



BAZE PODATAKA

Model Entiteti-Veže **(Entity-Relationship Model)**



Model Entiteti-Veze

- Definicija
- Ograničenja
- Dijagram E-V
- Odluke u Dizajnu
- Skupovi Slabih Entiteta
- Proširenje modela E-V
- Dizajn Bankarskog Sistema



Definicija modela Entiteti-Veze

Bazira se na percepciji realnog okruženja:

- Baza podataka se modelira kao:
 - Skup Entiteta,
 - Veza između entiteta.
- **Entitet (Entity)** je objekat koji postoji i razlikuje se od drugih objekata.
 - Primer: neka osoba, kompanija, događaj, ljubav
- Entiteti su opisani *atributima (attribute)*
 - Primer: ljudi imaju *imena* i *adrese*
- **Skup Entiteta (Entity Set)** je skup entiteta istog tipa.
 - Primer: skup svih ljudi, kompanija, drveća, praznika



Skupovi Entiteta *klijent* i *zajam*

customer_id customer_ customer_ customer_ loan_ amount
 name street city number

321-12-3123	Jones	Main	Harrison
019-28-3746	Smith	North	Rye
677-89-9011	Hayes	Main	Harrison
555-55-5555	Jackson	Dupont	Woodside
244-66-8800	Curry	North	Rye
963-96-3963	Williams	Nassau	Princeton
335-57-7991	Adams	Spring	Pittsfield

customer

L-17	1000
L-23	2000
L-15	1500
L-14	1500
L-19	500
L-11	900
L-16	1300

loan



Veza i Skup Veza (1)

- **Vezama** (Relationship) su prikazani odnosi između entiteta

Primer:

Hayes *depositor* A-102
customer entity relationship set *account* entity

- **Skup Veza** (Relationship set) je matematička relacija između $n \geq 2$ entiteta, od kojih svaki pripada nekom skupu entiteta

$$\{(e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

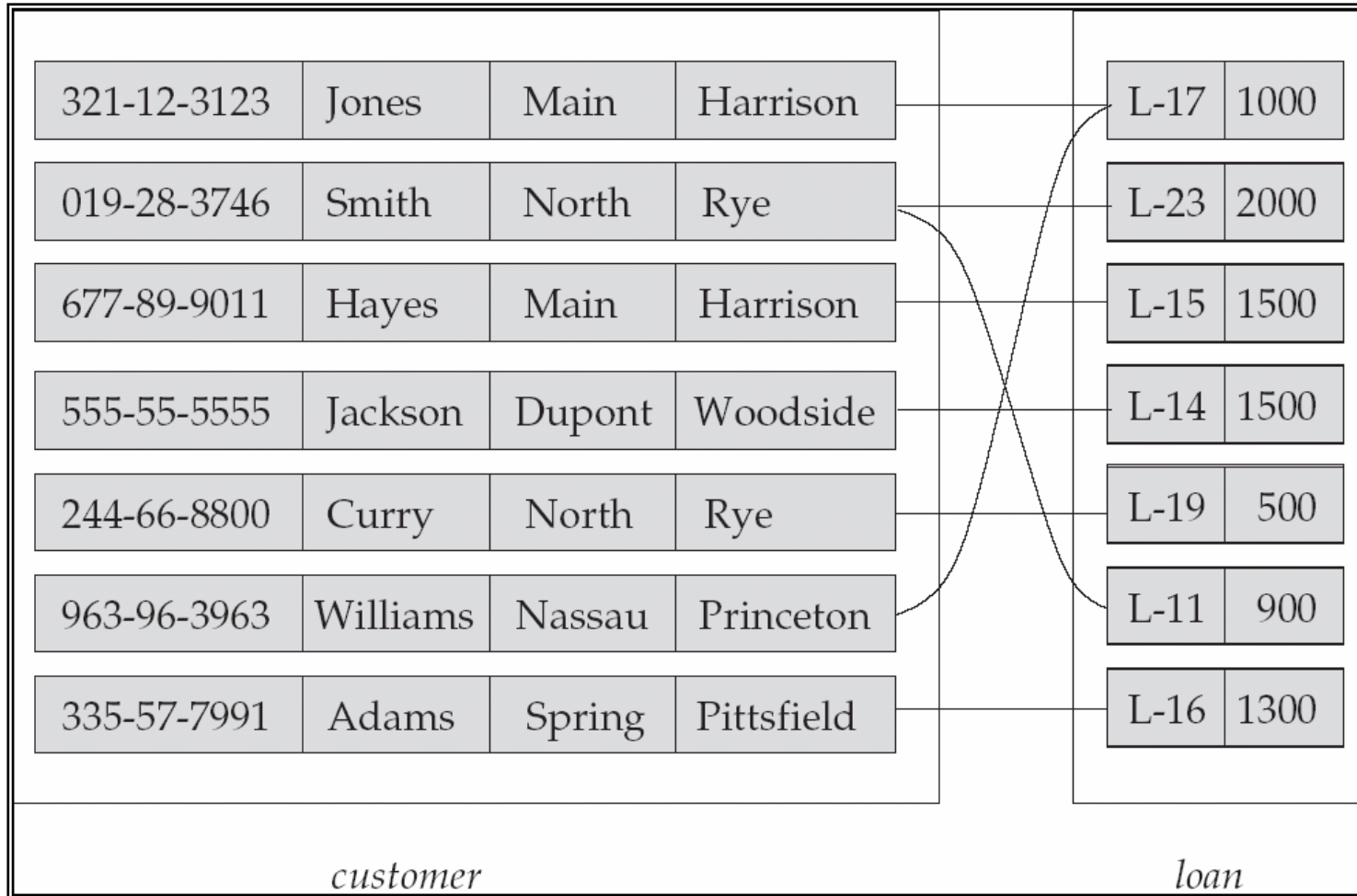
gde je (e_1, e_2, \dots, e_n) veza

– Primer:

$$(\text{Hayes}, \text{A-102}) \in \textit{depositor}$$



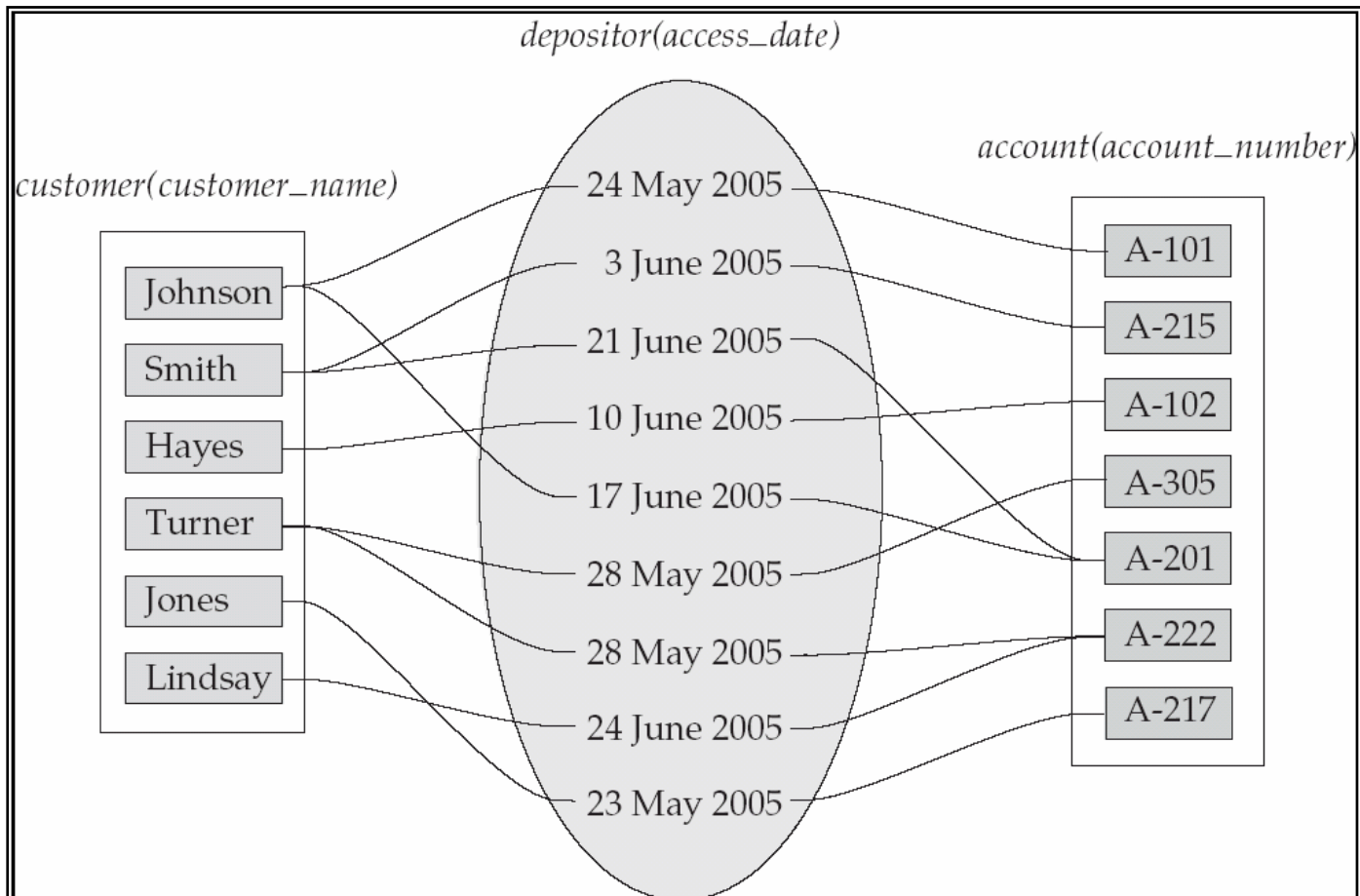
Skup Veza *pozajmiti* (2)





Skup Veza (3)

- Skup Veza može takođe imati *attribute*.
- Na primer, skup veza *ulaganje* (*depositor*) između skupova entiteta *klijent* (*customer*) i *račun* (*account*) može imati atribut *datum-pristupa* (*access-date*)





Stepen Skupa Veza

- Odnosi se na broj skupova entiteta koji učestvuju u skupu veza.
- Skupovi Veza koji uključuju dva skupa entiteta su **binarni** (ili stepena dva). Generalno, većina skupova veza u bazama podataka su binarne.
- Skupovi Veza mogu uključivati i više nego dva skupa entiteta.
 - ▶ Primer: Pretpostavimo da zaposleni banke mogu imati poslove (odgovornosti) u više ekspozitura, i to različite poslove u različitim ekspoziturama. Tada važi ternarni skup veza između skupova entiteta zaposleni(*employee*), posao(*job*), i ekspozitura(*branch*)
- Skupovi Veza koji uključuju više nego dva skupa entiteta su retki. Većina veza je binarna. (Više o tome kasnije)



Atributi

- Entitet je predstavljen skupom **Atributa**, kojima su opisane osobine pojedinih entiteta u skupu entiteta.

