

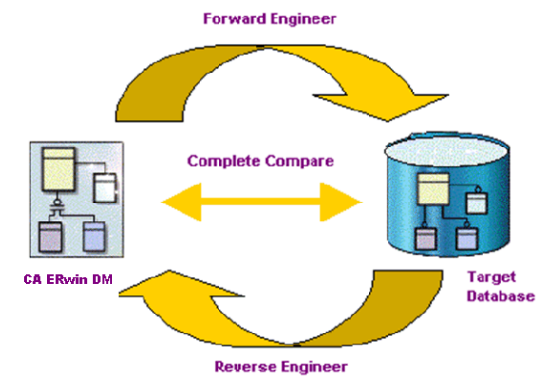
ERWin

SI4SAB



Šta je CA ERwin Data Modeler?

- Alat baziran na Windows grafičkom korisničkom interfejsu
- Kreiranje i održavanje modela relacione baze podataka
 - Logical data model – model poslovnih pravila i zahtevnosti
 - Physical data model – model za ciljni server i bazu podataka
- Forward engineering
 - Model -> Baza podataka
- Reverse engineering
 - Baza podataka -> Model
- Migracija sa jednog ciljnog servera na drugi
- Complete Compare technology
 - Upoređivanje modela sa bazom i prikaz razlika



Notacije

Cardinality Description	IDEF1X Notation		IE Notation	
	Identifying	Non-identifying	Identifying	Non-identifying
One to zero, one, or more				
One to one or more				
One to zero or one				
Zero or one to zero, one, or more (non-identifying only)				
Zero or one to zero or one (non-identifying only)				

Osnovni koncepti modelovanja

- Tipovi modela
- Entiteti
- Tabele
- Atributi i kolone
- Primarni i strani ključevi
- Odnosi

Tipovi modela

Logical

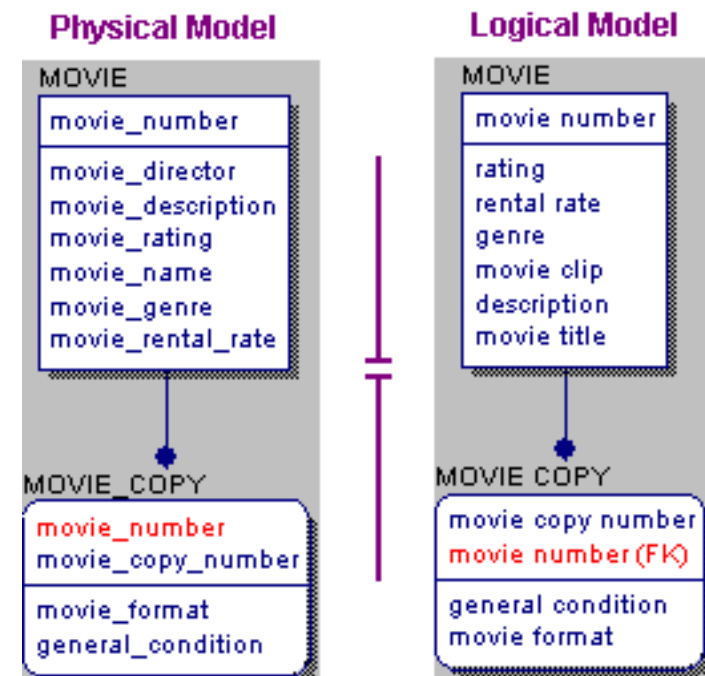
Konceptualni model koji sadrži objekte kao što su entiteti, atributi, grupe ključeva.

Physical

Model vezan za bazu podataka i sadrži objekte kao što su tabele, kolone, tipovi podataka.

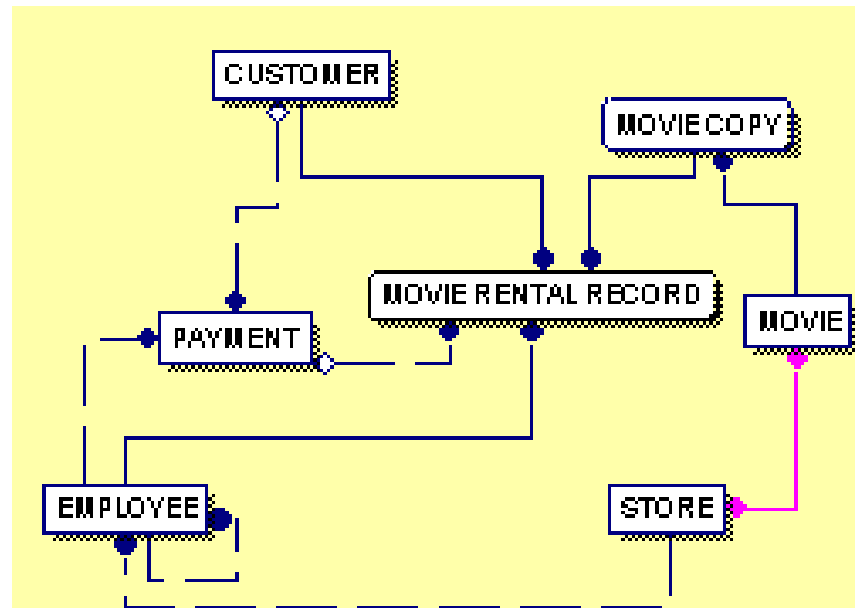
Logical/Physical

Model koji sadrži logički i fizički model.



Entiteti

Entitet je logički objekat koji predstavlja osobu, mesto, ili stvar o kojoj organizacija održava informacije.



Tabele

Tabela u fizičkom modelu odgovara entitetu u logičkom. U tabelama se čuvaju podaci u bazi podataka.

Customer Table in ERwin

CUSTOMER		
cust_no		
name		
address		

Customer Table in the database

cust_no	name	address
1234	Brown	5 Main St
2345	Jones	129 C
3456	Smith	77 W

Tabele

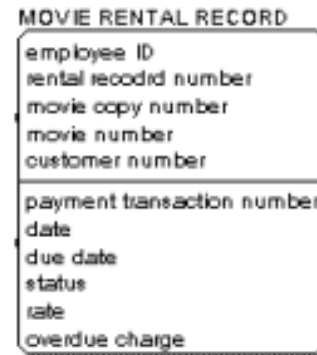
Postoje dve vrste entiteta/tabela:

- Independent Entity
 - Instance možemo odrediti bez određivanja odnosa sa drugim entitetima.
- Dependent Entity
 - Instance ne možemo odrediti bez određivanja odnosa sa drugim entitetima.

Independent Entity

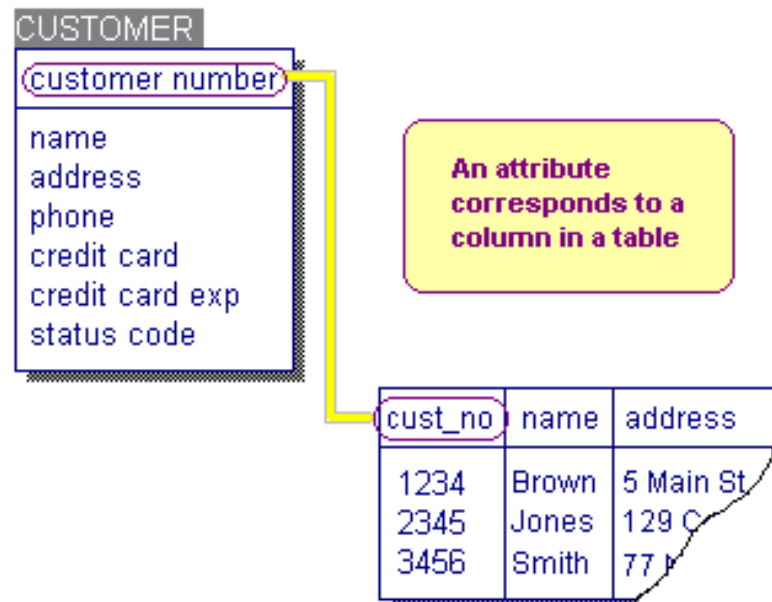


Dependent Entity

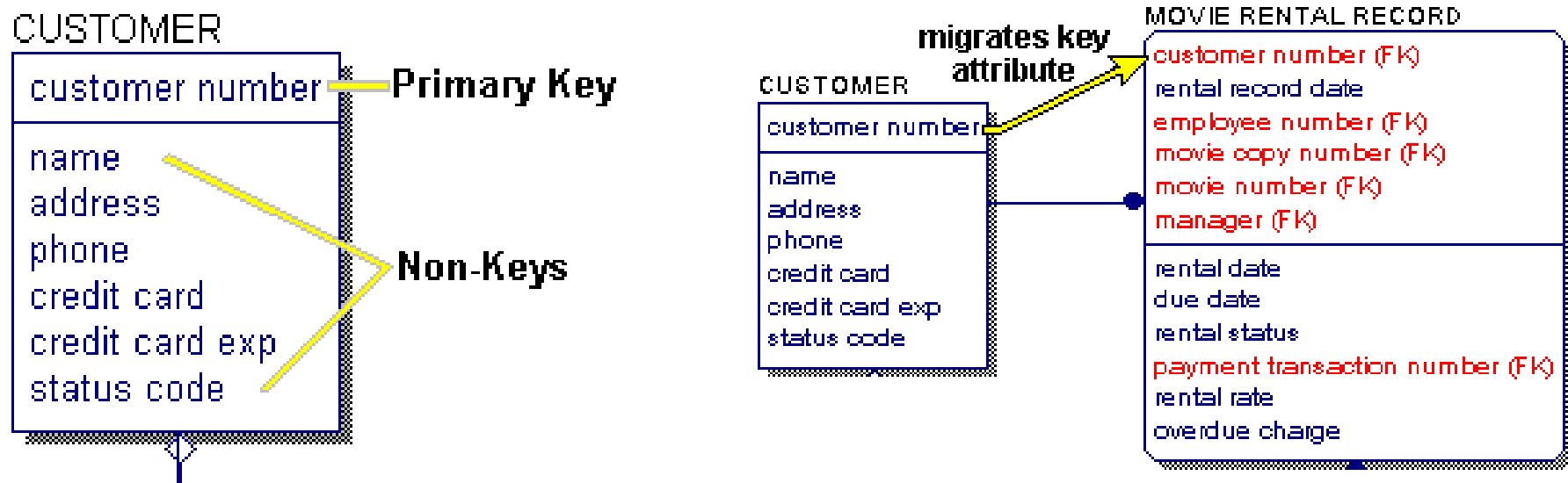


Atributi i kolone

Atributi u logičkom i kolone u fizičkom modelu.

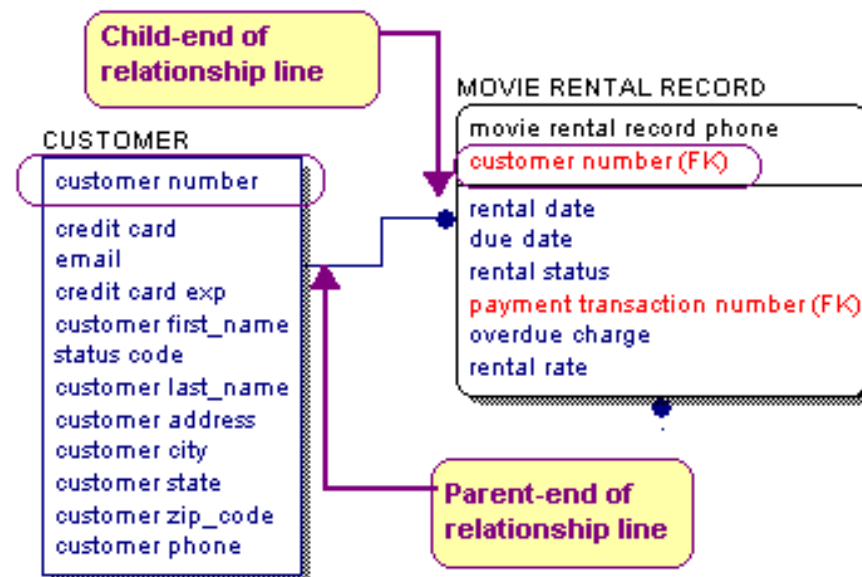


Primarni i strani ključevi



Odnosi

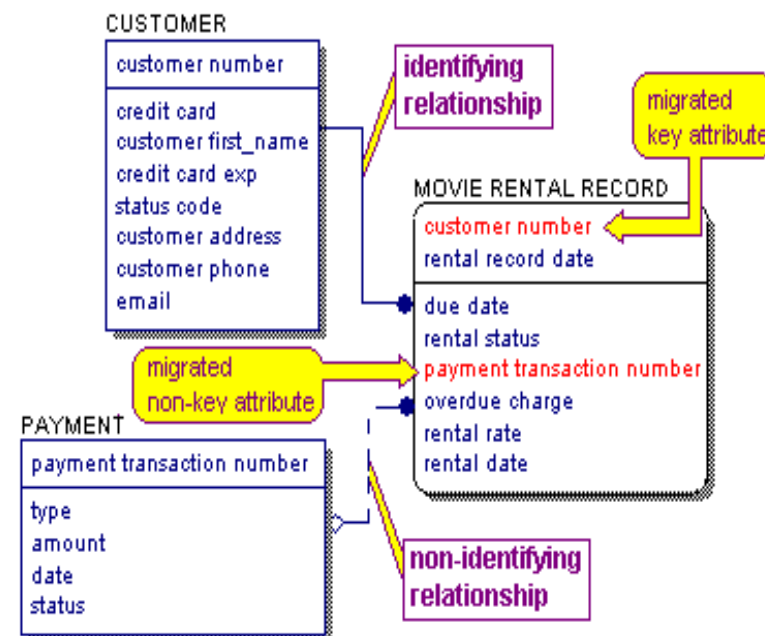
Predstavljani su punim ili isprekidanim linijama između entiteta.



Odnosi

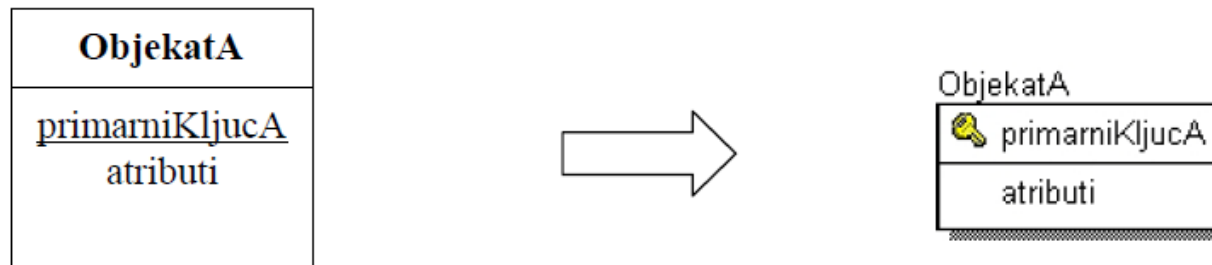
Postoje dva tipa odnosa:

- Identifying Relationship
 - Prikazan punom linijom
 - Primarni ključ roditelja migrira u primarni ključ deteta kao strani ključ
- Non-Identifying Relationship
 - Prikazan isprekidanom linijom
 - Primarni ključ roditelja migrira u ne-ključnu oblast deteta i postaje običan atribut



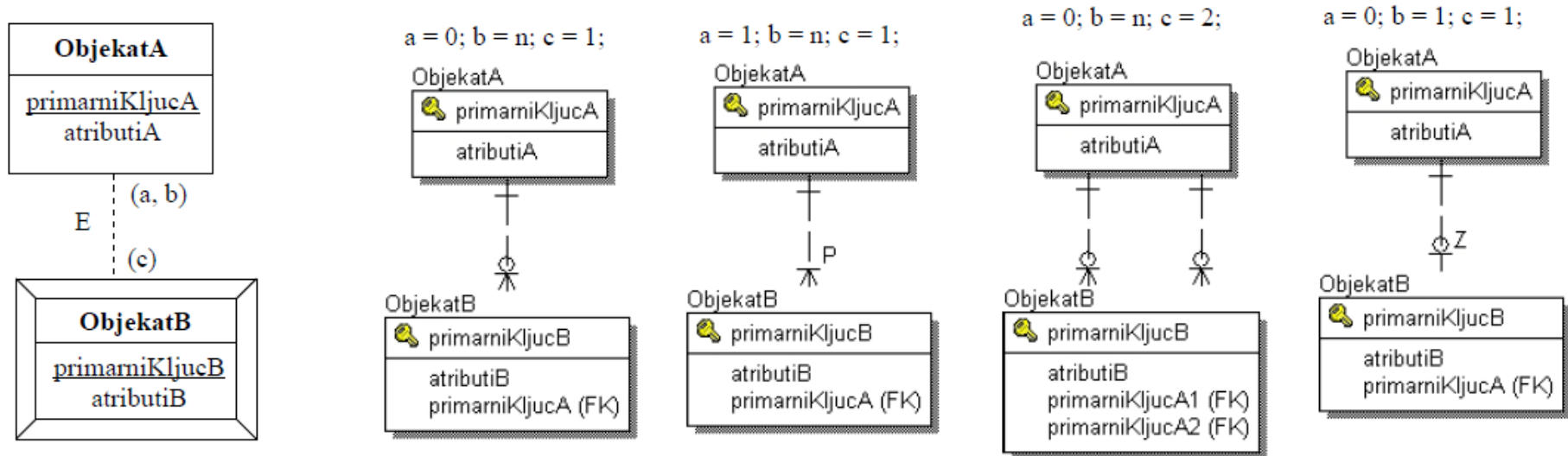
Pravila preslikavanja iz ER u IE

Nezavisni(jaki) ovjekti

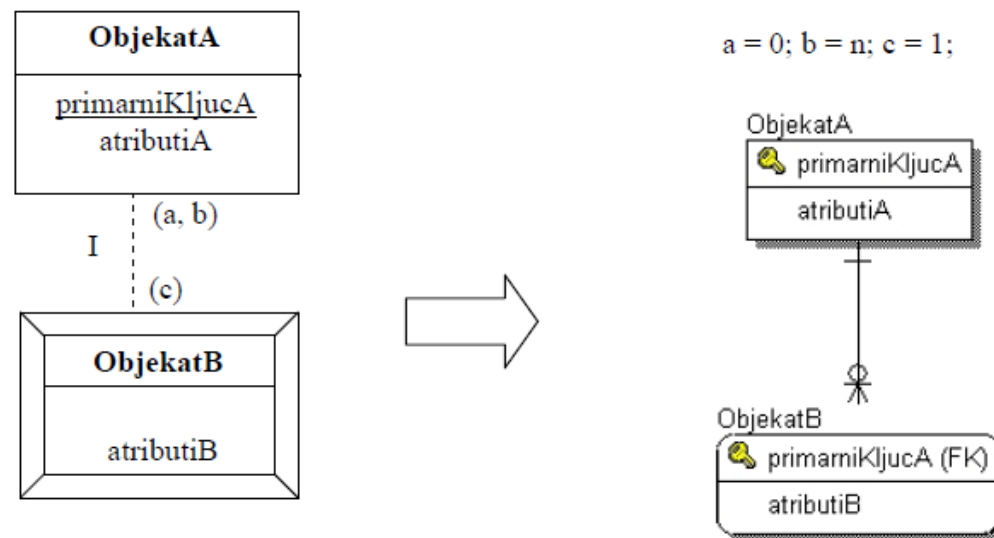


Pravila preslikavanja iz ER u IE

Zavisni (slabi) objekti



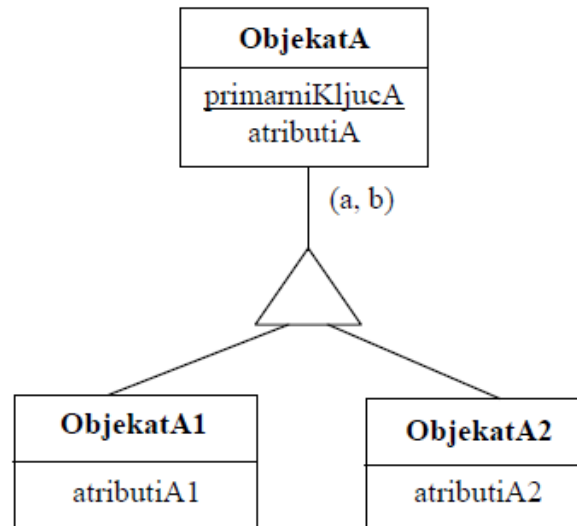
Pravila preslikavanja iz ER u IE



Ostali slučajevi izgledaju isto kao kod egzistencijalne zavisnosti, s tom razlikom što je linija veze puna i entitet ObjekatB izgleda kao na prethodnoj slici.

