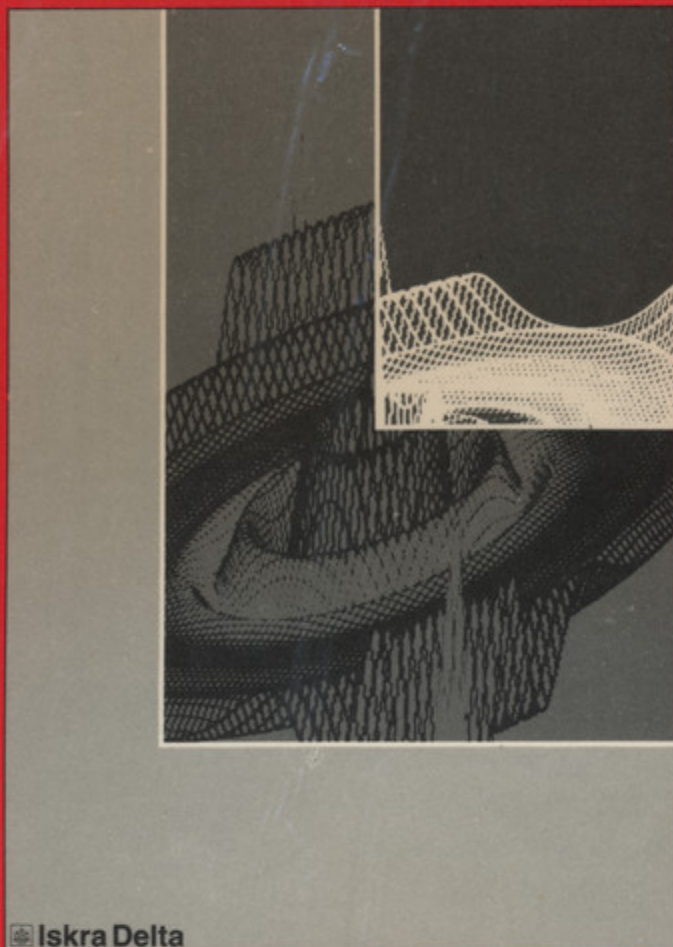


DB/VM



računalniški sistemi delta



O

PRIROČNIK ZA ORGANIZATORJE  
PRIRUČNIK ZA ORGANIZATORE  
ПРИРУЧНИК ЗА ОРГАНИЗАТОРЕ  
ПРИРАЧНИК ЗА ОРГАНИЗАТОРИ

DB/VM Verzija 1.0

PODATKOVNE STRUKTURE DB/VM

Priročnik za organizatorje

Koda 16 783 044

Preliminarna izdaja: Januar 1984

ISKPA DELTA

Operacijski sistem: DELTA/V, DELTA/M

Pred vami je preliminarna izdaja priročnika o podatkovnih strukturah pod delovnim imenom DP/VF.

Programski proizvod DP/VF je grajen po enotni ISKRA DELTA ARHITECTURI, ki med drugim obsega tudi kompleks programskih orodij. Sedaj sodijo novi proizvodi, kot so generatorji programov, ekrančnih slik in izpisov, slovarji podatkov in procedur, ter drugi proizvodi, ki jih najevljamo za leto 1984. Enovita arhitektura povezuje posamezne programske proizvode v funkcionalne enote. Še več - vse proizvode, torej tudi podatkovne strukture DP/VF - lahko uporabljamo pod DELTA/W ali DELTA/V operacijskih sistemov. To pa pomeni uporabo iste dokumentacije, iste tehnologije in istega znanja.

Verjetno se vam bodo ob branju priročnika porodile nove ideje, našli boste boljše razlage, odkrili morebitne napake in podobno. Za vsako vašo pripombo, poslano na obrazcu v prilogi, se vam zahvaljujemo.

ISKRA DELTA ODNA INTERAKCIJA - OPIS

- 3.2.2 INTERAKCIJA ZA PROGRAMIRANJE
- 3.2.2.1 PROCES "FC" - FORNAT
- 3.2.2.2 PROCES "FF" - RAZIRNEN



Prav tako je potrebno izdati priročnik o izdelavi in uporabi...

Prav tako je potrebno izdati priročnik o izdelavi in uporabi...

Prav tako je potrebno izdati priročnik o izdelavi in uporabi...

Pridržiujemo si izključno avtorsko pravico do programskega proizvoda, opisanega v tem priročniku, vključno z vso priloženo dokumentacijo.

Pridržiujemo si pravico do sprememb brez predhodnega opozorila in ne prevzamamo odgovornosti za posledice morebitne neveljavnosti informacij v priročniku s proizvodov ter za morebitne napake.

Jarčino za uporabnost programskih proizvodov na orreni, ki jo je dosegla in instalirala DO ISKRA DELTA.

ISKRA DELTA  
LJUBLJANA  
JUGOSLAVIJA



POGLAVJE 1

1.1	PROGRAMSKE KOMPONENTE DE/VM SISTEMA . . . . .	1-1
1.2	STRUKTURA SISTEMA DB/VM . . . . .	1-1
1.2.1	PODSISTEM DDP . . . . .	1-1
1.2.2	PODSISTEM DBF . . . . .	1-3
1.2.3	PODSISTEM DML . . . . .	1-3
1.2.4	DE MONITOR . . . . .	1-3
1.3	OPIS BAZE PODATKOV DE/VM . . . . .	1-4
1.3.1	FIZIČNA STRUKTURA PODATKOV NA DISKOVNIH ENOTAH . . . . .	1-8

POGLAVJE 2 PROGRAM ZA DEFINIRANJE PODATKOVNIH STRUKTUR DDP

2.1	DDP KREIRANJE . . . . .	2-3
2.1.1	KREIRANJE SCHEME . . . . .	2-4
2.1.2	KREIRANJE PODROČJA . . . . .	2-9
2.1.3	KREIRANJE PODSCHEME . . . . .	2-13
2.1.4	DEFINIRANJE FIZIČNE STRUKTURE . . . . .	2-20
2.1.4.1	KREIRANJE FIZIČNE STRUKTURE . . . . .	2-20
2.1.4.2	RAZŠIRITEV FIZIČNE STRUKTURE . . . . .	2-22
2.2	DDP MODIFICIRANJE . . . . .	2-23
2.3	DDP BRISANJE . . . . .	2-24
2.3.1	BRISANJE SCHEME . . . . .	2-25
2.3.2	BRISANJE PODROČIJ . . . . .	2-26
2.3.3	BRISANJE PODSCHEM . . . . .	2-28

POGLAVJE 3 PROGRAM ZA FORMATIRANJE FIZIČNE STRUKTURE BAZE PODATKOV DBF

3.1	OPIS DEF PROGRAMA . . . . .	3-1
3.1.1	PROGRAM DEF V OKOLJU OPERACIJSKEGA SISTEMA . . . . .	3-2
3.1.2	IZVAJANJE PROGRAMA DEF V OKOLJU OPERACIJSKEGA SISTEMA . . . . .	3-2
3.1.2.1	AKTIVIRANJE PROGRAMA DEF . . . . .	3-2
3.1.2.2	NIZ ODGOVOROV . . . . .	3-2
3.1.2.3	PROGRAMSKE KONVENCIJE . . . . .	3-3
3.2	OPIS INTERAKCIJE . . . . .	3-4
3.2.1	UVODNA INTERAKCIJA - OPIS . . . . .	3-5
3.2.2	INTERAKCIJA ZA POSAMEZNI PROCES - OPIS . . . . .	3-7
3.2.2.1	PROCES "FO" - FORMAT . . . . .	3-7
3.2.2.2	PROCES "EX" - RAZŠIRITEV . . . . .	3-9

3.2.2.3	PROCES "FB" - FORMAT IN BRISANJE POVEZAV . . . . .	3-11
3.2.3	OPIS ZAKLJUČNE INTERAKCIJE . . . . .	3-12
3.3	DEF DIAGNOSTIKA NAPAK . . . . .	3-14

POGLAVJE 4 PROGRAM ZA KUNUNICIRANJE Z DPM KOMPONENTAMI DEC

4.1	OPIS PROGRAMA DEC . . . . .	4-1
4.1.1	PROGRAM DEC V OKOLJU OPERACIJSKEGA SISTEMA . . . . .	4-2
4.1.2	IZVAJANJE PROGRAMA DEC V OKOLJU OPERACIJSKEGA SISTEMA . . . . .	4-2
4.1.3	AKTIVIRANJE PPROGRAMA DEC . . . . .	4-2
4.1.3.1	NIZ ODGCVOROV . . . . .	4-2
4.1.3.2	PROGRAMSKE KONVFNCIJE . . . . .	4-3
4.2	OPIS INTERAKCIJE . . . . .	4-4
4.2.1	UVODNA INTERAKCIJA - OPIS . . . . .	4-5
4.2.2	INTERAKCIJA ZA POSAMEZNO FUNKCIJO - OPIS . . . . .	4-6
4.2.2.1	FUNKCIJA "DBRUN" . . . . .	4-6
4.2.2.2	FUNKCIJA "DSTOP" . . . . .	4-7
4.2.2.3	FUNKCIJA "LGRUN" . . . . .	4-8
4.2.2.4	FUNKCIJA "LGSTOP" . . . . .	4-8
4.2.2.5	FUNKCIJA "LGMON" . . . . .	4-8
4.2.2.6	FUNKCIJA "DEMON" . . . . .	4-8
4.3	DEC DIAGNOSTIKA NAPAK . . . . .	4-9

DODATEK A OPTIMIZIRANJE SISTEMA DE/VM

A.1	OBLIKOVANJE V/I PODROČIJ . . . . .	A-1
A.2	OBLIKOVANJE PODROČIJ . . . . .	A-2
A.3	OBLIKOVANJE FIZIČNE STRUKTURE . . . . .	A-3
A.4	OBLIKOVANJE LOGIČNIH BLOKOV . . . . .	A-3
A.5	OBLIKOVANJE VELIKOSTI NIZOV ZA SAMOSTOJNE ZAPISE . . . . .	A-5

DODATEK B PRIKERI

B.1	POROČILO O KREIRANJU PODROČJA "PRODAJ1" . . . . .	B-1
B.2	COBOLSKA DEFINICIJA PODSHEME "PRODAJ1T1" . . . . .	B-8
B.3	PRIMER DIALOGA . . . . .	P-11
B.3.1	KREIRANJE SHEME . . . . .	P-11
B.3.2	KREIRANJE PODROČJA . . . . .	P-14
B.3.3	KREIRANJE PODSHEMF . . . . .	P-16
B.3.4	KREIRANJE FIZIČNE STRUKTURE . . . . .	P-21

INDEKS

## PREDGOVOR

Ta priročnik vsebuje informacije, ki jih potrebuje organizator podatkovnih struktur (Data Base Administrator) za uporabo sistema DB/VM na računalnikih DELTA. Zaradi obsežnosti snovi, se priročnik ukvarja izključno s programskimi rešitvami, ki so vključene v DE/VM V1.0. Zato je teoretično znanje s področja baz podatkov in datotečnih sistemov, posebno pa še praktične izkušnje, koristna in zaželjena osnova za strokovnjake, ki bodo organizirali in upravljali podatkovne strukture s pomočjo sistema DB/VM.

Priročnik je razdeljen v tri dele:

1. PRVI DEL vsebuje splošen opis komponent sistema DE/VM, njihovo medsebojno povezanost in integracijo z operacijskim sistemom.
2. DRUGI DEL opisuje definiranje podatkovnih struktur s procesiranjem podatkovnih definicij DDP.
3. TRETJI DEL vsebuje informacije o servisnih programih, ki so namenjeni organizatorju podatkovnih struktur.

Priročnik vsebuje tudi dodatke, ki s primeri ilustrirajo uporabo sistema DE/VM.



- 3.2.2.3 PROCES TISK - ...
- 3.2.3 CPIS ZAKLJUČEK ...
- 3.5 DIF DIAGNOSTIKA ...

4. PROGRAM ZA KUMUNICIRANJE ...

- 4.1 CPIS PROGRAMA DEC
- 4.1.1 PROGRAM DEC V OKOLJU OPERACIJSKE ...
- 4.1.2 IZVAJANJE PROGRAMA DEC V OKOLJU ...
- 4.1.3 AKTIVIRANJE PROGRAM ...
- 4.1.3.1 NIZ ODGOVOROV
- 4.1.3.2 PROGRAMSKE KONVENCIJE
- 4.2 CPIS INTERAKCIJE

Priloga je razdeljen v tri dele:

1. Priloga vsebuje opis komponent sistema ...
2. Priloga vsebuje opis delovanja sistema ...
3. Priloga vsebuje opis ...

Priloga je razdeljen v tri dele:

1. Priloga vsebuje opis komponent sistema ...
2. Priloga vsebuje opis delovanja sistema ...
3. Priloga vsebuje opis ...

## UVOD

Sistem DE/VM predstavlja vmesni element med fizičnimi podatki na diskih in logično sliko podatkov, kot jih uporabljajo programi. Pri tem se želi doseči naslednje cilje:

1. Neodvisnost programov od fizične strukture podatkov;
2. Vzpostavljanje logičnih povezav med različnimi podatki;
3. Eliminacija podvajanja podatkov;
4. Zaščita podatkov pred nepooblaščenimi osebam;
5. Enostaven programski dostop do podatkov;
6. Centralizirano upravljanje podatkovnih struktur;
7. Ohranjanje integritete podatkovnih struktur v primeru izpada posameznih komponent ali celotnega sistema;
8. Možnost postopnega razvoja podatkovnih struktur v skladu z razvojem in večanjem informacijskega sistema.

Sistem DE/VM uresničuje koncept sheme in podsheme. V shemi je definirana celotna struktura podatkov, podsheme pa so programsko okno do podatkov na nivoju posameznih programov ali obdelav. Neodvisnost programov od podatkov je realizirana tako, da podsheme niso deli programov ampak se polnijo v programe v času izvajanja v zato predviden prostor. Podsheme omogočajo, da ima program (ali obdelava) dostop do tistih podatkov, ki jih potrebuje, nima pa dostopa do ostalih podatkov. Prav tako spremembe na strukturah, ki

določen program ne zadevajo, nanj nimajo nikakršnega vpliva. Shema omogoča, da je za programe popolnoma transparentno, kje se podatki fizično nahajajo, saj podsheme vsebujejo izključno logične informacije. Sistem beleženja sprememb na bazi podatkov je realiziran na nivoju logičnih transakcij (ki so sestavljene iz večjega števila DML funkcij) za vsak program posebej. To omogoča vzpostavljanje ažurnega stanja sistema po nenamernih prekinitvah (Warm restart).

Sistem DB/VM definira programske in upravljalne operacije za:

1. Bazo podatkov DB/VM.
2. Sekvenčne in tekstovne datoteke.

Informacije v tem priročniku so pretežno s področja baze podatkov, saj so te strukture kompleksnejše od konvencionalnih datotek.

Baza podatkov DB/VM sodi v skupino mrežnih baz podatkov. Omogoča realizacijo mreže medsebojno povezanih zapisov brez podvajanja podatkov. Pri tem je potrebno poudariti veliko učinkovitost sistema DB/VM, ki je bila dosežena predvsem z dobro integracijo z operacijskim sistemom. Monitor baze podatkov lahko do skrajnosti izkoristi vse razpoložljive programske in aparaturne možnosti sistemov DELTA. Velikost tega monitorja je lahko od 16K do 1024K besed, odvisno od razpoložljive velikosti pomnilnika in baze podatkov.

Programski dostop do podatkov omogočajo ukazi DML, ki se dajo uporabljati v vseh standardnih prevajalnikih na sistemih DELTA. Definirane so štiri odnove operacije (GET, INSERT, REWRITE, DELETE) s katerimi se da zelo enostavno izvajati programske operacije nad zapisi.

Sistem zaščitnih gesel na nivoju sheme, področja in podsheme v veliki meri preprečuje nepredvideno pristopanje k podatkom, ki so shranjeni na fizičnih nosilcih v taki obliki, da brez poznavanja logičnih opisov niso razpoznavni. S tem je sistem zaščiten tako pred nenamernimi napakari, kot pred zlorabo.

Treba pa je poudariti, da kakovost podatkovne strukture določenega informacijskega sistema ni odvisna samo od kakovosti razpoložljive programske opreme, temveč v veliki meri tudi od strokovnjakov, ki načrtujejo, upravljajo in razvijajo to podatkovno strukturo. Pri tem je seveda zelo važno razumevanje in poznavanje programske opreme, s katero je baza podatkov realizirana.



## POGLAVJE 1

### 1.1 PROGRAMSKIE KOMPONENTE DE/VM SISTEMA

Sistem DE/VM je paket monitorjev in servisnih programov, ki omogočajo definiranje podatkovnih struktur (SHEMA, PODROČJE, PODSHEMA, FIZIČNA STRUKTURA), kreiranje in upravljanje podatkovne strukture in programski dostop do podatkov.

### 1.2 STRUKTURA SISTEMA DE/VM

Sistem DE/VM sestavljajo štirje podsistemi:

1. Podsistem za definiranje podatkovnih struktur (DDP);
2. Podsistem servisnih programov za kreiranje in upravljanje podatkovnih struktur (DEF);
3. Podsistem za realizacijo programskega dostopa do baze podatkov (DML);
4. Podsistem za izvajanje V/I operacij (DFM).

#### 1.2.1 PODSISTEM DDP

Podsistem DDP predstavlja interaktivni program, s pomočjo katerega se definirajo in upravljajo definicijske komponente baze podatkov:

- SHEMA
- PODROČJE
- PODSHEMA
- FIZIČNA STRUKTURA

#### SHEMA

Shema vsebuje logičen opis celotne baze podatkov to je definicija vseh zapisov, podatkovnih polj, ključev in povezav med zapisi. Podatkovna polja se opisujejo podobno, kot v programskem jeziku COBOL. SHEMA je identificirana s svojim imenom in zaščitena z geslom.

#### PODROČJE

Področje zajema tisti del sheme, h kateremu se pristopa istočasno. Uvedba področij omogoča določene optimizacije, ki so opisane v poglavju o optimizacijah (Dodatek A). PODROČJE je identificirano z imenom in zaščiteno z geslom.

#### PODSHEMA

Podshema opisuje zapise, ki so na voljo posameznim programom ali skupinam programov. Opisuje delne zapise, za katere je značilno, da vsebujejo definicijo nekaterih (ali vseh) podatkovnih polj, programske pravice dostopa in način pristopa (pri podrejenih ime SET-a). Ti delni zapisi se imenujejo programski zapisi.

Ker je pogosto potrebno definirati v okviru iste podsheme več kot eno obliko zapisa (zaradi različnih podatkovnih polj, povezav ali programskih pravic) se lahko definira do 16 programskih zapisov, ki se nanašajo na isti zapis iz SHEME.

