Univerzitet u Novom Sadu
Fakultet tehničkih nauka

Mr Ivan Luković,
dipl. inž. informatike

INTEGRACIJA ŠEMA MODULA BAZE PODATAKA INFORMACIONOG SISTEMA

- doktorska disertacija -

Mentor
Dr Pavle Mogin,
redovni profesor

Novi Sad,
1995.

UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA

Ključna dokumentaciona informacija

Redni broj
RBR

Identifikacioni broj
IBR

Tip dokumenta
TD **monografska publikacija**

Tip zapisa
TZ **tekstualni štampani materijal**

Vrsta rada
VR **doktorska disertacija**

Autor
AU **mr Ivan Luković**

Mentor/Komentor
MN **dr Pavle Mogin, redovni profesor**

Naslov rada
NS **Integracija šema modula baze podataka informacionog sistema**

Jezik publikacije
JZ **srpski**

Zemlja publikovanja
ZP **Jugoslavija**

Uže geografsko područje
UGP **Vojvodina**

Godina
GO **1995.**

Izdavač
IZ **autorski reprint**

Mesto i adresa
MS **Trg D. Obradovića 5, 21000 Novi Sad, YU**

UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA

Ključna dokumentaciona informacija

Fizički opis rada
(*broj pogl. / strana / lit. citata / tabela / slika / grafika / priloga*)
FO **[6 / 142 / 49 / 0 / 90 / 1 / 3]**

Naučna oblast
OB **Informaciono-upravljački sistemi**

Naučna disciplina
DIS **Projektovanje baza podataka**

Predmetna odrednica / ključne reči
PO **Relacioni model podataka, CASE alati, Automatizovano projektovanje šeme baze podataka, Integracija šema baza podataka, Prototipski pristup, Generalizovane zavisnosti podataka**

UDK **519.688:519.683:681.3.06→519.58:510.51/.53**

Čuva se
ČU **u biblioteci Fakulteta tehničkih nauka u Novom Sadu, Trg D. Obradovića 6, 21000 Novi Sad, YU**

Važna napomena
VN **ne postoji**

Izvod
IZ **Paralelan i nezavisan rad više projektanata na različitim modulima (podsistemima) nekog informacionog sistema, identifikovanim saglasno početnoj funkcionalnoj dekompoziciji realnog sistema, nužno dovodi do međusobno nekonzistentnih rešenja šema modula baze podataka. Rad se bavi pitanjima identifikacije i razrešavanja problema, vezanih za automatsko otkrivanje kolizija, koje nastaju pri paralelnom projektovanju različitih šema modula i problema vezanih za integraciju šema modula u jedinstvenu šemu baze podataka informacionog sistema.**

Identifikovani su mogući tipovi kolizija šema modula, formulisan je i dokazan potreban i dovoljan uslov stroge i intenzionalne kompatibilnosti šema modula, što je omogućilo da se, u formi algoritama, prikažu postupci za ispitivanje

UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA

Ključna dokumentaciona informacija

stroge i intenzionalne kompatibilnosti šema modula. Formalizovan je i postupak integracije kompatibilnih šema u jedinstvenu (strogo pokrivajuću) šemu baze podataka. Dat je, takođe, prikaz metodologije primene algoritama za testiranje kompatibilnosti i integraciju šema modula u jedinstvenu šemu baze podataka informacionog sistema.

Datum prihvatanja teme od strane NN veća
DP **08. 07. 1994.**

Datum odbrane
DO

Članovi komisije (naučni stepen / ime i prezime / zvanje / fakultet)
KO

dr Pavle Mogin, redovni profesor,
Fakultet tehničkih nauka, Novi Sad

dr Branislav Lazarević, redovni profesor,
Fakultet organizacionih nauka, Beograd

dr Ilija Ćosić, redovni profesor,
Fakultet tehničkih nauka, Novi Sad

dr Dušan Malbaški, vanredni profesor,
Fakultet tehničkih nauka, Novi Sad

dr Vojislav Mišić, docent,
Elektrotehnički fakultet, Beograd

UNIVERSITY OF NOVI SAD
FACULTY OF TECHNICAL SCIENCES

Key words documentation

Accession Number ANO	
Identification Number INO	
Document Type DT	**Monographic publication**
Type of Record TR	**Textual printed article**
Contents Code CC	**PhD thesis**
Author AU	**Ivan Luković, MSc**
Mentor/Comentor MN	**Pavle Mogin, PhD, full professor**
Title TI	**Integration of Information System Database Module Schemas**
Language of Text LT	**Serbian**
Country of Publication CP	**Yugoslavia**
Locality of Publication LP	**Vojvodina**
Publication Year PY	**1995.**
Publisher PB	**author's reprint**

UNIVERSITY OF NOVI SAD
FACULTY OF TECHNICAL SCIENCES

Key words documentation

Publication Place
PL **Trg D. Obradovića 5, 21000 Novi Sad, YU**

Physical Description
(*Number of chapters / pages / lit. citations / tables / figures / graphs / appendixes*)
PD **[6 / 142 / 49 / 0 / 90 / 1 / 3]**

Scientific Field
SF **Information and Management Systems**

Scientific Discipline
SD **Database Design**

Subject / Key Words
CX **Relational Data Model, CASE Tools, Automated Database Schema Design, Database Schema Integration, Prototyping, Generalized Data Dependencies**

UC **519.688:519.683:681.3.06→519.58:510.51/.53**

Holding Data
HD **Faculty of Technical Sciences Library,**
Trg D. Obradovića 6, 21000 Novi Sad, YU

Note
N **none**

Abstract
AB **Parallel and independent work of a number of designers on different information system modules (i.e. subsystems), identified by the initial real system functional decomposition, necessarily leads to mutually inconsistent database (db) module schemas. The thesis considers the problems concerning automatic detection of collisions, that can appear during the simultaneous design of different db module schemas, and integration of db module schemas into the unique information system db schema.**

All possible types of db module schema collisions have been identified. Necessary and sufficient condition of strong and intensional db module schema compatibility has been formulated and proved. It has enabled to formalize the process of

UNIVERSITY OF NOVI SAD
FACULTY OF TECHNICAL SCIENCES

Key words documentation

db module schema strong and intensional compatibility checking and to construct the appropriate algorithms. The integration process of the unique (strong covering) db schema, on the basis of compatible db module schemas, is formalized, as well. The methodology of applying the algorithms for compatibility checking and unique db schema integration is also presented.

Accepted by the Scientific Board on
ASB **July, 08th 1994.**

Defended on
DE

Thesis Defend Board (name / surname / title / faculty)
DB

Pavle Mogin, PhD, full professor,
Faculty of Technical Sciences, Novi Sad

Branislav Lazarević, PhD, full professor,
Faculty of Organization Sciences, Belgrade

Ilija Ćosić, PhD, full professor,
Faculty of Technical Sciences, Novi Sad

Dušan Malbaški, PhD, associated professor
Faculty of Technical Sciences, Novi Sad

Vojislav Mišić, PhD, associated professor
Faculty of Electrical Sciences, Belgrade

Predgovor

U predgovoru magistarskog rada, koji sam sačinio pre dve i po godine, između ostalog je napisano: "Potreba da se razvoj nekog automatizovanog informacionog sistema obavi u relativno kratkom roku, pri čemu treba ispuniti niz vrlo strogih projektantskih i korisničkih zahteva, inicirala je automatizaciju postupaka projektovanja i realizacije informacionih sistema." Rezultat napora velikog broja naučnika i stručnjaka iz oblasti informatike "se ogleda kako u velikom broju teoretskih rešenja, tako i u čitavom spektru CASE (Computer Aided Software Engineering) alata, bez kojih bi razvoj informacionih sistema danas bio gotovo nezamisliv". Ono što sada može da se doda ovim konstatacijama, nakon samo dve i po godine, jeste da se "pritisak" na rokove izrade i kvalitet softverskih proizvoda, kao i "pritisak" projektantskih i korisničkih zahteva vrlo intenzivno pojačava. S druge strane, u komercijalnoj upotrebi su se pojavili softverski alati (a među njima i CASE alati) za razvoj informacionih sistema, takvi da operativno podržavaju niz tehničkih rešenja, standarda i koncepata koji su pre tri-četiri godine bili tek "stidljivo" spominjani, ili čak nepoznati. Sve ovo je doprinelo da istraživanje u oblasti teorije i prakse projektovanja baza podataka, kojoj ovaj doktorski rad pripada, još više dobije na značaju i aktuelnosti.

Pokazuje se da je posao projektanta šeme baze podataka visoko intelektualan i teoretski, ali istovremeno i inženjerski, odnosno "zanatski" orijentisan. Veliko će zadovoljstvo i motivacija biti za mene, ukoliko se pokaže da praktična primena rezultata ovog istraživanja bar malo olakšava teoretski deo posla projektanta i čini rezultat projektovanja kvalitetnijim.

Želim da izrazim svoju zahvalnost svima koji su mi pomagali i podržavali me u izradi doktorske disertacije. Zadovoljstvo je bilo sarađivati s članovima komisije dr Branislavom Lazarevićem, dr Ilijom Ćosićem, dr Dušanom Malbaškim i dr Vojislavom Mišićem. Posebnu čast mi čini plodonosna saradnja s profesorom dr Pavlom Moginom, koja je započela mojim dolaskom na Institut za industrijske sisteme Fakulteta tehničkih nauka, maja 1991. godine. Jedan od, za mene do sada najznačajnijih, rezultata te saradnje i nesebične pomoći profesora Mogina, predstavlja i ova doktorska disertacija.

u Novom Sadu, *autor*
oktobra 1995. godine

Sadržaj

1. Poglavlje

Uvod

Razvoj nekog informacionog sistema se može smatrati uspešnim samo ukoliko se taj informacioni sistem osposobi da u zadovoljavajuće kratkom vremenu pruži odgovore na sve predviđene informacione zahteve korisnika i da svoju funkciju obavlja uz optimalno angažovanje materijalnih i ljudskih resursa. Razvoj informacionih sistema većih proizvodnih ili uslužnih preduzeća predstavlja kompleksan poduhvat, za koji je bitno da bude zasnovan na metodološkim principima. Dosledna primena tih principa treba da dovede do proizvoda koji je u stanju da u predviđenom vremenu ispuni zadatu funkciju cilja. ***Motivacija*** istraživanja, reprezentovanog radom "*Problem integracije šema modula baze podataka informacionog sistema*" je da u tom smislu dâ određeni doprinos.

Svaki složeniji informacioni sistem se sastoji od funkcionalno zaokruženih celina - modula. Modul predstavlja model veće funkcionalne celine realnog sistema, tzv. poslovne funkcije. Informacionom sistemu, kao celini, odgovara jedinstvena šema baze podataka, dok svakom pojedinačnom modulu odgovara šema modula baze podataka (kraće šema modula). Zbog složenosti informacionog sistema, ukazuje se potreba da više projektanata relativno nezavisno i paralelno radi na oblikovanju pojedinačnih šema modula i aplikacija koje će se koristiti pri obavljanju radnih zadataka odgovarajuće poslovne funkcije. Činjenica da poslovne funkcije realnog sistema nisu međusobno nezavisne, znači da većina aplikacija budućeg informacinog sistema mora biti razvijena nad jedinstvenom šemom baze podataka informacionog sistema, a ne nad pojedinačnim šemama modula. Zbog toga, dobijene šeme modula treba objediniti u integrisanu šemu baze podataka.

Paralelan rad više projektanata na podsistemima, identifikovanim saglasno početnoj funkcionalnoj dekompoziciji realnog sistema, nužno dovodi do međusobno nekonzistentnih rešenja šema modula. Ukoliko se kolizije, iskazane putem različitih šema modula ne otklone i ukoliko se ne obezbede valjani postupci za njihovu integraciju u jedinstvenu šemu baze podataka, baza podataka takvog informacionog sistema neće verno odslikavati stanje realnog sistema, što znači da ni informacioni sistem, kao takav, neće biti upotrebljiv.

Pokazuje se da postupci otkrivanja i otklanjanja navedenih kolizija i integracije šeme baze podataka predstavljaju kompleksne aktivnosti, koje od projektanta traže visoki stepen razumevanja teoretskih osnova problema i veliko vremensko angažovanje u njihovom razrešavanju.

Predmet istraživanja rada koji se nalazi pred čitaocem, predstavlja identifikacija i razrešavanje problema, vezanih za automatsko otkrivanje kolizija, koje nastaju pri paralelnom projektovanju različitih šema modula i problema vezanih za integraciju šema modula u jedinstvenu šemu baze podataka informacionog sistema.

Krajnji ***cilj istraživanja***, prikazanog u ovom radu, predstavlja ugradnja postupaka i algoritama za integraciju šema modula u CASE alat, namenjen za automatizovano projektovanje šeme relacione baze podataka i generisanje prototipa aplikacija. Automatizacijom postupaka integracije šema modula u jedinstvenu šemu baze podataka i njihovom ugradnjom u CASE alat se stvaraju mogućnosti za povećanje produktivnosti rada projektanata informacionog sistema, uz istovremeno poboljšanje kvaliteta projekta integrisane šeme baze podataka.

Rad se bazira na osnovnim teoretskim postavkama relacionog modela podataka. Korišćeni su, takođe, elementi teorije skupova, relacija, grafova i predikatskog računa. U radu se insistiralo na uspostavljanju matematički preciznog definicionog sistema i formulisanju važnijih tvrdnji u obliku teorema, koje su dokazane. Insistiralo se, takođe, na algoritamskom prikazu dobijenih rešenja, kao i prikazu njihove primene na realnom primeru. Na početku istraživanja, reprezentovanog ovim radom, pošlo se od ***pretpostavke*** da se:

- mogu identifikovati potrebni i, eventualno, dovoljni uslovi da integrisana šema baze podataka predstavlja proširenje pojedinačnih šema modula. Poznati su slučajevi, kod kojih za date pojave nad šemama modula ne postoji pojava nad integrisanom šemom baze podataka, iako su sve šeme relacija i ograničenja pojedinačnih šema modula pokriveni integrisanom šemom baze podataka,
- može identifikovati postupak za integraciju šema modula u jedinstvenu šemu baze podataka informacionog sistema, pri čemu postupak integracije treba da obezbedi generisanje takve šeme baze podataka koja predstavlja proširenje šema modula,
- može automatizovati postupak integracije šema modula.

Materija, koja se u radu izlaže, je ***strukturirana*** u devet celina: *Uvod*, četiri tematska poglavlja, *Zaključak* i tri priloga: *Literatura*, *Pregled korišćenih oznaka i skraćenica* i *Indeks*.

Drugo poglavlje, pod naslovom *Definicije osnovnih pojmova i polazne pretpostavke*, daje kratak prikaz osnovnih pojmova relacionog modela podataka, koji se u radu koriste. Nakon toga, prikazani su: osnovna koncepcija CASE alata za automatizovano generisanje šeme baze podataka i prototipa aplikacija i metodološki aspekti razvoja informacionog sistema, na kojima se rad temelji.

U trećem poglavlju, naslovljenom *Familije generalizovanih zavisnosti*, detaljnije se diskutuje problem interpretacije generalizovanih zavisnosti i načini prikaza ostalih tipova zavisnosti podataka pomoću generalizovanih E-zavisnosti i generalizovanih T-zavisnosti. Potom se uvode pojmovi ranga i konačne specifikabilnosti familije pojava da bi, nakon toga, bio razmatran problem projekcije i prirodnog spoja familije pojava s obzirom na određene tipove zavisnosti podataka. Problem projekcije i prirodnog spoja familija pojava predstavlja osnov za rešavanje problema integracije šema relacionih baza podataka.

Na početku četvrtog poglavlja *Integracija šema modula baze podataka* se uvode pojmovi proširenja šeme relacije, proširenja relacione šeme baze podataka i integrativnog svojstva, koji se koriste za definisanje kriterijuma konzistentnosti šema modula i kriterijuma, koje mora da zadovolji integrisana šema. Nakon toga se, u formi algoritama, prezentira postupak analize konzistentnosti šema modula i na kraju se daje prikaz algoritma integracije dve šeme modula u novu, jedinstvenu, relacionu šemu.

Peto poglavlje *Metodološki aspekti primene postupka integracije šeme baze podataka* se, na početku, bavi pitanjima metodologije primene CASE alata za automatizovano generisanje šeme baze podataka i prototipa aplikacija u postupku razvoja informacionog sistema, a zatim se detaljnije razmatraju metodološki aspekti postupka testiranja međusobne kompatibilnosti šema modula i integracije šeme modula u jedinstvenu šemu baze podataka. Nakon toga se, na jednom praktičnom primeru, demonstrira primena rezultata (odnosno algoritama), prikazanih u četvrtom poglavlju.

Projektanti, koji pri oblikovanju relacionih šema modula i integrisane šeme baze podataka, koriste CASE alat u koji su ugrađeni algoritmi za otkrivanje kolizija šema modula i integraciju šeme modula u jedinstvenu šemu baze podataka, bivaju oslobođeni znatnog dela posla, koji su, pri tradicionalnom načinu rada, morali manuelno da obave. U okviru pristupa koji se u ovom radu predlaže, projektantima stoji na raspolaganju niz algoritama, koji u sebi sublimiraju rezultate kompleksnih teoretskih istraživanja i tehnika projektovanja šeme relacione baze podataka. Pri tome, projektanti treba da budu osposobljeni samo za ekspertsku analizu rezultata koje ti algoritami produkuju (na nivou studije slučaja, prikazane u petom poglavlju rada), bez zalaženja u sam način rada algoritama, čime se može očekivati povećanje produktivnosti njihovog rada i dolazak do kvalitetnijeg projekta šeme baze podataka informacionog sistema.

2. Poglavlje

Definicije osnovnih pojmova i polazne pretpostavke

U ovom poglavlju se daje kratki prikaz osnovnih postavki relacionog modela podataka i metodologije razvoja informacinog sistema na kojoj je ovaj rad zasnovan. Pretpostavlja se da je čitalac upoznat s konceptima relacionog modela podataka na nivou **[PBG**, **M**, **U]**, tako da se njihova detaljnija objašnjenja ovde izostavljaju.

2.1. Osnovne postavke relacionog modela podataka

2.1.1. Obeležje, domen, torka, relacija i familija

Osnovna pretpostavka od koje polazi relacioni model podataka je pretpostavka o egzistenciji:

- konačnog, nepraznog skupa ***obeležja*** (***atributa***) $U = \{A_1,...,A_n\}$ koji se naziva ***univerzalni skup obeležja*** i
- konačnog nepraznog skupa (konačnih ili beskonačnih) skupova konstanti $\mathcal{D} = \{D_1,...D_k\}$, koji se naziva ***skup domena***.

Definicija 2.1. Neka su dati univerzalni skup obeležja $U = \{A_1,...,A_n\}$ i skup domena $\mathcal{D} = \{D_1,...D_k\}$. ***Domensko određenje*** skupa U je bilo koja funkcija dom: $U \rightarrow \mathcal{D}$ koja svakom obeležju pridružuje tačno jedan domen: $(\forall A_i \in U)(dom(A_i) \in \mathcal{D})$. Struktura $(U, \mathcal{D}, dom)$ se naziva ***primitivna šema relacije***. **[PBG]**

Definicija 2.2. Neka je data primitivna šema relacije $(U, \mathcal{D}, dom)$ i neka je $DomU = \bigcup_{A_i \in U} dom(A_i)$. Skup svih ***torki*** (***n-torki***, ***slogova***), u oznaci $Tuple(U)$, je skup:

$$Tuple(U) = \{t: U \rightarrow DomU \mid (\forall A_i \in U)(t(A_i) \in dom(A_i))\}.$$ **[M]**

Torke t_1 i t_2 su ***jednake*** $(t_1 = t_2)$ ako i samo ako je $(\forall A_i \in U)(t_1(A_i) = t_2(A_i))$.

Definicija 2.3. Neka je data proizvoljna torka $t \in Tuple(U)$ i skup obeležja $X \subseteq U$. ***Restrikcija torke*** t na skup obeležja X je torka $t[X] \in Tuple(X)$, takva da je $(\forall A_i \in X)(t[X](A_i) = t(A_i))$. **[PBG, M]**

Definicija 2.4. Relacija nad primitivnom šemom $(U, \mathcal{D}, dom)$, u oznaci $r(U)$ je svaki konačan podskup skupa torki: $r(U) \subseteq Tuple(U) \wedge |\, r(U)| < \infty$. **[PBG]**

Definicija 2.5. Familija relacija nad primitivnom šemom $(U, \mathcal{D}, dom)$, u oznaci $\mathcal{F}(U)$, ili $SAT(U)$, predstavlja skup svih relacija $r(U)$ nad $(U, \mathcal{D}, dom)$. **[GZ]**

Ukoliko je skup torki $Tuple(U)$ konačan, tada važi da je $SAT(U) = \mathcal{P}(Tuple(U))$, pri čemu je sa $\mathcal{P}(X)$ označen partitivni skup skupa X.

Ukoliko se iz konteksta može zaključiti o kojoj primitivnoj šemi relacije se radi, umesto oznake $r(U)$ upotrebljavaće se oznaka r, a umesto oznake $\mathcal{F}(U)$ upotrebljavaće se oznaka $\mathcal{F}$.

Definicija 2.6. Neka je data relacija $r(U)$ i skup obeležja $X \subseteq U$. ***Projekcija*** relacije r na skup obeležja X, u oznaci $\Pi_X(r)$, je funkcija Π: $\mathcal{P}(U) \times SAT(U) \to SAT(X)$, data izrazom:

$$\Pi_X(r) = \Pi(X, r) = \{\, t \in Tuple(X) \,|\, (\exists t' \in r)(t'[X] = t)\}.$$ **[PBG, M]**

Definicija 2.7. Neka je data familija relacija $\mathcal{F} = SAT(U)$ i skup obeležja $X \subseteq U$. ***Projekcija*** familije $\mathcal{F}$ na skup obeležja X, u oznaci $\Pi_X(\mathcal{F})$, odnosno $\Pi_X(SAT(U))$, je skup relacija:

$$\Pi_X(\mathcal{F}) = \{\Pi_X(r) \in SAT(X) \,|\, r \in SAT(U)\}.$$ **[GZ]**

Definicija 2.8. Neka su date relacije $r(R)$ i $s(S)$. ***Prirodni spoj relacija*** r i s, u oznaci $r \rhd\!\lhd s$, je funkcija $\rhd\!\lhd$: $SAT(R) \times SAT(S) \to SAT(RS)$, data izrazom:

$$r \rhd\!\lhd s = \{\, t \in Tuple(RS) \,|\, t[R] \in r \wedge t[S] \in s\}.$$ **[PBG, M]**

Definicija 2.9. Neka su date familije $\mathcal{F}_1 = SAT(R)$ i $\mathcal{F}_2 = SAT(S)$. ***Prirodni spoj familija*** $\mathcal{F}_1$ i $\mathcal{F}_2$, u oznaci $\mathcal{F}_1 \rhd\!\lhd \mathcal{F}_2$, odnosno $SAT(R) \rhd\!\lhd SAT(S)$, je skup relacija:

$$\mathcal{F}_1 \rhd\!\lhd \mathcal{F}_2 = \{\, r \rhd\!\lhd s \in SAT(RS) \,|\, r \in SAT(R) \wedge s \in SAT(S)\}.$$ **[GZ]**

2.1.2. Šema relacije, pojava šeme relacije i dekompozicija

Definicija 2.10. Šema relacije nad primitivnom šemom $(U, \mathcal{D}, dom)$ predstavlja par (R, C), pri čemu je $R \subseteq U$ skup obeležja, a C skup ograničenja šeme relacije, definisan nad skupom obeležja R. Šema relacije (U, C) se naziva ***šema univerzalne relacije***. **[PBG]**

Činjenica da relacija $r(R)$ zadovoljava neko ograničenje podataka $\gamma \in C$ biće obeležavana kao $r(R) \models \gamma$. U tom slučaju se kaže da $r(R)$ predstavlja ***model*** za ograničenje γ. Činjenica da relacija $r(R)$ zadovoljava sva ograničenja iz skupa ograničenja C biće obeležavana kao $r(R) \models C$. U tom slučaju se kaže da $r(R)$ predstavlja ***model*** za skup ograničenja C.

Definicija 2.11. Neka je data šema relacije (U, C). Za ograničenje $\gamma \in C$ se kaže da je ***trivijalno ograničenje***, ako $(\forall r \in SAT(U))(\, r \models \gamma)$.

Definicija 2.12. Neka je data šema relacije (U, C). Za ograničenje γ se kaže da je ***logička posledica*** (***konsekvenca***) skupa ograničenja C, u oznaci $C \models \gamma$, ako važi:

$$(\forall r \in SAT(U))(r \models C \Rightarrow r \models \gamma).$$

Za skup ograničenja C' se kaže da predstavlja logičku posledicu skupa ograničenja C, u oznaci $C \models C'$, ako važi da $(\forall \gamma \in C')(C \models \gamma)$.

Definicija 2.13. Za skupove ograničenja C i C' se kaže da su ***ekvivalentni***, u oznaci $C \equiv C'$, ako važi da je $C \models C' \wedge C' \models C$. U tom slučaju, skup C predstavlja ***pokrivač*** skupa C' i skup C' predstavlja pokrivač skupa C. **[M]**

Definicija 2.14. Relacija $r(R)$ predstavlja ***pojavu*** (***instancu***) nad šemom relacije (R, C), ako važi da je $r(R) \models C$. **[PBG, Mo3]**

Definicija 2.15. Familija pojava (***familija instanci***) nad šemom (R, C), u oznaci $SAT(R, C)$, predstavlja skup svih pojava nad šemom (R, C): $SAT(R, C) = \{r \in SAT(R) | r \models C\}$. **[GZ]**

U nastavku rada će biti korišćena pretpostavka o postojanju šeme univerzalne relacije (**URSA**)[*]. Jedna od posledica ove pretpostavke jeste da se svakom obeležju dodeljuje jedinstvena uloga (semantika). Drugim rečima, zabranjuje se egzistencija istog obeležja s više različitih uloga (***homonima***). Na osnovu **URSA**, definiše se pojam dekompozicije šeme univerzalne relacije.

Definicija 2.16. Skup šema relacija $S = \{(R_i, C_i) | i \in \{1,...,n\}\}$ predstavlja ***dekompoziciju*** primitivne šeme relacije (U, $\mathcal{O}$, dom), u oznaci $S = d(U)$, ako važi:

$$(\forall i \in \{1,...,n\})(R_i \subseteq U \wedge R_i \neq \emptyset) \wedge \bigcup_{i=1}^{n} R_i = U \quad . \textbf{[M]}$$

2.1.3. Šema baze podataka, pojava baze podataka i dekompozicija

Definicija 2.17. Struktura (S, I), pri čemu je $S = \{(R_i, C_i) | i \in \{1,...,n\}\}$ skup šema relacija, a I skup međurelacionih ograničenja se naziva ***šema relacione baze podataka*** (**BP**). **[Mo3]**

Definicija 2.18. Neka je data šema baze podataka (S, I), gde je $S = \{(R_i, C_i) | i \in \{1,...,n\}\}$. ***Pojava*** nad šemom baze podataka (***baza podataka***) predstavlja skup relacija $\{r_i(R_i) | i \in \{1,...,n\}\}$, takav da je $(\forall i \in \{1,...,n\})(r_i \in SAT(R_i, C_i))$ i da globalno zadovoljava sva ograničenja iz skupa $I \cup \left(\bigcup_{i=1}^{n} C_i \right)$. **[Mo3]**

Definicija 2.19. Šema baze podataka (S, I), gde je $S = \{(R_i, C_i) | i \in \{1,...,n\}\}$, predstavlja ***dekompoziciju*** šeme univerzalne relacije (U, C), ako je: $S = d(U) \wedge (\forall i \in \{1,...,n\})(C \models C_i)$.

[*] **U**niversal **R**elation **S**cheme **A**ssumption

2.1.4. Tipovi ograničenja

2.1.4.1. Funkcionalna zavisnost

Definicija 2.20. Izraz oblika $X \to Y$, gde su $X \subseteq U$ i $Y \subseteq U$ proizvoljni skupovi obeležja se naziva ***funkcionalna zavisnost*** (***fd***). **[M, PBG]**

Skup svih mogućih *fd* nad skupom obeležja U je skup: $FD(U) = \{X \to Y \mid X, Y \subseteq U\}$. Važi da je $|FD(U)| = |\mathcal{P}(U)|^2$.

Skup obeležja leve strane *fd* $f: X \to Y \in FD(U)$ se obeležava sa $lhs(f)$ ($lhs(f) = X$), dok se skup obeležja desne strane obeležava sa $rhs(f)$ ($rhs(f) = Y$).

Definicija 2.21. Za relaciju $r(U)$ i *fd* $X \to Y$ važi da $r \models X \to Y$ ako

$$(\forall u, v \in r)(u[X] = v[X] \Rightarrow u[Y] = v[Y]). \quad \textbf{[M, PBG]}$$

Može se pokazati da je $X \to Y$ trivijalna *fd* ako i samo ako je $Y \subseteq X$. **[Lu2]**

Definicija 2.22. Neka je dat skup *fd* $\Gamma \subseteq FD(U)$. ***Zatvarač*** skupa *fd* s obzirom na $FD(U)$, u oznaci Γ^+, je skup $\Gamma^+ = \{f \in FD(U) \mid \Gamma \models f\}$. **[PBG]**

Definicija 2.23. Projekcija (***restrikcija***) skupa *fd* $\Gamma \subseteq FD(U)$ na skup obeležja $X \subseteq U$, u oznaci $\Pi_X(\Gamma)$, odnosno $\Gamma|_X$, je skup $\Pi_X(\Gamma) = \{W \to Y \in \Gamma^+ \mid WY \subseteq X\}$. **[M, PBG]**

Definicija 2.24. Data je šema relacije (R, Γ), pri čemu je $\Gamma \subseteq FD(U)$. ***Superključ*** šeme (R, Γ) je skup obeležja $K \subseteq R$, takav da $K \to R \in \Gamma^+$. **[Y]**

Definicija 2.25. Ključ šeme relacije (R, Γ), $\Gamma \subseteq FD(U)$, je superključ K za koji važi svojstvo minimalnosti: $\neg(\exists K' \subset K)(K' \to R \in \Gamma^+)$. **[U, Y]**

Osnovna pravila izvođenja za funkcionalne zavisnosti:

- *FD1* - ***refleksivnost***: $Y \subseteq X \subseteq U \vdash X \to Y$.
- *FD2* - ***proširenje***: $X \to Y,\ W \subseteq U \vdash XW \to YW$.
- *FD3* - ***tranzitivnost***: $X \to Y,\ Y \to Z \vdash X \to Z$.

Skup pravila $AP = \{FD1, FD2, FD3\}$ se naziva skup ***Armstrongovih*** pravila (umesto $X \cup Y$ će se koristiti skraćena notacija XY). **[Y, U, M]**.

Definicija 2.26. Dokaz (***niz izvođenja, dedukcijska sekvenca***) zavisnosti $f \in FD(U)$ iz skupa $\Gamma \subseteq FD(U)$, u oznaci $\Gamma \vdash_{AP} f$, gde je AP skup pravila izvođenja, je niz zavisnosti $f_1, \ldots, f_k$, takav da važi:

1. $(\forall f_i \in \{f_1, \ldots, f_k\})(f_i \in \Gamma \vee f_1, \ldots, f_{i-1} \vdash_{AP} f)$,
2. $f_k = f$.

Iz Γ se može dobiti f samo primenom *konačno mnogo puta* pravila izvođenja iz skupa AP.

Skup armstrongovih pravila AP je ***korektan*** i ***kompletan***, tj:

$$(\forall f \in FD(U))(\Gamma \vdash_{AP} f \Rightarrow \Gamma \models f) \wedge (\forall f \in FD(U))(\Gamma \models f \Rightarrow \Gamma \vdash_{AP} f). \ \textbf{[M, Y, Lu2]}$$

Definicija 2.27. Neka su dati skup *fd* $\Gamma \subseteq FD(U)$ i skup obeležja $X \subseteq U$. ***Zatvarač*** skupa obeležja X s obzirom na Γ je skup obeležja, u oznaci X_Γ^+, dat izrazom:

$$X_{\Gamma}^{+} = \{A \in U \mid \Gamma \vdash_{AP} X \to A\}. \quad \textbf{[U]}$$

S obzirom na korektnost i kompletnost armstrongovih pravila izvođenja sledi da je: $X_{\Gamma}^{+} = \{A \in U \mid \Gamma \models X \to A\}$. Na osnovu ove činjenice i definicije zatvarača skupa *fd* proizilazi ekvivalencija sledeće tri tvrdnje:

1. $\Gamma \models X \to A$,
2. $X \to A \in \Gamma^{+}$ i
3. $A \in X_{\Gamma}^{+}$.

Na osnovu ovih ekvivalencija, test da li je neka *fd* logička posledica datog skupa *fd* (***implikacioni problem***) će se obavljati izračunavanjem zatvarača leve strane date *fd* **[M**, **U**, **Lu2]** i testiranjem da li je svako obeležje desne strane date *fd* sadržano u izračunatom zatvaraču, ili ne.

Definicija 2.28. Neka je data *fd* $X \to Z$ i skup *fd* $\Gamma \subseteq FD(U)$. $X \to Z$ je ***tranzitivna*** *fd* s obzirom na Γ, ako $(\exists Y \subseteq U \setminus XZ)(X \to Y \in \Gamma^{+} \wedge Y \to X \notin \Gamma^{+} \wedge Y \to Z \in \Gamma^{+})$. **[M]**

Definicija 2.29. Neka je dat skup *fd* $\Gamma \subseteq FD(U)$ i *fd* $X \to Y \in \Gamma^{+}$. $X \to Y$ je ***nepotpuna*** (***parcijalna***) *fd* s obzirom na Γ, ako $(\exists X' \subset X)(X' \to Y \in \Gamma^{+})$.

Zavisnost koja nije parcijalna se naziva ***potpuna*** (***levo redukovana***) *fd*. **[M]**

Definicija 2.30. Neka je dat skup *fd* $\Gamma \subseteq FD(U)$ i parcijalna *fd* $X \to Y$. Obeležje $A \in X$, za koje važi da je $Y \subseteq (X \setminus \{A\})_{\Gamma}^{+}$ se naziva ***suvišno*** (***redundantno***) obeležje. Ukoliko, pri tome, važi da je $X \subseteq (X \setminus \{A\})_{\Gamma}^{+}$, obeležje A se naziva ***osnovno suvišno*** obeležje.

Definicija 2.31. Neka je dat skup *fd* $\Gamma \subseteq FD(U)$ i *fd* $X \to Y \in \Gamma$. $X \to Y$ je ***redundantna*** (***suvišna***) *fd* ako $\Gamma \setminus \{X \to Y\} \equiv \Gamma$. **[M]**

Može se pokazati **[Lu2]** da je svaka parcijalna *fd*, bez osnovnih suvišnih obeležja istovremeno i redundantna *fd*. Ova činjenica predstavlja osnov za jedno od poboljšanja tzv. modifikovanog algoritma sinteze, o kojem će biti reči u tački 2.2.2.3.

Neredundantni pokrivač skupa *fd* Γ, u oznaci $np(\Gamma)$, je takav pokrivač skupa Γ, koji ne sadrži redundantne *fd*. ***Kanonički pokrivač*** skupa *fd* Γ, u oznaci $kp(\Gamma)$, je takav neredundantni pokrivač skupa Γ, za koji važi da sadrži samo levo redukovane *fd*, oblika $X \to A$, pri čemu je $X \subseteq U$ i $A \in U$. **[M]**

Šema relacije (U, C) je u ***prvoj normalnoj formi*** (*1NF*) ukoliko su za svaku pojavu $r(U)$, sve vrednosti svih obeležja $A_i \in U$ ***atomarne***. Element domena $d \in dom(A_i)$ se smatra atomarnim, ukoliko d nije skup, ili niz nekih drugih elemenata iz $dom(A_i)$. Šema relacione BP (S, I) je u *1NF* ukoliko su sve šeme relacije iz skupa S u *1NF*. **[M]**

Neka je data šema relacije (R, Γ), $\Gamma \subseteq FD(U)$ i neka je $\mathcal{K}$ skup ključeva ove šeme. Neka je data unija svih ključeva iz skupa $\mathcal{K}$: $K_p(R) = \bigcup_{K_i \in \mathcal{K}} K_i$. Obeležje $A \in R$, za koje važi da je $A \in K_p(R)$, se naziva ***primarno*** obeležje. U protivnom, A je ***neprimarno*** (***sporedno***) obeležje šeme relacije (R, Γ). **[M]**

Definicija 2.32. Neka je data šema relacije (R, Γ), $\Gamma \subseteq FD(U)$ i neka je $\mathcal{K}$ skup ključeva ove šeme. (R, Γ) je u ***drugoj normalnoj formi*** (*2NF*), ako važi:

1. (R, Γ) je u *1NF* i

2. $(\forall A \in R)(A$ je **neprimarno** $\Rightarrow (\forall K_i \in \mathcal{K})(K_i \rightarrow A$ je **potpuna** $fd))$. **[M]**

2NF, drugim rečima, zabranjuje postojanje parcijalnih *fd* neprimarnih obeležja o nekom ključu šeme relacije. Šema BP (S, I) je u *2NF* ukoliko su sve šeme relacija iz skupa S u *2NF*.

Definicija 2.33. Neka je data šema relacije (R, Γ), $\Gamma \subseteq FD(U)$ i neka je $\mathcal{K}$ skup ključeva ove šeme. (R, Γ) je u ***trećoj normalnoj formi*** (*3NF*), ako važi:

1. (R, Γ) je u *2NF* i
2. $\neg(\exists A \in R)(\exists K_i \in \mathcal{K})(A$ je **neprimarno** $\wedge\ K_i \rightarrow A$ je **tranzitivna** $fd)$. **[M]**

3NF, dakle, zabranjuje postojanje tranzitivnih *fd* neprimarnih obeležja o nekom ključu šeme relacije. Šema BP (S, I) je u *3NF* ukoliko su sve šeme relacija iz skupa S u *3NF*.

2.1.4.2. Višeznačna zavisnost i zavisnosti spoja

Definicija 2.34. Izraz oblika $X \rightarrow\rightarrow Y$, gde su $X \subseteq U$ i $Y \subseteq U$ proizvoljni skupovi obeležja se naziva ***višeznačna zavisnost*** (***mvd***). **[M, PBG]**

Skup svih mogućih *mvd* nad skupom obeležja U je skup $MVD(U)=\{X \rightarrow\rightarrow Y \mid X,Y \subseteq U\}$. Važi da je $|MVD(U)| = |\mathcal{P}(U)|^2$.

Definicija 2.35. Za relaciju $r(U)$ i *mvd* $X \rightarrow\rightarrow Y$ važi da $r \models X \rightarrow\rightarrow Y$ ako

$$(\forall u, v \in r)(u[X] = v[X] \Rightarrow$$
$$(\exists t \in r)(t[XY] = u[XY] \wedge t[X\,(U \setminus Y)] = v[X\,(U \setminus Y)])). \quad \textbf{[M, PBG]}$$

Može se dokazati da je relacija $r(U)$ model za *mvd* $X \rightarrow\rightarrow Y$ ako i samo ako (akko) važi da je $r(U) = \Pi_{XY}(r) \rhd\lhd \Pi_{XZ}(r)$, pri čemu je $Z = U \setminus XY$ **[M, Lu2]**. Važi, takođe, da je $r \models X \rightarrow\rightarrow Y$ akko $r \models X \rightarrow\rightarrow Z$. Zbog toga se, vrlo često, višeznačna zavisnost navodi u notaciji: $X \rightarrow\rightarrow Y \mid Z$.

Može se pokazati da je $X \rightarrow\rightarrow Y$ trivijalna *mvd* akko je $XY = U$, ili $Y \subseteq X$. **[M, Lu2]**

Za *mvd* $X \rightarrow\rightarrow Y \mid Z$ se kaže da je ***ugrađena*** (*emvd*) u relaciju $r(U)$ ako je $\Pi_{XYZ}(r) \models X \rightarrow\rightarrow Y \mid Z$, pri čemu važi da je $XYZ \subset U$. U protivnom (ako je $XYZ = U$), *mvd* $X \rightarrow\rightarrow Y \mid Z$ se naziva ***potpunom*** (*fmvd*). **[M]**

Definicija 2.36. Izraz oblika $\rhd\lhd(X_1,...,X_n)$, pri čemu je n prirodni broj ($n \in \mathbf{N}$) i $(\forall i \in \{1,...,n\})(X_i \subseteq U)$ se naziva ***zavisnost spoja*** (*jd*). **[M, PBG]**

Skup svih mogućih *jd* nad skupom obeležja U je skup $JD(U) = \{\rhd\lhd(X_1,...,X_n) \mid n \in \mathbf{N} \wedge X_1...X_n \subseteq U\}$. Može se primetiti da je skup $JD(U)$ beskonačan, prebrojiv skup.

Definicija 2.37. Za relaciju $r(U)$ i *jd* $\rhd\lhd(X_1,...,X_n)$ važi da $r \models \rhd\lhd(X_1,...,X_n)$ ako

$$(\forall t_1,...,t_n \in r)((\forall i, j \in \{1,...,n\})(t_i[X_i \cap X_j] = t_j[X_i \cap X_j]) \Rightarrow$$
$$(\exists t \in r)(\forall i \in \{1,...,n\})(t[X_i] = t_i[X_i])). \quad \textbf{[M, PBG]}$$

Za *jd* $\rhd\lhd(X_1,...,X_n)$ se kaže da je ***ugrađena*** (*ejd*) u relaciju $r(U)$ ako je $r \models \rhd\lhd(X_1,...,X_n)$, pri čemu važi da je $X_1...X_n \subset U$. U protivnom (ako je $X_1...X_n = U$), *jd* $\rhd\lhd(X_1,...,X_n)$ se naziva ***potpunom*** (*fjd*). **[M, PBG]**.

Može se pokazati da je $r(U)$ model za potpunu *jd* $\bowtie(X_1,...,X_n)$ akko važi da je $r(U) = \Pi_{X_1}(r) \bowtie ... \bowtie \Pi_{X_n}(r)$ **[PBG]**. Može se, takođe, pokazati da je $r(U)$ model za ugrađenu *jd* $\bowtie(X_1,...,X_n)$ akko važi da je $\Pi_{X_1 ... X_n}(r) = \Pi_{X_1}(r) \bowtie ... \bowtie \Pi_{X_n}(r)$.

2.1.4.3. Zavisnost sadržavanja i referencijalni integritet

Definicija 2.38. Izraz oblika $R_1[X] \subseteq R_2[Y]$, gde su $R_1 \subseteq U$, $R_2 \subseteq U$, $X \subseteq R_1$ i $Y \subseteq R_2$ skupovi obeležja, takvi da postoji bijekcija $f: X \rightarrow Y$, pri čemu važi da $(\forall A_i \in X)(dom(A_i) \subseteq dom(f(A_i)))$ se naziva ***zavisnost sadržavanja*** (***ind***).

Definicija 2.39. Za *ind* $R_1[X] \subseteq R_2[Y]$ se kaže da važi nad relacijama $r_1(R_1)$ i $r_2(R_2)$ ako

$$(\forall u \in r_1)(\exists v \in r_2)(u[X] = v[Y]). \quad \textbf{[PBG]}$$

Ukoliko je $X \neq Y$, uobičajeno je da se X i Y predstavljaju putem ***nizova*** obeležja, pri čemu i-tom obeležju niza X odgovara i-to obeležje niza Y. Na taj način se zadaje i bijekcija $f: X \rightarrow Y$. Ukoliko se u daljem tekstu, pri navođenju zavisnosti sadržavanja, bijekcija f **ne** navede, onda će podrazumevati da su X i Y nizovi obeležja, a da je f zadata redosledom navođenja obeležja.

Zavisnost sadržavanja $R_1[X] \subseteq R_2[Y]$ se naziva ***uslovom referencijalnog integriteta*** (***rd***) ako u skupu šema relacija šeme BP postoji šema relacije (R_2, C_2), takva da je Y njen ključ. **[PBG]**

2.1.4.4. Tablo, generalizovane zavisnosti i Chase algoritam

U ovom delu rada se samo navode osnovne definicije, vezane za tablo, generalizovane zavisnosti i chase algoritam. Detaljniji prikaz pojmova, uvedenih ovim definicijama, će biti dat u okviru narednog poglavlja.

Definicija 2.40. Neka je dat skup obeležja $U = \{A_1,...,A_n\}$ i skup simbola (promenljivih) $Sym = \{x, y, z, x_1,...,x_k,..., x_1^1,...,x_m^l,...,a, b, c,...\}$. ***Tablo*** (***reprezent***) nad skupom obeležja U, u oznaci $T(U)$, predstavlja konačni skup predikata dužine n: $T(U) = \{ P_i^n(\lambda_i^1,...,\lambda_i^n) \mid i \in \{1,...,k\}\}$, takav da

1. $(\forall i \in \{1,...,k\})(\forall j \in \{1,...,n\})(\lambda_i^j \in Sym) \wedge$
2. $(\forall i \in \{1,...,k\})(\exists f_{P_i}: U \rightarrow Sym)(\forall A_j \in U)(f_{P_i}(A_j) = \lambda_i^j)$.

Saglasno prethodnoj definiciji, tablo se može predstaviti putem tabele (slika 2.1).

$T(U)$	A_1	A_2	...	A_n
P_1^n	λ_1^1	λ_1^2	...	λ_1^n
P_2^n	λ_2^1	λ_2^2	...	λ_2^n
...	...	...	...	...
P_k^n	λ_k^1	λ_k^2	...	λ_k^n

Slika 2.1.

Definicija 2.41. Interpretacija (***valuacija***) tabloa $T(U) = \{ P_i^n(\lambda_i^1,...,\lambda_i^n) \mid i \in \{1,...,k\}\}$, u oznaci $v(T(U))$, je relacija $v(T(U)) \in SAT(U)$, definisana funkcijom v: $T(U) \rightarrow DomU^n$, takvom da važi:

1. $(\forall i \in \{1,...,k\})(v(P_i^n(\lambda_i^1,...,\lambda_i^n)) = (v(\lambda_i^1),...,v(\lambda_i^n)) \in dom(A_1) \times ... \times dom(A_n)) \wedge$
2. $(\forall i_1, i_2 \in \{1,...,k\})(\forall j_1, j_2 \in \{1,...,n\})(\lambda_{i_1}^{j_1} = \lambda_{i_2}^{j_2} \Rightarrow v(\lambda_{i_1}^{j_1}) = v(\lambda_{i_2}^{j_2}))$.

Definicija 2.42. Projekcija tabloa $T(U) = \{ P_i^n(\lambda_i^1,...,\lambda_i^n) \mid i \in \{1,...,k\}\}$ na skup obeležja $X \subseteq U$, (takav da $|X| = k$), u oznaci $\Pi_X(T(U))$, predstavlja tablo, definisan na sledeći način:

$$\Pi_X(T(U)) = \{ Q_i^k(\lambda_i^1,...,\lambda_i^k) \mid (\exists P_i^n \in T(U))(\forall A_j \in X)(\lambda_i^j = f_{Q_i}(A_j) = f_{P_i}(A_j))\}.$$

Definicija 2.43. Izraz oblika $<T(U), T'(X)>$, pri čemu su $T(U) = \{ P_i^n(\lambda_i^1,...,\lambda_i^n) \mid i \in \{1,...,k\}\}$ i $T'(X) = \{ Q_i^m(x_i^1,\dots,x_i^m) \mid i \in \{1,...,l\}\}$ tabloi, takvi da važi

1. $X \subseteq U \wedge$
2. $(\forall Q_i^m(x_i^1,\dots,x_i^m) \in T'(X))(\forall A_j \in X)(x_i^j \in \Pi_{A_j}(T(U)))$,

se naziva ***Generalizovana T-zavisnost*** (***tgd***). **[PBG, Hu, SU]**

Skup svih mogućih *tgd* nad skupom obeležja U je skup $TGD(U) = \{<T(U), T'(X)> \mid X \subseteq U\}$. Može se primetiti da je skup $TGD(U)$ beskonačan, prebrojiv skup.

Ukoliko za proizvoljnu *tgd* $<T(U), T'(X)>$ važi da je $X = U$, onda se ona naziva ***potpuna*** *tgd* (*ftgd*). Za skup svih mogućih *ftgd* nad U će biti upotrebljena oznaka $FTGD(U)$. Ako za proizvoljnu *tgd* $<T(U), T'(X)>$ važi da je $X \subset U$, reč je o ***ugrađenoj*** *tgd* (*etgd*).

Ako za proizvoljnu *tgd* $<T(U), T'(X)>$ važi da je $|T'(X)| = 1$, takva *tgd* će biti prikazivana u obliku $<T(U), Q(x_1,...,x_m)>$.

Definicija 2.44. Za relaciju $r(U)$ i *tgd* $<T(U), T'(X)>$ važi da $r \models <T(U), T'(X)>$ ako

$$(\forall v: T(U) \rightarrow DomU^n)(v(T(U)) \subseteq r \Rightarrow v(T'(X)) \subseteq \Pi_X(r)).$$ **[PBG, Hu, SU]**

Definicija 2.45. Izraz oblika $<T(U), E>$, pri čemu je $T(U) = \{ P_i^n(\lambda_i^1,...,\lambda_i^n) \mid i \in \{1,...,k\}\}$ tablo, a $E = \{ Eq_{A_j}^i(\lambda_p^j, \lambda_q^j) \mid i \in \{1,...,l\}\}$ konačan skup predikata, takvih da važi

$$(\forall i \in \{1,...,l\})(A_j \in U \wedge \lambda_p^j \in \Pi_{A_j}(T(U)) \wedge \lambda_q^j \in \Pi_{A_j}(T(U)))$$

se naziva ***Generalizovana E-zavisnost*** (***egd***). **[PBG, Hu, SU]**

Skup svih mogućih *egd* nad skupom obeležja U se obeležava sa $EGD(U)$. Može se primetiti da je $EGD(U)$ beskonačan, prebrojiv skup.

Ako za proizvoljnu *egd* $<T(U), E>$ važi da je $|E| = 1$, takva *egd* će biti prikazivana u obliku $<T(U), Eq_{A_j}(\lambda_p^j, \lambda_q^j)>$.

Definicija 2.46. Za relaciju $r(U)$ i *egd* $<T(U), E>$ važi da $r \models <T(U), E>$ ako

$$(\forall v: T(U) \rightarrow DomU^n)(v(T(U)) \subseteq r \Rightarrow (\forall Eq_{A_j}(\lambda_p^j, \lambda_q^j) \in E)(v(\lambda_p^j) = v(\lambda_q^j))).$$ **[PBG, Hu]**

Pomoću *egd* i *tgd* se mogu prikazati svi ostali tipovi zavisnosti podataka (*fd*, *mvd*, *jd*, *ind* i *rd*), o čemu će više biti reči u narednom poglavlju. Skup svih generalizovanih zavisnosti (***gd***), u oznaci $GD(U)$, je skup $GD(U) = EGD(U) \cup TGD(U)$.

Definicija 2.47. Neka je dat skup *gd* $\Gamma \subseteq GD(U)$. ***Zatvarač*** skupa *gd* s obzirom na $GD(U)$, u oznaci Γ^*, je skup $\Gamma^* = \{\gamma \in GD(U) | \Gamma \models \gamma\}$. **[Hu]**

Definicija 2.48. Neka je data relacija $r(U)$ i skup *gd* $\Gamma \subseteq GD(U)$. ***Chase*** sekvenca s obzirom na Γ predstavlja konačan niz relacija $(r_0,...,r_n)$, takav da važi:

1. $r_0 = r$,
2. $(\forall i \in \{0,...,n-1\})(\exists \gamma \in \Gamma)(\exists j \in \{i+1,...,n\})(r_i \not\models \gamma \wedge r_j \models \gamma)$. **[Hu, PBG, M]**

Definicija 2.49. Neka je data chase sekvenca $(r_0,...,r_{i-1})$ $(i > 0)$ i neka $(\exists \gamma \in \Gamma)(r_{i-1} \not\models \gamma)$. ***Chase modifikacija*** relacije r_{i-1} (tj. sekvence $(r_0,...,r_{i-1})$) predstavlja formiranje relacije r_i (tj. sekvence $(r_0,...,r_{i-1}, r_i)$) na sledeći način:

1. Ako je $\gamma = <T(U), T'(U)> \in FTGD(U)$, pri čemu je $v(T(U)) \subseteq r_{i-1}$, tada je $r_i = r_{i-1} \cup v(T'(U))$.
2. Ako je $\gamma = <T(U), Eq_{A_j}(\lambda_p^j, \lambda_q^j)> \in EGD(U)$, pri čemu je $v(T(U)) \subseteq r_{i-1}$, tada je

$$r_i = \mu(r_{i-1}),$$

pri čemu je funkcija μ: $SAT(U) \rightarrow SAT(U)$ definisana na sledeći način:

$$(\forall t \in r_{i-1})(\forall A_k \in U)\left(\mu(t(A_k)) = \begin{cases} t(A_k), & A_k \neq A_j \vee t(A_j) \notin \{v(\lambda_p^j), v(\lambda_q^j)\} \\ \min(v(\lambda_p^j), v(\lambda_q^j)), & A_k = A_j \wedge t(A_k) \in \{v(\lambda_p^j), v(\lambda_q^j)\} \end{cases}\right).$$

[Hu]

Na osnovu definicija 2.48. i 2.49, definisan je postupak formiranja chase sekvence - chase algoritam. Ukoliko za chase sekvencu $(r_0,...,r_n)$ važi da se ne može dalje proširivati, onda se ona naziva ***maksimalnom***, i uvodi se oznaka $r_n = CH_\Gamma(r)$. Ukoliko za maksimalnu chase sekvencu važi da $(\forall \gamma \in \Gamma)(CH_\Gamma(r) \models \gamma)$ onda je $CH_\Gamma(r) \in SAT(U, \Gamma)$.

2.2. Polazne pretpostavke metodologije razvoja informacionog sistema

Projektovanje šeme BP je važan zadatak u razvoju informacionog sistema (IS). Kvalitet projekta šeme BP ostvaruje značajan uticaj na troškove izrade programske podrške i eksploatacione performanse IS. Verovatno najveće probleme u projektovanju šeme BP sa stotinak ili više šema relacija predstavljaju:

- definisanje ograničenja, koja će verno odslikavati pravila poslovanja u realnom sistemu i
- primena kompleksnih tehnika strukturiranja šeme BP.

S druge strane, izrada aplikacija, od kojih svaka sadrži nekoliko programa, predstavlja zadatak koji zahteva dosta vremena i koji uobičajeno obavlja veći broj ljudi.

Problem izrade kvalitetne šeme BP i kvalitetnih aplikacija IS-a, uz približno optimalno angažovanje materijalnih, ljudskih i vremenskih resursa, se rešava ili makar ublažava, upotrebom odgovarajućeg CASE alata. CASE alat, čiji je cilj da ublaži, ili u krajnjoj liniji reši navedene probleme, treba da, između ostalih, poseduje celine za:

- izradu konceptualne šeme BP,
- automatsko oblikovanje implementacione šeme BP i
- generisanje prototipa aplikacija.

Primena takvog CASE alata ima punog opravdanja u razvoju složenih IS, koji treba da automatizuju veći broj poslovnih funkcija preduzeća. Metodologija razvoja takvih IS treba da omogući paralelan i relativno nezavisan rad većeg broja projektanata. Iz tog i iz prethodno navedenih razloga, neophodno je da metodologija razvoja IS bude oslonjena na odgovarajući CASE alat, a to znači da mora biti prilagođena upotrebi konkretnog CASE alata. Koncepcija CASE alata, isto tako, mora da se uklapa u metodološki pristup razvoja IS-a. Drugim rečima, postoji obostrani uticaj na relaciji CASE alat - metodologija razvoja IS.

U nastavku će biti dat prikaz:

- metodoloških aspekata razvoja IS-a, oslonjenog na CASE alat namenjen za automatizovano projektovanje šeme BP i generisanje prototipa aplikacija i
- osnovne koncepcije CASE alata, koji će pri razvoju IS biti upotrebljen.

2.2.1. Metodološki aspekti razvoja IS

Posmatrani pristup razvoju IS je zasnovan na kombinaciji metodologija ***životnog ciklusa*** i ***prototipskog razvoja***. Za metodologiju projektovanja IS putem CASE alata za automatizovano generisanje šeme BP i prototipa aplikacija su bitni pojmovi **[MLK2**, **LM3**, **ML2]**:

- poslovna funkcija,
- zajednički dokument,
- matrica zajednički dokumenti/poslovne funkcije,
- zajednički činilac poslovanja,
- šema modula BP,
- šema BP informacionog sistema i
- programski podsistem.

Poslovna funkcija (***podsistem***) predstavlja veću, relativno nezavisnu, celinu realnog sistema, čiji je cilj obavljanje srodne grupe poslovnih zadataka. Primeri poslovnih funkcija: komercijalna funkcija, knjigovodstvo, planiranje i praćenje proizvodnje, upravljanje proizvodnjom, itd. Poslovne funkcije nisu međusobno nezavisne. U poslovnoj funkciji postoje procesi čije odvijanje zavisi od ishoda nekih procesa iz drugih poslovnih funkcija.

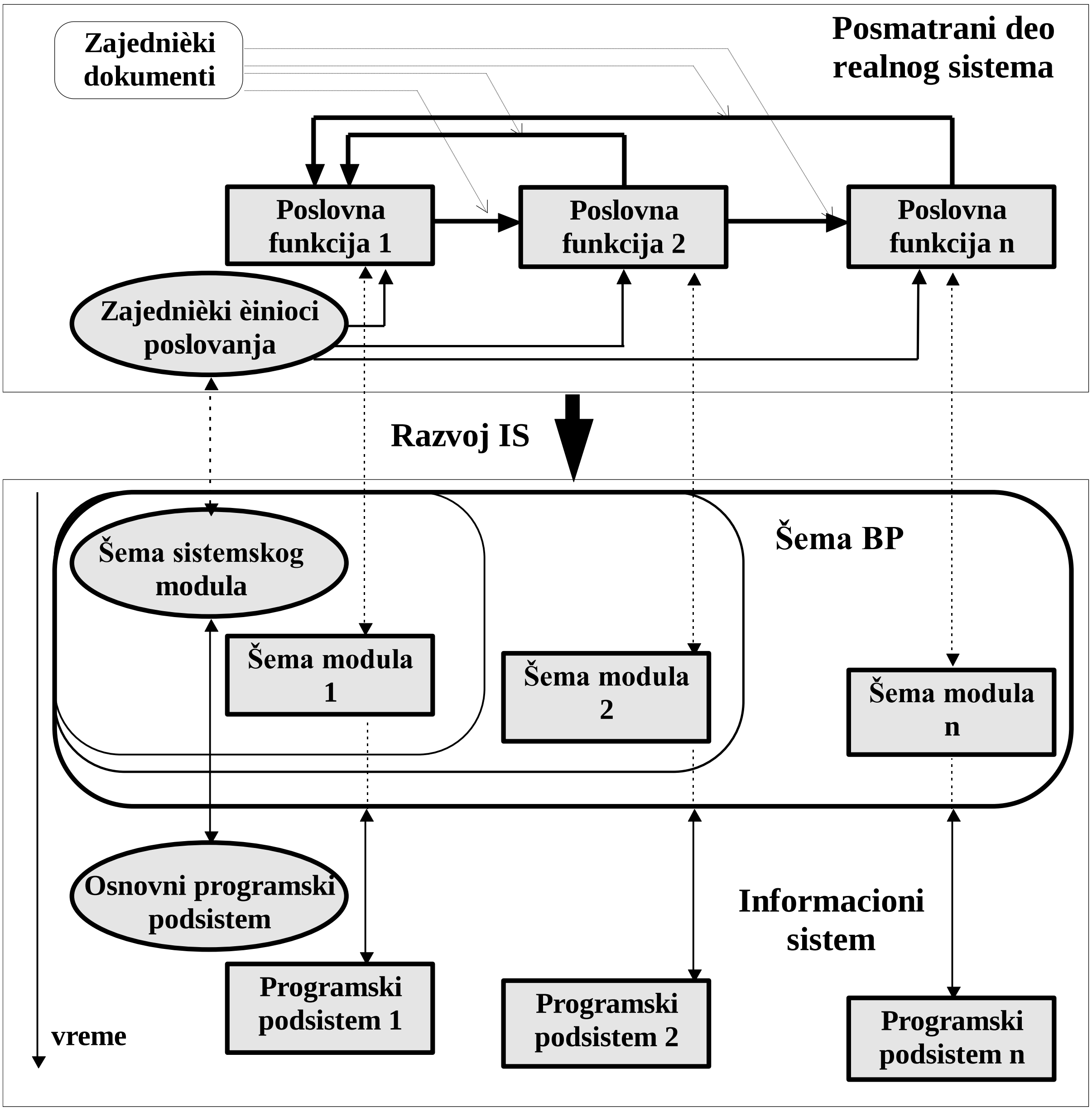
Zajednički dokumenat je takav nosilac informacija, koji se stvara u jednoj poslovnoj funkciji, a njegovi podaci se koriste, i eventualno modifikuju, u drugim poslovnim funkcijama. Putem zajedničkih dokumenata se reprezentuju tokovi podataka između poslovnih funkcija, saglasno međuzavisnostima procesa različitih poslovnih funkcija (slika 2.2). Poslovna funkcija, u kojoj se stvara neki dokument, naziva se izvornom funkcijom zajedničkog dokumenta. Kao primeri zajedničkih dokumenata, mogu se navesti: razni skladišni dokumenti, narudžbenice, radne liste, itd.

Matrica zajednički dokumenti/poslovne funkcije sadrži informacije o tome u kojoj poslovnoj funkciji se zajednički dokumenat stvara, a u kojim poslovnim funkcijama se koristi i, eventualno, modifikuje. Zaglavlje vrsta matrice sadrži nazive zajedničkih dokumenata, zaglavlje kolona sadrži nazive poslovnih funkcija, a elementi matrice uzimaju vrednosti iz skupa $\{\boldsymbol{S}(\text{stvara}),\ \boldsymbol{K}(\text{koristi}),\ \boldsymbol{M}(\text{modifikuje}),\ \Delta(\text{blanko})\}$. Matrica zajednički dokumenti/poslovne funkcije reprezentuje međuzavisnosti poslovnih funkcija realnog sistema.

Zajednički činilac poslovanja je činilac poslovanja, za čije podatke je karakteristično:

- da se koristi u više od jedne poslovne funkcije (slika 2.2) i
- da mu je životni vek relativno dugačak (duži od godinu dana).

Primeri zajedničkih činilaca poslovanja su materijali, rezervni delovi, osnovna sredstva, zaposleni, poslovni partneri, organizaciona i prostorna struktura realnog sistema, itd. Činioci poslovanja se, uobičajeno, reprezentuju putem kataloga činilaca.



Slika 2.2.

Šema modula BP (ili kraće ***šema modula***) predstavlja imenovani par $N_m(S_m, I_m)$, pri čemu je N_m naziv poslovne funkcije, S_m je skup šema relacija modula, a I_m je skup međurelacionih ograničenja modula. U postupku razvoja IS, svakoj poslovnoj funkciji se pridružuje jedna šema modula. Skupu zajedničkih činilaca poslovanja se, takođe, pridružuje jedna šema modula (slika 2.2), tzv. ***šema sistemskog modula*** (***jezgro***, ***zajednička šema modula***).

Šema BP informacionog sistema je imenovani par $N(S, I)$, gde je N naziv informacionog sistema, S je skup šema relacija u *3NF*, a I skup međurelacionih ograničenja. Svaka šema relacije skupa S predstavlja proširenje šema relacija iz različitih šema modula, koje sadrže bar jedan isti ključ. Skup ograničenja I je unija skupova ograničenja šema modula. Pojam proširenja šeme relacije će, u ovom radu, biti posebno obrađen.

Programski (***softverski***) ***podsistem*** predstavlja softverski proizvod koji se koristi pri izvršavanju radnih zadataka jedne poslovne funkcije. U postupku razvoja IS, svakoj poslovnoj funkciji se pridružuje jedan programski podsistem. Skupu zajedničkih činilaca poslovanja se, takođe, pridružuje jedan programski podsistem, tzv. ***osnovni programski podsistem*** (slika

2.2), koji služi za formiranje i ažuriranje kataloga zajedničkih činilaca poslovanja. Programski podsistemi se realizuju nad šemom BP IS-a.

Programski podsistem sadrži više ***aplikacija***. Jedna aplikacija se, u principu, vezuje za jedan tip radnog mesta i koristi se pri obavljanju radnih zadataka vezanih za taj tip radnog mesta.

Aplikacija se sastoji se od jednog, ili više ***programa***, strukturiranih u stablo ***menija aplikacije***. U postupku projektovanja softverskog podsistema, svakom programu se pridružuje ***programska specifikacija***, koja sadrži:

- sliku ekranske forme,
- podšemu i
- specifikaciju procedura.

Slikom ekranske forme se daje izgled ekrana, putem kojeg će korisnik realizovati svoje informacione zahteve.

Podšema programske specifikacije predstavlja imenovani par $N_p(S_p, I_p)$, gde je N_p ime programske specifikacije, S_p je skup šema relacija (dobijen na osnovu skupa S šeme BP $N(S, I)$), takav da svojim obeležjima pokriva sve informacione zahteve korisnika, koje datim programom treba razrešiti, a $I_p \subseteq I$ je skup međurelacionih ograničenja podšeme. **[Lu2]**

Specifikacija procedura definiše postupke koje program treba da izvršava.

Metodološki pristup razvoja IS, podržan CASE alatom za automatizovano generisanje šeme BP, prati poznate faze metodologije životnog ciklusa: strategija, snimanje i analiza, projektovanje, realizacija, testiranje, uvođenje, upotreba i održavanje. Jedan od nedostataka klasične metode životnog ciklusa je u tome što krajnji korisnik postaje svestan funkcionalnosti i upotrebljivosti aplikacije, tek kada je ona već uvedena u operativnu upotrebu. Kao posledica, pojavljuje se veliki broj zahteva za intervencijama, iako je aplikacija tek na početku korišćenja. Ovo može prouzrokovati potrebu da se aplikacija, ili čak cela šema modula, preprojektuje. U cilju prevazilaženja, ili makar ublažavanja navedenog problema, kombinuju se dobre strane metodologije životnog ciklusa s prototipskim pristupom. Ova kombinacija treba da ispuni dva cilja:

- dosledno primeniti metode i tehnike životnog ciklusa i
- pružiti na korišćenje krajnjem korisniku (odbacivi) prototip aplikacije, u ranoj fazi projektovanja šeme BP.

Prvi cilj znači da se faze i koraci metodologije životnog ciklusa ne smeju preskakati tokom projektovanja IS, dok drugi cilj znači da je potrebno njihovo značajno vremensko preklapanje.

Pomoću prototipa aplikacije, projektant konačno treba da dobije od krajnjeg korisnika precizne informacije, od važnosti za projektovanje šeme modula i programskog podsistema. S druge strane, krajnji korisnici treba da se priviknu na logiku i način funkcionisanja aplikacije još u ranim fazama razvoja IS. **[Da, ML2]**

U nastavku rada će, ukratko, biti dat opšti prikaz metodologije razvoja IS, bazirane na kombinaciji metodologija životnog ciklusa i prototipskog razvoja. Tokom prikaza faza životnog ciklusa, biće ukazivano i na specifičnosti prototipskog pristupa.

Faza strategije

Tokom ***faze strategije*** razvoja IS vrši se preliminarna identifikacija poslovnih funkcija i osnovnih struktura (funkcionalne, organizacione i prostorne) realnog sistema sa definisanjem korisničkih zahteva u cilju ugovaranja. Pored toga, definišu se:

- glavni plan i potplanovi projekta razvoja IS,
- organizacija i program rada na projektu i
- koncepcijsko idejno rešenje IS-a.

U ovoj fazi se utvrđuju prioriteti realizacije programskih podsistema za odgovarajuće poslovne funkcije. Prvi prototip, za koji se (saglasno prototipskom pristupu) može pretpostaviti da već postoji je prototip osnovnog programskog podsistema **[ML2]**. Razlog za ovu pretpostavku leži u činjenici da je osnovni programski podsistem, uglavnom, invarijantan s obzirom na različite realne sisteme. Krajnjim korisnicima se, već ovoj fazi, može obezbediti upotreba prototipa osnovnog programskog podsistema. Tokom njegove upotrebe, precizno se identifikuju svi zajednički činioci poslovanja sa osnovnim skupovima obeležja i procedurama za evidentiranje promene stanja zajedničkih činilaca poslovanja.

Implementacija osnovnog programskog podsistema u ovoj fazi predstavlja jedno preklapanje faze strategije i ostalih faza metodologije životnog ciklusa.

Faza snimanja i analize

Faza snimanja i analize se sprovodi na nivou svake poslovne funkcije. Tokom snimanja se:

- identifikuju izlazne informacije, koje korisnici poslovne funkcije očekuju od novog IS,
- identifikuju, ili kreiraju dokumenti i procedure, putem kojih se evidentiraju promene stanja realnog sistema,
- identifikuju, ili kreiraju dokumenti koji svojim sadržajem pokrivaju očekivane izlaze IS,
- identifikuju, ili kreiraju zajednički dokumenti, putem kojih se prikazuju ulazni i izlazni tokovi informacija između poslovnih funkcija i
- formira matrica zajednički dokumenti/poslovne funkcije.

Dokumenti koji se koriste za evidentiranje promene stanja realnog sistema mogu da posluže i kao osnov za formalizovanje ograničenja upotrebe podataka u realnom sistemu. Na osnovu dokumenata, tokom analize se formiraju tzv. tipovi formi. Tipovi formi se koriste za konceptualno modeliranje šeme modula BP. U tački 2.2.2. je dat detaljniji prikaz koncepta tipa forme.

Faza projektovanja i realizacije

U ***fazi projektovanja i realizacije*** izrađuje se, za svaku poslovnu funkciju, šema modula BP, polazeći od šeme sistemskog modula, koja prva mora biti gotova. Inicijalnim uključivanjem šeme sistemskog modula u sve ostale šeme modula i prethodnim identifikovanjem zajedničkih dokumenata (odnosno tzv. zajedničkih tipova formi, koji odgovaraju zajedničkim dokumentima) obezbeđuje se mogućnost paralelnog i relativno nezavisnog rada projektanata na različitim poslovnim funkcijama, što je još jedan preduslov primene prototipskog pristupa. Projektovanje šema modula se izvodi saglasno utvrđenim prioritetima razvoja programskih podsistema. Detaljniji opis postupka projektovanja šeme modula BP će biti dat u tački 2.2.2.3.

Šema BP informacionog sistema se dobija progresivnom ***integracijom*** šema modula poslovnih funkcija (slika 2.2). Postupak integracije šema modula u jedinstvenu šemu BP, preduslovi koje za sprovođenje postupka integracije treba obezbediti i metodološki aspekti primene postupka integracije su teme narednih poglavlja ovog rada.

U ovoj fazi se pristupa projektovanju programskog podsistema poslovne funkcije i generisanju prototipa programskog podsistema. Generisanje prototipova se, takođe, sprovodi saglasno prioritetima realizacije programskih podsistema poslovnih funkcija.

Proces izrade programskih podsistema se, kao što je već naglašeno, može paralelizovati, inicijalnim uključivanjem tipova formi sistemskog modula i zajedničkih tipova formi, u skup tipova formi datog programskog podsistema.

Prototipski pristup u fazama analize, projektovanja i realizacije

Cilj prototipskog pristupa je da se prototip aplikacije generiše još u ranoj fazi projektovanja. Na taj način, faze analize, projektovanja, realizacije, testiranja i upotrebe se preklapaju i, prema tome, "prolaze" najmanje dva puta. **[ML2]**

I Iteracija. Saglasno početnom sagledavanju problematike i eventualnom prethodnom znanju o razmatranom problemu, projektant formira preliminarni skup tipova formi, bez pretenzije da on bude kompletan. Na osnovu takvog skupa tipova formi, projektuje se prva verzija šeme modula BP i integriše se u tekuću šemu BP. Generišu se prototipovi aplikacija softverskog podsistema i uvode se u upotrebu.

