

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

EDITOR PASPORTIZACE VUT

SEMESTRÁLNÍ PROJEKT
TERM PROJECT

AUTOR PRÁCE
AUTHOR

Bc. Daniel Bierza

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

EDITOR PASPORTIZACE VUT
PASPORTIZATION EDITOR BUT

SEMESTRÁLNÍ PROJEKT
TERM PROJECT

AUTOR PRÁCE
AUTHOR

Bc. Daniel Bierza

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. Jaromír Marušínek, Ph.D., MBA

BRNO 2007

Abstrakt

V této práci budu prezentovat problematiku pasportizace. Provedu analýzu současného stavu u budov VUT. Popíši možná řešení pasportizace VUT v budoucnosti. Zaměřím se na analýzu formátu „obr“ pomocí reverzního inženýrství. Provedu analýzu získaných dat. Popíši způsoby uložení pasportizačních dat. Navrhnou grafický prohlížeč a editor pasportizace.

Klíčová slova

Pasportizace, reverzní inženýrství, formát obr, možnosti zobrazení grafických dat formát DXF a DWG.

Abstract

I will present the issue of passportization in my work. I will analyze the current status of the BUT buildings. I will describe possible solutions of passportization at the BUT in the future. I will focus on the analysis of the "obr" format through the method of reverse engineering. I will do the analysis of the acquired data. I will describe the way of saving of the passportization information. I will design a graphic browser and a passportization editor.

Keywords

Passportization, reverse engineering, format obr, graphic data representation possibilities, format DXF and DWG.

Citace

Bierza Daniel: Editor pasportizace VUT. Brno, 2009, diplomová práce, FIT VUT v Brně.

Editor pasportizace VUT

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Jaromíra Marušince, Ph.D., MBA.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Daniel Bierza
27. 1. 2009

Poděkování

Rád bych poděkoval vedoucímu mé diplomové práce za vypsání této práce, která byla pro mne i zábavou. Objevil jsem nové věci, které budou pro můj další profesní růst přínosem. Děkuji svým rodičům a přítelkyni, kteří se stali čtenáři, korektory a recenzenty této práce.

© Daniel Bierza, 2009.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	3
2	Pasportizace	4
3	Analýza pasportizace VUT v současnosti	7
3.1	Informační systém Apolo	8
3.2	Grafický editor pasportizace GT facility	9
3.3	Grafický editor pasportizace svopcad6	10
3.4	Výhody a nevýhody současného stavu	11
4	Pasportizace VUT v budoucnosti	12
4.1	Možnosti řešení pasportizace VUT	12
4.2	Přechod mezi současným a budoucím řešením	14
5	Získání pasportizačních dat z binárního souboru	15
5.1	Reverzní inženýrství	15
6	Způsoby uložení získaných pasportizačních dat	17
7	Způsoby rovinného vykreslení	20
8	CAD systémy	22
8.1	Aplikace AutoCAD	22
8.2	Aplikace MicroStation	23
9	Schéma praktického postupu	25
10	Analýza binárního souboru pomocí reverzního inženýrství	28
10.1	Umístění dat v binárním souboru	28
10.2	Rozpoznání typu pasportizačních dat	30
10.3	Získání parametrů grafických objektů	32
11	Analýza získaných dat	37
11.1	Způsoby uložení grafických objektů	37
11.2	Napojení pasportizačních dat na databázi	39
11.3	Struktura pro načítání pasportizačních dat	40
12	Zobrazení načtených dat	41
12.1	Zobrazení načtených dat knihovnou GDI+	41
12.2	Zobrazení načtených dat programem AutoCAD	44
13	Vhodný způsob uložení dat	47
14	Závěr	49

Seznam obrázků

Obrázek 1 Procesy facility managementu.	4
Obrázek 2 Facility management - Provozní efektivita.....	5
Obrázek 3 Facility management VUT.	7
Obrázek 4 Grafická primitiva v Svopcad6.....	10
Obrázek 5 Základní operace s objekty.	11
Obrázek 6 Napojení systému Alfa na současný řešení VUT.	13
Obrázek 7 Schéma přechodu mezi současným a budoucím řešením.	14
Obrázek 8 Schéma praktického postupu.....	27
Obrázek 9 Struktura souboru formátu obr.....	29
Obrázek 10 Jednotná hlavička grafických objektů.	33
Obrázek 11 Rozložení bitu určujících počet vrcholů lomené čáry.	34
Obrázek 12 Převod čtyř bytů obsahující souřadnici na číslo.....	36
Obrázek 13 Způsob uložení elipsy s různým natočením.	38
Obrázek 14 Způsob uložení n-úhelníku.	38
Obrázek 15 Různá kvalita vykreslení. HightQuality, HightSpeed, Default.....	43
Obrázek 16 Zobrazení grafických dat převodem do formátu DXF.	44
Obrázek 17 Možnosti uložení dat v programu AutoCAD.	45
Obrázek 18 Převod a uložení grafických dat do souboru XML.	47

Seznam tabulek

Tabulka 1 Porovnání velikostí souborů ve formátech DXF a DWG.	18
Tabulka 2 Porovnání velikostí souborů ve formátech DXF, DWG a DGN.	19
Tabulka 3 Charakteristické vlastnosti objektů.	31
Tabulka 4 Rozdělení objektů podle délky.....	32
Tabulka 5 Srovnání časové závislosti na kvalitě.	43

1 Úvod

Během posledních dvou let studia jsem měl možnost nahlédnout do problematiky pasportizace budov VUT. Toto téma mne zaujalo svou složitostí i důležitostí pro VUT a proto oceňuji, že toto téma bylo vypsáno jako jedno z témat diplomových prací.

Na počátku cesty bylo nutné se seznámit se stavem pasportizace VUT a na základě toho jsem si zvolil postup řešení diplomové práce, který vycházel i z potřeb VUT a problémů spojených s pasportizací VUT.

Jedním z úkolů, který budeme řešit bude analýza současného řešení pasportizace VUT a možnosti řešení této problematiky v budoucnu. VUT používá pro pasportizaci program GT facility, který ukládá data do binárních souborů. Neexistuje možnost tyto data exportovat se zachováním vazby na informace uložené v databázi. Naším hlavním cílem bude zbavit VUT závislosti na tomto programu. Analyzujeme jaké možnosti program GT facility umožňuje. Bude nutná analýza těchto binárních souborů pomocí reverzního inženýrství. Pokusíme se zjistit, jakým způsobem jsou uložena grafická data a jejich vazba na databázi, která je stěžejní. Pokud zjistíme strukturu souborů a grafických objektů, nebude problém tyto informace načíst.

Součástí této práce je i návrh grafického zobrazení, které bude sloužit pro kontrolu načtených grafických objektů. Zjistíme, jaká možná řešení lze použít pro zobrazení a následnou editaci. Jednoduchý grafický editor načtených dat bude navržen především pro možnost malých úprav před přechodem na jiné řešení pasportizace.

Podstatnou částí naší práce bude zjištění vhodných způsobů uložení načtených dat. Tento úkol je o to obtížnější, že není zatím znám produkt, který bude používán. Výstup z celé práce musí být proto natolik obecný, aby bylo možné jej importovat do nového řešení pasportizace VUT bez ztráty dat.

2 Pasportizace

Pojem pasportizace není standardizovaný název procesu. Pasportizace by se dala opsat jako správa hmotného majetku, správa budov nebo správa prostor. Pro nejlepší vysvětlení tohoto pojmu lze použít mnohem širší pojem Facility management. Facility management se skládá z velkého počtu procesů (Obrázek 1). Každý proces je svébytná činnost, avšak pouze všechny činnosti řízené jako celek mohou přinést konkurenční výhodu podniku.

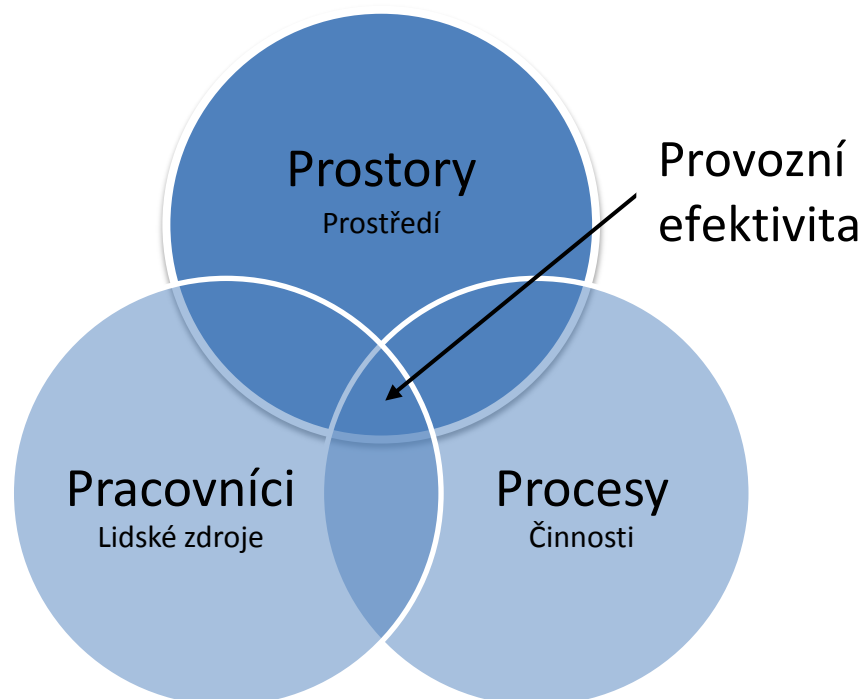


Obrázek 1 Procesy facility managementu.

Cílem facility managementu je posílit ty procesy v organizaci, pomocí nichž pracoviště a pracovníci podají nejlepší výkony a v konečném důsledku pozitivně přispějí k ekonomickému růstu a celkovému úspěchu organizace. U malých společností je vše relativně jednoduché a průhledné. S rozrůstáním společností či jejím vzrůstajícím vlivem v komerčním prostředí se stává efektivita fungování společnosti stále více prioritní záležitostí. Tyto činnosti

zajišťují zázemí, což převážně znamená, že zajišťují prostředí, ve kterém pracují jednotliví zaměstnanci, ať se jedná o generálního ředitele či pomocného skladníka. Tito všichni potřebují pro zajištění jejich práce množství zásadních či nevýznamných služeb a pomoci tak, aby se mohli plně věnovat „své“ činnosti, kterou mají v popisu práce. Facility management má za úkol toto vše zajistit a to v podobě, která je nákladově optimální, pro pracovníka nejpříjemnější, legislativně a formálně regulérní, ekologická a energeticky efektivní a odpovídající firemním standardům. [1]

Pro sladění pracovního prostředí, pracovníků a pracovních činností je nutné skloubit principy obchodní administrativy, architektury, humanitních a technických věd. Při sladění nastává provozní efektivita (Obrázek 2). V grafu u pracovníků a procesů se jedná vždy o činnosti zajišťované nebo určené pro osoby. Třetí oblast je pro facility management specifická. Správa prostor s efektivním využitím místa pracovníky a procesy se dá nazvat například pasportizace.



Obrázek 2 Facility management - Provozní efektivita.

Pasportizace je technická evidence objektů. Může popisovat nejenom plochu místností, ale i kdo místnost využívá a pro jaký účel, jaké obsahuje místnost vybavení a harmonogram využití místnosti. Může obsahovat i

informace, na které jističe jsou napojena světla a zásuvky. Množství evidovaných informací se liší podle počtu pasportizovaných budov a prostředků na pasportizaci vyhrazených.

Programy pro pasportizaci kombinují informační databázové systémy a grafické prostředí. Pro efektivní práci s evidovanými informacemi by plán budovy, zobrazený jako pouhý výpis místností s přidruženými informacemi, byl těžko představitelný. Proto je velmi důležitou částí grafické prostředí. Pro označení těchto grafických prostředí se používá zkratka CAFMⁱ systém. Grafická data jsou vkládána většinou z CADⁱⁱ podkladů projektové dokumentace nebo překreslením do CAFM systému.[2]

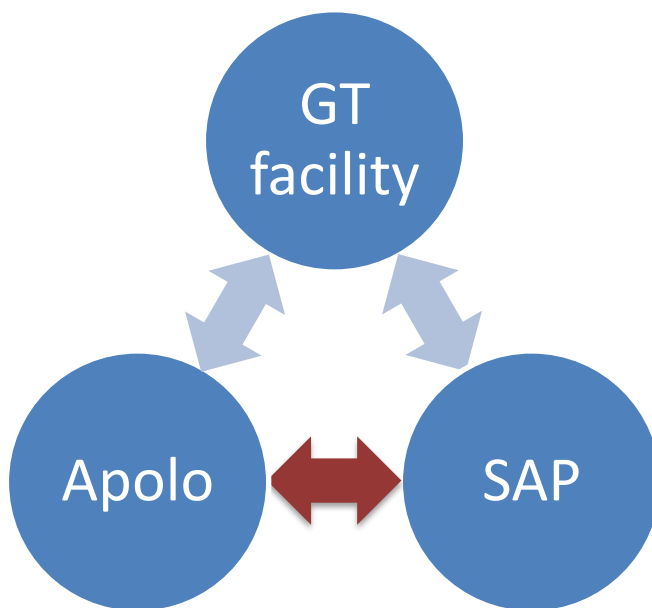
Grafická prostředí pracují většinou v rovinném zobrazení (2D). Toto zobrazení je plně postačující pro pasportizaci. Rovinné zobrazení je výhodné pro rychlé vykreslování velkého množství dat. Z prostorového zobrazení (3D) je oproti 2D zobrazení možné získat objemy místností, plochy stěn a obložení, plochy maleb. 3D zobrazení je praktické pro možnost rychlé prostorové orientace.

CAFM systémy kombinují informační databázové systémy a grafické prostředí, jak jsem již uvedl výše. V praxi se jedná o propojení grafického objektu (zásuvka, místnost, budova nebo pozemek) s alfanumerickou informací uloženou v databázi (číslo jističe, kontakt na osobu v místnosti, správce budovy, rozloha nebo míra rizika v záplavovém území). Výsledkem tohoto propojení je silný nástroj, který dokáže vytvářet grafické i alfanumerické výstupy, které lze využít pro plánování efektivního využití prostoru pracovníky a procesy.

3 Analýza pasportizace VUT v současnosti

Pro správu budov Vysokého učení technického se používá několik specializovaných programů (Obrázek 3), které jsou vzájemně provázané databází. Finance VUT se spravují v aplikaci SAP. SAP eviduje například plochu budov nebo místností. Pokud je místnost pronajata, tak komu. Z finančního pohledu je také důležité vybavení místností. Informace, který učitel v jaké třídě, kdy a jaký předmět učí, je pro SAP nepodstatné. Tato informace je naopak velmi důležitá pro plánování provozu budov. Nejen o tuto činnost se stará program Apollo, který navíc slouží i jako prohlížeč pasportizace budov a místností VUT. Třetí program s názvem GT facility slouží jako grafický editor pasportizace budov a místností VUT. Tento program je CAFMⁱ systém, který spravuje technickou dokumentaci budov VUT.

