

# Osnovni Pojmovi

---

ENTITET - Entitet je cinilac poslovanja u realnom svetu, dok je klasa entiteta skup slicnih entiteta.

POVEZNIK - Poveznik predstavlja vezu izmedju dva ili vise entiteta ili prethodno definisanih poveznika, dok je klasa poveznika skup veza prethodno definisanih klasa entiteta ili prethodno definisanih klasa poveznika.

OBELEZJE - Obelezje je osobina klase realnih entiteta ili poveznika. Obelezje moze da bude elementarno, slozeno i skupovno.

DOMEN - Domen predstavlja skup mogucih vrednosti obelezja. Mogu biti primitivni i izvedeni.

PODATAK - Podatak je uredjena cetvorka (Entitet, Obelezje, Vreme, Vrednost). Ako se eksplicitno navede samo vrednost, a implicitno nisu navedeni obelezje, entitet i vreme onda to nije podatak, jer smisao nije odredjen.

TIP ENTITETA - Tip entiteta predstavlja model klase realnih entiteta u informacionom svetu. Pojava tipa entiteta je model klase jednog realnog entiteta. Indentifikator tipa entiteta je skup obelezja i ima za zadatak da pronadje jedinstven nacin za oznacavanje bilo koje pojave tipa entiteta. Indentifikatori mogu biti interni i eksterni. Kljuc tipa entiteta je minimalni interni indentifikator tipa entiteta, takodje postoji i primarni kljuc to je jedan, izabrani kljuc iz skupa kljuceva i cesto se oznacava podvlacenjem.

TIP POVEZNIKA - Tip poveznika povezuje dva ili vise tipa entiteta ili prethodno definisanih tipova poveznika. Pojava tipa poveznika prikazuje jedan poveznik u realnom sistemu. Indentifikator tipa poveznika je niz i ima za zadatak da pronadje jedinstven nacin za oznacavanje bilo koje pojave tipa poveznika.

STRUKTURE PODATAKA - Strukture podataka se dele na 2 kategorije:

1. Prema nivou apstrakcije pridruzene sematike:

- logicka struktura obelezja
- logicka struktura podataka
- fizicka struktura podataka

2. Prema mogucem broju direktnih predhodnika i sledbenika cvora grafa

- mrezne strukture podataka
- strukture tipa stable
- linearne strukture podataka (ciklicne i aciklicne)

- Logicka struktura obelezja je struktura nad skupom tipova entiteta, tipova poveznika i njihovih atributa.

- Logicka struktura podataka definise se nad skupom podataka, putem posebne relacije, definise se u granicama zadate LSO. LSO predstavlja model za LSP. LSP se moze vizuelno prikazati pomocu grafova i tabela.

- Fizicka struktura podataka je LSP, smestena na memorijski medijum.

# Koncepcija Baze Podataka

---

BAZA PODATAKA - Osnovna ideja Baze podataka je bila da se svi podaci smestaju u jednu veliku datoteku. Uklanja se redundantno memorisanje podataka tj. ne vrsi se viestruko memorisanje istih ili slicnih podataka. Uvodi se softverski proizvod za kreiranje i koriscenje baze podataka (SUBP – sistem za upravljanje bazama podataka). Uvodi se visekorisnicki pristup i autorizacija korisnika. SUBP sadrzi: jezik za opis podataka, jezik za manipulisanje podacima i upitni jezik. SUBP je veoma skup i sadrzi veoma komplikovanu instalaciju i administraciju. Nije namenjen za skladistenje vremenskih sesija ili pretragu po tekstu.

SEMA BAZE PODATAKA - Program koji koristi usluge SUBP poznaje samo semu BP kao logicku kategoriju. Nad semom BP koristi logicku strukturu podataka i ne sme da vodi racuna o fizickoj strukturi podataka. Efekti uvođenja koncepta seme BP su: smanjenje redundantnosti podataka, smanjenje zavisnosti programa i seme BP od promena FSP, i uvođenje uloga (projektant i administrator).

PODSEMA BAZE PODATAKA – Promene seme BP mogu izazvati ceste i nepotrebne promene postojećih programa, i onda se otezano održava softverska podrška IS. Rešenje ovog problema je uvođenje koncepta i novog sloja – podsema. To je LSO, dobijena na osnovu dela seme BP. Projektuje se kao i sema BP, u procesu razvoja IS. Predstavlja model dela BP.