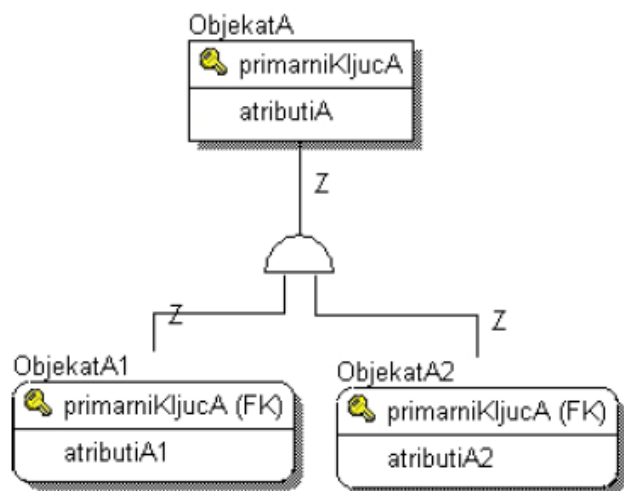
Pravila preslikavanja iz ER u IE

Specijalizacija

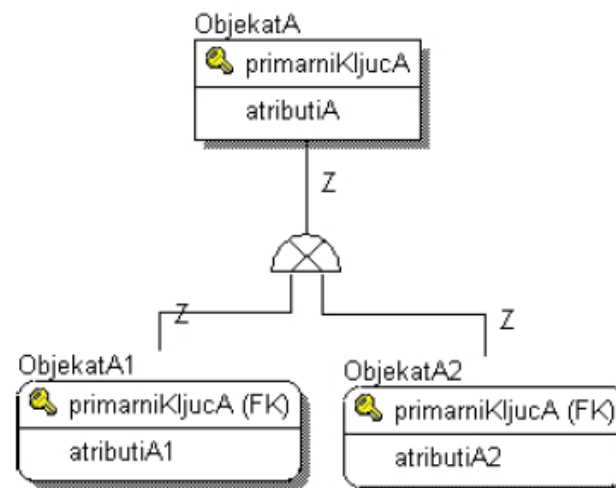


Pravila preslikavanja iz ER u IE

$b > 1$, inkluzivna

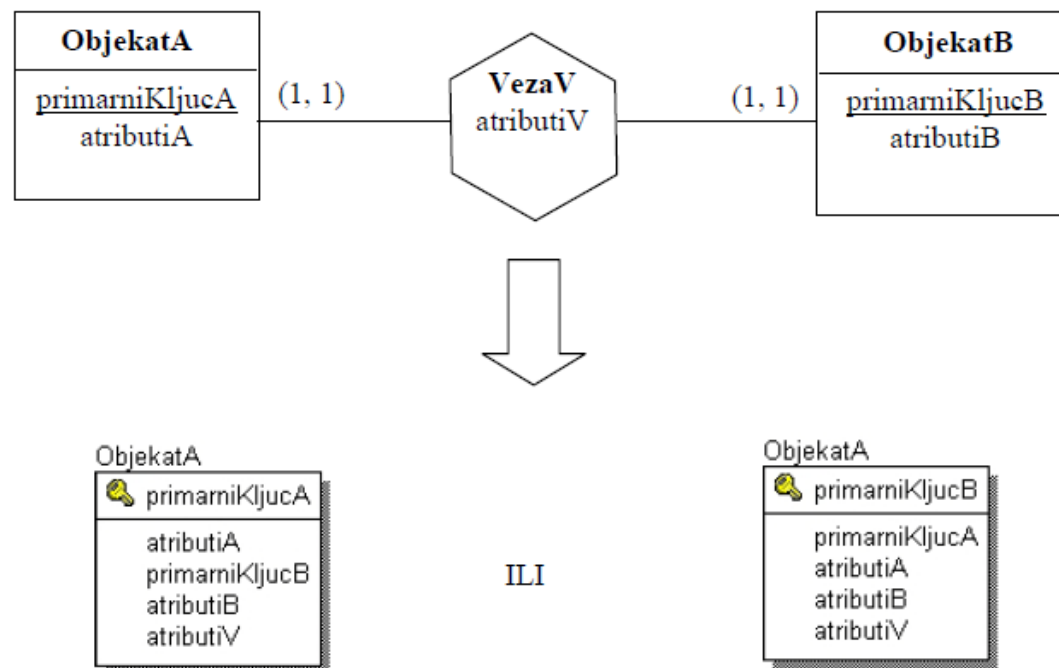


$b = 1$, ekskluzivna

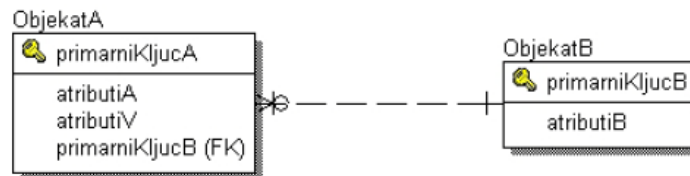
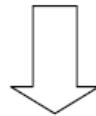
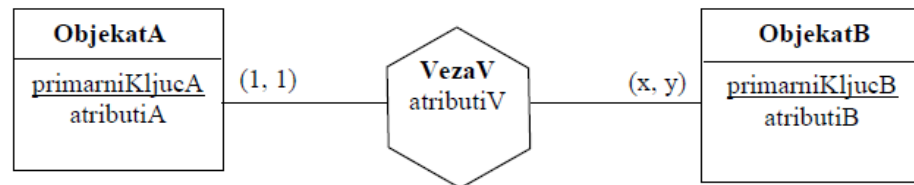


Pravila preslikavanja iz ER u IE

Veze

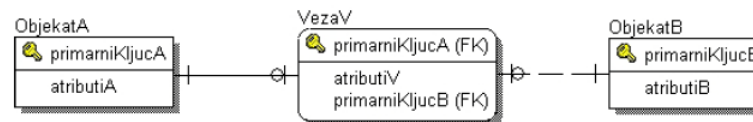
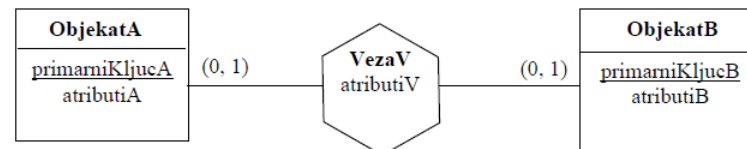


Pravila preslikavanja iz ER u IE

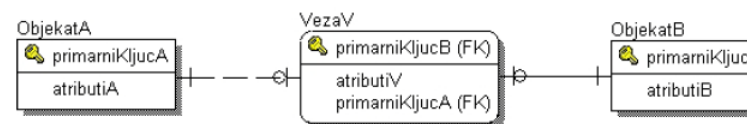


ako je $x = 0, y = n$

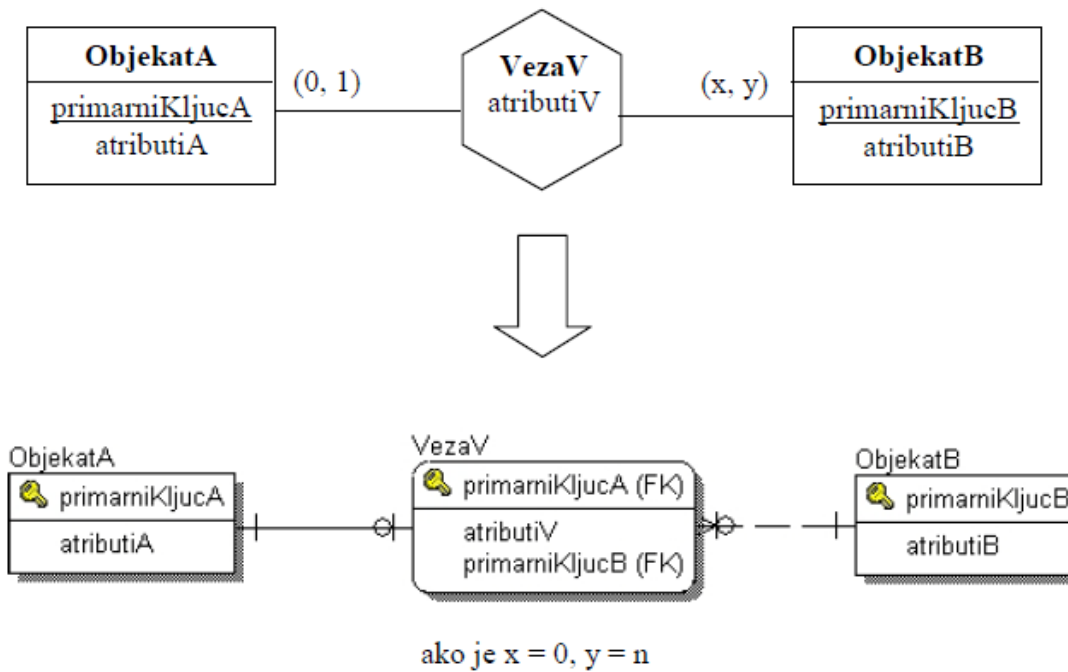
Pravila preslikavanja iz ER u IE



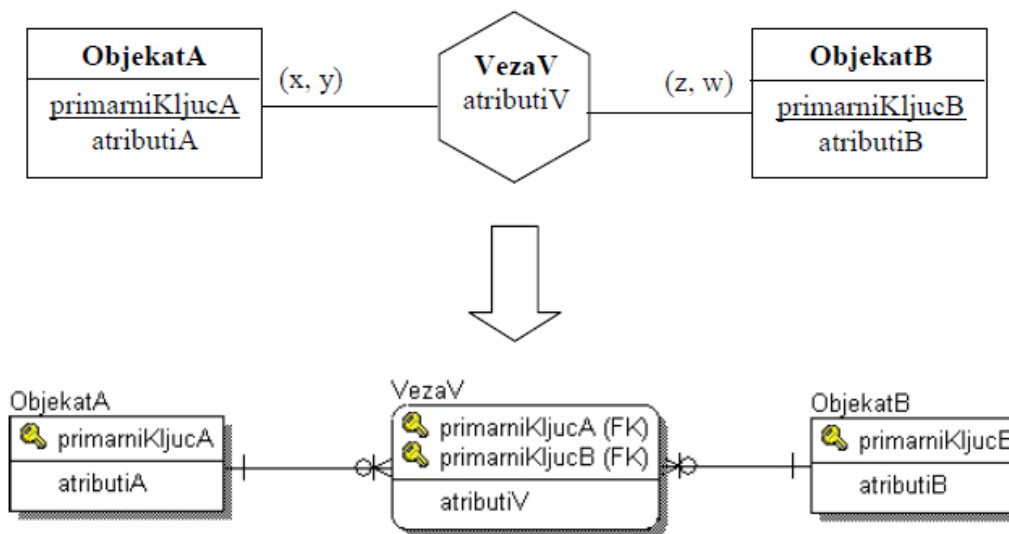
ILI



Pravila preslikavanja iz ER u IE



Pravila preslikavanja iz ER u IE



ako je $x = 0, y = n, z = 0, w = n$;
u slučaju drugih kardinalnosti,
menjaju se simboli na vezama

Constraints

Ukoliko su nam potrebna ograničenja za naše atribute u tabeli možemo ih koristiti na sledeći način.

Za atribut Status definišemo ograničenje pod nazivom „Ograničenje“ koje koristi validaciju „Pravilo“

The screenshot shows the 'Entity 'Osoba' Attribute 'Status' Editor' window. The 'Entity' is set to 'Osoba'. The 'Attribute' is 'Status'. The 'Logical Data Type' is 'CHAR'. The 'Primary Key' checkbox is checked. The 'Foreign Key' and 'Logical Only' checkboxes are unchecked.

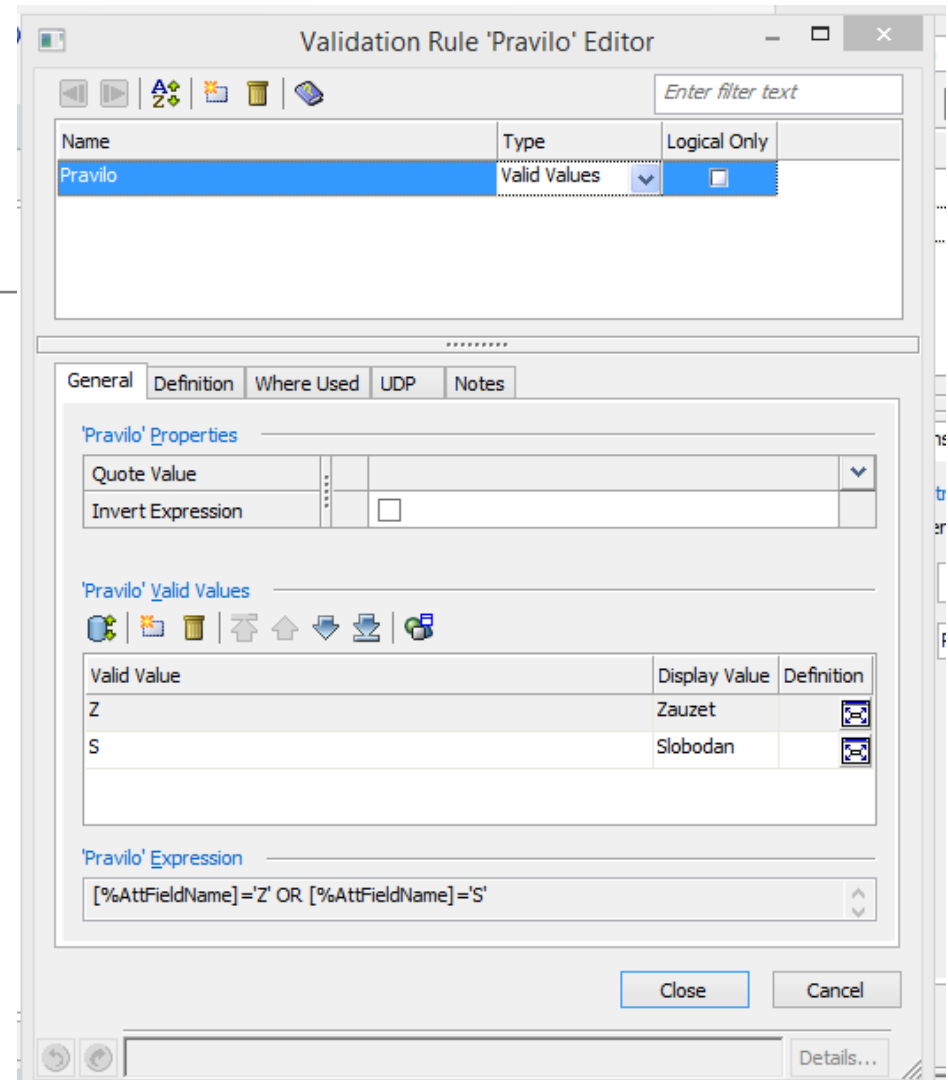
Name	Parent Domain	Logical Data Type	Primary Key	Foreign Key	Logical Only
SifO	Num...	INTEGER	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Status	String	CHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

The 'Constraint' tab is selected, and the 'Check Constraints' section is highlighted with a red box. The 'Use Inherited Constraint' checkbox is unchecked. The 'Name' field contains 'Ogranicenje' and the 'Validation' dropdown is set to 'Pravilo'.

Other tabs visible include: General, Link, Key Groups, Style, Definition, Where Used, UDP, History.

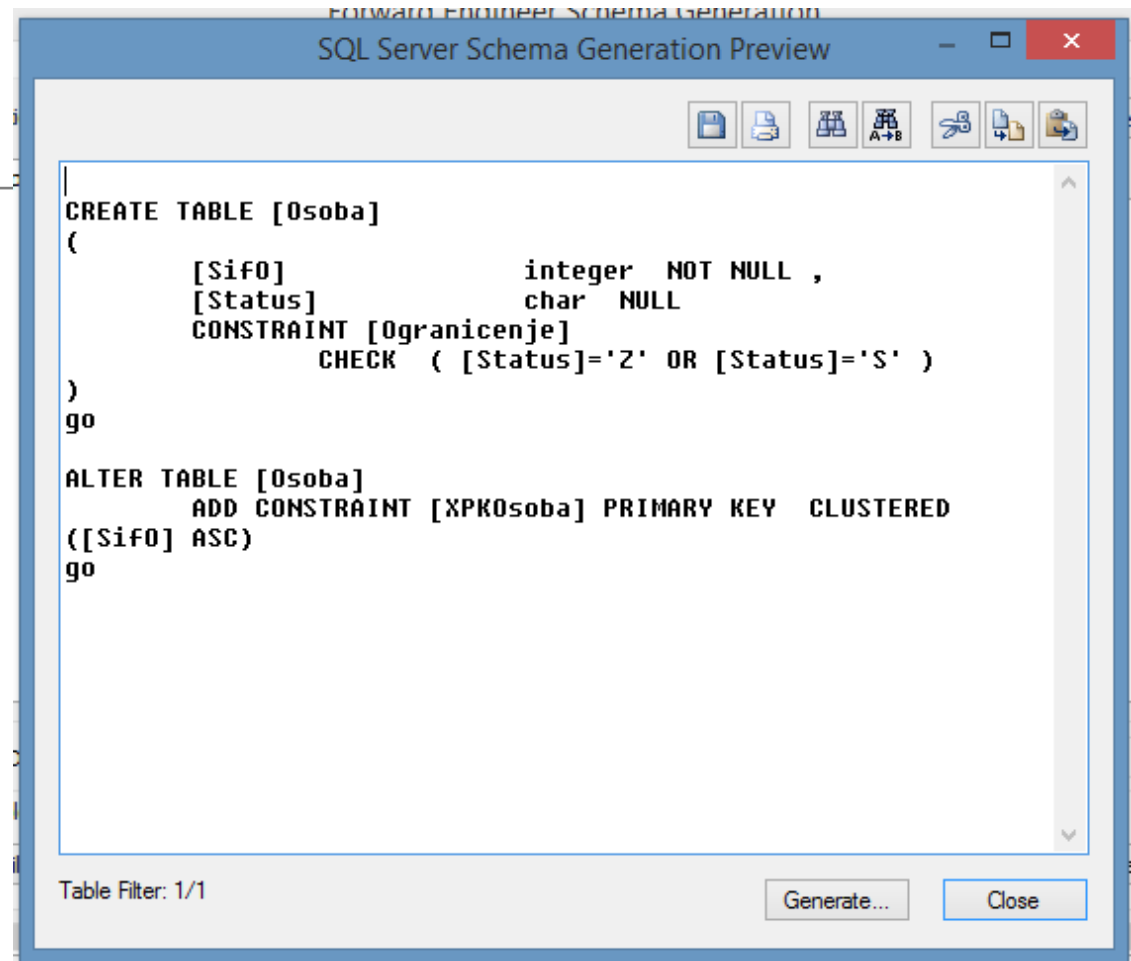
Constraints

Validacija je takva da dozvoljava samo dve određene vrednosti za atribut na koje je primenjeno i to „Z“ za „Zauzet“ i „S“ za „Slobodan“.



Constraints

U SQL skriptu je generisano ograničenje za atribut Status tabele Osoba.



The screenshot shows a window titled "SQL Server Schema Generation Preview" with a toolbar at the top containing icons for save, print, refresh, and other actions. The main area contains the following SQL code:

```
CREATE TABLE [Osoba]
(
    [Sif0]                integer NOT NULL ,
    [Status]              char  NULL
    CONSTRAINT [Ogranicenje]
        CHECK ( [Status]='Z' OR [Status]='S' )
)
go

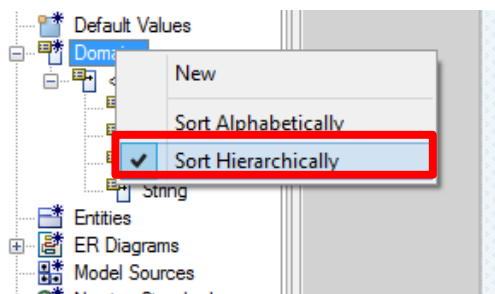
ALTER TABLE [Osoba]
    ADD CONSTRAINT [XPKOsoba] PRIMARY KEY CLUSTERED
([Sif0] ASC)
go
```

At the bottom of the window, there is a "Table Filter: 1/1" label and two buttons: "Generate..." and "Close".

Domeni

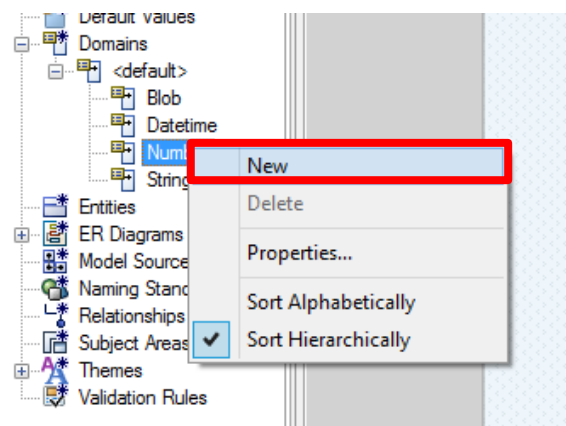
Olakšavaju stvaranje atributa. Različiti entiteti mogu imati isti naziv atributa (npr. Naziv, Adresa, Broj ...) pa iz tog razloga stvaramo domene koji će zajedno predstavljati skup različitih naziva atributa. Za svaki domen se definiše određeni tip podatka tako da firma ima mogućnost da tačno definiše tipove svojih atributa i da ih članovima firme na korišćenje.

Domeni

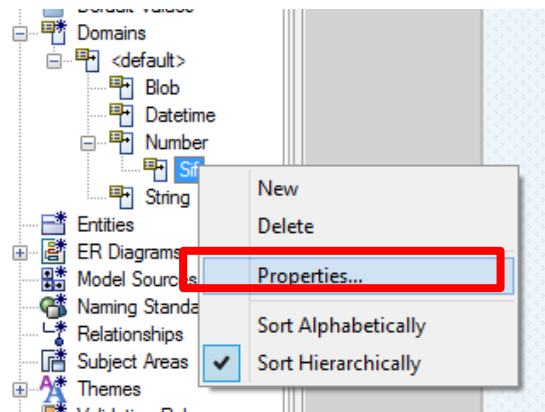


Radi boljše preglednosti izaberemo hierarhijski prikaz.

