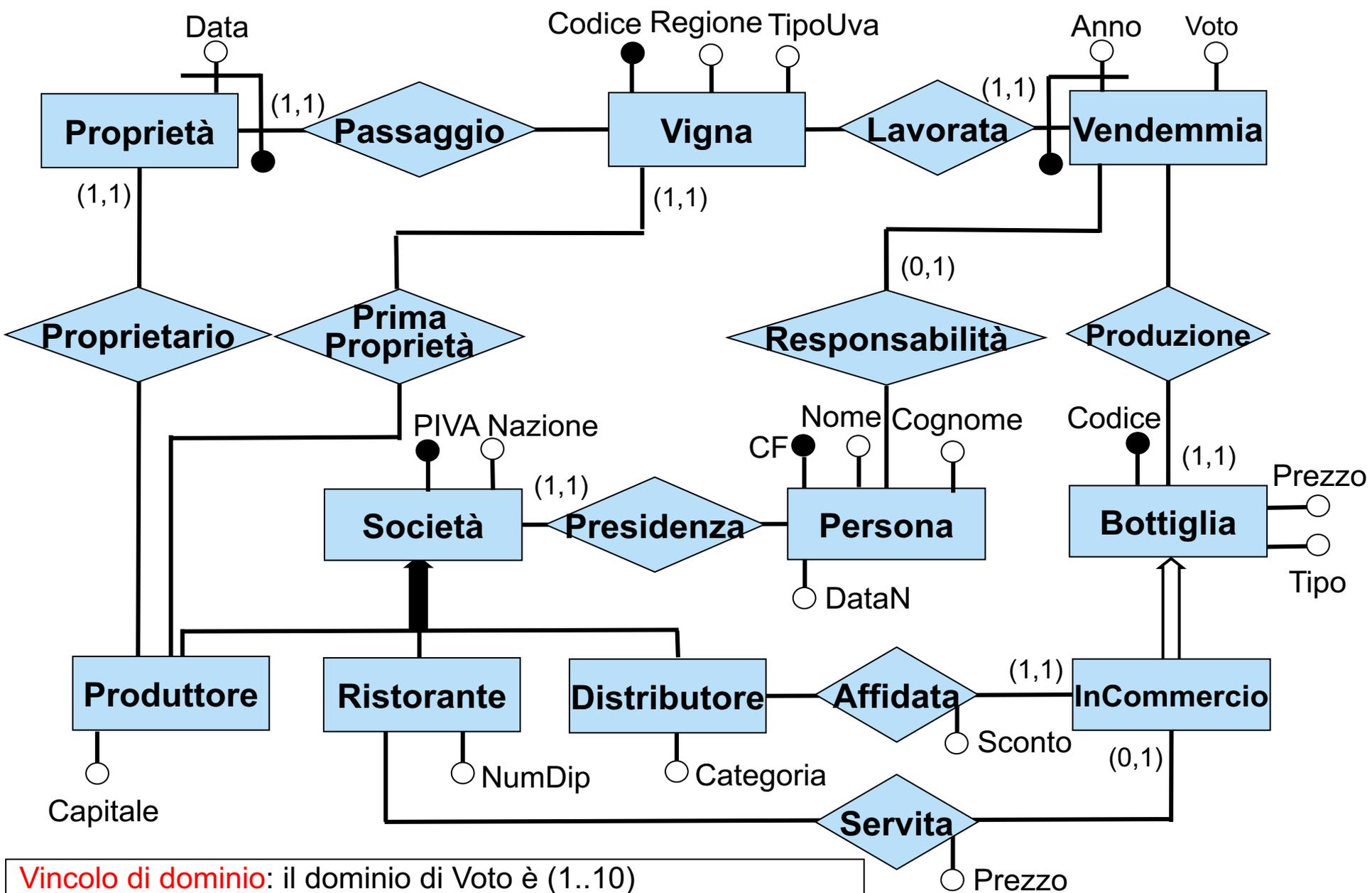


Basi di dati

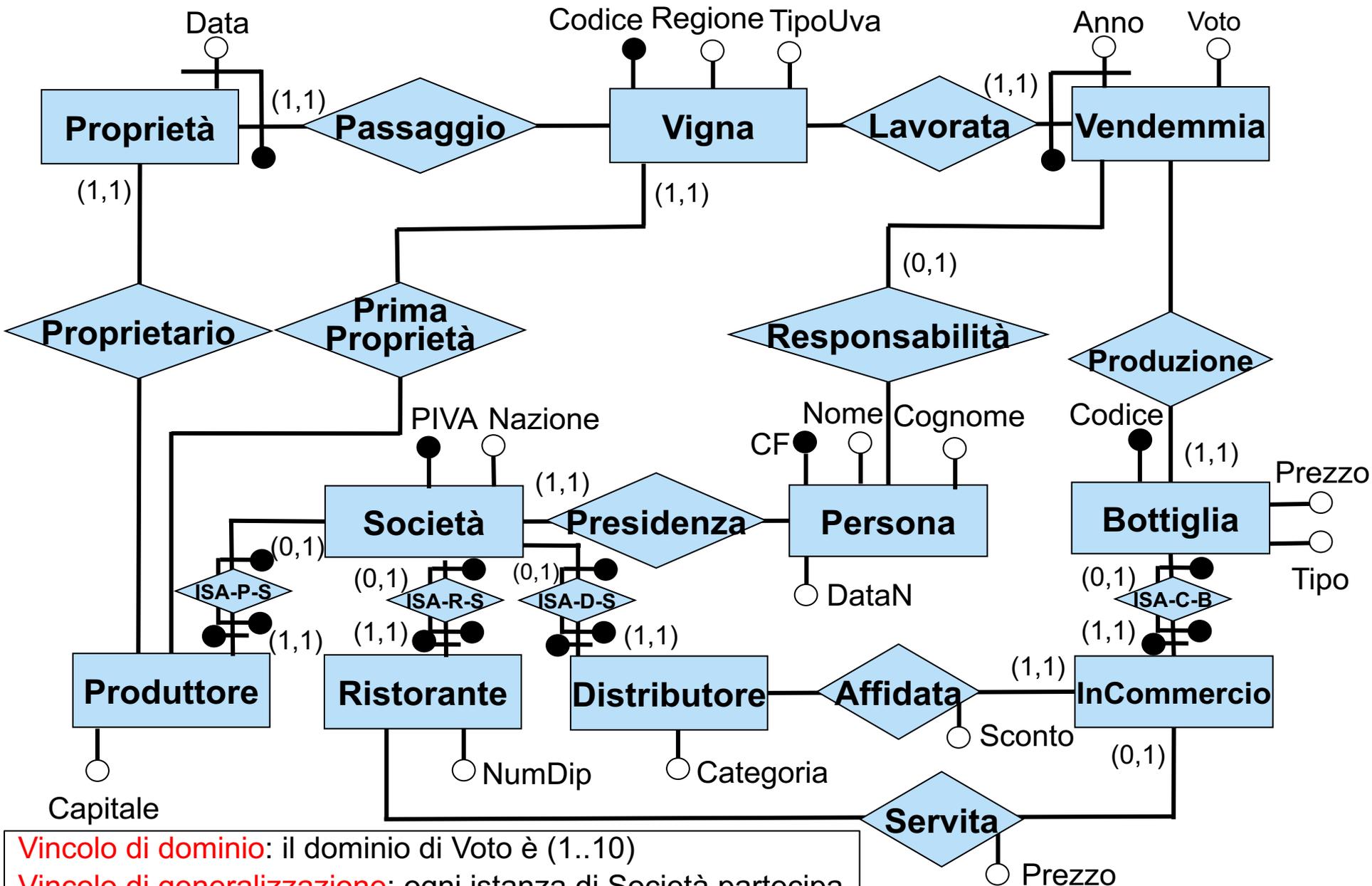
Soluzione per l'appello del 25-01-2019

Anno Accademico 2018/19

Problema 1 – Schema concettuale



Problema 1 – Schema concettuale ristrutturato



Vincolo di dominio: il dominio di Voto è (1..10)
Vincolo di generalizzazione: ogni istanza di Società partecipa ad una ed una sola tra ISA-P-S, ISA-R-S e ISA-D-S

Problema 2 - Schema logico dopo la traduzione diretta

Vigna(Codice, Regione, TipoUva)

foreign key: Vigna[Codice] \subseteq PrimaProprietà[Vigna]

PrimaProprietà(Vigna, Produttore)

foreign key: PrimaProprietà[Vigna] \subseteq Vigna[Codice]

foreign key: PrimaProprietà[Produttore] \subseteq Produttore[PIVA]

Proprietà(Vigna, Data, TipoUva)

foreign key: Proprietà[Vigna] \subseteq Vigna[Codice]

foreign key: Proprietà[Vigna,Data] \subseteq Proprietario[Vigna, Codice]

Proprietario(Vigna, Data, Produttore)

foreign key: Proprietario[Vigna,Data] \subseteq Proprietà[Vigna,Data]

foreign key: Proprietario[Produttore] \subseteq Produttore[PIVA]

Società(PIVA, Nazione)

foreign key: Società[PIVA] \subseteq Presidenza[PIVA]

vincolo di generalizzazione: Società[PIVA] \equiv Produttore[PIVA] \cup Ristorante[PIVA] \cup Distributore[PIVA]