II Iteracija (i naredne iteracije). Tokom testiranja i ***operativne upotrebe prototipa*** od strane korisnika, projektant prikuplja sve relevantne i detaljne informacije, potrebne za strukturiranje tipova formi i šeme modula. Na osnovu tih informacija, formira se konačni skup tipova formi, preprojektovanjem postojećih i dodavanjem novih tipova formi. Generiše se konača verzija šeme modula i i ponovo se integriše u jedinstvenu šemu BP. Potrebno je, takođe, prilagoditi ili ponovo generisati aplikacije.

Pored pozitivnih efekata, koji se očekuju od ovakve kombinacije metodologije životnog ciklusa i prototipskog pristupa, postoje i sledeći nedostaci, koje treba prevazići:

- prva i konačna verzija šeme modula mogu biti kolizione i
- ukoliko postoje kolizije, podaci, unešeni u bazu putem prototipa aplikacije, moraju biti restrukturirani tako da odgovaraju opisu integrisane šeme BP.

Problem kolizija šema modula je tema četvrtog poglavalja ovog rada. Restrukturiranje podataka koje je posledica postojanja kolizija se obavlja putem dostupnih alata konkretnog relacionog sistema za upravljanje bazama podataka (RSUBP).

Motivaciju za uvođenje prototipskog pristupa razvoja IS predstavlja problem da se snimanje realnog sistema ne može valjano obaviti na klasičan način, u kojem korisnik nema pred sobom neko konkretno rešenje aplikacija programskih podsistema. Prema **[Da]**, klasičan pristup realizacije softverskog proizvoda dovodi do toga da se zahtevi za značajnim intervencijama na već "gotovom" softverskom proizvodu pojavljuju kada je već 99% sredstava za razvoj tog prozivoda potrošeno. S druge strane, prototipski pristup treba da obezbedi da se intervencije na tekućoj verziji softverskog proizvoda vrše u situaciji kada je potrošeno samo 20% sredstava, namenjenih za razvoj softverskog proizvoda. Na taj način, u trenutku kada se potroši 99% sredstava, softverski prozivod u funkcionalnom i estetskom smislu mora biti "zaokružen", a korisnik zadovoljan načinom njegovog funkcionisanja.

Zahtev za obezbeđenjem operativnog korišćenja prototipa aplikacija programskog podsistema, odnosno potreba da se jednom uneti podaci putem prototipa ne odbacuju, dovodi do značajne kompleksnosti postupaka integracije šeme BP, a time i ovog rada u celini, o čemu će posebno biti reči u četvrtom poglavlju.

2.2.2. Osnovna koncepcija CASE alata za automatizovano generisanje šeme BP i prototipa aplikacija

Primena CASE alata, čiji je zadatak automatizovano generisanje šeme BP i prototipa aplikacija mora da započne odmah po završetku faze definisanja strategije razvoja IS, nakon što se precizno identifikuju zajednički činioci poslovanja, sa svojim skupovima obeležja. Cilj primene CASE alata u ovoj fazi jeste ***automatizovano generisanje*** narednih verzija:

- šeme sistemskog modula i
- prototipa osnovnog programskog podsistema.

CASE alat se, potom, upotrebljava u fazi analize svake poslovne funkcije realnog sistema u cilju automatizacije postupaka:

- generisanja šeme modula poslovne funkcije,
- integracije šeme modula poslovne funkcije u jedinstvenu šemu BP i
- generisanja prototipa programskog podsistema poslovne funkcije.

Primena CASE alata u fazi analize može da započne kada se identifikuju izlazne informacije, koje korisnici poslovne funkcije očekuju od novog IS, procedure i dokumenti, putem kojih se evidentiraju promene stanja poslovnog sistema i koji svojim sadržajem pokrivaju očekivane izlaze IS, kao i zajednički dokumenti i matrica zajednički dokumenti/poslovne funkcije.

Osnovni koncept CASE alata, prilagođenog pristupu razvoja IS-a, opisanom u tački 2.2.1, jeste koncept tipa forme. Pojam tipa forme se, u kraćim crtama, razmatra u tački 2.2.2.1, dok se detaljniji opis ovog koncepta može naći u **[MLK1**, **Lu2**, **LM1**, **MKu]**. Koncept tipa forme se, u nešto drugačijem obliku, pojavljuje u **[CMN**, **DL]**.

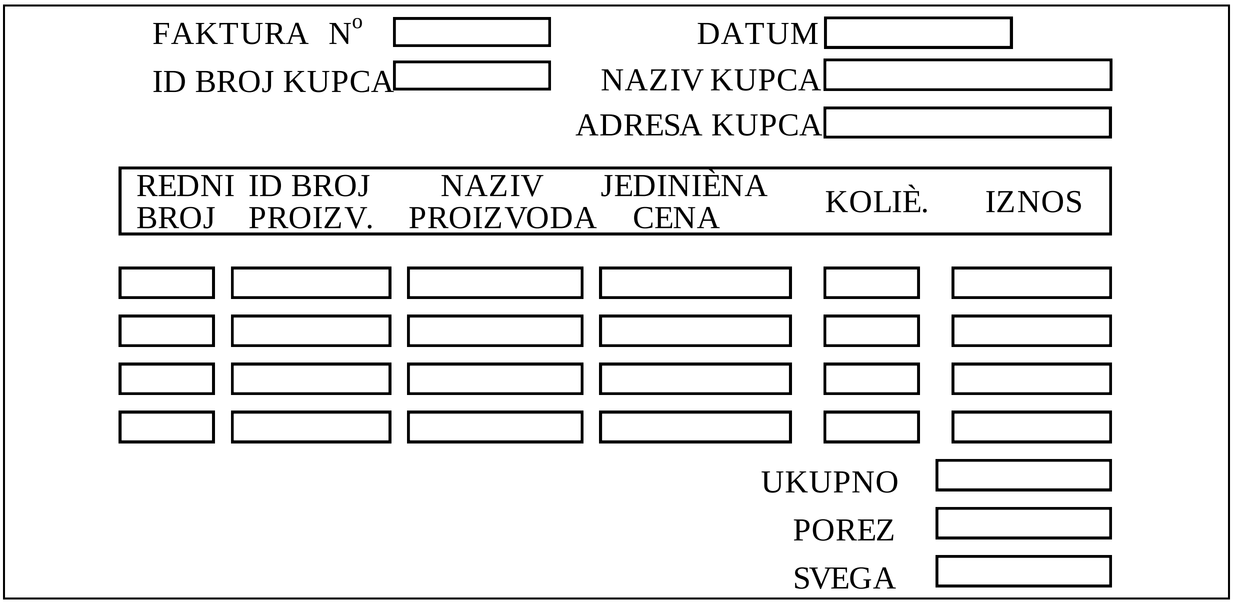
2.2.2.1. Koncept tipa forme

Komunikacija između korisnika i automatizovanog IS se realizuje putem ekranskih i štampanih formi. Te forme mogu predstavljati izvor ne samo za identifikaciju skupa obeležja šeme baze podataka, već i skupa ograničenja. Da bi forme predstavljale izvor za definisanje skupa ograničenja, potrebno je uvesti precizna pravila za njihovo struktuiranje i pravila za izvođenje zaključaka o egzistenziji ograničenja.

Primer 2.1. Na slici 2.3. je prikazan pojednostavljeni oblik forme pod nazivom *FAKTURA*. U prazna polja se upisuju podaci. Semantiku tih podataka definišu nazivi ispisani ispred ili iznad polja. Podaci sa ove forme se mogu grupisati u dve logičke celine. Jednu čine podaci o samoj fakturi sa zbirom, porezom i ukupnim iznosom, kao i podaci o kupcu. Ta celina se može nazvati *zaglavlje fakture*. Drugu celinu predstavljaju podaci o *stavkama fakture*. Ovakve logičke celine podataka se mogu nazvati objektima forme.

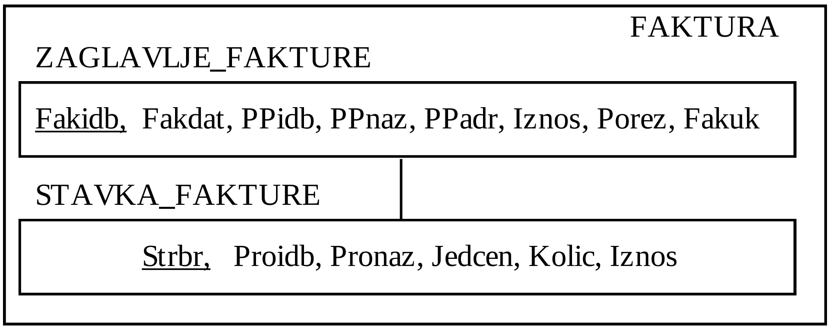
U principu, svaki podatak u objektu predstavlja jednu vrednost odgovarajućeg obeležja buduće šeme baze podataka. Nazivi tih obeležja se, u opštem slučaju, razlikuju od naziva polja na formi.

Generalizacijom analize sadržaja ekranskih i štampanih formi dolazi se do pojma tipa forme. Tip forme je struktura stabla čije čvorove predstavljaju tipovi objekata. Neka je $W(\mathcal{F})$ skup obeležja tipa forme sa nazivom $\mathcal{F}$. Svaki tip objekta tipa forme $\mathcal{F}$ je imenovani par $N(Q, \mathcal{K})$, gde je N naziv tipa objekta, $Q = \{A_1,...,A_k\} \subseteq W(\mathcal{F})$, a $\mathcal{K}$ skup ključeva tipa objekta.

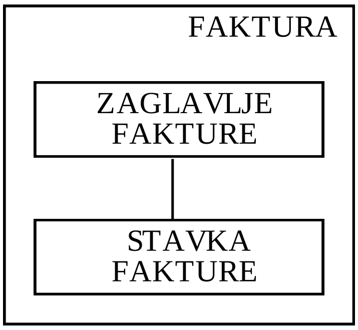


Slika 2.3.

Primer 2.2. Na slici 2.4. je prikazana geometrijska reprezentacija tipa forme, dobijenog generalizaciojm forme sa slike 2.3. Obeležja ključa svakog tipa objekta su podvučena. Na slici 2.5. je prikazano dalje uopštenje ovog tipa forme. □



Slika 2.4.



Slika 2.5.

Uvođenje pojma tipa forme zahteva da se definiše i pojam pojave tipa forme. Pojava tipa forme je struktura stabla, čiji koren sadrži jednu pojavu tipa objekta u korenu stabla tipa forme, a čvorove na svakom nižem nivou hijerarhije čini nula ili više pojava svakog tipa objekta na odgovarajućem nivou hijerarhije stabla tipa forme.

Nakon ovog, neformalnog razmatranja pojma tipa forme, slede definicije koncepata tipa forme i pojave tipa forme.

Definicija 2.50. Tip forme nad skupom obeležja $W(\mathcal{F})$ predstavlja imenovanu strukturu $\mathcal{F}(\mathcal{O}, \psi, C_{\mathcal{F}}, A_{\mathcal{F}}(x))$, pri čemu je:

- $\mathcal{O} = \{(Q_i, \mathcal{K}_o^i) \mid i \in \{1,...,l\}\}$ skup tipova objekata, $Q_i \subseteq W(\mathcal{F})$ skup obeležja, a $\mathcal{K}_o^i = \{K_j \subseteq Q_i \mid j \in \{1,...,m\}\}$ skup ključeva tipa objekta.
- $\psi \subseteq \mathcal{O} \times \mathcal{O}$ relacija, kojom se definiše ***struktura stabla tipa forme*** nad skupom tipova objekata. Važi da je $(N_i, N_j) \in \psi$ ukoliko postoji pojava tipa objekta N_i, za koju se vezuje više od jedne pojave tipa objekta N_j. (Definicijom 2.51. će biti formalizovani pojmovi: "pojava tipa objekta" i "pojava tipa forme".)
- $C_{\mathcal{F}}$ skup ograničenja, opisanih sledećim izrazima:

(2.1) $$\bigcup_{i=1}^{l} N_i = W(\mathcal{F})$$

(2.2) $$(\forall N_i, N_j \in \mathcal{O})(i \neq j \Leftrightarrow Q_i \cap Q_j = \emptyset)$$

(2.3) $$(\forall N_i \in \mathcal{O})(\forall A, B \in W(\mathcal{F}))((A \in Q_i \wedge B \in Q_i) \Leftrightarrow Ibp(\mathcal{F}, A, B)),$$

pri čemu je predikat $Ibp(\mathcal{F}, A, B)$) definisan na sledeći način:

$$(\forall p_{\mathcal{F}} \in SP_{\mathcal{F}})(|A(p_{\mathcal{F}})| = |B(p_{\mathcal{F}})|).$$

$SP_{\mathcal{F}}$ označava skup svih mogućih pojava tipa forme $\mathcal{F}$, a $|A(p_{\mathcal{F}})|$ označava broj pojava obeležja A, na pojavi $p_{\mathcal{F}} \in SP_{\mathcal{F}}$.

(2.4) $$(\forall N_i, N_j \in \mathcal{O})(((N_i, N_j) \in \psi \wedge (\forall K \in \mathcal{K}_o^j)) \Rightarrow Joo(N_i, N_j, K)),$$

pri čemu predikat $Joo(N_i, N_j, K)$ znači da ključ K jedinstveno određuje sve pojave tipa objekta N_j koje su direktno podređene jednoj pojavi tipa objekta N_i i ne postoji pravi podskup od K sa istim svojstvom.

Neka je $N_k \in \mathcal{O}$ koren strukture stabla tipa forme. Važi da je:

(2.5) $$(\forall K \in \mathcal{K}_o^k)(Jof(N_k, K)),$$

pri čemu predikat $Jof(N_k, K)$ znači da ključ K jedinstveno određuje sve pojave korenskog tipa objekta N_k, tj. jedinstveno određuje sve pojave tipa forme $\mathcal{F}$ i ne postoji pravi podskup od K sa istim svojstvom.

- $A_{\mathcal{F}}(x)$ predikat kojim se definiše način upotrebe tipa forme $\mathcal{F}$, pri čemu je $x \in \{u, r\}$. Ukoliko se tip forme upotrebljava i za ažuriranje i za upite, interpretacija predikata $A_{\mathcal{F}}(x)$ će biti $A_{\mathcal{F}}(u)$. Ukoliko se, međutim, tip forme koristi samo za izražavanje upita nad BP, interpretacija predikata će biti $A_{\mathcal{F}}(r)$. **[MLK1, Lu2]**

Na osnovu interpretacije predikata $A_{\mathcal{F}}(x)$, skup tipova formi se deli na klasu tipova formi za ažuriranje (određenu interpretacijom $A_{\mathcal{F}}(u)$) i klasu tipova formi za upite (određenu interpretacijom $A_{\mathcal{F}}(r)$). Skup tipova formi za ažuriranje će biti obeležen sa $\mathcal{SF}u$, dok će skup tipova formi za upite biti obeležen sa $\mathcal{SF}r$.

Definicija 2.51. Neka je dat tip forme $\mathcal{F}$.

1. Neka je $N_k(Q_k, \mathcal{K}_o^k)$ korenski objekat strukture tipa forme. ***Pojava tipa forme***, tj. ***pojava korenskog tipa objekta*** N_k je funkcija $p(N_k)$: $Q_k \rightarrow \bigcup_{A_i \in Q_k} dom(A_i)$, takva da važi:
$$(\forall A_i \in Q_k)(p(N_k)(A_i) \in dom(A_i)).$$
2. Neka je tip objekta $N_j(Q_j, \mathcal{K}_o^j)$ direktno podređen tipu objekta $N_i(Q_i, \mathcal{K}_o^i)$ u strukturi stabla tipa forme i neka je sa $P(N_i) = \{p_l(N_i) \mid l = 1,...,n\}$ označen skup svih pojava tipa objekta N_i, unutar jedne pojave tipa forme $\mathcal{F}$. ***Pojava tipa objekta*** N_j, saglasno jednoj pojavi tipa objekta $p_l(N_i) \in P(N_i)$, je funkcija $p(N_j, p_l(N_i))$: $Q_j \rightarrow \bigcup_{A_i \in Q_j} dom(A_i)$, takva da važi:
$$(\forall A_i \in Q_j)(p(N_j, p_l(N_i))(A_i) \in dom(A_i)).$$ **[Lu2, MLK2]**

2.2.2.2. Oblikovanje polaznog skupa ograničenja

Pomoću tipa forme se, implicitno, zadaju ograničenja na osnovu kojih će se projektovati šema modula. Neka ograničenja buduće šeme modula i šeme BP slede iz specifikacije domena i obeležja, kao primitivnih koncepata CASE alata. U ovom delu rada će, ukratko, biti prikazani tipovi ograničenja koji proizilaze iz specifikacije domena i obeležja i tipovi ograničenja koji se mogu iskazati pomoću tipova formi. Detaljnije o izvođenju zaključaka o egzistenciji ograničenja se može naći u **[MLK1**, **Lu2**, **MLK2**, **LKS]**.

Specifikacija domena i obeležja

Funkcija *dom*: $U \to \mathcal{D}$, data definicijom 2.1, iskazuje tzv. ***domensko ograničenje*** (***integritet domena***). Za svako obeležje $A \in U$ se putem funkcije *dom* definiše skup mogućih vrednosti. Pri tome, univerzalni skup obeležja predstavlja skup svih obeležja, definisanih tipovima formi iz klase za ažuriranje:

(2.6) $$U = \bigcup_{F \in SFu} W(F).$$

Definicija 2.52. Obeležja A_i, $A_j \in U$ se smatraju ***domenski kompatibilnim obeležjima***, u oznaci $dcomp(A_i, A_j)$, ako važi da je $dom(A_i) \cap dom(A_j) \neq \emptyset$.

Pored toga što nasleđuje ograničenja na skup dozvoljenih vrednosti, zadatih domenom, obeležje poseduje specifikaciju iz koje se mogu izvesti dodatna ograničenja. Specifikacija obeležja, između ostalog, treba da sadrži definicije:

- pripadnosti šemi BP,
- elementarnosti,
- rekurzivne funkcije izvođenja i
- funkcije preimenovanja.

Definicija 2.53. Specifikacija pripadnosti šemi BP obeležja $A \in U$, u oznaci $dbs(A)$, pri čemu važi $dbs(A) \in \{Y, N\}$, se zadaje na sledeći način:

- ako je A obeležje šeme BP, tada i samo tada je $dbs(A) = Y$,
- ako A nije obeležje šeme BP, tada i samo tada je $dbs(A) = N$. **[Lu2**, **LKS]**

Skup svih obeležja šeme BP će biti označen sa U_{BP}:

(2.7) $$U_{BP} = \{A \in U \mid dbs(A) = Y\}.$$

Definicija 2.54. Specifikacija elementarnosti obeležja $A \in U$, u oznaci $el(A)$, pri čemu važi $el(A) \in \{E, D, R\}$, se zadaje na sledeći način:

- ako je A ***elementarno*** obeležje, tada i samo tada je $el(A) = E$,
- ako je A ***izvedeno*** obeležje, tada i samo tada je $el(A) = D$ i
- ako je A ***preimenovano*** obeležje, tada i samo tada je $el(A) = R$. **[LKS]**

Elementarno obeležje je obeležje čija vrednost ne zavisi od vrednosti bilo kog drugog obeležja iz skupa U. Vrednost izvedenog obeležja se određuje na osnovu vrednosti nekih drugih obeležja iz univerzalnog skupa, dok se vrednost preimenovanog obeležja određuje na osnovu vrednosti drugog obeležja, čijim preimenovanjem je dato obeležje nastalo.

Svako elementarno obeležje mora biti sadržano u šemi BP:

(2.8) $$(\forall A \in U)(el(A) = E \Rightarrow dbs(A) = Y).$$ **[LKS]**

Definicija 2.55. Funkcija izvođenja obeležja je rekurzivna funkcija *der*: $U \to \mathcal{P}(U)$, pri čemu za svako obeležje $A \in U$ važi da:

- $el(A) \in \{E, R\} \Leftrightarrow der(A) = \emptyset$,
- $el(A) = D \Leftrightarrow der(A) \neq \emptyset$,

i važi da je graf, definisan funkcijom *der*: (U, φ), pri čemu je $\varphi = \{(A_i, A_j) \in U^2 \mid A_i \in der(A_j)\}$, ***aciklički usmereni graf***. **[Lu2**, **LKS]**

Definicija 2.56. Funkcija preimenovanja obeležja je funkcija ren: $\{A \in U \mid el(A) = R\} \rightarrow U$, pri čemu važi $dcomp(A, ren(A))$.

Funkcija *der* predstavlja osnov za formalizaciju tzv. pravila poslovanja, dok funkcija *ren* predstavlja osnov za formalizaciju onih referencijalnih integriteta buduće šeme BP koji su posledica uloge preimenovanog obeležja.

Specifikacija elementarnosti obeležja, funkcija izvođenja i funkcija preimenovanja predstavljaju semantičke koncepte modela podataka, jer ukazuju na međuzavisnosti semantika koje obeležja izražavaju. Koncepti elementarnosti i funkcije izvođenja nisu inicijalno uključeni u definiciju relacionog modela podataka.

Specifikacija obeležja na tipu forme

Prilikom izrade tipova formi iz klase za ažuriranje, za svako obeležje tipa forme se mora naglasiti da li se njegova vrednost obavezno zadaje pri ažurranju podataka, ili može ostati nedefinisana.

Definicija 2.57. Specifikacija obaveznosti obeležja $A \in W(\mathcal{F})$, na tipu forme $\mathcal{F} \in \mathcal{SF}u$, u oznaci $man(\mathcal{F}, A)$, pri čemu važi $man(\mathcal{F}, A) \in \{M, O\}$, se zadaje na sledeći način:

- ako se vrednost obeležja A mora ***obavezno*** zadati pri ažuriranju stanja BP putem tipa forme $\mathcal{F}$, tada i samo tada je $man(\mathcal{F}, A) = M$, i takvo obeležje se naziva ***obavezno*** obeležje,
- ako se vrednost obeležja A ne mora zadati pri ažuriranju stanja BP putem tipa forme $\mathcal{F}$, tada i samo tada je $man(\mathcal{F}, A) = O$, i takvo obeležje se naziva ***neobavezno*** (***opcionalno***).

Specifikacija obazveznosti predstavlja jedan od elemenata, na osnovu kojih se pri projektovanju šeme modula izvodi zaključak o dozvoli, ili zabrani nula vrednosti za obeležje u šemi relacije.

Pored ograničenja zadatih skupom $C_{\mathcal{F}}$ tipa forme $\mathcal{F}$, upotreba obeležja na tipu forme podrazumeva i poštovanje sledećih ograničenja:

- Ključna obeležja tipa forme moraju biti sadržana u šemi BP i moraju biti obavezna:

  (2.9) $$(\forall A \in K_p(\mathcal{F}))(dbs(A) = Y \wedge man(\mathcal{F}, A) = M),$$

  gde je $K_p(\mathcal{F}) = \{A \in W(\mathcal{F}) \mid (\exists N_i \in \mathcal{O})(\exists K_i \in \mathcal{K}_o^i)(A \in K_i)\}$.
- Sva obeležja tipa forme koja nisu sadržana u šemi BP, moraju biti preimenovana, ili izvedena:

  (2.10) $$(\forall A \in W(\mathcal{F}) \backslash U_{BP})(el(A) \in \{R, D\}).$$

Primena funkcija izvođenja i preimenovanja obeležja zahteva da se definiše i pojam proširenog skupa obeležja tipa forme. U sastav ovog skupa treba da uđu, pored obeležja tipa forme, i sva obeležja koja učestvuju u izvođenju, ili preimenovanju obeležja datog tipa forme.

Definicija 2.58. Prošireni skup obeležja tipa forme $\mathcal{F}$, u oznaci $W_{ex}(\mathcal{F})$, je skup obeležja, definisan rekurzivno:

1. $(\forall A \in W(\mathcal{F}))(A \in W_{ex}(\mathcal{F}) \wedge (el(A) = R \Rightarrow ren(A) \in W_{ex}(\mathcal{F})))$.
2. $(\forall A \in W_{ex}(\mathcal{F}))(el(A) = D \Rightarrow der(A) \subseteq W_{ex}(\mathcal{F}))$.
3. $W_{ex}(\mathcal{F})$ se dobija primenom konačno mnogo puta 1. i 2. **[Lu2]**

Na osnovu definicije 2.55, (pošto funkcija *der* definiše aciklički usmereni graf) garantuje se da će postupak formiranja proširenog skupa obeležja tipa forme, impliciran definicijom

2.58, okončati. Prošireni skup obeležja tipa forme je od važnosti za postupak projektovanja podšeme tipa forme **[Lu2]**.

Ograničenja koja se specificiraju putem tipova formi

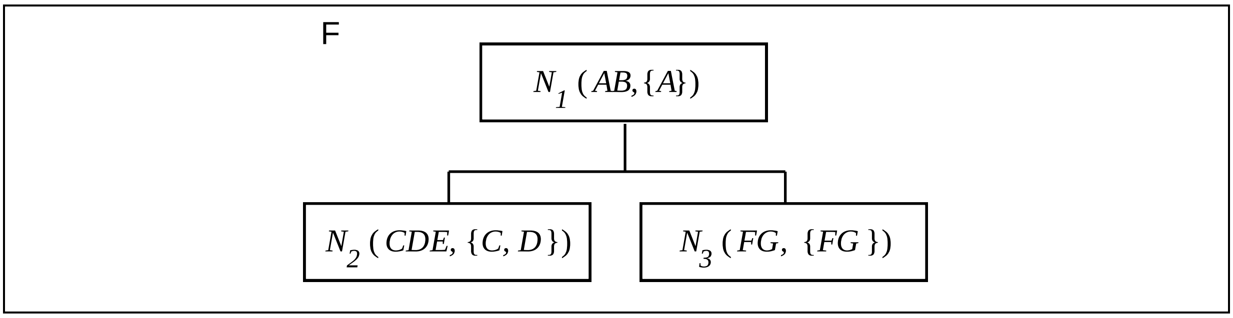
Na osnovu isprojektovanih tipova formi, u koje su ugrađena ograničenja uočena u realnom sistemu, moguće je izvesti zaključke o egzistenciji:

- funkcionalnih zavisnosti,
- ugrađenih višeznačnih i ugrađenih zavisnosti spoja i
- nefunkcionalnih odnosa.

Ključevi tipova objekata i struktura stabla tipova objekata nose informaciju o skupu *fd* $F(\mathcal{F})$, definisanog putem tipa forme $\mathcal{F}$. Obeležja svakog tipa objekta su funkcionalno zavisna od unije po jednog od ključeva tipova objekata koji se nalaze na putu od korenskog do datog tipa objekta. Sa $\mathcal{X}(N_i)$, gde je $N_i \in \mathcal{O}(\mathcal{F})$, će biti označen skup svih unija po jednog ključa od korenskog do objekta N_i. Skup *fd*, definisan tipom forme $\mathcal{F}$, je skup:

(2.11) $F(\mathcal{F}) = \{X \rightarrow A \in FD(W) \mid (\exists N_j \in \mathcal{O}(\mathcal{F}))(A \in Q(N_j) \wedge X \in \mathcal{X}(N_j))\}$. **[MLK1, Lu2]**

Primer 2.3. Na slici 2.6. je prikazan tip forme $\mathcal{F}$. Skup netrivijalnih *fd*, određen ovim tipom forme je $F(\mathcal{F}) = \{A \rightarrow B, AC \rightarrow D, AC \rightarrow E, AD \rightarrow C, AD \rightarrow E\}$.



Slika 2.6.

Odgovarajuća struktura tipa forme nosi informaciju o ugrađenoj zavisnosti spoja, koja važi u realnom sistemu. Insistira se na pojmu "ugrađena" jer se putem tipa forme iskazuju odnosi između podataka samo jednog dela realnog sistema.

Neka je dat skup listova strukture stabla tipa forme:

$$\mathcal{LO}(\mathcal{F}) = \{N_1, \ldots N_k\} \subseteq \mathcal{O}(\mathcal{F}).$$

Tipom forme $\mathcal{F}$ je izražena *jd*:

(2.12) $J(\mathcal{F}) = \rhd\lhd(X_1, \ldots, X_k)$, **[MLK1, Lu2]**

takva da se za svaki $N_i \in \mathcal{LO}(\mathcal{F})$, bira tačno jedan $X_i \in \mathcal{X}(N_i)$ $(i \in \{1, \ldots, k\})$.

Neka je *jd* $J_{ex}(\mathcal{F})$ definisana na sledeći način:

(2.13) $J_{ex}(\mathcal{F}) = \rhd\lhd\left(\{X_i \cup Q_i \mid (\exists N_i \in \mathcal{O})(X_i \in \mathcal{X}(N_i))\}\right).$

Može se pokazati **[Lu2]** da važi ekvivalencija ugrađenih *jd* $J(\mathcal{F})$ i $J_{ex}(\mathcal{F})$ s obzirom na skup *fd* tipa forme $F(\mathcal{F})$:

$$\{J(\mathcal{F})\} \cup F(\mathcal{F}) \equiv \{J_{ex}(\mathcal{F})\} \cup F(\mathcal{F}).$$

Primer 2.4. Jedna ugrađena *jd*, iskazana tipom forme $\mathcal{F}$ sa slike 2.6. je:

$$J(\mathcal{F}) = \rhd\lhd(AC, AFG) \text{ , a } J_{ex}(\mathcal{F}) = \rhd\lhd(AB, ACDE, AFG).$$

Svaki put dužine jedan, ili veće od jedan, koji počinje u korenu stabla, a završava se u listu, definiše jednu ***nefunkcionalnu zavisnost*** (*nfd*). Skup *nfd*, definisan tipom forme $\mathcal{F}$ je skup:

(2.14) $NF(\mathcal{F}) = \{X \rightarrow \theta_i \mid (\exists N_j \in \mathcal{LO}(\mathcal{F}))(X \in \mathcal{X}(N_j)) \wedge |X| > 1\}$, **[MLK2, Lu2]**

pri čemu $\theta_i \in \Theta = \{\theta_1,...,\theta_n\}$ predstavlja ***neinterpretirano***, nepostojeće, označeno obeležje. Ukoliko se, putem dva različita tipa forme, definiše isti nefunkcionalni odnos, onda će desne strane tog nefunkcionalnog odnosa biti različito označena neinterpretirana obeležja.

Nfd $X \rightarrow \theta_i$ ukazuje da postoji nefunkcionalni odnos između bar dva prava neprazna podskupa $Y, Z \subset X$, pri čemu je $Y \not\subseteq Z$ i $Z \not\subseteq Y$. Y i Z su ključevi dva različita tipa objekta. *Nfd* $X \rightarrow \theta_i$ iskazuje odnos više-više ($M : N$) između objekata, reprezentovanih sa Y i Z.

Neke *nfd* mogu biti pokrivene levom stranom odgovarajuće *fd*, definisane istim tipom forme. Zbog toga se definiše skup nepokrivenih *nfd*:

(2.15) $LNF(\mathcal{F}) = NF(\mathcal{F}) \setminus \{X \rightarrow \theta_i \in NF(\mathcal{F}) \mid (\exists f \in F(\mathcal{F}))(X = lhs(f))\}$.

Primer 2.5. Skupovi *nfd*, iskazani tipom forme $\mathcal{F}$ sa slike 2.6. su:

$$NF(\mathcal{F}) = \{AC \rightarrow \theta_1, AD \rightarrow \theta_2, AFG \rightarrow \theta_3\} \text{ i } LNF(\mathcal{F}) = \{AFG \rightarrow \theta_3\}.$$

2.2.2.3. Projektovanje šeme modula baze podataka informacionog sistema

Polazni skup ograničenja

Tipovi formi iz klase za ažuriranje nose informaciju o potrebnoj strukturi BP pa se putem ovih formi moraju definisati sva potrebna obeležja i sva ograničenja šeme BP. Polazni skup ograničenja, na osnovu kojeg se vrši projektovanje relacione šeme modula je skup:

(2.16) $ICS = \Gamma \cup NF \cup JF$,

pri čemu su Γ, NF i JF skupovi *fd*, *nfd* i *jd*, respektivno, izvedeni na osnovu tipova formi iz klase za ažuriranje:

(2.17) $$\Gamma = \bigcup_{\mathsf{F_i} \in \mathsf{SFu}} \left(F(\mathsf{F_i})\big|_{U_{BP}} \right),$$

(2.18) $$NF = \bigcup_{\mathsf{F_i} \in \mathsf{SFu}} \left(NF(\mathsf{F_i}) \right),$$

(2.19) $$JF = \bigcup_{\mathsf{F_i} \in \mathsf{SFu}} \left\{ J(\mathsf{F_i}) \right\}.$$

$F(\mathsf{F_i})\big|_{U_{BP}}$ je projekcija skupa *fd* tipa forme na skup obeležja šeme BP. Ova projekcija skupa *fd* je važna zbog toga što se na tipu forme mogu pojaviti izvedena, ili preimenovana obeležja iz skupa $U \setminus U_{BP}$, koja, prilikom projektovanja šeme modula, ne treba uzimati u obzir.

Normalizacija konceptualne šeme modula

Relaciona šema modula BP se automatizovano generiše postupkom normalizacije, zasnovanom na ***modifikovanom algoritmu sinteze*** **[MLK2, Lu2]**, koji se u ovoj tački ukratko opisuje. Ulaz za modifikovani algoritam sinteze predstavlja polazni skup ograničenja, definisan izrazom (2.16), a izlaz predstavlja relaciona šema modula u *3NF*. Klasični algoritam sinteze **[BB]** sadrži sledećih šest koraka:

1^0 redukcija levih strana *fd* (i *nfd*),
2^0 uklanjanje redundantnih *fd* i generisanje kanoničkog pokrivača,
3^0 particioniranje skupa zavisnosti saglasno istim levim stranama *fd* (i *nfd*),
4^0 spajanje skupova sa ekvivalentnim levim stranama,
5^0 eliminacija tranzitivnih zavisnosti i
6^0 formiranje skupa šema relacija.

Modifikovani algoritam sinteze, koji se u ovom pristupu upotrebljava, sadrži određena poboljšanja vezana za smanjenje složenosti algoritma (predložena u **[DM]**) i tretman skupa *nfd NF* u koracima 1^0, 3^0, 4^0 i 6^0. Algoritam je, takođe, proširen novim koracima:

7^0 pronalaženje kandidata za primarni ključ,
8^0 prostiranje primarnih ključeva i
9^0 generisanje skupa međurelacionih ograničenja.

U 1^0 koraku algoritma sinteze se vrši eliminacija osnovnih suvišnih obeležja kako iz levih strana parcijalnih *fd* skupa Γ, tako i iz *nfd* skupa *NF*:

(2.20) $$(\forall X\to B \in \Gamma)(\exists A \in X)(X \subseteq (X\setminus\{A\})^+_\Gamma \Rightarrow \Gamma \leftarrow (\Gamma\setminus\{X\to B\}) \cup \{X\setminus\{A\}\to B\}),$$

(2.21) $$(\forall X\to\theta_i \in NF)(\exists A \in X)(X \subseteq (X\setminus\{A\})^+_\Gamma \Rightarrow NF \leftarrow (NF\setminus\{X\to\theta_i\}) \cup \{X\setminus\{A\}\to\theta_i\}).$$

Ne postoji poseban korak elminacije ostalih suvišnih obeležja **[DM]**, već se ona uklanjaju u koraku 2^0, eliminacijom redundantnih zavisnosti.

CASE alat generiše poseban izveštaj o svim uklonjenim osnovnim suvišnim obeležjima i tipovima formi kojma je neko suvišno obeležje definisano. Razlog za ovo leži u činjenici da pojava osnovnih suvišnih obeležja može da ukaže na eventualne logičke greške, koje ne mogu biti razrešene formalnom zamenom skupa obeležja X, skupom obeležja $X\setminus\{A\}$, na levoj strani *fd* $X\to B$, gde je $B \in U_{BP}$ ili $B \in \Theta$. Jedino meritorno tumačenje nastale situacije može dati projektant, na osnovu generisanog izveštaja.

U koraku 2^0 se vrši uklanjanje redundantnih *fd* iz skupa Γ i formira se kanonički pokrivač $kp(\Gamma)$:

(2.22) $$(\forall X\to A \in \Gamma)(A \in X^+_{\Gamma\setminus\{X\to A\}} \Rightarrow \Gamma \leftarrow (\Gamma\setminus\{X\to A\})).$$

U koraku 3^0 se skup zavisnosti $NF \cup kp(\Gamma)$ particionira na podskupove $G(X_i)$, saglasno istim levim stranama: $G_X = \{G(X_i) \mid i \in \{1,...,n\}\}$, pri čemu važi:

(2.23) $$(\forall i \in \{1,...,n\})(G(X_i) = \{f \in NF \cup kp(\Gamma) \mid lhs(f) = X_i\}).$$

U koraku 4^0 se uniraju podskupovi $G(X_i)$ s ekvivalentnim levim stranama i generiše skup *fd* J koje su posledica ekvivalentnih levih strana:

(2.24) $$J = \{f \in \Gamma^+ \mid (\exists G(X_i), G(X_j) \in G_X)((X_i)^+_\Gamma = (X_j)^+_\Gamma \wedge lhs(f) = X_i \wedge rhs(f) = X_j)\},$$

(2.25) $$(\forall G(X_i) \in G_X)(\exists G(X_j) \in G_X)((X_i)^+_\Gamma = (X_j)^+_\Gamma \Rightarrow$$
$$(G(X_i) \leftarrow (G(X_i)\setminus\{X_i\to A \mid A \in X_j\}) \cup (G(X_j)\setminus\{X_j\to A \mid A \in X_i\})) \wedge (G_X \leftarrow G_X \setminus G(X_j)).$$

Korakom 5^0 se vrši eliminacija tranzitivnih zavisnosti iz podskupova $G(X_i)$:

(2.26) $$(\forall G(X_i) \in G_X)(\forall X\to A \in G(X_i))(A \in X^+_{(\Sigma\cup J)\setminus\{X\to A\}} \Rightarrow G(X_i) \leftarrow G(X_i)\setminus\{X\to A\}),$$

pri čemu je $\Sigma = \bigcup_{G(X_i) \in G_x} G(X_i)$.

U koraku 6^0 se formira skup šema relacija šeme modula, tako što od svakog nepraznog podskupa $G(X_i)$ nastaje po jedna šema relacije:

(2.27) $S = \{(R_i, \mathcal{K}_i) \mid (\exists G(X_i) \in G_X)($

$$\mathcal{K}_i = \{K \subseteq U \mid (\exists f \in G(X_i))(K = lhs(f))\} \wedge R_i = Syn(\Theta_i' \cup Q_i)\}),$$

pri čemu su Θ' i Q_i skupovi neinterpretiranih i realnih obeležja podskupa $G(X_i)$:

$$\Theta_i' = \{\theta_j \in \Theta \mid (\exists f \in G(X_i))(rhs(f) = \theta_j)\},$$

$$Q_i = \{A \in U_{BP} \mid (\exists f \in G(X_i))(rhs(f) = A)\},$$

a *Syn* je funkcija generisanja skupa obeležja šeme relacije:

$$R_i = Syn(\Theta_i' \cup Q_i) = Q_i \bigcup \left(\bigcup_{K \in \mathcal{K}_i} K \right).$$

Ukoliko važi da je $|\Theta_i'| \geq 1 \wedge Q_i \neq \emptyset$, ispituje se da li je svako neinterpretirano obeležje "pokriveno" nekim realnim obeležjem sa istog tipa forme:

(2.28) $(\forall \theta_j \in \Theta_i')(\exists A \in Q_i)(\exists \mathcal{F} \in \mathcal{SF}u)(\theta_j \in \Theta(\mathcal{F}) \wedge A \in W(\mathcal{F}))$,

pri čemu $\Theta(\mathcal{F}) \subseteq \Theta$ označava skup neinterpretiranih obeležja tipa forme $\mathcal{F}$ i važi da je:

$$(\forall \mathcal{F}_i, \mathcal{F}_j \in \mathcal{SF}u)(i \neq j \Rightarrow \Theta(\mathcal{F}_i) \cap \Theta(\mathcal{F}_j) = \emptyset).$$

Naime, neka je $X = YZ$, gde je$Y \neq \emptyset$, $Z \neq \emptyset$, i $Y \neq Z$. Svaki $\theta_j \in \Theta_i'$ može nositi informacije kako o istim, tako i o različitim ulogama Y i Z u njihovom nefunkcionalnom odnosu. Ove uloge mogu, ali ne moraju biti pokrivene nekim obeležjem iz Q_i. CASE alat treba da vodi računa o mestu nastanka θ_j i proverava da li je svaki $\theta_j \in \Theta_i'$ pokriven nekim $A \in Q_i$. Ukoliko to nije ispunjeno, tada CASE generiše izveštaj koji sadrži *nfd*, čija neinterpretirana obeležja nisu pokrivena realnim obeležjima, i nazive tipova formi koji definišu takve *nfd*. Na osnovu generisanog izveštaja, projektant treba da dâ meritorno tumačenje nastale situacije.

Motivacija za proširenje algoritma sinteze koracima 7^0 i 8^0 leži u činjenici da algoritam sinteze slučajnim izborom postavlja ekvivalentne ključeve jedne šeme realcije, kao strane ključeve u druge šeme relacija. Takvo prostiranje ključeva može uzrokovati probleme vezane za definisanje SQL upita, definisanje zavisnosti sadržavanja i dovesti do narušavanja nezavisnosti šeme BP (nezavisnosti, kako je definisana u **[Sc, Sa, HS]**). Da bi se ovi problemi eliminisali, potrebno je transformisati skup šema relacija $S = \{(R_i, \mathcal{K}_i) \mid i \in \{1,...,n\}\}$, dobijen algoritmom sinteze, u skup šema relacija $S' = \{(R_i', \mathcal{K}_i') \mid i \in \{1,...,n\}\}$, takav da je:

- $\bigcup_{R_i \in S} R_i = \bigcup_{R_i' \in S'} R_i'$ *),
- $\Gamma_S \equiv \Gamma_{S'}$, pri čemu je:

  $$\Gamma_S = \bigcup_{R_i \in S} \Gamma_i \ \text{ i } \ \Gamma_{S'} = \bigcup_{R_i' \in S'} \Gamma_i',$$

  gde je:

  $\Gamma_i = \Gamma(R_i) = \{X \rightarrow A \mid AX \subseteq R_i \wedge X \in \mathcal{K}_i\}$ i $\Gamma_i' = \Gamma'(R_i') = \{X \rightarrow A \mid AX \subseteq R_i' \wedge X \in \mathcal{K}_i'\}$,
- da za svaku šemu relacije $(R_i', \mathcal{K}_i') \in S'$, ako je to moguće, važi da se samo pravi, ili nepravi podskup jednog od njenih ekvivalentnih ključeva (primarni ključ) javlja kao podskup skupa obeležja neke druge šeme relacije iz S' :

---

*) Formula $\forall (R_i, \mathcal{K}_i) \in S$ æe, radi lakše èitljivosti, u nekim sluèajevima biti skraæeno data kao $\forall R_i \in S$.

(2.29) $(\forall(R_j', \mathcal{K}_j') \in S')(i \neq j \wedge (R_i)^+_\Gamma \subseteq (R_j)^+_\Gamma \Rightarrow (\exists! K \in \mathcal{K}_i')(K \subseteq (R_j)^+_{\Gamma_{S'} \setminus \Gamma_i'})),$

- skup S' je u *3NF*.

U slučaju da uslov (2.29) nije ispunjen za svaku šemu relacije $(R_i', \mathcal{K}_i') \in S'$, CASE alat generiše izveštaj u kojem navodi sve šeme relacija za koje uslov (2.29) ne važi i navodi šeme relacija zbog kojih se ne može izvršiti prostiranje primarnog ključa date šeme relacije. Pored toga, automatski se ispituje da li je šema modula BP nezavisna, pri čemu se uslov nezavisnosti izražava putem sledeće definicije.

Definicija 2.59. Šema baze podataka (S, I) je ***nezavisna***, ako važi

(2.30) $(\forall(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j) \in S)(\forall K \in \mathcal{K}_i)((R_i \setminus K) \cap (R_j)^+_{\Gamma_S \setminus \Gamma_i} = \varnothing).$ **[Sc]**

Ukoliko uslov nezavisnosti nije ispunjen, generiše se izveštaj, u kojem se navode parovi šema relacija koji ne ispunjavaju navedeni uslov i navodi se da li je (prema **[Sc]**) reč o **A** tipu, ili **B** tipu zavisnosti. U slučaju egzistencije A, ili B tipa zavisnosti, potrebno je definisati posebna međurelaciona ograničenja, koja će se kontrolisati pri ažuriranju kritičnih šema relacija (zbog kojih je narušena nezavisnost šeme). Ova tema je detaljno obrađena u radovima **[MR1]**, **[MR2]** i **[Ri]**.

U koraku 9^0 se formira skup međurelacionih ograničenja šeme modula i za svako obeležje svih šema relacija iz skupa S', definiše se dozvola (zabrana) nula vrednosti.

Osnovu za generisanje većine međurelacionih ograničenja predstavlja tzv. graf zatvaranja, koji se formira neposredno po generisanju skupa šema relacija šeme modula.

Definicija 2.60. Osnovni graf zatvaranja šeme BP (S, I) je graf $\mathcal{G} = (S, \rho)$, pri čemu je relacija ρ definisana na sledeći način:

(2.31) $\rho = \{(R_i, R_j) \in S^2 \mid R_j \subseteq (R_i)^+_\Gamma \wedge i \neq j \wedge$
$\neg(\exists R_k \in S)(k \neq i \wedge k \neq j \Rightarrow R_j \subseteq (R_k)^+_\Gamma \wedge R_k \subseteq (R_i)^+_\Gamma)\}$ **[MK, Lu2]**

Tranzitivni zatvarač relacije ρ, u oznaci $\tilde{\rho}$, je relacija definisana na sledeći način:

$$\tilde{\rho} = \{(R_i, R_j) \in S^2 \mid R_j \subseteq (R_i)^+_\Gamma\}.$$

Struktura $\tilde{\mathcal{G}} = (S, \tilde{\rho})$ se naziva tranzitivni graf zatvaranja šeme BP.

Na osnovu definicije osnovnog grafa zatvaranja, sledi da je $\mathcal{G}$ aciklički, usmereni graf. Osnovni graf zatvaranja definiše skup referencijalnih integriteta:

(2.32) $RI(\mathcal{G}) = \{R_i[K_j] \subseteq R_j[K_j] \mid (R_i, R_j) \in \rho \wedge K_j \subseteq R_i \wedge K_j \in \mathcal{K}_j\},$

i skup pravila poslovanja:

(2.33) $BR(\mathcal{G}) = \{\rhd\lhd(R_i, R_{i_1}, ..., R_{i_m})[K_j] \subseteq R_j[K_j] \mid (R_i, R_j) \in \rho \wedge K_j \not\subseteq R_i \wedge$
$MinCl(K_j, R_i, R_{i_1}, ..., R_{i_m}) \wedge m \geq 1 \wedge K_j \in \mathcal{K}_j\},$

pri čemu se predikat $MinCl(K_j, R_i, R_{i_1}, ..., R_{i_m})$ interpretira na sledeći način:

$$(\forall k \in \{i_1, ..., i_m\})(R_k \subseteq (R_i)^+_{\Gamma \setminus \Gamma_j} \wedge K_j \not\subseteq (R_i R_{i_1} \dots R_{i_m}) \setminus R_k) \wedge K_j \subseteq R_i R_{i_1} \dots R_{i_m}.$$

Na osnovu tipova formi iz skupa $\mathcal{SF}u$ i grafa $\mathcal{G}$, automatski se generiše skup zavisnosti sadržavanja:

(2.34) $ID(\mathcal{G}, \mathcal{SF}u) = \{R_j[K_j] \subseteq R_i[K_j] \mid (R_i, R_j) \in \rho \wedge$
$\wedge (\exists K_i \in \mathcal{K}_i)(K_j \subset K_i) \wedge \neg Fexst(\mathcal{SF}u, K_j)\},$

pri čemu se predikat $Fexst(\mathcal{SF}u, K_j)$ interpretira na sledeći način:

$$(\exists \mathcal{F}_j \in \mathcal{SF}u)(\exists N_j \in \mathcal{LO}(\mathcal{F}_j))(K_j \in \mathcal{X}(N_j)),$$

gde je $\mathcal{LO}(\mathcal{F}_j)$ skup listova strukture stabla tipa forme, a $\mathcal{X}(N_j)$ skup unija po jednog ključa od korenskog do tipa objekta N_j.

Pored automatskog generisanja skupa *ind* $ID(\mathcal{G}, \mathcal{SF}u)$, projektant interaktivno zadaje skup *ind*:

(2.35) $$ID(\mathcal{G}, desig) = \{R_j[K_j] \subseteq R_i[K_j] \mid (R_i, R_j) \in \rho \wedge (\forall K_i \in \mathcal{K}_i)(K_j \not\subseteq K_i) \wedge K_j \subseteq R_i \wedge desig(R_i, R_j)\}.$$

Interpretacija predikata $desig(R_i, R_j)$ je $I(desig(R_i, R_j)) = true$, ako i samo ako je projektant, u interakciji s CASE alatom, označio da postoji zavisnost sadržavanja između šema relacija $(R_i, \mathcal{K}_i)$ i $(R_j, \mathcal{K}_j)$.

Na osnovu funkcije preimenovanja obeležja i specifikacije pripadnosti šemi BP, izvodi se zaključak o egzistenciji skupa referencijalnih integriteta:

(2.36) $$RI_r(ren, dbs) = \{R_i[X_j] \subseteq R_j[K_j] \mid (\forall A \in X_j)(dbs(A) = Y \wedge (A \in K_j \vee (\exists B \in K_j)(A = ren(B)))) \wedge (R_i = R_j \Rightarrow X_j \notin \mathcal{K}_j)\}.$$

Funkcija izvođenja obeležja *der* u kombinaciji sa specifikacijom pripadnosti šemi BP *dbs*, može nositi informacije o različitim tipovima pravila poslovanja. Da bi se takva pravila poslovanja mogla formalno iskazati, trebalo bi uvesti posebnu specifikaciju funkcije, kojom se izražava ***način*** izračunavanja vrednosti nekog izvedenog obeležja A, na osnovu pojedinačnih vrednosti, ili skupova vrednosti obeležja iz skupa $der(A)$. To, međutim, izlazi iz okvira ovog rada, tako da ćemo samo pretpostaviti da postoji skup takvih pravila poslovanja i označićemo ga sa $BR(der, dbs)$.

U slučaju da sintetizovana šema narušava uslov nezavisnosti dat izrazom (2.30), definišu se posebna međurelaciona ograničenja (u nekim slučajevima su to referencijalni integriteti), koja će se kontrolisati prilikom ažuriranja relacija, zbog kojih je narušen uslov nezavisnosti. Ako za $(R_i, \mathcal{K}_i)$, $(R_j, \mathcal{K}_j) \in S$ važi $(R_i \setminus K) \cap (R_j)^+_{\Gamma_S \setminus \Gamma_i} \neq \emptyset$, onda međurelaciono ograničenje treba da obuhvati kako šeme relacija $(R_i, \mathcal{K}_i)$, $(R_j, \mathcal{K}_j)$, tako i sve šeme iz grafa zatvaranja $\mathcal{G}$, koje su podređene šemi relacije $(R_j, \mathcal{K}_j)$ i potrebne su da bi se ostvario uslov $(R_i \setminus K) \cap (R_j)^+_{\Gamma_S \setminus \Gamma_i} \neq \emptyset$. Problem određivanja takvih međurelacionih ograničenja je detaljnije obrađen u **[MR2]**, a za potrebe ovog rada će biti pretpostavljeno da postoji skup takvih ograničenja $DR(\mathcal{G}, dep)$.

Skup međurelacionih ograničenja šeme modula I predstavlja uniju:

(2.37) $$I = RI(\mathcal{G}) \cup BR(\mathcal{G}) \cup ID(\mathcal{G}, \mathcal{SF}u) \cup ID(\mathcal{G}, desig) \cup RI_r(ren, dbs) \cup BR(der, dbs) \cup DR(\mathcal{G}, dep).$$ **[GLM]**

Definicija 2.61. Specifikacija dozvole nula vrednosti za obeležje $A \in U$, nad šemom BP (S, I), u oznaci $null(R_i, A)$, pri čemu važi:

$$(\forall A \in U)(\forall R_i \in S)(A \in R_i \Leftrightarrow null(R_i, A) \in \{Y, N\}),$$

zadaje se na sledeći način:

1. ako su za obeležje A dozvoljene nula vrednosti unutar pojave nad šemom relacije $(R_i, \mathcal{K}_i)$, tada i samo tada je $null(R_i, A) = Y$, a
2. ako za obeležje A nisu dozvoljene nula vrednosti unutar pojave nad šemom relacije $(R_i, \mathcal{K}_i)$, tada i samo tada je $null(R_i, A) = N$. **[GLM]**

CASE alat automatski inicijalizuje specifikaciju dozvole nula vrednosti *null*, koristeći se sledećim pravilima zaključivanja: **[GLM]**

- $(\forall R_i \in S)(\forall A \in K_p(R_i))(null(R_i, A) = N)$, gde je K_p skup primarnih obeležja šeme relacije:
$$K_p(R_i) = \bigcup_{K_j \in \mathrm{K}_i} K_j .$$
- $$(\forall (R_i, \mathcal{K}_i) \in S)(\forall A \in R_i \backslash K_p(R_i))(\exists \mathcal{F} \in \mathcal{SF}u)(\exists N_i \in \mathcal{O}(\mathcal{F}))$$ $$((\mathcal{K}_i \cap \mathcal{X}(N_i) \neq \varnothing \wedge (A \notin Q(N_i) \vee man(\mathcal{F}, A) = O)) \Rightarrow null(R_i, A) = Y).$$
- $$(\forall R_i \in S)(\forall A \in R_i \backslash K_p(R_i))(\forall \mathcal{F} \in \mathcal{SF}u)(\forall N_i \in \mathcal{O}(\mathcal{F}))$$ $$((\mathcal{K}_i \cap \mathcal{X}(N_i) \neq \varnothing \Rightarrow (A \in Q(N_i) \wedge man(\mathcal{F}, A) = M)) \Rightarrow null(R_i, A) = N).$$

Specifikacija dozvole nula vrednosti postaje sastavni deo lokalnog skupa ograničenja svake šeme relacije, isprojektovane šeme modula BP.

3. Poglavlje

Familije generalizovanih zavisnosti

U prethodnom poglavlju su date definicije pojmova generalizovane T-zavisnosti (*tgd*) i generalizovane E-zavisnosti (*egd*). Pomoću *tgd* i *egd* se, na tabelarni način, koristeći simboličke domene, iskazuju ograničenja podataka realnog sistema. U ovom poglavlju će biti detaljnije diskutovane definicije interpretacije generalizovanih zavisnosti i načini prikaza ostalih tipova zavisnosti pomoću *tgd* i *egd*. Potom će biti uvedeni pojmovi ranga i konačne specifikabilnosti familije pojava, da bi nakon toga, bio razmatran problem projekcije i prirodnog spoja familije pojava s obzirom na određene tipove zavisnosti podataka. Problem projekcije i prirodnog spoja familija pojava predstavlja osnov za rešavanje problema integracije šema relacionih BP.

$T(U)$	A	B	C	D	E
P_1	x_1	y_1	z_1	u_1	w_1
P_2	x_1	y_1	z_2	u_2	w_2

Slika 3.1.

r_1	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
t_4	a_1	b_1	c_1	d_2	e_1

r_2	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
t_4	a_1	b_1	c_2	d_2	e_2

Slika 3.2.

Primer 3.1. Neka je dat skup obeležja $U = ABCDE$. Na slici 3.1. je prikazan tablo $T(U)$, a na slici 3.2. relacije $r_1, r_2 \in SAT(U)$. Za dati skup generalizovanih zavisnosti $\Gamma = \{\gamma_1, \gamma_2\}$, koji se sastoji iz *tgd* $\gamma_1 : \langle T(U), Q(x_1, y_1, z_1, u_1, w_2)\rangle$ i *egd* $\gamma_2 : \langle T(U), Eq_C(z_1, z_2)\rangle$, može se zaključiti da važi $r_1 \in SAT(U, \Gamma)$ i $r_2 \notin SAT(U, \Gamma)$, pri čemu je $r_2 \not\models \gamma_1$ i $r_2 \not\models \gamma_2$.

Činjenica $r_1 \in SAT(U, \Gamma)$ se može dokazati korišćenjem definicija 2.44. i 2.46. Za svaku interpretaciju v tabloa $T(U)$, takvu da je $v(T(U)) \subseteq r_1$ treba utvrditi da važi $v(x_1, y_1, z_1, u_1, w_2) \in r_1$ i $v(z_1) = v(z_2)$. Postoji osam mogućih interpretacija tabloa $T(U)$, za koje važi da je $v(T(U)) \subseteq r_1$, i one su prikazane, tabelarno, na slici 3.3. Za slučaj interpretacije $v_1(T(U))$, $v(x_1, y_1, z_1, u_1, w_2) = t_3 \in r_1$. Analogno, i za ostalih sedam interpretacija se može zaključiti da važi $v(x_1, y_1, z_1, u_1, w_2) \in r_1$. Za svaku od interpretacija $v_1, v_2, v_3, v_4, v_5, v_6, v_7$ i v_8 se, takođe, može zaključiti da važi $v(z_1) = v(z_2)$.

$v_1(T(U))$	A	B	C	D	E
$P_1 (P_2)$	a_1	b_1	c_1	d_1	e_1
$P_2 (P_1)$	a_1	b_1	c_1	d_2	e_2

$v_2(T(U))$	A	B	C	D	E
$P_1 (P_2)$	a_1	b_1	c_1	d_1	e_2
$P_2 (P_1)$	a_1	b_1	c_1	d_2	e_1

$v_3(T(U))$	A	B	C	D	E
$P_1 (P_2)$	a_1	b_1	c_1	d_1	e_1
$P_2 (P_1)$	a_1	b_1	c_1	d_1	e_2

$v_4(T(U))$	A	B	C	D	E
$P_1 (P_2)$	a_1	b_1	c_1	d_1	e_2
$P_2 (P_1)$	a_1	b_1	c_1	d_2	e_2

$v_5(T(U))$	A	B	C	D	E
$P_1 (P_2)$	a_1	b_1	c_1	d_1	e_1
$P_2 (P_1)$	a_1	b_1	c_1	d_2	e_1

$v_6(T(U))$	A	B	C	D	E
$P_1 (P_2)$	a_1	b_1	c_1	d_2	e_1
$P_2 (P_1)$	a_1	b_1	c_1	d_2	e_2

$v_7(T(U))$	A	B	C	D	E
P_1, P_2	a_1	b_1	c_1	d_1	e_1

$v_8(T(U))$	A	B	C	D	E
P_1, P_2	a_1	b_1	c_1	d_2	e_2

Slika 3.3.

Činjenice $r_2 \not\models \gamma_1$ i $r_2 \not\models \gamma_2$ se mogu dokazati korišćenjem negacija definicija 2.44. i 2.46. To znači da je dovoljno naći po jednu interpretaciju tabloa $T(U)$, takvu da je $v(T(U)) \subseteq r_2$, a da ne važi $v(x_1, y_1, z_1, u_1, w_2) \in r_2$, odnosno $v(z_1) = v(z_2)$. Interpretacija $v_1(T(U))$, pri čemu je $v_1(P_1) = (a_1, b_1, c_1, d_2, e_2)$, a $v_1(P_2) = (a_1, b_1, c_1, d_1, e_1)$, obara *tgd* γ_1, pošto $v(x_1, y_1, z_1, u_1, w_2) = (a_1, b_1, c_1, d_2, e_1) \notin r_2$. Interpretacija $v_9(T(U)) \subseteq r_2$, koja obara *egd* γ_2, budući da je $v_9(z_1) \neq v_9(z_2)$, prikazana je na slici 3.4.

$v_9(T(U))$	A	B	C	D	E
P_1	a_1	b_1	c_1	d_1	e_1
P_2	a_1	b_1	c_2	d_2	e_2

Slika 3.4.

Primer 3.2. Neka je dat skup obeležja $U = ABCDE$, relacija $r_2 \in SAT(U)$ i skup *gd* $\Gamma = \{\gamma_1, \gamma_2\}$, iz prethodnog primera. Na slici 3.5. su prikazane relacije chase sekvence ($r_2, r_{21}, r_{22}, r_{23}, r_{24}, r_{25}, r_{26}, r_{27}, r_{28}$). Relacije r_{21}, r_{22}, r_{23} i r_{24} su nastale modifikacijama relacije r_2 po *tgd* γ_1, dok su r_{25}, r_{26}, r_{27} i r_{28} nastale modifikacijama po *egd* γ_2. Može se zaključiti da je $r_{28} = CH_{\Gamma}(r_2)$, što znači da je zadovoljeno $r_{28} \in SAT(U, \Gamma)$.

r_{21}	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
	a_1	b_1	c_1	d_2	e_1
t_4	a_1	b_1	c_2	d_2	e_2

r_{22}	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
	a_1	b_1	c_1	d_2	e_1
t_4	a_1	b_1	c_2	d_2	e_2
	a_1	b_1	c_2	d_2	e_1

r_{23}	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
	a_1	b_1	c_1	d_2	e_1
t_4	a_1	b_1	c_2	d_2	e_2
	a_1	b_1	c_2	d_2	e_1
	a_1	b_1	c_2	d_1	e_1

r_{24}	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
	a_1	b_1	c_1	d_2	e_1
t_4	a_1	b_1	c_2	d_2	e_2
	a_1	b_1	c_2	d_2	e_1
	a_1	b_1	c_2	d_1	e_1
	a_1	b_1	c_2	d_1	e_2

r_{25}	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
	a_1	b_1	c_1	d_2	e_1
t_4	a_1	b_1	c_2	d_2	e_2
	a_1	b_1	c_2	d_2	e_1
	a_1	b_1	c_2	d_1	e_2

r_{26}	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
	a_1	b_1	c_1	d_2	e_1
t_4	a_1	b_1	c_2	d_2	e_2
	a_1	b_1	c_2	d_2	e_1

r_{27}	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
	a_1	b_1	c_1	d_2	e_1
t_4	a_1	b_1	c_2	d_2	e_2

r_{28}	A	B	C	D	E
t_1	a_1	b_1	c_1	d_1	e_1
t_2	a_1	b_1	c_1	d_2	e_2
t_3	a_1	b_1	c_1	d_1	e_2
	a_1	b_1	c_1	d_2	e_1

Slika 3.5.

Ukoliko se pretpostavi da simboli tabloa (elementi skupa *Sym*, uvedenog definicijom 2.40) predstavljaju, istovremeno, i elemente domena odgovarajućih obeležja i za interpretaciju tabloa $T(U)$ se upotrebi ***funkcija identičnog preslikavanja*** : $v(T(U)) = T(U)$, tada se *chase* modifikacije mogu direktno primenjivati na tablou $T(U)$. Na taj način se dobija na preglednosti razmatranog problema, pri čemu se opštost razmatranja ne umanjuje. U nastavku rada će, po potrebi, biti korišćena mogućnost direktnih *chase* modifikacija tabloa.

3.1. Generalizovani način prikaza zavisnosti podataka

3.1.1. Generalizovani način prikaza funkcionalnih zavisnosti

Funkcionalne zavisnosti se mogu prikazati putem *egd*, čiji tablo sadrži dva predikata, a predikati *Eq* se definišu za svako obeležje desne strane date *fd*.

Teorema 3.1. Neka je data *fd* $X \to A$, pri čemu je $X \subseteq U$ i $A \in U$, gde je U univerzalni skup obeležja i neka je data *egd* $<T_{XA}(U), Eq_A(a_1, a_2)>$, takva da je $T_{XA}(U) = \{P_1, P_2\}$, pri čemu važi da je $(\forall B \in X)(f_{p_1}(B) = f_{p_2}(B)) \wedge f_{p_1}(A) = a_1 \wedge f_{p_2}(A) = a_2$ (grafička reprezentacija tabloa je data na slici 3.6, pri čemu je oznakom "-" naznačeno da simbol na datoj poziciji tabloa nije od važnosti). Neka je $r \in SAT(U)$. Važi $r \models <T_{XA}(U), Eq_A(a_1, a_2)>$, ako i samo ako je $r \models X \to A$.

$T_{XA}(U)$	X	A	$U \setminus X$
	x_1	a_1	-
	x_1	a_2	-

Slika 3.6.

Dokaz. ($\Rightarrow$) Neka je $v(T_{XA}(U)) = \{t_1, t_2\}$ proizvoljna interpretacija tabloa $T_{XA}(U)$, takva da je $v(T_{XA}(U)) \subseteq r$. Pošto je $t_1[X] = t_2[X]$, saglasno *fd* $X \to A$, važi da je $t_1[A] = t_2[A]$, odnosno $v(a_1) = v(a_2)$. Pošto je $v(T_{XA}(U))$ proizvoljna interpretacija, sledi da $(\forall v(T_{XA}(U)))(v(T_{XA}(U)) \subseteq r \Rightarrow v(a_1) = v(a_2))$, odakle sledi da $r \models <T_{XA}(U), Eq_A(a_1, a_2)>$.

($\Leftarrow$) Neka su $t_1, t_2 \in r$ proizvoljno odabrane torke relacije r, takve da važi $t_1[X] = t_2[X]$. Neka je $v(T_{XA}(U))$ takva interpretacija tabloa $T_{XA}(U)$, za koju važi da je $v(T_{XA}(U)) = \{t_1, t_2\}$. Na osnovu pretpostavke o važenju *egd* $<T_{XA}(U), Eq_A(a_1, a_2)>$, sledi da je $v(a_1) = v(a_2)$, odnosno $t_1[A] = t_2[A]$. Pošto su t_1 i t_2 proizvoljno odabrane torke, zaključuje se da $(\forall t_1, t_2 \in r)(t_1[X] = t_2[X] \Rightarrow t_1[A] = t_2[A])$, odnosno $r \models X \to A$.

Na osnovu teoreme 3.1. i pravila dekompozicije desne strane, sledi da se bilo koja *fd* $X \to Y$, pri čemu je $X, Y \subseteq U$, može prikazati putem ekvivalentne *egd* $<T_{XY}(U), E>$, takve da za svako obeležje $A \in Y$ postoji jedan predikat $Eq_A(a_1, a_2) \in E$: $E = \{Eq_A(a_1, a_2) \mid A \in Y\}$.

Primer 3.3. Generalizovana E-zavisnost iz primera 3.1. γ_2: $<T(U), Eq_C(z_1, z_2)>$ je ekvivalentna funkcionalnoj zavisnosti $AB \to C$.

3.1.2. Generalizovani način prikaza višeznačnih zavisnosti

Višeznačne zavisnosti se mogu prikazati putem *tgd*, čiji prvi tablo sadrži dva predikata, s jednakim simbolima nad obeležjima leve strane date *mvd* i različitim simbolima nad ostalim obeležjima, a drugi tablo sadrži jedan predikat koji predstavlja kombinaciju simbola predikata prvog tabloa.

Teorema 3.2. Neka je dat skup obeležja $R \subseteq U$ i *mvd* $X \rightarrow\rightarrow Y$, pri čemu je $X, Y \subseteq R$, gde je U univerzalni skup obeležja i neka je data *tgd* $<T_{XY}(U), Q(R)>$, takva da je $T_{XY}(U) = \{P_1, P_2\}$, pri čemu važi da je $(\forall A \in X)(f_{p_1}(A) = f_{p_2}(A)) \wedge (\forall A \in R \setminus X)(f_{p_1}(A) \neq f_{p_2}(A))$, a Q predikat nad skupom obeležja R, takav da važi $(\forall A \in XY)(f_Q(A) = f_{p_1}(A)) \wedge (\forall A \in R \setminus XY)(f_Q(A) = f_{p_2}(A))$ (grafička reprezentacija tabloa je data na slici 3.7, pri čemu je oznakom "-" naznačeno da simbol na datoj poziciji tabloa nije od važnosti). Neka je $r \in SAT(R)$. Važi $r \models <T_{XY}(U), Q(R)>$, ako i samo ako je $r \models X \rightarrow\rightarrow Y$.

$T_{XY}(U)$	X	Y	$R \setminus XY$	$U \setminus R$
	x_1	y_1	z_1	-
	x_1	y_2	z_2	-

$Q(R)$	X	Y	$R \setminus XY$	$U \setminus R$
	x_1	y_1	z_2	-

Slika 3.7.

Dokaz. $(\Rightarrow)$ Neka je $v(T_{XY}(U)) = \{t_1, t_2\}$ proizvoljna interpretacija tabloa $T_{XY}(U)$, takva da je $v(T_{XY}(U)) \subseteq r$. Pošto je $t_1[X] = t_2[X]$, saglasno *mvd* $X \rightarrow\rightarrow Y$, važi da $(\exists t' \in r)(t'[XY] = t_1[XY] \wedge t'[R \setminus XY] = t_2[R \setminus XY])$. Saglasno uslovu $(\forall A \in XY)(f_Q(A) = f_{p_1}(A)) \wedge (\forall A \in R \setminus XY)(f_Q(A) = f_{p_2}(A))$, na osnovu kojeg je formiran predikat $Q(R)$ i prethodnoj činjenici da za neku torku t' važi: $t' \in r \wedge t'[XY] = t_1[XY] \wedge t'[R \setminus XY] = t_2[R \setminus XY]$, zaključuje se da je ispunjen uslov $v(\{Q(R)\}) = \{t'\} \subseteq r$. Pošto je $v(T_{XY}(U))$ proizvoljna interpretacija, sledi da $(\forall v(T_{XY}(U))(v(T_{XY}(U)) \subseteq r \Rightarrow v(\{Q(R)\}) \subseteq r)$, odakle sledi da $r \models <T_{XY}(U), Q(R)>$.

$(\Leftarrow)$ Neka su $t_1, t_2 \in r$ proizvoljno odabrane torke relacije r, takve da važi $t_1[X] = t_2[X]$. Neka je $v(T_{XY}(U))$ takva interpretacija tabloa $T_{XY}(U)$, za koju važi da je $v(T_{XY}(U)) = \{t_1, t_2\}$. Na osnovu pretpostavke o važenju *tgd* $<T_{XY}(U), Q(R)>$, sledi da je $v(\{Q(R)\}) \subseteq r$, a na osnovu načina formiranja predikata $Q(R)$ sledi da $(\exists t' \in r)(t'[XY] = t_1[XY] \wedge t'[R \setminus XY] = t_2[R \setminus XY])$, pri čemu navedeno svojstvo ispunjava torka $v(Q(R))$. Pošto su t_1 i t_2 proizvoljno odabrane torke, zaključuje se da $(\forall t_1, t_2 \in r)(t_1[X] = t_2[X] \Rightarrow (\exists t' \in r)(t'[XY] = t_1[XY] \wedge t'[R \setminus XY] = t_2[R \setminus XY]))$, odnosno $r \models X \rightarrow\rightarrow Y$.

Može se primetiti da ukoliko se kao uslov prethodne teoreme uvede jednakost $R = U$, tada je u pitanju ekvivalentnost *ftgd* i *fmvd*. Ukoliko, međutim, važi da je $R \subset U$, onda je reč o ekvivalentnosti *etgd* i *emvd*.

Primer 3.4. Generalizovana T-zavisnost iz primera 3.1. $<T(U), Q(x_1, y_1, z_1, u_1, w_2)>$ je ekvivalentna skupu višeznačnih zavisnosti $\{AB \rightarrow\rightarrow C, AB \rightarrow\rightarrow D, AB \rightarrow\rightarrow E\}$.

3.1.3. Generalizovani način prikaza zavisnosti spoja

Zavisnosti spoja se mogu prikazati putem *tgd*, čiji prvi tablo sadrži onoliko predikata, koliko komponenti sadrži *jd*, s jednakim simbolima nad obeležjima koja pripadaju presecima komponenti date *jd*. Drugi tablo sadrži jedan predikat koji predstavlja kombinaciju simbola predikata prvog tabloa.