V další analýze se zaměřím na programy Apollo a GT facility, protože program SAP nepracuje s grafickým formátem obr, ale pouze s databází na něj napojenou. Programy Apollo a GT facility jsou důležitým výchozím bodem pro analýzu souboru formátu obr a jeho vazby na databázi.



Obrázek 3 Facility management VUT.

3.1 Informační systém Apolo

Program Apolo je rozsáhlý systém, který obsahuje velké množství modulů. Obsahuje moduly pro rozvrh, publikace, obědy a mnoho dalších. My se zaměříme na modul spojený s pasportizací VUT. Tento modul se nazývá „Budovy a místnosti“. Je určen pro jednoduché prohlížení pasportizace budov a místností Vysokého učení technického. Obsahuje detailní přehled všech budov a místností včetně fotografií, výkresů objektů, leteckých snímků areálů a statistických informací. V programu se lze jednoduše přepínat mezi jednotlivými budovami Vysokého učení technického a zobrazovat jejich fotografie, letecké snímky a zobrazovat výkresy jednotlivých podlaží. Fotografie a letecké snímky nejsou ke všem budovám dostupné. Seznam objektů lze seskupovat a zobrazit podle různých kritérií. Základní seskupení je podle lokalit, kde je seznam rozdělen na Brno – sever, Brno – střed a Brno – jih. Seskupení podle fakult zobrazí všechny fakulty VUT a seskupení podle areálů seskupí budovy podle ulic, na kterých sídlí.[3]

Tento modul umožňuje i rychlé vyhledávání a zobrazení místností. Místnost lze jednoznačně vyhledávat podle jejího inventárního čísla, čísla místnosti či zkratky budovy. Další možností je vyhledávání místnosti podle osoby, která v ní sídlí nebo podle čísla telefonní klapky v místnosti. V tomto případě není úspěšné vyhledání zajištěno, protože mezi místnostmi a osobami/klapkami neexistuje jednoznačné propojení. Když je v zobrazeném seznamu místností nalezena požadovaná místnost, pak může být poklepnutím myši přímo zobrazena na výkrese k ní příslušející. Místnosti lze filtrovat podle jejich účelu. Lze zobrazit například jen hygienická zařízení, technické místnosti nebo kanceláře.

Ke každé místnosti zakreslené do výkresu jsou přiřazeny informace s ní spojené. Mezi základní informace, které lze získat o jednotlivých místnostech, patří číslo místnosti a číslo výkresu, ve kterém je místnost zakreslena. Údaj využívaný nejen při vyhledávání je název místnosti. U každé místnosti je zaznamenána plocha a výška. Mezi další významné údaje patří zařazení, účel místnosti a podlaží, ve kterém se místnost nachází. S každou místností může být spojena i osoba, která má v místnosti pracovnu. Detailní informace o obsazení místnosti nejsou přiřazeny jednoznačně. Například v místnosti B206

pracuje ing. Petr Matoušek, Ph.D. Při získávání informací zjistíme nejen podrobnosti o jeho osobě, ale i o prodejně knih, sídlící v místnosti P206. Systém vyhledá pouze podle čísla místnosti. U každé místnosti lze editovat vybavení, které je v ní obsaženo. Vybavení místnosti lze vybrat pouze z pevného seznamu zařízení, mezi která patří bariera, bílá tabule, černá tabule, meotar, ozvučení, počítač, projektor, video a zatemnění. U vybavení lze nastavit, zda zařízení je přítomno nebo není, ale počet nebo typ zařízení již není možné nastavit.

Program Apollo umožňuje statistické vyhodnocení všech údajů, ke kterým přistupuje. Lze zjistit užitečnou plochu všech budov fakulty i počet obsazených místností fakulty. Tyto údaje lze zobrazit i v grafech.

Při zobrazení jakéhokoli výkresu se automaticky do složky Temp ukládá výkres ve formátu obr a z něj poté program získává informace a vykresluje je. Tyto soubory budou vstupem pro program převádějící binární soubor obr do čitelného formátu.

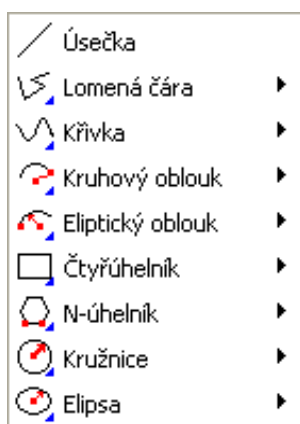
3.2 Grafický editor pasportizace GT facility

Program GT facility je CAFMⁱ systém, který je propojen s dalšími částmi zajišťujícími Facility management VUT pomocí databáze. Tento program slouží jako základní editor pro pasportizaci budov a místností Vysokého učení technického. V tomto editoru lze kreslit výkresy budov i dělat změny v již hotových výkresech. Tento program je propojen s databází a jakákoli změna se projeví okamžitě. Grafická data se ukládají do souboru formátu obr. Tento soubor obsahuje i vazby s databází. V případě, že potřebujeme změnit grafický editor, dají se exportovat veškerá grafická data, ale vazba na databázi se ztratí. Právě tento problém činí VUT na GT facility závislé. Pro analýzu chování programu GT facility lze použít i program svopcad6, který je velmi podobný, ale nepracuje přímo s databází. Pro důležitost pasportizačních dat je vhodnější použít pro analýzu program svopcad6, který pracuje pouze se soubory a připojení k databázi by mělo jít simulovat pomocí lokální databáze, ale tato možnost se nám nepodařila ověřit.

3.3 Grafický editor pasportizace svopcad6

Primární určení programu svopcad6 je kreslení a editování dvourozměrných výkresů budov. Vstupem i výstupem editoru svopcad6 jsou binární soubory formátu obr. Tento program je vhodný pro analýzu formátu obr pomocí reverzního inženýrství, protože není vázán na databázi a pasportizační data VUT. Není tedy možné poškodit velmi důležitá data. Editor umožňuje vektorový export do formátů dxf, vtx, emf, wmf, obx. Vektorový export ovšem nezachovává vazby s databází, uložené v souboru typu obr. Rastrový export umožňuje uložit výřez z právě zobrazené části výkresu do základních rastrových formátů. Důležitou součástí svopcad6 je import dat ze základních vektorových formátů mezi které patří dxf, dgn, shp, vtx, ocd, obx. Import se využívá pro vkládání nových výkresů dodaných k novým budovám.

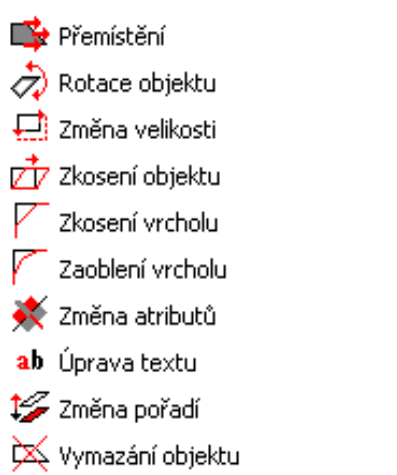
Svopcad6 umožňuje kreslení mnoha grafických primitiv (Obrázek 4), které mají mnoho dodatečných vlastností. Některé vlastnosti jsou u všech objektů stejné, například barva, tloušťka nebo typ vyplnění, který se zadává i u neuzavřených objektů. Důležitým parametrem u každého objektu je hladina, která udává příslušnost nakresleného objektu do skupiny. Hladinou mohou být obvodové zdi, dveře, okna a další stavební prvky. Svopcad6 umožňuje zobrazit jen určité hladiny, což zrychluje například hledání chyb nebo úpravu jen určité skupiny objektů v budově.



Obrázek 4 Grafická primitiva v Svopcad6.

Program Svopcad6 umožňuje základní operace s grafickými primitivami (Obrázek 5). Všechny operace však nejsou přístupné u všech objektů, například pro lomenou čáru nedokáže Svopcad6 provést operaci zkosení nebo zaoblení vrcholů. Pokud se objekt při rotaci ocitne částečně nebo zcela mimo

povolené rozměry výkresu, tak zmizí a zobrazí se v poslední správné poloze až po použití lupy, která způsobí jeho znovu vykreslení.



Obrázek 5 Základní operace s objekty.

3.4 Výhody a nevýhody současného stavu

Program GT facility je používán VUT již několik let. Byla do něj převedena projektová dokumentace téměř všech budov VUT. Možnost zobrazení a úpravy všech těchto výkresů v jednom programu jsou hlavní výhodou. Představa úpravy výkresu na papíře je nepředstavitelná. Pokud by se jednalo o editaci pouze CAD systémem nepropojeném s informací o místnosti, vedlo by to k množství ruční práce při převodu informací z výkresu do databáze a naopak.

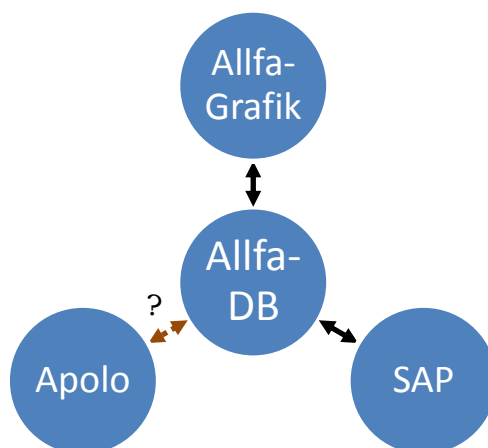
Mezi hlavní nevýhody pasportizace VUT patří nejednotnost výkresů. Výkresy byly tvořeny bez jakýchkoli sjednocujících zásad způsobů kreslení. Vzhled výkresu záleží na tom, kdy a kdo jej kreslil. Pokud jsou objekty umístěny do hladin, je to zcela chaotické. Možná příčina tohoto stavu je uživatelská nepřívětivost grafických editorů GT facility a svopcad6. Při kreslení jsou často uživateli zobrazovány špatné údaje. Pokud zadáme poloměr kružnice 100 a znovu zobrazíme informace o této kružnici, je udán poloměr 200, i když je ve skutečnosti 100. Některé parametry může uživatel změnit, ale nic se nestane. V porovnání s programem AutoCAD je grafický editor pro pasportizaci VUT velmi nepříjemně ovladatelný a práce s ním je pomalejší.

4 Pasportizace VUT v budoucnosti

Vzhledem k nevýhodám současného řešení pasportizace VUT probíhá výběrové řízení na pasportizační řešení pro VUT. To, že se budoucí software řešení pasportizace právě vybírá, má velký vliv na mou diplomovou práci. Pokud by byl znám software, který bude VUT používat v budoucnu a bylo by známé i řešení uložení dat, závěr diplomové práce by měl jasný cíl. Museli bychom výstup této práce připravit pro vstup nového produktu. Tím, že je vše v procesu výběru, musíme uložení získaných dat řešit co nejobecnější formou, ale zároveň i tak, aby se dala získaná data v případě potřeby zobrazit a editovat. Software pro pasportizaci nabízí v České republice jen několik firem. Každá přistupuje k pasportizaci jedinečným způsobem. Pro získání přehledu o možnostech, které mohou nastat při výběru možného nástupce GT facility, jsme vytvořili přehled nabídek, které se na českém trhu nacházejí.

4.1 Možnosti řešení pasportizace VUT

Jedním z možných nástupců GT facility je produkt Allfa společnosti Nemetschek. Systém Allfa je CAFMⁱ systém, který propojuje grafická 2D/3D data a relační databázový systém, obsahující související informace. Pro import dat do systému lze použít výkresy z CADⁱⁱ systémů ve formátu DXF^{ix}, DWG^{viii}, DGN^{iv}. Systém Allfa lze napojit na ekonomické řešení SAP R/3. Napojení na Apolo by asi firma Nemetschek zajistila přes společnou databázi (Obrázek 6). Allfa systém obsahuje dvě prostředí Allfa-DB a Allfa-Grafik. Allfa-DB zajišťuje operace nad databází bez grafické informace. Lze v něm editovat parametry budov, prostorů nebo místností, také vybavení místností nábytkem a ostatními zařízeními. Prostor Allfa-Grafik je CAFM část systému, která edituje grafická data.



Obrázek 6 Napojení systému Alfa na současný řešení VUT.

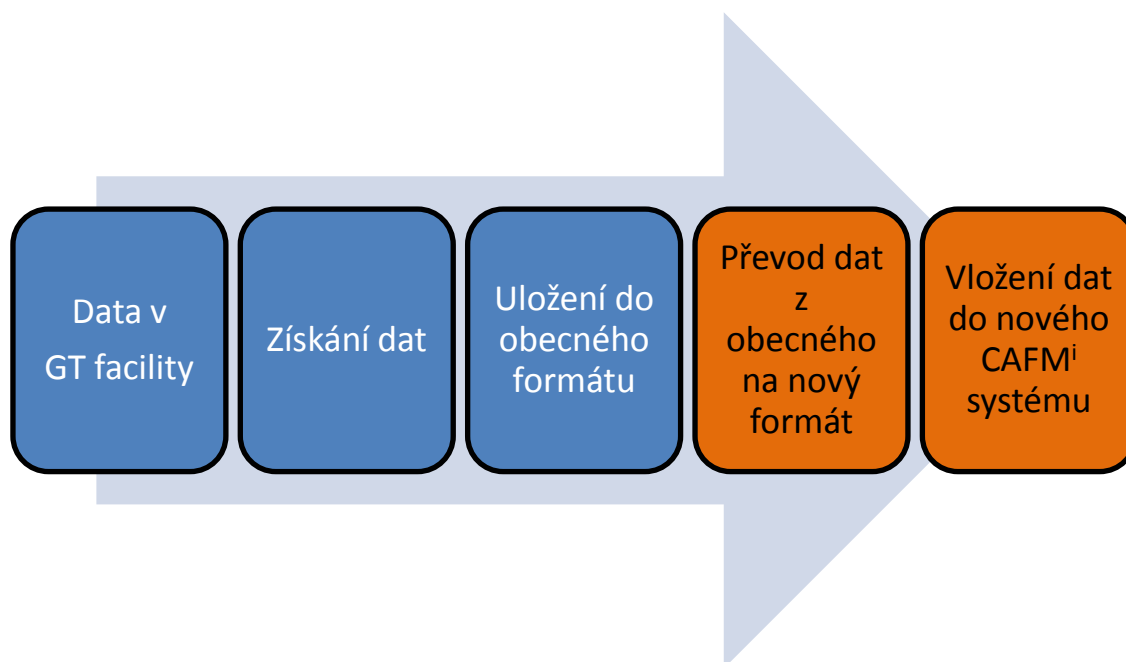
Dalším produktem, který lze použít jako náhradu GT facility, je systém Autodesk Topobase. Tato aplikace je postavena na aplikacích AutoCAD Map 3D, Autodesk MapGuide a databázovém úložišti Oracle Spatial. Autodesk Topobase je postaven na technologii Microsoft .NET a podporuje i budoucí 64bitové platformy. Databázové úložiště je otevřené dalším aplikacím, výkon se zvyšuje zpracováním topologických operací přímo Oracle serverem. Autodesk Topobase integruje data z aplikací pro počítačově podporované navrhování CAD, z databází majetku, z geografických informačních systémů a z papírových i elektronických zdrojů informací o zákaznících do centralizované databáze prostorových informací o infrastruktuře. Autodesk Topobase je čtyřrozměrný systém. Čtvrtým rozměrem je čas. Lze zobrazovat jak minulý, tak i budoucí stav dat.

