexadecimální notací (příklad zápisu replika ID „85256C39:001AF5EA“). Replikace je inicializována manuálním spouštěním nebo automatickou úlohou serveru. Cílem replikace je udržovat obsah databází aktuální na všech serverech nebo u klientů. [1]

5. POUŽITÉ NÁSTROJE

5.1 LotusScript

„LotusScript je objektově orientovaný programovací jazyk pro LN verze 4. Programy napsané v LotusScriptu nelze kombinovat se vzorci nebo jednoduchými akcemi. K psaní vzorců slouží programovací jazyk vzorců Notes. Skripty a vzorce se připojují podle potřeby k různým objektům Notes. Funkce, které lze použít, určuje konkrétní objekt. LotusScript obsahuje předem definované třídy, jejich vlastnosti a metody. LN automaticky řídí nahrávání a překlad uživatelských skriptů a zařazují do nich definice tříd Notes.“ [1]

Obecně lze říci, že LotusScript lze použít k následujícím úkonům. Přístup k ostatním programům a práci s daty, dále řízení aplikací pro Domino, a to včetně funkcionality aplikací či správy nebo manipulace s daty. Díky této ideální kombinaci vlastností se LotusScript rychle stal nejoblíbenějším programovacím jazykem pro systém Notes/Domino.

LotusScript se jazyku vzorců podobá tím, že umožňuje přístup k datům různých typů a následnou práci s těmito daty. Můžete tedy vybrané proměnné přiřadit nějakou hodnotu a následně se na tuto hodnotu odkazovat prostřednictvím názvu této proměnné. A stejně jako v jazyce vzorců i v jazyce LotusScript platí, že chcete-li pomocí nějaké standardní funkce zpracovat určitá data, pak tato data musíte zvolené funkci předat prostřednictvím nějakého parametru. [1]

5.2 Jazyk vzorců

„Jazyk vzorců je prvním programovacím jazykem systému Lotus Notes, který byl a je součástí všech verzí tohoto systému. Nicméně „věk“ tohoto jazyka vůbec neznamena, že by měl být považován jen za jakousi „dědičnou“ součást systému. Naopak: jazyk vzorců nadále zůstává integrální součástí sady jazyků podporovaných systémem Notes/Domino. Jazyk vzorců, někdy též označovaný jako makrojazyk Notes, je sadou @vzorců, s jejichž pomocí můžete napsat základní funkce, přičemž formát zápisu je velmi jednoduchý a snadno srozumitelný. Vzorce mohou být využity k vytváření aplikací určených jak pro klientský program Notes, tak i pro webový prohlížeč (případně pro smíšené aplikace). Díky jednoduchosti jazyka vzorců mohou i „neprogramátoři“ vytvářet užitečné aplikace, aniž by se museli učit „těžké“ jazyky, mezi něž patří např. LotusScript i Java. I přesto je však funkčnost, kterou jazyk vzorců nabízí, poměrně značná. „

Jazyk vzorců je poměrně jednoduchý a srozumitelný jazyk, který se vcelku snadno a rychle naučí jak zkušený vývojáři, tak i nováčci. Mnozí „ostřílení a zkušený vývojáři aplikací pro Notes/Domino začali právě tímto jazykem. Lze říci, že jazyk vzorců je i jemným úvodem do objektového modelu a funkčnosti systému Notes. Někteří vývojáři, kteří s vývojem aplikací pro Notes teprve začínají, považují jazyk vzorců za výchozí bod, jehož zvládnutí je předpokladem pro přechod ke skriptovacím jazykům.“ [1]

Jazyk vzorců vždy byl – a i v budoucnosti bude – velmi výkonným nástrojem pro tvorbu robustních aplikací pro systém Notes/Domino.

Má totiž následující výhody:

- Snadno se používá;
- Je výkonný a velmi rychlý;
- Naučí se ho rychle jak nováčci, tak i zkušený programátoři;

- Kód napsaný v jazyce vzorců může být využit na různých místech vašich aplikací;
- Použití může vést k rozšíření funkčnosti a zkrácení času potřebného pro vývoj.[1]

5.3 Pole v Lotus Notes Dokumentu

Každý dokument v Lotus Notes databázích (např. obyčejný e-mail) se skládá z různých polí (fields).

Typ prvku	Typ Pole
Text (Text)	Text, Dialogový seznam, Zaškrťovací políčko, Přepínač, Heslo, Časové pásmo, Barva
Jména (Names)	Jména
Autoři (Authors)	Autoři
Čtenáři (Readers)	Čtenáři
Číslo (Number)	Číslo
Čas/Datum (Date/Time)	Datum a čas
Vzorec (Formula)	Vzorec
RTF (Rich text)	Text ve Formátu RTF, RTF se seznamem položek

Obr. 3 Typy prvků a polí. [1]

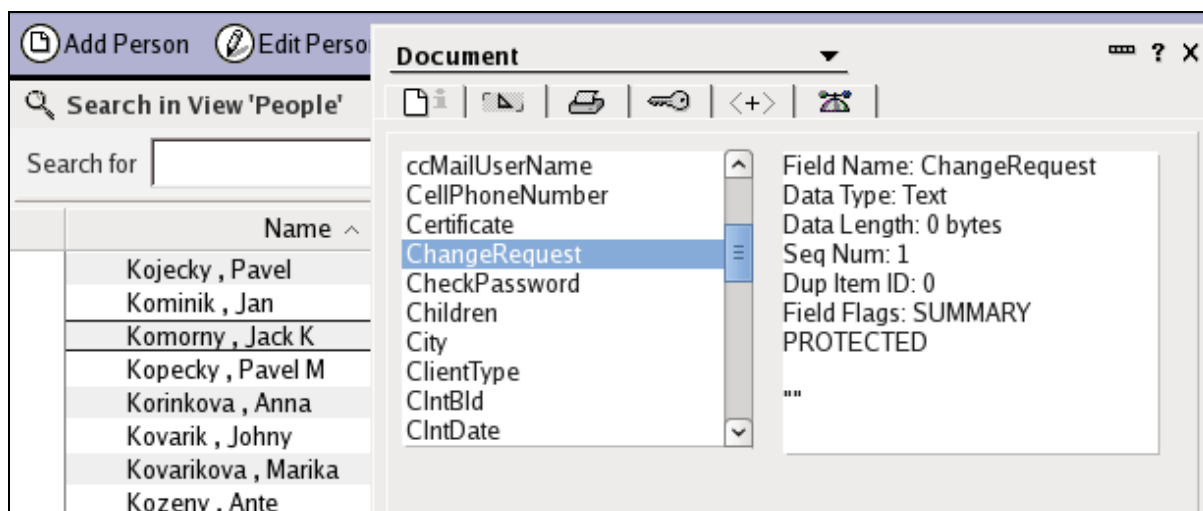
Uživatelské rozhraní umožňuje vidět pouze ty pole, které jsou pro uživatele důležité.

Notesový dokument z pohledu uživatele vypadá pouze takhle

	To:	Jan Kojecy/Czech Republic
	Cc:	
	Bcc:	
	Subject:	Test-02372Y - Mailrouting issue
	Default custom expiration date: 03/23/2014 This message will be sent encrypted.	
<p>Hello,</p> <p>this is a test email to test mail routing regarding case number <u>02372Y</u></p> <p>Please don't reply,</p> <p>Thank you for understanding</p> <p>Best regards,</p> <p>Jan <u>Kojecy</u></p>		

Obr. 4 E-mail z pohledu uživatele.

Pokud chceme zjistit hodnotu některých polí v dokumentu, můžeme to jednoduše udělat tak, že označíme dokument a pravým tlačítkem myši klikneme na vlastnosti dokumentu.



Obr. 5 Vlastnosti dokumentu.

Jak už je ale vidět na samotném obrázku, je takovéto čtení polí velice nepřehledné. Proto se nabízí možnost použít nějaký nástroj na přehlednější zobrazení jednotlivých polí.

Pokud takovýto dokument zkopírujeme do speciální databáze zvané „Notes Document Viewer“ tak uvidíme i ostatní pole.

5.3.1 Nástroj „Notes Document Viewer“

Takhle vypadá notesový dokument v databázi zvané „Notes Document Viewer“

Notes Document Viewer [®] - Mayflower Software	
\$Abstract (TEXT) :	
\$AutoSpell (TEXT) :	0
\$devopt_basic_moods (TEXT) :	N
\$EncryptionStatus (TEXT) :	1
\$KeepPrivate (TEXT) :	
\$KeepSent (NOTEREFS) :	9875BC805D769D5EC1257B37006C7554
\$Mailer (TEXT) :	Lotus Notes Release 8.5.3 September 15, 2011
\$MessageID (TEXT) :	<OF9875BC80.5D769D5E-ONC1257B37.006C7554-C1257B37.006D5BEA@LocalDomain>
\$Moods (TEXT) :	N
\$Orig (NOTEREFS) :	9875BC805D769D5EC1257B37006C7554
\$Seal () :	
\$SignatureStatus (TEXT) :	0
\$StorageTo (TEXT) :	1
\$TUA (NOTEREFS) :	BBD2DAAD6A5CC8E9C1257B37006ED1A6
\$UpdatedBy (NAMES) :	CN=Jan Kojecky/OU=Czech Republic/O= ,CN=D06MC462/OU=06/OU=M/O= ;CN=Jan Kojecky/OU=Czech Republic/O=
AltFrom (TEXT) :	
BlindCopyTo (NAMES) :	

Obr. 6 Zobrazení dokumentu v „Notes Document Viewer“. [16]

Je zřejmé, že pole dokumentu zobrazené tímto nástrojem jsou mnohem přehlednější.

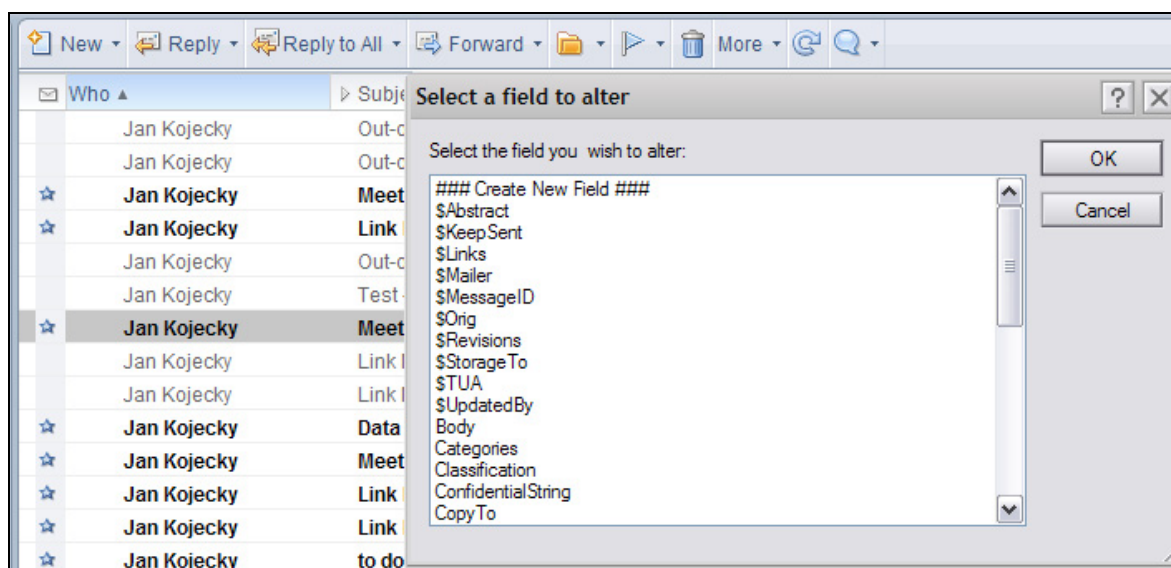
5.3.2 Tlačítko „Special Button“

Pomocí různých nástrojů můžeme tedy přehledněji zobrazit všechna pole v nějakém dokumentu.

Pomocí jednoduchého tlačítka „Special Button“ můžeme upravit hodnoty jednotlivých polí v dokumentu.

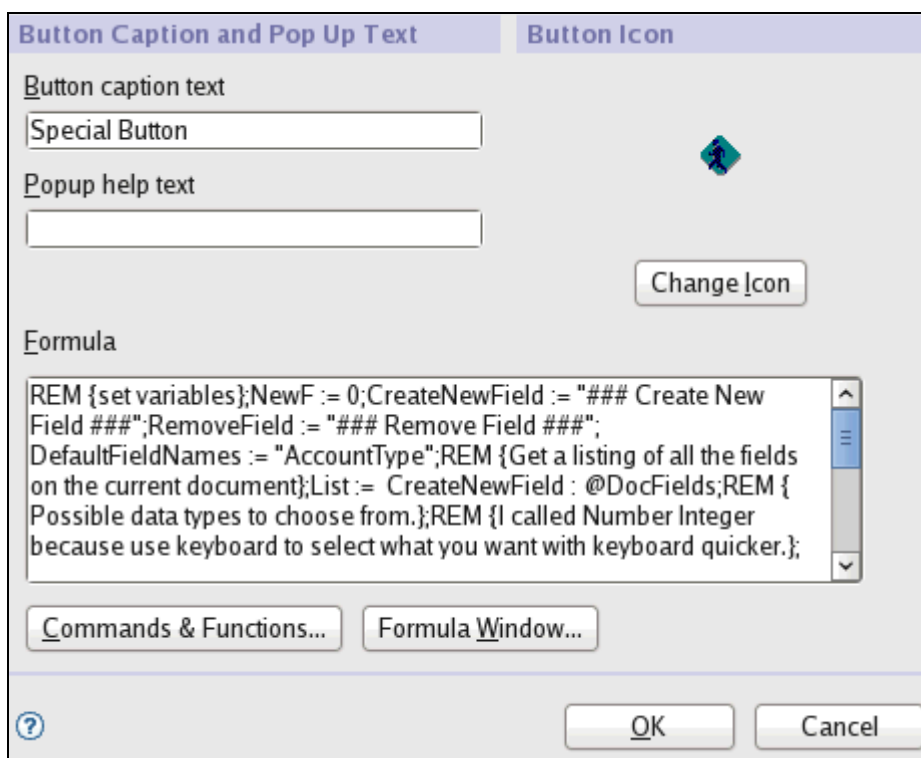
Jednoduše označíme příslušný dokument, klikneme na tlačítko „Special Button“ a zobrazí se nám seznam polí, které můžeme upravit. Můžeme taktéž vytvořit úplně nové pole, nebo vymazat nějaké existující.

