

NEKI PROBLEMI VEZANI ZA USPOSTAVLJANJE STRUKTURE MREŽE

U radu je analiziran problem prikazivanja odnosa između elemenata strukture mreže. Kao rezultat činjenice da opis odnosa u kojem međusobno stoje dva elementa u strukturi mreže nije funkcijski ovisan ni o jednom od ta dva elementa, opis odnosa mora biti memoriran kao zaseban logički slog. Medjutim, taj logički slog nije element strukture mreže, nego dio sistema kojim se prikazuje odnos. Kao što je u radu pokazano logički slog, koji sadrži opis odnosa, može biti povezan s drugim slogovima u strukturi samo u točno odredjenim slučajevima.

Semantička struktura predstavlja skup odnosa među entitetima promatranog skupa. Prilikom memoriranja podataka opis svakog entiteta smješten je u zasebnom logičkom slogu. Ovi logički slogovi moraju na memoriji biti povezani u strukturu koja će prikazivati odnose među entitetima. Pod semantičkom strukturom podrazumijevamo, dakle, s jedne strane, odredjeni skup odnosa među entitetima promatranog skupa, a s druge strane, prikaz tih odnosa u strukturi podataka.

Način na koji se struktura podataka može uspostaviti ograničen je karakteristikama opreme i softvera koji koristimo. U okviru tako odredjenih granica za uspostavljanje strukture podataka odabiru se one mogućnosti koje najbolje odgovaraju sadržaju koji struktura prikazuje. Način uspostavljanja strukture ovisi prvenstveno o broju elemenata koje jednom elementu strukture neposredno prethode, odnosno neposredno iza njega slijede u strukturi. U skladu s tim ove semantičke strukture podijeljene su u tri grupe:

- linearne,
- hijerarhijske i
- strukture mreže.

U biti struktura mreže predstavlja najopćenitiji slučaj semantičke strukture. Kod nje ne postoji nikakvo ograničenje broja elemenata. Drugim riječima svaki element strukture mreže može imati po volji mnogo elemenata koji mu neposredno prethode, odnosno neposredno iza njega u strukturi slijede. Kod hijerarhijske i linearne strukture taj broj elemenata je ograničen. Zbog

toga se hijerarhijska struktura može smatrati posebnim slučajem strukture mreže, a linearna struktura posebnim slučajem hijerarhijske. (Tkalac, 1979).

