

RELACIJSKE IN OBJEKTNE PODATKOVNE BAZE V GIS/LIS SISTEMIH

dr. Radoš Šumrada
FAGG-Oddelek za geodezijo, Ljubljana
Prispelo za objavo: 21.1.1994

Izveleček

Objektna orientacija je tržno zelo popularna oznaka, ki se pogosto uporablja za zelo različne pomenne in za vse mogoče potrebe. V članku so osnovne značilnosti objektnih in zlasti objektno orientiranih podatkovnih baz. Opisane so njihove vsebinske in tehnološke značilnosti v primerjavi s splošno razširjenimi relacijskimi podatkovnimi bazami. Predstavljen je tudi tehnološki in procesni pomen objektno orientiranih podatkovnih baz za uporabo v GIS/LIS sistemih.

Ključne besede: DBMS, GIS/LIS, objektno podatkovne baze, objektno orientirani podatkovni model, relacijske podatkovne baze, RDBMS, uporabniki, značilnosti

Abstract

At present object orientation is quite a buzz word often applied to various meanings and all kinds of needs. This article outlines basic characteristics of object and especially object-oriented databases, their conceptual and technological framework as compared to widely spread relational databases. The article presents also the technological and processing importance of object-oriented databases for application in GIS/LIS systems.

Keywords: characteristics, DBMS, GIS/LIS, object databases, object-oriented data model, RDBMS, relational databases, users

1. UVOD

Odkar so se okoli leta 1980 pojavili tržni softverski paketi za podporo relacijskemu podatkovnemu modelu, se pojmuje relacijski sistem za vodenje podatkovne baze (DBMS) kot vrhunski model tehnologije podatkovnih baz. Relacijski podatkovni model temelji na skrbno izdelani in dovršeni relacijski teoriji, ki je zasnovana na izpeljavi iz matematične teorije množic. Teorija relacijskega podatkovnega modela ima že več kot dvajsetletno zgodovino in več kot petnajstletno operativno tradicijo. V devetdesetih letih sta bili doseženi kulminacija in tržna prevlada relacijskih podatkovnih baz. Temelji na vzporednem skokovitem razvoju ustreznih hardverskih ter softverskih računalniških sposobnosti ob hkratnem stalnem padanju cen računalniške opreme in operativnih stroškov. Relacijske podatkovne

baze so v zadnjih nekaj letih dosegle vrhunec vsestranske operativne uporabe v zelo raznolikih uporabniških okoljih.

Nekateri uporabniki so medtem prišli do spoznanja, da je relacijski podatkovni model nezadosten oziroma neprimeren za podporo nekaterim zvrstem aplikacij. To velja zlasti v primerih kompleksnih podatkovnih zgradb, kot so na primer računalniška grafika, CAD, CAM, CASE, znanstvene in medicinske aplikacije, ekspertni sistemi ter zlasti sistemi GIS/LIS. Vsi ti uporabniki ustvarjajo in uporabljajo sestavljene objekte ter potrebujejo karakteristike DBMS-sistemov, ki se zelo razlikujejo od operativnosti tradicionalnih podatkovnih baz. Pojavile so se zahteve po drugačnih sistemih DBMS-ja, ki bodo ohranili vse glavne lastnosti tradicionalnih podatkovnih baz ter bodo hkrati sposobni trajno shranjevati zahtevne podatkovne strukture in povezave med njimi.

Zadnjih nekaj let se tako vlagajo veliki napor v raziskave in aplikativni razvoj sistemov objektnih podatkovnih baz (ODMS - Object Data Management Systems), ki lahko mnogo bolj učinkovito podpirajo zahtevne podatkovne modele in aplikacije. Celotno področje objektnih podatkovnih baz je nova računalniška domena. Sistemi objektnih podatkovnih baz so v intenzivnem razvoju in so predmet številnih raziskav. Obstaja velika skupina uporabnikov z aplikacijami, ki postavljajo nove, vendar podobne podatkovne zahteve. Od podatkovne baze pričakujejo podporo za upravljanje z veliko količino soodvisnih in vgnezenih podatkov s poljubno sestavljeno strukturo, katerim mora biti zagotovljena hkratna razvojna podpora za različne inačice podatkovnih modelov in alternativnih rešitev.

