

Projektiranje informacijskih sustava

Modeliranje podataka
Ak. god. 2009/2010



Model podataka

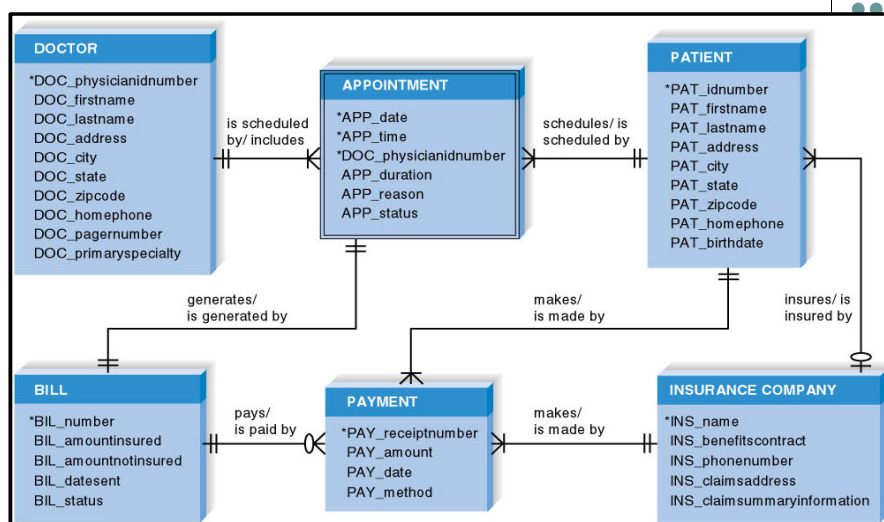
- Model podataka je formalni način predstavljanja podataka koji se koriste ili stvaraju u poslovnom procesu.
- U fazi analize razvija se logički model podataka koji se prebacuje u fizički model podataka tijekom faze dizajna.
- Logički model podataka opisuje podatke bez da sugeriraju kako se oni pohranjuju, kreiraju ili mijenjaju.
- Model podataka treba biti balansiran sa modelom procesa.

2

Model podataka

- Najčešće korištena tehnika za modeliranje podataka je ERD (*entity-relationship diagram*) (Peter Chen, 1976).
- ERD dijagram grafički prikazuje informacije koje se kreiraju, pohranjuju ili koriste u sustavu.
- Sastoji se od entiteta (koncepta) koji predstavljaju kolekciju sličnih informacija (npr. entitet osobni podaci sadrži ime, prezime, godinu rođenja,..., entitet adresa sadrži ime ulice, kućni broj, grad,...) prikazanih pravokutnikom.
- Linije između entiteta definiraju relacije među podacima.
- Pored relacija koje definiraju poslovna pravila nad podacima (npr. svaka osoba ima adresu), navode se i dodatne oznake koje definiraju poslovna pravila (npr. jedna osoba se može povezati sa jednom ili više adresa).

3



4

ELEMENTI ERD-a



	IDEF1X	Chen	Crow's Foot
An ENTITY: ✓ Is a person, place, or thing ✓ Has a singular name spelled in all capital letters ✓ Has an identifier ✓ Should contain more than one instance of data	ENTITY-NAME 	ENTITY-NAME 	ENTITY-NAME
An ATTRIBUTE: ✓ Is a property of an entity ✓ Should be used by at least one business process ✓ Is broken down to its most useful level of detail	ENTITY-NAME 		ENTITY-NAME
A RELATIONSHIP: ✓ Shows the association between two entities ✓ Has a parent entity and a child entity ✓ Is described with a verb phrase ✓ Has cardinality (1 : 1, 1 : N, or M : N) ✓ Has modality (null, not null) ✓ Is dependent or independent			