Teorema 3.3. Neka je dat skup obeležja $R \subseteq U$ i *jd* $\triangleright\triangleleft(X_1,...,X_n)$, pri čemu je $\bigcup_{i=1}^{n} X_i = R$, gde je U univerzalni skup obeležja i neka je data *tgd* $<T(U), Q(R)>$, takva da je $T(U) = \{P_1,...,P_n\}$, pri čemu važi da je:

(3.1) $(\forall i, j \in \{1,...,n\})((\forall A \in X_i \cap X_j)(f_{P_i}(A) = f_{P_j}(A)) \wedge (\forall A \in R \setminus (X_i \cap X_j))(f_{P_i}(A) \neq f_{P_j}(A))),$

a Q predikat nad skupom obeležja R, takav da važi:

(3.2) $(\forall i \in \{1,...,n\})(\forall A \in X_i)(f_Q(A) = f_{P_i}(A)).$

Neka je $r \in SAT(R)$. Važi $r \models <T(U), Q(R)>$, ako i samo ako je $r \models \triangleright\triangleleft(X_1,...,X_n)$.

Dokaz. ($\Rightarrow$) Neka je $v(T(U)) = \{t_1,...,t_n\}$ proizvoljna interpretacija tabloa $T(U)$, takva da je $v(T(U)) \subseteq r$. Pošto je, saglasno formuli (3.1), $(\forall i, j \in \{1,...,n\})(t_i[X_i \cap X_j] = t_j[X_i \cap X_j])$, na osnovu *jd* $\triangleright\triangleleft(X_1,...,X_n)$ se izvodi zaključak da $(\exists t \in r)(\forall i \in \{1,...,n\})(t[X_i] = t_i[X_i])$. Saglasno uslovu (3.2), na osnovu kojeg je formiran predikat $Q(R)$ i prethodnoj činjenici da za neku torku t važi: $t \in r \wedge (\forall i \in \{1,...,n\})(t[X_i] = t_i[X_i])$, zaključuje se da je ispunjen uslov $v(\{Q(R)\}) = \{t\} \subseteq r$. Pošto je $v(T(U))$ proizvoljna interpretacija, sledi da $(\forall v(T(U))(v(T(U)) \subseteq r \Rightarrow v(\{Q(R)\}) \subseteq r)$, odakle sledi da $r \models <T(U), Q(R)>$.

($\Leftarrow$) Neka su $t_1,...,t_n \in r$ proizvoljno odabrane torke relacije r, takve da važi $(\forall i, j \in \{1,...,n\})(t_i[X_i \cap X_j] = t_j[X_i \cap X_j])$. Neka je $v(T(U))$ takva interpretacija tabloa $T(U)$, za koju važi da je $v(T(U)) = \{t_1,...,t_n\}$. Na osnovu pretpostavke o važenju *tgd* $<T(U), Q(R)>$, sledi da je $v(\{Q(R)\}) \subseteq r$, a na osnovu načina formiranja predikata $Q(R)$ sledi da $(\exists t \in r)(\forall i \in \{1,...,n\})(t[X_i] = t_i[X_i])$, pri čemu navedeno svojstvo ispunjava torka $v(Q(R))$. Pošto su $t_1,...,t_n$ proizvoljno odabrane torke, zaključuje se da $(\forall t_1,...,t_n \in r)((\forall i, j \in \{1,...,n\})(t_i[X_i \cap X_j] = t_j[X_i \cap X_j]) \Rightarrow (\exists t \in r)(\forall i \in \{1,...,n\})(t[X_i] = t_i[X_i]))$, odnosno $r \models \triangleright\triangleleft(X_1,...,X_n)$.

Može se primetiti da ukoliko se kao uslov prethodne teoreme uvede jednakost $R = U$, tada je u pitanju ekvivalentnost *ftgd* i *fjd*. Ukoliko, međutim, važi da je $R \subset U$, onda je reč o ekvivalentnosti *etgd* i *ejd*.

Primer 3.5. Generalizovana T-zavisnost iz primera 3.1. $<T(U), Q(x_1, y_1, z_1, u_1, w_2)>$ je ekvivalentna zavisnosti spoja $\triangleright\triangleleft(ABC, ABD, ABE)$.

Primer 3.6. Generalizovana T-zavisnost $<T(U), Q(a_1, b_1, c_1, d_2)>$, pri čemu je $T(U)$ prikazan na slici 3.8, je ekvivalentna ugrađenoj zavisnosti spoja $\triangleright\triangleleft(ABC, BCD, CDA)$.

$T(U)$	A	B	C	D	E
	a_1	b_1	c_1	d_1	e_1
	a_2	b_1	c_1	d_2	e_2
	a_1	b_2	c_1	d_2	e_3

Slika 3.8.

3.1.4. Generalizovani način prikaza zavisnosti sadržavanja

Zavisnosti sadržavanja se iskazuju putem samih tabloa, za razliku od *fd* i *jd* koje se iskazuju, redom, putem *egd* i *tgd*.

Teorema 3.4. Neka su dati skupovi obeležja $R_1 \subseteq U$, $R_2 \subseteq U$, $X \subseteq R_1$ i $Y \subseteq R_2$, *id* $R_1[X] \subseteq R_2[Y]$, s bijekcijom $h : X \rightarrow Y$, takvom da $(\forall A_i \in X)(dom(A_i) \subseteq dom(h(A_i)))$ i tablo $T(U)$, za koji važi $(\forall A \in X \setminus Y)(\forall P_i \in T(U))(\exists P_j \in T(U))(f_{P_i}(A) = f_{P_j}(h(A)))$. Za bilo koje dve relacije $r_1 \in SAT(R_1)$ i $r_2 \in SAT(R_2)$, takve da postoji interpretacija tabloa v: $T(U) \rightarrow DomU^n$, za koju je zadovoljeno da $\Pi_{R_1}(v(T(U))) \subseteq r_1$ i $\Pi_{R_2}(v(T(U))) \subseteq r_2$, važi da zadovoljavaju *id* $R_1[X] \subseteq R_2[Y]$.

Dokaz. Iz uslova $(\forall A \in X \setminus Y)(\forall P_i \in T(U))(\exists P_j \in T(U))(f_{P_i}(A) = f_{P_j}(h(A)))$ sledi da je $\Pi_X(T(U)) \subseteq \Pi_Y(T(U))$. S druge strane, treba dokazati da, pod navedenim uslovima teoreme, važi da je $(\forall t_1 \in r_1)(\exists t_2 \in r_2)(t_1[X] = t_2[Y])$. Neka je $t_1 \in r_1$ proizvoljno odabrana torka relacije r_1. Pošto, po pretpostavci teoreme, postoji interpretacija v, takva da je $\Pi_{R_1}(v(T(U))) \subseteq r_1$, znači da u tablou $T(U)$ postoji predikat, npr. P_1, za koji važi da je $(v(P_1))[X] = t_1$. Na osnovu činjenice $\Pi_X(T(U)) \subseteq \Pi_Y(T(U))$, sledi da u tablou $T(U)$ postoji predikat, npr. P_2, za koji važi da je $P_1[X] = P_2[Y]$. Budući da je $\Pi_{R_2}(v(T(U))) \subseteq r_2$, može se zaključiti da $(v(P_2))[Y] \in r_2$, što znači da relacija r_2 sadrži neku torku t_2, takvu da je $t_1[X] = t_2[Y]$. □

Primer 3.7. Na slici 3.9. je prikazan tablo $T(U)$, kojim je reprezentovana *id* $ABC[BC] \subseteq CDE[DE]$ (pri čemu je $h(B) = D$ i $h(C) = E$). Može se zapaziti da važi inkluzija $\Pi_{BC}(T(U)) \subseteq \Pi_{DE}(T(U))$. Na slici 3.10. su prikazane relacije $r_1 \in SAT(ABC)$ i $r_2 \in SAT(CDE)$, takve da zadovoljavaju *id* $ABC[BC] \subseteq CDE[DE]$ i postoji interpretacija tabloa v, takva da važi $\Pi_{ABC}(v(T(U))) \subseteq r_1$ i $\Pi_{CDE}(v(T(U))) \subseteq r_2$.

$T(U)$	A	B	C	D	E
	x_1	y_1	z_1	u_1	w_1
	x_2	y_2	z_2	y_1	z_1
	x_1	y_2	z_2	y_2	z_2

Slika 3.9.

r_1	A	B	C
	a_1	b_1	c_1
	a_2	b_2	c_2
	a_1	b_2	c_2
	a_3	b_3	c_3
	a_3	b_2	c_2

r_2	C	D	E
	c_1	d_1	e_1
	c_2	b_1	c_1
	c_2	b_2	c_2
	c_3	d_3	e_3
	c_2	b_3	c_3

Slika 3.10.

3.2. Konačna specifikabilnost, rang i red familije

S obzirom na konačnost univerzalnog skupa obeležja U, može se zaključiti da su skupovi zavisnosti $FD(U)$ i $MVD(U)$ konačni. S druge strane, pošto je skup simbola (simbolički domen) Sym, na osnovu kojeg se formiraju *gd*, beskonačan, zaključuje se da je skup *gd* $GD(U) = TGD(U) \cup EGD(U)$ takođe beskonačan i prebrojiv skup. To znači da je formalno moguće konstruisati šemu relacije (U, Σ), $\Sigma \subseteq GD(U)$, takvu da je $|\Sigma| = \infty$. Takva šema relacije, međutim, nema nikakvog smisla ni s teoretskog ni s praktičnog stanovišta, jer je, u slučaju $|\Sigma| = \infty$, problem testiranja pripadnosti relacije familiji $r \in SAT(U, \Sigma)$ poluodlučiv. Drugim rečima, algoritam testa $r \in SAT(U, \Sigma)$ će sigurno okončati samo ukoliko je zadovoljeno da $r \notin SAT(U, \Sigma)$.

Definicija 3.1. Familija pojava $\mathcal{F} = SAT(R, \Sigma)$, pri čemu je $R \subseteq U$ i $\Sigma \subseteq GD(R)$, se naziva ***konačno specifikabilna*** familija, ako važi da je Σ konačan skup *gd* ($|\Sigma| < \infty$). □ **[Hu]**

Pošto su skupovi $FD(U)$, $MVD(U)$ i $JD(U)$ konačni, zaključuje se da svaka *fd*-familija, *mvd*-familija i *jd*-familija (tj. familija $SAT(U, \Sigma)$, za koju važi da je $\Sigma \subseteq FD(U)$, $\Sigma \subseteq MVD(U)$ i $\Sigma \subseteq JD(U)$, respektivno) mora biti konačno specifikabilna.

Definicija 3.2. Za *tgd* $<T(U), Q(x_1,...,x_m)>$, odnosno *egd* $<T(U), Eq_{A_j}(\lambda^j_p, \lambda^j_q)>$, se kaže da ima ***rang*** n ($n \in \mathbf{N}$ je prirodan broj), ako je zadovoljeno da $|T(U)| \leq n$. □ **[Hu]**

Definicija 3.3. Za familiju pojava $\mathcal{F} = SAT(R, \Sigma)$, pri čemu je $R \subseteq U$ i $\Sigma \subseteq GD(R)$, se kaže da ima ***rang*** n ($n \in \mathbf{N}$), ako svaka $gd \in \Sigma$ ima rang n. □ **[Hu]**

Lema 3.1. Familija pojava $\mathcal{F} = SAT(R, \Sigma)$, pri čemu je $R \subseteq U$ i $\Sigma \subseteq GD(R)$, je konačno specifikabilna ako i samo ako ima rang n ($n \in \mathbf{N}$). **[Hu]**

Dokaz. ($\Rightarrow$) $\mathcal{F}$ ima rang n. Neka su sve $gd \in \Sigma$ formirane tako da je za svaki $A \in R$ upotrebljen konačan skup od n simbola $Sym(A) = \{a_1,...,a_n\} \subseteq Sym$. (Ukoliko za neku $gd \in \Sigma$ ovaj uslov ne bi bio ispunjen, onda bi se ona, izomorfnim transformacijama, mogla transformisati u ekvivalentnu *gd*, takvu da se navedeni uslov ispuni). Pošto je skup upotrebljenih simbola za formiranje svih tabloa generalizovanih zavisnosti konačan, sledi da i skup Σ mora biti konačan, te je $\mathcal{F}$ konačno specifikabilna.

($\Leftarrow$) $\mathcal{F}$ je konačno specifikabilna familija. Pošto je tablo svake $gd \in \Sigma$ konačan skup predikata, sledi da svaka $gd \in \Sigma$ ima rang koji odgovara rangu one *gd*, čiji tablo sadrži maksimalan broj predikata. Neka taj broj iznosi n. Sledi da familija $\mathcal{F}$ ima rang n. □

Definicija 3.4. Familija pojava $\mathcal{F} = SAT(R, \Sigma)$, pri čemu je $R \subseteq U$ i $\Sigma \subseteq GD(R)$, ima ***red podinstance*** (kraće ***red***) n, ako svaka relacija $r \in SAT(R)$, za koju važi tzv. ***pravilo podinstance reda*** n, pripada familiji $\mathcal{F}$:

(3.3) $$(\forall r \in SAT(R))(pravilo\_podinstance(r, n) \Rightarrow r \in \mathcal{F}),$$

pri čemu se predikat $pravilo\_podinstance(r, n)$ interpretira na sledeći način:

(3.4) $$(\forall s \subseteq r)(|s| \leq n \Rightarrow (\exists s' \in \mathcal{F})(s \subseteq s' \subseteq r)).$$ □ **[Hu]**

Lema 3.2. *Fd*-familija $\mathcal{F} = SAT(U, \Gamma)$, $\Gamma \subseteq FD(U)$, ima red podinstance $n = 2$.

Dokaz. Neka je $r \in SAT(U)$ proizvoljno izabrana relacija, za koju važi predikat *pravilo_podinstance*$(r, 2)$. Uslov (3.4), u tom slučaju, ima oblik: $(\forall t_1, t_2 \in r)(\exists s \in SAT(U, \Gamma))(\{t_1, t_2\} \subseteq s \subseteq r)$. S obzirom da je $\mathcal{F}$ *fd*-familija, minimalna relacija s, za koju navedeni uslov važi je $s = \{t_1, t_2\}$, odakle sledi da $(\forall t_1, t_2 \in r)(\{t_1, t_2\} \in SAT(U, \Gamma))$, što znači da važi uslov $(\forall t_1, t_2 \in r)(\forall X \rightarrow Y \in \Gamma)(t_1[X] = t_2[X] \Rightarrow t_1[Y] = t_2[Y])$. Prema definiciji 2.21, zaključuje se da $(\forall X \rightarrow Y \in \Gamma)(r \models X \rightarrow Y)$, odnosno $r \in SAT(U, \Gamma)$, što je trebalo dokazati. □

Lema 3.3. *Mvd*-familija $\mathcal{F} = SAT(U, \Gamma)$, $\Gamma \subseteq MVD(U)$, ima red podinstance $n = 2$.

Dokaz. Neka je $r \in SAT(U)$ proizvoljno izabrana relacija, za koju važi predikat *pravilo_podinstance*$(r, 2)$. Uslov (3.4), u tom slučaju, ima oblik: $(\forall t_1, t_2 \in r)(\exists s \in SAT(U, \Gamma))(\{t_1, t_2\} \subseteq s \subseteq r)$. S obzirom da je $\mathcal{F}$ *mvd*-familija, znači da za $(\forall X \rightarrow\rightarrow Y \in \Gamma)(\exists t \in s)$ $(t[XY] = t_1[XY] \wedge t[X(U \setminus Y)] = t_2[X(U \setminus Y)])$. Pošto važi da je $\{t_1, t_2\} \subseteq s \subseteq r$, zaključuje se, saglasno definiciji 2.35, da je ispunjen uslov $(\forall X \rightarrow\rightarrow Y \in \Gamma)(r \models X \rightarrow\rightarrow Y)$, odnosno $r \in SAT(U, \Gamma)$, što je trebalo dokazati. □

Na osnovu definicije 3.4. se, za svaku familiju $\mathcal{F}$ koja poseduje *konačan* red podinstance, može testirati da li za proizvoljno odabranu relaciju r važi tvrdnja: $r \in \mathcal{F}$. Problem utvrđivanja odgovora na pitanje $r \in \mathcal{F}$ je, u datom slučaju, odlučiv, eksponencijalne složenosti. Eksponencijalna složenost testa je posledica eksponencijalne složenosti problema generisanja svih podskupova relacije r, kardinalnosti manje ili jednake n, pri čemu je n red podinstance familije $\mathcal{F}$.

Lema 3.4. Familija pojava $\mathcal{F} = SAT(R, \Sigma)$, pri čemu je $R \subseteq U$ i $\Sigma \subseteq GD(R)$, ima konačan red podinstance n ($n \in \mathbf{N}$), ako i samo ako ima rang n. □ **[Hu]**

Teorema 3.5. Neka je data familija pojava $\mathcal{F} = SAT(R, \Sigma)$, pri čemu je $R \subseteq U$ i $\Sigma \subseteq GD(R)$. Sledeće tvrdnje su ekvivalentne:

1. $\mathcal{F}$ ima rang n ($n \in \mathbf{N}$),
2. $\mathcal{F}$ ima red podinstance n,
3. $\mathcal{F}$ je konačno specifikabilna.

Dokaz. Lema 3.1. i lema 3.4. □ **[Hu]**

Primer 3.8. Familija pojava $\mathcal{F} = SAT(ABCDEFG, \Sigma)$, pri čemu je $\Sigma = \{AB \rightarrow C, D \rightarrow E, \rhd\lhd(AB, D, FG)\}$ je konačno specifikabilna, ranga, tj. reda podinstance $n = 3$. □

Primer 3.9. Familija pojava $\mathcal{F} = SAT(U, \Sigma)$, pri čemu je $U = ABC$, a $\Sigma = \{<T_i(U), Q_i(a_i, b_{i+1}, c_{i+2})> \mid i \in \mathbf{N}\}$, tako da $(\forall i \in \mathbf{N})(T_i(U) = \{P_j(a_j, b_j, c_j) \mid j \in \{1,...,i+2\}\})$ nije konačno specifikabilna i nema konačan rang. □

3.3. Projekcija familije generalizovanih zavisnosti

U ovom delu se razmatra problem zatvorenosti i konačne specifikabilnosti projekcije familije pojava. Biće pokazano da postoje konačno specifikabilne familije, čija projekcija ne predstavlja konačno specifikabilnu familiju. Biće, takođe, utvrđeni potrebni i dovoljni uslovi da određene vrste konačno specifikabilnih familija budu zatvorene s obzirom na operator projekcije.

Definicija 3.5. Konačno specifikabilna familija pojava $SAT(R, \Sigma)$, pri čemu je $R \subseteq U$ i Σ skup zavisnosti određenog tipa, je ***zatvorena s obzirom na operator projekcije*** na skup obeležja $\emptyset \neq X \subseteq R$, ako važi da postoji konačno specifikabilna familija $SAT(X, \Delta)$, takva da je $\Pi_X(SAT(R, \Sigma)) = SAT(X, \Delta)$, pri čemu su sve zavisnosti iz skupa Δ istog tipa kao i zavisnosti skupa Σ. □

Teorema 3.6. Postoji *fd*-familija $\mathcal{F} = SAT(R, \Gamma)$, čija projekcija na neki skup obeležja $\emptyset \neq X \subset R$ ne predstavlja konačno specifikabilnu familiju. **[Hu]**

Dokaz. Kao primer *fd*-familije, čija projekcija na dati skup obeležja X ne predstavlja konačno specifikabilnu familiju, biće upotrebljena familija $\mathcal{F} = SAT(R, \Gamma)$, pri čemu je $R = ABCDE$ i $\Gamma = \{A \to C, B \to C, CD \to E\}$. Neka je dat skup obeležja $X = ABDE$ i neka je $\Pi_X(SAT(R, \Gamma)) = SAT(X, \Delta) = \mathcal{F}'$, gde je $\Delta = GD(U)$. Dokaz će biti baziran na obratu po kontrapoziciji, tako da se uvodi suprotna pretpostavka, po kojoj je $\mathcal{F}'$ konačno specifikabilna. Saglasno teoremi 3.5, $\mathcal{F}'$ ima konačan rang, odnosno red podinstance, npr. n. Neka je data relacija $r \in SAT(X)$, $r = \{t_1, ..., t_n, t_{n+1}\}$, konstruisana na sledeći način:

$$(3.5)\quad (\forall i \in \{1,...,n+1\})(t_i(A) = \lceil i/2 \rceil \wedge t_i(B) = \lfloor i/2 \rfloor + 1 \wedge t_i(E) = i\,) \wedge t_i(D) = \begin{cases} i, & 1 \le i \le n \\ 1, & i = n+1 \end{cases}.$$

Na slici 3.11. je prikazana relacija $r \in SAT(X)$, za slučaj ranga $n = 3$.

r	A	B	D	E
t_1	1	1	1	1
t_2	1	2	2	2
t_3	2	2	3	3
t_4	2	3	1	4

Slika 3.11.

U nastavku će biti prvo pokazano da važi $r \in SAT(X, \Delta)$. Pošto $\mathcal{F}'$ ima red podinstance n, dovoljno je, prema (3.4), pokazati da $(\forall s \subseteq r)(|s| \le n \Rightarrow (\exists s_1 \in \mathcal{F}')(s \subseteq s_1 \subseteq r))$. Neka je $s \subseteq r$ proizvoljno odabran podskup od r, za koji važi da je $|s| \le n$. Neka je $s = r - \{t_i\}$, $i \in \{1,...,n+1\}$. Od relacije s se može napraviti relacija $s' \in SAT(R)$, $s' = \{u_1, ..., u_{i-1}, u_{i+1}, ..., u_{n+1}\}$, na sledeći način (slika 3.12, za $n = 3$ i $i = 2$):

$$(3.6)\qquad (\forall j \in \{1,..., i-1, i+1,..., n+1\})(u_j[X] = t_j) \wedge u_j(C) = \begin{cases} 1, & 1 \le j < i \\ 2, & i < j \le n+1 \end{cases}.$$

s	A	B	D	E
t_1	1	1	1	1
t_3	2	2	3	3
t_4	2	3	1	4

s'	A	B	C	D	E
u_1	1	1	1	1	1
u_3	2	2	2	3	3
u_4	2	3	2	1	4

Slika 3.12.

Na osnovu (3.5) i (3.6) se može zaključiti da:

- $s' \models A \rightarrow C$, jer $(\forall j, k \in \{1,..., i-1, i+1,..., n+1\})(u_j(C) \neq u_k(C) \Rightarrow u_j(A) \neq u_k(A))$,
- $s' \models B \rightarrow C$, jer $(\forall j, k \in \{1,..., i-1, i+1,..., n+1\})(u_j(C) \neq u_k(C) \Rightarrow u_j(B) \neq u_k(B))$ i
- $s' \models CD \rightarrow E$, jer $(\forall j, k \in \{1,..., i-1, i+1,..., n+1\})(u_j(CD) \neq u_k(CD))$,

odakle sledi da $s' \in SAT(R, \Gamma)$. Na osnovu (3.6) važi da je $s = \Pi_X(s')$, te je prema definiciji projekcije familije, $s \in \Pi_X(SAT(R, \Gamma)) = SAT(X, \Delta)$. Pošto navedena osobina važi za svaki $s \subseteq r$, takav da je $|s| \leq n$, zaključuje se da $r \in SAT(X, \Delta)$.

Pošto važi $r \in SAT(X, \Delta)$, na osnovu pretpostavke teoreme $\Pi_X(\mathcal{F}) = \mathcal{F}'$, proizilazi da mora biti zadovoljen uslov $(\exists r' \in SAT(R, \Gamma))(\Pi_X(r') = r)$. Relacija $r' = \{u_1,...,u_{n+1}\}$ se mora konstruisati polazeći od relacije r (slika 3.13), tako da bude ispunjen uslov:

(3.7) $$(\forall i \in \{1,...,n+1\})(u_j[X] = t_j).$$

Na osnovu *fd* $A \rightarrow C \in \Gamma$ i činjenice da je $(\forall i \in \{1,...,\lfloor n/2 \rfloor\})(u_{2i-1}(A) = u_{2i}(A))$ zaključuje se da mora važiti $(\forall i \in \{1,...,\lfloor n/2 \rfloor)(u_{2i-1}(C) = u_{2i}(C))$. Slično, na osnovu *fd* $B \rightarrow C \in \Gamma$ i činjenice da je $(\forall i \in \{1,...,\lfloor n/2 \rfloor\})(u_{2i}(B) = u_{2i+1}(B))$ zaključuje se da mora važiti $(\forall i \in \{1,..., \lfloor n/2 \rfloor)(u_{2i}(C) = u_{2i+1}(C))$. To, dalje, znači da važi $(\forall i \in \{1,..., n-1)(u_i(C) = u_{i+1}(C))$, što za posledicu ima $u_1(C) = u_{n+1}(C)$ (slika 3.13). Pošto je, prema (3.5) i (3.7), zadovoljeno $u_1(D) = u_{n+1}(D)$, znači da važi $u_1[CD] = u_{n+1}[CD]$, tako da, na osnovu *fd* $CD \rightarrow E$, mora biti zadovoljeno $u_1(E) = u_{n+1}(E)$. Ovo, međutim, obara uslov (3.7), pošto je $t_1(E) \neq t_{n+1}(E)$, što znači da $\neg(\exists r' \in SAT(R, \Gamma))(\Pi_X(r') = r)$. Sledi da je pretpostavka o konačnoj specifikabilnosti familije $SAT(X, \Delta) = \mathcal{F}'$ pogrešna, čime je dokazana tvrdnja teoreme. □

r	A	B	D	E
t_1	1	1	1	1
t_2	1	2	2	2
t_3	2	2	3	3
t_4	2	3	1	4

r'	A	B	C	D	E
u_1	1	1	1	1	1
u_2	1	2	1	2	2
u_3	2	2	1	3	3
u_4	2	3	1	1	1

Slika 3.13.

Značaj teoreme 3.6. je u tome što dokazuje da projekcija *fd*-familije nije, u opštem slučaju, zatvorena operacija. Drugim rečima, projekcija proizvoljno formirane *fd*-familije ne mora biti *fd*-familija. Analogan zaključak se može izvesti i za *mvd*-familije. U literaturi (**[Hu]**) je formulisana i dokazana teorema, analogna teoremi 3.6, po kojoj postoji *mvd*-familija $\mathcal{F}$, čija projekcija na neki skup obeležja $X \subset R$ ne predstavlja konačno specifikabilnu familiju.

Posledica 3.1. Sledeće familije pojava nisu zatvorene s obzirom na operator projekcije:

1. *fd*-familija,
2. konačno specifikabilna *egd*-familija,
3. *mvd*-familija,
4. *jd*-familija,
5. konačno specifikabilna *tgd*-familija i
6. konačno specifikabilna *gd*-familija.

Dokaz. Stav 1. je posledica teoreme 3.6. i definicije 3.5. Stav 2. je posledica teoreme 3.1, dok je stav 3. dokazan u **[Hu]**. Stav 4. je posledica činjenice da je *mvd* istovremeno i *jd*, a stav 5. je posledica teoreme 3.3. Stav 6. je posledica činjenice da su *egd* i *tgd* istovremeno i *gd*. □

U nastavku tačke 3.3. će biti prezentirani potrebni i dovoljni uslovi da *fd*-familije pojava budu zatvorene s obzirom na operaciju projekcije.

3.3.1. Projekcija familije funkcionalnih zavisnosti

Teorema 3.7. Neka je $\mathcal{F}$ proizvoljno odabrana *fd*-familija čija projekcija na neki skup obeležja $\Pi_X(\mathcal{F})$ nije *fd*-familija. Tada $\Pi_X(\mathcal{F})$ nije konačno specifikabilna familija. □ **[Hu]**

Pored tvrdnje teoreme 3.7, važi i obratna tvrdnja, da ako $\Pi_X(\mathcal{F})$ nije konačno specifikabilna faimilija, tada $\Pi_X(\mathcal{F})$ ne može biti *fd*-familija, pošto su sve *fd*-familije konačno specifikabilne. Saglasno teoremi 3.7, da bi *fd*-familija bila zatvorena s obzirom na projekciju, potrebno je i dovoljno utvrditi da je projekcija *fd*-familije takođe *fd*-familija. Iz tvrdnji teorema 3.6. i 3.7, zaključuje se da postoje *fd*-familije, čija projekcija na neki skup obeležja ne predstavlja *fd*-familiju. Primer takve *fd*-familije je dat u okviru dokaza teoreme 3.6.

Teorema 3.8. Neka je data *fd*-familija $SAT(R, \Gamma)$, $R \subseteq U$, $\Gamma \subseteq FD(R)$ i skup obeležja $\varnothing \neq X \subseteq R$, pri čemu važi da je $\Pi_X(SAT(R, \Gamma)) = SAT(X, \Delta) \wedge \Delta \subseteq FD(X)$. Tada važi ekvivalencija skupova $fd\ \Delta \equiv \Pi_X(\Gamma)$. **[GZ]**

Dokaz. Treba utvrditi da važi $\Delta \models \Pi_X(\Gamma)$ i $\Pi_X(\Gamma) \models \Delta$.

($\Rightarrow$) Neka je $s \in SAT(X, \Delta)$ proizvoljno odabrana relacija. Tada je $s \in \Pi_X(SAT(R, \Gamma))$, što znači da postoji $r \in SAT(R, \Gamma)$, tako da je $s = \Pi_X(r)$. Pošto važi $r \models \Gamma$ i $\Gamma \models \Pi_X(\Gamma)$, sledi da je $r \models \Pi_X(\Gamma)$ i zaključuje se (na osnovu definicije *fd*) da $\Pi_X(r) \models \Pi_X(\Gamma)$, odnosno $s \models \Pi_X(\Gamma)$, što znači da $(\forall s \in SAT(X))(s \models \Delta \Rightarrow s \models \Pi_X(\Gamma))$, tj. $\Delta \models \Pi_X(\Gamma)$.

($\Leftarrow$) Uvodi se suprotna pretpostavka: $\Pi_X(\Gamma) \not\models \Delta$. Pošto je $\Delta \subseteq FD(X)$, sledi da $(\exists V \to Y \in \Delta)(\Pi_X(\Gamma) \not\models V \to Y)$. Iz činjenica $\Gamma \models \Pi_X(\Gamma)$, $\Pi_X(\Gamma) \not\models V \to Y$ i $VY \subseteq X$ sledi da $\Gamma \not\models V \to Y$. Pošto je $\Gamma \not\models V \to Y$, zaključuje se da postoji relacija $r \in SAT(R, \Gamma)$, takva da $r \not\models V \to Y$. Budući da je $VY \subseteq X$, znači da važi $\Pi_X(r) \not\models V \to Y$. Pri tome je zadovoljeno da $\Pi_X(r) \in \Pi_X(SAT(R, \Gamma)) = SAT(X, \Delta)$. Činjenice $\Pi_X(r) \in SAT(X, \Delta)$ i $\Pi_X(r) \not\models V \to Y$ su kontradiktorne s obzirom na pretpostavku $V \to Y \in \Delta$, što obara polaznu pretpostavku $\Pi_X(\Gamma) \not\models \Delta$. □

Obrat tvrđenja, datog teoremom 3.8, međutim, ne važi. To znači da, u opštem slučaju, jednakost $\Pi_X(SAT(R, \Gamma)) = SAT(X, \Pi_X(\Gamma))$ **nije** zadovoljena, a $\Delta \equiv \Pi_X(\Gamma)$ predstavlja samo potreban uslov za $\Pi_X(SAT(R, \Gamma)) = SAT(X, \Delta) \wedge \Delta \subseteq FD(X)$. Pošto uvek važi $\Gamma \models \Pi_X(\Gamma)$, znači da se može zaključiti $\Pi_X(SAT(R, \Gamma)) \subseteq SAT(X, \Pi_X(\Gamma))$. Pri tome je, dakle, u opštem slučaju $SAT(X, \Pi_X(\Gamma)) \not\subseteq \Pi_X(SAT(R, \Gamma))$.

Primer 3.10. Primer iz dokaza teoreme 3.6, po kojem je $R = ABCDE$, $\Gamma = \{A \to C, B \to C, CD \to E\}$ i $X = ABDE$ pokazuje da postoje slučajevi u kojima je $SAT(X, \Pi_X(\Gamma)) \not\subseteq \Pi_X(SAT(R, \Gamma))$. Može se zaključiti da za relaciju $r(X)$, prikazanu na slici 3.11, važi da je $r \in SAT(X, \Pi_X(\Gamma))$, pri čemu je $\Pi_X(\Gamma) = \{AD \to E, BD \to E\}$, ali da je $(\forall r' \in SAT(R, \Gamma))(r \neq \Pi_X(r'))$ (videti dokaz teoreme 3.6. i sliku 3.13), što znači da $r \notin \Pi_X(SAT(R, \Gamma))$. □

U nastavku će biti formulisan dovoljan, ali ne i potreban uslov za $\Pi_X(SAT(R, \Gamma)) = SAT(X, \Pi_X(\Gamma))$.

Teorema 3.9. Neka je data *fd*-familija $SAT(R, \Gamma)$, $R \subseteq U$, $\Gamma \subseteq FD(R)$ i skup obeležja $\varnothing \neq X \subseteq R$. Ukoliko je ispunjeno da:

(3.8) $\quad \Delta^+ = \Pi_X(\Gamma) \wedge$

(3.9) $(\forall Y \to A \in \Gamma^+)((A \in X \setminus Y \wedge \neg(\exists Z \subset Y)(Z \to A \in \Gamma^+)) \Rightarrow$

$$((Y \subseteq X) \vee (Y \not\subseteq X \wedge X \to Y \notin \Gamma^+ \wedge (\forall B \in R \setminus X)(|dom(B)| \geq |dom[X]|)))),$$

pri čemu je za $X = A_1 \ldots A_p$, $dom[X] = dom(A_1) \times \ldots \times dom(A_p)$, tada je:

$$\Pi_X(SAT(R, \Gamma)) = SAT(X, \Delta).$$ **[KMo, GZ]**

Dokaz. ($\Rightarrow$) Inkluzija $\Pi_X(SAT(R, \Gamma)) \subseteq SAT(X, \Delta)$ sledi iz uslova (3.8) i činjenice $\Gamma \models \Pi_X(\Gamma)$.

($\Leftarrow$) Da bi se dokazala inkluzija $SAT(X, \Delta) \subseteq \Pi_X(SAT(R, \Gamma))$, posmatra se proizvoljno odabrana relacija $s \in SAT(X, \Delta)$. Treba dokazati da, pod navedenim uslovima teoreme, važi da $(\exists r \in SAT(R, \Gamma))(s = \Pi_X(r))$, odakle direktno sledi da $s \in \Pi_X(SAT(R, \Gamma))$. Koristeći obrat po kontrapoziciji, uvodi se suprotna pretpostavka:

(3.10) $$(\forall r \in SAT(R, \Gamma))\ (s \neq \Pi_X(r)).$$

Neka se, na osnovu relacije $s = \{t_1,...,t_n\}$, formira relacija $r_e \in SAT(R)$, $r_e = \{u_1,...,u_n\}$, što znači da je $|s| = |r_e|$. Saglasno pretpostavci teoreme (3.9), moguća su dva netrivijalna slučaja:

(3.11) $(\forall Y \to A \in \Gamma^+)((A \in X \setminus Y \wedge \neg(\exists Z \subset Y)(Z \to A \in \Gamma^+)) \Rightarrow (Y \subseteq X))$, ili

(3.12) $(\exists Y \to A \in \Gamma^+)(A \in X \setminus Y \wedge \neg(\exists Z \subset Y)(Z \to A \in \Gamma^+) \wedge Y \not\subseteq X \wedge X \to Y \notin \Gamma^+) \wedge$

$$(\forall B \in R \setminus X)(\ |dom(B)| \geq |dom[X]|\ ).$$

U slučaju da važi (3.11), r_e se formira tako da bude zadovoljeno:

(3.13) $$(\forall i \in \{1,...,n\})(u_i[X] = t_i) \wedge (\forall i, j \in \{1,...,n\})(u_i[R \setminus X] = u_j[R \setminus X]).$$

Neka je $Y \to A \in \Gamma^+$ proizvoljna netrivijalna, potpuna *fd*. Ako je $A \in X \setminus Y$, tada važi, na osnovu (3.11), $Y \subseteq X$, što znači da $s \models Y \to A$, odakle se zaključuje da $r_e \models Y \to A$. Ako, međutim, važi da $A \notin X \setminus Y$, na osnovu (3.13), proizilazi da je $(\forall u_i, u_j \in r_e)(u_i(A) = u_j(A))$, na osnovu čega, takođe, sledi činjenica $r_e \models Y \to A$. Može se, dakle, zaključiti da $r_e \in SAT(R, \Gamma)$, što obara pretpostavku (3.10).

U slučaju da važi (3.12), tj. $(\forall B \in R \setminus X)(\ |dom(B)| \geq |dom[X]|\ )$, r_e se može formirati na sledeći način:

(3.14) $$(\forall i \in \{1,...,n\})(u_i[X] = t_i) \wedge (\forall i, j \in \{1,...,n\})(\forall A \in R \setminus X)(i \neq j \Rightarrow u_i(A) \neq u_j(A)).$$

Treba zapaziti da je $\Pi_X(r_e) = s$ i $r_e \models \Pi_X(\Gamma)$. Nad relacijom r_e se može primeniti *chase* algoritam, koji generiše sekvencu $(r_e = r_0, r_1, ..., r_{p-1}, r_p)$. Pošto je $\Gamma \subseteq FD(R)$, rezultat primene *chase* algoritma je relacija $r_p = CH_\Gamma(r_e) \in SAT(R, \Gamma)$, pri čemu je $|CH_\Gamma(r_e)| \leq |r_e|$. Saglasno pretpostavci (3.10), važi $s \neq \Pi_X(CH_\Gamma(r_e))$, što je moguće samo u slučaju da je *chase* algoritam promenio vrednost makar jednog obeležja $A \in X$ relacije r_e, saglasno nekoj *fd* iz skupa $\Gamma^+ \setminus \Pi_X(\Gamma)$, tako da je ispunjeno:

(3.15) $$(\exists u_i \in r_e)(\exists u_j' \in CH_\Gamma(r_e))(\ u_i[R \setminus X] = u_j'[R \setminus X] \wedge u_j'[X] \notin s).$$

Pri tome, važi $u_i[X] \in s$ i $u_i[X] \neq u_j'[X]$. Neka je za obeležje $A \in X$ ispunjeno da $u_i(A) \neq u_j'(A)$, što znači da je *chase* modifikacija tabele r_e napravljena po osnovu netrivijalne, potpune *fd* $Y \to A \in \Gamma^+$. Pri tome mora da važi $Y \not\subseteq X$ (zbog uslova $Y \to A \in \Gamma^+ \setminus \Pi_X(\Gamma)$). Pretpostavimo da je *chase* algoritmom, po osnovu *fd* $Y \to A$, modifikovana relacija r_{k-1} i dobijena nova relacija sekvence r_k ($k \in \{2,..., p\}$). Ova modifikacija znači da u relaciji r_{k-1} moraju postojati dve torke na osnovu kojih je data modifikacija napravljena: $(\exists u_j'', u_l'' \in r_{k-}$

$_{1})(u_j\text{"}[Y] = u_l\text{"}[Y] \wedge u_j\text{"}(A) \neq u_l\text{"}(A))$. Iz činjenica $Y \setminus X \neq \emptyset$ i $u_j\text{"}[Y] = u_l\text{"}[Y]$, odnosno $u_j\text{"}[Y \setminus X] = u_l\text{"}[Y \setminus X]$ i načina formiranja relacije r_e (uslov (3.14), po kojem je $u_j[Y \setminus X] \neq u_l[Y \setminus X]$), zaključuje se da u *chase* sekvenci mora postojati relacija r_m, takva da je $m \in \{1,...,k - 1\}$, $r_m \models X \to Y \setminus X$ i $r_{m-1} \not\models X \to Y \setminus X$, pri čemu $X \to Y \setminus X$ može biti kako potpuna, tako i nepotpuna *fd*. Sledi da je $X \to Y \setminus X \in \Gamma^+$, odnosno $X \to Y \in \Gamma^+$, što je kontradiktorno pretpostavci teoreme (3.12). Na taj način je pobijena i hipoteza (3.10). □

Primer 3.11. Dati su $R = ABCDE$, $\Gamma = \{A \to C, B \to C, CD \to E\}$ i $X = ABDE$. U okviru teoreme 3.6. je pokazano da $SAT(X, \Pi_X(\Gamma)) \not\subseteq \Pi_X(SAT(R, \Gamma))$, odakle sledi da je narušena i polazna pretpostavka teoreme 3.9. Pri tome je narušen uslov (3.9), odnosno (3.12), jer za potpunu *fd* $CD \to E \in \Gamma^+$ važi da $CD \not\subseteq X \wedge X \to CD \in \Gamma^+$. □

Primer 3.12. Dati su $R = ABC$, $\Gamma = \{A \to B, A \to C\}$ i $X = AB$. U ovom slučaju je uslov (3.9) zadovoljen, tako da važi $\Pi_X(SAT(R, \Gamma)) = SAT(X, \Pi_X(\Gamma))$, pri čemu je $\Pi_X(\Gamma) \equiv \{A \to B\}$. □

3.3.2. Izvedene F-zavisnosti i projekcija familije funkcionalnih zavisnosti

U ovoj tački se uvodi pojam izvedene F-zavisnosti, kao specijalnog slučaja *egd*. Pomoću izvedenih F-zavisnosti se može izraziti potreban i dovoljan uslov da projekcija *fd*-familije bude, takođe, *fd*-familija. Za definisanje pojma izvedene F-zavisnosti je bitan pojam lanca, koji će u nastavku biti definisan.

Definicija 3.6. Dat je univerzalni skup obeležja U.

1. Izraz oblika (A), pri čemu je $A \in U$ obeležje iz univerzalnog skupa, se naziva ***lanac*** nad skupom U. Umesto izraza (A) se, skraćeno, može koristiti izraz A.
2. Neka su $\xi_1,..., \xi_n$, $n \in \mathbf{N}$, konačni skupovi lanaca L^i_j, nad skupom obeležja U: $(\forall i \in \{1,...,n\})(\xi_i = \{L^i_1,..., L^i_{n_i}\})$. Izraz oblika $(\xi_1,.., \xi_n)$ predstavlja ***lanac*** nad U.
3. ***Lanac*** je sve i samo ono što se dobije primenom, konačno mnogo puta, pravila 1. i 2. □ **[GZ]**

Primer 3.13. Neka je dat univerzalni skup obeležja $U = ABCDEFGH$. Obeležja A, B, C i D predstavljaju lance i skupove lanaca, istovremeno. (A, B) i (C, D) predstavljaju lanace. $\{(A, B), (C, D)\}$ i F su dva skupa lanaca, dok je $(\{(A, B), (C, D)\}, F)$ lanac ova dva skupa. □

Definicija 3.7. Neka je dat univerzalni skup obeležja U, skup $Y \subseteq U$ i skup lanaca ξ, $attr(\xi) \subseteq U$, pri čemu je sa $attr(\xi)$ označen skup svih obeležja, koja učestvuju u skupu lanaca ξ. Izraz oblika $\xi \to Y$ se naziva izvedena F-zavisnost (***ifd***) nad skupom obeležja U. □ **[GZ]**

Skup svih mogućih *ifd* nad skupom U je $IFD(U) = \{\xi \to Y \mid attr(\xi) \subseteq U, Y \subseteq U\}$. $IFD(U)$ predstavlja beskonačan, prebrojiv skup zavisnosti.

U cilju definisanja interpretacije izvedene F-zavisnosti, uvodi se sledeća notacija. Neka je data proizvoljna relacija $r \in SAT(U)$ i neka su $u, v \in r$:

- $u\, X_r\, v$, pri čemu je $X \subseteq U$, označava činjenicu da je $u[X] = v[X]$,
- $u\, L_r\, v$, pri čemu je $L = (\xi_1,.., \xi_n)$ lanac nad skupom obeležja U, označava činjenicu da:

$$(\exists u_1,..., u_{n-1} \in r)(u\ \xi_{1_r}\, u_1 \wedge u_1\ \xi_{2_r}\, u_2 \wedge ... \wedge u_{n-2}\ \xi_{n-1_r}\, u_{n-1} \wedge u_{n-1}\ \xi_{n_r}\, v),$$

- $u\, \xi_r\, v$, pri čemu je $\xi = \{L_1,...,L_k\}$ skup lanaca, označava činjenicu da:

$$(\forall L \in \{L_1,...,L_k\})(u\, L_r\, v).$$

Ukoliko se relacija r podrazumeva, onda se izrazi $u\, X_r\, v$, $u\, L_r\, v$ i $u\, \xi_r\, v$, skraćuju, redom, kao $u\, X\, v$, $u\, L\, v$ i $u\, \xi\, v$.

Definicija 3.8. Za relaciju $r(U)$ i *ifd* $\xi \rightarrow Y$ važi da $r \models \xi \rightarrow Y$ ako:

$$(\forall u, v \in r)(u\, \xi\, v \Rightarrow u[Y] = v[Y]). \qquad \textbf{[GZ]}$$

Primer 3.14. Neka su dati $U = ABCDE$ i *ifd* $\gamma : \{(A, B), (C, D)\} \rightarrow E$. *Ifd* γ se interpretira na sledeći način: $(\forall u, v \in r)(\exists u_1, u_2 \in r)((u(A) = u_1(A) \wedge u_1(B) = v(B) \wedge u(A) = u_2(A) \wedge u_2(B) = v(B)) \Rightarrow u(E) = v(E))$. Na slici 3.14. je prikazana relacija r, takva da je $r \models \gamma$. Može se proveriti da jedine dve torke za koje važi pretpostavka *ifd* su t_1 i t_3: $t_1\{(A, B), (C, D)\}_r\, t_3$. U tom slučaju važi i $t_1\, E_r\, t_3$, odakle sledi i činjenica $r \models \gamma$. □

Primer 3.15. Dat je skup obeležja $X = ABDE$ i *ifd* $\{(A, B, A), D\} \rightarrow E$. Neka je $r \in SAT(X)$ relacija, prikazana na slici 3.11. Može se utvrditi da $r \not\models \{(A, B, A), D\} \rightarrow E$, pošto važi da je $t_1\, \{(A, B, A), D\}\, t_4$, ali je $t_1(E) \neq t_4(E)$. □

r	A	B	C	D	E
t_1	$\boldsymbol{a_1}$	b_1	$\boldsymbol{c_1}$	d_1	$\boldsymbol{e_1}$
t_2	$\boldsymbol{a_1}$	$\boldsymbol{b_2}$	c_2	d_2	e_2
t_3	a_2	$\boldsymbol{b_2}$	c_3	$\boldsymbol{d_3}$	$\boldsymbol{e_1}$
t_4	a_3	b_3	$\boldsymbol{c_1}$	$\boldsymbol{d_3}$	e_2

Slika 3.14.

Može se videti da je bilo koja *fd* $X \rightarrow Y$ istovremeno *ifd* $\{X\} \rightarrow Y$, odnosno $X \rightarrow Y$, saglasno konvenciji da je $\{X\} = X$. Na taj način, važi $FD(U) \subseteq IFD(U)$.

Moguće je, takođe, dokazati činjenicu da je $EGD(U) \models IFD(U)$, tj. da se svaka *ifd* može prikazati putem odgovarajuće *egd*.

Primer 3.16. *Ifd* $\{(A, B), (C, D)\} \rightarrow E$ je ekvivalentna *egd* $<T(U), Eq_E(u_1, u_3)>$, pri čemu je tablo $T(U)$ prikazan na slici 3.15. □

$T(U)$	A	B	C	D	E
P_1	x_1	y_1	z_1	w_1	u_1
P_2	x_1	y_2	z_2	w_2	u_2
P_3	x_2	y_2	z_3	w_3	u_3
P_4	x_3	y_3	z_1	w_3	u_4

Slika 3.15.

Može se dokazati da za *ifd* važe sledeća pravila izvođenja (iz kojih, kao specijalan slučaj, mogu biti izvedena pravila za *fd FD1-FD3*):

- *IFD1* - ***refleksivnost***: $\xi \rightarrow Y,\ W \subseteq Y \ \vdash\ \xi \rightarrow W$,
- *IFD2* - ***uniranje***: $\xi_1 \rightarrow Y,\ \xi_2 \rightarrow Z \ \vdash\ \xi_1 \cup \xi_2 \rightarrow YZ$,
- *IFD3* - ***lančanje***: $\xi_1 \rightarrow Y,\ \xi_2 \rightarrow Y \ \vdash\ \{(\xi_1, \xi_2)\} \rightarrow Y$ i
- *IFD4* - ***tranzitivnost***: $\xi \rightarrow Y,\ Y \rightarrow Z \vdash \xi \rightarrow Z$.

Primer 3.17. Neka su dati R, Γ i X iz primera 3.10: $R = ABCDE$, $\Gamma = \{A\to C, B\to C, CD\to E\}$, $X = ABDE$. Iz $A\to C$ i $B\to C$, na osnovu *IFD3*, sledi $\{(A, B)\}\to C$. Iz $\{(A, B)\}\to C$ i $D\to D$, po pravilu *IFD2*, se izvodi $\{(A, B), D\}\to CD$, što u kombinaciji sa $CD\to E$, po *IFD4*, daje $\{(A, B), D\}\to E$. Zaključuje se da važi $\Gamma \models \{(A, B), D\}\to E$.

S druge strane je $\Pi_X(\Gamma) \not\models \{(A, B), D\}\to E$, iako je ispunjeno $attr(\{(A, B), D\}\to E) \subseteq X$. Za relaciju r, prikazanu na slici 3.16, važi da $r \models \Pi_X(\Gamma) = \{AD\to E, BD\to E\}$ i $r \not\models \{(A, B), D\}\to E$. □

r	*A*	*B*	*D*	*E*
t_1	*1*	*1*	*1*	*1*
t_2	*1*	*2*	*2*	*2*
t_3	*2*	*2*	*1*	*3*

Slika 3.16.

Definicija 3.9. Neka su dati skupovi lanaca ξ_i, ξ_j i obeležje $A \in U$. ***Zamena obeležja** A* ***skupom lanaca** ξ_i **u skupu lanaca*** ξ_j, u oznaci $\mu_{\xi_i\to A}[\xi_j]$, se definiše na sledeći način:

1. Neka je dat lanac $L = (A)$, odnosno $L = A$. Tada je $\mu_{\xi_i\to A}[L] = (\xi_i)$.
2. Neka je dat lanac $L = (B)$, odnosno $L = B$, pri čemu je $B \neq A$. Tada je $\mu_{\xi_i\to A}[L] = L$.
3. Neka je dat lanac $L = (\xi'_1, ..., \xi'_k)$. Tada je $\mu_{\xi_i\to A}[L] = (\mu_{\xi_i\to A}[\xi'_1], ..., \mu_{\xi_i\to A}[\xi'_k])$.
4. Neka je $\xi_j = \{L_1,...,L_m\}$. Tada je $\mu_{\xi_i\to A}[\xi_j] = \{\mu_{\xi_i\to A}[L_1],...,\mu_{\xi_i\to A}[L_k]\}$. □

Primer 3.18. Neka su dati skupovi lanaca $\xi = \{(A, B), (C, D), A\}$ i $\xi' = \{(E, F)\}$. Zamena obeležja A u ξ, skupom ξ', je: $\mu_{\xi'\to A}[\xi] = \mu_{\xi'\to A}[\{(A, B), (C, D), A\}] = \{\mu_{\xi'\to A}[(A, B)], \mu_{\xi'\to A}[(C, D)], \mu_{\xi'\to A}[A]\} = \{(\mu_{\xi'\to A}[A], \mu_{\xi'\to A}[B]), (\mu_{\xi'\to A}[C], \mu_{\xi'\to A}[D]), (\{(E, F)\})\} = \{(\{(E, F)\}, B), (C, D), (\{(E, F)\})\}$. □

Lema 3.5. Neka su date *ifd* $\xi_i \to A$ i $\xi_j \to Y$. Važi sledeća tvrdnja:

(3.16) $$\{\xi_i \to A, \xi_j \to Y\} \models \mu_{\xi_i\to A}[\xi_j]\to Y.$$

Dokaz. Ako $A \notin attr(\xi_j)$, tada je $\mu_{\xi_i\to A}[\xi_j] = \xi_j$ pa je dokaz trivijalan. Neka je $A \in attr(\xi_j)$ i neka je $r \in SAT(U)$ relacija za koju važi $r \models \{\xi_i \to A, \xi_j \to Y\}$. Treba dokazati da je $r \models \mu_{\xi_i\to A}[\xi_j]\to Y$. Neka su $u, v \in r$ proizvoljno odabrane torke za koje važi $u\ \mu_{\xi_i\to A}[\xi_j]\ v$. Za svaku pojavu skupa lanaca ξ_i u okviru ξ_j, u interpretaciji skupa lanca ξ_j će se pojaviti izraz oblika $u'\ \xi_i\ u''$, pri čemu su $u', u'' \in r$. Na osnovu *ifd* $\xi_i \to A$, iz $u'\ \xi_i\ u''$ sledi da važi $u'\ A\ u''$. S obzirom na način formiranja skupa lanaca $\mu_{\xi_i\to A}[\xi_j]$ iz skupa ξ_j, može se zaključiti da važi uslov $u\ \xi_j\ v$. Na osnovu *ifd* $\xi_j \to Y$ se zaključuje da važi $u\ Y\ v$, što znači da $(\forall u, v \in r)(u\ \mu_{\xi_i\to A}[\xi_j]\ v \Rightarrow u\ Y\ v)$, odnosno $r \models \mu_{\xi_i\to A}[\xi_j]\to Y$. □

Primer 3.19. Neka je dat skup *ifd* $\Sigma = \{\{(B, C)\}\to A, \{A, (C, D)\}\to E\}$. Važi da je $\Sigma \models \{(\{(B, C)\}), (C, D)\}\to E$, pošto je $\{(\{(B, C)\}), (C, D)\} = \mu_{\{(B, C)\}\to A}[\{A, (C, D)\}]$. Neka je data relacija $r \in SAT(ABCDE)$ i neka su $u, v \in r$ torke za koje važi $u\ \{(\{(B, C)\}), (C, D)\}\ v$. Ovaj izraz se interpretira na sledeći način: $u\ \{(B, C)\}\ v \wedge u\ (C, D)\ v$. Iz $\{(B, C)\}\to A$ sledi da je $u\ A\ v \wedge u\ (C, D)\ v$, odnosno $u\ \{A, (C, D)\}\ v$, a na osnovu $\{A, (C, D)\}\to E$ proizilazi $u\ E\ v$. □

U nastavku rada će izraz oblika $\mu_{\xi_1\to A_1}[\mu_{\xi_2\to A_2}[...\mu_{\xi_n\to A_n}[\xi]...]]$ biti skraćeno zapisivan u obliku: $\mu_{\xi_1\to A_1,\ \xi_2\to A_2,...,\ \xi_n\to A_n}[\xi]$.

Primeri 3.10. i 3.17. ilustruju situaciju u kojoj projekcija posmatrane *fd*-familije nije *fd*-familija. U isto vreme, projektovanjem skupa *fd* Γ na skup obeležja X, dolazi do narušavanja *ifd* γ, za koju važi da je $attr(\gamma) \subseteq X$. U ovim činjenicama leži motivacija za definisanje potrebnog i dovoljnog uslova da projekcija *fd*-familije bude takođe *fd*-familija.

Neka je $\Gamma \subseteq IFD(U)$. Simbolom Γ^{ifd} će biti označen skup svih *ifd* iz $IFD(U)$, koje su logička posledica skupa Γ:

(3.17) $$\Gamma^{ifd} = \{\gamma \in IFD(U) \mid \Gamma \models \gamma\}.$$

Definicija 3.10. Projekcija (***restrikcija***) skupa *ifd* $\Gamma \subseteq IFD(U)$ na skup obeležja $X \subseteq U$, u oznaci $\Pi_X(\Gamma^{ifd})$, odnosno $\Gamma^{ifd}|_X$, je skup $\Pi_X(\Gamma^{ifd}) = \{\gamma \in \Gamma^{ifd} \mid attr(\gamma) \subseteq X\}$. **[GZ]**

Treba primetiti da za skup *fd* $\Gamma \subseteq FD(U)$ važi $\Gamma^+ \subseteq \Gamma^{ifd}$ i $\Pi_X(\Gamma) \subseteq \Pi_X(\Gamma^{ifd})$ jer je $FD(U) \subseteq IFD(U)$.

Teorema 3.10. Neka je data *fd*-familija $SAT(R, \Gamma)$, $R \subseteq U$, $\Gamma \subseteq FD(R)$ i skup obeležja $\varnothing \neq X \subseteq R$. Neka je ispunjeno da $(\forall B \in R \setminus X)(\ |dom(B)| \geq |dom[X]|\ )$, pri čemu je za $X = A_1 ... A_p$, $dom[X] = dom(A_1) \times ... \times dom(A_p)$. Sledeće tvrdnje su ekvivalentne:

1. $\Pi_X(SAT(R, \Gamma)) = SAT(X, \Delta)$ i
2. $\Delta^+ = \Pi_X(\Gamma^+) \wedge \Pi_X(\Gamma^{ifd}) \subseteq \Delta^{ifd}$. **[GZ]**

Dokaz. $(1 \Rightarrow 2)$ Činjenica $\Delta^+ = \Pi_X(\Gamma^+)$ direktno sledi na osnovu teoreme 3.8. Da bi se dokazala inkluzija $\Pi_X(\Gamma^{ifd}) \subseteq \Delta^{ifd}$, uzima se proizvoljna *ifd* $\gamma \in \Pi_X(\Gamma^{ifd})$. Pošto je, na osnovu definicije projekcije skupa *ifd*, $\Pi_X(\Gamma^{ifd}) \subseteq \Gamma^{ifd}$, znači da je $\gamma \in \Gamma^{ifd}$, odakle sledi da $(\forall r \in SAT(R, \Gamma))(r \models \gamma)$. Iz činjenice $attr(\gamma) \subseteq X$ proizilazi da $(\forall s \in \Pi_X(SAT(R, \Gamma)))(s \models \gamma)$. Na osnovu 1. se zaključuje da $(\forall s \in SAT(R, \Delta))(s \models \gamma)$, što znači da $\Delta \models \gamma$, odnosno $\gamma \in \Delta^{ifd}$.

$(2 \Rightarrow 1)$ Činjenica $\Pi_X(SAT(R, \Gamma)) \subseteq SAT(X, \Delta)$ sledi direktno iz pretpostavke $\Delta^+ = \Pi_X(\Gamma^+)$. Da bi se dokazala tvrdnja 1, neophodno je pokazati da važi $SAT(X, \Delta) \subseteq \Pi_X(SAT(R, \Gamma))$. Posmatra se proizvoljno odabrana relacija $s \in SAT(X, \Delta)$. Treba dokazati da, pod navedenim uslovima teoreme, važi da $(\exists r \in SAT(R, \Gamma))(s = \Pi_X(r))$, odakle direktno sledi da je $s \in \Pi_X(SAT(R, \Gamma))$. Koristeći se obratom po kontrapoziciji, uvodi se suprotna pretpostavka:

(3.18) $$(\forall r \in SAT(R, \Gamma))\ (s \neq \Pi_X(r)).$$

Neka se, na osnovu relacije $s = \{t_1,...,t_n\}$, formira relacija $r_e \in SAT(R)$, $r_e = \{u_1,...,u_n\}$ $(|s| = |r_e|)$. Pošto važi $(\forall B \in R \setminus X)(\ |dom(B)| \geq |dom[X]|\ )$, r_e se može formirati na sledeći način:

(3.19) $$(\forall i \in \{1,...,n\})(u_i[X] = t_i) \wedge (\forall i, j \in \{1,...,n\})(\forall A \in R \setminus X)(i \neq j \Rightarrow u_i(A) \neq u_j(A)).$$

Treba zapaziti da je $\Pi_X(r_e) = s$ i $r_e \models \Delta^{ifd}$. Nad relacijom r_e se može primeniti *chase* algoritam, koji generiše sekvencu $(r_e = r_0, r_1,...,r_{p-1}, r_p)$. Pošto je $\Gamma \subseteq IFD(R)$, rezultat primene *chase* algoritma je relacija $r_p = CH_\Gamma(r_e) \in SAT(R, \Gamma)$, pri čemu je $|CH_\Gamma(r_e)| \leq |r_e|$. Saglasno pretpostavci (3.18), važi $s \neq \Pi_X(CH_\Gamma(r_e))$, što je moguće samo u slučaju da je *chase* algoritam promenio vrednost makar jednog obeležja $A \in X$ relacije r_e, saglasno nekoj netrivijalnoj *ifd* $\xi \to A \in \Gamma^{ifd} \setminus \Delta^{ifd}$, tako da je ispunjeno:

(3.20) $$(\exists u_j' \in CH_\Gamma(r_e))(u_j'[X] \notin s).$$

Neka je *chase* algoritam modifikovao vrednost obeležja $A \in X$, salgasno *ifd* $\xi \to A$, u k-tom koraku ($1 \le k \le p$), što znači da

(3.21) $(\forall i \in \{0,..., k - 1\})(r_i \not\models \xi \to A) \wedge (\forall i \in \{k,..., p\})(r_i \models \xi \to A)$.

Pošto je $\xi \to A \notin \Delta^{ifd}$, na osnovu stava 2, sledi da $\xi \to A \notin \Pi_X(\Gamma^{ifd})$. Prema definiciji za $\Pi_X(\Gamma^{ifd})$, i činjenici da $\xi \to A \in \Gamma^{ifd}$, proizilazi da mora biti $attr(\xi) \setminus X \neq \emptyset$. Neka je $Y = \{B_1,...,B_l\} = attr(\xi) \setminus X$, $l \ge 1$. Iz uslova (3.21) sledi da:

(3.22) $(\forall B_i \in Y)(\exists u_p, u_q \in r_k)(u_p(B_i) = u_q(B_i))$.

S druge strane, na osnovu (3.19), važi da $(\forall B_i \in Y)(\forall u_p, u_q \in r_0)(u_p(B_i) \neq u_q(B_i))$, što znači da je *chase* algoritmom u koraku m_i ($1 \le m_i < k$) modifikovana neka torka relacije r_{m_i} iz *chase* sekvence, u cilju ostvarenja uslova (3.22), iz čega proizilazi da za svako obeležje $B_i \in Y$ mora postojati netrivijalna *ifd* $\xi_i \to B_i \in \Gamma^{ifd}$, takva da važi:

(3.23) $(\forall j \in \{0,..., m_i - 1\})(r_j \not\models \xi_i \to B_i) \wedge (\forall j \in \{m_i,..., p\})(r_i \models \xi_i \to B_i) \wedge attr(\xi_i) \subseteq X$.

Ukoliko bi se dogodilo da za neku *ifd* $\xi_i \to B_i$, važi da $attr(\xi_i) \not\subseteq X$, onda bi se analiza, prezentovana putem kriterijuma (3.22) i (3.23), rekurzivno ponovila, što bi, uz primenu leme 3.5, značilo da postoji neka druga *ifd* $\xi_i' \to B_i$, takva da je $attr(\xi_i') \subseteq X$.

Iz *ifd* $\xi \to A$, $\xi_1 \to B_1,..., \xi_l \to B_l \in \Gamma^{ifd}$, primenom leme 3.5. l puta, izvodi se *ifd* $\xi' \to A \in \Gamma^{ifd}$, pri čemu je $\xi' = \mu_{\xi_1 \to B_1,..., \xi_l \to B_l}[\xi]$, za koju važi da je $attr(\xi') \subseteq X$. Iz ove činjenice sledi da je $\xi' \to A \in \Pi_X(\Gamma^{ifd})$ pa je, po pretpostavci 2, $\xi' \to A \in \Delta^{ifd}$, pri čemu je, saglasno (3.21), $r_e \not\models \xi' \to A$. Ovo je, međutim, kontradiktorno činjenici $r_e \models \Delta^{ifd}$, što obara suprotnu pretpostavku (3.18). □

Primer 3.20. Neka su dati R, Γ i X iz primera 3.17: $R = ABCDE$, $\Gamma = \{A \to C, B \to C, CD \to E\}$, $X = ABDE$. U okviru primera 3.17. je pokazana implikacija $\Gamma \models \{(A, B), D\} \to E$, a pošto je $attr(\{(A, B), D\} \to E) \subseteq X$, važi da je $\{(A, B), D\} \to E \in \Pi_X(\Gamma^{ifd})$. Neka je $\Delta = \Pi_X(\Gamma)$. U primeru 3.17. je pokazano da $\Delta \not\models \{(A, B), D\} \to E$, odnosno da $\{(A, B), D\} \to E \notin \Delta^{ifd}$. Proizilazi, dakle, da $\Pi_X(\Gamma^{ifd}) \not\subseteq \Delta^{ifd}$, što znači da je, saglasno teoremi 3.10, $\Pi_X(SAT(R, \Gamma)) \neq SAT(X, \Delta)$.

r_e	A	B	C	D	E
t_1	1	1	1	1	1
t_2	1	2	2	2	2
t_3	2	2	3	1	3

Slika 3.17.

Neka je na osnovu relacije r, prikazane na slici 3.16, za koju važi $r \in SAT(X, \Delta)$, formirana, prema uslovu (3.19), relacija r_e (prikazana na slici 3.17). Pošto je $u_1\ (A, B)\ u_3 \wedge u_1\ D\ u_3$, na osnovu $\{(A, B), D\} \to E \in \Gamma^{ifd}$, vrši se *chase* modifikacija relacije r_e i formira relacija r_1, prikazana na slici 3.18. Za r_1, međutim, važi da $\Pi_X(r_1) \neq r$. □

r_e	A	B	C	D	E
t_1	1	1	1	1	1
t_2	1	2	2	2	2
t_3	2	2	3	1	1

Slika 3.18.

3.4. Prirodni spoj familija generalizovanih zavisnosti

U ovom delu se razmatra problem zatvorenosti prirodnog spoja konačno specifikabilnih familija pojava. Postoje konačno specifikabilne familije, čiji prirodni spoj ne predstavlja konačno specifikabilnu familiju. Biće utvrđeni potrebni i dovoljni uslovi da prirodni spoj *fd*-familija bude, takođe, *fd*-familija. Rešavanje problema zatvorenosti prirodnog spoja konačno specifikabilnih familija predstavlja osnov za obezbeđenje spojivosti šema relacija bez gubitaka informacija.

Definicija 3.11. Konačno specifikabilne familije pojava $SAT(R, \Sigma)$ i $SAT(S, \Delta)$, pri čemu je $RS \subseteq U$, a Σ i Δ skupovi zavisnosti određenog tipa, su ***zatvorene s obzirom na operator prirodnog spajanja***, ako važi da postoji konačno specifikabilna familija $SAT(RS, \Omega)$, takva da je $SAT(R, \Sigma) \rhd\lhd SAT(S, \Delta) = SAT(RS, \Omega)$, pri čemu su sve zavisnosti iz skupa Ω istog tipa kao i zavisnosti iz skupova Σ i Δ. □

U cilju razmatranja problema konačne specifikabilnosti prirodnog spoja dve familije pojava, uvodi se pojam proširenja tabloa.

Definicija 3.12. Neka je dat tablo $T_X(X)$ i neka je $X \subseteq R \subseteq U$. Tablo $T_R(R)$ predstavlja ***proširenje*** tabloa $T_X(X)$, u oznaci $T_R(R) \supset T_X(X)$, ako je ispunjeno:

1. $\Pi_X(T_R) = T_X$,
2. $|T_R| = |T_X|$ i
3. $(\forall P_i, P_j \in T_R)(\forall A \in R \setminus X)(f_{p_i}[X] \neq f_{p_j}[X] \Rightarrow f_{p_i}(A) \neq f_{p_j}(A))$. □ **[Hu]**

Lema 3.6. Date su *egd* $\gamma_X : <T_X(X), Eq_A(a_1, a_2)>$ i $\gamma_R : <T_R(R), Eq_A(a_1, a_2)>$, pri čemu je $T_R(R) \supset T_X(X)$. Važi da je:

$$(\forall r \in SAT(R))(r \models \gamma_R \Leftrightarrow \Pi_X(r) \models \gamma_X). \tag{3.24}$$

Dokaz. ($\Rightarrow$) Neka je za $r \in SAT(R)$ zadovoljeno da $r \models \gamma_R$. Neka je v' interpretacija tabloa $T_X(X)$, za koju važi da je $v'(T_X) \subseteq \Pi_X(r)$. Prema tvrdnji 1, definicije 3.12, sledi da je $v'(\Pi_X(T_R)) \subseteq \Pi_X(r)$. Na osnovu tvrdnje 2. i 3, definicije 3.12, i činjenice da je interpretacija tabloa funkcija, može se zaključiti da postoji interpretacija v, tabloa T_R, takva da je $v|_X = v'$ i $v(T_R) \subseteq r$. Iz $r \models \gamma_R$ i $v(T_R) \subseteq r$, sledi da je $v(a_1) = v(a_2)$, a pošto je $A \in X$, znači da važi $v'(a_1) = v'(a_2)$, odakle sledi da je $\Pi_X(r) \models \gamma_X$.

($\Leftarrow$) Neka je za $r \in SAT(R)$ zadovoljeno da $\Pi_X(r) \models \gamma_X$. Neka je v interpretacija tabloa $T_R(R)$, za koju važi da je $v(T_R) \subseteq r$. Sledi da je $\Pi_X(v(T_R)) \subseteq \Pi_X(r)$. Iz definicije 3.12, sledi da je $\Pi_X(v(T_R)) = v(\Pi_X(T_R)) = v(T_X) \subseteq \Pi_X(r)$, a iz $\Pi_X(r) \models \gamma_X$ proizilazi da je $v(a_1) = v(a_2)$, odakle sledi tvrdnja $r \models \gamma_R$. $\Box$

Teorema 3.11. Prirodni spoj dve konačno specifikabilne *egd*-familije pojava $SAT(R, \Sigma)$ i $SAT(S, \Delta)$, pri čemu je $RS \subseteq U$, predstavlja konačno specifikabilnu *gd*-familiju. **[Hu]**

Dokaz. Neka je $W = RS$ i neka je na osnovu Σ i Δ formiran skup *gd* Ω, na sledeći način:

$$\Omega = \Sigma_W \cup \Delta_W \cup \{\rhd\lhd(R, S)\},$$

pri čemu su:

$$\Sigma_W = \{<T_W(W), Eq_A(a_1, a_2)> \mid (\exists <T_R(R), Eq_A(a_1, a_2)> \in \Sigma)(T_W \supset T_R)\} \text{ i}$$

$$\Delta_W = \{<T_W(W), Eq_A(a_1, a_2)> \mid (\exists <T_S(S), Eq_A(a_1, a_2)> \in \Delta)(T_W \supset T_S)\}.$$

Da bi se dokazala tvrdnja teoreme, dovoljno je dokazati da važi:

$$SAT(W, \Omega) = SAT(R, \Sigma) \rhd\lhd SAT(S, \Delta).$$

Pošto je skup Ω konačan, sledi da je $SAT(W, \Omega)$ konačno specifikabilna *gd* familija.

($\Rightarrow$) Dokazuje se inkluzija $SAT(W, \Omega) \subseteq SAT(R, \Sigma) \rhd\lhd SAT(S, \Delta)$. Neka je $r \in SAT(W, \Omega)$. Iz $r \models \Sigma_W$, na osnovu leme 3.6, sledi da $\Pi_R(r) \models \Sigma$, odnosno $\Pi_R(r) \in SAT(R, \Sigma)$. Analogno, važi $\Pi_S(r) \models \Delta$, tj. $\Pi_S(r) \in SAT(S, \Delta)$. Pošto je $r \models \rhd\lhd(R, S)$, važi da je $r = \Pi_R(r) \rhd\lhd \Pi_S(r)$, odakle sledi da $r \in SAT(R, \Sigma) \rhd\lhd SAT(S, \Delta)$.

