

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Absolvování individuální odborné praxe

Individual Professional Practice in the Company

2011

David Holuša

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 6. května 2011

.....

Rád bych na tomto místě poděkoval mému vedoucímu odborné praxe ve firmě Webcom, ing. Davidu Volnému, za ochotu a trpělivost, kterou se mnou měl po celou dobu absolvování praxe.

Abstrakt

Tato bakalářská práce se zabývá stručným popisem průběhu absolvování individuální odborné praxe ve firmě Webcom a.s. V úvodu práce je popsáno zaměření firmy Webcom a.s. První kapitola je věnována stručnému úvodu do problematiky systému Microsoft Dynamics NAV. Následující kapitola obsahuje trojici vybraných úloh, které byli na odborné praxi studentovi zadány. U každé s těchto úloh je lehký úvod do problematiky a popis funkčnosti řešení. První úloha se zabývá problematikou tvorby nových společností v systému NAV. Druhá úloha ukazuje využití webové služby Medidata. Poslední úloha popisuje práci se souborovým systémem a správu obrázků v systému NAV. Závěrečná kapitola obsahuje zhodnocení dosažených výsledků a vlastní přínos studenta.

Klíčová slova: Databáze, Informační systém, Microsoft Dynamics NAV, Navision, C/AL, Medidata

Abstract

Bachelor work is concerned with brief description of process of individual professional practise in the company Webcom a.s. In the beginning described the focus of the company Webcom a.s. The first chapter is devoted to a brief introduction on the Microsoft Dynamics NAV. The following chapter contains three selected tasks that were on the practical training of students entered. For each of these tasks is easy introduction, a description of the functional and description of the solution. The first task deals with the creation of new companies in the NAV system. The second task demonstrates the use of Web services Medidata. The last task describes how to work with filesystem and manage files in NAV system. The final chapter contains an evaluation of the results and the student's own contribution.

Keywords: Database, Information System, Microsoft Dynamics NAV, Navision, C/AL, Medidata

Seznam použitých zkratk a symbolů

NAV	– (Microsoft Dynamics Navision 5.0) Informační systém z řady Microsoft Dynamics.
C/SIDE	– (Client/Server Integrated Development Environment) Klientské grafické rozhraní systému NAV.
C/AL	– (Client/server Application Language) Procedurální programovací jazyk.
SIFT	– (Sum Index Flow Technology) Databázová technologie systému NAV.
OLAP	– (On Line Analytical Processing) Rychlá analýza sdílených vícerozměrných informací.
DBMS	– (Database Management System) Systém řízení báze dat.
HTTP	– (Hypertext Transfer Protocol) Internetový protokol.
HTTPS	– (Hypertext Transfer Protocol Secure) Zabezpečený internetový protokol.

Obsah

1	Úvod	4
1.1	O firmě Webcom	4
1.2	Microsoft Dynamics	4
2	Microsoft Dynamics NAV	5
2.1	Databáze	5
2.2	Objekty aplikace	6
2.3	Jazyk C/AL	7
3	Moje práce	8
3.1	Nástroj pro tvorbu nové společnosti	8
3.2	Webová služba Medidata	10
3.3	Správa obrázků zboží	12
4	Závěr	16
5	Reference	17
	Přílohy	I
A	Obrázky	I

Seznam obrázků

1	Ukázka technologie SIFT	6
2	Tvorba nové společnosti	9
3	Nastavení nové šablony	9
4	Hromadný import obrázků	13
5	Přehled uložených obrázků	14
6	C/SIDE Klient	II

Seznam tabulek

1	Objekty aplikace	7
2	Význam znaků v názvu souboru	15

1 Úvod

1.1 O firmě Webcom

V rámci mé bakalářské práce jsem se rozhodl vykonávat odbornou praxi ve společnosti WEBCOM a.s., kde jsem nastoupil již v květnu roku 2010, na pozici Junior developer.

Společnost WEBCOM a.s. je největší partner společnosti Microsoft v oblasti poskytování informačních systémů pro řízení podniků a státních institucí v České republice. Poskytuje SW řešení pro střední a velké společnosti zaměřené převážně na modulové systémy Microsoft Dynamics. Nabízí modulové informační systémy pro řízení obchodu a vztahu se zákazníky, moduly pro ekonomiku, controlling, logistiku, dopravu a další IS řešení na zakázku.

Firma WEBCOM a.s. má již více než 210 spokojených zákazníků s implementovaným řešením Microsoft Dynamics, a také zákazníky, kteří používají internetové aplikace založené na produktech společnosti Microsoft.

