

NĚKOLIK POZNÁMEK K FORMÁLNÍM TECHNIKÁM NÁVRHU OBJEKTOVÝCH DATABÁZÍ

COMMENTS TOWARDS THE FORMAL TECHNIQUES OF THE OBJECT DATABASE DESIGN

Vojtěch Merunka

Anotace:

Příspěvek prezentuje současný stav poznání v oblasti formálních technik návrhu objektových databází a naznačuje možné směry vývoje.

Klíčová slova:

datová normalizace, objektový datový model, relační datový model, funkční závislosti atributů.

Abstract:

The paper will present the current state in the area of formal techniques for object database development, and will introduce possible trends into the near future.

Keywords:

data normalisation, object data model, relational data model, functional dependencies among attributes.

ÚVOD

Databázové systémy jsou založené na různých datových modelech. Jde o datový model síťový (a jeho variantu hierarchický datový model), relační, objektově relační a objektově orientovaný. (Někteří autoři také považují za databázové datové modely ještě modely fulltextové, hypertextové a modely založené na sémantických sítích).

Dnes dominuje relační datový model, který ve většině aplikačních oblastí postupně nahradil databáze založené na síťovém datovém modelu. Dnešní praxe však ukazuje, že relační databáze začínají být postupně nahrazovány databázemi objektovými. Pod obecným označením „objektové databáze“ se však skrývají dva vzájemně odlišné datové modely:

1. Objektově relační datový model představuje evoluční trend vývoje. Jde o doplnění relačního datového modelu o možnost práce s některými datovými strukturami, které známe z oblasti objektově orientovaných programovacích jazyků. Většina výrobců velkých relačních databázových systémů (např. Oracle) zvolila tuto variantu. Objektově relační datový model ale ve svých principech zůstává původním relačním datovým modelem.
2. Objektově orientovaný datový model (ODM) představuje revoluční trend vývoje. Jde o nový datový model, který není postaven jako rozšíření relačního datového modelu. Do jisté míry zde jde o renesanci původního síťového datového modelu, který je doplněn o možnost práce s objekty tak, jak je známe z objektového programování.

Hlavním motivem pro vznik ODM byly problémy s ukládáním a zpracováním objektů v relačních databázích. Relační datový model (RDM) objektovému programování nevyhovuje, protože je příliš jednoduchý. Z tohoto důvodu vznikl tlak na konstrukci nových databázových systémů, které by lépe dokázaly pracovat s objekty.

Objektový a relační datový model se od sebe výrazně liší. Tabulky jsou v ODM pouze jedna z možných forem výstupní prezentace uložených dat. ODM se nicméně může podobat strukturám síťových databází, jak jsme je znali v systémech IDMS. Na ODM můžeme nahlížet jako na renesanci síťového datového modelu. Při určité míře zjednodušení lze připustit vztah:

síťový datový model + objektové typy dat + polymorfismus = objektový datový model.

OBJEKTOVÉ DATABÁZE V SOUČASNÉ PRAXI

Relačně objektová technologie je dnes rozšířenější. „Nerelační“ objektově orientovaný datový model má však následující přednosti:

1. Lépe podporuje datové struktury, které známe z objektově orientovaných programovacích jazyků. Není třeba datové struktury tolik transformovat, aby byly uložitelné do databáze. Existují již prakticky použitelné systémy (např. Gemstone, ObjectStore, O2, Versant, ...), které dovolují v databázi zpracovávat objekty ve stejném tvaru, jak se s nimi nakládá v objektových programovacích jazycích.
2. Protože navazuje na síťový datový model, tak má předpoklady pro efektivnější způsoby zpracování dotazů ve srovnání s relačním datovým modelem. Tato vlastnost se projevuje hlavně u složitých datových struktur, které by se podle relačního datového modelu musely rozkládat do mnoha vzájemně provázaných relačních tabulek.

Příčiny, proč jsou zatím objektové databáze méně v praxi rozšířené, než relační, jsou pravděpodobně následující:

1. Malá praktická znalost objektových databází v komunitě tvůrců softwaru.
2. Dostupnost systémů na trhu a jejich cena. Velké relačně objektové systémy jsou levnější než objektové. (například komerční cena systému Gemstone je asi dvojnásobná než cena Oracle).
3. Konservativní myšlení potenciálních uživatelů a samozřejmě také potřeba zpracovávat již vytvořené báze dat v relačních systémech.
4. Chybějící standardy a nedostatečná podpora metod analýzy a návrhu. Návrh standardu ODMG 3.0 není všemi výrobci respektován. Modelovací jazyk UML nepodporuje všechny konstrukce potřebné k modelování objektové databáze. Metody používané pro návrh relačních databází nejsou vhodné k plnému využití možností objektových databází.

