



VISOKA POSLOVNA ŠKOLA STRUKOVNIH STUDIJA ČAČAK

SEMINARSKI RAD

Prostorne baze podataka

Mentor: _____
Profesor: _____

Student: _____
Br.Indeksa: _____

Prostorne baze podataka

S A D R Ž A J

1. UVOD.....	3
2. RELACIJSKI I OBJEKTNI PRISTUP RUKOVANJU PROSTORNIM OBJEKTIMA U BAZAMA PODATAKA.....	4
2.1. RELACIJSKI PRISTUP	4
2.2. OBJEKTNI PRISTUP	5
3. PROSTORNE BAZE PODATAKA - DOSADAŠNJA POSTIGNUĆA.....	7
3.1. KONCEPT PROSTORA	7
3.2. MODEL PROSTORNIH PODATAKA I UPITNI JEZIK	7
3.3. IZVOĐENJE PROSTORNIH UPITA	8
3.4. ORGANIZACIJA DATOTEKA PROSTORNIH PODATAKA I INDEKSI	10
4. MODELIRANJE 3D OBJEKATA U GEO-DBMS SUSTAVIMA.....	12
4.1. 3D PRIMITIVNI TIP PODATAKA	12
4.2. PROVJERA VALJANOSTI	14
4.3. PROSTORNI INDEKSI.....	14
5. 3D PROSTORNE OPERACIJE U GEO-DBMS SUSTAVIMA	15
5.1. KARAKTERISTIKE POLIEDRA	15
5.2. 3D TIPOVI PODATAKA U DBMS SUSTAVIMA	16
5.2.1. <i>Geometrijski tip podatka</i>	16
5.2.2. <i>Topološki tip podatka</i>	17
5.3. IMPLEMENTACIJA U DBMS SUSTAVIMA	18
5.4. TOPOLOŠKE OPERACIJE	18
5.4.1. <i>Prostorne operacije za geometrijski tip podataka</i>	19
5.4.2. <i>Prostorne operacije za topološki tip podataka</i>	20
6. DOHVAT 3D GEO-DBMS PODATAKA KORIŠTENJEM WEB TEHNOLOGIJA	21
6.1. 3D GEO-OBJEKTI U DBMS SUSTAVIMA	21
6.2. VRML I X3D	21
6.3. PRISTUPI	22
6.3.1. <i>ASP, VRML i MS Access</i>	22
6.3.2. <i>XSQL, X3D i Oracle</i>	24
7. ZAKLJUČAK	26

Literatura

Prostorne baze podataka

Sažetak: U ovom je seminarskom radu dan kratak pregled prostornih baza podataka. Opisane su neke općenite stvari poput različitih pristupa rukovanju prostornim podacima, dosadašnja postignuća na području organizacije datoteka prostornih podataka i indeksa, načini modeliranja 3D prostornih objekata, prostorne operacije nad 3D objektima, te mogućnosti njihove vizualizacije.

Ključne riječi: DBMS, indeks, 3D, poliedar, vizualizacija

1. Uvod

Zadatak ovog seminarskog rada bio je pronaći 5 znanstvenih članaka o prostornim bazama podataka koristeći baze znanstvenih časopisa na web-u, proučiti ih te sažeti u jednu cjelinu. Kao polazna osnova za pretraživanje korišteno je web sučelje Centra za online baze podataka, a zatim pretraživači baza znanstvene materije kao što su Scopus, ScienceDirect, SpringerLink te mnogi drugi. U tim je bazama moguće pronaći popis gotovo svih relevantnih znanstvenih članaka o zadanoj temi. Neke od njih nude samo sažetke, a neke i cjelokupne radove koje je moguće preuzeti najčešće u pdf formatu.

Ovaj je seminarski rad koncipiran na način da je naglasak stavljen na 3D prostorne podatke, njihovo modeliranje i održavanje u prostornim bazama podataka te njihovu vizualizaciju kao vrlo važne i neizostavne stavke svakog prostornog pretraživanja. Odabrani su 3D prostorni podaci upravo iz razloga što u okviru prostornih baza podataka postoje još mnoga neistražena područja koja su na razini 2D podataka već odavno riješena.

Poglavlja u ovome radu organizirana su na način da se svako poglavlje temelji na jednom znanstvenom članku. Prvo poglavlje daje kratku usporedbu relacijskog i objektnog pristupa kod baza podataka (Crowther i Hartnett 2001). Drugo poglavlje govori o modelima prostornih podataka, organizaciji datoteka prostornih podataka i indeksima (Shekhar i dr. 1999). U trećem je poglavlju riječ o modeliranju 3D prostornih objekata (Arens i dr. 2005). Četvrto poglavlje bavi se 3D prostornim operacijama (Chen i dr. 2008), a peto poglavlje vizualizacijom 3D objekata (Vries i dr. 2003).

2. Relacijski i objektni pristup rukovanju prostornim objektima u bazama podataka

Jedno od temelja za prostorne analize jest objekt. Objekti mogu biti stvarne stvari koje nas okružuju, ali i razne apstrakcije. Postoji čitav niz struktura u obliku baza podataka koje se mogu koristiti za pohranu prostornih podataka. One uključuju relacijske sustave za upravljanje bazama podataka (engl. Relational Database Management System, RDBMS), objektno orijentirane sustave za upravljanje bazama podataka (engl. Object Oriented Database Management System, OODBMS) i objektno-relacijske sustave za upravljanje bazama podataka (engl. Object-relational Database Management System, ORDBMS). Svaki od tih sustava zasnovan je na određenom pristupu rukovanju podacima.

2.1. Relacijski pristup

Osnova relacijskog pristupa jest prikaz podataka pomoću relacija (Slika 1). Uobičajeni izraz u svakodnevnoj upotrebi za relaciju jest tablica. Izraz je baziran na vizualnom izgledu relacije koja se u aplikacijama najčešće prikazuje kao pravokutno područje koje se sastoji od stupaca (atributa) i redaka (n-torki) za koje vrijedi:

- Sve vrijednosti unutar jednog atributa su istog tipa.
- Svaka vrijednost za sebe unutar n-torke predstavlja samo određeni broj ili skup znakova i ništa više. Ako se promatra samo jedna vrijednost, ne može se ništa doznati o ostalim vrijednostima atributa, niti o ostalim vrijednostima u n-torci.
- Unutar jedne relacije ne smiju postojati dvije n-torke s identičnim vrijednostima svih atributa.
- Redoslijed n-torki unutar relacije je potpuno nebitan.
- Svi atributi unutar relacije moraju imati različita imena. Njihov redoslijed također nije bitan.
- Različite relacije mogu imati ista imena atributa, ali u bazi ne mogu postojati dvije relacije s istim imenom.

Entitet je objekt realnog svijeta o kojemu se informacije skupljaju i obrađuju. Svaki entitet je u bazi podataka predložen jednom relacijom odnosno tablicom.

Atribut je svojstvo određenog objekta o kojemu skupljamo podatke u bazi podataka. Domena atributa je skup svih vrijednosti koje atribut može poprimiti.

U svrhu očuvanja konzistencije podataka potrebno je uvesti neka ograničenja:

- ključ (engl. key),
- cjelovitost entiteta (engl. entity integrity) i
- referencijalna cjelovitost (engl. referential integrity).

Ograničenje ključa, odnosno primarni ključ (engl. primary key) jest atribut čija vrijednost jednoznačno određuje pojavljivanje svake n-torke. U jednoj relaciji ne mogu postojati dvije n-torke sa istim vrijednostima primarnog ključa.

Osim primarnog ključa, postoje i strani ključevi (engl. foreign key). Strani ključ je atribut jedne relacije, koji se u drugoj relaciji pojavljuje kao primarni ključ.

Sljedeće ograničenje je cjelovitost entiteta. Ono je određeno tako da primarni ključ svake n-torke bude različit od *null* vrijednosti. Veza između pojedinih tablica unutar baze podataka se ostvaruje na logičkoj razini upotrebom primarnih i stranih ključeva.

Pravilo referencijalne cjelovitosti glasi: baza podataka ne smije sadržavati vrijednosti stranog ključa za kojeg ne postoje odgovarajuće vrijednosti primarnog ključa u baznoj relaciji.

primarni ključ

n-torka

Atribut 1 (PK)	Atribut 2	Atribut 3	Atribut 4
1	A	10	A1
2	A	20	A2
3	B	10	A1
4	C	40	B1
5	D	10	B2

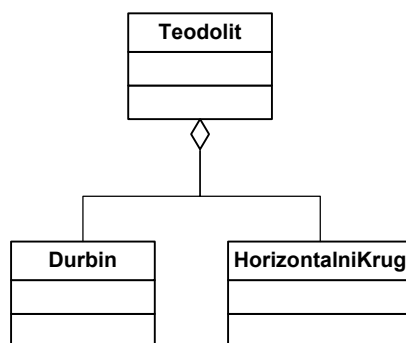
Slika 1. Relacija (tablica)

2.2. Objektni pristup

Objektni pristup uvodi pojam klase. Klase je najlakše objasniti ako ih usporedimo sa stvarima iz prirode koje nas okružuju. Sve te stvari imaju razne attribute (svojstva) i ponašaju se na razne načine. To ponašanje možemo predočiti kao niz operacija. Također, stvari možemo smjestiti u kategorije. Upravo te kategorije u svijetu objektnog modeliranja i programiranja predstavljaju klase. Klasa je kategorija ili grupa stvari koje imaju iste attribute i isto se ponašaju. Objekt je instanca klase.

Za razliku od relacijskih baza podataka, veze kod objektnih baza podataka jesu ili oblika "je" (engl. "is a") ili oblika "ima" (engl. "has a"). Prvi oblik omogućuje izgradnju hijerarhije te dozvoljava nasljeđivanje. Drugi oblik dozvoljava strukture koje pokazuju agregaciju, tj. na koji je način objekt sačinjen od drugih objekata.