Notesový dokument při úpravě pomocí „Special Button“



Obr. 7 Použití „Special Button“.

Na obrázku č. 8 vidíme ukázkou skriptu v jazyce vzorců.



Obr. 8 Ukázkou skriptu „Special Button“.

5.3.3 Nástroj „ScanEZ“

Pokud bychom chtěli vidět všechna pole a případně je upravovat, můžeme také použít nástroj „ScanEZ“. Pomocí „ScanEZ“ můžeme provádět následující akce:

- Nalezení problémového dokumentu v databázi;
- Porovnávání libovolného počtu dokumentů a následné zobrazení polí, kterými se liší;
- Hromadné úpravy dokumentů a polí;
- Lokalizování replikačních konfliktů a následné vyřešení;
- Přímý pohled do ACL (Access Control List).

Drag a column header here to group by that content.								
x/y	Name	N.	S..	D.	I..	Value		Modification date
ab	\$Abstract	1	1	0	44			23-03-2013 09:10:26 PM
ab	\$AutoSpell	1	1	0	22	0		23-03-2013 09:10:26 PM
ab	\$devopt_basic_...	1	1	0	5	N		23-03-2013 09:10:26 PM
ab	\$KeepPrivate	1	1	0	43			23-03-2013 09:10:26 PM
	\$KeepSent	1	1	0	10	[UNID] 9875BC805D769D5EC1257...		23-03-2013 09:10:26 PM
ab	\$Mailer	1	1	0	12	Lotus Notes Release 8.5.3 Septem...		23-03-2013 09:10:26 PM
ab	\$MessageID	1	1	0	13	<OF9875BC80.5D769D5E-ONC125...		23-03-2013 09:10:26 PM
ab	\$Moods	1	1	0	2	N		23-03-2013 09:10:26 PM
	\$Orig	1	1	0	17	[UNID] 9875BC805D769D5EC1257...		23-03-2013 09:10:26 PM
ab	\$StorageTo	1	1	0	11	1		23-03-2013 09:10:26 PM
	\$TUA	1	1	0	54	[UNID] BBD2DAAD6A5CC8E9C125...		23-03-2013 09:10:26 PM
	\$UpdatedBy	3	1	0	53	CN=Jan Kojecky/OU=Czech Repub...		23-03-2013 09:10:26 PM
1.2	_ViewIcon	1	1	0	7	0		23-03-2013 09:10:26 PM
ab	AltFrom	1	1	0	25			23-03-2013 09:10:26 PM
	BlindCopyTo	1	1	0	34			23-03-2013 09:10:26 PM
	Body	1	1	0	41	[rich text cannot be viewed in list] ...		23-03-2013 09:10:26 PM
ab	Categories	1	1	0	18			23-03-2013 09:10:26 PM
ab	Classification	1	1	0	46			23-03-2013 09:10:26 PM

Obr. 9 Notesový dokument. v zobrazení pomocí „ScanEZ“ nástroje. [15]

5.4 Nástroj „Violet UML Editor“

Popsat jednotlivé procesy slovy by bylo velmi složité. Proto jsem chtěl tyto procesy zvýraznit pomocí diagramů. Na vytvoření těchto diagramů jsem si vybral nástroj „Violet UML Editor“ [17]. „Na začátku by bylo určitě vhodné vysvětlit samotný pojem UML. UML neboli Unified Modeling Language je jazyk, pomocí kterého vizuálně modelujeme systémy. Nejedná se o metodiku (nestanovuje postupy vývoje), ale jedná o obecný a univerzálně použitelný jazyk. Základními stavebními bloky jazyka jsou předměty (elementy modelu), vztahy (pojítka mezi předměty) a diagramy (pohledy na modely UML). Diagramy chování slouží k popisu funkcionality systému. Konkrétnějším typem diagramů chování jsou diagramy interakce, které ukazují vzájemnou spolupráci jednotlivých komponent. Mezi diagramy chování patří diagram případů užití, stavový diagram a diagram aktivit.“ [13]. Nejvhodnějším typem diagramu pro moje účely byl diagram aktivit, který jsem nakonec zvolil.

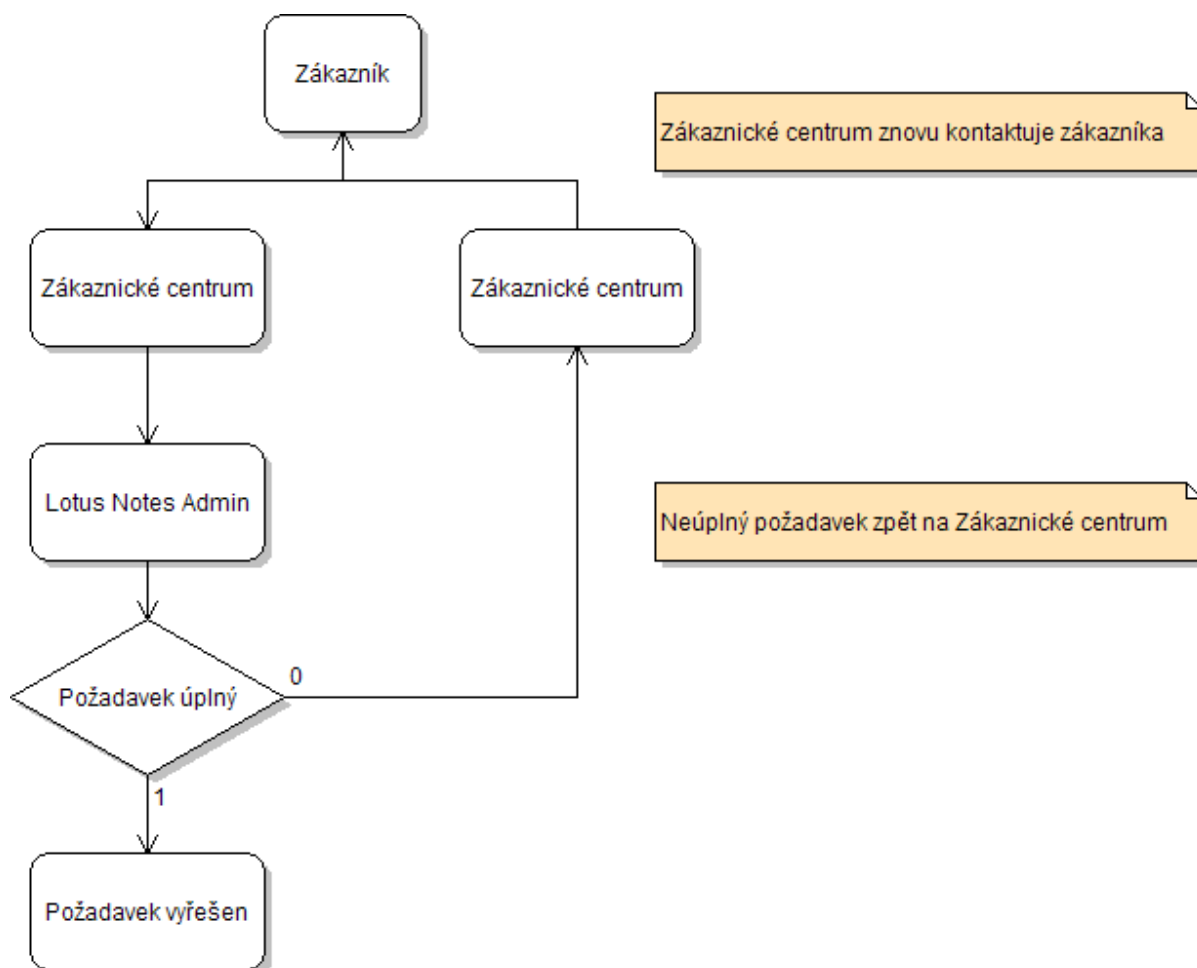
6. ANALÝZA

6.1 Analýza dokumentace na zákaznickém centru.

Jak už jsem se dříve zmínil, tak rozdělení práce na tři úrovně nám přináší velkou výhodu. Aby to ovšem fungovalo dle modelu, je potřeba aby každý tým v této tříúrovňové hierarchii jasně dodržoval pravidla hry.

Jelikož tikety přicházejí k nám na druhou úroveň od zákaznického centra, je potřeba nejprve analyzovat současný stav dokumentace a procesu přímo na první úrovni.

I když pracovníci zákaznického centra mají požadavky posoudit a zjistit, jestli je potřeba nějaké schválení, nebo jestli obsahuje všechny potřebné informace, tak jsme stále dostávali spoustu tiketů bez informací. Agenti zákaznického centra totiž většinou posílali tikety na náš tým a náš tým tam následně dával instrukce, jak se mají tyto tikety řešit. Takže agenti první úrovně využívali tuto možnost a pouze tikety posílali a čekali na detailní instrukce od týmu druhé úrovně. Počínání agenta bez dokumentace je vidět na obrázku č. 10.



Obr. 10 Počínání agenta bez dokumentace a následná cesta tiketu.

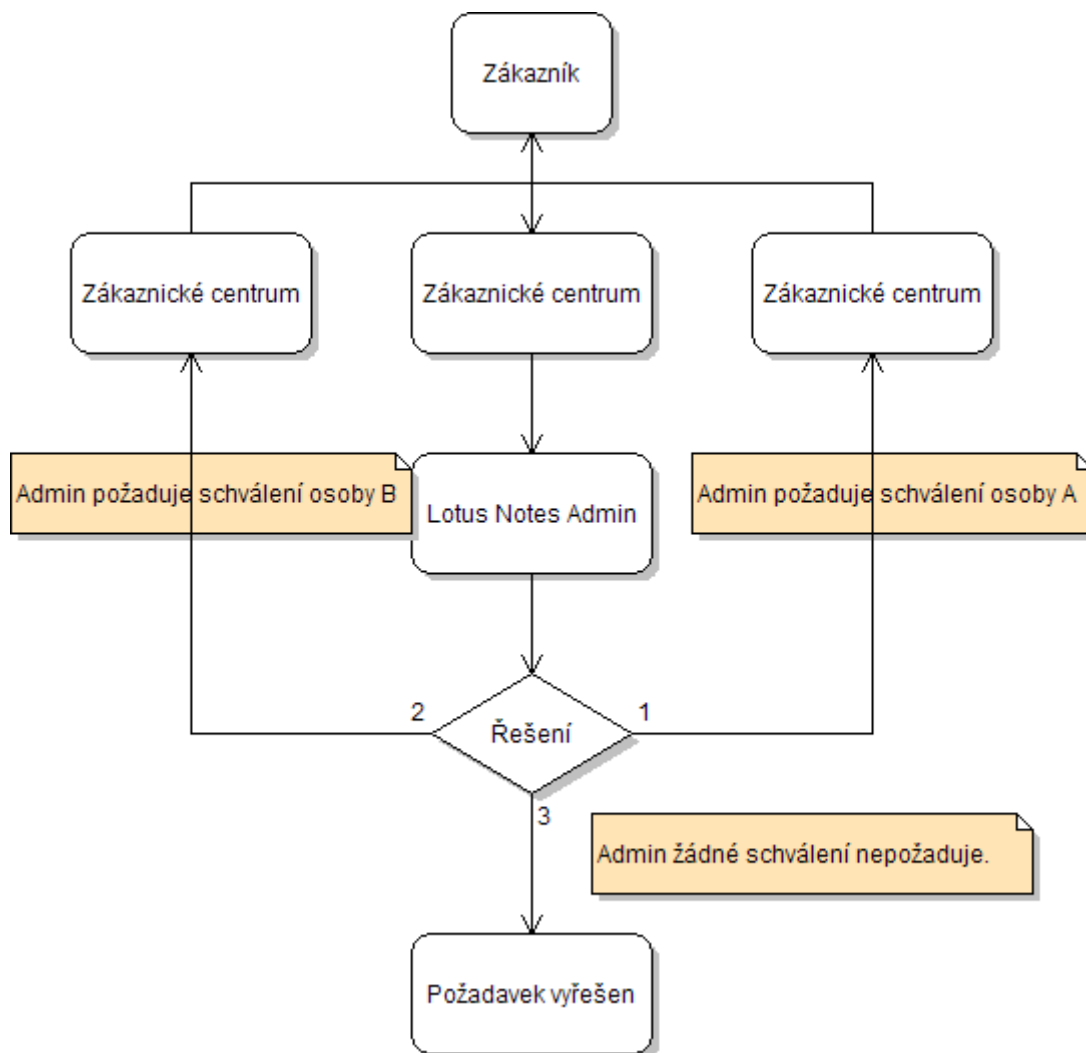
6.2 Analýza dokumentace na druhé úrovni

Když jsem přišel do týmu Lotus Notes administrace a začal se podílet na řešení klientských problémů, tak jsem využíval externí databáze v podstatě hledáním řešení, které už někdo v minulosti řešil.

Sice už i v této době existovala interní databáze, ale řešení jednotlivých klientských problémů nebylo ve většině případů detailně popsáno, nebo dokumenty úplně chyběly. Další věcí, co jsem si všiml, je, že ostatní administrátoři měly různý přístup k řešení jednotlivých problémů, či splnění nějakého požadavku od zákazníka. Abych to podrobněji vysvětlil – dejme tomu, že řešení, například notes klient problémů, může mít více řešení a bude částečně záležet na jednotlivém administrátorovi, jak se k tomu postaví a jaké řešení dohledá v nějaké externí databázi. Pokud každé řešení bude fungovat, pak je v podstatě jedno jaké řešení jeden či druhý administrátor použil.