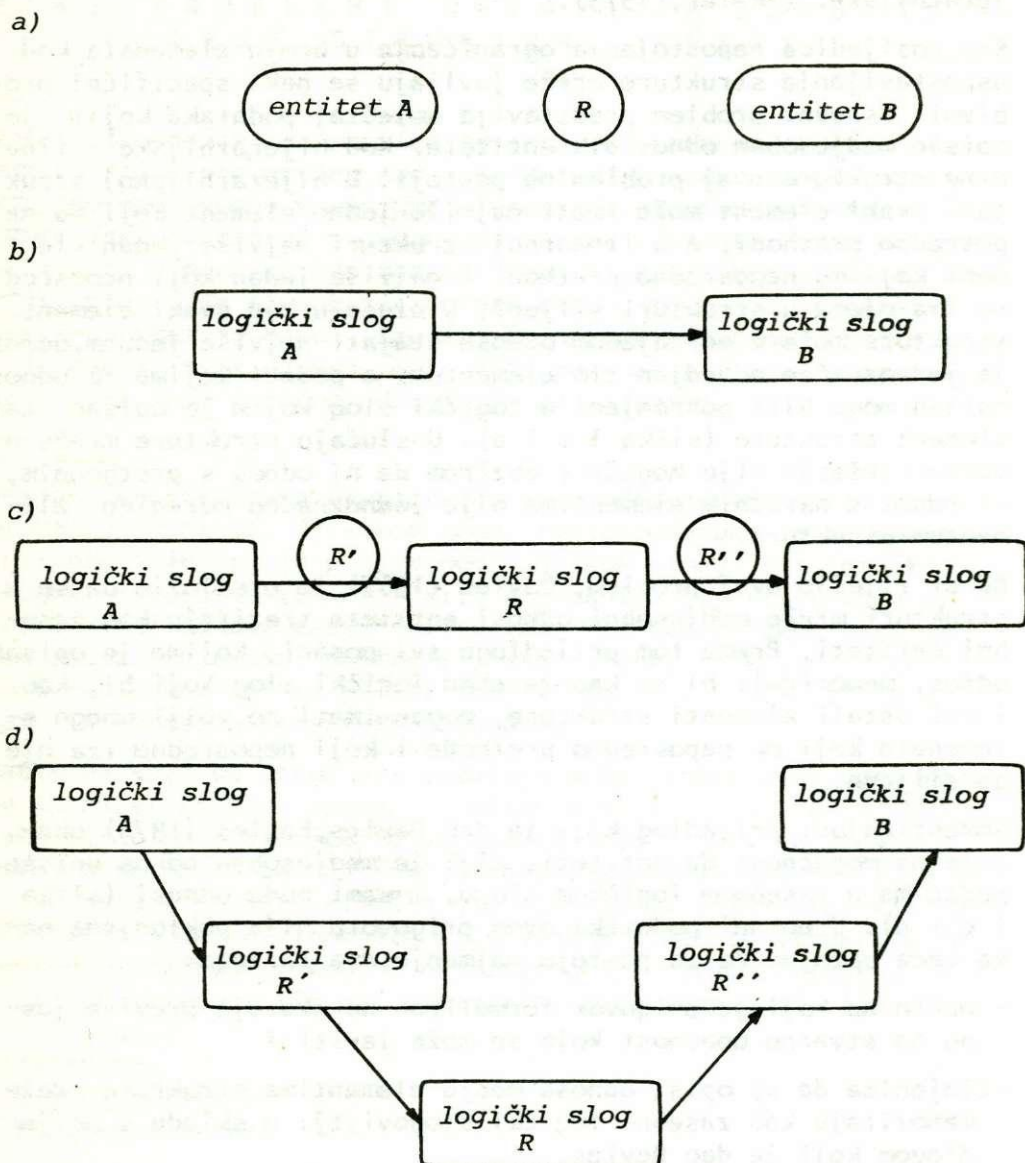
Kao posljedica nepostojanja ograničenja u broju elemenata kod uspostavljanja strukture mreže javljaju se neki specifični problemi. Osnovni problem predstavlja smještaj podataka kojim je opisan medjusoban odnos dva entiteta. Kod hijerarhijske i linearne strukture ovaj problem ne postoji. U hijerarhijskoj strukturi svaki element može imati najviše jedan element koji mu neposredno prethodi, a u linearnoj strukturi najviše jedan element koji mu neposredno prethodi i najviše jedan koji neposredno iza njega u strukturi slijedi. U slučaju kad svaki element strukture može u određenom odnosu stajati najviše jednom, odnos je jednoznačno određen tim elementom, a podaci kojima je odnos opisan mogu biti pohranjeni u logički slog kojim je opisan taj element strukture (slika 1 a i b). U slučaju strukture mreže ovakvo rješenje nije moguće s obzirom da ni odnos s prethodnim, ni odnos s narednim elementima nije jednoznačno određen elementom strukture.

Da bi riješio ovaj problem, Davies (1967) je predložio da se u strukturi mreže medjusobni odnosi entiteta tretiraju kao zasebni entiteti. Prema tom prijedlogu svi podaci, kojima je opisan odnos, memorirali bi se kao zaseban logički slog koji bi, kao i svi ostali elementi strukture, mogao imati po volji mnogo elemenata koji mu neposredno prethode i koji neposredno iza nje ga slijede.

Komentirajući prijedlog koji je dao Davies, Engles (1970) ukazuje na mogućnost da entiteti, čiji je medjusoban odnos opisan podacima u zasebnom logičkom slogu, i sami budu odnosi (slika 1 c i d). U obradi podataka ovom prigovoru nije poklonjena neka veća pažnja. Za to postoje najmanje dva razloga:

- način na koji je prigovor formuliran ne ukazuje previše jasno na stvarnu opasnost koja se može javiti i
- činjenica da se opisi odnosa medju elementima strukture mreže memoriraju kao zasebni logički slogovi, tj. u skladu s prijedlogom koji je dao Davies.