## Model Podataka

---

MODEL PODATAKA – Predstavlja uredjenu trojuku (S, I, O) gde je S – strukturalna komponentna koja služi za modeliranje LSO, I – integritetna komponenta koja služi za modeliranje ograničenja nad podacima u BP i O – operacijska komponentna koja služi za modeliranje dinamike izmene stanja. Modeli podataka se dele na: Model tipova entiteta i poveznika, relacioni model, mrežni model, hijerarhijski model, XML model, Objektno relacioni model, Objektno orijentisani model, Logicki model.

## ER Model

---

ER DIJAGRAM – Je pogodna dijagramska tehnika za predstavljanje modela statičke strukture realnog sistema. ER model podataka uživa popularnost zbog dijagramskog nacina prikaza seme BP.

OGRANICENJE DOMENA – Predstavlja strukturu koja se sastoji od naziva domena, ograničenje domena i predifinisana vrednost domena. Ograničenje domena se definiše primenom izabranog pravila za specificiranje korisnicki definisanog domena, pravila su: pravilo nasledjivanja, pravilo tipa sloga, pravilo tipa skupa, pravilo tipa izbora. Specijalni slucaj je Nula vrednost. Ona se oznacava sa simbolom w, u praksi to je oznaka NULL. Moguca znacenje nula vrednost su: nepoznata – postojeca vrednost obelezja, nepostojeca vrednost obelezja, neinformativna vrednost obelezja.

OGRANICENJE VREDNOSTI OBELEZJA – Definiše se za svako obeležje tipa, to je struktura koja se sastoji od Domena i NULL vrednosti može imati dozvolu ili zabranu dodele nula vrednosti obeležju. Takodje Domen i Null su obavezne komponente.

OGRANICENJE POJAVE TIPRA – Definiše ograničenje na moguće vrednosti podataka unutar iste pojave TE ili TP. Predstavlja skup ograničenja vrednosti obeležja, kojem je pridodat logici uslov.

**KARDINALITET TIPA POVEZNIKA** – Je uređen par koji se sastoji od minimalnog kardinaliteta (0, 1) i maksimalnog kardinaliteta (1, n) s tim da je n veće ili jednako od 2. Ograničava u koliko će se pojava tipa poveznika učestvovati jedna, bilo koja pojava povezanog tipa. Definise se za svaki povezani tip. Postoje 3 opšte grupe maksimalnih kardinaliteta to su: M:N, N:1 i 1:1.

**INTEGRITET TIPA** – Može biti ograničenje ključa za TE ili TP. Postoje 3 opšte grupe maksimalnih kardinaliteta to su: M:N, N:1 i 1:1. Gerund predstavlja tip entiteta dobijen transformacijom tipa poveznika, tj. tip poveznika, koji predstavlja povezani tip u nekom drugom tipu poveznika. Gerund ima dvojaku ulogu, istovremeno je i tip entiteta i tip poveznika. Tip poveznika je za neke druge, povezane tipove, a tip entiteta u nekim drugim tipovima poveznika.

**GERUND** – U ER modelu gerund je: TP nastao transformacijom TP i TP koji predstavlja povezani tip u nekom drugom tipu poveznika. Ima dvojaku ulogu kao TP i TE. Gerund se koristi kada ne mogu proizvoljne kombinacije pojava nekih tipova biti sadržane u pojavi posmatranog tipa poveznika. Gerund se upotrebljava kada ne mogu proizvoljne kombinacije pojava nekih tipova biti sadržane u pojavi posmatranog tipa poveznika i postoji pravilo koje kombinacije pojava tih tipova mogu biti sadržane u pojavi posmatranog tipa poveznika.

**AGREGACIJA** – Obezbeđuje objedinjavanje složenije ER strukture. Cela ER struktura se posmatra kao jedna tip entiteta. Najjednostavniji primer agregacije je gerund

**ID ZAVISNOST** – Postoje 2 tipa ID zavisnosti:

1. Egzistencijalna – javlja se između pojava dva tipa entiteta i postoji kada je minimalni kardinalitet poveznika jednak 1.

2. Identifikaciona – je poseban slučaj egzistencijalne zavisnosti i postoji kada su i minimalni i maksimalni kardinaliteti TP prema slabom TE jednaki 1. Uvodi klasifikaciju TP na neidentifikacioni TP i identifikacioni TP. Identifikacioni TP predstavlja identifikacionu zavisnost slabog TE. Ukazuje da svaka pojava zavisnog TE može identifikovati samo uz pomoć identifikatora nadređenog TE. Ključ zavisnog TE se formira korišćenjem ključa nadređenog TE. Zavisni TE može posedovati neprazan skup sopstvenih obeležja.