($\Leftarrow$) Dokazuje se inkluzija $SAT(R, \Sigma) \rhd\lhd SAT(S, \Delta) \subseteq SAT(W, \Omega)$. Neka je $r \in SAT(R, \Sigma) \rhd\lhd SAT(S, \Delta)$. Sledi da $r \models \rhd\lhd(R, S)$. Sledi, takođe, da $(\exists r_R \in SAT(R, \Sigma))(\exists r_S \in SAT(S, \Delta))(r = r_R \rhd\lhd r_S)$, odakle se zaključuje da je $\Pi_R(r) = r_R$ i $\Pi_S(r) = r_S$. Pošto važi $r_R \models \Sigma$, na osnovu leme 3.6, sledi $r \models \Sigma_W$. Analogno se zaključuje da $r \models \Delta_W$. Iz $r \models \{\rhd\lhd(R, S)\} \cup \Sigma_W \cup \Delta_W$, proizilazi da je $r \in SAT(W, \Omega)$. $\Box$

Posledica 3.2. Za *fd*-familije $SAT(R, \Gamma_1)$ i $SAT(S, \Gamma_2)$, pri čemu je $RS \subseteq U$, važi da je:

$$SAT(R, \Gamma_1) \rhd\lhd SAT(S, \Gamma_2) = SAT(RS, \Gamma),$$

gde je $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \{\rhd\lhd(R, S)\}$.

Dokaz. Sledi iz teoreme 3.11. $\Box$

Za razliku od konačno specifikabilnih *egd*-familija, prirodni spoj konačno specifikabilnih *gd*-familija ne mora biti konačno specifikabilan! Dokaz ove tvrdnje sledi na osnovu sledeće teoreme, čiji dokaz se može naći u **[Hu]**.

Teorema 3.12. Postoje *mvd*-familije pojava $SAT(R, \Sigma)$ i $SAT(S, \Delta)$, čiji prirodni spoj ne predstavlja konačno specifikabilnu familiju. $\Box$ **[Hu]**

Iz teoreme 3.12. sledi da prirodni spoj konačno specifikabilnih *gd*-familija nije, u opštem slučaju, zatvoren s obzirom na prirodno spajanje. Bez obzira na važenje teoreme 3.11, isti zaključak se može izvesti i za konačno specifikabilne *egd*-familije. Činjenica da konačno specifikabilne *egd*-familije nisu zatvorene s obzirom na prirodno spajanje, proizilazi na osnovu sledeće teoreme.

Teorema 3.13. Postoje *fd*-familije $\mathcal{F}_1 = SAT(R, \Gamma_1)$ i $\mathcal{F}_2 = SAT(S, \Gamma_2)$, čiji prirodni spoj ne predstavlja *fd*-familiju.

Dokaz. Saglasno posledici 3.2, dovoljno je pronaći *fd*-familije $\mathcal{F}_1$ i $\mathcal{F}_2$, takve da važi $\Gamma_1 \cup \Gamma_2 \not\models \bowtie(R, S)$. Neka je $R = AB$, $\Gamma_1 = \{A \to B\}$, $S = BC$ i $\Gamma_2 = \{C \to B\}$. Na slici 3.19. je prikazana relacija r, za koju važi da $r \models \Gamma_1 \cup \Gamma_2$, $r \not\models \bowtie(R, S)$, $\Pi_R(r) \in \mathcal{F}_1$ i $\Pi_S(r) \in \mathcal{F}_2$. □

r	A	B	C
	a_1	b_1	c_1
	a_2	b_1	c_2

Slika 3.19.

Posledica 3.3. Sledeće familije pojava nisu zatvorene s obzirom na operator prirodnog spoja:

1. *fd*-familija,
2. konačno specifikabilna *egd*-familija,
3. *mvd*-familija,
4. *jd*-familija,
5. konačno specifikabilna *tgd*-familija i
6. konačno specifikabilna *gd*-familija.

Dokaz. Stav 1. je posledica teoreme 3.13. Stav 2. je posledica teoreme 3.1, dok je stav 3. dokazan u **[Hu]** (teorema 3.12). Stav 4. je posledica činjenice da je *mvd* istovremeno i *jd*, a stav 5. je posledica teoreme 3.3. Stav 6. je posledica činjenice da su *egd* i *tgd* istovremeno i *gd*. □

U nastavku će biti definisan potreban i dovoljan uslov da prirodni spoj dve *fd*-familije bude zatvoren. U tom cilju, potrebno je formulisati lemu o spojivosti dve relacije bez gubitaka informacija.

Lema 3.7. Data je relacija $r \in SAT(U)$ i *fd* $X \to Y \in FD(U)$. Ako važi $r \models X \to Y$ tada je $r = \Pi_{XY}(r) \bowtie \Pi_{X(U \setminus Y)}(r)$. **[M, Lu1]**

Dokaz. ($\Rightarrow$) Prvo će biti dokazano da važi $r \subseteq \Pi_{XY}(r) \bowtie \Pi_{X(U \setminus Y)}(r)$. Neka je $t \in r$. Tada $t[XY] \in \Pi_{XY}(r) \wedge t[X(U \setminus Y)] \in \Pi_{X(U \setminus Y)}(r)$, odakle proizilazi $t[XY]\, t[X(U \setminus Y)] \in \Pi_{XY}(r) \bowtie \Pi_{X(U \setminus Y)}(r)$, odnosno $t \in \Pi_{XY}(r) \bowtie \Pi_{X(U \setminus Y)}(r)$.

($\Leftarrow$) Neka je $t \in \Pi_{XY}(r) \bowtie \Pi_{X(U \setminus Y)}(r)$. Sledi da je $t[XY] \in \Pi_{XY}(r) \wedge t[X(U \setminus Y)] \in \Pi_{X(U \setminus Y)}(r)$. Neka su torke $t_1, t_2 \in r$, takve da važi $t_1[XY] = t[XY]$, odnosno $t_2[X(U \setminus Y)] = t[X(U \setminus Y)]$. Pošto je $t_1[X] = t[X]$ i $t[X] = t_2[X]$, tada važi $t_1[X] = t_2[X]$, a na osnovu $r \models X \to Y$ proizilazi $t_1[Y] = t_2[Y]$, tako da je $t_2[Y] = t[Y]$. Iz $t_2[X(U \setminus Y)] = t[X(U \setminus Y)]$ i $t_2[Y] = t[Y]$ se zaključuje da je $t_2 = t$, a pošto je $t_2 \in r$, znači da $t \in r$, tj. $\Pi_{XY}(r) \bowtie \Pi_{X(U \setminus Y)}(r) \subseteq r$.

Iz $r \subseteq \Pi_{XY}(r) \bowtie \Pi_{X(U \setminus Y)}(r)$ i $\Pi_{XY}(r) \bowtie \Pi_{X(U \setminus Y)}(r) \subseteq r$ sledi tvrdnja teoreme. □

Teorema 3.14. Date su *fd*-familije $\mathcal{F}_1 = SAT(R, \Gamma_1)$ i $\mathcal{F}_2 = SAT(S, \Gamma_2)$, pri čemu je $RS \subseteq U$. Sledeće tvrdnje su ekvivalentne:

1. $SAT(R, \Gamma_1) \bowtie SAT(S, \Gamma_2)$ predstavlja *fd*-familiju.
2. $SAT(R, \Gamma_1) \bowtie SAT(S, \Gamma_2) = SAT(RS, \Gamma_1 \cup \Gamma_2)$.
3. $\Gamma_1 \models R \cap S \to R \vee \Gamma_2 \models R \cap S \to S$. **[GZ]**

Dokaz. Tvrdnja teoreme će biti dokazana dokazivanjem implikacija $2 \Rightarrow 1$, $1 \Rightarrow 3$ i $3 \Rightarrow 2$.

($2 \Rightarrow 1$) Činjenica 1. sledi direktno iz 2, pošto je $SAT(RS, \Gamma_1 \cup \Gamma_2)$ *fd*-familija.

($3 \Rightarrow 2$) Prvo će biti pokazano da važi $SAT(R, \Gamma_1) \bowtie SAT(S, \Gamma_2) \subseteq SAT(RS, \Gamma_1 \cup \Gamma_2)$. Neka je data proizvoljna relacija $r \in SAT(R, \Gamma_1) \bowtie SAT(S, \Gamma_2)$. Neka je $attr(\Gamma_1)$ skup obeležja, upotrebljenih za formiranje Γ_1. Pošto je $attr(\Gamma_1) \subseteq R$, sledi da $\Pi_R(r) \models \Gamma_1$. Analogno, važi $\Pi_S(r) \models \Gamma_2$. Iz $\Pi_R(r) \models \Gamma_1$ i $\Pi_S(r) \models \Gamma_2$ proizilazi da $\Pi_R(r) \bowtie \Pi_S(r) \models \Gamma_1 \cup \Gamma_2$. Pošto je $r = \Pi_R(r) \bowtie \Pi_S(r)$, sledi da $r \in SAT(RS, \Gamma_1 \cup \Gamma_2)$.

Sada će biti pokazano da važi $SAT(RS, \Gamma_1 \cup \Gamma_2) \subseteq SAT(R, \Gamma_1) \bowtie SAT(S, \Gamma_2)$. Neka je $r \in SAT(RS, \Gamma_1 \cup \Gamma_2)$. Na osnovu pretpostavke 3. važi $r \models R \cap S \rightarrow R \vee r \models R \cap S \rightarrow S$, tako da se na osnovu leme 3.7, može zaključiti da važi $r = \Pi_R(r) \bowtie \Pi_S(r)$, što znači da $r \in SAT(R, \Gamma_1) \bowtie SAT(S, \Gamma_2)$.

($1 \Rightarrow 3$) Neka je $SAT(R, \Gamma_1) \bowtie SAT(S, \Gamma_2) = SAT(RS, \Gamma)$, pri čemu je $\Gamma \subseteq FD(RS)$. Neka je data proizvoljna relacija $r \in SAT(RS, \Gamma)$. Pošto je $r = \Pi_R(r) \bowtie \Pi_S(r)$, pri čemu $\Pi_R(r) \models \Gamma_1$ i $\Pi_S(r) \models \Gamma_2$, zaključuje se da $r \models \Gamma_1 \cup \Gamma_2$, odnosno $(\Gamma_1 \cup \Gamma_2)^+ \subseteq \Gamma^+$. Uvodi se suprotna pretpostavka: $\Gamma_1 \not\models R \cap S \rightarrow R \wedge \Gamma_2 \not\models R \cap S \rightarrow S$. Neka je $r' = \{t_1, t_2\}$, pri čemu važi:

(3.25) $$(\forall A \in (R \cap S)^+_\Gamma)(t_1(A) = t_2(A)) \text{ i}$$

(3.26) $$(\forall A \notin (R \cap S)^+_\Gamma)(t_1(A) \neq t_2(A)).$$

Lako se može utvrditi da je $r' \in SAT(RS, \Gamma)$. Na osnovu negirane pretpostavke 3. i izraza (3.25) i (3.26), zaključuje se da za najmanje dva obeleležja $A \in R \setminus S$ i $B \in S \setminus R$ mora da važi $t_1(A) \neq t_2(A)$ i $t_1(B) \neq t_2(B)$. Zaključuje se, takođe, da je $\Pi_R(r') = \{u_1, u_2\}$, pri čemu je $u_1 \neq u_2$ i $u_1[R \cap S] = u_2[R \cap S]$. Analogno je $\Pi_S(r') = \{u_3, u_4\}$, pri čemu je $u_3 \neq u_4$ i $u_3[R \cap S] = u_4[R \cap S]$. Sledi da je $u_1[R \cap S] = u_2[R \cap S] = u_3[R \cap S] = u_4[R \cap S]$, što znači da je $\Pi_R(r') \bowtie \Pi_S(r') = \{u_1 u_3, u_1 u_4, u_2 u_3, u_2 u_4\} \supset r'$, a to je kontradiktorno činjenici $r' = \Pi_R(r') \bowtie \Pi_S(r')$, odnosno polaznoj pretpostavci 1. □

4. Poglavlje

Integracija šema modula baze podataka

U delu 2.2.2.3. je opisan postupak projektovanja šema modula BP, na osnovu konceptualne šeme BP, iskazane putem tipova formi. Zadatak ovog poglavlja je da formalizuje postupak integracije dve, proizvoljno odabrane, šeme modula u jednu, integrisanu šemu modula BP. Integraciji šema modula u jedinstvenu šemu treba da prethodi analiza konzistentnosti, kojom će se utvrditi da li se šeme modula mogu integrisati, ili ne. Pri tome se pretpostavlja da nad šemama modula mogu postojati pojave, koje je krajnji korisnik formirao koristeći se prototipovima programskih podsistema. U slučaju da se analizom utvrdi da su šeme modula, sa stanovišta integracije nekonzistentne, potrebno precizirati uzroke koji su prouzrokovali nekonzistentnost. Uzroci nekonzistentnosti šema modula se, generalno, mogu svrstati u dve kategorije:

- nekonzistencije intenzionalnog nivoa (tj. nekonzistencije samih šema modula) i
- nekonzistencije ekstenzionalnog nivoa (tj. nekonzistencije pojava šema modula).

U ovom poglavlju se analiziraju obe kategorije nekonzistentnosti. Razmatranja, vezana za integraciju šema modula će se temeljiti na pretpostavakama i definicijama, uvedenim u 2. poglavlju i teoretskim analizama, prikazanim u 3. poglavlju.

Na početku poglavlja će biti uvedeni pojmovi proširenja šeme relacije, proširenja relacione šeme BP i integrativnog svojstva. Na osnovu ovih pojmova, biće definisani kriterijumi konzistentnosti šema modula i kriterijumi, koje mora da zadovolji integrisana šema. Nakon toga, u formi algoritama, biće prikazan postupak analize konzistentnosti šema modula, a na kraju će biti prezentovan algoritam integracije dve šeme modula u novu, jedinstvenu, šemu modula.

4.1. Proširenje šeme relacije i šeme baze podataka

U ovom delu rada se uvode koncepti proširenja šeme relacije i proširenja šeme BP. Pomoću ovih koncepata će biti formulisani kriterijumi međusobne konzistentnosti dve šeme modula, odnosno mogućnosti njihove integracije.

Definicija 4.1. Date su šeme relacija (R, Σ) i (X, Δ), takve da je $X \subseteq R \subseteq U$, pri čemu je U univerzalni skup obeležja. (R, Σ) predstavlja ***proširenje*** šeme (X, Δ), u oznaci $(R, \Sigma) \supset (X, \Delta)$, ako važi:

$$\Pi_X(SAT(R, \Sigma)) = SAT(X, \Delta).$$ □ **[Sc, KMo, LM4]**

Saglasno definiciji projekcije familije pojava, činjenica $(R, \Sigma) \supset (X, \Delta)$ je ekvivalentna formulama:

(4.1) $$(\forall r \in SAT(R, \Sigma))(\exists s \in SAT(X, \Delta))(s = \Pi_X(r)) \wedge$$

(4.2) $$(\forall s \in SAT(X, \Delta))(\exists r \in SAT(R, \Sigma))(s = \Pi_X(r)).$$

Intuitivno, $(R, \Sigma) \supset (X, \Delta)$ znači da se svaka pojava nad (X, Δ) može bez gubitaka informacija, ili narušavanja skupa zavisnosti Σ proširiti do pojave nad (R, Σ). Svaka pojava nad (R, Σ) sadrži, takođe, odgovarajuću pojavu nad (X, Δ).

Primer 4.1. Date su šeme relacija $(ABC, \{A \to B, B \to C\})$ i $(AC, \{A \to C\})$, pri čemu je $X = AC$ i $\Gamma = \{A \to B, B \to C\}$. Može se proveriti da su uslovi teoreme 3.9. zadovoljeni (pošto je $\{A \to C\}^+ = \Pi_{AC}(\Gamma)$ i $C \to A \notin \Gamma^+$ i za $A \to C \in \Gamma^+$ važi $A \subseteq X$), na osnovu čega se zaključuje da važi $(ABC, \{A \to B, B \to C\}) \supset (AC, \{A \to C\})$. □

Primer 4.2. Date su šeme relacija (R, Γ) i (X, Δ), $R = ABCDE$, $\Gamma = \{A \to C, B \to C, CD \to E\}$, $X = ABDE$ i $\Delta = \{AD \to E, BD \to E\}$. (Šema relacije (R, Γ) je ista kao i u slučaju dokaza teoreme 3.6). Treba zapaziti da je $\Delta^+ = \Pi_X(\Gamma^+)$. Pošto za netrivijalnu *ifd* $\{(A, B), D\} \to E$ važi da je $\{(A, B), D\} \to E \in \Pi_X(\Gamma^{ifd}) \setminus \Delta^{ifd}$ (videti primer 3.17), znači da $\Pi_X(\Gamma^{ifd}) \not\subseteq \Delta^{ifd}$. Na osnovu teoreme 3.10. sledi da je $\Pi_X(SAT(R, \Gamma)) \neq SAT(X, \Delta)$, što znači da (R, Γ) ne predstavlja proširenje šeme relacije (X, Δ), što će biti označeno kao $(R, \Gamma) \not\supset (X, \Delta)$. □

Vidljivo je da teorema 3.9. definiše dovoljan (ali ne i potreban) uslov da bude $(R, \Gamma) \supset (X, \Delta)$, pri čemu je $\Gamma, \Delta \subseteq FD(U)$. Teorema 3.10, u slučaju da je $\Gamma, \Delta \subseteq FD(U)$, definiše potreban i dovoljan uslov za $(R, \Gamma) \supset (X, \Delta)$.

Definicija 4.2. Date su relacione šeme BP (S_1, I_1) i (S_2, I_2). (S_1, I_1) predstavlja ***proširenje*** šeme (S_2, I_2), u oznaci $(S_1, I_1) \supset (S_2, I_2)$, ako:

(4.3) $$(\forall (X, \Delta) \in S_2)(\exists (R, \Sigma) \in S_1)((R, \Sigma) \supset (X, \Delta)).$$ □

Prema pristupu koji se prezentuje ovim radom, (S_1, I_1) će reprezentovati jedinstvenu (integrisanu) šemu BP, dok će (S_2, I_2) reprezentovati šemu modula. Svaka pojava šeme BP, dakle, sadrži odgovarajuću pojavu šeme modula, i obrnuto, svaka pojava šeme modula se može proširiti do pojave šeme BP.

Lokalna ograničenja svih šema relacija bilo koje šeme modula (S_n, I_n), izgenerisane putem modifikovanog algoritma sinteze (videti tačku 2.2.2.3) se iskazuju putem skupova ključeva. Imajući u vidu činjenicu da isprojektovana šema modula zadovoljava uslov *3NF*, na osnovu skupova ključeva se može formirati kanonički pokrivač polaznog skupa *fd*. Neka je za neku šemu BP (S, I) ispunjen uslov $(S, I) \supset (S_n, I_n)$, pri čemu je (S_n, I_n) šema modula BP, izgenerisana modifikovanim algoritmom sinteze. Neka su date šeme relacija $(R_i, \mathcal{K}_i) \in S_n$ i $(R_i', \mathcal{K}_i') \in S$ ($\mathcal{K}_i$ i $\mathcal{K}_i'$ su skupovi ključeva), pri čemu je ispunjeno $(R_i', \mathcal{K}_i') \supset (R_i, \mathcal{K}_i)$. Na osnovu teoreme 3.8, sledi da je

(4.4) $$\Gamma(R_i') \models \Gamma(R_i),$$

pri čemu su $\Gamma(R_i)$ i $\Gamma(R_i')$ skupovi *fd*, izvedeni, redom, na osnovu skupova ključeva $\mathcal{K}_i$ i $\mathcal{K}_i'$: $\Gamma(R_i) = \{X \to A \mid XA \subseteq R_i \wedge X \in \mathcal{K}_i\}$ i $\Gamma(R_i') = \{X \to A \mid XA \subseteq R_i' \wedge X \in \mathcal{K}_i'\}$.

Saglasno izrazu (4.4), iz uslova $(S, I) \supset (S_n, I_n)$ proizilazi da važi:

$$(4.5) \qquad \Gamma(S) \models \Gamma(S_n),$$

gde su $\Gamma(S) = \bigcup_{R_i \in S} \Gamma(R_i)$ i $\Gamma(S_n) = \bigcup_{R_i \in S_n} \Gamma(R_i)$.

Lema 4.1. Za šeme modula (S, I) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27), (2.29) i (2.37)), pri čemu je $(S, I) \supset (S_n, I_n)$, važi implikacija $\Gamma(S) \models \Gamma(S_n)$.

Dokaz. Tvrdnja teoreme sledi iz definicije 4.2. i teoreme 3.8. □

4.2. Međusobna kompatibilnost šema modula BP

Intuitivno, šeme modula su međusobno kompatibilne na intenzionalnom nivou, ukoliko se mogu proširiti (integrisati) u jedinstvenu šemu BP, a kompatibilne su na ekstenzionalnom nivou, ukoliko se sve (proizvoljne) pojave šema modula mogu proširiti do pojave jedinstvene šeme BP. U nastavku će prvo biti formalno definisani pojmovi integrativnog svojstva šeme BP i međusobne kompatibilnosti dve proizvoljne šeme BP, pri čemu se ne specificira tip ograničenja koji je zastupljen u pripadajućim šemama relacija. Nakon toga, definicija međusobne kompatibilnosti će biti primenjena na šeme modula BP, izgenerisane putem modifikovanog alogirtma sinteze. Pre same definicije integrativnog svojstva, potrebno je uvesti nove oznake.

Date su šeme relacionih BP (S_n, I_n), (S_m, I_m) i (S, I), takve da važi $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$. Za proizvoljnu šemu $(R, \mathcal{K}) \in S$ će biti označen:

- sa $Ext(R, \mathcal{K})$ skup svih šema relacija iz (S_n, I_n) i (S_m, I_m), za koje $(R, \mathcal{K})$ predstavlja proširenje:

  $$(4.6) \qquad Ext(R, \mathcal{K}) = \{(R_i, \mathcal{K}_i) \in S_m \cup S_n \mid (R, \mathcal{K}) \supset (R_i, \mathcal{K}_i)\} \text{ i}$$

- sa $SAT(Ext(R, \mathcal{K}))$ skup svih kombinacija pojava nad šemama iz $Ext(R, \mathcal{K})$, pri čemu je $Ext(R, \mathcal{K}) = \{(R_1, \mathcal{K}_1),...,(R_k, \mathcal{K}_k)\}$:

  $$(4.7) \qquad SAT(Ext(R, \mathcal{K})) = \{(r_1,..., r_k) \in SAT(R_1, \mathcal{K}_1) \times ... \times SAT(R_k, \mathcal{K}_k) \mid (\forall i \in \{1,...,k\})((R_i, \mathcal{K}_i) \in Ext(R, \mathcal{K}))\}.$$

Za proizvoljnu kombinaciju pojava $(r_1,..., r_k) \in SAT(Ext(R, \mathcal{K}))$, sa $Ext_R(r_1,..., r_k)$ će, ukoliko postoji, biti označena pojava r nad šemom relacije $(R, \mathcal{K})$, nastala proširivanjem relacija $r_1,..., r_k$:

$$(4.8) \qquad r = Ext_R(r_1,..., r_k) \Leftrightarrow r \in SAT(R, \mathcal{K}) \wedge (\forall i \in \{1,...,k\})(r_i \subseteq \Pi_{R_i}(r)).$$

Neka je $(r_1^n,...,r_p^n)$ pojava BP nad šemom (S_n, I_n), a $(r_1^m,...,r_q^m)$ pojava BP nad šemom (S_m, I_m). Sa $Ext(S)$ će biti označen skup po jedne pojave $Ext_R(r_1,..., r_k)$ za svaku šemu relacije iz skupa S, za koju $Ext_R(r_1,..., r_k)$ postoji, saglasno pojavama $(r_1^n,...,r_p^n)$ i $(r_1^m,...,r_q^m)$:

$$(4.9) \quad Ext(S) = \{Ext_R(r_1,..., r_k) \mid (R, \mathcal{K}) \in S \wedge (\forall i \in \{1,...,k\})(r_i \in \{r_1^n,...,r_p^n, r_1^m,...,r_q^m\})\}.$$

Definicija 4.3. Date su relacione šeme BP (S_n, I_n), (S_m, I_m) i (S, I), takve da važi:

$$(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n).$$

(S_n, I_n) i (S_m, I_m) zadovoljavaju ***integrativno svojstvo*** s obzirom na (S, I), u oznaci $(S, I) = Int((S_n, I_n), (S_m, I_m))$, ako za bilo koje dve pojave BP $(r_1^n, ..., r_p^n)$ i $(r_1^m, ..., r_q^m)$ nad šemama, redom, (S_n, I_n) i (S_m, I_m), postoji $Ext(S)$, takav da važi:

1. $(\forall(R, \mathcal{K}) \in S)(NonEqDk(Ext(R, \mathcal{K})) \Rightarrow (\exists r \in SAT(R, \mathcal{K}))(r = Ext_R(r_1, ..., r_k) \in Ext(S)))$ i
2. $Ext(S)$ zadovoljava sva međurelaciona ograničenja iz skupa I,

pri čemu se predikat $NonEqDk(Ext(R, \mathcal{K}))$ interpretira putem formule:

(4.10) $(\exists(R_i, \mathcal{K}_i), (R_i, \mathcal{K}_i) \in Ext(R, \mathcal{K}))(\exists K_i \in \mathcal{K}_i \setminus \mathcal{K}_j)(K_i \cap R_j \neq \emptyset)$. □

Iz definicije 4.3. se može zaključiti da dve šeme zadovoljavaju integrativno svojstvo s obzirom na treću šemu BP, ako se bilo koja njihova pojava može proširiti do pojave nad trećom šemom.

Primer 4.3. Date su šeme BP (S_m, I_m), (S_n, I_n) i (S, I), takve da je:

- $S_m = \{(AC, \{A \to C\})\}$,
- $I_m = \emptyset$,
- $S_n = \{(AB, \{A \to B\}), (BC, \{B \to C\})\}$,
- $I_n = \{AB[B] \subseteq BC[B]\}$,
- $S = \{(ABC, \{A \to B, B \to C, AB[B] \subseteq BC[B]\})\}$ i
- $I = \emptyset$.

Pod pretpostavkom da je $|dom(B)| \geq |dom(A)|$, može se zaključiti, na osnovu teoreme 3.9, da važi $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$. Uslov $|dom(B)| \geq |dom(A)|$ se zahteva zbog toga što za *fd* $B \to C$ važi $C \in X$ i $B \not\subseteq X$, gde je $X = AC$, pri čemu je ispunjen uslov teoreme 3.9. $X \to B \notin \Gamma^+$, gde je $\Gamma = \{A \to B, B \to C\}$. Integrativno svojstvo $(S, I) = Int((S_n, I_n), (S_m, I_m))$, međutim, ne važi. Primer relacija $r_1 \in SAT(AC, \{A \to C\})$, $r_2 \in SAT(AB, \{A \to B\})$ i $r_3 \in SAT(BC, \{B \to C\})$, koje obaraju integrativno svojstvo (preciznije, stav 1. definicije 4.3) su date na slici 4.1. Predikat $NonEqDk(Ext(R, \mathcal{K}))$ važi za $Ext(ABC, \{A \to B, B \to C, AB[B] \subseteq BC[B]\})$, pošto je za šeme relacija $(AB, \{A\}), (BC, \{B\}) \in Ext(R, \mathcal{K})$ i ključ $B \in \{B\} \setminus \{A\}$ zadovoljeno $B \cap AB = B \neq \emptyset$. S druge strane, ne postoji pojava $Ext_{ABC}(r_1, r_2, r_3)$, što je ilustrovano primerom na slici 4.1. □

r_1	A	C
	a_1	c_1
	a_2	c_2

r_2	A	B
	a_1	b_1
	a_2	b_1

r_3	B	C
	b_1	c_1

Slika 4.1.

Stav 1. definicije 4.3. zahteva dodatni komentar, vezan za "oslabljenje" uslova, po kojem bi, za svaku šemu relacije $(R, \mathcal{K}) \in S$, bilo zadovoljeno:

(4.11) $(\exists r \in SAT(R, \mathcal{K}))(r = Ext_R(r_1, ..., r_k) \in Ext(S))$,

uvođenjem implikacije $NonEqDk(Ext(R, \mathcal{K})) \Rightarrow$ (4.11). Situacija $\neg NonEqDk(Ext(R, \mathcal{K}))$:

(4.12) $(\forall(R_i, \mathcal{K}_i), (R_i, \mathcal{K}_i) \in Ext(R, \mathcal{K}))(\forall K_i \in \mathcal{K}_i \setminus \mathcal{K}_j)(K_i \cap R_j = \emptyset)$

može, takođe, dovesti do slučaja $(\forall r \in SAT(R, \mathcal{K}))(r \notin Ext(S))$, za neke pojave BP $(r_1^n,...,r_p^n)$ i $(r_1^m,...,r_q^m)$. Pomenuti slučaj se, praktično, ne događa, iako je situacija $\neg NonEqDk(Ext(R, \mathcal{K}))$, uobičajena i česta, dok je, s druge strane, uslov $\neg NonEqDk(Ext(R, \mathcal{K})) \wedge$ (4.11) suviše restriktivan sa stanovišta problema integrativnog svojstva dve šeme BP. Sledeći primer ilustruje ove zaključke.

Primer 4.4. Date su šeme BP (S_m, I_m), (S_n, I_n) i (S, I), takve da je:

- $S_m = \{(AB, \{A \to B\})\}$,
- $I_m = \emptyset$,
- $S_n = \{(ABC, \{A \to BC\})\}$,
- $I_n = \emptyset$,
- $S = \{(ABC, \{A \to BC\})\}$ i
- $I = \emptyset$.

Na osnovu teoreme 3.9. se zaključuje da važi $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$, dok se na osnovu definicije 4.3. može zaključiti da važi integrativno svojstvo $(S, I) = Int((S_n, I_n), (S_m, I_m))$, jer za sve šeme relacija $(R, \mathcal{K}) \in S$ važi $\neg NonEqDk(Ext(R, \mathcal{K}))$, a skup međurelacionih ograničenja $I = \emptyset$ je trivijalno zadovoljen, za bilo koje pojave BP nad (S_m, I_m) i (S_n, I_n).

Neka je $r_1 \in SAT(AB, \{A \to B\})$, $(AB, \{A \to B\}) \in S_m$, $r_2 \in SAT(ABC, \{A \to BC\})$, $(ABC, \{A \to BC\}) \in S_n$ i $r_3 \in SAT(ABC, \{A \to BC\})$, $(ABC, \{A \to BC\}) \in S$. Mogućnost nastanka situacije $\Pi_{AB}(r) \neq r \vee r \neq r_2$ se izbegava uvođenjem posebnog protokola o dozvoli ažuriranja relacija r_1 i r_2. Prema takvom protokolu, vrednost obeležja B se može zadavati (ili ažurirati) samo na nivou jedne od relacija r_1 i r_2. □

Definicija 4.4. Relacione šeme BP (S_n, I_n) i (S_m, I_m) su ***međusobno kompatibilne*** (***konzistentne***), u oznaci $(S_m, I_m) \approx (S_n, I_n)$, ako postoji šema BP (S, I), takva da važi:

1. $Int((S_n, I_n), (S_m, I_m))$,
2. $\Sigma(S) \models \Sigma(S_n) \cup \Sigma(S_m)$ i
3. $\Sigma(S) \cup I \models I_n \cup I_m$,

pri čemu je $\Sigma(S') = \bigcup_{(R_i, \Sigma) \in S'} \Sigma$, $S' \in \{S, S_n, S_m\}$, skup *gd* odgovarajuće šeme. Šema BP (S, I) se naziva ***pokrivajuća šema*** za (S_n, I_n) i (S_m, I_m). **[LM4]** □

Na osnovu uslova 1, definicije 4.4, zaključuje se da su međusobno kompatibilne šeme ***ekstenzionalno kompatibilne***, što znači da se pojave nad šemama (S_n, I_n) i (S_m, I_m) mogu proširiti do pojava nad pokrivajućom šemom (S, I).

Pošto uslov integrativnog svojstva $Int((S_n, I_n), (S_m, I_m))$ sadrži, kao potreban, uslov $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$, znači da ekstenzionalna kompatibilnost šema modula povlači za sobom i intenzionalnu kompatibilnost. Pri tome će se pod intenzionalno kompatibilnim šemama BP podrazumevati šeme (S_n, I_n) i (S_m, I_m), za koje, između ostalih, važe uslovi 2. i 3. definicije 4.4, a umesto uslova 1. definicije 4.4. se pojavljuje uslov $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$. Pojam intenzionalne kompatibilnosti će kasnije biti formalno uveden.

Lema 4.2. Za šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)), važi da je $(S_m, I_m) \approx (S_n, I_n)$, ukoliko postoji šema BP (S, I), takva da je $Int((S_n, I_n), (S_m, I_m))$ i $\Sigma(S) \cup I \supseteq I_n \cup I_m$, pri čemu je $\Sigma(S)$ skup svih lokalnih ograničenja (ne samo *fd*) šeme (S, I).

Dokaz. Na osnovu leme 4.1. sledi da je $\Gamma(S) \models \Gamma(S_n) \wedge \Gamma(S) \models \Gamma(S_m)$, čime je ispunjen uslov 2. definicije 4.4. Uslov 3. sledi direktno iz činjenice $\Sigma(S) \cup I \supseteq I_n \cup I_m$ i pretpostavke o postojanju šeme univerzalne relacije. □

Treba zapaziti da za međusobno kompatibilne šeme modula (S_m, I_m) i (S_n, I_n), pri čemu je $Int((S_n, I_n), (S_m, I_m))$ (odakle sledi da je $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$), važi:

(4.13) $$U_{BP}(S) \supseteq U_{BP}(S_m) \cup U_{BP}(S_n),$$

gde je sa $U_{BP}(S') = \bigcup_{(R_i, K_i) \in S'} R_i$, $S' \in \{S, S_n, S_m\}$, označen skup svih obeležja šeme S'.

Primer 4.5. Date su šeme BP (S_m, I_m), (S_n, I_n) i (S, I) iz primera 4.3. Pošto je narušeno integrativno svojstvo, (S, I) ne predstavlja pokrivajuću šemu za (S_m, I_m) i (S_n, I_n), što znači da ni uslov $(S_m, I_m) \approx (S_n, I_n)$ nije ispunjen. Treba zapaziti da, za razliku od (S_m, I_m) i (S_n, I_n), (S, I) ne zadovoljava uslov *3NF* i da zavisnost sadržavanja $AB[B] \subseteq BC[B]$, u okviru šeme relacije $(ABC, \{A \rightarrow B, B \rightarrow C, AB[B] \subseteq BC[B]\})$, predstavlja redundantno ograničenje. □

Primer 4.6. Date su šeme BP (S_m, I_m), (S_n, I_n) i (S, I), takve da je:

- $S_m = \{(AB, \{A \rightarrow B\})\}$,
- $I_m = \varnothing$,
- $S_n = \{(AC, \{A \rightarrow C\})\}$,
- $I_n = \varnothing$,
- $S = \{(ABCD, \{A \rightarrow B, A \rightarrow C, C \rightarrow D\})\}$ i
- $I = \varnothing$.

Važi da je $(S_m, I_m) \approx (S_n, I_n)$, odnosno da je (S, I) pokrivajuća šema za (S_m, I_m) i (S_n, I_n). Uslovi 1. i 2. definicije 4.3. i uslov 3. definicije 4.4. su trivijalno zadovoljeni, a uslov 2. definicije 4.4. važi jer je $\{A \rightarrow B\} \subseteq \{A \rightarrow B, A \rightarrow C, C \rightarrow D\}$ i $\{A \rightarrow C\} \subseteq \{A \rightarrow B, A \rightarrow C, C \rightarrow D\}$. □

Međusobno kompatibilne šeme modula se, teoretski, mogu integrisati u bilo koju pokrivajuću šemu BP. S projektantskog (i praktičnog) stanovišta, međutim, od važnosti je tzv. strogo pokrivajuća šema. Sledećom definicijom se uvode projektantski kriterijumi, koje pokrivajuća šema mora da zadovolji, kako bi predstavljala integrisanu šemu datih šema modula, za koje važi da su kompatibilne na ekstenzionalnom nivou. Neki od tih kriterijuma odgovaraju uslovima koje treba da zadovolji svaka šema modula BP, dobijena pomoću modifikovanog algoritma sinteze.

Definicija 4.5. Date su šeme modula (S_n, I_n) i (S_m, I_m), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27), (2.29) i (2.37)), takve da važi $(S_m, I_m) \approx (S_n, I_n)$. (S_n, I_n) i (S_m, I_m) se nazivaju ***međusobno strogo kompatibilnim*** (***konzistentnim***) šemama, u oznaci $(S_m, I_m) \cong (S_n, I_n)$, ukoliko postoji pokrivajuća šema BP (S, I), za koju važi:

1. $U_{BP}(S) = U_{BP}(S_m) \cup U_{BP}(S_n)$,
2. (S, I) zadovoljava uslov *3NF*,
3. $\Gamma(S) \equiv \Gamma(S_n) \cup \Gamma(S_m)$,

4. $(\forall(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j) \in S)(i \neq j \Rightarrow (R_i)^+_{\Gamma(S)} \neq (R_j)^+_{\Gamma(S)})$ i

5. $(\forall(R_i, \mathcal{K}_i) \in S)(\exists(R_j, \mathcal{K}_j) \in S_m \cup S_n)((R_i, \mathcal{K}_i) \supset (R_j, \mathcal{K}_j))$.

(S, I) se naziva ***strogo pokrivajuća šema*** za (S_m, I_m) i (S_n, I_n). □

Narednom definicijom se utvrđuju projektantski kriterijumi koje treba da zadovolje dve šeme modula i njihova pokrivajuća šema, kada je u pitanju intenzionalni nivo kompatibilnosti.

Definicija 4.6. Date su šeme modula (S_n, I_n) i (S_m, I_m), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27), (2.29) i (2.37)). (S_n, I_n) i (S_m, I_m) se nazivaju ***međusobno intenzionalno kompatibilnim*** (***konzistentnim***) šemama, u oznaci $(S_m, I_m) \cong_I (S_n, I_n)$, ukoliko postoji šema BP (S, I), za koju važi:

1. $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$,
2. $U_{BP}(S) = U_{BP}(S_m) \cup U_{BP}(S_n)$,
3. (S, I) zadovoljava uslov *3NF*,
4. $\Gamma(S) \equiv \Gamma(S_n) \cup \Gamma(S_m)$,
5. $\Gamma(S) \cup I \models I_m \cup I_n$,
6. $(\forall(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j) \in S)(i \neq j \Rightarrow (R_i)^+_{\Gamma(S)} \neq (R_j)^+_{\Gamma(S)})$ i
7. $(\forall(R_i, \mathcal{K}_i) \in S)(\exists(R_j, \mathcal{K}_j) \in S_m \cup S_n)((R_i, \mathcal{K}_i) \supset (R_j, \mathcal{K}_j))$.

(S, I) se naziva ***intenzionalno pokrivajuća šema*** za (S_m, I_m) i (S_n, I_n). □

Za razliku od definicije 4.5, definicija 4.6. ne zahteva da šeme modula (S_m, I_m) i (S_n, I_n) zadovoljavaju integrativno svojstvo, odnosno stavove 1. i 2. definicije 4.3.

Može se jednostavno dokazati da modifikovani algoritam sinteze, prikazan u tački 2.2.2.3, obezbeđuje da šema modula (S_n, I_n) zadovoljava i sledeće uslove:

(4.14) (S_n, I_n) zadovoljava *3NF*,

(4.15) $\Gamma(S_n) \equiv \Gamma_n$,

pri čemu je Γ_n skup *fd* modula, dat izrazom (2.17) i

(4.16) $(\forall(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j) \in S_n)(i \neq j \Rightarrow (R_i)^+_{\Gamma(S_n)} \neq (R_j)^+_{\Gamma(S_n)})$.

S obzirom da izrazi (4.14) i (4.16) odgovaraju, redom, stavovima 2. i 4, definicije 4.5, može se zaključiti da skup šema relacija strogo pokrivajuće šeme ispunjava projektantske uslove, implicirane modifikovanim algoritmom sinteze.

Primer 4.7. Šema BP (S, I) iz primera 4.6. ne predstavlja strogo pokrivajuću šemu za (S_m, I_m) i (S_n, I_n), iz istog primera, jer su narušeni stavovi 1, 2. i 3. definicije 4.5. (S, I) ne predstavlja, takođe, ni intenzionalno pokrivajuću šemu BP za (S_m, I_m) i (S_n, I_n). □

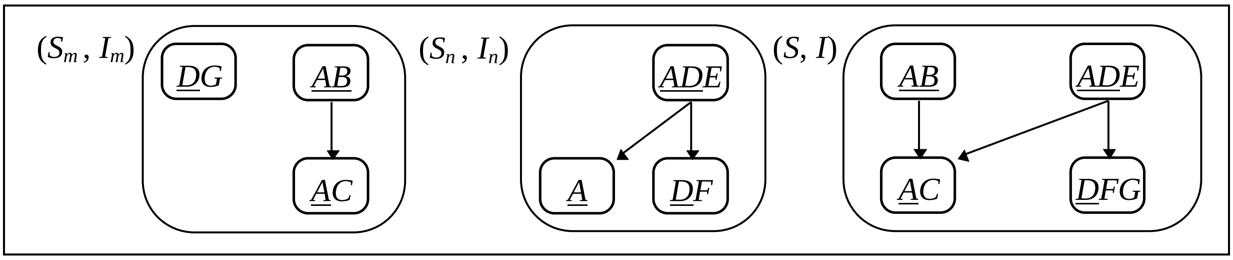
Primer 4.8. Date su šeme BP (S_m, I_m), (S_n, I_n) i (S, I), takve da je:

- $S_m = \{(AB, \emptyset), (AC, \{A \rightarrow C\}), (DG, \{D \rightarrow G\})\}$,
- $I_m = \{AB[A] \subseteq AC[A]\}$,

- $S_n = \{(A, \emptyset), (ADE, \{AD \rightarrow E\}), (DF, \{D \rightarrow F\})\}$,
- $I_n = \{ADE[D] \subseteq DF[D], ADE[A] \subseteq A[A]\}$,
- $S = \{(AB, \emptyset), (AC, \{A \rightarrow C\}), (ADE, \{AD \rightarrow E\}), (DFG, \{D \rightarrow FG\})\}$ i
- $I = \{ AB[A] \subseteq AC[A], ADE[D] \subseteq DFG[D], ADE[A] \subseteq AC[A]\}$.

Na slici 4.2. su prikazani grafovi zatvaranja datih šema BP. Ključevi šema relacija su podvučeni.

(S, I) predstavlja strogo pokrivajuću šemu za (S_m, I_m) i (S_n, I_n), što znači da je ispunjen uslov $(S_m, I_m) \cong (S_n, I_n)$. □



Slika 4.2.

Lema 4.3. Neka je data strogo pokrivajuća šema (S, I) za šeme modula (S_m, I_m) i (S_n, I_n). Važi da je $I \equiv I_n \cup I_m$.

Dokaz. $(\Rightarrow)$ Implikacija $I \models I_n \cup I_m$ sledi iz stava 3. definicije 4.4. $(\Sigma(S) \cup I \models I_n \cup I_m)$ i činjenica da (S, I) zadovoljava uslov *3NF* i da je $\Gamma(S) \equiv \Gamma(S_n) \cup \Gamma(S_m)$, što znači da se ni jedno međurelaciono ograničenje skupa $I_n \cup I_m$ ne može transformisati u ograničenje skupa $\Gamma(S)$.

$(\Leftarrow)$ Treba dokazati da je $I_n \cup I_m \models I$. Uvodi se suprotna pretpostavka $I_n \cup I_m \not\models I$. Neka su $(r_1^n,..., r_p^n)$ pojava BP nad šemom (S_n, I_n) i $(r_1^m,..., r_q^m)$ pojava nad (S_m, I_m), takve da postoje proširene pojave $\{r_1,...,r_k\} \subseteq Ext(S)$, koje narušavaju neko ograničenje $ic \in I$. Saglasno pretpostavci teoreme, važi $(S, I) = Int((S_m, I_m), (S_n, I_n))$, što znači da (prema stavu 2. definicije 4.3) $Ext(S)$ mora biti pojava za (S, I). To je kontradiktorno zaključku da $Ext(S)$ narušava $ic \in I$, što znači da važi $I_n \cup I_m \models I$. □

Treba napomenuti da je drugi deo dokaza leme 4.3. sproveden zahvaljujući uslovu ekstenzionalne kompatibilnosti šema BP, koji u paksi može biti vrlo strog, o čemu će kasnije u ovom poglavlju biti više reči.

Iz leme 4.3. i definicija 4.5. i 4.6. sledi zaključak da su strogo kompatibilne šeme modula istovremeno i intenzionalno kompatibilne:

$$(S_m, I_m) \cong (S_n, I_n) \Rightarrow (S_m, I_m) \cong_I (S_n, I_n). \tag{4.17}$$

Obrnuto tvrđenje, međutim, ne važi, pošto za intenzionalnu kompatibilnost šema ne važi uslov $I \equiv I_n \cup I_m$, već samo važi $\Gamma(S) \cup I \models I_m \cup I_n$.

Lema 4.4. Ukoliko za šeme modula (S_m, I_m) i (S_n, I_n) postoji strogo pokrivajuća šema (S, I), onda je ona jedinstvena.

Dokaz. Ako važi $(S_m, I_m) \cong (S_n, I_n)$, tada jedinstvenost strogo pokrivajuće šeme sledi direktno iz definicije 4.5. i leme 4.3. □

Ciljevi ovog poglavlja rada su da se za dve proizvoljne šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze:

- utvrde potrebni i dovoljni uslovi, kojima će biti obezbeđeno važenje kriterijuma $(S_m, I_m) \cong (S_n, I_n)$, odnosno $(S_m, I_m) \cong_I (S_n, I_n)$,
- formalizuje postupak testova $(S_m, I_m) \cong (S_n, I_n)$ i $(S_m, I_m) \cong_I (S_n, I_n)$, saglasno prethodno utvrđenim dovoljnim uslovima za $(S_m, I_m) \cong (S_n, I_n)$ i $(S_m, I_m) \cong_I (S_n, I_n)$ i
- utvrdi postupak integracije šema modula, odnosno generisanja strogo pokrivajuće šeme za (S_n, I_n) i (S_m, I_m).

4.3. Uslovi međusobne stroge kompatibilnosti

Primer 4.3. predstavlja jednu moguću ilustraciju za šeme modula koje nisu međusobno strogo kompatibilne. Pored toga što je narušeno integrativno svojstvo, razlog ove nekompatibilnosti leži i u činjenici da *fd* $A \rightarrow C$ postaje tranzitivna s obzirom na skup *fd* $\Gamma(S_n) \cup \Gamma(S_m)$, iako to svojstvo nije imala s obzirom na skup $\Gamma(S_m)$. Sledeći primeri ilustruju ostale karakteristične situacije strogo nekompatibilnih šema modula.

Primer 4.9. Date su šeme modula BP (S_m, I_m) i (S_n, I_n), takve da je:

- $S_m = \{(ABC, \{AB \rightarrow C\})\}$,
- $I_m = \emptyset$,
- $S_n = \{(AD, \{A \rightarrow D\}), (BD, \{D \rightarrow B\})\}$ i
- $I_n = \{AD[D] \subseteq BD[D]\}$.

Ne postoji strogo pokrivajuća šema za (S_m, I_m) i (S_n, I_n). Bilo koja pokrivajuća šema (S', I') ne može zadovoljiti uslov *3NF*, jer u S' mora postojati šema relacije koja predstavlja proširenje za $(ABC, \{AB \rightarrow C\})$, a s obzirom na uslov $\Sigma(S') \models \Sigma(S_n) \cup \Sigma(S_m)$, zavisnost $AB \rightarrow C$ postaje nepotpuna. Razlog ove nekompatibilnosti, dakle, leži u činjenici da *fd* $AB \rightarrow C$ postaje nepotpuna s obzirom na skup *fd* $\Gamma(S_n) \cup \Gamma(S_m)$, iako to svojstvo nije imala s obzirom na skup $\Gamma(S_m)$. □

Primer 4.10. Date su šeme modula BP (S_m, I_m) i (S_n, I_n), takve da je:

- $S_m = \{(AB, \{A \rightarrow B\})\}$,
- $I_m = \emptyset$,
- $S_n = \{(BC, \{B \rightarrow C\}), (AC, \{C \rightarrow A\})\}$ i
- $I_n = \{BC[C] \subseteq AC[C]\}$.

Ne postoji strogo pokrivajuća šema za (S_m, I_m) i (S_n, I_n). Jedina šema BP koja zadovoljava stavove 1, 2, 3. i 4, definicije 4.5, je (S', I'), pri čemu je $S' = \{(ABC, \{A \rightarrow B, B \rightarrow C, C \rightarrow A\})\}$ i $I = \emptyset$. Šema relacije $(ABC, \{A \rightarrow B, B \rightarrow C, C \rightarrow A\})$, međutim, ne predstavlja proširenje ni jedne šeme relacije iz skupa $S_m \cup S_n$, jer je narušen uslov $\Delta^+ = \Pi_X(\Gamma)$, stava 2, teoreme 3.10, pošto za *fd* $B \rightarrow A \in \Pi_{AB}(\{A \rightarrow B, B \rightarrow C, C \rightarrow A\})$ važi da $\Gamma(S_m) = \{A \rightarrow B\} \not\models B \rightarrow A$. □

Primer 4.11. Date su šeme modula BP (S_m, I_m) i (S_n, I_n):

- $S_m = \{(ABC, \{A \rightarrow B, B \rightarrow A, B \rightarrow C\})\}$,

- $I_m = \emptyset$,
- $S_n = \{(BCD, \{B \to C, B \to D, CD \to B\})\}$ i
- $I_n = \emptyset$.

Za šeme modula (S_m, I_m) i (S_n, I_n) bi se moglo, na osnovu teoreme 3.10, zaključiti da postoji šema (S, I), $S = \{(ABCD, \{A, B, CD\})\}$, $I = \emptyset$ (lokalni skup ograničenja je izražen putem zavisnosti ključa), takva da je $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$, pod uslovom da je $|dom(D)| \geq |dom(B)|$. (S, I) zadovoljava stavove 1 - 5. definicije 4.5, ali ne predstavlja pokrivajuću šemu za (S_m, I_m) i (S_n, I_n), jer je narušeno integrativno svojstvo $(S, I) = Int((S_n, I_n), (S_m, I_m))$ (stav 1. definicije 4.3), što je ilustrovano primerom, prikazanom na slici 4.3, za $r_1 \in SAT(ABC, \{A, B\})$ i $r_2 \in SAT(BCD, \{B, CD\})$. Stav 1. definicij 4.3. je narušen, jer za šeme relacija $(R_m, \mathcal{K}_m) = (ABC, \{A, B\})$ i $(R_n, \mathcal{K}_n) = (BCD, \{B, CD\})$ važi da je $CD \cap ABC = C \neq \emptyset$, pri čemu je $CD \in \mathcal{K}_n \setminus \mathcal{K}_m$. Treba primetiti da su (S_m, I_m) i (S_n, I_n) intenzionalno kompatibilne šeme, ali zbog narušavanja stava 1. definicije 4.3. (integrativno svojstvo), nisu ekstenzionalno kompatibilne. □

r_1	A	B	C
	a_1	b_1	c_1
	a_2	b_2	c_2

r_2	B	C	D
	b_1	c_1	d_1
	b_2	c_1	d_2

Slika 4.3.

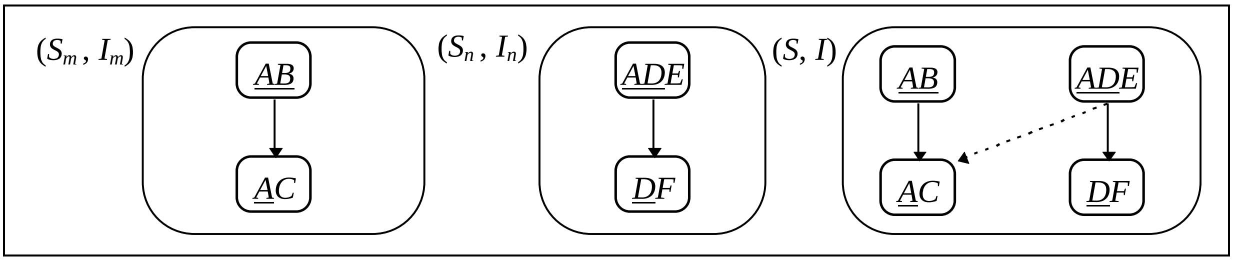
Primer 4.12. Date su šeme BP (S_m, I_m), (S_n, I_n) i (S, I), takve da je:

- $S_m = \{(AB, \emptyset), (AC, \{A \to C\})\}$,
- $I_m = \{AB[A] \subseteq AC[A]\}$,
- $S_n = \{(ADE, \{AD \to E\}), (DF, \{D \to F\})\}$,
- $I_n = \{ADE[D] \subseteq DF[D]\}$,
- $S = \{(AB, \emptyset), (AC, \{A \to C\}), (ADE, \{AD \to E\}), (DF, \{D \to F\})\}$ i
- $I = \{ AB[A] \subseteq AC[A], ADE[D] \subseteq DF[D], ADE[A] \subseteq AC[A]\}$.

Na slici 4.4. su prikazani grafovi zatvaranja datih šema relacija. Ključevi šema relacija su podvučeni.

(S, I) ne predstavlja strogo pokrivajuću šemu za (S_m, I_m) i (S_n, I_n), jer je narušena tvrdnja leme 4.3. Treba zapaziti da se referencijalni integritet $ADE[A] \subseteq AC[A] \in I$ ne može pojaviti u skupu I_m, ili I_n, što znači da je $I \not\equiv I_n \cup I_m$. Može se, pri tome, zapaziti da je narušeno integrativno svojstvo (stav 2. definicije 4.3), pošto lokalno ažurirana relacija nad šemom $(ADE, \{AD \to E\})$ može sadržati torke, koje narušavaju pravilo $ADE[A] \subseteq AC[A]$.

S druge strane, može se utvrditi da važe uslovi 1-7. definicije 4.6, što znači da su (S_m, I_m) i (S_n, I_n) intenzionalno kompatibilne šeme. □



Slika 4.4.

Pre nego se formulišu potrebni i dovoljni uslovi da šeme modula budu međusobno strogo kompatibilne, neophodno je opisati tipove međusobnih kolizija šema modula. Sledeće definicije uvode, redom, pojmove kolizije tranzitivnosti, kolizije funkcionalni-nefunkcionalni odnos, kolizije ekvivalentnih ključeva i kolizije integrativnog svojstva.

U nastavku teksta, sa Γ_{mn} će biti označen skup svih *fd* koje su posledica ključeva svih šema relacija iz šema modula (S_m, I_m) i (S_n, I_n): $\Gamma_{mn} = \Gamma(S_n) \cup \Gamma(S_m)$.

Definicija 4.7. Šeme modula (S_m, I_m) i (S_n, I_n) su ***tranzitivno kolizione***, u oznaci $TranC(S_m, S_n)$, ako važi:

(4.18) $$(\exists(R_i, \mathcal{K}_i) \in S_m)(\exists(R_j, \mathcal{K}_j) \in S_n)(\mathcal{K}_i \cap \mathcal{K}_j = \emptyset \wedge Hte(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})),$$

pri čemu se predikat $Hte(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})$ interpretira putem formule:

$$(R_i \setminus K_p(R_i)) \cap (R_j \setminus K_p(R_j)) \neq \emptyset \wedge (R_i \subseteq (R_j)^+_{\Gamma_{mn}} \veebar R_j \subseteq (R_i)^+_{\Gamma_{mn}}),$$

gde $K_p(R)$ označava skup primarnih obeležja odgovarajuće šeme relacije, a $\veebar$ označava ekskluzivnu disjunkciju formula. □ **[LM4]**

Primer 4.13. Šeme modula BP (S_m, I_m) i (S_n, I_n) iz primera 4.3. su tranzitivno kolizione. Šeme relacija, zbog kojih važi $TranC(S_m, S_n)$ su $(AC, \{A\})$ i $(BC, \{B\})$, jer važi $B \subseteq (A)^+_{\Gamma_{mn}}$.

□

Definicija 4.8. Šeme modula (S_m, I_m) i (S_n, I_n) su u ***koliziji funkcionalni-nefunkcionalni odnos*** (***fnf-koliziji***), u oznaci $FnfC(S_m, S_n)$, ako važi:

(4.19) $$(\exists(R_i, \mathcal{K}_i) \in S_m \cup S_n)(Fnf(\mathcal{K}_i, \Gamma_{mn})),$$

pri čemu se predikat $Fnf(\mathcal{K}_i, \Gamma_{mn})$ interpretira putem formule:

$$(\exists K_i \in \mathcal{K}_i)(\exists X \subset K_i)(X \neq \emptyset \wedge X \subseteq (K_i \setminus X)^+_{\Gamma_{mn}}).$$ □ **[LM4]**

Primer 4.14. Šeme modula BP (S_m, I_m) i (S_n, I_n) iz primera 4.9. su u *fnf*-koliziji. Šema relacije, zbog koje važi $FnfC(S_m, S_n)$ je $(ABC, \{AB\})$, jer važi $B \subseteq (A)^+_{\Gamma_{mn}}$. □

Definicija 4.9. Šeme modula (S_m, I_m) i (S_n, I_n) su u ***koliziji ekvivalentnih ključeva***, u oznaci $EqKeyC(S_m, S_n)$, ako važi:

(4.20) $$(\exists(R_i, \mathcal{K}_i), \in S_m)(\exists(R_j, \mathcal{K}_j) \in S_n)(\mathcal{K}_i \cap \mathcal{K}_j = \emptyset \wedge Eqk(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})),$$

pri čemu se predikat $Eqk(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})$ interpretira putem formule:

$$(R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}} \wedge ((\exists K_j \in \mathcal{K}_j)(K_j \subseteq (R_i)^+_{\Gamma(S_m)}) \vee (\exists K_i \in \mathcal{K}_i)(K_i \subseteq (R_j)^+_{\Gamma(S_n)})).$$ □

Primer 4.15. Šeme modula BP (S_m, I_m) i (S_n, I_n) iz primera 4.10. su u koliziji ekvivalentnih ključeva. Šeme relacija sa skupovima ključeva $\mathcal{K}_i \cap \mathcal{K}_j = \varnothing$, zbog kojih je $EqKeyC(S_m, S_n)$, su npr. $(AB, \{A \to B\})$ i $(AC, \{C \to A\})$, jer važi $(A)^+_{\Gamma_{mn}} = (C)^+_{\Gamma_{mn}} \wedge A \subseteq (AC)^+_{\{C \to A, B \to C\}}$. □

Primećuje se da kolizija $EqKeyC(S_m, S_n)$ zahteva da postoje šeme relacija $(R_i, \mathcal{K}_i)$ i $(R_j, \mathcal{K}_j)$ sa svojstvom $\mathcal{K}_i \cap \mathcal{K}_j = \varnothing \wedge Eqk(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})$, koje su isključivo iz različitih modula. Naredni primer ilustruje situaciju koja ne izaziva koliziju ekvivalentnih ključeva, jer ne postoje šeme relacija iz različitih modula sa svojstvom $\mathcal{K}_i \cap \mathcal{K}_j = \varnothing \wedge Eqk(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})$.

Primer 4.16. Date su šeme modula BP (S_m, I_m) i (S_n, I_n):

- $S_m = \{(AB, \{A, B\})\}$,
- $I_m = \varnothing$,
- $S_n = ((A, \{A\}), (B, \{B\}))$ i
- $I_n = \varnothing$.

(S_m, I_m) i (S_n, I_n) nisu u $EqKeyC(S_m, S_n)$ koliziji. Strogo pokrivajuća šema za (S_m, I_m) i (S_n, I_n) je šema (S_m, I_m). □

Primer 4.17. Šeme modula BP (S_m, I_m) i (S_n, I_n):

- $S_m = \{(AB, \{A, B\}), (CD, \{C, D\})\}$,
- $I_m = \varnothing$,
- $S_n = ((BC, \{B, C\}), (D, \{D\}))$ i
- $I_n = \varnothing$,

nisu u $EqKeyC(S_m, S_n)$ koliziji, jer za bilo koju kombinaciju šema relacija $(R_i, \mathcal{K}_i)$ i $(R_j, \mathcal{K}_j)$, za koju važi $\mathcal{K}_i \cap \mathcal{K}_j = \varnothing \wedge (R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}}$, može se zaključiti da ne važi drugi deo kriterijuma $Eqk(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})$. Strogo pokrivajuća šema za (S_m, I_m) i (S_n, I_n) je šema (S, I), $S = (ABCD, \{A, B, C, D\})$, $I = \varnothing$. □

Definicija 4.10. Šeme modula (S_m, I_m) i (S_n, I_n) su u ***koliziji integrativnog svojstva***, u oznaci $IntC(S_m, S_n)$, ako važi:

(4.21) $$IntCDk(S_m, S_n) \vee IntCLi(S_m, S_n) \vee IntCGr(S_m, S_n).$$

U nastavku se daju interpretacije predikata $IntCDk(S_m, S_n)$, $IntCLi(S_m, S_n)$ i $IntCGr(S_m, S_n)$.

- Predikat $IntCDk(S_m, S_n)$ (***kolizija dela ključa***) se interpretira putem formule:

  (4.22) $$(\exists(R_i, \mathcal{K}_i) \in S_m)(\exists(R_j, \mathcal{K}_j) \in S_n)((R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}} \wedge DInt(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma(S_m), \Gamma(S_n))),$$

  gde je $\Gamma_{mn} = \Gamma(S_n) \cup \Gamma(S_m)$, dok se predikat $DInt(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma(S_m), \Gamma(S_n))$ definiše putem formule:

$$(\exists K_i \in \mathcal{K}_i \setminus \mathcal{K}_j)(K_i \cap (R_j)^+_{\Gamma_{mn} \setminus \Gamma(R_i)} \neq \varnothing) \vee (\exists K_j \in \mathcal{K}_j \setminus \mathcal{K}_i)(K_j \cap (R_i)^+_{\Gamma_{mn} \setminus \Gamma(R_j)} \neq \varnothing).$$

- Predikat $IntCLi(S_m, S_n)$ (***lokalna kolizija skupova međurelacionih ograničenja*** I_m i I_n) se interpretira putem formule:

(4.23) $(\exists(R_{m_1}, \mathcal{K}_{m_1}),...,(R_{m_k}, \mathcal{K}_{m_k}) \in S_m)(\exists(R_{n_1}, \mathcal{K}_{n_1}),...,(R_{n_k}, \mathcal{K}_{n_k}) \in S_n)($

$$(\forall i \in \{1,...,k\})(\mathcal{K}_{m_i} \cap \mathcal{K}_{n_i} \neq \varnothing) \wedge Lint(\{R_{m_1},..., R_{m_k}\}, I_m, \{R_{n_1},..., R_{n_k}\}, I_n)),$$

a značenje predikata $Lint(Si_m, I_m, Si_n, I_n)$ je dato formulom:

$$(\exists ic \in IC_n)(obuhvata(ic, Si_n)) \wedge \neg(\exists ic \in IC_m)(obuhvata(ic, Si_m)) \vee$$
$$\neg(\exists ic \in IC_n)(obuhvata(ic, Si_n)) \wedge (\exists ic \in IC_m)(obuhvata(ic, Si_m)),$$

gde $IC_n \subseteq I_n$, označava, redom, bilo koji skup ograničenja $ID(\mathcal{G}_n, \mathcal{SF}u_n)$, $ID(\mathcal{G}_n, desig)$, ili $DR(\mathcal{G}_n, dep)$, saglasno izrazu (2.37), a analogno važi i za $IC_m \subseteq I_m$. Predikat *obuhvata*(*ic*, *Si*) znači da međurelaciono ograničenje *ic* obuhvata šeme relacija, zadate skupom *Si*.

- Predikat $IntCGr(S_m, S_n)$ (***globalna kolizija skupova međurelacionih ograničenja*** I_m i I_n) se interpretira putem formule:

(4.24) $(\exists(R_i, \mathcal{K}_i) \in S_m)(\exists(R_j, \mathcal{K}_j) \in S_n)(\mathcal{K}_i \cap \mathcal{K}_j = \varnothing \wedge (RICls(R_j, R_i, \Gamma_{mn}, I_{mn}) \vee$

$$RICls(R_i, R_j, \Gamma_{mn}, I_{mn}) \vee RIRen(R_i, R_j, I_{mn}) \vee BRCl(R_i, R_j, I_{mn}) \vee BRCl(R_j, R_i, I_{mn}) \vee$$
$$BrDer(R_i, R_j, I_{mn}))),$$

gde je $I_{mn} = I_m \cup I_n$. Predikat $RICls(R_x, R_y, \Gamma_{mn}, I_{mn})$ ("referencijalni integritet zatvaranja") se interpretira kao:

$$(R_x \subset (R_y)^+_{\Gamma_{mn}} \wedge R_y[K_x] \subseteq R_x[K_x] \in RI \wedge RI(\mathcal{G}_m) \cup RI(\mathcal{G}_n) \not\models R_y[K_x] \subseteq R_x[K_x]),$$

pri čemu je *RI* skup svih referencijalnih integriteta. Predikat $RIRen(R_i, R_j, I_{mn})$ ("referencijalni integritet preimenovanja") se interpretira kao:

$$(\exists ri_r \in RI_r(ren, dbs))(obuhvata(ri_r, \{R_i, R_j\}) \wedge I_m \cup I_n \not\models ri_r),$$

gde $RI_r(ren, dbs)$ predstavlja opšti skup referencijalnih integriteta preimenovanja (dat izrazom (2.36)), pri čemu su funkcije *ren* i *dbs* nezavisne od izgenerisanih šema modula. Predikat $BRCl(R_x, R_y, I_{mn})$ ("pravilo poslovanja zatvaranja") se interpretira kao:

$$(\exists ic \in BR)(R_x \subset (R_y)^+_{\Gamma_{mn}} \wedge obuhvata(ic, \{R_x, R_y\}) \wedge BR(\mathcal{G}_n) \cup BR(\mathcal{G}_m) \not\models ic),$$

gde je *BR* skup svih pravila poslovanja oblika $\triangleright\triangleleft(R_x, R_{x_1},..., R_{x_m})[K_y] \subseteq R_y[K_y]$. Predikat $BrDer(R_i, R_j, I_{mn})$ ("pravilo poslovanja funkcije izvođenja") se interpretira kao:

$$(\exists br_d \in BR(der, dbs))(obuhvata(br_d, \{R_i, R_j\}) \wedge I_m \cup I_n \not\models br_d),$$

gde $BR(der, dbs)$ predstavlja opšti skup pravila poslovanja, koja su posledica funkcije izovđenja uloge obeležja *der*, date definicijom 2.56 . □

Primer 4.18. Date su šeme modula BP (S_m, I_m) i (S_n, I_n) iz primera 4.11. Skupovi ključeva odgovarajućih šema relacija su: $\mathcal{K}_m = \{A, B\}$ i $\mathcal{K}_n = \{B, CD\}$. Za prikazane šeme modula važi $IntC(S_m, S_n)$, tj. $IntCDk(S_m, S_n)$, jer je $\mathcal{K}_m \cap \mathcal{K}_n \neq \varnothing$, pri čemu važi $CD \in \{B, CD\} \setminus \{A, B\} \wedge CD \cap (ABC)^+_{\{A\to B, B\to A, B\to C\}} = C \neq \varnothing$. □

Primer 4.19. Šeme modula BP (S_m, I_m) i (S_n, I_n) iz primera 4.10. su i u koliziji integrativnog svojstva, jer važi $IntCDk(S_m, S_n)$. Može se proveriti da za šeme relacija $(BC, \{B\to C\})$ i $(AC, \{C\to A\})$ važi $(B)^+_{\Gamma_{mn}} = (C)^+_{\Gamma_{mn}} \wedge B \notin \{C\} \setminus \{B\} \wedge B \cap (AC)^+_{\{C\to A, A\to B\}} \neq \varnothing$. □

Primer 4.20. Za šeme modula BP (S_m, I_m) i (S_n, I_n) iz primera 4.12. važi kolizija $IntC(S_m, S_n)$, specijalno $IntCGr(S_m, S_n)$. Za referencijalni integritet $ADE[A] \subseteq AC[A]$ se može zaključiti sledeće: $RI(\mathcal{G}_m) \cup RI(\mathcal{G}_n) \not\models ADE[A] \subseteq AC[A]$. □

Naredni primeri ilustruju još neke situacije šema modula koje su u koliziji integrativnog svojstva.

Primer 4.21. Date su šeme modula BP (S_m, I_m) i (S_n, I_n):

- $S_m = \{(ABE, \{A \to B, B \to A, B \to E\}), (CE, \{E \to C\})\}$,
- $I_m = \{ABE[E] \subseteq CE[E]\}$,
- $S_n = \{(BCD, \{B \to C, B \to D, CD \to B\})\}$ i
- $I_n = \emptyset$.

Za prikazane šeme modula važi $IntC(S_m, S_n)$, tj. $IntCDk(S_m, S_n)$, pošto za šeme relacija $(ABE, \{A, B\})$ i $(BCD, \{B, CD\})$ važi $CD \in \{B, CD\} \setminus \{A, B\} \wedge CD \cap (ABE)^+_{\Gamma_{mn} \setminus \{B \to CD, CD \to B\}} = C \neq \emptyset$. Posmatra se šema (S, I):

$$S = \{(ABCDE, \{A, B, CD\}), (CE, \{E\})\},\ I = \{ABCDE[CE] \subseteq CE[CE]\}.$$

(S, I) zadovoljava stavove 1 - 5. definicije 4.5, ali ne predstavlja pokrivajuću šemu za (S_m, I_m) i (S_n, I_n), jer je narušeno integrativno svojstvo $(S, I) = Int((S_n, I_n), (S_m, I_m))$ (stav 2. definicije 4.3), odnosno tvrđenje leme 4.3. ($I \not\equiv I_n \cup I_m$). Ovo je ilustrovano primerom, prikazanom na slici 4.5, za relacije $r_1 \in SAT(ABE, \{A, B\})$, $r_2 \in SAT(BCD, \{B, CD\})$, $r_3 \in SAT(CE, \{E\})$ i $r \in SAT(ABCDE, \{A, B, CD\})$. Narušeno je međurelaciono ograničenje $ABCDE[CE] \subseteq CE[CE] \in DR(\mathcal{G}, dep)$, koje je posledica činjenice da je šema (S, I) zavisna (prema **[Sc]**, u pitanju je B-zavisnost). □

r_1	A	B	E
	a_1	b_1	e_1
	a_2	b_2	e_2

r_2	B	C	D
	b_1	c_1	d_1
	b_2	c_1	d_2

r_3	E	C
	e_1	c_1
	e_2	c_2

r	A	B	C	D	E
	a_1	b_1	c_1	d_1	e_1
	a_2	b_2	c_1	d_2	e_2

Slika 4.5.

Primer 4.22. Date su šeme modula BP (S_m, I_m) i (S_n, I_n):

- $S_m = \{(ABD, \{AB \to D\}), (AC, \{A \to C\})\}$,
- $I_m = \{ABD[A] \subseteq AC[A], AC[A] \subseteq ABD[A]\}$,
- $S_n = \{(ABD, \{AB \to D\}), (AC, \{A \to C\})\}$ i
- $I_n = \{ABD[A] \subseteq AC[A]\}$.

