

MIGRACIJA POSLOVNOG OBJEKTNOG MODELA U RELACIONU BAZU PODATAKA

Dražen Brđanin, Slavko Marić, Ognjen Čalić, *Elektrotehnički fakultet Banja Luka*
Siniša Macan, *Direkcija za implementaciju CIPS projekta*
Krstan Bošnjak, *Slobomir P univerzitet*

Sadržaj - U ovom radu opisana je migracija poslovnog objektnog modela u relaciju bazu podataka. Kao polazna osnova uzet je dijagram biznis (poslovnih) klasa iz poslovnog objektnog modela realizovanog u okviru poslovnog modela na bazi biznis profila UML-a. Cilj migracije je relacioni model baze podataka na bazi UML profila za projektovanje baze podataka. Opisana metodologija ilustrovana je na primjeru sistema za migracije stranaca.

1. UVOD

UML [1] je jezik za vizuelno modelovanje softverskih sistema. Postao je vodeći jezik modelovanja implementiran u većini CASE alata i danas, *de facto*, predstavlja industrijski standard za modelovanje softvera. Standardni UML predstavlja otvoren sistem. Osnovni koncept moguće je proširivati definisanjem korisničkih proširenja, odnosno specijalizovati za određenu oblast uvođenjem profila. Postoji više profila, kao što su npr. profil za poslovno modelovanje, profil za RT sisteme, profil za projektovanje baze podataka i dr.

U ovom radu opisana je migracija iz poslovnog objektnog modela, realizovanog pomoću profila za poslovno modelovanje, u relacioni model baze podataka, realizovan pomoću profila za projektovanje baze podataka. Osim same metodologije za transformaciju dijagrama poslovnih klasa u relacionu šemu baze podataka, višena je i analiza pojedinih aspekata te transformacije.

2. POSLOVNI OBJEKTNI MODEL

Poslovni model, prema biznis profilu UML-a [1], organizovan je u dva modela: **model poslovnih slučajeva upotrebe i poslovni objektni model**.

Model poslovnih slučajeva upotrebe predstavlja pogled na poslovni sistem izvana i opisuje poslovni sistem i njegove odnose sa vanjskim sistemima kroz poslovne slučajeve upotrebe (PSU). PSU se može smatrati poslovnim procesom, odnosno konkretnom funkcijom poslovnog sistema koju on obezbjeđuje vanjskim sistemima – poslovnim učesnicima, tako što kroz niz akcija učesniku stvara neku konkretnu prepoznatljivu vrijednost. Model PSU sadrži opise učesnika i PSU, te njihove interakcije, što se prikazuje dijagramima PSU. Model PSU, pored specifikacije sadrži i odgovarajući opis realizacije identifikovanih PSU. Realizacija PSU, grupe ili sistema PSU, dokumentuje se tekstualno i grafički [2]. Tekstualni opis podrazumijeva specifikaciju karakteristika i aktivnosti datog procesa. Za grafičko predstavljanje toka aktivnosti mogu da se koriste dijagram aktivnosti [1] ili dijagram sekvence [3]. Priroda procesa (tačno definisan niz akcija) upućuje na korištenje dijagrama aktivnosti. Najčešće se koristi dijagram aktivnosti visokog nivoa [2], kojim se prikazuju samo makroaktivnosti, bez zalaženja u detalje (odgovornosti radnika za sprovođenje aktivnosti i objekti koje ti radnici koriste). Dodatno se mogu uvesti plivačke staze da bi se prikazao raspored aktivnosti po organizacionim jedinicama ili radnim grupama, što se preporučuje kod saradnje većeg broja organizacionih jedinica.