Ale když si vezmeme kategorii tiketů (vytvořit databázi, smazat databázi, apod.), je potřeba, aby zákazník dostal vždy jasné, a stejné instrukce jakým způsobem se jeho požadavek řeší. Nejdůležitější kategorií těchto tiketů jsou požadavky na jednotlivé přístupy. Ať už se jedná o přístup do databáze, nebo práva rezervovat si nějakou zasedací místnost, nebo přístup do e-mailové schránky nějakého uživatele, který třeba onemocněl a nemůže někomu tento přístup sám delegovat. Dokážete si představit nespokojenost zákazníka, když jeden tiket byl vyřešen bez schválení příslušné osoby, a pokud se stejný tiket dostal k jinému agentovi, ten požadoval schválení od manažera, a například další administrátor požadoval schválení od osoby z personálního oddělení?

Na obrázku č. 11 vidíme proces počínání administrátora bez dokumentace.



Obr. 11 Počínání administrátora bez dokumentace a následná cesta tiketu.

Zákazník tedy vůbec neví, kdo má příslušný požadavek schválit, doba řešení tiketu se tímto samozřejmě prodlužuje.

Výsledkem takového způsobu podpory může být pouze nespokojenost zákazníka.

6.3 Analýza technické práce na druhé úrovni

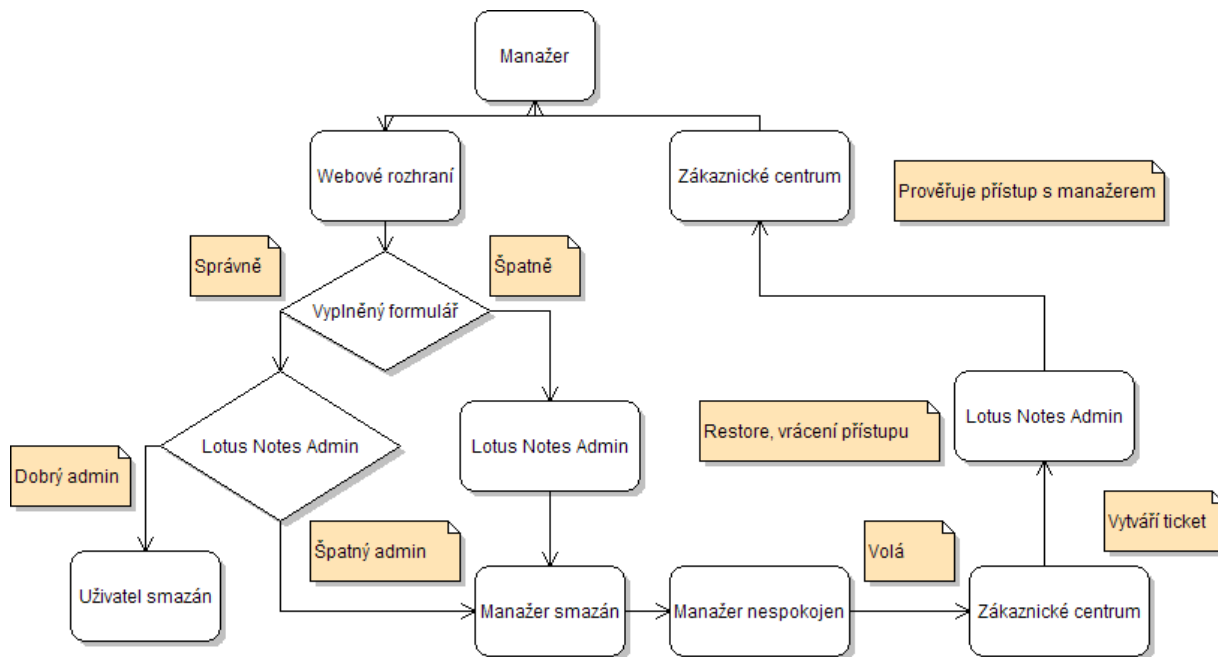
Na druhé úrovni pracujeme s hotovými databázemi, které většinu požadavků vykonávají automaticky. Zákazník už má vytvořené databáze například na celkovou administraci jednotlivých uživatelů nebo jednotlivých databázi.

Tyto databáze zpracovávají spoustu různých požadavků najednou (vytvoření uživatele, přejmenování uživatele atd.).

Všechny tyto globální nástroje a databáze jsou aplikovatelné pouze na uživatele v globální doméně, proto na podporu uživatelů v lokálních doménách je stále aplikována, řekněme manuální podpora, vytváření uživatele klasicky přes Lotus Notes administrátora apod. Toto vytváří spoustu opakované práce. Jakmile se zvyšuje opakovaná práce, tak dochází k většímu zatížení na samotné zaměstnance, a to může způsobit častější chyby apod.

6.3.1 Upozornění na smazání účtů VIP osob – současný stav

Jelikož se zabýváme i požadavky na smazání účtů uživatelů a provádíme spoustu tiketů za den, je potřeba se vyhnout jakýmkoliv chybám. První krok, jak se vyhnout těmto chybám, je mít správně zdokumentovaný proces a zvýrazněné části tiketu, na které je potřeba se zaměřit. Dále je důležité nespřést si žadatele s osobou, která má být smazána. V dokumentaci je samozřejmě popsáno, že je potřeba zjišťovat, jestli je uživatel aktivní či ne. To znamená, přistupoval-li v poslední době na server a na základě toho se rozhodnout, zda uživateli účet smazat či ne. Ale vždy existuje možnost, že pracovník zákaznického centra, či administrátor druhé úrovně udělá chybu a zadá špatné jméno do příslušného formuláře na smazání účtu. Na obrázku č. 12 je vidět původní proces s možnými důsledky.



Obr. 12 Mazání účtů VIP osob – současný stav.

Na základě těchto zmíněných chyb dochází ke smazání účtů samotných manažerů. I při následné rychlé reakci zákaznického centra a administrátora na vrácení účtu manažera zpět do produkce, dochází ke ztracení určitého počtu příchozích e-mailů a také ke ztracení přístupu do některých dokumentů. Jedná se konkrétně o hodnoty v autorských a čtenářských polích těchto dokumentů.

Při smazání celého účtu, je totiž jméno manažera smazáno ze všech přístupových bodů, ať už se jedná o přístupové skupiny, nebo přístupy na různé servery, nebo již zmíněné hodnoty v polích dokumentů.

Například v určité databázi jsou dokumenty, které vyžadují manažerovo schválení. Protože manažerovo jméno bylo ale vymazáno ze čtenářských polí těchto dokumentů, tak jednoduše nastane situace, že manažer tyto dokumenty v určité databázi vůbec neuvidí, takže k žádnému schválení nedojde a to má samozřejmě další důsledky.

6.3.2 Analýza procesu při přejmenování uživatele a následná úprava profilu v databázi personálního oddělení

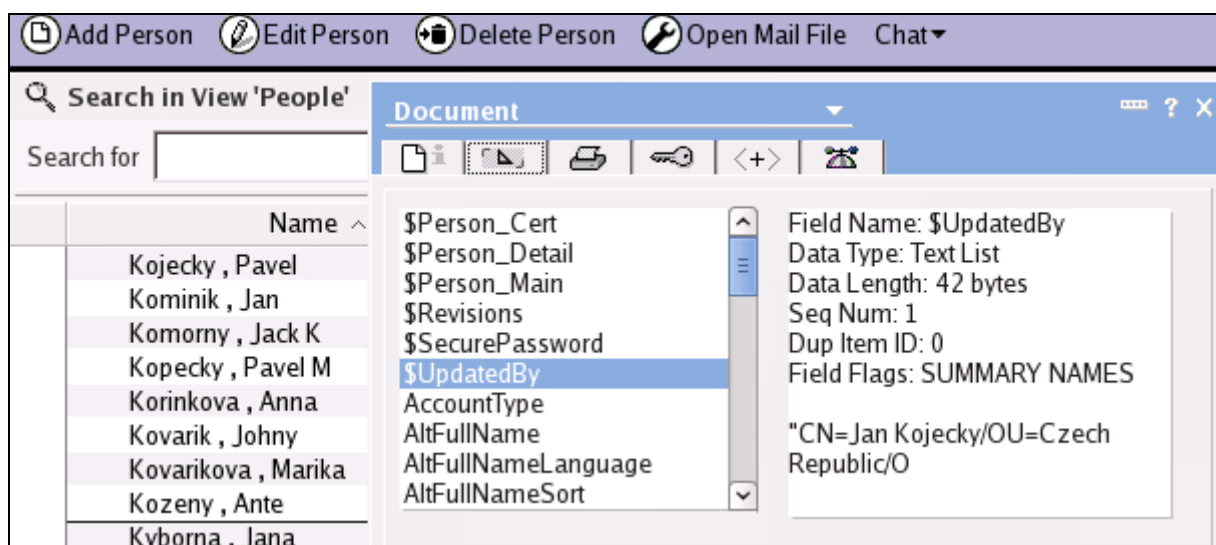
Při schvalování požadavků v databázi na administraci uživatelů, jsme byli požádáni zákazníkem jedné země, že je potřeba při přejmenování uživatele následně doplnit informace v databázi personálního oddělení. V jiných zemích existuje propojení mezi týmem, který požadavek na změnu jména vytváří, a personálním oddělením. Nejdříve jsme se tedy snažili vykomunikovat se zákazníkem z dané země, aby se přizpůsobil globálním pravidlům. Ovšem výsledek této komunikace byl negativní.

Začali jsme tedy při každé změně jména informovat personální oddělení prostřednictvím e-mailu. To znamená, jakmile administrátor zaznamenal, že požadavek na změnu jména pochází z dané země, tak jednoduše napsal e-mail personálnímu oddělení. Docházelo ovšem k mnoha chybám. Jednak si administrátor při schvalování většího počtu požadavků, takového požadavku vůbec nevšiml a jednak, když si všiml tak e-mail neposlal ihned a později na to třeba zapomněl.

Dostávali jsme tedy často reklamace, že informace nejsou včas a kompletně předávány personálnímu oddělení.

6.3.3 Analýza zjištění kdo poslední upravil dokument

Při řešení jakéhokoliv problému první analýza spočívá v kontrole hodnot různých polí v příslušných dokumentech. Velmi důležité je zjistit, kdo a kdy tyto hodnoty v dokumentu nastavil. V dokumentu se nachází pole „UpdatedBy“



Obr. 13 Pole v dokumentu „UpdatedBy“.

Úkon, při němž administrátor zjišťuje, kdy byl naposledy příslušný dokument upraven a kdo se podílel na této úpravě, se provádí několikrát denně.

Zjišťovat to z vlastností dokumentu „UpdateBy“ je velice zdlouhavé a nepřehledné.

6.3.4 Analýza přidávání uživatelů nebo úprava polí v jednotlivých dokumentech v jakékoliv databázi

Velmi často je potřeba upravit nějaká pole v dokumentu. Například při rezervaci zasedacích místností. Přichází nám požadavek od majitele nějaké místnosti, abychom dali přístupová práva na možnost rezervace dalším uživatelům. Takže administrátor upraví příslušný dokument a zapíše notesové jméno uživatele do určitého pole tohoto dokumentu. To by bylo v pořádku, ale stále více přichází tiketů na to, aby uživatel měl práva rezervovat zasedací místnosti na celé lokaci příslušné firmy. Může se tedy jednat například o dvacet místností, takže je potřeba upravit dvacet dokumentů. Upravovat ale dvacet dokumentů tím způsobem, že každý otevřeme a následně upravujeme je z časového hlediska nepřijatelné.

6.3.5 Analýza nastavení tlačítek

Každý člen našeho týmu má vytvořenou tlačítkovou lištu a na ní má různá tlačítka, která si buď vytvořil sám, nebo převzal skript na tyto tlačítka od jiného administrátora.

Proces je současně nastaven tak, že když přijde do našeho týmu nováček, dostane e-mail, kde jsou jednotlivé skripty pro různá tlačítka. Tyto skripty si administrátor následně zkopíruje do designu a nastaví si pro každý skript tlačítko v tlačítkové liště.

Toto řešení se jeví poněkud zdlouhavé, protože administrátor potřebuje používat větší počet tlačítek. Dále je absolutně nepřijatelné pro samotného zákazníka. Kdybychom mu poslali skripty v textovém formátu a instrukcemi, jak si má tyto jednotlivá tlačítka zprovoznit, tak jednak by nevěděl jak to má sám nastavit, a jednak není z časového hlediska možné, aby lokální administrátor zprovoznil tlačítka pro větší počet uživatelů.

6.3.6 Analýza migrace uživatele z lokální do globální domény a naopak

Někdy dochází k situaci, že se uživatel velké mezinárodní společnosti stěhuje do jiné země, nebo je vedením firmy určen na pozici v jiné pobočce této firmy, která se ovšem nachází v jiném státě. Pokud se oba státy nachází v globální doméně, tak je takovýto přesun jednoduchý. Ke komplikacím ovšem dochází při přesunu uživatele mezi lokální a globální doménou.

Původní proces přesunu uživatele z lokální domény do globální probíhal tak, že se nechal smazat starý účet v lokální doméně a následně se uživateli vytvořil nový účet v globální doméně. Pokud uživatel potřeboval nějaká data z předchozího účtu, tak použil lokální kopii e-mailové schránky. Protože by bylo pro zákazníka mnohem pohodlnější používat původní e-mailovou schránku i nadále, tak zákazník vznesl požadavek, abychom začali používat řešení kompletní migrace.

Uživatele je potřeba zmigrovat z lokální domény do globální. Protože je potřeba udělat hodně kroků, a následně každý krok komunikovat s lokální podporou, je potřeba komunikovat každou akci e-mailem s lokální podporou. Tiketovací systém v tomto případě ke komunikaci nelze použít, protože se tento systém v lokálních doménách nepoužívá, tudíž není možné takovýto tiket poslat na lokální podporu.

Rovněž je potřeba zmínit, že celý migrační proces může trvat i několik dní. To znamená, že na jedné migraci musí pracovat i více administrátorů, jak z našeho týmu, tak i z lokální podpory.