V nasprotju z relacijskim podatkovnim modelom še ne obstaja formalno enoten in uveljavljen objektno orientirani podatkovni model. V zadnjih letih je bilo narejenih veliko raziskav in poizkusov v tej smeri. V literaturi je veliko celovitih objektnih podatkovnih modelov ter pristopov. Vsaka hitro razvijajoča se disciplina računalniške znanosti ima svoje vzpone in padce. Sistemi objektnih podatkovnih baz so še področje raziskovanja in intenzivnega eksperimentiranja računalniške znanosti, ki postopoma prehaja v svoje tržno obdobje (Simon 1993). Na tem področju že obstaja formalna organizacija za strokovno povezovanje OMG (Object Management Group) in njena posebna poslovno orientirana podskupina ODMG (Object Database Management Group), v kateri je več vodilnih proizvajalcev objektnih podatkovnih baz (Atwood 1993). Skupina ODMG je sestavila standard ODMG-93 za osnovne potrebe objektnih podatkovnih baz. Uradni osnutek standarda ODMG-93, ki je bil predvidoma objavljen do konca leta 1993, sestoji iz specifikacije značilnosti in vsebine objektnega modela, ODMS-ja ter dostopnih povezav za objektno orientirane programske jezike.

2. OBJEKTNE PODATKOVNE BAZE

Objektna podatkovna baza je sistem za upravljanje podatkovne baze, ki opravlja s stanovitnimi objekti. Takšni sistemi združujejo objektni podatkovni model in sposobnosti DBMS-ja v funkcionalno in operativno celoto. Objektne podatkovne baze (Cattell 1991) se glede na splošne lastnosti in sposobnosti lahko razdelijo v naslednje vrste:

- objektno orientirane podatkovne baze

- razširjene relacijske podatkovne baze
- funkcionalno semantične podatkovne baze
- enostavne sisteme za upravljanje z objekti
- generatorje podatkovnih baz.

Objektno orientirane podatkovne baze in razširjene relacijske podatkovne baze sta tisti dve obliki sistemov objektnih podatkovnih baz, ki zaslužita večino pozornosti; sta najbolj razširjeni, tehnološko najnaprednejši in največkrat izvedeni obliki objektnih podatkovnih baz. Objektno orientirani sistemi za vodenje podatkovnih baz so edine izvirne objektno orientirane podatkovne baze, ki temeljijo na objektno orientiranem podatkovnem modelu ter predstavljajo razširitev oziroma nadgradnjo določenega objektno orientiranega programskega jezika. V številni heterogeni literaturi o objektnih podatkovnih bazah se pojavljajo različne terminološke oznake. Ne uporabljajo se tudi enotne kratice in interpretacije. Zasedimo lahko predvsem naslednje štiri akronime, katerih pomen je, kakor sledi iz naslednje tabele, vsebinsko različen.

<i>ODMS</i>	<i>Object Data Management System</i>
<i>ODBMS</i>	<i>Object DataBase Management System</i>
<i>OODBS</i>	<i>Object Oriented DataBase System</i>
<i>OODBMS</i>	<i>Object Oriented DataBase Management System</i>

Prvi dve kratici, ODMS in ODBMS, sta sinonima ter imata vsebinsko enak pomen. Uporabljata se za označevanje celotne družine objektnih podatkovnih baz, ne glede na dejanski generični tip posamezne podatkovne baze. V skladu s (Cattell 1991) je primernejša uporaba oblike ODMS (sistem za upravljanje z objektnimi podatki) za označevanje celotne skupine sicer različnih sistemov objektnih podatkovnih baz. Drugi dve kratici, OODBS in OODBMS, sta tudi enakovredna sinonima ter imata vsebinsko enak pomen. Uporabljata se za označevanje dejanske podskupine objektno orientiranih podatkovnih baz, ki je podrazred ODMS-ja. V skladu s (Cattell 1991) in (Atkinson et al. 1989) je primernejša uporaba oblike OODBS-ja (sistem objektno orientirane podatkovne baze) za označevanje generičnih, objektno orientiranih podatkovnih baz. Te temeljijo ali so podaljšek določenega objektno orientiranega programskega jezika oziroma predstavljajo njegovo nadgradnjo v smislu upravljanja z obstojnimi (persistentnimi) objekti.