Poslovni objektni model predstavlja pogled na poslovni sistem iznutra i u potpunosti opisuje procese, procedure i ponašanja radnika, te korištene resurse i njihovu organizaciju. Poslovni objektni model obično uključuje detaljne dijagrame aktivnosti, dijagrame interakcije i dijagrame biznis klasa [2]. Potpun opis procesa daje se detaljnim dijagramima aktivnosti - dijagrami makroaktivnosti upotpunjeni odgovornostima radnika, korištenim objektima i objektnim tokovima. Područja odgovornosti radnika modeluju se korištenjem plivačkih staza. Dijagram aktivnosti je proceduralno orijentisan i nije prilagođen za objektno orijentisanu analizu i dizajn sistema. Zbog toga poslovni objektni model uključuje i neki od OO dijagrama interakcije, dijagram saradnje ili dijagram sekvence [4]. Dijagram saradnje podesan je za fokusiranje na strukturalnu komponentu, ali se ne preporučuje, zbog nedovoljne preglednosti, u slučaju složenijih procesa. Dijagram sekvence omogućava fokusiranje na vremensku komponentu procesa. Organizacijom dosta podsjeća na dijagram aktivnosti, pa se preporučuje za korištenje u objektnom modelu. Da bi se u narednim fazama projektovanja IS omogućilo lakše modelovanje klasa, poslovni objektni model uključuje i dijagram biznis klasa. Dijagram biznis klasa prikazuje statičku strukturu i veze između klasa, i kao takav predstavlja osnovu za projektovanje relacione baze podataka.

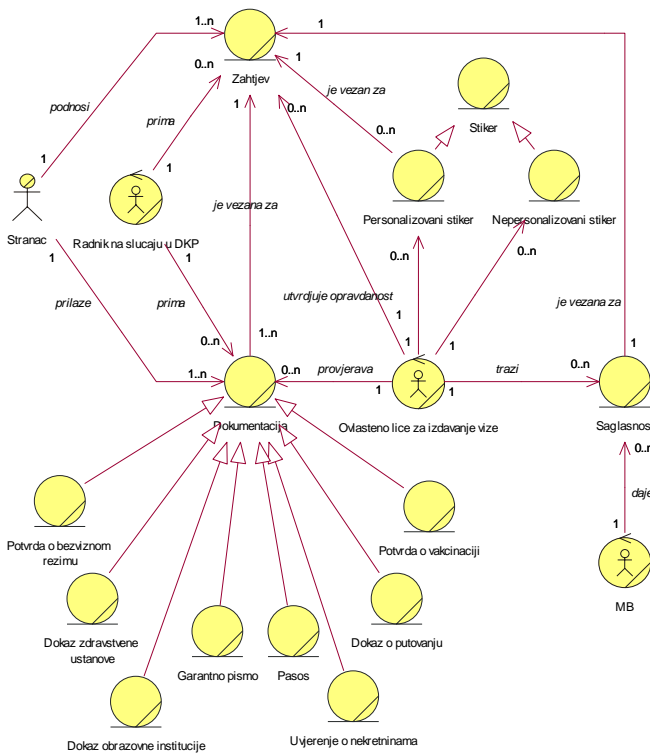
UML kao standardni jezik za modelovanje propisuje notaciju i semantiku, ali ne i standardni proces modelovanja. U literaturi je predloženo više postupaka, a prema [5] proces se odvija kroz sljedeće faze:

1. specifikacija PSU (dijagram PSU),
2. tekstualni opis realizacije,
3. grafički prikaz realizacije (dijagram makroaktivnosti),
4. detaljan opis realizacije (detaljan dijagram aktivnosti),
5. dijagram sekvence,
6. dijagram biznis klasa.

Identifikovani PSU i poslovni učesnici, radnici i poslovni objekti mogu da se iskoriste za **prevođenje poslovnih modela u polazne sistemske modele**. Model PSU prevodi se u sistemski model slučajeva upotrebe, a poslovni objektni model u sistemski model dizajna. Pri prevođenju modela PSU, svaki PSU je kandidat za sistemski slučaj upotrebe, a svaki poslovni učesnik i svaki radnik je potencijalni sistemski učesnik. Pri prevođenju objektnog modela, svaki identifikovani entitet (učesnici, radnici i objekti) je kandidat za sistemsku klasu, a svaka poruka u dijagramu sekvence je potencijalna metoda sistemske klase.