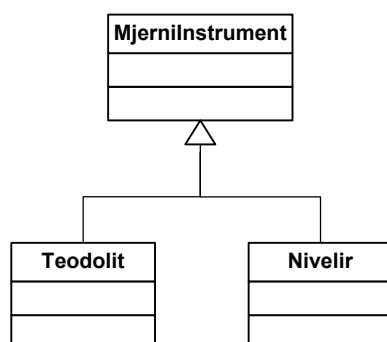
UML (engl. Unified Modeling Language, UML) dijagram klasa (Slika 2) mogao bi se interpretirati kao: teodolit sadrži ("has a") durbin i horizontalni krug.



Slika 2. Agregacija klasa

Nažalost, trenutno postoji relativno malo komercijalnih baza podataka koje mogu implementirati strukturu "has a" direktno. Umjesto toga, implementacija se obavlja korištenjem tehnika relacijskih baza podataka i veza "jedan na mnogo" (engl. "one to many"), tj. pohranom stranog ključa u svaku od podklasa.

Iako su Objektno orijentirani sustavi za upravljanje bazama podataka relativno rijetki, hibridni objektno-relacijski su mnogo učestaliji (Oracle, PostgreSQL). Oni također nemaju mogućnost "has a" struktura, tj. agregacija, ali zato imaju mogućnost "is a" struktura, tj. generalizacija ili nasljeđivanja (Slika 3). To omogućuje da hijerarhijski niže klase nasljeđuju attribute od klasa koje su hijerarhijski više.



Slika 3. Generalizacija klasa

Ovaj bi se UML dijagram klasa mogao interpretirati kao: teodolit je ("is a") mjerni instrument i nivelir je ("is a") mjerni instrument.

3. Prostorne baze podataka - dosadašnja postignuća

Sustavi za upravljanje prostornim bazama podataka (engl. Spatial Database Management System, SDBMS) imaju za cilj učinkovito upravljanje podacima vezanim uz prostor. Prostorne baze podataka jesu aktivno područje istraživanja već preko 20 godina. Rezultati tih istraživanja, npr. višedimenzionalni prostorni indeksi, korišteni su u raznim područjima. Trenutna istraživanja usmjerena su na poboljšanje funkcionalnosti i učinkovitosti. Glavna pokretačka sila za to proizlazi iz potreba postojećih aplikacija poput GIS aplikacija (engl. Geographic Information Systems, GIS), CAD aplikacija (engl. Computer Aided Design, CAD), DWH aplikacija za skladištenje podataka (engl. Data Warehousing, DWH), te raznih drugih.

3.1. *Koncept prostora*

Postoje različiti načini modeliranja i prikaza prostornih podataka. Oni se oslanjaju na stroge i ponekad komplicirane matematičke temelje koji mogu znatno otežati izradu modela. Međutim, za potrebe izučavanja prostornih baza podataka i manipulacija prostornim podacima možemo pojednostaviti pristup i ograničiti se na nekoliko osnovnih geometrijskih koncepata koji zadovoljavaju naše potrebe:

Prije svega, područje interesa je R^d , zajedno sa euklidskom udaljenošću. Nazivamo ga još i euklidski prostor te ukoliko nije drugačije specificirano, najčešće pretpostavljamo da je dimenzija $d=2$.

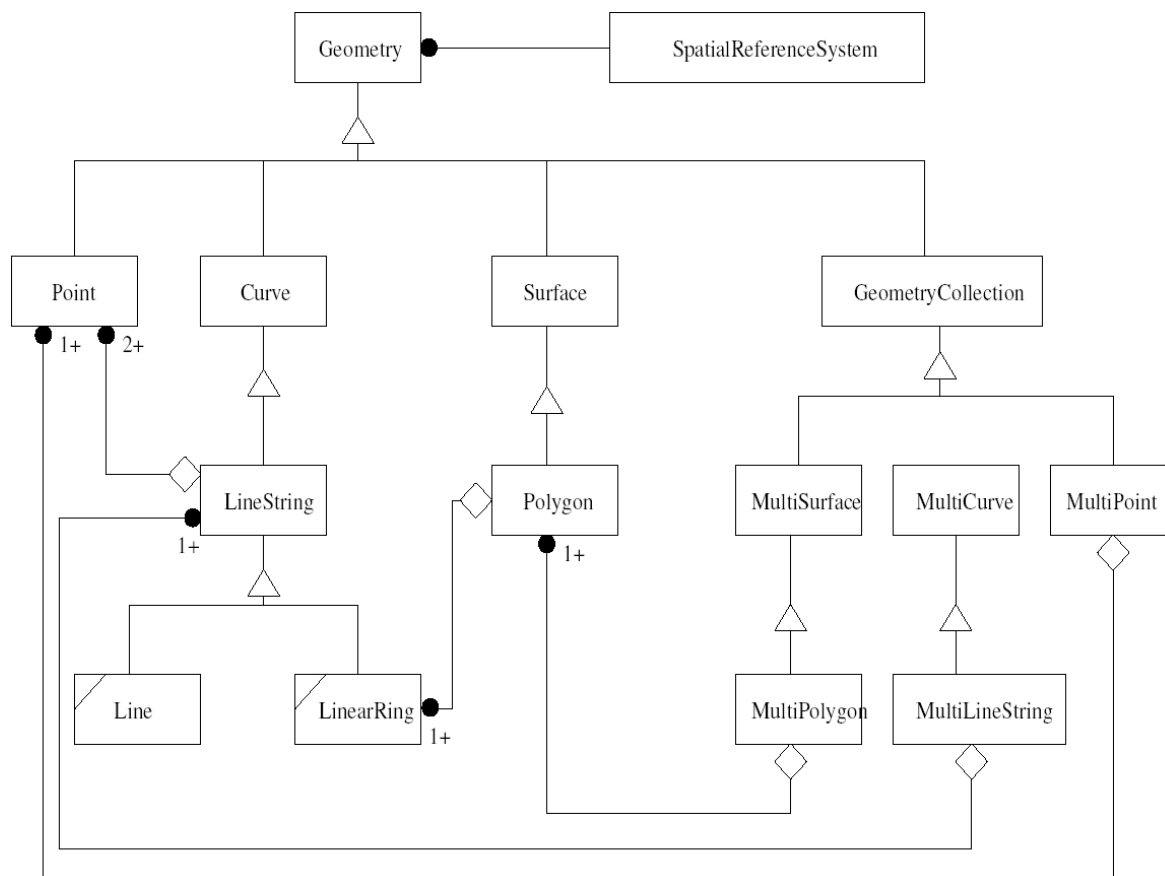
Točke su elementi prostora. Točka ima par (kartezijevih) koordinata koje najčešće označavamo sa x (apcisa) i y (ordinata). Iako su točke u euklidskom prostoru iznimno korisne za prikaz položaja i oblika objekata Zemljine površine, potrebno je naglasiti da je Zemlja približno kugla (dakle, zakrivljena površina) te je za prikaz geografskih entiteta u ravnini potrebna njihova konverzija primjenom neke kartografske projekcije.

I konačno, najčešće usmjeravamo svu pažnju na područje R^2 koje sadrži relevantne objekte. Ono je ograničeno dovoljno velikim četverokutom čiji su rubovi paralelni s osima koordinatnog sustava. To se područje naziva još i ugrađen prostor (engl. embedded space), a ponekad i prostor pretraživanja (engl. search space).

3.2. *Model prostornih podataka i upitni jezik*

Postoje dva uobičajena modela prostornih informacija: model temeljen na području (engl. field-based model) i model temeljen na objektima (engl. object-based model). Kod modela temeljenog na području, svakoj točki prostora pridružena je jedna ili više vrijednosti atributa, definiranih kao neprekidne funkcije od x i y . Npr. nadmorska je visina tako funkcija definirana preko x i y , čiji je rezultat vrijednost varijable h za svaku točku 2D prostora. Model temeljen na objektima tretira prostor kao da je nastanjen entitetima sačinjenim od tri komponente: opis, prostorna komponenta i identitet (kako bi se međusobno razlikovali budući da mogu dijeliti iste osobine, tj. imati isti opis).