Engles polazi od sloga i odnosa i postavlja pitanje na što se odnose slogovi koji neposredno prethode, odnosno neposredno iza njega slijede u strukturi. Ako se podje od slogova kojima su opisani entiteti, prigovor se može formulirati na slijedeći



Slika 1

način: veze kojima su ti slogovi povezani sa slogom odnosa odgovaraju i prikazuju odnose u kojima stoje promatrani entiteti i entitet odnosa. Postoji mogućnost da u odredjenim slučajevima i svojstva tih odnosa moraju biti smještena u zasebne logičke slogove itd. (slika 1 c i d). Očito je da bi u takvim slučajevima rješenje koje je predložio Davies bilo neprihvatljivo. Umjesto da riješi pitanje prikazivanja odnosa između dva promatrana entiteta, ono bi generiralo beskonačan broj entiteta odnosa i veza među njima.

Teoretski problem prikazivanja odnosa među elementima strukture mreže nije riješen. U praksi se kod uspostavljanja strukture mreže koriste različita rješenja; od kojih su najčešća:

- transformacija strukture mreže u skup hijerarhijskih struktura,
- upotreba tzv. bit-slike i
- memoriranje opisa odnosa u obliku zasebnih logičkih slogova.

Ovo posljednje rješenje vjerojatno je najpovoljnije i najviše se koristi u praksi. Bilo da se radi o uspostavljanju odredjene strukture mreže ili o izradi softvera za podršku strukturi mreže, kod primjene ovog rješenja u velikoj mjeri se oslanjamo na iskustvo i intuiciju.

Zbog svega što je rečeno potrebno je odgovoriti na pitanje u kojim slučajevima se mogu javiti problemi na koje ukazuje Engles, odnosno kakav je položaj logičkih slogova koji sadrže opis odnosa unutar strukture mreže.

Raspored podataka u semantičkoj strukturi

Da bismo ilustrirali problem smještaja podataka kojima je opisan odnos između dva entiteta u semantičkoj strukturi, poslužit ćemo se primjerom računa i uplata. Radi veće preglednosti u primjerima je račun opisan sa svega dvije informacije (identifikator računa i iznos na koji račun glasi) isto kao i uplata (identifikator uplate i iznos na koji glasi). Odnos između računa i uplate kojom se račun podmiruje opisan je samo jednom informacijom (iznos koji se uplatom podmiruje na računu).

Sve informacije kojima je opisan račun pohranjene su u logičkom slogu računa, a sve informacije kojima je opisana uplata prikazane su podacima u logičkom slogu uplate. Informacije kojima je opisan odnos među entitetima smješteni su, ovisno o

tipu semantičke strukture, u logičkom slogu jednog od dva međusobno povezana entiteta ili u zasebnom logičkom slogu odnosa.

Vrsta semantičke strukture ovisi o ukupnom broju računa, odnosno o ukupnom broju uplata koji su međusobno povezani. Ovisno o tome da li se iznos na određenom računu uvijek podmiruje u cijelosti s jednom jedinom uplatom, ili može biti podmiren u nekoliko navrata s različitim uplatama, te da li se jednom uplatom podmiruje iznos samo na jednom ili mogu biti podmireni iznosi na nekoliko različitih računa, račun i uplata mogu međusobno stajati u odnosu:

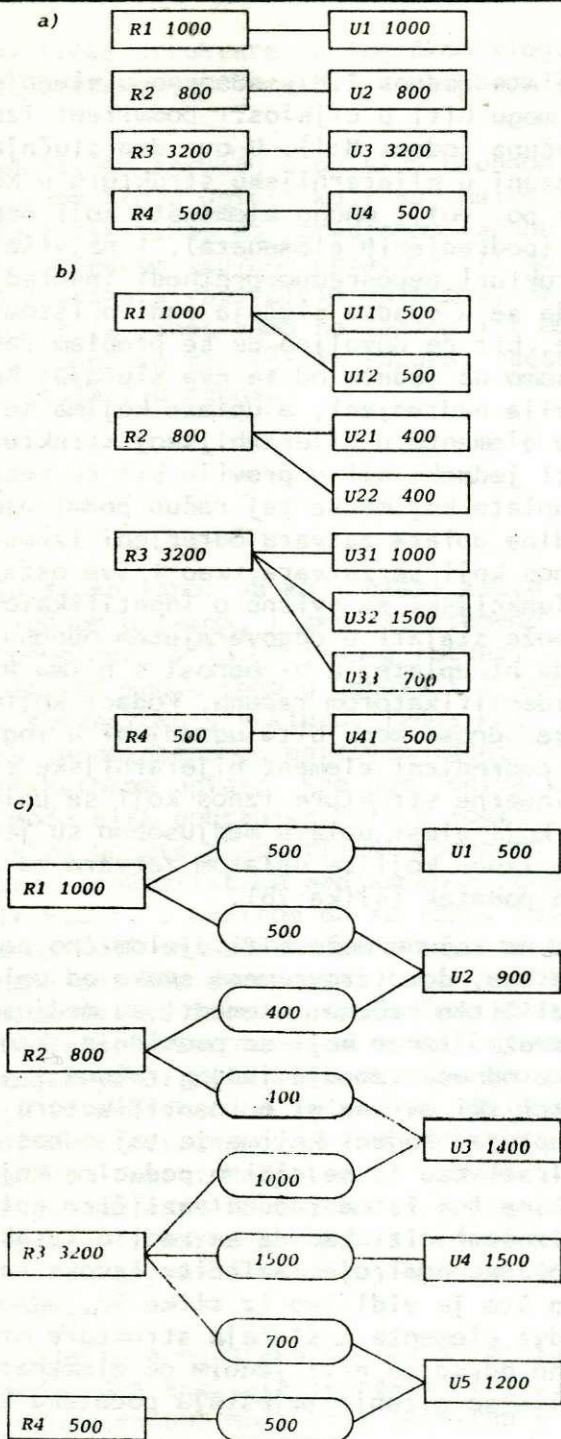
- 1 : 1
- 1 : N
- M : 1
- M : N

U slučaju da se iznos na koji glasi jedan račun u cijelosti podmiruje jednom uplatom, i ako istovremeno jednom uplatom ne može biti podmireno više od jednog računa, entiteti: račun i uplata nalaze se međusobno u odnosu 1:1, odnosno povezani su u linearnu semantičku strukturu. Iznos koji se uplatom podmiruje na računu jednoznačno je određen identifikatorom računa, kao i identifikatorom uplate. Kao posljedica toga iznos koji se podmiruje može biti pohranjen bilo u logički slog računa bilo u logički slog uplate. Ovo vrijedi općenito za sve podatke kojima su opisane karakteristike međusobnog odnosa elemenata u linearnoj strukturi. S obzirom da su iznos računa i iznos uplate kao i iznos koji se uplatom podmiruje na računu međusobno jednaki, nema potrebe da se ovaj treći iznos memorira kao poseban podatak (slika 2a). Naprosto ćemo smatrati da je iznos koji se podmiruje prikazan iznosom na koji glasi račun ili uplata. Nešto drugačija bi bila situacija da smo promatrali period vremena koji je protekao od izdavanja do podmirivanja računa. Podatak kojim je opisana ova karakteristika odnosa morao bi biti memoriran kao takav. Medjutim, i u ovom slučaju podatak bi mogao biti pohranjen bilo u jedan ili drugi logički slog. Općenito, u slučaju linearne strukture podaci kojima je opisan odnos mogu biti pohranjeni u bilo koji od dva međusobno povezana logička sloga.