3. UPORABNIKI OBJEKTNIH PODATKOVNIH BAZ

Objektno podatkovne baze predvsem niso nadomestilo ali zamenjava za relacijske podatkovne baze. Omogočajo samo bolj uspešno in bolj učinkovito uporabo tehnologije podatkovnih baz v nekaterih posebnih uporabniških okoljih. Med neposredne uporabnike objektnih podatkovnih baz spadajo zlasti aplikacije z naslednjih uporabniških področij: CAD, CAM, CASE, GIS/LIS, obdelava besedil, znanost in raziskave na splošno, medicina, organska kemija, inženirstvo znanja ter izvedenski sistemi in sistemi za delo v realnem času (Cattell 1991). Vsa ta aplikativna področja imajo veliko podobnih zahtev po upravljanju s podatki. To je morda presenetljivo, ker imajo vse te številne aplikacije zelo različne podatkovne zahteve,

vendar imajo vse enak razvojni princip. Vse te aplikacije uporabljajo ljudje za načrtovanje umetnih izdelkov ob podpori računalnikov. Predvsem prva tri aplikativna področja so tista, ki so dejansko motivirala razvoj objektnih podatkovnih baz in so dejansko največ pridobila s sodobnim razvojem sistemov za upravljanje objektnih podatkovnih baz.

4. RAZLIKE MED RELACIJSKIMI IN OBJEKTNO ORIENTIRANIMI PODATKOVNIMI BAZAMI

Glavna razlika med relacijskimi ter objektnimi podatkovnimi bazami je predvsem v načinu podpore uporabniškega dostopa do podatkov. Osnovna razlika je v pasivnem in aktivnem obnašanju celotnega podpornega sistema oziroma načinom, kako je podporni DBMS realiziran (Ozkarahan 1990). V tradicionalni pasivni relacijski podatkovni bazi so vsi podatki shranjeni samo kot zbirka strogo formatiranih zapisov v normaliziranih relacijskih tabelah. Ko je treba določene shranjene podatke procesirati in manipulirati, se za to lahko uporabi standardni neproceduralni jezik SQL ali pa se izdelata posebna uporabniška aplikacija v izbranem proceduralnem višjem programskem jeziku (3GL). Uporabnik lahko vidi samo logično predstavitev podatkovne baze kot normalizirane tabele, ki jih je treba ustrezno procesno spojiti, da se obnovi prvotni koncept podatkovnega modela. V večini relacijskih podatkovnih baz obstajata predvsem dva načina dostopa do podatkov. Prvi dostopni način je s pomočjo navedbenega jezika SQL. Uporabnik opredeli, kakšne podatke želi in mu ni treba vedeti, kako se bodo podatki poiskali. Interpreter SQL prevede stavke SQL v ustrezne postopkovne klice, ki so shranjeni v posebni knjižnici procesnih rutin. Takšen pristop štiti interno podatkovno neodvisnost oziroma notranjo organizacijo podatkov pred uporabniškimi posegi (Stein 1993). V sodobnih relacijskih podatkovnih bazah je SQL jezik pogosto tudi ustrezno proceduralno nadgrajen, kar omogoča uporabo SQL jezika v povezavi z določenimi postopkovnimi sposobnostmi (4GL).

