

# **Basi di dati**

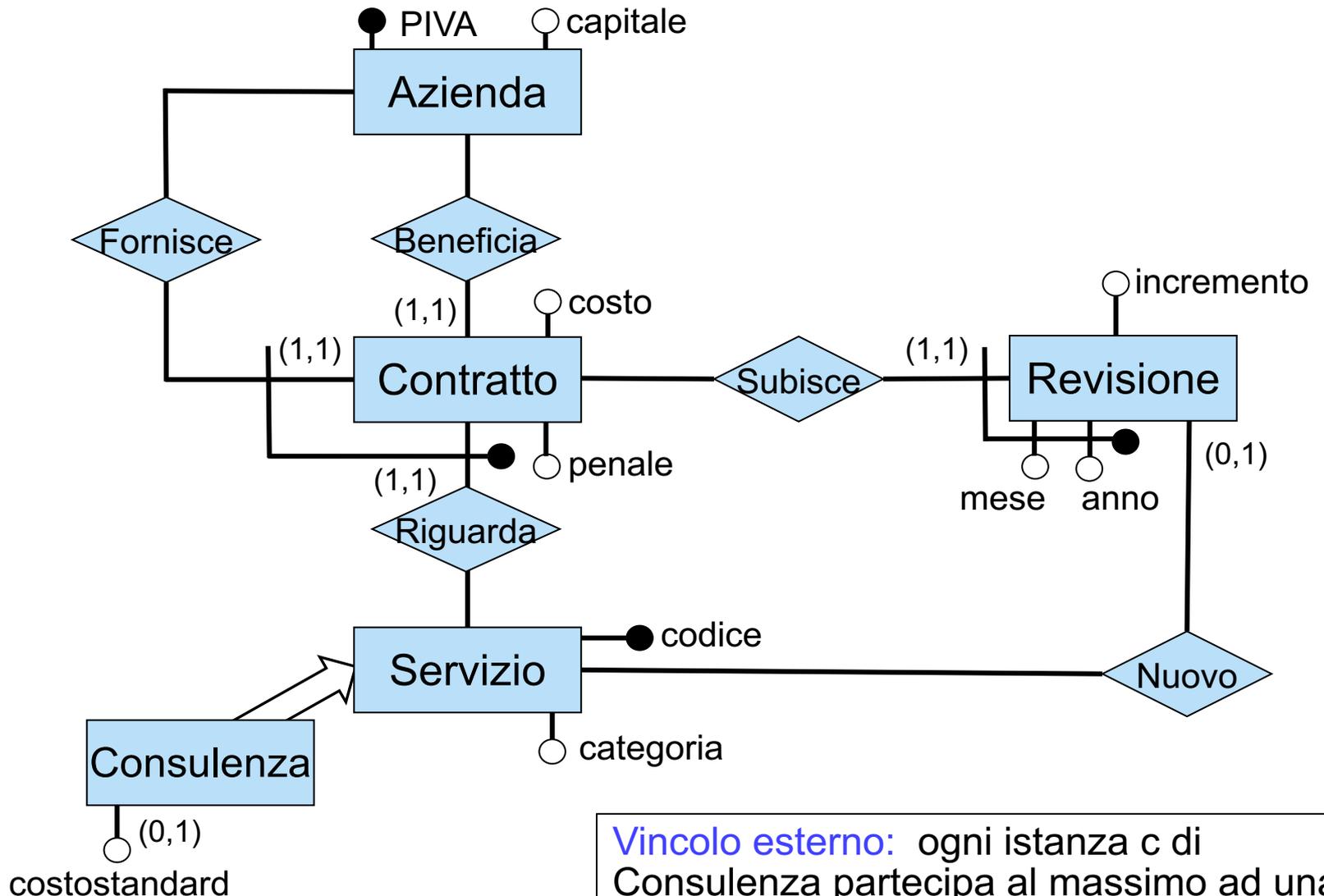
**Soluzioni dei problemi proposti  
nell'appello del 20-01-2023  
Compito A**