Poleg tega lahko podshema vsebuje tudi specifikacije sekvenčnih datotek. Te so sestavljene iz specifikacije datoteke (standardna specifikacija); tipa datoteke (vhodna, vhodno/izhodna, izhodna, izhodna z atributom CR/LF) in maksimalne dolžine zapisa.

DE/VM predvideva naslednjo zaščito na nivoju programskih zapisov:

- GTP - dovoljeno je branje po fizični sekvenci;
- GET - dovoljeno je branje zapisa;
- INS - dovoljeno je dodajanje zapisa;
- DEL - dovoljeno je brisanje zapisa;
- RWR - dovoljeno je spreminjanje zapisa.

Podshema je identificirana z imenom in zaščitena z geslom. FIZIČNA STRUKTURA vsebuje opis velikosti podatkovnih nizov in specifikacije kontejnerskih datotek.

### 1.2.2 PODSISTEM DBF

DEF omogoča primarno kreiranje in upravljanje fizične strukture baze podatkov. Deli diska, ki se uporabljajo za bazo podatkov morajo biti v naprej formatirani. To se doseže z uporabo funkcije "FO". Lahko se formatira celotna fizična struktura ali pa posamezni nizi. Pri formatiranju nizov podrejenih zapisov je potrebno poleg tega še izbrisati povezovalne informacije v nadrejenih zapisih. To omogoča funkcija "FB". Poleg tega se da nize, ki vsebujejo podrejene zapise s funkcijo "EX" povečevati.

### 1.2.3 PODSISTEM DML

DML je izveden s pomočjo stavka "CALL" v gostujočem programskem jeziku. Ko se izvede ustrezen stavek "CALL" se prenese kontrola v poseben podprogram, ki predstavlja DML komponento programske opreme DB/VM. Komunikacija med aplikativnimi programi in sistemom DB/VM je izvedena preko parametrov in internih registrov. Podrobnosti so opisane v Programerskem priročniku za programsko opremo DB/VM. Dostop do podatkov je zaščiten, saj ga lahko izvaja samo oseba, ki pozna ustrezno geslo podsheme.

### 1.2.4 DE MONITOR

DE monitor je kompleksna programska celota in je sestavljena iz petih programskih komponent:

Program DBC starta in zaustavlja ostale komponente DE monitorja na osnovi interaktivnih ukazov.

Program DEM izvaja programske in fizične operacije na bazi podatkov. Zaradi tega se imenuje jedro DB/VM sistema. Jedro je programirano tako, da po potrebi širi svoj naslovni prostor od 16k do 1024K v odvisnosti od velikosti aktivnega področja in vrste V/I operacij programov, ki so vključeni v sistem DB/VM.

LOGGER izvaja programske in fizične operacije na sekvenčnih datotekah in vrši beleženje sprememb (logging) na bazi podatkov.



DEG vrši kontrolo nad aktivnostjo programov, ki so se s pomočjo ukaza HEL vključili v sistem DE/VM. V primeru nepredvidenih prekinitev izvede za take programe ukaz EYF in s tem zagotavlja nemoteno delovanje sistema.

TP DRIVER omogoča komunikacijo med aplikativnimi programi in komponentami DE monitorja. TP DRIVER je komunikacijsko orodje, ki je maksimalno prilagojeno specifičnim lastnostim operacijskega sistema. Omogoča, da se programske komponente sistema DE/VM premikajo po glavnem pomnilniku (shuffling) in po potrebi začasno izločijo na disk (task check-pointing).

### 1.3 OPIS BAZE PODATKOV DE/VM

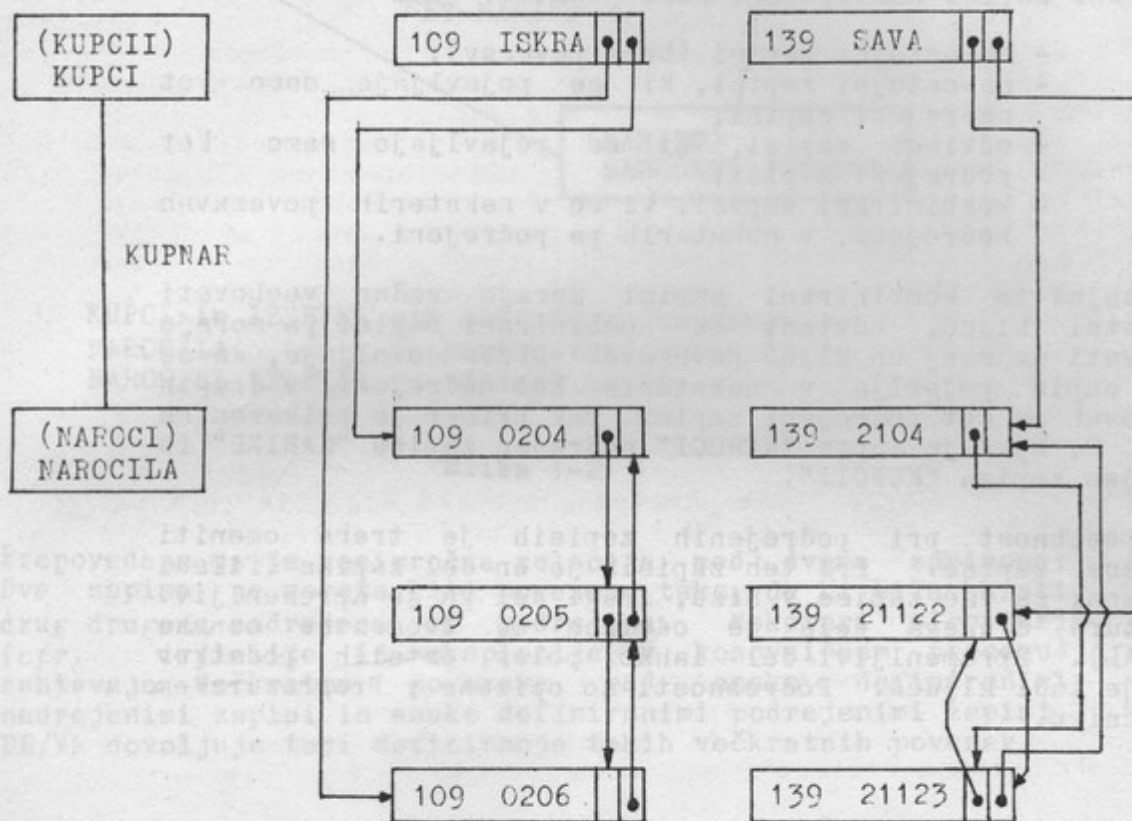
Bazo DE/VM podatkov predstavlja množica zapisov, ki so medsebojno povezani v skladu s predpisi. Oboje, zapise in njihove medsebojne povezave (relacije) predpiše upravljelec baze podatkov v SHEMI. Zapis predstavlja normirano obliko podatkov (podatkovnih polj), ki opisujejo določen objekt. Vsi enako definirani zapisi (tisti, ki imajo enake podatkovna polja) predstavljajo niz: npr. vsi zapisi, ki vsebujejo osnovne podatke o kupcih, predstavljajo niz kupcev. Posamezna podatkovna polja lahko predstavljajo različne podatke (verbalne, količinske, idikativne itd.). Določena podatkovna polja so definirana kot ključi, ki služijo za dostop do zapisa in za vzpostavljanje povezav med (različnimi) zapisi. Pogosto se kot ključ uporabljajo različne šifre, ime in priimek, poštna številke, tekoče številke dokumentov, datumi, klasifikacijske številke ipd. Glede na različne tipe zapisov DE/VM razlikuje dva tipa ključev:

1. Direktni ključ se pojavlja pri nadrejenih zapisih. Ta ključ se uporablja kot argument za direktni dostop do enega samega zapisa v množici vseh enakodefiniranih nadrejenih zapisov.
2. Ključ povezave se pojavlja pri podrejenih zapisih. Uporablja se za vzpostavljanje povezave med nadrejenimi in podrejenimi zapisi in za oblikovanje verig podrejenih zapisov. Služi tudi, kot argument za dostop na začetek ali konec verige podrejenih zapisov.

Med zapisi se lahko definirajo medsebojne relacije ali SET-i. Primer povezave med zapisom "KUPCII" in "NAROCI":

SET KUPNAR  
OWNER IS KUPCII  
MEMBER IS NAROCI USING SIFKUP.

Ta definicija pomeni, da je KUPECII nadrejeni zapis, NAROCCI pa podrejeni zapis v povezavi KUPNAR. Kriterij povezave je enakost direktnega ključa v zapisu KUPCII in ključa povezave SIFKUP v zapisu NAROCCI. Praktično to pomeni, da pripada vsak zapis NAROCCI natanko določenemu zapisu KUPCII. Tudi v resnici bi bila nesmiselna odprta naročila, ki ne bi pripadala nobenemu kupcu. Obratno seveda ne drži, prav možno je namreč, da kakšen kupec nima nobenega odprtega naročila. Množica nadrejenih zapisov v neki relaciji je vedno en sam zapis, množica podrejenih zapisov pa je lahko prazna.



Slika 1-1

Iz slike 1 se vidi, da vsebuje nadrejeni zapis kazalca na prvi in zadnji zapis v verigi podrejenih zapisov. Podrejeni zapisi so medsebojno obojestransko povezani. To omogoča programske operacije nad temi zapisi v smeri od začetka verige proti koncu in obratno.

Baza podatkov DE/VM predvideva, da mora nadrejeni zapis vsebovati direktni ključ. Na sliki 1 je šifra kupca 139 direktni ključ za zapis SAVA. Prav tako vsebujejo vsi podrejeni zapisi v tej relaciji vrednost 139 v polju ključa povezave. Povezava se tvori (ali briše) avtomatsko pri dodajanju oziroma brisanju podrejenih zapisov, na osnovi enakosti direktnega ključa v nadrejenem zapisu in ključa povezave v podrejenem zapisu. V nizu nadrejenih zapisov ni dovoljeno podvajanje direktnih ključev, medtem ko so ključi povezave v nizu podrejenih zapisov lahko podvojeni.

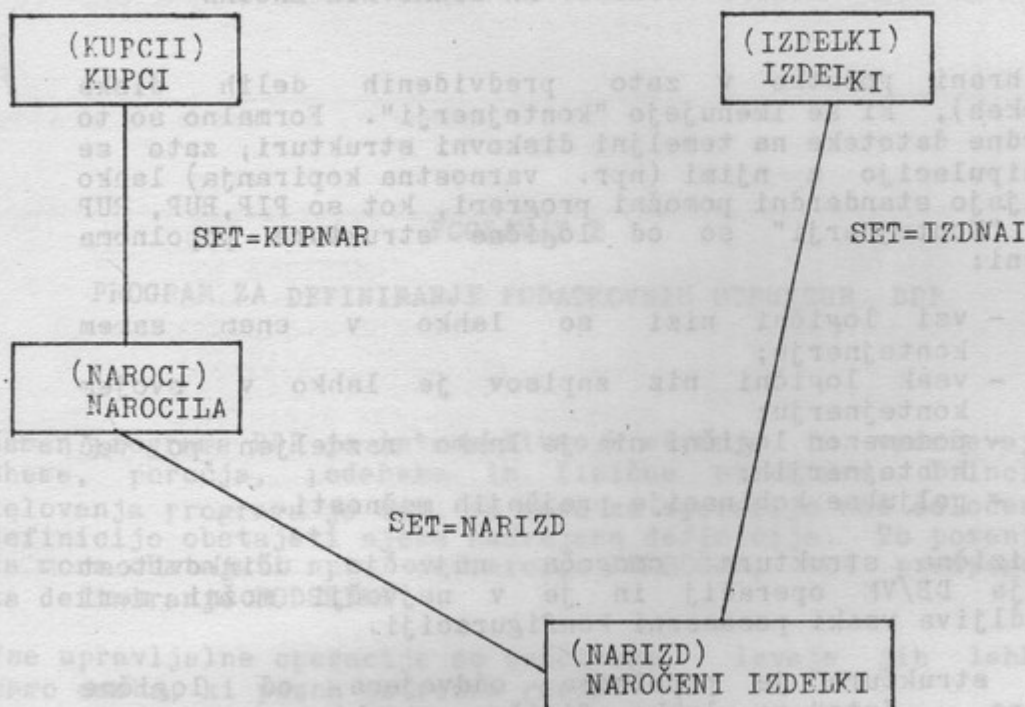
Mrežna arhitektura baze podatkov DE/VM dovoljuje da posamezni zapisi nastopajo v bazi podatkov kot:

- samostojni zapisi (brez povezav);
- samostojni zapisi, ki se pojavljajo samo kot nadrejeni zapisi;
- odvisni zapisi, ki se pojavljajo samo kot podrejeni zapisi;
- kombinirani zapisi, ki so v nekaterih povezavah nadrejeni, v nekaterih pa podrejeni.

Samostojni in kombinirani zapisi morajo vedno vsebovati direktni ključ, odvisni in kombinirani zapisi pa morajo vsebovati najmanj en ključ povezave. DE/VM dovoljuje, da se isti zapis pojavlja v nekaterih kot nadrejeni, v drugih povezavah pa kot podrejeni zapis. Tak primer je prikazan na sliki 2, kjer je zapis "NAROČI" nadrejen zapisu "NARIZD" in podrejen zapisu "KUPCII".

Kot posebnost pri podrejenih zapisih je treba omeniti označene zapise. Pri teh zapisih je en del zapisa (fiksni del) enak za vse zapise v nizu, drugi del pa je spremenljiv. Struktura drugega dela je odvisna od dvomestne oznake (REQUAL). Spremenljivi del lahko poleg ostalih podatkov vsebuje tudi ključe. Podrobnosti so opisane v Programerskem priročniku.





KUPCI in IZDELKI sta samostojna zapisa,  
 NAROČILA je kombinirani zapis in  
 NAROČENI IZDELKI je odvisni zapis.

Slika 1-2

Prepovedana pa je recipročna relacija med dvema zapisoma: Dve zapisa ne moreta biti povezana tako, da bi bila hkrati drug drugemu nadrejena in podrejena. Nekateri algoritmi (npr. implozije in eksplozije v kosovničnem procesu) zahtevajo večkratne povezave med enako definiranimi nadrejenimi zapisi in enako definiranimi podrejenimi zapisi. DE/VM dovoljuje tudi definiranje takih večkratnih povezav.

### 1.3.1 FIZIČNA STRUKTURA PODATKOV NA DISKOVNIH ENOTAH

DE/VM hrani podatke v zato predvidenih delih diska (datotekah), ki se imenujejo "kontejnerji". Formalno so to standardne datoteke na temeljni diskovni strukturi, zato se za manipulacijo z njimi (npr. varnostna kopiranja) lahko uporabljajo standardni pomožni programi, kot so PIP, BUP, RUP itd. "Kontejnerji" so od logične strukture popolnoma neodvisni:

- vsi logični nizi so lahko v enem samem kontejnerju;
- vsak logični niz zapisov je lahko v svojem kontejnerju;
- posamezen logični niz je lahko razdeljen po več kontejnerjih;
- poljubne kombinacije prejšnjih možnosti.

Taka fizična struktura omogoča največjo učinkovitost izvajanja DE/VM operacij in je v največji možni meri prilagodljiva vsaki posamezni konfiguraciji.

Fizična struktura je popolnoma oddvojena od logične strukture. Zato se lahko fizična struktura spreminja neodvisno od logične strukture. Posamezni nizi se lahko povečujejo, lahko se spreminja faktorje blokiranja, prav tako se logične nize v kontejnerjih lahko prerazporedi. Vse te spremembe se izvajajo centralizirano, zato je verjetnost napake manjša. Poleg tega tovrstne spremembe nimajo nobenega vpliva na programe, ki uporabljajo bazo podatkov. Spremembe v okolju (npr. povečano število diskovnih enot) nimajo nobene posledice za programe.

DE/VM zapisi so na disku shranjeni v obiki logičnih blokov. Ti bloki so lahko dolgi od 1 do 16 fizičnih sektorjev to je od 512 do 4096 bytov. Vsebujejo lahko enega ali več zapisov. Vse V/I operacije se fizično izvajajo v dolžini logičnega bloka.

V/I področje je prostor v programu DE/VM, kamor se prebere oziroma od koder se vpiše logične bloke. Kadar je V/I področje deljivo med večimi različnimi zapisi je njegova velikost enaka velikosti največjega pripadajočega bloka.

## POGLAVJE 2

### PROGRAM ZA DEFINIRANJE PODATKOVNIH STRUKTUR DDP

Namen programa DDP je interaktivno kreiranje in vzdrževanje sheme, poročja, podsheme in fizične strukture. Princip delovanja programa je tak, da mora za operacijo nad določeno definicijo obstajati njena nadrejena definicija. To pomeni, da mora obstajati npr. definicija PODROČJA, kot predpogoj za definiranje PODSHEME.

Vse upravljalne operacije so zaščitene. Izvaja jih lahko samo oseba, ki pozna ustrezna gesla.

Nekaj splošnih napotkov:

1. Pogoj za uporabo programa DDP je, da je uporabnik privilegiran.
2. Posamezna kreiranja je mogoče prekiniti z navodilom "END.". Prekinjeno kreiranje je mogoče nadaljevati z ukazom ŽIME SHEME ali ŽIME POROČJA ali ŽIME PODSHEME.

Program DDP aktiviramo z ukazom:

```
>DDP<CR>
```

Pokaže se prvi ekran, ki je namenjen izboru funkcije.



OPIS DDP DIALOGA:

DDP KREIRANJE	1
DDP MODIFICIRANJE	2
DDP BRISANJE	3
KONEC (CTRL/Z)	4
=====	
IZBRANI PROCES	:

2.1 DDP KREIRANJE

Pokaže se ekran za kreiranje.

DDP KREIRANJE	V1.0
=====	
KREIRANJE SHEME	1
KREIRANJE PODROČJA	2
KREIRANJE PODSHEME	3
KREIRANJE FIZIČNE STRUKTURE	4
KONEC (CTRL/Z)	5
=====	
IZBERI PROCES	:

2.1.1 KREIRANJE SCHEME

Vprašanje

Odgovor

IME SCHEME (6)

: PRODAJ

OPIS:

- PRODAJ je primer za ime sheme.
- Ime sheme je 6 alfanumeričnih znakov.
- Nenačrtan odgovor <CR> ni mogoč.

Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer za geslo sheme.
- Geslo je 6 alfanumeričnih znakov.
- Nenačrtan odgovor <CR> ni mogoč.

Vprašanje

Odgovor

IME ZAPISA V SHEMI (6)

: KUPCII

OPIS:

- KUPCII je primer.
- Ime zapisa v shemi je 6 alfanumeričnih znakov.  
Če "0" ne more biti prvi znak imena.
- 0 pomeni konec vnosov imen zapisov.
- Nenačrtan odgovor ni mogoč
- Navodilo DISPLAY je prikaz vseh vnešenih zapisov.