Problem smještaja podataka kojima su opisane karakteristike odnosa između dva entiteta je nešto drugačiji u slučaju da iznos na jednom računu može biti podmiren u nekoliko navrata s

nekoliko manjih uplata (odnos 1:N), odnosno u slučaju da jednom većom uplatom mogu biti u cijelosti podmireni iznosi dva ili više manjih računa (odnos M:1). U ova dva slučaja elementi su međusobno povezani u hijerarhijsku strukturu u kojoj svaki element može imati po volji mnogo elemenata koji neposredno iza njega slijede (podredjenih elemenata), i najviše jedan element koji mu u strukturi neposredno prethodi (nadredjeni element). S obzirom da se u obadva slučaja radi o istom tipu semantičke strukture, bit će dovoljno da se problem rasporeda podataka analizira samo na jednom od ta dva slučaja. Kod odnosa 1:N račun predstavlja nadredjeni, a uplate kojima se račun podmiruje podredjene elemente u hijerarhijskoj strukturi. Iznos na računu može biti jednak, ali u pravilu bit će veći od iznosa na koji glase uplate kojima se taj račun podmiruje. S obzirom da svaka pojedina uplata zatvara odredjeni iznos samo na jednom računu, iznos koji se zatvara, kao i sve ostale karakteristike odnosa funkcijski su ovisne o identifikatoru uplate. Kako jedan račun može stajati u odgovarajućem odnosu s više uplata, očito je da ni uplate, a ni odnosi s njima nisu jednoznačno odredjeni identifikatorom računa. Podaci kojima su opisane karakteristike odnosa mogu biti ugrađeni u logički slog uplate, odnosno u podredjeni element hijerarhijske strukture. Kao i u slučaju linearne strukture iznos koji se uplatom podmiruje i iznos na koji glasi uplata međusobno su jednaki pa nema potrebe da se iznos koji se uplatom zatvara na računu memorira kao zaseban podatak (slika 2b).

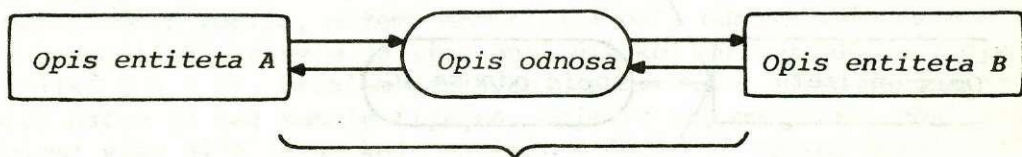
Konačno, ako iznos na računu može biti djelomično podmiren s više različitih uplata, dok istovremeno svaka od uplata može podmiriti više različitih računa, elementi su međusobno povezani u strukturu mreže. Iznos koji se podmiruje, kao i sve ostale karakteristike odnosa između jednog računa i jedne uplate, nije više funkcijski ovisan ni o identifikatoru računa, ni o identifikatoru uplate. Podaci kojima je taj odnos opisan ne mogu se više tretirati kao da se radi o podacima koji opisuju karakteristike računa (na istom računu različite uplate podmirit će različite iznose) niti kao da se radi o karakteristikama uplate (ista uplata podmiruje različite iznose na različitim računima). Kao što je vidljivo iz slike 2c, odnos u kojem međusobno stoje dva elementa u slučaju strukture mreže ne može biti jednoznačno odredjen niti jednim od elemenata. Kao posljedica toga javlja se pitanje smještaja podataka kojima je taj odnos opisan.



Slika 2

Problem smještaja podataka kojima je opisan odnos izmedju dva elementa strukture karakterističan je za strukturu mreže i javlja se neovisno o vrsti entiteta, vrsti odnosa i promatranim karakteristikama odnosa. U pravilu ovaj problem rješava se tako da se opis odnosa izdvaja i memorira kao zaseban logički slog.

Pomoću logičkog sloga koji sadrži opis odnosa i sistema veza kojima je taj slog povezan s opisima odgovarajućih entiteta prikazan je u strukturi mreže odnos izmedju dva entiteta (slika 3). Opis odnosa je prema tome dio sistema pomoću kojeg se u strukturi mreže prikazuje odnos izmedju dva entiteta.