**ISA HIJERARHIJA** – Poseban koncept TP, zahteva uvođenje superklase i potklase. Pojmovi super i potklase se uvode da bi se izbegle nula vrednosti, da bi se izbeglo definisanje TP koji nema mnogo smisla i da bi model statičke strukture realnog sistema bio semantički bogatiji. Specijalizacija se vrši na osnovu vrednosti nekog skupa klasifikacionih obeležja. U TE superklase ostaju sva zajednička obeležja i primarni ključ dok u TE podklase ostaju samo specifična obeležja. Tip ISA hijerarhije se određuje na osnovu min. i max. kardinaliteta. Ako je min 0 to je parcijalna a ako je 1 to je totalna. Ako je max 1 to je nepresekna a ako je N to je presekna. Ključ svake podklase je primarni ključ superklase. Podklase mogu imati svoje sopstvene ključeve. Podklasa može imati ulogu superklase u drugoj ISA hijerarhiji.

**KATEGORIZACIJA** – Poseban koncept TP, koji zahteva uvođenje pojma kategorije. Kategorija predstavlja posebnu vrstu tipa TE ili TP – gerunda. Jedan TE se povezuje s više kategorija. Ne postoji id zavisnost posmatranog TE od kategorije ili obratno. Može a ne mora postojati skup klasifikacionih obeležja kategorije.

## Relacioni Model Podataka

**SKUP PRIMITIVNIH I SLOZENIH KONCEPTA RMP** – Za opis LSO (nivo intenzije) i LSP (nivo ekstenzije). Nivo intenzije se sastoji od domena, obeležja, seme BP, seme relacije i skupa obeležja. Nivo ekstenzije se sastoji od Baze podataka, Realcije, torke, podatka i vrednosti.

**TORKA** – je jedna pojava entiteta ili poveznika. Pomocu torke svakom obeležju iz nekog skupa obeležja dodeljujemo konkretnu vrednost.

RELACIJA – je konacan skup torki, predstavlja skup realnih entiteta ili poveznika. U relaciji se ne mogu pojaviti dve identicne torke, to je onda ista torka samo dva puta prikazana. Poredak torki u relaciji ne utice na informacije koje sa sobom nosi relacija. Relacije se uobicajeno prikazuju pomocu tabela.

OSNOVNE SKUPOVNE OPERACIJA NAD RELACIJAMA – Unija, torka moze da se pojavi samo u jednoj relaciji da bi se upisala u novu. Presek, torka mora biti u obe relacije da bi se upisala u novu i Razlika torka mora postojati samo u jednoj relaciji da bi se upisala.

SELEKCIJA – Omogucava izbor torki relacije po nekom kriterijumu. Logickom formulom izrazava se kriterijum po kojem se torke selektuju. Bice selektovane samo one torke, za koje je formula tacna.

PROJEKCIJA RELACIJE – Izvajanje vrednosti pojedinih kolona iz relacije. Projektovanje relacije na podskup skupa obelezja.

SPOJEVI –

1. Prirodni spoj – spajanje torki razlicitih relacija po osnovu istih vrednosti zajednickih obelezja.
2. Dekartov proizvod relacije – spajanje formiranjem svih mogucih kombinacija torki iz dve relacije.
3. Theta spajanje relacija – selektovanje torki po nekom kriterijumu iz dekartovog proizvoda relacija.
4. Spoljni spoj - omogucava spajanje dve tabele po vrednosti nekog atributa , ali i ukljucivanje onih torki iz druge tabele, koje ne zadovoljavaju uslov jednakosti.

OBLAST DEFINISANOSTI U RMP –

1. Vanrelaciono ogranicenje – definise se izvan konteksta seme relacije
2. Jednorelaciono ogranicenje – definise se nad tacno jednom semom relacije
3. Veserelaciono ogranicenje – definise se nad skupom ili nizom sema relacija, koji sadrzi bar dva clana

OBLAST INTERPRETACIJE U RMP –

1. Ogranicenje vrednosti – interpretira se nad tacno jednom vrednoscu nekog obelezja
2. Ogranicenje torke – interpretira se nad tacno jednom torkom bilo koje relacije
3. Relaciono ogranicenje – interpretira se nad skupom torki bilo koje relacije
4. Medjurelaciono ogranicenje – interpretira se nad barem dve, bilo koje relacije

OGRANICENJE TORKE – Izrazava ogranicenje na moguće vrednosti unutar jedne torke. Predstavlja skup ogranicenja vrednosti obelezja, kojem je pridodat logicki uslov.