Vprašanje

Odgovor

TIP ZAPISA V SHEMI (S,O,R,X) : 0

OPIS:

- C je primer.
- Tip zapisa je lahko:
  - S = lastnik zapisa je sistem.
  - O = Zapis vsebuje direktni ključ (samostojni zapis).
  - R = Zapis vsebuje ključe povezave (podrejeni zapis).
  - X = Zapis vsebuje direktni ključ in ključe povezave (kombinirani zapis).
- Nenaglašen odgovor <CR> je znak "S".

Vprašanje

Odgovor

"KUPCII" OPIS DOLŽINE KONTROLNEGA POLJA : X(6)

OPIS:

- KUPCII je ime zapisa (izpiše program).
- X(6) je primer odgovora.
- Odgovor je sobelski opis PICTURE.
- Nenaglašen odgovor ni mogoč.
- Kontrolno polje je mogoče strukturirati (glej: opis dolžine podatkovnega polja).

Vprašanje

Odgovor

"KUPCII" IME PODATKOVNEGA POLJA : IMEKUP

CPIS:

- KUPCII je ime zapisa (izpiše program).
- IMEKUP je primer odgovora.
- Možni odgovori so:
  - eeeeeee je ime podatkovnega polja v zapisu xxxxxx.
    - ime podatkovnega polja sestavlja 6 alfanumeričnih znakov, prvi znak ne sme biti nič (0);
    - rezervirana imena so REQUAL, OWNKEY, SYNKEY, FILLER;
    - imena podatkovnih polj za posamezni zapis morajo biti unikatna;
    - največje število podatkovnih polj za zapis je 256.
  - nneeeeeee je opis nivoja posameznega podatkovnega polja. Nivo "10" je struktura predhodnega podatkovnega polja brez nivoja, nivo "15" je struktura podatkovnega polja z nivojem "10" itd.
    - vrednost "nn" je 10, 15, 20, 25 in 30;
    - nivo "05" je rezerviran za elemente vnešene brez nivoja.
  - REQUAL=kk je oznake samo za zapise tipa "M in X", kar omogoča večkratno strukturiranje enega podatkovnega polja.
    - oznako sestavljata dva alfanumerična znaka;
    - oznake morajo biti unikatne;
    - za tem navodilom lahko vnašamo samo podatkovna polja v nivojih "nn";
    - navodilo se lahko uporabi večkrat;
    - s tem navodilom se lahko vnašajo samo podatkovna polja z nivojem.
- E pomeni konec vnosov podatkovnih polj.
- Nenaglašen odgovor ni mogoč.
- Navodilo DISPLAY je prikaz vseh vnešenih polj.

CPIS:

- KUPCII je primer.
- Ime zapisa v zbirki je 6 alfanumeričnih znakov. Prvi znak ne sme biti nič (0).
- E pomeni konec vnosov imen zapisov.
- Nenaglašen odgovor ni mogoč.
- Navodilo DISPLAY je prikaz vseh vnešenih zapisov.

Vprašanje

Odgovor

OPIS DOLŽINE PODATKOVNEGA POLJA : X(50)

OPIS:

- X(50) je primer opisa dolžine podatkovnega polja.

Drugi primeri:

AAAAAAAAA ali A(9)  
XXXXX ali X(5)  
999999 ali 9(6)  
9(4) COMP ali 9999 COMP  
S9(5)V99 COMP-3

- Vsa podatkovna polja se opisujejo s cobolskim opisom PICTURE.

- A pomeni alfa podatkovna polja.
- X pomeni alfanumerična podatkovna polja.
- 9 pomeni numerična podatkovna polja.
- S pomeni naglašeni predznak za numerična podatkovna polja.

(Obvezno za COMP-3 podatkovna polja)

- V pomeni decimalno vejico v numeričnih podatkovnih poljih.
- COMP numerična podatkovna polja v binarnem formatu (programsko sinhronizirana na mejo besede). S pravilno razvrstitvijo podatkovnih polj se da izogniti vrivanju praznega mesta zaradi poravnave pomnilniških rezervacij.
- Tabela rezervacij pomnilniških lokacij za COMP navodilo.
 

S9	do	S9(4)	1 beseda (2 byte)
S9(5)	do	S9(9)	2 besedi (4 byti)
S9(10)	do	S9(18)	4 besede (8 bytov)
- COMP-3 pakirani decimalni format. Dve številki sta shranjeni v enem bytu, predznak pa v skrajnem desnem bytu pakiranega polja. (Programska poravnava na mejo besede).
- Kenaglašen odgovor ni mogoč.

OPIS:

- PICTURE je primer za ime področja.
- Ime področja je ime dolgo sedem znakov. Prvih šest alfanumeričnih znakov je ime skema, sedmi alfanumerični znak pa označuje področje v skema.
- Kenaglašen odgovor ni mogoč.

NA EKLANU SE POJAVI BPOHOČILO "...KREIRAN DELOVNE PIZE...".



Vprašanje

Odgovor

IME SETA OWNER "KUPCII" MEMBER "NAROCI" : KUPNAR

OPIS:

- KUPCII je ime zapisa tipa "Owner".
- NAROCI je ime zapisa tipa "Member".
- KUPNAR je primer imena "Set-a".
- Ime seta je šest (6) alfanumeričnih znakov.
- B pomeni, da ni "Set-a" med zapisoma "Owner in Member".  
Dva zapisa povezuje lahko več "SET-ov".
- Navodilo DISPLAY pomeni prikaz vseh setov.
- Nenaglašen odgovor ni mogoč.

Vprašanje

Odgovor

"NAROCI" KLJUC PRIPADNOSTI ZA SET : SIFKUP

OPIS:

- NAROCI je ime zapisa "Member";
- SIFKUP je primer podatkovnega polja v zapisu "Member";
- dolžina vnesenega podatkovnega polja iz "Member" zapisa mora biti enaka dolžini kontrolnega polja iz zapisa "Owner";
- nenaglašen odgovor ni mogoč;
- program poišče prvi vneseni zapis "Owner" in prvi zapis "Member".
- sekvenca se nadaljuje z naslednjim zapisom "Member".
- po vprašanjih za prvi zapis "Owner" in vse zapise "Member" se ista sekvenca nadaljuje z naslednjim "Owner" zapisom.
- <CR> vrne sekvenco vprašanj in vnos imena seta. Predhodno vnešeno ime seta je neveljavno.

## 2.1.2 KREIRANJE PODROČJA

Vprašanje

Odgovor

IME SCHEME (6)

: PRODAJ

OPIS:

- PRODAJ je primer imena sheme kateri pripada področje (area).
- Ime sheme je šest alfanumeričnih znakov.
- Nenaglašen odgovor ni mogoč.

Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer gesla.
- Geslo je šest alfanumeričnih znakov.
- Nenaglašen odgovor ni možen.

Vprašanje

Odgovor

IME PODROČJA (7)

: PRODAJ1

OPIS:

- PRODAJ1 je primer za ime področja.
- Ime področja je ime dolgo sedem znakov. Prvih šest alfanumeričnih znakov je ime sheme, sedmi alfanumerični znak pa označuje področje v shemi.
- Nenaglašen odgovor ni mogoč.

NA EKRANU SE POJAVI SPOROČILO "...KREIRAM DELOVNE NIZE...".

Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer gesla področja.
- Geslo področja je šest alfanumeričnih znakov.
- Nenaglašen odgovor ni možoč.

NA EKTRAN SE IZPIŠEJO VSI ZAPISI SCHEME.

Vprašanje

Odgovor

IME ZAPISA V PODROČJU (6)

: KUPCII

OPIS:

- KUPCII je primer za ime zapisa.
- Ime zapisa je dolgo šest alfanumeričnih znakov.
- D označuje konec vnosov zapisov.
- Navodilo DISPLAY pomeni ponoven prikaz vseh zapisov - označeni so že izbrani.

Vprašanje

Odgovor

ŠTEVILO AKTIVNIH PROGRAMOV

: 11

OPIS:

- 11 je primer odgovora.
- Odgovor je maksimalno število programov aktivnih istočasno na PODROČJU.
- vrednost "tt" je od "2" do "52".
- <CR> je nenaglašen odgovor čigar vrednost je "10".



Vprašanje \_\_\_\_\_ Odgovor

ČAS DOSTOPA DO ZAKLENJENIH ZAPISOV : 62

OPIS:

- 62 je primer.
- Odgovor je časovni interval v sekundah, preden je zapis h kateremu pristopimo razpoložljiv za drugi program.
- vrednost odgovora je med "0" in "999" sekundami.
- <CR> je nenaglašen odgovor čigar vrednost je "60".

Vprašanje \_\_\_\_\_ Odgovor

ŠTEVILO ZAKLENJENIH ZAPISOV : 6

OPIS:

- 6 je primer.
- Odgovor je število vrednosti med "0" in "999".
- <CR> nenaglašen odgovor je 10.

Vprašanje \_\_\_\_\_ Odgovor

"KUPCII" IME V/I PODROČJA (6) : IOKUPC

OPIS:

- KUPCII je ime zapisa (izpiše program).
- IOKUPC je primer za ime V/I področja.
- Ime V/I področja je alfanumerično in dolgo šest (6) znakov.
- ime V/I področij morajo biti različna za zapise "Owner" in zapise "Member".
- nenaglašen odgovor <CR> pomeni, da je ime V/I področja enako imenu zapisa.



### 2.1.3 KREIRANJE PODSHEME

Vprašanje

Odgovor

IME SHEME (6)

: PRODAJ

OPIS:

- PRODAJ je primer za ime sheme.
- Ime sheme je dolgo šest (6) alfanumeričnih znakov.
- Nenaglašen odgovor <CR> je nemogoč.

NA EKRANU SE IZPIŠE SPOROČILO "...KREIRAM DELOVNE NIZE ..."

Vprašanje

Odgovor

GESLO (6)

:

OPIS:

- Odgovor je šest (6) alfanumeričnih znakov, ki pa se na ekranu ne prikaže.
- Nenaglašen odgovor <CR> ni mogoč.

NA EKRANU SE IZPIŠE SPOROČILO "...KREIRAM DELOVNE NIZE...".

Vprašanje

Odgovor

IME PODROČJA (7)

: PRODAJ1

OPIS:

- PRODAJ1 je primer.
- Ime področja je dolgo sedem alfanumeričnih znakov.
- Nenaglašen odgovor ni mogoč.



Vprašanje

Odgovor

GESLO (6)

OPIS:

- Odgovor je šest (6) alfanumeričnih znakov, ki pa se na ekranu ne prikaže.
- Nenaglašen odgovor <CR> ni mogoč.

NA EKRANU SE IZPIŠE SPOROČILO "...KREIRAM DELOVNE NIZE..."

Vprašanje

Odgovor

IME PODSHEME (9)

: PRODAJ1TT

OPIS:

- PRODAJ1TT je primer.
- Ime podsheme je dolgo devet (9) alfanumeričnih znakov.
  - Prvih šest znakov je ime sheme.
  - Sedmi znak enolično označuje področje.
  - Osmi in deveti znak enolično označujeta podshemo.
- Nenaglašen odgovor je nemogoč.

Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer.
- Odgovor je geslo podsheme, dolgo šest (6) alfanumeričnih znakov.
- Nenaglašen odgovor je nemogoč.

Vprašanje

Odgovor

IME PROJEKTA (8)

: TESTNIO1

OPIS:

- TESTNIO1 je primer.
- Ime projekta je lahko največ osem (8) alfanumeričnih znakov.
- Nenaglašen odgovor ni mogoč.

Vprašanje

Odgovor

NAČIN PRISTOPA V PODROČJE (R,U)

: U

OPIS:

- U je primer.
- Predvidena sta dva načina pristopa:
  - R (READONLY) pomeni da se podatke lahko samo bere.
  - U (UPDATE) pomeni da se podatke lahko dodaja, ažurira, briše in bere.
- Nenaglašen odgovor ni mogoč.

NA EKTRAN SE IZPIŠEJO VSA IMENA ZAPISOV PODROČJA.

Vprašanje

Odgovor

IME ZAPISA V PODSHEMI (6)

: KUPCII

OPIS:

- KUPCII je primer za ime zapisa.
- D pomeni konec vnosa zapisov.
- Z navodilom DISPLAY se na ekran ponovno izpišejo vsa imena zapisov - posebej označeni so že izbrani
- Nenaglašen odgovor ni mogoč.

Vprašanje

Odgovor

NAČIN PRISTOPA K IZBRANEMU ZAPISU (S,P) : S

OPIS:

- S je primer odgovora.
- Odgovor je lahko:
  - S (SHARED) za vse podsheme.
  - P (PRIVILEGED) za podshemo.
- Nenagljen odgovor <CR> pomeni znak "S".

Vprašanje

Odgovor

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"KUPCII001" NAČIN DELA S POLJI : 1

OPIS:

- 1 je primer odgovora.
- Odgovor je izbran način dela. Lahko se navede tudi več načinov hkrati. Na primer:123 ali 1 2 3 ali 1,2,3 ali 231 ...
- KUPCII001 je programsko ime zapisa sestavljeno iz naslednjih delov:
  - prvih šest mest je ime zapisa;
  - preostala tri mesta, so tekoča številka programskega zapisa z začetkom ena (001).
- D konec izbora programskih zapisov za pripadajoči zapis.

NA EKTRAN SE IZPIŠEJO VSI SETI PODREJENEGA ZAPISA IN IZBRANIH NADREJENIH V PODSHEMI.



Vprašanje

Odgovor

-----  
IZBERI SET ZA PODATKOVNA POLJA (6) : KUPNAR  
-----

OPIS:

- KUPNAR je primer.
- To vprašanje se pojavi takrat, ko se uporabljajo podrejeni zapisi oziroma nizi.
- <CR> samo pri kombiniranih zapisih ni izbranega seta.

NA EKRAK SE IZPIŠE SPOROČILO: LISTI JE DODANO SIFKUP POLJE ZA KLJUČ SETA. TO SPOROČILO POVE, DA JE DDP SAM DODAL POLJE KLJUČA ZA NADREJENI ZAPIS.

-----  
NA EKRAK SE IZPIŠE PRIPADAJOČA LISTA PODATKOVNIH POLJ.  
-----

Vprašanje

Odgovor

-----  
"KUPCIOO1" IZBERI PODATKOVNA POLJA : ALL.  
-----

OPIS:

- ALL. je primer odgovora.
- KUPCII je ime niza (izpiše program).
- 001 je zaporedna številka programskega zapisa (izpiše program).
- Požna odgovora sta :
  - šest alfanumeričnih znakov.
  - ALL. je izbor vseh polj za pripadajoči zapis.
- B pomeni konec izbora elementov.
- Z navodilom DISPLAY se na ekran izpišejo vsi elementi, označeni so že izbrani.
- Nenaglašen odgovor <CR> pomeni: izbrano je prvo podatkovno polje ali naslednje od predhodnega izbranega.

-----  
NA EKRAK SE IZPIŠE SPOROČILO "... KREIRAN DELOVNE NIZE...".  
-----

Vprašanje

Odgovor

IME SEKVENČNEGA ZAPISA PODSHEME (6) : KUPNAR

OPIS:

- KUUUPP je primer.
- Odgovor je šest (6) alfanumeričnih znakov.
- B pomeni konec vnosa.
- Nenaglašen odgovor ni mogoč.

Vprašanje

Odgovor

DOLŽINA ZAPISA : 20

OPIS:

- 20 je primer.
- Odgovor je celo število med 14 in 510. S tem parametrom se za vrsto CF definira fiksna dolžina zapisa, za tipa CV in CR pa maksimalna dolžina.

Vprašanje

Odgovor

NAČIN DOSTOPA (IN,OU,IO) : IN

OPIS:

- IN je primer.
- Možni odgovori so:
  - IN dovoljeno je čitanje.
  - OU dovoljeno je pisanje.
  - IO dovoljeno je čitanje in pisanje.

Vprašanje

Odgovor

ZAŠČITA DOSTOPA (SN,NS)

: SN

OPIS:

- SN je primer.
- Možni odgovori so:
  - SN možen je sočasen dostop več programom.
  - NS ekskluzivni dostop.
- Nenaglašen odgovor ni predviden.

Vprašanje

Odgovor

ZAPIS IMA "CR/LF" KARAKTERISTIKO (D,N)

: D

OPIS:

- D je primer odgovora.
- Možna odgovora sta:
  - D pomeni da ima zapis "CR/LF" karakteristiko izpisa na izhodne medije.
  - N pomeni da zapis nima teh karakteristik.

Vprašanje

Odgovor

SPECIFIKACIJA NIZA

: DRAO:KUPNAR.SEQ

OPIS:

- DRAO:KUPNAR.SEQ je primer standardne specifikacije.
- D je terminator.
- <CR> ni predvidena nenaglašena vrednost.

NA EKTRAN SE IZPIŠE SPOROČILO "...KREIRAM DELOVNE NIZE..."



2.1.4 DEFINIRANJE FIZIČNE STRUKTURE

---

DE/VM - KREIRANJE	V1.0
=====	=====
KREIRANJE FIZIČNE STRUKTURE	1
RAZŠIRITEV FIZIČNE STRUKTURE	2
KONEC	3
=====	=====
IZBERI PROCES	:

---

2.1.4.1 KREIRANJE FIZIČNE STRUKTURE -

Vprašanje Odgovor

---

IME SHEME (6) : PRODAJ

---

OPIS:

- PRODAJ je primer.
  - Odgovor je ime sheme dolgo šest (6) alfanumeričnih znakov.
  - Nenaglašen odgovor ni mogoč.
  - "END." ni dovoljen.
- 

Vprašanje Odgovor

---

GESLO (6) :

---

OPIS:

- Odgovor je geslo sheme dolgo šest (6) alfanumeričnih znakov, ki pa se na ekranu ne prikazuje.
- Nenaglašen odgovor ni mogoč.
- "END." ni dovoljen.

NA EKРАНУ SE IZPIŠE SPOROČILO "...KREIRAN DELOVNE NIZE..." .

Vprašanje

Odgovor

SPECIFIKACIJA KONTEJNERJA

: DRAO:PRODAJA1.CON

OPIS:

- DRAO:PRODAJA1.CON je standardna specifikacija.
- D je terminator.
- Več kot 32 kontejnerjev ni mogoče specificirati.
- Nenaglašen odgovor <CR> ni predviden.
- Z navodilom DISPLAY se na ekranu pojavi izpis vseh kontejnerjev in izbranih zapisov.
- "END." ni dovoljen.

Vprašanje

Odgovor

IZBERI ZAPISE ZA KONTEJNER (6)

: KUPCII

OPIS:

- KUPCII je primer za ime zapisa v SHEMI.
- Odgovor je šest alfanumeričnih znakov.
- Več kot 32 zapisov ni mogoče izbrati v kontejner.
- En zapis se lahko ponovi največ 16-krat.
- Z navodilom DISPLAY se na ekranu izpišejo vsi izbrani zapisi v kontejnerju.
- D pomeni konec vnosov za kontejner.
- Nenaglašen odgovor ni predviden.
- "END." ni dovoljen.