Přes uvedené problémy však existují důvody se domnívat, že význam objektových databází v blízké budoucnosti poroste, protože již dnes existuje celá řada aplikací, kde objektové databáze prakticky prokazují svoje přednosti. Společnou vlastností těchto aplikací je velké množství komplexních datových struktur a jejich proměnlivost za chodu systému, které způsobují problémy relačním databázím. Takové systémy mohou pracovat až se stovkami a tisíci různých vzájemně poskládaných datových typů reprezentovaných třídami objektů. Dotazy nad takovými objekty ještě navíc vyžadují vysokou míru vzájemného polymorfismu. (V takových systémech kupříkladu potřebujeme klást dotazy nad množinami obsahující prvky různého typu. Zároveň očekáváme, že při přidání nového datového typu se nebudou muset přepisovat již hotové dotazy.) Typickým příkladem takových systémů jsou datové sklady,

kteřé jsou charakteristické dlouhodobým shromažďováním velkého množství nově vznikajících různorodých dat. Takové systémy jsou charakteristické nejen pro řízení velkých podniků, ale také v různých evidenčních systémech státní správy, zdravotnických systémech, informačních systémech obsahujících ekologické informace, zemědělských informačních systémech, historiografických informačních systémech atp.

Na druhou stranu je třeba poznamenat, že relační databáze fungují velmi dobře v oblastech, kde během života systému nedochází k požadavku na změnu struktury databáze a na přidávání dalších datových typů. Relační systém může být výkonný i pokud se databáze skládá z velkého množství záznamů, ale uložených v malém počtu jednoduše strukturovaných relačních tabulek.

STAV VÝZKUMU OBJEKTOVÝCH DATABÁZÍ V ČR

Bohužel se v ČR dnes objektovými databázemi žádné pracoviště soustavně nezabývá. Dílčí práce byly vykonány v první polovině 90. let na Fakultě elektrotechniky a informatiky VUT Brno a na Matematicko fyzikální fakultě Komenského univerzity v Bratislavě (za bývalého Československa). Práce obou týmů vedly ke konstrukci experimentálních databázových systémů. Slovenský systém byl později dopracován ve finském projektu tvorby metamodelovacího nástroje na universitě v Jyväskylä – nyní jedním z celosvětově používaným CASE nástrojem Metaedit™. Dnes je situace taková, že na univerzitách v ČR se až na výjimky objektové databáze neobjevují ani ve výuce.

STAV VÝZKUMU OBJEKTOVÝCH DATABÁZÍ VE SVĚTĚ

Ve světě je několik univerzitních pracovišť, které se objektovými databázemi zabývají (např. CERN, Université de Genève, Vrije Universiteit Brusel a celá řada v USA jako např. MIT a Stanford University). Výsledky jejich práce jsou využívány v praxi při konstrukci objektových databází. Na internetu existuje mezinárodní sdružení ODMG – Object Database Management Group (www.odmg.org).

Problematika objektových databází je od poloviny 90. let diskutována na odborných konferencích. Od konce 90. let vycházejí v zahraničí odborné publikace, které se zabývají především vlastnostmi vybraných objektových databází. Přestože již existuje mnoho dílčích teoretických prací, které jednotlivě dokazují účelnost objektově orientovaného datového modelu v databázových systémech, tak se zatím v oblasti metod analýzy a návrhu objektových databázových aplikací používají jen postupy původně určené pro práci s relačními systémy a nebo jen intuitivní přístupy založené na zkušenosti s imperativními objektově orientovanými programovacími jazyky. Uvažuje se o využití refaktoringu, návrhových vzorů a agilních metodik. Domníváme se tedy, že je vhodné se věnovat výzkumu v oblasti specializovaných metod analýzy a návrhu objektových bází dat.

SOUČASNÉ METODY NÁVRHU OBJEKTOVÝCH DATABÁZÍ

Objektový datový model není nadstavbou relačního datového modelu. Je tedy otázkou, jaké metody návrhu použít. Pro relační datový model je k dispozici:

1. datová normalizace,
2. metoda syntézy atributů a
3. metoda datové dekompozice podle funkčních závislostí atributů.