Řešení od Firmy Tesco SW a.s. je řešení FaMa +. Toto řešení zahrnuje jak grafický editor, tak pokrývá procesy provozu, správy a údržby movitého i nemovitého majetku. Systém lze napojit na SAP. Podrobnosti o způsobech napojení na jiné programy a způsob ukládání dat není firmou zveřejněn.

Envis Facility Management je produkt společnosti DS Soft Olomouc. Tento software je postavený na Microsoft.NET. Datové úložiště je na serveru Microsoft SQL Server. Program parametrizovatelný s možností definice a konfigurace vlastních stavebních a technologických entit z něj činí naprosto otevřený systém. Systém poskytuje prostředí nejen pro evidenci pasportu, ale i pro plánování procesů údržby, revize a plánování preventivních prohlídek.

4.2 Přejchod mezi současným a budoucím řešením

Nejdůležitější otázkou je způsob přenosu dat z prostředí v současné době používaného GT facility do nového CAFMⁱ systému. Data jsou uložena v souborech ve formátu obr a s vazbou v databázi na informace o jednotlivých objektech. V této práci se zabýváme získáním pasportizačních dat z GT facility a vhodným uložením těchto dat (Obrázek 7). Po vybrání nástupce GT facility bude následovat převod dat do formátu tohoto nástupce a následné vložení těchto dat.



Obrázek 7 Schéma přechodu mezi současným a budoucím řešením.

5 Získání pasportizačních dat z binárního souboru

Současný systém pasportizace ukládá veškerá data do binárních souborů. Tento fakt znemožňuje přechod na jiný systém. Řešení by bylo možné, pokud by existovala možnost, jak data exportovat bez ztráty informace. Tato možnost však není k dispozici. Druhým řešením by bylo získat dokumentaci k binárnímu formátu od dodavatelské firmy. Tento krok však firma ASP a.s. nechystá. Tato firma by mohla také sama exportovat data například do formátu DXF i se zachováním vazby. U tohoto řešení je však podobná situace jako u předešlého. Jediná možnost získání dat zůstává analýza formátu pomocí reverzního inženýrství.

5.1 Reverzní inženýrství

Reverzní inženýrství označuje proces, jehož cílem je odkrýt princip fungování zkoumaného předmětu, jeho vlastností, vnitřních vztahů, architektury nebo designu. Reverzní inženýrství má za cíl sestavení ekvivalentního předmětu, který však není kopií originálu. Mezi obvyklé postupy reverzního inženýrství patří měření a analýza struktury předlohy a zkoumání vnitřních vztahů částí systému. Výstupem reverzního inženýrství je obnovení části návrhu nebo jen dokumentace, které mohou být podkladem k vylepšení designu nebo například k bezpečnostnímu auditu. Reverzní inženýrství se nepoužívá jen v informatice, ale nachází uplatnění i v jiných oborech lidské činnosti. V architektuře se reverzní inženýrství používá pro získání 3D modelu budov z naměřených významných bodů. Významné uplatnění našlo ve vojenství pro kopírování technologií. Sověti během let 1945 – 1949 „vyvinuli“ strategický bombardér Tupolev Tu - 4, který byl téměř dokonalou kopií amerického strategického bombardéru B-29. CIA se bránila proti úniku strategických technologií podstrčením několika technologií se skrytými chybami. V tomto případě se reverzní inženýrství stalo příčinou výbuchu sibiřského plynovodu, který je po atomových výbuších dosud největší umělá exploze vůbec.[4]

Reverzní inženýrství v informatice je definováno jako proces analýzy předmětného systému s cílem identifikovat komponenty systému a jejich vzájemné vazby nebo vytvořit reprezentaci systému v jiné formě nebo na vyšší úrovni abstrakce. V praxi jsou to dva hlavní druhy. V prvním případě jsou dostupné zdrojové kódy, ale chybí nebo není kompletní dokumentace. Cílem reverzního inženýrství je v tomto případě vytvoření nebo zkvalitnění dokumentace k dostupným zdrojovým kódům. Tyto případy se musí často řešit i po odchodu klíčového zaměstnance. V druhém případě nejsou dostupné zdrojové kódy ani dokumentace. Tento proces se někdy nazývá „Reverse code engineering“. Reverzní inženýrství může být prováděno různými metodami. Jedna ze základních metod je pozorování skrze výměnu informací, která je nejčastěji používána v protokolovém reverzním inženýrství. Tato metoda sleduje například chování na sběrnici nebo v síti. Po provedení analýzy chování je možné vytvořit produkt napodobující chování vzoru. Druhou metodou je takzvaná „disassembly“, kdy se reverzní inženýrství provádí analýzou strojového kódu programu. Tato metoda je časově velmi náročná. Třetí často používanou metodou je „decompilation“. Tato metoda se snaží převést dostupný strojový kód programu do vyššího programovacího jazyku.

6 Způsoby uložení získaných pasportizačních dat

Cílem diplomové práce je získat pasportizační data ze stávajícího CAFMⁱ systému a připravit je na vložení do nového systému. Výběrové řízení je stále v plném proudu, není znám vhodný formát dat, který by umožnil nový systém importovat. Na otázku, jak uložit načtená data, lze proto v současné době jen stěží odpovědět. Vycházíme z toho, že se jedná o data, které obsahují velice důležitou položku a to vazbu na databázi. A dále jsou to především grafická data. Mezi jednu z možností patří použití databáze jako dočasného úložiště pasportizačních dat. Toto řešení má výhodu a zároveň nevýhodu v přístupu k této databázi. Pasportizační data je nutné chránit před zneužitím, proto je nelokální databáze jen pro dočasné uložení zbytečně komplikovaná. Pro uložení vektorové grafiky s přidanou informací lze použít mnoho formátů určených pro grafiku nebo i obecný formát XMLⁱⁱⁱ, který je určen především pro výměnu dat mezi aplikacemi a pro publikování dokumentů. XML nepopisuje strukturu dokumentu z hlediska věcného obsahu jednotlivých částí, ale pouze pomocí značek určuje jejich význam.

Mezi formáty používané v CADⁱⁱ systémech patří například DXF^{ix}, DWG^{viii} nebo DGN^{iv}. Tyto formáty se používají běžně pro uložení projektové dokumentace. Velkou výhodou je možnost editace dat v CAD nástrojích, které jsou právě určeny pro vytváření a editaci projektové dokumentace.

Formát DWG byl vytvořen společností Autodesk a je nativním uložením dat z programu AutoCAD. Existuje 18 verzí tohoto formátu. Téměř každá verze AutoCADu přinesla vylepšení formátu DWG. Přes množství formátů je koncovka souboru stále DWG. Tento formát není popsán, ale je možno použít knihovnu ObjectDBX, která je určena pro vytváření, editaci a zobrazení tohoto formátu. Formát DWG se snaží Open Design Alliance dokumentovat a snaží se z něj vytvořit otevřený standard. Tato aliance vytvořila dokumentovaný formát OpenDWG. S příchodem knihovny verze DWGdirect 2.1.0 podala společnost Autodesk na Open Design Alliance žalobu a soud rozhodl

předběžným opatřením o stažení tohoto produktu. Přesto se OpenDWG nachází v mnoha CAD nástrojích „non-Autodesk aplikace“.[5]

Formát DXF byl vytvořen také společností Autodesk, ale na rozdíl od formátu DWG je kompletně dokumentován, protože slouží jako přechod mezi CAD programy. AutoCAD se stával s každou verzí více komplexní, toto vedlo k vytváření složitějších objektů, které jsou špatně dokumentované nebo zcela nedokumentované. Tato nevýhoda vede většinu tvůrců CAD softwaru k používání DWG formátu jako primárního formátu pro výměnu dat. Pro DWG je však nutné použít knihovnu Open Design Alliance, jak je popsáno výše. Formát DXF oproti DWG má větší požadavky na prostor. Sice s každou novou verzí se velikost formátu DWG zmenšuje, ale totéž neplatí u formátu DXF, u kterého jsou změny spíše opačné (Tabulka 1)¹. Řešením tohoto problému je komprimace. Komprimace textového formátu DXF zmenší jeho velikost dokonce pod velikost souboru uloženého ve formátu DWG.[6]

Verze AutoCAD	DWG (KB)	DXF (KB)	zip DXF (KB)
2007, 2008, 2009	254	1 636	115
2004, 2005,2006	230	1 635	115
2000, 2000i, 2002	738	1 544	115
R14	741	1 117	108

Tabulka 1 Porovnání velikostí souborů ve formátech DXF a DWG.

Formát souborů DGN byl vytvořen jako nativní formát pro uložení dat z programu MicroStation od firmy Bentley Systems. Na rozdíl od formátů DXF a DWG se styly čar a písma ukládají mimo soubor. Soubory mají koncovku DGN nezávisle na verzi formátu. Existují dva formáty souborů DGN. První verze byla vytvořená v roce 1982 a byla používána až do verze programu sedm, která přišla v roce 2000. Tento formát je dnes označován jako DGN V7, ale před rokem 2000 byl označován pouze jako DGN. Označení je odvozeno od poslední verze MicroStation 7.x. Formát DGN V7 je odvozen od formátu ISFF^v, ale na rozdíl od této předlohy nebyl nikdy zdokumentován. Nástupce formátu DGN V7 je formát DGN V8. Tento formát je navržen tak, aby ho nebylo nutné

¹ Analýzu porovnání velikostí souborů ve formátech DXF a DWG jsme provedli na pasportu budovy FIT (2834 grafických entit) vycházející z importu souboru „VUT_bozetechova_02c_opravene.obr“ do AutoCAD 2007 následně do MicroStation 8.1.

měnit po dobu dalších patnácti až dvaceti let. Formát je velmi flexibilní, protože umožňuje vkládat data ve formátu XML jako dodatečný popis dat. Formát umožňuje sledovat historii změn ve výkresu a ukládat nezávislé modely do jednoho dokumentu. Open Design Alliance, která formát DGN V8 dokumentovala, označila tento formát jako jeden z nejlépe navržených formátů pro CAD systémy v historii. Formát DGN V8 je oproti předchozí verzi mnohem méně náročný na místo (Tabulka 2), ale pokud dojde ke kompresi formátu DGN V7, tak je výsledný soubor menší než ve verzi DGN V8. Toto je obdobné jako u formátu DXF a DWG. V porovnání s DWG má formát DGN V8 rozšíření o digitální oprávnění, digitální podpisy a rozšířené sdílení, které umožňuje víceuživatelský přístup k souborům.

Formát	Velikost (KB)
DWG 2007	254
DWG 2007 – zip	219
DXF 2007	1 636
DXF 2007 – zip	115
DGN V8	316
DGN V8 – zip	260
DGN V7	1 546
DGN V7 – zip	198

Tabulka 2 Porovnání velikostí souborů ve formátech DXF, DWG a DGN.

7 Způsoby rovinného vykreslení

Pokud si položíme otázku, jaké nástroje lze použít pro rovinné vykreslování, tak si nejprve musíme odpovědět, pro jaký operační systém a v jakém jazyku chceme programovat. Z níže vypsanych knihoven lze sice použít podstatnou část na většině operačních systémů, ale někdy je třeba překonávat problémy spojené s použitím v jiném operačním systému, než pro který byla knihovna původně určena. Pro vykreslování lze použít například knihovny Cairo, GDI^{vi}, GDI+ nebo grafickou knihovnu Xlib.

Xlib je grafická knihovna protokolu klienta X Window System. X Windows je grafické uživatelské prostředí, které vzniklo pro prostředí systému Unix, ale dá se použít i v operačním systému Windows. Knihovna Xlib zjednodušuje programování, protože programátor ušetří implementace X protokolu. Xlib dokáže shlukovat požadavky a tím zrychluje vykreslování. Je postavena na programovacím jazyku C.

Grafická knihovna Cairo poskytuje vektorové aplikační rozhraní a mezi hlavní výhody patří její platformová nezávislost. Umožňuje hardwarovou akceleraci, což může zvýšit její rychlost. Knihovna byla implementována v programovacím jazyce C, ale je možné ji použít i v jazycích C++, Lisp, Squeak, Java, Ruby. Knihovna Cairo umožňuje výstup do mnoha prostředí a formátů například X Window System, Win32 GDI, Max OS X Quartz, PNG souborů, Postscript a SVG souborů.

Grafická knihovna GDI je součástí WinAPI. Je prvkem mezi aplikací a hardwarem, který vytváří abstraktní zobrazovací zařízení (Device Context). Programátor pracuje s každým výstupním zařízením bez nutnosti znát parametry zařízení. Tato abstrakce umožňuje pracovat s tiskárnou i monitorem stejně. Grafická knihovna umožňuje vykreslovat grafická primitiva, ale neumožňuje vykreslit přímo okna, menu nebo tlačítka. Tyto funkce umožňuje uživatelský subsystém, který se nachází v knihovně user32.dll a je na GDI postavený.

Grafická knihovna GDI+ nahrazuje a rozšiřuje grafickou knihovnu GDI s příchodem operačního programu Windows XP. Pro použití ve starších verzích systému Windows je nutné použít knihovnu gdiplus.dll. Pro použití grafické knihovny v jiném operačním systému než Windows je nutné simulovat prostředí .NET a Framework. Pro tyto případy je možné využít projekty Mono nebo DotGNU. Výhody GDI jako abstraktní zobrazovací zařízení jsou zachovány. V GDI+ došlo k zapouzdření grafických objektů a operací do objektů. Tímto krokem získalo GDI+ snadnost použití, ale zpomalilo se a zvýšila se paměťová náročnost. Knihovna byla rozšířena o 2D Anti-aliasing, koordinaci pomocí plovoucí čárky, alpha blending, gradiální stínování, podporu formátů JPEG a PNG a v neposlední řadě podporu pro skládání afinní transformace v 2D zobrazení.[7]

8 CAD systémy

Snaha oprostit se od papíru a rýsovacího prkna na konci sedmdesátých let vedla k vytvoření prvních rýsovacích programů, obecně nazývaných CADⁱⁱ. Tyto programy se staly významnou oblastí výpočetní techniky využívanou v praxi. Rozšířily možnosti a zvětšily efektivitu tvorby projektové dokumentace. CAD systémy přinášejí nejenom možnost vytvářet realistickou geometrii objektů, ale i možnost získávat dodatečné informace pomocí analýzy nebo výpočtů nad objekty. Pro vytváření projektové dokumentace budov se používají především CAD systémy.