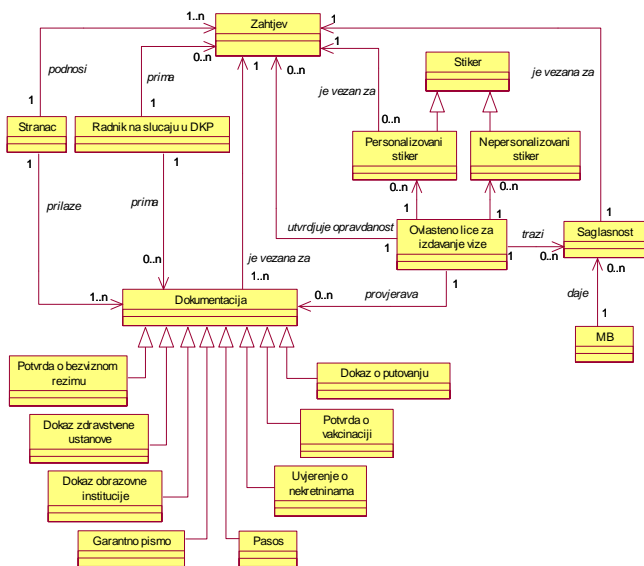
3. MIGRACIJA DIJAGRAMA BIZNIS KLASA U DIJAGRAM KLASA

Predmet ovog rada je migracija poslovnog objektnog modela, odnosno dijagrama biznis klasa, u relacionu šemu baze podataka. Dijagram biznis klasa kreira se u posljednjoj fazi poslovnog modelovanja kao apstrakcija svih identifikovanih objekata i objektnih tokova te njihovih međusobnih veza kao i veza sa učesnicima i radnicima u sistemu. Za ilustraciju je uzet jedan segment dijagrama biznis klasa sistema za izdavanje viza iz poslovnog modela sistema za migracije stranaca [6], slika 1.



Sl. 1. Polazni dijagram biznis klasa.

Prvi korak prema relacionoj bazi podataka je migracija dijagrama biznis klasa u dijagram klasa ciljnog sistema. Uvažavajući prethodno navedena pravila za transformaciju poslovnih modela u sistemske modele, direktnim mapiranjem svih poslovnih učesnika, radnika i korištenih resursa (objekata) iz dijagrama biznis klasa u sistemske klase, dobijamo dijagram klasa prikazan na slici 2.

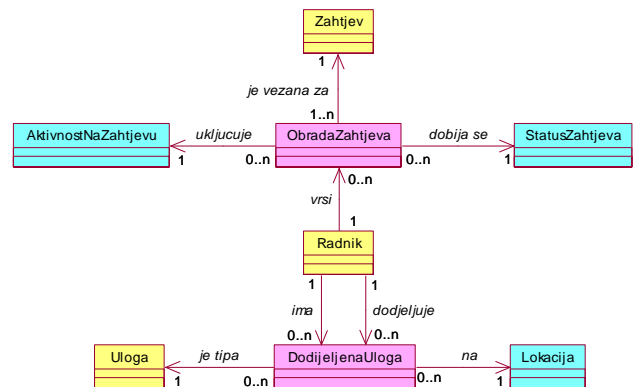


Sl. 2. Dijagram klasa dobijen direktnim mapiranjem biznis klasa.

Dijagram klasa dobijen direktnim mapiranjem biznis klasa treba da prođe višestruka rafiniranja. Prvu modifikaciju zahtijevaju veze generalizacije/specijalizacije klasa. U datom primjeru, imamo generalizaciju priloga zahtjevu. Vidljivo je da u sistemu postoji veći broj različitih priloga, a dodatni problem predstavljaju i potencijalno novi oblici ili vrste priloga, koji mogu da se pojave u praksi s obzirom na prirodu problema (nova zakonska regulativa i sl). Ciljna struktura treba da je

dovoljno fleksibilna da omogući manipulaciju različitim i proširljivim skupom dokumenata, a istovremeno da obezbijedi nadležnim organima mogućnost primjene različitih analitičkih tehnika i pretraživanje priloženih dokumenata. Detaljnom analizom poslovnih objekata i korisničkih zahtjeva, dolazi se do zaključka da veći broj različitih dokumenata, iako heterogene prirode, može da se svrsta u nekoliko tipičnih grupa i predstavi sa tri značajne klase. Ovo grupisanje izvedeno je po porijeklu i značaju priloga, kao što je prikazano na slici 4. Dodatno je uvedena klasa koja predstavlja odgovarajuću nomenklaturu – šifarnik vrste priloga. Osim ove klase, biće uvedene još neke klase, ali o dodatnim klasama biće riječi kasnije, jer one predstavljaju treću značajnu modifikaciju dijagrama klasa.