Bohužel pro objektový datový model zatím není žádná všeobecně uznávaná a používaná technika nebo metoda návrhu. Je možné převzít relační techniky, ale potom dostaneme jen „relační databázi v objektovém prostředí“ a nevyužijeme všechny vlastnosti, které ODM má. Jinou možností je převzít schéma objektů a tříd tak, jak jsou navrženy v aplikaci, která má

s databází pracovat. To už je lepší, protože právě proto byl objektový datový model vyvinut. Jenomže struktura objektů výhodná pro algoritmy v softwarové aplikaci může komplikovat jejich efektivní databázové zpracování.

Čtenáři se mohou v různých pramenech setkat s různými tvrzeními o objektových databázích, které tuto problematiku zjednodušují a prohlašují například, že objektovou databázi není třeba normalizovat a že objektová databáze je mnohonásobně rychlejší než relační. Tvrzení o zbytečnosti datové normalizace můžeme nalézt i v renomované literatuře.

Tvrzení o rychlosti platí, ale jen pro případ, kdy se podaří navrhnout objektové schéma tak, aby obsahovalo přímo propojené objekty mezi sebou na rozdíl od relační databáze, která musí používat spojení od cizího klíče z jedné tabulky na primární klíč druhé tabulky. Tedy jinými slovy pokud se podaří objektové schéma navrhnout v duchu zásad síťového datového modelu.

S normalizací to je ještě složitější. Uvažuje se o využití refaktoringu a návrhových vzorů. Určitý pokrok učinil Kent Beck, jeden z pionýrů agilních metodik. Prezентuje první tři „objektové normální formy“, které se týkají správného návrhu struktury tříd objektů a jsou odvozeny z relačních normálních forem. V relačním datovém modelu jsou jednotkou funkční závislosti samotné atributy – tedy datové složky v záznamech a ne celé záznamy. Proto se bez normalizace v RDM neobejdeme. Objektový datový model pracuje s objekty, které navenek vystupují jako nedělitelné jednotky. Proto není problém normalizace tak naléhavý jako v RDM, ale neznamená to, že neexistuje.

Při tvorbě objektové databáze jsme tedy zatím odkázáni jen na zkušenost. Podle většiny prakticky orientovaných pramenů však lze popsat postup, jak objektovou databázi prakticky vytvořit:

1. Sestavit konceptuální model úlohy. Zde je možné použít diagram tříd UML nebo Chenův konceptuální ER diagram s rozšířením na vazby generalizace-specializace a kontejnery. Zatím ale modelujeme jen množiny. Atributy zde rozpoznané budou později implementovány nejen jako datové složky ale také jako metody objektů. V tomto kroku se to ještě nerozlišuje.
2. K prvkům množin najít potřebné třídy. Bohužel diagram tříd UML nemá zvláštní symboly pro množiny a pro třídy. Pokud máme množiny z objektů jen jedné třídy, tak to nevádí, ale jinak musíme použít stereotypy nebo přidat nový symbol.
3. Rozhodnout, které atributy budou implementovány datovými složkami a které metodami. Tyto metody pak naprogramovat.
4. Naplnit databázi daty. Tedy vytvářet objekty (instance) tříd a ukládat je do vybraných množin. Vytvořit objekt jako instanci třídy nestačí. Takový objekt totiž ještě není prvkem žádné množiny v databázi.

OTÁZKA FORMÁLNÍCH TECHNIK NÁVRHU

I v objektovém datovém modelování jsou užitečné formální techniky správného návrhu datové struktury. Zatím chybějí, ale potřebujeme je ze stejných důvodů, jako to činíme u relačních databází – tedy pro předcházení možných problémů s redundancí a konzistencí dat.

Relační techniky návrhu ale není možné jednoduše převzít z následujících důvodů:

1. Objekty mají nejen datové složky, ale i metody. Je tedy otázkou, co budeme považovat za "atribut". Zajímavý pohled přináší opět Kent Beck, který zavádí pojem

"behaviour" objektu jako spojení jeho datových složek a metod (Mírou koheze mezi datovými složkami a metodami se dokonce zabývá jedna z jeho "objektových normálních forem"). Pro potřeby databázového modelování ale většina autorů pracuje jen s pojmem "atribut", což jsou zvnějšku viditelné datové vlastnosti objektu. U takových atributů se potom nerozlišuje, zda mají implementační původ v datové složce nebo v metodě.