Zkratka CAD se překládá jako počítačem podporované projektování, ale původní význam byl počítačem podporované kreslení. CAD systémy zastřešují širokou oblast návrhu. CAD aplikace vždy obsahují grafické, geometrické, matematické a inženýrské nástroje pro kreslení plošných výkresů a modelování objektů a dějů reálného světa. Pokročilejší řeší výpočty, analýzy a řízení systémů (výroby, zařízení). Blízkým příbuzným je také oblast počítačových vizualizací, protože virtuální 3D návrhy jsou často klientům prezentovány ve formě fotorealistických vizualizací. [8]

CAD systémy lze rozdělit na obecné a specializované. Mezi obecnými patří k nejpoužívanějším AutoCAD a MicroStation. Specializované lze dále rozdělit podle odvětví. Pro strojírenství se používá SolidWorks nebo Solid Edge, který je pro rovinné kreslení zdarma. V elektronice se používá OrCAD. Pro liniové a dopravní stavby InRoad, InRail nebo AutoCAD Civil 3D. Ve stavebnictví se používá AutoCAD Architecture nebo ArchiCAD. Pro správu budov se používá Topobase firmy Autodesk, Alfa Nemetschek, AMES firmy Berit, GTFacility firmy ASP, FaMa+ firmy TESCO SW a.s., Envis Facility Management firmy DS SOFT OLOMOUC a v budoucnu modul REM pro SAP^{vii}.

8.1 Aplikace AutoCAD

AutoCAD je CAD nástroj od společnosti Autodesk. Patří mezi obecné CAD nástroje. Vývoj začal již v roce 1982 a od té doby vzešlo 23 verzí. AutoCAD je použitelný pouze v operačním systému Windows od verze Release 14 vydané v

únoru 1997. Původně však byl určen i pro Unix a Macintosh. S velkým množstvím verzí tohoto programu vznikl i větší množství verzí formátu pro ukládání dat. DWG^{viii} formát vytvořený pro AutoCAD lze uložit do osmnácti verzí. Problém přechodu je vyřešen tak, že každá verze může načíst starší verzi formátu a každá verze se dokáže uložit do staršího formátu. Takto zpětně kompatibilní je však jen pro několik předchozích verzí. AutoCAD 2004 až 2006 umožňuje uložit pouze do verze 2000, ale AutoCAD 2007 až po verzi AutoCAD R14, což umožňuje otevřít soubor až v 8 verzích tohoto programu. Pokud by uživatel potřeboval uložit do starší verze nebo chtěl přenést data do jiné aplikace, může použít formát DXF^{ix}. Formát DXF byl vytvořen pro výměnu dat mezi programy. Vznikl současně s formátem DWG, který sice přesně reprezentuje nativní data AutoCADu, ale Autodesk nepublikoval specifikaci. Open Design Alliance vytvořila formát OpenDWG, který je plně specifikovaný, ale není úplně kompatibilní. Firma Autodesk dnes nabízí knihovnu ObjectDBX, která slouží právě pro vykreslování, čtení a zápis formátu DWG.

AutoCAD lze použít jako CAD aplikaci, ale i jako platformu pro vývoj uživatelských CAD aplikací. Již první verze AutoCADu byly otevřené pro rozšiřující nadstavby. První se dalo rozšířit program pouze pomocí AutoLISPu, později pomocí jazyka C, objektových jazyků C++ a Visual Basic. Nyní je možné rozšířit jakýmkoli jazykem platformy .NET. Tato otevřenost umožnila vznik tisíců nadstaveb od malých doplňků až po velké CAD nástroje.

8.2 Aplikace MicroStation

System MicroStation je grafický editor určený pro vytvoření a editaci technické dokumentace v různých oborech praxe. Program MicroStation je produkt firmy Bentley Inc. Mezi první uživatele v ČR patřila armáda, která tento systém použila v geodezii. System MicroStation je považován v geodezii za standard.

Program MicroStation používá pro uložení nativní formát souborů DGN^{iv}. Lze však použít i formáty souborů DXF a DWG vytvořené pro program AutoCAD. Od verze MicroStation 8.x je možno ukládat soubory s digitálním oprávněním a digitálním podpisem. Na rozdíl od AutoCADu, který ukládá

soubory do nedokumentovaného formátů DWG, používá MicroStation 8.x otevřený formát (OpenDGN). Tento formát umožňuje použití i třetí stranou v samostatných a na MicroStation nezávislých aplikacích. Program umožňuje také uložení informace o provedených změnách.

MicroStation V8 vyniká jako grafická vývojová platforma umožňující tvorbu specializovaných aplikací, které jsou vyvíjeny jak společností Bentley Systems, tak nezávislými vývojáři. Součástí MicroStationu V8 jsou propracovaná aplikační programovací rozhraní (API), která mohou programátoři využít k rozšíření standardních funkcí nebo k tvorbě specializovaných profesních aplikací. Jedním z dostupných API je i Visual Basic for Applications (VBA) společnosti Microsoft. Visual Basic není pouze výkonným programovacím nástrojem představujícím průmyslový standard, ale také snadno použitelným prostředím, které může využívat prakticky kdokoli. Jednoduchost a široká dostupnost nástrojů VBA poskytuje uživatelům MicroStationu V8 možnost snadných vlastních úprav. MicroStation V8 obsahuje také MicroStation Development Language (MDL), což je nativní programovací prostředí MicroStationu umožňující bezproblémovou integraci profesionálních aplikací s MicroStationem. MDL používají vývojáři Bentley k doplňování funkcí do MicroStationu. Prostředí MDL je dodáváno s MicroStationem a dovoluje vytvářet, editovat a ladit knihovny vlastních funkcí. MicroStation V8 rovněž podporuje aplikace napsané v jazycích C, C++ a Java.[9]

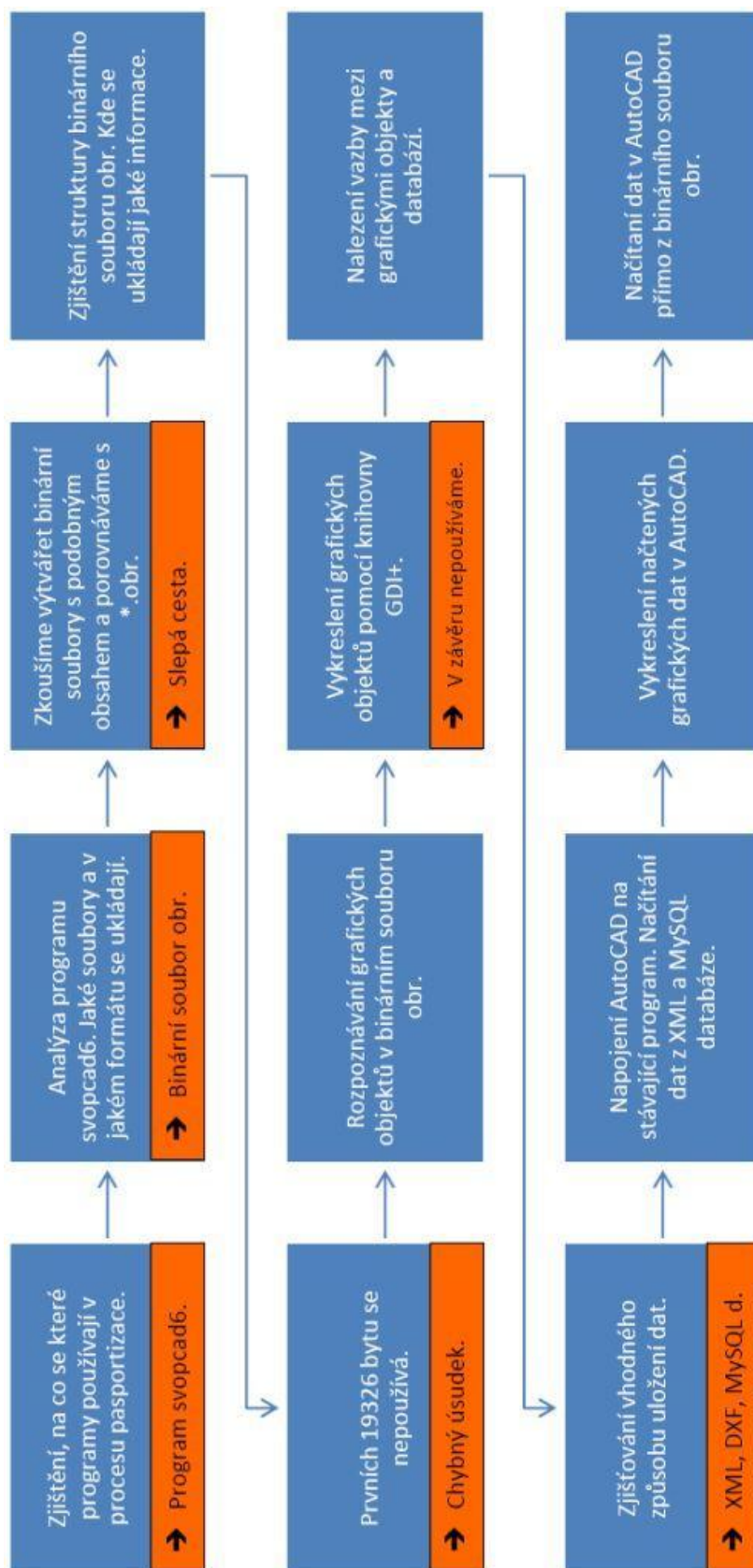
9 Schéma praktického postupu

Ve schématu (**Chyba! Nenalezen zdroj odkazů.**) jsme zobrazili kroky, které vedly od zadání až k cíli naší práce. Ve schématu jsou také zobrazeny dobré i špatné výstupy z jednotlivých kroků. Na začátku celého procesu bylo zjištění současného stavu pasportizace VUT. VUT používá dva grafické editory: GT facility a svopcad6. Pro náš cíl jsme se rozhodli použít program svopcad6, který nám posloužil pro vytváření binárních souborů formátu obr, které byly vstupem pro reverzní inženýrství. Zpočátku jsme se pokoušeli vytvářet binární soubory s podobným obsahem, ale tato cesta se ukázala slepou. Postupným vytvářením grafických objektů a změnami jejich parametrů jsme analyzovali strukturu formátu a následně zjistili klíčové parametry jednotlivých objektů, podle kterých se dají rozpoznat v souborech formátu obr. Výsledkem této části byl chybný úsudek o umístění podstatných informací až za 19326 bytem. Avšak tato část souboru byla vyhrazena informacím o hladinách a textových nástrojích. Další náročnou částí bylo analyzovat parametry jednotlivých grafických objektů. Opět jsme vytvářeli objekty a měnili jejich jednotlivé argumenty. Tímto jsme získali informaci o umístění a způsobu uložení všech parametrů grafických objektů.

Ve chvíli, kdy jsme načetli grafická data, bylo třeba ověřit správnost načítání i u pasportizačních souborů budov VUT. Tyto soubory obsahují tisíce grafických objektů. Z těchto důvodů jsme vykreslili načtená data pomocí knihovny GDI+. Po ověření správnosti načtení grafických dat jsme vybírali vhodný způsob jejich uložení. Vybrali jsme obecný značkovací jazyk XML pro uložení původního způsobu uložení dat. U každého objektu jsme také uložili binární popis pro možnost zpětného opravení případné chyby. Druhým vybraným formátem byl formát DXF. Zvolili jsme ho pro jeho přenositelnost mezi CAD systémy. Výhoda tohoto řešení byla především možnost editace načtených dat.

Velkou změnou při naší cestě k cíli bylo napojení našeho kódu na program AutoCAD. Tímto se vyřešilo nejenom zobrazení grafických objektů, ale i jejich editace a zjednodušené ukládání do formátů DXF a DWG. Data jsme původně načítali ze souborů vygenerovaných z předchozího programu nebo načítali z databáze, která však sloužila pouze jako dočasné úložiště. Toto nepohodlí pro uživatele jsme vyřešili tak, že jsme implementovali načtení formátu obr přímo v programu AutoCAD.

Náročnější částí bylo vykreslení grafických objektů v programu AutoCAD. Museli jsme nejprve převést data z původního systému uložení na způsob vykreslení používaný programem AutoCAD. Tímto byly naplněny všechny cíle, které jsme si na počátku stanovili.



Obrázek 8 Schéma praktického postupu.

10 Analýza binárního souboru pomocí reverzního inženýrství

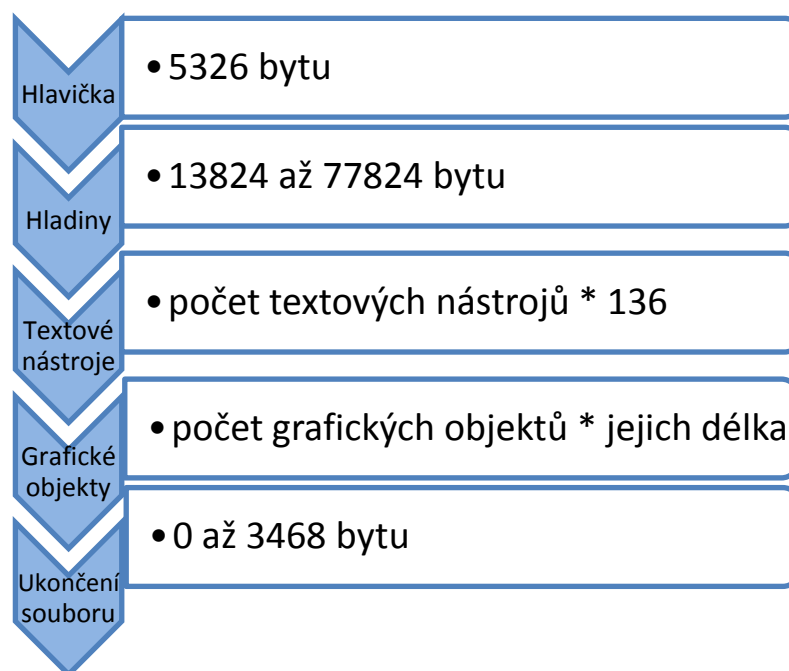
Veškerá grafická data pasportizace VUT jsou uložena do binárních souborů formátu obr. Pro analýzu těchto souborů jsme zvolili metodu reverzního inženýrství. Tato metoda je sice pracná, ale jiná možnost neexistuje. Soubory ve formátu obr byly vytvořeny v programu GT facility, ale pro analýzu je vhodnější program svopcad6, který ukládá do stejného formátu jako GT facility, ale není napojen na databázi a ukládá soubory pouze lokálně. Lokální uložení je výhodné ze dvou důvodů. Prvním důvodem je snadnost přístupu k nakresleným výkresům, které se dají editovat ihned po uložení. Kdežto GT facility ukládá soubory na server, ke kterému jsou nutná oprávnění, které jsme nezískali z bezpečnostních důvodů. Druhým důvodem jsou již zmiňovaná bezpečnostní opatření. Při zásahu do existujících dat může dojít k nenávratné ztrátě velmi cenných dat. V programu svopcad6 lze vytvořit výkres bez vazby na databázi a tím vznikne podobné prostředí jako v programu GT facility.