Vprašanje

Odgovor

"KUPCII" VELIKOST NIZA ZAPISOV : 10000

OPIS:

- 10000 je primer odgovora.
- KUPCII je ime zapisa (izpiše program).
- Odgovor je velikost niza zapisov izražena v številu logičnih zapisov.
- Z navodilom DISPLAY se na ekran izpišejo vsi zapisi v kontejnerju.
- Nenaglašen odgovor ni predviden.
- "END." ni dovoljen.

Vprašanje

Odgovor

"KUPCII" ŠTEVILO ZAPISOV V BLOKU : 14

OPIS:

- 14 je primer.
- KUPCII je ime zapisa (izpiše program).
- Odgovor je število zapisov v bloku (program poskuša optimizirati, če je možno).
- <CR> pomeni, da program sam izračuna število zapisov v bloku.
- Najdaljši dovoljen blok je 8K bytov.
- "END." ni dovoljen.

NA EKTRAN SE IZPIŠE SPROČILO "...KREIRAM DELOVNE NIZE...",  
KAR PREDSTAVLJA KREIRANJE FIZIČNE STRUKTURE.

#### 2.1.4.2 RAZŠIRITEV FIZIČNE STRUKTURE -

Postopek je isti kot za kreiranje fizične strukture. Če se ugotovi, da skreirani opis fizične strukture ne ustreza, ga je možno razširiti. "CTRL/Z in "END." nista dovoljena.





2.3 DDP BRISANJE

DE/VM - DDP BRISANJE	V1.0
PRISANJE SHEME	1
PRISANJE PODROČJA	2
PRISANJE PODSHEME	3
KONEC (CTRL/Z)	4
IZBERI PROCES	:

Vprašanje

Odgovor

"KUPCII" ŠTEVILO ZAPISOV V BLOKU : 14

OPIS:

- 14 je pravi.
- KUPCII je ime zapisa (izpiše program).
- Odgovor je število zapisov v bloku (program poskuša optimizirati, če je možno).
- <CR> pomeni, da program sam izračuna število zapisov v bloku.
- Najdaljši dovoljen blok je 64 bajtov.
- "ZFD" ni dovoljen.

NA EKRAHU SE IZPIŠE OPOROČILO "...KUPCII...".  
KAR PREDSTAVLJA KUPCII JE FIZIČNA STRUKTURA.

2.1.4.2 FIZIČNE STRUKTURE

Pomembno je, da se pri brisanju fizične strukture. Če se ugotovi, da fizična struktura ne ustreza, je možno razbiti. "ZFD" ni dovoljena.

2.3.1 ERISANJE SCHEME

Vprašanje

Odgovor

IME SCHEME (6)

: PRODAJ

OPIS:

- PRODAJ je primer za ime sheme.
- Ime sheme je 6 alfanumeričnih znakov.
- Nenaglašen odgovor <CR> ni mogoč.

Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer za geslo sheme.
- Geslo je 6 alfanumeričnih znakov.
- Nenaglašen odgovor <CR> ni mogoč.

NA EKTRAN SE IZPIŠE CELOTNA SCHEMA Z VSEMI PRIPADAJČIMI  
PODROČJI IN PODSCHEMAMI.

Vprašanje

Odgovor

BRISANJE SCHEME, PODROČIJ, PODSCHEM (D/N)

: D

OPIS:

- D je primer odgovora.
- Edina možna odgovora sta D (da) ali N (ne).



2.3.2 BRISANJE PODROČIJ

Vprašanje

Odgovor

IME SCHEME (6)

: PRODAJ

OPIS:

- PRODAJ je primer za ime sheme.
- Ime sheme je 6 alfanumeričnih znakov.
- Nenačlajšen odgovor <CR> ni mogoč.

Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer za geslo sheme.
- Geslo je 6 alfanumeričnih znakov.
- Nenačlajšen odgovor <CR> ni mogoč.

Vprašanje

Odgovor

IME PODROČJA (7)

: PRODAJ1

OPIS:

- PRODAJ1 je primer za ime področja.
- Ime področja je ime dolgo sedem znakov. Prvih šest alfanumeričnih znakov je ime sheme, sedmimi alfanumerični znak pa označuje področje v shemi.
- Nenačlajšen odgovor ni mogoč.

Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer gesla področja.
- Geslo področja je šest alfanumeričnih znakov.
- Nenaglašen odgovor ni mogoč.

NA EKTRAN SE IZPIŠE CELOTNO PODROČJE S PRIPADAJOČIMI  
PODSHEMAMI.

Vprašanje

Odgovor

BRISANJE PODROČJA, PODSHEM (D/N)

: D

OPIS:

- D Je primer odgovora.
- Edina možna odgovora sta D (da) ali N (ne).

2.3.3 BRISANJE PODSHEM

Vprašanje

Odgovor

IME SHEME (6)

: PRODAJ

OPIS:

- PRODAJ je primer za ime sheme.
- Ime sheme je 6 alfanumeričnih znakov.
- Nenaglašen odgovor <CR> ni mogoč.

Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer za geslo sheme.
- Geslo je 6 alfanumeričnih znakov.
- Nenaglašen odgovor <CR> ni mogoč.

Vprašanje

Odgovor

IME PODROČJA (7)

: PRODAJ1

OPIS:

- PRODAJ1 je primer za ime področja.
- Ime področja je ime dolgo sedem znakov. Prvih šest alfanumeričnih znakov je ime sheme, sedmi alfanumerični znak pa označuje področje v shemi.
- Nenaglašen odgovor ni mogoč.



Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer gesla področja.
- Geslo področja je šest alfanumeričnih znakov.
- Nenaglašen odgovor ni mogoč.

Vprašanje

Odgovor

IME PODSHEME (9)

: PRODAJ1T1

OPIS:

- PRODAJ1T1 je primer.
- Ime podsheme je dolgo devet (9) alfanumeričnih znakov in prvi znak ne sme biti "0".
  - Prvih šest znakov je ime sheme.
  - Sedmi znak enolično označuje področje.
  - Osmi in deveti znak enolično označujeta podshemo.
- Nenaglašen odgovor ni mogoč.

Vprašanje

Odgovor

GESLO (6)

: PRODAJ

OPIS:

- PRODAJ je primer.
- Odgovor je geslo podsheme, dolgo šest (6) alfanumeričnih znakov.
- Nenaglašen odgovor je nemogoč.

NA EKTRAN SE IZPIŠE PODSHEMA.

Vprašanje

Odgovor

BRISANJE PODSHEME (D/N)

: D

OPIS:

- D je primer odgovora.
- Edina možna odgovora sta D (da) ali N (ne).

POGLAVJE 3

PROGRAM ZA FORMATIRANJE FIZIČNE STRUKTURE BAZE PODATKOV DEF

3.1 OPIS DEF PROGRAMA

Zaradi razumljivosti nadaljnega besedila je najprej potrebno opisati lastnosti posameznih tipov zapisov. DB/VM loči tri (3) tipe zapise, ki se razlikujejo po lastnostih:

1. SAMOSTOJNI TIP ZAPISA - niz, ki ima en sam direktni ključ.
2. ODVISNI TIP ZAPISA - niz, ki ima enega ali več ključev povezave.
3. KOMBINIRANI TIP ZAPISA - niz, ki ima en sam direktni ključ, vendar lahko vsebuje tudi enega ali več ključev povezave.

Program DEF vsebuje niz vprašanj s pomočjo katerih uporabnik specificira parametre, ki so potrebni za definiranje procesa. Izvajajo se algoritmi, ki preverjajo sintaktično in logično pravilnost parametrov ter vsklajenost uporabniških zahtev z okoljem. Na zahtevo se kreira niz z odgovori na posamezna vprašanja, ki se lahko uporabi za ponovno izvajanje enakega procesa.

DEF izvaja tri vrste osnovnih procesov:

- "FO" ..... formatiranje logičnih nizov - vse tri vrste.
- "EX" ..... razširitev - ODVISNI TIP ZAPISA.
- "BL" ..... brisanje povezav - SAMOSTOJNI TIP ZAPISA.

V nadaljnjem tekstu so procesi pisani s šiframi. Proces "FB" je sestavljen iz procesov "FO" in "BL".



### 3.1.1 PROGRAM DBF V OKOLJU OPERACIJSKEGA SISTEMA

Vsi procesi, ki jih izvaja DBF, so privilegirani, zato jih lahko izvaja samo upravljalec podatkov pod pogojem, da pozna privilegirano geslo za sistem in gesla vseh področij nad katerimi želi izvajati operacije. Zaradi narave operacij, ki se izvajajo je obvezno, da so vsi ostali uporabniki neaktivni na prizadetih področjih in na podatkih samih.

### 3.1.2 IZVAJANJE PROGRAMA DBF V OKOLJU OPERACIJSKEGA SISTEMA

Programski DBF paket lahko izvaja samo uporabnik, ki pozna privilegirano geslo.

#### 3.1.2.1 AKTIVIRANJE PROGRAMA DBF

Operacije, ki so potrebne za aktiviranje programa DBF:

- a) Vključitev s privilegiranim geslom.
- b) Aktiviranje programa DBF:

>DEF<CR>

ali

>DEF žnizodg<CR>

kjer "nizodg" pomeni specifikacijo niza odgovorov.

#### 3.1.2.2 NIZ ODGOVOROV

Niz odgovorov vsebuje uporabnikove odgovore na posamezna vprašanja programa DBF. Vsi odgovori se ponovno preverjajo zaradi možnosti spremembe v okolju. Če v okolju ni prišlo do nobene spremembe, je verifikacija odgovorov uspešna, sicer pa program DBF ugotovi napako. V tem primeru javlje ustrezno diagnozo in preide v interaktivni režim. Na osnovi diagnoze se uporabnik odloča o nadaljevanju oziroma prekinitvi programa.

### 3.1.2.3 PROGRAMSKE KONVENCIJE -

Kot v vseh podobnih primerih so tudi v programu DBF uporabljene določene konvencije:

- a) Pri vprašanjih kjer je odgovor samo "D" ali "N" je nenaglašen odgovor vedno "N".
- b) Vse specifikacije nizov so standardne specifikacije.
- c) Oznake RMS statusov so standardne.
- d) Odgovor <CTRL Z> na katerokoli vprašanje pomeni takojšen zaključek programa DEF z brisanjem niza odgovorov.
- e) Nenaglašen odgovor je <CR>.

Vprašanje

Odgovor

DEF PODROČJA (?)

PRODAJI

CPIS:

- PRODAJI je prazen odgovor.
- Odgovor je dolg sedem alfanumeričnih znakov.
- Te vnosi mora vsebovati isto področje.
- Ni nenaglašenege odgovora na to vprašanje.

### 3.2 OPIS INTERAKCIJE

Interakcija je sestavljena iz treh delov :

- UVODNA INTERAKCIJA
- INTERAKCIJA ZA POSAMEZNI PROCES
- ZAKLJUČNA INTERAKCIJA

V nadaljnjem tekstu so detaljni opisi vsakega dela posebej.

#### 3.1.2.1 AKTIVIRANJE PROGRAMA DBF

Operacije, ki so potrebne za aktiviranje programa DBF:

- a) Vključitev s privilegiranim geslom.
- b) Aktiviranje programa DBF:

DBF<CR>

CR

DBF<CR>

DBF<CR>      nize identifikacije nize

#### 3.1.2.2 NIZ ODGOVOROV

Niz odgovorov vsebuje uporabnikove odgovore na posamezna vprašanja programa DBF. Vsi odgovori so podrobno preverjeni zaradi možnosti napake v obliki napake v obliki ni prišlo do nobene napake, je v primeru uspešnih odgovorov uspešna, sicer pa program DBF ugotovi napako. V tem primeru javlja ustrezno diagnostiko in preide v interaktivni režim. Na osnovi diagnostike se uporabnik odloči o nadaljevanju oziroma prekinitvi programa.

### 3.2.1 UVODNA INTERAKCIJA - OPIS

V interaktivnem delu /režimu/ uporabnik odgovarja na vprašanja v zvezi s konvencijo specifikacije nizov in izvorno definicijo podatkovne strukture.

Vprašanje \_\_\_\_\_ Odgovor \_\_\_\_\_

VNESI SPEC. ZA NIZ Z ODGOVORI : PRODAJ1FO

OPIS:

- PRODAJ1FO je primer odgovora.
- Odgovor je dolg do trideset alfanumeričnih znakov.
- Tu se specificira ime niza v katerem bodo shranjeni odgovori za bodočo uporabo programa DEF.
- Pravilen odgovor je standardna specifikacija niza kot je definirana v uporabniškem priročniku za operacijski sistem, vendar brez verzije niza.
- Nenaglašen odgovor <CR> pomeni, da se niz z odgovori ne kreira.
- V primeru napačnega vnosa, izpiše program vrsto napak, odgovor pa se mora vnesti ponovno.

Vprašanje \_\_\_\_\_ Odgovor \_\_\_\_\_

IME PODROČJA (7) : PRODAJ1

OPIS:

- PRODAJ1 je primer odgovora.
- Odgovor je dolg sedem alfanumeričnih znakov.
- Ta vnos mora vsebovati ime področja.
- Ni nenaglašenege odgovora na to vprašanje.



Vprašanje

Odgovor

GESLO PODROČJA

OPIS:

- Odgovor je PRODAJ, ki pa se na ekranu ne pokaže.
- Odgovor je šest alfanumeričnih znakov.
- Ta vnos specificira geslo za obravnavano področje.
- V primeru nepravilnega odgovora, se operacija ne more nadaljevati.
- Nenaglašen odgovor je veljaven samo v primeru, če področje ni zaščiteno z geslom.

Vprašanje

Odgovor

PROCES (FO, EX, FE) ali "D"

: FO

OPIS:

- FO je primer odgovora.
- Odgovor je dolg dva alfa znaka v primeru procesa ali "D".
- Šifra procesa, ki ga želimo izvajati.
  - FO ..... Format
  - FB ..... Format in Blank
  - EX ..... EXTend - povečanje odvisnega niza.
  - D ..... Konec markiranja in začetek izvajanja.
- V primeru napačnega vnosa, izpiše program vrsto napake, odgovor pa se mora vnesti ponovno.
- Nenaglašenega odgovora ni.
- Od tu naprej vnaša uporabnik odgovore na vprašanja, ki so odvisna od izbranega procesa.

### 3.2.2 INTERAKCIJA ZA POSAMEZNI PROCES - OPIS

#### 3.2.2.1 PROCES "FO" - FORMAT -

Vzroki za formatiranje posameznih tipov zapisov ali celotnega področja so lahko različni.

1. Formiranje novega področja ali dodajanje novega tipa zapisa v obstoječe področje.
2. Spreminjanje posameznih tipov zapisov ( npr. dodajanje novih polj).
3. Spreminjanje povezovalnih poti med strukturami.
4. Slabe performanse pri uporabi ( npr. veliko število sinonimov).
5. Povečanje prostora za posamezne tipe zapisov.

Vsaka od teh sprememb ima za posledico spremembo v definiciji področja ali formiranje nove definicije. Pri spremembi definicije področja je potrebno izvesti kopiranje spremenjenih tipov zapisov preden se izvede formatiranje. V primeru, da fizični nizi že obstajajo se pri procesih "FO" in "FE" ne kreira nova verzija, pač pa se reformatira že obstoječe nize.

Vsa navodila in opozitve so opremljene s programi DDP.

OPIS DIALOGA ZA PROCES "FO":

Vprašanje

Odgovor

LISTA TIPOV ZAPISOV ALI ALL.

: ALL.

CPIS:

- ALL. je primer odgovora.
- Odgovor je lista logičnih imen nizov, ločenih z vejico.
- Ta vnos specificira listo logičnih imen tipov zapisov, ki morajo biti definirana v definiciji področja.
- Ni nenaglašene odgovora.
- Če je katerokoli logično ime v listi napačno se izpiše opozorilo z navedbo napačnega imena, ki se seveda ne upošteva. Vsa pravilna imena se upoštevajo. Odgovor ".ALL" pomeni formatiranje celotnega PODROČJA.
- Če je navedena lista logičnih imen tipov zapisov, jih program označi, ter ponovi vprašanje "PROCES (FO,EX,FB) ali "D" :".
- Če v vrstici ni dovolj prostora za celotno listo, potem lahko na ponovljeno vprašanje "PROCES (FO,EX,FB) ali "D" : " ponovno vnesemo željeni proces in vtiskamo ostanek liste.
- Če je odgovor "ALL." se vprašanje "PROCES (FO,EX,FB) ali "D" : " ne ponovi.



### 3.2.2.2 PROCES "EX" - RAZŠIRITEV -

Klasični postopek za razširitev NIZA ZAPISOV je kompliciran in dolgotrajen. Sestavljen je iz naslednjih korakov:

1. Prepis niza zapisov v sekvenčno obliko.
2. Popravek področja.
3. Formatiranje niza zapisov.
4. Eventuelno brisanje povezav.
5. Ponovno polnjenje niza zapisov.

Opisani postopek je kompliciran in pri velikih nizih zapisov tudi zelo dolgotrajen. Rešitev za enostavno in hitro razširitev odvisnih nizov zapisov je proces "EX" programa DBF. Proces "EX" je zaenkrat mogoč samo pri odvisnih zapisih, ker po potreba za razširitvi pri teh zapisih navečkrat pojavi. Pri tipih zapisov, kjer je definiran direktni ključ brez duplikacije, se ponavadi lahko vnaprej predvidi število možnih zapisov, zato zaenkrat za razširitev le-teh ostane prej opisani postopek.

Postopek "EX" poteka v naslednjih korakih:

1. Modifikacija področja.+
2. Izvajanje procesa "EX" s programom DBF.

---

+ Vsa navodila in omejitve so opisane pri programu DDP.



DE/VM - PRIRIČNIK ZA ORGANIZATORJE  
OPIS INTERAKCIJE

DIALOG ZA PROCES "EX":

Vprašanje \_\_\_\_\_ Odgovor

LISTA NIZOV TIPA 02 : NARIZD

OPIS:

- NARIZD je primer odgovora.
- Odgovor je lista logičnih imen odvisnih zapisov, ločenih z vejico.
- Ta vnos specificira listo logičnih imen odvisnih zapisov, ki morajo biti definirana v definiciji področja.
- Če je katerokoli logično ime v listi napačno, ali je specificirano logično ime samostojnega zapisa ali kombiniranega zapisa, se izpiše opozorilo z navedbo logičnega imena, ki se seveda ne upošteva.
- V primeru, da je specificirano logično ime odvisnega tipa zapisa za katerega ni spremenjeno število zapisov v področju DEF, to ugotovi, vendar ne izpiše opozorila. Proces "EX" se ne izvaja za tak tip zapisa.
- DEF označi pravilna logična imena tipov zapisov in ponovi se vprašanje "PROCES (FO,EX,FB) ali "D" :".
- Ni nenaglačenega odgovora.