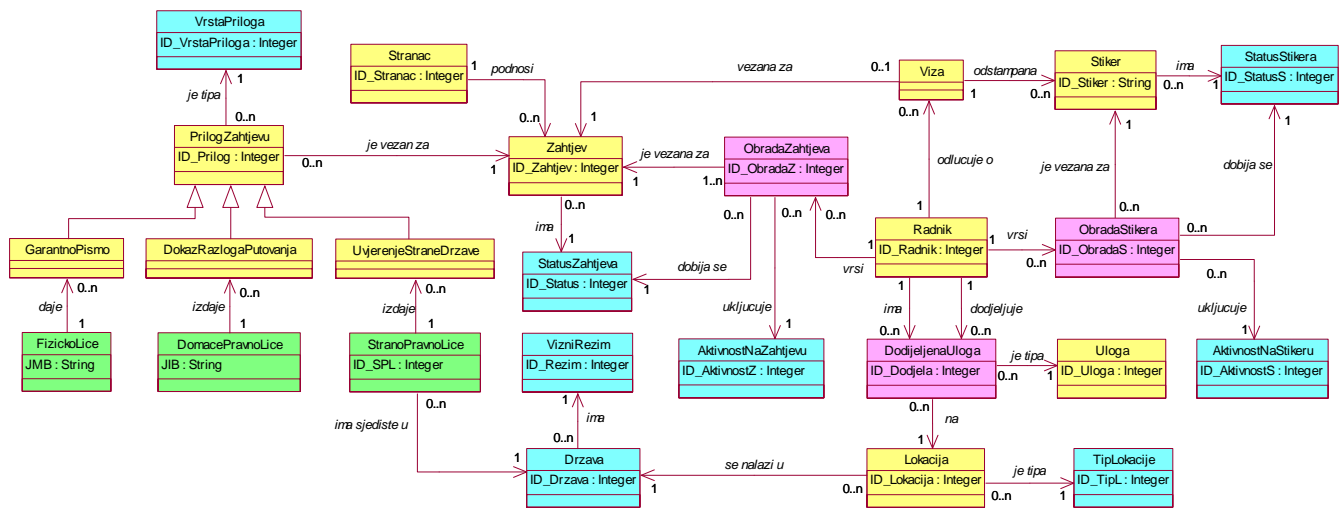
Dругu značajnu modifikaciju dijagrama klasa uslovljava veći broj različitih uloga u sistemu: lice ovlašteno za prijem zahtjeva, lice ovlašteno za izdavanje vize kao i druge uloge (koje nisu prikazane na ovom dijagramu), te veći broj različitih aktivnosti nad istim poslovnim objektima. Osim toga, različite uloge mogu biti povjerene istim radnicima, ali isto tako, postojeće uloge mogu zbog prirode sistema (širok spektar različitih aktivnosti) biti redefinisane kroz skup novih uloga u sistemu koje mogu biti povjerene različitim radnicima u okviru različitih organa ili organizacionih jedinica. Dodatni problem mogu da predstavljaju i potencijalno nove aktivnosti nad istim poslovnim objektima. Ciljna baza podataka treba da ima dovoljno robusnu i fleksibilnu strukturu tako da omogućava efikasne izmjene i u pogledu skupa različitih uloga i u pogledu skupa različitih aktivnosti nad poslovnim objektima. Moguće rješenje ovog problema prikazano je na slici 3.



Sl. 3. Pojednostavljenje problema heterogenog skupa aktivnosti i uloga u sistemu.

Da bi se omogućilo praćenje statusa zahtjeva nakon svake obrade, dodatno je uvedena i odgovarajuća nomenklatura statusa zahtjeva. Osim toga, da bi se postiglo brže dobijanje trenutnog statusa zahtjeva može se uvesti i redundantna veza između zahtjeva i statusa zahtjeva, ali o redundantnim vezama biće riječi kasnije, jer one predstavljaju četvrtu značajnu modifikaciju dijagrama klasa. Isti princip iskorišten je i za praćenje aktivnosti nad ostalim poslovnim objektima u sistemu. Interesantna je i dvostruka asocijacija radnika i dodijeljene uloge u sistemu. Prva asocijacija predstavlja vezu između radnika i uloge koju on ima na nekoj lokaciji, a druga asocijacija predstavlja vezu sa radnikom koji mu je tu ulogu dodijelio.