Primer:

Customer=(*customer_id*, *customer_name*, *customer_street*, *customer_city*)
loan = (*loan_number*, *amount*)

- **Domen** – skup dozvoljenih vrednosti svakog atributa
- Tipovi Atributa:
 - *Prosti (Simple)* i *Kompozitni (Composite)* atributi.
 - *Sa Jednom (Single-valued)* i *Više (Multi-valued)* vrednosti
 - Primer: više-vrednosni (multivalued) atribut: *phone_numbers*
 - *Sračunljivi (Derived)* atributi
 - Mogu se sračunati iz drugih atributa
 - Primer: **starost**, sračunava se iz **datum_rođenja**



Kompozitni Atributi

Composite
Attributes

name

first_name middle_initial last_name

address

street city state postal_code

Component
Attributes

street_number street_name apartment_number

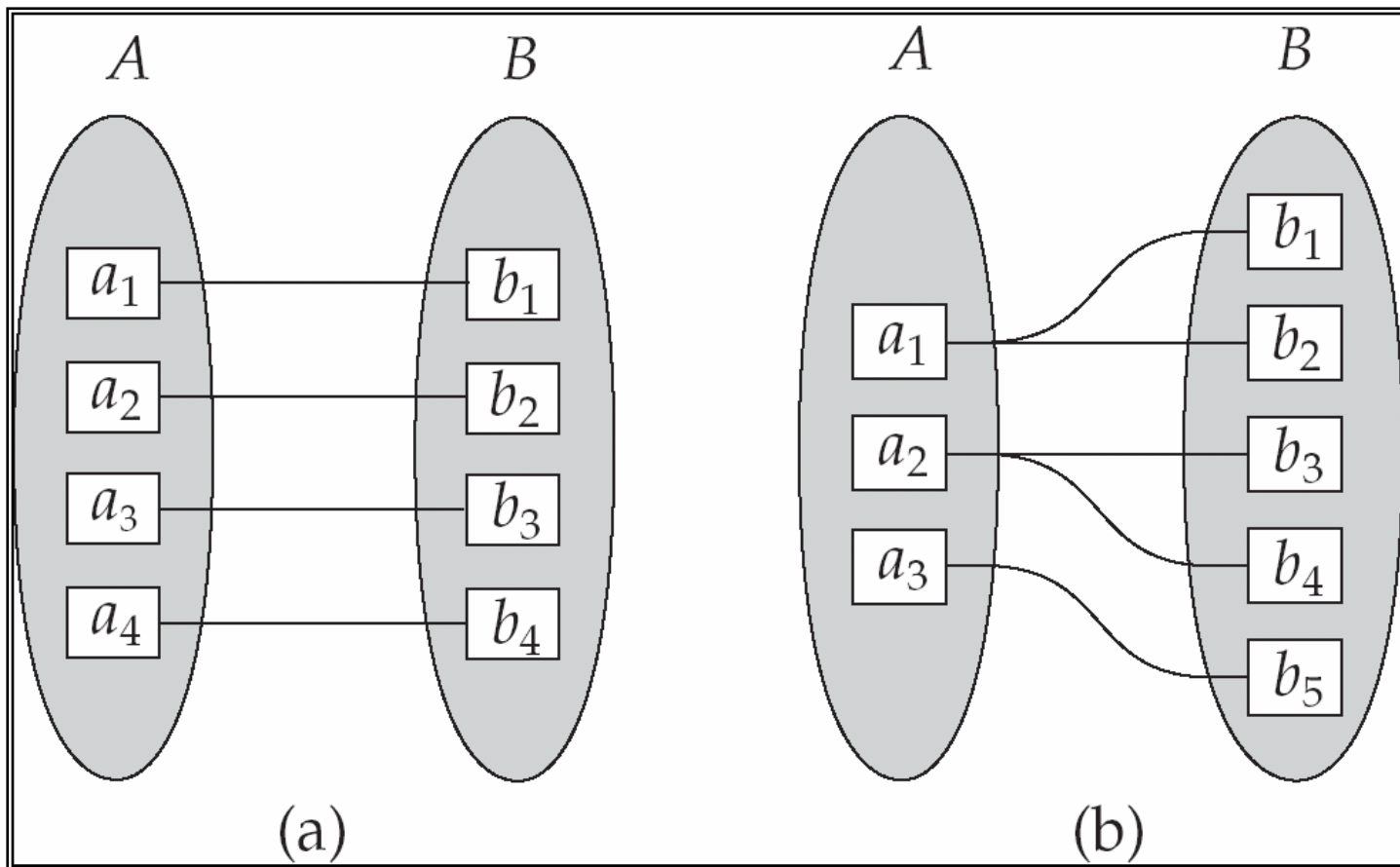


Kardinalnost mapiranja (1)

- Izražava broj entiteta koji se mogu dovesti u vezu sa drugim entitetom.
- Najupotrebljivija za opisivanje binarnog skupa veza.
- Za binarni skup veza kardinalnost mapiranja može biti:
 - Jedan-u-jedan (One to one)
 - Jedan-u-više (One to many)
 - Više-u-jedan (Many to one)
 - Više-u-više (Many to many)



Kardinalnost mapiranja (2)



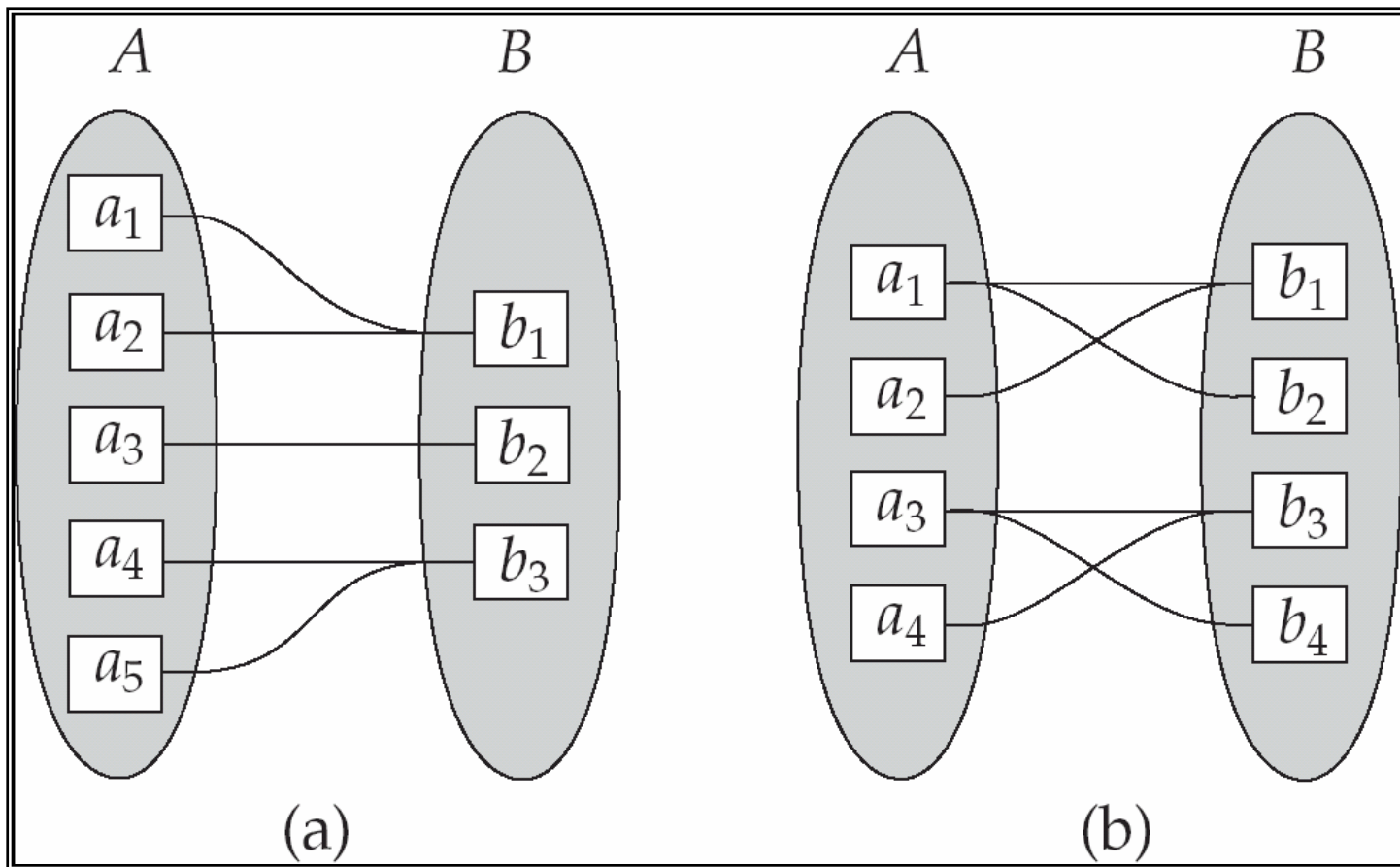
Jedan-u-jedan

Jedan-u-više

Primedba: Neki elementi u A i B ne moraju biti mapirani ni na jedan element drugog skupa



Kardinalnost mapiranja (3)



Više-u-jedan

Više-u-više

Primedba: Neki elementi u A i B ne moraju biti mapirani ni na jedan element drugog skupa



Ključ

- **Super ključ** (super key) skupa entiteta je skup jednog ili više atributa čije vrednosti jednoznačno određuju svaki entitet.
- **Kandidat ključ** (candidate key) skupa entiteta je minimalni super ključ
 - *Customer_id* je kandidat ključ za *customer*
 - *account_number* je kandidat ključ za *account*
- **Primarni ključ** (primary key) je izabrani kandidat ključ.