3.2.2.3 PROCES "FB" - FORMAT IN BRISANJE POVEZAV -

Mnogokrat se pojavi potreba za reformatiranjem nizov odvisnih ali kombiniranih zapisov. Zaradi skladnosti podatkov je v tem primeru potrebno brisati povezave v zapisih lastnikov. V primeru, da povezave v zapisih lastnikov, niso izbrisane lahko pride do različnih napak, ki ponavadi, pomenijo da bo potrebno izvesti postopek obnove baze podatkov. Zato je bil izdelan postopek "FB", katerega značilnost je v tem, da na podlagi liste tipov zapisov za formatiranje avtomatsko briše tudi vse povezave v zapisih lastnikov, ki niso označeni za format. Na ta način se poenostavi proces reorganizacije odvisnih ali kombiniranih zapisov. Koraki potrebni za reorganizacijo nizov odvisnih ali kombiniranih zapisov:

1. Prepis niza odvisnih ali kombiniranih zapisov v sekvenčno obliko.
2. DBF-proces "FB" za niz odvisnih ali kombiniranih zapisov.
3. Polnjenje odvisnega ali kombiniranega niza zapisov.

DIALOG ZA PROCES "FB":

Dialog je popolnoma enak, kot pri procesu "FORMAT".

### 3.2.3 OPIS ZAKLJUČNE INTERAKCIJE

V ZAKLJUČNI INTERAKCIJI DEF NAJPREJ IZPIŠE VSE NIZE ZAPISOV, KI SO OZNAČENI IN ZRAVEN OZNAČENE PROCESSE V OBLIKI :

- \*\*\*\* DBF - OZNAČENI PROCESI
- \*\*\*\* KUPCII-FO, IZDLKI-FO, NAROČI-FO,
- \*\*\*\* NARIZD-FO

Če pride do konfliktna situacije pri označevanju procesov se uporabi naslednja hierarhija procesov : "FO" - "EX" - "BL"

Npr.: Za niz zapisov SAMOO1 želimo označiti procese "FO" , "BL". V tem primeru se avtomatsko označi najvišji proces, to je "FO".

Vprašanje

Odgovor

-----  
\*\*\*\*Če NIZ obstaja velja REFORMAT za "FO" in "FB"

ŽELIŠ IZVAJATI PROCESSE (D/N) : N  
-----

CPIS:

- N je primer odgovora.
- Možna odgovora sta "D" ali "N". Odgovor "N" pomeni da označenih procesov ne želimo izvajati. Če smo navedli specifikacijo za niz z odgovori se le-ta formira in DEF konča.
- Odgovor "D" pomeni, da želimo izvajati označene procese. Niz z odgovori se ravno tako formira, seveda če smo navedli specifikacijo.
- Nenaglašen odgovor je "N".

Če je bil odgovor "D" ali če smo DBF aktivirali indirektno, se začno izvajati označeni procesi.

DEF IZPIŠE NA ZAČETKU PROCESIRANJA VSAKEGA NIZA SPOROČILO:

\*\*\*\* DEF - P R O C E S I R A M KUPCII

Če pride do napake med procesiranjem katerega koli tipa zapisa, se izvede IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.

- 22 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 27 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 30 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 33 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 36 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 39 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 42 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 45 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 48 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 51 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 54 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 57 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 60 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 63 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 66 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 69 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 72 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 75 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 78 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 81 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 84 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 87 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.
- 90 Napaka pri IOT in zadnji tip zapisa, ki je bil izpisan ni pravilno obdelan.



### 3.3 DEF DIAGNOSTIKA NAPAK

Linija napake je sestavljena iz dveh delov :

- predrona - \*\*\*\* DEF

- ostali del sporočila

V opisu je naveden samo ostali del sporočila.

#### 02 Napaka pri OPEN niza odgovorov

Neppravilno odpiranje niza odgovorov. Napaka je indenticirana v RMS statusu.

#### 06 Niz odgovorov ne obstaja

Napaka pri odpiranju niza odgovorov. Navadno je razlog nepravilna specifikacija niza pri indirektnem procesu.

#### 08 Napačna dolžina vnosa

Neppravilno število vnečenih znakov.

#### 11 PODROČJE ni včitano

Včitavanje definicije PODROČJA ni bilo uspešno, ponavadi zaradi napake pri bodpiranju ali branju niza definicije. Napaka je indenticirana v RMS statusu. Če smo navedli

#### 17 Neppravilno geslo

Vnešeno geslo ni enako geslu iz definicije PODROČJA.

#### 19 Neppravilno ime PODROČJA

Ime izvorne definicije ne ustreza imenu v definiciji podatkovne strukture.

#### 20 Napaka pri procesiranju niza odgovorov

Napaka pri čitanju niza odgovorov. Napaka je identificirana v RMS statusu.

22 Napačno ime, ZAPIS že označen - xxxxxx

Ime podatkovnega niza ni definirano v definiciji podatkovne strukture ali pa je vnešeni niz že označen. To opozorilo se pojavi tudi pri nesmiselnih kombinacijah procesov.  
xxxxxx - ime TIPA ZAPISA za katerega se izpiše opozorilo.

27 Niz definicije področja je poškodovan

Napaka pri včitavanju niza definicije PODROČJA. Normalno niz ne vsebuje definicije PODROČJA.

30 Napačen odgovor

Na mestu, kjer se pričakuje odgovor "D" ali "N" je bil vtipkan drugačen odgovor.

60 Napaka pri včitavanju ukazne vrstice

Napaka je indentificirana s statusom direktive/\$DSW/.

71 Neprivilegiran uporabnik - OPROSTI.

Terminal iz katerega je startan program DBF ni privilegiran.

72 Napaka direktive \$DSW = XXXX

Napaka pri izvajanju systemske direktive /\$DSW/.

73 R M S napaka STS = XXXX ,STV = YYYY niz :ER.....

Vse RMS napake.

76 Napaka pri CREATE, OPEN, CLOSE ali EXTEND.

Ta napaka se izpiše na začetku kateregakoli procesa in to v primeru, ko se KONTEJNER ne da kreirati ali odpreti - "FO" ali "FB" , ali KONTEJNER ne obstoja - "EX","BL"

3.3 ali "IN". Lahko se pojavi tudi pri zapiranju KONTEJNERja.

77 \*\*\*\* Niz ni zvezen \*\*\*\*

To je samo opozorilo. DBF poizkuša kreirati zvezen niz. Če ni dovolj zveznega prostora, se izpiše opozorilo, DBF pa kreira nezvezen niz.

78 Proces lahko samo FO,EX ali FB

Vtipkana je bila napačna šifra procesa.

81 Premalo prostora za buffer.

DBF nima dovolj prostora za vsaj en logični blok. Ta napaka se pojavi pri zelo velikih PODROČJIH, v katerih so definirani veliki bloki za zapise.

99 Interna napaka

Interna napaka v programu DBF.



## POGLAVJE 4

### PROGRAM ZA KUMUNICIRANJE Z DEM KOMPONENTAMI DEC

#### 4.1 OPIS PROGRAMA DEC

Program DEC vsebuje niz vprašanj s pomočjo katerih uporabnik specificira parametre, ki so potrebni za definiranje procesa. Izvajajo se algoritmi, ki preverjajo sintaktično in logično pravilnost parametrov ter vsklajenost uporabniških zahtev z okoljem. Na zahtevo se kreira niz z odgovori na posamezna vprašanja, ki se lahko uporabi za ponovno izvajanje enakega procesa.

Osnovna funkcija programa DEC je kontrola okolja za DEM, LOGER in DEG ter aktiviranje in zaustavljanje le-teh. Dodane so tudi funkcije za kontrolo aktivnosti posameznih komponent.

DEC izvaja šest osnovnih procesov:

- "DBRUN" ..... priprava okolja za DEM, LOGER in DEG ter zagon le-teh.
- "DBSTOP" ..... zaustavljanje DEM-ja.
- "DBMON" ..... izpis različnih informacij o aktivnostih na DEM-ju.
- "LGRUN" ..... priprava okolja za LOGER in zagon le-tega.
- "LGSTOP" ..... zaustavljanje LOGER-ja.
- "LGMON" ..... izpis različnih informacij o aktivnostih na LOGER-ju.



#### 4.1.1 PROGRAM DEC V OKOLJU OPERACIJSKEGA SISTEMA

Vsi procesi, ki jih izvaja DEC, so privilegirani, zato jih lahko izvaja samo upravljalec podatkov pod pogojem, da pozna privilegirano geslo za sistem in gesla vseh področij nad katerimi želi izvajati operacije.

#### 4.1.2 IZVAJANJE PROGRAMA DEC V OKOLJU OPERACIJSKEGA SISTEMA

Program DEC lahko izvaja samo uporabnik, ki pozna privilegirano geslo.

#### 4.1.3 AKTIVIRANJE PROGRAMA DEC

Operacije, ki so potrebne za aktiviranje programa DEC:

a) Vključitev s privilegiranim geslom.

b) Aktiviranje programa DEC:

>DEC<CR>

ali

>DEC Žnizodg<CR>

kjer "nizodg" pomeni specifikacijo niza odgovorov.

##### 4.1.3.1 NIZ ODGOVOROV -

Niz odgovorov vsebuje uporabnikove odgovore na posamezna vprašanja programa DEC. Vsi odgovori se ponovno preverjajo zaradi možnosti spremembe v okolju. Če v okolju ni prišlo do nobene spremembe, je verifikacija odgovorov uspešna, sicer pa program DEC ugotovi napako. V tem primeru javlja ustrezno diagnozo in preide v interaktivni režim. Na osnovi diagnoze se uporabnik odloča o nadaljevanju oziroma prekinitvi programa.

#### 4.1.3.2 PROGRAMSKE KONVENCIJE -

Kot v vseh podobnih primerih so tudi v programu DEC uporabljene določene konvencije:

- a) Pri vprašanjih kjer je odgovor samo "D" ali "N" je nenaglašen odgovor vedno "N".
  - b) Vse specifikacije nizov so standardne specifikacije.
  - c) Oznake RMS statusov so standardne.
  - d) Odgovor <CTRL Z> na katerokoli vprašanje pomeni takojšen zaključek programa DEC z brisanjem niza odgovorov.
  - e) Nenaglašen odgovor je <CR>.
- Odgovor na vprašanje je primer.  
- Tu se specificira ime niza za niz v katerem bodo shranjeni odgovori za bodočo uporabo programa DEC.  
- Pravilen odgovor je standardna specifikacija niza kot je definirana v uporabniškem priročniku za operacijski sistem, vendar brez versije niza.  
- Nenaglašen odgovor <CR>, pomeni da se niz s odgovori ne kreira.  
- V primeru napadnega vnosa izpiše program vrsto napake, odgovor pa se mora vnesti ponovno.

Vprašanje \_\_\_\_\_ Odgovor \_\_\_\_\_

FUNKCIJA (DBX, LGX) ali "D" : DEBRUN

- OPIS:
- DEBRUN je primer.
  - Odgovor so trije alfanumerični znaki ali "4".
  - Ta vnos mora vsebovati ime funkcije, ki jo želimo izvajati ali "D", ki pomeni konec.
  - DB ..... funkcije za DB.
  - LG ..... funkcije za LOGEN.
  - X ..... tri različne funkcije - STOP, RUN, MON.
  - Nenaglašen odgovor ni mogoč.



4.2.1 UVODNA INTERAKCIJA - OPIS

V interaktivnem delu /režimu/ uporabnik odgovarja na vprašanja v zvezi s konvencijo specifikacije nizov in izvorno definicijo podatkovne strukture.

Vprašanje Odgovor

---

VNESI SPEC. ZA NIZ Z ODGOVORI : PRODAJ1TU

---

OPIS:

- PRODAJ1TU je primer.
- Odgovor je do trideset alfanumeričnih znakov.
- Tu se specificira ime niza za niz v katerem bodo shranjeni odgovori za bodočo uporabo programa DEC.
- Pravilen odgovor je standardna specifikacija niza kot je definirana v uporabniškem priročniku za operacijski sistem, vendar brez verzije niza.
- Nenaglašen odgovor <CR>, pomeni da se niz z odgovori ne kreira.
- V primeru napačnega vnosa izpiše program vrsto napake, odgovor pa se mora vnesti ponovno.

Vprašanje Odgovor

---

FUNKCIJA (DBX, LGX) ali "D" : DBRUN

---

OPIS:

- DBRUN je primer.
- Odgovor so trije alfanumerični znaki ali "D".
- Ta vnos mora vsebovati ime funkcije, ki jo želimo izvajati ali "D", ki pomeni konec.
- DB ..... funkcije za DBM.
- LGX ..... funkcije za LOGER.
- X ..... tri različne funkcije - STOP, RUN, MCN.
- Nenaglašen odgovor ni mogoč.



4.2.2 INTERAKCIJA ZA POSAMEZNO FUNKCIJO - OPIS

4.2.2.1 FUNKCIJA "DBRUN"

Funkcija "DBRUN" se uporablja za aktiviranje vseh potrebnih komponent za delo z bazo podatkov. V primeru kakršnekoli napake v okolju ali v posameznih komponentah, ali v primeru nepoznavanja gesla za področje, se aktiviranje ne more izvesti.

Če je bilo preverjanje okolja, področja in gesla uspešno se aktivira DBM komponenta, implicitno pa se aktivirata tudi LOGGER in DEG komponenti, če še nista aktivirani.

OPIS DIALOGA ZA FUNKCIJO "DBRUN":

Vprašanje: ..... Odgovor: .....

ŠT. TER. ZA DB/VM KOMPONENTE-TT: 0

OPIS:

- 0 je primer odgovora.
- Odgovor je ASCII oktalna številka. Ta vnos mora vsebovati številko terminala na katerem se bodo izvajale DF/VM komponente.
- Terminal, ki je naveden, mora biti privilegirano logiran.
- Logične enote morajo biti povezane s fizičnimi tudi na navedenem terminalu - ASN.
- Nenaglašen odgovor pomeni, da bodo DE/VM komponente aktivirane na terminalu na katerem se izvaja DBC.

10 7, 3  
9 7, 2  
8 5, 3

Vprašanje

Odgovor

IME PODROČJA (7)

: PRODAJ1

OPIS:

- PRODAJ1 je primer.
- Odgovor je sedem alfanumeričnih znakov.  
Ta vnos mora vsebovati ime področja.
- Ni nenaglašene odgovora na to vprašanje.

Vprašanje

Odgovor

GESLO PODROČJA

: PRODAJ

OPIS:

- PRODAJ je primer odgovora.
- Odgovor je šest alfanumeričnih znakov.  
Ta vnos specificira geslo za obravnavano področje.
- V primeru nepravilnega odgovora, DEM ne bo aktiviran.
- Nenaglašen odgovor je veljaven samo v primeru, če področje ni zaščiteno z geslom.

#### 4.2.2.2 FUNKCIJA "DBSTOP" -

Funkcija "DBSTOP" se uporablja za zaustavljanje DBM komponente.

DIALOG ZA FUNKCIJO "DBSTOP":

Dialog za funkcijo "DBSTOP" je enak kot pri funkciji "DERUN"

Na vprašanji "IME PODROČJA" in "GESLO PODROČJA" je dovoljen nenaglašen odgovor, s katerim zaustavimo DEM ne glede na trenutno aktivno področje. Zaustavljanje se dovoli samo v slučaju, če je število aktivnih programov 0.

#### 4.2.2.3 FUNKCIJA "LGRUN" -

Funkcija "LGRUN" se uporablja za aktiviranje LOGER  
komponente.

DIALOG ZA FUNKCIJO "LGRUN":

Za funkcijo "LGRUN" niso potrebne druge informacije.

#### 4.2.2.4 FUNKCIJA "LGSTOP" -

Funkcija "LGSTOP" se uporablja za zaključek dela LOGER  
komponente.

DIALOG ZA FUNKCIJO "LGSTOP":

Za funkcijo "LGSTOP" niso potrebne druge informacije.

#### 4.2.2.5 FUNKCIJA "LGMON" -

Funkcija "LGMON" se uporablja za kontrolo dela LOGER  
komponente. "LGMON" za sedaj še ni realizirana.

#### 4.2.2.6 FUNKCIJA "DEMON" -

Funkcija "DEMON" se uporablja za kontrolo dela DEM  
komponente. "DEMON" za sedaj še ni realizirana.

#### 4.3 DEC DIAGNOSTIKA NAPAK

Linija napake je sestavljena iz dveh delov :

- predpona - \*\*\*\* DEC
- ostali del sporočila

V opisu je naveden samo ostali del sporočila.

#### 02 Napaka pri odpiranju niza odgovorov (OPEN)

Nepravilno odpiranje niza odgovorov. Napaka je indentificirana v RMS statusu.

#### 06 Niz odgovorov ne obstaja

Napaka pri odpiranju niza odgovorov. Navadno je razlog nepravilna specifikacija niza pri indirektnem procesu.

#### 08 Napačna dolžina vnosa

Nepravilno število vnešenih znakov.

#### 11 PODROČJE ni aktivno.

Navedeno je napačno PODROČJE ali pa DBM dela z drugim PODROČJEM.

#### 17 Nepravilno geslo

Vnešeno geslo ni enako geslu iz definicije PODROČJA.

#### 20 Napaka pri obdelovanju niza odgovorov

Napaka pri čitanju niza odgovorov. Napaka je indentificirana v RMS statusu.

#### 30 Napačen odgovor

Na mestu, kjer se pričakuje odgovor "D" ali "N" je bila



4.2. vtipkan drugačen odgovor.

60. Napaka pri včitavanju ukazne vrstice

Napaka je identificirana s statusom direktive/\$DSW/.

71 Neprivilegirani uporabnik - OPROSTI

Terminal na katerem je bil startan program DEC ni privilegirani.

78 Proces je lahko samo DBRUN, DBMON, DESTOP, LGRUN, LGMON, LGSTOP.

Vtipkana je bila napačna šifra procesa.

79 Komunikacijska napaka

Napaka je nastala pri komunikaciji z DBM-jem ali LOGER-jem, kar v večini primerov pomeni, da vsaj eden od navedenih programov ni aktiven.

80 Status = XXXX

Izpiše se DBM status.

81 Ime P O D R O Č J A = XXXXXXXXXX

Informativno sporočilo.

82 Število aktivnih programov = YYY

Informativno sporočilo.

83 <<DATA BASE MONITOR>> zaustavljen

Informativno sporočilo.

84 PRIVILEGIRANA OPERACIJA - dovoljena na TTO: se dovoli samo iz

Informativno sporočilo.

Zaustavljanje programa DBM ali programa LOGER brez navajanja imena PODROČJA in GESLA se dovoli samo iz

konzole (TTO:) in v primeru če je ŠTEVILO AKTIVNIH PROGRAMOV več kot 0.

85 <<DATA BASE MONITOR>> ni izločen

Informativno sporočilo.