V roce 2010 byla oceněna jako jedna ze tří společností v České republice titulem President's Club 2010. Navíc obstála v celosvětovém srovnání a jako jediná v České republice a jedna ze tří společností v rámci střední a východní Evropy, jí bylo uděleno ocenění Inner Circle 2010, které získávají partneři skupiny Microsoft Dynamics s nejlepšími obchodními výsledky po splnění dalších výběrových kritérií za uplynulé období. Svoji stabilitu na trhu a pozici silného partnera společnosti Microsoft potvrdila také tím, že v roce 2010 získala renomované ocenění Microsoft Dynamics Reseller of the Year Finalist pro střední a východní Evropu.

1.2 Microsoft Dynamics

Jak již bylo zmíněno, společnost WEBCOM a.s. se zabývá modulovými systémy Microsoft Dynamics. Řešení Microsoft Dynamics tvoří řada integrovaných, adaptabilních podnikových řešení.

1. Microsoft Dynamics NAV
2. Microsoft Dynamics AX
3. Microsoft Dynamics CRM

Všechna tato řešení fungují podobně jako další známý software společnosti Microsoft a spolupracují s ním, což znamená snadné osvojení a snížení rizika spojeného s implementací nového řešení. Dále automatizují a zjednodušují procesy související s financemi, vztahy se zákazníky a odběratelsko-dodavatelským řetězcem, a pomáhají v dosahování podnikatelské úspěšnosti.

2 Microsoft Dynamics NAV

V této kapitole se stručně seznámíme s tímto informačním systémem. Seznámíme se s jazykem C/AL a podíváme se na strukturu databáze.

Tak jako ostatní produkty z řady Microsoft Dynamics, je i NAV ekonomickým informačním systémem pro řízení firemních procesů.

Jedná se o dvouvrstvou aplikaci obsahující systém řízení báze dat (DBMS), které je na straně serveru a grafické rozhraní (C/SIDE), které je na straně klienta. C/SIDE klient (obr.6) zároveň plní roli vývojového prostředí jazyka C/AL.

Poznámka 2.1 Nejnovější verze systému NAV (Microsoft Dynamics NAV 2009 R2), je již postavena na třívrstvé architektuře. Klientskou vrstvu pak rozdělujeme na tenkého a tlustého klienta. Střední vrstvu tvoří aplikační server.

2.1 Databáze

Nyní se podíváme na rozdělení a datový model databáze v systému NAV.

NAV databáze je postavena na relačním datovém modelu a podporuje následující datová uložště:

1. *Standard Dynamics NAV database* - jedná se o nativní databázi reprezentovanou souborem s příponou *.fdb*. Tento soubor je možno načíst i bez DBMS a pracovat tak s databází v jednoruživatelském režimu. Pro řízení souběhu se zde využívá optimistického přístupu pomocí verzování databáze.
2. *SQL Server database* - řízení souběhu je v tomto případě řešeno zamykáním objektů.

2.1.1 Rozdělení

Databáze se dělí na 3 hlavní části:

1. *Objekty aplikace* - jsou rozděleny do sedmi základních objektů. Podrobněji si je popíšeme později.
2. *Firemní data* - jelikož systém umožňuje vytvářet neomezené množství společností v rámci jedné databáze, jsou i data jednotlivých společností logicky oddělena.
3. *Společná data pro všechny firmy* - v této části databáze jsou uložena všechna data, která jsou společná pro všechny vytvořené společnosti.

2.1.2 Technologie SIFT

Zajímavou databázovou technologií systému NAV je SIFT.

Dvě hlavní výhody SIFT spočívají ve zlepšené výkonnosti poskytování informací a OLAP, což umožňuje koncovému uživateli získat ze systému podrobné informace, které potřebuje.

Další výhodou je, že metoda SIFT kalkuluje všechny součty v programu a neukládá je. To má za následek, např. rychlejší účtovací proces a zlepšenou optimistickou konkurenci. SIFT je do programu zabudována pomocí FlowFields, FlowFilters a SumIndexFields.

Příklad 2.1

Mějme tabulku *Customer* z obr. 1. Tato tabulka obsahuje dva FlowField. Atribut s názvem *Any Entries* je FlowField typu Exist, a atribut *Balance* FlowField typu Sum. Obrázek 1 ilustruje fungování technologie SIFT.