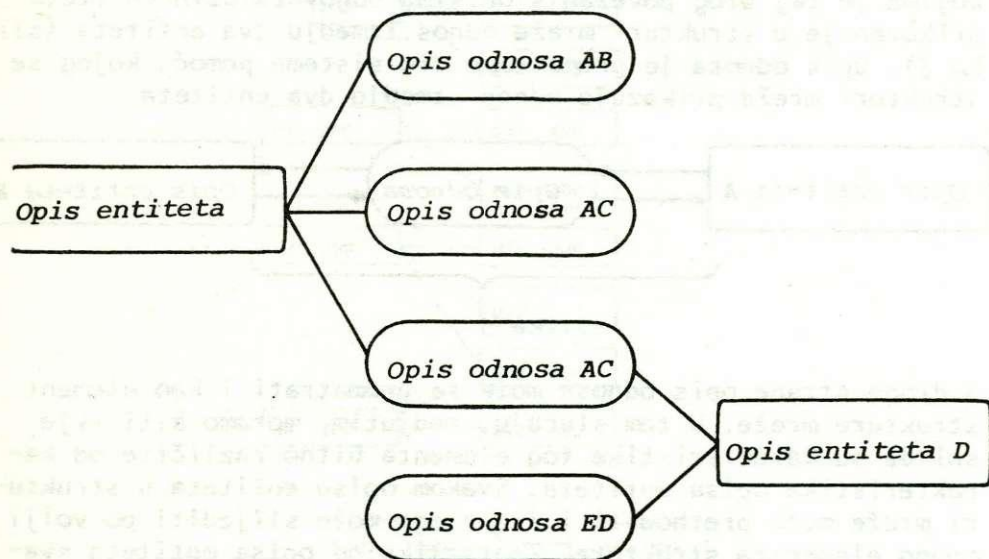
Slika 3

S druge strane opis odnosa može se promatrati i kao element strukture mreže. U tom slučaju, međjutim, moramo biti svjesni da su karakteristike tog elementa bitno različite od karakteristika opisa entiteta. Svakom opisu entiteta u strukturi mreže može prethoditi i iza njega može slijediti po volji mnogo elemenata strukture. Za razliku od opisa entiteta svakom opisu odnosa uvijek prethodi i iza njega slijedi samo jedan element strukture mreže. Ako promatramo samo jedan opis entiteta i sve s njim povezane opise odnosa, vidimo da su oni međjusobno povezani u hijerarhijsku strukturu (slika 4) u kojoj je opis entiteta nadređen, a opisi odnosa podređeni elementi.

Kao što smo ranije vidjeli, svi podaci kojima su opisane karakteristike odnosa medju elementima u hijerarhijskoj strukturi funkcijski su ovisni o podređenim elementima strukture i mogu u njih biti smješteni.

Vratimo se na Daviesov prijedlog i Englesov prigovor. Očito je da opis odnosa mora biti memoriran kao zaseban logički slog. Tako dugo dok je opis odnosa povezan samo s jednim slogom koji mu u strukturi prethodi i jednim koji slijedi neposredno iza njega, ne postoji opasnost od komplikacija o kojima govori Engles. Postavlja se, međjutim, pitanje da li se može javi-

ti potreba da opis odnosa bude povezan s varijabilnim brojem logičkih slogova u strukturi te da li bismo u tom slučaju odnos tretirali kao entitet.



Slika 4

Odnos između dva entiteta definiran je s jedne strane relacijom (kupac, dobavljač, roditelj, bračni drug itd.), a s druge strane entitetima koji u odnosu stoje. Karakteristike svakog odnosa možemo podijeliti u dvije grupe:

- karakteristike koje su rezultante odgovarajućih karakteristika entiteta koji u odnosu stoje i
- karakteristike relacije.

Prva grupa karakteristika specifična je za svaki pojedini odnos, dok će karakteristike iz druge grupe biti jednake za sve odnose koji su definirani istom relacijom. Uzmimo na primjer relaciju bračni drug. Svi bračni drugovi imaju određene dužnosti i prava. Ove karakteristike odnosa proizlaze iz karakte

ristika relacije i jednake su za sve bračne drugove. Jasno je, međutim, da će svaki pojedini brak imati niz karakteristika koje su posljedica odgovarajućih karakteristika bračnih drugova (zdravstveno stanje, navike, obrazovanje itd.).