5

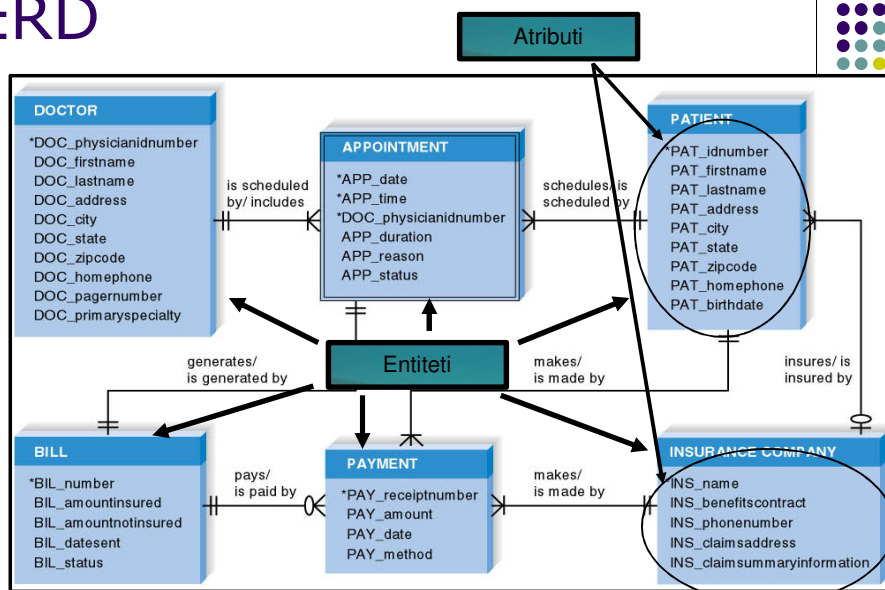
ERD



- Tri osnovna elementa ERD-a su:
 - entitet, atribut i relacija
- Entitet je osnovni građevni blok ERD-a. To može biti osoba, odgađaj, mjesto ili stvar, tj. bilo što o čemu se skupljaju podaci (student, ispit, ...).
- Svaki entitet ima Ime i Identifikator. Ime entiteta je imenica i piše se velikim slovima. Identifikator jednoznačno određuje jednu instancu nekog entiteta.
- Važno svojstvo entiteta je da predstavlja nešto što može imati puno instanci (kao klasa u OO programiranju). Npr. profesor je entitet, dok Maja Štula ne može biti entitet. Znači kada postoji samo jedna instanca nekog podatka onda se ne definira kao entitet.
- Atributi su dostupne informacije o entitetu, označavaju se imenicom, mogu se koristiti i kao identifikatori entiteta. (npr. entitet PROFESOR ima attribute ime, prezime, smjer, ...). Za svaki entitet moguće je definirati veliki broj entiteta, ali potrebno je ograničiti se samo na one entitete koji se stvarno i koriste u poslovnom procesu.