Produttore(PIVA, Capitale)

foreign key: Produttore[PIVA] \subseteq Società[PIVA]

vincolo di generalizzazione: Produttore[PIVA] \cap Ristorante[PIVA] = \emptyset

vincolo di generalizzazione: Produttore[PIVA] \cap Distributore[PIVA] = \emptyset

Ristorante(PIVA, NumDip)

foreign key: Ristorante[PIVA] \subseteq Società[PIVA]

vincolo di generalizzazione: Ristorante[PIVA] \cap Distributore[PIVA] = \emptyset

Distributore(PIVA, Categoria)

foreign key: Distributore[PIVA] \subseteq Società[PIVA]

Presidenza(Società, Persona)

foreign key: Presidenza[Società] \subseteq Società[PIVA]

foreign key: Presidenza[Persona] \subseteq Persona[CF]

Vendemmia(Vigna, Anno, Voto)

foreign key: Vendemmia[Vigna] \subseteq Vigna[Codice]

vincolo di dominio: il dominio di Voto è (1..10)

Problema 2 - Schema logico dopo la traduzione diretta

Responsabilità(Vigna, Anno, Persona)

foreign key: Responsabile[Vigna,Anno] \subseteq Vendemmia[Vigna,Anno]

foreign key: Responsabile[Persona] \subseteq Persona[CF]

Persona(CF, Nome, Cognome, DataN)

Bottiglia(Codice, Prezzo, Tipo)

foreign key: Bottiglia[Codice] \subseteq Produzione[Bottiglia]

Produzione(Bottiglia, Vigna, Anno)

foreign key: Produzione[Bottiglia] \subseteq Bottiglia[Codice]

foreign key: Produzione[Vigna,Anno] \subseteq Vendemmia[Vigna,Anno]

InCommercio(Codice)

foreign key: InCommercio[Codice] \subseteq Bottiglia[Codice]

foreign key: InCommercio[Codice] \subseteq Affidata[Bottiglia]

Affidata(Bottiglia, Distributore, Sconto)

foreign key: Affidata[Bottiglia] \subseteq InCommercio[Codice]

foreign key: Affidata[Distributore] \subseteq Distributore[PIVA]

Servita(Bottiglia, Ristorante, Prezzo)

foreign key: Servita[Bottiglia] \subseteq InCommercio[Codice]

foreign key: Servita[Ristorante] \subseteq Ristorante[PIVA]

Problema 2 – Ristrutturazione dello schema logico

- *Indicazione 1:* quando si accede ai dati relativi ad una bottiglia interessa sempre sapere la vendemmia che l'ha prodotta, l'eventuale distributore al quale è stata affidata con il relativo sconto e l'eventuale ristorante al quale è stata venduta con il relativo prezzo.
 - Accorpamento di Bottiglia con InCommercio, Affidata e Servita, con l'aggiunta di tutti i vincoli necessari
- *Indicazione 2:* quando si accede ai dati relativi ad una società si vuole sempre sapere se è un produttore (e in questo caso si vuole sapere il capitale sociale), oppure un distributore (e in questo caso si vuole sapere la categoria), oppure un ristorante (e in questo caso si vuole sapere il numero di dipendenti).
 - Accorpamento di Società con Produttore, Distributore e Ristorante, con l'aggiunta di tutti i vincoli necessari. Si noti che anche i vincoli verso Produttore, Distributore e Ristorante vanno modificati opportunamente

Problema 2 - Schema logico dopo la ristrutturazione

Vigna(Codice, Regione, TipoUva)

foreign key: Vigna[Codice] \subseteq PrimaProprietà[Vigna]

PrimaProprietà(Vigna, Produttore)

foreign key: PrimaProprietà[Vigna] \subseteq Produttore[PIVA]

Proprietà(Vigna, Data, TipoUva)

inclusione: Proprietà[Vigna] \subseteq Vigna[Codice]

Proprietario(Vigna, Data, Produttore)

foreign key: Proprietario[Vigna,Data] \subseteq Proprietà[Vigna,Data]

inclusione: Proprietario[Produttore] \subseteq (select PIVA from Società where Tipo = 'produttore')

Società(PIVA, Nazione, Tipo, Capitale*, NumDip*, Categoria*)

foreign key: Società[PIVA] \subseteq Presiede[PIVA]

vincolo di dominio: Tipo = 'produttore' OR Tipo = 'distributore' OR Tipo = 'ristorante'

vincolo di tupla: Capitale non è null se e solo se Tipo = 'produttore'

vincolo di tupla: NumDip non è null se e solo se Tipo = 'ristorante'

vincolo di tupla: Categoria non è null se e solo se Tipo = 'distributore'

