

1. Úvod

V r. 1994 firma MicroTool uvedla na softwarový trh upgrade CASE/4/0 ve verzi 4.1., který přinesl uživatelům CASE/4/0 další zlepšení. Jeho ovládání se stalo jednodušším a modelování navrhovaného IS preciznějším v oblastech:

- automatizovaného převádění konceptuálního (ER) modelu na model logický (relační),
- jednoduchého modelování řídicí funkce v diagramu informačních toků s vnořeným diagramem přechodů stavů,
- intuitivního chápání entity jako objektu nesoucího kromě svých atributů také své operace při dynamickém modelování,
- zobrazování toků: - datových,
 - materiálových,
 - událostních.

Nasazení jakéhokoliv CASE do projekční činnosti musí nevyhnutelně předcházet zvládnutí metodologie systémové analýzy a designu IS, kterou CASE podporuje. Nedokonalé zvládnutí, či dokonce absence metodologie vede jednak k podcenění nebo naopak přecenění CASE nástroje, což má za následek provádění nákladné a málo efektivní projekční činnosti.

Jaké metodologie naše projekční a programátorská veřejnost ve své činnosti vesměs využívá? Ve většině organizací se dodnes vychází ze zásad metodiky "Pokyny pro budování ASŘ", které podporují metodu strukturovaného programování založenou na funkční abstrakci. Metodologie tvorby informačního systému se potom především opírá o hierarchickou dekompozici IS a sémantický popis funkcí, dat a procesů. Z hlediska používaných ručních i automatizovaných podpor projekčních prací a dokumentačních standardů, které byly často neaktuální a málo využívány, byly vyvinuty takové metody jako HIPO, HIT a různé metainformační systémy jako byl SADEX, ale mnohde postačil i jednoduchý textový editor a pod..

Fakt, že se nedařilo projektovou dokumentaci udržovat v aktuálním stavu po celou dobu životního cyklu projektu, se skutečností, že do informatiky vstupovali lidé často profesně nedostatečně kvalifikovaní, přispěl k jakési metodologické nedůslednosti až nekázní v projekčních a programátorských postupech. To je dědictví minulosti, které v myslech mnohých "praktiků" dodnes přežívá a s kterým se musí nově vznikající softwarová pracoviště, jež projektují náročné a rozsáhlé uživatelské aplikace, vyrovnávat.

Cestou k tomu, jak spojit "místní projekční kulturu" [1] se světově standardizovanými metodologiemi, jsou investice do proškolení pracovníků těmito metodologiemi s návazností na nákup CASE nástrojů. Příklady takových nabídek můžeme nalézt u řady firem, jako jsou ITC PragoData, LBMS CZ, Komix a pod.. Za příklad cesty profesionalizace projekčních prací v podmínkách středoevropské kultury lze považovat technologii "PDIT" projektování informačních systémů v architektuře client/server vyvinutou firmou ITC PragoData, a.s.. PIDT vychází z obecné metodologie britské firmy LBMS Systems Engineering. Zařazení CASE/4/0 mezi nástroje, které tuto metodologii podporují, zvyšuje jeho užitnou hodnotu i uplatnění na softwarovém trhu, jež požaduje splnění požadavků normy ISO-9003.

2. Volba přístupu k tvorbě informačního systému.

Přístupy k tvorbě informačního systému mohou být ovlivněny:

1. Funkční typovostí projekčního záměru.
2. Datovou typovostí vyplývající z hromadného zpracování dat
3. Inovačním charakterem dosud nerealizované části informačního systému.

Filosofii tvorby projektů ovlivněných funkční a datovou typovostí projekčního záměru lze realizovat "ad hoc" přístupem, v němž hrají podstatnou roli zkušenost a osvědčené postupy projektantů. Naopak u inovačních projektů je třeba uplatňovat "vyvážený" přístup s významnou měrou intuice a novátorství projektantů.

Z hlediska samotného projektu IS můžeme výše vyjmenované vlivy na jejich tvorbu akceptovat v podobě analýzy funkční, datové a událostní při respektování pravidla, že čím je systém pro uživatele jednodušší, tím je pro tvůrce složitější. Zvolit analytický přístup k tvorbě informačního systému v podstatě znamená rozhodnout se, kterým diagramem ve strukturované metodologii začít.

Jestliže informační systém má kopírovat stávající organizační strukturu (a odpovídat některé známé standardní podobě organizace), je vhodné zvolit funkční přístup a začít diagramem hierarchické struktury. Celá architektura IS je tím pádem odvozena ze známé funkční dekompozice a představuje určitý typ standardního informačního systému (na př. informační systém podniku).

Při znalosti některé dominantní skupiny dat (jako při zpracování MTZ, odbytu, účetnictví, mezd a pod.), je možno zvolit datový přístup a začít diagramem entit a relací. Volba datového přístupu se převážně vztahuje na úlohy evidenčního charakteru s dávkovým zpracováním velkých objemů dat a nehodí se na projektování interaktivně pracujících systémů.

Informační systémy, jejichž struktura není dosud stabilizována či standardizována a systémy, jejichž architektura do značné míry závisí na informační potřebě uživatele, je vhodné začít projektovat od diagramů informačních toků. Systémy s vyjmenovanými vlastnostmi patří k těm, jejichž potřebu vyvolávají měnící se společensko ekonomické podmínky a nestačily se v rámci stávajících IS dosud stabilizovat. Příkladem za všechny může sloužit

"systém obchodních a marketingových informací", jehož potřebu vyvolává prosazující se tržní mechanismus a snaha podniků uplatnit se na zahraničních trzích.