6

ERD

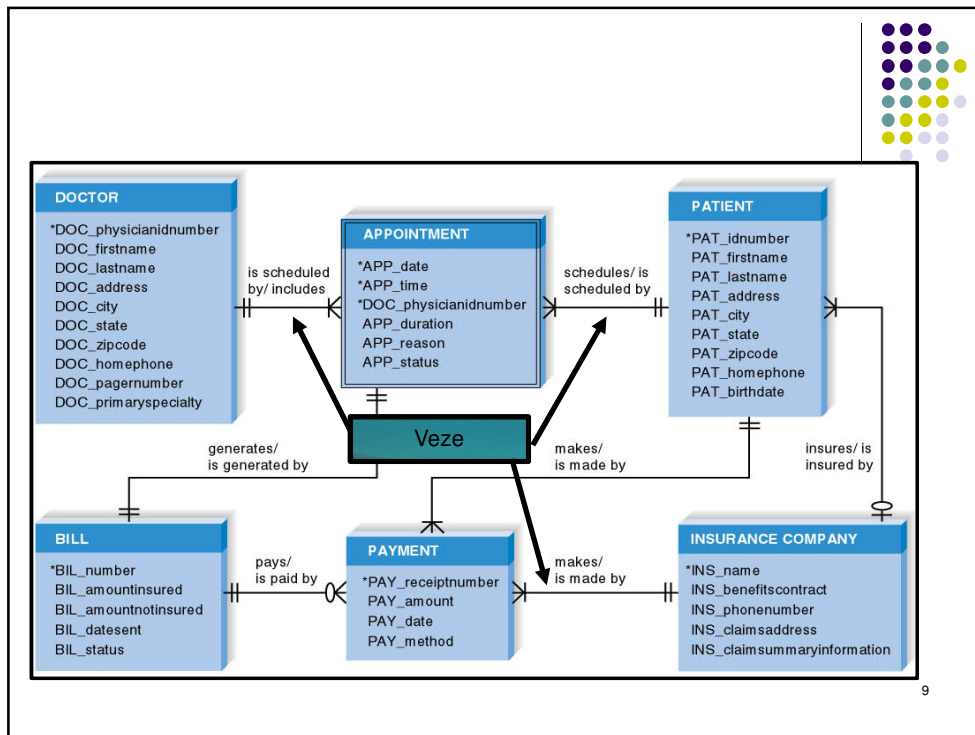


7

ERD

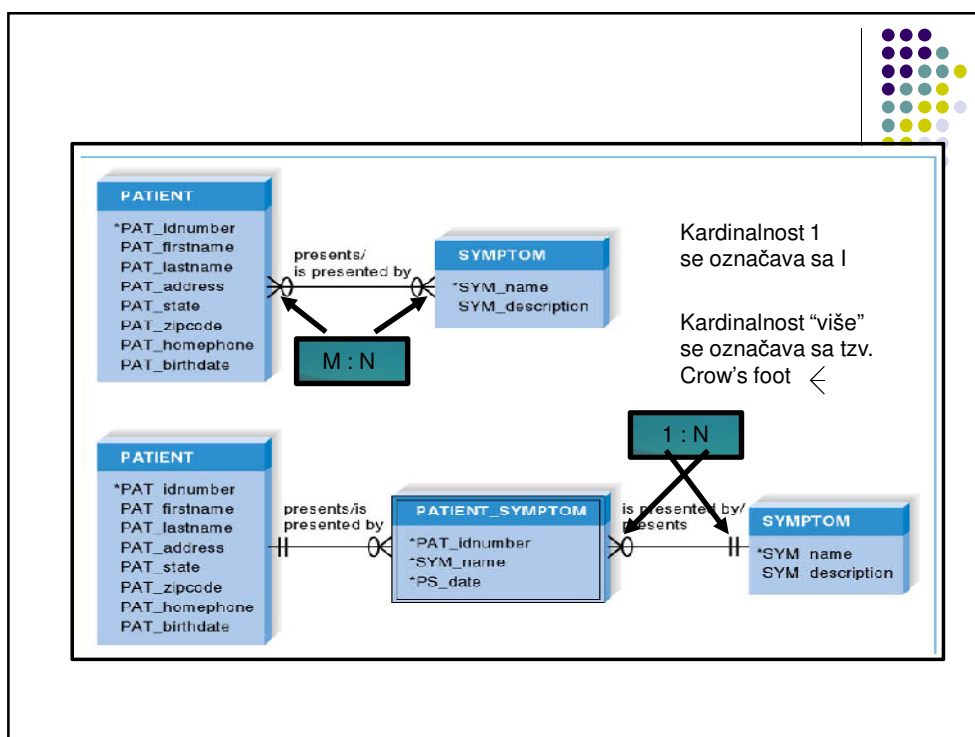
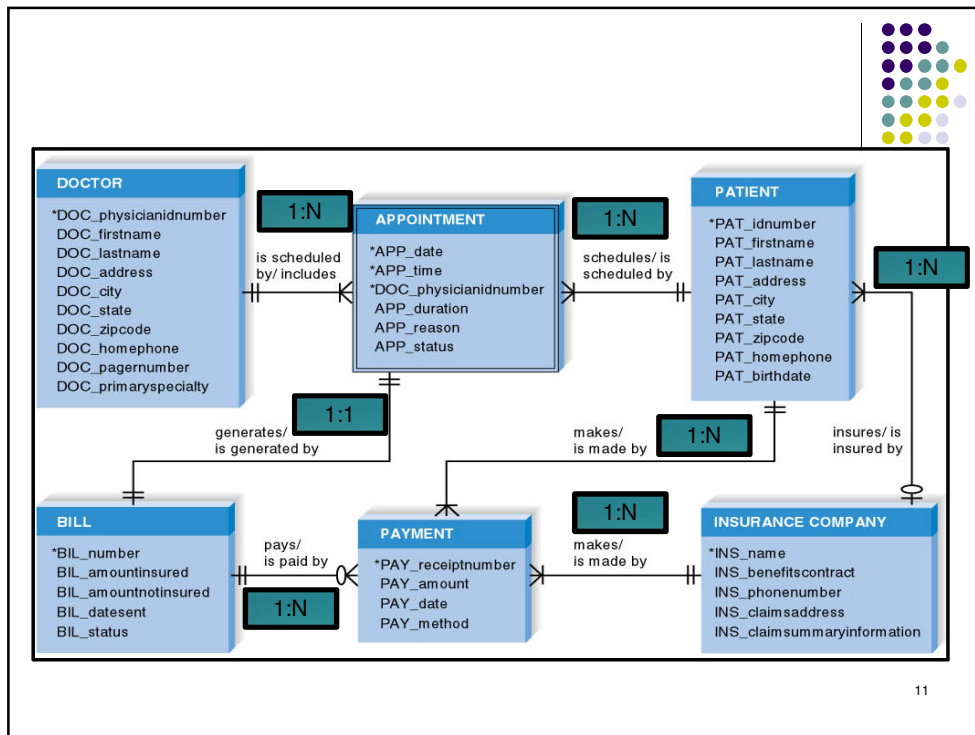
- Veza/relacija je asocijacija među entitetima.
- Prvi entitet u relaciji je entitet roditelj, a drugi entitet je dijete.
- Relacije bi trebale biti imenovane glagolom.
- Relacije su dvosmjerne.