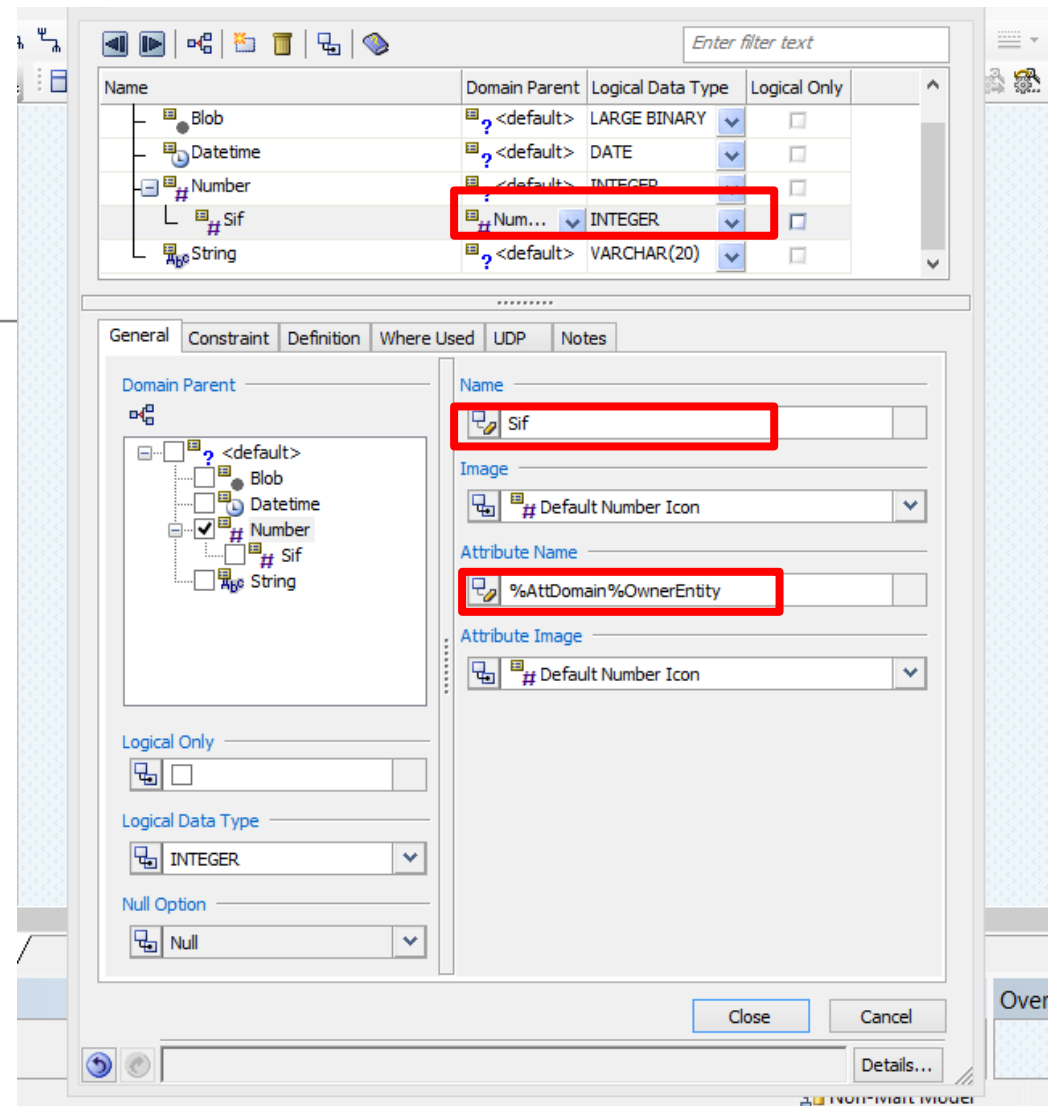
Kreiramo novi domen.



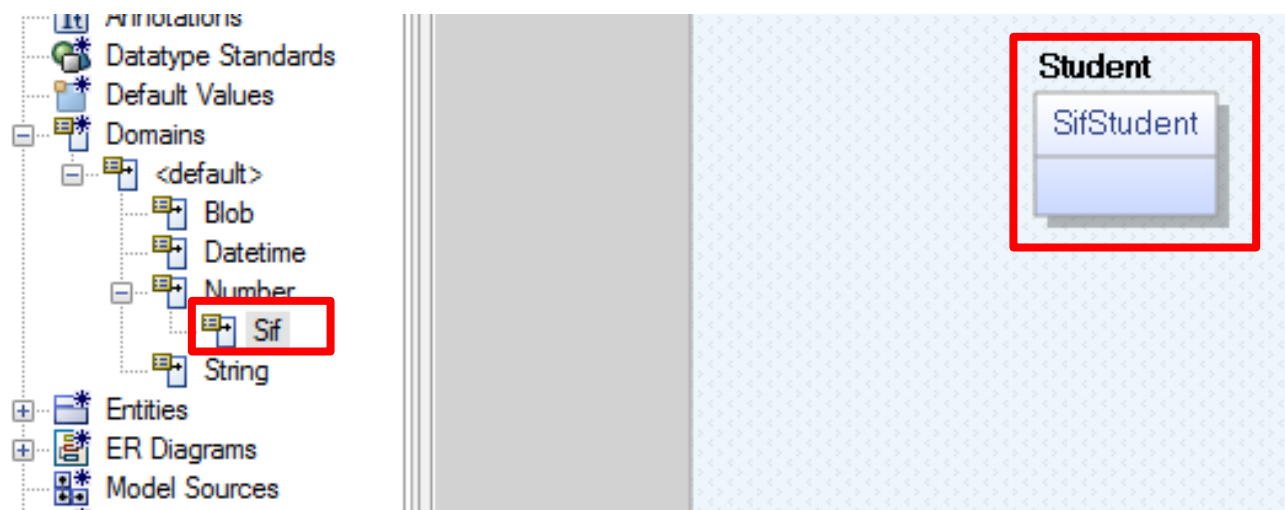
Domeni



Svaki domen ima svoj tip.
Za naziv se mogu koristiti makroi
kao što je %OwnerEntity.

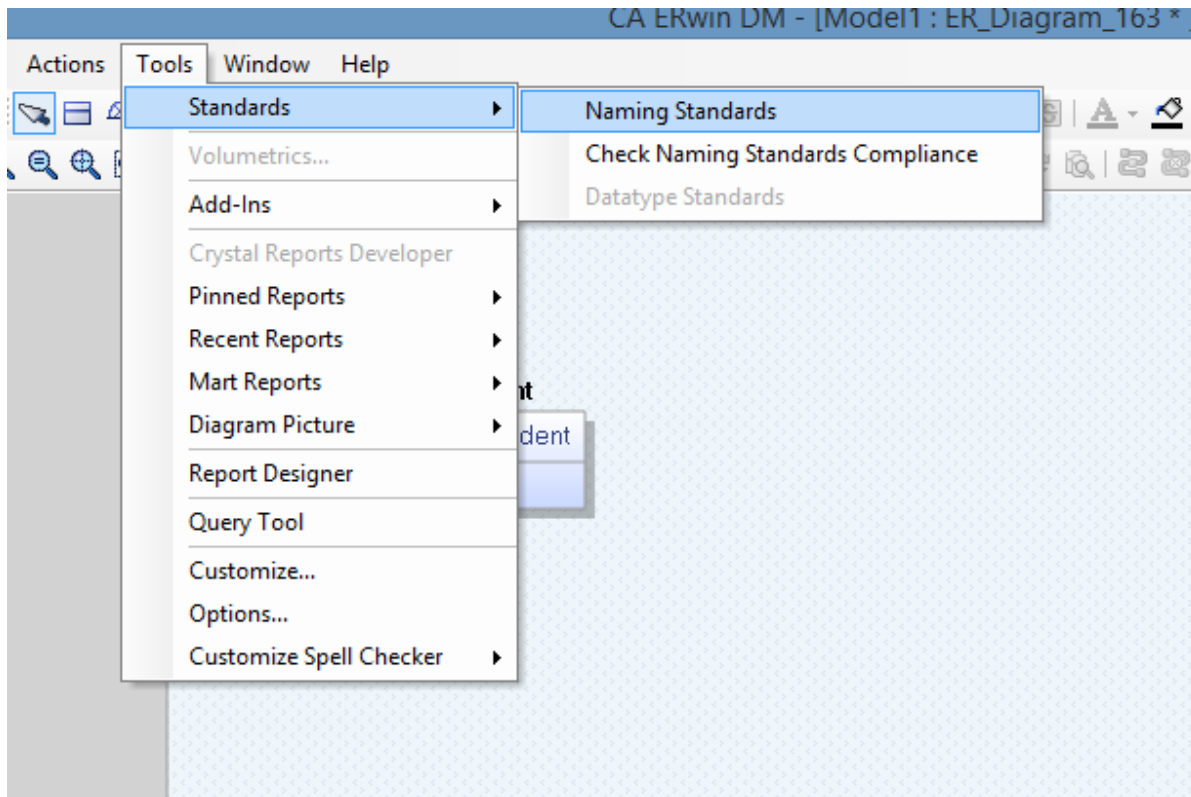


Domeni



Domen „Sif“ zbog makroa kao atribut ima naziv „SifStudent“.

Naming Standards



Npravimo novi standard za imenovanje.

Naming Standards

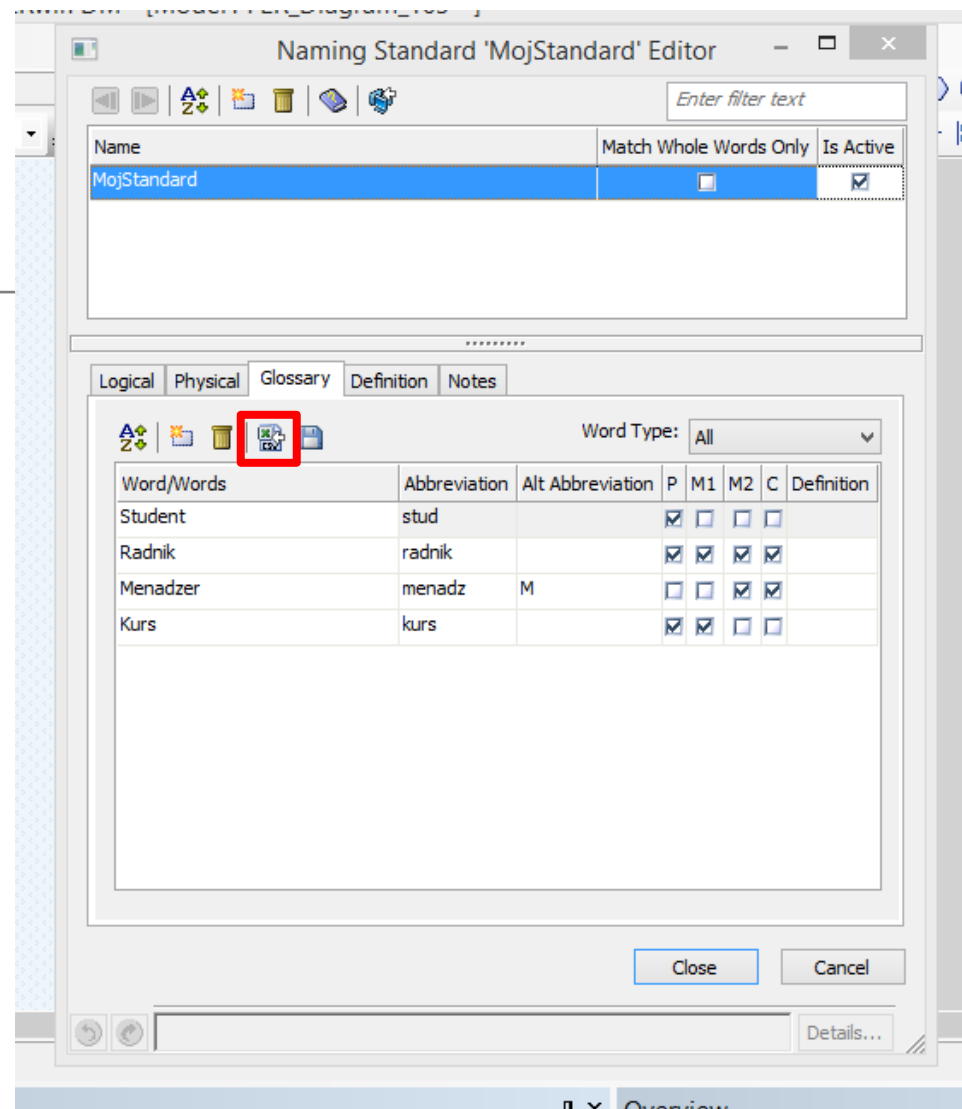
The screenshot shows the Microsoft Excel interface with the following data in the spreadsheet:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Student	stud		X									
2	Kurs	kurs		X	X								
3	Radnik	radnik		X	X	X	X						
4	Menadzer	menadz	M			X	X						
5													
6													
7													
8													
9													
10													
11													
12													
13													

Standarde obično definišemo u CSV fajlu.

Naming Standards

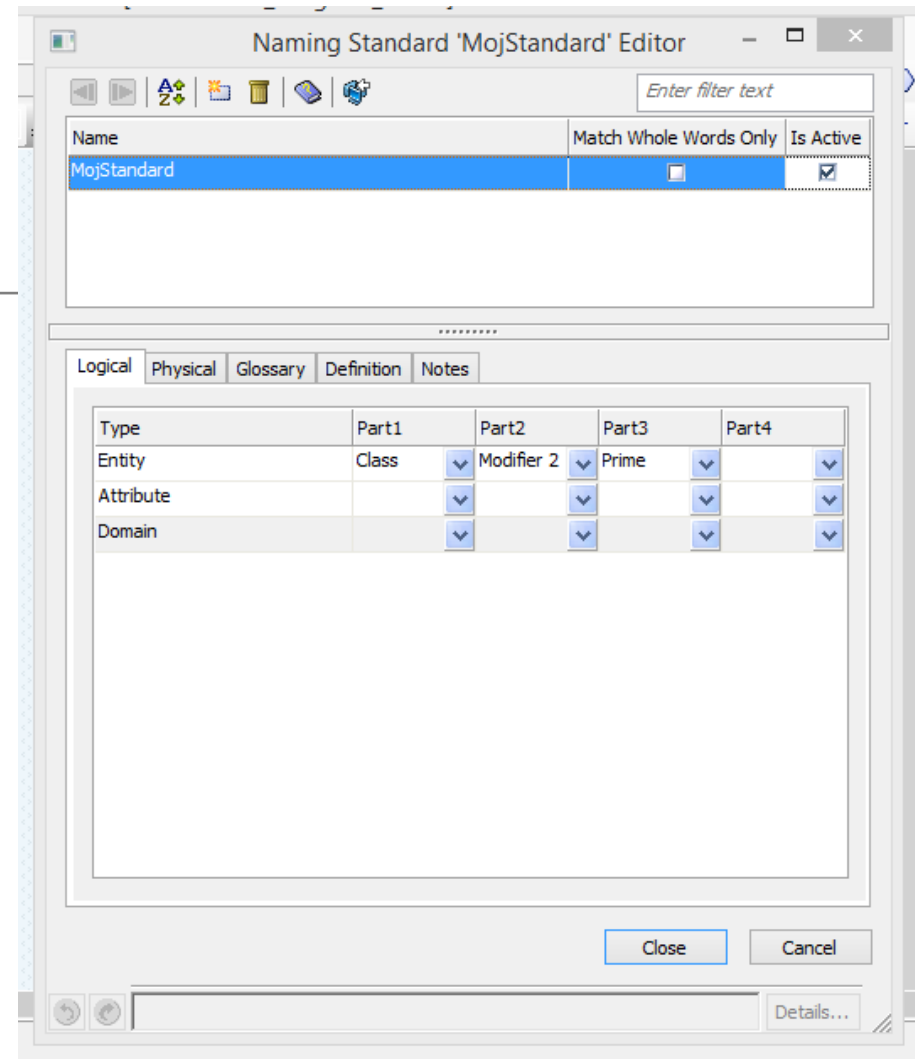
Importujemo naš CSV fajl i pravila za imenovanja se pojave u listi.



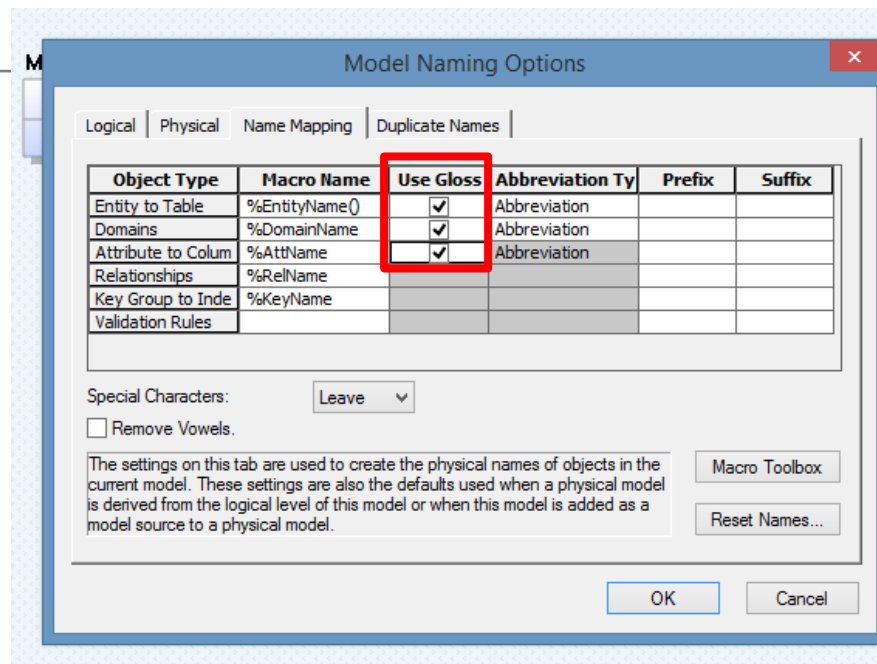
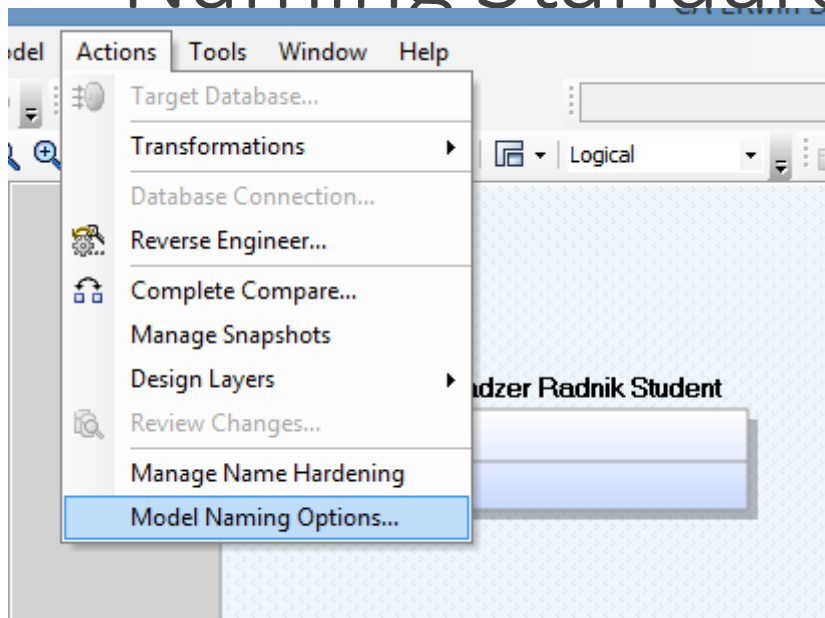
Naming Standards

Definišemo pravila za imenovanje entiteta, atributa i domena. U ovom primeru Entitet se sastoji od tri reci od kojih je prva tipa „Class“, druga tipa „Modifier 2“ i treća tipa „Prime“

Pravilo obavezno označiti kao aktivno.

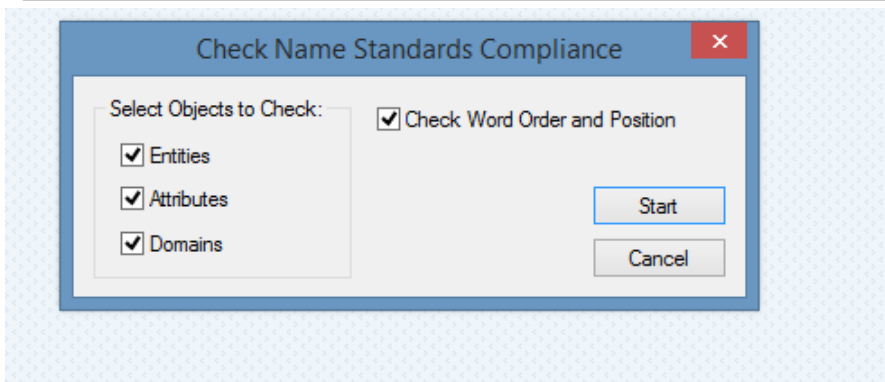


Naming Standards



Da bi naša pravila za konverziju naziva iz logičkog u fizički model bila korišćena, potrebno je čekirati tip objekta za koji želimo da naša pravila budu primenjena.