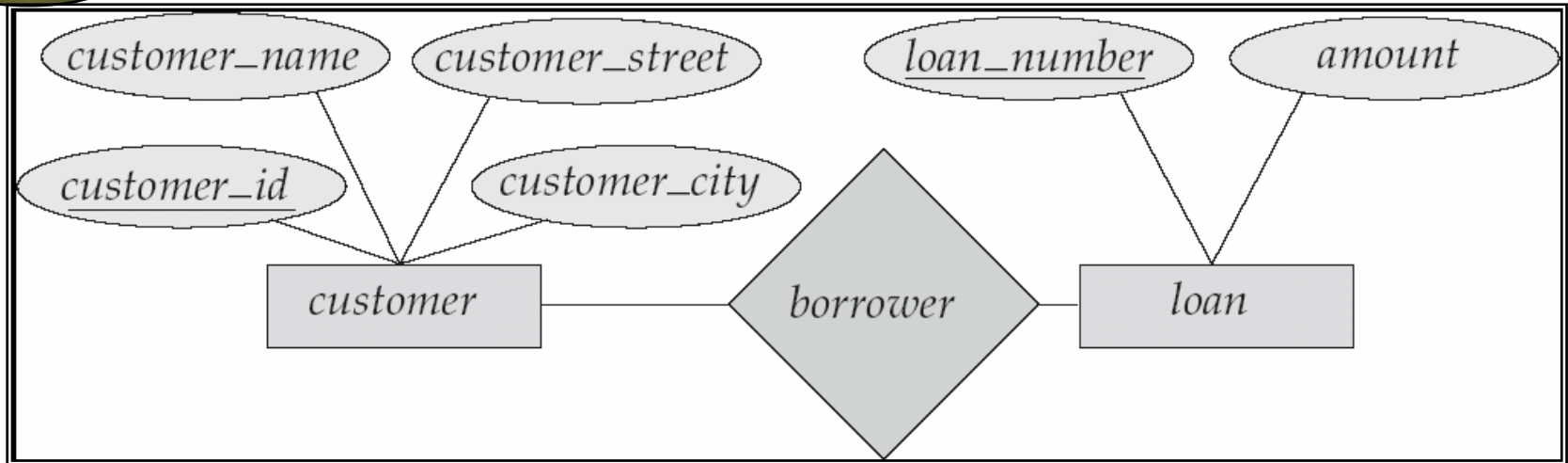


Ključ Skupa Veza

- Skup primarnih ključeva skupova entiteta u nekoj vezi je super ključ skupa veza.
 - (*customer_id*, *account_number*) je super ključ za *depositor*
 - *Primedba: to znači da par skupova entiteta može imati najviše jednu vezu u datom skupu veza.*
 - Primer: Ako želimo da pratimo sve *access_dates* za svaki *account* za svakog *customer*, ne možemo pretpostaviti vezu za svaki pristup. Možemo koristiti više-vrednosni (multivalued) atribut
- Mora se uzeti u obzir kardinalnost mapiranja skupa veza pri odluci o kandidat ključu
- Potrebno je uzeti u obzir semantiku skupa veza pri izboru *primarnog ključa* u slučaju više od jednog kandidat ključa



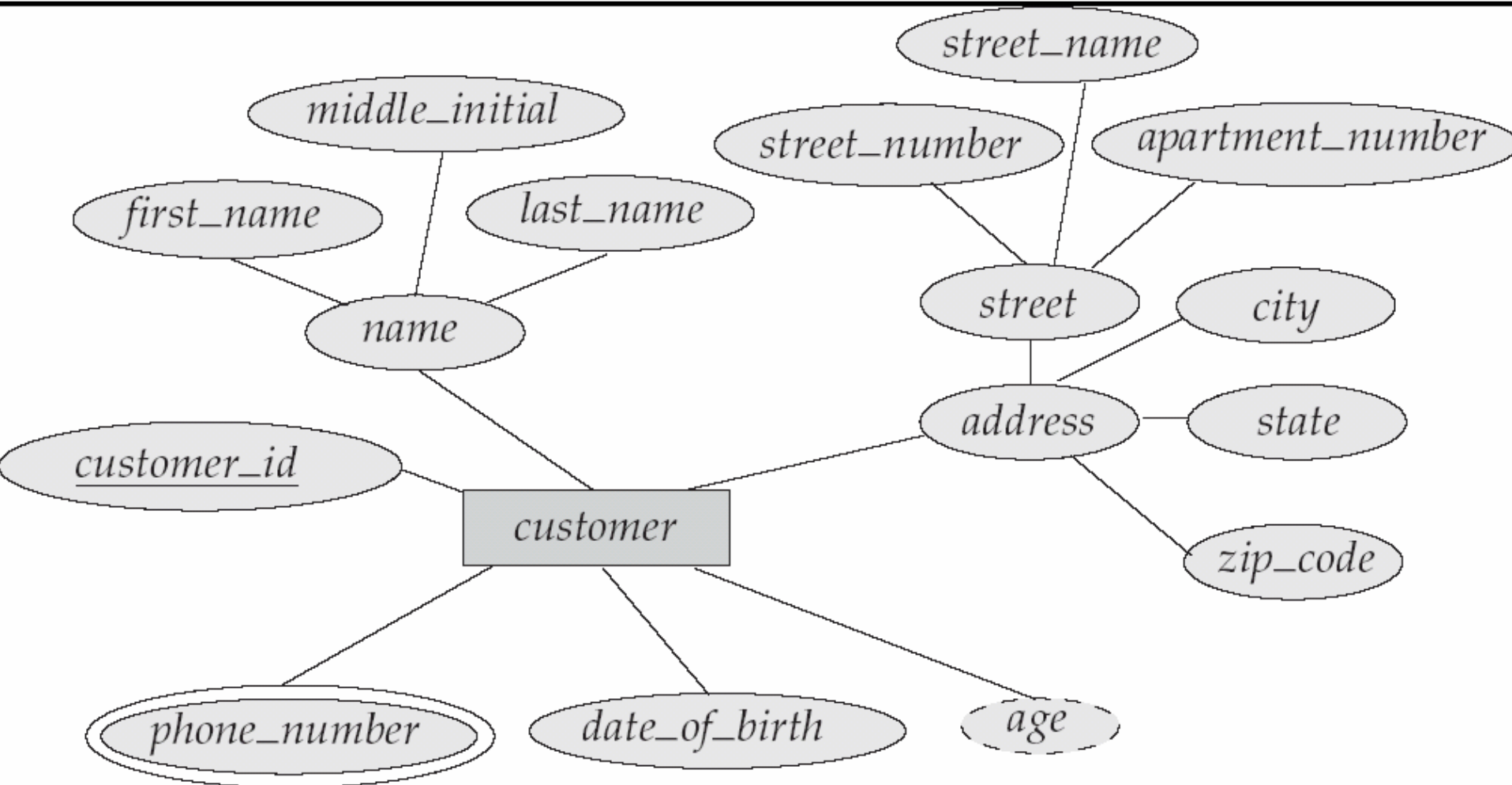
Dijagram E-V



- Pravougaonici predstavljaju skupove entiteta.
- Rombovi predstavljaju skupove veza.
- Linije povezuju attribute sa skupovima entiteta i skupove entiteta sa skupovima veza.
- Elipse predstavljaju attribute
 - Dvostruke elipse predstavljaju više-vrednosne attribute.
 - Isprekidane elipse predstavljaju sračunljive attribute.
- Podvučeno ime označava attribute primarnog ključa