Dругi dostopni način je v relacijskih podatkovnih bazah mogoč iz višjih proceduralnih programskih jezikov, v katere se vgradijo potrebni SQL stavki. Tako izdelani uporabniški programi se prevedejo ter absolutizirajo v hitrejšo in učinkovitejšo izvedbeno kodo. Izdelani uporabniški program, ki ob podpori relacijskega DBMS-ja izbere ustrezne podatke iz fizične podatkovne baze na disku, jih najprej začasno shrani v vmesnem hitrem pomnilniku. Sledi izvedba ustreznih manipulacij s podatki. Rezultat poizvedovanj se lahko pregleduje s pomočjo posebnih kazalcev na izbrane zapise. Vsakič, ko kurzor napreduje, se prepíše vsebina nove izbrane vrstice v posebej prirejene programske spremenljivke. Če so se podatki tudi spremenili, se rezultati transakcij ob podpori relacijskega DBMS-ja zapišejo nazaj na ustrezno mesto v pasivni podatkovni bazi na zunanjem pomnilniškem mediju. Neproceduralni stavki SQL se niso mogli nikoli zadovoljivo vklopiti v katerikoli proceduralni višji programski jezik. Programerji so dejansko programirali v dveh različnih jezikih hkrati (Atwood 1993). S pojavom bolj in bolj zahtevnih ter sestavljenih podatkovnih zgradb z zapletenimi povezavami je postalo zelo težko in nevarno obvladovati dostop do takšnih podatkov s pomočjo takšnega nerodnega kurzorskega vmesnika.

V objektno orientirani podatkovni bazi se uporabniška zahteva prek sistema za upravljanje baze direktno posreduje ustreznemu objektu v podatkovni bazi. Če objekt sporočilo razume, nanj odgovori na konsistenten način, tako da sproži ustrezno metodo razreda, ki izvede potrebno aktivnost, spremeni podatke ali pa posreduje sporočilo drugemu objektu. Objekti komunicirajo med seboj samo s pomočjo sporočil, ki lahko posredujejo tudi potrebne podatkovne vrednosti med objekti. Vse uporabniške zahteve se za razliko od relacijskih podatkovnih baz v celoti razrešijo znotraj objektov samih.

Celoten koncept objektov ter razredov podatkovnega modela je enolično upodobljen v ustreznem logičnem konceptu in interni arhitekturi objektno orientirane podatkovne baze. Vse povezave objektne podatkovne baze z objektno orientiranim programskim jezikom temeljijo na enakem objektnem modelu ter enakem definicijskem (ODL) in manipulacijskem (OML) jeziku (Atwood 1993). Vsi enaki pojavi obstojnih objektov v podatkovni bazi in enaki pojavi minljivih objektov v objektno orientiranem programskem jeziku pripadajo istemu objektnemu tipu. Vsi trajni in začasni objekti se uporabljajo identično v objektno definicijskem, manipulacijskem in poizvedovalnem jeziku. Ti so samo ustrezna nadgradnja istega objektno orientiranega programskega jezika. Takšen objektni pristop predstavlja dramatično poenostavitev v primerjavi z uporabniškimi dostopi do podatkov v relacijskih podatkovnih bazah.

Primarni cilj relacijske podatkovne baze je tako predvsem v čim večji podatkovni neodvisnosti. Podatkovna struktura mora biti popolnoma neodvisna od vseh procesov, ki uporabljajo podatke. Relacijska tehnologija podatkovnih baz je načrtovana predvsem za podporo različnim aplikativnim procesom, ki se lahko neprestano in poljubno spreminjajo. Podatki se lahko hitro ter učinkovito uporabijo tudi za različne nepredvidljive in naključne aplikacije. Prvenstveni cilj objektno orientirane tehnologije je enkapsulacija podatkov in procesnih funkcij v skoraj neodvisne razrede. Vse vrednosti atributov v vseh pojavih objektov so načeloma dostopne samo s pomočjo določenih metod razredov. Dodatni cilj objektno orientirane tehnologije je zagotovitev maksimalne ponovne uporabe že razvitih in preizkušenih razredov, oziroma že izdelane programske kode. Naslednja tabela, ki je povzeta (Martin et al. 1993) ter ustrezno razširjena, prikazuje pregledno primerjavo vseh glavnih lastnosti in razlik med relacijskimi ter objektno orientiranimi podatkovnimi bazami.