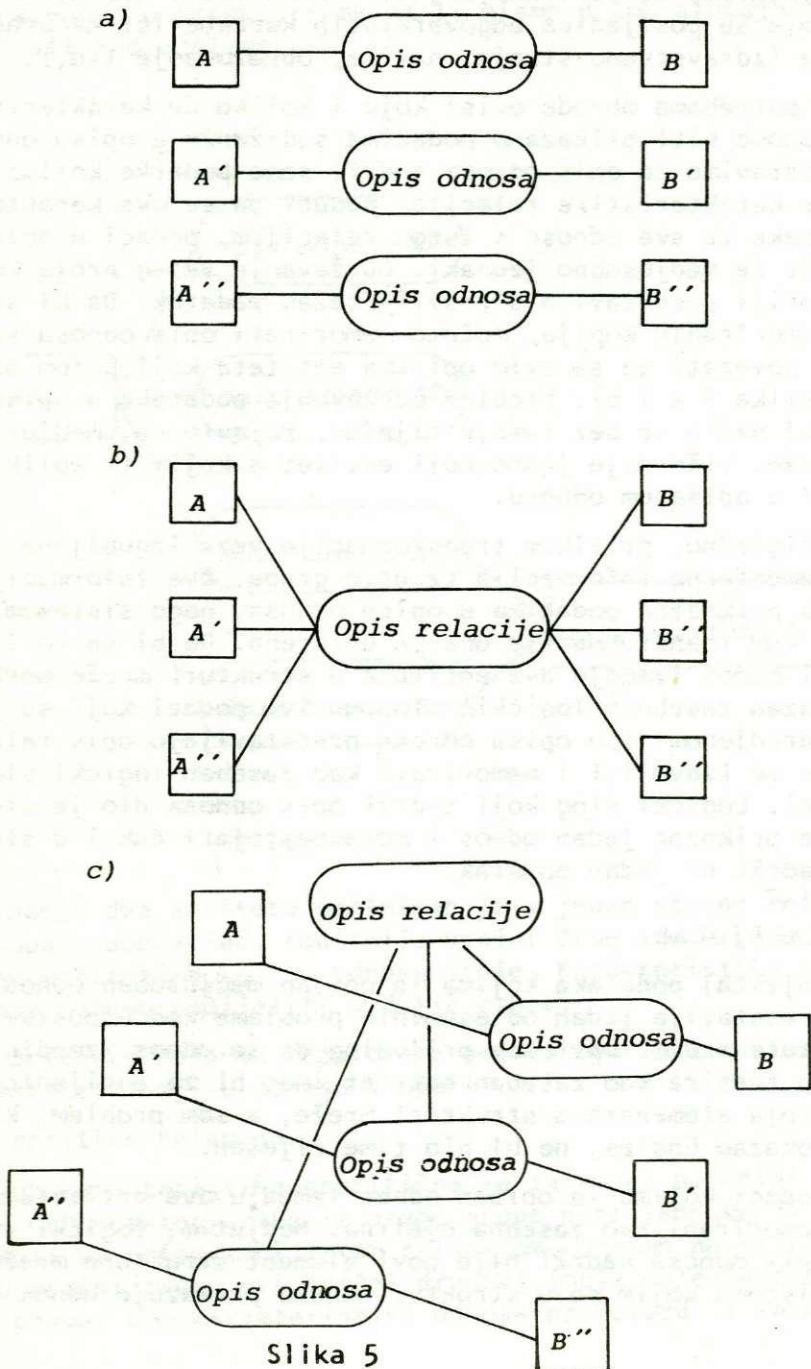
O potrebama obrade ovisi koje i koliko će karakteristika nekog odnosa biti prikazano podacima sadržanim u opisu odnosa. Pretpostavimo da opis odnosa sadrži samo podatke kojima su prikazane karakteristike relacije. Budući da su ove karakteristike jednake za sve odnose s istom relacijom, podaci u opisima odnosa bit će međusobno jednaki. Održavanje većeg broja kopija na memoriji predstavlja u pravilu težak zadatak. Da bi se izbjeglo memoriranje kopija, možemo memorirati opis odnosa samo jednom i povezati ga sa svim opisima entiteta koji u tom odnosu stoje (slika 5 a i b). Problem održavanja podataka u opisu odnosa na taj način je bez sumnje riješen. Pojavio se, međutim, novi problem: više nije jasno koji entitet s kojim (i koliko puta) stoji u opisanom odnosu.