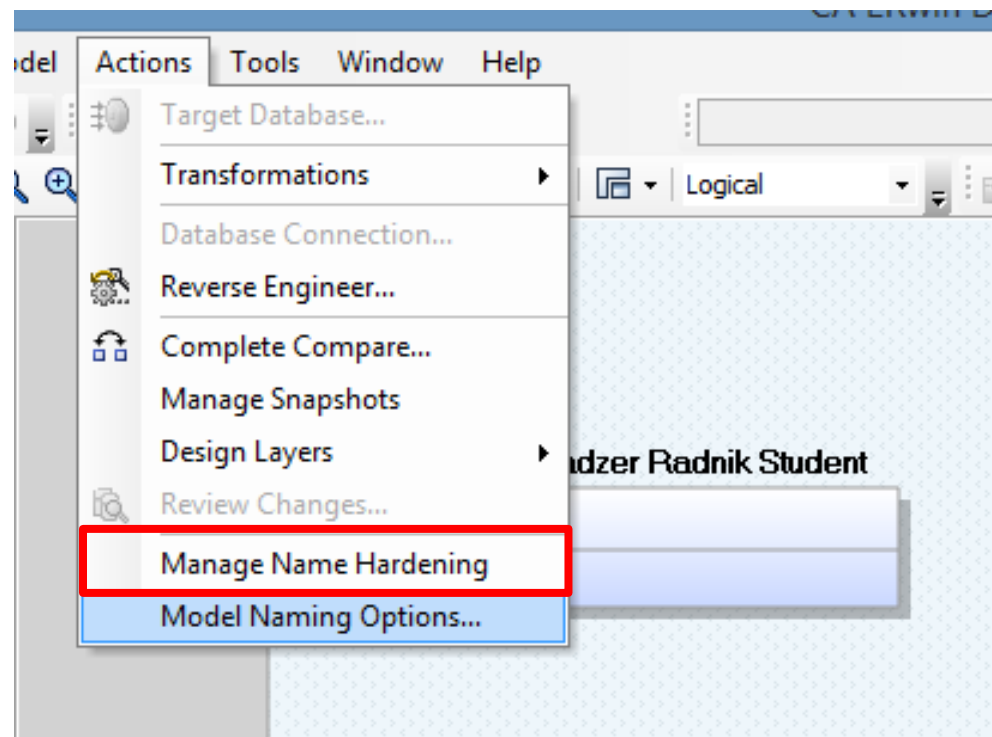
Naming Standards



Entitet „Menadzer Radnik Student“ je uspeo da prođe proveru, jer je nazvan po definisanom pravilu.

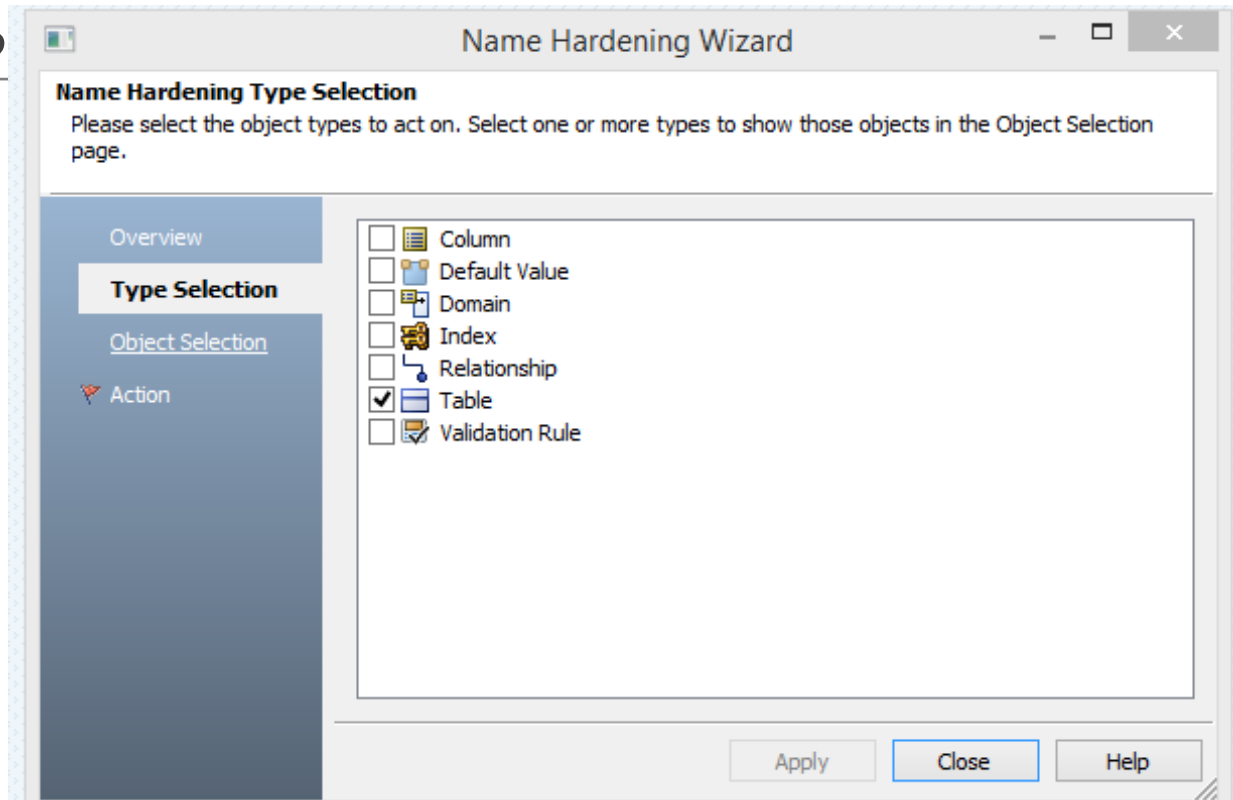
Name Hardening

Ukoliko želimo da se neka od naših pravila imenovanja ne primenjuju u prevođenju naziva iz logičkog modela u fizički možemo koristiti opciju „Name Hardening“.



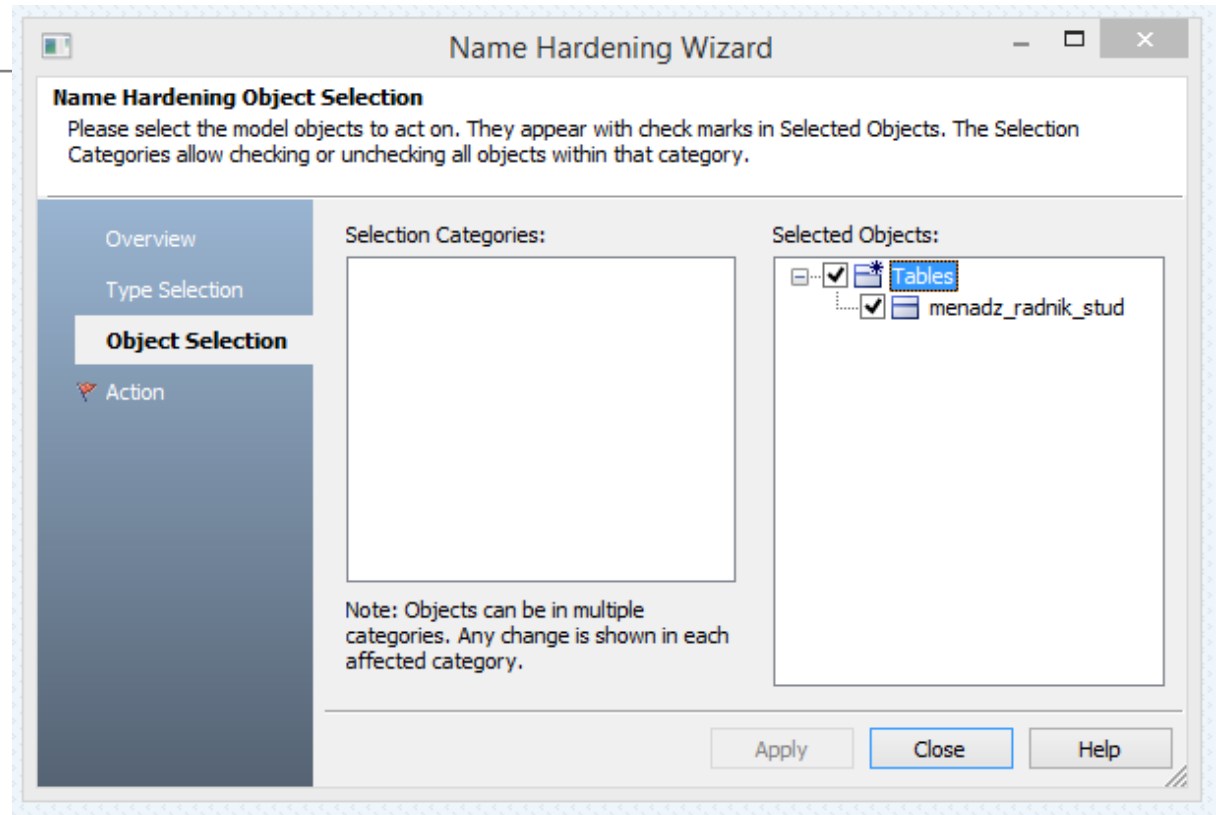
Name Hardening

Izaberemo tip koji želimo da zaključamo.



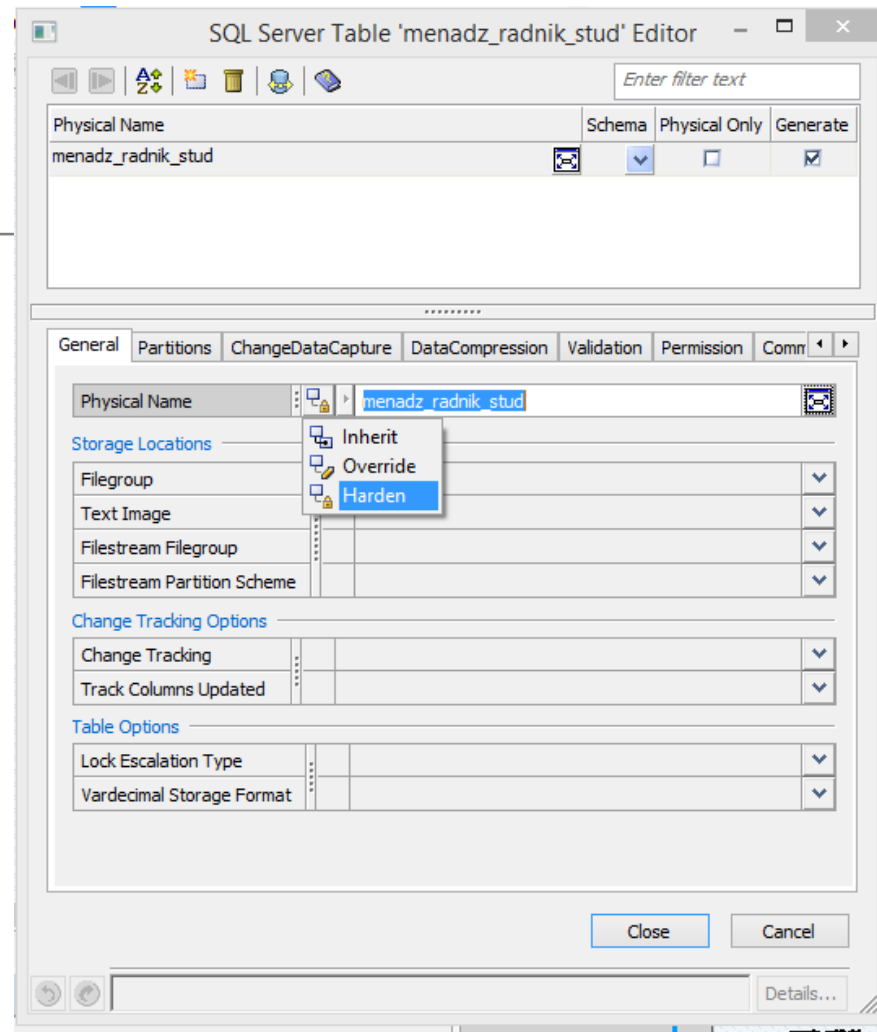
Name Hardening

Zatim i objekat koji zaključavamo.



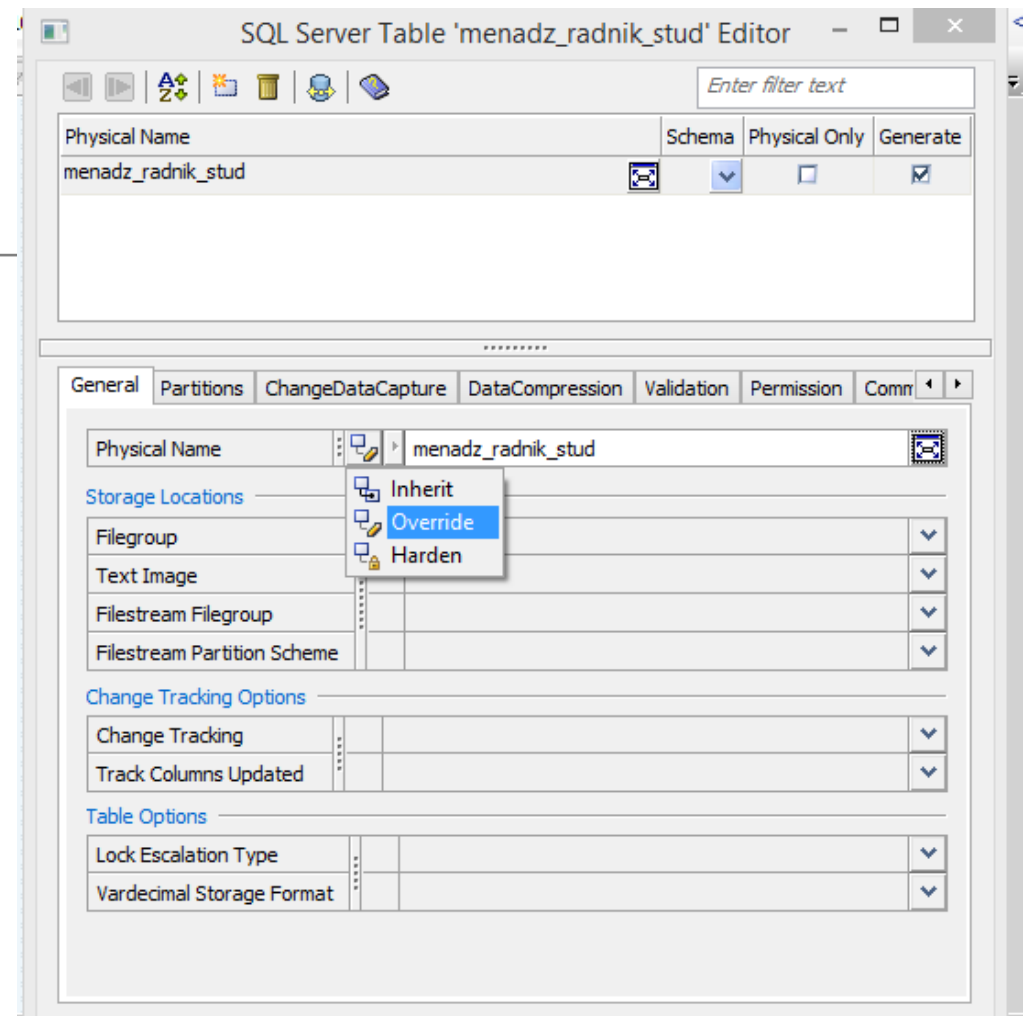
Name Hardening

Kao što je prikazano, ime našeg entiteta je zaključano (Harden – iskaljeno).



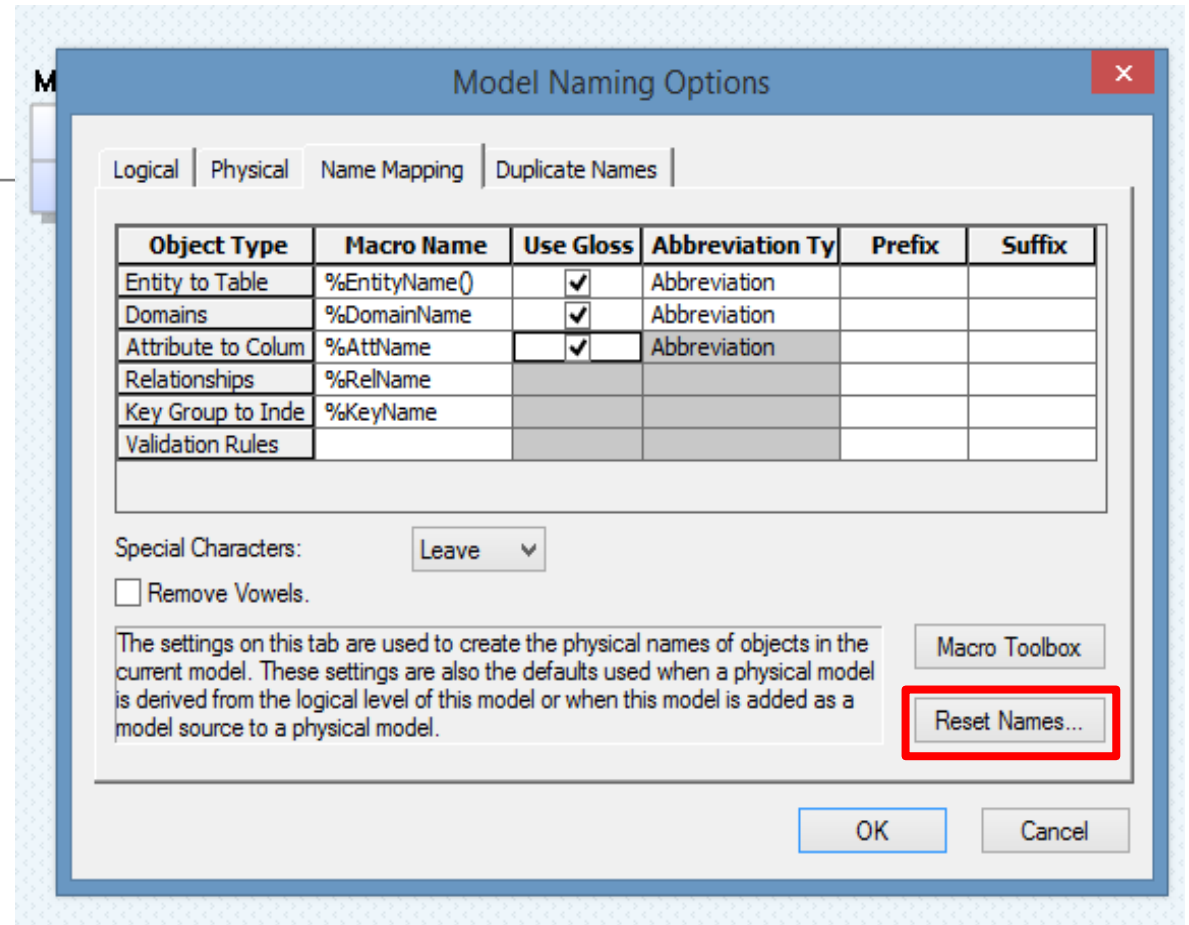
Name Hardening

Problem je što prilikom otključavanja ime postane „Override“ a ne „Inherit“ tako da se naša pravila još uvek neće primenjivati.



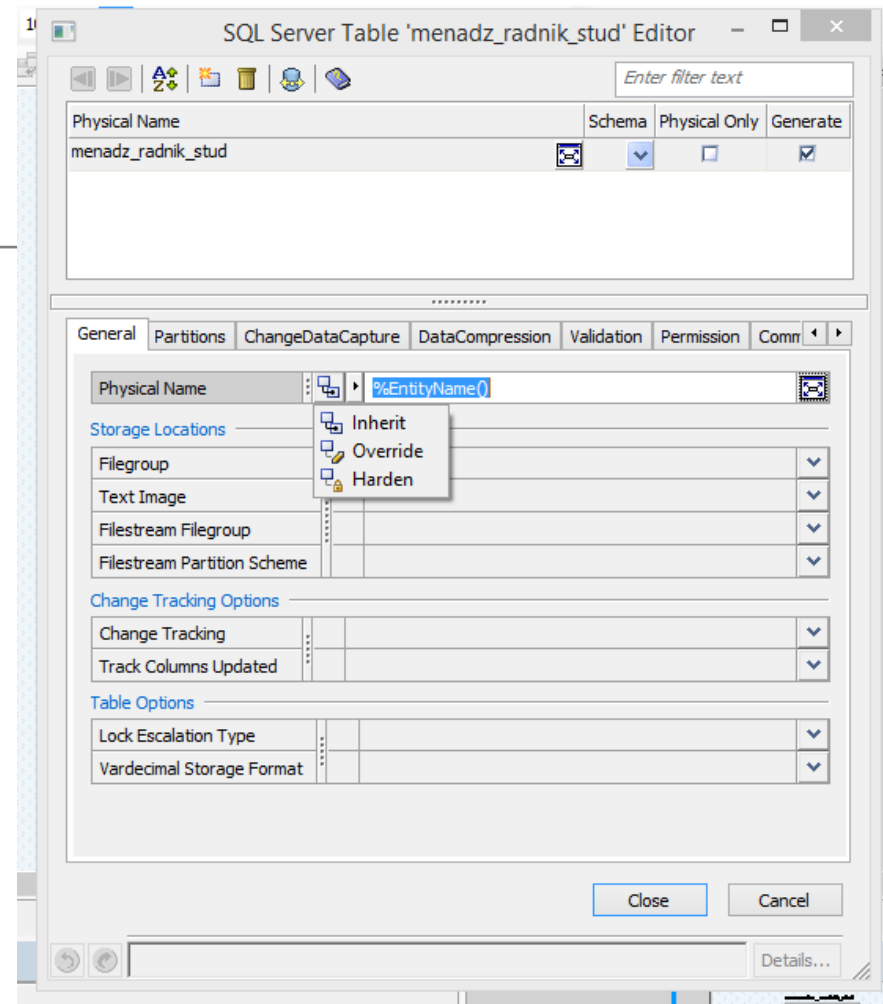
Name Hardening

Rešenje za ovo je da resetujemo sva imena klikom na „Reset Names“.



Name Hardening

Sada je naziv naše tabele tipa „Inherit“ tako da će se pravila imenovanja primenjivati.