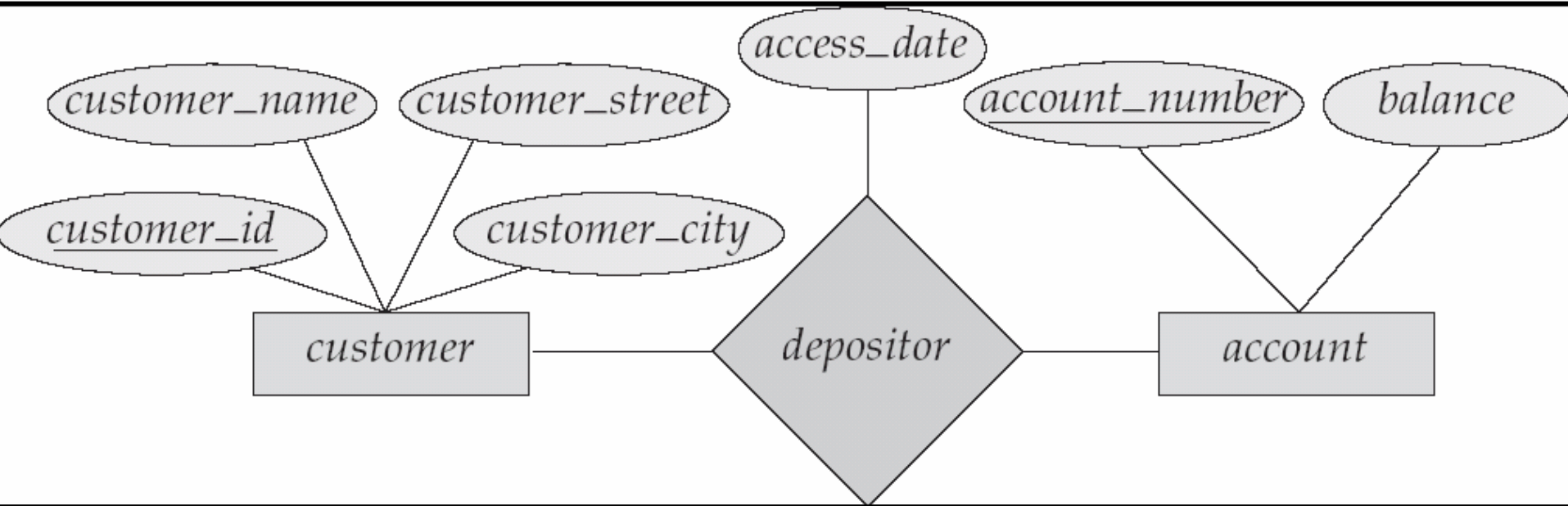


Dijagram E-V sa Kompozitnim, Više-vrednosnim, i Sračunljivim Atributima





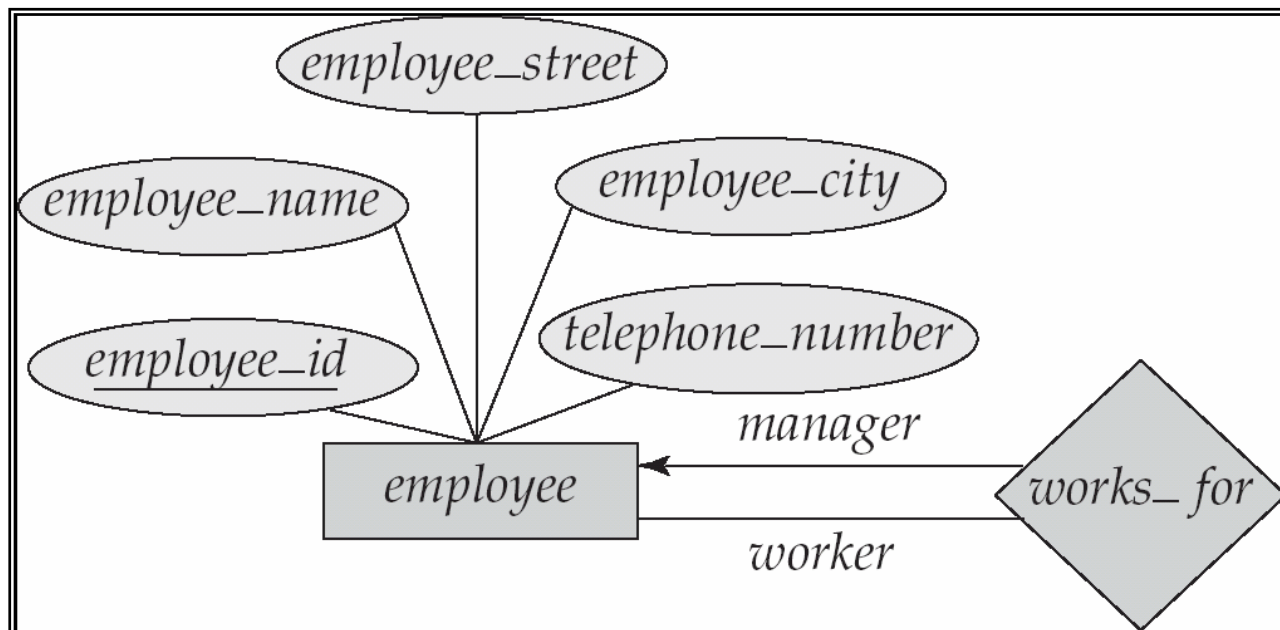
Atributi Skupa Veza





Uloge

- Skupovi entiteta u vezi ne moraju biti isključivi.
- Labele "manager" i "worker" se nazivaju **Uloge**; specificiraju kako entiteti *employee* učestvuju u skupu veza *works_for*.
- Uloge se označavaju labelama linija koje povezuju skupove veza i entiteta.
- Uloge su obično implicitne, specificiraju se samo ako nije očigledna semantika veze.





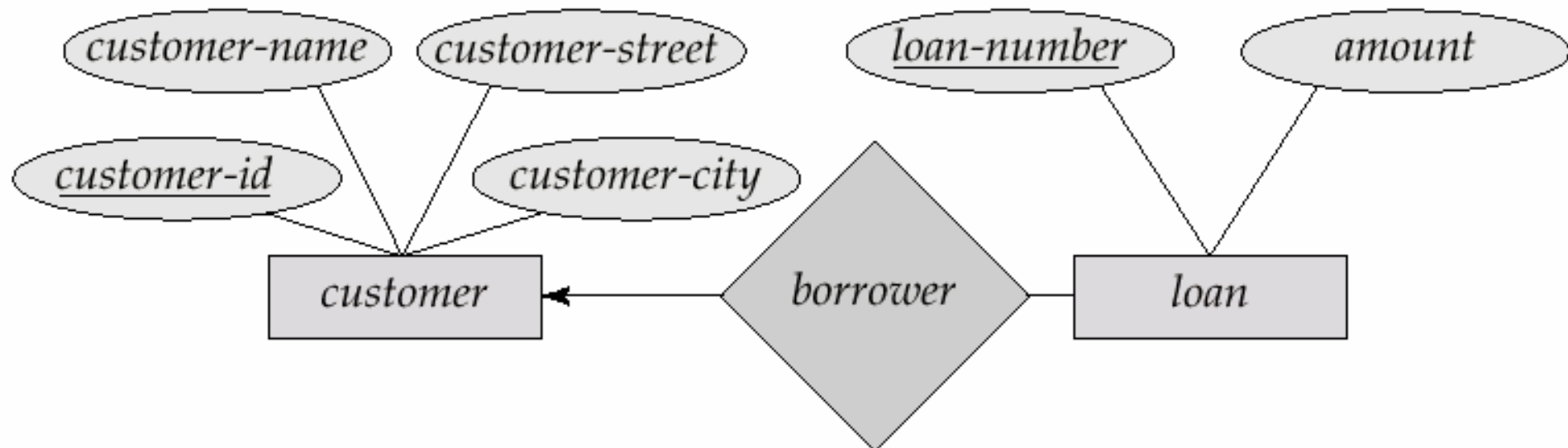
Izražavanje Kardinalnosti Mapiranja

- Kardinalnost mapiranja se izražava usmerenom linijom (\rightarrow), označavajući “jedan,” ili neusmerenom linijom ($—$), označavajući “više”.
- Veza **jedan-u-jedan**:
 - *customer* je povezan sa najviše jednim *loan* vezom *borrower*
 - *loan* je povezan sa najviše jednim *customer* vezom *borrower*



Veza Jedan-u-Više

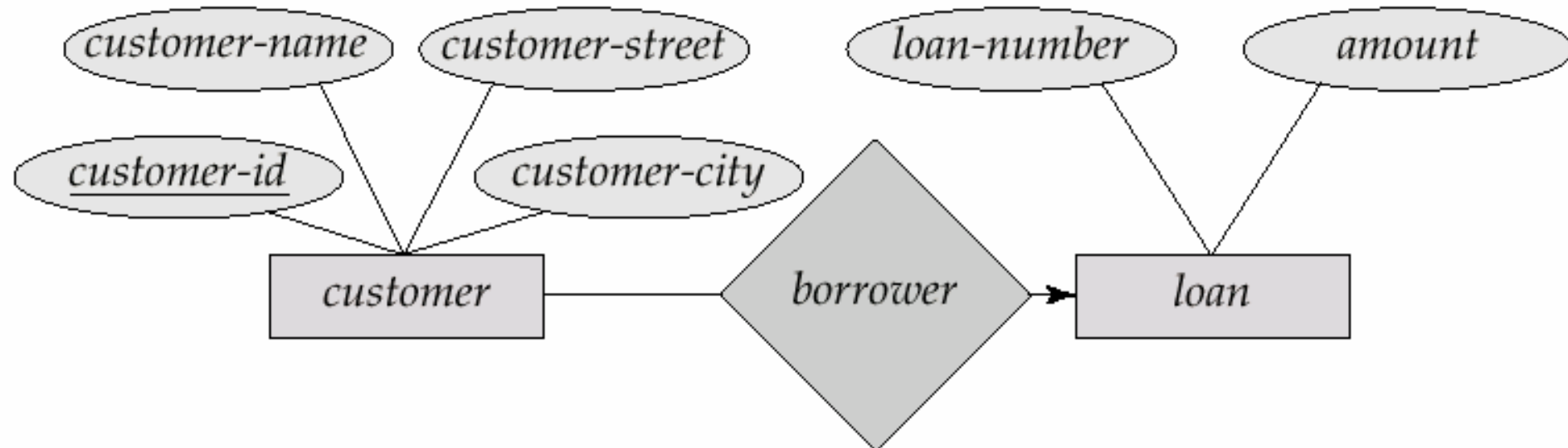
- U vezi **jedan-u-više** *loan* je povezan sa najviše jednim *customer* vezom *borrower*, a *customer* je povezan sa više (uključujući 0) *loans* vezom *borrower*