Presidenza(Società, Persona)

foreign key: Presidenza[Società] \subseteq Società[PIVA]

foreign key: Presidenza[Persona] \subseteq Persona[CF]

Vendemmia(Vigna, Anno, Voto)

foreign key: Vendemmia[Vigna] \subseteq Vigna[Codice]

vincolo di dominio: il dominio di Voto è (1..10)

Responsabilità(Vigna, Anno, Persona)

foreign key: Responsabile[Vigna,Anno] \subseteq Vendemmia[Vigna,Anno]

foreign key: Responsabile[Persona] \subseteq Persona[CF]

Persona(CF, Nome, Cognome, DataN)

Bottiglia(Codice, Prezzo, Tipo, Distributore*, Sconto*, Ristorante*, Prezzo*)

foreign key: Bottiglia[Codice] \subseteq Produzione[Bottiglia]

inclusione: Bottiglia[Distributore] \subseteq (select PIVA from Società where Tipo = 'distributore')

inclusione: Bottiglia[Ristorante] \subseteq (select PIVA from Società where Tipo = 'ristorante')

vincolo di tupla: Distributore è null se e solo se Sconto è null

vincolo di tupla: se Distributore è null allora Ristorante è null

vincolo di tupla: Ristorante è null se e solo se Prezzo è null

Produzione(Bottiglia, Vigna, Anno)

foreign key: Produzione[Bottiglia] \subseteq Bottiglia[Codice]

foreign key: Produzione[Vigna,Anno] \subseteq Vendemmia[Vigna,Anno]

Problema 2 - Schema logico dopo la ristrutturazione

Le viste per ricostruire le relazioni che sono state eliminate dalla ristrutturazione sono:

Vista Produttore(PIVA, Capitale) =

(select PIVA, Capitale from Società where Tipo = 'produttore')

Vista Ristorante(PIVA, NumDip) =

(select PIVA, NumDip from Società where Tipo = 'ristorante')

Vista Distributore(PIVA, Categoria) =

(select PIVA, Categoria from Società where Tipo = 'distributore')

Vista InCommercio(Codice) =

(select Codice from Bottiglia where Distributore is not null)

Vista Affidata(Bottiglia, Distributore, Sconto) =

(select Codice, Distributore, Sconto from Bottiglia where Distributore is not null)

Vista Servita(Bottiglia, Ristorante, Prezzo) =

(select Codice, Ristorante, Prezzo from Bottiglia where Ristorante is not null)

Problema 3: soluzione

```
select regione, nome
from citta c
where numabit > (select avg(numabit)
                 from citta
                 where regione = c.regione)
```

Problema 4: soluzione

$\text{PROJ}_{\text{regione}}(\text{citta}) - \text{PROJ}_{\text{regione}}(\text{SEL}_{\text{numabit} > 50000}(\text{citta}))$

Problema 5: soluzione

- i. Un vincolo di integrità è una regola che si esprime a livello di schema e che deve valere per tutte le basi di dati che sono istanze dello schema. Lo schema concettuale S contiene tre vincoli di cardinalità, associati rispettivamente ai ruoli C di R , C di Q ed E di R .
- ii. Esiste un'istanza I dello schema S in cui ciascuna delle entità E ed F ha esattamente una istanza. Consideriamo una istanza I dello schema S in cui sia E sia F hanno una istanza. Ovviamente, siccome F ISA E , l'oggetto che è istanza di F è esattamente lo stesso che è istanza di E . Inoltre, poiché il vincolo di cardinalità minimo sul ruolo E di Q è pari a 1, e Q ISA R , avremo che esisterà una istanza di Q a cui partecipa f nel ruolo E e questa sarà anche istanza di R . Quindi, l'istanza I è così composta: $Istanze(I, F) = \{f\}$, $Istanze(I, E) = \{f\}$, $Istanze(I, C) = \{c\}$, $Istanze(I, Q) = \{ \langle E:f, C:c \rangle \}$, $Istanze(I, R) = \{ \langle E:f, C:c \rangle \}$.
- iii. Non esiste un'istanza dello schema S in cui l'entità F ha due istanze e l'entità C ha esattamente una istanza. Infatti, se in una istanza I dello schema S , F ha due istanze, chiamiamole f_1 ed f_2 , e C ha esattamente una istanza c , allora, poiché a causa del vincolo di cardinalità minimo sul ruolo E di Q ogni istanza di F deve partecipare a Q nel ruolo E , avremo le istanze $\langle E:f_1, C:c \rangle$ e $\langle E:f_2, C:c \rangle$ in Q . Ma siccome Q ISA R , queste due coppie sono anche istanze di R e quindi la cardinalità massima 1 di C nel ruolo C di R viene violata.