Lokální i globální administrátoři mají různé úkoly, které musejí splnit v určitém časovém sledu. Někdy nelze jeden úkol udělat, dokud druhý administrátor předchozí úkol

nedokončí. V nejhorším případě se může stát, že takovýto úkol je technicky možné udělat, ale ještě nebyl dokončen úkol předchozí. Například lokální administrátor smaže dokument uživatele z lokálního adresáře a v globálním adresáři jsou e-maily stále přesměrovány na lokální doménu. To znamená, že všechny příchozí e-maily jsou posílány do lokálního adresáře, ale zde už nemají být komu doručeny, protože se příslušný dokument uživatele v lokálním adresáři prostě nenachází.

Stávalo se nám tedy, že nebyly doručeny některé e-maily, nebo uživateli nefungoval účet tak, jak by měl, například blackberry nebylo správně nastavené atd.

To samozřejmě vedlo k nespokojenosti zákazníka, a protože se vesměs jednalo o VIP osoby, pak nastával velký tlak na naše vedení a tím i na náš tým.

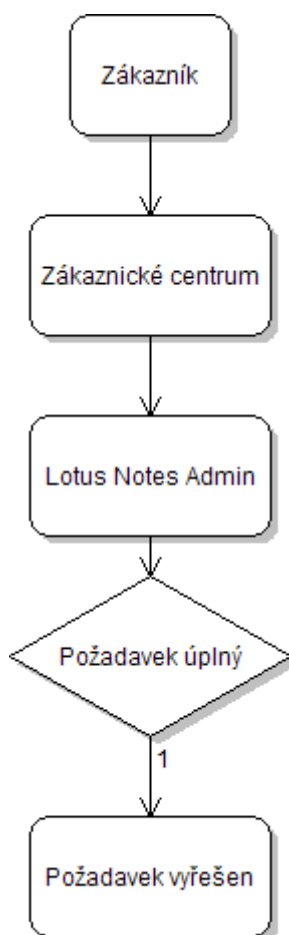
7. NÁVRH RACIONALIZOVANÉHO ŘEŠENÍ

7.1 Návrh dokumentace na zákaznickém centru.

Nejlepším řešením by bylo dosáhnout toho, aby se tikety řešili co nejrychleji, to znamená, aby cesta takového tiketu byla co nejkratší, v nejlepším případě by tiket vyřešil pracovník první úrovně.

Cílem je tedy zvětšit procento tiketů vyřešených přímo na zákaznickém centru. V závislosti na tom snížit počet tiketů, které cestují na druhou úroveň podpory.

Dalším cílem je zmenšit množství tiketů bez dostatečných informací, tím by se zmenšil počet transferů těchto tiketů mezi týmy první a druhé úrovně. Administrátor druhé úrovně by totiž nemusel tento tiket posílat zpátky na zákaznické centrum, aby pracovník zjistil potřebné informace od zákazníka, ale pracovník by rovnou tiket vyřešil. Na obrázku č. 14 vidíme, jak by takovýto proces měl vypadat.



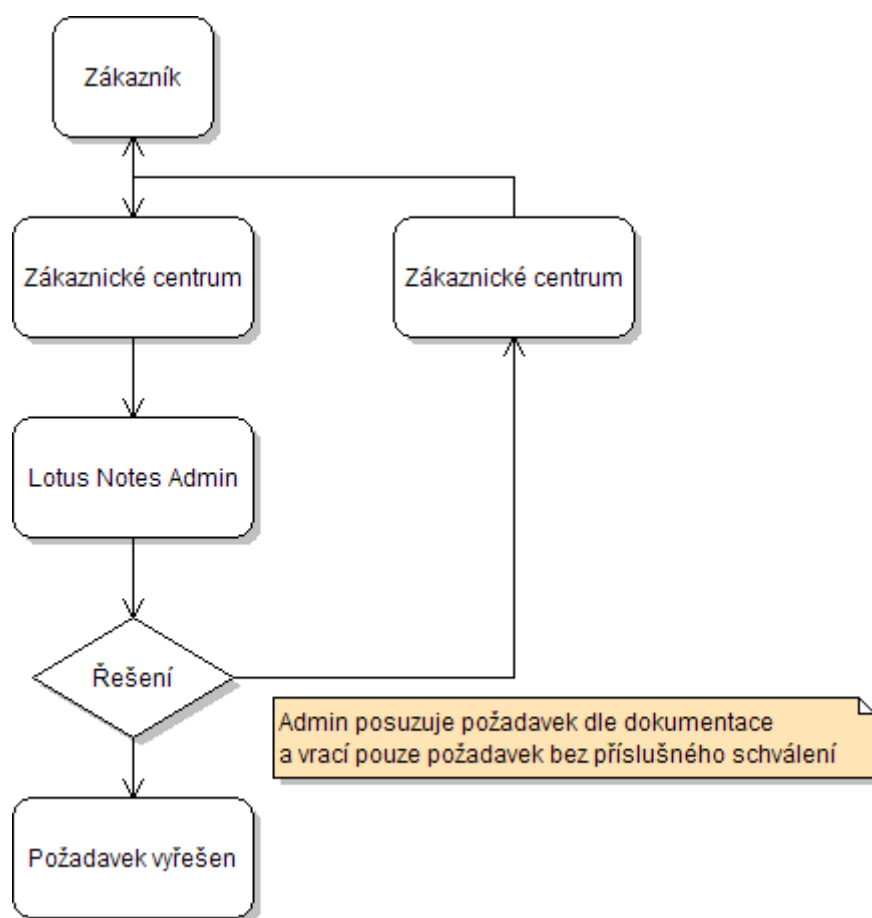
Obr. 14 Proces s dokumentací na první úrovni.

7.2 Návrh dokumentace na druhé úrovni

Bylo potřeba s tím tedy něco udělat, aby naše práce byla prezentována zákazníkovi jako práce týmu, který má jasně definované procesy, kterých se drží a klade se zde důraz zejména na zabezpečení různých dokumentů, či soukromých e-mailů.

Takže cílem je nejen udělat příslušné dokumenty na všechny procesy, ale také je nasdílet ostatním lidem v týmu. Zároveň pomůžou příchozím nováčkům k rychlejší integraci do týmu.

Na obrázku č. 15 vidíme, jak postupuje administrátor na základě příslušné dokumentace.



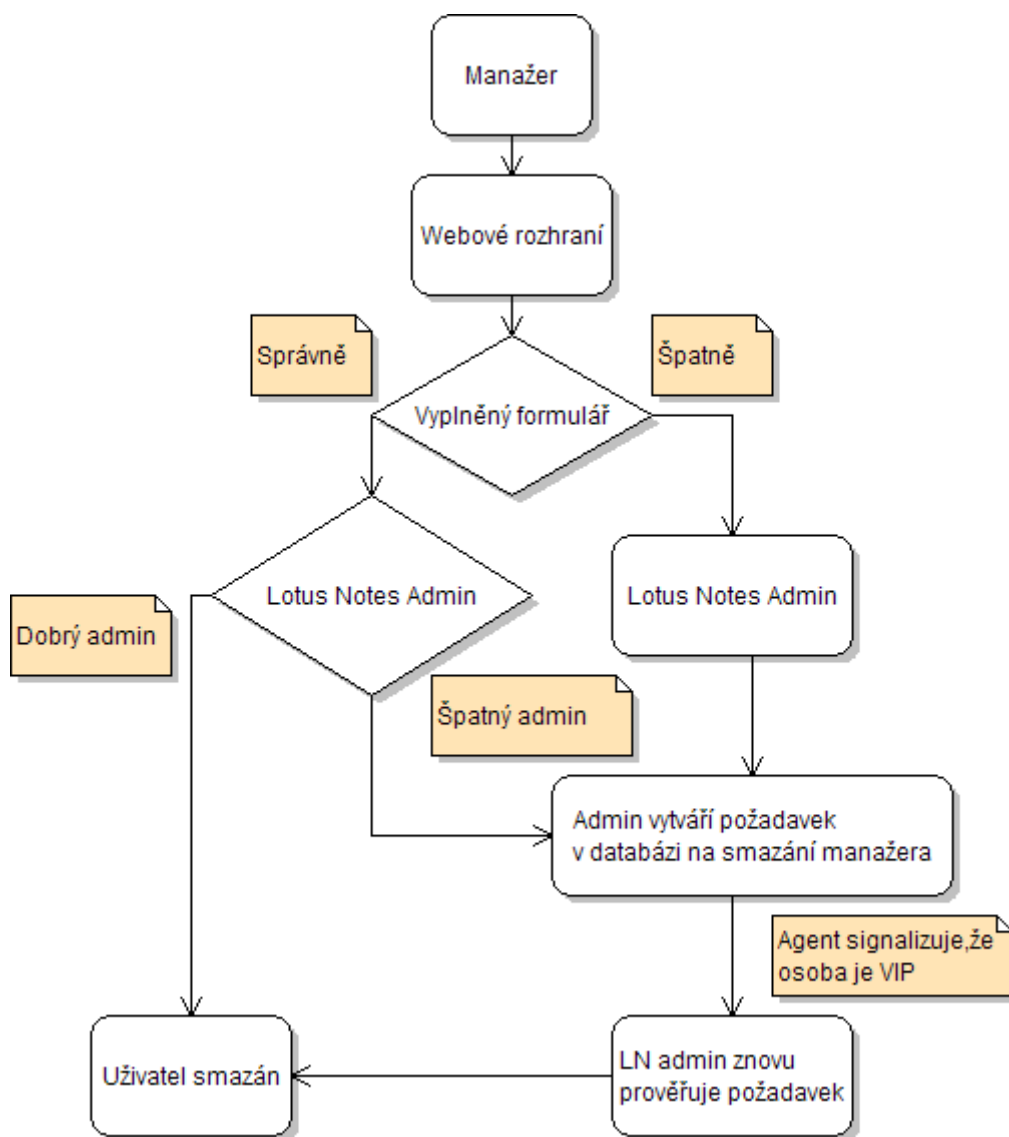
Obr. 15 Proces s dokumentací na druhé úrovni.

7.3 Návrh technické práce na druhé úrovni

7.3.1 Návrh upozornění na smazání účtů VIP osob

Cílem je samozřejmě zmenšit počet chyb při procesu smazání účtů, protože v případě chyby je důsledkem smazání účtu VIP osoby.

Bylo by dobře upozornit v databázi určené na správu uživatelů na takovýto požadavek, respektive nějak ho zvýraznit. Na obrázku. 16 vidíme proces, když agent upozorní administrátora.



Obr. 16 Proces s upozorněním na VIP osoby.

7.3.2 Návrh informovat personální oddělení při přejmenování uživatele

Samozřejmě, že by bylo nejjednodušší nastavit konsolidaci do této příslušné databáze, ale to se ukázalo jako nemožné.

Bylo proto potřeba stávající řešení nějak automatizovat. Nejprve bylo potřeba upozornit administrátora na tento požadavek a následně zautomatizovat posílání e-mailů personálnímu oddělení.

7.3.3 Návrh na zjištění, kdo upravoval dokument

Bylo potřeba najít rychlejší a přehlednější způsob na zjištění informace, kdo upravoval příslušný dokument.

7.3.4 Návrh na přidávání uživatelů nebo úprava polí v jednotlivých dokumentech v jakékoliv databázi

Cílem je upravit jakýkoliv počet dokumentů, aniž bychom museli vytvářet agenta v designu a pokaždé tohoto agenta upravovat v závislosti na změně jmen přidávaných uživatelů.

Je potřeba najít řešení rychlé, a také aby formát vložených hodnot do jednotlivých polí, byl správný. Takže je potřeba agenta nadefinovat tak aby nebylo možné vložit libovolný text, ale aby jméno uživatele muselo být vybráno z globálního adresáře.

7.3.5 Návrh řešení automatické distribuce uživatelsky definované tlačítkové lišty

Bylo potřeba najít řešení, které by všem novým členům našeho týmu, nebo jednotlivým uživatelům nastavilo tlačítkovou lištu se všemi potřebnými tlačítky.

Cílem je tedy poslat uživateli pouze jeden e-mail s jedním tlačítkem, po jehož stisknutí se mu automaticky vytvoří nová tlačítková lišta, obsahující všechna potřebná tlačítka.

7.3.6 Návrh řešení na migraci uživatele

Začal jsem přemýšlet, jak docílit toho, aby tento migrační proces mohl probíhat rychle a úspěšně. První myšlenkou, která mě napadla, bylo, že by globální administrátoři dostali přístup do lokálních domén a mohli tedy celý proces zvládnout sami a nepotřebovali vůbec s lokálními administrátory tento proces komunikovat.

Toto eventuální řešení ale nebylo možné z procesního hlediska provést. Je potřeba zmínit, že lokální administrátoři jsou přímými zaměstnanci zákazníka a ne tedy zaměstnanci firmy, která mu celkovou podporu zajišťuje. A proto je tedy nutné z hlediska přístupů zachovat výchozí nastavení, to znamená, že do globální domény mají přístup pouze globální administrátoři a do lokálních domén pouze administrátoři lokální.

Jediné možné řešení tedy je zdokonalit vzájemnou spolupráci lokálních a globálních administrátorů. Podrobně tedy povinnost administrátora zaznamenat každý krok a časově sladit jednotlivé úkoly jednotlivých administrátorů.

8. REALIZACE ŘEŠENÍ

8.1 Vytvoření dokumentace na první úrovni

Bylo potřeba udělat revalidaci současných dokumentů ve všech interních databázích zákaznického centra. Dále případně vytvořit všechny potřebné dokumenty na všechny příslušné kategorie tiketů.

Proto jsem následně komunikoval s vedením zákaznického centra, abychom dali přístup všem pracovníkům, aby využily příslušné dokumenty z naší interní databáze. Dostali jsme se ale do problémů, že zákaznické centrum už mělo svou interní databázi a z časových důvodů pracovníci nemohli používat více různých databází. Proto jsem si zažádal o přístupy do jednotlivých databází na všech zákaznických centrech a zkontroloval dokumenty v těchto jejich databázích.

Následně jsem analyzoval, jestli tyto tikety stále přicházejí bez informací a bez příslušných schválení a překvapivě výsledek nebyl vůbec potěšující.