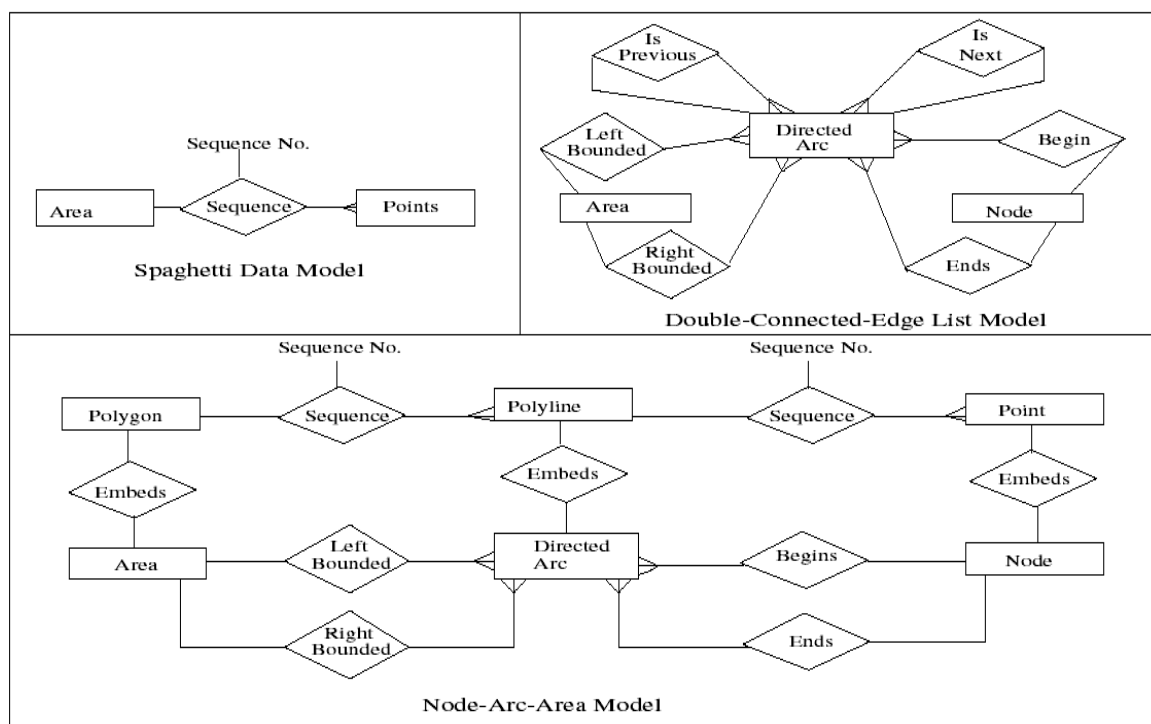
Implementacija modela prostornih podataka u kontekstu objektno-relacijskih baza podataka sastoji se od seta tipova prostornih podataka i operacija nad njima. U proteklih desetak godina odrađen je velik posao na izradi prostornih apstraktnih tipova podataka (engl. Abstract Data Type, ADT) i njihovom ugrađivanju u upitni jezik. Slika 4. prikazuje hijerarhijski prikaz tipova prostornih podataka.



Slika 4. Hijerarhijski prikaz tipova prostornih podataka

3.3. Izvođenje prostornih upita

Učinkovito izvođenje prostornih upita zahtjeva učinkovit prikaz i učinkovite algoritme. Uobičajeni prikaz prostornih podataka u objektnom modelu uključuje špageti model, NAA model i DCEL model. Možemo ih prikazati ER dijagramom (engl. Entity-Relationship diagram, dijagram odnosa entiteta) (Slika 5).



Slika 5. ER dijagrami uobičajenih prikaza prostornih podataka

U špageti modelu, geometrija bilo kojeg prostornog objekta iz skupa prostornih objekata, opisana je neovisno o ostalim prostornim objektima iz tog skupa. Kod takvog modela nema pohranjene topologije te se svi topološki odnosi računaju tek na upit. Ovakva struktura također implicira i redundanciju podataka. Npr. granica između dviju graničnih područja prikazana je dvaput. Ovaj jednostavni model omogućuje heterogeni prikaz koji sadrži točke, polilinije i područja bez ikakvih ograničenja. Valja napomenuti da se ovdje polilinije mogu presijecati u ravnini bez eksplicitnog pohranjivanja točaka presijecanja u bazu podataka. Glavna prednost ovakvog pristupa je njegova jednostavnost. Međusobna neovisnost pohranjenih objekata omogućuje krajnjem korisniku jednostavan unos novih objekata u skup. S druge strane, nedostatak ovakvog modela uglavnom je nepostojanje informacija o topološkim odnosima prostornih objekata, poput graničenja i sadržavanja. Ne postoji direktan način da se otkrije dijele li granice dvaju poligona zajedničku točku. Nadalje, kako ne postoji dijeljenje informacija, podaci se pohranjuju sa određenom dozom redundancije što može predstavljati problem zbog ionako već velike količine podataka u modelu.

Temeljni entiteti koji sačinjavaju NAA model (engl. Node Arc Area, NAA) jesu čvor (točka), usmjereni luk (polilinija određene orijentacije koja počinje u nekom čvoru i završava u nekom čvoru) i površina (2D veličina zatvorena lukovima). U ovom modelu svaki usmjereni luk ima jedan početni i jedan završni čvor. Svaki čvor mora biti početni čvor, završni čvor, ili oboje, barem jednom usmjerenom luku. Jedan ili više usmjerenih lukova zatvaraju svaku površinu. Usmjereni lukovi presijecaju se samo u čvorovima. Svaki usmjereni luk ima točno jednu površinu s lijeve strane i jednu površinu s desne strane. Svaka površina mora biti lijeva površina, desna površina, ili oboje, barem jednom usmjerenom luku.

DCEL (engl. Doubly Connected Edge List, DCEL) je poznata varijanta topološke strukture. Princip je isti kao i kod topološkog modela, ali je naglasak stavljen na linijski segment umjesto na luk. Svakom linijskom segmentu (bridu) pridružena su dva pokazivača (engl. pointer): jedan na sljedeći segment i jedan na prethodni.

Prostorni upiti su često izvođeni korištenjem tehnika filtriranja i poboljšavanja (engl. filter and refine techniques). Najprije se koristi približna geometrija predstavljena minimalnim ortogonalnim graničnim pravokutnicima raširenih prostornih objekata, kako bi se u kratkom vremenu filtriranjem izbacili svi irelevantni objekti. Potom se koristi točna geometrija za preostale prostorne objekte kako bi se dovršilo izvođenje prostornog upita. Strategije za prostorne upite uključuju skeniranje i pretraživanje indekasa u kombinaciji sa algoritmima za pretraživanje ravnine. Strategije za prostorna spajanja uključuju ugniježdene petlje, usporedbe stabala (kada su indeksi prisutni na svim sudjelujućim relacijama), te dijeljenje prostora (kada su indeksi odsutni). Kako bi se ubrzali izračuni kod velikih prostornih objekata (poligoni često imaju i preko 1000 bridova), koriste se indeksi objekata kod raširenog filtriranja. Strategije poput aproksimacije objekata i usporedbe stabala potječu od prostornih baza podataka, ali se potencijalno mogu primijeniti i u ostalim područjima sličnih karakteristika.

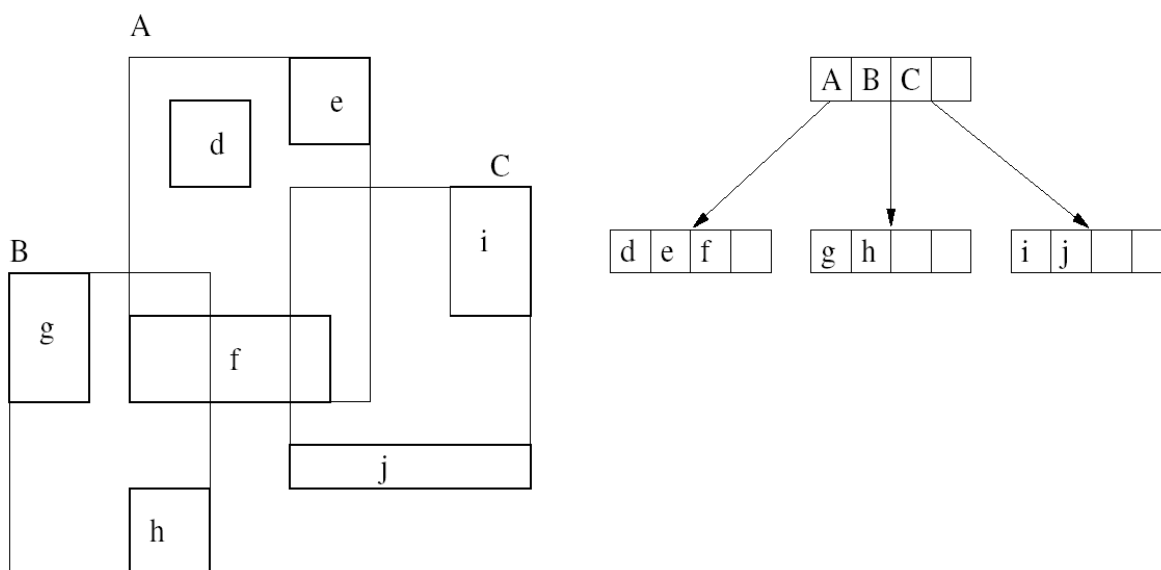
3.4. Organizacija datoteka prostornih podataka i indeksi

Fizička izvedba prostornih baza podataka je takva da nastoji optimizirati instrukcije uređajima za pohranu podataka kod obavljanja uobičajenih operacija nad datotekama s prostornim podacima. Kreiranje datoteka za sekundarnu pohranu uključuje metode ugrožnjavanja (engl. clustering) te metode prostornog raspršenja (engl. hashing). Tehnika prostornog ugrožnjavanja je mnogo zahtjevnija od klasičnog ugrožnjavanja zbog toga što ne postoji prirodna dosljednost u višedimenzionalnom prostoru u kojem se prostorni podaci nalaze. Komplikacije se ovdje pojavljuju samo zbog činjenice što je diskovni uređaj za pohranu podataka, logički gledano, jednodimenzionalni uređaj. Stoga je potrebno obaviti preslikavanje iz višedimenzionalnog prostora u jednodimenzionalni prostor na način da elementi koji su bliže u prostoru budu preslikani bliže na liniji i da se dvije točke iz prostora nikada ne preslikaju u istu točku na liniji. Predloženo je nekoliko metoda preslikavanja od kojih niti jedna nije idealna. Najpoznatije od njih uključuju: rednu krivulju, Hilbertovu krivulju i Z - krivulju (Slika 6).