***Maurizio Lenzerini***

Anno Accademico 2022/23

# Problema 1 – Schema ER

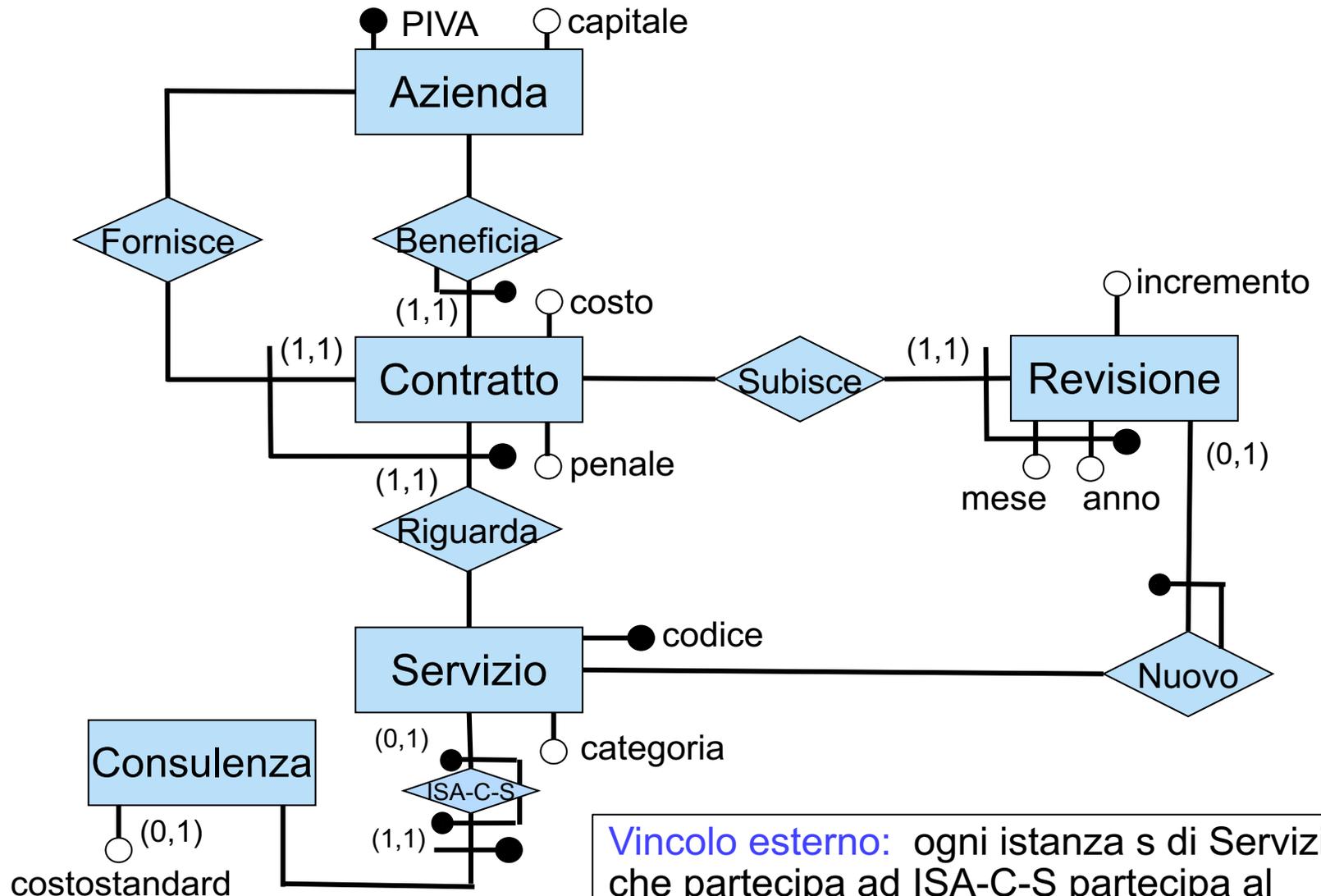
Schema concettuale:



**Vincolo esterno:** ogni istanza c di Consulenza partecipa al massimo ad una istanza di Riguarda.  
**Vincolo di dominio:** incremento > 0

# Problema 2 – Ristrutturazione Schema ER

Schema concettuale ristrutturato:



**Vincolo esterno:** ogni istanza s di Servizio che partecipa ad ISA-C-S partecipa al massimo ad una istanza di Riguarda.  
**Vincolo di dominio:** incremento > 0

## Problema 2 – Traduzione diretta

Schema logico prodotto  
dalla traduzione diretta:

Azienda(piva, capitale)

Contratto(fornitore, servizionativo, costo, penale)

**foreign key:** Contratto[fornitore]  $\subseteq$  Azienda[piva]

**foreign key:** Contratto[servizionativo]  $\subseteq$  Servizio[codice]

**foreign key:** Contratto[fornitore, servizionativo]  $\subseteq$  Beneficia[fornitorecontratto, servizionativocontratto]

**vincolo inter-relazionale:** per ogni tupla t1 di Contratto tale che t1[servizionativo] è in Consulenza[codice],  
non esiste altra tupla t2 di Contratto tale che t2[servizionativo] = t1[servizionativo]

Beneficia(fornitorecontratto, servizionativocontratto, beneficiario)

**foreign key:** Beneficia[fornitorecontratto, servizionativocontratto]  $\subseteq$  Contratto[fornitore, servizionativo]