Forward Engineering

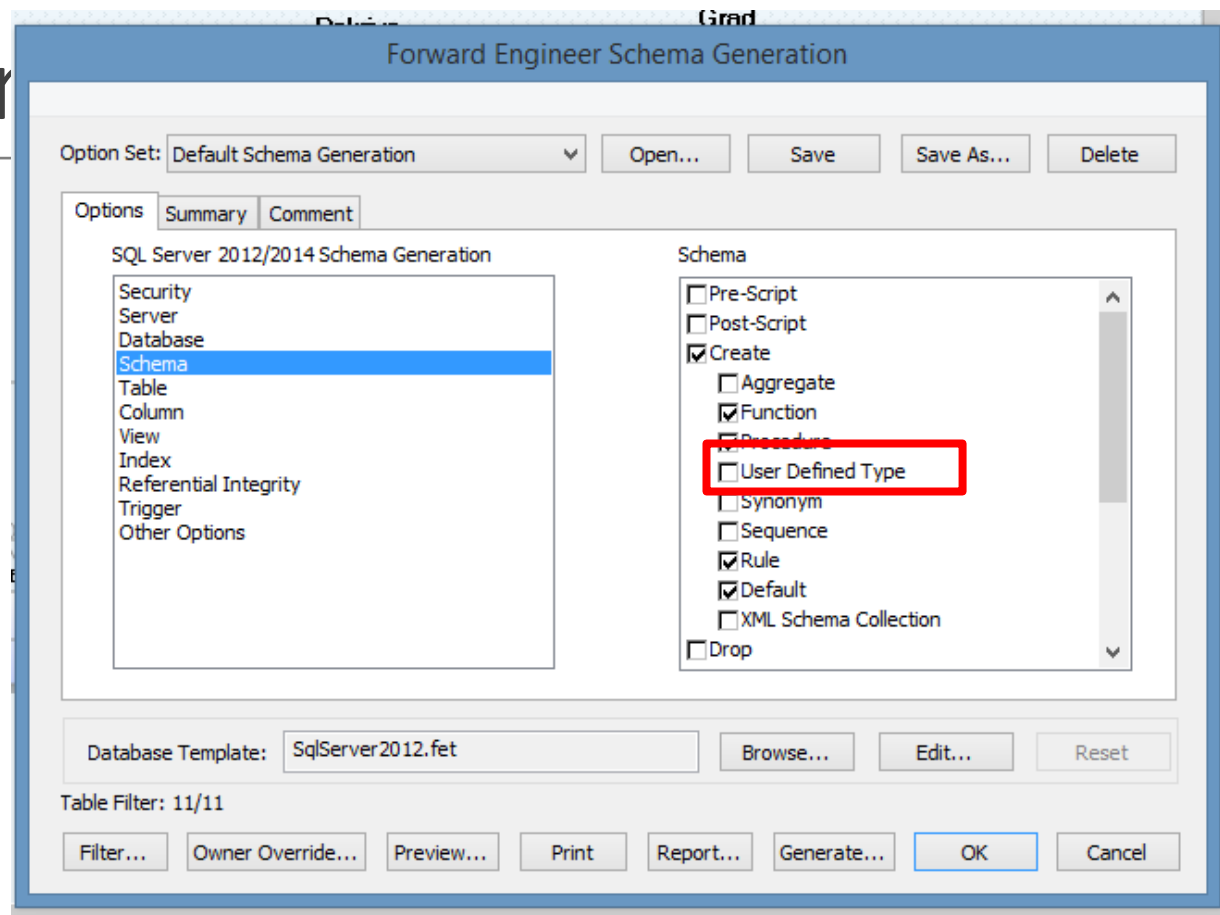
Forward engineering nam omogućava da iz modela napravimo bazu za različite ciljne servere.



Klikom na forward engineering otvoriće se novi prozor.

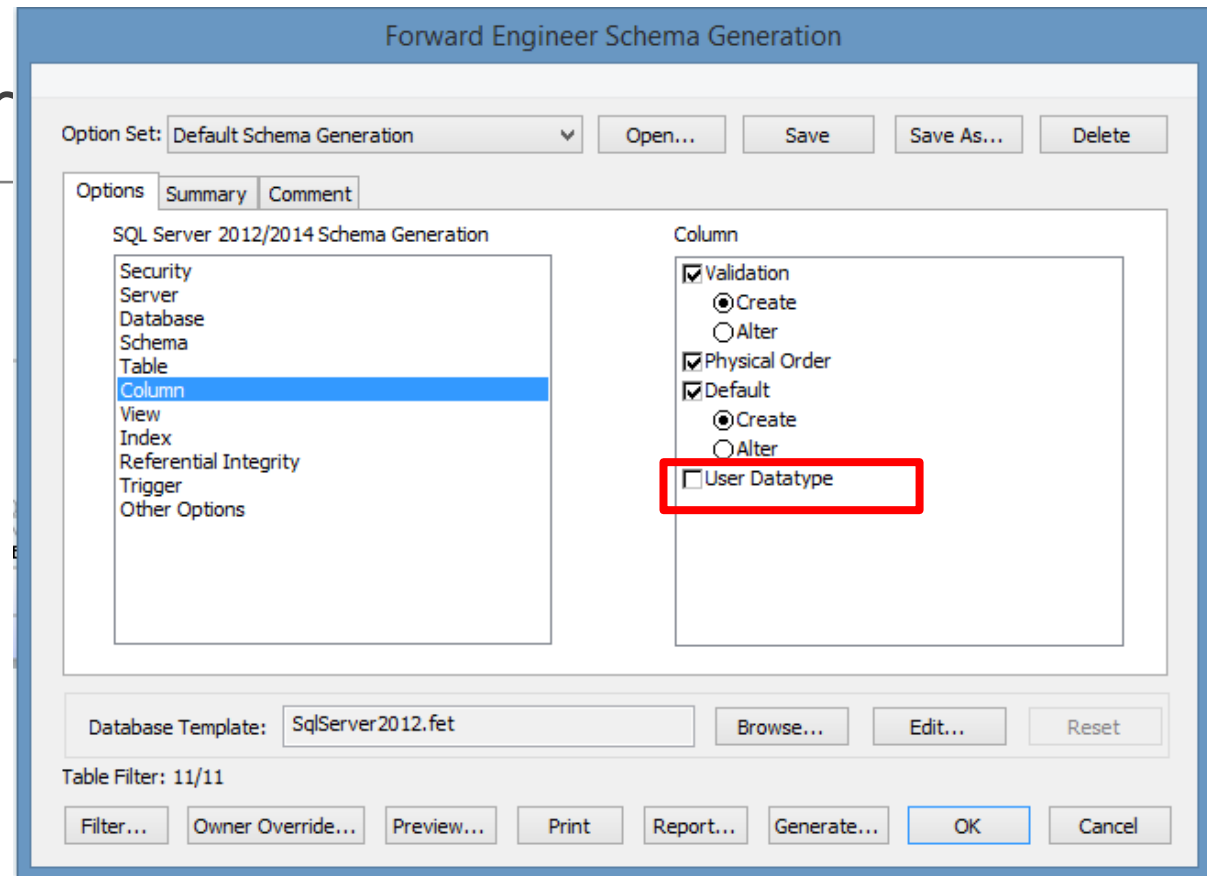
Forward Engineer

Isključimo „User Defined Type“ ukoliko ne želimo da generišemo tipove u SQL skriptu koji će oslikati domene napravljene u ERWin alatu.



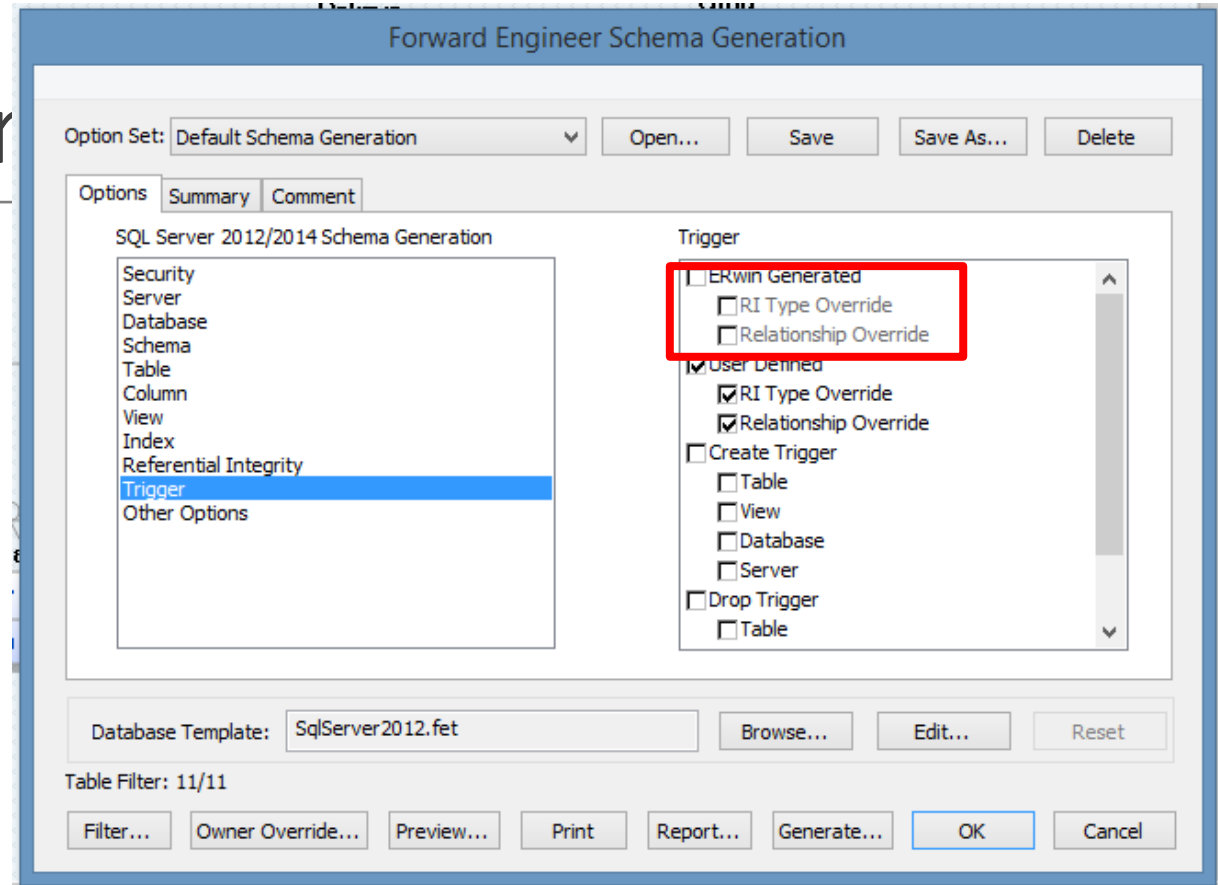
Forward Engineer

Isključimo „User Datatype“ ukoliko ne želimo da naši atributi budu tipa nekog postojećeg domena već da budu onog tipa koji je definisan za odgovarajući domen.



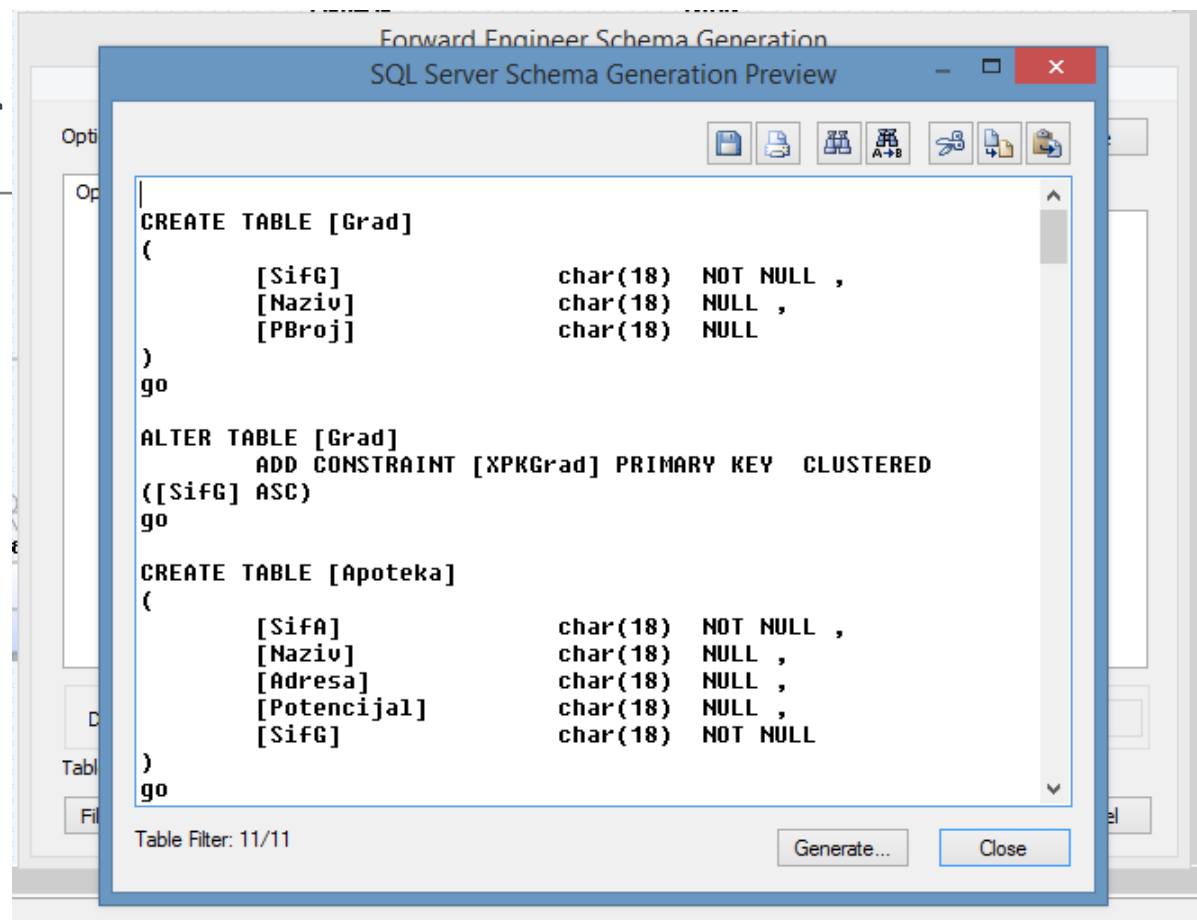
Forward Engineer

Isključimo „ERWin Generated“ trigere kako u SQL skriptu ne bismo imali automatski generisane trigere od strane alata.

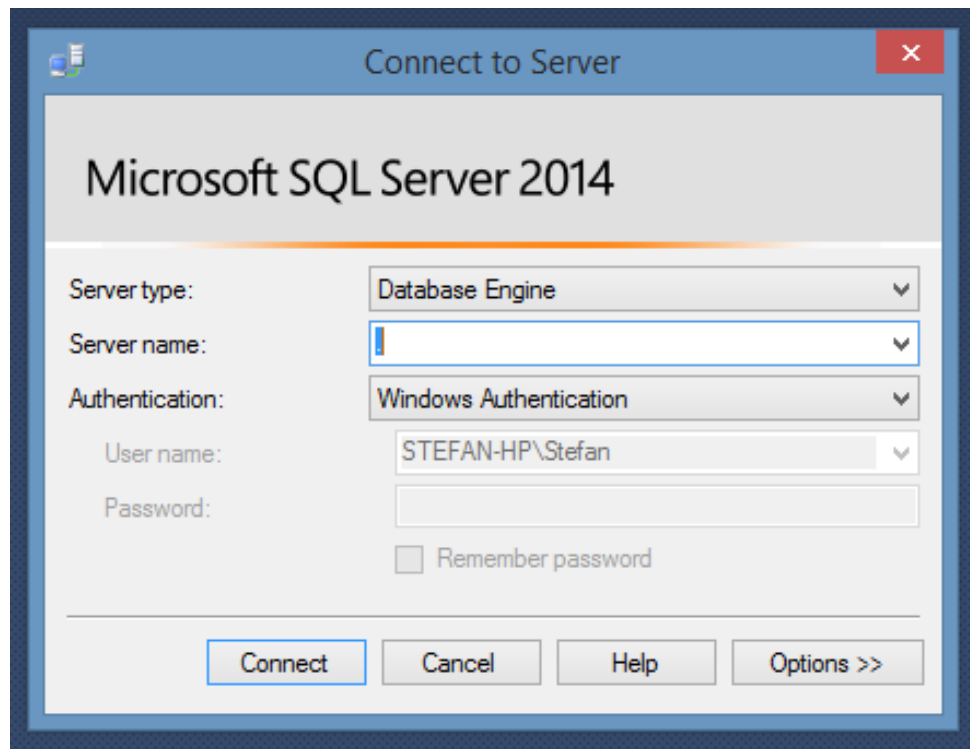


Forward Engineer

Klikom na „Preview“ otvara se prozor sa SQL skriptom koja služi za kreiranje baze.



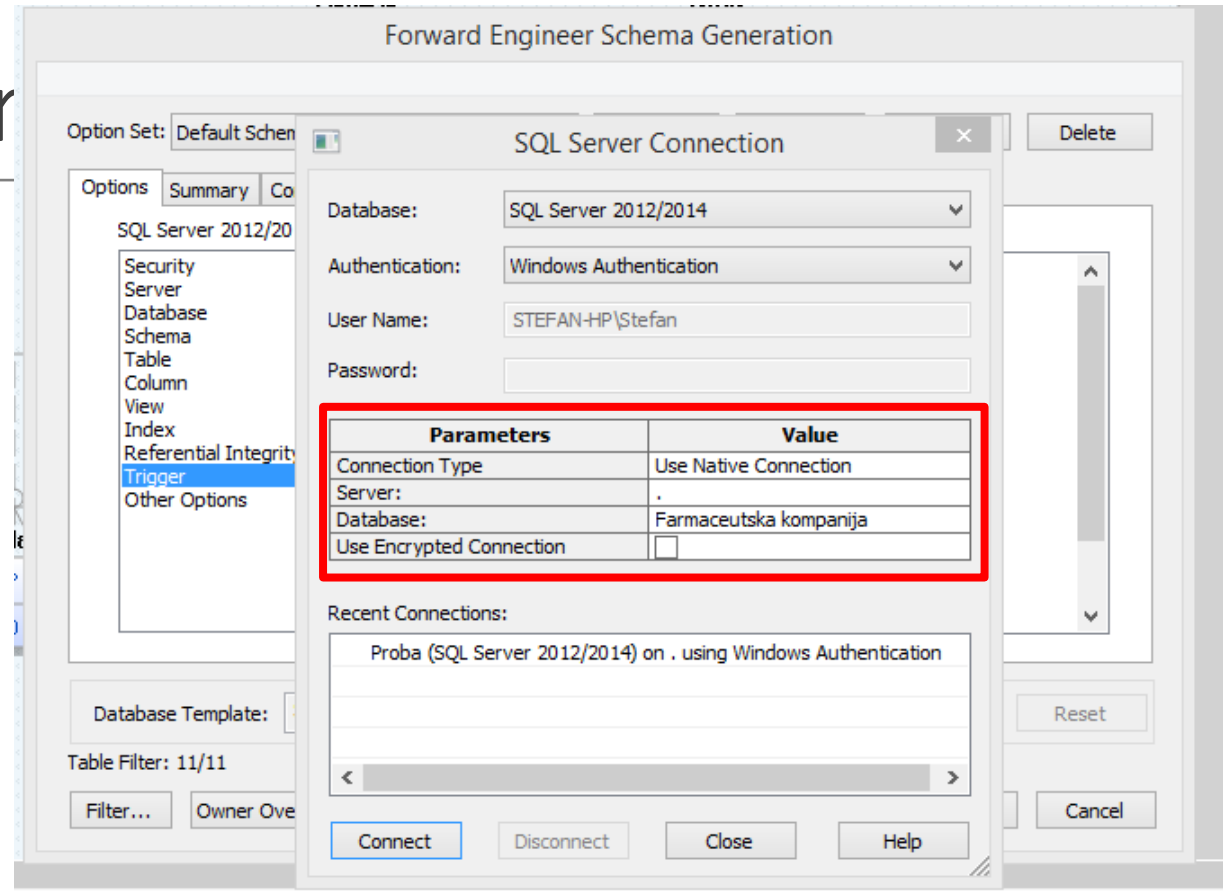
Forward Engineering



Po otvaranju Microsoft SQL Server Management Studio alata biramo server na koji želimo da se povežemo. Server name „.“ označava lokani server.