Slika 6. Krivulje za linearizaciju višedimenzionalnog prostora

Klasični jednodimenzionalni indeksi poput B^+ stabla (engl. B^+ tree) mogu se koristiti za prostorne podatke linearizacijom višedimenzionalnog prostora pomoću krivulje kao što je Z - krivulja (Slika 6). Velik broj prostornih indeksa bio je istražen za višedimenzionalni euklidski prostor. Jedna od prvih metoda dohvata prostornih podataka, kreirana za manipulaciju raširenih objekata, bila je Guttmanovo R-stablo (engl. R-tree). R-stablo je visoko usklađena prirodna ekstenzija B^+ stabla za više dimenzije. Upotrebljava se za indeksiranje višedimenzionalnih informacija kao što su npr. (x, y) koordinate prostornih geografskih podataka. Struktura R-stabla dijeli prostor pomoću hijerarhijski ugniježđenih i po mogućnosti preklopljenih minimalnih graničnih pravokutnika (engl. Minimum Bounding Rectangles, MBR) (Slika 7).



Slika 7. Prostorni objekti organizirani prema R-stablo hijerarhiji

Minimalni granični pravokutnik često se još naziva i granični okvir (engl. bounding box). Svaki čvor R-stabla ima varijabilni broj unosa. Svaki unos unutar čvora koji nije završni čvor pohranjuje dva podatka: način za identifikaciju čvora djeteta (engl. child node) i granični okvir svih unosa unutar čvorova djece.

Algoritmi za umetanje i brisanje koriste granične okvire čvorova kako bi osigurali da obližnji elementi budu smješteni u isti završni čvor (novi element će ući u onaj završni čvor koji zahtijeva najmanje proširenje svog graničnog okvira). Svaki unos unutar završnog čvora pohranjuje dvije informacije: način za identifikaciju stvarnog elementa podatka i granični okvir elementa podatka.

Na sličan način, algoritmi za pretraživanje koriste granične okvire kod donošenja odluke dali pretraživati unutar čvora djeteta. Na taj način, većina čvorova ostaje netaknuta tijekom pretrage.

Postoje algoritmi i za dijeljenje čvorova kada postanu preveliki. Rezultat toga jesu kvadratni i linearni podtipovi R-stablo struktura. Postoje još mnoge varijacije R-stablo struktura, a osnovna svrha njihovog postojanja jest pronalaženje novih strategija za održavanje i usklađivanje stabla u slučaju dijeljenja, te minimiziranje preklopa graničnih okvira u svrhu smanjivanja vremena pretraživanja.

4. Modeliranje 3D objekata u geo-DBMS sustavima

Geo-DBMS sustavi omogućuju upravljanje velikom količinom prostornih podataka kojima može istovremeno pristupati više korisnika. Ti se prostorni skupovi podataka uglavnom sastoje od 2D podataka, iako je sve veći broj aplikacija ovisan o 3D podacima. Neki od tipičnih primjera za to jesu aplikacije za 3D katastar, telekomunikacije i urbano planiranje. Današnji geo-DBMS sustavi ne podržavaju 3D primitivne tipove podataka, ali 3D prostorni objekti mogu biti modelirani pomoću 2D primitivnih tipova podataka kao što su npr. poligoni. To je ostvarljivo zahvaljujući 3D koordinatama, koje su podržane u geo-DBMS sustavima. Kod takvog pristupa, više 2D poligona omeđuje 3D objekt. Ti 2D poligoni mogu biti pohranjeni kao jedan zapis u obliku skupine poligona (engl. multi-polygon) ili više zapisa pri čemu se svaki poligon zasebno pohranjuje. Nedostatak pravih 3D primitivnih tipova podataka u geo-DBMS sustavima uzrokuje dva osnovna problema:

- Geo-DBMS sustavi ne prepoznaju 3D prostorne objekte jer nemaju ugrađene 3D primitivne tipove podataka kojim bi ih modelirali. Kod izvođenja DBMS funkcija to rezultira time da one ne rade kako treba. Kada se 3D objekti pohranjuju, bilo kroz jedan ili više zapisa, ne postoji veza između različitih 2D poligona koji tvore neki objekt. Ne postoji provjera valjanosti unesenih poligona te je moguće unijeti bilo koji skup poligona. Osnovna mana toga je što se neke koordinate mogu pojaviti i više puta čime se povećava rizik od nekonzistentnosti. Također, ne postoje informacije o vanjskim ili unutarnjim granicama poliedra.
- Ako su 2D poligoni koji omeđuju 3D objekt pohranjeni u obliku više zapisa, pojavljuje se veza $1:n$ između objekta i pojedinih zapisa. Jasnije i efikasnije administriranje ovakvih velikih skupova podataka iziskuje veze $1:1$ između objekta u stvarnosti i objekta u bazi podataka.

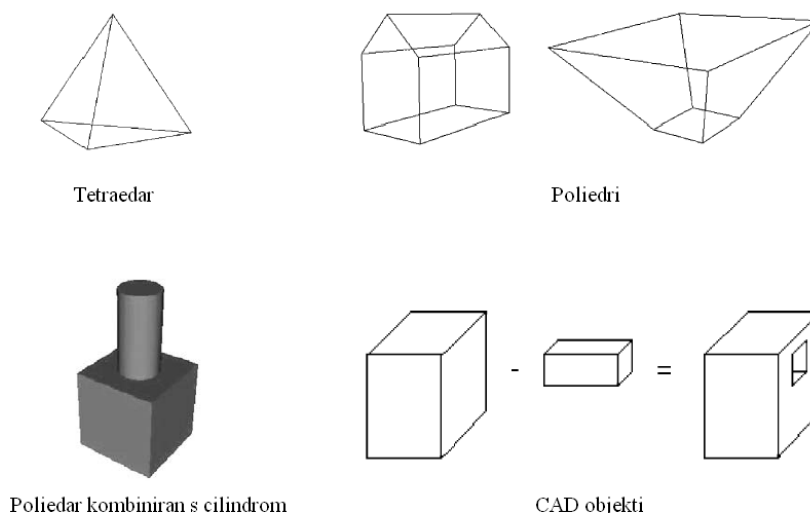
4.1. 3D primitivni tip podataka

Trenutno, geo-DBMS sustavi imaju mogućnost pohrane, provjere valjanosti i pretraživanja prostornih podataka u 2D prostoru. 2D prostorni objekti su pohranjeni kao 2D primitivni tipovi podataka (poligoni). Međutim, za pohranu 3D prostornih objekata bez nailaženja na spomenute probleme, potreban je 3D primitivni tip podataka. Postoje razni 3D primitivni objekti (Slika 8) koji mogu biti korišteni za modeliranje 3D prostornih objekata:

- *Tetraedar*: to je najjednostavniji 3D primitivni objekt. Sastoji se od 4 trokuta koji tvore zatvoreni objekt u 3D prostoru. Objekt je jasno definiran budući da tri točke svakog trokuta uvijek leže u istoj ravnini. Relativno je jednostavno kreirati funkcije koje bi radile s njim. Loša strana je ta što bi bio potreban veliki broj tetraedara za konstrukciju stvarnog objekta, što odudara od principa veze $1:1$ između objekta u stvarnosti i njegovog prikaza u bazi podataka.
- *Poliedar*: to je ekvivalent poligonu, ali u 3D prostoru. Načinjen je od ravnih ploha koje zatvaraju 3D prostor. Prednost mu je ta što jedan poliedar može biti ekvivalent jednom stvarnom objektu. Budući da poliedar može imati rupe u vanjskoj i unutarnjoj granici, pomoću njega se mogu modelirati

mnoge vrste objekata. Mana je ta što će operacije sa zonama (engl. buffer) rezultirati ne-poliedarskim objektima jer će sadržavati sferne i cilindrične dijelove koji se ne mogu prikazati poliedrima. Rješenje tog problema leži u aproksimaciji rezultata operacija sa zonama.

- *Poliedar kombiniran sa sfernim i cilindričnim dodacima*: to je ekvivalent sadašnjih 2D geometrijskih modela podataka većine geo-DBMS sustava. Na ovaj način moguće je modelirati 3D objekte još realističnije. Međutim, ovdje također postoje problemi pri operacijama sa zonama, a i samo modeliranje predstavlja dosta kompleksan zadatak.
- *CAD objekti*: ovdje postoje razne mogućnosti, međutim ti objekti ili ne odgovaraju OGC (engl. Open Geospatial Consortium, OGC) specifikacijama za 2D geometrijski model podataka ili su previše kompleksni za modeliranje izvan okruženja naprednog grafičkog korisničkog sučelja.



Slika 8. 3D primitivni objekti

Implementacija 3D primitivnih tipova podataka trebala bi voditi prema ispravnim objektima i jednostavnom kreiranju učinkovitih algoritama. Također bi trebalo uzimati u obzir i veličinu redundancije kod njihove pohrane. Evaluacija mogućih 3D primitivnih tipova podataka dana je tablicom (Tablica 1).

Tablica 1. Evaluacija mogućih 3D primitivnih tipova podataka

	Provjera valjanosti	Realističnost	Modeliranje	Algoritmi
Tetraedar	++	+	-	++
Poliedar	+	+	++	+
Poliedar kombiniran sa sfernim i cilindričnim dodacima	-	++	-	+/-
CAD objekti	-	++	+/-	--

Temeljem evaluacije mogućih 3D primitivnih tipova podataka, zaključeno je kako je poliedar najviše zadovoljio postavljene kriterije te je stoga najpogodniji za implementaciju u prostornim bazama podataka.

4.2. Provjera valjanosti

Provjera prostornih podataka prilikom unošenja u bazu podataka od velike je važnosti. Provjera valjanosti bitna je kako bi se moglo ispravno manipulirati objektima. Provjera valjanosti čovjeku može izgledati jednostavna, međutim računalu je potrebna velika količina pravila za provjeru prostornih podataka. Funkcije za provjeru valjanosti sadrže vrijednost tolerancije kao ulazni parametar. Ona je uvedena kako bi se riješio problem blagog odstupanja točaka, koje čine pojedine poligone, od zajedničke ravnine. Do toga može doći zbog geodetskih metoda mjerenja te zbog zaokruživanja koordinata kod računalne obrade mjerenja. Stoga, ravne plohe poliedara jesu ravne, ali unutar neke tolerancije. Važno je da njena vrijednost iz navedenih razloga ne bude nula, ali je isto tako bitno da ne bude niti prevelika jer bi tada i neispravni objekti bili prihvaćani kao valjani. Dobra mjera za toleranciju je standardno odstupanje geodetskih mjerenja.