**foreign key:** Beneficia[beneficiario]  $\subseteq$  Azienda[piva]

Servizio(codice, categoria)

Consulenza(codice, costostandard\*)

**foreign key:** Consulenza[codice]  $\subseteq$  Servizio[codice]

Revisione(fornitorecontratto, servizionativocontratto, mese, anno, incremento)

**foreign key:** Revisione[fornitorecontratto, servizionativocontratto]  $\subseteq$  Contratto[fornitore, servizionativo]

**vincolo di dominio:** incremento > 0

Nuovo(fornitorecontratto, servizionativocontratto, mese, anno, servizio)

**foreign key:** Nuovo[fornitorecontratto, servizionativocontratto, mese, anno]  $\subseteq$

Revisione[fornitorecontratto, servizionativocontratto, mese, anno]

**foreign key:** Nuovo[servizio]  $\subseteq$  Servizio[codice]

# Problema 2 – Ristrutturazione dello schema logico

## Schema logico prodotto dalla ristrutturazione:

1. La prima indicazione di progetto induce un accorpamento tra Contratto e Beneficia, fortemente accoppiate.
2. Sempre la prima indicazione induce anche un accorpamento tra Revisione e Nuovo, debolmente accoppiate.
3. La seconda indicazione induce un accorpamento tra Servizio e Consulenza, debolmente accoppiate.

Azienda(piva,capitale)

Contratto(fornitore,servizionativo,costo,penale,beneficiario)

**foreign key:** Contratto[fornitore]  $\subseteq$  Azienda[piva]

**foreign key:** Contratto[servizionativo]  $\subseteq$  Servizio[codice]

**foreign key:** Contratto[beneficiario]  $\subseteq$  Azienda[piva]

**vincolo inter-relazionale:** per ogni tupla t1 di Contratto tale che esiste una tupla t2 di Servizio con  
t2[codice] = t1[servizionativo] e t2[flagconsulenza] = true, non esiste altra tupla  
t3 di Contratto tale che t2[servizionativo] = t1[servizionativo]

Servizio(codice,categoria,flagconsulenza,costostandard\*)

Revisione(fornitorecontratto,servizionativocontratto,mese,anno,incremento,nuovoservizio\*)

**foreign key:** Revisione[fornitorecontratto,servizionativocontratto]  $\subseteq$  Contratto[fornitore,servizionativo]

**foreign key:** Revisione[nuovoservizio]  $\subseteq$  Servizio[codice]

**vincolo di dominio:** incremento > 0

Definiamo anche le viste per ricostruire le relazioni accorpate:

View Beneficia(f,s,b) = select fornitore, servizionativo,beneficiario from Contratto

View Consulenza(c,s) = select codice,costostandard from Servizio where flagconsulenza

View Nuovo(f,s,m,a,n) = select fornitorecontratto,servizionativocontratto,mese,anno,nuovoservizio  
from Revisione

## Problema 3 – Ulteriori indicazioni di progetto

1. La prima indicazione induce la seguente definizione del vincolo di foreign key da Revisione a Contratto:

```
foreign key (fornitorecontratto, servizionativocontratto) references Contratto on delete cascade
```

2. La seconda indicazione induce la definizione di questo trigger:

```
create or replace function blocca_update_contratto() returns trigger as  
$$ BEGIN RETURN NULL; END;  
$$ language plpgsql;
```

```
create trigger trigger_update_contratto before update on Contratto  
for each row execute procedure blocca_update_contratto();
```

3. La terza indicazione induce la definizione di questo trigger:

```
create or replace function inserimento_revisione() returns trigger as  
$$ BEGIN
```

```
    IF (select flagconsulenza from Servizio where codice = NEW.servizionativocontratto)  
        and not
```

```
        (select flagconsulenza from Servizio where codice = NEW.nuovoservizio)
```

```
    THEN RETURN NULL;
```

```
    ELSE RETURN NEW;
```

```
    END IF;
```

```
END;
```

```
$$ language plpgsql;
```

```
create trigger trigger_inserimento_revisione before insert on Revisione  
for each row execute procedure inserimento_revisione();
```

# Problema 4 – testo e soluzione

**Testo:** Riferendosi allo schema logico prodotto per il problema 2, scrivere una query SQL che per ogni contratto C restituisca la partita IVA dell'azienda fornitrice di C, il codice del servizio nativo di C, la categoria del servizio nativo di C ed il numero delle revisioni subite da C che hanno cambiato il servizio oggetto del contratto.