Sljedeća faza u projektovanju dijagrama klasa jeste upotpunjavanje dijagrama klasa dodatnim klasama, kao što su odgovarajuće nomenklature ili druge potrebne klase. Ova faza direktno je u vezi sa identifikacijom potrebnih atributa. Po završetku ove faze, dijagram klasa može da se okarakterise kao FA (*fully attributed*) model.



Sl. 4. Modifikovani dijagram klasa.

Posljednju fazu predstavlja identifikacija i uspostavljanje redundantnih asocijacija između pojedinih klasa, kako bi se omogućilo brže pretraživanje i dobijanje potrebnih informacija u eksploataciji baze podataka.

Na slici 4 dat je potpuni dijagram klasa dobijen prethodno opisanim modifikacijama polaznog dijagrama klasa. Iz objektivnih razloga prikazani su samo oni atributi koji su bitni za ilustraciju kreiranja relacione šeme. Cilj ovog rada jeste relaciona šema baze podataka, pa je FA model krajnji cilj, bez potrebe za definisanjem metoda u klasama.

4. MIGRACIJA U RELACIONU ŠEMU

UML, kao jezik za objektno-orientisano modelovanje, nije u potpunosti prilagođen za projektovanje i modelovanje relacione baze podataka. Zato je razvijen poseban profil za projektovanje baza podataka [7]. Ovaj profil nije ugrađen u standardni UML, ali je implementiran u nekim CASE alatima (Rational Rose) i omogućava automatizovanu migraciju FA dijagrama klasa u relacionu šemu baze podataka. Automatizovano generisanje relacione šeme iz FA dijagrama klasa samo je jedna od mogućnosti profila za projektovanje baze podataka. Za razliku od standardne ER (*entity-relationship*) notacije, ovaj profil omogućava i modelovanje drugih koncepata, kao što su pogledi, trigeri, pohranjene procedure, indeksi, ograničenja i dr. Osim logičkog projektovanja baze, omogućeno je fokusiranje i na aspekte fizičke implementacije baze podataka, kao što su npr. distribuiranost, segmentiranost itd. Predmet ovog rada je migracija poslovnog objektnog modela u relacionu šemu, pa je pažnja usmjerena samo na modelovanje strukture.

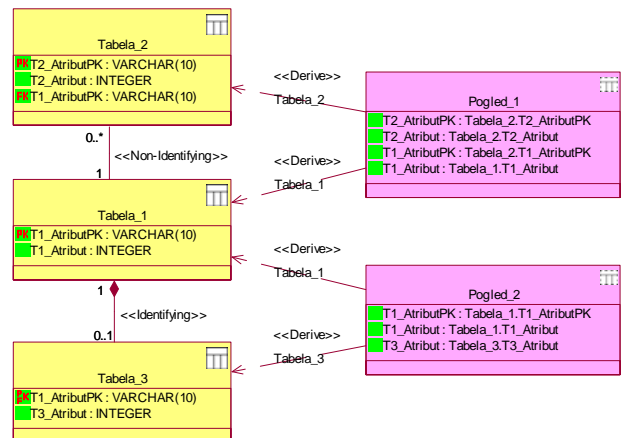
Profil za projektovanje baze podataka omogućava kreiranje i manipulaciju modelom baze. Uveden je i novi tip dijagrama – dijagram baze podataka (*database diagram*). Ovaj tip dijagrama omogućava korištenje koncepata karakterističnih za bazu podataka (tabele, pogledi, relacije, ključevi, domeni ...). Osim dijagrama baze podataka, model baze podataka omogućava korištenje i drugih, netipiziranih, dijagrama, u kojima mogu da se koriste i neki drugi koncepti, kao što su koncepti karakteristični za sistemski model implementacije ili model dizajna.

Koncepti uvedeni profilom za projektovanje baze podataka, ilustrovani su na generičkom dijagramu baze podataka, datom na slici 5. Tabela (detalj ①) se modeluje kao klasa sa stereotipom <<Table>>. Za ilustraciju ovog značajnog koncepta kreiran je i poseban simbol (*icon*). Prilikom kreiranja relacione šeme iz

dijagrama klasa, sve klase se automatski prevode u tabele, a svi atributi klase u kolone tabele. Kolone su modelovane stereotipom <<Column>>. Kolona koja predstavlja primarni ključ označava se sa PK (⑤). Prilikom automatizovanog kreiranja šeme iz dijagrama klasa, primarni ključevi automatski se kreiraju na osnovu atributa koji su označeni kao atributi koji identifikuju klasu.