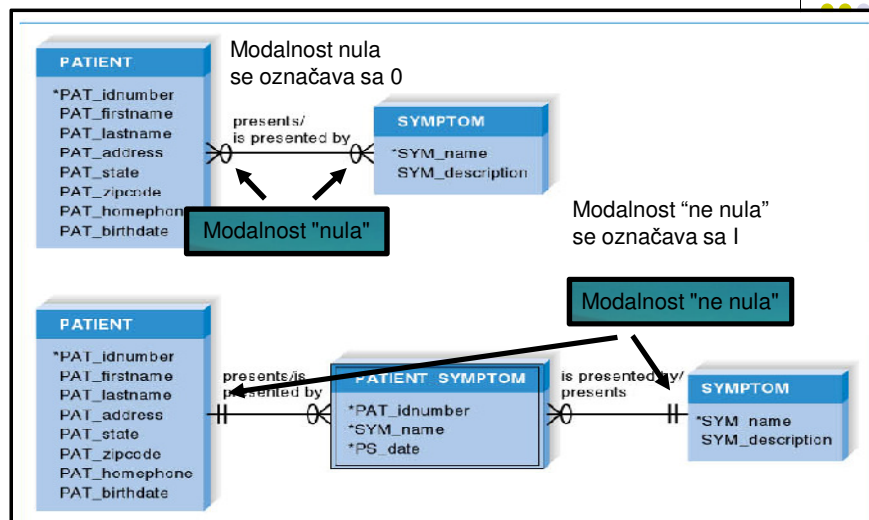
8



ERD

- Binarne relacije definirane su sa dva svojstva:
 - Kardinalnost - određuje koliko instanci entiteta roditelja je asocirano sa koliko instanci entiteta djeteta. Moguće kardinalnosti relacije su 1:1, 1:N ili M:N
 - Modalnost - određuje da li instanca *child* entiteta može postojati bez instance *parent* entiteta, moguće modalnosti su "nula" ili "ne nula".