<i>Relacijske podatkovne baze</i>	<i>Objektno orientirane podatkovne baze</i>
<i>Prvenstvena cilja sta popolna podatkovna neodvisnost in njihova enostavnost.</i>	<i>Prvenstvena cilja sta enkapsulacija in razredna procesna neodvisnost.</i>
<i>Shranjeni so v glavnem samo podatki kot atributi v tabelah.</i>	<i>V podatkovni bazi so shranjeni atributi in vse procesne metode objektov.</i>
<i>Podatke si lahko delijo različni procesi in so načrtovani za poljubno aplikacijo.</i>	<i>Podatke objektov je možno uporabljati samo s pomočjo vgrajenih metod.</i>
<i>Podatki so pasivni. Določene operacije v bazi se lahko avtomatsko sprožijo.</i>	<i>Objekti so aktivni. Sprejeto sporočilo sproži ustrezno reakcijo oziroma metodo objekta.</i>
<i>Vsi procesi in aplikacije, ki uporabljajo podatke, se lahko neprestano poljubno ter neodvisno spreminjajo.</i>	<i>Razredi so načrtovani za maksimalno ponovno uporabo in so zato stabilni ter se predvidoma malo spreminjajo.</i>
<i>Podatki se lahko fizično reorganizirajo, ne da bi bile zato prizadete aplikacije.</i>	<i>Notranjost razredov se lahko spreminja, ne da bi se zato spremenil njihov vmesnik.</i>
<i>Uporabnik vidi podatke kot enostavne tabele s stolpci in vrsticami.</i>	<i>Podatkovne strukture so lahko zelo kompleksne, česar pa uporabniki ne opazijo.</i>
<i>Podatki so shranjeni deljeno v različnih normaliziranih tabelah. Tabele je možno enostavno spajati med seboj.</i>	<i>Podatki so lahko povezani z dinamičnimi kazalci in referenčnimi sezname, kar se manifestira v odličnih procesnih lastnostih.</i>
<i>Normalizacija podatkov pomaga pri zmanjšanju redundance v podatkih.</i>	<i>Neredundantnost podatkov in metod se doseže z enkapsulacijo in dedovanjem.</i>
<i>Neproceduralni standardni jezik SQL se uporablja za vse vrste manipulacij s podatki in ključna poizvedovanja.</i>	<i>Vse procesne metode v podatkovni bazi so programirane v izbranem objektno orientiranem programskem jeziku.</i>
<i>Naključna vprašanja so možna in se enostavno izražajo v neproceduralnem jeziku SQL. Neproceduralnost jezika SQL je njegova glavna pomanjkljivost.</i>	<i>Naključna vprašanja predstavljajo problem in se rešujejo s programiranjem metod v izbranem objektno orientiranem jeziku. Vnaprej se mora predvideti večina vprašanj.</i>
<i>Učinkovitost se zmanjšuje sorazmerno hitro s kompleksnostjo podatkovne zgradbe in številom spojenih tabel.</i>	<i>Objektno orientirane podatkovne baze imajo mnogo boljše procesne lastnosti zlasti za aplikacije s kompleksnimi podatki.</i>
<i>Podatkovni model relacijskih tabel se bistveno razlikuje od podatkovnega modela med analizo in načrtovanjem.</i>	<i>Podatkovni modeli za analizo, načrtovanje, izvedbo, programiranje in strukturo podatkovne baze so praktično enaki.</i>

5. ZAKLJUČEK

Pomembno je vprašanje, ali se lahko ter v kolikšni meri se lahko predvidi prihodnost in razvoj objektnih podatkovnih baz. Hkrati je zanimiva primerjava takšnih razvojnih predvidevanj z uporabo ter bodočim razvojem konkurenčnih relacijskih podatkovnih baz. Na razvojnem področju bodo v devetdesetih letih objektne podatkovne baze deležne večine raziskav, tako kot so to že bili v drugi polovici osemdesetih let (Simon 1993). Prihodnost nadaljnjega prodora in uveljavitve objektnih podatkovnih baz na tržišču pa je bolj negotova. Trenutno so sicer na tržišču številni paketi sistemov ODMS, vendar pa je njihov delež trga zelo majhen. Če pa

upoštevamo število uporabnikov CAD, CASE in GIS, lahko ugotovimo, da njihov odziv na novo tehnologijo objektnih podatkovnih baz do sedaj še ni bil zelo množičen. Manevrski prostor zožujejo tudi stalne izboljšave in razširitve relacijskih podatkovnih baz, ki so stalno prisotne in konkurenčne na istem tržišču.