Zaměstnanci zákaznického centra totiž pořád dostávali detailní instrukce v tiketech a tak namísto hledání v jejich interní databázi stačilo tiket poslat na druhou úroveň a tiket přišel zpátky s detailními instrukcemi. To bylo z hlediska rychlosti vyřešení tiketu neakceptovatelné řešení. Takže jsem vytvořil speciální dokument v naší interní databázi. Tento dokument obsahuje jednoduchou tabulku všech druhů požadavků a zároveň dokument ID příslušného dokumentu v interních databázích jednotlivých zákaznických centrech.

Takže administrátor druhé úrovně už do tiketů, který vrací zpět, nezasílá detailní popis, jak a co se má udělat, ale pouze jednoduchou poznámku: „prosím postupujte dle procesu popsaného v dokumentu Z-10XXYY“. Tímto je zaměstnanec první úrovně přinucen si otevřít vlastní interní databázi a zároveň se takto automaticky zaškoluje na využívání této interní databáze. Automaticky se tímto zmenší počet transferů tiketu.

8.2 Vytvoření dokumentace na druhé úrovni

Protože jsem si všiml, že se problémy s Lotus Notes vlastně opakují tak jsem si začal vytvářet detailní dokumenty na řešení jednotlivých problémů.

Začal jsem vlastně vytvářením Word dokumentů, které jsem uspořádal do různých složek. Složky byly pojmenovány podle různých kategorií problémů. Jednotlivé názvy dokumentů obsahovali chybové hlášky, součástí dokumentů byly také chybové hlášky a detailní popis problémů, dále popis řešení z pohledu administrátora, a pokud se jednalo o lokální problém, tak detailní instrukce pro zaměstnance první úrovně, jak tyto problémy řešit. To bylo pro mé potřeby dostačující.

Protože ale pracujeme v týmu a tento tým provádí podporu 24 hodin denně, bylo potřeba tyto znalosti nasdílet celému týmu. Jak jsem si v budoucnosti všiml, sdílení těchto informací byla vlastně klíčová věc, která vedla k jednotnému přístupu řešení jednotlivých problémů či požadavků. Výsledek tohoto bylo jednak zrychlení práce jednotlivých agentů, ale také jasné a detailní instrukce zákazníkovi.

Jelikož v týmu se začali objevovat další nováčci a já jsem je dostal na starosti zaučit, tak jsem s nimi začal sdílet tyto dokumenty. To bylo po nějaký čas také dostačující, ale samozřejmě jelikož dokumenty nebyly sdílené, tak jakákoliv úprava těchto dokumentů nebyla nasdílena zbytku týmu. Začal jsem tedy přemýšlet o vytvoření Lotus notes

databáze, která by byla ideální na shromažďování těchto dokumentů a zároveň přístupná, nejen pro členy našeho týmu, ale také pro členy jiných týmů. Protože už ve firmě existovala šablona na takovouto databázi, tak jsem pouze zažádal o vytvoření této databáze pro náš tým. Zároveň jsem byl zvolen majitelem této databáze a tím jsem měl pod kontrolou přidávání přístupů pro členy našeho týmu a také pro další členy z jiných týmů.

V databázi jsem vytvořil strukturu hlavních kategorií. Dále jsem převedl moje lokální dokumenty do této databáze. Pro zjednodušení jsem ze začátku pouze přidal tyto dokumenty jako přílohu do nově vytvořených notes dokumentů, ale následně se to ukázalo jako nevýhodné, protože nešlo používat „text free search“. Je to tím, že „text free search“ neprohledává obsah jednotlivých příloh, takže jsem musel tyto Word dokumenty převést do Lotus notes formátu.

Z hlediska bezpečnosti je nejdůležitější zabezpečení a dodržování standardů. Na tuto kategorii jsem vytvořil speciální dokument v naší databázi nazvaný schválení a normy „Approvals and Standards“. Tento dokument popisuje všechny přístupy a zároveň je to takový výtah z norem naší podporované firmy. Ve standardech jsou definované například jména uživatele, syntax jména databázi, přesměrování e-mail adresy do externí domény apod. Díky tomuto dokumentu koncový uživatel vždy dostane jasné instrukce, například s jakým jménem může být uživatel vytvořen a zároveň dostane link do firemních standardů s příslušnou kapitolou. Tyto firemní normy jsou přístupné každému uživateli podporované firmy přes intranet.

Vytvořením dokumentů práce neskončila. Vždy když jsem vytvořil nový dokument, bylo potřeba e-mailem informovat ostatní členy týmu, k jakým změnám došlo apod. I když byla vytvořena příslušná dokumentace, ne všichni administrátoři ji buďto používali nebo nevěděli, kde odpovídající dokument hledat.

Protože by bylo velice naivní si představit, že si každý administrátor bude brát sám požadavky z řady, a bude sám aktivně splňovat každou další povinnost, je potřeba práci v týmu efektivně rozdělit.

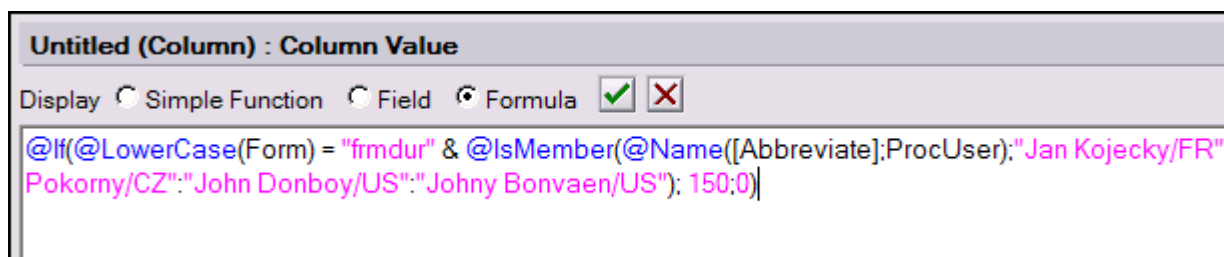
Práce lotus notes administrátora je strašně komplexní, takže kromě správné dokumentace je také důležité rozdělení jednotlivých povinností v týmu. Proto v týmu už používáme tabulku, ve které vedoucí na týdenní bázi rozdělí různé povinnosti jednotlivým administrátorům. Do této tabulky, kde jsou rozepsány různé povinnosti, jsem tedy přidal ještě jeden sloupec. Do tohoto sloupce jsem umístil název dokumentu a zároveň odkaz na tento dokument. Zodpovědné osobě pro daný úkol tedy stačí kliknout na příslušný odkaz a otevře se mu přímo dopovídající dokument pro danou problematiku.

8.3 Realizace technické práce na druhé úrovni

8.3.1 Upozornění na smazání účtu VIP osoby

Rozhodl jsem pro řešení nějak upozornit administrátora na VIP osobu. Tohle řešení bylo uděláno jednoduchým sloupcem v pohledu, do kterého jsem dal seznam všech VIP osob, takže pokud je tato osoba v seznamu, tak se v databázi vedle požadavku na smazání objeví upozornění „!“.

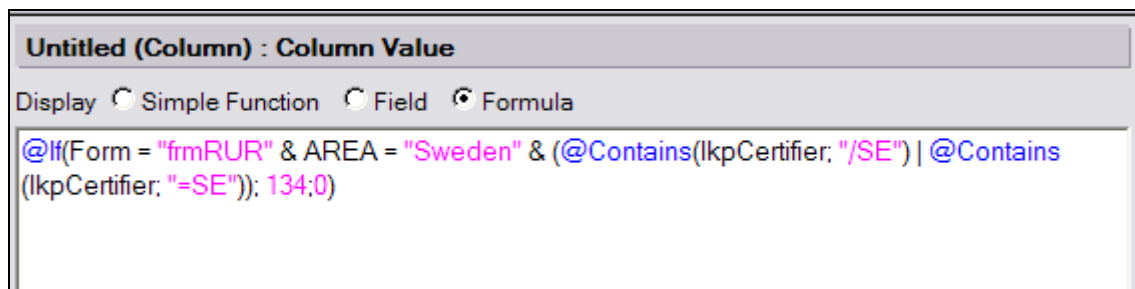
Na obrázku č. 17 je ukázka skriptu upozornění na VIP osoby.



Obr. 17 Ukázka skriptu upozornění na VIP osoby.

8.3.2 Vytvoření agenta „Informuj HR“

Nejprve jsem zase vytvořil sloupec v pohledu, který upozorní administrátora, že se jedná o požadavek na přejmenování účtu z dané země. Takže u každého požadavku tohoto typu se objeví červená ikonka/obálka, upozorňující administrátora, že je potřeba udělat nějakou akci. Na obrázku č. 18 je ukázka skriptu upozornění na přejmenování uživatele.



Obr. 18 Ukázka skriptu upozornění na přejmenování uživatele.

Dále jsem vytvořil jednoduchého agenta, který pošle e-mail informující o změně jména daného uživatele na příslušné personální oddělení.

Domino administrátor pouze vybere příslušný požadavek a označí ho a následně spustí agenta. Na obrázku č. 19 je ukázka skriptu agenta „Informuj HR“

```
△ SE Name Change Notification - Initialize

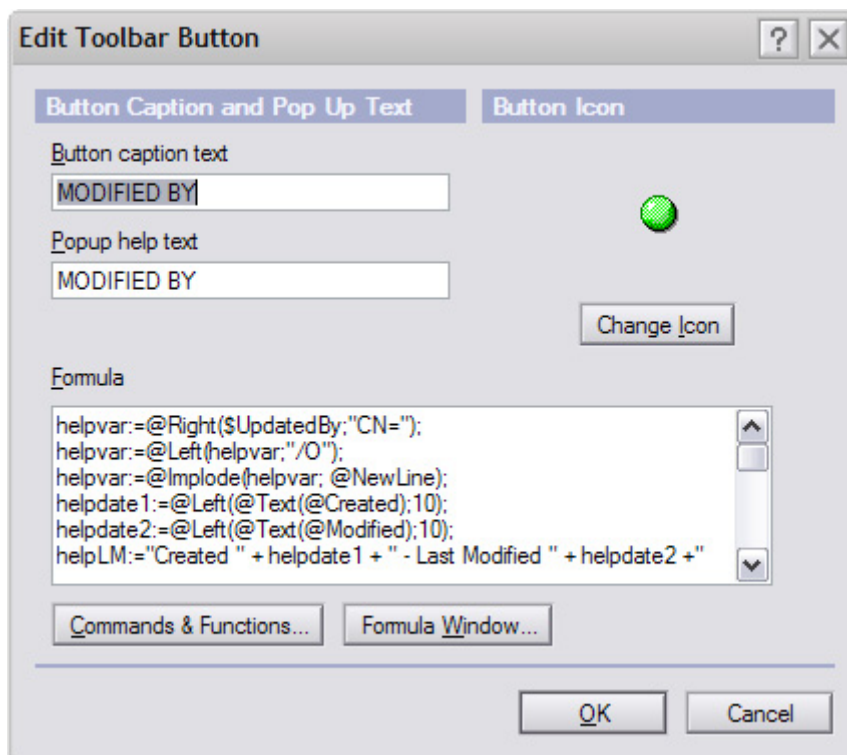
memodoc.Form="Memo"
memodoc.Subject="The LN account: " + nam.Abbreviated + " was renamed."
memodoc.SendTo="HR-sweden@se.com"

Set richStyle = session.CreateRichTextStyle
Set body = New NotesRichTextItem( memodoc, "Body" )
Call body.AppendText( "Hello," )
Call body.AddNewline(1)
Call body.AddNewline(1)
Call body.AppendText("The LN account: " + nam.Abbreviated + " was renamed.")
Call body.AddNewline(1)
Call body.AppendText("New Name: ")
richStyle.Bold = True
Call body.AppendStyle(richStyle)
Call body.AppendText(doc.NEWNAME(0))
Call body.AddNewline(1)
richStyle.Bold = False
Call body.AppendStyle(richStyle)
Call body.AppendText("New Internet Address: ")
richStyle.Bold = True
```

Obr. 19 Ukázka skriptu Agentu „Informuj HR“.

8.3.3 Vytvoření tlačítka „Modified by“

Vytvořil jsem tedy tlačítko „Modified by“ Stačí pouze označit příslušný dokument a zmáčknout tlačítko „Modified by“ a zobrazí se nám datum, kdy byl dokument vytvořen a také kdo všechno prováděl jakékoliv úpravy v dokumentu. Na obrázku č. 20 je ukázka skriptu „Modified by“



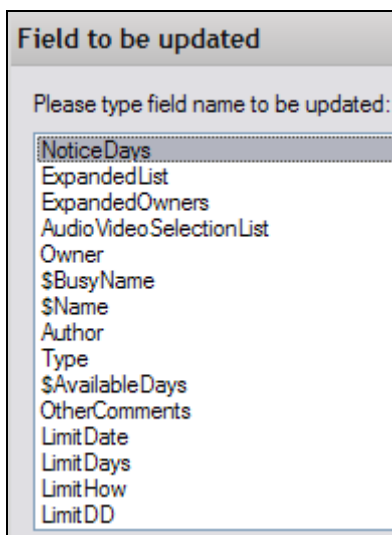
Obr. 20. Ukázka skriptu „MODIFIED BY“.

8.3.4 Vytvoření agenta „Add/Remove“

Vytvořil jsem agenta, pomocí něhož můžeme přidat či odebrat uživatele z většího počtu dokumentů.

Na příkladu je ukázáno využití agentu v databázi na rezervaci zasedacích místností.

1. Označíme dokumenty, které chceme změnit.
2. Vybereme, jakou akci chceme provést. (přidat, odebrat, přejmenovat)
3. Vybereme, jaké pole chceme upravit. (viz. obr. 21)



Obr. 21 Výběr pole při použití agenta „Add/Remove“.