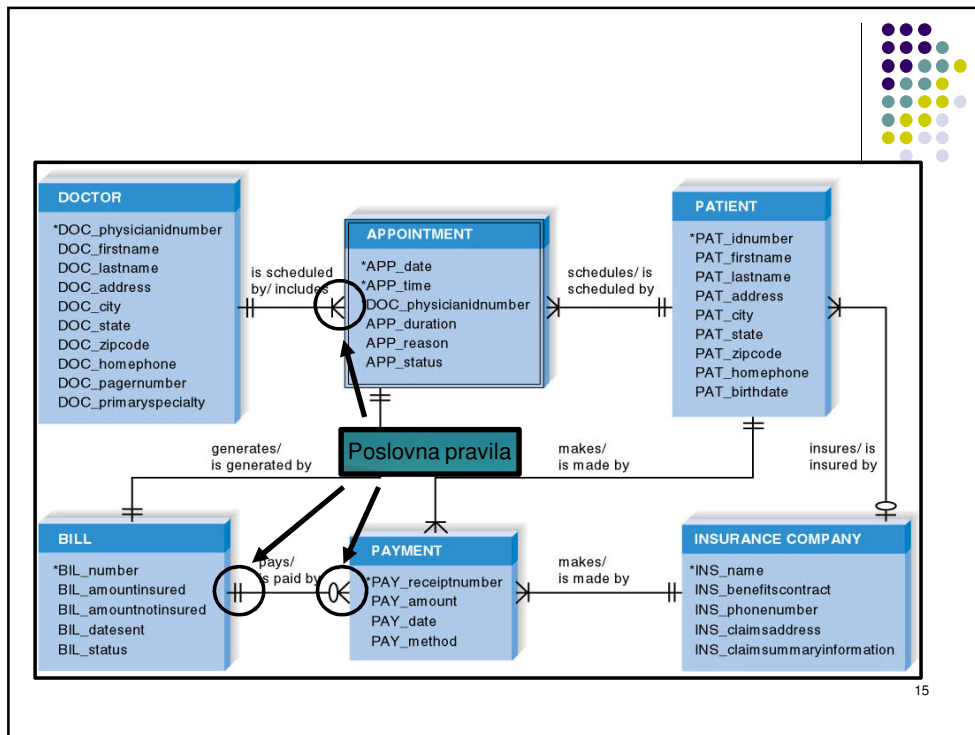


13

ERD

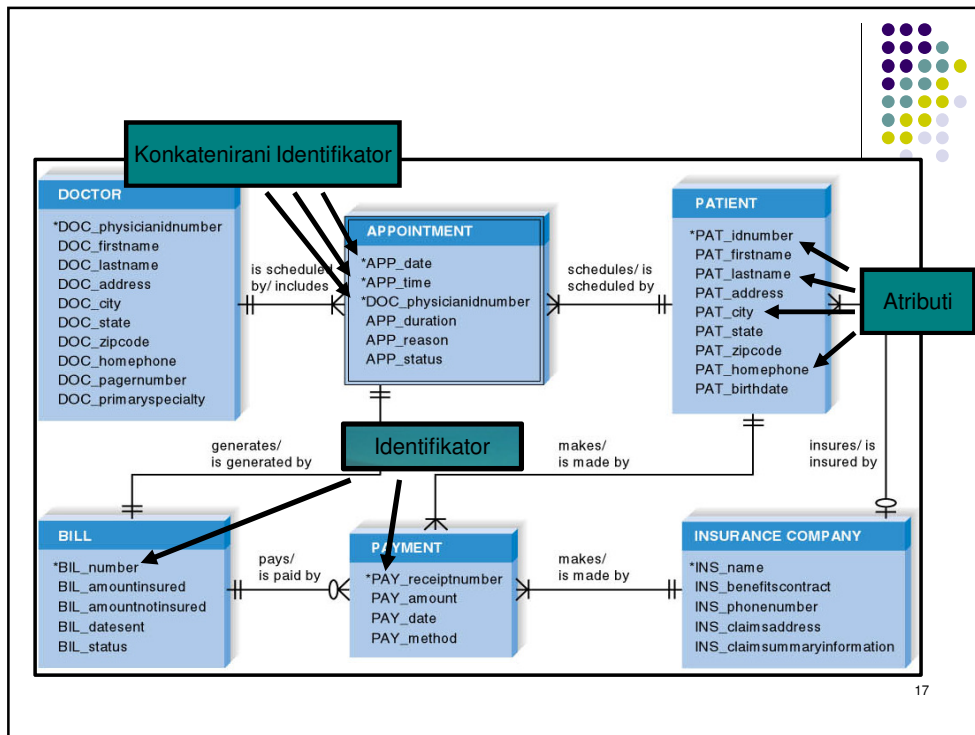
- Poslovna pravila su ograničenja koja se trebaju poštivati tijekom rada sustava.
- ERD simboli omogućavaju prikazivanje poslovnih pravila.
 - Kada instanca jednog entiteta mora postojati da bi instanca drugog entiteta postojala
 - Kada je instanca jednog entiteta povezana sa samo jednom ili sa više instanci drugog entiteta

14



Identifikatori entiteta

- Jadan ili više atributa entiteta mogu poslužiti kao identifikatori entiteta (*entity identifier*), koji jedinstveno identificiraju svaku instancu entiteta
- Spojeni identifikator (*concatenated identifier*) se sastoji od nekoliko atributa
- Identifikator može biti i neki "umjetni podatak" poput ID broja za određeni entitet (*ID number*)
- Identifikatori se točno određuju u fazi dizajna sustava



ERD

- Zavisni entitet:
 - u slučaju kada dijete entitet zahtjeva atribut od entiteta roditelj da bi jedinstveno identificirali instancu onda se dijete entitet naziva zavisni entitet
 - relacija koja uključuje zavisni dijete entitet naziva se identificirajuća relacija
- Nezavisni entitet:
 - entitet koji se može identificirati bez pomoći nekog drugog entiteta
 - relacija koja uključuje nezavisni dijete entitet zove se neidentificirajuća relacija
- Presjecajući entitet:
 - postoji u svrhu prikupljanja informacija o relaciji između dva entiteta
 - dodaju se u podatkovni model da bi pohranili informacije o dvoje entiteta koji dijele M:N relaciju
 - nazivaju se još i asocijativni entiteti
 - proces dodavanja presjecajućeg entiteta naziva se razdvajanje M:N relacije
 - postoje tri koraka za dodavanje presjecajućeg entiteta:
 - uklanjanje M:N relacije i umetanje novog entiteta između dva postojeća
 - dodavanje dvije 1:N relacije u podatkovni model
 - dodjela imena presjecajućem entitetu

18

Postupak izrade ERD



- Postupak izrade ERD-a se sastoji od tri koraka:
 1. Identifikacija entiteta
 2. Identifikacija atributa entiteta
 3. Identifikacija relacija

19

Balansiranje dijagrama entiteta i relacija (ERDs) s dijagramom toka podataka (DFDs)



- Iako se model procesa fokusira na procese poslovnih sustava, sastoji se od dvije podatkovne komponente:
 - tok podataka
 - spremište podataka
- Svrha je da se ilustrira koji podaci su korišteni i kreirani od strane procesa i gdje se drže, tj. gdje su pohranjeni ti podaci
- Te komponente dijagrama toka podataka trebaju biti balansirane s dijagramom entiteta i relacija
- Mnogi CASE alati nude mogućnost identifikacije problema kod balansiranja između DFD-a i ERD-a.