V prvním kroku při analýze jsme postupně vytvářeli grafické objekty, hladiny a textové nástroje a hledali jejich umístění v binárním souboru. Následovala identifikace jednotlivých objektů v souboru. Po nalezení umístění a identifikaci objektů jsme postupovali podobně i u hledání jednotlivých parametru objektů.

10.1 Umístění dat v binárním souboru

Znalost přesné struktury souboru nám ulehčuje načítání jednotlivých typů informací. Při zjišťování struktury jsme postupně ukládali soubory s různým obsahem a porovnávali je s prázdným souborem. Tímto postupem jsme zjistili, v jakém pořadí se informace ukládají a které informace se ukládají na stejné místo. Tato analýza měla jednu podstatnou nevýhodu. Naše znalost programu

svopcad6 nebyla v počátku naší práce dostatečná a proto se některé vlastnosti, které lze nastavovat a ukládat, nezohlednily v analýze. Předpoklad, že vlastnosti textu se nastavují jako u ostatních grafických objektů přes vlastnosti objektů, byla chybná. Vlastnosti textu se ve skutečnosti přednastavují pomocí textového nástroje, který lze definovat před zadáním textu a který se taktéž ukládá. Druhý parametr souboru, který nebyl ve struktuře zpočátku hledán a rozeznán, byly hladiny. Tyto dva špatné předpoklady vedly k domněnce, že prvních 19326 bytů není využito pro přenos informace. Ve skutečnosti nyní načítáme data již z pozice 5326 (Obrázek 9). Před touto pozicí nepředpokládáme žádná data vztahující se ke grafickým objektům.



Obrázek 9 Struktura souboru formátu obr.

První uložená data jsou jména hladin. Hladin je pevný počet 256 a každý soubor obsahuje informace o jménu každé hladiny. Jméno hladiny má pevnou délku. Z tohoto popisu lze udělat závěr, že vyhrazené místo pro popis hladin má konstantní délku, ale skutečnost, že každá hladina může být uložena čtyřmi způsoby, umožňuje uložit informace o hladinách v rozsahu 13824 až 77824 bytů. Novější soubory mají délku vymezenou pro informace o hladinách na 71194 bytů, ale starší soubory jen 13860 bytů. Tento rozdíl připisujeme různým verzím programu.

V souboru za informacemi o hladinách se nachází informace o textových nástrojích. Počet bytů pro tato data je dynamicky se měnící podle počtu vytvořených textových nástrojů. Velikost se dá tedy odvodit z počtu textových nástrojů krát 136 bytů.

Poslední námi načítanou částí jsou grafické objekty. Místo, které zabírají, ovlivňuje počet a typ grafických objektů. Za těmito daty se nacházejí u starších souborů opět data, která nesouvisí s žádnými grafickými objekty. Jedná se o čtyři byty stále se opakující. Délka je proměnná. V souborech, které jsme vytvořili v programu svopcad6, je délka těchto bytů 3468. Ale u staršího výkresu části historického areálu kláštera fakulty informačních technologií je délka těchto bytů pouze 2318. U nových výkresů je délka těchto bytů vždy nulová.

10.2 Rozpoznání typu pasportizačních dat

Analýza struktury nám ukázala, kde se jaké data ukládají. Dalším krokem byla analýza rozdílů mezi grafickými objekty. Výsledkem bylo jednoznačné rozlišení typu grafického objektu při načítání.

Postup byl podobný jako při analýze struktury souboru. Pro každý objekt jsme vytvořili výkresy, které obsahovaly žádný, jeden a několik těchto objektů. Vzniklé soubory jsme porovnávali a zjistili, jak se objekt ukládá do souboru. Výsledkem je soubor vlastností grafických objektů, které každému načtenému objektu přesně přiřadí typ. Určující vlastností objektu jsou vždy první dva byty, které má každý typ objektu jedinečné (Tabulka 3).

Objekt	Počáteční byty objektu (hex)	Délka (byte)
bod	70 17	28
čára	18 15	28
lomená čára	88 13	>= 46
oblouková křivka	88 18	>= 52
pravoúhlá lomená čára	40 1F	>= 46
křivka	D4 17	>= 52
kruh	44 16	36
elipsa	45 16	36
kruhový oblouk	A8 16	44
eliptický oblouk	A9 16	44
obdélník	7C 15	36
obdélník se zaoblenými hranami	86 15	40
sít	7C 17	40
mnohoúhelník	E0 15	38
Hladina	10 27	54,90 278,304
Textový nástroj	01 00 2E	136

Tabulka 3 Charakteristické vlastnosti objektů.

Další podstatnou vlastností objektů je délka, kterou zabírají v souboru (Tabulka 3). Objekty se dají rozdělit na dva typy podle délky. Prvním typem jsou objekty neměnné délky (Tabulka 4). U těchto objektů je pevný počet parametrů, a tudíž délka zůstává stejná. Mezi tyto objekty patří i hladina, protože její délka je pevná a mění se jen s verzí formátu souboru. Druhým typem jsou objekty, u kterých je proměnný počet parametrů a tudíž proměnná délka. Mezi tyto objekty patří například lomená čára, která svou délku mění podle počtu vrcholů, kterými je zadána. Počet vrcholů může být až 8191. Pokud bude lomená čára složena z 8191 vrcholů, bude zabírat 65558 bytů. Zjistili jsme, že každý vrchol je uložen na 8 bytech a hlavičku tvoří 30 bytů. Další objekty jsou ukládány obdobně. Větší problémy nám činilo načtení celého složeného objektu. Může se skládat z 16 777 215 grafických objektů. Problém nebyl v délce, ale v tom, že grafické objekty v složeném objektu se ukládaly úplně stejně. Rozdíl byl pouze v hlavičce, která tyto objekty předcházela. Tuto hlavičku jsme původně přehlédli a dospěli k špatnému závěru, že program svopcad6 neukládá složené objekty dohromady.

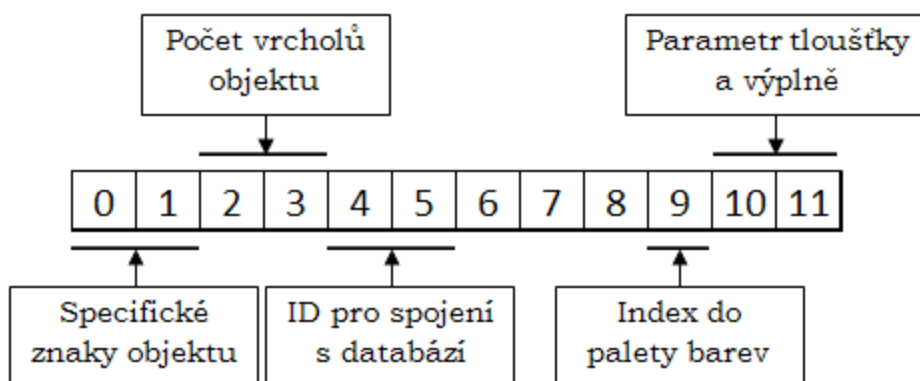
Objekty s pevným počtem parametrů	Objekty s proměnným počtem parametrů
čára	lomená čára
kruhový oblouk	pravoúhlá lomená čára
eliptický oblouk	křivka
obdélník	oblouková křivka
obdélník se zaoblenými hranami	křivka vytvořená tahem ruky
sít	křivka vytvořená analyticky
mnohoúhelník	složený objekt
kruh	
elipsa	
Hladina	
Textový nástroj	

Tabulka 4 Rozdělení objektů podle délky.

10.3 Získání parametrů grafických objektů

Ve chvíli, kdy jsme při analýze souboru ve formátu obr znali strukturu souboru a dokázali rozpoznat od sebe jednotlivé uložené objekty, jsme přistoupili k další fázi analýzy. Tou bylo rozpoznání všech ukládaných parametrů objektů a jejich převod na čitelná data. Zjistili jsme, že hlavička všech grafických objektů má stejný tvar (Obrázek 10). První dva byty jsou určeny pro rozpoznání objektu. V následujících dvou může být uložen počet vrcholů. Počet vrcholů je uložen u objektů s proměnným počtem parametrů. V šestém a sedmém bytu je uloženo číslo, které spojuje grafický objekt s databází. V programu GT facility se grafický objekt propojuje s databází pomocí textu, který přesně určuje umístění objektu v areálech VUT. Například se může jednat o text BOZ01N01102, který popisuje areál Božetěchova, 1. budova, 1. NP, první patro, místnost 02. Tento text se neukládá do souboru obr, ale vkládá se přímo do databáze a do souboru ukládá pouze ukazatel na záznam v databázi. Devátý byte je index do palety barev. Tato paleta obsahuje 256 barev. Index do palety barev převádíme na odpovídající barvu uloženou pomocí tří složek RGB^x. Kdyby barva indexu v GT facility odpovídala barvě v indexu programu AutoCAD, který bychom mohli považovat mezi CAD nástroji za standart, tak by převod nebyl nutný. Barvy indexů jsou však

voleny podle jiného systému, na který se nám nepodařilo přijít. Desátý a jedenáctý byte informuje o vzhledu výplně a tloušťky. Přesto, že lze uložit zvlášť tloušťku a zvlášť styl čáry, tak editory svopcad6 a GT facility neumožňují nakreslit tlustší a zároveň čerchovanou čáru. Vždy lze vybrat jen jednu z možností. Výplň je ukládána i v případě, kdy grafický objekt není uzavřen. Použití prvních 11 bytů je vždy stejné.



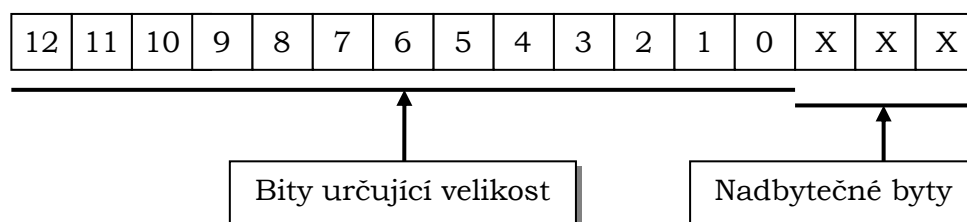
Obrázek 10 Jednotná hlavička grafických objektů.

Rozdílné hlavičky mají uložené hladiny. Jak jsem již popisoval, jsou čtyři různé délky uložených hladin. Uložená data jsou však vždy stejná. Rozdíl v délce dělá pouze počet nulových bytů. Každé jméno hladiny má délku 16 znaků. Tato délka je pevná, a pokud uživatel zadá pouze čtyři znaky, uloží se tyto znaky, za nimi následuje nulový byte a poté text, který tam byl před změnou. Tento fakt způsoboval problémy, protože jsem předpokládal, že se uloží text a poté následují již jen prázdné znaky. Tohoto špatného předpokladu jsem si všiml jen náhodou, když jsem omylem přejmenoval hladinu kratším textem, než byl původně. Stejně vlastnosti má i grafický objekt text, který také může obsahovat minulý text za nulovým bytem. Na příkladu uložení hladiny lze ukázat zbytečnou velikost většiny objektů. Pokud započteme dva byty na rozpoznání, jeden jako číslo hladiny a šestnáct bytů pro jméno hladiny, tak dostaneme číslo devatenáct bytů. Pokud jsou některé hladiny uloženy na 304 bytech, tak je jasné, že místo není využito zcela efektivně.

Dalším zajímavě uloženým grafickým objektem je text. Délka textu není nijak omezena. Při ukládání do souboru formátu obr je text rozdělen na texty o délce 57 znaků. To, jestli za uloženým textovým blokem pokračuje další

textový blok, který má být spojen s předešlým blokem, se pozná podle posledního bytu. Pokud je poslední byte roven číslu 87, tak další blok je navazující. Tímto způsobem může být uložen libovolně dlouhý text.

Skupina grafických objektů, které obsahují proměnný počet parametrů, vždy obsahuje v hlavičce číslo, které udává počet těchto parametrů. Tuto informaci používáme pro výpočet délky zabírané grafickým objektem. Původní předpoklad byl, že všechny objekty budou mít počet parametrů uložený stejným způsobem, ale ukázalo se, že jsou mezi nimi malé rozdíly. Například u lomené čáry jsou poslední tři bity nevyužité (Obrázek 11), a proto při převodu tyto bity nezahrnujeme. Maximální počet bodů je tedy 8 192. Možná by nyní každý předpokládal, že grafický objekt pravoúhlá lomená čára, která je vlastně pouze lomenou čárou, která při kreslení umožňuje zalomení do pravých úhlů, se bude ukládat totožně. Tento předpoklad je však špatný. U pravoúhlé lomené čáry jsou poslední čtyři bity nevyužity. Takovéto malé rozdíly má mnoho objektů.

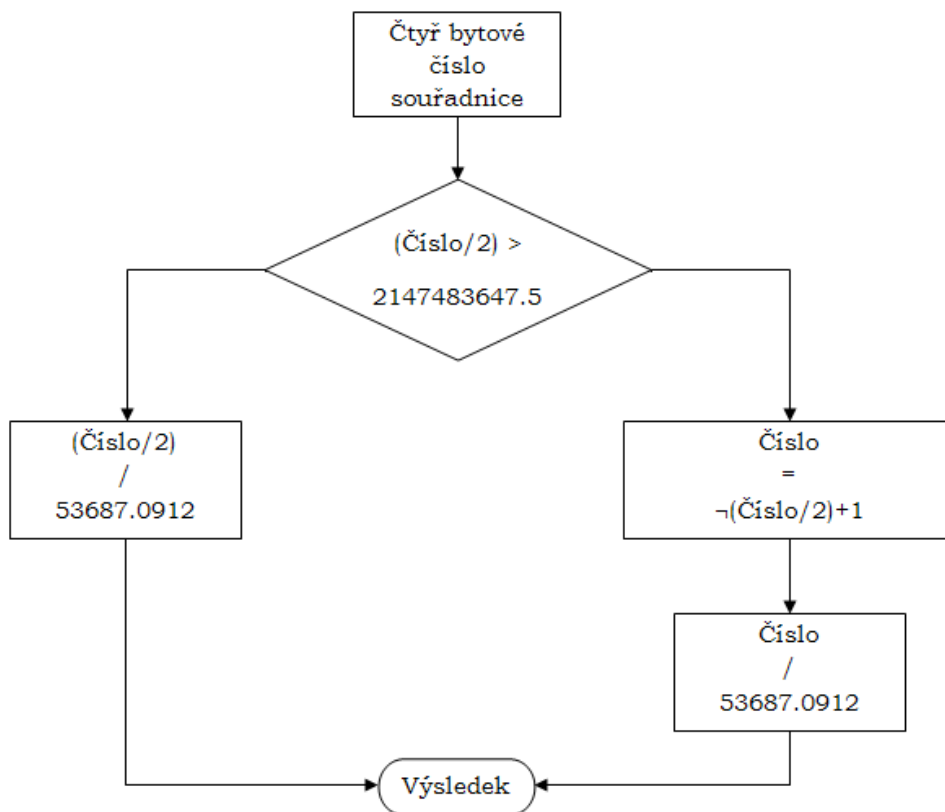


Obrázek 11 Rozložení bitu určujících počet vrcholů lomené čáry.