KLJUC SEME RELACIJE – Minimalni podskup skupa obelezja seme relacije, na osnovu kojeg se jedinstveno moze identifikovati svaka torka relacije nad datom semom. U odredenim situacijama skup ogranicenja seme relacije zadaje se samo kao skup kljuceva.

OGRANICENJE KLJUCA – Za sva obelezja kljuka nula vrednosti su zabranjenje. Vrste obelezja seme relacije, s obzirom na kljuceve su : primarno (pripada barem jednom kljucu seme relacije) i neprimarno obelezje (ne pripada ni jednom kljucu seme relacije). Svaka sema relacije mora posedovati najmanje jedan kljuc koji moze biti: ekvivalenti (svi kljucevi skupa kljuceva) ili primarni kljuc (jedan izabrani kljuc, od svih ekvivalentnih kljuceva). U SQL se javlja UNIQUE ogranicenje

obezbedjuje jedinstvenost vrednosti jedne ili vise kolona bazne tabele. UNIQUE ogranicenje je zadovoljeno ako i samo ako ne postoje dva reda bazne tabele sa istim definisanim (NOT NULL) vrednostima u kolonama nad kojima je ogranicenje specificirano.

OGRANICENJE REFERENCIJANOG INTEGRITETA - Referencijalno ogranicenje realizuje referencijalni integritet na taj nacin sto specificira jednu ili vise kolona bazne tabele kao referencirajuce kolone i njima odgovarajuce referencirane kolone u nekoj (ne neophodno razlicitoj) baznoj tabeli.

Funkcionalna zavisnost  $X \rightarrow Y$  je nepotpuna ako sadrži logički suvišno obeležje na levoj strani.

FUNKCIONALNA ZAVISNOST –Atribut Y relacije R je potpuno funkcionalno zavisn od atributa X relacije R ako je funkcionalno zavisn od atributa X, a nije funkcionalno zavisn ni od jednog pravog podskupa atributa X. Ako je poznata X vrednost, poznata je i Y vrednost. Svako X vrednosti odgovara samo jedna Y vrednost.1

Zatvarač (zatvorenje) skupa funkcionalnih zavisnosti (oznaka  $F^+$ ) je skup koji sadrži sve logičke posledice od F.

ZATVARAC OBELEZJA – Se izracunava na osnovu f-nominalnih zavisnosti  $F=\{AB \rightarrow C, C \rightarrow D, D \rightarrow F\}$ ,  $(AB)^+ = ABCDF$

Funkcionalna zavisnost  $X \rightarrow Y$  je tranzitivna ako važi  $X \rightarrow Y$  i  $Y \rightarrow Z$  , a ne važi da je  $Y \rightarrow X$  .

# Datotecki Sistem Operativnog Sistema

---

TABELA UREDJAJA – Zapis sa podacima, neophodnim za koriscenje uredjaja. Formira se prilikom inicijalizacije OS ili montiranja uredjaja. Sadrzi podatke o karakteristikama uredjaja – disk jedinice. Sadrzi adrese rutina za upravljanje fizickim U/I za dati uredjaj.

SISTEMSKA TABELA DATOTEKE – Predstavlja trajan zapis o datoteci koji odrzava OS. Formira se prilikom kreiranja datoteke, a unistava se prilikom brisanja. Ucitava se u OM, kada se datoteka operativno koristi. Modifikuje se prilikom izmene sadrzaja datoteke.

ALOKACIONA TABELA DATOTEKE – Alokaciona tabela za fajlove korišćena na DOS i Microsoft Windows operativnim sistemima. Verzije FAT12, FAT16 i FAT32 označavaju 12-bitnu, 16-bitnu i 32-bitnu dubinu tabele. Ovaj je fajl sistem podržan od strane gotovo svih operativnih sistema za ličnu ili poslovnu primenu. Široka rasprostranjenost omogućava prenos podataka sa velikog broja različitih računara starije i novije generacije.

TABELA LOGICKIH IMENA DATOTEKA –

TABELA OTVORENIH DATOTEKA – Niz zapisa o otvorenim datoteka u celom sistemu. Svako otvaranje datoteke je jedan zapis u TOD. Svaki zapis sadrzi podatke, neophodne da bi se obavlja razmena podataka izmedju datoteke i aplikativnog programa.