4. Vybereme uživatele z adresáře, kterého chceme přidat či odebrat.

Na obrázku č. 22 je ukázka skriptu tlačítka „Add/Remove“

```

REPLACE, ADD , USER - (Options)

UserToBeAdded = workspace.PickListStrings(PICKLIST_NAMES)

While Not(doc Is Nothing)
  IsMember=False
  CurrGroup = doc.GetItemValue(FieldToBeUpdated)
  Forall aname In CurrGroup
    If Not Strcompare(aname, UserToBeAdded(0), 4) = 0 Then
      IsMember = False
    Else
      IsMember = True
      Exit Forall
    End If
  End Forall

  If IsMember =False Then
    NewGroup = Arrayappend(CurrGroup,UserToBeAdded(0))
    Call doc.ReplaceItemValue(FieldToBeUpdated, NewGroup)
    Call doc.Save(False, False)
  End If

  Set doc = currcollection.GetNextDocument(doc)
Wend
Exit Sub

Remove_user:

UserToBeRemoved = workspace.PickListStrings(PICKLIST_NAMES)

```

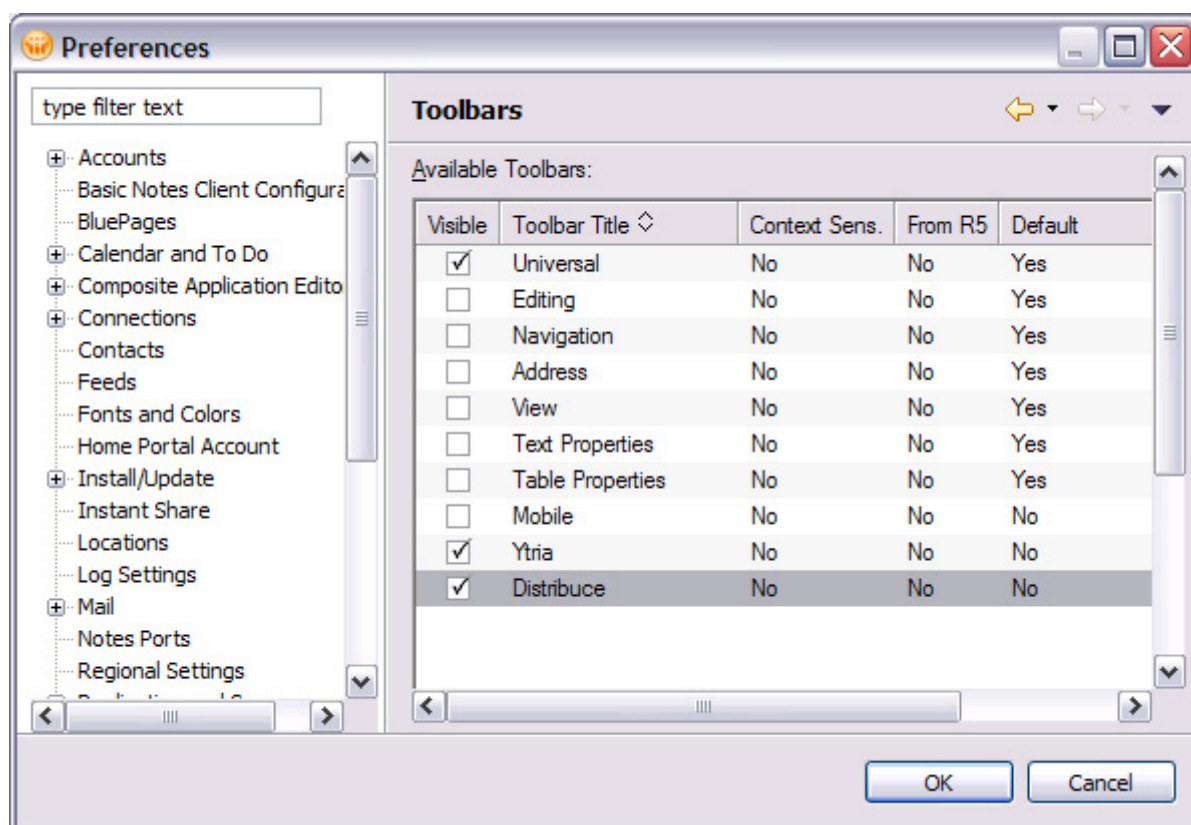
Obr. 22 Ukázka skriptu tlačítka „Add/Remove“.

8.3.5 Realizace automatické distribuce uživatelsky definované tlačítkové lišty

Na jednoduchém příkladu budu demonstrovat celý proces, na konci kterého je vytvoření tlačítkové lišty u nového člena v administračním týmu, nebo u nějakého koncového zákazníka.

Jako příklad jsem zvolil následující tlačítka – „Special Button“, „Updated by“, „Expiration date“.

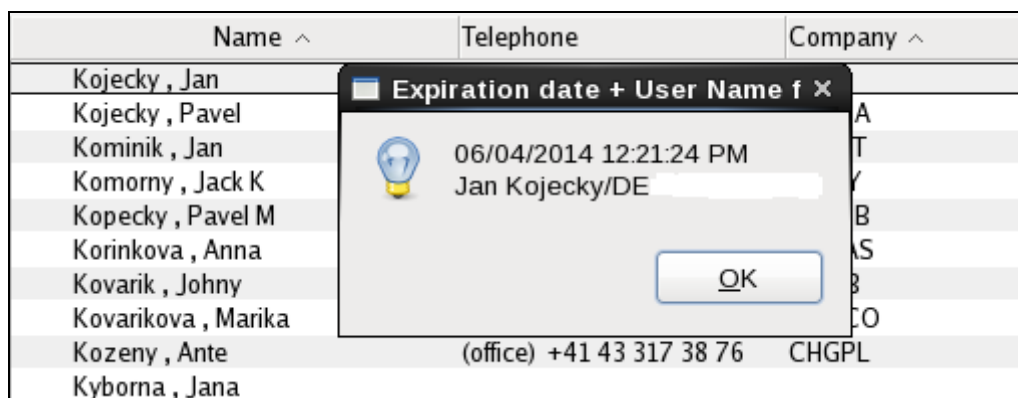
Nejdříve jsem vytvořil tlačítkovou lištu „Distribuce“ (viz obr. 23) na svém klientu, ve kterém jsem vytvořil příslušná tlačítka.



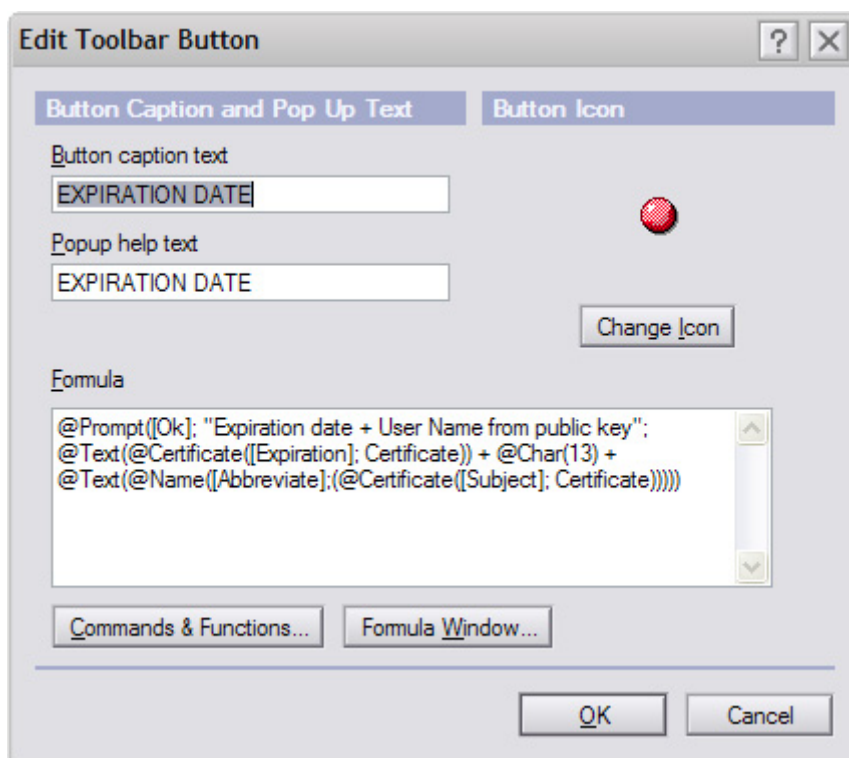
Obr. 23 Tlačítková lišta „Distribuce“.

Například tlačítko „Expiration Date“. Používáme ho velice často, když potřebujeme zjistit, kdy vyprší lhůta nějakému účtu a je potřeba ho recertifikovat přes Domino Administrátora. Zároveň nám dává důležitou informaci o jméně příslušného účtu. Nezobrazuje hodnotu z pole „Jména“ ale informaci načítá z veřejného klíče, který je uložen v každém osobním dokumentu uživatele v adresáři.

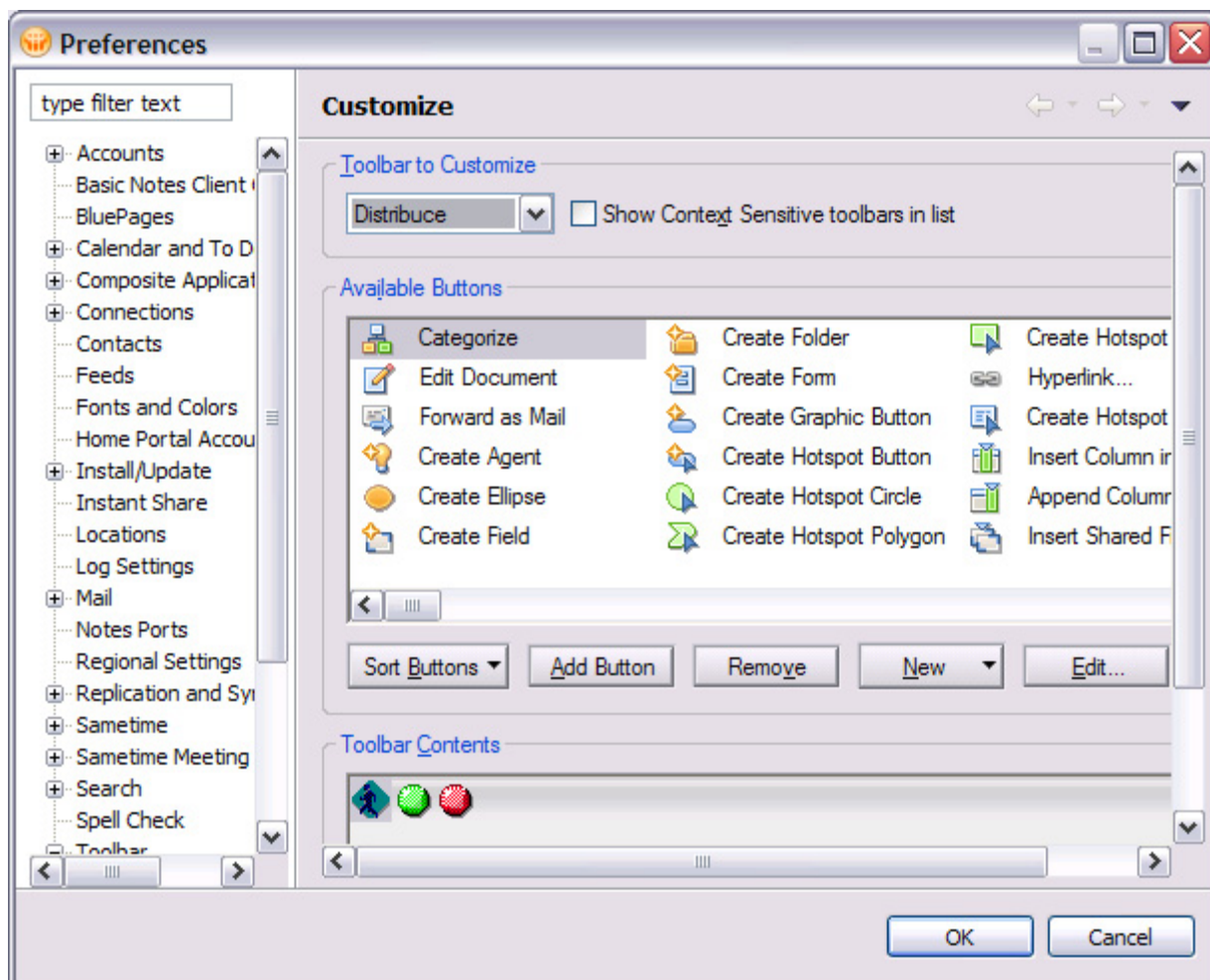
Na obrázku č. 24 je použití tlačítka „Expiration date“.



Obr. 24 Použití tlačítka „Expiration date“.