Za prikazane šeme modula važi $IntC(S_m, S_n)$, tj. $IntcLi(S_m, S_n)$. Posmatra se skup *id* $ID(\mathcal{G}_m, \mathcal{SF}u_m) \subseteq I_m$, formiran prema izrazu (2.34): $ID(\mathcal{G}_m, \mathcal{SF}u_m) = \{AC[A] \subseteq ABD[A]\}$. Pri tome je, dakle, $IC_m = ID(\mathcal{G}_m, \mathcal{SF}u_m) = \{AC[A] \subseteq ABD[A]\} \subseteq I_m$. S druge strane, skup međurelacionih ograničenja I_n neće sadržati $id\ AC[A] \subseteq ABD[A]$, pod pretpostavkom da je $ID(\mathcal{G}_n, \mathcal{SFu}_n) = \varnothing$, što, dalje, znači da je $IC_n = ID(\mathcal{G}_n, \mathcal{SFu}_n) = \varnothing$. Kritično međurelaciono ograničenje je $ic = AC[A] \subseteq ABD[A]$, jer za $Si_m = \{ABD, AC\}$ važi $obuhvata(ic, Si_m)$, dok za $Si_n = \{ABD, AC\}$ predikat $obuhvata(ic, Si_n)$ ne važi. (S_m, I_m) i (S_n, I_n) nisu strogo kompatibilne jer se može narušiti integrativno svojstvo (stav 2. definicije 4.3), tako što će se u pojavi šeme modula (S_n, I_n) narušiti $AC[A] \subseteq ABD[A] \notin I_n$. □

Primer 4.23. Date su šeme modula BP (S_m, I_m) i (S_n, I_n):

- $S_m = \{(ADE, \{AD \rightarrow E\})\}$,
- $I_m = \varnothing$,
- $S_n = \{(ABC, \{AB \rightarrow C\}), (CD, \{C \rightarrow D\})\}$ i
- $I_n = \{ABC[C] \subseteq CD[C]\}$.

Za prikazane šeme modula važi $IntC(S_m, S_n)$, tj. $IntCGr(S_m, S_n)$. Razlog za koliziju leži u činjenici da se za pravilo poslovanja $\rhd\lhd(ABC, CD)[AD] \subseteq ADE[AD]$, za koje je ispunjeno $\Gamma_{mn} \models \rhd\lhd(ABC, CD)[AD] \subseteq ADE[AD]$, može zaključiti:

$$BR(\mathcal{G}_m) \cup BR(\mathcal{G}_n) \not\models \rhd\lhd(ABC, CD)[AD] \subseteq ADE[AD],$$

čime se narušava stav 2. definicije 4.3. □

4.4. Potreban i dovoljan uslov međusobne stroge kompatibilnosti

Sledeće leme stvaraju osnove za definisanje potrebnog i dovoljnog uslova međusobne stroge kompatibilnosti šema modula. U okviru dokaza lema će, implicitno, biti uveden i postupak formiranja strogo pokrivajuće šeme, odnosno postupak integracije šema modula u jednstvenu šemu BP.

Lema 4.5. Date su strogo kompatibilne šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)). (S_m, I_m) i (S_n, I_n) nisu u koliziji tranzitivnosti.

Dokaz. Treba dokazati da važi implikacija $(S_m, I_m) \cong (S_n, I_n) \Rightarrow \neg TranC(S_m, S_n)$. Uvodi se suprotna pretpostavka, $TranC(S_m, S_n)$, pri čemu je za šeme relacija $(R_i, \mathcal{K}_i) \in S_m$ i $(R_j, \mathcal{K}_j) \in S_n$ ispunjen uslov $\mathcal{K}_i \cap \mathcal{K}_j = \varnothing \wedge Hte(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})$. Neka je, pri tome, $K_i \subseteq (K_j)^+_{\Gamma_{mn}}$, gde je $K_i \in \mathcal{K}_i$. (U slučaju da je $K_j \subseteq (K_i)^+_{\Gamma_{mn}}$, za $K_j \in \mathcal{K}_j$, dokaz je analogan.) Neka je (S, I) strogo pokrivajuća šema i neka je $(R, \mathcal{K}) \in S$ šema relacije, za koju važi $(R, \mathcal{K}) \supset (R_j, \mathcal{K}_j)$. Sledi da je $R_j \subseteq R$. S druge strane, iz $K_i \subseteq (K_j)^+_{\Gamma_{mn}}$ i $K_j \not\subseteq (K_i)^+_{\Gamma_{mn}}$ sledi da je netrivijalna $fd\ K_j \rightarrow (R_i \setminus K_p(R_i)) \cap (R_j \setminus K_p(R_j)) \in \Gamma^+_{mn}$ tranzitivna. Zbog $R_j \subseteq R$ i stava 3. definicije 4.5, važi da je $K_j \rightarrow (R_i \setminus K_p(R_i)) \cap (R_j \setminus K_p(R_j)) \in (\Gamma(R))^+$, što znači da $(R, \mathcal{K})$ ne zadovoljava *3NF*. To je u kontradikciji s pretpostavkom da je $(S_m, I_m) \cong (S_n, I_n)$. □

Lema 4.6. Date su strogo kompatibilne šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)). (S_m, I_m) i (S_n, I_n) nisu u koliziji funkcionalni-nefunkcionalni odnos.

Dokaz. Treba dokazati da važi implikacija $(S_m, I_m) \cong (S_n, I_n) \Rightarrow \neg FnfC(S_m, S_n)$. Uvodi se suprotna pretpostavka, $FnfC(S_m, S_n)$, pri čemu je za šemu relacije $(R_i, \mathcal{K}_i) \in S_m \cup S_n$ ispunjen uslov $Fnf(\mathcal{K}_i, \Gamma_{mn})$, tako da za $K_i \in \mathcal{K}_i$ i $X \subset K_i$, $X \neq \emptyset$, važi $X \subseteq (K_i \setminus X)^+_{\Gamma_{mn}}$. Neka je (S, I) strogo pokrivajuća šema i neka je $(R, \mathcal{K}) \in S$ šema relacije, za koju važi $(R, \mathcal{K}) \supset (R_i, \mathcal{K}_i)$. Sledi da je $R_i \subseteq R$, što znači da $K_i \in \mathcal{K}$, tako da je *fd* $K_i \rightarrow R_i \in (\Gamma(R))^+$ nepotpuna. Zbog toga $(R, \mathcal{K})$ ne zadovoljava *3NF*, a to je u kontradikciji s pretpostavkom da je $(S_m, I_m) \cong (S_n, I_n)$. □

Lema 4.7. Date su strogo kompatibilne šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)). (S_m, I_m) i (S_n, I_n) nisu u koliziji ekvivalentnih ključeva.

Dokaz. Treba dokazati da važi implikacija $(S_m, I_m) \cong (S_n, I_n) \Rightarrow \neg EqKeyC(S_m, S_n)$. Uvodi se suprotna pretpostavka, $EqKeyC(S_m, S_n)$, pri čemu je za šeme relacija $(R_i, \mathcal{K}_i) \in S_m$ i $(R_j, \mathcal{K}_j) \in S_n$ ispunjen uslov $\mathcal{K}_i \cap \mathcal{K}_j = \emptyset \wedge Eqk(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})$. Neka je (S, I) strogo pokrivajuća šema i neka je $(R, \mathcal{K}) \in S$ šema relacije, za koju važi $Ext(R, \mathcal{K}) \supseteq \{(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j)\}$. Saglasno stavovima 2, 3. i 4. definicije 4.5, skup ključeva $\mathcal{K}$ se sastoji od svih ključeva šema relacija iz $Ext(R, \mathcal{K})$:

(4.25) $$\mathcal{K} = \bigcup_{(R_k, \mathcal{K}_k) \in Ext(R, \mathcal{K})} \mathcal{K}_k .$$

Neka je $|Ext(R, \mathcal{K})| = p$ i neka je za svaki $k \in \{1,..., p\}$, $K_k \in \mathcal{K}_k$ proizvoljno odabrani ključ šeme relacije $(R_k, \mathcal{K}_k) \in Ext(R, \mathcal{K})$. Saglasno (4.25) i činjenici $(R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}} \wedge (\exists K_j \in \mathcal{K}_j)(K_j \subseteq (R_i)^+_{\Gamma(S_m)}) \vee (\exists K_i \in \mathcal{K}_i)(K_i \subseteq (R_j)^+_{\Gamma(S_n)})$, postoji neredundantni ciklični niz *fd* $(K_1 \rightarrow K_2, ..., K_{p-1} \rightarrow K_p, K_p \rightarrow K_1)$, takav da važi:

$$\Gamma(R) \models \{K_1 \rightarrow K_2, ..., K_{p-1} \rightarrow K_p, K_p \rightarrow K_1\} \wedge K_i, K_j \in \{K_1, ..., K_p\}.$$

Pošto je niz $(K_1 \rightarrow K_2, ..., K_{p-1} \rightarrow K_p, K_p \rightarrow K_1)$ cikličan, može se zaključiti da važi:

$$\{K_1 \rightarrow K_2, ..., K_{p-1} \rightarrow K_p, K_p \rightarrow K_1\} \models \{K_i \rightarrow K_{(i+1) \bmod p}, K_{(i+1) \bmod p} \rightarrow K_i\},$$

odnosno: $\Gamma(R) \models \{K_i \rightarrow K_{(i+1) \bmod p}, K_{(i+1) \bmod p} \rightarrow K_i\}$, pri čemu je $K_i \in \mathcal{K}_i \wedge K_{(i+1) \bmod p} \notin \mathcal{K}_i$. S obzirom da je niz $(K_1 \rightarrow K_2, ..., K_{p-1} \rightarrow K_p, K_p \rightarrow K_1)$ neredundantan, sledi da $K_{(i+1) \bmod p} \in R_i$, što znači da se $K_{(i+1) \bmod p}$ pojavljuje kao strani ključ šeme relacije $(R_i, \mathcal{K}_i)$.

Iz činjenica $\Gamma(R) \models \{K_i \rightarrow K_{(i+1) \bmod p}, K_{(i+1) \bmod p} \rightarrow K_i\}$ i $K_{(i+1) \bmod p} \in R_i$, sledi da je $\Pi_{R_i}(\Gamma(R)) \models \{K_i \rightarrow K_{(i+1) \bmod p}, K_{(i+1) \bmod p} \rightarrow K_i\}$. S druge strane, iz $K_{(i+1) \bmod p} \notin \mathcal{K}_i$ sledi da $\Gamma(R_i) \not\models \{K_{(i+1) \bmod p} \rightarrow K_i\}$, što znači da je $(\Pi_{R_i}(\Gamma(R)))^+ \neq (\Gamma(R_i))^+$. Činjenica $(\Pi_{R_i}(\Gamma(R)))^+ \neq (\Gamma(R_i))^+$ obara tvrdnju 2. teoreme 3.10, što je u kontradikciji s pretpostavkom da je $(S_m, I_m) \cong (S_n, I_n)$. □

Lema 4.8. Date su strogo kompatibilne šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)). (S_m, I_m) i (S_n, I_n) nisu u koliziji integrativnog svojstva.

Dokaz. Treba dokazati da važi implikacija $(S_m, I_m) \cong (S_n, I_n) \Rightarrow \neg IntC(S_m, S_n)$. Uvodi se suprotna pretpostavka, $IntC(S_m, S_n)$.

(**i**) Neka važi $IntCDk(S_m, S_n)$, pri čemu je za šeme relacija $(R_i, \mathcal{K}_i) \in S_m$ i $(R_j, \mathcal{K}_j) \in S_n$ ispunjen uslov $(R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}} \wedge Int(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma(S_m), \Gamma(S_n))$. Posmatra se slučaj ključa $K_i \in \mathcal{K}_i \setminus \mathcal{K}_j$, takvog da važi $K_i \cap (R_j)^+_{\Gamma_{mn} \setminus \Gamma(R_i)} \neq \emptyset$. (Dokaz je analogan, ukoliko se posmatra "simetrična" situacija, za neki ključ $K_j \in \mathcal{K}_j \setminus \mathcal{K}_i$). Neka je (S, I) strogo pokrivajuća šema i neka je $(R, \mathcal{K}) \in S$ šema relacije, za koju važi $Ext(R, \mathcal{K}) \supseteq \{(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j)\}$. Pri tome je, zbog $(R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}}$, zadovoljeno $\mathcal{K} \supseteq \mathcal{K}_i \cup \mathcal{K}_j$. Posmatraju se relacije $r \in SAT(R, \mathcal{K})$, $r_i \in SAT(R_i, \mathcal{K}_i)$ i $r_j \in SAT(R_j, \mathcal{K}_j)$. Moguća su dva slučaja: $K_i \cap R_j \neq \emptyset$ i $K_i \cap R_j = \emptyset$.

(∗) Pretpostavlja se da važi $K_i \cap R_j \neq \emptyset$. Posmatra se neko obeležje $A \in K_i \cap R_j$. Neka je relacija r_i formirana tako da postoje torke $u_i, v_i \in r_i$, $u_i \neq v_i$, $u_i(A) \neq v_i(A)$, pri čemu za neke torke $u, v \in r$, $u \neq v$, važi $u[R_i] = u_i \wedge v[R_i] = v_i$, što znači da je $u(A) \neq v(A)$. S druge strane, neka su $u_j, v_j \in r_j$ torke, takve da važi $u_j \neq v_j$ i $u[R_j] = u_j \wedge v[R_j] = v_j$. Pošto $K_i \notin \mathcal{K}_j$, sledi da $(\exists r_j' \in SAT(R_i, \mathcal{K}_i))(\exists u_j', v_j' \in r_j')(u_j'[R_j \setminus A] = u_j[R_j \setminus A] \wedge v_j'[R_j \setminus A] = v_j[R_j \setminus A] \wedge u_j'(A) = v_j'(A))$. Iz ovog zaključka i činjenice da je $\mathcal{K} \supseteq \mathcal{K}_j$, proizilazi da mora biti $u[R_j] = u_j' \wedge v[R_j] = v_j'$, odnosno $u(A) = v(A)$, jer je $A \in R_j$. Ovo je, međutim, kontradiktorno prethodnom zaključku $u(A) \neq v(A)$. Time je narušeno integrativno svojstvo (stav 1. definicije 4.3), a to je u kontradikciji s polaznom pretpostavkom $(S_m, I_m) \cong (S_n, I_n)$.

(∗∗) Pretpostavlja se da važi $K_i \cap R_j = \emptyset$. Posmatra se neko obeležje $A \in K_i \cap (R_j)^+_{\Gamma_{mn} \setminus \Gamma(R_i)}$. Sa $(R_k, \mathcal{K}_k) \in S_m \cup S_n$ $(k \neq i)$ će biti označena šema relacije za koju važi $A \in R_k \setminus K_p(R_k)$ i $R_k \subseteq (R_j)^+_{\Gamma_{mn} \setminus \Gamma(R_i)}$. Pored toga, mora postojati šema relacije $(R', \mathcal{K}') \in S$, takva da je $(R_k, \mathcal{K}_k) \in Ext(R', \mathcal{K}')$. Neka su date i relacije $r_k \in SAT(R_k, \mathcal{K}_k)$ i $r' \in SAT(R', \mathcal{K}')$. Pošto važi $fd\ R_j \rightarrow R_k \in \Gamma_{mn} \setminus \Gamma(R_i)^+$, svakoj torki $t_j \in r_j$ odgovara *jedna i samo jedna* torka $t_k \in r_k$, koja će biti obeležena sa $t_k = t_j^+(R_k)$. Pošto je $(R_j, \mathcal{K}_j) \in Ext(R, \mathcal{K})$ i $(R_k, \mathcal{K}_k) \in Ext(R', \mathcal{K}')$, tada važi i $fd\ R \rightarrow R' \in \Gamma(S)^+$, tako da svakoj torki $t \in r$ odgovara *jedna i samo jedna* torka $t' \in r'$, koja će biti obeležena sa $t' = t^+(R')$.

Neka je relacija r_i formirana tako da postoje torke $u_i, v_i \in r_i$, $u_i \neq v_i$, $u_i(A) \neq v_i(A)$, pri čemu za neke torke $u, v \in r$, $u \neq v$, važi $u[R_i] = u_i \wedge v[R_i] = v_i$, odnosno $u(A) \neq v(A)$. Neka za torke $u', v' \in r'$ važi $u' = u^+(R')$ i $v' = v^+(R')$, što znači da je $u'(A) \neq v'(A)$. S druge strane, neka su $u_j, v_j \in r_j$ torke, takve da važi $u_j \neq v_j$ i $u[R_j] = u_j \wedge v[R_j] = v_j$ i neka su $u_k, v_k \in r_k$ torke, za koje je zadovoljeno $u_k = u_j^+(R_k) \wedge v_k = v_j^+(R_k)$. S obzirom da važi $K_i \rightarrow R_j \notin \Gamma_{mn} \setminus \Gamma(R_i)^+$, za pojavu $r_k \in SAT(R_k, \mathcal{K}_k)$ je dozvoljeno da bude $u_k(A) = v_k(A)$, odakle sledi da mora biti $u'(A) = v'(A)$, jer $(R_k, \mathcal{K}_k) \in Ext(R', \mathcal{K}')$. Činjenice $u'(A) = v'(A)$ i $u'(A) \neq v'(A)$ su, međutim, kontradiktorne. Na taj način je oboren stav 2. definicije 4.3, što znači da je narušeno integrativno svojstvo, a time i polazna pretpostavka $(S_m, I_m) \cong (S_n, I_n)$.

(**ii**) Neka važi $IntCLi(S_m, S_n)$ i neka za skupove Si_m i Si_n, koji reprezentuju ekvivalentne parove šema relacija iz S_m i S_n, važi:

$$(\exists ic \in IC_n)(obuhvata(ic, Si_n)) \wedge \neg(\exists ic \in IC_m)(obuhvata(ic, Si_m)).$$

("Simetričan" slučaj se analogno dokazuje.) Polazi se od činjenice da se svi tipovi ograničenja, pokriveni skupovima IC_n (tj. $ID(\mathcal{G}_n, \mathcal{SFu})$, $ID(\mathcal{G}_n, desig)$, ili $DR(\mathcal{G}_n, dep)$) i IC_m mogu definisati samo na nivou šeme modula. Saglasno pretpostavci $\mathcal{K}_{m_i} \cap \mathcal{K}_{n_i} \neq \emptyset$ izraza

(4.23), odnosno činjenici da će za svaki par šema $(R_{m_i}, \mathcal{K}_{m_i})$, $(R_{n_i}, \mathcal{K}_{n_i})$ postojati tačno jedno proširenje, za međurelaciono ograničenje $ic \in IC_n$ će važiti da je $ic \in I$, pri čemu je (S, I) strogo pokrivajuća šema. Činjenica $ic \in I$ je, međutim, kontradiktorna stavu 2. definicije 4.3, jer je nad šemom (S_m, I_m) moguće konstruisati pojavu, koja će narušiti ograničenje ic, pošto $ic \notin I_m$. Na taj način je oborena polazna pretpostavka $(S_m, I_m) \cong (S_n, I_n)$.

(**iii**) Neka važi $IntCGr(S_m, S_n)$, pri čemu je za neke šeme relacija $(R_i, \mathcal{K}_i) \in S_m$ i $(R_j, \mathcal{K}_j) \in S_n$ zadovoljen neki od predikata $RICls(R_i, R_j, \Gamma_{mn}, I_{mn})$, $RICls(R_j, R_i, \Gamma_{mn}, I_{mn})$, $RIRen(R_i, R_j, I_{mn})$, $BRCl(R_i, R_j, I_{mn})$, $BRCl(R_j, R_i, I_{mn})$, ili $BRDer(R_i, R_j, I_{mn})$. U svakom od navedenih slučajeva (saglasno interpretacijama navedenih predikata), postoji međurelaciono ograničenje $ic \in I$, gde je (S, I) strogo pokrivajuća šema, takvo da je $I_m \cup I_n \not\models ic$, odakle sledi $I \not\equiv I_n \cup I_m$. Ova činjenica je, saglasno lemi 4.3, u kontradikciji s polaznom pretpostavkom $(S_m, I_m) \cong (S_n, I_n)$. □

Lema 4.9. Date su šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)). Ako su (S_m, I_m) i (S_n, I_n) u koliziji ekvivalentnih ključeva, tada su one i u koliziji integrativnog svojstva $IntCDk(S_m, S_n)$.

Dokaz. Treba dokazati implikaciju $EqKeyC(S_m, S_n) \Rightarrow IntCDk(S_m, S_n)$.

Neka važi $EqKeyC(S_m, S_n)$, pri čemu je za šeme relacija $(R_i, \mathcal{K}_i) \in S_m$ i $(R_j, \mathcal{K}_j) \in S_n$ ispunjeno $\mathcal{K}_i \cap \mathcal{K}_j = \varnothing \wedge Eqk(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})$:

$$(R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}} \wedge ((\exists K_j \in \mathcal{K}_j)(K_j \subseteq (R_i)^+_{\Gamma(S_m)}) \vee (\exists K_i \in \mathcal{K}_i)(K_i \subseteq (R_j)^+_{\Gamma(S_n)})).$$

Pretpostavlja se da važi $(\exists K_j \in \mathcal{K}_j)(K_j \subseteq (R_i)^+_{\Gamma(S_m)})$. (U slučaju $(\exists K_i \in \mathcal{K}_i)(K_i \subseteq (R_j)^+_{\Gamma(S_n)})$, dokaz je analogan.) Ukoliko za neki ključ $K_j \in \mathcal{K}_j$ važi da je $K_j \subseteq (R_i)^+_{\Gamma(S_m)}$, tada, pošto je $\Gamma_{mn} \setminus \Gamma(R_j) \models \Gamma(S_m)$, proizilazi $K_j \cap (R_i)^+_{\Gamma_{mn} \setminus \Gamma(R_j)} \neq \varnothing$, odakle sledi egzistencija kolizije $IntCDk(S_m, S_n)$. □

Na osnovu leme 4.9. se zaključuje da važi $\neg IntCDk(S_m, S_n) \Rightarrow \neg EqKeyC(S_m, S_n)$, odnosno $\neg IntC(S_m, S_n) \Rightarrow \neg EqKeyC(S_m, S_n)$. To omogućava da se u narednoj lemi uslov $\neg EqKeyC(S_m, S_n)$ izostavi iz njene pretpostavke.

Lema 4.10. Date su šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)). Ukoliko (S_m, I_m) i (S_n, I_n) nisu u koliziji tranzitivnosti, koliziji funkcionalni-nefunkcionalni odnos i koliziji integrativnog svojstva i ako važi:

(4.26) $$(\forall (R_i, \mathcal{K}_i) \in S_m)(\forall (R_j, \mathcal{K}_j) \in S_n)(\mathcal{K}_i \cap \mathcal{K}_j \neq \varnothing \Rightarrow$$
$$(\forall K_i, K_j \in (\mathcal{K}_i \cup \mathcal{K}_j) \setminus (\mathcal{K}_i \cap \mathcal{K}_j))(\,|\,dom[K_i]\,| = |\,dom[K_j]\,|\,)),$$

tada su (S_m, I_m) i (S_n, I_n) strogo kompatibilne.

Dokaz. Treba dokazati, pod pretpostavkom važenja uslova (4.26), implikaciju:

$$\neg(TranC(S_m, S_n) \vee FnfC(S_m, S_n) \vee IntC(S_m, S_n)) \Rightarrow (S_m, I_m) \cong (S_n, I_n),$$

odnosno, treba pokazati da postoji strogo pokrivajuća šema (S, I).

Neka je sa $Ekviv(S_m \cup S_n)$ označen skup klasa ekvivalencije (particija skupa $S_m \cup S_n$), saglasno istim zatvaračima skupova obeležja šema relacija:

(4.27) $Ekviv(S_m \cup S_n) = \{S_{Eq}(R_i) \subseteq S_m \cup S_n \mid (R_i, \mathcal{K}_i) \in S_m \cup S_n \wedge$

$$(\forall (R_j, \mathcal{K}_j) \in S_m \cup S_n)((R_j, \mathcal{K}_j) \in S_{Eq}(R_i) \Leftrightarrow (R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}})\},$$

gde je $\Gamma_{mn} = \Gamma(S_n) \cup \Gamma(S_m)$. Posmatra se šema (S, I), takva da važi:

(4.28) $$S = \{(R, \mathcal{K}) \mid (\exists S_{Eq} \subseteq Ekviv(S_m \cup S_n))(R = \bigcup_{(R_i, \mathrm{K}_i) \in S_{Eq}} R_i \wedge \mathcal{K} = \bigcup_{(R_i, \mathrm{K}_i) \in S_{Eq}} \mathrm{K}_i)\}.$$

Činjenica da je šema $(R, \mathcal{K}) \in S$ formirana iz klase ekvivalencije $S_{Eq} \in Ekviv(S_m \cup S_n)$ će biti označavana sa $(R, \mathcal{K}) = Uniran(S_{Eq})$.

Skup međurelacionih ograničenja šeme (S, I) je dat kao:

(4.29) $$I = I_n \cup I_m .$$

U nastavku će biti dokazano da šema (S, I), zadata izrazima (4.28) i (4.29), predstavlja strogo pokrivajuću šemu za (S_m, I_m) i (S_n, I_n), odnsno da zadovoljava sve uslove definicija 4.5. i 4.3.

(**i**) Tvrđenje $U_{BP}(S) = U_{BP}(S_m) \cup U_{BP}(S_n)$ proizilazi direktno iz izraza (4.28). $U_{BP}(S) \subseteq U_{BP}(S_m) \cup U_{BP}(S_n)$ sledi iz činjenice da je za svaku šemu $(R, \mathcal{K}) \in S$ skup obeležja R formiran samo na osnovu skupova obeležja nekih šema relacija iz $S_m \cup S_n$, dok je $U_{BP}(S_m) \cup U_{BP}(S_n) \subseteq U_{BP}(S)$ posledica činjenice da su sve šeme relacija iz skupa $S_m \cup S_n$ učestvovale u formiranju neke šeme relacije iz S.

(**ii**) Dokazuje se da važi $\Gamma(S) \equiv \Gamma(S_n) \cup \Gamma(S_m)$.

($\Rightarrow$) Posmatra se proizvoljna šema relacije $(R_k, \mathcal{K}_k) \in S_m \cup S_n$. saglasno izrazu (4.28) i činjenici da $Ekviv(S_m \cup S_n)$ definiše particiju skupa $S_m \cup S_n$, postoji jedna šema relacije $(R, \mathcal{K}) \in S$, takva da je $R_k \subset R$ i $\mathcal{K}_k \subset \mathcal{K}$, odakle proizilazi $\Gamma(S) \models \Gamma(R) \models \Gamma(R_k)$. Pošto za svaku šemu relacije $(R_k, \mathcal{K}_k) \in S_m \cup S_n$ važi $\Gamma(S) \models \Gamma(R_k)$, zaključuje se da $\Gamma(S) \models \Gamma(S_n) \cup \Gamma(S_m)$.

($\Leftarrow$) Neka je $(R, \mathcal{K}) \in S$ proizvoljno odabrana šema relacije. Pošto je $\mathcal{K} = \bigcup_{(R_i, \mathrm{K}_i) \in S_{Eq}} \mathrm{K}_i$ i $S_{Eq} \subseteq S_m \cup S_n$, zaključuje se da važi $\Gamma(S_n) \cup \Gamma(S_m) \models \Gamma(R)$. Budući da za svaku šemu $(R, \mathcal{K}) \in S$ važi $\Gamma(S_n) \cup \Gamma(S_m) \models \Gamma(R)$, proizilazi da $\Gamma(S_n) \cup \Gamma(S_m) \models \Gamma(S)$.

Iz $\Gamma(S) \models \Gamma(S_n) \cup \Gamma(S_m)$ i $\Gamma(S_n) \cup \Gamma(S_m) \models \Gamma(S)$ se zaključuje da je $\Gamma(S) \equiv \Gamma(S_n) \cup \Gamma(S_m)$.

(**iii**) Treba dokazati da (S, I) zadovoljava uslov *3NF*. Polazi se od činjenice da (S_m, I_m) i (S_n, I_n) zadovoljavaju uslov *3NF*. Pošto važi $\neg FnfC(S_m, S_n)$, zaključuje se da za svaku šemu relacije $(R, \mathcal{K}) \in S$, za proizvoljne $K \in \mathcal{K}$ i $A \in R \setminus K_p(R)$ važi da je *fd* $K \rightarrow A$ potpuna s obzirom na skup $\Gamma(S)$. Slično, zbog $\neg TranC(S_m, S_n)$, važi da $K \rightarrow A$ nije tranzitivna s obzirom na $\Gamma(S)$. $(R, \mathcal{K})$, dakle, zadovoljava uslov *3NF*.

(**iv**) Dokazuje se da važi $(\forall (R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j) \in S)(i \neq j \Rightarrow (R_i)^+_{\Gamma(S)} \neq (R_j)^+_{\Gamma(S)})$. Uvodi se suprotna pretpostavka, prema kojoj za šeme $(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j) \in S$, $R_i \neq R_j$, važi da je $(R_i)^+_{\Gamma(S)} = (R_j)^+_{\Gamma(S)}$. Neka je $(R_i, \mathcal{K}_i) = Uniran(S_{Eq_i})$ i $(R_j, \mathcal{K}_j) = Uniran(S_{Eq_j})$, pri čemu mora da važi $S_{Eq_i} \cap S_{Eq_j} = \varnothing$, jer je $R_i \neq R_j$. Prema (4.27), znači da za proizvoljne šeme relacija $(R_p, \mathcal{K}_p) \in S_{Eq_i}$ i $(R_q, \mathcal{K}_q) \in S_{Eq_j}$ važi da je $(R_p)^+_{\Gamma(S)} \neq (R_q)^+_{\Gamma(S)}$, a pošto je $\mathcal{K}_p \subseteq \mathcal{K}_i$ i $\mathcal{K}_q \subseteq \mathcal{K}_i$, sledi $(R_i)^+_{\Gamma(S)} \neq (R_j)^+_{\Gamma(S)}$, što obara suprotnu pretpotavku.

(**v**) Biće dokazano da je $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$. Neka je izabrana proizvoljna šema relacije $(R_k, \mathcal{K}_k) \in S_m \cup S_n$ i neka je $(R, \mathcal{K}) = Uniran(S_{Eq}(R_k))$, iz čega sledi da je $R_k \subseteq R$ i $\mathcal{K}_k \subseteq \mathcal{K}$, odnosno $\Pi_{R_k}(\Gamma(R)) \equiv \Gamma(R_k)$. Za bilo koje obeležje $A \in R_k$ i proizvoljni ključ $K \in \mathcal{K}$ je zadovoljeno da $K \rightarrow A \in \Gamma(R)$. Ako je $K \in \mathcal{K}_k$ tada, na osnovu teoreme 3.9, direktno sledi da je $(R, \mathcal{K}) \supset (R_k, \mathcal{K}_k)$. Neka postoji $K \in \mathcal{K}$, takav da $K \notin \mathcal{K}_k$ i neka je $K_k \in \mathcal{K}_k$ proizvoljno odabrani ključ šeme relacije $(R_k, \mathcal{K}_k)$. Pošto je (S, I) u *3NF* i važi $K \rightarrow K_k \in \Gamma(R) \wedge K_k \rightarrow K \in \Gamma(R)$, jer je zadovoljeno $\neg EqKeyC(S_m, S_n)$, kao posledica $\neg IntC(S_m, S_n)$, može se zaključiti da $(\forall f \in \Pi_{R_k}(\Gamma(R)^{ifd}))(f \in \Gamma(R_k)^{ifd})$. Kako važi (4.26), na osnovu teoreme 3.10, proizilazi $(R, \mathcal{K}) \supset (R_k, \mathcal{K}_k)$. S obzirom da je $(R_k, \mathcal{K}_k)$ proizvoljno odabrana šema relacije, sledi da je $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$.

(**vi**) Uslov $(\forall (R_i, \mathcal{K}_i) \in S)(\exists (R_j, \mathcal{K}_j) \in S_m \cup S_n)((R_i, \mathcal{K}_i) \supset (R_j, \mathcal{K}_j))$ sledi direktno iz izraza (4.28) i prethodno dokazane činjenice $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$.

(**vii**) Pošto važi $(S, I) \supset (S_m, I_m) \wedge (S, I) \supset (S_n, I_n)$, dokaz da je zadovoljeno integrativno svojstvo $(S, I) = Int((S_m, I_m), (S_n, I_n))$ se svodi na dokaz da važe stavovi 1. i 2. definicije 4.3. Uvodi se suprotna pretpostavka, po kojoj je $(S, I) \neq Int((S_m, I_m), (S_n, I_n))$.

(∗) Pretpostavlja se da je narušen stav 1. definicije 4.3. Neka je $(R, \mathcal{K}) \in S$ i neka važi $NonEqDk(Ext(R, \mathcal{K})) \wedge (\forall r \in SAT(R, \mathcal{K}))(r \notin Ext(S))$. S druge strane, neka je $(R, \mathcal{K}) = Uniran(S_{Eq})$, pri čemu je $S_{Eq} \in Ekviv(S_m \cup S_n)$. Činjenica da za svaku relaciju $r \in SAT(R, \mathcal{K})$ važi $r \notin Ext(S)$ znači da za neku kombinaciju relacija $(r_1, ..., r_k) \in SAT(S_{Eq})$ (pri čemu je $S_{Eq} = Ext(R, \mathcal{K})$) nije moguće formirati $Ext_R(r_1, ..., r_k)$. Saglasno izrazu (4.8), proiziliazi da za bar dve relacije $r_i, r_j \in \{r_1, ..., r_k\}$, $r_i \in SAT(R_i, \mathcal{K}_i)$, $r_j \in SAT(R_j, \mathcal{K}_j)$, $(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j) \in S_{Eq}$, i za svaku relaciju $r \in SAT(R, \mathcal{K})$ važi $r_i \neq \Pi_{R_i}(r) \vee r_j \neq \Pi_{R_j}(r)$. Prema (4.27), pripadnost klasi ekvivalencije $(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j) \in S_{Eq}$ implicira $(R_i)^+_{\Gamma(S)} = (R_j)^+_{\Gamma(S)}$, a s obzirom na uslov $NonEqDk(Ext(R, \mathcal{K}))$, zaključuje se da mora biti zadovoljeno $(\exists K_i \in \mathcal{K}_i \setminus \mathcal{K}_j)(K_i \cap R_j \neq \varnothing) \vee (\exists K_j \in \mathcal{K}_j \setminus \mathcal{K}_i)(K_j \cap R_i \neq \varnothing)$.

Neka je za neki ključ $K_i \in \mathcal{K}_i \setminus \mathcal{K}_j$ ispunjen uslov $K_i \cap R_j \neq \varnothing$ (za "simetričan" slučaj, dokaz je analogan) i neka je dato obeležje $A \in K_i \cap R_j$, takvo da za torke $u_i, v_i \in r_i$, $u_j, v_j \in r_j$, $u, v \in r$, $u_i \neq v_i$ i $u_j \neq v_j$, važi $u_i[R_i] = u \wedge v_i[R_i] = v \wedge u_j[R_j] = u \wedge v_j[R_j] = v \wedge u_i(A) = v_i(A) \wedge u_j(A) \neq v_j(A)$. Situacija $A \in K_i \cap R_j$ znači da $(\exists K_j \in \mathcal{K}_j \setminus \mathcal{K}_i)(K_j \cap (R_i)^+_{\Gamma_{mn} \setminus \Gamma(R_j)} \neq \varnothing)$, što je kontradiktorno pretpostavci $\neg IntC(S_m, S_n)$.

(∗∗) Pretpostavlja se da je narušen stav 2. definicije 4.3. Neka je $(r_1^n, ..., r_p^n)$ pojava BP nad šemom (S_n, I_n), $(r_1^m, ..., r_q^m)$ pojava nad (S_m, I_m), a $Ext(S) = \{Ext_R(r_1, ..., r_k) \mid (r_1, ..., r_k) \in SAT(S_{Eq}) \wedge S_{Eq} \in Ekviv(S_m \cup S_n)\}$ skup pojava nad šemama relacija iz S, dobijen proširenjem relacija iz $(r_1^n, ..., r_p^n)$ i $(r_1^m, ..., r_q^m)$. Neka je $ic \in I$ međurelaciono ograničenje, koje skup $Ext(S)$ narušava. Pošto je $I = I_n \cup I_m$, važi $ic \in I_n \cup I_m$, tako da $(r_1^n, ..., r_p^n)$, ili $(r_1^m, ..., r_q^m)$ narušava ic, što je kontradiktorno činjenici da su $(r_1^n, ..., r_p^n)$ i $(r_1^m, ..., r_q^m)$ pojave nad šemama modula.

Prema (∗) i (∗∗), zaključuje se da važi $(S, I) = Int((S_m, I_m), (S_n, I_n))$. □

Uslov (4.26) se može iskazati i u obliku $(\forall K_i \in (\mathcal{K}_i \cup \mathcal{K}_j) \setminus (\mathcal{K}_i \cap \mathcal{K}_j))(|dom[K_i]| = \infty)$. U praksi je, formalno, nemoguće obezbediti beskonačnost domena, ali je moguće obezbediti dovoljno veliki opseg mogućih vrednosti za svako obeležje ključa, tako da navedeni uslov bude praktično ostvarljiv. Praktične implikacije uslova (4.26) su da sistemom šifriranja treba predvideti da svakoj aktuelnoj vrednosti ključa $K \in \mathcal{K}_i \cap \mathcal{K}_j$ može biti dodeljena neka vrednost ključa $K_i \in (\mathcal{K}_i \cup \mathcal{K}_j) \setminus (\mathcal{K}_i \cap \mathcal{K}_j)$.

Teorema 4.1. Date su šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)), pri čemu važi uslov (4.26). (S_m, I_m) i (S_n, I_n) su međusobno strogo kompatibilne, ako i samo ako nisu u tranzitivnoj koliziji, *fnf*-koliziji i koliziji integrativnog svojstva.

Dokaz. Treba dokazati da važi ekvivalencija:

$$(S_m, I_m) \cong (S_n, I_n) \Leftrightarrow \neg(TranC(S_m, S_n) \vee FnfC(S_m, S_n) \vee IntC(S_m, S_n)).$$

($\Rightarrow$) Proizilazi iz lema 4.5, 4.6. i 4.8.

($\Leftarrow$) Direktno sledi iz leme 4.10. □

Teorema 4.2. Date su strogo kompatibilne šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)), na osnovu skupova tipova formi za ažuriranje, redom, $\mathcal{SFu}_m$ i $\mathcal{SFu}_n$. Strogo pokrivajuća šema(S, I), formirana putem izraza (4.28) i (4.29), predstavlja izlaz modifikovanog algoritma sinteze, primenjenog na skup tipova formi za ažuriranje $\mathcal{SFu} = \mathcal{SFu}_m \cup \mathcal{SFu}_n$.

Dokaz. Saglasno definiciji 4.5, uslovima (4.14), (4.15) i (4.16) i izrazu (4.28), sledi da skup šema relacija strogo pokrivajuće šeme S ispunjava sve projektantske uslove, implicirane modifikovanim algoritmom sinteze, za ulazni skup tipova formi $\mathcal{SFu} = \mathcal{SFu}_m \cup \mathcal{SFu}_n$. Pored toga, potrebno je dokazati da skup međurelacionih ograničenja $I = I_n \cup I_m$ zadovoljava formulu (2.37). Uvodi se suprotna pretpostavka, po kojoj je:

$$I \not\equiv RI(\mathcal{G}) \cup BR(\mathcal{G}) \cup ID(\mathcal{G}, \mathcal{SFu}) \cup ID(\mathcal{G}, desig) \cup RI_r(ren, dbs) \cup BR(der, dbs) \cup DR(\mathcal{G}, dep),$$

gde je $\mathcal{G}$ graf zatvaranja šeme (S, I). Pošto je $I = I_n \cup I_m$, pri čemu I_m i I_n zadovoljavaju (2.37), sledi da mora biti $RI(\mathcal{G}) \cup BR(\mathcal{G}) \cup ID(\mathcal{G}, \mathcal{SFu}) \cup ID(\mathcal{G}, desig) \cup RI_r(ren, dbs) \cup BR(der, dbs) \cup DR(\mathcal{G}, dep) \models I$. Prema tome, analizira se situacija $I \not\models RI(\mathcal{G}) \cup BR(\mathcal{G}) \cup ID(\mathcal{G}, \mathcal{SFu}) \cup ID(\mathcal{G}, desig) \cup RI_r(ren, dbs) \cup BR(der, dbs) \cup DR(\mathcal{G}, dep)$. Moguća su tri slučaja.

(**i**) Neka je $ic \in (RI(\mathcal{G}) \cup BR(\mathcal{G}) \cup RI_r(ren, dbs) \cup BR(der, dbs)) \setminus I$, tako da $I_n \cup I_m \not\models ic$. Sledi da su (S_m, I_m) i (S_n, I_n) u koliziji integrativnog svojstva, zbog važenja predikata $IntCGr(S_m, S_n)$.

(**ii**) Situacija $ic \in (ID(\mathcal{G}, \mathcal{SFu}) \cup ID(\mathcal{G}, desig)) \setminus I$, tako da $I_n \cup I_m \not\models ic$, je nemoguća, jer važi $\neg IntCLi(S_m, S_n)$, što znači da je $ID(\mathcal{G}, \mathcal{SFu}) = ID(\mathcal{G}_m, \mathcal{SFu}_m) \cup ID(\mathcal{G}_n, \mathcal{SFu}_n)$ i $ID(\mathcal{G}, desig) = ID(\mathcal{G}_m, desig) \cup ID(\mathcal{G}_n, desig)$.

(**iii**) Situacija $ic \in DR(\mathcal{G}, dep) \setminus I$, tako da $I_n \cup I_m \not\models ic$, je nemoguća jer mora biti zadovoljeno $DR(\mathcal{G}, dep) = DR(\mathcal{G}_m, dep) \cup DR(\mathcal{G}_n, dep)$. □

Jednakost $DR(\mathcal{G}, dep) = DR(\mathcal{G}_m, dep) \cup DR(\mathcal{G}_n, dep)$ zahteva dodatni komentar. Skupovi $DR(\mathcal{G}_m, dep)$ i $DR(\mathcal{G}_n, dep)$ se kreiraju u interakciji s projektantom (videti **[MR2]**), nakon generisanja skupa šema relacija modula, prema izrazu (2.27). Po integraciji strogo kompatibilnih šema modula, strogo pokrivajuća šema modula (S, I) mora biti nezavisna, ako su polazne šeme modula bile nezavisne. Može se pokazati da strogo pokrivajuća šema modula ne može biti B-zavisna, jer bi B-zavisnost, u tom slučaju, značila pojavu kolizije integrativnog svojstva $IntCDk(S_m, S_n)$. Moguća je, međutim, pojava slučaja:

$$(4.30) \qquad (\exists (R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j) \in S)(\exists K \in \mathcal{K}_i)((R_i \setminus K) \cap (R_j)^+_{\Gamma(S)\setminus\Gamma(R_i)} \neq \emptyset),$$

koji bi ukazao na eventualnu A-zavisnost, odnosno na potrebu definisanja međurelacionog ograničenja $ic \in DR(\mathcal{G}, dep) \setminus (DR(\mathcal{G}_m, dep) \cup DR(\mathcal{G}_n, dep))$. Definisanje (u interakciji s projektantom) ograničenja *ic* s navedenom osobinom se ne sme dozvoliti, jer bi time moglo doći do narušavanja integrativnog svojstva (stava 2, definicije 4.3), odnosno ekstenzionalne kompatibilnosti. Naredni primer ilustruje jednu takvu situaciju.

Primer 4.24. Date su strogo kompatibilne, nezavisne, šeme modula BP (S_m, I_m) i (S_n, I_n):

- $S_m = \{(ABE, \{AB \to E\}), (AC, \{A \to C\}), (CD, \{C \to D\})\}$,
- $I_m = \{ABE[A] \subseteq AC[A], AC[C] \subseteq CD[C]\}$,
- $S_n = \{(ABE, \{AB \to E\}), (BC, \{B \to C\}), (CD, \{C \to D\})\}$ i
- $I_n = \{ABE[B] \subseteq BC[B], BC[C] \subseteq CD[C]\}$.

Strogo pokrivajuća šema za (S_m, I_m) i (S_n, I_n) je (S, I):

$$S = S_m \cup S_n = \{(ABE, \{AB \to E\}), (AC, \{A \to C\}), (BC, \{B \to C\}), (CD, \{C \to D\})\},$$

$$I = I_m \cup I_n = \{ABE[A] \subseteq AC[A], ABE[B] \subseteq BC[B], AC[C] \subseteq CD[C]\}.$$

(S, I) zadovoljava kriterijum zavisnosti (4.30), što bi mogao biti povod za definisanje pravila poslovanja $\rhd\lhd(ABE, BC)[AC] \subseteq AC[AC] \in DR(\mathcal{G}, dep) \setminus (DR(\mathcal{G}_m, dep) \cup DR(\mathcal{G}_n, dep))$ (ili, alternativno, pravila $\rhd\lhd(ABE, AC)[BC] \subseteq BC[BC]$). Definisanjem ovakvog pravila bi se, međutim, narušilo integrativno svojstvo, jer takvo pravilo ne bi bilo posledica skupa međurelacionih ograničenja $I_m \cup I_n$. Nedefinisanjem pomenutog pravila poslovanja, obeležje *C* dobija ulogu obeležja - homonima, pošto ima dve različite uloge - jednu u kontekstu šeme relacije $(AC, \{A \to C\})$ i drugu u kontekstu šeme relacije $(BC, \{B \to C\})$. Drugim rečima, obeležje *C* može poprimiti takve vrednosti u okviru pojava nad šemama $(AC, \{A \to C\})$ i $(BC, \{B \to C\})$, koje će, globalno, narušiti tranzitivnu *fd* $AB \to C \in \Gamma^+_{mn}$, što znači da nije sve jedno da li se vrednost obeležja *C* dobija u nekom upitu na osnovu zadate vrednosti obeležja *A*, ili na osnovu zadate vrednosti obeležja *B*. Time se, u krajnjoj liniji, narušava i polazna pretpostavka o postojanju šeme univerzalne relacije. □

4.5. Potreban i dovoljan uslov međusobne intenzionalne kompatibilnosti

Zahtev za ekstenzionalnom kompatibilnošću, koja je implicirana stavovima 1. i 2, definicije 4.3. (tj. integrativnim svojstvom), u praksi može predstavljati dosta strogo ograničenje. Opravdanje za ovu tvrdnju leži u činjenici da pojava kolizija integrativnog svojstva $IntC(S_m, S_n)$ često ne mora značiti da se šeme (S_m, I_m) i (S_n, I_n) ne mogu integrisati u pokrivajuću šemu (S, I), već da integraciji pojava nad šemama (S_m, I_m) i (S_n, I_n) mora prethoditi restruktuiranje podataka u cilju njihovog usaglašavanja sa skupom ograničenja $\Gamma(S) \cup I$, intenzionalno pokrivajuće šeme (S, I). Zbog toga, uslov međusobne intenzionalne kompatibilnosti $(S_m, I_m) \cong_I (S_n, I_n)$ može biti, s praktičnog stanovišta, zadovoljavajući kriterijum mogućnosti integracije u pokrivajuću šemu BP.

U nastavku rada se formuliše potreban i dovoljan uslov međusobne intenzionalne kompatibilnosti šema modula.

Teorema 4.3. Date su šeme modula (S_m, I_m) i (S_n, I_n), izgenerisane modifikovanim algoritmom sinteze (izrazi (2.27) i (2.37)), pri čemu važi uslov (4.26). (S_m, I_m) i (S_n, I_n) su međusobno intenzionalno kompatibilne, ako i samo ako nisu u tranzitivnoj koliziji, *fnf*-koliziji i koliziji ekvivalentnih ključeva.

Dokaz. Treba dokazati da važi ekvivalencija:

$$(S_m, I_m) \cong_I (S_n, I_n) \Leftrightarrow \neg(TranC(S_m, S_n) \vee FnfC(S_m, S_n) \vee EqKeyC(S_m, S_n)).$$

($\Rightarrow$) Proizilazi iz lema 4.5, 4.6. i 4.7.

($\Leftarrow$) Treba dokazati, pod pretpostavkom važenja uslova (4.26), implikaciju:

$$\neg(TranC(S_m, S_n) \vee FnfC(S_m, S_n) \vee EqKeyC(S_m, S_n)) \Rightarrow (S_m, I_m) \cong_I (S_n, I_n),$$

odnosno, treba pokazati da postoji intenzionalno pokrivajuća šema (S, I). U nastavku će biti dokazano da šema (S, I), pri čemu je skup šema relacija zadat izrazom (4.28), i važi $I \supseteq I_m \cup I_n$, predstavlja intenzionalno pokrivajuću šemu za (S_m, I_m) i (S_n, I_n), odnsno da zadovoljava uslove 1-7. definicije 4.6.

(**i**) Uslov 2. definicije 4.6. se dokazuje na isti način kao u stavu (**i**) leme 4.10.
(**ii**) Uslov 4. definicije 4.6. se dokazuje na isti način kao u stavu (**ii**) leme 4.10.
(**iii**) Uslov 3. definicije 4.6. se dokazuje na isti način kao u stavu (**iii**) leme 4.10.
(**iv**) Uslov 6. definicije 4.6. se dokazuje na isti način kao u stavu (**iv**) leme 4.10.
(**v**) Uslov 1. definicije 4.6. se dokazuje na isti način kao u stavu (**v**) leme 4.10.
(**vi**) Uslov 7. definicije 4.6. se dokazuje na isti način kao u stavu (**vi**) leme 4.10.
(**vii**) Uslov 5. definicije 4.6. sledi direktno iz pretpostavke $I \supseteq I_m \cup I_n$. □

4.6. Algoritmi testa međusobne stroge kompatibilnosti

Teorema 4.1. predstavlja osnov za formalizaciju postupka testiranja međusobne stroge kompatibilnosti šema modula, dok teorema 4.3. predstavlja osnov za formalizaciju postupka testiranja međusobne intenzionalne kompatibilnosti šema modula. U nastavku će biti prikazani algoritmi testa kolizije tranzitivnosti, *fnf*-kolizije, kolizije integrativnog svojstva i kolizije ekvivalentnih ključeva, odnosno testa potrebnog i dovoljnog uslova međusobne stroge kompatibilnosti i međusobne intenzionalne kompatibilnosti šema modula.

Algoritmi za testiranje kolizija su konstruisani tako da u slučaju otkrivanja kolizionih situacija formiraju izveštaje koji mogu sadržati ***poruke*** (algoritam 4.7), ***upozorenja*** i ***greške***. Pri tome, izveštaj će sadržati odgovarajuće *upozorenje* u svim situacijama u kojima je narušena ekstenzionalna (a samim tim i stroga) kompatibilnost, ili su otkriveni mogući homonimi, dok će se *greške* pojaviti u svim situacijama narušavanja intenzionalne kompatibilnosti. *Poruke* (u algoritmu 4.7) ukazuju na mogućnost eventualnog narušavanja ekstenzionalne kompatibilnosti.

Navedeni algoritmi za otkrivanje kolizija pozivaju algoritam:

Zatvarač(**U**(*Att*, Γ), **UI**(), **I**(*ClosAtt*))

čiji je zadatak da, na osnovu zadatog skupa obeležja *Att* i skupa *fd* Γ, izračuna zatvarač skupa obeležja $ClosAtt = (Att)^+_\Gamma$. Za potrebe ovog rada je dovoljno napomenuti da je algoritam ***Zatvarač***, u najlošijem slučaju, kvadratne složenosti s obzirom na kardinalnost skupa *fd* Γ (opis samog algoritma se može naći u literaturi **[M**, **U**, **PBG]**).

U nastavku ovog poglavlja, uz svaki algoritam je data i teorema kojom se obrazlaže korektnost razmatranog algoritma i daje procena njegove složenosti. Složenost svih algoritama je procenjena saglasno pretpostavci da ako su X i Y skupovi obeležja (šeme relacije), tada se testovi $X \subseteq Y$, $X \not\subseteq Y$, $X = \emptyset$, $X \neq \emptyset$ i skupovne operacije $X \cup Y$, $X \setminus Y$, $X \cap Y$

smatraju *elementarnim* koracima algoritma (koracima složenosti $O(1)$). Algoritmi su prezentovani u nerekurzivnoj formi.

ALGORITAM 4.1. TEST KOLIZIJE TRANZITIVNOSTI

PROCES *Test_kolizije_tranzitivnosti* (**U**(S_m, Γ_m, S_n, Γ_n)), **UI**(), **I**(*Ind*, *TranC_Rep*))
(∗ **U:** S_m, S_n - Skupovi šema relacija šema modula, izgenerisani modifikovanim algoritmom sinteze (izraz (2.27))
$\Gamma_m = \Gamma(S_m)$, $\Gamma_n = \Gamma(S_n)$ - Skupovi *fd* šema (S_m, I_m) i (S_n, I_n), dati izrazom (2.17)
I: $Ind \in \{true, false\}$ - Indikator uspešnosti testa $TranC(S_m, S_n)$
TranC_Rep - Izveštaj s upozorenjima i greškama koje ukazuju na kolizije. ∗)
POČETAK PROCESA *Test_kolizije_tranzitivnosti*
  POSTAVI $\Gamma_{mn} \leftarrow \Gamma_m \cup \Gamma_n$
  POSTAVI *Ind* ← *true* (∗ Inicijalno uspešan test ∗)
  POSTAVI *TranC_Rep* ← ∅
  RADI *izračunaj_zatvarače* ***ZA*** $(\forall (R_i, \mathcal{K}_i) \in S_m \cup S_n)$
    POZOVI *Zatvarač*(R_i, Γ_{mn}, Cl_R_i)
  KRAJ RADI *izračunaj_zatvarače*
  RADI *prolaz_kroz_m* ***ZA*** $(\forall (R_i, \mathcal{K}_i) \in S_m)$
    RADI *prolaz_kroz_n* ***ZA*** $(\forall (R_j, \mathcal{K}_j) \in S_n)$
      POSTAVI $TrAtt \leftarrow (R_i \setminus K_p(R_i)) \cap (R_j \setminus K_p(R_j))$
      AKO JE $(\mathcal{K}_i \cap \mathcal{K}_j = \emptyset) \wedge (TrAtt \neq \emptyset)$ ***TADA***
        AKO JE $(R_i \subseteq Cl\_R_j) \wedge (R_j \not\subseteq Cl\_R_i)$ ***TADA***
          POSTAVI *Ind* ← *false* (∗ Neuspešan test ∗)
          POSTAVI *TranC_Rep* ← *TranC_Rep* ∪ {(‘*Greška*’, $\mathcal{K}_i$, $\mathcal{K}_j$, *TrAtt*, $R_j \rightarrow R_i$, ‘*Tranzitivna obeležja u šemi relacije:*’, R_j)}
        INAČE AKO JE $(R_i \not\subseteq Cl\_R_j) \wedge (R_j \subseteq Cl\_R_i)$ ***TADA***
          POSTAVI *Ind* ← *false* (∗ Neuspešan test ∗)
          POSTAVI *TranC_Rep* ← *TranC_Rep* ∪ {(‘*Greška*’, $\mathcal{K}_i$, $\mathcal{K}_j$, *TrAtt*, $R_i \rightarrow R_j$, ‘*Tranzitivna obeležja u šemi relacije:*’, R_i)}
        INAČE AKO JE $(R_i \not\subseteq Cl\_R_j) \wedge (R_j \not\subseteq Cl\_R_i)$ ***TADA***
          POSTAVI *TranC_Rep* ← *TranC_Rep* ∪ {(‘*Upozorenje*’, $\mathcal{K}_i$, $\mathcal{K}_j$, *TrAtt*, $R_i \not\leftrightarrow R_j$, ‘*Mogući homonimi*’)}
        INAČE (∗ $(R_i \subseteq Cl\_R_j) \wedge (R_j \subseteq Cl\_R_i)$ ∗)
          POSTAVI *TranC_Rep* ← *TranC_Rep* ∪ {(‘*Upozorenje*’, $\mathcal{K}_i$, $\mathcal{K}_j$, *TrAtt*, $R_i \leftrightarrow R_j$, ‘*Moguća kolizija ekvivalentnih ključeva*’)}
      KRAJ AKO
    KRAJ RADI *prolaz_kroz_n*
  KRAJ RADI *prolaz_kroz_m*
KRAJ PROCESA *Test_kolizije_tranzitivnosti*

Slika 4.6.

Na slici 4.6. je prikazan algoritam testa kolizije tranzitivnosti $TranC(S_m, S_n)$.

Teorema 4.4. Algoritam 4.1. je korektan, polinomijalne složenosti ($O(n^3)$) u najlošijem slučaju.

Dokaz. (*Korektnost*) Kompozicija petlji *prolaz_kroz_m* i *prolaz_kroz_n* obezbeđuje da se ispitaju svi parovi šema relacija $(R_i, \mathcal{K}_i) \in S_m$ i $(R_j, \mathcal{K}_j) \in S_n$ i utvrdi da li postoji takav par šema relacija za koji je zadovoljen kriterijum:

$$(\mathcal{K}_i \cap \mathcal{K}_j = \emptyset) \wedge ((R_i \setminus K_p(R_i)) \cap (R_j \setminus K_p(R_j)) \neq \emptyset) \wedge (R_j \subseteq Cl\_R_i \veebar R_i \subseteq Cl\_R_j).$$

Pri tome, saglasno definiciji 4.6, vrednost promenljive *Ind* $\in$ {*true*, *false*} odgovara logičkoj vrednosti interpretacije predikata $TranC(S_m, S_n)$.

(*Složenost*) Neka je $n = max(|S_m \cup S_n|, |\Gamma_{mn}|)$. Složenost algoritma *Zatvarač* je $O(n^2)$, a složenost petlje *izračunaj_zatvarače* je $O(n^2) \cdot O(n) = O(n^3)$. Ako je $m = max(|S_m|, |S_n|)$ $(m < n)$, tada je složenost kompozicije petlji *prolaz_kroz_m* i *prolaz_kroz_n* $O(m^2)$, što znači da je složenost celog algoritma data kao $O(n^3) + O(m^2) = O(n^3)$. □

Na slici 4.8. je prikazan algoritam testa *fnf*-kolizije $FnfC(S_m, S_n)$. Algoritam poziva proces *Redukcija_skupa*, čiji je zadatak da za zadati skup obeležja i zadati skup *fd* odredi da li postoji obeležje koje je funkcionalno zavisno od ostalih obeležja skupa, i koji je prikazan na slici 4.7.

Teorema 4.5. Algoritam 4.3. je korektan, polinomijalne složenosti $(O(n^3))$ u najlošijem slučaju.

Dokaz. (*Korektnost*) Korektnost algoritma *Redukcija_skupa* proizilazi iz korektnosti algoritma *Zatvarač*.

Petlja *redukcija_ključa* obezbeđuje da se za neku šemu $(R_i, \mathcal{K}_i) \in S_m \cup S_n$ i neki ključ $K_i \in \mathcal{K}_i$ ispita da li važi $(\exists X \subset K_i)(X \neq \emptyset \wedge X \subseteq (K_i \setminus X)^+_{\Gamma_{mn}})$ i, ako je uslov ispunjen, utvrđuje jedan od maksimalanih skupova X s navedenom osobinom. Iteracija *testiraj_ključ* omogućava da se, ukoliko postoji, pronađe jedan ključ $K_i \in \mathcal{K}_i$, takav da važi prethodno navedeno svojstvo, dok se petljom *ispitaj_ključeve* obezbeđuje testiranje svih šema relacija $(R_i, \mathcal{K}_i) \in S_m \cup S_n$. Saglasno definiciji 4.7, vrednost promenljive *Ind* $\in$ {*true*, *false*} odgovara logičkoj vrednosti interpretacije predikata $FnfC(S_m, S_n)$.

ALGORITAM 4.2. REDUKCIJA SKUPA OBELEŽJA

```
PROCES Redukcija_skupa (U(Att, Γ), UI(), I(A))
(* U: Att, Γ          -  Skup obeležja i skup fd
   I: A               -  Obeležje za koje važi A ∈ (Att \ {A})+Γ, ili specijalna oznaka '*'
*)
POČETAK PROCESA Redukcija_skupa
   POSTAVI A ← *
   RADI test_skupa_obeležja (∀B ∈ Attr) I DOK JE (A = *)
      POZOVI Zatvarač(Att \ {B}, Γ, Cl_AttB)
      AKO JE B ∈ Cl_AttB TADA
         POSTAVI A ← B
      KRAJ AKO
   KRAJ RADI test_skupa_obeležja
KRAJ PROCESA Redukcija_skupa
```

Slika 4.7.

ALGORITAM 4.3. TEST KOLIZIJE FUNKCIONALNI-NEFUNKCIONALNI ODNOS

```
PROCES Test_fnf_kolizije (U(Sm, Γm, Sn, Γn)), UI(), I(Ind, FnfC_Rep))
(* U: Sm, Sn, Γm, Γn        -  Skupovi šema relacija i fd šema modula (izraz (2.27), tj. (2.17))
   I: Ind ∈ {true, false}   -  Indikator uspešnosti testa FnfC(Sm, Sn)
      FnfC_Rep              -  Izveštaj s greškama koje ukazuju na kolizije, za Ind = false.
*)
POČETAK PROCESA Test_fnf_kolizije
```

POSTAVI $\Gamma_{mn} \leftarrow \Gamma_m \cup \Gamma_n$
POSTAVI *Ind* $\leftarrow$ *true* (∗ Inicijalno uspešan test ∗)
POSTAVI *FnfC_Rep* $\leftarrow \emptyset$
RADI *ispitaj_ključeve* ***ZA*** $(\forall(R_i, \mathcal{K}_i) \in S_m \cup S_n)$
POSTAVI *Ind_pom* $\leftarrow$ *false*
RADI *testiraj_ključ* ***ZA*** $(\forall K_i \in \mathcal{K}_i)$ ***I DOK JE*** $\neg$*Ind_pom*
POSTAVI $RedK_i \leftarrow K_i$
POSTAVI $X \leftarrow \emptyset$
RADI *redukcija_ključa*
POZOVI *Redukcija_skupa*$(RedK_i, \Gamma_{mn}, A)$)
AKO JE $A \neq *$ ***TADA***
POSTAVI $RedK_i \leftarrow RedK_i \setminus \{A\}$
POSTAVI $X \leftarrow X \cup \{A\}$
KRAJ AKO
KRAJ RADI *redukcija_ključa* ***KAD BUDE*** $(A = *)$
AKO JE $X \neq \emptyset$ ***TADA***
POSTAVI *Ind* $\leftarrow$ *false* (∗ Neuspešan test ∗)
POSTAVI *Ind_pom* $\leftarrow$ *true*
POSTAVI *FnfC_Rep* $\leftarrow$ *FnfC_Rep* $\cup$
{(‘*Greška*’, $(R_i, \mathcal{K}_i)$, $RedK_i \rightarrow X$, ‘*Funkcionalni odnos u okviru ključa*’)}
KRAJ AKO
KRAJ RADI *testiraj_ključ*
KRAJ RADI *ispitaj_ključeve*
KRAJ PROCESA *Test_fnf_kolizije*

Slika 4.8.

(*Složenost*) Pod pretpostavkom (koja je u ovom slučaju ispunjena) da je $|\Gamma| >> |Att|$, složenost algoritma *Redukcija_skupa* se svodi na složenost algoritma *Zatvarač* i iznosi $O(m^2)$, pri čemu je $m = |\Gamma|$.

Petlja *redukcija_ključa* algoritma 4.3. se izvršava, u najgorem slučaju, $|K_i| - 1$ puta, a pošto je $|K_i| << |S_m \cup S_n|$, može se pretpostaviti da je njena složenost $O(1) \cdot O(m^2) = O(m^2)$. Složenost petlje *testiraj_ključ* je, takođe, $O(1) \cdot O(m^2) = O(m^2)$, jer je $|\mathcal{K}_i| << |S_m \cup S_n|$, dok je složenost algoritma 4.3. određena složenošću petlje *ispitaj_ključeve*, koja se, u najgorem slučaju, izvršava $O(n) \cdot O(n^2) = O(n^3)$, gde je $n = max(|\Gamma|, |S_m \cup S_n|)$. □

Na slici 4.9. je prikazan algoritam testa kolizije dela ključa $IntCDk(S_m, S_n)$, na slci 4.11. je dat algoritam testa lokalne kolizije skupova međurelacionih ograničenja $IntCLi(S_m, S_n)$, koji poziva pomoćni proces ‘*Test_skupa_id*’, prikazan na slici 4.10, dok se na slici 4.12. nalazi algoritam testa globalne kolizije skupova međurelacionih ograničenja $IntCGr(S_m, S_n)$. Algoritam testa $IntCGr(S_m, S_n)$ poziva pomoćni proces testa direktnog zatvaranja dve šeme relacije ‘*Dir_zatvaranje*’, prikazan na slici 4.11.

ALGORITAM 4.4. TEST KOLIZIJE DELA KLJUČA

PROCES *Test_IntCDk* (**U**$(S_m, \Gamma_m, S_n, \Gamma_n)$), **UI**(), **I**(*Ind*, *IntCDk_Rep*))
(∗ **U:** $S_m, S_n, \Gamma_m, \Gamma_n$ - Skupovi šema relacija i *fd* šema modula (izraz (2.27), tj. (2.17))
I: $Ind \in \{true, false\}$ - Indikator uspešnosti testa $IntCDk(S_m, S_n)$
IntCDk_Rep - Izveštaj s upozorenjima koja ukazuju na kolizije, za *Ind* = *false*. ∗)
POČETAK PROCESA *Test_IntCDk*

POSTAVI $\Gamma_{mn} \leftarrow \Gamma_m \cup \Gamma_n$
POSTAVI *Ind* $\leftarrow$ *true* (∗ Inicijalno uspešan test ∗)
POSTAVI *IntCDk_Rep* $\leftarrow \emptyset$
RADI *izračunaj_zatvarače* **ZA** $(\forall(R_i, \mathcal{K}_i) \in S_m \cup S_n)$
POZOVI *Zatvarač*$(R_i, \Gamma_{mn}, Cl\_R_i)$
KRAJ RADI *izračunaj_zatvarače*
RADI *prolaz_kroz_m* **ZA** $(\forall(R_i, \mathcal{K}_i) \in S_m)$
RADI *prolaz_kroz_n* **ZA** $(\forall(R_j, \mathcal{K}_j) \in S_n)$
AKO JE $(R_i \subseteq Cl\_R_j) \wedge (R_j \subseteq Cl\_R_i)$ **TADA** (∗ $(R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}}$ ∗)
(∗ Blok 1: Test kriterijuma $(\exists K_i \in \mathcal{K}_i \setminus \mathcal{K}_j)(K_i \cap (R_j)^+_{\Gamma_{mn} \setminus \Gamma(R_i)} \neq \emptyset)$ ∗)
POSTAVI *Ind_pom* $\leftarrow$ *false*
POZOVI *Zatvarač*$(R_j, \Gamma_{mn} \setminus \Gamma(R_i), ClS\_R_j)$
RADI *testiraj_ključ* **ZA** $(\forall K_i \in \mathcal{K}_i \setminus \mathcal{K}_j)$ **I DOK JE** $\neg$*Ind_pom*
AKO JE $K_i \cap ClS\_R_j \neq \emptyset$ **TADA**
POSTAVI *Ind* $\leftarrow$ *false* (∗ Neuspešan test ∗)
POSTAVI *Ind_pom* $\leftarrow$ *true*
POSTAVI *IntCDk_Rep* $\leftarrow$*IntCDk_Rep* $\cup$ {(‘*Upozorenje*’, $(R_i, \mathcal{K}_i)$, $(R_j, \mathcal{K}_j)$,
$R_i \leftrightarrow R_j$, $K_i \cap (R_j)^+_{\Gamma_{mn} \setminus \Gamma(R_i)} \neq \emptyset$, ‘*Deo ključa u šemi relacije:*’, R_j)}
KRAJ AKO
KRAJ RADI *testiraj_ključ*
(∗Blok 2: Test kriterijuma $(\exists K_j \in \mathcal{K}_j \setminus \mathcal{K}_i)(K_j \cap (R_i)^+_{\Gamma_{mn} \setminus \Gamma(R_j)} \neq \emptyset)$ ∗)
POSTAVI *Ind_pom* $\leftarrow$ *false*
POZOVI *Zatvarač*$(R_i, \Gamma_{mn} \setminus \Gamma(R_j), ClS\_R_i)$
RADI *testiraj_ključ* **ZA** $(\forall K_j \in \mathcal{K}_j \setminus \mathcal{K}_i)$ **I DOK JE** $\neg$*Ind_pom*
AKO JE $K_j \cap ClS\_R_i \neq \emptyset$ **TADA**
POSTAVI *Ind* $\leftarrow$ *false* (∗ Neuspešan test ∗)
POSTAVI *Ind_pom* $\leftarrow$ *true*
POSTAVI *IntCDk_Rep* $\leftarrow$*IntCDk_Rep* $\cup$ {(‘*Upozorenje*’, $(R_i, \mathcal{K}_i)$, $(R_j, \mathcal{K}_j)$,
$R_i \leftrightarrow R_j$, $K_j \cap (R_i)^+_{\Gamma_{mn} \setminus \Gamma(R_j)} \neq \emptyset$, ‘*Deo ključa u šemi relacije:*’, R_i)}
KRAJ AKO
KRAJ RADI *testiraj_ključ*
KRAJ AKO (∗ $(R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}}$ ∗)
KRAJ RADI *prolaz_kroz_n*
KRAJ RADI *prolaz_kroz_m*
KRAJ PROCESA *Test_IntCDk*

Slika 4.9.

ALGORITAM 4.5. TEST SKUPA ZAVISNOSTI SADRŽAVANJA

PROCES *Test_skupa_id*(**U**(S_y), **UI**$(IC_x, IC_y$, *Ind*, *IntCLi_Rep*), **I**())
POČETAK PROCESA *Test_skupa_id*
RADI *međurelaciona_ograničenja* **ZA** $(\forall ic_x \in IC_x)$
POSTAVI *Compat* $\leftarrow$ *true* (∗ Inicijalno, postoji kompatibilno ograničenje $ic_y \in IC_y$ ∗)

```
      POSTAVI Si_y ← ∅
      RADI kompatibilne_šr ZA (∀(R_x , 𝒦_x) ∈ Si_x )(obuhvata(ic_x , Si_x)) I DOK JE Compat
         POSTAVI Nađen ← false
         RADI traženje_kompat_šr (∀(R_y , 𝒦_y) ∈ S_y) I DOK JE ¬Nađen
            AKO JE 𝒦_x ∩ 𝒦_y ≠ ∅ TADA
               POSTAVI Nađen ← true
               POSTAVI Si_y ← Si_y ∪ {(R_y , 𝒦_y)}
            KRAJ AKO
            AKO JE ¬Nađen TADA
               POSTAVI Compat ← false
            KRAJ AKO
         KRAJ RADI traženje_kompat_šr
      KRAJ RADI kompatibilne_šr
      AKO JE Compat TADA            (∗ Pronađen je skup kompatibilnih šema relacija S_y ∗)
         POSTAVI Nađen ← false
         RADI kompatibilan_ic ZA (∀ic_y ∈ IC_y) I DOK JE ¬Nađen
            AKO JE obuhvata(ic_y , Si_y) TADA
               POSTAVI Nađen ← true
            KRAJ AKO
         KRAJ RADI kompatibilan_ic
         AKO JE ¬Nađen TADA
            POSTAVI Ind ← false                    (∗ Neuspešan test ∗)
            POSTAVI IntCLi_Rep ← IntCLi_Rep ∪ {(‘Upozorenje’, ic_x , Si_x ,
                           ‘Ne postoji ekvivalentno ograničenje u modulu:’, y)}
         INAČE
            POSTAVI IC_x ← IC_x \ {ic_x}
            POSTAVI IC_y ← IC_y \ {ic_y}
         KRAJ AKO
      KRAJ AKO                  (∗ Compat ∗)
   KRAJ RADI međurelaciona_ograničenja
KRAJ PROCESA Test_skupa_id
```

Slika 4.10.

Teorema 4.6. Algoritam 4.4. je korektan, polinomijalne složenosti ($O(n^4)$) u najlošijem slučaju.