Veza Više-u-Jedan

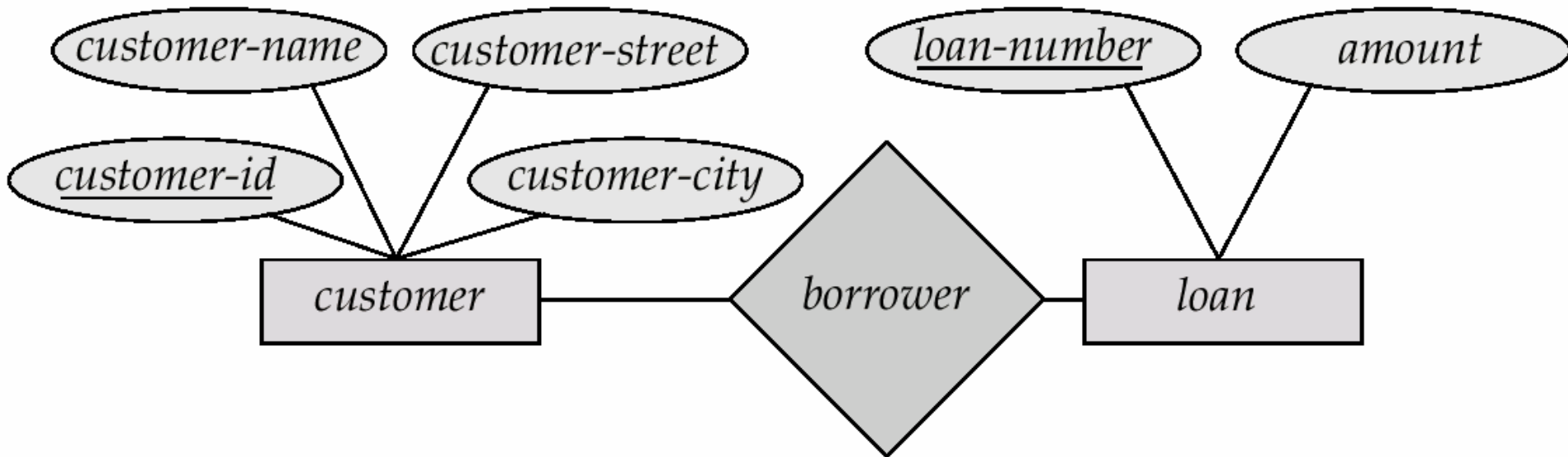
- U vezi **više-u-jedan** *loan* je povezan sa više (uključujući 0) *customers* vezom *borrower*, a *customer* je povezan sa najviše jednim *loan* vezom *borrower*





Veza Više-u-Više

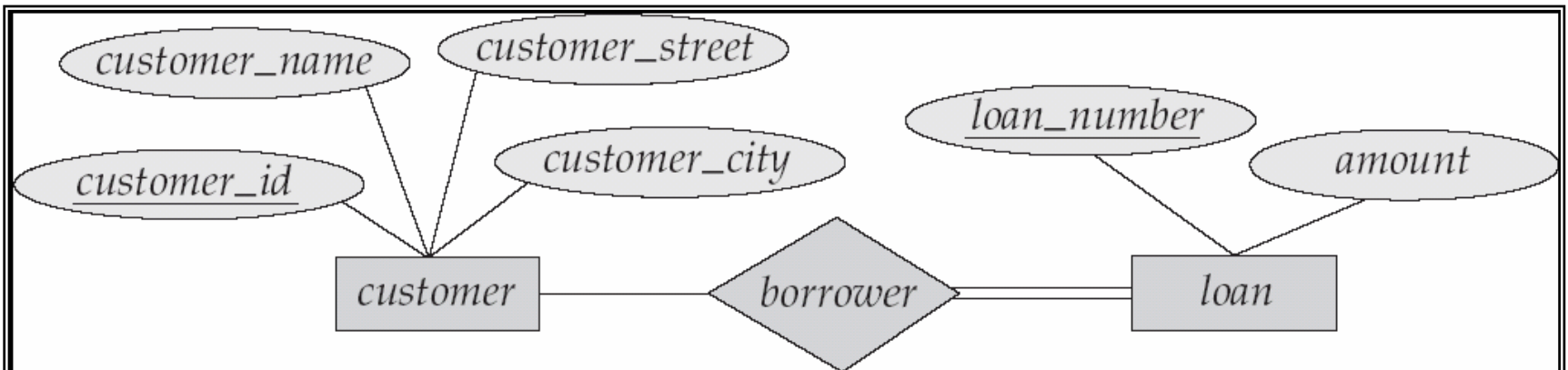
- *customer* je povezan sa više (uključujući 0) *loans* vezom *borrower*
- *loan* je povezan sa više (uključujući 0) *customers* vezom *borrower*





Učešće Skupa Entiteta u Skupu Veza

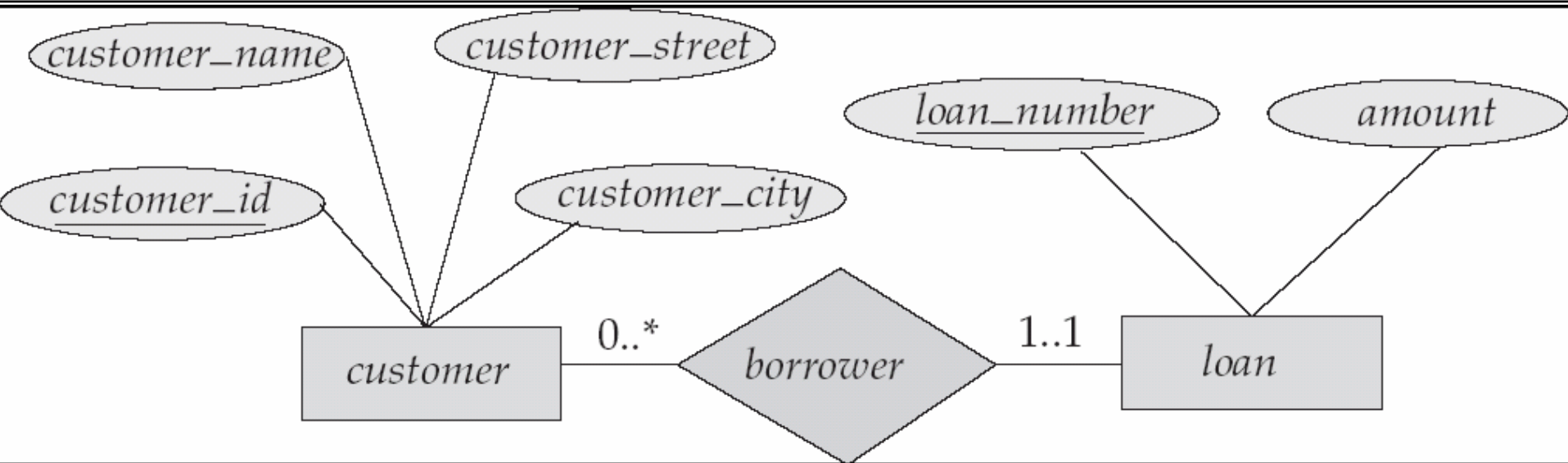
- **Totalno učešće** (predstavljeno duplom linijom): svaki entitet u skupu entiteta učestvuje u najmanje jednoj vezi u skupu veza
 - Na primer učešće *loan* u *borrower* je totalno
 - ▶ svaki *loan* mora imati *customer* povezanog vezom *borrower*
- **Parcijalno učešće**: neki entiteti ne moraju učestvovati ni u jednoj vezi u skupu veza
 - Primer: učešće *customer* u *borrower* je parcijalno





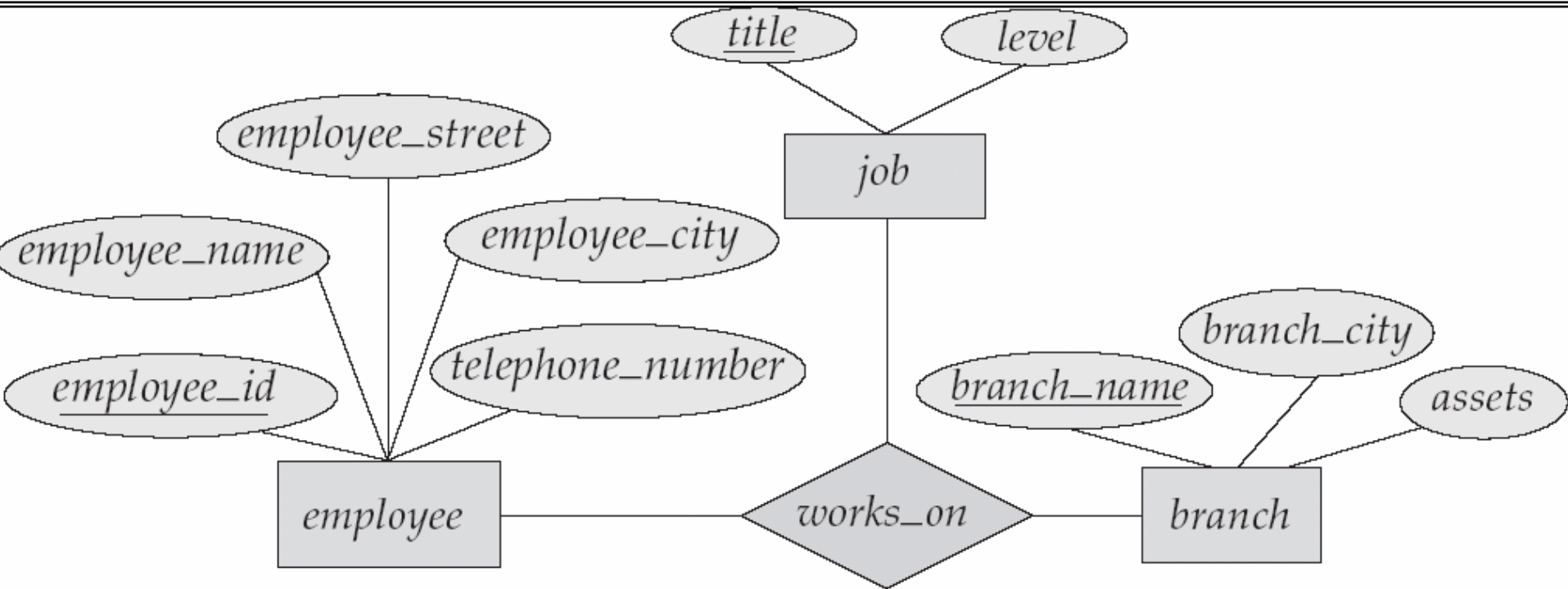
Alternativna notacija kardinalnosti veza