Funkcije za provjeru valjanosti prostornih podataka sastoje se od nekoliko zasebnih testova. Prvi test odnosi se na provjeru pohrane podataka. Nakon toga slijedi ispitivanje ravnoće ploha koje čine poliedar. Sljedeći test je provjera dali poliedar zatvara jedan volumen u 3D prostoru. Ako poliedar zadovoljava ove testove, znači da je valjan kao cjelina. Potrebno je još ispitati plohe koje ga sačinjavaju. Svakoj od njih mora biti moguće odrediti površinu i nijedna ne smije sama sebe presijecati. Posljednji test provjere valjanosti ispituje dali su vrhovi koji čine pojedine poligone ispravno orijentirani. Gledajući s vanjske strane, vrhovi koji čine poligone moraju biti poredani u smjeru obrnutom od kazaljke na satu, tj. ako se promatra iz unutrašnjosti objekta, u smjeru kazaljke na satu. Ako su zadovoljeni svi kriteriji provjere valjanosti, prostorni objekt se može smatrati valjanim.

4.3. Prostorni indeksi

Uz uporabu 3D primitivnih tipova podataka, potrebno je razmotriti i primjenu 3D prostornih indeksa. U mnogim prostornim aplikacijama, dimenzije u x,y ravnini su mnogo veće od onih u smjeru z osi. Npr. tipičan plan grada pokriva područje od 5 x 5 km sa zgradama visine do 50 m. To, a i činjenica da upiti u pravilu nastoje pronaći objekte u konkretnoj (x,y) okolini, može učiniti 3D prostorne indekse manje korisnim kod takvih aplikacija. Međutim, neki testovi potvrđuju i opravdanost uporabe 3D prostornih indeksa u specifičnim situacijama.

5. 3D prostorne operacije u geo-DBMS sustavima

Sustav za upravljanje bazama podataka, ili DBMS, računalni je softverski program osmišljen kao sredstvo za upravljanje svim bazama podataka koje su trenutno instalirane na tvrdom disku sustava ili mreži. Postoje različite vrste sustava za upravljanje bazama podataka, od kojih su neke dizajnirane za nadzor i odgovarajuću kontrolu baza podataka konfiguriranih za posebne namjene. U prostornim bazama podataka, prostorni su tipovi podataka obično definirani kao apstraktni tipovi podataka (engl. Abstract Data Type, ADT), odnosno učahureni tipovi podataka zajedno s prostornim operacijama. Na implementacijskoj razini, mogu se definirati prostorni indeksi na prostornim ADT-ovima. Prostorni objekt je instanca prostornog tipa, te može imati 0 (točka), 1 (linija), 2 (poligon) i 3 (tijelo) dimenzije. Svi podaci pohranjeni u DBMS sustavu su u konačnici u binarnom obliku. Prostorni DBMS sustavi imaju unaprijed definiran način organiziranja binarnih podataka za prikaz geometrije. Taj je unaprijed definiran način organiziranja binarnih podataka ugrađen u DBMS u obliku tipa podataka, kao što je SDO_GEOMETRY u Oracle-u ili ST_GEOMETRY u IBM-ovom DB2. Budući da je ovaj tip podataka ugrađen u DBMS nazivaju ga izvornim geometrijskim tipom. Međutim, nedostatak 3D primitivnih tipova podataka, kao što je poliedar, rezultira nedostatkom 3D prostornih operacija.

5.1. Karakteristike poliedra

Poliedar je geometrijsko je tijelo omeđeno ravnim plohamo odnosno poligonima. Te plohe se zovu strane. Dužine u kojima se sastaju dvije susjedne strane poliedra se zovu bridovi poliedra, a točke u kojima se sastaju susjedni bridovi su vrhovi poliedra. Poligoni koji čine poliedar moraju biti ravni, tj. sve se točke pojedinog poligona moraju nalaziti u istoj ravnini.

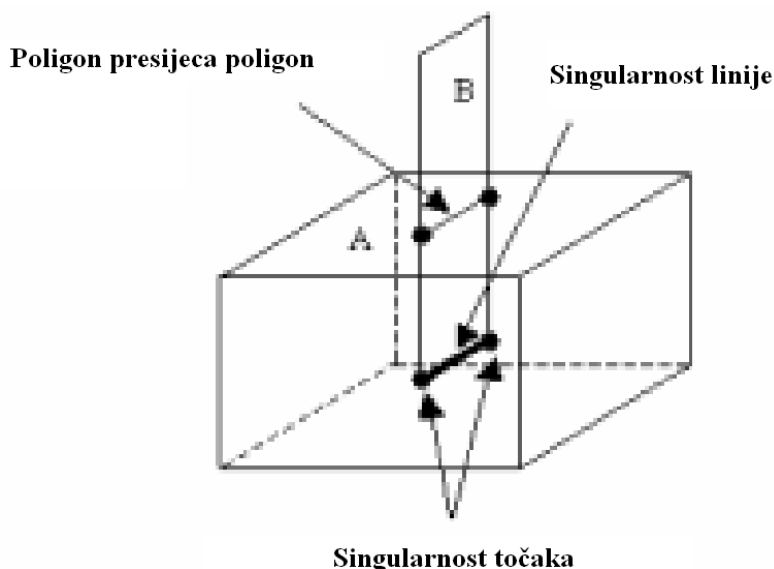
Valjani poligon mora zadovoljavati sljedeće karakteristike:

- Plosnatost - svi poligoni koji omeđuju poliedar moraju biti ravni. To znači da svi vrhovi koji sudjeluju u konstrukciji pojedinog poligona moraju biti u istoj ravnini. Plosnatost poligona može se potvrditi standardnom jednadžbom ravnine, pri čemu je vektor (A,B,C) normala na tu ravninu:

$$Ax + By + Cz + D = 0$$

- Poliedar mora biti jednovolumni objekt – skup poligona koji čine poliedar moraju zatvarati jedan volumen.
- Svaki brid ima samo 2 vrha.
- Početna i završna točka poligona su iste i pohranjuju se samo jednom. Npr. poligon kojeg čine četiri točke (a,b,c,d) će biti pohranjen kao (a,b,c,d,a) umjesto (a,b,c,d,e).
- Poligon mora imati površinu.
- Linije koje čine poligon se ne smiju presijecati.
- Singularnost poliedra nije dozvoljena, tj. niže-dimenzijski objekti ne smiju postojati unutar viših dimanzija. Npr. točka ne smije postojati unutar linije, poligona ili poliedra, linija ne smije postojati unutar poligona ili poliedra.

Zahvaljujući tom pravilu može se direktno izbjeći presijecanje poligona (Slika 9). Svi poligoni koji presijecaju druge poligone neće biti pohranjeni kao dio poliedra.



Slika 9. Presijecanje poligona uzrokuje singularnost točaka i linija

5.2. 3D tipovi podataka u DBMS sustavima

Postojeći prostorni objekti dostupni u PostgreSQL-u prilično su ograničeni na 2D, ali se pojavljuju i u trodimenzionalnom prostoru. 3D primitivni objekti, poput poliedra, nisu dostupni. Kao što je već rečeno, poliedar je izgrađen od skupa poligona koji omeđuju zatvoreni objekt. Većina DBMS sustava ima tendenciju pohranjivanja skupine poligona na način da su vrijednosti koordinata redundantne, tj. koordinate zajedničkih vrhova se pohranjuju više puta. Svaki položaj točke sastoji se od 3 vrijednosti koordinata (x,y,z). Kada bi se za skupinu poligona (engl. multi-polygon) ili poliedar pohranjivale sve vrijednosti koordinata, pohrana podataka bila bi ogromna. Međutim, taj je problem prevladan predstavljanjem svake lokacije vrha s jedinstvenim identifikatorom (ID). Zahvaljujući ovoj metodi, svaka grupa od tri vrijednosti koordinata (x,y,z) će se ponovno koristiti za svaki zajednički vrh i bit će predstavljena jedinstvenima identifikatorom, čime se direktno smanjuje prostor za pohranu poliedra. Uz pohranu jedinstvenih identifikatora vrhova, pohranjuju se i neke dodatne informacije poput ukupnog broja poligona, vrhova itd.

5.2.1. Geometrijski tip podatka

Sljedeći primjer prikazuje geometrijski tip podatka za poliedar:

```
SELECT * FROM GM_BODYTABLE WHERE PID = 1;
```

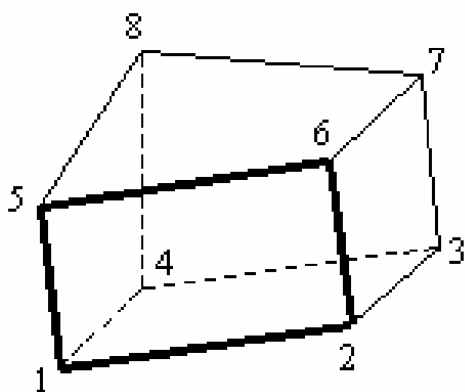
(Za geometrijski tip podatka)

```
POLYHEDRON(PolygonInfo(6,24),SumVertexList(8),SumPolygonList(
4,4,4,4,4,4),VertexList(100.0,100.0,100.0,400.0,100.0,100.0,4
00.0,400.0,100.0,100.0,400.0,100.0,100.0,100.0,400.0,400.0,10
```

```
0.0,400.0,400.0,400.0,400.0,100.0,400.0,400.0),PolygonList(1,
2,6,5,2,3,7,6,3,4,8,7,4,1,5,8,5,6,7,8,1,4,3,2))
```

- `PolygonInfo(6,24)` predstavlja 6 poligona i 24 jedinstvena identifikatora u listi poligona (`PolygonList`),
- `SumVertexList(8)` predstavlja ukupan broj vrhova,
- `SumPolygonList(4,4,4,4,4,4)` predstavlja ukupan broj vrhova za svaki poligon (ukupan broj poligona je 6),
- `VertexList()` predstavlja listu koordinatnih vrijednosti svih vrhova (bez redundancije),
- `PolygonList()` daje informaciju o svim poligonima navođenjem njihovih jedinstvenih identifikatora.