<i>Customer (Table data)</i>				
Customer	Name	Country/Region Code	Balance (FlowField)	Any Entries (FlowField)
10000	Windy City Solutions	US	60	Yes
10010	Modern Cars Inc.	US	90	Yes
10020	Jean Saint Laurent	FR	210	Yes
10030	Russel Publishing	UK	0	No
10040	La Cuisine Française	FR	300	Yes

<i>Customer Entry (Table data)</i>			
Customer	Date	Comment	Amount
10000			10
10000			20
10000			30
10010			40
10010			50
10020			60
10020			70
10020			80
10040			90
10040			100
10040			110

Virtual part of the table data

Obrázek 1: Ukázka technologie SIFT

2.2 Objekty aplikace

Objekty aplikace mohou být považovány za fyzické soubory jako v tradičním informačním systému. V databázi existuje sedm typů objektů popsanych v tabulce 1.

C/SIDE klient obsahuje nástroj zvaný *Object Designer*, který umožňuje spuštění a editaci všech zmíněných objektů.

Typ objektu	K čemu se používá?
<i>Table</i>	Tabulka slouží k uložení aktuálních dat.
<i>Form</i>	Formulář umožňuje přístup k informacím uložených v tabulce. Slouží ke vkládání nových dat nebo k prohlížení již uložených dat.
<i>Report</i>	Report slouží k prezentaci souhrnných dat.
<i>DataPort</i>	DataPort slouží k importu/exportu dat z/do jiných programů.
<i>XMLport</i>	XMLport slouží k import a exportu dat v XML formátu.
<i>Codeunit</i>	Obsahuje definované funkce napsané v C/AL kódu. Tyto funkce pak mohou být volány z ostatních objektů aplikace.
<i>MenuSuite</i>	Obsahuje menu, které jsou zobrazeny v navigačním panelu.

Tabulka 1: Objekty aplikace

2.3 Jazyk C/AL

Jazyk C/AL vychází z jazyka Pascal. Jedná se tedy o procedurální jazyk, který navíc umožňuje volat některé databázové operace (např. COMMIT). V systému NAV slouží pro psaní veškeré logiky. Pro psaní C/AL kódu slouží C/AL editor, který je k dispozici v C/SIDE klientu. Většina zdrojového kódu a aplikační logiky je napsána v Codeunit objektech. Pokud se tedy budeme nadále bavit o psaní zdrojového kódu, budeme pro zjednodušení uvažovat psaní Codeunity.

V tomto jazyce rozlišujeme dva druhy datových typů:

1. *Základní* - Option, Integer, Decimal, Char, Duration, Text, Code, ...
2. *Komplexní* - Blob, Table, Record, InStream, OutStream, ...

Proměnné je možné deklarovat globálně, pro celou Codeunitu, nebo lokálně v rámci jedné funkce. Pokud deklaruje proměnnou jako globální, je potřeba myslet na její inicializaci. Pokud však deklaruje proměnnou lokálně, v rámci jedné funkce, tak je inicializace proměnné zajištěna s každým novým voláním funkce.

Výpis 1 v příloze A, která je k dispozici na doloženém CD, obsahuje ukázkou procedury napsanou v jazyku C/AL. Tato procedura vrací všechny zákazníky, kteří mají zadané příjmení.

3 Moje práce

Během své odborné praxe jsem převážně pracoval na menších úkolech, jako jsou supportní úpravy a údržba databáze. Příkladem supportní úpravy může být přidání nějakého políčka na formulář nebo přidání nějakých informací do reportu. Těchto úprav jsem dělal opravdu mnoho a mám za to, že nemá význam se zde o nich zmiňovat.

Mimo tyto drobnější úpravy, již existujících objektů, jsem také pracoval na tvorbě úplně nových nástrojů a funkcí. Tyto nástroje jsem se rozhodl zde popsat podrobněji.

3.1 Nástroj pro tvorbu nové společnosti

Jak už jsme si v předchozích kapitolách řekli, tak v systému NAV můžeme v rámci jedné databáze vytvářet libovolný počet společností a tím logicky oddělit jejich data. Této možnosti využívají i někteří ze zákazníků společnosti Webcom. Ve většině případů však dochází k vytváření společností se stejnou nebo podobnou parametrizací některých číselníků. Je tedy zřejmé, že by bylo vhodné parametrizaci číselníků automatizovat.

Při vytváření nové společnosti v systému NAV máme na výběr z několika předdefinovaných šablon společností (viz. obr.2).