- Opseg vrednosti dat brojevima označava kardinalnost mapiranja (min..max)





Dijagram E-V sa ternarnom Vezom





Ograničenja kardinalnosti na ternarnim vezama

- Dozvoljena je najviše jedna strelica iz ternarne (ili većeg stepena) veze za označavanje ograničenja kardinalnosti
- Npr. strelica iz *works_on* prema *job* označava da svaki *employee* radi na najviše jednom poslu u bilo kojoj ekspozituri.
- Ako postoji više nego jedna strelica, dve su moguće interpretacije značenja.
 - Npr. ternarna veza R između A , B i C sa strelicom prema B i C može značiti
 1. svaki entitet iz A je u vezi sa jednim entitetom iz B i C ili
 2. svaki par entiteta iz (A, B) je u vezi sa jednim entitetom iz C , i svaki par (A, C) je u vezi sa jednim entitetom iz B
 - Da bi se izbegla konfuzija zabranjuje se više od jedne strelice



Odluke u Dizajnu

- **Korišćenje skupova entiteta vs. atributa**
Izbor uglavnom zavisi od strukture sistema koji modeliramo, i od semantike atributa.
- **Korišćenje skupova entiteta vs. skupova veza**
Moguće uputstvo je određivanje skupa veza za opis aktivnosti između entiteta
- **Binarni vs. n -arni skupovi veza**
Mada je moguće prevesti bilo koji nebinarni (n -arni, za $n > 2$) skup veza u više posebnih binarnih skupova veza, n -arni skup veza jasnije pokazuje da više entiteta učestvuje u jednoj vezi.
- **Korišćenje atributa skupova veza**



Binarne vs. Ne-Binarne Veze

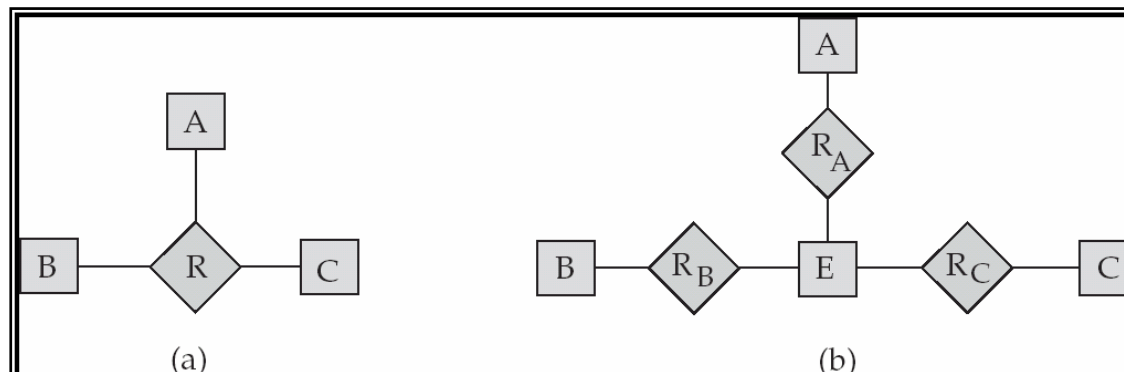
- Neke veze koje izgledaju ne-binarne mogu biti bolje predstavljene kao binarne veze
 - Npr. Ternarnu vezu *roditelji*, koja povezuje *dete* sa ocem i majkom, je bolje zameniti sa dve binarne veze, *otac* i *majka*
 - Korišćenje dve binarne veze dozvoljava parcijalnu informaciju (npr. Poznata je samo majka)
 - Ali postoje neke veze koje su prirodno ne-binarne
 - Primer: *works_on*



Konvertovanje

ne-binarnih u binarne veze (1)

- Generalno, svaka ne-binarna veza može biti predstavljena binarnim vezama kreiranjem veštačkih skupova entiteta.
 - Zameniti vezu R između skupova entiteta A , B i C sa skupom entiteta E , i tri skupa veza:
 1. R_A , koja povezuje E i A
 2. R_B , koja povezuje E i B
 3. R_C , koja povezuje E i C
 - Uvesti poseban identifikacioni atribut za E
 - Dodati attribute veze R u E
 - Za svaku vezu (a_i, b_i, c_i) u R , uvesti
 1. novi entitet e_i u skup entiteta E
 2. dodati (e_i, a_i) u R_A
 3. dodati (e_i, b_i) u R_B
 4. dodati (e_i, c_i) u R_C





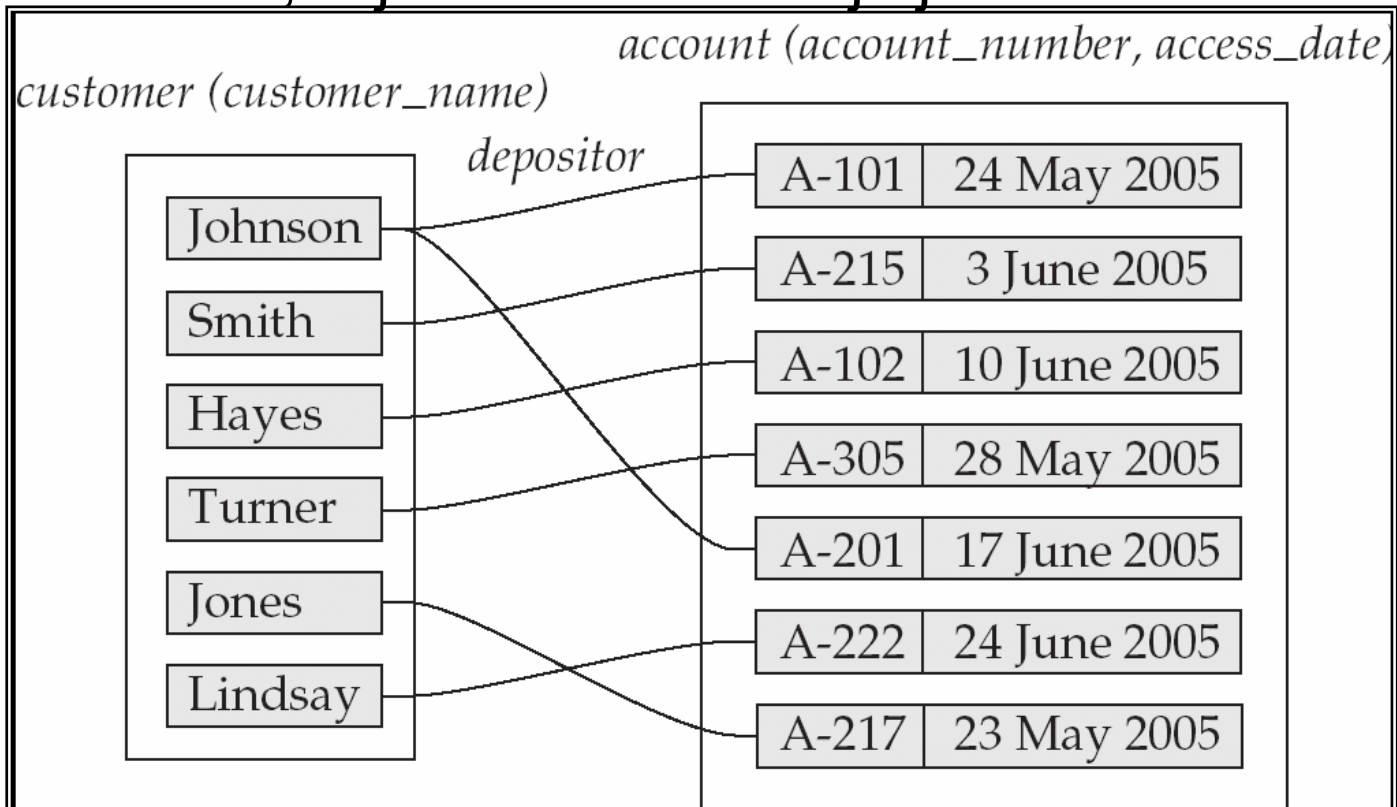
Konvertovanje ne-binarnih u binarne veze (2)

- Takođe je potrebno prevesti ograničenja
 - Prevođenje svih ograničenja nije uvek moguće
 - Može biti instanci u prevedenoj šemi koje ne odgovaraju ni jednoj instanci veze R
 - Zadatak: *dodati ograničenja u veze R_A , R_B i R_C* da bi novi entiteti odgovarali tačno jednom entitetu u svakom od skupova entiteta A , B i C
 - Može se izbeći uvođenje identifikacionog atributa pravljenjem E kao skupa slabih entiteta koji se identifikuje posredstvom tri skupa veza



Uticaj kardinalnosti mapiranja na dizajn E-V

- Moguće *access-date* uvesti kao atribut za entitet *account*, umesto kao atribut veze, ako svaki račun može pripadati samo jednom klijentu
- Odnosno, veza računa sa klijentom je više u jedan, ili equivalently, klijent sa računom je jedan u više