Grafički prikaz navedenog primjera prikazuje slika (Slika 10).



Ukupan broj poligona = 6

Ukupan broj vrhova = 8

Ukupan broj vrhova prvog poligona = 4

Prvi poligon = (1,2,6,5)

Slika 10. Primjer strukture poliedra za geometrijski tip podatka

5.2.2. Topološki tip podatka

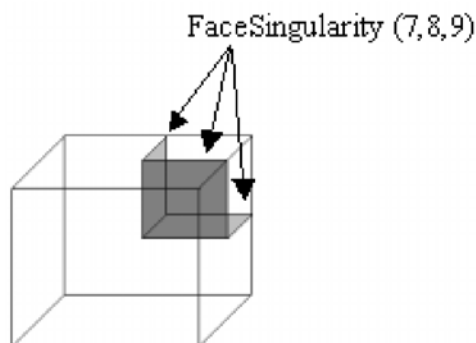
Za poliedar se kod topološkog tipa podataka pohranjuju samo jedinstveni identifikatori niže-dimenzionalnih objekata. U ovom su slučaju jedinstveni identifikatori lica (engl. face) pohranjeni kao ulazni skup podataka za poliedar (Slika 11).

```
SELECT * FROM TP_BODYTABLE WHERE PID = 1;
```

(Za topološki tip podatka)

```
4,POLYHEDRON(FaceInfo(6,3),Face(1,2,3,4,5,6),
FaceSingularity(7,8,9))
```

- `FaceInfo(6,3)` predstavlja 6 poligona i 3 singularnosti poliedra,
- `Face(1,2,3,4,5,6)` predstavlja jedinstvene identifikatore svih lica koja čine poliedar,
- `FaceSingularity(7,8,9)` predstavlja sve singularnosti poligona unutar poliedra.



Slika 11. Primjer strukture poliedra za topološki tip podatka

5.3. Implementacija u DBMS sustavima

Većina komercijalnih DBMS sustava omogućuje korisnicima kreiranje novih tipova podataka i funkcija. Takvi tipovi podataka kreirani od strane korisnika mogu biti pisani u C, C++ ili Javi. Moguće je koristiti i proceduralne jezike poput PL/SQL-a, ali njihova uporaba u npr. PostgreSQL-u, koji je pisan u C jeziku, obično rezultira lošijom izvedbom.

5.4. Topološke operacije

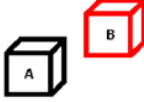
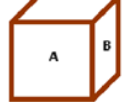
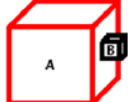
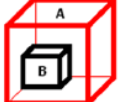
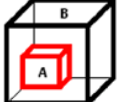
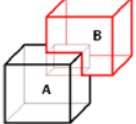
Ovdje predstavljene topološke operacije temelje se na modelu '4-presjeka' te proširuju na 3D prostor. Model '4-presjeka' koristi se kod opisivanja binarnih topoloških relacija između dva objekta (A i B) pri čemu se promatraju 4 presjeka. Model se može predstaviti matricom 2x2:

$$\begin{pmatrix} A \cap B & A \cap \partial B \\ \partial A \cap B & \partial A \cap \partial B \end{pmatrix}$$

gdje je:

A = unutrašnjost objekta A; ∂A = granica objekta A; B = unutrašnjost objekta B; ∂B = granica objekta B.

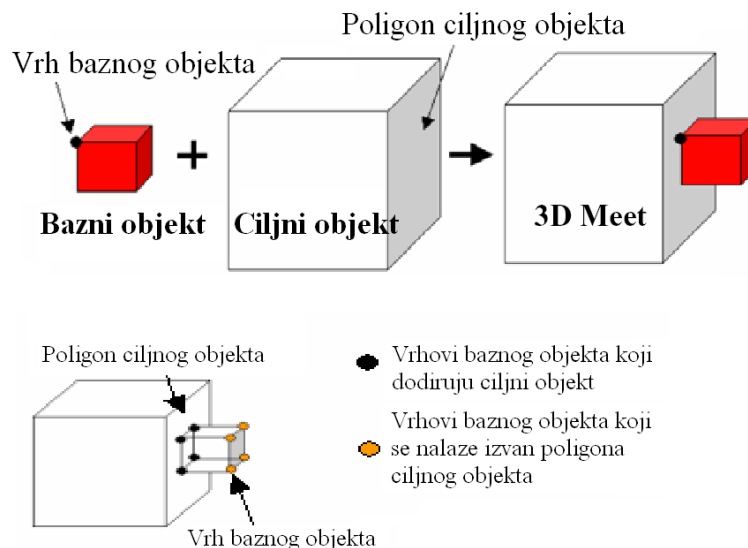
Uz uporabu oznaka za prazan skup ($\emptyset$) i ne-prazan skup ($\neg\emptyset$), kod 4-presjeka može se razlikovati $2^4 = 16$ binarnih topoloških relacija. Njih 8 je moguće realizirati za homogene dvodimenzionalne zatvorene objekte (regije) u 2D prostoru, ali se isto tako mogu primijeniti i na 3D objekte (Slika 12).

 $A \begin{pmatrix} B & \partial B \\ \emptyset & \emptyset \end{pmatrix}$ $\partial A \begin{pmatrix} \emptyset & \emptyset \\ \emptyset & \emptyset \end{pmatrix}$ disjoint	 $A \begin{pmatrix} B & \partial B \\ \neg \emptyset & \neg \emptyset \end{pmatrix}$ $\partial A \begin{pmatrix} \emptyset & \emptyset \\ \emptyset & \emptyset \end{pmatrix}$ contains	 $A \begin{pmatrix} B & \partial B \\ \neg \emptyset & \emptyset \end{pmatrix}$ $\partial A \begin{pmatrix} \neg \emptyset & \emptyset \\ \neg \emptyset & \emptyset \end{pmatrix}$ inside	 $A \begin{pmatrix} B & \partial B \\ \neg \emptyset & \emptyset \end{pmatrix}$ $\partial A \begin{pmatrix} \emptyset & \neg \emptyset \\ \emptyset & \neg \emptyset \end{pmatrix}$ equal
 $A \begin{pmatrix} B & \partial B \\ \emptyset & \emptyset \end{pmatrix}$ $\partial A \begin{pmatrix} \emptyset & \neg \emptyset \\ \emptyset & \neg \emptyset \end{pmatrix}$ meet	 $A \begin{pmatrix} B & \partial B \\ \neg \emptyset & \neg \emptyset \end{pmatrix}$ $\partial A \begin{pmatrix} \emptyset & \neg \emptyset \\ \emptyset & \neg \emptyset \end{pmatrix}$ covers	 $A \begin{pmatrix} B & \partial B \\ \neg \emptyset & \emptyset \end{pmatrix}$ $\partial A \begin{pmatrix} \neg \emptyset & \neg \emptyset \\ \neg \emptyset & \neg \emptyset \end{pmatrix}$ coveredBy	 $A \begin{pmatrix} B & \partial B \\ \neg \emptyset & \neg \emptyset \end{pmatrix}$ $\partial A \begin{pmatrix} \neg \emptyset & \neg \emptyset \\ \neg \emptyset & \neg \emptyset \end{pmatrix}$ overlap

Slika 12. Primjena 2D topoloških relacija nad 3D objektima

5.4.1. Prostorne operacije za geometrijski tip podataka

Za topološke operacije kod geometrijskih tipova podataka bit će uzeti u obzir vrhovi baznog objekta i poligoni ciljnog objekta (Slika 13). Mogući relativni položaji između njih mogu biti: unutar, izvan, dodiruju se. Ti će se odnosi koristiti za određivanje na koji se način ta dva poliedra presijecaju.



3D topološke operacije	Unutar	Izvan	Dodiruju se
Meet	NE	DA	DA

Slika 13. Bazni i ciljni objekt kod 3D operacija

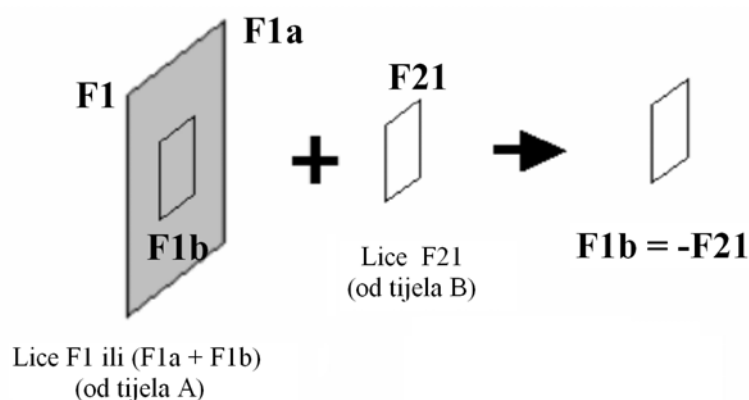
Cjelokupan prikaz odnosa između baznog i ciljnog objekta za sve prethodno navedene topološke operacije dat je tablicom (Tablica 2).