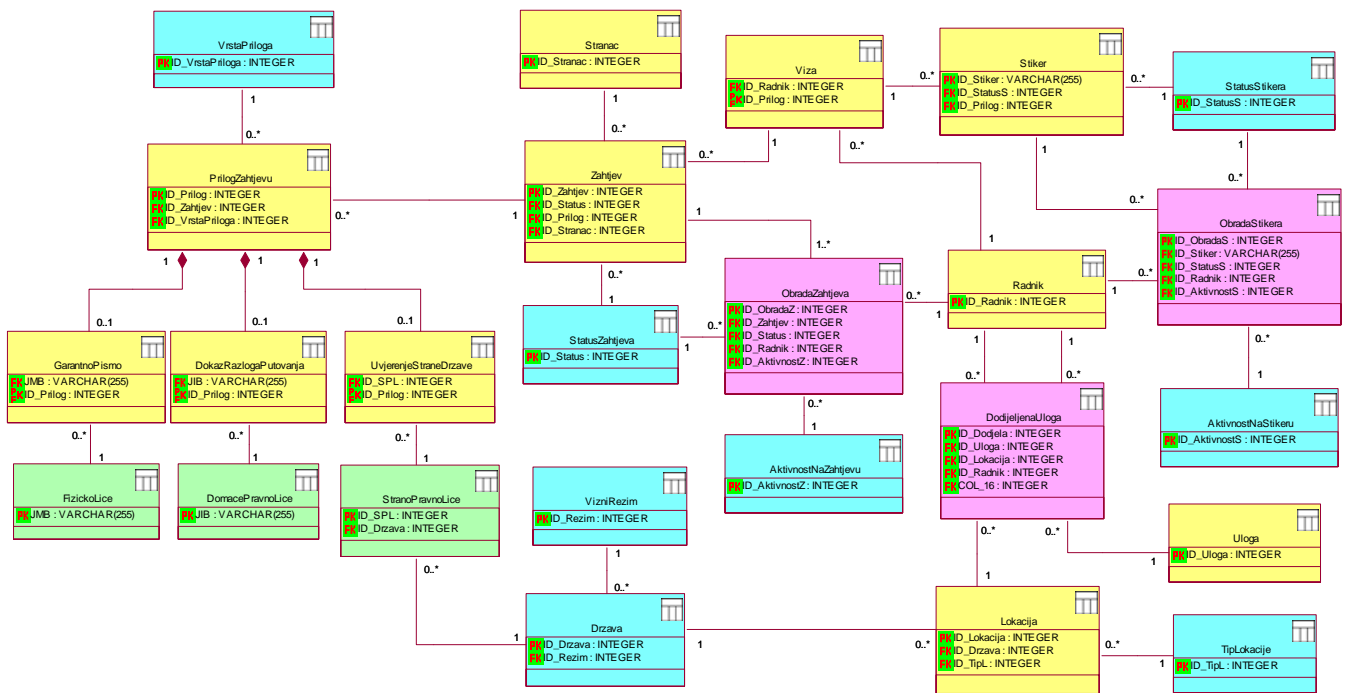
Drugi značajan koncept predstavlja pogled (*view*), koji se modeluje kao klasa sa stereotipom <<view>> (②). Pogled se povezuje sa tabelama nad kojima je definisan kao zavisnost sa stereotipom <<derive>>. Omogućeno je potpuno modelovanje pogleda, uz definisanje odgovarajućeg SQL upita.

Između pojedinih tabela moguće je uspostaviti identifikujuću odnosno neidentifikujuću relaciju. Neidentifikujuća relacija (③) modeluje se kao <<Non-Identifying>> asocijacija. Ova relacija obezbeđuje automatsku migraciju odgovarajućeg primarnog ključa iz tabele na strani "1" u strani ključ u tabeli na strani "0..*" (⑥). Strani ključevi u tabeli označavaju se sa FK. Identifikujuća relacija (④) modeluje se kao agregacija sa stereotipom <<Identifying>>. Ova relacija vrši automatsku migraciju primarnog ključa iz tabele na strani "1" u primarni/strani ključ (PFK) u zavisnoj tabeli (⑦).