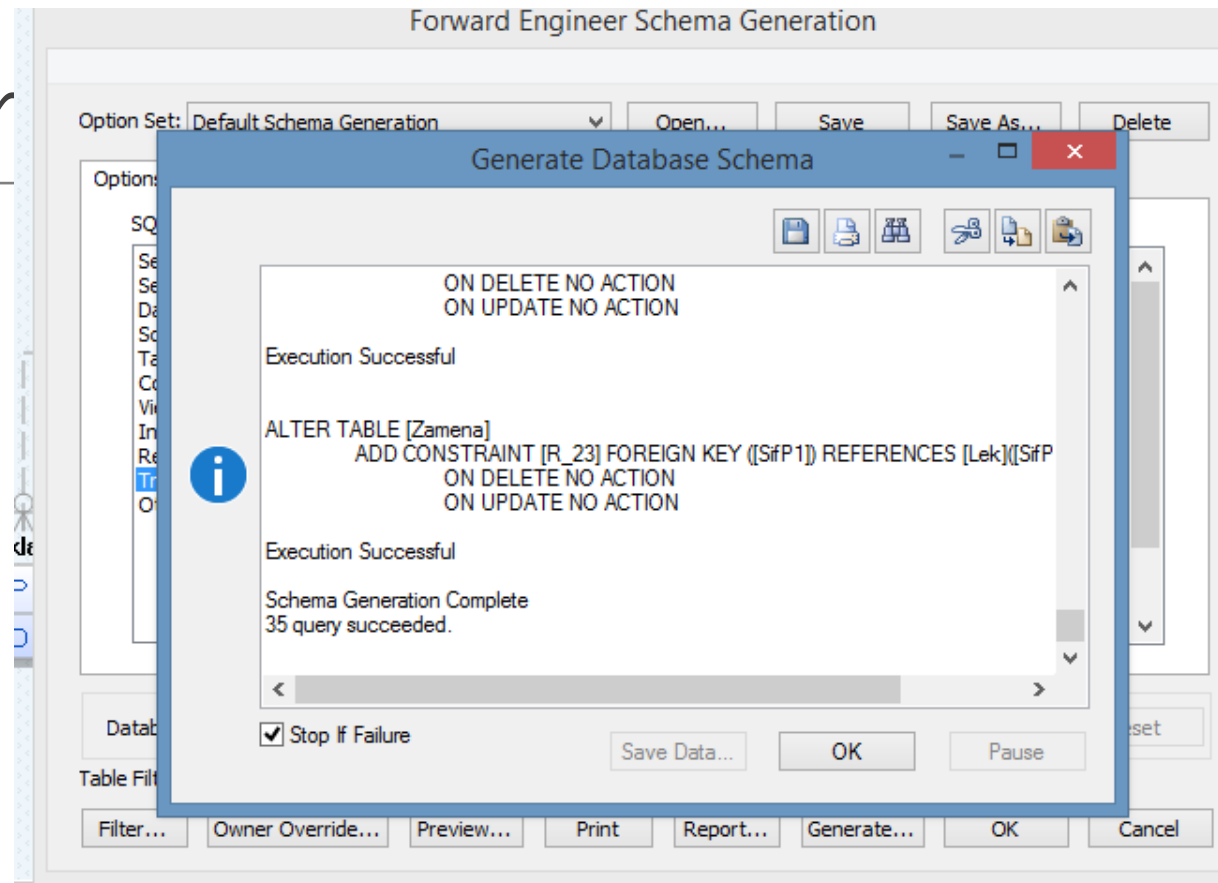
Forward Engineer

U ERWin alatu se takođe povežemo na isti server i to na bazu „Farmaceutska kompanija“ jer za nju želimo da generišemo tabele iz našeg modela.



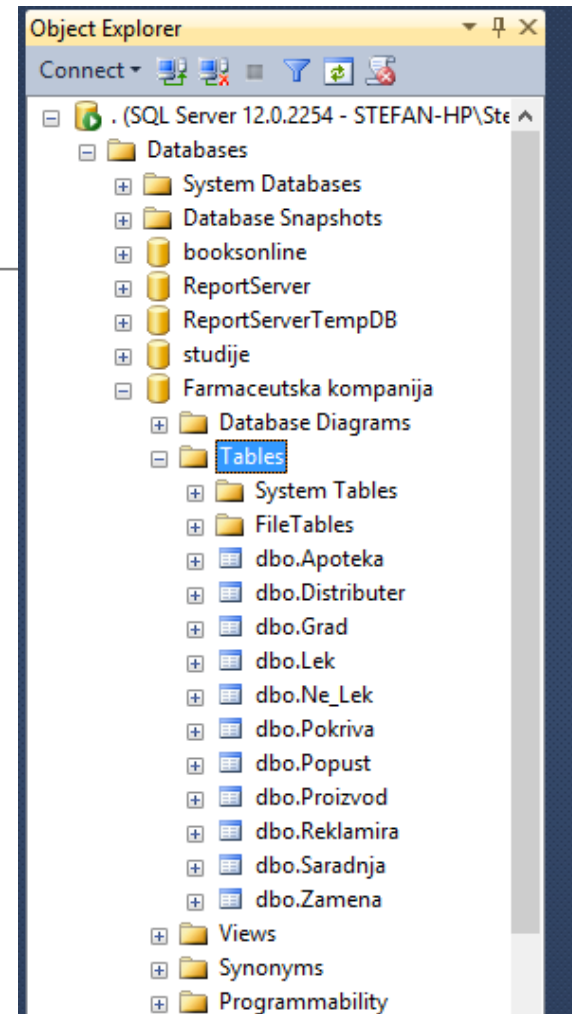
Forward Engineer

Klikom na „Generate“ alat ispisuje da je izvršavanje uspešno obaljeno.



Forward Engineering

Kao rezultat ovoga u Management studiu vidimo izgenerisane tabele.



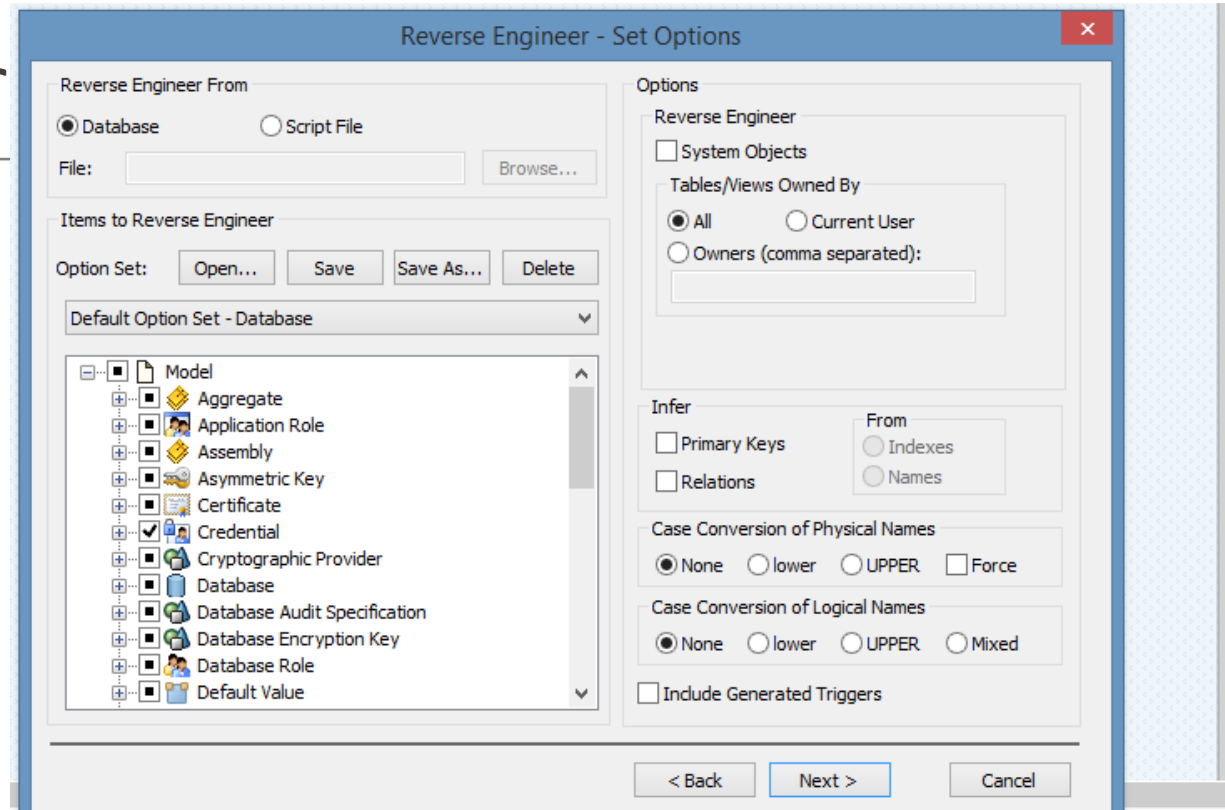
Reverse Engineering

Reverse Engineering nam omogućava da iz baze napravimo model.



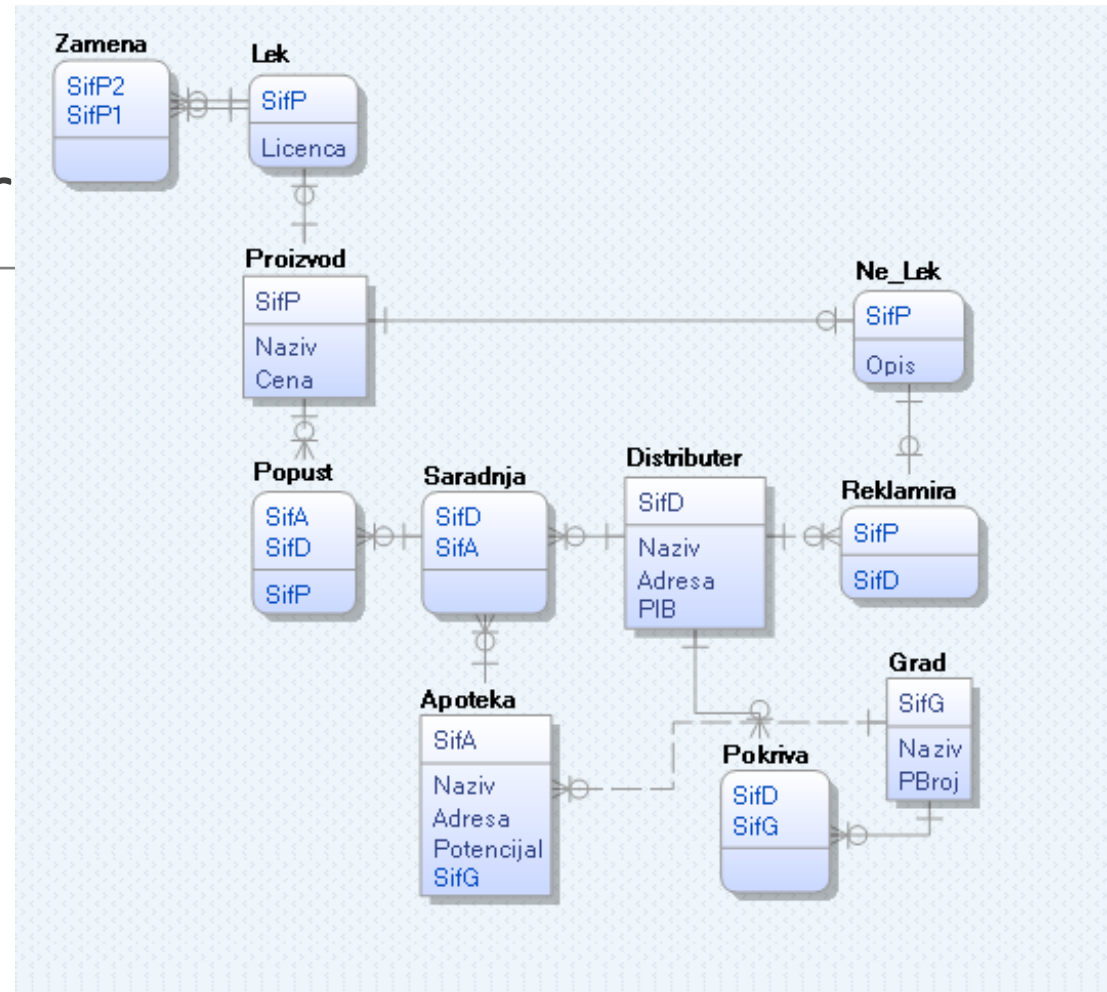
Reverse Engineer

Izaberemo šta iz baze želimo da uvezemo u naš model.



Reverse Engineer

Alat ovim postupkom generiše novi model.

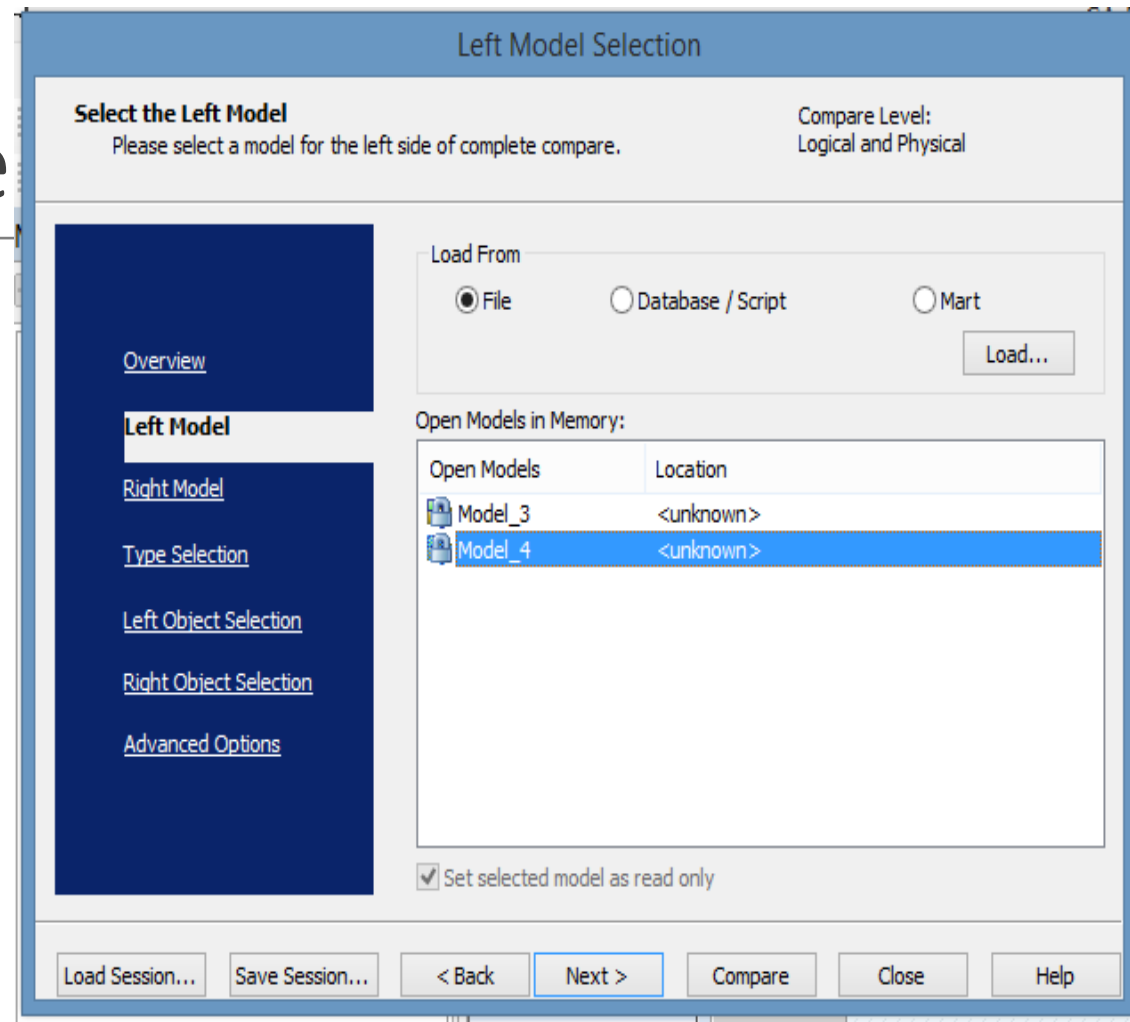


Complete Compare

Complete Compare nam omogućava da uporedimo model, bazu i skript u svim mogućim kombinacijama.

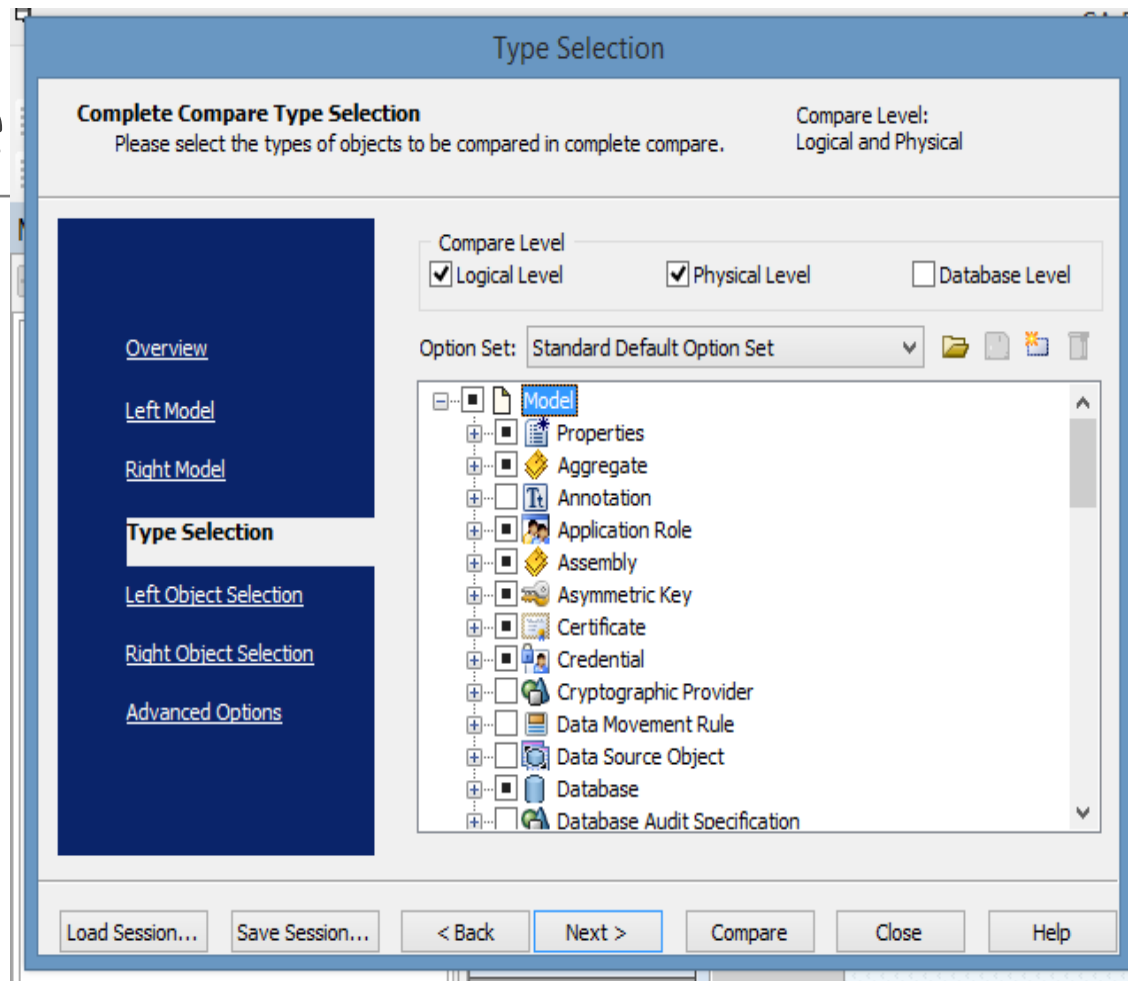
Ovde je dat primer sa uprodeđivanjem dva modela.

Biramo levi model, analogno i desni.



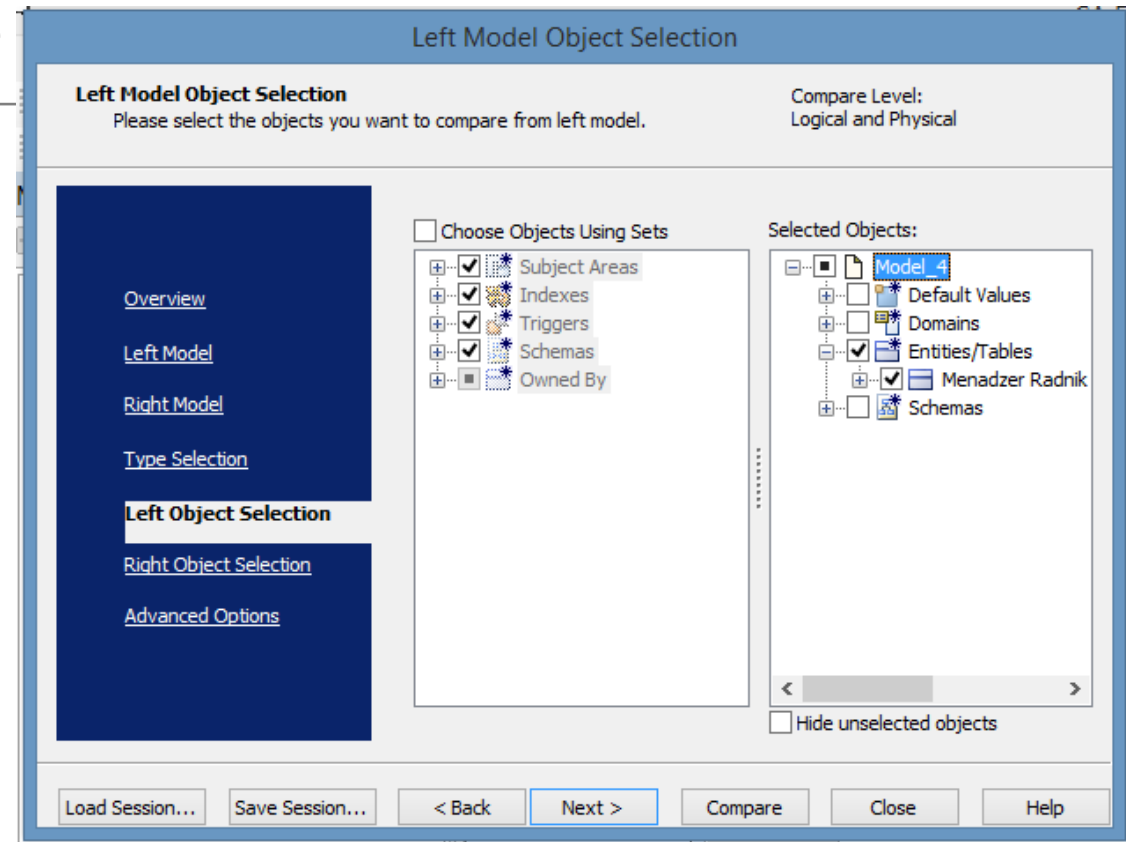
Complete Compare

Zatim tipove koje želimo da uporedimo.