86 Vsi potrebni ASSGN-i za PODROČJE morajo biti izvedeni.

Informativno sporočilo.

Logične enote navedene pri kreiranju fizične strukture (DDP) morajo biti povezane s fizičnimi enotami preden se aktivira DBM ali LOGER.

87 <<DATA BASE MONITOR>> program ne obstaja/

Usodna napaka.

DATA BASE MONITOR ni aktiven.

88 << LOGER MONITOR >> zaustavljen

Informativno sporočilo.

89 << LOGER MONITOR >> ni zaustavljen

Informativno sporočilo.

99 Interna napaka

Napaka v programu DBC.

100 Proces še ni vključen

Informativno sporočilo.

- Če zapis deli V/I področje samo s nekaterimi in odvisnimi zapisi.
- Če zapis deli V/I področje samo s nekaterimi samostojnimi zapisi.
- Če ima zapis svoje nedeljive področje;
- Če ima zapis toliko V/I področij, kolikor je simultano aktivnih programov, ki delajo V/I operacije s temi zapisi.

Tri odvisnih in kombiniranih zapisih je treba upoštevati dejstvo, da so ti zapisi povezani s svetilni nadrejenimi

- 60 Napaka pri uveljavljanju ukazov vrstice  
 Informativno sporočilo.
- 61 Napaka je identificirana in izločen  
 Informativno sporočilo.
- 62 Neprivilegirani uporabnik - izločen  
 Vsi postopki LOGER-ja v LOGER-ju bodo prvi izvedeni na  
 privilegirani terminali.  
 Informativno sporočilo.
- 63 Neprivilegirani uporabnik - izločen  
 Informativno sporočilo.
- 64 <<DATA BASE MONITOR>> program ne obstaja  
 Informativno sporočilo.
- 65 Uporabna napaka.  
 Informativno sporočilo.
- 66 Napaka nastala pri komunikaciji z LOGER-jem.  
 Informativno sporočilo.
- 67 <<LOGGER MONITOR >> zavezanjen  
 Informativno sporočilo.
- 68 Status = XXXX  
 Informativno sporočilo.
- 69 <<LOGGER MONITOR >> ni zavezanjen  
 Informativno sporočilo.
- 70 Informativno sporočilo.  
 Informativno sporočilo.
- 71 Informativno sporočilo.  
 Informativno sporočilo.
- 72 Informativno sporočilo.  
 Informativno sporočilo.
- 73 Informativno sporočilo.  
 Informativno sporočilo.
- 74 Informativno sporočilo.  
 Informativno sporočilo.
- 75 Informativno sporočilo.  
 Informativno sporočilo.
- 76 Informativno sporočilo.  
 Informativno sporočilo.
- 77 Informativno sporočilo.  
 Informativno sporočilo.
- 78 Informativno sporočilo.  
 Informativno sporočilo.
- 79 Informativno sporočilo.  
 Informativno sporočilo.
- 80 Informativno sporočilo.  
 Informativno sporočilo.
- 81 Informativno sporočilo.  
 Informativno sporočilo.
- 82 Informativno sporočilo.  
 Informativno sporočilo.
- 83 Informativno sporočilo.  
 Informativno sporočilo.
- 84 Informativno sporočilo.  
 Informativno sporočilo.
- 85 Informativno sporočilo.  
 Informativno sporočilo.
- 86 Informativno sporočilo.  
 Informativno sporočilo.
- 87 Informativno sporočilo.  
 Informativno sporočilo.
- 88 Informativno sporočilo.  
 Informativno sporočilo.
- 89 Informativno sporočilo.  
 Informativno sporočilo.
- 90 Informativno sporočilo.  
 Informativno sporočilo.
- 91 Informativno sporočilo.  
 Informativno sporočilo.
- 92 Informativno sporočilo.  
 Informativno sporočilo.
- 93 Informativno sporočilo.  
 Informativno sporočilo.
- 94 Informativno sporočilo.  
 Informativno sporočilo.
- 95 Informativno sporočilo.  
 Informativno sporočilo.
- 96 Informativno sporočilo.  
 Informativno sporočilo.
- 97 Informativno sporočilo.  
 Informativno sporočilo.
- 98 Informativno sporočilo.  
 Informativno sporočilo.
- 99 Informativno sporočilo.  
 Informativno sporočilo.
- 100 Informativno sporočilo.  
 Informativno sporočilo.

## DODATEK A

### OPTIMIZIRANJE SISTEMA DB/VM

Sistem DB/VM je oblikovan tako, da se lahko instalira na vse računalnike DELTA, od najmanjših do največjih. Razumljivo je, da cilji optimizacij na velikih sistemih popolnoma drugačni, kot na malih računalnikih. V prvem primeru se skuša doseči čim krajši odzivni čas, saj mora DE/VM podpirati veliko število simultano aktivnih programov. Na malih sistemih pa se želi čimbolj zmanjšati pomnilniški prostor, ki ga uporablja sistem DB/VM. Ti dve zahtevi pa sta protislovnii.

#### A.1 OBLIKOVANJE V/I PODROČIJ

Na odzivni čas najbolj vpliva število posegov na diskovne enote. Te se da zmanjšati s pravilno uporabo V/I področij. Uporaba teh področij se po svojem konceptu razlikuje od koncepta pri konvencionalnih datotekah.

Pri samostojnih zapisih je uporaba V/I področij razmeroma enostavna. Najslabši odzivni čas je v primeru, da samostojni zapis deli V/I področje z vsemi drugimi zapisi. Izboljšuje pa se v naslednjih primerih:

- če zapis deli V/I področje samo z nekaterimi drugimi samostojnimi in odvisnimi zapisi.
- če zapis deli V/I področje samo z nekaterimi samostojnimi zapisi.
- če ima zapis svoje nedeljivo področje;
- če ima zapis toliko V/I področij, kolikor je simultano aktivnih programov, ki delajo V/I operacije s temi zapisi.

Pri odvisnih in kombiniranih zapisih je treba upoštevati dejstvo, da so ti zapisi povezani s svojimi nadrejenimi



zapisi. Praktično to pomeni, da je treba obravnavati pri optimizaciji V/I področij tudi vse samostojne zapise, s katerimi je določen zapis povezan. Posamezne funkcije DML imajo različne zahteve glede V/I področij za nadrejene zapise:

- GETP in GETD funkciji sta ovisni samo od V/I področij odvisnega zapisa;
- GETG in GETR funkciji sta odvisni od V/I področij odvisnega zapisa in od V/I področij samostojnega zapisa glede na pripadajoč set;
- INSG, INSA, INSB in DELG funkcije so odvisne od V/I področij odvisnega zapisa in od vseh V/I področij samostojnih zapisov, s katerimi je v povezavi.

Treba je zagotoviti več V/I področij za tiste zapise, na katerih se pogosteje izvajajo funkcije DML.

## A.2 OBLIKOVANJE PODROČIJ

Pri oblikovanju posameznih področij je treba analizirati vsebino programov, ki so simultano aktivni. Zato se praviloma oblikuje za isto bazo podatkov (shema) več področij. Tipična področja, ki se pojavljajo skoraj pri vsaki bazi podatkov so:

- Področje za primarno polnjenje;
- Področje za interaktivni režim dela;
- Področje za paketni režim dela.

Posebno na malih sistemih se priporoča večje število področij, saj gre navadno manjše število simultano aktivnih programov. Zato je priporočljivo analizirati tudi dinamiko aktivnosti. V situacijah, ko se da definirati ponovljive skupine simultanih programov, se področje optimizira glede na te programe. Kadar se takih skupin ne da predvideti, je koristno, če se za več zapisov (ki naj ne bi bili med seboj povezani in imajo enako dolžino logičnega zapisa) definira skupno V/I področje, ki se večkrat ponovi.

### A.3 OBLIKOVANJE FIZIČNE STRUKTURE

Cilj optimizacije fizične strukture je zmanjševanje premikov ročic na diskovnih enotah. Z vidika DB/VM bi bilo idealno, da bi bila vsaka množica enakih zapisov na svoji diskovni enoti. Ker to praktično ni mogoče, je treba upoštevati naslednja pravila za razvrščanje nizov po diskovnih enotah:

- Minimalna zahteva je, da obstajata vsaj dva kontejnerja, od katerih eden vsebuje samostojne, drugi pa odvisne zapise. Vsak od navedenih kontejnerjev naj bo na svoji diskovni enoti.

- V kolikor je na voljo več diskovnih enot, je treba razporediti nize in kontejnerje tako, da so na ločenih diskih tisti nizi, h katerim se pristopa simultano.

Pri tem je treba upoštevati, da pomenita npr. funkciji GETG in GETR za odvisni zapis poseg najprej v niz samostojnih zapisov in potem poseg v niz odvisnih zapisov.

- Za določene kratke nize, h katerim se pristopa pogosto, se lahko definira toliko V/I področij, da so vsi zapisi v glavnem pomnilniku. V tem primeru tega niza ni več treba upoštevati pri optimizacijah.

### A.4 OBLIKOVANJE LOGIČNIH BLOKOV

Dolžina logičnih blokov tudi vpliva na velikost potrebnega glavnega pomnilnika za V/I področja, poleg tega pa tudi na izkoristek diskovnega prostora. Ker zapisi niso deljeni med logičnimi bloki, je treba upoštevati dejstvo, da je del bloka neizkoriščen. Dolžina neizkoriščenega dela bloka se lahko izračuna:

$L = \text{dolžina bloka} - \text{dolžina zapisa} * \text{število zapisov v bloku}$

Dolžina zapisa za samostojne zapise:

$L = \text{dolžina podatkov} + c + \text{število podrejenih zapisov} * c$

$c=4$  : število zapisov v nizu < 32768

$c=8$  : število zapisov v nizu > 32767

Dolžina zapisa za odvisne zapise:

$L = \text{dolžina podatkov} + \text{število povezav} * c$

Dolžina podatkovnega dela kombiniranega zapisa je enaka kot pri odvisnem zapisu, področje indeksov pa je pod kontrolo sistema DB/VM in se nanj ne more vplivati.

Najbolje je, da se dolžina bloka oblikuje tako, da v bloku ni neizkoriščenega prostora. DB/VM je optimiziran za zapise dolge do 256 znakov, zato je priporočljivo, da zapisi niso daljši, če to ni nujno potrebno.

Zaradi internega razreševanja sinonimov je zelo nepriporočljivo oblikovanje logičnih blokov tako, da je v enem bloku en sam zapis. To onemogoča, da bi bil sinonim lahko razrešen v svojem bloku, kar zahteva dodatne V/I operacijo na disku. To se vidi iz kratkega opisa tega algoritma.

Procesor sinonimov razrešuje trke pri dodajanju novih zapisov po tej shemi:

1. Ugotavlja ali je zapis, ki trenutno zaseda naslovno celico tja razporejen zaradi vrednosti ključa (pravi sinonim) ali pa ga je tja razporedil algoritem za iskanje prazne celice (navidezni sinonim).

Tu se razreševanje razveja v dve smeri, v eni se razrešujejo pravi sinonimi, v drugi pa navidezni sinonimi.

Razreševanje pravih sinonimov:

2. V kolikor za naslovno celico že obstaja veriga sinonimov, se naslovi in prebere zadnji zapis v verigi, sicer se predpostavi, da je edini zapis v verigi tudi zadnji.

3. Poišče se nezasedena celica. Za to iskanje je značilno, da se najprej skuša najti nezasedeno celico v bloku, kjer je zadnji zapis. Če take celice ni, se skuša najti nezasedeno celico v sekciji, kjer se nahaja zadnji zapis. Če tudi tu ni razpoložljive prazne celice, se išče prazno celico v sekcijah, ki sledijo.



4. V prazno celico se vpiše novi zapis.
5. Ker so sinonimi medsebojno obojestransko povezani s kazalci (veriga naprej in veriga nazaj), je potrebno hkrati popraviti kazalce verige sinonimov.

Razreševanje navidezni sinonimov:

6. Poišče se nezasedeno celico in se vanjo premesti zapis, ki je označen kot navidezni sinonim. Pri tem se popravi tudi kazalce v verigi sinonimov, ki ji ta zapis pripada. S tem se sprosti celica, ki jo je prej zasedal navidezni sinonim.

R.1

7. V sproščeno celico se vpiše novi zapis, ki v tem primeru ne more imeti sinonimov.

#### A.5 OBLIKOVANJE VELIKOSTI NIZOV ZA SAMOSTOJNE ZAPISE

Ker se za dostop do samostojnih (lahko tudi do kombiniranih) zapisov uporablja kriptografski (hash) algoritem, je zelo ugodno, da je število razpoložljivih celic vsaj za petnajst odstotkov večje kot je dejanskih zapisov. S tem se pomembno izboljša pristopni čas do vseh zapisov v bazi podatkov. Ko število zapisov preseže to mejo, je tak niz treba povečati.



4. V prazno celico se vpiše koda, ki jo določa ena od naslednjih pogojev:

5. Ker so sinonimi med seboj povezani, je potrebno hkrati popraviti kodo za vse sinonime. (veriga besed) in verige besed, ki jih vsebujejo.

Določimo potek algoritma, ki bo določal, ali je potrebno popraviti kodo za vse sinonime. Pri tem moramo upoštevati, da se morajo popraviti kodi za vse sinonime, ki so povezani s kodo, ki jo popravljamo. To se vidi iz kretkega opisa tega algoritma.

6. Ko se popravijo kodi za vse sinonime, ki so povezani s kodo, ki jo popravljamo, moramo preveriti, ali so nastale nove kodi, ki so povezane s kodo, ki jo popravljamo. Če so nastale nove kodi, moramo te kodi popraviti. To se vidi iz kretkega opisa tega algoritma.

7. V prazno celico se vpiše koda, ki jo določa ena od naslednjih pogojev: (veriga besed) in verige besed, ki jih vsebujejo. Potrebno hkrati popraviti kodo za vse sinonime. Določimo potek algoritma, ki bo določal, ali je potrebno popraviti kodo za vse sinonime. Pri tem moramo upoštevati, da se morajo popraviti kodi za vse sinonime, ki so povezani s kodo, ki jo popravljamo. To se vidi iz kretkega opisa tega algoritma.

#### 4.2. OPREDELJEVALNI ALGORITMI ZA SAMOSTOJNI RAZREŠEVANJE

Ker so besede med seboj povezane, je potrebno hkrati popraviti kodo za vse besede, ki so povezane s kodo, ki jo popravljamo. Določimo potek algoritma, ki bo določal, ali je potrebno popraviti kodo za vse besede, ki so povezane s kodo, ki jo popravljamo. Pri tem moramo upoštevati, da se morajo popraviti kodi za vse besede, ki so povezane s kodo, ki jo popravljamo. To se vidi iz kretkega opisa tega algoritma.

#### Razreševanje besed:

1. V prazno celico se vpiše koda, ki jo določa ena od naslednjih pogojev: (veriga besed) in verige besed, ki jih vsebujejo. Potrebno hkrati popraviti kodo za vse sinonime. Določimo potek algoritma, ki bo določal, ali je potrebno popraviti kodo za vse sinonime. Pri tem moramo upoštevati, da se morajo popraviti kodi za vse sinonime, ki so povezani s kodo, ki jo popravljamo. To se vidi iz kretkega opisa tega algoritma.
2. Poiščite se nezasedene celice. Če se izkaže, da so celice nezasedene, moramo te celice popraviti. Če se izkaže, da so celice zasedene, moramo te celice popraviti. To se vidi iz kretkega opisa tega algoritma.

\*\* D E T A RACUNALNI SISTEMI \*\*  
"PRODAJ1" SHEMA BAZNE PODATKOV  
"PRODAJ1" SHEMA BAZNE PODATKOV

INIZI ZAPISA = KUPCI  
TIP ZAPISA = BAZNA  
INIZI V/I PODROČJA = KUPCI  
DODATEK B

PRIMERI  
IME SHEME  
OPIS

B.1 POROČILO O KREIRANJU PODROČJA "PRODAJ1"

OPIS	IMEN	ŠIFRA	ŠIFRA
STAVILO ZAKLEMLJENIH ZAPISOV	10	10	10
GAS DOSTOJA DO ZAKLEMLJENIH ZAPISOV	10	10	10
STAVILO AKTIVNIH PROGRAMOV	10	10	10
OPIS NIZA	10	10	10

FIZICNA VELOCOST NIZA ZAPISOV  
LOGICNA VELOCOST NIZA ZAPISOV  
STAVILO V ZAKLEMLJENIH ZAPISOV  
DOLZINA ZAPISA

DEFINICIJA FIZICNE STRUKTURE  
KONTAINER  
IME ZAPISA  
VARIJACIJA

DRAG:ŠPRODAJ1ŠPRODAJ1.COM

KONTAINER  
IME ZAPISA  
VARIJACIJA

DRAG:ŠPRODAJ1ŠPRODAJ1.COM

DRAG:ŠPRODAJ1ŠPRODAJ1.COM

DRAG:ŠPRODAJ1ŠPRODAJ1.COM

DP/VM - PRIROČNIK ZA ORGANIZATORJE  
 POROČILO O KREIRANJU PODROCJA "PRODAJ1"

\*\* D E L T A RACUNALNISKI SISTEMI \*\*

"PRODAJ" SHEMA BANKE PODATKOV 8/11/82 11:16:41 STRAN :1  
 "PRODAJ1" PODROCJE BANKE PODATKOV

SPECIFIKACIJE SHEME :

IME SHEME = PRODAJ  
 GESLO = PRODAJ

SPECIFIKACIJE PODROCJA :

IME PODROCJA = PRODAJ1  
 GESLO PODROCJA = PRODAJ  
 STEVILO AKTIVNIH PROGRAMOV = 10  
 CAS DOSTOPA DO ZAKLENJENIH ZAPISOV = 60  
 STEVILO ZAKLENJENIH ZAPISOV = 10

SPECIFIKACIJE V/I PODROCIJ:

IME STEVILO DOLZINA

IOKUPC	2	1024
IOIZDL	1	1024
IONARO	2	512
IONARI	2	512

DEFINICIJA FIZICNE STRUKTURE:

KONTEJNER	IME ZAPISA	VELIKOST	BLOKI
DRAO:ŠPRODAJACPRODAJA1.CON	KUPCII	12	8
	IZDLKI	18	
DRAO:ŠPRODAJACPRODAJA.CON	NAROCI	12	4
	NARIZD	20	

DE/VM - PRIROČNIK ZA ORGANIZATORJE  
 POROČILO O KREIRANJU PODROCJA "PRODAJ1"

\*\* D E L T A RACUNALNISKI SISTEMI \*\*  
 "PRODAJ" SHEMA BANKE PODATKOV 8/11/82 11:16:41 STRAN :2  
 "PRODAJ1" PODROCJE BANKE PODATKOV