Tyto šablony mají předdefinovaná data pro daný typ společnosti. Zdá se, že šablona je přesně to, co by nám pomohlo při automatizaci číselníků. Ovšem NAV, standardně neumožňuje vytváření vlastních šablon. Proto bude zapotřebí vytvořit nástroj, který by toto umožnil a zároveň využil standardního nástroje pro tvorbu společnosti.

Bude tedy zapotřebí vytvořit vzorové společnosti a ty naparametrizovat. Z těchto vzorových společností pak budou vznikat jednotlivé šablony.

Nyní si popíšeme funkčnost celého nástroje a následně i jeho implementaci.

3.1.1 Popis funkčnosti

Obecný popis

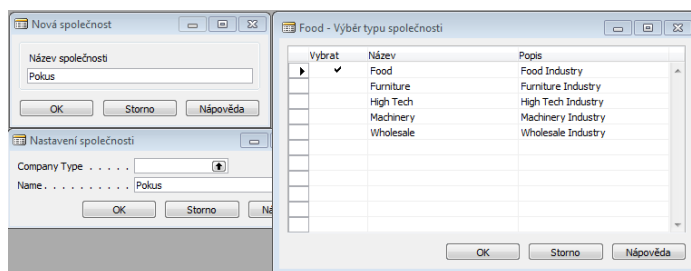
Pokud budeme chtít vytvořit novou šablonu, musíme se nejdříve připojit do databáze a načíst danou vzorovou společnost. Vycházejme ze skutečnosti, že vzorová společnost je již náležitě naparametrizována. Pokud jsou všechna vzorová data připravena, můžeme spustit nástroj pro tvorbu společnosti.

Po spuštění nástroje je potřeba provést inicializaci všech naparametrizovaných objektů. Všechny tyto objekty se po inicializaci zobrazí a můžeme s nimi dále pracovat.

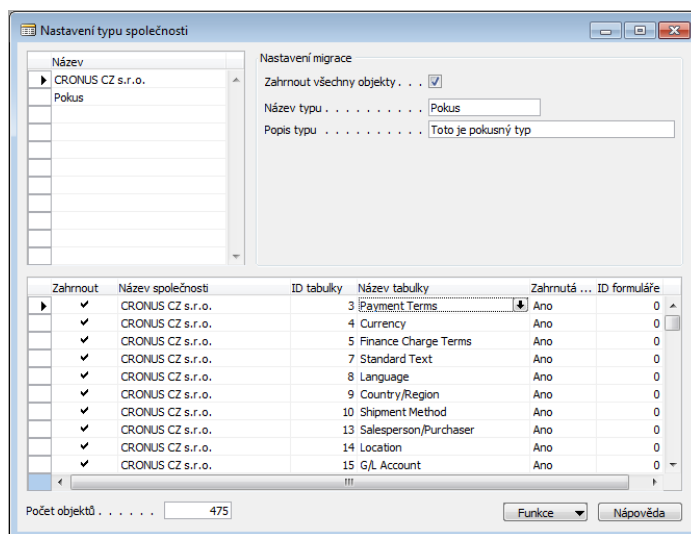
Je třeba vzít v úvahu, že ne vždy budou pro vytvoření šablony potřebné všechny objekty a jejich pole. Jak můžeme z obr. 3 vidět, je možné zahrnout pouze ty objekty a pole, které jsou pro šablonu potřebné.

Pokud jsou nastaveny všechny potřebné objekty a pole, můžeme přistoupit k samotné tvorbě nové šablony společnosti. Pro spuštění je ještě potřeba vyplnit název a popis nové šablony. Poté se vytvoří nová šablona společnosti.

Nástroj dále umožňuje aktualizaci objektů, které jsou zahrnuty do šablony. *Aktualizace* porovná všechny objekty vzorové společnosti s objekty, které jsou vybrané do šablony.



Obrázek 2: Tvorba nové společnosti



Obrázek 3: Nastavení nové šablony

Pokud dojde ke změně jakéhokoli objektu ve vzorové společnosti, jsme na to upozorněni, můžeme do šablony zahrnout nový aktualizovaný objekt. Pokud potvrdíme nahrazení, bude stávající objekt nahrazen. Pokud bude ve vzorové společnosti vytvořen nový objekt, můžeme jej stejným způsobem zahrnout do šablony.

Stejně jako je možnost šablony vytvářet, tak je zde také možnost vybranou šablonu smazat.

Práce s šablonou

Nyní se podíváme jak prakticky funguje načítání nových šablon při standardním vytváření společnosti v systému NAV.