Očigledno, prilikom transformacije veza izgubljena je jedna elementarna informacija iz prve grupe. Ova informacija nije bila prikazana podacima u opisu odnosa, nego sistemom veza i prilikom transformacije ona je uništena. Da bi se to izbjeglo, svaki odnos između dva entiteta u strukturi mreže mora biti prikazan zasebnim logičkim slogom. Svi podaci koji su jednaki u odredjenom tipu opisa odnosa predstavljaju opis relacije i mogu se izdvojiti i memorirati kao zaseban logički slog (slika 5c). Logički slog koji sadrži opis odnosa dio je sistema kojim je prikazan jedan odnos i mora postojati čak i u slučaju da ne sadrži ni jedan podatak.

Zaključak

Smještaj podataka kojima je opisan međusoban odnos entiteta predstavlja jedan od osnovnih problema kod uspostavljanja strukture mreže. Daviesov prijedlog da se odnos između dva entiteta tretira kao zaseban entitet imao bi za posljedicu povećanje broja elemenata u strukturi mreže, a sam problem, kao što je pokazao Engles, ne bi bio time riješen.

Podaci kojima je opisan odnos između dva entiteta moraju biti memorirani kao zasebna cjelina. Međutim, logički slog koji opis odnosa sadrži nije novi element strukture mreže, nego dio sistema kojim se u strukturi mreže prikazuje odnos između dva



Slika 5

entiteta. Za razliku od elemenata strukture mreže (logičkih slogova koji sadrže opise entiteta) kojima unutar strukture može neposredno prethoditi i iza njih neposredno slijediti po volji velik broj elemenata, broj logičkih slogova koji su povezani s jednim opisom odnosa jest za dani tip odnosa određen. U najjednostavnijem slučaju opisu odnosa neposredno prethodi i iza njega neposredno slijedi po jedan opis entiteta. U složenijim slučajevima opis odnosa pored toga može biti povezan i s jednim ili više logičkih slogova koji sadrže opis relacije.

Svaki logički slog s kojim je povezan opis odnosa ima u odnosu na njega ulogu hijerarhijski nadređenog elementa. Na taj način je logički slog koji sadrži opis odnosa povezan u biti u određen broj međusobno neovisnih hijerarhijskih struktura. Očito je da se položaj i uloga logičkog sloga koji sadrži opis odnosa u strukturi mreže bitno razlikuje od položaja i uloge logičkog sloga koji sadrži opis entiteta. Prema tome odnos se ne tretira kao novi entitet.

L I T E R A T U R A

- Davies, C.T. (1967): "A LOGICAL CONCEPT FOR THE CONTROL AND MANAGEMENT OF DATA", Report AR-0803-00, International Business Machines Corporation, Systems Development Division, Poughkeepsie, New York.
- Engles, R.W. (1970): "A TUTORIAL ON DATA BASE ORGANIZATION" Report TR-00.2004 International Business Machines Corporation, Systems Development Division, Poughkeepsie, New York.
- Tkalac, S. (1979) "STRUKTURA I ORGANIZACIJA PODATAKA", Fakultet organizacije i informatike, Varaždin.

Primljeno: 1980-12-27

*Tkalac S. Some Problems Connected with the Establishing of a
Network Structure*

S U M M A R Y

The problem of presenting relations between elements in a network structure is analysed. As a result of the fact that description of the relation between two elements in a network structure is not functionally dependent on any of the two elements, it has to be memorized as a separate logical record. However, the logical record is not an element of the structure but a part of a system which presents the relation. As is shown in the paper, the logical record which contains the description of a relation can be connected with other records in the structure only under strictly defined conditions.