

Jednoduchý systém pro podporu projektování

Bohumil Krejčířik

1. Ideová část

1.1. Problémy datové analýzy v rozsáhlejších projektech

Při návrhu a realizaci rozsáhlejších projektů se setkáváme s mnoha nepříjemnými faktory. Jedním z nich je i obtížné provádění datové analýzy, která je předpokladem pro vznik datové základny projektu. Rozsah takového velkého projektu předpokládá většinou paralelní práci několika projektantů s více projekty, nebo jeho částí. To samozřejmě stěžuje organizaci a řízení takových projektů. Předpokládaná datová základna těchto projektů je často vysoce distribuovaná a heterogenní. To je zcela samozřejmé při dnešní dostupnosti rozmanité výpočetní techniky a při požadavcích uživatelů na zkvalitnění přístupu k informacím, především v čase. To pak vyžaduje interaktivní prostředí, lokální datové základny a jejich spojení s centrální datovou základnou, využití sítí apod. Existuje samozřejmě řada softwareových prostředků pro podporu těchto prací, především v zahraničí. Ty však nejsou často dostupné včas, kdy je již nejvyšší čas začít projektovat, kdy se blíží první dodávka nového hardware apod. Často hraje roli i cena těchto produktů.

Výše uvedené problémy jsou typické pro podnik, který se ocitl v situaci, kdy je nezbytné stávající výpočetní systém nahradit nebo doplnit vyšším, spolehlivějším. Na něm se již tentokrát vybuduje skutečné ASŘ podniku, perfektně fungující informační systém. Tak se alespoň zdůvodňuje nákup takové techniky. Aby nově budovaný informační systém alespoň zčásti vyhovoval představě, aby se učinilo zadost slibům i uživatelům, je nutno projekt k takovému cíli zřít. Nezbytným prostředkem pro takovou práci, alespoň z hlediska datové analýzy je vytvoření datového modelu.

1.2. Datový model

Datový model popisuje skutečné entity a skutečné vazby mezi nimi tak, jak objektivně existují v realitě, která je předmětem projektu. V případě výše zmíněného ASŘ to bude podnik. Jde tedy o popis objektivní skutečnosti. Jiný pohled na tuto skutečnost pak mohou mít samozřejmě jednotliví uživatelé projektovaného informačního systému. Uvažujeme-li například entitu VYROBEK nebo MATERIAL, jiný pohled na ní bude mít a tím i jiné informace požadovat uživatel zabývající se účet-

nictvím, jiné zase uživatel z výroby, zásobování, prodeje atd. Některé údaje v dané entitě jsou samozřejmě společné. Datový model sjednocuje tyto pohledy jednotlivých uživatelů.

V literatuře se často rozlišují tři různé úrovně pohledu na data:

- **Externí** - to jsou individuální pohledy jednotlivých uživatelů,
- **Konceptuální** - to je sjednocující pohled, který odpovídá datovému modelu,
- **Interní** - to je skutečné, fyzické rozčlenění a uložení dat na paměťovém mediu v různých prostředcích výpočetní techniky, zapojených do daného informačního systému.

Měla by pak existovat korespondence mezi jednotlivými úrovněmi, realizovaná například mapovacím mechanismem, organizačně apod.

Datový model lze pak chápat jako metainformační systém, který modeluje objekty v realitě a vazby mezi nimi. Jedná se samozřejmě o objekty, které jsou předmětem projektu. Jako takový by měl mít datový model tyto základní vlastnosti:

- Nezávislost na skutečném, fyzickém uložení dat a na mechanismech tato data obsluhujících.
- Data by v něm měla být prezentována v normalizovaném tvaru (tzv. třetí normální forma) čili i bez redundancí.
- Musí být dostatečně otevřený, předpokládá se možnost změn a doplňků, stěží bude někdy definitivně uzavřen.
- Mechanismus, který ho obsluhuje musí být jednoduchý a dostatečně pružný s maximálními vnitřními kontrolami.
- Výstupy, které poskytuje, musí mít jednotnou formu, být srozumitelné a dostupné všem projektantům.

1.3. Jak využívat datový model

Projektant, který realizuje některou část projektu používá datový model jako podklad pro svoji datovou analýzu. Určuje, která data bude pro svoji část projektu využívat a jakým způsobem je bude využívat (čtení, modifikace atd.). Datový model mu poskytuje informace o vazbách / závislostech mezi daty, se kterými pak bude muset ve svém projektu počítat. Datový model mu dále poskytuje informace o tom, kdo, který projektant event. uživatel projektu daná data zakládá, obhospodařuje atd.