Šablona je reprezentována Xml souborem, který je uložen v instalační složce NAV. Soubor je pojmenován stejně jako název šablony (*jeho struktura je popsána na doloženém CD v příloze A*).

Z příkladu vidíme, že kořenovým tagem je `DataList`. V tomto tagu jsou obsaženy všechny záznamy jednotlivých tabulek. Jako názornou ukázkou jsem vybral uložení jednoho záznamu tabulky `Currency` (Měna). Jak je vidět z ukázky, tak každá tabulka začíná tagem "`<Názevtabulky>List`", v našem případě `CurrencyList`. Tag `TableID` nám udává číslo objektu, pod kterým je uloženo v databázi. Tag `FormID` nám udává číslo přehledového formuláře. Tag s názvem tabulky označuje jeden záznam daného objektu, v našem případě se jedná o tag s názvem `Currency`. Dále jsou vypsány hodnoty jednotlivých atributů.

Všechny šablony společnosti jsou shromážděny v konfiguračním souboru (*jeho struktura je popsána na doloženém CD v příloze A*). Konfigurační soubor obsahuje názvy a popisy všech existujících šablon a cesty k datovým souborům. Tato konfigurace se pak načítá při standardním vytváření společnosti v systému NAV (viz. obr.2).

Abychom tedy využili standardního nástroje, bude zapotřebí vytvořit datový soubor nové šablony, ve formátu `Xml` a modifikovat konfigurační soubor.

3.1.2 Návrh datové struktury

Jak z funkčnosti vyplývá, je zapotřebí ukládat objekty a jejich konfiguraci (*viz. kapitola Popis funkčnosti*). Abychom dosáhli perzistentního uložení objektů, bude zapotřebí navrhnout vhodnou datovou strukturu ve formě databázových tabulek (*viz. datový slovník přílohy A na CD*).

- Tabulka *Migration Table Export* - v této tabulce jsou uloženy všechny objekty určené pro vytvářenou šablonu. Dále se ukládá konfigurace daného objektu. Při vytváření nové šablony jsou zahrnuty všechny objekty, které jsou uloženy v této tabulce a jsou označeny k zahrnutí do šablony.
- Tabulka *Migration Table Field Export* - v této tabulce jsou uloženy všechny pole vybraných objektů. Při vytváření nové šablony jsou zahrnuty pouze ty atributy, které jsou označeny k zahrnutí do šablony.

Pro funkci aktualizace, o které jsem se již zmiňoval v předchozí kapitole, navrhne dvě tabulky. Při spuštění aktualizace objektů dochází k porovnávání objektů vzorové společnosti a objektů uložených v tabulce *Migration Table Export*. Výsledky porovnání se uloží do následujících tabulek.

- Tabulka *Log Update Migration* - v této tabulce jsou uloženy všechny objekty, které byli změněny nebo přidány.
- Tabulka *Log Update Migration Field* - v této tabulce jsou uloženy všechny atributy, ve kterých došlo ke změně nebo byly přidány.

3.2 Webová služba Medidata

Společnost Medidata s.r.o. zajišťuje snadnou a bezpečnou komunikaci ve zdravotnictví. Umožňuje přenášet objednávky, dodací listy, faktury, recepty a jiné doklady. Vše je řízeno transakčním serverem Medidata.

Jeden ze zákazníků společnosti Webcom a.s. dodává mimo jiné kancelářské vybavení do zdravotnických zařízení. Bude proto zapotřebí zajistit komunikaci s transakčním serverem Medidata, získat všechny objednávky ze zásobníku nezpracovaných dokumentů a naimportovat do systému NAV.

Nyní se podíváme na to, jak probíhá komunikace s transakčním serverem Medidata.

3.2.1 Popis funkčnosti

Komunikace je založena na přístupu *klient - server* přes protokoly HTTP nebo HTTPS a probíhá v systému BizTran. Pro přenosy se používá hostname serveru *www.medidata.cz*. Cesta k virtuálnímu adresáři je */dbbiztran1/* pro HTTP protokol nebo pro HTTPS bez klientského certifikátu.

Všechny dokumenty zasílané na virtuální adresář serveru jsou zabaleny do obálky zprávy, která nese obecné základní údaje:

- Odesílatel, identifikace přes unikátní ID klienta
- Ověření odesílatele (klíč)
- Komu je zpráva adresována; ID klienta
- Typ dokumentu, který zpráva dále obsahuje a jejich číselný kód typu. (Např. objednávka = 10, dodací list = 20)
- Identifikace transakce, pokud se jedná o zprávu, která navazuje na již existující transakci
- Čas vytvoření a verzi zprávy

Zprávy jsou zasílány ve formátu Xml. Xml zpráva musí mít odpovídající strukturu, která je definována schématem *Message*, což je hlavní obal všech zpráv zasílaných do systému *BizTran*. Schéma si můžeme prohlédnout v příloze. V příloženém schématu jsou také detailně popsány jednotlivé atributy zprávy.