Cesta událostně orientovaného analytického přístupu, kdy se provádí analýza datové (Flow) orientovaných událostí, je vhodná pro moderně pojeté řešení informačních systémů. Událostně orientovaný přístup k analýze vytváří podmínky pro takový způsob navrhování IS, který v projektu pasivuje funkční specifikaci a upřednostňuje události. Způsob zobrazování výsledků systémové analýzy pomocí dynamických modelů je založen na principu objektově orientovaného modelování (OOM).

3. Technika objektového modelování.

Základním principem objektově orientovaného modelování je abstrakce dat reprezentovaná objektem, jež je definován jak svými atributy, tak svým chováním. Objektově orientované modely se používají k napodobení dynamických vlastností systémů, slouží ke komunikaci expertů z dané problemové oblasti, dají se jimi vyjádřit interakce modelovaných prvků dynamického systému, slouží k navrhování programů, databází a přípravě dokumentace.

Objektově orientované myšlení vychází z principů strukturovaného myšlení a respektuje časové souvislosti změn objektů v dynamickém systému. Myšlení, které se historicky vyvíjelo od funkčního, přes strukturované až k objektově orientovanému představuje gradaci poznávacích a tvůrčích procesů v oblasti systémové analýzy, modelování a projektování IS. Tato gradace je stupňována vývojem výpočetní techniky, programovacích jazyků a softwarových podpor analytických softwarových aplikací může, ale nemusí být objektově orientovanou. To závisí jak na způsobu myšlení, tak na volbě implementačního prostředí. Takovéto pojetí vývoje se nazývá technikou objektového modelování (TOM).

TOM je metodologie, která kombinuje tři pohledy na modelovaný systém a vyznačuje se:

- datovou reprezentací systému, t.j. datovými strukturami,
- funkční reprezentací, t.j. transformací dat a
- reprezentací chování, t.j. sekvencí stavů a přechodových procesů.

Všechny tři druhy pohledů jsou při projektování IS realizovány systémem modelů objektových, funkčních a dynamických.

Struktura prvních dvou typů modelů je dostatečně známá, neboť v podstatě představují rozšíření odpovídajících technik ze strukturované analýzy. Dynamické modely popisují podmínky a průběh změn v informačním systému.

Graficky jsou dynamické modely v projektech IS reprezentovány diagramy datových toků s řídicími funkcemi a s vnořenými stavovými diagramy aplikace. Každý stavový diagram vyjadřuje sekvenci stavů a přechodů, jež jsou v aplikaci přípustné pro operace odpovídajících tříd objektů. Přechody ve stavových diagramech korespondují s funkcemi ve funkčním modelu tak, že funkce jsou na

základě nabytých stavů spouštěny prostřednictvím událostí. Po vykonání funkce se aplikace dostává do nových stavů, které ve stavových diagramech nastavují nové podmínky pro spouštění dalších funkcí.

4. Návrh aplikace "Sledování vybraných obchodních a marketingových informací" v architektuře klient/server.

Pro potřebu softwarového nabídkového řízení podniku se strojírenským charakterem výroby byl v minulosti vyprojektován a na úrovni prototypingu realizován systém obchodních a marketingových informací. Jelikož při formulaci tohoto úkolu nebyla přesně specifikována funkční struktura informačního systému a jeho návrh představoval dosud nerealizovanou inovaci, jevil se k jeho vyprojektování metodologicky nejvhodnějším nástrojem událostně orientovaný přístup.

Seznam datově orientovaných událostí vytypovaných cestou řízených rozhovorů, kontextový diagram aplikace i návrh relačního modelu přispěly ke koncepci prototypu informačního systému pokrývajícího nabídkové řízení obchodního úseku, sledování plnění zakázkové činnosti ve výrobě až po stadium expedice.

Návrh implementace předpokládá, že informační systém bude pro svou autonomii provozován v lokální počítačové síti. V prvním stadiu vývoje budou centrální databáze včetně databází všech klientů umístěny na DB serveru. Později se uvažuje o distribuovaném řešení aplikace.

5. Závěr

Účelem obrazových příloh je naznačit způsob použití dosud málo vyzkoušeného událostně orientovaného přístupu v projekční metodologii a poukázat na jeho možnosti při projektování informačních systémů. Událostně orientovaný přístup, jak bylo uvedeno, je spjat především s interaktivně pracujícími informačními systémy. Jinými slovy, je to přístup, který možná víc než jiné respektuje individuální požadavky uživatele (proto byl za aplikační realizaci zvolen systém obchodních a marketingových informací).

Zvláštností událostně orientovaného přístupu je, že napohled složité procesy probíhající v realitě, se promítají do databáze prostřednictvím vytypovaných událostí jednoduchou množinou aktualizací a dotazovacích transakcí (I,M,D,R). Událostně orientovaný přístup se jeví nejen jako možný, ale i jako vhodný způsob, jak přejít od filosofie strukturované analýzy k objektově orientované metodologii a přitom využívat dosud strukturovaně orientovaných CASE nástrojů a relačních databázových systémů.

Literatura

- [1] ITC PragoData: CASE/4/0, Příručka k české verzi. ITC PragoData, Praha, 1994.
- [2] Král, J., Demmer, J.: Softwarové inženýrství. Academia, Praha 1991.
- [3] Salemi, J.: Databáze klient/server. UNIS publishing, Brno 1993.

Doc.Ing. Jiří Schindler, CSc
EkF VŠB-TU Ostrava
Sokolská 33
701 00 Ostrava
tel.: 069/6225744/2311
e-mail: jiri.schindler@vsb.cz