20

Balansiranje dijagrama entiteta i relacija (ERDs) s dijagramom toka podataka (DFDs)



- Trebamo provjeriti da li su svi podaci u spremištima i tokovima podataka u procesnom modelu uključeni negdje kao entitet ili atribut entiteta u modelu podataka.
- Želimo osigurati da model u potpunosti sadrži podatke predstavljene u procesnom modelu.
- Ako to nije osigurano onda je model podataka nepotpun.
- Svi podatkovni elementi trebali bi se pojaviti kao dio spremišta podataka i toka podataka u procesnom modelu.
- Ako neki podatkovni elementi nedostaju u procesnom modelu, trebamo provjeriti da li su ti podaci zaista potrebni u radu sustava.

21

CRUD



- Koristan alat za jasno prikazivanje relacija između procesnog modela i modela podataka je CRUD (Create, Read, Update, Delete) matrica.
- To je tablica koja opisuje kako procesi koriste podatke unutar sustava.
- CRUD matrica(e) se najprije razvija na logičkom procesu i modelu podataka, a onda se preispituje u fazi dizajna.
- Matrica također pruža važne informacije za specifikaciju programa zato jer pokazuje točno kako su podaci korišteni i stvarani od strane glavnih procesa sustava.

22

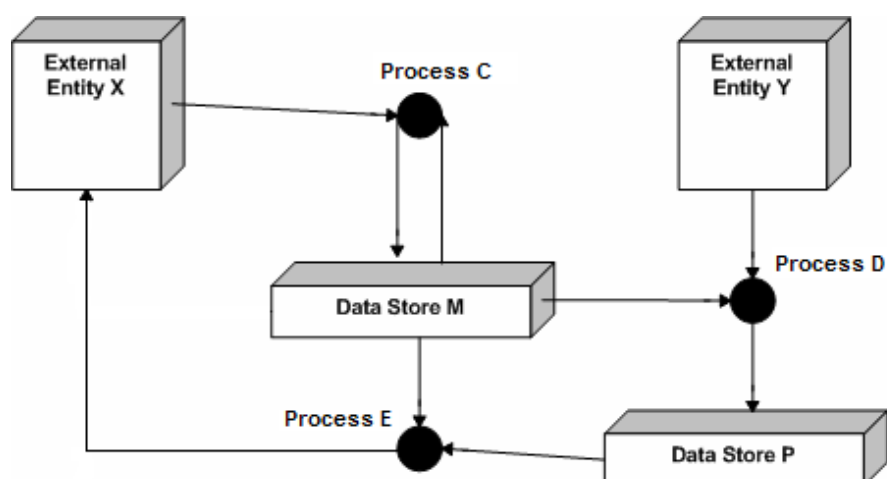
Kreiranje CRUD matrice



- Da bi kreirali CRUD matricu, nacrtamo tablicu navodeći sve procese u prvoj liniji, a entitete i njihove attribute u prvom stupcu s lijeve strane (ili obrnuto).
- Zatim ovisno o informacijama u procesnom modelu analitičar ispunjava svaku ćeliju tablice sa C,R,U,D slovom ili ostavlja praznu da bi opisao interakciju pojedinog procesa sa pojedinim entitetom.

23

Primjer



Primjer



	Process C	Process D	Process E
Data Entity M			
Attribute M-1	CRUD	R	R
Attribute M-2	CRUD		R
Attribute M-3	CRUD	R	
Attribute M-4	CRUD		R
Data Entity P			
Attribute P-1		C	R
Attribute P-2		C	
Attribute P-3		C	R