IME ZAPISA = KUPCII  
 TIP ZAPISA = OWNER  
 IME V/I PODROCJA = IOKUPC

IME POLJA	DOLZINA	COBOLSKI OPIS
OWNKEY	6	PIC X(6)
IMEKUP	50	PIC X(50)
NASLOV	60	PIC X(60)
TELEFO	9	PIC 9(9)
SIFDEO	6	PIC X(6)

OPIS NIZA :

FIZICNA VELOKOST NIZA ZAPISOV = 12  
 LOGICNA VELIKOST NIZA ZAPISOV = 11  
 STEVILO ZAPISOV V BLOKU = 6  
 DOLZINA ZAPISA = 148

DEFINICIJA FIZICNE STRUKTURE:

KONTEJNER	IME ZAPISA	VELIKOST	BLOKI
DRAO:ŠPRODAJĆPRODAJA1.CON	KUPCII	12	8



\*\* D E L T A RACUNALNISKI SISTEMI \*\*

"PRODAJ" SHEMA BANKE PODATKOV 8/11/82 11:16:41 STRAN:3  
 "PRODAJ1" PODROCJE BANKE PODATKOV

IME ZAPISA = IZDLKI  
 TIP ZAPISA = OWNER  
 IME V/I PODROCJA = IOIZDL

IME POLJA	INA	COBOLSKI OPIS	DOLZINA
OWNKEY	12	PIC X(12)	12
IMEIZD	50	PIC X(60)	50
CENAIZ	9	PIC 9(7)V9(2)	9
KOLICI	10	PIC 9(7)V9(3)	10

OPIS NIZA

FIZICNA VELOKOST NIZA ZAPISOV = 18  
 LOGICNA VELOKOST NIZA ZAPISOV = 17  
 STEVILO ZAPISOV V BLOKU = 9  
 DOLZINA ZAPISA = 108

DEFINICIJA FIZICNE STRUKTURE:

KONTEJNER	IME ZAPISA	VELIKOST	BLOKI
DRAO:ŠPRODAJ1	PRODAJA1	.CON	8
	IZDLKI		8

KONTEJNER	IME ZAPISA	VELIKOST	BLOKI
DRAO:ŠPRODAJ1	PRODAJA1	.CON	8
	KUPC11		12
	IZDLKI		18
DRAO:ŠPRODAJ1	PRODAJA1	.CON	8
	NABOJ		12
	NARIS		20

DE/VM - PRIROČNIK ZA ORGANIZATORJE  
 POROČILO O KREIRANJU PODROCJA "PRODAJ1"

\*\* D E L T A RACUNALNISKI SISTEMI \*\*  
 "PRODAJ" SHEMA BANKE PODATKOV 8/11/82 11:16:41 STRAN :4  
 "PRODAJ1" PODROCJE BANKE PODATKOV

IME ZAPISA = NAROCI  
 TIP ZAPISA = OWNER IN MEMBER  
 IME V/I PODROCJA = IONARO

IME POLJA DOLZINA COBOLSKI OPIS

OWNKEY	5	PIC X(5)
SIFKUP	6	PIC X(6)
DATNAR	6	PIC 9(6)
ROKDOB	6	PIC 9(6)

OPIS NIZA :

FIZICNA VELOKOST NIZA ZAPISOV	=	12
LOGICNA VELIKOST NIZA ZAPISOV	=	10
STEVILLO ZAPISOV V BLOKU	=	12
DOLZINA ZAPISA	=	40

DEFINICIJA FIZICNE STRUKTURE:

KONTEJNER	IME ZAPISA	VELIKOST	BLOKI
DRAO:ŠPRODAJĆPRODAJA2.CON	NAROCI	12	4

\*\* D E L T A RACUNALNISKI SISTEMI \*\*  
 "PRODAJ" SHEMA BANKE PODATKOV 8/11/82 11:16:41 STRAN :5  
 "PRODAJ1" PODROČJE BANKE PODATKOV

IME ZAPISA = NARIZD  
 TIP ZAPISA = MEMBER  
 IME V/I PODROČJA = IONARI

IME POLJA	DOLZINA	COBOLSKI OPIS
STVNAR	5	PIC X(5)
SIFIZD	12	PIC X(12)
NARKOL	8	PIC 9(5)V9(3)
DOBKOL	8	PIC 9(5)V9(3)

OPIS NIZA :

FIZICNA VELOKOST NIZA ZAPISOV	=	20
LOGICNA VELIKOST NIZA ZAPISOV	=	18
STEVILLO ZAPISOV V BLOKU	=	10
DOLZINA ZAPISA	=	50

DEFINICIJA FIZICNE STRUKTURE:

KONTEJNER	IME ZAPISA	VELIKOST	BLOKI
DRAO:ŠPRODAJČPRODAJA2.CON	NARIZD	20	4





B.2 COBOLSKA DEFINICIJA PODSHEME "PRODAJ1T1"

```

*-----*
* 8/11/82 7:33:22 PRODAJ1T1.LIB
*-----*
01 SHEMA PIC X(6) VALUE "PRODAJ".
01 PODROCJE PIC X(7) VALUE "PRODAJ1".
01 PODSHEMA PIC X(9) VALUE "PRODAJ1T1".
01 PROJEKT PIC X(8) VALUE "DEMODEM ".
01 GESLO PIC X(6).
*-----*
* ZAPIS = KUPCII
* TIP ZAPISA = OWNER
* FUNKCIJA = GETP GET
*-----*
01 KUPCII001 PIC X(9) VALUE "KUPCII001".
01 KUPCII-001.
   05 KUPCIIOWNKEY PIC X(6).
   05 KUPCIIIMEKUP PIC X(50).
   05 KUPCIINASLOV PIC X(60).
   05 KUPCIITELEFO PIC 9(9).
   05 KUPCIISIFDEO PIC X(6).
*-----*
* ZAPIS = KUPCII
* TIP ZAPISA = OWNER
* FUNKCIJA = GET INS RWR
*-----*
01 KUPCII002 PIC X(9) VALUE "KUPCII002".
01 KUPCII-002.
   05 KUPCIIOWNKEY PIC X(6).
   05 KUPCIIIMEKUP PIC X(50).
   05 KUPCIINASLOV PIC X(60).
   05 KUPCIITELEFO PIC 9(9).
   05 KUPCIISIFDEO PIC X(6).
*-----*
* ZAPIS = IZDLKI
* TIP ZAPISA = OWNER
* FUNKCIJA = GETP GET
*-----*
01 IZDLKIO01 PIC X(9) VALUE "IZDLKIO01".
01 IZDLKI-001.
   05 IZDLKIOWNKEY PIC X(12).
   05 IZDLKICENAIZ PIC 9(7)V9(2).
   05 IZDLKIKOLICI PIC 9(7)V9(3).
*-----*
* ZAPIS = IZDLKI
* TIP ZAPISA = OWNER
* FUNKCIJA = GET INS RWR
*-----*
01 IZDLKIO02 PIC X(9) VALUE "IZDLKIO02".
01 IZDLKI-002.
   05 IZDLKIOWNKEY PIC X(12).
  
```

DB/VM - PRIROČNIK ZA ORGANIZATORJE  
 COBOLSKA DEFINICIJA PODSHEME "PRODAJIT1"

```

05 IZDLKIIMEIZD PIC X(60).
05 IZDLKICENAIZ PIC 9(7)V9(2).
05 IZDLKIKOLICI PIC 9(7)V9(3).
*=====
* ZAPIS = IZDLKI
* TIP ZAPISA = OWNER
* FUNKCIJA = GET INS RWR
*=====
01 IZDLKIOO3 PIC X(9) VALUE "IZDLKIOO3".
01 IZDLKI-003.
05 IZDLKIOWNKEY PIC X(12).
05 IZDLKICENAIZ PIC 9(7)V9(2).
05 IZDLKIKOLICI PIC 9(7)V9(3).
*=====
* ZAPIS = NAROCI
* TIP ZAPISA = OWNER IN MEMBER
* FUNKCIJA = GETP GET
*=====
01 NAROCIOO1 PIC X(9) VALUE "NAROCIOO1".
01 NAROCI-001.
05 NAROCIOWNKEY PIC X(5).
05 NAROCISIFKUP PIC X(6).
05 NAROCIDATNAR PIC 9(6).
05 NAROCIROKDOB PIC 9(6).
*=====
* ZAPIS = NAROCI
* TIP ZAPISA = OWNER IN MEMBER
* FUNKCIJA = GET INS RWR
*=====
01 NAROCIOO2 PIC X(9) VALUE "NAROCIOO2".
01 NAROCI-002.
05 NAROCIOWNKEY PIC X(5).
05 NAROCISIFKUP PIC X(6).
05 NAROCIDATNAR PIC 9(6).
05 NAROCIROKDOB PIC 9(6).
*=====
* ZAPIS = NAROCI
* TIP ZAPISA = OWNER IN MEMBER
* FUNKCIJA = GET DEL
*=====
01 NAROCIOO3 PIC X(9) VALUE "NAROCIOO3".
01 NAROCI-003.
05 NAROCIOWNKEY PIC X(5).
05 NAROCISIFKUP PIC X(6).
05 NAROCIDATNAR PIC 9(6).
05 NAROCIROKDOB PIC 9(6).
*=====
* ZAPIS = NARIZD
* TIP ZAPISA = MEMBER
* FUNKCIJA = GETP GET
*=====
01 NARIZDOO1 PIC X(9) VALUE "NARIZDOO1".
    
```

```
01 NARIZD-001.
    05 NARIZDSTVNAR PIC X(5).
    05 NARIZDSIFIZD PIC X(12).
    05 NARIZDNARKOL PIC 9(5)V9(3).
    05 NARIZDDOBKOL PIC 9(5)V9(3).
*-----*
* ZAPIS = NARIZD
* TIP ZAPISA = MEMBER
* FUNKCIJA = GET INS RWR
*-----*
01 NARIZD002 PIC X(9) VALUE "NARIZD002".
01 NARIZD-002.
    05 NARIZDSTVNAR PIC X(5).
    05 NARIZDSIFIZD PIC X(12).
    05 NARIZDNARKOL PIC 9(5)V9(3).
    05 NARIZDDOBKOL PIC 9(5)V9(3).
*-----*
* ZAPIS = NARIZD
* TIP ZAPISA = MEMBER
* FUNKCIJA = GET RWR
*-----*
01 NARIZD003 PIC X(9) VALUE "NARIZD003".
01 NARIZD-003.
    05 NARIZDSTVNAR PIC X(5).
    05 NARIZDSIFIZD PIC X(12).
    05 NARIZDDOBKOL PIC 9(5)V9(3).
*-----*
* ZAPIS = NARIZD
* TIP ZAPISA = MEMBER
* FUNKCIJA = GET DEL
*-----*
01 NARIZD004 PIC X(9) VALUE "NARIZD004".
01 NARIZD-004.
    05 NARIZDSTVNAR PIC X(5).
    05 NARIZDSIFIZD PIC X(12).
    05 NARIZDNARKOL PIC 9(5)V9(3).
    05 NARIZDDOBKOL PIC 9(5)V9(3).
*-----*
* ZAPIS = NARIZD
* TIP ZAPISA = MEMBER
* FUNKCIJA = GET INS RWR
*-----*
01 NARIZD005 PIC X(9) VALUE "NARIZD005".
01 NARIZD-005.
    05 NARIZDSTVNAR PIC X(5).
    05 NARIZDSIFIZD PIC X(12).
    05 NARIZDNARKOL PIC 9(5)V9(3).
    05 NARIZDDOBKOL PIC 9(5)V9(3).
*-----*
* ZAPIS = NARIZD
* TIP ZAPISA = MEMBER
* FUNKCIJA = GET INS RWR
*-----*
```



B.3 PRIMER DIALOGA

B.3.1 (KREIRANJE) SCHEMA

DE/VM - DDP

V1.0

```

=====
DDP KREIRANJE                1
DDP MODIFICIRANJE           2
DDP BRISANJE                 3
KONEC (CNTRL/Z)             4
=====
    
```

IZBRANI PROCES

1

DDP KREIRANJE

V1.0

```

=====
KREIRANJE SCHEMA             1
KREIRANJE PODROCJA           2
KREIRANJE PODSCHEMA          3
KREIRANJE FIZICNE STRUKTURE  4
KONEC (CNTRL/Z)             5
=====
    
```

IZBERI PROCES

1

IME SCHEMA (6)

MEMBER "NAROCI" : PRODAJ

GESLO

MEMBER :

```

=====
IME ZAPISA V SCHEMI (6)      : KUPCII
TIP ZAPISA V SCHEMI (S,O,M,X) : C
IME ZAPISA V SCHEMI (6)      : IZDLKI
TIP ZAPISA V SCHEMI (S,O,M,X) : 0
IME ZAPISA V SCHEMI (6)      : NAROCI
TIP ZAPISA V SCHEMI (S,O,M,X) : X
IME ZAPISA V SCHEMI (6)      : NARIZD
TIP ZAPISA V SCHEMI (S,O,M,X) : M
IME ZAPISA V SCHEMI (6)      : D
    
```

"KUPCII" OPIS DOLZINE KONTROLNEGA POLJA : X(6)

"KUPCII" IME PODATKOVNEGA POLJA : IMEKUP

OPIS DOLZINE PODATKOVNEGA POLJA : X(50)

"KUPCII" IME PODATKOVNEGA POLJA : NASLOV

OPIS DOLZINE PODATKOVNEGA POLJA : X(60)

"KUPCII" IME PODATKOVNEGA POLJA : TELEFO

OPIS DOLZINE PODATKOVNEGA POLJA : 9(9)

"KUPCII" IME PODATKOVNEGA POLJA : SIFDEO

OPIS DOLZINE PODATKOVNEGA POLJA : X(6)

"KUPCII" IME PODATKOVNEGA POLJA : D

"IZDLKI" OPIS DOLZINE KONTROLNEGA POLJA : X(12)

"IZDLKI" IME PODATKOVNEGA POLJA : IMEIZD

OPIS DOLZINE PODATKOVNEGA POLJA : X(60)



DE/VM - PRIROČNIK ZA ORGANIZATORJE  
PRIMER DIALOGA

```

"IZDLKI" IME PODATKOVNEGA POLJA          : CENAIZ
OPIS DOLZINE PODATKOVNEGA POLJA          : 9(7)V9(2)
"IZDLKI" IME PODATKOVNEGA POLJA          : KOLICI
OPIS DOLZINE PODATKOVNEGA POLJA          : 9(7)V999
"IZDLKI" IME PODATKOVNEGA POLJA          : D
=====
OPIS DOLZINE PODATKOVNEGA POLJA          : X(5)
"NAROCI" IME PODATKOVNEGA POLJA          : SIFKUP
OPIS DOLZINE PODATKOVNEGA POLJA          : X(6)
"NAROCI" IME PODATKOVNEGA POLJA          : DATNAR
OPIS DOLZINE PODATKOVNEGA POLJA          : 9(6)
"NAROCI" IME PODATKOVNEGA POLJA          : ROKDOB
OPIS DOLZINE PODATKOVNEGA POLJA          : 9(6)
"NAROCI" IME PODATKOVNEGA POLJA          : D
=====
"NARIZD" IME PODATKOVNEGA POLJA          : STVNAR
OPIS DOLZINE PODATKOVNEGA POLJA          : X(5)
"NARIZD" IME PODATKOVNEGA POLJA          : SIFIZD
OPIS DOLZINE PODATKOVNEGA POLJA          : X(12)
"NARIZD" IME PODATKOVNEGA POLJA          : NARKOL
OPIS DOLZINE PODATKOVNEGA POLJA          : 9(5)V999
"NARIZD" IME PODATKOVNEGA POLJA          : DOBKOL
OPIS DOLZINE PODATKOVNEGA POLJA          : 9(5)V999
"NARIZD" IME PODATKOVNEGA POLJA          : D
=====
IME SETA, OWNER "KUPCII" MEMBER "NAROCI" : KUPNAR
"NAROCI" KLJUC PRIPADNOSTI ZA SET         : SIFKUP
IME SETA, OWNER "KUPCII" MEMBER "NAROCI" : D
IME SETA, OWNER "KUPCII" MEMBER "NARIZD" : D
IME SETA, OWNER "IZDLKI" MEMBER "NAROCI" : D
IME SETA, OWNER "IZDLKI" MEMBER "NARIZD" : IZDNAI
"NARIZD" KLJUC PRIPADNOSTI ZA SET         : SIFIZD
IME SETA, OWNER "IZDLKI" MEMBER "NARIZD" : D
IME SETA, OWNER "NAROCI" MEMBER "NARIZD" : NARNAI
"NARIZD" KLJUC PRIPADNOSTI ZA SET         : STVNAR
IME SETA, OWNER "NAROCI" MEMBER "NARIZD" : D

"... KREIRAM DELOVNE NIZE ..."
Ž <EOF>

DDP KREIRANJE
=====
KREIRANJE SCHEME                : 1
KREIRANJE PODROCJA              : 2
KREIRANJE PODSCHEME             : 3
KREIRANJE FIZICNE STRUKTURE     : 4
KONEC (CNTRL/Z)                 : 5
=====
IZBERI PROCES                    : 5
DE/VM - DDP
=====

```

DE/VM - PRIROČNIK ZA ORGANIZATORJE  
PRIMER DIALOGA

DDP KREIRANJE	1
DDP MODIFICIRANJE	2
DDP BRISANJE	3
KONEC (CNTRL/Z)	4

-----  
IZBRANI PROCES :4

KREIRANJE SHEME	1
KREIRANJE PODROČJA	2
KREIRANJE PODSEME	3
KREIRANJE FIZIČNE STRUKTURE	4
KONEC (CNTRL/Z)	5

-----  
IZBRANI PROCES :4

DDP - VM/DB	1
DDP KREIRANJE	2
DDP MODIFICIRANJE	3
DDP BRISANJE	4
KONEC (CNTRL/Z)	5

-----  
IZBRANI PROCES :4

SHEMA (S)	1
OBMOČJE (O)	2

"... KIRIRAN DELOVNE NISB ..."