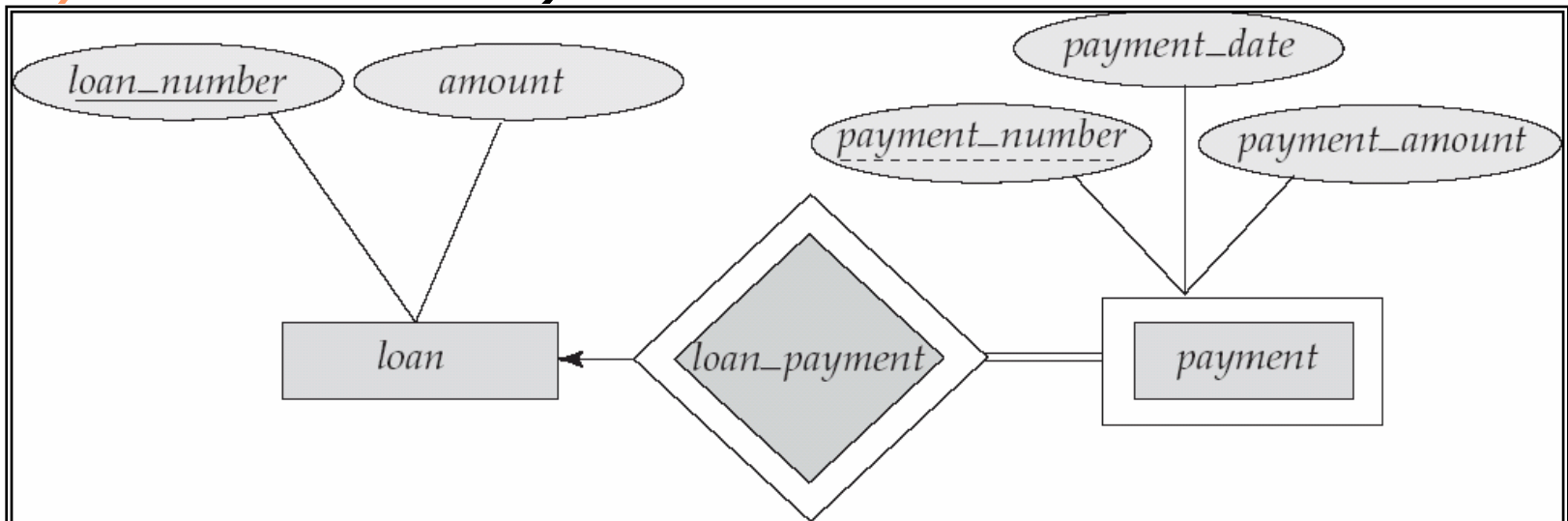
Skupovi Slabih Entiteta (1)

- Skup entiteta koji nema primarni ključ se označava kao **Skup Slabih Entiteta** (*Weak Entity Set*).
- Egzistencija skupa slabih entiteta zavisi od postojanja **identifikacionog skupa entiteta** (*identifying entity set*)
 - mora biti u vezi sa identifikacionim skupom entiteta postedstvom totalnog, jedan-u-više skupa veza od identifikacionog prema skupu slabih entiteta
 - Identifikaciona veza se označava dvostrukim rombom
- **Diskriminator** skupa slabih entiteta je skup atributa na osnovu kojih se mogu razlikovati entiteti u skupu slabih entiteta.
- **Primarni ključ** skupa slabih entiteta se formira od primarnog ključa jakog skupa entiteta od koga je skup slabih entiteta egzistencijalno zavistan, plus diskriminator skupa slabih entiteta.



Skupovi Slabih Entiteta (2)

- Skup slabih entiteta se označava dvostrukim pravougaonikom.
- Diskriminator skupa slabih entiteta se označava isprekidanom linijom.
- *payment_number* – diskriminator skupa entiteta *payment*
- Primarni ključ za *payment* – (*loan_number*, *payment_number*)





Skupovi Slabih Entiteta (3)

- Primedba: Primarni ključ skupa jakih entiteta se eksplicitno ne nalazi u skupu slabih entiteta, već je dat implicitno posredstvom identifikacione veze.
- Npr., ako je *loan_number* eksplicitno dat, *payment* može postati jak entitet, ali je tada veza između *payment* i *loan* duplicirana implicitnom vezom definisanom posredstvom atributa *loan_number* koji je zajednički za *payment* i *loan*



Skupovi Slabih Entiteta (Primeri)

- U modelu univerziteta, *course* je jak entitet, a *course_offering* može biti modeliran kao slab entitet
- Diskriminator *course_offering* može biti *semester* (uključuje godinu) i *section_number* (ako ima više delova)
- Ako modeliramo *course_offering* kao jak entitet tada *course_number* modeliramo kao atribut.
Onda je veza sa *course* implicitna posredstvom atributa *course_number*

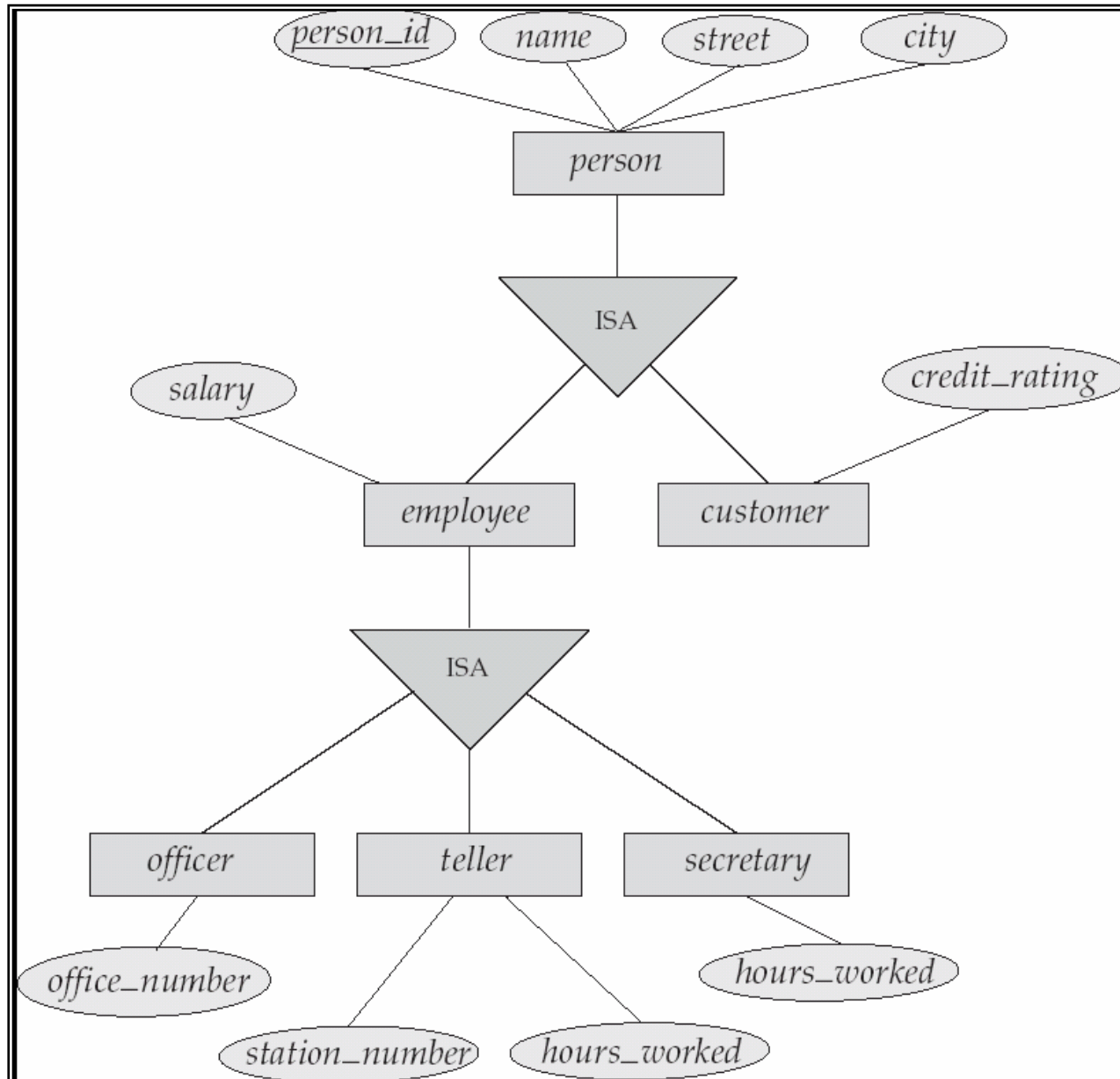


Specijalizacija

- **Top-down proces dizajna**; Delimo skup entiteta na više podskupova entiteta čime ističemo razlike koje među njima postoje.
- Podskupovi entiteta postaju skupovi entiteta nižeg nivoa u odnosu na polazni skup i imaju neke attribute ili učestvuju u nekim vezama koje nema ili u kojima ne učestvuje skup entiteta višeg nivoa.
- Označava se trouglom sa labelom **ISA** (Npr. *customer* "is a" *person*).
- **Nasleđivanje (inheritance) atributa** – skupovi entiteta nižeg nivoa nasleđuju sve attribute i veze skupa entiteta višeg nivoa sa kojim su povezani.



Specijalizacija (Primer)





Generalizacija

- **Bottom-up proces dizajna**– grupisanje više skupova entiteta koji imaju zajedničke osobine u skup entiteta višeg nivoa.
- Specijalizacija i generalizacija su inverzije jedna druge; Predstavljene su na isti način u dijagramu E-V.
- Koriste se saglasno tome da li želimo da istaknemo razlike(specijalizacija) ili sličnosti(generalizacija) koje postoje između pojedinih skupova entiteta.