Projektant pak do datového modelu zavádí nové informace o datech a vazbách mezi nimi, jak to vyplývá z potřeb jeho části projektu. Tak se datový model postupně kompletuje a zároveň dokumentuje současnou znalost datových objektů projektu, jejich vztahy a vlastnosti.

2. Návrh datového modelu jako podpory pro projektování

Navrhovaný datový model vychází z modelu Entity-Relationship, což je snad vhodné vzhledem k trvající konjunktuře relačních databází (ty však nejsou podmínkou pro realizaci projektu). Pracuje s následujícími základními prvky:

- ENTITA** - Objekt (reálný, jako např. VAGON, DODAVATEL, nebo abstraktní, např. HISTORIE_UDALOSTI) o němž chceme vést záznam v datové základně. Není podstatné, zda se jedná o centrální datovou základnu (např. SQL/DS) či o lokální (např. DBase apod.), pokud je projekt realizován na více různých strojích a s různým software.
- POLOŽKA** - Údaj popisující vlastnost/stav/atribut entity.
- RELACE** - Vztah mezi dvěma entitami.
- PRIMÁRNÍ KLÍČ** - Položky, které entitu jednoznačně identifikují.
- CIZÍ KLÍČ** - Položky, které zprostředkují vazbu z entity, v níž jsou obsažené (podřízená entita) na jinou entitu (nadřazenou). Svým pořadím a atributy musí korespondovat s primárním klíčem nadřazené entity.
- PROJEKTY** - Dekomponovaná část celkového projektu. Projektanti vyvíjejí tyto části projektu paralelně, právě tak budou tyto části realizované ve funkcích sdílet společnou datovou základnu projektu.
- PRACOVISTĚ** - Uživatel projektovaného informačního systému. Jeden prostředek výpočetní techniky (terminál, PC), využívaný dvěma různými uživateli může představovat dvě pracoviště.
- PŘÍSTUPY** - Matice entity x pracoviště. Určuje, jak se z daného pracoviště bude přistupovat k dané entitě.

Koncepce datového modelu předpokládá, že za něj někdo odpovídá a udržuje jej. Může to být správce datového modelu, ale i hlavní projektant, koordinátor projektu či obdobná funkce. Ten musí určovat, zda ve sporných případech míní různí projektanti tutéž entitu, nebo jde o entity různé, koordinovat celý průběh výstavby datového modelu.

V první úrovni jsou (nejlépe formou diskuzí) definovány entity a jejich jednoznačné identifikace. Užitečný je grafický záznam životního cyklu entity (např. formou vývojového diagramu), který nejlépe upřesní stavy, jichž může entita nabývat a který může sloužit jako podklad při řešení sporných případů. Dále je třeba zachytit vztahy mezi entitami, opět nejlépe graficky E-R diagram. Lze již určit, která

pracoviště v rámci kterého projektu a jakým způsobem budou s entitou pracovat. Je také možno postupně definovat položky entit, ale ty se upřesní až v další úrovni, při definici funkcí nad daty, kdy se může ukázat potřeba dalších pomocných položek apod. V této druhé úrovni je možno vyhledem k definovaným položkám specifikovat primární a cizí klíče. Celý postup probíhá v praxi v iteracích, stav projektů je na různé úrovni. Ideální by samozřejmě bylo pracovat postupně, definitivně určit entity, položky, klíče apod., datový model uzavřít a pak teprve přistoupit k definici funkcí nad datovým modelem, programovým zadáním atd.

3. Technické řešení

Většinu práce s datovým modelem lze samozřejmě automatizovat. To zaručí snadnost zavádění nových informací, oprav a rychlé využívání informací v datovém modelu obsažených. To umožňuje vestavění potřebných automatických kontrol, které by se jinak ručně těžko prováděly. Následuje popis souborů, které by měly postačovat pro udržení základních informací v datovém modelu. U každého souboru jsou uvedeny jeho položky s podrobným popisem významu a povolených hodnot.