Tablica 2. Odnosi između baznog i ciljnog objekta

3D topološke operacije	Unutar	Izvan	Dodiruju se
Equal (jednak)	×	×	✓
Meet (dodiruje)	×	✓	✓
Covers (prekriva)	✓	×	✓
CoveredBy (je prekriven)	✓	×	✓
Contains (sadrži)	✓	×	×
Inside (unutar)	✓	×	×
Disjoint (odvojen)	×	✓	×
Overlap (preklapa se)	✓	✓	✓

5.4.2. Prostorne operacije za topološki tip podataka

Konvencionalni topološki tipovi podataka uključuju izradu definicija primitivnih tipova podataka i konstrukcije objekata. Osnovna svrha tih implementacija je održavanje 3D topologije, koja će biti korištena za vizualizaciju i prostorne analize. Strategija implementacije 3D topoloških operacija za 3D topološke strukture sastoji se od definiranja sličnosti između dvaju objekata. Odnos između tijela A i tijela B postoji samo u slučaju kada slično lice pohranjuje objekt A i objekt B (Slika 14).



Slika 14. Određivanje sličnosti lica dvaju tijela

6. Dohvat 3D geo-DBMS podataka korištenjem Web tehnologija

Prostorni podaci koji se koriste u GIS-u dio su cjelokupnog sustava. Stoga postoji rastuća potreba za centralnim sustavom za upravljanje bazama podataka (engl. Database Management System, DBMS), tj. sustavom u kojem se prostorni i ne-prostorni podaci održavaju u zajedničkom okružju. Budući da su tipovi prostornih podataka implementirani u (objektno-)relacijske sustave za upravljanje bazama podataka, mnogi su takvi sustavi sposobni održavati prostorne podatke u 2D. Osim same potrebe za takvim integriranim DBMS sustavima, prisutan je i trend za modeliranjem svijeta u tri dimenzije.

Ovdje prezentirano istraživanje bit će fokusirano na pitanje kako pretraživati i vizualizirati 3D geo-podatke korištenjem različitih Web tehnologija. Bit će korištena dva pristupa: jedan zasnovan na Microsoftovoj platformi, sa ASP (engl. Active Server Pages, ASP), MS Access i VRML (engl. Virtual Reality Modeling Language, VRML) tehnologijom i drugi zasnovan na XSQL (engl. Extensible Structured Query Language, XSQL), Oracle i X3D (engl. Extensible 3D, X3D) tehnologiji.

6.1. 3D geo-objekti u DBMS sustavima

GIS se razvio u prostorni sustav za upravljanje bazama podataka u kojem se prostorni i ne-prostorni podaci održavaju u integriranom okružju. Kako bi se to postiglo, većina DBMS sustava je implementirala prostorne tipove podataka i prostorne operatore više ili manje usklađene s OGC (engl. Open Geospatial Consortium, OGC) specifikacijama. Implementacija se sastoji od SQL dodataka koji podržavaju pohranu, dohvat, pretraživanje i ažuriranje jednostavnih prostornih elemenata (točke, linije, poligoni). OGC specifikacije su još uvijek 2D, isto kao i implementacija prostornih tipova podataka u DBMS sustave. Dostupne su neke 3D funkcionalnosti poput duljine i opsega te prostorno indeksiranje u 3D prostoru. Iako trenutno DBMS sustavi ne podržavaju 3D objekte, upotrebom z-koordinate se 0D, 1D i 2D objekti mogu pohraniti u 3D.

6.2. VRML i X3D

Godine 1994. Web3D konzorcij je lansirao VRML koji je 3 godine kasnije postao međunarodna ISO norma. Temelj za razvoj VRML-a bila je potreba za jednostavnim razmjenskim formatom za 3D informacije. VRML format se temelji na najviše korištenoj semantici modernih 3D aplikacija: hijerarhijske transformacije, modeli osvjetljenja, točke gledišta, geometrija, magla, animacije, karakteristike i tekstone materijala.

Razvoj VRML-a je prekinut budući da je Web3D konzorcij započeo razvoj XML (engl. Extensible Markup Language, XML) verzije VRML-a, nazvane X3D, kako bi je integrirao sa drugim web tehnologijama i alatima. Struktura podataka X3D dokumenta je slična strukturi podataka VRML datoteke. Stoga, barem što se tiče temeljnog modela podataka, na X3D treba gledati kao na podskupinu VRML-a. Razlika leži jedino u korištenoj sintaksi. Za razliku od VRML-a, X3D je kodiran u

XML-u te koristi tagove za strukturiranje. To je velika prednost za dohvat podataka, zbog lakoće korištenja XML-a u web aplikacijama.

6.3. *Pristupi*

Osnovna ideja oba pristupa bila je organizirati 3D geo-objekte u DBMS sustav te ih pretraživati i vizualizirati putem web preglednika. Geo-objekti sadrže i prostornu i ne-prostornu informaciju. Prostornu informaciju moguće je vizualizirati nakon konverzije u VRML ili X3D, a ne-prostornu uz pomoć dinamičkih HTML stranica.

Na zahtjev računala klijenta uspostavlja se veza sa DBMS sustavom te se odabere prostorna informacija i konvertira u X3D/VRML. Dodatak (engl. plug-in) klijentovom web pregledniku omogućuje pregledavanje VRML ili X3D podataka. VRML i X3D pružaju i mogućnost pokretanja skripte kada korisnik odabere neki objekt. Ta se funkcionalnost koristi kako bi se dohvatili ne-prostorni podaci vezani uz 3D geo-objekte. Putem VRML/X3D dodatka za web preglednik šalje se zahtjev računalu poslužitelju. Poslužitelj zaprima i interpretira dolazeće informacije te šalje HTML sa potrebnim informacijama natrag do klijentovog web preglednika.

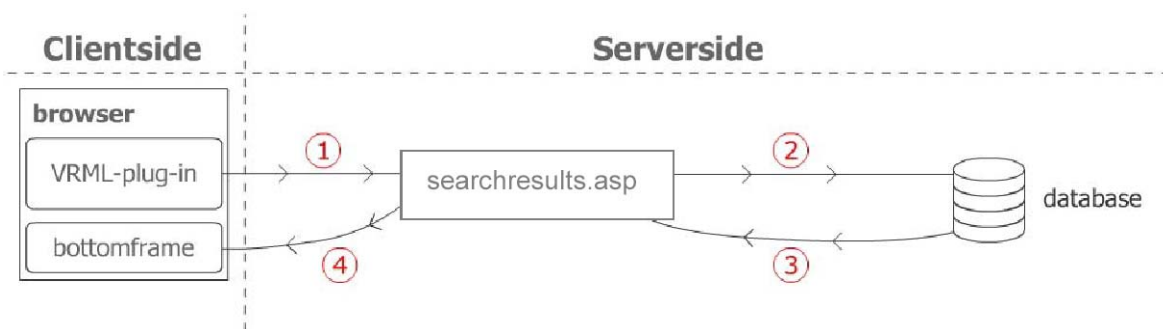
Za dohvat prostornih i ne-prostornih informacija iz DBMS sustava potrebna je tehnologija za komunikaciju između klijenta i baze podataka na poslužitelju. Za tu komunikaciju dostupno je više različitih tehnologija (ColdFusion, ASP.NET, ASP, JSP, PHP). Izbor korištene tehnologije ovisi o korištenom web poslužitelju.

Kako bi se pokazale mogućnosti pretraživanja 3D geo-objekata pomoću web klijenta, korišten je jednostavniji pristup zasnovan na Microsoftovoj tehnologiji. Nakon što je prvi pristup pokazao zadovoljavajuće rezultate, osmišljen je drugi mnogo napredniji pristup.

6.3.1. ASP, VRML i MS Access

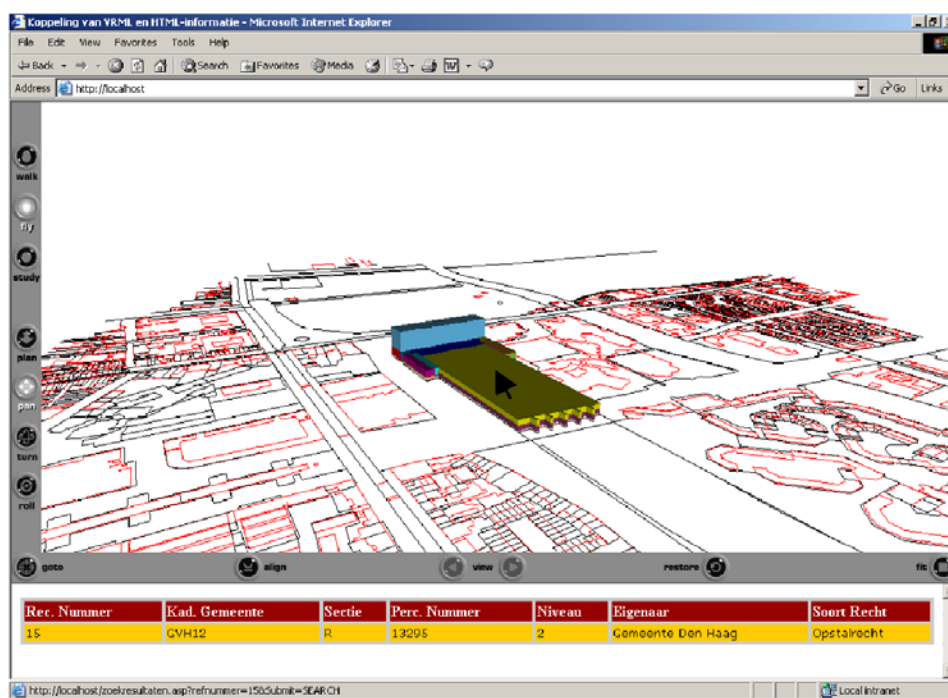
Kod prvog se pristupa samo ne-prostorni podaci mogu dinamički dohvatiti iz baze podataka. Za prostornu informaciju, prethodno se stvara (statička) VRML datoteka koja sadrži sve geo-3D objekte iz skupa podataka. Korištenjem Java programa, geografske informacije pohranjene u pojednostavljenom prostornom modelu u DBMS sustavu (Oracle) pretvaraju se u VRML datoteku. Za svaki objekt pohranjena je atributna informacija (poput npr. vlasništva) u MS Accessu.