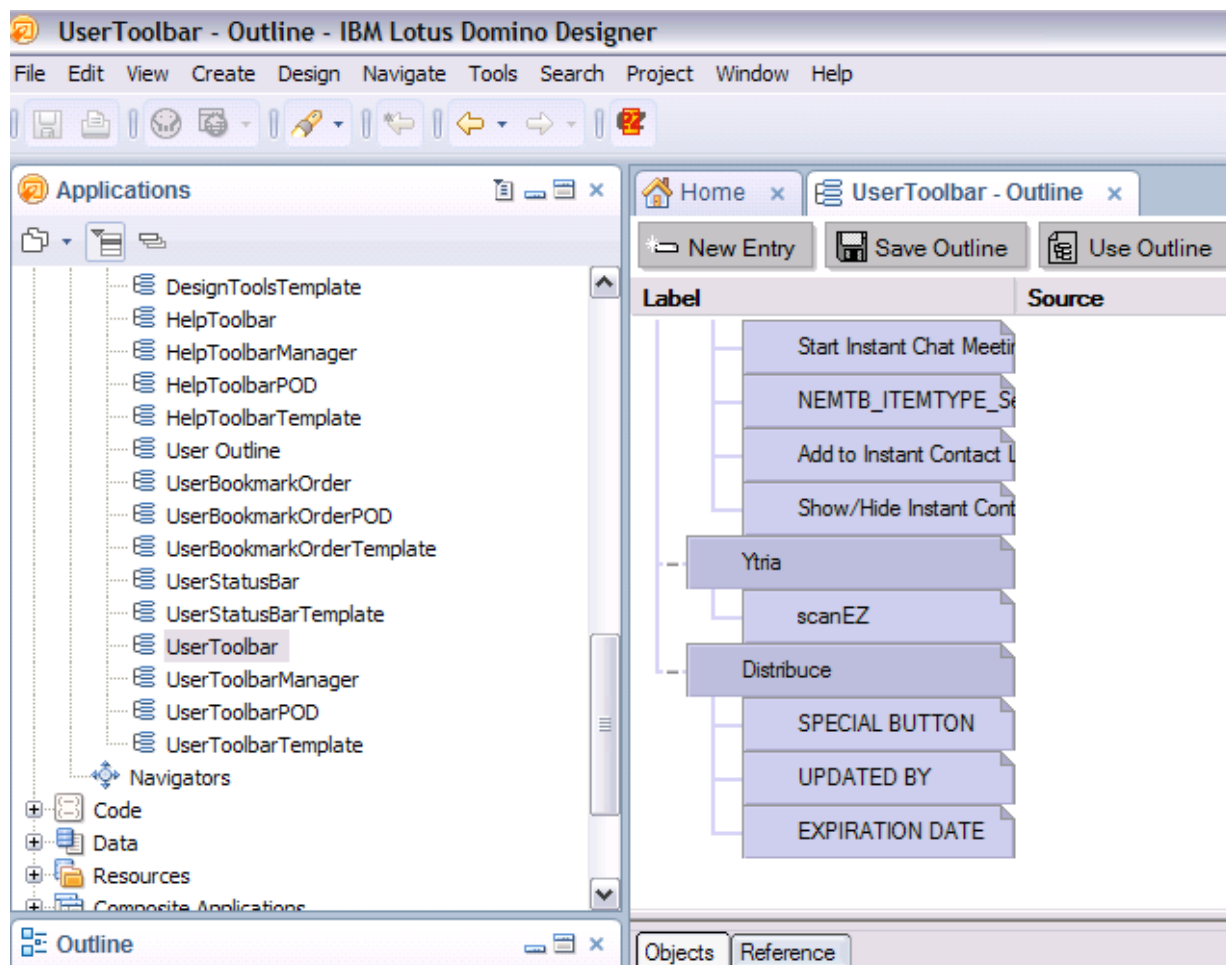


Obr. 25 Ukázka skriptu tlačítka „Expiration Date“.



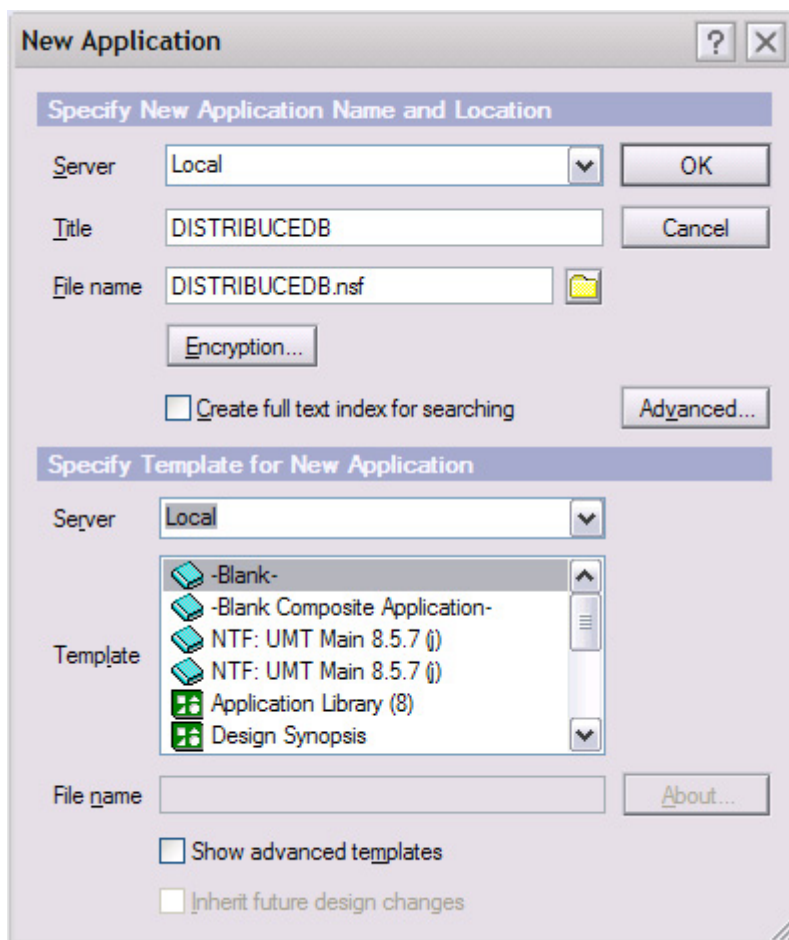
Obr. 26 Ukázka tlačítkové lišty „Distribuce“ se třemi tlačítky.

Otevřel jsem si lokální databázi „bookmark.nsf“ v designeru a našel jsem si osnovu (Outline) „Distribuce“



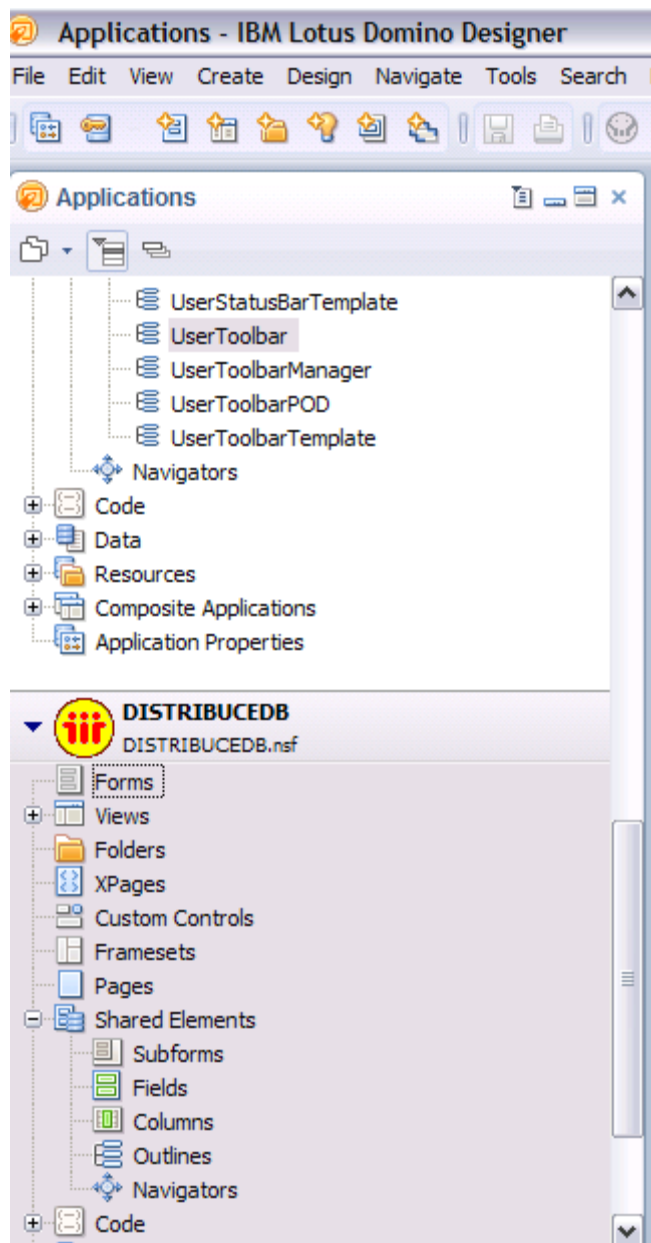
Obr. 27 Osnova (Outline) „Distribuce“.

Dále jsem si vytvořil novou prázdnou databázi „Distribuce.nsf“. (viz. obr. 28)



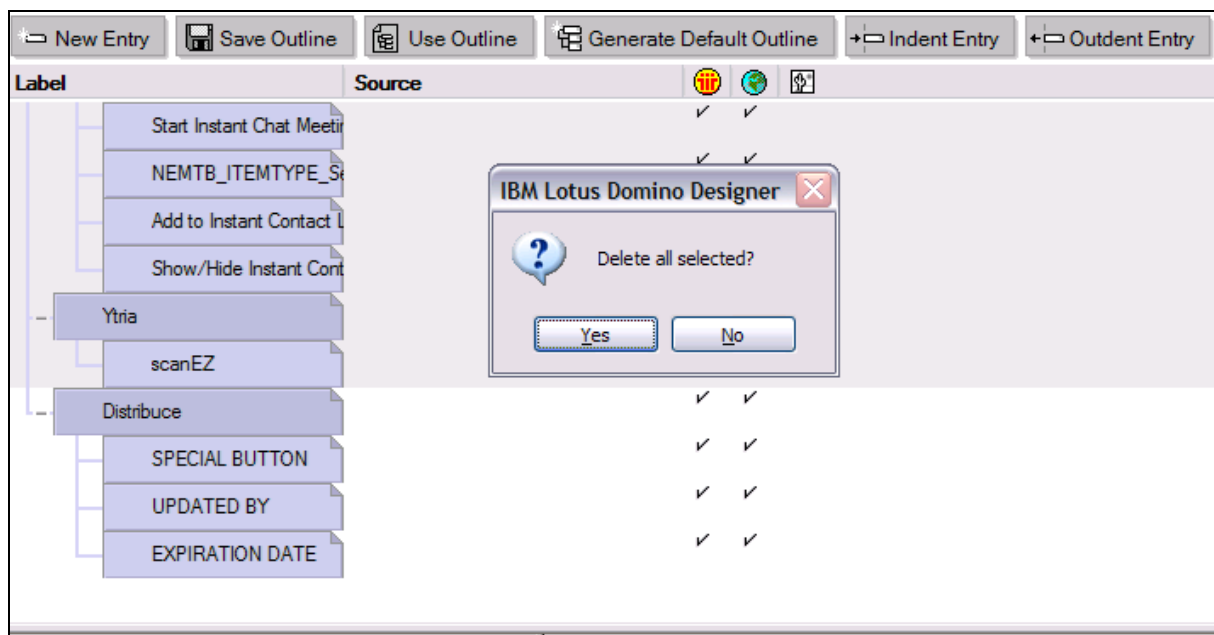
Obr. 28 Vytvoření prázdné databáze.

Tuto novou databázi jsem si otevřel v designu a nakopíroval jsem osnovu „UserToolbar“ z databáze „bookmark.nsf“ do databáze „distributedb.nsf“ (viz obr. 29).



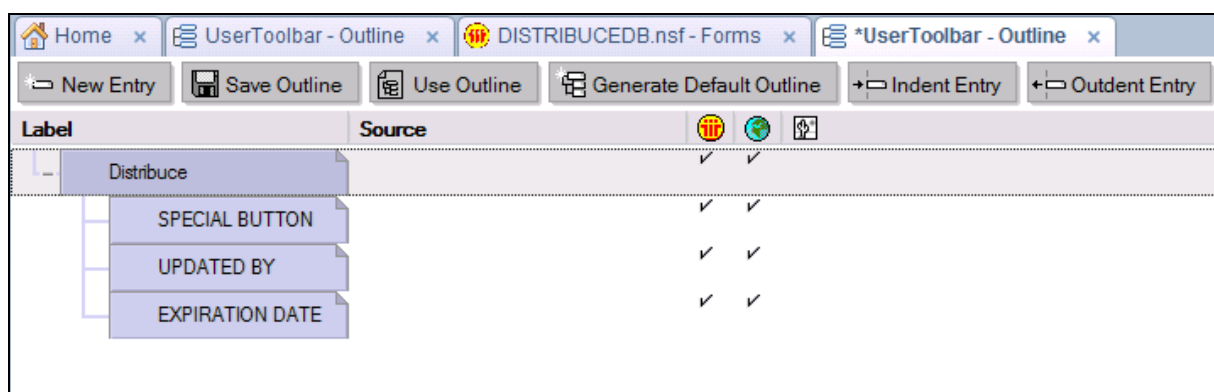
Obr. 29 Design databáze „distributedb.nsf“.

Po nakopírování osnovy „UserToolbar“ z databáze „bookmark.nsf“ do databáze „distributedb.nsf“ jsem smazal všechny uživatelské lišty kromě té, kterou chci distribuovat. (viz obr. 30).



Obr. 30 Smazání ostatních tlačítkových lišt.

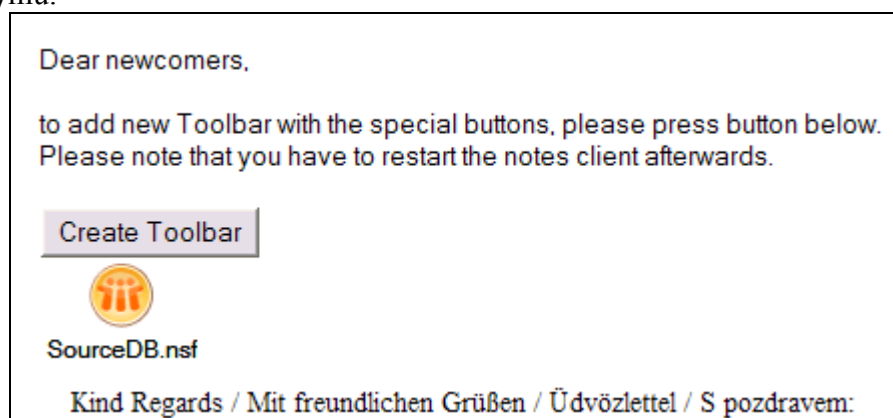
Po smazání mi v databázi „Distribuce.nsf“ zůstala pouze lišta „Distribuce“ (viz obr. 30).