Jak jsem se již v zadání zmínil, tak naším úkolem je zpracovat pouze dokumenty typu *Objednávka*. Zpráva objednávky je určena schématem *Order*, na které se také můžete podívat v příloze. V příloze jsou opět detailně popsány všechny atributy zprávy. Zpráva s objednávkou musí být zapouzdřena do schématu *Message*. Abychom objednávku úspěšně zpracovali, potřebujeme provést tři následující kroky:

1. Žádost o objednávku
2. Vyzvednutí objednávky
3. Povtrzení o zpracování objednávky

Pojďme se nyní na jednotlivé kroky podívat podrobněji.

Žádost o objednávku

Klient, v tomto případě tedy dodavatel, musí odeslat na server žádost o vyzvednutí objednávky. Odpovědí je buď zpráva se staženou objednávkou anebo zpráva, která informuje, že na serveru není pro dodavatele v danou chvíli žádná nezpracovaná objednávka.

Při žádosti o vyzvednutí objednávky se odesílá pouze hlavička s identifikačními údaji a zpět jako odpověď vrací server zprávu s dokumentem (objednávkou), pokud je na serveru k dispozici v zásobníku nezpracovaných dokumentů. Existuje několik způsobů jak určit, kterou objednávku ze serveru vyzvednout:

1. Server ze zásobníku vrací klientovi první objednávku (tj. bere se nejdéle čekající dokument ze zásobníku).
2. Vyzvednutí objednávky od konkrétního odběratele.
3. Vyzvednutí konkrétní objednávky, pokud je dodavateli známo ID transakce.

Vyzvednutí objednávky

Při úspěšném odeslání požadavku vrací server zprávu s návratovými hodnotami. Pokud je v zásobníku požadovaný doklad, je vrácen spolu se zprávou.

Potvrzení o zpracování objednávky

Doklady, v našem případě objednávky, které byly vyzvednuty ze zásobníku serveru, je nutno po úspěšném zpracování ze zásobníku přesunout. Tuto událost vyvolá klient, v tomto případě tedy dodavatel, který si objednávku vyzvedl. Zaslání potvrzení o zpracování bychom však měli provést až po úspěšném načtení objednávky do systému NAV tak, aby v případě jakéhokoli problému nedošlo ke ztrátě dokladu.

3.3 Správa obrázků zboží

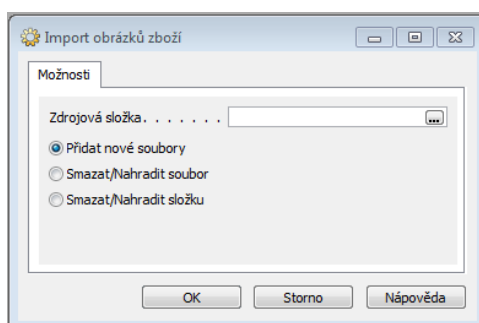
Dalším požadavkem je vytvoření centrálního úložiště produktových obrázků přes systém NAV. Obrázky budou ukládány na souborový systém serveru. Do databáze se budou ukládat pouze odkazy k jednotlivým obrázkům. Systém NAV bude také vytvářet požadovanou strukturu souborového systému.

Dále bude zapotřebí zajistit import/export obrázků. Import bude ukládat obrázky do systému NAV a zároveň na souborový systém serveru.

Nyní se podíváme na podrobnější popis celé funkčnosti.

3.3.1 Popis funkčnosti

Celá funkčnost pro správu produktových obrázků se skládá ze dvou nástrojů.



Obrázek 4: Hromadný import obrázků

1. Import obrázků zboží

Prvním nástrojem je procesní report, který slouží k hromadnému importu obrázků.

Jedná se o grafické rozhraní (obr.4), ve kterém si můžeme zadat zdrojovou složku s obrázky připravenými k importu.

Poznámka 3.1 Obrázky musí splňovat dohodnutý formát názvu, který budeme specifikovat později.