Při analýze byly jedním z velkých překvapení grafické objekty založené na křivce. Při kreslení křivky se zadávají body a je vykreslována interpolační křivka. Při ukládání nejsou ukládány tyto body, ale křivka je převedena na mnoho úseček, které se následně uloží. Při kreslení křivky vytvořené tahem ruky se při tahu myši vytváří lomená čára s velkým počtem vrcholů. Při tahu se s rostoucí délkou křivky postupně snižuje počet vytvářených vrcholů. Toto omezení není asi cílené, protože nižší počet vrcholů je způsoben pouze pomalostí programu svopcad6, který celou čáru překresluje při každém pohybu a tudíž nezaznamená celý pohyb myši.

Pokud bychom měřili časovou náročnost podle poměru čas k velikosti načítané informace, nejnáročnější v tomto ohledu by bylo načtení polohy. Všechny objekty mají jeden nebo více bodů udávajících polohu. Každý bod je definován souřadnicí x a y. V editoru svopcad6 lze nastavit nejmenší kladnou

souřadnici na hodnotu 0.001 a největší kladnou na 10 000. Stejný rozsah je i pro záporné hodnoty. Soubor uložený ve formátu obr má souřadnici uloženou ve čtyřech bytech. Čtyři byty odpovídají číselnému rozsahu -2 147 483.000 až 2 147 483.000. Soubor obr však umožňuje uložit souřadnici v rozsahu 10 000.000 až -10 000.000. Tato neshoda byla příčinou velmi obtížného rozpoznání načítaných souřadnic. Pro získání správné hodnoty je nutný převod, ve kterém se nachází několik na první pohled nejasných čísel (Obrázek 12). Protože je číslo uloženo v doplňkovém kódu, nejprve testuji, v jaké části rozsahu se nachází. Pokud je číslo v rozsahu 0 až 2147483647.5, je to číslo kladné, ale pokud je číslo v rozsahu 2147483647.5 až 4294967295, je záporné a musí se převést na kladné. Záporné číslo se neguje a je k němu přičtena jednička (Pravá část diagramu). Po převodu na kladné číslo, pokud bylo záporné, následuje vydělení kladného čísla číslem 53687.0912. Číslo 53687.0912 vzniklo vydělením čísla 2147483647.5 (maximální hodnota uloženého kladného čísla) číslem 40000 (maximálním číslem uloženým ve formátu obr). Výsledek se poté vynásobí -1, pokud bylo číslo záporné. Kdyby svopcad6 ukládal v rozsahu 10000.000 až -10000.000 bez nadbytečných převodů, vešlo by se číslo uchováající souřadnici do 7 bytů. Při použití 7 bytů by byla úspora jak místa, tak v rychlosti načítání. Jeden výkres části historického areálu kláštera fakulty informačních technologií obsahuje 2652 položek. Z převážné části jsou tyto položky čáry a lomené čáry s průměrným počtem bodu 6. Celkový počet uložených bodů je přibližně 8840. Každý bod se skládá ze dvou souřadnic, takže celková úspora místa je 17680B. Celková velikost souboru je 140 288B.



Obrázek 12 Převod čtyř bytů obsahující souřadnici na číslo.

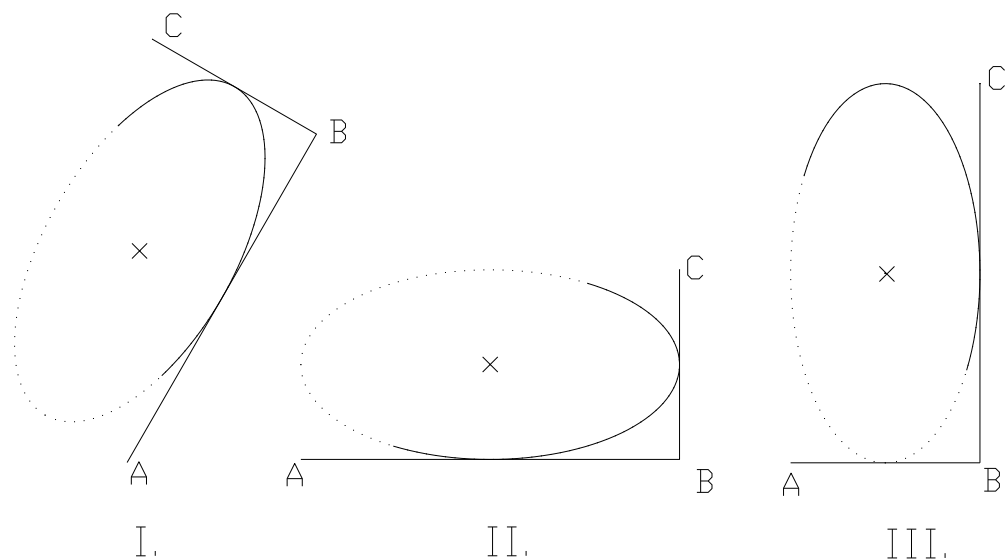
11 Analýza získaných dat

Pomocí reverzního inženýrství jsme úspěšně načetli data z binárního souboru formátu obr. Pro pochopení, jakým způsobem jsou data uložena, jsme opět použili reverzní inženýrství. U jednoduchých objektů jako je čára, lomená čára nebo bod jsou uloženy souřadnice jednotlivých bodů a při vykreslení nenastane žádný problém. Avšak grafické objekty jako je elipsa, čtverec, eliptický oblouk, n-úhelník jsou uloženy pouze pomocí tří bodů. Z tohoto důvodu bylo nezbytné provést analýzu způsobu uložení, která byla nutná pro vykreslení. Způsob uložení křivky je také jiný, než by se dalo očekávat, ale následné vykreslení spíše ulehčí.

11.1 Způsoby uložení grafických objektů

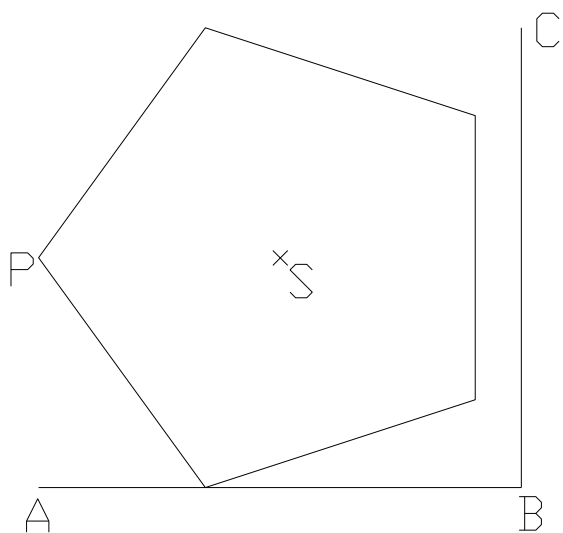
Grafické objekty elipsa, kružnice, eliptický oblouk a kruhový oblouk mají velmi podobné uložení. Eliptický oblouk a kruhový oblouk navíc obsahují počáteční a koncový úhel. Tyto objekty popisují pouze tři body. Pomocí těchto bodů je zachycen tvar a pozice. Na obrázku (Obrázek 13) jsou znázorněny tři body, jejichž spojnice opisují elipsu. Z bodů A a C lze získat střed elipsy. Ze středu přímky CB a středu elipsy získáme délku hlavní poloosy. Obdobně ze středu přímky AB a středu elipsy získáme délku vedlejší poloosy. První elipsa je natočena o úhel 60° , druhá elipsa je natočena o 0° a třetí elipsa je natočena o 90° . U třetí elipsy, která je natočena o 90° , je zvláštní umístění bodů A, B a C. Dalo by se předpokládat, že přímka AB bude opět podél delší strany a BC podél kratší, ale umístění je posunuté. Natočení elipsy lze zjistit z natočení přímky AB k ose x. U tohoto grafického objektu jsou tři body nejúspěšnějším řešením.

Způsob uložení kružnice na rozdíl od elipsy není příliš úsporný. Kružnice je opět uložena pomocí tří bodů, které ji opisují pomocí přímek mezi těmito body. Pro uložení kružnice by přitom stačily dva body. Natočení kružnice nemá význam, stejně tak jako třetí bod. Kružnicový oblouk je vyřešen, stejně jako u elipsy, dvěma body, určujícími počáteční a koncový úhel.



Obrázek 13 Způsob uložení elipsy s různým natočením.

N-úhelník patří mezi další zajímavě uložené objekty. Předpoklad, že objekt je uložen pomocí středu, jednoho z vrcholů, počtu hran případně úhlu natočení, byl nesprávný. S úhlem natočení nemusíme počítat, protože v programu svopcad6 je možno zadat tento úhel, ale po potvrzení se vždy zadané číslo přepíše na nulu. Uložení tohoto grafického objektu je opět pomocí tří bodů (Obrázek 14). Z bodů A, B a C lze zjistit střed S a počáteční bod P. Se znalostí počtu hran není již obtížné získat i zbylé vrcholy.



Obrázek 14 Způsob uložení n-úhelníku.

Dalším grafickým objektem, který není uložen předpokládaným (způsobem, je křivka. Program svopcad6 umožňuje kreslit standardní, obloukovou, od ruky a analytickou křivku, která se zadává rovnicí. Všechny tyto křivky se ukládají stejným způsobem. Řídící body, pomocí kterých je křivka zadána, nejsou uloženy, křivka je rozložena na přímky a ty jsou ukládány. Při analýze vyvstala otázka, zda řídicí body zpětně rekonstruovat a uložit jen tyto body. Rozhodli jsme se však zachovat veškeré vlastnosti všech objektů, tak jak je program ukládá. Tento způsob nechává volbu způsobu uložení, až na fázi, kdy data budeme vkládat do nového systému pro pasportizaci.

11.2 Napojení pasportizačních dat na databázi

Jak již bylo zmíněno, hlavním cílem této práce je získat vazbu grafických objektů na databázi. Touto vazbou jsou propojena grafická data, například obdélník, s popisem v databázi, například číslo místnosti, rozvrh nebo jméno nájemce. Program svopcad6 umožňuje grafické objekty exportovat do vektorových formátů, ale vazba na databázi se ztratí. Při hledání vazby na databázi bylo velkou komplikací nemožnost vytvářet data, která by obsahovala tuto vazbu. Program svopcad6 umožňuje práci pouze s lokálními soubory. K programu GT facility jsme neměli oprávnění vytvářet nové soubory a k stávajícím jsme neměli přístup. Tento velký hendikep jsme kompenzovali několika ukázkovými soubory, které tuto vazbu obsahovaly. Z analýzy těchto souborů vzešla hypotéza, že index je do databáze uložen na čtvrtý a pátý byte. Tato hypotéza se podařila ověřit až v závěru, kdy jsme obdrželi několik malých výkresů, u kterých jsme znali popisky místností i identifikační čísla s nimi spojená v databázi.

Vazba grafického objektu je vždy číslo v rozsahu 1 až 65 536. Pro spojení s informací v databázi je nutné znát i identifikační číslo, které má přiřazen každý soubor, protože každý soubor začíná číslovat tyto vazby od čísla 1. Proto je nutný pro získání informace z databáze právě název souboru nebo jeho identifikační číslo, protože bez něj existuje mnoho duplicitních záznamů uložených v databázi.

11.3 Struktura pro načítání pasportizačních dat

Při načítání grafických dat ze souborů formátu obr jsou načítána data, která mají společné vlastnosti. Mezi společné vlastnosti patří identifikační číslo umožňující spojení s databází a vrstva. Výhodné bylo využít pro grafické objekty polymorfismu, který dovoluje zacházet stejně s podobnými objekty. Toho můžeme využít při exportu do XML, vykreslení v programu AutoCAD nebo při samotném načtení. Vytvořili jsme proto abstraktní třídu, která obsahuje abstraktní metody pro vykreslení, vykreslení v programu AutoCAD, binární import a export nebo ukládání do databáze. Následně jsme vytvořili pro každý grafický objekt třídu, odvozenou od této abstraktní třídy, a překryli jednotlivé metody podle vlastností objektu. Vytvořením abstraktní třídy, ze které ostatní třídy dědí společné vlastnosti a překrývají abstraktní metody, jsme mohli použít pro načítání velkého množství grafických objektů generickou kolekci. V této části jsme vycházeli z informací v knize [10].

12 Zobrazení načtených dat

V této diplomové práci jsme zatím převážně řešili načítání dat z formátu obr. Získání dat má rozhodující vliv na výsledek této práce. Část, kterou nyní budeme řešit, je však také důležitá. Pokud bychom předpokládali, že všechna data načtena ze souborů formátu obr byla načtena správně, pak by tato část byla opravdu nepodstatná. Mohli bychom zobrazit grafická data pouze jako výpis ze souborů XML nebo DXF, ale pokud se jedná o výkresy, které obsahují až 12 000 grafických objektů, tak touto metodou nejsme schopni odhalit při načítání žádnou chybu. Grafickým zobrazením získáme ideální možnost kontroly načtených údajů.

K zobrazení grafických dat můžeme přistoupit dvěma způsoby. První způsob je kompletně vyřešit vykreslení s možností jednoduché editace grafických objektů a následným uložením provedených změn do vhodného formátu. Druhou možností je vytvořit DLL knihovnu, která dokáže načíst data ze souboru formátu obr a tuto knihovnu importovat do již existujícího CAD systému. Součástí této knihovny musí být část, která grafická data vykreslí. Tudíž by se mohlo zdát, že prvním i druhé řešení je stejně náročné. Velký rozdíl je však v kvalitě editace. Editace grafických prvků může být vyřešena velmi jednoduše, ale u této části vývoje by mělo vstoupit na scénu nejdůležitější kritérium a to pohled koncového uživatele. Podstatným důvodem vzniku celé této práce je uživatelsky nepřívětivé prostředí předchozího řešení. Mohli bychom být troufalí a myslet si, že kvalitní rovinný editor pro pasportizaci nebude obtížné sestavit. Volbou prvního řešení by zajisté vznikl kvalitní rovinný editor, ale nevznikne volbou druhého řešení kvalitnější? Pokud je možnost použít CAD nástroje, které se vyvíjejí od roku 1982 a na jejich vývoji se podílí stovky lidí, měli bychom této možnosti využít.