25

Primjer



ACTIVITIES	DATA ENTRIES										
	Catalog	Customer	Inventory item	Order	Order item	Order transaction	Package	Product item	Return item	Shipment	Shipper
Look up item availability			R								
Create new order		CRU	RU	C	C	C	R	R		C	R
Update order		RU	RU	RUD	RUD	RUD	R	R		CRUD	R
Look up order status		R		R	R	R				R	R
Record order fulfillment					RU					RU	
Record back order					RU					CRU	
Create order return		CRU		RU		C			C		
Provide catalog info	R		R				R	R			
Update customer account		CRUD									
Distribute promotional package	R	R	R				R	R			
Create customer charge adjustment		RU				CRUD					
Update catalog	RU		R				RU	R			
Create special promotion	R		R				R	R			
Create new catalog	C		R				CRU	R			

C = Creates new data, R = Reads existing data, U = Updates existing data, D = Deletes existing data

26

CRUD matrica



- Možemo identificirati mjesta gdje su atributi možda izostavljeni iz spremišta podataka ili entiteta.
- Nadalje, možemo provjeriti da se svaki atribut kreira, čita, ažurira i briše negdje u procesnom modelu.
- Ako nije pročitao od strane nekog procesa, onda atribut vjerojatno nije potreban.
- Ako nije kreiran ili ažuriran, atribut vjerojatno treba dodati u tok podataka u procesnom modelu.

27

Validacija ERD-a



- Normalizacija je tehnika koja se koristi za validaciju podatkovnih modela, to je proces u kojem se niz pravila primjenjuje na logički model da bi se odredilo koliko je dobro model formiran.
- Uobičajeno se koriste tri normalizacijska pravila ili forme.
- Normalizacijom se iz postojećih podataka nastoji ukloniti redundantne podatke tj. iste podatke koji postoje na dva mjesta (direktna redudancija) ili podatke koji se mogu izvući iz drugih podataka (indirektna redudancija), a zadržati integritete podataka.

28

Normalizacija – 1NF



- Logički model podataka je u prvoj normalnoj formi (1 Normal Form (1NF)) ako ne sadrži ponavljajuće atribute (atributi koji sadrže višestruke vrijednosti za jednu instancu).
- Podaci se dovode u 1NF:
 - Uklanjanjem višestrukog pojavljivanja istog stupca u tablici.
 - Kreiranjem posebne tablice za svaki skup povezanih podataka te identificiranjem jedinstvenim ključem svakog redka tablice.

29

Normalizacija – 1NF

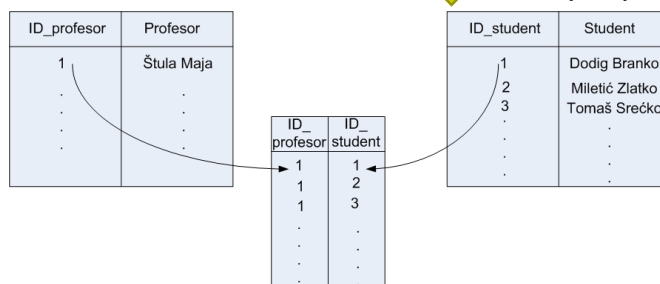


- Što ako dođe treći student?
- Što ako se promjeni prezime jednog studenta?

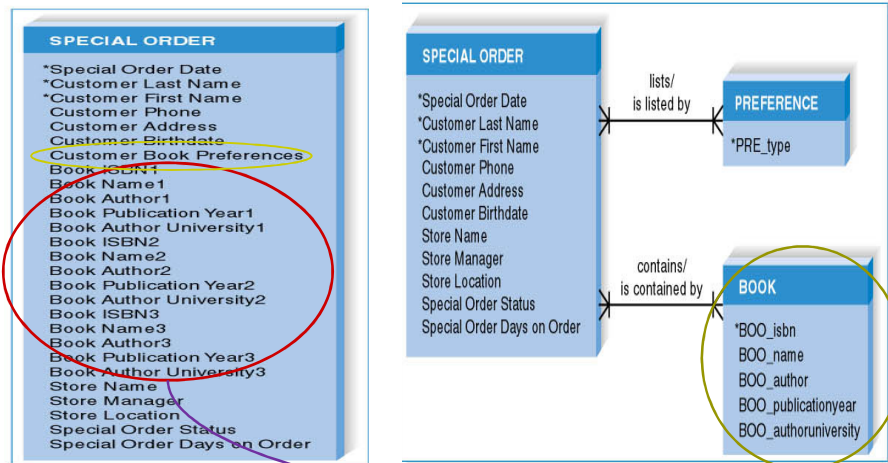
Profesor	Student 1	Student 2	Student 3
Štula Maja	Dodig Branko	Miletić Zlatko	Tomaš Srećko
.	.	.	.
.	.	.	.
.	.	.	.