Za komunikaciju između Internet klijenta i poslužitelja korišten je ASP. Za komunikaciju između MS Accessa i ASP-a kreirana je ODBC konekcija. Sve operacije izvršavane su na strani servera, a ne na strani klijenta (Slika 15).



Slika 15. Model prvog pristupa

Za ovaj pristup korišteno je sučelje u obliku HTML stranice sastavljene od dva dijela: jedan za prikaz VRML-a i drugi za prikaz atributnih informacija (Slika 16). Nakon otvaranja VRML datoteke u web pregledniku i odabira nekog objekta, generira se URL koji sadrži jedinstveni identifikator odabranog objekta. Na temelju toga, ASP stranica pronalazi traženu atributnu informaciju u MS Accessu te ju prikazuje u web pregledniku u obliku dinamičkog HTML-a.



Slika 16. Rezultat prvog pristupa u web pregledniku

Ovaj pristup ima tri glavna nedostatka:

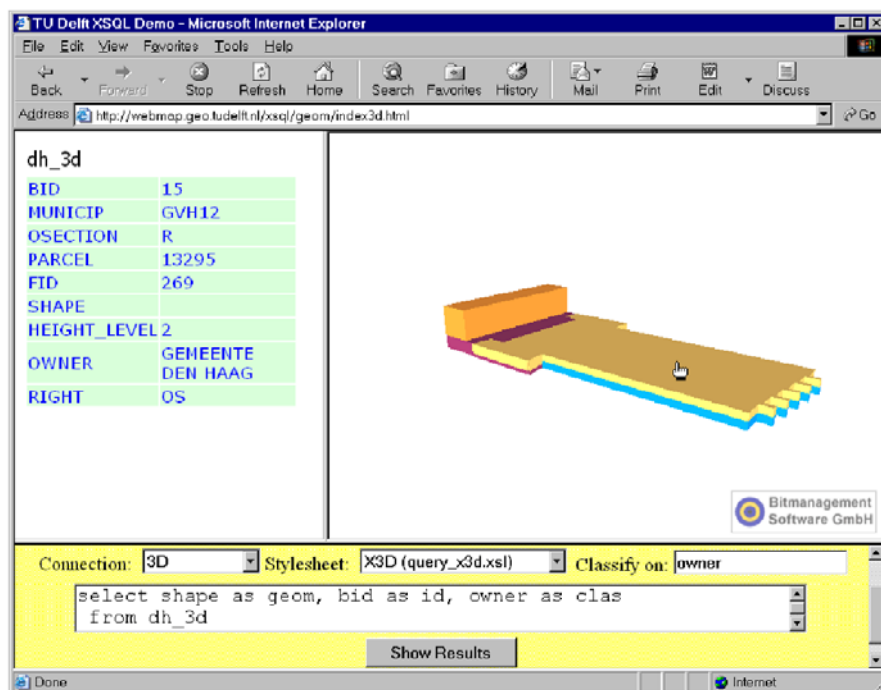
- Transformacija prostornih podataka u 3D format nije izvršena direktno, već je najprije iz podataka pohranjenih u bazi podataka proizvedena statička VRML datoteka.
- Ne-prostorni i prostorni podaci nisu pohranjeni u istom DBMS sustavu.
- MS Access ne podržava prostorne tipove podataka.

6.3.2. XSQL, X3D i Oracle

Za razliku od prethodnog, temelj drugog pristupa čine Oracle-ov XSQL Servlet i X3D. Servleti su Java klase koje djeluju u okruženju web poslužitelja. One obrađuju zahtjeve proslijeđene od strane web poslužitelja, nakon što su njemu proslijeđeni od strane klijenta. U tom smislu, XSQL obavlja istu ulogu kao i ASP kod prvog pristupa. XSQL 'stranica' je kombinacija XSQL tagova i SQL naredbi. Naredni generički primjer prikazuje XSQL stranicu sa SELECT naredbom koja vraća sve retke iz tablice:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="{@xsl}xsl"?>
<mymap connection="{@con}" xmlns:xsql="urn:oracle-xsql" >
<xsql:query rowset-element="" max-rows="5000" >
select t.{@geom}.sdo_gtype as gtype,
t.{@geom}.sdo_elem_info as info,
t.{@geom}.sdo_ordinates as geom,
{@idcol} as id
from {@table} t
</xsql:query>
```

XSQL stranica ima .xsql ekstenziju. Kada web poslužitelj zaprimi zahtjev za .xsql dokument, stranica se proslijeđuje XSQL servletu. Servlet obrađuje stranicu: kreira se veza prema bazi podataka, nakon čega se izvodi SELECT naredba. Rezultat upita koji se vraća iz baze je već u XML formatu te se prepisuje u X3D nakon čega se može vizualizirati u web pregledniku (Slika 17). Do atributnih informacija se dolazi po istom principu kao i kod prvog pristupa.



Slika 17. Rezultat drugog pristupa u web pregledniku

Drugi se pristup razlikuje od prvog po nekoliko stvari:

- X3D je korišten kao 3D grafički format umjesto VRML-a.
- Transformacija u 3D format je izvedena u realnom vremenu, direktnim pristupom Oracle-u.
- Prostorni i ne-prostorni podaci pohranjeni su u istom DBMS sustavu (u ovom slučaju čak i u istoj tablici).

Drugi je pristup neovisan o platformi te se zbog Java servlet tehnologije može implementirati i na Microsoft i na Unix web poslužiteljima. Također, iako je XSQL Servlet dio Oracle tehnologije, može se koristiti u kombinaciji i s ostalim DBMS sustavima (MySQL, PostgreSQL itd.).

7. Zaključak

Nije potrebno naglašavati kako su baze podataka prisutne u svim postojećim sektorima. Geodetskoj struci posebno je zanimljiv onaj sektor koji se bavi prostorom, njegovim modeliranjem i upravljanjem. U kontekstu struke prostor je trenutno uglavnom sveden na dvije dimenzije, dok je treća prisutna, ali najčešće u obliku nekog dodatnog atributa. Tendencije su da se prostorom upravlja kroz baze podataka na način da se u njima pohranjuju podaci o prostoru bez ukidanja pojedinih dimenzija. Na taj bi se način riješili mnogi problemi i ograničenja uzrokovana dosadašnjim pristupom. Međutim, i sam pristup modeliranja prostora u 3 dimenzije uzrokuje mnoge poteškoće. Sustavi za upravljanje bazama podataka jednostavno još nisu spremni za obavljanje potrebnih zadataka u 3D. Kao što je već istaknuto u ovom radu, postoje primjeri implementacija 3D tipova podataka i funkcija u DBMS sustavima, međutim to još uvijek nije na onoj razini kakva je postignuta sa 2D tipovima podataka. Bit će potrebno još mnogo istraživačkog rada kako bi se postigla potpuna funkcionalnost 3D baza podataka.

Literatura:

- Crowther, P., Hartnett, J. (2001): Handling Spatial Objects in a GIS Database - Relational v Object Oriented Approaches. Proceedings of the 6th International Conference on GeoComputation, University of Queensland, Brisbane, Australia, 2001.
- Shekhar, S., Chawla, S., Ravada, S., Fetterer, A., Liu, X., Lu, C. T. (1999): Spatial databases-accomplishments and research needs. IEEE Transactions on Knowledge and Data Engineering, Minneapolis, 1999.
- Arens, C., Stoter, J., Oosterom, P. van (2005): Modeling 3D spatial objects in a geo-DBMS using a 3D primitive. Computers & Geosciences, Vol. 31, 2005.
- Chen, T.K., Abdul-Rahman, A., Zlatanov, S. (2008): 3D Spatial Operations for Geo-DBMS - Geometry VS. Topology. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 37, Beijing, 2008.
- Vries, M.E. de, Stoter, J.E. (2003): Accessing a 3D geo-DBMS using Web technology. ISPRS Joint Workshop on 'Spatial, Temporal and Multi-Dimensional Data Modeling and Analysis', Québec, 2003.

Spatial databases

Abstract: *This paper shows a brief review of spatial databases. It describes some general things such as different approaches of handling spatial data, previous achievements in the field of organization of spatial data files and indices, ways of modelling 3D spatial objects, spatial operations of 3D objects and possibilities of their visualization.*

Keywords: *DBMS, index, 3D, polyhedron, visualization*

Gotovi seminarski, maturski, maturalni i diplomski radovi iz raznih oblasti, lektire , puškice, tutorijali, referati - specijalizovan tim za usluge visokokvalitetnog pisanja, istraživanja i obradu teksta za kompletan region Balkana.

Posetite nas na sajtovima ispod:

WWW.MATURSKIRADOVI.NET

WWW.SEMINARSKIRAD.ORG

WWW.MATURSKI.NET

WWW.MATURSKI.ORG

WWW.SEMINARSKIRAD.INFO

Dostupni smo Vam 24h 365 dana u godini.

Za gotove verzije rada obratiti se na mail:

maturskiradovi.net@gmail.com

061/ 11-00-105

Seminarski, diplomski, maturski radovi, prevodi na engleski i eseji...