Dokaz. (*Korektnost*) Kompozicija petlji *prolaz_kroz_m* i *prolaz_kroz_n* obezbeđuje da se ispitaju svi parovi šema relacija $(R_i , \mathcal{K}_i) \in S_m$ i $(R_j , \mathcal{K}_j) \in S_n$ i utvrdi da li postoji takav par šema relacija za koji je zadovoljen kriterijum $(\exists(R_i , \mathcal{K}_i) \in S_m)(\exists(R_j , \mathcal{K}_j) \in S_n)((R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}} \wedge Int(R_i , \mathcal{K}_i , R_j , \mathcal{K}_j , \Gamma(S_m), \Gamma(S_n)))$. Deo programa, označen komentarom ‘Blok 1’ obezbeđuje test uslova $(\exists K_i \in \mathcal{K}_i \setminus \mathcal{K}_j)(K_i \cap (R_j)^+_{\Gamma_{mn} \setminus \Gamma(R_i)} \neq \emptyset)$, dok deo, označen komentarom ‘Blok 2’ obezbeđuje simetričan test $(\exists K_j \in \mathcal{K}_j \setminus \mathcal{K}_i)(K_j \cap (R_i)^+_{\Gamma_{mn} \setminus \Gamma(R_j)} \neq \emptyset)$. Saglasno definiciji 4.9, vrednost promenljive $Ind \in \{true, false\}$ odgovara logičkoj vrednosti interpretacije predikata $IntCDk(S_m , S_n)$.

(*Složenost*) Složenost petlje *izračunaj_zatvarače* je $O(n^2) \cdot O(n) = O(n^3)$, gde je $n = max(|S_m \cup S_n|, |\Gamma_{mn}|)$. Ako je $m = max(|S_m|, |S_n|)$ $(m < n)$, tada je složenost kompozicije petlji *prolaz_kroz_m* i *prolaz_kroz_n* $O(m^2) \cdot O(n^2)$, jer je složenost petlje *testiraj_ključ* $O(1)$. To znači da je složenost celog algoritma u najlošijem slučaju data kao $O(n^3) + O(m^2) \cdot O(n^2) = O(n^4)$. □

ALGORITAM 4.6. TEST LOKALNE KOLIZIJE MEĐURELACIONIH OGRANIČENJA

PROCES *Test_IntCLi*(**U**(S_m, ICl_m, S_n, ICl_n, *obuhvata*), **UI**(), **I**(*Ind*, *IntCLi_Rep*))
(∗ **U:** S_m, S_n - Skupovi šema relacija modula (datih izrazom (2.27))
$ICl_m \subseteq I_m$, $ICl_n \subseteq I_n$ - Skupovi međurelacionih ograničenja:
$ICl_m = ID(\mathcal{G}_m, \mathcal{SFu}_m) \cup ID(\mathcal{G}_m, desig) \cup DR(\mathcal{G}_m, dep)$
$ICl_n = ID(\mathcal{G}_n, \mathcal{SFu}_n) \cup ID(\mathcal{G}_n, desig) \cup DR(\mathcal{G}_n, dep)$
obuhvata(*ic*, *Si*) - Predikat: ‘ograničenje *ic* obuhvata skup šema relacija *Si*’
I: $Ind \in \{true, false\}$ - Indikator uspešnosti testa $IntcLi(S_m, S_n)$
IntCLi_Rep - Izveštaj s upozorenjima koja ukazuju na kolizije, za *Ind* = *false*.
∗)
POČETAK PROCESA *Test_IntCLi*
POSTAVI *Ind* ← *true* (∗ Inicijalno uspešan test ∗)
POSTAVI *IntCLi_Rep* ← ∅
POZOVI *Test_skupa_id*(S_n, $ID(\mathcal{G}_m, \mathcal{SFu}_m)$, $ID(\mathcal{G}_n, \mathcal{SFu}_n)$, *Ind*, *IntCLi_Rep*)
POZOVI *Test_skupa_id*(S_m, $ID(\mathcal{G}_n, \mathcal{SFu}_n)$, $ID(\mathcal{G}_m, \mathcal{SFu}_m)$, *Ind*, *IntCLi_Rep*)
POZOVI *Test_skupa_id*(S_n, $ID(\mathcal{G}_m, desig)$, $ID(\mathcal{G}_n, desig)$, *Ind*, *IntCLi_Rep*)
POZOVI *Test_skupa_id*(S_m, $ID(\mathcal{G}_n, desig)$, $ID(\mathcal{G}_m, desig)$, *Ind*, *IntCLi_Rep*)
POZOVI *Test_skupa_id*(S_n, $DR(\mathcal{G}_m, dep)$, $DR(\mathcal{G}_n, dep)$, *Ind*, *IntCLi_Rep*)
POZOVI *Test_skupa_id*(S_m, $DR(\mathcal{G}_n, dep)$, $DR(\mathcal{G}_m, dep)$, *Ind*, *IntCLi_Rep*)
KRAJ PROCESA *Test_IntCLi*

Slika 4.11.

Teorema 4.7. Algoritam 4.6. je korektan, polinomijalne složenosti ($O(n^3)$) u najlošijem slučaju.

Dokaz. (*Korektnost*) Sekvenca poziva procedure *Test_skupa_id* (algoritam 4.5) u okviru procedure *Test_IntCLi* obezbeđuje da se ispitaju sva međurelaciona ograničenja skupova ICl_m i ICl_n. Kompozicija petlji *međurelaciona_ograničenja*, *kompatibilne_šr* i *traženje_kompat_šr*, procedure *Test_skupa_id*, omogućava da se za svako međurelciono ograničenje $ic_x \in IC_x$ $(x \in \{m, n\})$ i skup Si_x, takav da $obuhvata(ic_x, Si_x)$, ispita da li:

$$(\exists Si_y \subseteq S_y)((|Si_x| = |Si_y| = k \wedge (\forall i \in \{1,...,k\})(\mathcal{K}_x \cap \mathcal{K}_y \neq \emptyset)) \Rightarrow (\forall ic_y \in IC_y)(\neg obuhvata(ic_y, Si_y))),$$

pri čemu je $x, y \in \{m, n\}$. Navedeni uslov znači egzistenciju kolizije $IntCLi(S_m, S_n)$. Saglasno definiciji 4.9, vrednost promenljive $Ind \in \{true, false\}$ odgovara logičkoj vrednosti interpretacije predikata $IntcLi(S_m, S_n)$.

(*Složenost*) Složenost algoritma 4.6. je određena složenošću procedure *Test_skupa_id*. Složenost petlje *kompatibilan_ic* se može proceniti kao $O(m)$, gde je $m = max(|IC_x|, |IC_y|)$, tako da se složenost petlje *traženje_kompat_šr* i *kompatibilne_šr* (jer je $|Si_x| << |S_y|$) ocenjuje kao $O(m) \cdot O(n) = O(n^2)$, gde je $n = |S_y|$, $m < n$. Složenost petlje *međurelaciona_ograničenja*, tj. celokupnog algoritma, u najlošijem slučaju je $O(m) \cdot O(n^2) = O(n^3)$. □

Što se tiče procene složenosti algoritma 4.6, treba napomenuti da ukoliko se pretpostavi da važi $m^2 << n$ (što je u praksi, gotovo uvek, slučaj), tada je njegova složenost u najlošijem slučaju linearna $O(n)$.

ALGORITAM 4.7. TEST DIREKTNOG ZATVARANJA ŠEMA RELACIJA

PROCES *Dir_zatvaranje* (**U**((R_x, $\mathcal{K}_x$, S_x), (R_y, $\mathcal{K}_y$, $\mathcal{G}_y$, CL_y), CL_{mn}), **UI**(*Ind*, *IntCGr_Rep*), **I**(*Tranzitivan*))

(∗ Algoritam ispituje uslov $R_x \subset (R_y)^+_{\Gamma_{mn}}$ i testira da li za ograničenje $ic \in RI \cup BR$, za koje je
ispunjeno *obuhvata*(ic, {R_x, R_y}), važi $I_m \cup I_n \models ic$.
Značenje ulaznih i ulazno-izlaznih parametara je opisano u okviru algoritma 4.8.
I: *Tranzitivan* ∈ {*true*, *false*} - Indikacija tranzitivnog/direktnog zatvaranja $R_x \subset (R_y)^+_{\Gamma_{mn}}$
Contin ∈ {*true*, *false*} - Indikacija potrebe nastavka testa direktnog zatvaranja
∗)

POČETAK PROCESA *Dir_zatvaranje*
  POSTAVI *Contin* ← *true* (∗ Inicijalno potreban nastavak testa ∗)
  AKO JE $(R_x \subset Cl_{mn}\_R_y) \wedge (R_y \not\subseteq Cl_{mn}\_R_x)$ ***TADA***
    POSTAVI *Tranzitivan* ← *false* (∗ Inicijalno direktno zatvaranje ∗)
    POSTAVI $S_{xy} \leftarrow (S_x \cup S_y) \setminus \{(R_x, \mathcal{K}_x), (R_y, \mathcal{K}_y)\}$
    POSTAVI *SEqX* ← ∅
    POSTAVI *SEqY* ← ∅
    RADI *ispitaj_tranzitivnost* ***ZA*** $(\forall (R_k, \mathcal{K}_k) \in S_{xy})$ ***I DOK JE*** ¬*Tranzitivan*
      AKO JE $(R_x \subset Cl_{mn}\_R_k) \wedge (R_k \subset Cl_{mn}\_R_y)$ ***TADA***
        POSTAVI *Tranzitivan* ← *true* (∗ Tranzitivno zatvaranje ∗)
      INAČE AKO JE $(R_x \subseteq Cl_{mn}\_R_k) \wedge (R_k \subseteq Cl_{mn}\_R_x) \wedge (R_k \subset Cl_{mn}\_R_y)$ ***TADA***
        POSTAVI $SEqY \leftarrow SEqY \cup \{(R_k, \mathcal{K}_k)\}$ (∗ $(R_k, \mathcal{K}_k) \in S_y$ ∗)
      INAČE AKO JE $(R_x \subset Cl_{mn}\_R_k) \wedge (R_k \subseteq Cl_{mn}\_R_y) \wedge (R_y \subseteq Cl_{mn}\_R_k)$ ***TADA***
        POSTAVI $SEqX \leftarrow SEqX \cup \{(R_k, \mathcal{K}_k)\}$ (∗ $(R_k, \mathcal{K}_k) \in S_x$ ∗)
      KRAJ AKO
    KRAJ RADI *ispitaj_tranzitivnost*
    AKO JE ¬*Tranzitivan* ***TADA***
      POSTAVI $RefAttr \leftarrow K_p(R_x) \cap Cl_y\_R_y$
      RADI *ispitaj_ekvivalentnost_y* ***ZA*** $(\forall (R_k, \mathcal{K}_k) \in SEqY)$ ***I DOK JE*** $(RefAttr \neq \emptyset)$
        (∗ $R_x \subset (R_y)^+_{\Gamma_{mn}} \wedge R_y \not\subseteq (R_x)^+_{\Gamma_{mn}} \wedge (R_x)^+_{\Gamma_{mn}} = (R_k)^+_{\Gamma_{mn}}$ ∗)
        POSTAVI $RefAttr \leftarrow RefAttr \setminus Cl_y\_R_k$
      KRAJ RADI *ispitaj_ekvivalentnost_y*
      AKO JE $RefAttr = \emptyset$ ***TADA***
        RADI *ispitaj_ekvivalentnost_x* ***ZA*** $(\forall (R_k, \mathcal{K}_k) \in SEqX)$ ***I DOK JE*** $(RefAttr \neq \emptyset)$
          (∗ $R_x \subset (R_y)^+_{\Gamma_{mn}} \wedge R_y \not\subseteq (R_x)^+_{\Gamma_{mn}} \wedge (R_y)^+_{\Gamma_{mn}} = (R_k)^+_{\Gamma_{mn}}$ ∗)
          POSTAVI $RefAttr \leftarrow RefAttr \setminus Cl_x\_R_k$
        KRAJ RADI *ispitaj_ekvivalentnost_x*
        AKO JE $RefAttr \neq \emptyset$ ***TADA***
          POSTAVI *Contin* ← *false* (∗ Nepotreban nastavak testa ∗)
          POSTAVI $Pkey \leftarrow R_x \cap Cl_{mn}\_R_y$

POSTAVI *IntCGr_Rep* ← *IntCGr_Rep* ∪ {(‘*Poruka*’,
ic: $(R_y, \mathcal{K}_y)[Pkey] \subseteq (R_x, \mathcal{K}_x)[Pkey]$, ‘*Moguće nepokriveno međurelaciono ograničenje*’)}
INAČE (∗ Detektovana moguća kolizija ∗)
KRAJ AKO (∗ $RefAttr \neq \varnothing$ ∗)
KRAJ AKO (∗ $RefAttr = \varnothing$ ∗)
KRAJ AKO (∗ ¬*Tranzitivan* ∗)
AKO JE ¬*Tranzitivan* ∨ $RefAttr \neq \varnothing$ **TADA**
POSTAVI *Contin* ← *false* (∗ Nepotreban nastavak testa ∗)
POSTAVI *Ind* ← *false* (∗ Neuspešan test ∗)
POSTAVI *Pkey* ← $R_x \cap Cl_{mn}\_R_y$
POSTAVI *IntCGr_Rep* ← *IntCGr_Rep* ∪ {(‘*Upozorenje*’,
ic: $(R_y, \mathcal{K}_y)[Pkey] \subseteq (R_x, \mathcal{K}_x)[Pkey]$, ‘*Nepokriveno međurelaciono ograničenje*’)}
KRAJ AKO
KRAJ AKO (∗ $(R_x \subset Cl_{mn}\_R_y) \wedge (R_y \not\subseteq Cl_{mn}\_R_x)$ ∗)
KRAJ PROCESA *Dir_zatvaranje*

Slika 4.12.

Teorema 4.8. Algoritam 4.8. je korektan, polinomijalne složenosti ($O(n^3)$) u najlošijem slučaju.

Dokaz. (*Korektnost*) Kompozicija petlji *prolaz_kroz_m* i *prolaz_kroz_n* obezbeđuje da se za svaki par šema relacija $(R_i, \mathcal{K}_i)$, $\in S_m$ i $(R_j, \mathcal{K}_j) \in S_n$, za koji je $\mathcal{K}_i \cap \mathcal{K}_j = \varnothing$, ispita važenje uslova (iz definicije 4.9) $RICls(R_j, R_i, \Gamma_{mn}, I_{mn}) \vee RICls(R_i, R_j, \Gamma_{mn}, I_{mn}) \vee RIRen(R_i, R_j, I_{mn}) \vee BRCl(R_i, R_j, I_{mn}) \vee BRCl(R_j, R_i, I_{mn}) \vee BRDer(R_i, R_j, I_{mn})$. Važenje predikata $RICls(R_i, R_j, \Gamma_{mn}, I_{mn})$ i $BRCl(R_i, R_j, I_{mn})$ se ispituje prvim pozivom procedure *Dir_zatvaranje*, dok se simetrični slučajevi $RICls(R_j, R_i, \Gamma_{mn}, I_{mn})$ i $BRCl(R_j, R_i, I_{mn})$ ispituju drugim pozivom procedure *Dir_zatvaranje*.

Procedura *Dir_zatvaranje* ispituje za svaki par šema relacija $(R_x, \mathcal{K}_x)$ i $(R_y, \mathcal{K}_y)$ iz različitih modula, uslov $R_x \subset (R_y)^+_{\Gamma_{mn}} \wedge R_y \not\subseteq (R_x)^+_{\Gamma_{mn}}$, što može biti indikacija za egzistenciju ograničenja $ic \in RI \cup BR$, takvog da važi $I_{mn} \not\models ic$ i *obuhvata*$(ic, \{R_x, R_y\})$. Petlja *ispitaj_tranzitivnost* omogućava da se utvrdi je li *ic* strogo tranzitivno međurelaciono ograničenje, pošto strogo tranzitivna međurelaciona ograničenja ne moraju obavezno izazvati koliziju. Pored toga, izveštaj *IntCGr_Rep* ne treba opterećivati kolizijama tranzitivnih ograničenja. Petlje *ispitaj_ekvivalentnost_y* i *ispitaj_ekvivalentnost_x* obezbeđuju da se za međurelaciono ograničenje grafa zatvaranja $ic \in I$ utvrdi da li je pokriveno odgovarajućim međurelacionim ograničenjem skupa $I_m \cup I_n$.

Važenje predikata $RIRen(R_i, R_j, I_{mn})$ se ispituje putem testova: $(\exists K_j \in \mathcal{K}_j)(\exists X_j \in R_i)(R_i[X_j] \subseteq R_j[K_j] \in RI_r(ren, dbs) \setminus I_{mn})$ i $(\exists K_i \in \mathcal{K}_i)(\exists X_i \in R_j)(R_j[X_i] \subseteq R_i[K_i] \in RI_r(ren, dbs) \setminus I_{mn})$. Ukoliko se, primenom algoritma, zaključi da neko od ograničenja $R_i[X_j] \subseteq R_j[K_j]$, ili $R_j[X_i] \subseteq R_i[K_i]$ pripada skupu $RI_r(ren, dbs) \setminus I_{mn}$, tada, saglasno definiciji funkcije *ren*, ono sigurno ne može biti ni posledica skupa I_{mn}.

Važenje predikata $BRDer(R_i, R_j, I_{mn})$ se ispituje putem testa $(\exists br \in BRd)(obuhvata(br, \{R_i, R_j\}))$. Ukoliko se zaključi da postoji pravilo poslovanja $br \in BR(der, dbs) \setminus I_{mn}$, tada, saglasno definiciji funkcije *der*, ono ne može biti ni posledica skupa I_{mn}.

ALGORITAM 4.8. TEST GLOBALNE KOLIZIJE MEĐURELACIONIH OGRANIČENJA

PROCES *Test_IntCGr* (**U**(S_m, I_m, Γ_m, $\mathcal{G}_m$, S_n, I_n, Γ_n, $\mathcal{G}_n$, *obuhvata*), **UI**(), **I**(*Ind*, *IntCGr_Rep*))

(∗ **U:** S_m, S_n - Skupovi šema relacija modula (datih izrazom (2.27))
I_m, I_n - Skupovi međurelacionih ograničenja šema modula (izraz (2.37))
Γ_m, Γ_n - Skupovi *fd* šema modula (S_m, I_m) i (S_n, I_n) (izraz (2.17))
$\mathcal{G}_m = (S_m, \rho_m)$ - Graf zatvaranja šeme modula (S_m, I_m) (izraz (2.31))
$\mathcal{G}_n = (S_n, \rho_n)$ - Graf zatvaranja šeme modula (S_n, I_n) (izraz (2.31))
obuhvata(*ic*, *Si*) - Predikat: 'ograničenje *ic* obuhvata skup šema relacija *Si*'
I: *Ind* ∈ {*true*, *false*} - Indikator uspešnosti testa *IntcGr*(S_m, S_n)
IntCGr_Rep - Izveštaj upozorenjima koja ukazuju na kolizije, za *Ind* = *false* i porukama koje ukazuju na mogućnost nastanka globalne kolizije.
∗)

POČETAK PROCESA *Test_IntCGr*
POSTAVI *Ind* ← *true* (∗ Inicijalno uspešan test ∗)
POSTAVI *IntCGr_Rep* ← ∅
POSTAVI $\Gamma_{mn} \leftarrow \Gamma_m \cup \Gamma_n$
POSTAVI $I_{mn} \leftarrow I_m \cup I_n$
POSTAVI $CL_{mn} \leftarrow \emptyset$ (∗ Skup zatvarača svih šema relacija s obzirom na Γ_{mn} ∗)
POSTAVI $CL_m \leftarrow \emptyset$ (∗ Skup zatvarača šema relacija modula *m* s obzirom na Γ_m ∗)
RADI *izračunaj_zatvarače_m* ***ZA*** $(\forall(R_i, \mathcal{K}_i) \in S_m)$
POZOVI *Zatvarač*(R_i, Γ_{mn}, Cl_{mn}_R_i)
POSTAVI $CL_{mn} \leftarrow CL_{mn} \cup \{Cl_{mn}\_R_i\}$
POZOVI *Zatvarač*(R_i, Γ_m, Cl_m_R_i)
POSTAVI $CL_m \leftarrow CL_m \cup \{Cl_m\_R_i\}$
KRAJ RADI *izračunaj_zatvarače_m*
POSTAVI $CL_n \leftarrow \emptyset$ (∗ Skup zatvarača šema relacija modula *n* s obzirom na Γ_n ∗)
RADI *izračunaj_zatvarače_n* ***ZA*** $(\forall(R_i, \mathcal{K}_i) \in S_n)$
POZOVI *Zatvarač*(R_i, Γ_{mn}, Cl_{mn}_R_i)
POSTAVI $CL_{mn} \leftarrow CL_{mn} \cup \{Cl_{mn}\_R_i\}$
POZOVI *Zatvarač*(R_i, Γ_n, Cl_n_R_i)
POSTAVI $CL_n \leftarrow CL_n \cup \{Cl_n\_R_i\}$
KRAJ RADI *izračunaj_zatvarače_n*
RADI *prolaz_kroz_m* ***ZA*** $(\forall(R_i, \mathcal{K}_i) \in S_m)$
RADI *prolaz_kroz_n* ***ZA*** $(\forall(R_j, \mathcal{K}_j) \in S_n)$
AKO JE $\mathcal{K}_i \cap \mathcal{K}_j = \emptyset$ ***TADA***
POZOVI *Dir_zatvaranje* ((R_i, $\mathcal{K}_i$, S_m), (R_j, $\mathcal{K}_j$, $\mathcal{G}_n$, CL_n), CL_{mn}, *Ind*, *IntCGr_Rep*, *Contin*)
AKO JE *Contin* ***TADA*** (∗ Neophodan nastavak testa globalne kolizije ∗)
POZOVI *Dir_zatvaranje* ((R_j, $\mathcal{K}_j$, S_n), (R_i, $\mathcal{K}_i$, $\mathcal{G}_m$, CL_m), CL_{mn}, *Ind*, *IntCGr_Rep*, *Contin*)
AKO JE *Contin* ***TADA***
POSTAVI $RId_r \leftarrow RI_r(ren, dbs) \setminus I_{mn}$ (∗ Generisanje skupa RId_r ∗)
AKO JE $(\exists K_j \in \mathcal{K}_j)(\exists X_j \in R_i)(R_i[X_j] \subseteq R_j[K_j] \in RId_r)$ ***TADA***
POSTAVI *Ind* ← *false* (∗ Neuspešan test ∗)
POSTAVI *IntCGr_Rep* ← *IntCGr_Rep* ∪ {('*Upozorenje*', *ri*: $(R_i, \mathcal{K}_i)[X_j] \subseteq (R_j, \mathcal{K}_j)[K_j]$, '*Nepokriveni referencijalni integritet preimenovanja*')}

INAČE AKO JE $(\exists K_i \in \mathcal{K}_i)(\exists X_i \in R_j)(R_j[X_i] \subseteq R_i[K_i] \in RId_r)$ **TADA**
POSTAVI *Ind* ← *false* (∗ Neuspešan test ∗)
POSTAVI *IntCGr_Rep* ← *IntCGr_Rep* ∪ {(‘*Upozorenje*’,
ri: $(R_j, \mathcal{K}_j)[X_i] \subseteq (R_i, \mathcal{K}_i)[K_i]$, ‘*Nepokriveni referencijalni integritet preimenovanja*’)}
INAČE
POSTAVI $BR_d \leftarrow BR(der, dbs) \setminus I_{mn}$ (∗ Generisanje skupa BR_d ∗)
AKO $(\exists br \in BRd)(obuhvata(br, \{R_i, R_j\}))$**TADA**
POSTAVI *Ind* ← *false* (∗ Neuspešan test ∗)
POSTAVI *IntCGr_Rep* ← *IntCGr_Rep* ∪ {(‘*Upozorenje*’,
br: $\{(R_i, \mathcal{K}_i), (R_j, \mathcal{K}_j)\}$, ‘*Nepokriveno pravilo poslovanja nad:*’, R_i, R_j)}
KRAJ AKO
KRAJ AKO
KRAJ AKO
KRAJ AKO
KRAJ AKO (∗ $\mathcal{K}_i \cap \mathcal{K}_j = \emptyset$ ∗)
KRAJ RADI *prolaz_kroz_n*
KRAJ RADI *prolaz_kroz_m*
KRAJ PROCESA *Test_IntCGr*

Slika 4.13.

Na osnovu svega iznesenog, saglasno definiciji 4.9, vrednost promenljive *Ind* ∈ {*true*, *false*} odgovara logičkoj vrednosti interpretacije predikata $IntcGr(S_m, S_n)$.

(*Složenost*) Složenost algoritma 4.7. *Dir_zatvaranje* je određena složenošću petlje *ispitaj_tranzitivnost* i iznosi $O(n)$, $n = |S_m| + |S_n|$. (Pošto važi da je $|SEqX| << S_x$ i $|SEqY| << S_y$, sledi da se složenost petlji *ispitaj_ekvivalentnost_x* i *ispitaj_ekvivalentnost_y* aproksimira kao $O(1)$.) Složenost petlji *izračunaj_zatvarače_n* i *izračunaj_zatvarače_m* iznosi ukupno $O(1) \cdot O(n) \cdot O(n^2) = O(n^3)$. Složenost uslovnih naredbi kojima se testira $RIRen(R_i, R_j, I_{mn})$, može se proceniti kao $O(1)$, jer je $|\mathcal{K}_i| << n$. Složenost grupe naredbi za testiranje predikata $BRDer(R_i, R_j, I_{mn})$ se, takođe, može proceniti kao $O(1)$, pod pretpostavkom (koja je u praksi ispunjena) da je $BR_d << n$. Složenost kompozicije petlji *prolaz_kroz_m* i *prolaz_kroz_n* je $O(n^2) \cdot O(n) = O(n^3)$, tako da je, u najlošijem slučaju, algoritam 4.8. složenosti $O(n^3)$. □

Na slici 4.14. je prikazan algoritam testa $EqKeyC(S_m, S_n)$ - kolizije ekvivalentnih ključeva.

Teorema 4.9. Algoritam 4.9. je korektan, polinomijalne složenosti ($O(n^3)$) u najlošijem slučaju.

Dokaz. (*Korektnost*) Kompozicija petlji *polaz_kroz_m* i *prolaz_kroz_n* obezbeđuje da se za svaku kombinaciju dve šeme relacije iz skupova S_m i S_n, odgovarajućih šema modula, ispita važenje uslova $\mathcal{K}_i \cap \mathcal{K}_j = \emptyset \wedge Eqk(R_i, \mathcal{K}_i, R_j, \mathcal{K}_j, \Gamma_{mn})$, odnosno eventualne egzistencije kolizije ekvivalentnih ključeva $EqKeyC(S_m, S_n)$.

(*Složenost*) Složenost petlji *izračunaj_zatvarače_m* i *izračunaj_zatvarače_n* je $O(n^2) \cdot O(n) = O(n^3)$, gde je $n = max(|S_m|, |S_n|, |\Gamma_{mn}|)$. Složenost kompozicije petlji *prolaz_kroz_m* i *prolaz_kroz_n* iznosi $O(n^2) \cdot O(1) = O(n^2)$, jer se složenost uslova $(\exists K_j \in \mathcal{K}_j)(K_j \subseteq Cl_m\_R_i)$, odnosno $(\exists K_i \in \mathcal{K}_i)(K_i \subseteq Cl_n\_R_j)$ može proceniti kao $O(1)$, pošto je, u gotovo svim slučajevima, zadovoljeno $|\mathcal{K}_j| << n$, tj. $|\mathcal{K}_i| << n$. To znači da je složenost celog algoritma u najlošijem slučaju data kao $2 \cdot O(n^3) + O(n^2) = O(n^3)$. □

ALGORITAM 4.9. TEST KOLIZIJE EKVIVALENTNIH KLJUČEVA

PROCES *Test_EqKeyC*(**U**(S_m, Γ_m, S_n, Γ_n), **UI**(), **I**(*Ind*, *EqKeyC_Rep*))
(∗ **U:** S_m , S_n , Γ_m , Γ_n - Skupovi šema relacija i *fd* šema modula (izraz (2.27), tj. (2.17))
I: $Ind \in \{true, false\}$ - Indikator uspešnosti testa $EqKeyC(S_m , S_n)$
EqKeyC_Rep - Izveštaj s greškama koje ukazuju na kolizije, za *Ind* = *false*.
∗)
POČETAK PROCESA *Test_EqKeyC*
POSTAVI $Ind \leftarrow true$ (∗ Inicijalno uspešan test ∗)
POSTAVI $EqKeyC\_Rep \leftarrow \varnothing$
RADI *izračunaj_zatvarače_m* ***ZA*** $(\forall(R_i , \mathcal{K}_i) \in S_m)$
POZOVI $Zatvara\check{c}(R_i , \Gamma_m, Cl_m\_R_i)$
POZOVI $Zatvara\check{c}(R_i , \Gamma_{mn}, Cl_{mn}\_R_i)$
KRAJ RADI *izračunaj_zatvarače_m*
RADI *izračunaj_zatvarače_n* ***ZA*** $(\forall(R_i , \mathcal{K}_i) \in S_n)$
POZOVI $Zatvara\check{c}(R_i , \Gamma_n, Cl_n\_R_i)$
POZOVI $Zatvara\check{c}(R_i , \Gamma_{mn}, Cl_{mn}\_R_i)$
KRAJ RADI *izračunaj_zatvarače_n*
RADI *prolaz_kroz_m* ***ZA*** $(\forall(R_i , \mathcal{K}_i) \in S_m)$
RADI *prolaz_kroz_n* ***ZA*** $(\forall(R_j , \mathcal{K}_j) \in S_n)$
AKO JE $(\mathcal{K}_i \cap \mathcal{K}_j = \varnothing) \wedge (R_i \subseteq Cl_{mn}\_R_j) \wedge (R_j \subseteq Cl_{mn}\_R_i)$ ***TADA***
(∗ $\mathcal{K}_i \cap \mathcal{K}_j = \varnothing \wedge (R_i)^+_{\Gamma_{mn}} = (R_j)^+_{\Gamma_{mn}}$ ∗)
AKO JE $(\exists K_j \in \mathcal{K}_j)(K_j \subseteq Cl_m\_R_i)$ ***TADA*** (∗ $(K_j \subseteq (R_i)^+_{\Gamma(S_m)})$ ∗)
POSTAVI $Ind \leftarrow false$ (∗ Neuspešan test ∗)
POSTAVI $EqKeyC\_Rep \leftarrow EqKeyC\_Rep \cup \{($‘*Greška*’, $(R_i , \mathcal{K}_i), (R_j , \mathcal{K}_j),$
$R_i \leftrightarrow R_j$, ‘*Ekvivalentne šeme relacija*’$)\}$
INAČE AKO JE $(\exists K_i \in \mathcal{K}_i)(K_i \subseteq Cl_n\_R_j)$ ***TADA*** (∗ $K_i \subseteq (R_j)^+_{\Gamma(S_n)}$ ∗)
POSTAVI $Ind \leftarrow false$ (∗ Neuspešan test ∗)
POSTAVI $EqKeyC\_Rep \leftarrow EqKeyC\_Rep \cup \{($‘*Greška*’, $(R_i , \mathcal{K}_i), (R_j , \mathcal{K}_j),$
$R_i \leftrightarrow R_j$, ‘*Ekvivalentne šeme relacija*’$)\}$
KRAJ AKO
POSTAVI $S_n \leftarrow S_n \setminus \{(R_j , \mathcal{K}_j)\}$
KRAJ AKO
KRAJ RADI *prolaz_kroz_n*
KRAJ RADI *prolaz_kroz_m*
KRAJ PROCESA *Test_EqKeyC*

Slika 4.14.

Slika 4.15. daje prikaz algoritma za testiranje stroge ili intenzionalne kompatibilnosti dve šeme modula BP. Detaljnija analiza i načini tumačenja tipova grešaka i upozorenja koji se mogu pojaviti u okviru izveštaja *TranC_Rep*, *FnfC_Rep*, *IntCDk_Rep*, *IntCLi_Rep*, *IntCGr_Rep* i *EqKeyC_Rep* su dati u okviru petog poglavlja rada.

Teorema 4.10. Algoritam 4.10. je korektan, polinomijalne složenosti ($O(n^4)$) u najlošijem slučaju.

Dokaz. (*Korektnost*) Sledi na osnovu teorema 4.1. i 4.3-4.9.
(*Složenost*) Sledi na osnovu teorema 4.3-4.9. □

ALGORITAM 4.10. TEST STROGE ILI INTENZIONALNE KOMPATIBILNOSTI

PROCES *Stroga_kompatibilnost*(**U**(S_m, I_m, Γ_m, $\mathcal{G}_m$, S_n, I_n, Γ_n, $\mathcal{G}_n$, *obuhvata*), **UI**(), **I**(*Ind, TranC_Rep, FnfC_Rep, IntCDk_Rep, IntCLi_Rep, IntCGr_Rep, EqKeyC_Rep*))

(∗ **U:** S_m, S_n - Skupovi šema relacija modula (datih izrazom (2.27))
I_m, I_n - Skupovi međurelacionih ograničenja šema modula (izraz (2.37))
Γ_m, Γ_n - Skupovi *fd* šema modula (S_m, I_m) i (S_n, I_n) (izraz (2.17))
$\mathcal{G}_m = (S_m, \rho_m)$ - Graf zatvaranja šeme modula (S_m, I_m) (izraz (2.31))
$\mathcal{G}_n = (S_n, \rho_n)$ - Graf zatvaranja šeme modula (S_n, I_n) (izraz (2.31))
obuhvata(*ic, Si*) - Predikat: ‘ograničenje *ic* obuhvata skup šema relacija *Si*’
$ICl_m \subseteq I_m$, $ICl_n \subseteq I_n$ - Skupovi međurelacionih ograničenja:
$ICl_m = ID(\mathcal{G}_m, \mathcal{SFu}_m) \cup ID(\mathcal{G}_m, desig) \cup DR(\mathcal{G}_m, dep)$
$ICl_n = ID(\mathcal{G}_n, \mathcal{SFu}_n) \cup ID(\mathcal{G}_n, desig) \cup DR(\mathcal{G}_n, dep)$

I: *Ind* ∈ {*Strue, Itrue, false*} - Indikator uspešnosti testa kompatibilnosti:
Strue - strogo kompatibilne šeme modula,
Itrue - intenzionalno kompatibilne šeme modula,
false - nekompatibilne šeme modula
TranC_Rep, FnfC_Rep, IntCDk_Rep, IntCLi_Rep, IntCGr_Rep, EqKeyC_Rep
- Izveštaji s upozorenjima i greškama koji ukazuju na kolizije
∗)

POČETAK PROCESA *Stroga_kompatibilnost*
    POSTAVI *Ind* ← *false* (∗ Inicijalno neuspešan test ∗)
    POZOVI *Test_kolizije_tranzitivnosti* (S_m, Γ_m, S_n, Γ_n, *TranC_Ind, TranC_Rep*)
    POZOVI *Test_fnf_kolizije* (S_m, Γ_m, S_n, Γ_n, *FnfC_Ind, FnfC_Rep*)
    POZOVI *Test_IntCDk* (S_m, Γ_m, S_n, Γ_n, *IntCDk_Ind, IntCDk_Rep*)
    AKO JE *¬IntCDk* ***TADA***
        POZOVI *Test_EqKeyC*(S_m, Γ_m, S_n, Γ_n, *EqKeyC_Ind, EqKeyC_Rep*)
    KRAJ AKO
    POZOVI *Test_IntCLi*(S_m, ICl_m, S_n, ICl_n, *obuhvata, IntCLi_Ind, IntCLi_Rep*)
    POZOVI *Test_IntCGr* (S_m, I_m, Γ_m, $\mathcal{G}_m$, S_n, I_n, Γ_n, $\mathcal{G}_n$, *obuhvata, IntCGr_Ind, IntCGr_Rep*)
    AKO JE *TranC_Ind* ∧ *FnfC_Ind* ***TADA***
        AKO JE *IntCDk_Ind* ∧ *IntCLi_Ind* ∧ *IntCGr_Ind* ***TADA***
            POSTAVI *Ind* ← *Strue* (∗ Strogo kompatibilne šeme modula ∗)
        INAČE AKO JE *IntCDk_Ind* ∨ *EqKeyC_Ind* ***TADA***
            POSTAVI *Ind* ← *Itrue* (∗ Intenzionalno kompatibilne šeme modula ∗)
        KRAJ AKO
    KRAJ AKO
KRAJ PROCESA *Stroga_kompatibilnost*

Slika 4.15.

4.7. Algoritam integracije šema modula

Algoritam integracije strogo kompatibilnih šema modula u strogo pokrivajuću šemu modula je prikazan na slici 4.16.

ALGORITAM 4.11. INTEGRACIJA STROGO KOMPATIBILNIH ŠEMA MODULA

PROCES *Integracija* (**U**($S_m, I_m, \Gamma_m, \mathcal{G}_m, S_n, I_n, \Gamma_n, \mathcal{G}_n$, *obuhvata*), **UI**(), **I**((S, I), *Ind*, *Upz*))
(∗ **I:** (S, I) - Strogo pokrivajuća šema modula za šeme modula (S_m, I_m) i (S_n, I_n)
Upz - Upozorenje, u slučaju narušavanja stroge, tj. intenzionalne kompatibilnosti ∗)

POČETAK PROCESA *Integracija*
POSTAVI $Upz \leftarrow \emptyset$ (∗ Inicijalizacija upozorenja ∗)
POZOVI *Stroga_kompatibilnost*($S_m, I_m, \Gamma_m, \mathcal{G}_m, S_n, I_n, \Gamma_n, \mathcal{G}_n$, *obuhvata*, *Ind*, *TranC_Rep*, *FnfC_Rep*, *IntCDk_Rep*, *IntCLi_Rep*, *IntCGr_Rep*, *EqKeyC_Rep*)
AKO JE $Ind \in \{Itrue, Strue\}$ ***TADA*** (∗ $(S_m, I_m) \cong (S_n, I_n) \vee (S_m, I_m) \cong_I (S_n, I_n)$ ∗)
(∗ Formiranje skupa klasa ekvivalencija $Ekviv(S_m \cup S_n)$ ∗)
POSTAVI $Ekviv(S_m \cup S_n) \leftarrow \emptyset$
POSTAVI $S_{xy} \leftarrow S_m \cup S_n$
RADI *prolaz_kroz_mn* ***ZA*** $(\forall (R_i, \mathcal{K}_i) \in S_{xy})$
POSTAVI $S_{Eq}(R_i) \leftarrow \{(R_i, \mathcal{K}_i)\}$
POSTAVI $TestŠr \leftarrow \{(R_i, \mathcal{K}_i)\}$ (∗ Skup š. rel, za koji se traže ekvivalentne šeme ∗)
POSTAVI $S_m \leftarrow S_m \setminus \{(R_i, \mathcal{K}_i)\}$
POSTAVI $S_{xy} \leftarrow S_{xy} \setminus \{(R_i, \mathcal{K}_i)\}$
RADI *formiranje_klasa_ekv* ***DOK JE*** $TestŠr \neq \emptyset$
POSTAVI $NTestŠr \leftarrow \emptyset$
RADI *ekvivalentne_šr* ***ZA*** $(\forall (R_j, \mathcal{K}_j) \in TestŠr)$
RADI *traženje_ekviv* ***ZA*** $(\forall (R_k, \mathcal{K}_k) \in S_n)$
AKO JE $\mathcal{K}_j \cap \mathcal{K}_k \neq \emptyset$ ***TADA***
POSTAVI $S_{Eq}(R_i) \leftarrow S_{Eq}(R_i) \cup \{(R_k, \mathcal{K}_k)\}$
POSTAVI $NTestŠr \leftarrow NTestŠr \cup \{(R_k, \mathcal{K}_k)\}$
POSTAVI $S_n \leftarrow S_n \setminus \{(R_k, \mathcal{K}_k)\}$
POSTAVI $S_{xy} \leftarrow S_{xy} \setminus \{(R_i, \mathcal{K}_i)\}$
KRAJ AKO
KRAJ RADI *traženje_ekviv*
KRAJ RADI *ekvivalentne_šr*
POSTAVI $TestŠr \leftarrow NTestŠr$
POSTAVI $Temp \leftarrow S_n$ (∗ Zamena uloga skupova šema relacija S_m i S_n ∗)
POSTAVI $S_n \leftarrow S_m$
POSTAVI $S_m \leftarrow Temp$
KRAJ RADI *formiranje_klasa_ekv*
POSTAVI $Ekviv(S_m \cup S_n) \leftarrow Ekviv(S_m \cup S_n) \cup S_{Eq}(R_i)$
KRAJ RADI *prolaz_kroz_mn*
(∗ Formiranje strogo kompatibilne šeme (S, I) ∗)
POSTAVI $I \leftarrow I_m \cup I_n$
POSTAVI $S \leftarrow \emptyset$
RADI *prolaz_kroz_klase* ***ZA*** $(\forall S_{Eq} \in Ekviv(S_m \cup S_n))$
POSTAVI $(R, \mathcal{K}) \leftarrow \left(\bigcup_{(R_i, \mathcal{K}_i) \in S_{Eq}} R_i , \bigcup_{(R_i, \mathcal{K}_i) \in S_{Eq}} \mathcal{K}_i \right)$
POSTAVI $S \leftarrow S \cup \{(R, \mathcal{K})\}$
KRAJ RADI *prolaz_kroz_klase*
(∗ Kraj formiranja strogo kompatibilne šeme (S, I) ∗)
AKO JE *Ind* = *Itrue* ***TADA***

POSTAVI *Upz* ← ‘*Potrebno je izgenerisati nepokrivena* ’ +
‘*međurelaciona ograničenja:* $I \models I_n \cup I_m$.’ (* $(S_m, I_m) \cong_I (S_n, I_n)$ *)
KRAJ AKO
INAČE (* $(S_m, I_m) \cong (S_n, I_n) \vee (S_m, I_m) \cong_I (S_n, I_n)$ *)
POSTAVI *Upz* ← ‘*Nekompatibilne šeme modula se ne mogu integrisati* ’
KRAJ AKO (* $(S_m, I_m) \cong (S_n, I_n) \vee (S_m, I_m) \cong_I (S_n, I_n)$ *)
KRAJ PROCESA *Integracija*

Slika 4.16.

Teorema 4.11. Algoritam 4.11. je korektan, polinomijalne složenosti ($O(n^4)$) u najlošijem slučaju.

Dokaz. (*Korektnost*) Sledi na osnovu teorema 4.1, 4.3. i 4.10. i sledeće analize.

Korektnost bloka algoritma ‘Formiranje strogo kompatibilne šeme (S, I)’ sledi direktno iz teoreme 4.1, odnosno iz formula (4.28) i (4.29). Potrebno je obrazložiti korektnost bloka za formiranje klasa ekvivalencija.

Blok za formiranje klasa ekvivalencija je korektan zahvaljujući činjenici da ako neka klasa ekvivalencije $S_{Eq}(R_i)$ sadrži više od jedne šeme relacije, tada za svaku šemu relacije $(R_j, \mathcal{K}_j) \in S_{Eq}(R_i)$ važi da mora u suprotnom modulu od onoga, kojem pripada ta šema relacije, postojati bar jedna šema relacije $(R_k, \mathcal{K}_k) \in S_{Eq}(R_i)$, takva da je $\mathcal{K}_j \cap \mathcal{K}_k \neq \emptyset$. Ova tvrdnja je posledica činjenice da ne postoji kolizija $EqKeyC(S_m, S_n)$ (odnosno $IntCDk(S_m, S_n)$).

Pošto skup klasa $Ekviv(S_m \cup S_n)$ definiše particiju, jednom ispitivana šema relacije se može izostaviti iz daljeg rada, nakon formiranja klase kojoj pripada, jer se više ne može pojaviti kao kandidat za dodavanje u neku drugu klasu ekvivalencije.

(*Složenost*) Sledi na osnovu teoreme 4.10. i činjenice da je složenost algoritma diktirana složenošću procesa *Stroga_kompatibilnost*, koja iznosi ($O(n^4)$). Složenost bloka algoritma za formiranje skupa klasa $Ekviv(S_m \cup S_n)$, odnosno kompozicije petlji *prolaz_kroz_mn* i *formiranje_klasa_ekv* iznosi u najlošijem slučaju ($O(n^2)$), jer algoritam obezbeđuje da se svaka šema relacije iz skupa $S_m \cup S_n$ ispituje najviše dva puta kako bi se utvrdila eventualna egzistencija ekvivalentne šeme relacije iz suprotnog modula od onog, kojem data šema relacije pripada. Petlja *prolaz_kroz_klase* je linearne složenosti. □

5. Poglavlje

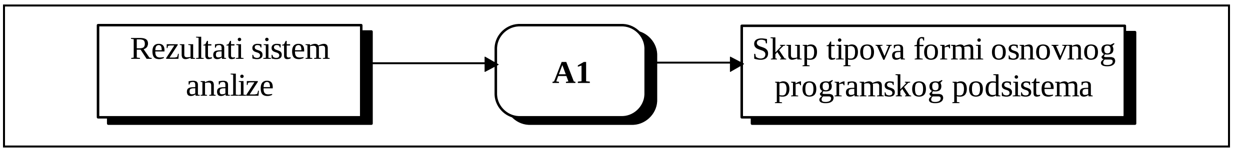
Metodološki aspekti primene postupka integracije šeme baze podataka

Na početku ovog poglavlja će biti, putem dijagramske tehnike, dat globalni prikaz načina primene CASE alata za automatizovano projektovanje šeme baze podataka i prototipa aplikacija. Nakon toga će biti detaljnije razmatrani metodološki aspekti postupaka testiranja međusobne kompatibilnosti šema modula BP i integracije šema modula u jedinstvenu šemu BP.

5.1. Upotreba CASE alata za automatizovano generisanje šeme BP i prototipa aplikacija

Primena CASE alata za automatizovano generisanje šeme BP i prototipa aplikacija se odvija saglasno metodološkom pristupu razvoja IS, opisanom u drugom poglavlju ovog rada. Prve aktivnosti su:

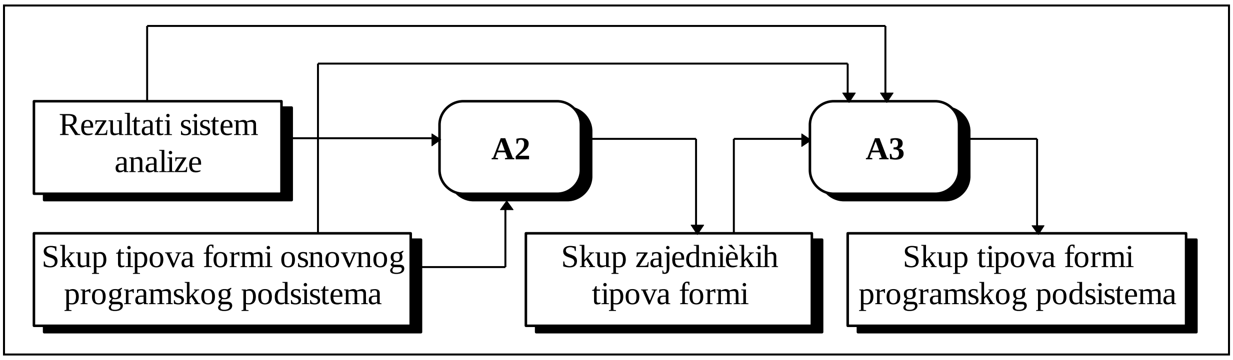
A1. Projektovanje tipova formi, namenjenih za kreiranje i modifikovanje podataka zajedničkih činilaca poslovanja (Slika 5.1).



Slika 5.1. Projektovanje tipova formi zajedničkih činilaca poslovanja

A2. Projektovanje zajedničkih tipova formi (tj. tipova formi koji nastaju od zajedničkih dokumenata i koriste se u najmanje dva programska podsistema). Ova aktivnost se paralelno odvija za svaki programski podsistem posebno (Slika 5.2).

A3. Projektovanje ostalih tipova formi, specifičnih samo za jedan programski podsistem (prvo onih koje su namenjene za ažuriranje stanja baze podataka, a zatim i onih koje su namenjene za izveštaje) (Slika 5.2).



Slika 5.2. Projektovanje zajedničkih i specifičnih tipova formi programskog podsistema

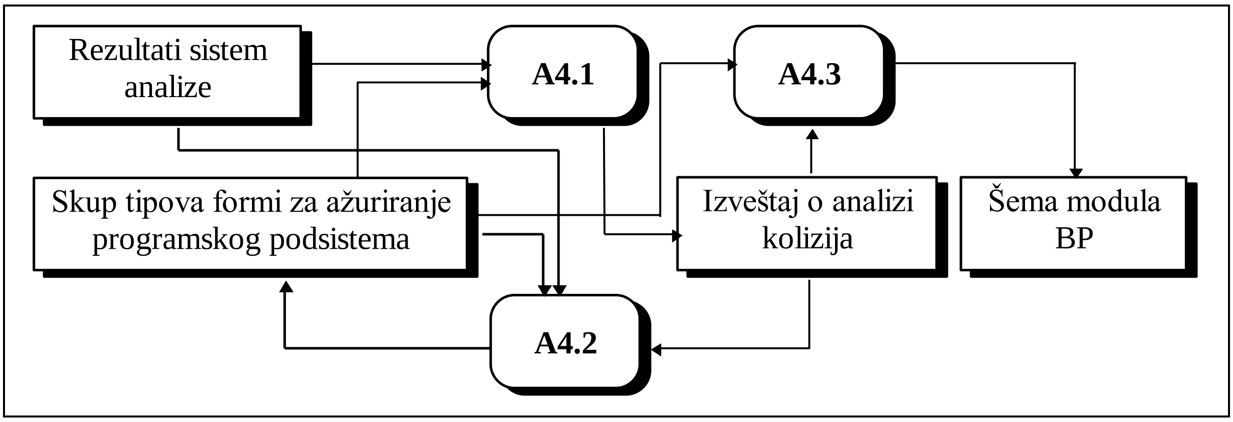
Aktivnosti A1-A3 izvode projektanti šema modula, koristeći se inteligentnim editorom za projektovanje tipova formi.

Sledeća aktivnost, sa svoje tri podaktivnosti:

A4. Projektovanje šeme modula BP (Slika 5.3), tj:

- **A4.1.** Analiza tipova formi i detekcija mogućih kolizija,
- **A4.2.** Otklanjanje detektovanih kolizija i
- **A4.3.** Oblikovanje polaznog skupa ograničenja i generisanje relacione šeme modula BP (podaktivnost može da započne tek pošto se razreše sve eventualne kolizije)

čini celinu s prethodnim aktivnostima. Podaktivnosti A4.1. i A4.3. su u potpunosti automatizovane, dok podaktivnost A4.2. izvodi projektant šeme modula, na isti način kao i aktivnosti A1-A3. Rezultat iterativne primene aktivnosti A1-A4 treba da bude šema modula BP. Šema modula BP se generiše modifikovanim algoritmom sinteze (videti tačku 2.2.2.3).



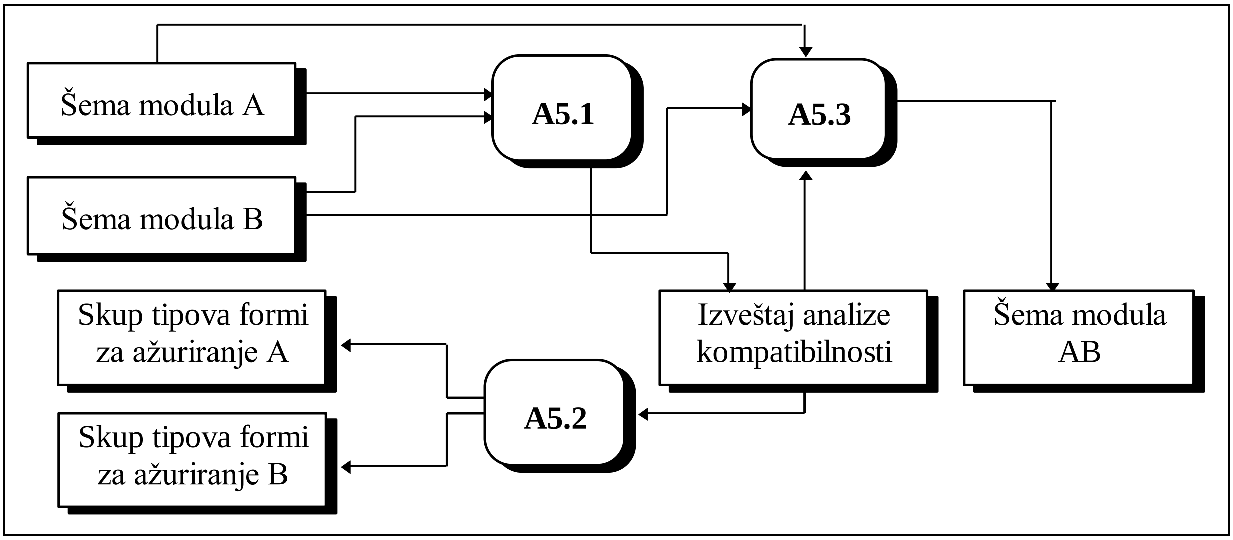
Slika 5.3. Analiza kolizija i projektovanje šeme modula programskog podsistema

Naredna aktivnost sa svoje tri podaktivnosti:

A5. Integracija šema modula BP (Slika 5.4):

- **A5.1.** Analiza kompatibilnosti šema modula,
- **A5.2.** Otklanjanje detektovanih kolizija i
- **A5.3.** Integracija šema modula (podaktivnost može započeti nakon što se otklone sve nekonzistentnosti između šema modula)

se sprovodi nakon uspešno obavljenih aktivnosti A1-A4. Otklanjanje nekompatibilnosti šema modula se vrši preprojektovanjem postojećih, ili dodavanjem novih tipova formi, zbog kojih je došlo do pojave kolizije između datih šema modula, što znači da se redosled aktivnosti A1-A5 primenjuje iterativno, sve dok podaktivnost A5.1 uspešno ne okonča, čime se stvaraju preduslovi za integraciju dve šeme modula. Sledeća tačka ovog poglavalja detaljnije razmatra pitanja sprovođenja aktivnosti A5.



Slika 5.4. Analiza kompatibilnosti i integracija šema modula

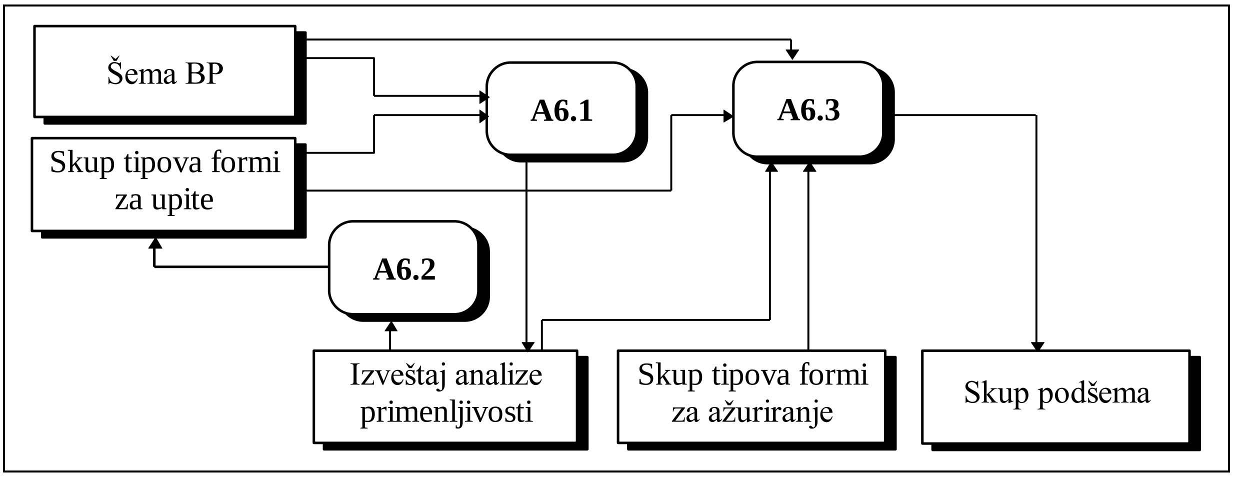
A6. Projektovanje podšema (Slika 5.5) s podaktivnostima:

A6.1. Analiza primenljivosti tipova formi za upite,

A6.2. Preprojektovanje (ili uklanjanje) neprimenljivih tipova formi i

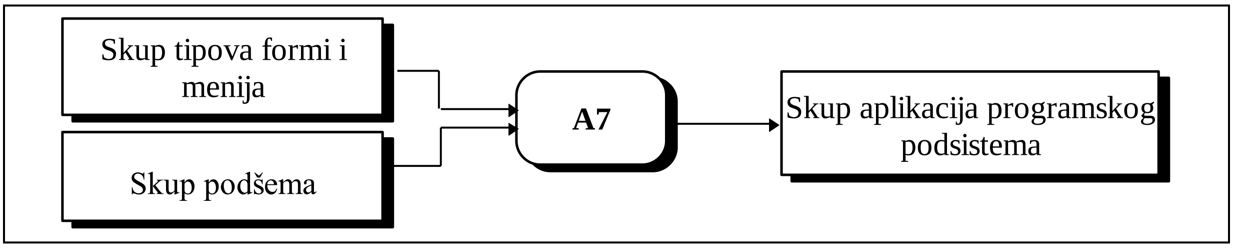
A6.3. Oblikovanje podšeme za svaki primenljivi tip forme,

predstavlja jednu od aktivnosti vezanih za projektovanje programskog podsistema. Podaktivnosti A6.1. i A6.3. se u potpunosti obavljaju automatizovano **[Lu2]**, dok podaktivnost A6.2. obavlja projektant šeme modula. Nakon završetka aktivnosti A6, za svaku programsku specifikaciju su definisani izgled ekranske forme i podšema (tj. pogled). Time se stvaraju preduslovi za sprovođenje aktivnosti:



Slika 5.5. Analiza primenljivosti i projektovanje podšema

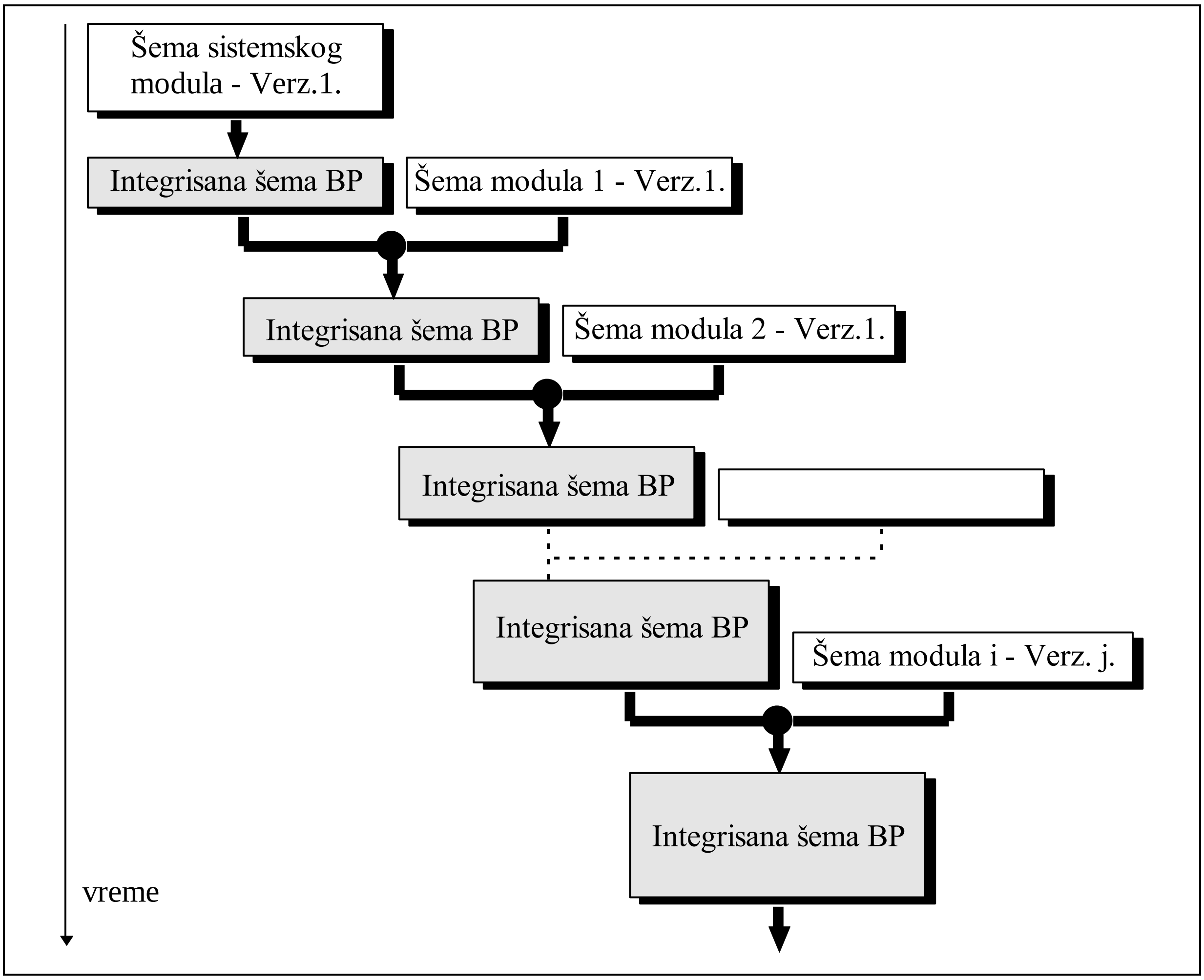
A7. Projektovanje skupa aplikacija programskog podsistema i generisanje prototipa aplikacija (Slika 5.6).



Slika 5.6. Projektovanje programskog podsistema

5.2. Analiza kompatibilnosti i integracija šema modula

Šema BP informacionog sistema se formira progresivnom integracijom šema modula BP. Proces integracije je vremenski određen prioritetima razvoja pojedinačnih programskih podsistema i odvija se tokom celog životnog ciklusa informacionog sistema.



Slika 5.7. Šematski prikaz postupka integracije šeme BP

Šematski prikaz postupka oblikovanja integrisane šeme BP je dat na slici 5.7. Postupak integracije zahteva da se uvede pojam ***tekućeg stanja šeme*** BP informacionog sistema. Kao inicijalno ***tekuće stanje šeme*** BP se uzima šema sistemskog modula, jer se ona prva projektuje. Nakon toga, svako novo stanje šeme BP se dobija integracijom tekućeg stanja šeme BP s nekom šemom modula i zatim se to novo stanje proglašava tekućim stanjem integrisane šeme BP, čime se postupak rekurzivno ponavlja.

Integracija šeme BP (aktivnost A5, odnosno podaktivnost A5.3) se odvija prema algoritmu 4.11, pri čemu se analiza stroge i intenzionalne kompatibilnosti (podaktivnost A5.1) sprovodi pozivom procesa *Stroga_kompatibilnost* (algoritam 4.10). Proces *Stroga_kompatibilnost* obezbeđuje formalnu detekciju:

- kolizije tranzitivnosti, putem algoritma 4.1,
- mogućih homonima (tj. neprimarnih obeležja, sadržanih u bar dve šeme relacije, takvih da jedna drugu ne zatvara), putem algoritma 4.1,
- kolizije funkcionalni-nefunkcionalni odnos, putem algoritma 4.3,
- kolizije dela ključa, putem algoritma 4.4,
- lokalne kolizije međurelacionih ograničenja, putem algoritma 4.6,
- globalne kolizije međurelacionih ograničenja, putem algoritma 4.8. i
- kolizije ekvivalentnih ključeva, putem algoritma 4.9.

Ostaje, međutim, problem nemogućnosti formalne detekcije ***sinonima*** - različitih obeležja, kojima je pridružena ista semantika. Kolizije sinonima jedino mogu razrešiti projektanti šema modula. U cilju lakšeg detektovanja kolizija sinonima, projektantima šema modula treba da stoje na raspolaganju sledeći izveštaji:

- Leksikografski uređeni spisak primarnih obeležja iz posmatrane šeme modula i tekuće verzije šeme BP,
- Pregled novoformiranih šema relacija šeme modula:

  $$Nove\_\check{S}R = \{(R_m, \mathcal{K}_m) \in S_m \mid \neg(\exists(R_n, \mathcal{K}_n) \in S_n)(\mathcal{K}_m \cap \mathcal{K}_n \neq \emptyset)\},$$

  pri čemu je (S_m, I_m) šema modula, a (S_n, I_n) tekuće stanje šeme BP i
- Pregled zajedničkih i posebnih obeležja šema relacija sadržanih i u šemi modula i u tekućoj šemi BP:

  $$Zajedni\check{c}ke\_\check{S}R = \{((R_m \setminus R_n, \mathcal{K}_m), (R_n \setminus R_m, \mathcal{K}_n), R_m \cap R_n) \mid (R_m, \mathcal{K}_m) \in S_m \wedge (R_n, \mathcal{K}_n) \in S_n \wedge \mathcal{K}_m \cap \mathcal{K}_n \neq \emptyset\},$$

  pri čemu je (S_m, I_m) šema modula, a (S_n, I_n) tekuće stanje šeme BP.

Sam proces integrisanja (A5.3) ne zahteva interakciju s projektantom šeme modula, pod pretpostavkom da su tekuće stanje šeme BP i šema modula strogo, ili intenzionalno kompatibilni. Analiza kompatibilnosti (A5.1), međutim, produkuje izveštaje *TranC_Rep*, *FnfC_Rep*, *IntcDk_Rep*, *IntCLi_Rep*, *IntCGr_Rep* i *EqKeyC_Rep* (izlazi algoritma 4.10), ukoliko šema BP i šema modula nisu strogo, odnosno intenzionalno kompatibilne. Projektant treba, tumačeći sadržaje tih izveštaja, da otkloni nastale kolizije preprojektovanjem postojećih, ili dodavanjem novih tipova formi za ažuriranje. Dodavanje novog tipa forme za ažuriranje se može realizovati na tri načina:

- preuzimanjem (pozajmljivanjem) tipa forme iz druge poslovne funkcije, pri čemu se modifikovanje pozajmljenog tipa forme ne dozvoljvava,
- kreiranjem novog tipa forme na bazi već postojećeg tipa forme (isprojektovanog u okviru matične, ili bilo koje druge poslovne funkcije), ili
- kreiranjem potpuno novog tipa forme.

Saglasno prototipskom pristupu razvoja informacionog sistema (ukratko prikazanom u drugom poglavlju rada), moguće je da se pojave kolizije između šeme modula i tekuće šeme BP koje su posledica činjenice da prva verzija šeme modula, na osnovu koje je izgenerisan i instaliran kod krajnjeg korisnika prototip aplikacije, nije u saglasnosti s novom verzijom iste šeme modula. Ukoliko je korisnik, putem prototipa aplikacije, izgenerisanog na osnovu prve verzije šeme modula, uneo podatke u bazu, tada projektant mora, pošto otkloni kolizije na nivou šema modula, da izvrši i *restrukturiranje* prethodno unetih podataka, tako da odgovaraju novom opisu integrisane šeme. Potreba za restrukturiranjem već unetih podataka se može javiti i u slučaju da se šema modula prvi put integriše u tekuću šemu BP. Restrukturiranje podataka se obavlja putem dostupnih alata konkretnog relacionog sistema za upravljanje bazama podataka koji je u upotrebi.

U nastavku poglavlja će, putem demo-primera, biti dat prikaz nekih karakterističnih slučajeva kolizija šeme modula i tekuće šeme BP, kao i načina za njihovo razrešavanje.

5.3. Analiza kompatibilnosti - studija slučaja

U ovoj tački će, putem pojednostavljenog primera poslovne funkcije “***Skladištenje***”, biti prikazan postupak projektovanja integrisane šeme BP. Pri tome se prezentira i projektovanje dela šeme sistemskog modula, kao i nekih zajedničkih tipova formi iz drugih poslovnih funkcija, koji se koriste u okvriu funkcije “Skladištenje”.

5.3.1. Projektovanje sistemskog modula (MSIS)

MSIS - Skup tipova formi

Neka je projektant sistemskog modula (**MSIS**) napravio tipove formi:

- *ORG_JEDINICA* (***OJ***) - formiranje evidencije o organizacionoj strukturi realnog sistema, slika 5.8,
- *MATERIJAL* (***MT***) - formiranje kataloga materijala (tj. evidentiranje osnovnih podataka o materijalu), slika 5.9,
- *POSL_PART* (***PP***) - formiranje kataloga (evidencije) poslovnih partnera, slika 5.10,
- *LOKACIJA* (***LO***) - formiranje kataloga lokacija na kojima su fizički smeštene poslovne celine realnog sistema, slika 5.11,

ORG_JEDINICA

OJidb, *OJnaz, OJnadr, Lokidb, Loknaz, LDeidb, LDenaz*

Slika 5.8.

MATERIJAL

Matidb, *Matnaz, Matjus, Matozn*

Slika 5.9.

POSL_PART

PPidb, *PPnaz, Adrptt, Adrmes, PPadr, PPtel*

Slika 5.10.

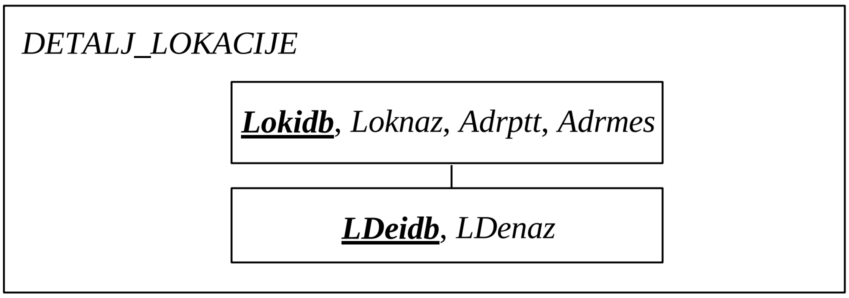
LOKACIJA

Lokidb, *Loknaz, Adrptt, Adrmes, Lokadr*

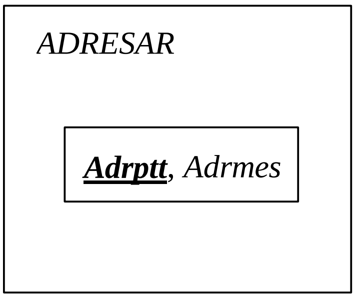
Slika 5.11.

- *DETALJ_LOKACIJE* (***LD***) - formiranje kataloga detalja (evidencije II nivoa detaljnosti) lokacija na kojima su fizički smeštene poslovne celine realnog sistema, slika 5.12. i

- *ADRESAR* - (***AD***) formiranje kataloga adresa, odnoso mesta, slika 5.13.



Slika 5.12.



Slika 5.13.