Rješenje



Normalizacija – 1NF



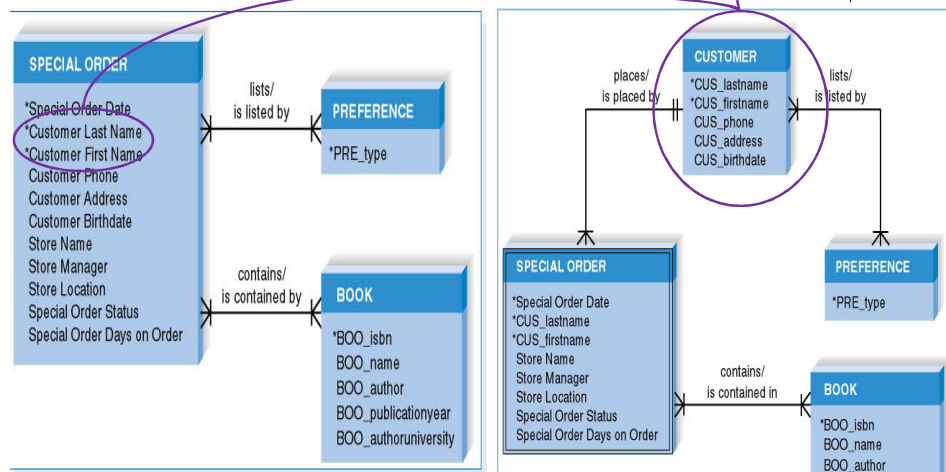
31

Normalizacija – 2NF

- Druga normalna forma (2NF) zahtjeva da svi entiteti budu u prvoj normalnoj formi (1NF).
- Podaci se dovode u 2NF:
 1. Ako entitet ima spojeni identifikator potražiti attribute koji ovise samo o dijelu identifikatora i odvojiti ih u zasebni entitet.

32

Normalizacija – 2NF



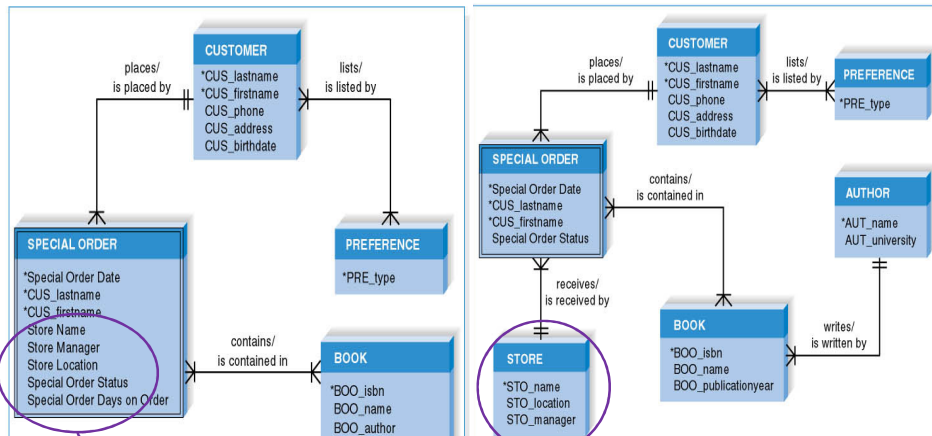
33

Normalizacija – 3NF

- Treća normalna forma (3NF) se može postići kada je model već i u prvoj (1NF) i u drugoj (2NF) normalnoj formi.
- Podaci se dovode u 3NF:
 1. Trebaju biti zadovoljeni svi uvjeti za prvu i drugu normalnu formu.
 2. Potražiti attribute koji ovise samo o drugom atributu koji nije identifikator i prebaciti ih u novi entitet tj. iz tablice treba ukloniti kolone koje nisu u potpunosti ovisne o primarnom ključu.

34

Normalizacija



35

Normalizacija



Do any attributes have multiple values for a single instance of an entity?	0 Normal Form Yes: Remove the repeating attributes and repeating groups. Create an entity that describes the attributes. Usually you will need to add a relationship to connect the old and new entities. No: The data model is in 1NF.
Is the identifier comprised of more than one attribute? If so, are any attribute values dependent on just part of the identifier?	1 Normal Form Yes: Remove the partial dependency. Move the attributes to an entity in which their values are dependent on the entire identifier. Usually you will need to create a new entity and add a relationship to connect the old and new entities. No: The data model is in 2NF.
Do any attribute values depend on an attribute that is not the entity's identifier?	2 Normal Form Yes: Remove the transitive dependency or derived attribute. Move the attributes to an entity in which their values are dependent on the identifier. Usually you will need to create a new entity and add a relationship to connect the old and new entities. No: The data model is in 3NF.
	3 Normal Form

36