TABELA OPISA DATOTEKE – Predstavlja prekopirani sadrzaj STD sa ATM u OM. Azurira se tokom upotrebe datoteke i prenosi se azurirani sadrzaj u STD, pri zavrsetku rada s datotekom.

SISTEMSKI POZIVI –

1. Create – kreiranje potpuno nove datoteke i otvaranje za pisanje. Poziva se na osnovu zadatih parametara, naziv sa putanjom, ovlasčenja, velicina...

2. Open – priprema datoteke za operativnu upotrebu. Poziva se na osnovu zadatih parametara, naziv sa putanjom, nacin otvaranja i eventualno ovlasčenja

3. Read – sistemski poziv za ucitavanje niza bajtova iz otvorene datoteke. Poziva se na osnovu zadatih parametara, logicka oznaka datoteke, odredisna promenljiva u OM, broj bajtova za citanje.

4. Write – sistemski poziv za zapisivanje niza bajtova u otvorenu datoteku. Poziva se na osnovu zadatih parametara, logicka oznaka datoteke, izvorna promenljiva u OM i broj bajtova za zapisivanje
5. Seek – pozicioniranje na zeljenu lokaciju. Poziva se na osnovu zadatih parametara, logicka oznaka datoteke, pomak i referenta tacka za pomak
6. Close – sistemski poziv za zatvaranje otvorene datoteke
7. Sync – sistemski poziv za praznjenje izmenjenih bafera
8. Delete – sistemski poziv za brisanje imena i unistavanje datoteke
9. Truncate – sistemski poziv za brisanje sadrzaja datoteke
10. Stat – preuzimanje informacije o datoteci

## Datoteke

---

SERIJSKA – Slogovi se smestaju jedan za drugim u sukcesivne memorijske lokacije. Redosled memorisanja slogova najcesce prema hronoloskom redosledu njihovog nastanka. Slogovi mogu, a i ne moraju biti blokirani. Serijska datoteka se generise najcesce u postupku obuhvata podataka. Postoje 2 nacina trazenja sloga: Trazenje slucajno odabranog i primena metode linearnog trazenja. Upis novog sloga u datoteku se postavlja na prvu slobodnu lokaciju na kraju datoteke, pre toga mora mu prethoditi jedno neuspesno trazenje. Brisanje postojeceg sloga prvo mora u prethoditi jedno uspesno trazenje, najcesce brisanje je logicko. Fizicko brisanje bi zahtevalo veliki broj pristupa. Za modifikaciju slogova takodje mora postojati jedno uspesno trazenje.

SEKVENCIJALNA – Slogovi su smesteni sukcesivno jedan za drugim. Logicki susedni slogovi smestaju se u fizicki susedne lokacije. Rastuce uredjenje po vrednosti kljuka, slog sa najmanjom vrednoscu kljuka smesta se u prvu lokaciju. Slogovi se smestaju u blokovima i pozeljno je da faktor blokiranja bude sto veci. Sekvencijalna datoteka se dobija sortiranjem serijske datoteke. Postoje 2 nacina trazenja sloga: Trazenje slucajno odabranog i logicki narednog. Obrada moze biti direktna (ima smisla ako je datoteka mala, tako da se moze smestiti u OM) i redosledna (iterativan proces, u kojem vodeca datoteka generise logicki naredne vrednosti kljuka za trazenje u obradivanoj sekvencijalnoj datoteci). Za upis novog sloga prvo mora da sledi jedno neuspesno trazenje i zatim pomeranje za jednu lokaciju udesno svih slogova sa vrednostima kljuka vecim od vrednosti kljuka novog sloga. Brisanje zahteva uspesno trazenje i pomeranje za jednu lokaciju ulevo svih slogova sa vecom vrednoscu kljuka, ako se brisanje vrsi fizicki. Modifikacija zahteva jedno uspesno trazenje.

STATICKA INDEKS SEKVENCIJALNA – Sadrzi 3 zone:

1. Primarna zona – slogovi uredjeni saglasno rastucim vrednostima kljuka, grupisani su u blokove i pozeljan je sto veci faktor blokiranja. Smestaju se u sukcesivne memorijske lokacije. Cilj je da se iskoristi pozeljne osobine sekvencijalne organizacije u redoslednoj obradi podataka.