2. Relační postupy datové normalizace jsou odvozeny od konceptu funkční závislosti mezi atributy. Atributy v relační databázi jsou "základní stavební jednotkou" a jsou to skalární a atomické hodnoty, z nichž se skládají jednotlivé relace. V objektovém datovém modelu ale pracujeme s celými objekty, které z "relačního" pohledu sdružují několik atributů dohromady. Navíc, atributy těchto objektů mohou být také dalšími objekty. To znamená, že "základní stavební jednotka" objektového datového modelu na rozdíl od relačního není ani atomická ani skalární.
3. Dalším problémem jednoduchého převzetí relačních technik je koncept identity objektů. V relačním datovém modelu je identita tvořena hodnotou vybraných atributů, které mají úlohu primárních klíčů relací. V objektovém datovém modelu je ale identita objektu zachovávána i v případech, kdy jsou mu změněny hodnoty všech jeho atributů.

ZÁVĚR

Je velká škoda, že tak perspektivní a prakticky již používaná technologie, jako jsou objektové databáze, nemá dosud srozumitelný a všeobecně uznávaný teoretický základ a na něm vybudované formální techniky návrhu. Je nepochybné, že se tímto tématem zabývá několik pracovišť ve světě, ale zatím nebyly publikovány žádné ucelené výsledky. Proto se domníváme, že blízká budoucnost přinese možná i několik alternativních přístupů.

Na ČZU objektovou technologii studujeme a vyučujeme již od roku 1991. Objektové databáze jsou praktickou součástí výuky i prací doktorandů Katedry informačního inženýrství. V letošním roce se naše katedra stala prvním a zatím jediným univerzitním pracovištěm v ČR, které dostalo akademickou licenci systému Gemstone. Pro řešení zde diskutovaného tématu náš kolektiv v letošním roce požádal Grantovou agenturu České republiky o podporu. Projekt je evidován pod číslem GAČR 201/05/0906 a bude-li přijat, tak práce na něm začnou v roce 2005.

LITERATURA

1. Barry D.: *The Object Database Handbook: How to Select, Implement, and Use Object-Oriented Databases*, ISBN 0471147184
2. Beck K.: *Agile Database Techniques - Effective Strategies for the Agile Software Developer*, John Wiley & Sons; ISBN 0471202835
3. Blaha M., Premerlani M.: *Object-Oriented Modeling and Design for Database Applications*, Prentice Hall, ISBN 0-13-123829-9
4. Bertino, E., Martino, L., *Object-Oriented Database Systems, Concepts and Architectures*, Addison-Wesley, 1993. ISBN 0-201-62439-7.
5. Carda A., Merunka V., Polák J.: *Umění systémového návrhu - objektově orientovaná tvorba informačních systémů pomocí původní metody BORM*. Grada, Praha 2003. ISBN 80-247-0424-2.
6. Catell R. G.: *The Object Data Standard: ODMG 3.0*, ISBN 1558606475
7. Delobel C., Lecluse C., Richard P.: *Databases: from Relational to Object-Oriented Systems*, International Thomson Computer Press, 1995. ISBN 1850321248
8. Embley D.: *Object Database Development: Concepts and Principles*, ISBN 0201258293
9. Gemstone Object Server - documentation & non-commercial version download, <http://www.gemstone.com>, <http://smalltalk.gemstone.com>.

10. Hruška T.: *Objektově orientované databázové technologie*, sborník konference Tvorba softwaru 1995, ISBN 80-901751-3-9
11. Kroha P.: *Objects and Databases*, McGraw Hill, London 1995, ISBN 0-07-707790-3.
12. Lacko B.: *Komparativní analýza strukturovaného a objektově orientovaného přístupu*, sborník konference OBJEKTY 1996. ČZU Praha 1996 str.69-76
13. Lacko B.: *Vademekum objektově orientované technologie*, sborník konference Programování, Ostrava 1993.
14. Loomis M., Chaundri A.: *Object Databases in Practice*, ISBN 013899725X
15. Merunka V.: *Objektový databázový systém Gemstone*, sborník konference OBJEKTY 2003. Ostrava 2003. ISBN 80-248-0274-0
16. Merunka V.: *Objekty v databázových systémech*, skriptum ČZU Praha, Praha 2002. ISBN 80-213-0318-2

Kontaktní adresa autora:

Ing. Vojtěch Merunka, Ph.D., Katedra informačního inženýrství PEF ČZU Praha,
telefon +420 22438 2272, e-mail: merunka@pef.czu.cz