U nastavku je dat opis semantike upotrebljenih obeležja, kao i specifikacija funkcija *dbs*, *el*, *ren* i *man*:

- *OJidb* - Identifikaciona oznaka organizacione jedinice, *dbs*(*OJidb*) = *Y*, *el*(*OJidb*) = *E*,
- *OJnaz* - Naziv organizacione jedinice, *dbs*(*OJnaz*) =*Y*, *el*(*OJnaz*) =*E*, *man*(*OJ*, *OJnaz*) =*M*,
- *OJnadr* - ID oznaka direktno nadređene organizacione jedinice, *dbs*(*OJnadr*) = *Y*, *el*(*OJnadr*) = *R*, *ren*(*OJnadr*) = *OJidb*, *man*(*OJ*, *OJnadr*) = *O*,
- *Matidb* - Identifikaciona oznaka materijala, *dbs*(*Matidb*) = *Y*, *el*(*Matidb*) = *E*,
- *Matnaz* - Naziv materijala, *dbs*(*Matnaz*) = *Y*, *el*(*Matnaz*) = *E*, *man*(*MT*, *Matnaz*) = *M*,
- *Matjus* - JUS oznaka materijala, *dbs*(*Matjus*) = *Y*, *el*(*Matjus*) = *E*, *man*(*MT*, *Matjus*) = *O*,
- *Matozn* - Stara oznaka materijala, *dbs*(*Matozn*) =*Y*, *el*(*Matozn*) =*E*, *man*(*MT*, *Matozn*) =*O*,
- *PPidb* - Identifikaciona oznaka poslovnog partnera, *dbs*(*PPidb*) = *Y*, *el*(*PPidb*) = *E*,
- *PPnaz* - Naziv poslovnog partnera, *dbs*(*PPnaz*) = *Y*, *el*(*PPnaz*) = *E*, *man*(*PP*, *PPnaz*) = *M*,
- *PPadr* - Adresa poslovnog partnera, *dbs*(*PPadr*) = *Y*, *el*(*PPadr*) = *E*, *man*(*PP*, *PPadr*) =*O*,
- *PPtel* - Adresa poslovnog partnera, *dbs*(*PPtel*) = *Y*, *el*(*PPtel*) = *E*, *man*(*PP*, *PPtel*) = *O*,
- *Lokidb* - Identifikaciona oznaka lokacije, *dbs*(*Lokidb*) = *Y*, *el*(*Lokidb*) = *E*, *man*(*OJ*, *Lokidb*) = *M*,
- *Loknaz* - Naziv lokacije, *dbs*(*Loknaz*) = *Y*, *el*(*Loknaz*) = *E*, *man*(*LO*, *Loknaz*) = *M*,
- *Lokadr* - Adresa lokacije, *dbs*(*Lokadr*) = *Y*, *el*(*Lokadr*) = *E*, *man*(*LO*, *Lokadr*) = *M*,
- *LDeidb* - Identifikaciona oznaka detalja lokacije, *dbs*(*LDeidb*) = *Y*, *el*(*LDeidb*) = *E*, *man*(*OJ*, *LDeidb*) = *M*,
- *LDenaz* - Naziv detalja lokacije, *dbs*(*LDenaz*) = *Y*, *el*(*LDenaz*) = *E*, *man*(*LO*, *LDenaz*) =*M*,
- *Adrptt* - PTT broj mesta, *dbs*(*Adrptt*) = *Y*, *el*(*Adrptt*) = *E*, *man*(*PP*, *Adrptt*) = *man*(*LO*, *Adrptt*) = *M*,
- *Adrmes* - Naziv mesta, *dbs*(*Adrmes*) = *Y*, *el*(*Adrmes*) = *E*, *man*(*AD*, *Adrmes*) = *M*.

MSIS - Šema modula BP

Na osnovu prikazanih tipova formi, automatski se generiše, modifikovanim algoritmom sinteze, šema sistemskog modula (S_s , I_s), sa skupom šema relacija S_s :

- *Materijal*({*Matidb*, *Matnaz*, *Matjus*, *Matozn*}, {*Matidb*}),
- *Posl_part*({*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*}, {*PPidb*}),
- *Org_jedinica*({*OJidb*, *OJnaz*, *OJnadr*, *Lokidb*, *LDeidb*}, {*OJidb*}),
- *Lokacija*({*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*}, {*Lokidb*}),
- *Detalj_lokacije*({*Lokidb*, *LDeidb*, *LDenaz*}, {*Lokidb* + *LDeidb*}),
- *Adresar*({*Adrptt*, *Adrmes*}, {*Adrptt*})

i skupom međurelacionih ograničenja I_s :

- *Posl_part*[*Adrptt*] $\subseteq$ *Adresar*[*Adrptt*],
- *Org_jedinica*[*Lokidb* + *LDeidb*] $\subseteq$ *Detalj_lokacije*[*Lokidb* + *LDeidb*],
- *Org_jedinica*[*OJnadr*] \ {*w*}[*)] $\subseteq$ *Org_jedinica*[*OJidb*],
- *Lokacija*[*Adrptt*] $\subseteq$ *Adresar*[*Adrptt*] i
- *Detalj_lokacije*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*].

Isprojektovana šema sistemskog modula postaje, istovremeno, početno stanje integrisane šeme BP (**ISBP**), koje će biti označena kao (S_0 , I_0).

5.3.2. Projektovanje modula "Skladištenje" (MSKL)

MSKL - Polazne pretpostavke

P1. U cilju isticanja razloga za inicijalno uključivanje tipova formi sistemskog modula u skup tipova formi ostalih modula, prva verzija modula ***Skladištenje*** (**MSKL**) će biti napravljena bez tipova formi sistemskog modula.

P2. Neka je projektant prvo napravio tipove formi koje odgovaraju skladišnim dokumentima. Ovo su, ujedno, zajednički tipovi formi, jer se skladišni dokumenti formiraju u ovoj poslovnoj funkciji, a koriste se i u drugim poslovnim funkcijama (u knjigovodstvu). Pored toga, neka su odmah kreirani i tipovi formi putem kojih se reguliše organizaciona struktura skladišta.

P3. Pretpostavlja se, takođe, da u okviru ove funkcije, već postoji evidencija materijala, preuzeta iz prethodnog informacionog sistema. Saglasno ovoj evidenciji, projektant pravi poseban tip forme.

MSKL - Polazni skup tipova formi

Saglasno uvedenom scenariju i potrebama ovoga rada, pretpostavlja se da su kreirani tipovi formi:

- *MATERIJALS* (***MS***) - formiranje kataloga materijala, slika 5.14,
- *SKLADIŠTE* (***SK***) - formiranje kataloga svih skladišta, slika 5.15,
- *SKL_ZONA* (***SZ***) - formiranje kataloga skladišnih zona, slika 5.16. i

MATERIJALS

Matozn, *Matnaz*, *Matjus*, *Matzal*

Slika 5.14.

SKLADIŠTE

Sklidb, *Sklnaz*, *Lokidb*, *Loknaz*

Slika 5.15.

*) *w* je oznaka za nula vrednost

SKL_ZONA

Sklidb, *Sklnaz, Lokidb, Loknaz*

Zonidb, *Zonnaz, ZLoidb, ZLonaz, LDeidb, LDenaz*

Slika 5.16.

OTPREMNICA

Sklidb, Zonidb, Posgod, Sdvrst, Sdtip, Sdidbr, *Sdatiz, PPidb, PPnaz*

Stvidb, ***Matozn***, *Matnaz, Stvkol*

Slika 5.17.

- *OTPREMNICA* (***OT***) - dokument kojim se reguliše izdavanje robe poslovnom partneru, slika 5.17.

U nastavku je za obeležja koja nisu upotrebljena u okviru MSIS, dat opis njihove semantike. Pored toga, data je specifikacija funkcija *dbs*, *el*, *ren*, *man* i *der*:

- *Matzal* - Ukupne zalihe materijala, *dbs*(*Matzal*) = *Y*, *el*(*Matzal*) = *E*,
- *Sklidb* - Identifikaciona oznaka skladišta, *dbs*(*Sklidb*) = *Y*, *el*(*Sklidb*) = *E*,
- *Sklnaz* - Naziv skladišta, *dbs*(*Sklnaz*) = *Y*, *el*(*Sklnaz*) = *E*, *man*(*SK*, *Sklnaz*) = *M*,
- *man*(*SK*, *Lokidb*) = *M*,
- *Zonidb* - Identifikaciona oznaka zone skladišta, *dbs*(*Zonidb*) = *Y*, *el*(*Zonidb*) = *E*,
- *Zonnaz* - Naziv zone skladišta, *dbs*(*Zonnaz*) = *Y*, *el*(*Zonnaz*) = *E*, *man*(*SZ*, *Zonnaz*) = *M*,
- *Zloidb* - Identifikaciona oznaka lokacije zone skladišta, *dbs*(*Zloidb*) = *Y*, *el*(*Zloidb*) = *R*, *ren*(*Zloidb*) = *Lokidb*, *man*(*SZ*, *Zloidb*) = *M*. Obeležje *Zloidb* je uvedeno zbog činjenice (za koju se u ovom, pojednostavljenom, primeru pretpostavlja da važi) da lokacija zone skladišta i lokacija skladišta ne moraju biti iste, a na bilo kom tipu forme jedno obeležje može biti upotrebljeno u najviše jednom tipu objekta,
- *Zlonaz* - Naziv lokacije zone, *dbs*(*Zlonaz*) = *N*, *el*(*Zlonaz*) = *R*, *ren*(*Zlonaz*) = *Loknaz*,
- *man*(*SZ*, *LDeidb*) = *M*,
- *Sdatiz* - Datum izdavanja dokumenta, *dbs*(*Sdatiz*) = *Y*, *el*(*Sdatiz*) =*E*, *man*(*OT*, *Sdatiz*)=*M*,
- *Posgod* - Poslovna godina u kojoj se dokument formira, *dbs*(*Posgod*) = *Y*, *el*(*Posgod*) =*D*, *der*(*Posgod*) = {*Sdatiz*},
- *Sdvrst* - Identifikaciona oznaka vrste skladišnog dokumenta, *dbs*(*Sdvrst*) = *Y*, *el*(*Sdvrst*) = *E*, *dom*(*Sdvrst*) = {*OT*, *PR*, *IU*, *II*} (*OT* - otpremnica, *PR* - prijemnica, *IU* - interni ulaz, *II* - interni izlaz),
- *Sdtip* - Tip dokumenta, *dbs*(*Sdtip*) = *Y*, *el*(*Sdtip*) = *E*, *dom*(*Sdtip*) = {*O*, *S*} (*+1* - običan, *-1* - storno dokument),
- *Sdidbr* - Redni (identifikacioni) broj dokumenta, *dbs*(*Sdidbr*) = *Y*, *el*(*Sdidbr*) = *E*,
- *man*(*OT*, *PPidb*) = *M*,
- *Stvidb* - Redni (identifikacioni) broj stavke dokumenta, *dbs*(*Stvidb*) = *Y*, *el*(*Stvidb*) = *E*,
- *man*(*OT*, *Matozn*) = *M* i

- *Stvkol* - Količina materijala na stavci, *dbs*(*Stvkol*) = *Y*, *el*(*Stvkol*) = *E*, *man*(*OT*, *Stvkol*)=*M*.

MSKL - Šema modula BP - V.1.

Na osnovu prikazanih tipova formi, generiše se prva verzija šeme MSKL (S_{m_1}, I_{m_1}), sa skupom šema relacija S_{m_1}:

- *Materijal*({*Matozn*, *Matnaz*, *Matjus*, *Matzal*}, {*Matozn*}),
- *Skladište*({*Sklidb*, *Sklnaz*, *Lokidb*, *Loknaz*}, {*Sklidb*}),
- *Skl_zona*({*Sklidb*, *Zonidb*, *Zonnaz*, *ZLoidb*, *LDeidb*, *LDenaz*}, {*Sklidb* + *Zonidb*}),
- *Skl_dokument*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Sdatiz*, *PPidb*, *PPnaz*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}),
- *Stavka_dokum*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Stvidb*, *Matozn*, *Stvkol*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Stvidb*, *Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Matozn*})

i skupom međurelacionih ograničenja I_{m_1}:

- *Skl_zona*[*Sklidb*] ⊆ *Skladište*[*Sklidb*],
- *Skl_dokument*[*Sklidb* + *Zonidb*] ⊆ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_dokum*[*Sklidb*+*Zonidb*+*Posgod*+*Sdvrst* +*Sdtip* + *Sdidbr*] ⊆ *Skl_dokument*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Skl_dokument*[*Sklidb*+*Zonidb*+*Posgod* +*Sdvrst* +*Sdtip* +*Sdidbr*] ⊆ *Stavka_dokum*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*] i
- *Stavka_dokum*[*Matozn*] ⊆ *Materijal*[*Matozn*].

ISBP / MSKL - Ispitivanje stroge kompatibilnosti I

Proces *Stroga_kompatibilnost* (algoritam 4.10) za ulazne šeme BP (S_s, I_s) i (S_{m_1}, I_{m_1}), daje izlazni rezultat *Ind* = *false*, što znači da je (S_s, I_s) $\ncong_I$ (S_{m_1}, I_{m_1}) i formira sledeće izveštaje o detektovanim kolizijama:

- za koliziju tranzitivnosti *TranC*(S_s, S_{m_1}), algoritam 4.1. formira izveštaj *TranC_Rep* (slika 5.18),
- za *fnf*-koliziju *FnfC*(S_s, S_{m_1}) je *FnfC_Rep* = ∅,
- za koliziju dela ključa *IntcDk*(S_s, S_{m_1}) je *IntcDk_Rep* = ∅,
- za lokalnu koliziju međurelacionih ograničenja *IntcLi*(S_s, S_{m_1}) je *IntCLi_Rep* = ∅,
- za globalnu koliziju međurelacionih ograničenja *IntCGr*(S_s, S_{m_1}) algoritam 4.8. formira izveštaj *IntCGr_Rep* (slika 5.19) i
- za koliziju ekvivalentnih ključeva je *EqKeyC*(S_s, S_{m_1}) je *EqKeyC_Rep* = ∅.

Svi izveštaji, koje produkuje proces *Stroga_kompatibilnost* (algoritam 4.10), kao što je već napomenuto u četvrtom poglavlju, sastoje se od sledećih elemenata: *poruka*, *upozorenja* i *grešaka*. Svaki element nekog izveštaja će biti numerisan rednim brojem, počevši od *1*. Izveštaji ove studije slučaja će, takođe, biti numerisani rednim brojevima. U nastavku rada, pozivanje na određenu poruku, upozorenje, ili grešku nekog konkretnog izveštaja, vršiće se po šemi:

*Naziv_izveštaja.**I**-<Rbr_izveštaja>**.**<Rbr_greške/upozorenja/poruke>,*

dok će se pozivanje na sam izveštaj vršiti po šemi:

*Naziv_izveštaja.**I**-<Rbr_izveštaja>.*

IZVEŠTAJ I-1: ***TranC_Rep*** - *Kolizija tranzitivnosti*	
Modul: **Sistemski** (**MSIS**)	Modul: **Skladištenje** (**MSKL**)
Materijal: $\mathcal{K}$ = {*Matidb*}	*Materijal*: $\mathcal{K}$ = {*Matozn*}
1. Greška: MSIS.*Materijal* → MSKL.*Materijal*, *TrAttr* = {*Matnaz*, *Matjus*, *Matzal*} *Tranzitivna obeležja u šemi relacije:* MSIS.*Materijal*.	
Posl_part: $\mathcal{K}$ = {*PPidb*}	*Skl_dokument*: $\mathcal{K}$ = {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}
2. Greška: MSKL.*Skl_dokument* → MSIS.*Posl_part*, *TrAttr* = {*PPnaz*} *Tranzitivna obeležja u šemi relacije:* MSKL.*Skl_dokument*.	
Lokacija: $\mathcal{K}$ = {*Lokidb*}	*Skladište*: $\mathcal{K}$ = {*Sklidb*}
3. Greška: MSKL.*Skladište* → MSIS.*Lokacija* *TrAttr* = {*Loknaz*} *Tranzitivna obeležja u šemi relacije:* MSKL.*Skladište*.	
Detalj_lokacije: $\mathcal{K}$ = {*Lokidb* + *LDeidb*}	*Skl_zona*: $\mathcal{K}$ = {*Sklidb* + *Zonidb*}
4. Upozorenje: MSIS.*Detalj_lokacije* $\not\leftrightarrow$ MSKL.*Skl_zona* *TrAttr* = {*LDenaz*} *Mogući homonimi.*	

Slika 5.18.

IZVEŠTAJ I-2: ***IntCGr_Rep*** - *Globalna kolizija međurelacionih ograničenja*	
Modul: **Sistemski** (**MSIS**)	Modul: **Skladištenje** (**MSKL**)
Materijal: *R* = {*Matidb*, *Matnaz*, *Matjus*, *Matozn*} $\mathcal{K}$ = {*Matidb*}	*Materijal*: *R* = {*Matozn*, *Matnaz*, *Matjus*, *Matzal*} $\mathcal{K}$ = {*Matozn*}
1. Upozorenje: *ic*: MSIS.*Materijal*[*Matozn*] ⊆ MSKL.*Materijal*[*Matozn*] *Nepokriveno međurelaciono ograničenje.*	
Posl_part: *R* = {*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*} $\mathcal{K}$ = {*PPidb*}	*Skl_dokument*: *R* = {*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Sdatiz*, *PPidb*, *PPnaz*} $\mathcal{K}$ = {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}
2. Upozorenje: *ic*: MSKL.*Skl_dokument*[*PPidb*] ⊆ MSIS.*Posl_part*[*PPidb*] *Nepokriveno međurelaciono ograničenje.*	

<table>
<tr><td>Lokacija:
R = {Lokidb, Loknaz, Lokadr, Adrptt}
$\mathcal{K}$ = {Lokidb}</td><td>Skladište:
R = {Sklidb, Sklnaz, Lokidb, Loknaz}
$\mathcal{K}$ = {Sklidb}</td></tr>
<tr><td colspan="2">3. Upozorenje: ic: MSKL.Skladište[Lokidb] ⊆ MSIS.Lokacija[Lokidb]
Nepokriveno međurelaciono ograničenje.</td></tr>
<tr><td>Lokacija:
R = {Lokidb, Loknaz, Lokadr, Adrptt}
$\mathcal{K}$ = {Lokidb}</td><td>Skl_zona:
R = {Sklidb, Zonidb, Zonnaz, ZLoidb, LDeidb, LDenaz}
$\mathcal{K}$ = {Sklidb + Zonidb}</td></tr>
<tr><td colspan="2">4. Upozorenje: ri: MSKL.Skl_zona[ZLoidb] ⊆ MSIS.Lokacija[Lokidb]
Nepokriveni referencijalni integritet preimenovanja.</td></tr>
<tr><td>Detalj_lokacije:
R = {Lokidb, LDeidb, LDenaz}
$\mathcal{K}$ = {Lokidb + LDeidb}</td><td>Skl_zona:
R = {Sklidb, Zonidb, Zonnaz, ZLoidb, LDeidb, LDenaz}
$\mathcal{K}$ = {Sklidb + Zonidb}</td></tr>
<tr><td colspan="2">5. Upozorenje:
ri: MSKL.Skl_zona[ZLoidb + LDeidb] ⊆ MSIS.Detalj_lokacije[Lokidb + LDeidb]
Nepokriveni referencijalni integritet preimenovanja.</td></tr>
</table>

Slika 5.19.

Na primer, referenciranje na drugu grešku izveštaja o kolizijama tranzitivnosti *TranC_Rep*, prikazanom na slici 5.18, će imati oblik:

TranC_Rep.I-1.2,

dok će pozivanje na sam izveštaj imati oblik:

TranC_Rep.I-1.

MSKL - Otklanjanje kolizija

Analizom izveštaja *TranC_Rep.I-1.* i *IntCGr_Rep.I-2.* se dolazi do sledećih zaključaka:

- Greška *TranC_Rep.I-1.1.* i upozorenje *IntCGr_Rep.I-2.1.* ukazuju na neusaglašenost ključeva šema relacija MSIS.*Materijal* i MSKL.*Materijal*, što prouzrokuje potrebu preprojektovanja tipa forme *MATERIJALS*, na osnovi tipa forme *MATERIJAL*. Preprojektovani tip forme *MATERIJALS* je prikazan na slici 5.20.
- Greška *TranC_Rep.I-1.1.* i upozorenje *IntCGr_Rep.I-2.1.* ukazuju na potrebu preprojektovanja tipa forme *OTPREMNICA*, zbog neusaglašenosti ključeva šema relacija MSIS.*Materijal* i MSKL.*Materijal*. Preprojektovani tip forme *OTPREMNICA* je prikazan na slici 5.21, pri čemu se, dodatno, modifkuje funkcija *man* tako da bude *man*(*OT*, *Matozn*) = *O* i *man*(*OT*, *Matidb*) = *M*.
- Greške *TranC_Rep.I-1.2*, *TranC_Rep.I-1.3*, upozorenje *TranC_Rep.I-1.4*, kao i upozorenja *IntCGr_Rep.I-2.2 - IntCGr_Rep.I-2.5*, ukazuju na potrebu preuzimanja tipova formi *POSL_PART*, *LOKACIJA* i *DETALJ_LOKACIJE* (a posredno i tipa forme *ADRESAR*, u cilju izbegavanja kolizija tranzitivnosti) u skup tipova formi MSKL. Druga mogućnost je da se, na osnovu postojećih tipova formi *POSL_PART*, *LOKACIJA* i *DETALJ_LOKACIJE*, naprave u okviru MSKL novi tipovi formi: *POSL_PARTS*,

LOKACIJAS i *DETALJ_LOKACIJES*, koji su prikazani, redom, na slikama 5.22, 5.23. i 5.24.

MATERIJALS

Matidb, *Matozn, Matnaz, Matjus, Matzal*

Slika 5.20.

OTPREMNICA

Sklidb, Zonidb, Posgod, Sdvrst, Sdtip, Sdidbr, *Sdatiz, PPidb, PPnaz*

Stvidb, ***Matidb***, *Matozn, Matnaz, Stvkol*

Slika 5.21.

POSL_PARTS

PPidb, *PPnaz, Adrptt, PPadr, PPtel*

Slika 5.22.

LOKACIJAS

Lokidb, *Loknaz, Lokadr, Adrptt*

Slika 5.23.

DETALJ_LOKACIJES

Lokidb, *Loknaz*

LDeidb, *LDenaz*

Slika 5.24.

MSKL - Šema modula BP - V.2.

Na osnovu tipova formi *POSL_PARTS* (slika 5.22), *LOKACIJAS* (slika 5.23), *DETALJ_LOKACIJES* (slika 5.24), *MATERIJALS* (slika 5.20), *SKLADIŠTE* (slika 5.15), *SKL_ZONA* (slika 5.16) i *OTPREMNICA* (slika 5.21), projektuje se druga verzija šeme MSKL (S_{m_2}, I_{m_2}), sa skupom šema relacija S_{m_2}:

- *Materijal*({*Matidb, Matozn, Matnaz, Matjus, Matzal*}, {*Matidb*}),
- *Posl_part*({*PPidb, PPnaz, PPadr, PPtel, Adrptt*}, {*PPidb*}),
- *Lokacija*({*Lokidb, Loknaz, Lokadr, Adrptt*}, {*Lokidb*}),
- *Detalj_lokacije*({*Lokidb, LDeidb, LDenaz*}, {*Lokidb* + *LDeidb*}),

- *Skladište*({*Sklidb*, *Sklnaz*, *Lokidb*}, {*Sklidb*}),
- *Skl_zona*({*Sklidb*, *Zonidb*, *Zonnaz*, *ZLoidb*, *LDeidb*}, {*Sklidb* + *Zonidb*}),
- *Skl_dokument*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Sdatiz*, *PPidb*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}),
- *Stavka_dokum*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Stvidb*, *Matidb*, *Stvkol*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Stvidb*, *Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Matidb*})

i skupom međurelacionih ograničenja I_{m_2}:

- *Detalj_lokacije*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*Sklidb*] $\subseteq$ *Skladište*[*Sklidb*],
- *Skladište*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*ZLoidb* + *LDeidb*] $\subseteq$ *Detalj_lokacije*[*Lokidb* + *LDeidb*],
- *Skl_dokument*[*PPidb*] $\subseteq$ *Posl_part*[*PPidb*],
- *Skl_dokument*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_dokum*[*Sklidb*+*Zonidb*+*Posgod*+*Sdvrst* +*Sdtip* + *Sdidbr*] $\subseteq$ *Skl_dokument*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Skl_dokument*[*Sklidb*+*Zonidb*+*Posgod* +*Sdvrst* +*Sdtip* +*Sdidbr*] $\subseteq$ *Stavka_dokum*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*] i
- *Stavka_dokum*[*Matidb*] $\subseteq$ *Materijal*[*Matidb*].

ISBP / MSKL - Ispitivanje stroge kompatibilnosti II

Proces *Stroga_kompatibilnost* (algoritam 4.10) za ulazne šeme BP (S_s, I_s) i (S_{m_2}, I_{m_2}), daje izlazni rezultat *Ind* = *Strue*, što znači da je $(S_s, I_s) \cong (S_{m_2}, I_{m_2})$. Pri tome je *TranC_Rep* = *FnfC_Rep* = *IntcDk_Rep* = *IntCLi_Rep* = *EqKeyC_Rep* = ∅, dok izveštaj *IntCGr_Rep* sadrži poruke (slika 5.25). Šeme BP (S_s, I_s) i (S_{m_2}, I_{m_2}) predstavljaju primer za situaciju u kojoj algoritam 4.8. (odnosno algoritam 4.7) ne indikuje globalnu koliziju međurelacionih ograničenja, a izveštaj *IntCGr_Rep.I-3.* sadrži poruke, tipa '*Moguće nepokriveno međurelaciono ograničenje*'. U nastavku se ovakva situacija detaljnije analizira.

IZVEŠTAJ I-3: ***IntCGr_Rep*** - *Globalna kolizija međurelacionih ograničenja*	
Modul: **Sistemski** (**MSIS**)	Modul: **Skladištenje** (**MSKL**)
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Posl_part*: *R* = {*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*} $\mathcal{K}$ = {*PPidb*}
1. Poruka: *ic*: MSKL.*Posl_part*[*Adrptt*] $\subseteq$ MSIS.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Lokacija*: *R* = {*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*} $\mathcal{K}$ = {*Lokidb*}
2. Poruka: *ic*: MSKL.*Lokacija*[*Adrptt*] $\subseteq$ MSIS.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	

Slika 5.25.

Poruka *IntCGr_Rep.I-3.1.* znači da je moguće narušiti referencijalni integritet MSKL.*Posl_part*[*Adrptt*] $\subseteq$ MSIS.*Adresar*[*Adrptt*], pod pretpostavkom da se dozvoljava ažuriranje relacije nad šemom MSKL.*Posl_part*, iako je postoji šema relacije MSIS.*Posl_part*, odnosno referencijalni integritet MSIS.*Posl_part*[*Adrptt*] $\subseteq$ MSIS.*Adresar*[*Adrptt*].

Slično, poruka *IntCGr_Rep.I-3.2.* znači da je moguće narušiti referencijalni integritet MSKL.*Lokacija*[*Adrptt*] $\subseteq$ MSIS.*Adresar*[*Adrptt*], u slučaju da se dozvoli ažuriranje relacije nad šemom MSKL.*Lokacija*[*Adrptt*], iako postoji šema MSIS.*Lokacija*[*Adrptt*].

Zbog prethodno navedenih razloga, poruke *IntCGr_Rep.I-3.1.* i *IntCGr_Rep.I-3.2.* iniciraju uvođenje sledeće pretpostavke:

P4. U šeme relacija, formirane na osnovu preuzetih (pozajmljenih i/ili preprojektovanih) tipova formi, ne smeju se dodavati novi podaci putem programskog podsistema datog modula.

U konkretnom slučaju, relacije nad šemama *Posl_part* i *Lokacija* se ne smeju ažurirati putem programskog podsistema MSKL, već samo putem programskog podsistema MSIS. Na taj način, poruke *IntCGr_Rep.I-3.1.* i *IntCGr_Rep.I-3.2.* ne dovode do kolizije integrativnog svojstva.

Nepoštovanje pretpostavke P4 bi za posledicu, u najboljem slučaju, imalo preinačenje referencijalnih integriteta:

$$Posl\_part[Adrptt] \subseteq Adresar[Adrptt] \text{ i } Lokacija[Adrptt] \subseteq Adresar[Adrptt]$$

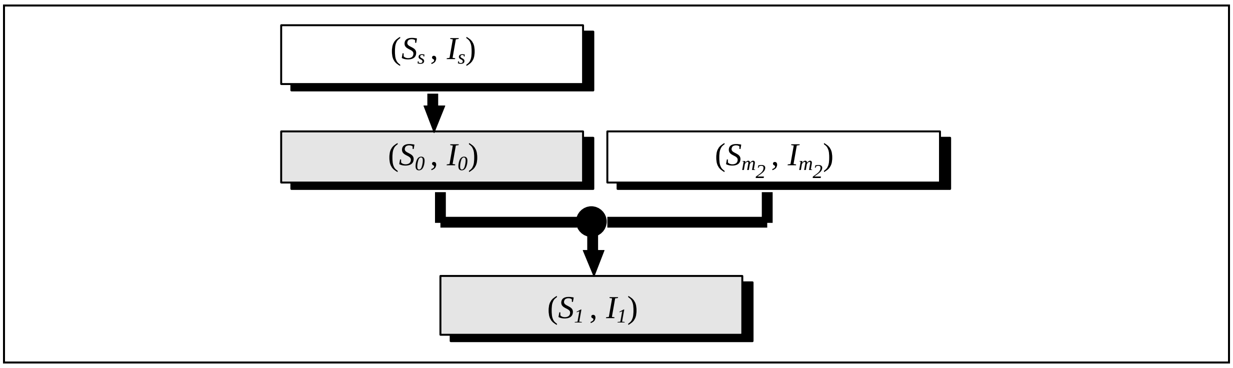
u “slabu” formu, redom:

$$Posl\_part[Adrptt] \setminus \{w\} \subseteq Adresar[Adrptt] \text{ i } Lokacija[Adrptt] \setminus \{w\} \subseteq Adresar[Adrptt],$$

što bi praktično dovelo do kolizije integrativnog svojstva.

ISBP / MSKL - Integracija šeme BP I

Saglasno činjenici da važi $(S_s, I_s) \cong (S_{m_2}, I_{m_2})$, vrši se integracija šeme MSIS, koja ujedno reprezentuje i prvo tekuće stanje šeme BP, sa drugm verzijom šeme MSKL (S_{m_2}, I_{m_2}) (algoritam 4.11). Kao rezultat (slika 5.26), dobija se novo tekuće stanje integrisane šeme BP (ISBP) (S_1, I_1), sa skupom šema relacija S_1:



Slika 5.26.

- *Materijal*({*Matidb, Matozn, Matnaz, Matjus, Matzal*}, {*Matidb*}),
- *Org_jedinica*({*OJidb, OJnaz, OJnadr, Lokidb, LDeidb*}, {*OJidb*}),
- *Adresar*({*Adrptt, Adrmes*}, {*Adrptt*}),

- *Posl_part*({*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*}, {*PPidb*}),
- *Lokacija*({*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*}, {*Lokidb*}),
- *Detalj_lokacije*({*Lokidb*, *LDeidb*, *LDenaz*}, {*Lokidb* + *LDeidb*}),
- *Skladište*({*Sklidb*, *Sklnaz*, *Lokidb*}, {*Sklidb*}),
- *Skl_zona*({*Sklidb*, *Zonidb*, *Zonnaz*, *ZLoidb*, *LDeidb*}, {*Sklidb* + *Zonidb*}),
- *Skl_dokument*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Sdatiz*, *PPidb*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}) i
- *Stavka_dokum*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Stvidb*, *Matidb*, *Stvkol*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Stvidb*, *Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Matidb*})

i skupom međurelacionih ograničenja I_1 :

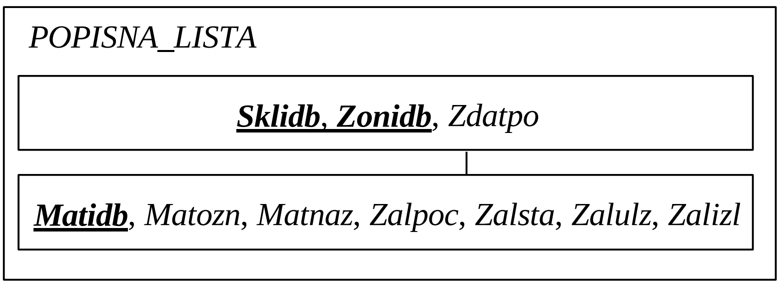
- *Posl_part*[*Adrptt*] $\subseteq$ *Adresar*[*Adrptt*],
- *Org_jedinica*[*Lokidb* + *LDeidb*] $\subseteq$ *Detalj_lokacije*[*Lokidb* + *LDeidb*],
- *Org_jedinica*[*OJnadr*] \ {*w*} $\subseteq$ *Org_jedinica*[*OJidb*],
- *Lokacija*[*Adrptt*] $\subseteq$ *Adresar*[*Adrptt*],
- *Detalj_lokacije*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*Sklidb*] $\subseteq$ *Skladište*[*Sklidb*],
- *Skladište*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*ZLoidb* + *LDeidb*] $\subseteq$ *Detalj_lokacije*[*Lokidb* + *LDeidb*],
- *Skl_dokument*[*PPidb*] $\subseteq$ *Posl_part*[*PPidb*],
- *Skl_dokument*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_dokum*[*Sklidb*+*Zonidb*+*Posgod*+*Sdvrst* +*Sdtip* + *Sdidbr*] $\subseteq$ *Skl_dokument*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Skl_dokument*[*Sklidb*+*Zonidb*+*Posgod* +*Sdvrst* +*Sdtip* +*Sdidbr*] $\subseteq$ *Stavka_dokum*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*] i
- *Stavka_dokum*[*Matidb*] $\subseteq$ *Materijal*[*Matidb*].

MSKL - Inovacija skupa tipova formi

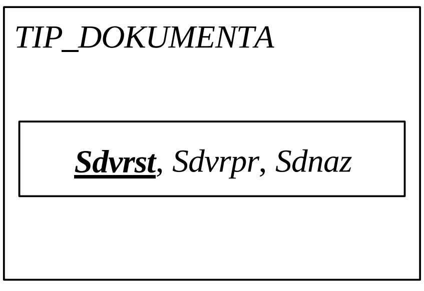
Saglasno novim rezultatima sistem-analize, projektant MSKL zaključuje sledeće:

- potrebno je evidentirati tipove skladišnih dokumenata i
- postoji poseban dokument - popisna lista, kojim se utvrđuje početno stanje zaliha na skladištu.

Na osnovu ovih zaključaka i rezultata sistem-analize, isprojektovana su dva nova tipa forme: *POPISNA_LISTA* (***PL***), slika 5.27. i *TIP_DOKUMENTA* (***TD***), slika 5.28.



Slika 5.27.



Slika 5.28.

Za novododata obeležja je dat opis njihove semantike i specifikacija funkcija *dbs*, *el*, *ren*, *man* i *der*:

- *Zdatpo* - Datum popisa stanja zaliha u zoni skladišta, *dbs*(*Zdatpo*) = *Y*, *el*(*Zdatpo*) = *E*,
- *Zalpoc* - Početno stanje zaliha, *dbs*(*Zalpoc*) = *Y*, *el*(*Zalpoc*) = *E*, *man*(*PL*, *Zalpoc*) = *M*,
- *Zalsta* - Tekuće stanje zaliha, *dbs*(*Zalsta*) = *Y*, *el*(*Zalsta*) = *D*, *man*(*PL*, *Zalsta*) = *M*, *der*(*Zalsta*) = {*Zalpoc*, *Zalulz*, *Zalizl*} (precizno, $Zalsta = Zalpoc + Zalulz - Zalizl \geq 0$),
- *Zalulz* - Ukupan ulaz materijala na zalihe, *dbs*(*Zalulz*) = *Y*, *el*(*Zalulz*) = *D*, *man*(*PL*, *Zalulz*) = *M*, *der*(*Zalulz*) = {*Matidb*, *Sklidb*, *Zonidb*, *Stvkol*} (precizno, $Zalulz = Zalulz + Sdtip * Stvkol$, za ulazne dokumente, pri čemu je početno stanje $Zalulz = 0$),
- *Zalizl* - Ukupan izlaz materijala sa zaliha, *dbs*(*Zalizl*) = *Y*, *el*(*Zalizl*) = *D*, *man*(*PL*, *Zalizl*) = *M*, *der*(*Zalizl*) = {*Matidb*, *Sklidb*, *Zonidb*, *Stvkol*} (precizno, $Zalizl = Zalizl + Sdtip * Stvkol$, za izlazne dokumente, pri čemu je početno stanje $Zalizl = 0$),
- *Sdvrpr* - Vrsta promene skladišnog dokumenta, *dom*(*Zalizl*) = {*1*, *-1*} (*1* - ulazni, *-1* - izlazni dokument), *dbs*(*Sdvrpr*) = *Y*, *el*(*Sdvrpr*) = *E*, *man*(*TD*, *Sdvrpr*) = *M*,
- *Sdnaz* - Naziv skladišnog dokumenta, *dbs*(*Sdnaz*) = *Y*, *el*(*Sdnaz*) = *E*, *man*(*TD*, *Sdnaz*) =*M*.

Pored navedenih definicija, projektant menja specifikaciju funkcije *der* za obeležje *Matzal*:

- *el*(*Matzal*) = *D*, *der*(*Matzal*) = {*Matidb*, *Zalsta*} (precizno *Matzal* predstavlja sumu zaliha materijala u svim zonama, svih skladišta: $Matzal = \sum_{(Sklidb, Zonidb)} Zalsta(Sklidb, Zonidb) \geq 0$,

  odnosno, izračunava se za svaki dokument po formuli $Matzal = Matzal + Sdvrpr * Sdtip * Stvkol$, pri čemu je početna vrednost data kao: $Matzal = \sum_{(Sklidb, Zonidb)} Zalpoc(Sklidb, Zonidb)$).

MSKL - Šema modula BP - V.3.

Na osnovu tipova formi *POSL_PARTS* (slika 5.22), *LOKACIJAS* (slika 5.23), *DETALJ_LOKACIJES* (slika 5.26), *MATERIJALS* (slika 5.20), *SKLADIŠTE* (slika 5.15), *SKL_ZONA* (slika 5.16), *OTPREMNICA* (slika 5.21), *POPISNA_LISTA* (slika 5.27) i *TIP_DOKUMENTA* (slika 5.28), projektuje se treća verzija šeme MSKL (S_{m_3}, I_{m_3}), sa skupom šema relacija S_{m_3}:

- *Materijal*({*Matidb*, *Matozn*, *Matnaz*, *Matjus*, *Matzal*}, {*Matidb*}),
- *Posl_part*({*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*}, {*PPidb*}),
- *Lokacija*({*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*}, {*Lokidb*}),
- *Detalj_lokacije*({*Lokidb*, *LDeidb*, *LDenaz*}, {*Lokidb* + *LDeidb*}),
- *Skladište*({*Sklidb*, *Sklnaz*, *Lokidb*}, {*Sklidb*}),
- *Skl_zona*({*Sklidb*, *Zonidb*, *Zonnaz*, *Zdatpo*, *ZLoidb*, *LDeidb*}, {*Sklidb* + *Zonidb*}),
- *Zalihe*({*Sklidb*, *Zonidb*, *Matidb*, *Zalpoc*, *Zalsta*, *Zalulz*, *Zalizl*}, {*Sklidb* + *Zonidb* + *Matidb*}),
- *Tip_dokumenta*({*Sdvrst*, *Sdvrpr*, *Sdnaz*}, {*Sdvrst*}),
- *Skl_dokument*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Sdatiz*, *PPidb*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}),
- *Stavka_dokum*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Stvidb*, *Matidb*, *Stvkol*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Stvidb*, *Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Matidb*})

i skupom međurelacionih ograničenja I_{m_3}:

- $Detalj\_lokacije[Lokidb] \subseteq Lokacija[Lokidb]$,
- $Skl\_zona[Sklidb] \subseteq Skladište[Sklidb]$,
- $Skladište[Lokidb] \subseteq Lokacija[Lokidb]$,
- $Skl\_zona[ZLoidb + LDeidb] \subseteq Detalj\_lokacije[Lokidb + LDeidb]$,

- *Zalihe*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Zalihe*[*Matidb*] $\subseteq$ *Materijal*[*Matidb*],
- *Skl_dokument*[*Sdvrst*] $\subseteq$ *Tip_dokumenta*[*Sdvrst*],
- *Skl_dokument*[*PPidb*] $\subseteq$ *Posl_part*[*PPidb*],
- *Skl_dokument*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_dokum*[*Sklidb*+*Zonidb*+*Posgod*+*Sdvrst* +*Sdtip* + *Sdidbr*] $\subseteq$ *Skl_dokument*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Skl_dokument*[*Sklidb*+*Zonidb*+*Posgod* +*Sdvrst* +*Sdtip* +*Sdidbr*] $\subseteq$ *Stavka_dokum*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Stavka_dokum*[*Sklidb* + *Zonidb* + *Matidb*] $\subseteq$ *Zalihe*[*Sklidb* + *Zonidb* + *Matidb*] i
- *BR_Nenegativna_zaliha*(*Tip_dokumenta*, *Skl_dokument*, *Stavka_dokum*, *Zalihe*), kao pravilo poslovanja koje je posledica (precizno definisane) funkcije izvođenja *der*(*Zalsta*), *der*(*Zalulz*) i *der*(*Zalizl*), a interpretira se putem formule (u kojoj *Insert*(t_i , *Stavka_dokum*) znači da se t_i upisuje u relaciju *Stavka_dokum*):

  Insert(t_i , *Stavka_dokum*) $\Leftrightarrow$

  $(\forall t \in Stavka\_dokum \rhd\lhd Skl\_dokument \rhd\lhd Tip\_dokumenta)((t[Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr] = t_i[Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr] \wedge t(Sdtip) * t(Sdvrpr) = -1) \Rightarrow$

  $(\forall t_j \in Zalihe)(t_i[Sklidb + Zonidb + Matidb] = t_j[Sklidb + Zonidb + Matidb] \Rightarrow t_i[Stvkol] \leq t_j[Zalsta]))).$

ISBP / MSKL - Ispitivanje stroge kompatibilnosti III

Proces *Stroga_kompatibilnost* za ulazne šeme BP (S_1, I_1) i (S_{m_3}, I_{m_3}), daje izlazni rezultat *Ind* = *Strue*, što znači da je (S_1, I_1) $\cong$ (S_{m_3}, I_{m_3}). Pri tome je *TranC_Rep* = *FnfC_Rep* = *IntcDk_Rep* = *IntCLi_Rep* = *EqKeyC_Rep* = $\emptyset$, dok izveštaj *IntCGr_Rep* sadrži poruke (slika 5.29).

Poruke *IntCGr_Rep.I-4.1.* i *IntCGr_Rep.I-4.2.* se tumače na isti način kao i poruke iz prethodne iteracije *IntCGr_Rep.I-3.1.* i *IntCGr_Rep.I-3.2*, dok poruke *IntCGr_Rep.I-4.3.* i *IntCGr_Rep.I-4.4.* zahtevaju da se eventualno uneti podaci u relacije *Skl_dokument* i *Stavka_dokum* prilikom instaliranja najnovije verzije ISBP (koja će biti prikazana u narednoj tački) dovedu u saglasnost s podacima relacija *Tip_dokumenta* i *Zalihe*, respektivno.

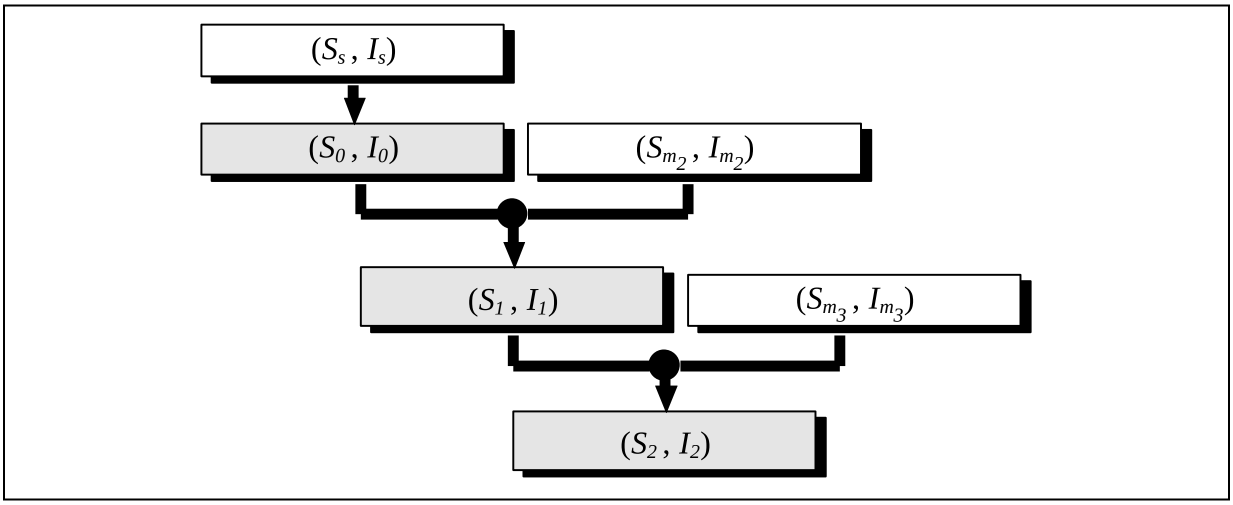
IZVEŠTAJ I-4: *IntCGr_Rep* - *Globalna kolizija međurelacionih ograničenja*	
Integrisana šema BP (**ISBP**)	Modul: **Skladištenje** (**MSKL**)
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Posl_part*: *R* = {*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*} $\mathcal{K}$ = {*PPidb*}
1. Poruka: *ic*: MSKL.*Posl_part*[*Adrptt*] $\subseteq$ ISBP.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Lokacija*: *R* = {*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*} $\mathcal{K}$ = {*Lokidb*}
2. Poruka: *ic*: MSKL.*Lokacija*[*Adrptt*] $\subseteq$ ISBP.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	

<table>
<tr><td>Skl_dokument:
R = {Sklidb, Zonidb, Posgod, Sdvrst, Sdtip, Sdidbr, Sdatiz, PPidb}
$\mathcal{K}$ = {Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr}</td><td>Tip_dokumenta:
R = {Sdvrst, Sdvrpr, Sdnaz}
$\mathcal{K}$ = {Sdvrst}</td></tr>
<tr><td colspan="2">3. Poruka: ic: ISBP.Skl_dokument[Sdvrst] $\subseteq$ MSKL.Tip_dokumenta[Sdvrst]
Moguće nepokriveno međurelaciono ograničenje.</td></tr>
<tr><td>Stavka_dokum:
R = {Sklidb, Zonidb, Posgod, Sdvrst, Sdtip, Sdidbr, Stvidb, Matidb, Stvkol}
$\mathcal{K}$ = {Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr + Stvidb, Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr + Matidb}</td><td>Zalihe:
R = {Sklidb, Zonidb, Matidb, Zalpoc, Zalsta, Zalulz, Zalizl}
$\mathcal{K}$ = {Sklidb + Zonidb + Matidb}</td></tr>
<tr><td colspan="2">4. Poruka: ic:
ISBP.Stavka_dokum[Sklidb + Zonidb + Matidb] $\subseteq$ MSKL.Zalihe[Sklidb + Zonidb + Matidb]
Moguće nepokriveno međurelaciono ograničenje.</td></tr>
</table>

Slika 5.29.

ISBP / MSKL - Integracija šeme BP II

Saglasno činjenici da važi $(S_1, I_1) \cong (S_{m_3}, I_{m_3})$, vrši se integracija tekuće verzije ISBP sa trećom verzijom šeme MSKL (S_{m_3}, I_{m_3}). Kao rezultat (slika 5.30), dobija se novo tekuće stanje ISBP (S_2, I_2), sa skupom šema relacija S_2:



Slika 5.30.

- *Materijal*({*Matidb*, *Matozn*, *Matnaz*, *Matjus*, *Matzal*}, {*Matidb*}),
- *Org_jedinica*({*OJidb*, *OJnaz*, *OJnadr*, *Lokidb*, *LDeidb*}, {*OJidb*}),
- *Adresar*({*Adrptt*, *Adrmes*}, {*Adrptt*}),
- *Posl_part*({*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*}, {*PPidb*}),
- *Lokacija*({*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*}, {*Lokidb*}),
- *Detalj_lokacije*({*Lokidb*, *LDeidb*, *LDenaz*}, {*Lokidb* + *LDeidb*}),
- *Skladište*({*Sklidb*, *Sklnaz*, *Lokidb*}, {*Sklidb*}),
- *Skl_zona*({*Sklidb*, *Zonidb*, *Zonnaz*, *Zdatpo*, *ZLoidb*, *LDeidb*}, {*Sklidb* + *Zonidb*}),

- *Zalihe*({*Sklidb*, *Zonidb*, *Matidb*, *Zalpoc*, *Zalsta*, *Zalulz*, *Zalizl*}, {*Sklidb* + *Zonidb* + *Matidb*}),
- *Tip_dokumenta*({*Sdvrst*, *Sdvrpr*, *Sdnaz*}, {*Sdvrst*}),
- *Skl_dokument*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Sdatiz*, *PPidb*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}),
- *Stavka_dokum*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Stvidb*, *Matidb*, *Stvkol*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Stvidb*, *Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Matidb*})

i skupom međurelacionih ograničenja I_2:

- *Posl_part*[*Adrptt*] $\subseteq$ *Adresar*[*Adrptt*],
- *Org_jedinica*[*Lokidb* + *LDeidb*] $\subseteq$ *Detalj_lokacije*[*Lokidb* + *LDeidb*],
- *Org_jedinica*[*OJnadr*] \ {*w*} $\subseteq$ *Org_jedinica*[*OJidb*],
- *Lokacija*[*Adrptt*] $\subseteq$ *Adresar*[*Adrptt*],
- *Detalj_lokacije*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*Sklidb*] $\subseteq$ *Skladište*[*Sklidb*],
- *Skladište*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*ZLoidb* + *LDeidb*] $\subseteq$ *Detalj_lokacije*[*Lokidb* + *LDeidb*],
- *Zalihe*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Zalihe*[*Matidb*] $\subseteq$ *Materijal*[*Matidb*],
- *Skl_dokument*[*Sdvrst*] $\subseteq$ *Tip_dokumenta*[*Sdvrst*],
- *Skl_dokument*[*PPidb*] $\subseteq$ *Posl_part*[*PPidb*],
- *Skl_dokument*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_dokum*[*Sklidb*+*Zonidb*+*Posgod*+*Sdvrst* +*Sdtip* + *Sdidbr*] $\subseteq$ *Skl_dokument*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Skl_dokument*[*Sklidb*+*Zonidb*+*Posgod* +*Sdvrst* +*Sdtip* +*Sdidbr*] $\subseteq$ *Stavka_dokum*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Stavka_dokum*[*Sklidb* + *Zonidb* + *Matidb*] $\subseteq$ *Zalihe*[*Sklidb* + *Zonidb* + *Matidb*] i
- *BR_Nenegativna_zaliha*(*Tip_dokumenta*, *Skl_dokument*, *Stavka_dokum*, *Zalihe*).

ISBP - zajednički tip forme modula PRODAJA (MPRO)

Projektanti modula "***Prodaja***" (**MPRO**) i MSKL su, tokom sistem-analize, zaključili da se u okviru funkcije "Prodaja" formira dokument NALOG_ZA_OTPREMU, kojim se, na osnovu porudžbenice, ili ugovora s kupcem, izdaje nalog za otpremu robe iz skladišta. U okviru funkcije "Skladištenje", otpremnice se formiraju isključivo na osnovu prethodno generisanog naloga za otpremu, pri čemu jedan nalog za otpremu može biti realizovan putem jedne, ili više otpremnica. To znači da NALOG_ZA_OTPREMU predstavlja zajednički dokument. Projektant MPRO pravi tip forme *NALOG_ZA_OTPREMU* (***NO***), prikazan na slici 5.31. *NALOG_ZA_OTPREMU* treba, pri tome, da predstavlja zajednički tip forme s obzirom na MSKL.

Definicije (novododatih) obeležja:

- *Nalidb* - Identifikaciona oznaka naloga za otpremu, *dbs*(*Nalidb*) = *Y*, *el*(*Nalidb*) = *E*,
- *man*(*NO*, *Sdatiz*) = *M*,
- *man*(*NO*, *PPidb*) = *M*,
- *Stnidb* - Identifikacioni broj stavke naloga za otpremu, *dbs*(*Stnidb*) = *Y*, *el*(*Stnidb*) = *E* i
- *man*(*NO*, *Stvkol*) = *M*.

NALOG_ZA_OTPREMU

<u>Nalidb</u>, *Sklidb, Zonidb, Sdatiz, PPidb, PPnaz*

<u>Stnidb</u>, ***<u>Matidb</u>***, *Matozn, Matnaz, Stvkol*

Slika 5.31.

MPRO - Šema modula BP - V.1.

Na osnovu preuzetih tipova formi *POSL_PARTS* (slika 5.22) i *MATERIJALS* (slika 5.20), kao i na osnovu tipa forme *NALOG_ZA_OTPREMU* (slika 5.31), projektuje se prva verzija šeme MPRO (S_{p_1}, I_{p_1}), sa skupom šema relacija S_{p_1}:

- *Materijal*({*Matidb, Matozn, Matnaz, Matjus, Matzal*}, {*Matidb*}),
- *Posl_part*({*PPidb, PPnaz, PPadr, PPtel, Adrptt*}, {*PPidb*}),
- *Nalog_za_otpremu*({*Nalidb, Sklidb, Zonidb, Sdatiz, PPidb*}, {*Nalidb*}),
- *Stavka_naloga*({*Nalidb, Stnidb, Matidb, Stvkol*}, {*Nalidb + Stnidb, Nalidb + Matidb*})

i skupom međurelacionih ograničenja I_{p_1}:

- *Nalog_za_otpremu*[*PPidb*] $\subseteq$ *Posl_part*[*PPidb*],
- *Stavka_naloga*[*Matidb*] $\subseteq$ *Materijal*[*Matidb*],
- *Stavka_naloga*[*Nalidb*] $\subseteq$ *Nalog_za_otpremu*[*Nalidb*] i
- *Nalog_za_otpremu*[*Nalidb*] $\subseteq$ *Stavka_naloga*[*Nalidb*].

ISBP / MPRO - Ispitivanje stroge kompatibilnosti I

Proces *Stroga_kompatibilnost* za ulazne šeme BP (S_2, I_2) i (S_{p_1}, I_{p_1}), daje izlazni rezultat *Ind* = *Itrue*, što znači da je (S_2, I_2) $\cong_I$ (S_{p_1}, I_{p_1}).

<table>
<tr><td colspan="2">IZVEŠTAJ I-5: TranC_Rep - Kolizija tranzitivnosti</td></tr>
<tr><td>Integrisana šema BP (ISBP)</td><td>Modul: Prodaja (MPRO)</td></tr>
<tr><td>Skl_dokument:
$\mathcal{K}$ = {Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr}</td><td>Nalog_za_otpremu:
$\mathcal{K}$ = {Nalidb}</td></tr>
<tr><td colspan="2">1. Upozorenje: ISBP.Skl_dokument $\not\leftrightarrow$ MPRO.Nalog_za_otpremu
TrAttr = {Sdatiz, PPidb}
Mogući homonimi.</td></tr>
<tr><td>Stavka_dokum:
$\mathcal{K}$ = {Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr + Stvidb, Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr + Matidb}</td><td>Stavka_naloga:
$\mathcal{K}$ = {Nalidb + Stnidb, Nalidb + Matidb}</td></tr>
<tr><td colspan="2">2. Upozorenje: ISBP.Stavka_dokum $\not\leftrightarrow$ MPRO.Stavka_naloga
TrAttr = {Stvkol}
Mogući homonimi.</td></tr>
</table>

Slika 5.32.

Formiraju se sledeći izveštaji o detektovanim kolizijama:

- za koliziju tranzitivnosti *TranC*(S_2, S_{p_1}), formira se izveštaj *TranC_Rep* (slika 5.32) i
- za globalnu koliziju međurelacionih ograničenja *IntCGr*(S_2, S_{p_1}) formira se izveštaj *IntCGr_Rep* (slika 5.33),

dok je *FnfC_Rep* = *IntcDk_Rep* = *IntCLi_Rep* = *EqKeyC_Rep* = ∅.

IZVEŠTAJ I-6: *IntCGr_Rep* - *Globalna kolizija međurelacionih ograničenja*	
Integrisana šema BP (**ISBP**)	Modul: **Prodaja** (**MPRO**)
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Posl_part*: *R* = {*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*} $\mathcal{K}$ = {*PPidb*}
1. Poruka: *ic*: MPRO.*Posl_part*[*Adrptt*] ⊆ ISBP.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	
Zalihe: *R* = {*Sklidb*, *Zonidb*, *Matidb*, *Zalpoc*, *Zalsta*, *Zalulz*, *Zalizl*} $\mathcal{K}$ = {*Sklidb* + *Zonidb* + *Matidb*}	*Stavka_naloga*: *R* = {*Nalidb*, *Stnidb*, *Matidb*, *Stvkol*} $\mathcal{K}$ = {*Nalidb* + *Stnidb*, *Nalidb* + *Matidb*}
2. Upozorenje: *ic*: MPRO.(*Stavka_naloga* ⊳⊲ *Nalog_za_otpremu*)[*Sklidb* + *Zonidb* + *Matidb*] ⊆ ISBP.*Zalihe*[*Sklidb* + *Zonidb* + *Matidb*][*)] *Nepokriveno međurelaciono ograničenje.*	

Slika 5.33.

MSKL / MPRO - Otklanjanje kolizija

Bez obzira što je testom kompatibilnosti utvrđeno da važi (S_2, I_2) $\cong_I$ (S_{p_1}, I_{p_1}), upozorenja i poruke izveštaja *TranC_Rep.I-5.* i *IntCGr_Rep.I-6.* ukazuju na potrebu otklanjanja određenih kolizija, koje se u nastavku diskutuju.

Upozorenje *TranC_Rep.I-5.1.* (slika 5.32) ukazuje na dve činjenice:

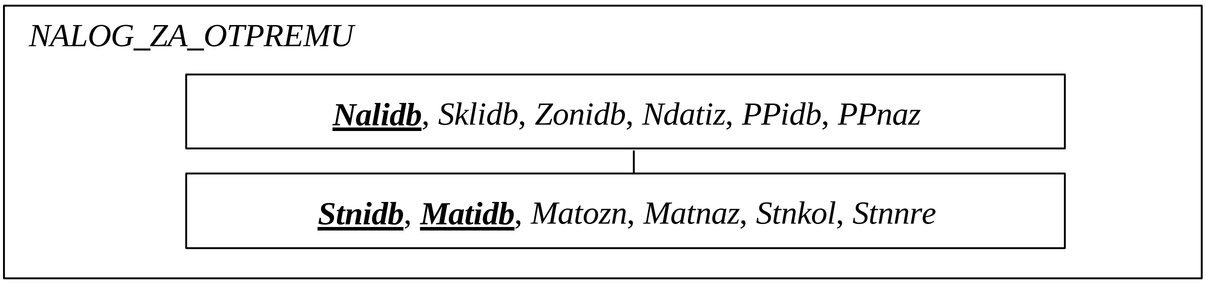
- Obeležje *Sdatiz* je homonim, jer jednom predstavlja datum izdavanja naloga za otpremu robe, a drugi put predstavlja datum izdavanja otpremnice. Zbog toga se, umesto obeležja *Sdatiz* uvodi obeležje *Ndatiz* (videti definiciju obeležja, uz sliku 5.34). Pored toga, menja se specifikacija funkcije *el* za obeležje *Sdatiz* (videti definiciju obeležja, uz sliku 5.35).
- Obeležje *PPidb* nije homonim, pošto se isti poslovni partner pojavljuje i na nalogu za otpremu i na otpremnici. Ova situacija bi nalagala definisanje posebnog međurelacionog ograničenja, koje bi obezbedilo kontrolu unosa obeležja *PPidb* na otpremnici, tako da odgovara unetoj vrednosti za *PPidb* na nalogu za otpremu. Ukoliko se, međutim, uzme u obzir i činjenica da se svaka otpremnica odnosi na tačno jedan nalog za otpremu, tada je potrebno preprojektovati tip forme *OTPREMNICA*, tako što će se u zaglavlje uvesti obeležje *Nalidb* i ukloniti obeležje *PPidb*. Obeležje *PPidb* se uklanja sa tipa forme *OT-*

[*)] Algoritam 4.7. bi, formalno posmatrano, umesto navedenog pravila poslovanja, u okviru ovog upozorenja, prikazao poruku *Stavka_naloga*[*Sklidb* + *Zonidb* + *Matidb*] ⊆ ISBP.*Zalihe*[*Sklidb* + *Zonidb* + *Matidb*], što je posledica Činjenice da navedeni algoritam tretira iskljuČivo problem meðusobnog zatvaranja ***parova*** šema relacija. Uz određene modifikacije algoritma 4.7, moguæe je automatski pružiti punu informaciju o pravilima poslovanja, analognim ovde posmatranom pravilu.

PREMNICA u cilju izbegavanja kolizije tranzitivnosti oblika ISBP.*Skl_dokument* → MPRO.*Nalog_za_otpremu*, do koje bi, inače, došlo zbog dodavanja obeležja *Nalidb* na tip forme *OTPREMNICA*. Umesto *PPidb*, uvodi se izvedeno obeležje *PPido* (slika 5.35), jer korisnik očekuje da se na otpremnici pojave osnovni podaci o poslovnom partneru.

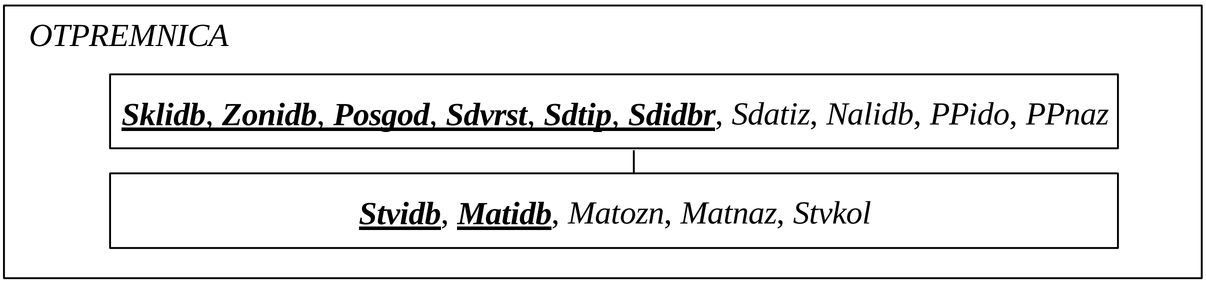
Upozorenje *TranC_Rep.I-5.2.* (slika 5.32) ukazuje na činjenicu da je obeležje *Stvkol* homonim (pošto količina robe na stavci otpremnice i količina robe na stavci naloga za otpremu ne moraju biti iste), te se zbog toga na tipu forme *NALOG_ZA_OTPREMU* uvodi novo obeležje *Stnkol* (videti definiciju obeležja, uz sliku 5.34) i menja se definicija obeležja *Stvkol* na tipu forme *OTPREMNICA* (videti definiciju obeležja, uz sliku 5.35).

Poruka *IntCGr_Rep.I-6.1.* ne dovodi do kolizije integrativnog svojstva, ukoliko se, saglasno pretpostavci P4, zabrani ažuriranje evidencije poslovnih partnera u okviru MPRO. Upozorenje *IntCGr_Rep.I-6.2.* ukazuje na potrebu preuzimanja tipa forme *POPISNA_LISTA* u skup tipova formi MPRO, kao zajedničkog tipa forme, u cilju obezbeđenja uvida u stanje zaliha robe pri kreiranju (ili ažuriranju) naloga za otpremu.



Slika 5.34.

- *Ndatiz* - Datum izdavanja naloga za otpremu, *dbs*(*Ndatiz*) = *Y*, *el*(*Ndatiz*) = *E*, *man*(*NO*, *Ndatiz*) = *M*,
- *Stnkol* - Količina materijala na stavci naloga za otpremu, *dbs*(*Stnkol*) = *Y*, *el*(*Stnkol*) = *E*, *man*(*NO*, *Stnkol*) = *M*,
- *Stnnre* - Neisporučena količina materijala na stavci naloga za otpremu, *dbs*(*Stnnre*) = *Y*, *el*(*Stnnre*) = *D*, *der*(*Stnnre*) = {*Stvkol*} (precizno, *Stnnre* = *Stnnre* - *Stvkol* ≥ *0*, pri čemu je početna vrednost za *Stnnre* data kao *Stnnre* = *Stnkol*).



Slika 5.35.

- *Sdatiz* - Datum izdavanja otpremnice, *dbs*(*Sdatiz*) = *Y*, *el*(*Sdatiz*) = *D*, *der*(*Sdatiz*) = {*Nalidb*, *Ndatiz*} (precizno, za zadati *Nalidb* mora biti *Sdatiz* ≥ *Ndatiz*),
- *PPido* - ID oznaka poslovnog partnera kojem se isporučuje roba po nalogu za otpremu, *dbs*(*PPido*) = *N*, *el*(*PPido*) = *D*, *der*(*PPido*) = {*Nalidb*, *PPidb*},
- *Stvkol* - Količina materijala na stavci otpremnice, *dbs*(*Stvkol*) = *Y*, *el*(*Stvkol*) = *E*.

Zbog definicije funkcije izvođenja *der* za obeležja *Sdatiz*, *PPido* i *Stnnre* (što je posledica činjenice da *NALOG_ZA_OTPREMU* predstavlja zajednički tip forme) u skup tipova formi MSKL će biti preuzet i tip forme *NALOG_ZA_OTPREMU* (kako bi se izbegle *IntCGr* kolizije).

MSKL - Šema modula BP - V.4.

Na osnovu tipova formi *POSL_PARTS* (slika 5.22), *LOKACIJAS* (slika 5.23), *DETALJ_LOKACIJES* (slika 5.24), *MATERIJALS* (slika 5.20), *SKLADIŠTE* (slika 5.15), *SKL_ZONA* (slika 5.16), *OTPREMNICA* (slika 5.35), *POPISNA_LISTA* (slika 5.27), *TIP_DOKUMENTA* (slika 5.28) i *NALOG_ZA_OTPREMU* (slika 5.34), projektuje se četvrta verzija šeme MSKL (S_{m_4}, I_{m_4}), sa skupom šema relacija S_{m_4}:

- *Materijal*({*Matidb*, *Matozn*, *Matnaz*, *Matjus*, *Matzal*}, {*Matidb*}),
- *Posl_part*({*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*}, {*PPidb*}),
- *Lokacija*({*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*}, {*Lokidb*}),
- *Detalj_lokacije*({*Lokidb*, *LDeidb*, *LDenaz*}, {*Lokidb* + *LDeidb*}),
- *Skladište*({*Sklidb*, *Sklnaz*, *Lokidb*}, {*Sklidb*}),
- *Skl_zona*({*Sklidb*, *Zonidb*, *Zonnaz*, *Zdatpo*, *ZLoidb*, *LDeidb*}, {*Sklidb* + *Zonidb*}),
- *Zalihe*({*Sklidb*, *Zonidb*, *Matidb*, *Zalpoc*, *Zalsta*, *Zalulz*, *Zalizl*}, {*Sklidb* + *Zonidb* + *Matidb*}),
- *Tip_dokumenta*({*Sdvrst*, *Sdvrpr*, *Sdnaz*}, {*Sdvrst*}),
- *Skl_dokument*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Sdatiz*, *Nalidb*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}),
- *Stavka_dokum*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Stvidb*, *Matidb*, *Stvkol*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Stvidb*, *Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Matidb*}),
- *Nalog_za_otpremu*({*Nalidb*, *Sklidb*, *Zonidb*, *Ndatiz*, *PPidb*}, {*Nalidb*}),
- *Stavka_naloga*({*Nalidb*, *Stnidb*, *Matidb*, *Stnkol*}, {*Nalidb* + *Stnidb*, *Nalidb* + *Matidb*})

i skupom međurelacionih ograničenja I_{m_4}:

- *Detalj_lokacije*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*Sklidb*] $\subseteq$ *Skladište*[*Sklidb*],
- *Skladište*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*ZLoidb* + *LDeidb*] $\subseteq$ *Detalj_lokacije*[*Lokidb* + *LDeidb*],
- *Zalihe*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Zalihe*[*Matidb*] $\subseteq$ *Materijal*[*Matidb*],
- *Skl_dokument*[*Sdvrst*] $\subseteq$ *Tip_dokumenta*[*Sdvrst*],
- *Skl_dokument*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_dokum*[*Sklidb*+*Zonidb*+*Posgod*+*Sdvrst* +*Sdtip* + *Sdidbr*] $\subseteq$ *Skl_dokument*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Skl_dokument*[*Sklidb*+*Zonidb*+*Posgod* +*Sdvrst* +*Sdtip* +*Sdidbr*] $\subseteq$ *Stavka_dokum*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Stavka_dokum*[*Sklidb* + *Zonidb* + *Matidb*] $\subseteq$ *Zalihe*[*Sklidb* + *Zonidb* + *Matidb*],
- *BR_Nenegativna_zaliha*(*Tip_dokumenta*, *Skl_dokument*, *Stavka_dokum*, *Zalihe*),

- *Skl_dokument*[*Nalidb* + *Sklidb* + *Zonidb*] $\subseteq$ *Nalog_za_otpremu*[*Nalidb* + *Sklidb* + *Zonidb*][*)],
- (*Stavka_naloga* $\bowtie$ *Nalog_za_otpremu*)[*Sklidb* + *Zonidb* + *Matidb*] $\subseteq$ *Zalihe*[*Sklidb* + *Zonidb* + *Matidb*],
- *Nalog_za_otpremu*[*PPidb*] $\subseteq$ *Posl_part*[*PPidb*],
- *Nalog_za_otpremu*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_naloga*[*Nalidb*] $\subseteq$ *Nalog_za_otpremu*[*Nalidb*],
- *Nalog_za_otpremu*[*Nalidb*] $\subseteq$ *Stavka_naloga*[*Nalidb*],
- *BR_Nenegativna_stavka_nal*(*Nalog_za_otpremu*, *Stavka_dokum*, *Stavka_naloga*), s interpretacijom (u kojoj *Insert*(t_i , *Stavka_dokum*) predstavlja pokušaj dodavanja nove torke u relaciju *Stavka_dokum*):

  $Insert(t_i\,,\ Stavka\_dokum) \Leftrightarrow$

  $(\forall t_j \in Nalog\_za\_otpremu \bowtie Stavka\_naloga)(t_i[Sklidb + Zonidb + Matidb] = t_j[Sklidb + Zonidb + Matidb] \Rightarrow t_i[Stvkol] \le t_j[Stnnre])$.

ISBP / MSKL - Ispitivanje stroge kompatibilnosti IV

Proces *Stroga_kompatibilnost* za ulazne šeme BP (S_2, I_2) i (S_{m_4}, I_{m_4}), daje izlazni rezultat *Ind* = *false*, što znači da je (S_2, I_2) $\ncong_I$ (S_{m_4}, I_{m_4}) i formira sledeće izveštaje o detektovanim kolizijama:

- za koliziju tranzitivnosti *TranC*(S_2, S_{m_4}), formira se izveštaj *TranC_Rep* (slika 5.36) i
- za globalnu koliziju međurelacionih ograničenja *IntCGr*(S_2, S_{m_4}) formira se izveštaj *IntCGr_Rep* (slika 5.37),

dok je *FnfC_Rep* = *IntcDk_Rep* = *IntCLi_Rep* = *EqKeyC_Rep* = $\emptyset$.

Greška *TranC_Rep.I-7.1.* je posledica činjenice da u integrisanoj šemi (S_2, I_2) egzistira stara verzija šeme relacije *Skl_dokument* (koja sadrži u sebi obeležje *PPidb*). To znači da se postupak integracije mora vratiti do one tačke gde je prvi put u integrisanu šemu BP dodata šema relacije *Skl_dokument*, i zatim, ponovo, započeti postupak integracije od te tačke (slika 5.38). Tumačenje poruka *IntCGr_Rep.I-8.1.* i *IntCGr_Rep.I-8.2.* je već dato u prethodnom tekstu.