Dále má uživatel na výběr ze tří možných voleb hromadného importu:

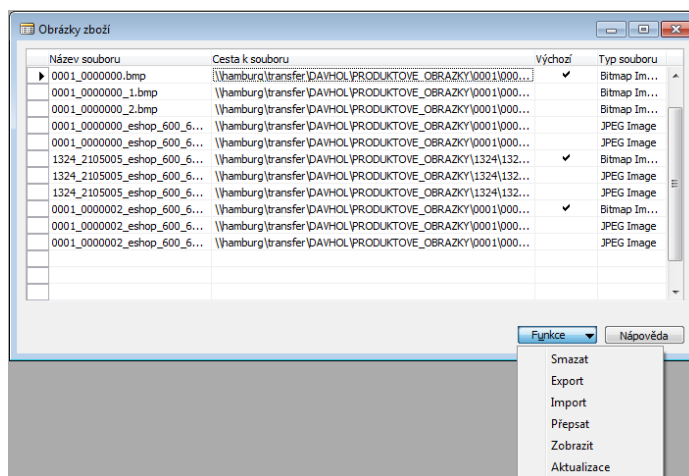
1. *Přidat nové soubory* - Tato volba zajistí přidání nových obrázků, které ještě v systému nejsou.
2. *Smazat/Nahradit soubor* - Tato volba zajistí přidání nových a nahrazení stávajících obrázků se stejným názvem.
3. *Smazat/Nahradit složku* - Tato volba zajistí nahrazení všech obrázků v dané složce.

Po zadání zdrojové složky a výběru volby můžeme hromadný import spustit. Po spuštění se nám zobrazí dialogové okno, které nám zobrazí průběh kopírování obrázků. Obrázky jsou zkopírovány do souborového systému serveru. Do databáze jsou pak uloženy odkazy k těmto obrázkům. Obrázky jsou v databázi přiřazeny k jednotlivému zboží. Číslo zboží je získáno z názvu souboru.

2. Obrázky zboží

Dalším nástrojem je přehledový formulář obrázků. Formulář je možno spustit ze standardní karty zboží. V tomto případě se přehled obrázků spustí s filtrem čísla zboží. Pokud bychom spustili formulář, zobrazí se nám samostatně přehled všech obrázků v databázi. Tato volba však není častá.

Na tomto formuláři máme k dispozici několik funkcí (viz. obr.5). Pojdme si tedy všechny tyto funkce popsat.



Obrázek 5: Přehled uložených obrázků

- *Smazat* - odstraní vybraný počet obrázků. Po spuštění se zobrazí dialogové okno s potvrzením o odstranění. Pokud odstranění výběru potvrdíme, budou všechny vybrané obrázky odstraněny jak z databáze, tak ze souborového systému serveru.
- *Export* - umožňuje stažení vybraných obrázků ze souborového systému serveru na zvolené umístění.
- *Import* - umožňuje import jednoho obrázku. Importovaný obrázek musí opět splňovat dohodnutý formát názvu souboru. Pokud se však pokusíme importovat soubor, který podle svého názvu náleží jinému zboží a máme nastaven filtr na jedno konkrétní zboží, bude název automaticky přepsán. Pokud bychom ale filtr nastaven neměli, importovaný obrázek se do databáze uloží pod číslem, které je určeno v jeho názvu (*jak je tomu u hromadného importu*).
- *Přepsat* - umožňuje nahrazení vybraného obrázku.
- *Zobrazit* - zobrazí vybraný obrázek ve výchozím prohlížeči systému.
- *Aktualizace* - provede aktualizace souborového systému vůči databázi a naopak.

Formát názvu souboru

Jednotlivé obrázky určené k importu jsou označeny předem dohodnutým názvoslovím (např. *1324_2105005_eshop_600_600_01.jpg*). Pojďme se nyní podívat na význam jednotlivých znaků v názvu souboru. Tabulka 2 obsahuje popis jednotlivých znaků.

<CallNo>_<ProductNo>_<Typ>_<Rozměry>_<Pořadí>.<Přípona>

Znak	Popis
<i>CallNo</i>	Volací znak produktu
<i>ProductNo</i>	Produktové číslo
<i>Typ</i>	Definuje kategorii pro kterou je obrázek určen (např. E-shop, webový portál, atd.)
<i>Rozměry</i>	Rozměry obrázku
<i>Pořadí</i>	Pořadové číslo obrázku
<i>Přípona</i>	Typ souboru