IME PODROČJA (P)	1
OBMOČJE (O)	2

O-KUPCI	1
X-NAROCI	2
M-NARIZD	3

IME ZAPISA V PODROČJU	1
IME ZAPISA V PODROČJU	2
IME ZAPISA V PODROČJU	3
IME ZAPISA V PODROČJU	4
IME ZAPISA V PODROČJU	5
IME ZAPISA V PODROČJU	6

STAVILO AKTIVNIH PROGRAMOV (P)	1
GAZ DOSTOPA DO ZAKLENJENIH ZAPISOV (EO)	2
STAVILO ZAKLENJENIH ZAPISOV (IO)	3

"KUPCI" IME V/I PODROČJA	1
"NAROCI" IME V/I PODROČJA	2
"NARIZD" IME V/I PODROČJA	3
"KUPCI" STAVILO PONOVITIV V/I PODROČJA	4
"NAROCI" STAVILO PONOVITIV V/I PODROČJA	5
"NARIZD" STAVILO PONOVITIV V/I PODROČJA	6

B.3.2 KREIRANJE PODROČJA

```
DE/VM - DDP                                V1.0
=====
DDP KREIRANJE                                1
DDP MODIFICIRANJE                            2
DDP ERISANJE                                  3
KONEC (CNTRL/Z)                              4
```

```
=====
IZBRANI PROCES                               :1
```

```
DDP KREIRANJE                                V1.0
=====
KREIRANJE SHEME                              1
KREIRANJE PODROCJA                            2
KREIRANJE PODSHEME                            3
KREIRANJE FIZICNE STRUKTURE                  4
KONEC (CNTRL/Z)                              5
```

```
=====
IZBERI PROCES                               :2
```

```
SHEME (6)                                     : PRODAJ
GESLO                                         :
```

```
"... KREIRAM DELOVNE NIZE ..."
```

```
IME PODROCJA (6)                             :PRODAJ1
GESLO                                         :
```

```
O-KUPCII      O-IZDLKI      X-NAROCI      M-NARIZD
```

```
IME ZAPISA V PODROCJU                        :KUPCII
IME ZAPISA V PODROCJU                        :IZDLKI
IME ZAPISA V PODROCJU                        :NAROCI
IME ZAPISA V PODROCJU                        :NARIZD
IME ZAPISA V PODROCJU                        :D
```

```
=====
STEVILO AKTIVNIH PROGRAMOV (10)             :
CAS DOSTOPA DO ZAKLENJENIH ZAPISOV (60)     :
STEVILO ZAKLENJENIH ZAPISOV (10)           :
```

```
=====
"KUPCII" IME V/I PODROCJA                    :IOKUPC
"IZDLKI" IME V/I PODROCJA                    :IOIZDL
"NAROCI" IME V/I PODROCJA                    :IONARO
"NARIZD" IME V/I PODROCJA                    :IONARI
"IOKUPC" STEVILO PONOVIJEV V/I PODROCJA     :2
"IOIZDL" STEVILO PONOVIJEV V/I PODROCJA     :
"IONARO" STEVILO PONOVIJEV V/I PODROCJA     :2
"IONARI" STEVILO PONOVIJEV V/I PODROCJA     :2
```



"... KREIRAN DELOVNE NIZE ..."

>Z <EOF>

DDP KFEIRANJE ..... V1.0

```

=====
KREIRANJE SHEME ..... 1
KREIRANJE PODROCJA ..... 2
KREIRANJE PODSCHEME ..... 3
KREIRANJE FIZICNE STRUKTURE ..... 4
KONEC (CNTRL/Z) ..... 5
=====

```

IZBERI PROCES ..... :5

DE/VM - DDP ..... V1.0

```

=====
DDP KFEIRANJE ..... 1
DDP MODIFICIRANJE ..... 2
DDP BRISANJE ..... 3
KONEC (CNTRL/Z) ..... 4
=====

```

IZBRANI PROCES ..... :4



B.3.3 KREIRANJE PODSHEME

```

DE/VM - DDP                                V1.0
=====
DDP KREIRANJE                               1
DDP MODIFICIRANJE                           2
DDP BRISANJE                                 3
KONEC (CNTRL/Z)                             4
=====
IZBRANI PROCES                               : 1
    
```

```

DDP KREIRANJE                                V1.0
=====
KREIRANJE SHEME                             1
KREIRANJE PODROCJA                           2
KREIRANJE PODSHEME                           3
KREIRANJE FIZICNE STRUKTURE                  4
KONEC (CNTRL/Z)                              5
=====
IZBRANI PROCES                               : 3
IME SHEME (6)                                : PRODAJ
GESLO                                          : PRODAJ
    
```

"...KREIRAM DELOVNE NIZE..."

```

IME PODROCJA (7)                             : PRODAJ1
GESLO                                          : PRODAJ
    
```

"...KREIRAM DELOVNE NIZE..."

```

IME PODSHEME (9)                             : PRODAJ1T1
GESLO                                          : PRODAJ
IME PROJEKTA (8)                             : DEMODEM
NACIN PRISTOPA V PODROCJE (R,U)              : U
    
```

O-KUPCII      O-IZDLKI      X-NAROCI      M-NARIZD

```

IME ZAPISA V PODSHEMI (6)                   : KUPCII
NACIN PRISTOPA K ZAPISU (S,P)                : S
IME ZAPISA V PODSHEMI (6)                   : IZDLKI
NACIN PRISTOPA K ZAPISU (S,P)                : S
IME ZAPISA V PODSHEMI (6)                   : NAROCI
NACIN PRISTOPA K ZAPISU (S,P)                : S
IME ZAPISA V PODSHEMI (6)                   : NARIZD
NACIN PRISTOPA K ZAPISU (S,P)                : S
IME ZAPISA V PODSHEMI (6)                   : D
    
```

O-KUPCII-T1    O-IZDLKI-T1    X-NAROCI-T1    M-NARIZD-T1

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR

DE/VM - PRIROČNIK ZA ORGANIZATORJE  
PRIMER DIALOGA

"KUPCII001" NACIN DELA S POLJI : 12  
KONTROLNO POLJE JE SELEKTIRANO

IMEKUP NASLOV TELEFO SIFDEO

"KUPCII001" IZBERI PODATKOVNA POLJA : ALL.

IMEKUP-01 NASLOV-01 TELEFO-01 SIFDEO-01

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"KUPCII002" NACIN DELA S POLJI : 35  
KONTROLNO POLJE JE SELEKTIRANO

IMEKUP NASLOV TELEFO SIFDEO

"KUPCII002" IZBERI PODATKOVNA POLJA : ALL.

IMEKUP-02 NASLOV-02 TELEFO-02 SIFDEO-02

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"KUPCII003" NACIN DELA S POLJI : D

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"IZDLKIO01" NACIN DELA S POLJI : 12  
KONTROLNO POLJE JE SELEKTIRANO

IMEIZD CENAIZ KOLICI

"IZDLKIO01" IZBERI PODATKOVNA POLJA : CENAIZ

"IZDLKIO01" IZBERI PODATKOVNA POLJA : KOLICI

"IZDLKIO01" IZBERI PODATKOVNA POLJA : D

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"IZDLKIO02" NACIN DELA S POLJI : 235  
KONTROLNO POLJE JE SELEKTIRANO

IMEIZD CENAIZ KOLICI

"IZDLKIO02" IZBERI PODATKOVNA POLJA : ALL.

IMEIZD-02 CENAIZ-02 KOLICI-02

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"IZDLKIO03" NACIN DELA S POLJI : 35  
KONTROLNO POLJE JE SELEKTIRANO

IMEIZD CENAIZ KOLICI

"IZDLKIO03" IZBERI PODATKOVNA POLJA : CENAIZ

"IZDLKIO03" IZBERI PODATKOVNA POLJA : KOLICI  
"IZDLKIO03" IZBERI PODATKOVNA POLJA : D

-----  
1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"IZDLKIO04" NACIN DELA S POLJI : D

-----  
1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"NAROCIO01" NACIN DELA S POLJI : 12

KUPCII-KUPNAR-NAROCI

-----  
IZBERI SET ZA PODATKOVNA POLJA :  
KONTROLNO POLJE JE SELEKTIRANO

SIFKUP-K DATNAR ROKDOB

"NAROCIO01" IZBERI PODATKOVNA POLJA : ALL.

SIFKUP-K-01 DATNAR-01 ROKDOB-01

-----  
1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"NAROCIO02" NACIN DELA S POLJI : 235

KUPCII-KUPNAR-NAROCI

-----  
IZBERI SET ZA PODATKOVNA POLJA : KUPNAR  
KONTROLNO POLJE JE SELEKTIRANO

SIFKUP-K DATNAR ROKDOB

"NAROCIO02" IZBERI PODATKOVNA POLJA : ALL.

SIFKUP-K-02 DATNAR-02 ROKDOB-02

-----  
1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR  
"NAROCIO03" NACIN DELA S POLJI : 24

KUPCII-KUPNAR-NAROCI

-----  
IZBERI SET ZA PODATKOVNA POLJA :  
KONTROLNO POLJE JE SELEKTIRANO

SIFKUP-K DATNAR ROKDOB

"NAROCIO03" IZBERI PODATKOVNA POLJA : ALL.

SIFKUP-K-03 DATNAR-03 ROKDOB-03

-----  
1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR



"NAROCIOO4" NACIN DELA S POLJI : D

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR

"NARIZDOO1" NACIN DELA S POLJI : 12

IZDLKI-IZDNAI-NARIZD NAROCI-NARNAI-NARIZD

IZBERI SET ZA PODATKOVNA POLJA : IZDNAI  
LISTI JE DODANO POLJE "SIFIZD" ZA KLJUC SETA

STVNAR-K SIFIZD-K-01 NARKOL DOBKOL

"NARIZDOO1" IZBERI PODATKOVNA POLJA : ALL.

STVNAR-K-01 SIFIZD-K-01 NARKOL-01 DOBKOL-01

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR

"NARIZDOO2" NACIN DELA S POLJI : 235

IZDLKI-IZDNAI-NARIZD NAROCI-NARNAI-NARIZD

IZBERI SET ZA PODATKOVNA POLJA : NARNAI  
LISTI JE DODANO POLJE "STVNAR" ZA KLJUC SETA

STVNAR-K-02 SIFIZD-K NARKOL DOBKOL

"NARIZDOO1" IZBERI PODATKOVNA POLJA : ALL.

STVNAR-K-02 SIFIZD-K-02 NARKOL-02 DOBKOL-02

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR

"NARIZDOO3" NACIN DELA S POLJI : 25

IZDLKI-IZDNAI-NARIZD NAROCI-NARNAI-NARIZD

IZBERI SET ZA PODATKOVNA POLJA : IZDNAI  
LISTI JE DODANO POLJE "STVNAR" ZA KLJUC SETA

STVNAR-K-03 SIFIZD-K NARKOL DOBKOL

"NARIZDOO3" IZBERI PODATKOVNA POLJA : SIFIZD

"NARIZDOO3" IZBERI PODATKOVNA POLJA : DOBKOL

"NARIZDOO3" IZBERI PODATKOVNA POLJA : D

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR



"NARIZOO4" NACIN DELA S POLJI : 24

IZDLKI-IZDNAI-NARIZD NAROCI-NARNAI-NARIZD

IZBERI SET ZA PODATKOVNA POLJA : NARNAI

LISTI JF DODANO POLJE "STVNAR" ZA KLJUC SETA

STVNAR-K-04 SIFIZD-K NARKOL DOBKOL

"NARIZOO1" IZBERI PODATKOVNA POLJA : ALL

STVNAR-K-04 SIFIZD-K-04 NARKOL-04 DOBKOL-04

1=GETP, 2=GET, 3=INS, 4=DEL, 5=RWR

"NARIZOO5" NACIN DELA S POLJI : D

IME SEKVENCIALNEGA ZAPISA PODSHEME (6) : D

"... KREIRAM DELOVNE NIZE ..."

>Ž <FOF>

DDP KFEIRANJE : V1.0

KREIRANJE SHEME 1

KREIRANJE PODROCJA 2

KREIRANJE PODSHEME 3

KREIRANJE FIZICNE STRUKTURE 4

KONEC (CNTRL/Z) 5

IZBERI PROCES : 5

DE/VM - DDP V1.0

DDP KREIRANJE 1

DDP MODIFICIRANJE 2

DDP BRISANJE 3

KONEC (CNTRL/Z) 4

IZBRANI PROCES : 4

B.3.4 KREIRANJE FIZIČNE STRUKTURE

DE/VM - DDP V1.0

=====

DDP KREIRANJE	1
DDP MODIFICIRANJE	2
DDP BRISANJE	3
KONEC (CNTRL/Z)	4

=====

IZBERI PROCES : 1

DE/VM - DDP KREIRANJE V1.0

=====

KREIRANJE SHEME	1
KREIRANJE PODROCJA	2
KREIRANJE PODSHEME	3
KREIRANJE FIZICNE STRUKTURE	4
KONEC (CNTRL/Z)	5

=====

IZBERI PROCES : 4

DE/VM - DDP KREIRANJE V1.0

=====

KREIRANJE FIZICNE STRUKTURE	1
RAZSIRITEV FIZICNE STRUKTURE	2
KONEC	3

=====

IZBERI PROCES : 1

IME SHEME (6) : PRODAJ

GESLO : PRODAJ

"... KREIRAM DELOVNE NIZE ..."

SPECIFIKACIJA KONTEJNERJA :ŠPRODAJĆPRODAJA1.COM

O-KUPCII O-IZDLKI X-NAROCI M-NARIZD

IZBERI ZAPISE ZA KONTEJNER :KUPCII

IZBERI ZAPISE ZA KONTEJNER :IZDLKI

IZBERI ZAPISE ZA KONTEJNER :D

"KUPCII" VELIKOST NIZA ZAPISOV :10

"KUPCII" STEVILO ZAPISOV V BLOKU :4

"IZDLKI" VELIKOST NIZA ZAPISOV :10

"IZDLKI" STEVILO ZAPISOV V BLOKU :9

SPECIFIKACIJA KONTEJNERJA :ŠPRODAJĆPRODAJA2.COM

O-KUPCII O-IZDLKI X-NAROCI M-NARIZD

DB/VM - PRIROČNIK ZA ORGANIZATORJE  
PRIMER DIALOGA

IZBERI ZAPISE ZA KONTEJNER :NAROCI  
 IZBERI ZAPISE ZA KONTEJNER :NARIZD  
 IZBERI ZAPISE ZA KONTEJNER :D

"NAROCI" VELIKOST NIZA ZAPISOV :10  
 "NAROCI" STEVILO ZAPISOV V BLOKU :9

"NARIZD" VELIKOST NIZA ZAPISOV :20  
 "NARIZD" STEVILO ZAPISOV V BLOKU :9

SPECIFIKACIJA KONTEJNERJA :D

"... KREIRAM DELOVNE NIZE ..."

SPECIFIKACIJA KONTEJNERJA :D

>Ž <EOF>  
 >Ž <EOF>  
 >Ž <EOF>

DB/VM - DDP KREIRANJE V1.0

KREIRANJE FIZICNE STRUKTURE 1  
 RAZSIRITEV FIZICNE STRUKTURE 2  
 KONEC 3

IZBERI PROCES : 3

DB/VM - DDP KREIRANJE V1.0

KREIRANJE SHEME 1  
 KREIRANJE PODROCJA 2  
 KREIRANJE PODSHEME 3  
 KREIRANJE FIZICNE STRUKTURE 4  
 KONEC (CNTRL/Z) 5

IZBERI PROCES : 5

DB/VM - DDP V1.0

DDP KREIRANJE 1  
 DDP MODIFICIRANJE 2  
 DDP BRISANJE 3  
 KONEC (CNTRL/Z) 4

IZBERI PROCES : 4



## INDEKS

- DB monitor, 1-3
- Direktni ključ, 1-4, 1-6
- Fizična struktura, 1-8
- Formatiranje, 1-3, 3-7, 3-11
- Ključ povezave, 1-4
- Kontejner, 1-8
- Logični blok, 1-8
- Niz zapisov, 1-4
- Označeni zapisi, 1-6
- Področje, 1-2
- Podshema, 1-2
- Podsistem DBF, 1-2
- Podsistem DDP, 1-1
- Podsistem DML, 1-3
- Procesor sinonimov, A-4
- Requal, 1-6, 2-6
- Shema, 1-2
- Struktura sistema DB/VM, 1-1
- V/I področje, 1-8

REPUBLIKA SRPSKA  
INSTITUT ZA ORGANIZACIJSKE IZMENJAVE  
BEOGRAD 16 783 044/  
prečišćena verzija

Izdajatelj:

© IZDAVAČKI TRŽNI KOMBINAT "VEŠTAČKI IZUMI"

LJUBLJANA  
1984



IZBERI ZAPISE ZA KONTEJNER :  
 IZBERI ZAPISE ZA KONTEJNER :  
 IZBERI ZAPISE ZA KONTEJNER :

"NAROCI" VELIKOST NIZA ZAPISOV :  
 "NAROCI" STEVILO ZAPISOV V BLOKU :

"NARIZD" VELIKOST NIZA ZAPISOV :  
 "NARIZD" STEVILO ZAPISOV V BLOKU :

SPECIFIKACIJA KONTEJNERJA :  
 ".... KREIRAN DELOVNE NIZE ..."

SPECIFIKACIJA KONTEJNERJA :  
 >E >VOP  
 >E >HOP  
 >E >EOP

DE/VN - DDP KREIRANJE :  
 KREIRANJE FIZICNE STRUKTURE :  
 RAZŠIRITEV FIZICNE STRUKTURE :  
 KONEC :

IZBERI PROCES :  
 Struktura sistema DV\N\Z\1

DE/VN - DDP KREIRANJE :  
 KREIRANJE SHEME :  
 KREIRANJE PODROČJA :  
 KREIRANJE PODSHEME :  
 KREIRANJE FIZICNE STRUKTURE :  
 KONEC (CNTRL/Z) :

IZBERI PROCES :  
 Struktura sistema DV\N\Z\1

DE/VN - DDP :  
 DDP KREIRANJE :  
 DDP MODIFICIRANJE :  
 DDP BRISANJE :  
 KONEC (CNTRL/Z) :

IZBERI PROCES :  
 Struktura sistema DV\N\Z\1

PRIPOMBE

VAŠE DELOVNO PODROČJE

- PROGRAMER  ORGANIZATOR  VODILNI DELAVEC  
 ŠTUDENT  DRUGO

DP/VP  
PRILGOČNIK ZA ORGANIZATORJE PODATKOVNIH STRUKTUR  
/Koda 16 783 044/  
preliminarna verzija

Izdajatelj:

© ISKRA DELTA, TRŽNO KOMUNICIRANJE, Parmova 41, Ljubljana

LJUBLJANA  
1984

IME PRIIMEK: \_\_\_\_\_ DATUM: \_\_\_\_\_  
ORGANIZACIJA: \_\_\_\_\_  
NASLOV: \_\_\_\_\_  
TELEF.: \_\_\_\_\_ TELEX: \_\_\_\_\_

PROJEKT STAVBY  
V RÁMCI VEŠKOPRÁVNÍHO  
VÝVOJE

PROJEKT STAVBY

PROJEKT STAVBY  
V RÁMCI VEŠKOPRÁVNÍHO  
VÝVOJE  
PROJEKT STAVBY  
V RÁMCI VEŠKOPRÁVNÍHO  
VÝVOJE

PROJEKT STAVBY  
V RÁMCI VEŠKOPRÁVNÍHO  
VÝVOJE  
PROJEKT STAVBY  
V RÁMCI VEŠKOPRÁVNÍHO  
VÝVOJE

PROJEKT STAVBY  
V RÁMCI VEŠKOPRÁVNÍHO  
VÝVOJE  
PROJEKT STAVBY  
V RÁMCI VEŠKOPRÁVNÍHO  
VÝVOJE