12.1 Zobrazení načtených dat knihovnou GDI+

V počátku naší práce bylo třeba ověřit správnost načítaných dat. Velmi efektivní řešení této potřeby je vizuální kontrola. Pro zobrazování rovinných

grafických objektů jsme vybrali knihovnu GDI+. Důvody výběru jsou popsány v sedmé kapitole.

Protože předpokládáme vykreslování velkého množství grafických objektů, řádově tisíce, rozhodli jsme se otestovat vliv kvality vykreslení na rychlost. Výsledkem je možnost nastavit parametry ovlivňující kvalitu tak, aby bylo vykreslení rychlé i při vykreslování tisíců prvků nebo bylo vykreslení natolik kvalitní, aby výstup bylo možné použít při dalším zpracování nebo uložit ve vysoké kvalitě.

Knihovna GDI+ má několik vlastností, které ovlivňují kvalitu vytvářené grafiky. První takovou vlastností je „SmoothingMode“, který nastavuje vyhlazování hran (antialiasing) při kreslení přímek a křivek. Při implicitním nastavení se hrany nevyhlazují, což způsobí, že šikmé přímky a křivky se nakreslí stupňovitě. Antialiasing způsobí, že místa napravo a nalevo vedle stupňů budou lehce stínovaná nebo vyplněna slabší transparentní barvou. Linie tak na pozorovatele působí hladším dojmem. Nejvyšší kvality dosáhneme při nastavení „HighQuality“ a největší rychlosti při nastavení „HighSpeed“. [11]

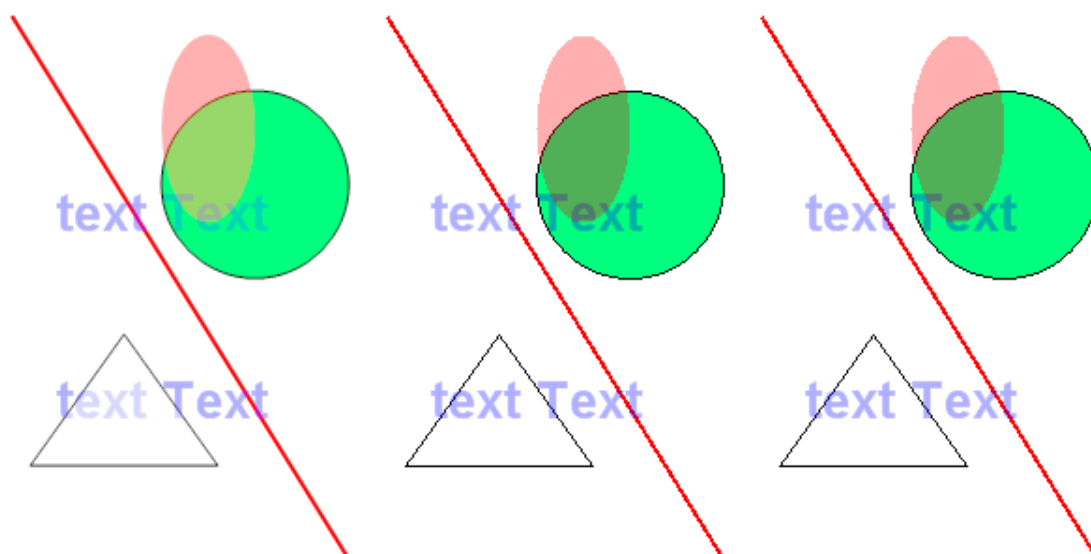
Dalším parametrem rozšiřujícím nastavení antialiasingu je „PixelOffsetMode“. Tento parametr nastavuje posunutí pixelu, čímž dosahuje změkčení stínů. Nejvyšší kvality dosáhneme při nastavení „HighQuality“ a největší rychlosti při nastavení „HighSpeed“ podobně jako u vlastnosti „SmoothingMode“. [11]

Vlastnost „InterpolationMode“ určuje interpolaci, která se používá při zvětšování nebo zmenšování obrázku. Nemá žádný vliv na kreslení grafických objektů jako přímek, obdélníků a kruhů. Nejlepší kvality dosáhneme pomocí parametru „HighQualityBicubic“. [11]

Vlastnost „CompositingQuality“ ovlivňuje kvalitu, pokud se kreslí transparentní grafické objekty. Při nastavení vysoké kvality parametrem „HighQuality“ je kontrast transparentně kreslených objektů podstatně vyšší než při nízké kvalitě. Nízká kvalita nastavená parametrem „HighSpeed“ vede k výraznému zrychlení. [11]

Srovnání vykreslených objektů jde vidět na obrázku (Obrázek 15). Defaultní nastavení kvality vykreslení grafických objektů má stejnou kvalitu

jako nastavení HightSpeed. Nej kvalitnější je první z trojice obrázků, u kterého byly všechny parametry nastaveny na nejvyšší rychlost.



Obrázek 15 Různá kvalita vykreslení. HightQuality, HightSpeed, Default.

Rychlost vykreslených dat je závislá na kvalitě nastavení vykreslování. Srovnání dopadlo podobně jako v předchozím případě (Tabulka 5). Obrázek (Obrázek 15) byl pro každou kvalitu vykreslen 5000x. Výsledné časy jsou dosti rozdílné, pokud porovnáváme vysokou kvalitu s defaultním nastavením a vysokou rychlostí. Rychlost u Default a HightSpeed je však stejná.

Kvalita	Čas
HightQuality	44,98s
HightSpeed	17,91s
Default	16,31s

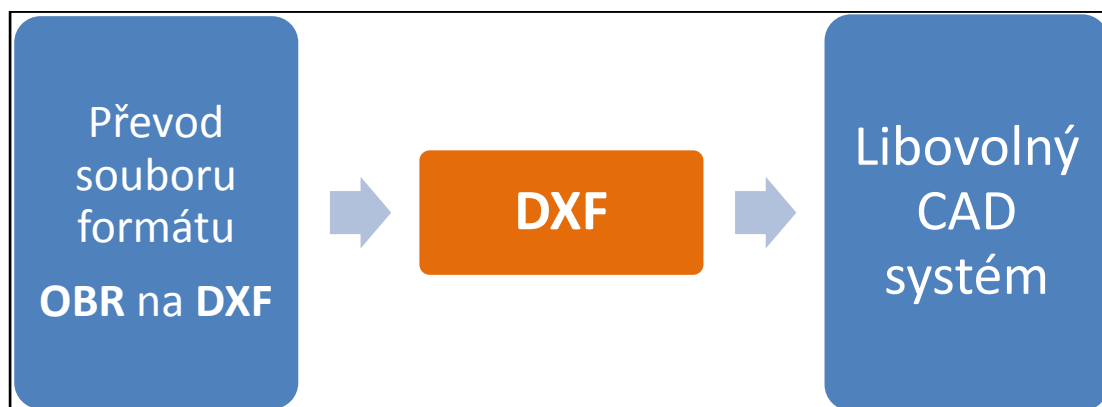
Tabulka 5 Srovnání časové závislosti na kvalitě.

Zobrazení načtených grafických objektů pomocí knihovny GDI+ jsme používali, dokud jsme nepřesunuli veškerý kód do programu AutoCAD. Přesunem jsme získali vylepšené možnosti zobrazení a hotové řešení editace. Zobrazení dat pomocí knihovny GDI+ nebylo zcela dokončeno. Zobrazovali jsme pouze jednoduché tvary jako přímka, složená přímka. Přesto toto zobrazení bylo dostačující pro zobrazení starších výkresů, kde se používaly v 95% pouze tyto grafické objekty. V současnosti není kód pro vykreslení pomocí GDI+ dokončen a používán, protože byl nahrazen kódem pro vykreslení v programu AutoCAD.

12.2 Zobrazení načtených dat programem AutoCAD

Výhody využití programu AutoCAD jako editoru souboru formátu obr jsme popsali na začátku této kapitoly. Původním záměrem bylo využít program AutoCAD jako editor, ale vstupem pro tento program neměl být soubor formátu obr, ale námi vytvořený soubor formátu DXF, který bychom generovali ze získaných dat (Obrázek 16). V tomto případě nebyla nutná instalace programu AutoCAD, ale dal se použít libovolný CAD systém, který umožňuje otevřít soubor formátu DXF. Vycházeli jsme z předpokladu, že formát DXF je kompletně zdokumentovaný a natolik rozšířený, že nebude problém ho importovat do libovolného CAD systému nebo do CAFMⁱ systému, který bude VUT využívat v budoucnosti. Prvním krokem při zobrazení dat ze souboru formátu obr by bylo získání dat a uložení do souboru formátu DXF. Tento krok by byl proveden v našem programu. V druhém kroku by uživatel zvolil libovolný prohlížeč nebo editor DXF souborů. Výhoda tohoto řešení byla hlavně v možnosti volby libovolného editoru a nevýhoda převodu byla především v nutnosti použití více programů.

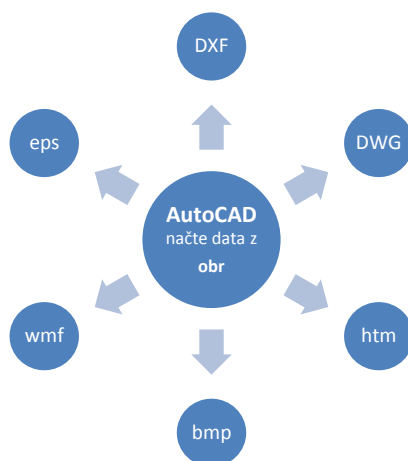
Tento přístup jsme implementovali pouze částečně, protože jsme se rozhodli využít možnosti vložení našeho kódu do programu AutoCAD a tím umožnit otevřít soubor formátu obr přímo v programu AutoCAD.



Obrázek 16 Zobrazení grafických dat převodem do formátu DXF.

Druhá možnost zobrazení grafických objektů je přímo implementovat kód pro aplikaci AutoCAD. Navrhli jsme dynamickou knihovnu, která po načtení v programu AutoCAD umožňuje načítat a vykreslovat data přímo ze souboru formátu obr. Po vykreslení těchto dat si uživatel může vybrat libovolný formát, do kterého data uloží. Výhodou tohoto řešení je především

uživatelská nenáročnost. Stačí zadat dva příkazy a poté může uživatel editovat grafická data nebo je uložit do všech verzí formátu DXF a DWG nebo exportovat jako obrázek (Obrázek 17). AutoCAD umožňuje také publikovat výkres jako internetovou stránku.



Obrázek 17 Možnosti uložení dat v programu AutoCAD.

Při vykreslování grafických dat v programu AutoCAD jsme řešili problém, jak převést způsob uložení grafických dat do způsobu vykreslení grafických dat v programu AutoCAD. Pokud je elipsa uložena třemi body (Obrázek 13) a v AutoCADu se dá zadat pouze pomocí parametrů střed, vektor hlavní poloosy a poměr hlavní a vedlejší poloosy, tak je nutné všechny elipsy před vykreslením převést na nový způsob uložení dat. Grafický objekt n-úhelník se dá kreslit v AutoCADu, ale po nakreslení je okamžitě převeden na křivku a nelze s ním již manipulovat jako s n-úhelníkem. V tomto případě opět tři body a počet hran převedeme na vrcholy n-úhelníku a z nich vytvoříme uzavřenou křivku. Touto operací však ztrácíme informaci, o jaký objekt se jednalo. Tento problém je taktéž u vykreslení kružnice. Kružnici vykreslujeme jako elipsu, která má poměr hlavní a vedlejší poloosy roven jedné. Pokud vykreslujeme lomenou čáru, opět využíváme křivky. Tato ztráta informace o typu objektu však nebude mít vliv na pochopení výkresu jako celku, protože není důležité, který grafický objekt byl vybrán pro nakreslení čtverce, ale je důležité, že uživatel pozná, že se jedná o čtvercovou místnost.

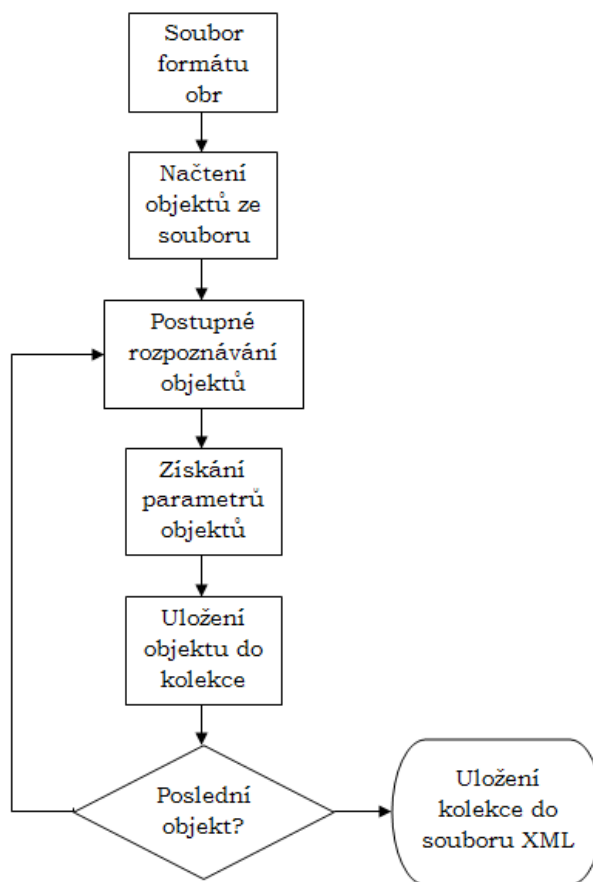
Zásadní otázkou při vykreslování grafických objektů bylo, jak přiřadit vazbu na databázi k vykreslenému objektu tak, aby byla uchována i po

uložení do souboru formátu DXF a DWG. Nalezli jsme vhodné řešení. Ke každému objekt lze přiřadit informaci spojující tento objekt s internetovou stránkou. Při ukázání na tento objekt se automaticky ukáže internetová stránka a její popis. Této vlastnosti jsme využili a ukládáme místo internetové adresy identifikační číslo místnosti spojující grafický objekt s informací v databázi. Možnost definovat vlastní parametry by nás ochudila o možnost přenést výkres do libovolného CAD systému, tudíž využití parametru internetové adresy se jeví jako ideální řešení.

13 Vhodný způsob uložení dat

Výstupem této práce by měla být data převedená do čitelného formátu, který budeme moci importovat do budoucího řešení pasportizace budov VUT. Toto je jediný požadavek na uložená data. Vzhledem k tomu, že neznáme budoucí software a formát, který bude podporovat, tak jsme se rozhodli pro výstup zvolit formát XML, DXF a DWG.