Tabulka 2: Význam znaků v názvu souboru

Struktura souborového systému

Jak už bylo řečeno, obrázky jsou ukládány do souborového systému na serveru. Aby bylo možné se souborovým systémem lépe a přehledně pracovat, je navržena následující struktura:

```
-PRODUKTOVE_OBRAZKY
--1324 (volací znak produktu)
---1324_2105001 (cislo produktu)
----soubor_1
----soubor_2
----soubor_n
---1324_2105002 (cislo produktu)
----soubor_1
----soubor_2
----soubor_n
--1325
---1325_2410201
----soubor_1
----soubor_2
----soubor_n
```

4 Závěr

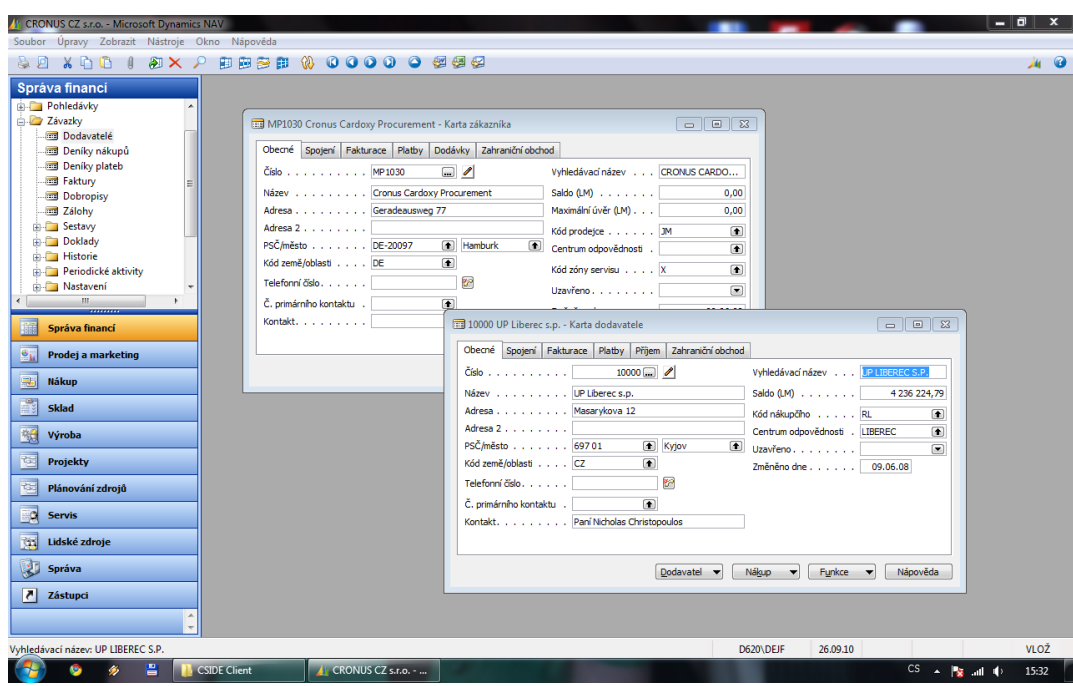
V průběhu praxe jsem úspěšně vyřešil všechny zadané úlohy a většina z nich byla nasazena do ostrého provozu u zákazníka. Každá z řešených úloh byla pro mě velkým přínosem. Měl jsem možnost rozšířit své znalosti z oblasti databází. Setkal jsem se s novými a zajímavými technologiemi. Poznal jsem metody a postupy práce v týmu. Získal jsem nový pohled na realizaci informačního systému.

Věřím, že všechny znalosti a zkušenosti nabyté na této praxi mi budou ku prospěchu pro další studium a pomůžou mi při výběru mého budoucího zaměstnání. Obzvláště bych rád využil zkušeností, které jsem získal při práci s databázemi, jelikož bych se rád v této oblasti dále specializoval.

5 Reference

- [adg07] MICROSOFT CORP. *Application Designer's Guide*. 2007. 618s. Dostupné na WWW: <http://www.microsoft.com/downloads/en/details.aspx?FamilyID=a3c9c3e3-11fc-446a-ac12-f6ad0749cb50>
- [nae01] NAVISION A/S. *Navision Attain Essentials*. 2001.
- [med10] DYMEŠ, Pavel. *Medidata - Popis elektronické komunikace klientů*. 2010. Verze 1.06.
- [msdn] MICROSOFT. *Microsoft Dynamics NAV 2009 Developer and IT Pro Documentation* [online]. 2011. Dostupné na WWW: <http://msdn.microsoft.com/en-us/library/bb219495.aspx>.

A Obrázky



Obrázek 6: C/SIDE Klient