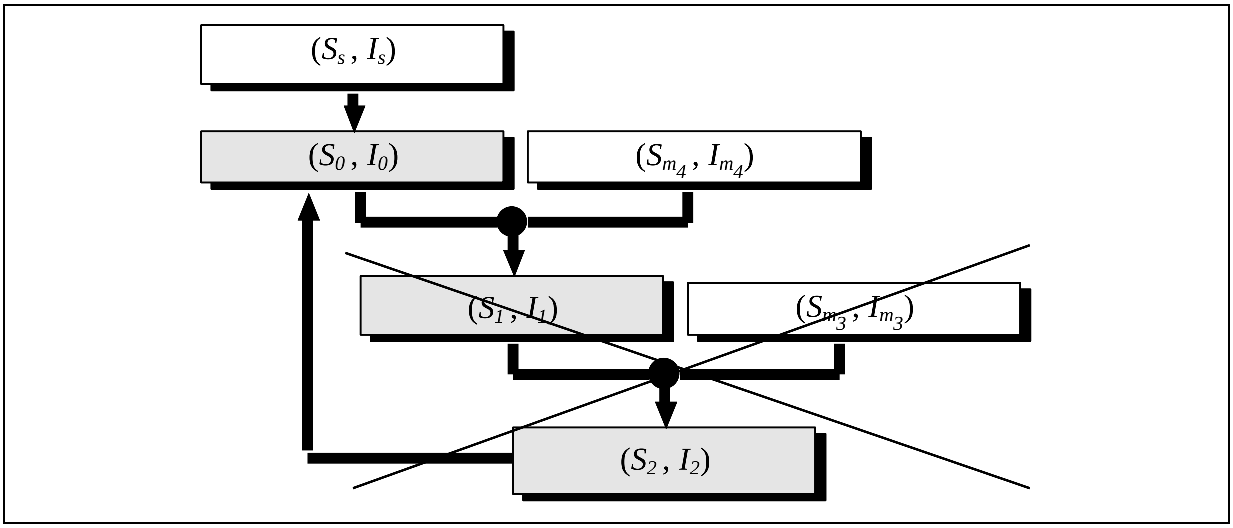
IZVEŠTAJ I-7: ***TranC_Rep*** - *Kolizija tranzitivnosti*	
Integrisana šema BP (**ISBP**)	Modul: **Skladištenje** (**MSKL**)
Skl_dokument: $\mathcal{K}$ = {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}	*Nalog_za_otpremu*: $\mathcal{K}$ = {*Nalidb*}
1. Greška: ISBP.*Skl_dokument* $\rightarrow$ MSKL.*Nalog_za_otpremu* *TrAttr* = {*PPidb*} *Tranzitivna obeležja u šemi relacije:* ISBP.*Skl_dokument*.	

Slika 5.36.

[*)] Ova zavisnost sadržavanja je posledica Činjenice da je data šema modula B-zavisna (prema **[Sc]**), jer se primarna obeležja *Sklidb* i *Zonidb* šeme relacije *Skl_dokument* javljaju kao neprimarna obeležja šeme relacije *Nalog_za_otpremu*, pri Čemu šema *Skl_dokument* zatvara šemu *Nalog_za_otpremu*, zbog propagacije ključa *Nalidb* u šemu *Skl_dokument*.

IZVEŠTAJ I-8: ***IntCGr_Rep*** - *Globalna kolizija međurelacionih ograničenja*	
Integrisana šema BP (**ISBP**)	Modul: **Skladištenje** (**MSKL**)
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Posl_part*: *R* = {*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*} $\mathcal{K}$ = {*PPidb*}
1. Poruka: *ic*: MSKL.*Posl_part*[*Adrptt*] ⊆ ISBP.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Lokacija*: *R* = {*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*} $\mathcal{K}$ = {*Lokidb*}
2. Poruka: *ic*: MSKL.*Lokacija*[*Adrptt*] ⊆ ISBP.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	

Slika 5.37.



Slika 5.38.

IZVEŠTAJ I-9: ***IntCGr_Rep*** - *Globalna kolizija međurelacionih ograničenja*	
Integrisana šema BP (**ISBP**)	Modul: **Skladištenje** (**MSKL**)
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Posl_part*: *R* = {*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*} $\mathcal{K}$ = {*PPidb*}
1. Poruka: *ic*: MSKL.*Posl_part*[*Adrptt*] ⊆ ISBP.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Lokacija*: *R* = {*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*} $\mathcal{K}$ = {*Lokidb*}
2. Poruka: *ic*: MSKL.*Lokacija*[*Adrptt*] ⊆ ISBP.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	

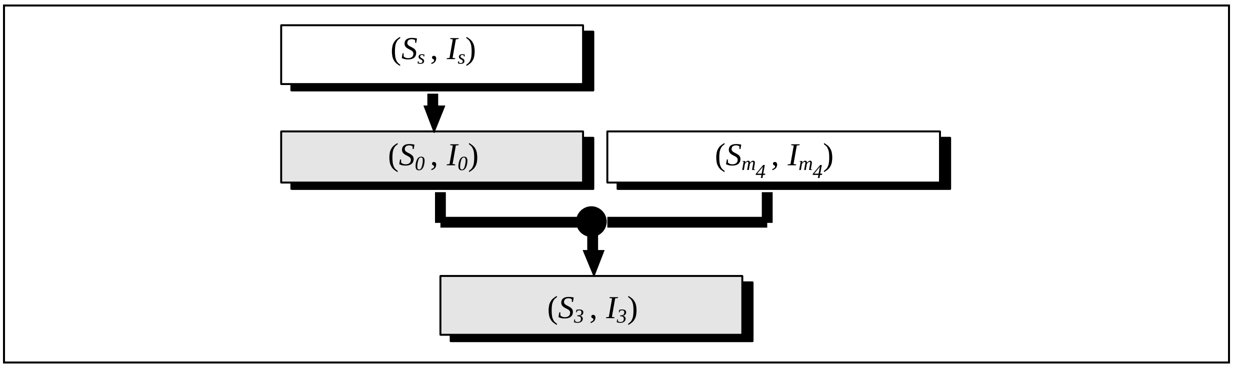
Slika 5.39.

ISBP / MSKL - Ispitivanje stroge kompatibilnosti V

Proces *Stroga_kompatibilnost* za ulazne šeme BP (S_0, I_0) i (S_{m_4}, I_{m_4}), daje izlazni rezultat *Ind* = *Strue*, što znači da je (S_0, I_0) $\cong$ (S_{m_4}, I_{m_4}), pri čemu je *TranC_Rep* = *FnfC_Rep* = *IntcDk_Rep* = *IntCLi_Rep* = *EqKeyC_Rep* = $\emptyset$, a izveštaj *IntCGr_Rep* sadrži poruke (slika 5.39), čije značenje je prethodno diskutovano.

ISBP / MSKL - Integracija šeme BP III

Saglasno činjenici da važi (S_0, I_0) $\cong$ (S_{m_4}, I_{m_4}), vrši se integracija tekuće verzije ISBP sa trećom verzijom šeme MSKL (S_{m_4}, I_{m_4}). Kao rezultat (slika 5.40), dobija se novo tekuće stanje ISBP (S_3, I_3), sa skupom šema relacija S_3:



Slika 5.40.

- *Org_jedinica*({*OJidb*, *OJnaz*, *OJnadr*, *Lokidb*, *LDeidb*}, {*OJidb*}),
- *Adresar*({*Adrptt*, *Adrmes*}, {*Adrptt*}),
- *Materijal*({*Matidb*, *Matozn*, *Matnaz*, *Matjus*, *Matzal*}, {*Matidb*}),
- *Posl_part*({*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*}, {*PPidb*}),
- *Lokacija*({*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*}, {*Lokidb*}),
- *Detalj_lokacije*({*Lokidb*, *LDeidb*, *LDenaz*}, {*Lokidb* + *LDeidb*}),
- *Skladište*({*Sklidb*, *Sklnaz*, *Lokidb*}, {*Sklidb*}),
- *Skl_zona*({*Sklidb*, *Zonidb*, *Zonnaz*, *Zdatpo*, *ZLoidb*, *LDeidb*}, {*Sklidb* + *Zonidb*}),
- *Zalihe*({*Sklidb*, *Zonidb*, *Matidb*, *Zalpoc*, *Zalsta*, *Zalulz*, *Zalizl*}, {*Sklidb* + *Zonidb* + *Matidb*}),
- *Tip_dokumenta*({*Sdvrst*, *Sdvrpr*, *Sdnaz*}, {*Sdvrst*}),
- *Skl_dokument*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Sdatiz*, *Nalidb*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*}),
- *Stavka_dokum*({*Sklidb*, *Zonidb*, *Posgod*, *Sdvrst*, *Sdtip*, *Sdidbr*, *Stvidb*, *Matidb*, *Stvkol*}, {*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Stvidb*, *Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr* + *Matidb*}),
- *Nalog_za_otpremu*({*Nalidb*, *Sklidb*, *Zonidb*, *Ndatiz*, *PPidb*}, {*Nalidb*}),
- *Stavka_naloga*({*Nalidb*, *Stnidb*, *Matidb*, *Stnkol*}, {*Nalidb* + *Stnidb*, *Nalidb* + *Matidb*})

i skupom međurelacionih ograničenja I_3:

- *Posl_part*[*Adrptt*] $\subseteq$ *Adresar*[*Adrptt*],
- *Org_jedinica*[*Lokidb* + *LDeidb*] $\subseteq$ *Detalj_lokacije*[*Lokidb* + *LDeidb*],
- *Org_jedinica*[*OJnadr*] \ {*w*} $\subseteq$ *Org_jedinica*[*OJidb*],
- *Lokacija*[*Adrptt*] $\subseteq$ *Adresar*[*Adrptt*],

- *Detalj_lokacije*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*Sklidb*] $\subseteq$ *Skladište*[*Sklidb*],
- *Skladište*[*Lokidb*] $\subseteq$ *Lokacija*[*Lokidb*],
- *Skl_zona*[*ZLoidb* + *LDeidb*] $\subseteq$ *Detalj_lokacije*[*Lokidb* + *LDeidb*],
- *Zalihe*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Zalihe*[*Matidb*] $\subseteq$ *Materijal*[*Matidb*],
- *Skl_dokument*[*Sdvrst*] $\subseteq$ *Tip_dokumenta*[*Sdvrst*],
- *Skl_dokument*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_dokum*[*Sklidb*+*Zonidb*+*Posgod*+*Sdvrst* +*Sdtip* + *Sdidbr*] $\subseteq$ *Skl_dokument*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Skl_dokument*[*Sklidb*+*Zonidb*+*Posgod* +*Sdvrst* +*Sdtip* +*Sdidbr*] $\subseteq$ *Stavka_dokum*[*Sklidb* + *Zonidb* + *Posgod* + *Sdvrst* + *Sdtip* + *Sdidbr*],
- *Stavka_dokum*[*Sklidb* + *Zonidb* + *Matidb*] $\subseteq$ *Zalihe*[*Sklidb* + *Zonidb* + *Matidb*],
- *BR_Nenegativna_zaliha*(*Tip_dokumenta*, *Skl_dokument*, *Stavka_dokum*, *Zalihe*),
- *Skl_dokument*[*Nalidb* + *Sklidb* + *Zonidb*] $\subseteq$ *Nalog_za_otpremu*[*Nalidb* + *Sklidb* + *Zonidb*],
- (*Stavka_naloga* $\rhd\lhd$ *Nalog_za_otpremu*)[*Sklidb* + *Zonidb* + *Matidb*] $\subseteq$ *Zalihe*[*Sklidb* + *Zonidb* + *Matidb*],
- *Nalog_za_otpremu*[*PPidb*] $\subseteq$ *Posl_part*[*PPidb*],
- *Nalog_za_otpremu*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_naloga*[*Nalidb*] $\subseteq$ *Nalog_za_otpremu*[*Nalidb*],
- *Nalog_za_otpremu*[*Nalidb*] $\subseteq$ *Stavka_naloga*[*Nalidb*] i
- *BR_Nenegativna_stavka_nal*(*Nalog_za_otpremu*, *Stavka_dokum*, *Stavka_naloga*).

MPRO - Šema modula BP - V.2.

Na osnovu preuzetih tipova formi *POSL_PARTS* (slika 5.22), *MATERIJALS* (slika 5.20), *NALOG_ZA_OTPREMU* (slika 5.34), i *POPISNA_LISTA* (slika 5.27), projektuje se druga verzija šeme MPRO (S_{p_2}, I_{p_2}), sa skupom šema relacija S_{p_2}:

- *Materijal*({*Matidb*, *Matozn*, *Matnaz*, *Matjus*, *Matzal*}, {*Matidb*}),
- *Posl_part*({*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*}, {*PPidb*}),
- *Nalog_za_otpremu*({*Nalidb*, *Sklidb*, *Zonidb*, *Ndatiz*, *PPidb*}, {*Nalidb*}),
- *Stavka_naloga*({*Nalidb*, *Stnidb*, *Matidb*, *Stnkol*}, {*Nalidb* + *Stnidb*, *Nalidb* + *Matidb*}),
- *Skl_zona*({*Sklidb*, *Zonidb*, *Zdatpo*}, {*Sklidb* + *Zonidb*}),
- *Zalihe*({*Sklidb*, *Zonidb*, *Matidb*, *Zalpoc*, *Zalsta*, *Zalulz*, *Zalizl*}, {*Sklidb* + *Zonidb* + *Matidb*})

i skupom međurelacionih ograničenja I_{p_2}:

- *Zalihe*[*Matidb*] $\subseteq$ *Materijal*[*Matidb*],
- *Zalihe*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Stavka_naloga*[*Nalidb*] $\subseteq$ *Nalog_za_otpremu*[*Nalidb*],
- *Nalog_za_otpremu*[*Nalidb*] $\subseteq$ *Stavka_naloga*[*Nalidb*],
- *Nalog_za_otpremu*[*Sklidb* + *Zonidb*] $\subseteq$ *Skl_zona*[*Sklidb* + *Zonidb*],
- *Nalog_za_otpremu*[*PPidb*] $\subseteq$ *Posl_part*[*PPidb*] i
- (*Stavka_naloga* $\rhd\lhd$ *Nalog_za_otpremu*)[*Sklidb* + *Zonidb* + *Matidb*] $\subseteq$ *Zalihe*[*Sklidb* + *Zonidb* + *Matidb*].

ISBP / MPRO - Ispitivanje stroge kompatibilnosti II

Proces *Stroga_kompatibilnost* za ulazne šeme BP (S_3, I_3) i (S_{p_2}, I_{p_2}), daje izlazni rezultat *Ind* = *Strue*, što znači da je $(S_3, I_3) \cong (S_{p_2}, I_{p_2})$, pri čemu je *TranC_Rep* = *FnfC_Rep* = *IntcDk_Rep* = *IntCLi_Rep* = *EqKeyC_Rep* = ∅, a izveštaj *IntCGr_Rep* sadrži poruke (slika 5.41).

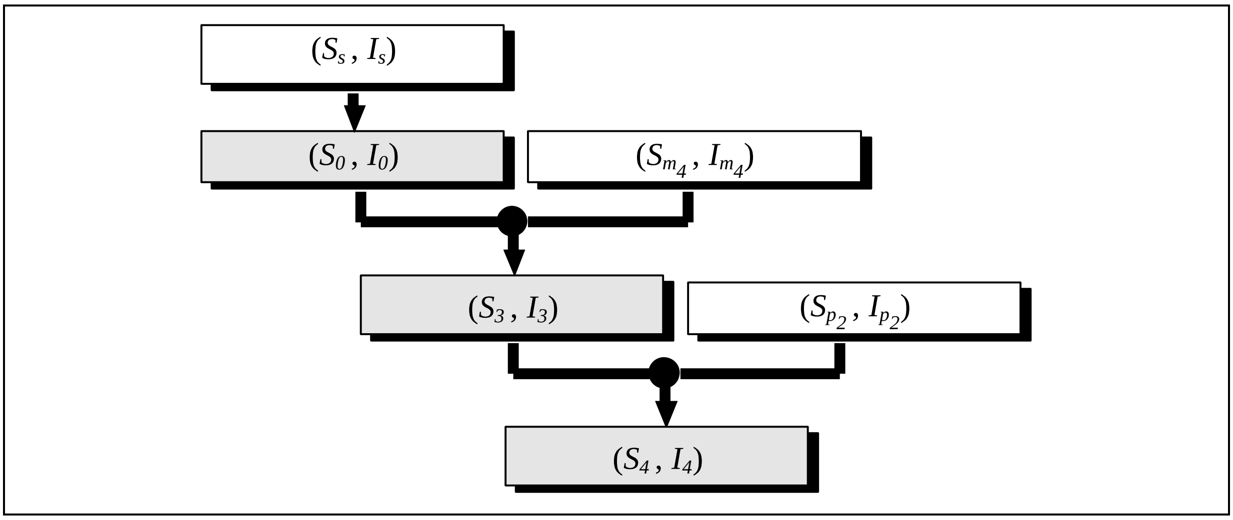
IZVEŠTAJ I-10: *IntCGr_Rep* - *Globalna kolizija međurelacionih ograničenja*	
Integrisana šema BP (**ISBP**)	Modul: **Prodaja** (**MPRO**)
Adresar: *R* = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Posl_part*: *R* = {*PPidb*, *PPnaz*, *PPadr*, *PPtel*, *Adrptt*} $\mathcal{K}$ = {*PPidb*}
1. Poruka: *ic*: MPRO.*Posl_part*[*Adrptt*] ⊆ ISBP.*Adresar*[*Adrptt*] *Moguće nepokriveno međurelaciono ograničenje.*	
Skladište: *R* = {*Sklidb*, *Sklnaz*, *Lokidb*} $\mathcal{K}$ = {*Sklidb*}	*Skl_zona*: *R* = {*Sklidb*, *Zonidb*, *Zdatpo*} $\mathcal{K}$ = {*Sklidb* + *Zonidb*}
2. Poruka: *ic*: MPRO.*Skl_zona*[*Sklidb*] ⊆ ISBP.*Skladište*[*Sklidb*] *Moguće nepokriveno međurelaciono ograničenje.*	

Slika 5.41.

Upozorenja *IntCGr_Rep.I-10.1* i *IntCGr_Rep.I-10.2.* ne dovode do kolizije integrativnog svojstva, što znači da se može pristupiti integraciji ISBP i MPRO.

ISBP / MPRO - Integracija šeme BP IV

Saglasno činjenici da važi $(S_3, I_3) \cong (S_{p_2}, I_{p_2})$, vrši se integracija tekuće verzije ISBP sa trećom verzijom šeme MPRO (S_{p_2}, I_{p_2}). Kao rezultat (slika 5.42), dobija se novo tekuće stanje ISBP (S_4, I_4), sa skupom šema relacija S_4:



Slika 5.42.

- *Org_jedinica*({*OJidb*, *OJnaz*, *OJnadr*, *Lokidb*, *LDeidb*}, {*OJidb*}),

- Adresar({Adrptt, Adrmes}, {Adrptt}),
- Materijal({Matidb, Matozn, Matnaz, Matjus, Matzal}, {Matidb}),
- Posl_part({PPidb, PPnaz, PPadr, PPtel, Adrptt}, {PPidb}),
- Lokacija({Lokidb, Loknaz, Lokadr, Adrptt}, {Lokidb}),
- Detalj_lokacije({Lokidb, LDeidb, LDenaz}, {Lokidb + LDeidb}),
- Skladište({Sklidb, Sklnaz, Lokidb}, {Sklidb}),
- Skl_zona({Sklidb, Zonidb, Zonnaz, Zdatpo, ZLoidb, LDeidb}, {Sklidb + Zonidb}),
- Zalihe({Sklidb, Zonidb, Matidb, Zalpoc, Zalsta, Zalulz, Zalizl}, {Sklidb + Zonidb + Matidb}),
- Tip_dokumenta({Sdvrst, Sdvrpr, Sdnaz}, {Sdvrst}),
- Skl_dokument({Sklidb, Zonidb, Posgod, Sdvrst, Sdtip, Sdidbr, Sdatiz, Nalidb}, {Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr}),
- Stavka_dokum({Sklidb, Zonidb, Posgod, Sdvrst, Sdtip, Sdidbr, Stvidb, Matidb, Stvkol}, {Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr + Stvidb, Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr + Matidb}),
- Nalog_za_otpremu({Nalidb, Sklidb, Zonidb, Ndatiz, PPidb}, {Nalidb}),
- Stavka_naloga({Nalidb, Stnidb, Matidb, Stnkol}, {Nalidb + Stnidb, Nalidb + Matidb})

i skupom međurelacionih ograničenja I_4:

- Posl_part[Adrptt] $\subseteq$ Adresar[Adrptt],
- Org_jedinica[Lokidb + LDeidb] $\subseteq$ Detalj_lokacije[Lokidb + LDeidb],
- Org_jedinica[OJnadr] \ {w} $\subseteq$ Org_jedinica[OJidb],
- Lokacija[Adrptt] $\subseteq$ Adresar[Adrptt],
- Detalj_lokacije[Lokidb] $\subseteq$ Lokacija[Lokidb],
- Skl_zona[Sklidb] $\subseteq$ Skladište[Sklidb],
- Skladište[Lokidb] $\subseteq$ Lokacija[Lokidb],
- Skl_zona[ZLoidb + LDeidb] $\subseteq$ Detalj_lokacije[Lokidb + LDeidb],
- Zalihe[Sklidb + Zonidb] $\subseteq$ Skl_zona[Sklidb + Zonidb],
- Zalihe[Matidb] $\subseteq$ Materijal[Matidb],
- Skl_dokument[Sdvrst] $\subseteq$ Tip_dokumenta[Sdvrst],
- Skl_dokument[Sklidb + Zonidb] $\subseteq$ Skl_zona[Sklidb + Zonidb],
- Stavka_dokum[Sklidb+Zonidb+Posgod+Sdvrst +Sdtip + Sdidbr] $\subseteq$ Skl_dokument[Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr],
- Skl_dokument[Sklidb+Zonidb+Posgod +Sdvrst +Sdtip +Sdidbr] $\subseteq$ Stavka_dokum[Sklidb + Zonidb + Posgod + Sdvrst + Sdtip + Sdidbr],
- Stavka_dokum[Sklidb + Zonidb + Matidb] $\subseteq$ Zalihe[Sklidb + Zonidb + Matidb],
- BR_Nenegativna_zaliha(Tip_dokumenta, Skl_dokument, Stavka_dokum, Zalihe),
- Skl_dokument[Nalidb + Sklidb + Zonidb] $\subseteq$ Nalog_za_otpremu[Nalidb + Sklidb + Zonidb],
- (Stavka_naloga $\bowtie$ Nalog_za_otpremu)[Sklidb + Zonidb + Matidb] $\subseteq$ Zalihe[Sklidb + Zonidb + Matidb],
- Nalog_za_otpremu[PPidb] $\subseteq$ Posl_part[PPidb],
- Nalog_za_otpremu[Sklidb + Zonidb] $\subseteq$ Skl_zona[Sklidb + Zonidb],
- Stavka_naloga[Nalidb] $\subseteq$ Nalog_za_otpremu[Nalidb],
- Nalog_za_otpremu[Nalidb] $\subseteq$ Stavka_naloga[Nalidb] i
- BR_Nenegativna_stavka_nal(Nalog_za_otpremu, Stavka_dokum, Stavka_naloga).

5.3.3. Dodatni primeri kolizionih šema (MXXX)

U ovoj tački će biti demonstrirani primeri kolizija *FnfC*, *IntCDk*, *IntCLi* i *EqKeyC*. U tom cilju, uvodi se hipotetički modul MXXX, koji treba integrisati sa tekućim stanjem ISBP (S_4 , I_4).

MXXX - Skup tipova formi

Neka je projektant modula MXXX napravio svoje tipove formi:

- *ORG_JEDINICAX* (***OX***) - korišćenje evidencije o organizacionoj strukturi realnog sistema, slika 5.43,
- *MATERIJALX* (***MX***) - korišćenje kataloga materijala, slika 5.44,
- *ADRESARX* (***AX***) - korišćenje kataloga poštanskih brojeva i mesta, slika 5.45,

i preuzeo tip forme *LOKACIJA* (slika 5.11).

ORG_JEDINICAX

OJidb, OJnadr, *OJnaz, Lokidb*

Slika 5.43.

MATERIJALX

Matidb, *Matnaz*, ***Matozn, Matsta***

Slika 5.44.

ADRESARX

Adrmes, *Adrptt*

Slika 5.45.

Definicija obeležja *Matsta*: status materijala, *dbs*(*Matsta*) = *Y*, *el*(*Matsta*) = *E*. Značenja svih ostalih obeležja tipova formi *ORG_JEDINICAX* i *MATERIJALX* su data u okviru tačke MSIS - skup tipova formi.

MXXX - Šema modula BP - V.1.

Na osnovu tipova formi *ORG_JEDINICAX* (SLIKA 5.43), *MATERIJALX* (slika 5.44), *ADRESARX* (slika 5.45) i *LOKACIJA* (slika 5.11), generiše se prva verzija šeme MXXX (S_{x_1} , I_{x_1}), sa skupom šema relacija S_{x_1} :

- *Materijal*({*Matidb, Matozn, Matsta, Matnaz*}, {*Matidb, Matozn + Matsta*}),
- *Org_jedinica*({*OJidb, OJnadr, OJnaz, Lokidb*}, {*OJidb + OJnadr*}),
- *Adresar*({*Adrmes, Adrptt*}, {*Adrmes*}),
- *Lokacija*({*Lokidb, Loknaz, Lokadr, Adrptt*}, {*Lokidb*})

i skupom međurelacionih ograničenja I_{x_1} :

- *Org_jedinica*[*OJnadr*] $\subseteq$ *Org_jedinica*[*OJidb*] i

- *Lokacija*[*Lokidb*] $\subseteq$ *Org_jedinica*[*Lokidb*] [*)].

ISBP / MXXX - Ispitivanje stroge kompatibilnosti I

Proces *Stroga_kompatibilnost* za ulazne šeme BP (S_4, I_4) i (S_{x_1}, I_{x_1}), daje izlazni rezultat *Ind* = *false*, što znači da je (S_4, I_4) $\not\cong_I$ (S_{x_1}, I_{x_1}) i formira sledeće izveštaje sa detektovanim kolizijama i upozorenjima:

- za *fnf*-koliziju *FnfC*(S_4, S_{x_1}), formira se izveštaj *FnfC_Rep* (slika 5.45),
- za koliziju dela ključa *IntcDk*(S_4, S_{x_1}), formira se izveštaj *IntcDk_Rep* (slika 5.46),
- za lokalnu koliziju međurelacionih ograničenja *IntcLi*(S_4, S_{x_1}), formira se izveštaj *IntcLi_Rep* (slika 5.47),
- za koliziju ekvivalentnin ključeva *EqKeyC*(S_4, S_{x_1}), formira se izveštaj *EqKeyC_Rep* (slika 5.48),

dok je *TranC_Rep* = *IntCGr_Rep* = $\emptyset$.

IZVEŠTAJ I-11: *FnfC_Rep* - *Kolizija funkcionalni-nefunkcionalni odnos*
MXXX.*Org_jedinica*: *R* = {*OJidb*, *OJnadr*, *OJnaz*, *Lokidb*} $\mathcal{K}$ = {*OJidb* + *OJnadr*}
1. Greška: *OJidb* $\rightarrow$ *OJnadr* - *Funkcionalni odnos u okviru ključa*

Slika 5.46.

IZVEŠTAJ I-12: *IntCDk_Rep* - *Kolizija dela ključa*	
Integrisana šema BP (**ISBP**)	Modul: **Fiktivni** (**MXXX**)
Materijal: *R* = {*Matidb*, *Matnaz*, *Matjus*, *Matozn*} $\mathcal{K}$ = {*Matidb*}	*Materijal*: *R* = {*Matidb*, *Matozn*, *Matsta*, *Matnaz*} $\mathcal{K}$ = {*Matidb*, *Matozn* + *Matsta*}
1. Upozorenje: ISBP.*Materijal* $\leftrightarrow$ MXXX.*Materijal*, $\{Matozn + Matsta\} \cap (\text{ISBP.}\,Materijal)^{+}_{\Gamma_{mn} \setminus \Gamma(R_i)} = \{Matozn\} \neq \emptyset$ *Deo ključa u šemi relacije:* ISBP.*Materijal*	

Slika 5.47.

IZVEŠTAJ I-13: *IntCLi_Rep* - *Lokalna kolizija međurelacionih ograničenja*
MXXX.*Lokacija*: *R* = {*Lokidb*, *Loknaz*, *Lokadr*, *Adrptt*} $\mathcal{K}$ = {*Lokidb*}
1. Upozorenje: *Lokacija*[*Lokidb*] $\subseteq$ *Org_jedinica*[*Lokidb*] *Ne postoji ekvivalentno ograničenje u modulu:* ISBP.

Slika 5.48.

[*)] Ovu zavisnost sadržavanja interaktivno zadaje projektant modula.

IZVEŠTAJ I-14: *EqKeyC_Rep* - *Kolizija ekvivalentnih ključeva*	
Integrisana šema BP (**ISBP**)	Modul: **Fiktivni** (**MXXX**)
Adresar: R = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrptt*}	*Adresar*: R = {*Adrptt*, *Adrmes*} $\mathcal{K}$ = {*Adrmes*}
1. Greška: ISBP.*Adresar* ↔ MXXX.*Adresar* *Ekvivalentne šeme relacija.*	

Slika 5.49.

Greška *FnfC_Rep.I-11.1.* je posledica loše definisanog ključa tipa forme *ORG_JEDINICAX* i razrešava se preuzimanjem tipa forme *ORG_JEDINCA* iz sistemskog modula, umesto tipa forme *ORG_JEDINCAX*.

Ekstenzionalna kolizija (upozorenje) *IntcDk_Rep.I-12.1.* se:

- može ignorisati, ili
- razrešiti na intenzionalnom nivou.

Ignorisanjem kolizije *IntcDk_Rep.I-12.1*, u integrisanoj šemi BP će se (nakon primene algoritma 4.11) naći šema relacije:

$$Materijal(\{Matidb, Matozn, Matsta, Matnaz\}, \{Matidb, Matozn + Matsta\}).$$

Nakon integracije je potrebno izvršiti eventualna prestrukturiranja podataka, sadržanih u relacijama nad šemama MSIS.*Materijal* i/ili MXXX.*Materijal*. Neka je mat_s pojava nad šemom MSIS.*Materijal*({*Matidb*, *Matozn*, *Matnaz*}, {*Matidb*}), a mat_x pojava nad šemom MXXX.*Materijal*({*Matidb*, *Matozn*, *Matsta*, *Matnaz*}, {*Matidb*, *Matozn* + *Matsta*}). Restrukturiranje podataka se, u ovom slučaju, može obaviti tako što se formira relacija *mat*, nad šemom *Materijal*({*Matidb*, *Matozn*, *Matsta*, *Matnaz*}, {*Matidb*, *Matozn* + *Matsta*}), na sledeći način:

$$mat = mat_x \cup \{t[Matidb+Matozn+Matnaz+Matsta] \mid t[Matidb] \in \Pi_{Matidb}(mat_s) \setminus \Pi_{Matidb}(mat_x) \wedge$$
$$t[Matidb + Matozn + Matnaz] \in mat_s \wedge t[Matsta] \in dom(Matsta) \setminus \{w\}\},$$

što znači da se, inicijalno, preuzimaju torke iz relacije mat_x, a preostale torke (iz relacije mat_s) se dopunjuju vrednostima za obeležje *Matsta*, tako da zadovolje ograničenje ključa *Matozn* + *Matsta*.

Razrešenje kolizije *IntcDk_Rep.I-12.1.* na intenzionalnom nivou može da se obavi na dva načina: preuzimanjem tipa forme *MATERIJAL* iz sistemskog modula, ili preprojektovanjem tipa forme *MATERIJAL* u sistemskom modulu, tako da odgovara tipu forme *MATERIJALX*. U slučaju da se projektanti, saglasno uočenim ograničenjima koja važe u realnom sistemu, opredele za drugo rešenje, potrebno je izgenerisati novu verziju šeme modula za MSIS i ponovo izvršiti njenu integraciju sa tekućim stanjem ISBP, što može predstavljati dosta obimnu aktivnost. U tom slučaju, ignorisanje navedene kolizije, praktično, ostaje zadovoljavajuće rešenje.

Upozorenje *IntcLi_Rep.I-13.1.* se, takođe, može otkloniti na dva načina: brisanjem zavisnosti sadržavanja *Lokacija*[*Lokidb*] $\subseteq$ *Org_jedinica*[*Lokidb*] iz MXXX, ili njenim uvođenjem u MSIS, ukoliko je to neophodno.

Greška *EqKeyC_Rep.I-14.1.* znači da su šeme relacija ISBP.*Adresar* i MXXX.*Adresar* ekvivalentne s obzirom na skup *fd* ISBP i MXXX, ali nemaju zajedničkih ključeva, odnosno da se ključevi tipova objekata formi *ADRESAR* i *ADRESARX*, redom, *Adrptt* i *Adrmes* međusobno funkcionalno određuju: $Adrptt \rightarrow Adrmes$ i $Adrmes \rightarrow Adrptt$. To svojstvo, međutim, nije lokalno zadovoljeno - na nivou šeme ISBP, ili MXXX. Kolizija *EqKeyC_Rep.I-14.1.* se može razrešiti preprojektovanjem tipa forme *ADRESAR* (proglašavanjem obeležja *Adrmes* za drugi ključ tipa objekta), ili tipa forme *ADRESARX* (proglašavanjem obeležja *Adrptt* za ključ tipa objekta ili/i ukidanjem ključa *Adrmes*), saglasno važećim ograničenjima u realnom sistemu.

5.4. Studija slučaja - završna analiza

U okviru završne analize prikazane studije slučaja - demo primera integracije šeme MSKL u jedinstvenu šemu BP, još jednom se apostrofiraju najznačajnije činjenice i zaključci, vezani za analizu izveštaja o kolizijama, produkovanih putem procesa *Stroga_kompatibilnost* (algoritam 4.10) i integraciju šema modula u jedinstvenu šemu BP. Pri tome će biti komentarisano sledeće:

- inicijalno uključivanje svih tipova formi sistemskog modula u skup tipova formi posmatranog modula,
- prioritet projektovanja i preuzimanja zajedničkih tipova formi,
- retrogradno kretanje postupka integracije i
- značaj upozorenja i poruka u postupku integracije.

Inicijalno uključivanje svih tipova formi MSIS u skup tipova formi posmatranog modula se, u praksi, pokazuje korisnim, jer se time izbegava pojava velikog broja kolizija, do kojih može doći usled eventualnih neusaglašenosti tipova formi posmatranog modula i MSIS. Razlog za inicijalno uključivanje tipova formi MSIS u posmatrani modul predstavlja činjenica da se u gotovo svakom modulu intenzivno javlja potreba za podacima o zajedničkim činiocima poslovanja, koji se evidentiraju u okviru MSIS. S druge strane, ukupan broj tipova formi u MSIS, uobičajeno, nije veliki, tako da projektanti drugih modula nisu "opterećeni" problemom broja preuzetih tipova formi iz MSIS. Sa stanovišta "dobrog" i "efikasnog" projektovanja može biti, takođe, zadovoljavajuća i varijanta delimičnog preuzimanja (samo potrebnih) tipova formi iz MSIS.

Da su u skup tipova formi MSKL (studija slučaja - potpoglavlje 5.3) inicijalno bili uključeni tipovi formi iz MSIS (suprotno pretpostavci P1), tada bi ispitivanje stroge kompatibilnosti I za MSIS i MSKL dalo pozitivan rezultat, čime bi se izbegla jedna iteracija u postupku integracije jedinstvene šeme BP.

Prioritet projektovanja i preuzimanja zajedničkih tipova formi može imati ne mali uticaj na obezbeđenje preduslova za paralelan rad više projektanata na različitim modulima, odnosno na povećanje produktivnosti projektovanja. Sa stanovišta postupka progresivne integracije jedinstvene šeme BP, visoki prioritet ovih aktivnosti znači:

- inicijalno preuzimanje zajedničkih tipova formi iz drugih modula koji su potrebni u posmatranom modulu i
- projektovanje zajedničkih tipova formi, koje preuzimaju projektanti drugih modula,

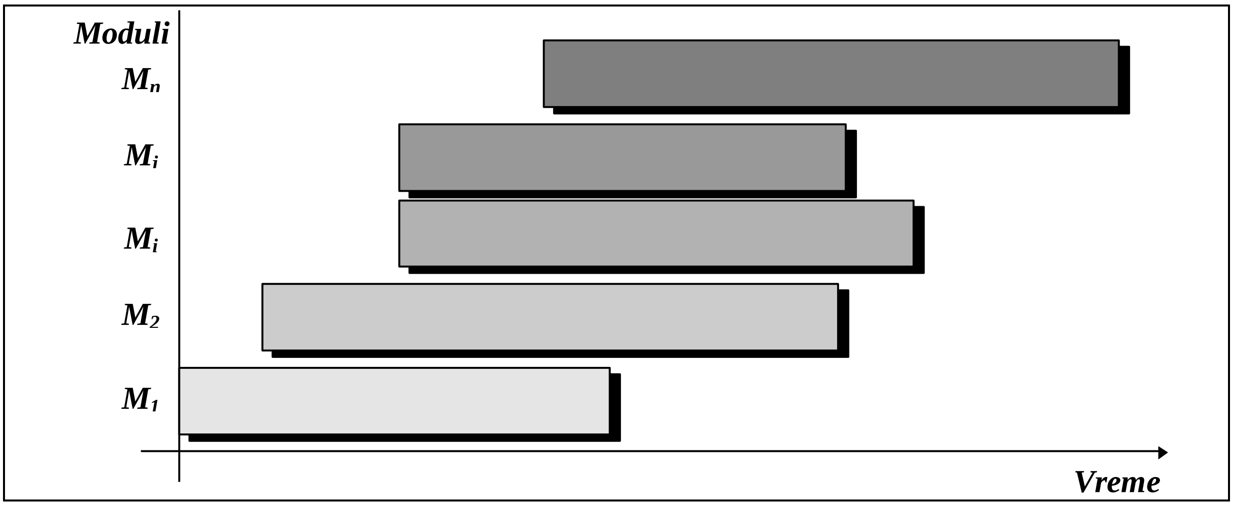
čime se izbegava pojava kolizija, do kojih može doći usled nekih neusaglašenosti tipova formi kreiranih u okviru posmatranog modula i preuzetih iz drugih modula.

Da je u skup tipova formi MSKL inicijalno bio uključen tip forme *NALOG_ZA_OTPREMU* iz MPRO, tada bi ispitivanje stroge kompatibilnosti IV za ISBP i MSKL dalo pozitivan rezultat, čime bi se izbegla još jedna iteracija u postupku integracije jedinstvene šeme BP.

Retrogradno kretanje postupka integracije znači da se u postupku integracije mogu pojaviti slučajevi kolizija koji zahtevaju poništavanje određenog broja koraka integracije i povratak na neko prethodno tekuće stanje integrisane šeme BP. Pri tome se sve "poništene" aktivnosti integracije moraju ponoviti, u cilju generisanja novog tekućeg stanja integrisane šeme BP.

Primer za retrogradno kretanje postupka integracije u studiji slučaja predstavlja povratak na tekuće stanje ISBP (S_0, I_0) (slika 5.38), koje je posledica ispitivanja stroge kompatibilnosti IV, tj. kolizija, prikazanih u izveštajima *TranC_Rep.I-7.* (slika 5.36) i *IntCGr_Rep.I-8.* (slika 5.37). Da je ovim retrogradnim kretanjem bila poništena bilo koja integracija neke druge šeme modula sa ISBP (osim šeme MSKL), ta integracija bi se (uz testiranje stroge kompatibilnosti) morala ponovo sprovesti.

Potreba inicijalnog preuzimanja zajedničkih tipova formi iz drugih modula i pojava retrogradnog kretanja postupka integracije iniciraju problem utvrđivanja "*prirodnog redosleda*" projektovanja i integracije šema modula. Postojanje prirodnog redosleda integracije je, u velikoj meri, determinisano prirodom i načinom obavljanja funkcija realnog sistema. Ukoliko se prirodni redosled projektovanja i integracije šema modula (ekspertskom analizom) "dobro definiše", tada bi se proces projektovanja modula, u vremenskom smislu, mogao odvijati, okvirno, prema šemi, prikazanoj na slici 5.50. Problem utvrđivanja egzistencije i "dobre definicije" prirodnog redosleda projektovanja i integracije izlazi izvan okvira ovog rada.



Slika 5.50.

Upozorenja i poruke, kao što se vidi iz algoritama 4.1-4.11, ne dovode do narušavanja intenzionalne kompatibilnosti. Postoje, međutim, situacije u kojima upozorenja i poruke mogu *ukazivati* kako na kolizije koje se moraju otkloniti na nivou tipova formi pre započinjanja samog procesa integracije, tako i na potrebu restrukturiranja podataka, u cilju njihovog dovođenja u saglasnost sa najnovijim stanjem integrisane šeme BP, koje će biti izgenerisano putem algoritma 4.11. Zbog toga se preko upozorenja i poruka *ne sme* olako prelaziti pri ekspertskoj analizi izgenerisanih izveštaja - izlaza algoritma 4.10. (procesa *Stroga_kompatibilnost*).

U ovoj studiji slučaja, izveštaji *TranC_Rep.I-5.* (slika 5.32) i *IntCGr_Rep.I-6.* (slika 5.33), koji su rezultat ispitivanja stroge kompatibilnosti I za ISBP i MPRO, sadrže isključivo upozorenja i poruke. Značaj tih upozorenja i poruka je takav, da se ne može pristupiti integraciji šema BP za ISBP i MPRO, pre nego se otklone kolizije, indikovane izveštajima *TranC_Rep.I-5.* i *IntCGr_Rep.I-6.*

Činjenice, na osnovu kojih se može izvesti zaključak o potrebi restrukturiranja podataka u cilju njihovog dovođenja u saglasnost sa najnovijim stanjem integrisane šeme BP, saglasno upozorenjima i porukama raspoloživih izveštaja, su:

- da li postoje i pojava nad tekućim stanjem integrisane šeme BP i pojava nad šemom modula koji se integriše, ili postoji samo jedna od ove dve pojave i koja je pojava u pitanju,
- da li uneti podaci narušavaju ograničenja najnovijeg stanja integrisane šeme BP i
- da li postoji potreba za uvođenjem i da li su primenjene potrebne organizacione mere u realnom sistemu (koje se ogledaju kroz protokole o pravima ažuriranja pojedinih relacija).

U studiji slučaja je dat primer moguće potrebe za restrukturiranjem podataka, koja je posledica kolizije dela ključa - upozorenja *IntcDk_Rep.I-12.1* (slika 5.47).

Kao primer organizacionih mera (protokola o pravima ažuriranja pojedinih relacija), u studiji slučaja se pojavljuje pretpostavka P4, po kojoj se relacije nad šemama *Posl_part* i *Lokacija* ne smeju ažurirati putem programskog podsistema MSKL, već samo putem programskog podsistema MSIS, čime se izbegavaju moguće kolizije, indikovane porukama *IntCGr_Rep.I-3.1.* i *IntCGr_Rep.I-3.2.* (slika 5.25).

Pored toga, potrebu za restrukturiranjem podataka u konkretnom slučaju, mogu izazvati i sve druge greške, upozorenja i poruke, ukoliko su u relacije nad šemama, obuhvaćenim tim greškama, upozorenjima i porukama, upisani podaci koji narušavaju određena, trenutno “nepokrivena” ograničenja.

6. Poglavlje

Zaključak

Rad *Problem integracije šema modula baze podataka informacionog sistema* se bavi pitanjima automatskog otkrivanja kolizija, koje nastaju pri paralelnom projektovanju različitih šema modula baze podataka i problemom izgradnje jedinstvene šeme baze podataka informacionog sistema.

Poslednjih sedam godina na Institutu za industrijske sisteme Fakulteta tehničkih nauka u Novom Sadu, razvija se CASE alat za automatizovano projektovanje šeme baze podataka i generisanje prototipa aplikacija, pod nazivom IIS∗CASE. U radu su prikazani osnovni principi primene i načina funkcionisanja alata IIS∗CASE, pri čemu je posebno naglašena njegova primena u domenu projektovanja integrisane šeme baze podataka. Istraživanje, čiji su rezultati prikazani u ovom radu, sprovedeno je u funkciji unapređenja postojeće i realizacije nove verzije alata IIS∗CASE, u cilju obezbeđenja automatizacije postupka projektovanja integrisane šeme baze podataka i stvaranja preduslova za prototipski pristup razvoja informacionog sistema. Pri tome, jedan od važnih zahteva je bio da se obezbede uslovi za operativno korišćenje prototipa aplikacija, što znači da se podaci, koje bi korisnik uneo putem nekog prototipa aplikacije, ne smeju odbaciti pri daljem unapređivanju prototipa ka konačnom softverskom proizvodu. Umesto toga, podatke po potrebi treba restrukturirati u cilju usaglašavanja s tekućom verzijom šeme baze podataka. Ovaj zahtev je, u ne maloj meri, uticao na povećanje složenosti razmatranih problema, a samim tim i rada u celini.

Najvažniji rezultati rada, u odnosu na koncepte primenjene u dostupnoj literaturi, sastoje se u sledećem:

- Uvedeni su pojmovi proširenja šeme baze podataka, integrativnog svojstva, strogo kompatibilnih šema, strogo pokrivajuće šeme relacione baze podataka, intenzionalno kompatibilnih šema modula i intenzionalno pokrivajuće šeme relacione baze podataka. Na taj način, formirani su kriterijumi (na intenzionalnom i ekstenzionalnom nivou) koje šeme modula treba da zadovolje, da bi mogle biti integrisane u jedinstvenu šemu baze podataka.
- Identifikovani su mogući tipovi kolizija šema modula, formulisan je i dokazan potreban i dovoljan uslov stroge i intenzionalne kompatibilnosti šema modula, što je omogućilo da se, u formi algoritama, prikažu postupci za ispitivanje stroge i intenzionalne kompatibilnosti šema modula. Formalizovan je i postupak integracije strogo kompatibilnih i intenzionalno kompatibilnih šema u jedinstvenu (strogo pokrivajuću) šemu baze podataka.

- Dat je prikaz metodologije primene algoritama za testiranje stroge i intenzionalne kompatibilnosti i integraciju šema modula u jedinstvenu šemu baze podataka informacionog sistema.
- Sistematizovano i strogo formalno je prezentiran pristup problemu zatvorenosti familije pojava s obzirom na operatore projekcije i prirodnog spoja. U okviru prezentacije ovog pristupa, formalizovani su načini prikaza svih ostalih tipova zavisnosti podataka putem generalizovanih tipova zavisnosti.
- Formalizovan je skup pravila izvođenja za izvedene F-zavisnosti, uveden pojam zamene obeležja skupom lanaca i pomoću leme 3.5. formulisano pravilo izvođenja za F-zavisnosti, zasnovano na zameni obeležja leve strane F-zavisnosti skupom lanaca. Ovo je, sve zajedno, obezbedilo pojednostavljenje dokaza teoreme 3.10. o potrebnom i dovoljnom uslovu da *fd*-familija bude zatvorena s obzirom na operator projekcije.
- Sistematizovani su mogući tipovi međurelacionih ograničenja šeme modula, odnosno integrisane šeme baze podataka i prikazani su načini za izvođenje zaključaka o njihovom važenju.

Veći deo prikazanih rezultata (algoritama) je već implementiran u okviru sadašnje verzije programskog proizvoda IIS∗CASE. Reč je o algoritmima za ispitivanje kolizije tranzitivnosti, kolizije funkcionalni-nefunkcionalni odnos i kolizije ekvivalentnih ključeva. Realizovan je, takođe, i algoritam integracije šema modula u jedinstvenu šemu baze podataka. Svi algoritmi su realizovani pomoću programskog jezika PROLOG (Arity-Prolog, verzija 5.1), koji je, kako se pokazuje, podesan za ove vrste problema, zbog visokog nivoa deklarativnosti. Algoritmi su praktično korišćeni pri projektovanju integrisane šeme baze podataka informacionog sistema jednog većeg jugoslovenskog preduzeća.

Svi rezultati ovog rada će biti implementirani u okviru nove verzije IIS∗CASE, čiji je razvoj u toku. Pri tome će se insistirati na objektno-orijentisanom pristupu razvoja algoritama, budući da se celokupan razvoj nove verzije IIS∗CASE bazira na objektnoj orijentaciji i softverskim alatima koji je podržavaju.

Kompletan rad je realizovan u roku od dve i po godine, dok je sama implementacija algoritama trajala oko tri meseca.

Nastavak istraživanja, vezanih za ovu problematiku, se može odvijati u tri pravca:

- Realizacija ekspertnog sistema za projektovanje integrisane šeme baze podataka informacionog sistema, baziranog na alatu IIS∗CASE. Ovakav ekspertni sistem bi sadržao analitičarsko i projektantsko znanje, vezano za određene oblasti interesovanja, koje bi bilo od koristi pri projektovanju tipova formi i analizi rezultata (i izveštaja), koje produkuju algoritmi IIS∗CASE-a.
- Automatsko generisanje (SQL) koda za realizaciju implementacione šeme baze podataka na konkretnim relacionim sistemima za upravljanje bazama podataka (RSUBP). Pri tome se misli kako na generisanje opisa relacija (tabela), tako i na generisanje kôda pravila i procedura za kontrolu međurelacionih ograničenja integrisane šeme baze podataka na nivou samog RSUBP-a, a ne pojedinačnih aplikacija.
- Automatsko generisanje programskih specifikacija i prototipa aplikacija, za konkretno okruženje RSUBP-a.

7. Prilog A

Literatura

[BB] Beeri C, Bernstein P.A, “Computational Problems Related the Design of Normal Form Relational Schemas”, *ACM Transactions on Database Systems*, Vol.4, No.1, March, 1979, pp. 30-59.

[C] Codd E. F, *The Relational Model for Database Management Version 2*, Addison-Wesley-Publishing-Company, USA, 1990.

[CMN] Choobineh J, Mannino V.M, Nunamaker F.J, Konsynski R.B, “An Expert Database Design System Based on Analysis of Forms”, *IEEE Transactions on Software Engineering*, Vol. 14, No. 2, February 1988, pp. 242-253.

[CS] Cvetković D, Simić S, *Diskretna matematika - matematika za kompjuterske nauke*, Naučna knjiga, Beograd, 1990.

[Da] Davis M. A, “Fifteen Principles of Software Engineering”, *Computer*, Vol. 27, No. 11, November 1994, pp. 94-101.

[DL] Diet J, Lochovsky F, “Interactive Specification and Integration of User Views Using Forms”, *Proceedings of the Eighth International Conference on Entity-Relationship Approach*, Toronto, Canada, 18-20. October, 1989, pp. 171-185.

[DM] Diedrich I, Milton J, “New Methods and Fast Algorithms for Database Normalization”, *ACM Transactions on Database Systems*, Vol 13, No. 3, Sept. 1988, pp. 339-365.

[Fa] Fagin R, “Horn Clauses and Database Dependencies”, *Journal of the ACM*, Vol. 29, No. 4, October 1982, pp. 952-985.

[FMU] Fagin R, Mendelzon O. A, Ullman D. J, “A Simplified Universal Relation Assumption and Its Properities”, *ACM Transactions on Database Systems*, Vol. 7, No. 3, September 1982, pp. 343-360.

[GLM] Govedarica M, Luković I, Mogin P, “Generisanje skupa međurelacionih ograničenja implementacione šeme baze podataka”, *XXXIX Konferencija ETRAN*, Zlatibor, 06-09. 06. 1995. (Prihvaćen i prezentovan rad)

[GML] Govedarica M, Mogin P, Luković I, “O kontroli zavisnosti sadržavanja u bazi podataka”, *Zbornik radova XXXVIII jugoslovenske konferencije ETRAN*, Niš, 07-09. 06. 1994, III sveska, pp. 101-102.

[GMV] Graham M. H, Mendelzon A. O, Vardi M. Y, “Notions of Dependency Satisfaction”, *Journal of the ACM*, Vol. 33, No. 1, January 1986, pp. 105-129.

[GZ] Ginsburg S, Zaiddan S. M, “Properties of Functional-Dependency Families”, *Journal of the ACM*, Vol. 29, No. 3, July 1982, pp. 678-698.

[HS] Honeyman P, Scoire E, “New Caracterization of Independence”, *Proceedings of the ACM-SIGMOD Conference*, 1983, pp. 92-96.

[Hu] Hull R, “Finitely Specifiable Implicational Dependency Families”, *Journal of the ACM*, Vol. 31, No. 2, April 1984, pp. 210-226.

[KMo] Kuzmanov S, Mogin P, “The Relation Scheme Extensions”, *Informatica*, 4/90, 1990, pp. 31-38.

[Ku] Kuzmanov S, *Prilozi teoriji nezavisnih i acikličkih proširenja šema relacionih baza podataka*, Doktorska disertacija, Prirodno matematički fakultet, Institut za matematiku, Novi Sad, oktobar 1989.

[LKS] Luković I, Karadžić Ž, Sič P, “Specifikacija obeležja i domena u rečniku podataka CASE alata”, *Zbornik radova XXXVIII jugoslovenske konferencije ETRAN*, Niš, 07-09. 06. 1994, III sveska, pp. 87-88.

[LM1] Luković I, Mogin P, “Oblikovanje skupa ograničenja baze podataka putem tipova formi”, *Zbornik radova XXXVII jugoslovenske konferencije ETAN*, Beograd, 20-23. 09. 1993, VIII sveska, pp. 61-66.

[LM2] Luković I, Mogin P, “Lossless Joins of Relational Database Views”, *Review of Research, Faculty of Science, Mathematics Series*, Novi Sad (prihvaćen za objavljivanje 01. 07. 1995)

[LM3] Luković I, Mogin P, “Metodološki aspekti primene CASE alata u postupku projektovanja informacionog sistema”, *I Simpozijum o računarskim naukama i informatici*, Brezovica, 04-08. 04. 1995. (Prihvaćen i prezentovan rad)

[LM4] Luković I, Mogin P, “Konzistentnost i integracija relacionih šema baza podataka”, *XXXIX Konferencija ETRAN*, Zlatibor, 06-09. 06. 1995. (Prihvaćen i prezentovan rad)

[Lu1] Luković I, *Implikacioni problem za višeznačne i funkcionalne zavisnosti u relacionim bazama podataka*, Diplomski rad, Visoke vojnotehničke škole KOV JNA, Vojno-tehnički fakultet, Zagreb, 11. 06. 1990.

[Lu2] Luković I, *Automatizovano generisanje podšeme relacione baze podataka putem formi*, Magistarski rad, Univerzitet u Beogradu, Elektrotehnički fakultet, Beograd, 18. 06. 1993.

[Lu3] Luković I, “Elementi semantičkog modela podataka u bazi podataka informacionog sistema za upravljanje proizvodnjom”, *Zbornik radova IX naučno-stručne konferencije Industrijski sistemi '93*, Novi Sad, 28-29. 06. 1993, pp. 2.49-2.52.

[M] Maier D, *The Theory of Relational Databases*, Computer Science Press, Inc. rockville, Maryland, 1983.

[Me] Mendelzon O. A, “Database states and Their Tableaux”, *ACM Transactions on Database Systems*, Vol. 9, No. 2, June 1984, pp. 264-282.

[Mi] Mišić V, *Automatsko generisanje aplikacija za baze podataka iz specifikacija*, doktorska disertacija, Elektrotehnički fakultet, Beograd, 1993.

[MK] Mogin P, Karadžić Ž, “Prostiranje primarnog ključa u šemi relacione baze podataka”, *Zbornik radova XV Simpozijuma o informacionim tehnologijama*, Sarajevo - Jahorina, januar 1991, pp. 106.1-8.

[MKu] Mogin P, Kuzmanov S, “IIS*CASE, a Database Schema Design Tool”, *Publications of the School of Engineering Sciences*, University of Novi Sad, 1991, Vol. 21 & 22, pp.126-140.

[ML1] Mogin P, Luković I, “O kontroli integriteta baze podataka”, *Zbornik radova XXXVII jugoslovenske konferencije ETAN*, Beograd, 20-23. 09. 1993, VIII sveska, pp. 55-60.

[ML2] Mogin P, Luković I, “A Prototyping CASE Tool”, *Proceedings for the Dedicated Conference on Rapid Prototyping in the Automotive Industries of the XXVIII International Symphosium on Automotive Technology and Automation*, Stuttgart, Germany, 18-22. 09. 1995, pp. 261-268.

[MLB] Mogin P, Luković I, Brkić M, “Projektantske i programerske podloge za obezbeđenje kvaliteta softvera”, *Monografija SISTEM KVALITETA * Kvalitet softvera * Softver za kvalitet*, Fakultet tehničkih nauka - Institut za industrijske sisteme i IIS - Istraživački i tehnološki centar, 1995, pp. 17-40.

[MLK1] Mogin P, Luković I, Karadžić Ž, “Ka automatskoj identifikaciji ograničenja baze podataka”, *Zbornik radova XXXVI jugoslovenske konferencije ETAN*, Kopaonik, 08-11. 06. 1992, Sveska IX, pp. 19-26.

[MLK2] Mogin P, Luković I, Karadžić Ž, “Relational Database Schema Design and Application Generating using IIS*CASE Tool”, *Proceedings of International Conference on Technical Informatics*, Timisoara, Romania, 16-19. 11. 1994, Vol. 5, pp. 49-58.

[Mo1] Mogin P, *Uvod u baze podataka*, Fakultet tehničkih nauka - Institut za industrijske sisteme, Novi Sad, 1989.

[Mo2] Mogin P, “IIS-CASE, alat za projektovanje šeme baze podataka”, *Zbornik radova naučne konferencije INDUSTRIJSKI SISTEMI - IS’90*, Novi Sad, 21-23. 06. 1990, pp. 381-390.

[Mo3] Mogin P, *Organizacija datoteka i uvod u baze podataka*, Viša škola za organizaciju i informatiku, Novi Sad, 1991.

[MR1] Mogin P, Ristić S, “Kandidati za primarni ključ šeme relacije”, *Zbornik radova XXXVIII jugoslovenske konferencije ETRAN*, Niš, 07-09. 06. 1994, III sveska, pp. 99-100.

[MR2] Mogin P, Ristić S, “O projektovanju A-nezavisne šeme relacione baze podataka”, *I Simpozijum o računarskim naukama i informatici*, Brezovica, 04-08. 04. 1995. (Prihvaćen i prezentovan rad)

[Ni] Nielsen J, “Iterative User-Interface Design”, *Computer*, Vol. 26, No. 11, November 1993, pp. 32-41.

[PBG] Paredaens J, Bra D. P, Gyssens M, Gucht V. D, *The Structure of the Relational Database Model*, Springer-Verlag, Berlin Heidelberg, 1989.

[Ri] Ristić S, *Istraživanje problema propagacije primarnog ključa u šemi relacione baze podataka*, Magistarski rad, Univerzitet u Novom Sadu, Ekonomski fakultet, Subotica, April 1994.

[Sa] Sagiv Y, "Characterization of Globally Consistent Databases and their Correct Access Path", *TR University of Illinois*, July, 1981.

[Sc] Scoire E, "Improving Database Schemes by Adding Attributes", *Proceedings of SIGACT - SIGMOD PODS*, 1983.

[SS] Simić D, Starčević D, "Objektni softver", *Info*, 2/94, pp. 31-35.

[SU] Sadri F, Ullman J. D, "Template Dependencies: A Large Class of Dependencies in Relational Databases and Its Complete Axiomatization", *Journal of the ACM*, Vol. 29, No. 2, April 1982, pp. 363-372.

[U] Ullman D. J, *Principles of Database Systems*, Computer Science Press, Inc. Rockville, Maryland, 1982.

[Y] Yang C. C, *Relational Databases*, Prentice-Hall, A Division of Simon & Schuster, Inc. Englewood Cliffs, New Jersey, 1986.

8. Prilog B

Pregled korišćenih oznaka i skraćenica

–A–

AP	Armstrongova pravila izvođenja
A, A_i	Atribut, Obeležje

–B–

BP	Baza podataka

–C–

CASE	Computer Aided Software Engineering

–D–

$d(U)$	Dekompozicija primitivne šeme relacije
$rhs(f)$	Desna strana funkcionalne zavisnosti
D, D_i, $dom(A)$	Domen
$dcomp(A_i, A_j)$	Domenski kompatibilna obeležja
2NF	Druga normalna forma

–E–

$C \equiv C'$	Ekvivalentni skupovi

–F–

$SAT(R, C)$	Familija pojava
$\mathcal{F}(U)$, $SAT(U)$	Familija relacija
der	Funkcija izvođenja obeležja
ren	Funkcija preimenovanja obeležja
$X \rightarrow Y$, fd, f	Funkcionalna zavisnost

–G–

$<T(U), E>$, $<T(U),\ Eq_{A_j}(\lambda_p^j, \lambda_q^j)>$

egd	Generalizovana E-zavisnost

$<T(U), T'(X)>$, $<T(U), Q(x_1,...,x_m)>$,

tgd	Generalizovana T-zavisnost
gd	Generalizovana zavisnost
$IntCGr(S_m, S_n)$	Globalna kolizija skupova međurelacionih ograničenja

–I–

IS	Informacioni sistem
$Int((S_n, I_n), (S_m, I_m))$	Integrativno svojstvo
ISBP	Integrisana šema baze podataka
$v(T(U))$	Interpretacija tabloa
ifd, $\xi \rightarrow Y$	Izvedena F-zavisnost
IntCGr_Rep	Izveštaj o globalnim kolizijama skupova međurelacionih ograničenja

IntcDk_Rep	Izveštaj o kolizijama dela ključa
EqKeyC_Rep	Izveštaj o kolizijama ekvivalentnih ključeva
FnfC_Rep	Izveštaj o kolizijama funkcionalni-nefunkcionalni odnos
TranC_Rep	Izveštaj o kolizijama tranzitivnosti
IntCLi_Rep	Izveštaj o lokalnim kolizijama skupova međurelacionih ograničenja

–K–

$kp(\Gamma)$	Kanonički pokrivač
$\lvert X \rvert$	Kardinalitet skupa X
$S_{Eq}(R_i)$, S_{Eq}	Klasa ekvivalencije saglasno istim zatvaračima
$IntcDk(S_m, S_n)$	Kolizija dela ključa
$EqKeyC(S_m, S_n)$	Kolizija ekvivalentnih ključeva
$FnfC(S_m, S_n)$	Kolizija funkcionalni-nefunkcionalni odnos
$IntC(S_m, S_n)$	Kolizija integrativnog svojstva
$TranC(S_m, S_n)$	Kolizija tranzitivnosti

–L–

$L_j^i = (\xi_1, .., \xi_n)$	Lanac
$lhs(f)$	Leva strana funkcionalne zavisnosti
$C \models \gamma$, $C \models C'$	Logička posledica
$IntCLi(S_m, S_n)$	Lokalna kolizija skupova međurelacionih ograničenja

–M–

$CH_I(r)$	Maksimalna *chase* sekvenca
$(S_m, I_m) \cong_I (S_n, I_n)$	Međusobna intenzionalna kompatibilnost
$(S_m, I_m) \approx (S_n, I_n)$	Međusobna kompatibilnost
$(S_m, I_m) \cong (S_n, I_n)$	Međusobna stroga kompatibilnost
$r(R) \models \gamma$	Model za ograničenje γ
$r(R) \models C$	Model za skup ograničenja C
MPRO	Modul "Prodaja" demo primera
MSKL	Modul "Skladištenje" demo primera

–N–

nfd, $X \to \theta_i$	Nefunkcionalna zavisnost
θ_i	Neinterpretirano obeležje
$np(\Gamma)$	Neredundantni pokrivač
$\Gamma \vdash_{AP} f$	Niz izvođenja (dokaz) za funkcionalne zavisnosti
w, w_i	Nula vrednost

–O–

$O(x)$	Ocena složenosti algoritma
$\mathcal{G} = (S, \rho)$	Osnovni graf zatvaranja

–P–

$N_p(S_p, I_p)$	Podšema
$(r_1^n, ..., r_p^n)$	Pojava baze podataka
$Ext_R(r_1, ..., r_k)$	Pojava nad šemom $(R, \mathcal{K})$, nastala proširivanjem relacija $r_1, ..., r_k$
$p_{\mathcal{F}}$	Pojava tipa forme
$p_l(N_i)$	Pojava tipa objekta
$Uniran(S_{Eq}(R_k))$	Pokrivajuća šema relacije za klasu ekvivalencije $S_{Eq}(R_k)$
ICS	Polazni skup ograničenja relacione šeme modula
ftgd	Potpuna generalizovana T-zavisnost
fjd	Potpuna zavisnost spoja
IFD3	Pravilo lančanja za izvedene F-zavisnosti
FD1	Pravilo refleksivnosti za funkcionalne zavisnosti
IFD1	Pravilo refleksivnosti za izvedene F-zavisnosti
FD2	Pravilo proširivanja za funkcionalne zavisnosti
FD3	Pravilo tranzitivnosti za funkcionalne zavisnosti
IFD4	Pravilo tranzitivnosti za izvedene F-zavisnosti
IFD2	Pravilo uniranja za izvedene F-zavisnosti
$f_{p_i}(A_j) = \lambda_i^j$	Pridruživanje promenljive tabloa obeležju
$(U, \mathcal{D}, dom)$	Primitivna šema relacije
$\mathcal{F}_1 \bowtie \mathcal{F}_2$	Prirodni spoj familija
$r \bowtie s$	Prirodni spoj relacija
$\Pi_X(\mathcal{F})$	Projekcija familije
$\Pi_X(r)$	Projekcija relacije

$\Pi_X(\Gamma)$, $\Gamma\|_X$	Projekcija skupa funkcionalnih zavisnosti
$\Pi_X(T(U))$	Projekcija tabloa
$W_{ex}(\mathcal{F})$	Prošireni skup obeležja tipa forme
$(S_1, I_1) \supset (S_2, I_2)$	Proširenje šeme baze podataka
$(R, \Sigma) \supset (X, \Delta)$	Proširenje šeme relacije
$T_R(R) \supset T_X(X)$	Proširenje tabloa
1NF	Prva normalna forma

–R–

$R_1[K_i] \subseteq R_2[K_i]$, *rd*	Referencijalni integritet
$r(U)$, r	Relacija
RSUBP	Relacioni sistem za upravljanje bazama podataka
$t[X]$	Restrikcija torke na X

–S–

MSIS	Sistemski modul demo primera
$\mathcal{D}$	Skup domena
$\Gamma(S)$, Γ_S	Skup funkcionalnih zavisnosti, određen ključevima skupa šema relacija S
$\Gamma(R_i)$, Γ_i	Skup funkcionalnih zavisnosti šeme $(R_i, \mathcal{K}_i)$
$F(\mathcal{F})$	Skup funkcionalnih zavisnosti tipa forme
Σ, Δ	Skup generalizovanih zavisnosti
$ID(\mathcal{G}, \mathcal{SF}u)$	Skup generisanih zavisnosti sadržavanja
$ID(\mathcal{G}, desig)$	Skup interaktivno zadatih zavisnosti sadržavanja
$Ekviv(S_m \cup S_n)$	Skup klasa ekvivalencije saglasno istim zatvaračima
$\mathcal{K}_{\mathcal{O}}$	Skup ključeva tipa objekta
$\xi_i = \{L_1^i, ..., L_{n_i}^i\}$	Skup lanaca
$\mathcal{LO}(\mathcal{F})$	Skup listova stabla tipa forme
I, I_m	Skup međurelacionih ograničenja
$DR(\mathcal{G}, dep)$	Skup međurelacionih ograničenja zavisnih šema
$NF(\mathcal{F})$	Skup nefunkcionalnih zavisnosti tipa forme
Θ	Skup neinterpretiranih obeležja
$\Theta(\mathcal{F})$	Skup neinterpretiranih obeležja tipa forme
$LNF(\mathcal{F})$	Skup nepokrivenih nefunkcionalnih zavisnosti
$attr(\xi_i)$	Skup obeležja skupa lanaca
U_{BP}, $U_{BP}(S)$	Skup obeležja šeme baze podataka
$W(\mathcal{F})$	Skup obeležja tipa forme
$C_{\mathcal{F}}$	Skup ograničenja tipa forme
$Ext(S)$	Skup pojava nad proširenim šemama relacija
$BR(der, dbs)$	Skup pravila poslovanja funkcije izvođenja obeležja
$BR(\mathcal{G})$	Skup pravila poslovanja grafa zatvaranja
$K_p(R)$	Skup primarnih obeležja šeme relacije
$RI(\mathcal{G})$	Skup referencijalnih integriteta grafa zatvaranja
$RI_r(ren, dbs)$	Skup referencijalnih integriteta preimenovanja
Sym	Skup simbola (promenljivih)
$FD(U)$	Skup svih funkcionalnih zavisnosti nad U
$EGD(U)$	Skup svih generalizovanih E-zavisnosti nad U
$TGD(U)$	Skup svih generalizovanih T-zavisnosti nad U
$GD(U)$	Skup svih generalizovanih zavisnosti nad U
$IFD(U)$	Skup svih izvedenih F-zavisnosti nad U
$SP_{\mathcal{F}}$	Skup svih pojava tipa forme
$P(N_i)$	Skup svih pojava tipa objekta unutar iste pojave tipa forme
$Tuple(U)$	Skup svih torki nad U
$\mathcal{X}(N_i)$	Skup svih unija po jednog ključa od korenskog do tipa objekta N_i
$MVD(U)$	Skup svih višeznačnih zavisnosti nad U
$JD(U)$	Skup svih zavisnosti spoja nad U
$Ext(R, \mathcal{K})$	Skup šema relacija za koje $(R, \mathcal{K})$ predstavlja proširenje
$\mathcal{SF}u$	Skup tipova formi za ažuriranje
$\mathcal{SF}r$	Skup tipova formi za izveštavanje (upite)
$\mathcal{O}(\mathcal{F})$, $\mathcal{O}$	Skup tipova objekata tipa forme
$null(R_i, A)$	Specifikacija dozvole (zabrane) nula vrednosti
$dbs(A)$	Specifikacija pripadnosti obeležja šemi baze podataka

$el(A)$	Specifikacija elementarnosti obeležja
$man(\mathcal{F}, A)$	Specifikacija obaveznosti obeležja na tipu forme
ψ	Struktura stabla tipa forme

–Š–

$N(S, I)$	Šema baze podataka informacionog sistema
$N_m(S_m, I_m)$	Šema modula baze podataka
(R, C)	Šema relacije
(S, I)	Šema relacione baze podataka

–T–

$T(U)$	Tablo, reprezent
$\mathcal{F}$	Tip forme
$A_{\mathcal{F}}(u)$	Tip forme za ažuriranje
$A_{\mathcal{F}}(r)$	Tip forme za izveštavanje
$N(Q, \mathcal{K})$	Tip objekta
t, t_i, u, v	Torka (N-torka)
$\tilde{G} = (S, \tilde{\rho})$	Tranzitivni graf zatvaranja
3NF	Treća normalna forma

–U–

etgd	Ugrađena generalizovana T-zavisnost
ejd	Ugrađena zavisnost spoja
DomU	Unija svih domena nad U
U	Univerzalni skup obeležja

–V–

$X \to\to Y$, $X \to\to Y \mid Z$, *mvd*	Višeznačna zavisnost

–Z–

$\mu_{\xi_i \to A}[\xi_j]$	Zamena obeležja A skupom lanaca ξ_i
Γ^+	Zatvarač skupa funkcionalnih zavisnosti
Γ^{ifd}	Zatvarač skupa izvedenih F-zavisnosti
Γ^*	Zatvarač skupa generalizovanih zavisnosti
X_Γ^+	Zatvarač skupa obeležja
$R_1[X] \subseteq R_2[Y]$, *ind*	Zavisnost sadržavanja
$\rhd\lhd(X_1,...,X_n)$, *jd*	Zavisnost spoja
$J(\mathcal{F})$, $J_{ex}(\mathcal{F})$	Zavisnost spoja tipa forme

9. Prilog C

Indeks

–E–

–F–

–G–

–H–

–I–

–K–

–L–

–M–

–N–

–O–

–P–

—R—

—S—

–Š–

–T–

–U–

–V–

–Z–

Objašnjenje UDK - Univerzalna Decimalna Dokumentacija:

519.688:519.683:681.3.06→519.58:510.51/.53

519.688	**Programi i algoritmi**
519.683	**Programske metode i sredstva**
681.3.06→519.58	**Softver**
510.51	**Teorija algoritama**
510.52	**Složenost algoritama**
510.53	**Algoritamski problemi**