Sl. 5. Generički dijagram baze podataka.

Na osnovu dijagrama klasa sa slike 4, uz prethodno opisane principe, dobija se relaciona šema baze podataka, prikazana na dijagramu baze podataka datom na slici 6. Pošto su na dijagramu klasa prikazani samo atributi koji identifikuju klase, na dijagramu baze podataka imamo tabele samo sa ključevima.



Sl. 6. Dijagram baze podataka za generisanu relacionu šemu.

5. ZAKLJUČAK

Iako prvenstveno namijenjen za vizuelno modelovanje softverskih sistema, zahvaljujući biznis profilu UML može uspješno da se koristi i za poslovno modelovanje, jer bogata notacija i semantika omogućavaju analizu svih aspekata poslovnog domena. Poseban značaj ima poslovno modelovanje UML-om ako je to osnova za razvoj informacionog sistema. Tada je notacija za poslovno modelovanje usklađena sa notacijom za sistemsko modelovanje, a korišteni koncepti omogućavaju lak prelazak iz poslovnog modela u sistemske modele. Ova migracija ilustrovana je transformacijom dijagrama biznis klasa u sistemski dijagram klasa.

Osim pravila za mapiranje dijagrama biznis klasa u sistemski dijagram klasa, ilustrovane su i neke naknadne karakteristične modifikacije dijagrama klasa sve do realizacije FA dijagrama, kao osnove za projektovanje relacione baze podataka.

Iako je UML jezik prevashodno namijenjen za OO analizu i dizajn, zahvaljujući profilu za projektovanje baze podataka, omogućeno je veoma jednostavno projektovanje relacione baze podataka, a sam proces generisanja relacione šeme na osnovu dijagrama klasa potpuno je automatizovan, što je u ovom radu i ilustrovano.

U odnosu na tradicionalnu ER notaciju, profil za projektovanje baze podataka omogućava modelovanje i pogleda, trigera, pohranjenih procedura itd. Osim modelovanja logičke strukture baze podataka, profil za projektovanje baze podataka omogućava i modelovanje nekih aspekata fizičke implementacije baze.

U ovom radu pokazana je metodologija za projektovanje relacione baze podataka uz isključivo korištenje UML-a od poslovnog modelovanja do modelovanja baze podataka, čime je, zahvaljujući odgovarajućim profilima standardnog jezika, pokazana efikasnost i jednostavnost u povezivanju i transformaciji različitih modela tokom procesa projektovanja.

LITERATURA

- [1] OMG, *Unified Modeling Language*, v1.4, 2001.
- [2] G. Övergaard, B. Selic, C. Bock, *Object Modeling with UML: Behavioral Modeling*, OMG, 2000.
- [3] K. Palmkvist, B. Selic, J. Warmer, *Object Modeling with UML: Advanced Modeling*, OMG, 2000.
- [4] N. Pan-Wei, *Effective Business Modeling with UML: Describing Business Use Cases and Realization*, Rose Architect e-zine, Novembar 2002.
- [5] D. Brđanin, S. Marić, K. Bošnjak, "Poslovno modelovanje na bazi biznis profila UML-a", *Zbornik radova XLVII Konferencije za ETRAN*, Herceg Novi, 2003, tom III, str. 140-143.
- [6] S. Marić, D. Brđanin, O. Čalić, *Informacioni sistem migracija u BiH: poslovni model podsistema za izdavanje viza*, Banja Luka, 2005.
- [7] E. Naiburg, R. Maksimchuk, *UML for Database Design*, Addison-Wesley, 2001.

Abstract – In this paper, one approach for the migration of the business object model into the relational database model has been presented. Business class diagram from business object model, based on UML business profile, has been taken as the basis for that migration. The target is the relational database model. The database design has been done by UML profile for database design. Described methodology is illustrated taking the system for foreigner's migration evidence as an example.

MIGRATION OF THE BUSINESS OBJECT MODEL INTO THE RELATIONAL DATABASE

Dražen Brđanin, Slavko Marić, Ognjen Čalić,
Siniša Macan, Krstan Bošnjak