Specializacija i Generalizacija

- Možemo imati više specijalizacija skupa entiteta zavisno od razlika koje želimo da istaknemo.
- Npr. *permanent_employee* vs. *temporary_employee*, i/ili *officer* vs. *secretary* vs. *teller*
- Svaki zaposleni može biti
 - član *permanent_employee* ili *temporary_employee*,
 - i takođe član *officer*, *secretary*, ili *teller*
- Veza ISA se označava i kao veza **superclass - subclass**



Ograničenja dizajna za Specializaciju i Generalizaciju (1)

- Ograničenja za uključivanje entiteta u skupove nižeg nivoa.
 - uslovom-definisana
 - Primer: svi klijenti preko 65 godina su članovi skupa entiteta *senior-citizen*; *senior-citizen* ISA *person*.
 - korisnički-definisana
- Ograničenja za određivanje da li entiteti mogu pripadati više nego jednom skupu entiteta nižeg nivoa u jednoj generalizaciji.
 - **Disjoint**
 - entitet može pripadati samo jednom skupu entiteta nižeg nivoa
 - U dijagramu E-V se označava upisivanjem *disjoint* uz ISA trougao
 - **Overlapping**
 - entitet može pripadati više nego jednom skupu entiteta nižeg nivoa



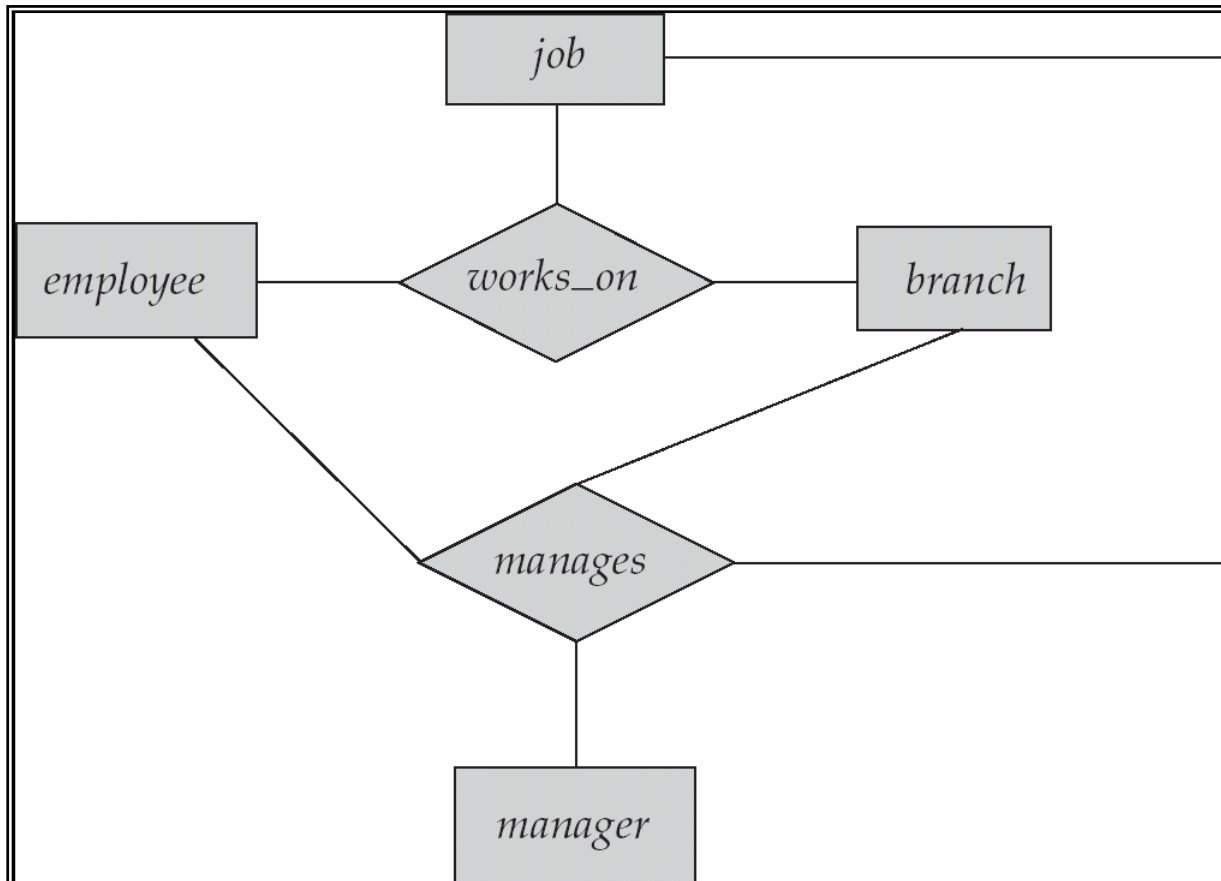
Ograničenja dizajna za Specializaciju i Generalizaciju (2)

- **Ograničenje Potpunosti** (Completeness constraint) – specificira da li entitet u skupu entiteta višeg nivoa mora pripadati najmanje jednom od skupova entiteta nižeg nivoa u generalizaciji.
 - **totalno** : entitet mora pripadati jednom od skupova entiteta nižeg nivoa (generalizacija *account* je totalna)
 - **parcijalno**: entitet ne mora pripadati ni jednom od skupova entiteta nižeg nivoa



Agregacija (1)

- Razmotrimo ranije pomenutu ternarnu vezu *works_on*
- Pretpostavimo da želimo da imamo informacije o menadžerima za poslove koje su zaposleni uradili u ekspoziturama



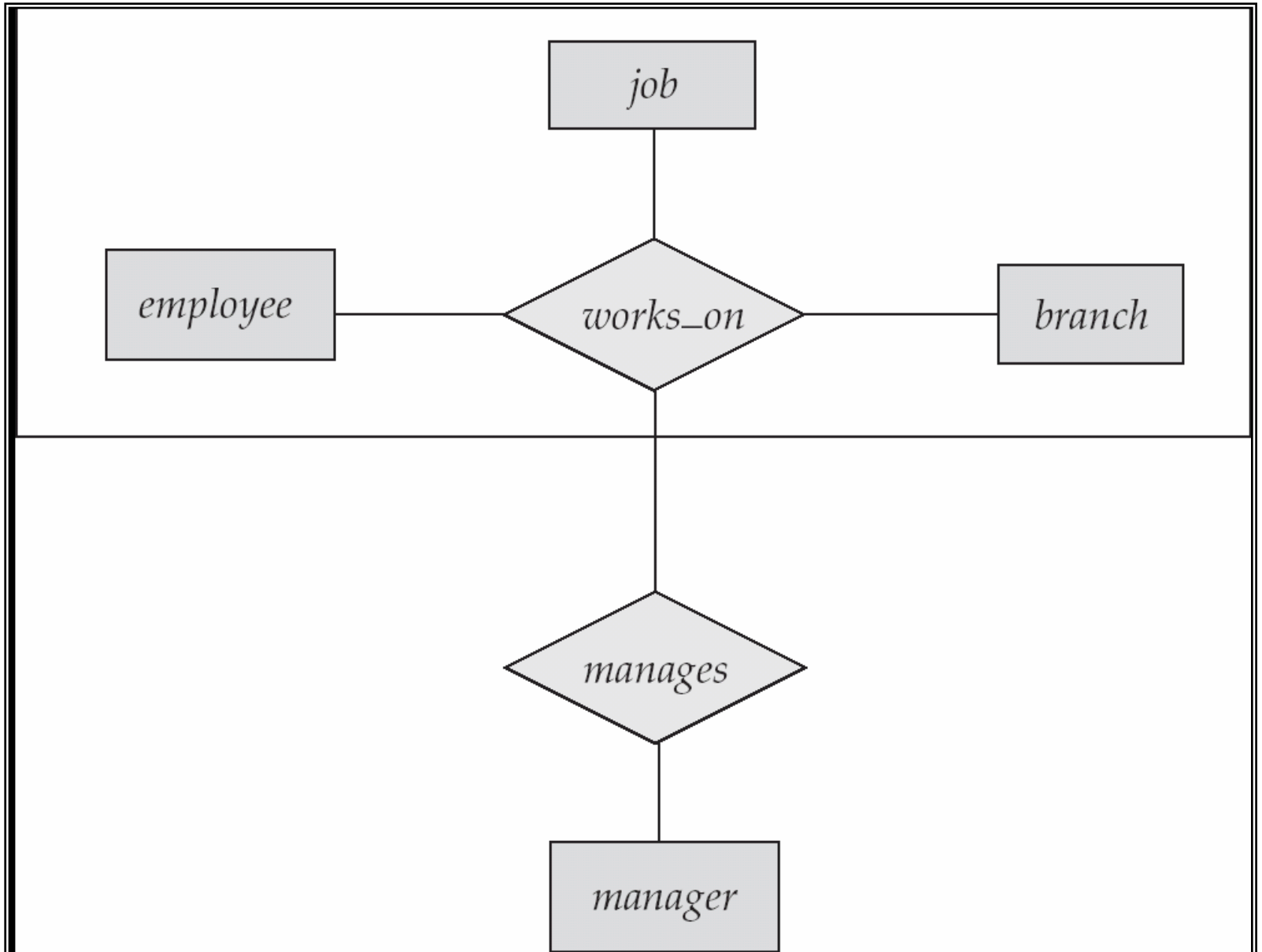


Agregacija (2)

- Skupovi veza *works_on* i *manages* se preklapaju
 - Svaka veza *manages* odgovara vezi *works_on*
 - Međutim, neke veze *works_on* ne moraju odgovarati ni jednoj vezi *manages*
 - Prema tome ne možemo izbaciti vezu *works_on*
- Ova redundantnost se može eliminisati **Agregacijom**
 - Vezu posmatramo kao abstraktni entitet višeg nivoa
 - Dozvoljava se veza između veza
 - Abstrakciju veze uključujemo u novi entitet
- Dobijamo sledeći dijagram:
 - *Zaposleni radi* neki *posao* u nekoj *ekspozituri*
 - Kombinacija *zaposleni*, *ekspozitura*, *posao* može imati pridruženog *menadžera*



Agregacija (3)





Odluke u procesu dizajna

- Korišćenje atributa ili skupa entiteta za predstavljanje objekta.
- Da li je realno okruženje bolje predstavljeno skupom entiteta ili skupom veza.
- Korišćenje ternarne veze ili para binarnih veza.
- Korišćenje skupa jakih ili slabih entiteta.
- Korišćenje specijalizacije/generalizacije – doprinosi modularnosti dizajna.
- Korišćenje agregacije – tretiranje agregiranog skupa entiteta kao celine bez razmatranja detalja njegove interne strukture.



Dijagram E-V za bankarski sistem