2. Zona indeksa – Reprezentativne vrednosti kljuka svakog bloka primarne zone propagirane u stablo. Elementi listova stabla sadrze po jednu vrednost kljuka iz svakog bloka. Elementi cvorova na visim nivoima hijerarhije stabla sadrze po jednu vrednost kljuka iz svakog direktno podredjenog cvora. Nakon kreiranja, ne azurira se. Vrste zone indeksa su: Zona indeksa s propagacijom najvećih vrednosti kljuka iz svakog bloka, u slucaju poslednjeg bloka propagira se najveća dozvoljena vrednost kljuka i Zona indeksa s propagacijom najmanjih vrednosti kljuka iz svakog bloka, u slucaju prvog bloka propagira se najmanja dozvoljena vrednost kljuka.

3. Zona prekoracenja – Sadrzi kompletne slogove datoteke kao i primarna zona, koji se upisuju u zonu prekoracenja pri upisu novih sloga koji se zovu prekoracioci. Svaki blok primarne zone moze imati svoje prekoracioce. Kada blok u primarnoj zoni nije kompletan upis novog sloga dovodi do pomeranja slogova u bloku, a u suprotnom upis svakog novog sloga dovodi do upisa jednog od slogova u zonu prekoracenja. Pokazivac na pocetak lanca moze se prikazati na 2 nacina: Direktno povezivanje sa listom stabla trazenja (pokazivac na pocetak lanca smesta se u odgovarajuci list) i Indirektno povezivanje sa listom stabla trazenja (pokazivac na pocetak lanca smesta se u prateci deo odgovarajuceg bloka u primarnoj zoni).

STATICKI RASUTA DATOTEKA SA PROBABILISTICKOM TRANSFORMACIJOM – Kod staticki rasute datoteke se da njeni slogovi nisu sukcesivno rasporedjeni u memoriju. Staticka je jer se memorija alocira unapred. Baket je isto sto i slog i on sadrzi nekoliko slogova. Adrese smestanja u memoriji se odredjuju na osnovu transformacije kljuca. Razlikujemo deterministicku i probalistiticku transformaciju. Kod deterministicke transformacija kljuca daje jednu adresu, a obrnut proces daje jedan kljuc. Kod probalisticke transformacije transformisanjem kljuca dobijamo jednu adresu, a obrnuto proces moze da da vise kljuceva. Probalisticke transformacije su: metoda ostatka pri deljenju, metoda centralnih cifara kvadrata kljuca i metoda preklapanja.

DINAMICKA RASUTA DATOTEKA – Osnovni nedostatak statickih rasutih datoteta je da dolazi do pojave prekoracilaca, dolazi do degradacije preformasi obrade podataka i uzajamna zavisnost metode transformacije kljuca u relativnu adresu. Za datoteke koje se skoro iskljucivo obradjuju u rezumu direktne obrade i cije se sadrzaj intenzivno menja moguće je uvesti rasutu organizaciju. Dinamicka rasuta se sastoji od 2 dela:

1. Adresar – sadrzi niz pokazivaca duzine  $2^d$  gde je  $d \geq 0$  sa adresama baketa i druga polja. To je mala pomocna struktura podataka i najcesce se realizuju kao linearna struktura. Zahteva relativno mali kapacitet memorijskog prostora

2. Zona podataka sa baketima – svaki baket sadrzi zaglavlje i bar jedan slog

Adresar i zona podataka se realizuju kao dve posebne datoteke.

B STABLO – Cvor sadrzi niz elemenata koji propagiraju vrednost kljuca iz primarne zone. Svaki cvor sadrzi max.  $2r$  elementa. Svaki cvor, izuzev korena sadrzi min.  $r$  elemenata. Koren sadrzi min. 1 element. Ekstremni slucajevi popunjenosti B stabla su: Poluprazno stablo (svi cvorovi osim korena sadrže po  $r$  elemenata i koren sadrži samo 1 element) i Kompletno stablo (svi cvorovi sadrže po  $2r$  elemenata). Upis novog elementa prethodi mu neuspesno trazenje elementa. Zapocinje u korenu i uvek se završava u listu. Zapocinje na mestu zaustavljanja neuspesnog trazenja u listu, list moze biti potpuno ili delimicno popunjen. Brisanje elemenata iz B stabla prethodi mu uspesno trazenje, element se sme fizicki izbrisati jedino ako se nalazi u listu. Postoje 2 tehnike brisanja elemenata iz stabla: Tehnika pozajmljivanja elemenata od suseda i Tehnika spajanja dva susedna cvora u jedan.