Formát XML jsme zvolili, protože umožňuje uložit jakákoli data. Pomocí serializace ukládáme celou kolekci načtených grafických dat přímo do formátu XML. Při načítání jednotlivé objekty rozpoznáme a uložíme do kolekce. Když načteme poslední objekt, tak celou kolekci uložíme do souboru formátu XML. Ke každému objektu ukládáme navíc i kopii binární reprezentace ze souboru formátu obr. Hlavní výhodou tohoto řešení je možnost využívat všechna načtená data a možnost zpětně zkontrolovat načtení z uložené binární části.



Obrázek 18 Převod a uložení grafických dat do souboru XML.

Další způsob uložení dat jsme zvolili formáty DXF a DWG. Toto uložení lze provést po načtení souboru obr do aplikace AutoCAD. Výhoda oproti XML je ta, že data lze zobrazit ve většině CAD systémů a editovat je. Nevýhodou tohoto řešení je nemožnost přenést veškeré informace z původního souboru obr do nového způsobu uložení. Pro formát DXF a DWG neexistuje grafický objekt kružnice nebo n-úhelník. Při převodu na tyto formáty ztratíme některé informace. Data se sice zobrazí úplně stejná, ale ve skutečnosti to bude složená čára a ne n-úhelník, co uvidí uživatel.

Kombinací těchto dvou možností získáme ideální řešení. Grafická data lze zobrazit a zkontrolovat správnost řešení díky formátům DXF a DWG. Do budoucího systému půjde převést všechny informace, které bude možno importovat.

14 Závěr

V diplomové práci jsem se zabýval pasportizací budov VUT. Hlavním cílem mé práce bylo připravit data pro přenos ze současného řešení pasportizace VUT do budoucího. VUT používá program GT facility a svopcad6 jako editory výkresů pro pasportizaci. V prvním kroku jsme zjistili, že veškerá pasportizační data program svopcad6 ukládá do binárního souboru formátu obr. U programu svopcad6 jsme zjistili všechny možnosti, které poskytuje. Následně jsme pomocí reverzního inženýrství analyzovali formát obr. Při analýze souboru jsme postupně odhalovali umístění ukládaných dat. Výsledkem byl popis struktury souboru a popis způsobu rozpoznání jednotlivých grafických objektů v souboru. Nejnáročnější fází analýzy bylo rozpoznání parametru grafických objektů a jejich převod na čitelné informace. Obzvláště komplikovaný byl způsob uložení souřadnic. Stěžejní částí bylo nalézt propojení grafických objektů s databází. Bylo nutné vypátrat umístění identifikačního čísla, propojujícího objekt s informací uloženou v databázi, u všech typů objektů. Ze získaných informací o formátu obr jsme mohli vytvořit program, který data z jakéhokoli souboru formátu obr dokáže načíst.

Požadavkem bylo zobrazit načtené grafické objekty a umožnit jejich jednoduchou editaci. Proto jsme vytvořili knihovnu pro rozpoznání, načtení a zobrazení grafických dat z formátu obr, která lze použít v programu AutoCAD. Tímto řešením jsme získali pro uživatele jednoduchou možnost načíst a zobrazit soubory formátu obr, ale hlavní výhodou tohoto řešení je využití jednoho z nejkvalitnějších programů pro editaci tvorby projektové dokumentace. Zároveň jsme zajistili možnost uložení načtených dat do formátů, které dokáže načíst téměř každý CAD systém. Současně s touto možností uložení jsme se rozhodli ukládat nepozměněná data do souboru formátu XML, protože vykreslením v programu AutoCAD se některé informace o objektech ztrácejí. Do souboru XML ukládáme ke každému načtenému objektu i jeho původní binární reprezentaci. Tato informace může být využita, pokud bude v našem řešení nalezena chyba nebo nekompletnost načtených informací.

Výsledkem této práce je možnost získat informace z původně nečitelného binárního souboru, ve kterém jsou uložena cenná pasportizační data naší univerzity. Veškerá načtená data jsme zobrazili v programu AutoCAD, ve kterém lze libovolně tato data editovat. Výstupem této práce jsou tedy data uložená z programu AutoCAD s tím nejdůležitějším atributem a to vazbou na databázi. Pro naši univerzitu je velkým přínosem také oprostění se od závislosti na firmě ASP.

Práce, která nás čeká po vybrání nástupce programu GT facility, je importovat data s vazbou na databázi do nového systému. Tato práce však bude už jen třešnička na dortu.

Při tvorbě této diplomové práce jsme se mnohokrát dostali do slepé uličky nebo na okraj hluboké propasti. Nejenom v analýze jsme podléhali špatným předpokladům nebo závěrům, ale i při možnostech uložení, zobrazení a editace jsme se vydali mnoha směry, které ne vždy vedli k ideálnímu řešení.

Pokud se ohlédneme za provedenou práci, musíme uznat, že bylo vynaloženo mnoho zbytečného úsilí na řešení, která jsme nakonec nepoužili.

K tomuto závěru musí dojít každý řešitel, který šel neprošlapanou cestou a sám hledal ideální řešení. Doufám, že já jsem se k té ideální cestě co nejvíce přiblížil.

Literatura

- [1] Vyskočil V., Štrup O., Pavlík M.: Facility management a Public Private Partnership, 1, Praha, Kamil Mařík – PROFESSIONAL PUBLISHING, 2007
- [2] Vyskočil V., Štrup O.: Podpůrné procesy a snižování režijních nákladů (Facility Management), 1, Praha, Kamil Mařík – PROFESSIONAL PUBLISHING, 2004
- [3] Vysoké učení technické v brně - Apollo nápověda, Dokument dostupný na URL <http://is.vutbr.cz/napoveda/apollo.php> (4. 1. 2008)
- [4] Reverse engineering, Dokument dostupný na URL http://en.wikipedia.org/wiki/Reverse_engineering (4. 1. 2008)
- [5] Fořt P., Kletečka J.: AutoCAD 2006, 1, Brno, Computer Press, a.s., 2006
- [6] Spielmann M., Špaček J.: AutoCAD Názorný průvodce pro verze 2004 až 2007, 1, Brno, Computer Press, a.s., 2006
- [7] Graphics Device Interface, Dokument dostupný na URL http://en.wikipedia.org/wiki/Graphics_Device_Interface (4. 1. 2008)
- [8] Computer aided design Dokument dostupný na URL http://cs.wikipedia.org/wiki/Computer_aided_design (13. 1. 2009)
- [9] MicroStation [www.gissoft.cz] - informace o programu, Dokument dostupný na URL <http://www.gissoft.cz/MicroStation/MicroStation> (15.1.2009)
- [10] Troelsen A.: C# a .NET 2.0 profesionálně, 1, Brno, ZONER Software s.r.o., 2006
- [11] Bayer J.: C# 2005 Velká kniha řešení, 1, Brno, Computer Press, a.s., 2007

Seznam příloh

Příloha 1. Manuál k ovládání programu v prostředí AutoCAD.

Příloha 2. CD obsahující text diplomové práce, zdrojové texty program a funkční program pro převod, uložení a zobrazení dat ze souboru formátu obr.

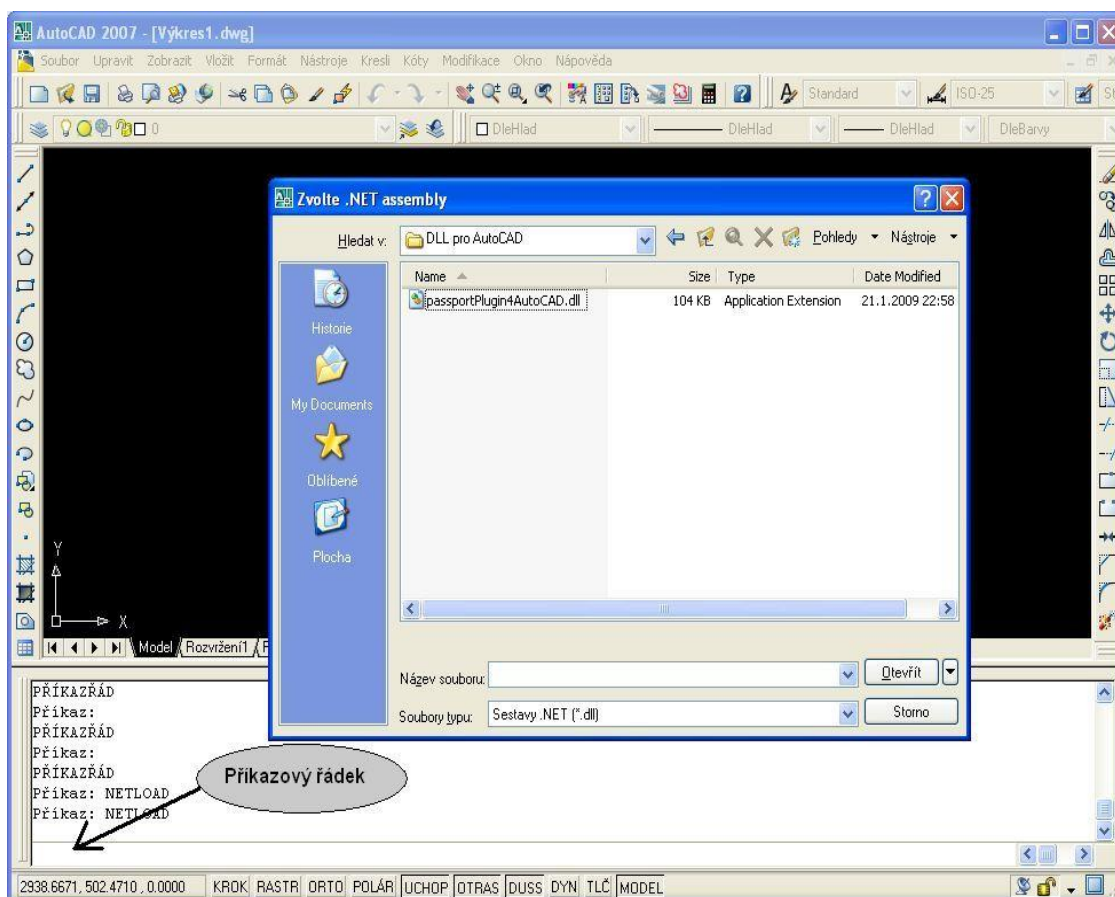
Seznam zkratk

-
- i Computer aided facility management (CAFM)
 - ii Computer-aided design (CAD), původně computer-aided drafting (CAD)
 - iii Extensible markup language (XML)
 - iv Design (DGN)
 - v Intergraph standard file format
 - vi Graphical device interface (GDI)
 - vii Systems - applications - products in data processing (SAP)
 - viii Drawing (DWG)
 - ix Drawing exchange format (DXF)
 - x Red green blue (RGB)

Příloha 1.

Manuál k ovládání programu v prostředí AutoCAD

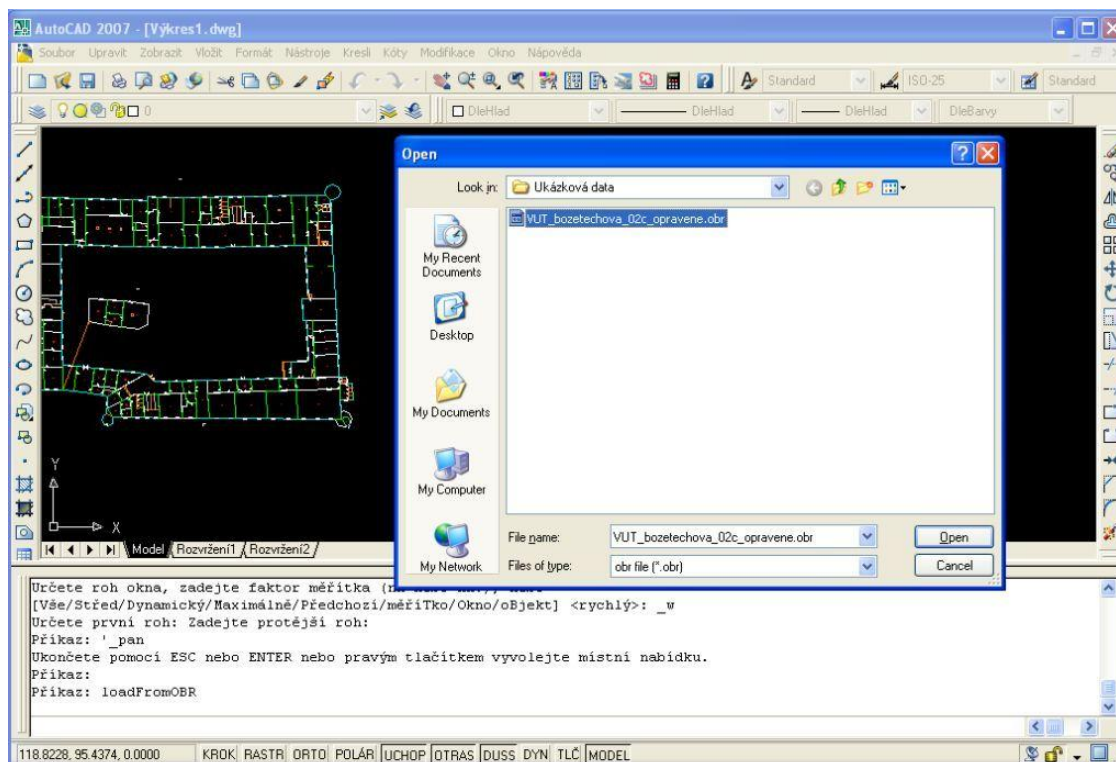
Program byl testován v programu AutoCAD 2007. Pokud chcete spustit jakýkoliv příkaz, je nutné nejprve načíst knihovnu „passportPlugin4AutoCAD.dll“ ze souboru „CD\DLL pro AutoCAD“. Knihovnu načtete příkazem NETLOAD. Příkazy se zadávají v příkazové řádce (Obrázek příloha 1 - 1) a potvrzují klávesou „enter“. Pro urychlení práce s programem AutoCAD stačí zadat pouze „net“ a zmáčknout klávesu „tabulátor“ a zbytek příkazu se doplní. Pokud zmáčknete klávesu „tabulátor“ podruhé, objeví se další příkaz začínající na písmena „net“.



Obrázek příloha 1 - 1 Načtení knihovny do programu AutoCAD.

Příloha 1.

Pokud jsme načetli úspěšně knihovnu do programu AutoCAD, můžeme začít zadávat příkazy pro ovládání programu. Pro načtení dat ze souboru formátu obr nebo pro jeho uložení do formátu XML je nutné opět zadat obdobný příkaz jako „NETLOAD“ (Obrázek příloha 1 - 2). Opět stačí zadat část příkazu a zmáčknout „tabulátor“.



Obrázek příloha 1 - 2 Načtení souboru obr v programu AutoCAD.

Seznam příkazů

- loadFromOBR – načte a zobrazí data ze souboru formátu obr
- loadFromXML – načte a zobrazí data ze souboru formátu XML
- loadFromOBRSaveTOXML – načte a zobrazí data ze souboru formátu obr a uloží data do souboru formátu XML

Všechny příkazy umožňují načítat větší množství souborů.