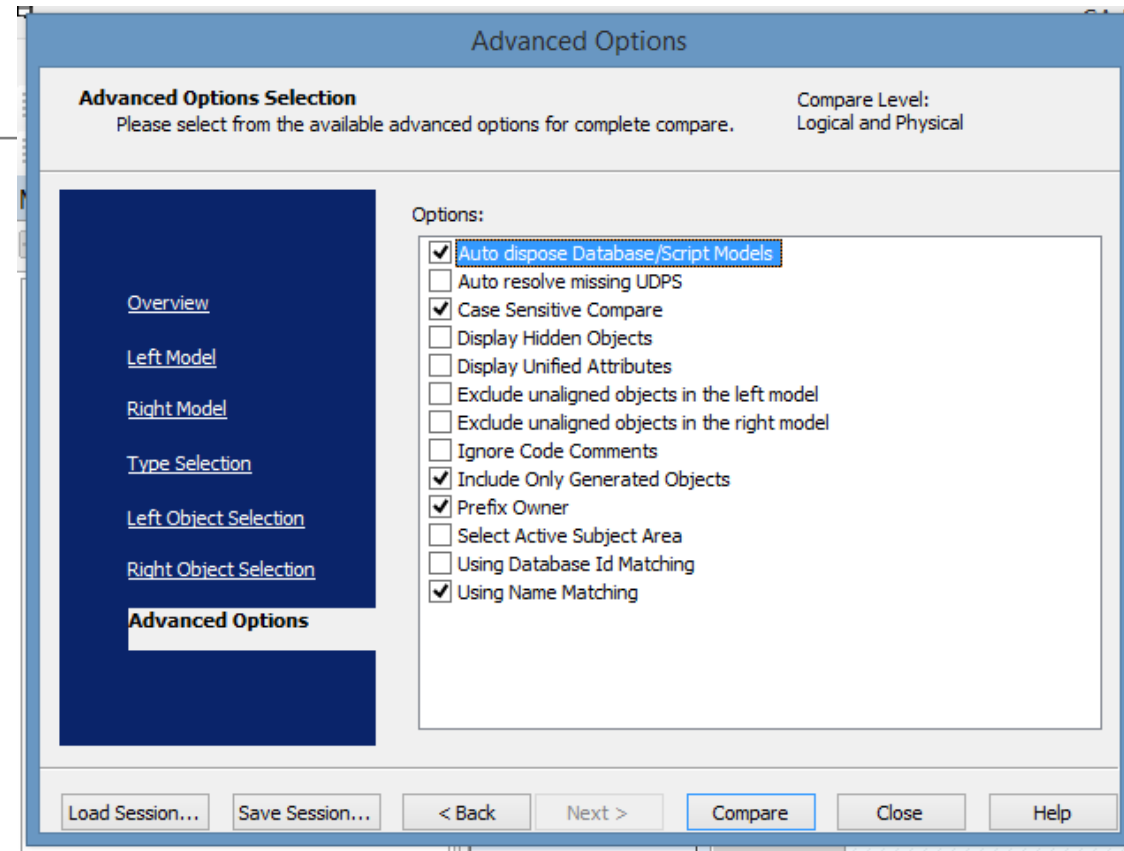
Complete Compare

Objekte iz levog i analogno iz desnog modela.



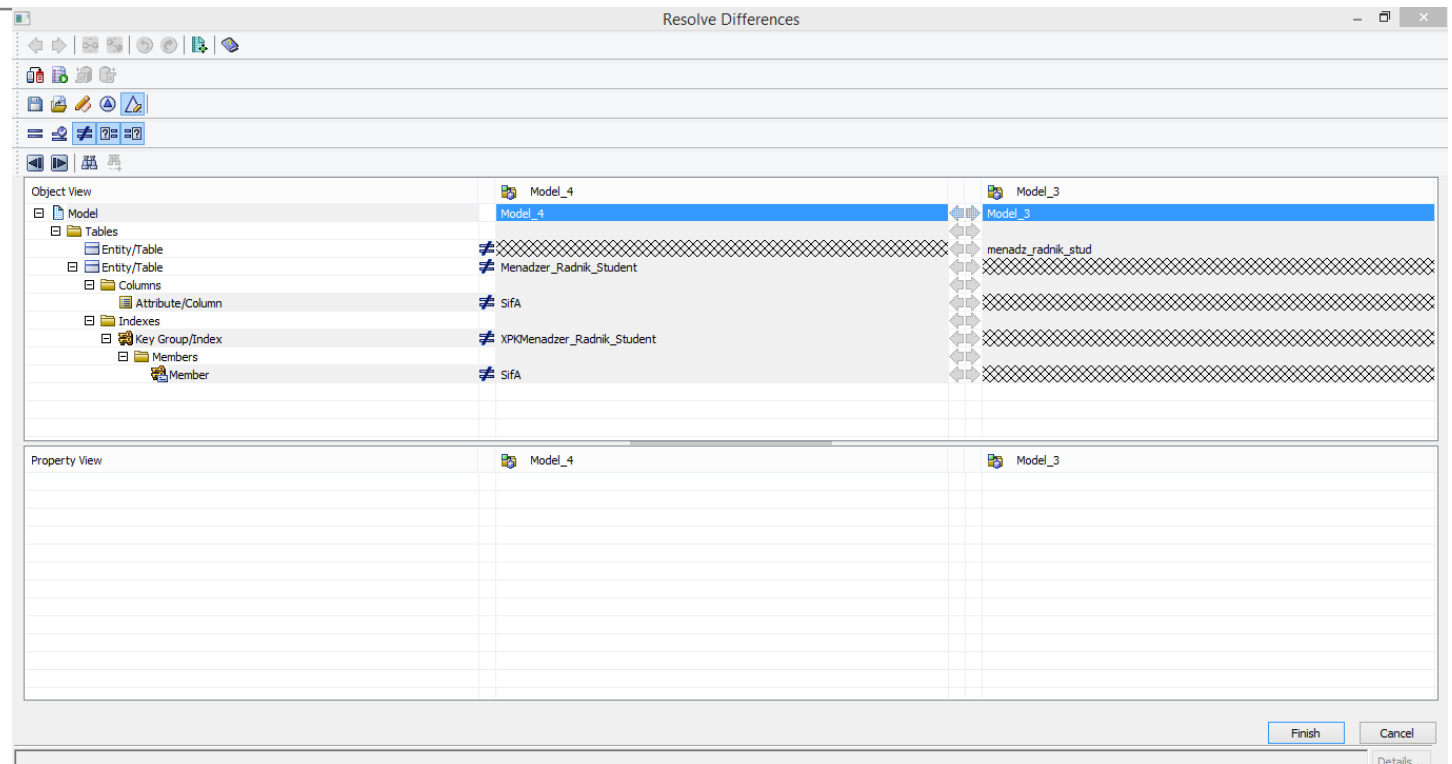
Complete Compare

Ukoliko ima potrebe podesimo napredne opcije.



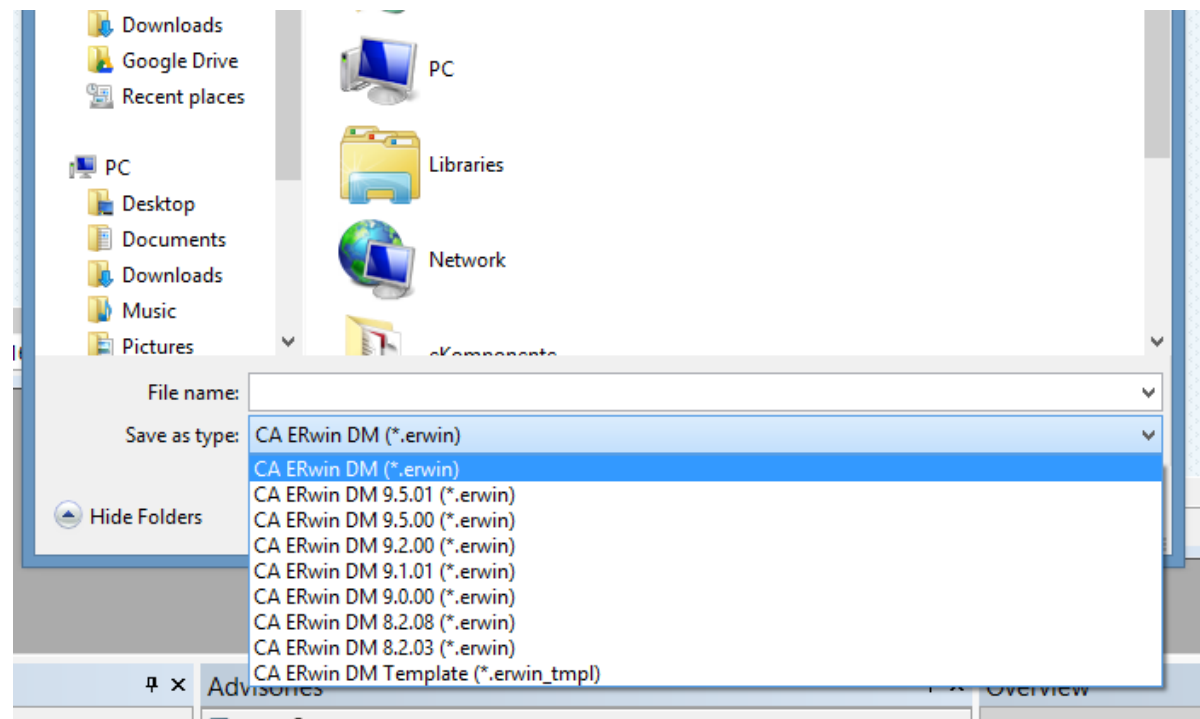
Complete Compare

Rezultat našeg upoređivanja.
Dostupne su opcije za
uređivanje ova dva modela
kako bi bili što sličniji.



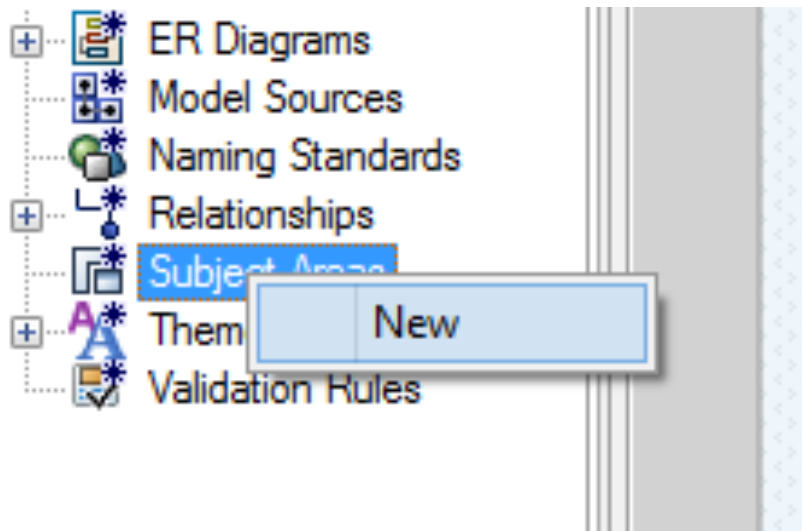
Templates

Sva podešavanja u ERWin-u se mogu sačuvati kao template koji će se kasnije koristiti u vidu fajla sa ekstenzijom .erwin_tmpl



Subject Areas

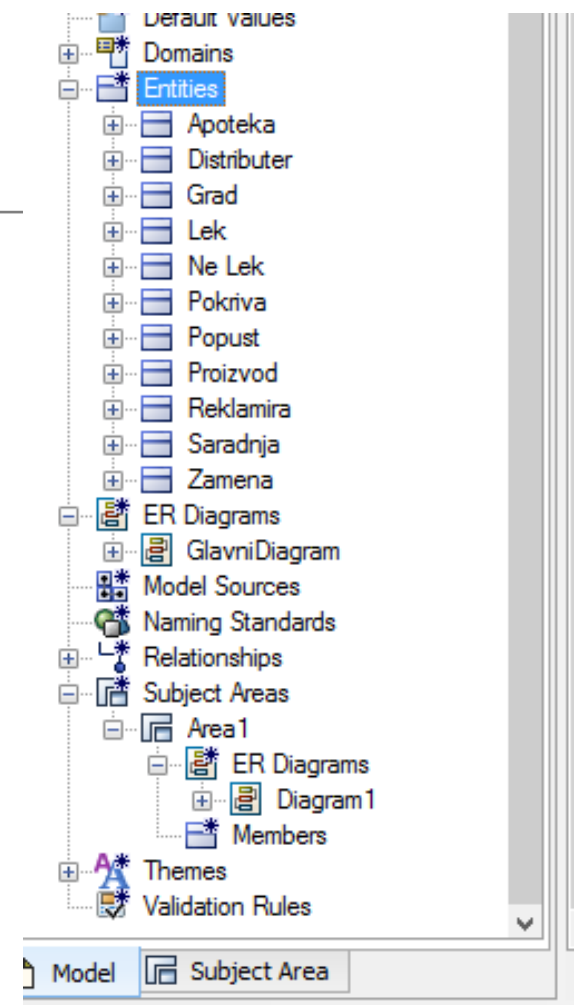
Usled mogućnosti da model neke baze bude veliki sa dosta entiteta i odnosa među entitetima potrebno je podeliti ga na određene celine koje se nazivaju „Subject Areas“.



Kreiramo novi „Subject Area“.

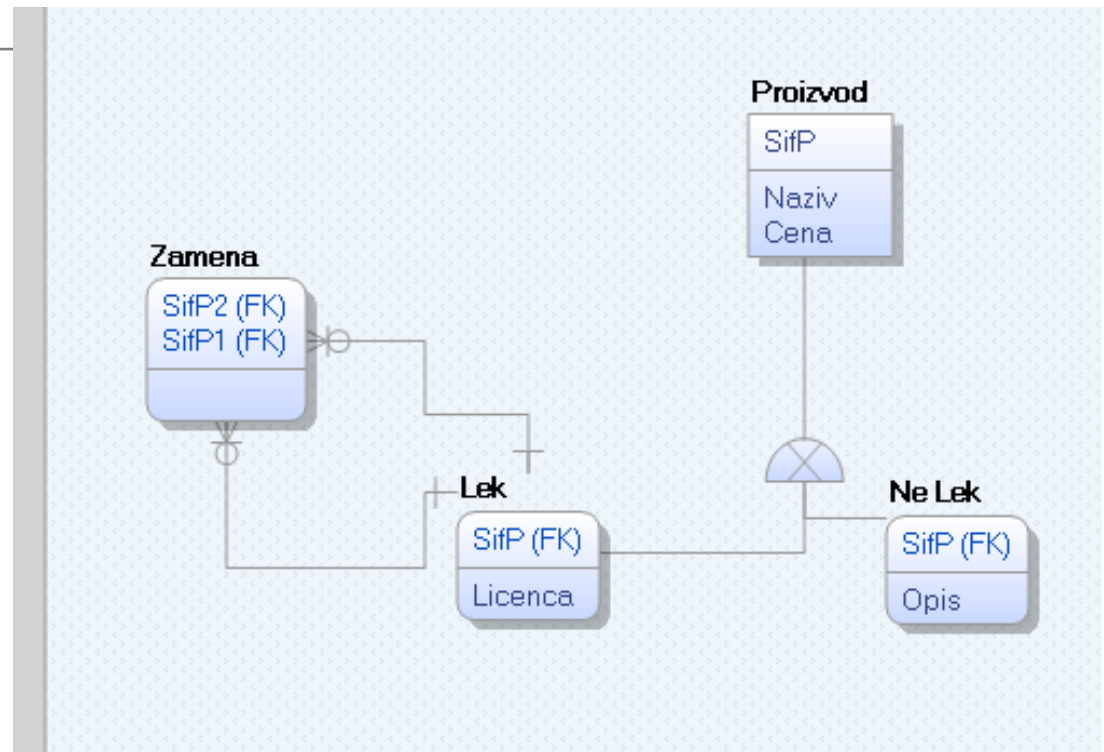
Subject Areas

Za novu oblast alat nam je generisao novi dijagram koji je prazan.
Prevlačenjem entiteta iz modela na dijagram novo kreirane oblasti
popunjavamo dijagram.

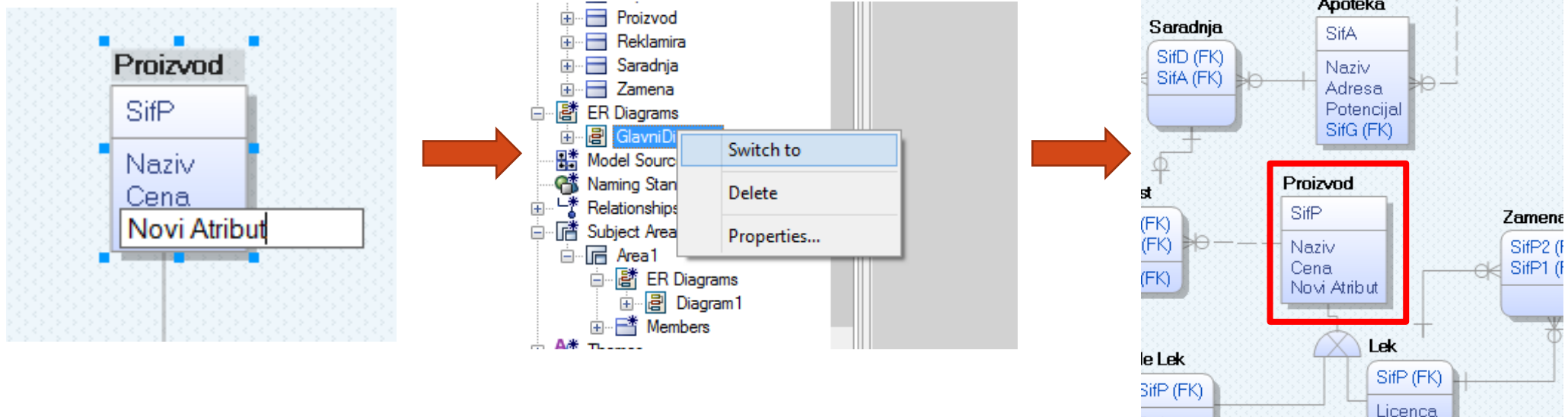


Subject Areas

Subjet area sa entitetima od interesa.



Subject Areas



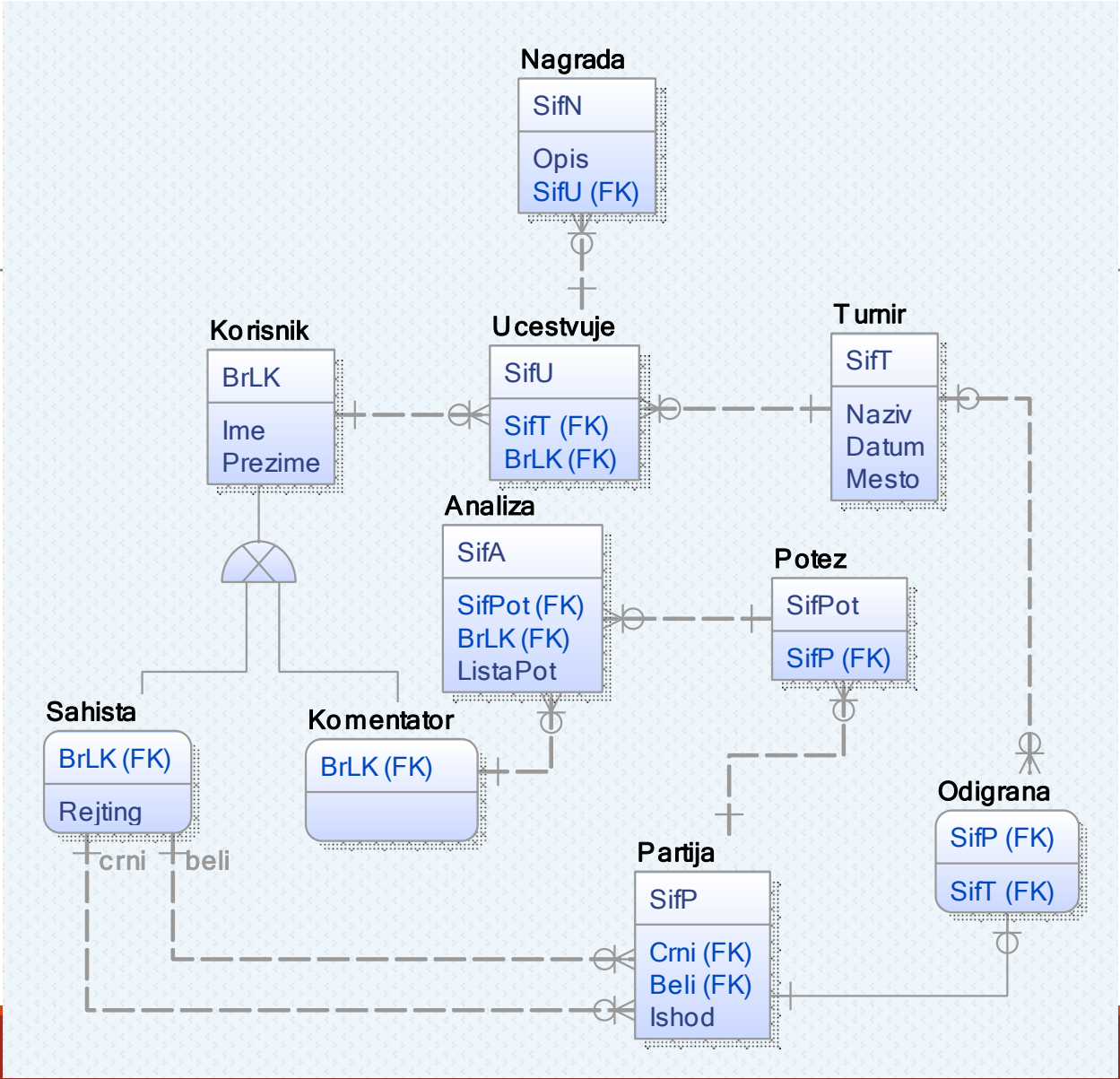
Svaka promena na jednom dijagramu biće automatski vidljiva i na svim ostalim dijagramima koji imaju izmenjeni entitet.