Obstojita vsaj dve ali celo več verjetnih alternativ razvoja tehnologije podatkovnih baz v devetdesetih letih (Cattell 1991). Po prvi se predvideva, da se bodo tradicionalne lastnosti podatkovnih baz združile z objektnimi karakteristikami v enoten podatkovni model. Po drugi se pričakuje, da se bo tržišče podatkovnih baz razdelilo. Pri posebnih uporabnikih, kot so na primer CAD, CASE, GIS, itd., bodo prevladale in se uveljavile objektne podatkovne baze. Tradicionalne in zlasti relacijske podatkovne baze pa bodo obdržale tržišče poslovnih aplikacij. Prva možnost razvoja se zdi verjetnejša, vendar pa izkušnje razvoja podatkovnih baz povedo, da je tržni prodor podatkovnih modelov mnogo počasnejši od raziskav, znanstvenih in tehnoloških dosežkov (Simon 1993). Več kot desetletna razlika v primeru relacijskih podatkovnih baz je dober dokaz. Najverjetneje je torej, da bo tržišče relacijskih in objektnih podatkovnih baz ostalo razdeljeno še leta. Obstajajo seveda še tudi druge alternative razvoja. Objektne podatkovne baze lahko v celoti zatonejo v pozabo. Nekonkurenčna cena softverskih paketov ODMS, kompleksnost njihove uporabe, na splošno draga izvedba podatkovnih baz, pomanjkanje tradicije in uporabniškega zaupanja ter majhno tržišče so nedvomni argumenti te možnosti. Tradicionalne in zlasti relacijske podatkovne baze se lahko postopoma dovolj razširijo. Sistemi CAD, CASE in GIS se lahko opremijo z internimi objektno orientiranimi podatkovnimi bazami. Zgodovina podatkovnih baz beleži veliko primerov, ko se vsebinsko in tehnološko najbolj napredni podatkovni modeli niso tudi uspešno tržno uveljavili. Semantični podatkovni model je zadnji takšen primer.

Viri:

- Atkinson, M. et al., 1989, *The Object-Oriented Database Systems Manifesto* (in Kim, Nicolas, and Nishio).
- Atwood, T., 1993, *ODMG-93, The Object DBMS Standard*, *Object Magazine*, September-October.
- Cattell, G.C.R., 1991, *Object Data Management, Object-Oriented and Extended Relational Database Systems*, Addison-Wesley.
- English, P.L., 1992, *Object Databases at Work*, DBMS.
- Goldberg, J.C., 1993, *Object Oriented Databases*, *Oracle Magazine*, Volume VII, Number 1, Winter.
- Martin, J. et al., 1993, *Principles of Object-Oriented Analysis and Design*, Prentice Hall, Inc., PTR Professional Technical Reference.
- Mowbray, J.T. et al., 1993, *Interoperability and CORBA Based Open Systems*, *Object Magazine*, September-October.
- Ozkarahan, E., 1990, *Database Management*, Prentice Hall, Inc.
- Reilly, D., 1993, *Object Oriented Databases*, *Computer Language*, January.
- Simon, R., 1993, *Object or Relational? A Guide for Selecting Database Technology*, Servio Corporation Publication.
- Stein, J., 1993, *ODBMS*, *Object Magazine*, September-October.
- Strand, J.E., 1993b, *Broker Handles Map Objects by Request*, *GIS World*, December.
- Strickland, H., 1993, *ODMG-93, The Object Database Standard for C++*, *C++ Report*, October.
- Vinoski, S., 1993, *Distributed Object Computing with CORBA*, *C++ Report*, July-August.

Recenzija: Janko Rozman
Franc Zakrajšek (v delu)