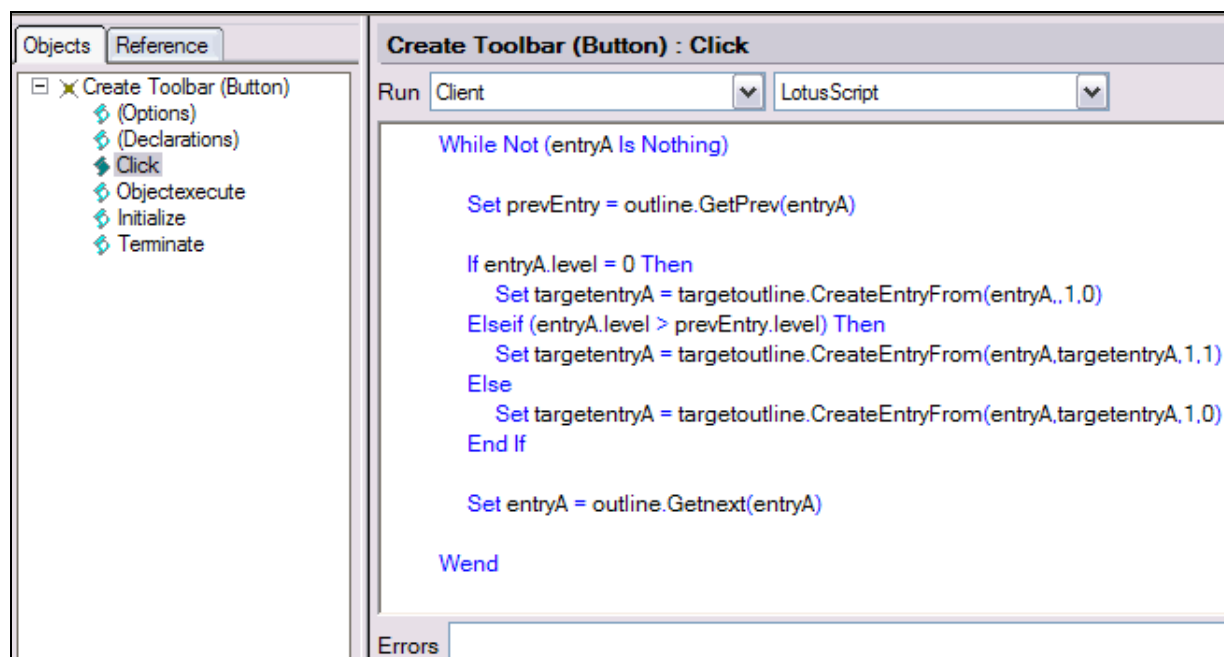
Obr. 31 Tlačítková lišta „Distribuce“.

Když tedy chceme někomu následně nastavit tlačítkovou lištu s tlačítky, tak stačí pouze vytvořit e-mail. Do e-mailu vložíme databázi „Distribuce.nsf“ jako přílohu.

Následující e-mail zobrazený na obrázku č. 32 pak pošleme uživateli nebo novým členům týmu.



Obr. 32 Ukázka e-mailu na vytvoření tlačítkové lišty.



Obr. 33 Ukázka skriptu tlačítka v e-mailu.

Uživatel po stisknutí tohoto tlačítka dostane instrukce restartovat notes klienta. Po restartu notes klienta, už uvidí novou tlačítkovou lištu a může začít používat tlačítka z přidané lišty.

8.3.6 Vytvoření migrační databáze

Vytvořil jsem tedy novou databázi, kterou jsem umístil na server přístupný zároveň pro administrátory globální a lokální domény. Proces funguje tak, že globální či lokální administrátor je zodpovědný při splnění příslušného úkolu i potvrdit splnění úkolu v této databázi. To znamená, že musí změnit status úkolu na „Completed“. Databáze je nastavena tak, že při změně statusu se ve vedlejší kolonce automaticky zobrazí datum, kdy byl úkol splněn a také jméno administrátora, který akci provedl.

Migrace tedy probíhá v následujících fázích:

1. Administrátor z globální domény nejdříve zadá do databáze všechny informace o migrovaném uživateli:
 - jméno uživatele,
 - datum migrace,
 - země, ze které je uživatel migrován,
 - země, do které je uživatel migrován,
 - administrátor původní domény,
 - administrátor budoucí domény,
 - původní server,
 - budoucí server,
 - globální hub server,
 - lokální hub server.

2. V přípravné fázi jednotliví administrátoři připravují příslušné kroky na migraci uživatele:
 - globální administrátor informuje lokálního o migraci,
 - globální administrátor kontroluje, jestli má uživatel přístup na server,
 - lokální administrátor vytváří repliku na hub serveru.
3. V samotném provedení migrace administrátor z globální domény:
 - odstraní pole „X_CONSOLIDATED“ z dokumentu v adresáři,
 - nastaví nové hodnoty polí v dokumentu v adresáři (server, doména, stát),
 - odstraní přesměrování adresy.

Na obrázku č. 34 vidíme ukázkou skriptu tlačítka statusu v migrační databázi.

The screenshot shows a LotusScript editor window with a table and a script. The table has three columns: 'Status', 'Date of completion', and 'FreezingDate'. The script is a LotusScript subroutine named 'Onchange' that updates the 'Date' field based on the 'Completed' field's value.

	Status	Date of completion
at the migration	Completed	Date
	Status	FreezingDate
ccess group	Completed	Date_1
	Status_1	FreezingDate_1

```

Completed (Field) : Onchange
Run Client LotusScript

Sub Onchange(Source As Field)
  Dim workspace As New NotesUIWorkspace
  Dim uidoc As NotesUIDocument
  Dim s As New NotesSession
  Dim NName As New NotesName( s.Username )
  Set uidoc = workspace.CurrentDocument
  If uidoc.FieldGetText("Completed") = "Completed" Then
    Call uidoc.FieldSetText( "Date", Str( Now ) + " - " + NName.Common )
  ElseIf uidoc.FieldGetText("Completed") = "None" Then
    Call uidoc.FieldSetText( "Date", "" )
  End If
End Sub

```

Obr. 34 Ukázka skriptu tlačítka Statusu.

9. TEST PROGRAMOVÉHO ŘEŠENÍ

Aby bylo možné příslušné řešení nebo agenta použít přímo v produkci, to znamená na databázích nacházejících se přímo na produkčních serverech, je potřeba takového řešení otestovat. To zaručuje vysokou kvalitu programového návrhu.

Testování nové šablony, například pro nějakou hlavní databázi, jsem provedl na testovacích serverech a to tím, že jsem na testovacím serveru nastavil i ostatní databáze, které spolupracují s touto databází. Pak jsem následně vytvořil všechny možné požadavky v dané databázi a sledoval jsem, jestli specifický požadavek provádí všechny příslušné úkony. Nejdůležitější věcí bylo nastavit všechny databáze a propojení těchto databází v testovacím prostředí stejně jako v produkčním prostředí. V minulosti se totiž stala situace, že nástroj, který výborně fungoval v testovacím prostředí, způsobil několik chyb po implementaci v produkčním prostředí.

Testování některých agentů není nutné provádět na testovacích serverech. Provedl jsem pouze testování na lokálním serveru. Udělal jsem to jednoduše tak, že jsem vytvořil kopii příslušné databáze z produkce ovšem bez dokumentů, následně jsem nakopíroval pouze pár dokumentů do této databáze a na těchto dokumentech jsem pak lokálně provedl samotný test.

Jednoduché agenty jsem otestoval přímo v produkci. Měl jsem například úkol upravit dvě stě dokumentů. Agentu jsem vždy nejprve spustil pouze na jednom dokumentu a pak jsem zkoumal, jestli vše proběhlo tak jak má. Agenti, kteří posílají informační e-maily, jsem testoval přesměrováním e-mailů na speciální testovací účty, které jsem vytvořil pro tento účel.

10. ZÁVĚR

Kompletní IT podpora velké mezinárodní společnosti je velmi složitý proces. Aby tato podpora byla úspěšná ve všech dílčích segmentech, ať už se jedná o rychlé vyřešení problémů, nebo rozšiřování nabídky pro zákazníka, je nutné mít ve firmě spoustu specializovaných pracovníků, kteří vykonávají velmi odbornou práci. Nejdůležitějším úkolem je ovšem zkoordinovat jednotlivé týmy, aby byli schopni respektovat dané procesy.

Důkladnou analýzou různých kategorií příchozích tiketů na druhou úroveň podpory jsem se mohl zaměřit na kritické procesy. Zkontroloval jsem existující dokumentaci na tyto procesy na zákaznickém centru, dále jsem vytvořil všechny další potřebné dokumenty pro tým první úrovně podpory. Docílil jsem také toho, že pracovníci první úrovně začali ke své práci tuto dokumentaci využívat.

Podářilo se mi dále zdokumentovat všechny procesy pro administrátory druhé úrovně podpory. Vytvořením přehledné struktury v databázi jsem svým kolegům zjednodušil hledání odpovídajících dokumentů.

Naprogramováním jednotlivých agentů jsem administrátorům ušetřil čas, také jsem eliminoval monotónní práci v týmu. Speciálními pohledy v určitých databázích jsem dokázal administrátory upozornit na nebezpečí možných chyb.

Je možné, že moje práce nemůže úplně zdokonalit celý proces kompletní podpory, ale přinejmenším jsem dosáhl těchto dílčích cílů:

- zmenšení počtu příchozích tiketů na druhou úroveň,
- zmenšení počtu chybných rušení účtů,
- zkvalitnění služeb pracovníků první úrovně,
- zrychlení práce administrátorů druhé úrovně,
- zlepšení koordinace mezi oběma úrovní podpory.
- zkvalitnění spolupráce administrátorů globální a lokální domény.

Tímto jsem určitě dosáhl zvýšení spokojenosti zákazníka, což je pro firmu, která se zabývá outsorsingem údržby počítačových aplikací ve velkých mezinárodních společnostech klíčová věc.

SEZNAM POUŽITÉ LITERATURY

- [1] BRIAN, Benz; Rocky, Oliver. *Mistrovství programování Lotus Notes a Domino*. Brno : Computer Press, 2005. ISBN 80-251-0750-7.
- [2] TVRDÍKOVÁ, M. *Zavádění a inovace informačních systémů ve firmách*. Praha : Grada, 2000. ISBN 80-7169-703-6.
- [3] MOLNÁR, Zd. *Efektivnost informačních systémů*. Praha : Grada, 2001. ISBN 80-247-0087-5.
- [4] VOŘÍŠEK, J. *Strategické řízení informačního systému a systémová integrace*. Praha : Management Press, 1997. ISBN 80-85943-40-9.
- [5] Světnotes.cz [online]. Brno (Česká republika): ABC Systems, s.r.o., 2002. [cit. 2013-05-20]. Dostupné z: <SvětNotes.cz>
- [6] HAMMER, Michael; Champy, James. *Reengineering – Radikální proměna firmy*, Praha : Management Press, 1995. ISBN 80-7261-028-7.
- [7] DONOVAN, J. *Business Re-Engineering with Information Technology*. New Jersey : P T R Prentice Hall, 1994.
- [8] FAŤUN, M.: Reengineering je důležitým nástrojem pro zefektivnění podnikových procesů. *Computerworld*, 1998, č. 3.
- [9] *FULL SPECTRUM SOFTWARE SUPPORT AND MAINTENANCE* [online]. [cit. 2013-05-20]. Dostupné z: <<http://www.embedded360.com/it-custom-application-services/software-support-and-maintenance.htm>>
- [10] Which issue trackers support sub-tickets, and how well do they work for bridging the gap between project managers and developers? *Stackoverflow* [online]. Sep 26 '08 at 15:30. [cit. 2013-05-20]. Dostupné z: <<http://stackoverflow.com/questions/140236/which-issue-trackers-support-sub-tickets-and-how-well-do-they-work-for-bridging>>
- [11] *Jemný úvod do Lotus Notes* [online]. Květen 1999. [cit. 2013-05-20]. Dostupný z: <<http://radimp.tripod.com/dedek/ld01.htm>>
- [12] ZEMAN, M. Systém řízení informační bezpečnosti podle normy ISO 27001. *Perspektivy kvality* [online]. 2007, č. 2. [cit. 2013-05-20]. Dostupné z: <<http://www.perspektivykvality.cz/k-hlavnimu-tematu/system-rizeni-informacni-bezpecnosti-podle-normy-iso-27001.html>>
- [13] MACÁK, J. Přehled UML diagramů. *JakubMacák*. [online]. On 5 Jan, 2013. [cit. 2013-05-20]. Dostupné z: <<http://www.jakubmacak.cz/prehled-uml-diagramu/>>
- [14] DOYLE, S. What's New in IBM Domino 9.0 Social Edition. *IBM Notes and Domino wiki* [online]. On 5 Jan, 2013. [cit. 2013-05-20]. Dostupné z: <http://www-10.lotus.com/ldd/dominowiki.nsf/dx/Whatapos_New_in_IBM_Domino_9.0_Social_Edition/>

POUŽITÉ NÁSTROJE

- [15] *ScanEZ* [online] Dostupné z:
<http://www.ytria.com/WebSite.nsf/WebPageRequest/Solutions_scanEZen?OpenDocument&Lang=en>
- [16] *Notes Document Viewer Version 2.0* [online]. Dostupné z:
<http://www.maysoft.com/web3.nsf/page/products-document_viewer-overview>
- [17] *Violet UML Editor* [online] Dostupné z:
<<http://alexdp.free.fr/violetumleditor/page.php>>
- [18] *IBM Lotus Notes 8.5.3* [online] Dostupné z:
<<http://www.softpedia.com/progDownload/IBM-Lotus-Notes-Download-175953.html>>

SEZNAM POUŽITÝCH ZKRATEK

IBM	International Business Machines
VŠB	Vysoká škola báňská
SMTP	Simple Mail Transfer Protocol
SAP	System Analysis and Program Development
NRPC	Notes Remote Procedure Call
ISO	International Organization for Standardization
POP3	Post Office Protocol
IMAP	Internet Message Access Protocol
LDAP	Lightweight Directory Access Protocol
MAPI	Messaging Application Programming Interface
HTTP	Hypertext Transfer Protocol
NNTP	Network News Transfer Protocol)
HTML	HyperText Markup Language
XHTML	<u>E</u> xtensible <u>H</u> yper <u>T</u> ext <u>M</u> arkup <u>L</u> anguage
XML	Extensible Markup Language
ACL	Access Control List
LN	Lotus Notes
UML	Unified Modeling Language