Zadaci



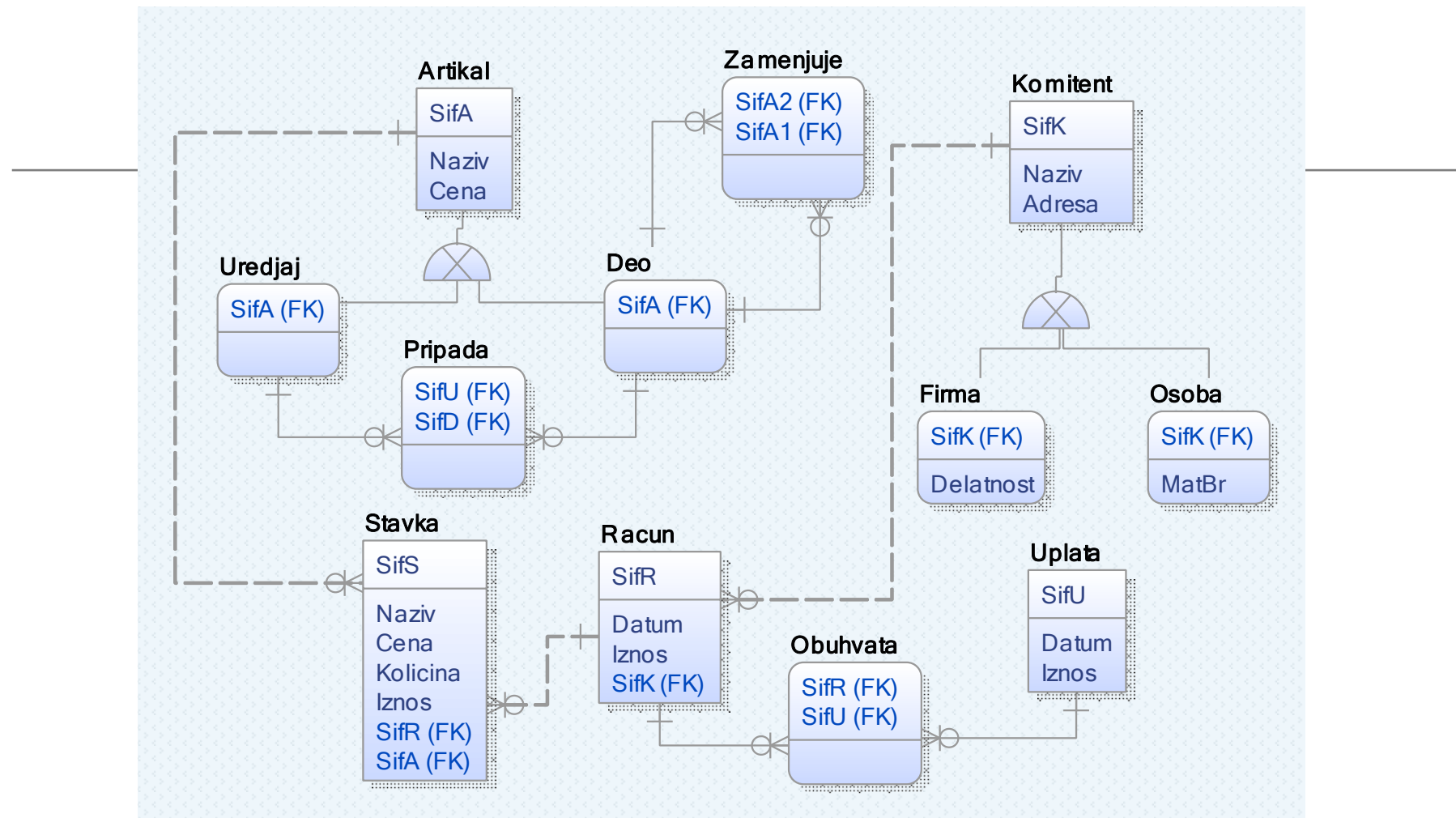
1. Šahovski savez

Projektovati sistem za potrebe šahovskog saveza. U sistemu treba voditi evidenciju o korisnicima (ime, prezime, br.l.k), od kojih neki mogu biti šahisti (rejting) ili komentatori. U bazi se vodi evidencija o održanim turnirima (naziv, datum i mesto održavanja), kao i korisnicima koji su učesnici turnira i svim šahovskim partijama odigranim na turniru. Partija može biti evidentirana iako nije odigrana u okviru turnira. Za svaku partiju mora da postoji podatak o igračima i o tome sa kojim su figurama igrali (belim ili crnim) kao i o krajnjem ishodu partije. Proizvoljan broj komentatora može da daje svoju analizu na svaki od poteza povučenog u partiji. Analiza se sastoji od liste mogućih poteza koji mogu da uslede nakon posmatranog. Na turniru se nekim učesnicima može dodeliti novčana nagrada.



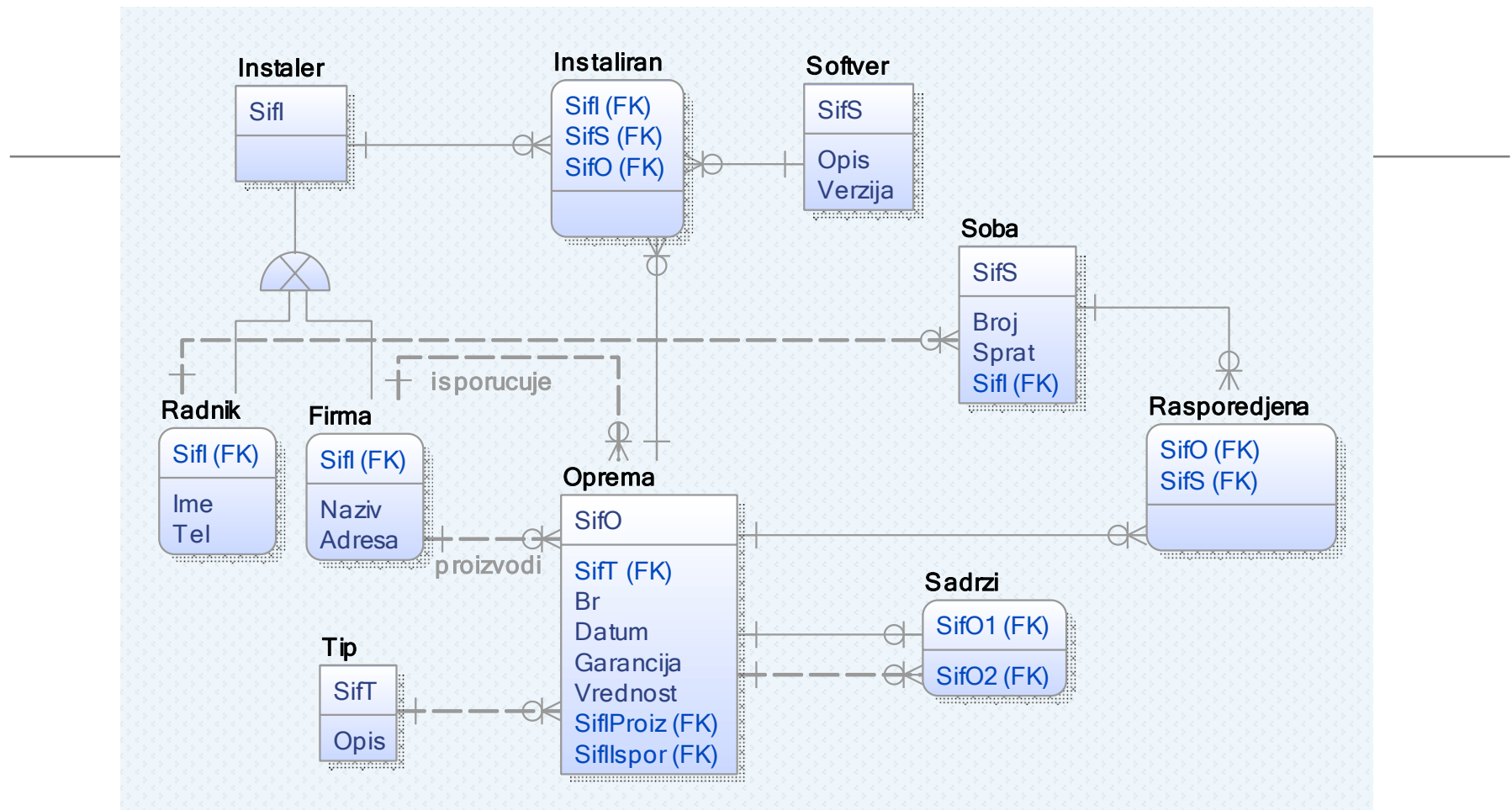
2. Prodavnica uređaja

Posmatrani sistem je jedna prodavnica uređaja, za koje se prati naziv i cena, i delova za koje se prate isti podaci. Prati se i to koji deo pripada kojim uređajima, kao i koji deo u slučaju nedostatka može biti zamenjen kojim delovima. Prodavnica radi sa komitentima, za koje se prate naziv i adresa, a oni su ili firme, za koje se prati i delatnost, ili osobe, za koje se prati i matični broj. Za svaku obavljenju kupovinu prodavnica komitentu ispostavlja račun sa unikatnim brojem, datumom i iznosom. Račun sadrži i jednu ili više stavki. Svaka stavka se odnosi na jedan uređaj ili deo (može i jedno i drugo u istom računu) i kao podatke sadrži šifru, naziv, cenu, količinu i iznos. Komitenti plaćaju ono što su kupili u vidu jedne ili više uplata za koje se prate datum i iznos. Uplata se može odnositi i na više računa.



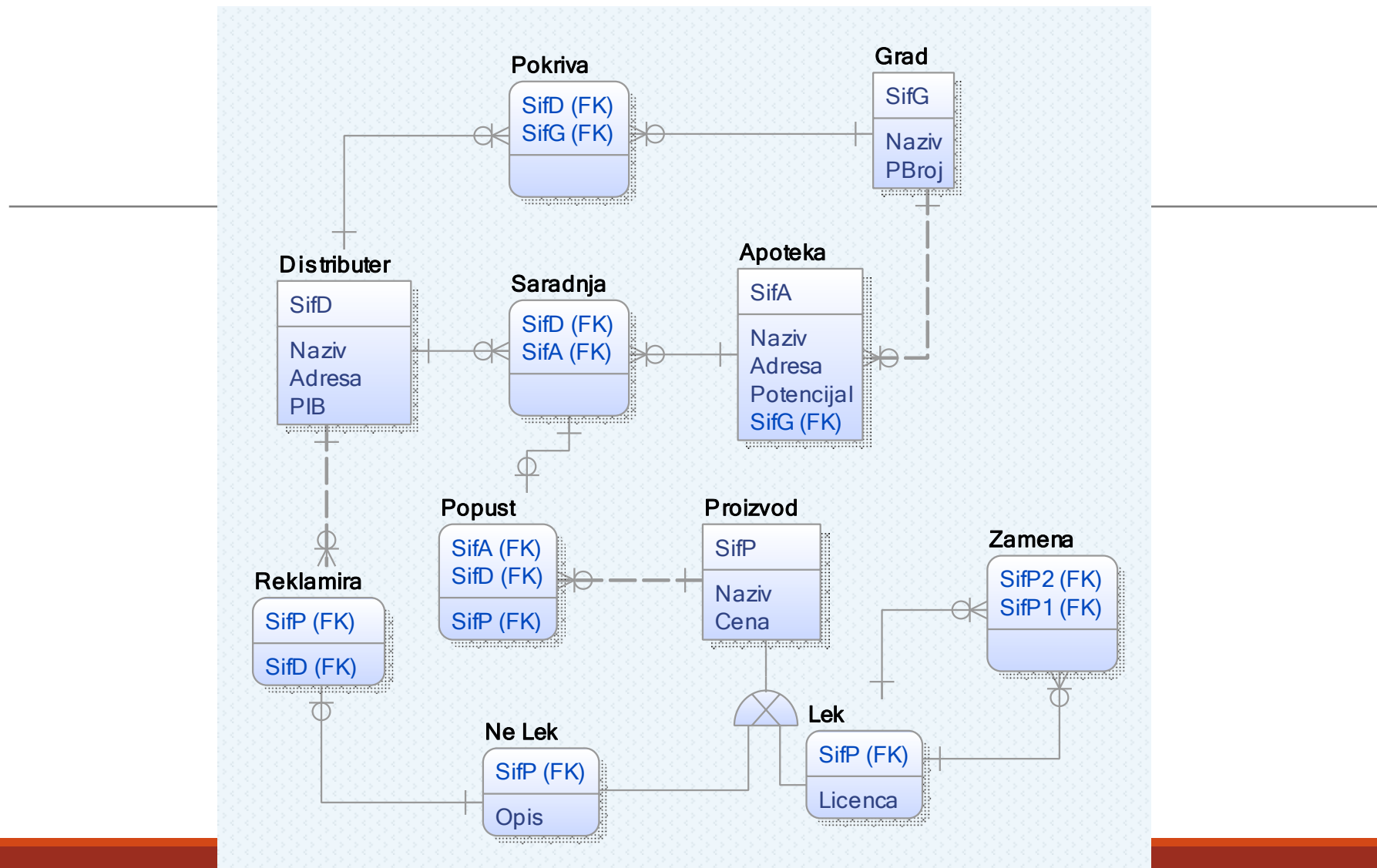
3. IT inventar

Projektovati bazu podataka za potrebe vođenja IT inventara (opreme) u nekoj kompaniji. U sistemu treba voditi evidenciju o inventarskom broju, datumu početka upotrebe, trajanju garancije, vrednosti, tipu i opisu opreme, a takođe i o nazivu i adresi firme koja je isporučila opremu a koja ujedno može, a ne mora, biti proizvođač te opreme. Oprema može biti raspoređena u jednoj ili više soba koje pripadaju kompaniji, pri tom svaka soba poseduje jedinstveni identifikacioni broj i informaciju na kom se spratu nalazi. Svaka soba mora imati definisanu odgovornu osobu (ime, broj telefona) koja je radnik kompanije. Oprema može da se sastoji iz više delova koji su takođe oprema. U sistemu je potrebno voditi evidenciju i o raznovrsnom softveru koji može biti instaliran na većem broju opreme, pri tom je potrebno voditi evidenciju o tome ko je i kada obavio instalaciju. Instalaciju može obaviti osoba koja je radnik kompanije ili neka od firmi sa kojom kompanija saraduje.



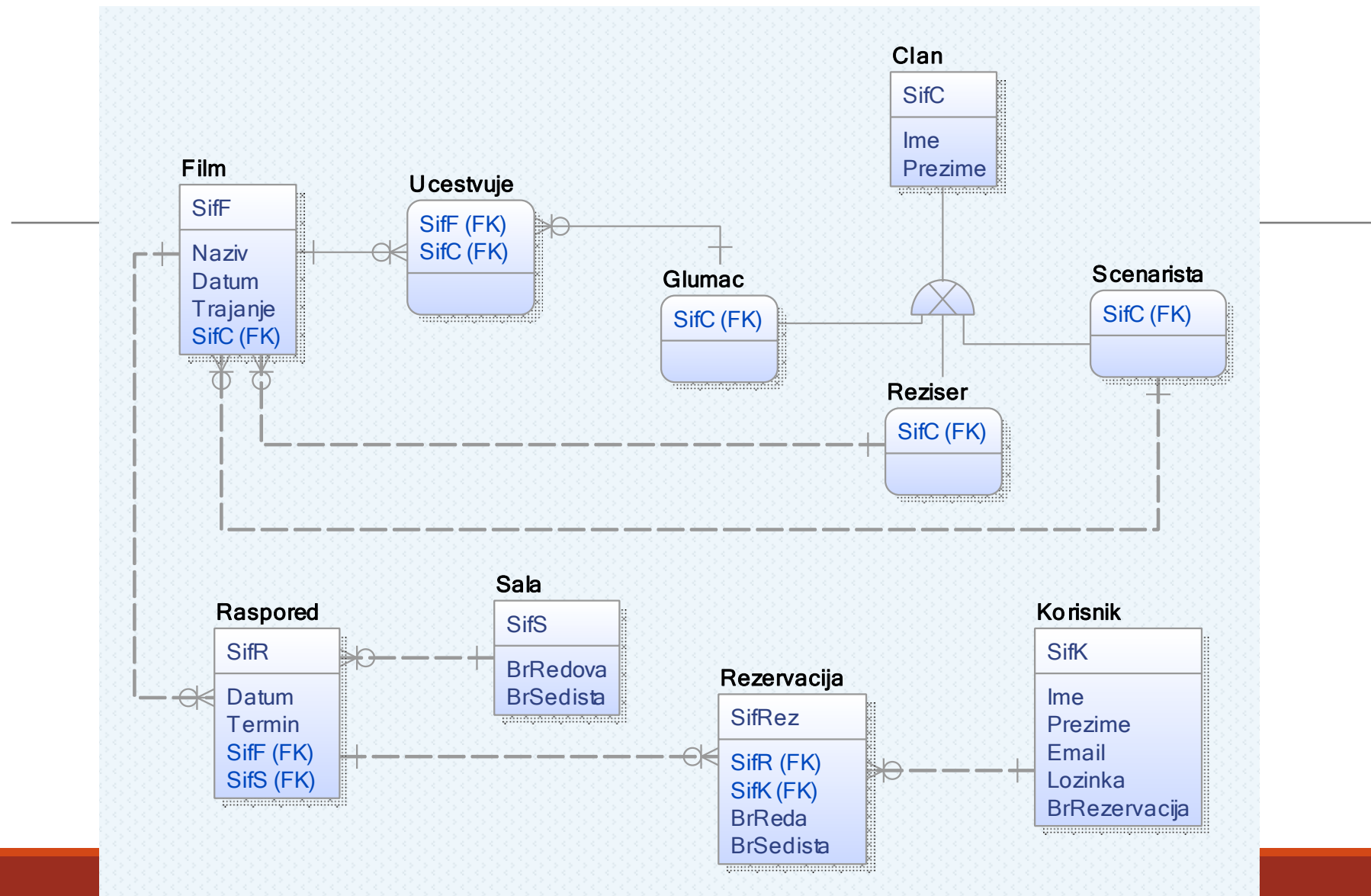
4. Farmaceutska kompanija

Posmatra se deo informacionog sistema farmaceutske kompanije. Kompanija poseduje veći broj distributera za koje se čuva naziv, adresa, PIB i informacije o gradovima koje taj distributer pokriva. Distributeri saraduju sa apotekama koje takođe mogu saradivati sa većim brojem distributera. Za svaku apoteku se prati njen potencijal prodaje, naziv, adresa i grad u kome se nalazi. Svaki distributer može svakoj od apoteka sa kojom saraduje obezbediti cenu bolju nego inače ali samo za maksimalno jedan od proizvoda kompanije. Kompanija ima proizvode (naziv, cena) koji su ili lekovi (licenca) ili ne-lek preparati (opis). Za svaki lek se može čuvati informacija o drugim lekovima koji mogu poslužiti kao zamena za taj lek. Dok se za ne-lek preparat može određenom distributeru odobriti reklamiranje.



5. Bioskop

Posmatra se informacioni sistem bioskopa. Postoje filmovi, koji imaju svoj naziv, datum izlaska, trajanje, režisera, scenaristu. U jednom filmu može učestvovati više glumaca, dok jedan glumac može učestvovati u više filmova. Član filmske ekipe može biti glumac, režiser ili scenarista i on ima svoje ime i prezime. U bioskopu postoje sale, gde se za svaku pamti broj redova i broj sedišta po redu (smatrati da svaki red u sali ima isti broj sedišta). U sistemu postoje i registrovani korisnici za koje se pamte ime, prezime, email adresa, lozinka i broj rezervacija. Svaki film se može naći u rasporedu prikazivanja, i on se može prikazivati u određenoj sali, određenog dana, u nekom terminu. Svaki registrovani korisnik može napraviti više rezervacija za određene termine u rasporedu prikazivanja. U svakoj rezervaciji se čuva rezervisani broj reda i sedišta u redu one sale za koju je termin rezervisan.



6. Firme

Posmatra se baza podataka za registrovanje firmi, radnika i radnih mesta. Za radnika se pamti ime, prezime, broj lične karte i kancelarija u kojoj radi. Za firmu se čuva naziv i PIB. Ona može imati više kancelarija. Kancelarija se nalazi u određenom gradu, ima svoju ulicu i broj, kao i broj radnika koji u njoj trenutno rade. Svaki grad ima svoj naziv i državu u kojoj se nalazi. Takođe, kancelarija ima definisanu listu funkcija, za koje se pamte nazivi, i broj slobodnih mesta za svaku od njih. Svaki radnik može obavljati najviše dve funkcije u nekoj firmi, a smatra se radnikom ukoliko obavlja bar jednu funkciju. Svaka funkcija ima definisanu platu čiji iznos zasebno definiše svaka firma.