ENTITY	- definice entity
Název entity	- jednoznačná identifikace v rámci souboru
Popis entity	- stručná charakteristika, titulek
Definice entity	- přesná, jednoznačná definice
třída entity	- zařazení entity do jedné ze tříd:
	Z = základní entita, popisuje většinou reálný objekt, může existovat samostatně.
	P = popisná entita, je vztažena k jiné entitě, popisuje její vlastnosti.
	C = Číselník, katalog, seznam s možností přiřazení dalších údajů.
	A = asociativní entita. Jednoznačně přiřazuje entity, které jsou ve vztahu m : n.
	W = pracovní entita, většinou podporuje nějaké technické řešení.
POLOŽKY	- definice položky
název položky	- jednoznačná identifikace v rámci souboru
popis položky	- stručná charakteristika, titulek
definice položky	- přesná, jednoznačná definice včetně kontrol
typ položky	- N = numerická
	- C = znaková

	-	D = datum
	-	T = čas
	-	M = dlouhá položka (MEMO, GRAPHIC apod.)
	-	L = logická
délka položky	-	včetně desetinných míst
počet des. míst	-	pouze pro typ položky N
symbol projektu	-	kde položka vzniká
ENT POL	-	přřazení položky k entitě
název entity	-	ze souboru ENTITY
název položky	-	ze souboru POLOZKY
RELACE	-	definice relace mezi dvěma entitami
název entity	-	podřizené
název entity	-	nadřizené
název relace	-	jednoznačný v rámci souboru
spodní hranice	-	počtu výskytů podřizené entity
horní hranice	-	počtu výskytů podřizené entity
spodní hranice	-	počtu výskytů nadřizené entity
horní hranice	-	počtu výskytů nadřizené entity
rušící pravidlo	-	
		R = restrict, nadřizenou entitu nelze zrušit, pokud existuje výskyt podřizené entity
		C = cascade, zruší se všechny podřizené entity včetně nadřizené
		S = set null, nadřizená entita se zruší, do cizího klíče podřizené se nastaví hodnota NULL ("prázdná hodnota").
popis relace	-	popis významu relace
PKLICE	-	definice primárního klíče
název entity	-	pro kterou primární klíč definujeme
název položky	-	součástí primárního klíče této entity
pořadí položky	-	v primárním klíči
CKLICE	-	definice cizího klíče
název klíče	-	jednoznačný v rámci souboru
název položky	-	která je jeho součástí
pořadí položky	-	v cizím klíči

PROJEKTY - **definice projektů**
symbol projektu - jednoznačný v rámci souboru
název projektu

PRACOVIŠTĚ - **definice pracovišť**
symbol pracoviště - jednoznačný v rámci souboru
název pracoviště

PŘÍSTUPY - **definice přístupů**
symbol projektu - v jehož rámci se k entitě přistupuje
symbol pracoviště - z něhož se s entitou manipuluje
název entity

příznak lokálního zpracování -

L = (lokální PC apod.)

C = centrální zpracování způsoby povolených přístupů k entitě

R = pouze číst

C = zakládat novou

D = rušit

M = modifikovat údaje které mění stav entity

U = opravovat ostatní údaje

Rozměry uvedených položek souborů závisí na rozsahu podporovaného projektu, ale i na dokumentačních zvyklostech apod. Právě tak implicitní hodnoty. Funkce, které budou nad soubory pracovat musí zajistit jejich integritu, to znamená, že například nelze v definici primárního klíče uvést položku, která by již nebyla definována v souboru POLOŽKY, nelze zrušit entitu, která je součástí některé relace, ale to jsou již problémy konkrétního programového zadání pro podobný podpůrný software. Úroveň komunikace s uživatelem takového podpůrného software záleží pak už snad pouze na časových možnostech navrhovatele.

Závěr

Při návrhu takového software by se však neměl pouštět ze zřetele hlavní důvod jeho vytvoření, to je jednoduchost, přehlednost a rychlost realizace. I výše uvedený návrh umožňuje udržovat informace o sledovaném projektu na právě dané úrovni poznání a při vhodně navržených funkcích umožní například:

- Kompletní informace o dekompozici projektu (části, projekty) a pracovištích pro manipulaci s daty.
- Popis entit, jejich struktury (položky), klíče a vazby mezi nimi, z nichž plynou nutné kontroly v aplikačních programech.
- Informace o tom, z kterého pracoviště, v rámci kterého projektu se k dané entitě přistupuje a jakým způsobem.
- Tvorba různých přehledů, výběrů, inverzních zobrazení, matic.
- Možnosti grafických výstupů apod.

Tato podpora pro projektování slouží nejen při vývoji projektu, ale i jako dokumentace distribuované báze dat.

Autor: Ing. Bohumil Krejčířík
Spolana
277 11 Neratovice
tel. 2151 kl. 3169