## Soluzione:

```
select c.fornitore, c.servizionativo, s.categoria, 0
from Contratto c join Servizio s on c.servizionativo = s.codice
where (c.fornitore,c.servizionativo) not in
      (select r.fornitorecontratto, r.servizionativocontratto
       from Revisione r
       where r.nuovoservizio is not null)
union
select c.fornitore, c.servizionativo, s.categoria, count(*)
from Contratto c join Servizio s on c.servizionativo = s.codice
where (c.fornitore,c.servizionativo) in
      (select r.fornitorecontratto, r.servizionativocontratto
       from Revisione r
       where r.nuovoservizio is not null)
group by c.fornitore, c.servizionativo, s.categoria
```

# Problema 5 – soluzione

## Prima parte

Relativamente alla prima parte del problema 5, ricordiamo solo la definizione di **superchiave** per lo schema di relazione R. Sia S un insieme non vuoto degli attributi di R.

1. Se r è una relazione coerente con R (ossia che soddisfa tutti i vincoli di R), S soddisfa la **condizione di superchiave in r** se non esistono due tuple in r che coincidono negli attributi S.
2. S è **una superchiave per R** se per ogni relazione r coerente con R, S soddisfa la condizione di superchiave in r.

## Seconda parte

Per la seconda parte dobbiamo individuare tutte le superchiavi per R, dove R è definita così:

$R(\underline{A}, B, \underline{C}, D)$

vincolo intra-relazionale: non esistono  $t_1, t_2 \in R$  tali che  $t_1[D]=t_2[D]$  e  $t_1[C] \neq t_2[C]$

Siccome  $\{A, C\}$  è la chiave primaria per S, le seguenti sono chiaramente superchiavi di R:  $\{A, C\}$ ,  $\{A, C, B\}$ ,  $\{A, C, D\}$ ,  $\{A, C, B, D\}$ .

Il problema è adesso verificare se ci sono altre superchiavi per R. Notiamo che il vincolo intra-relazionale si può esprimere così: **per qualunque coppia di tuple  $t_1, t_2 \in R$ , se  $t_1[D]=t_2[D]$ , allora  $t_1[C]=t_2[C]$** . Da questo si evince immediatamente che D si comporta come C rispetto alla condizione di superchiave. In particolare, si può verificare che  $\{A, D\}$  è una superchiave (e anche una chiave) per R semplicemente notando che se  $\{A, D\}$  non fosse una superchiave, allora esisterebbe una relazione r coerente con R che contiene due tuple  $t_1, t_2$  diverse tra loro tali che  $t_1[A] = t_2[A]$  e  $t_1[D] = t_2[D]$ . Ma questo implicherebbe  $t_1[A] = t_2[A]$  e  $t_1[C] = t_2[C]$ , contraddicendo il fatto che  $\{A, C\}$  rispetta la condizione di superchiave in r. Possiamo quindi concludere che anche le seguenti sono superchiavi di R, visto che contengono la chiave  $\{A, D\}$ :  $\{A, D\}$ ,  $\{A, D, B\}$ .

# Problema 5 – soluzione

*Terza parte (che nella correzione degli esami è stata considerata facoltativa)*

Dobbiamo infine motivare il perché non vi sono altre superchiavi per R. Notiamo che l'insieme di superchiavi che abbiamo individuato è

$\{A,C\}, \{A,C,B\}, \{A,C,D\}, \{A,C,B,D\}, \{A,D\}, \{A,D,B\}$

il che significa che è una superchiave per R ogni insieme di attributi di R che contiene A e C, oppure contiene A e D.

Consideriamo un qualunque insieme S di attributi di R che non contiene entrambi A e C e non contiene entrambi A e D. Se S è vuoto, non è una superchiave per definizione. Altrimenti, la ragione per cui S non è una superchiave per R è che possiamo definire una relazione r coerente con R che contiene due tuple diverse tra loro  $t_1, t_2$  che coincidono negli attributi in S, dimostrando che S non soddisfa la condizione di superchiave in r (il che appunto implica che S non è superchiave per R).

Per scegliere  $t_1$  e  $t_2$  distinguiamo due casi.

- 1. S contiene D** (e quindi non contiene A). Scegliamo un valore V e per ogni attributo  $x \in S$  poniamo  $t_1[x] = t_2[x] = V$  e anche  $t_1[C] = t_2[C]$ . Scegliamo due valori  $W_1$  e  $W_2$  diversi tra loro e per ogni attributo  $y \notin S$  diverso da C poniamo  $t_1[y] = W_1$  e  $t_2[y] = W_2$ . È facile verificare che abbiamo  $t_1[A] \neq t_2[A]$ ,  $t_1[D] = t_2[D]$  e  $t_1[C] = t_2[C]$  e quindi r soddisfa sia la condizione di superchiave per  $\{A,C\}$ , sia il vincolo intra-relazionale.
- 2. S non contiene D**. Scegliamo ancora un valore V e per ogni attributo  $x \in S$  poniamo  $t_1[x] = t_2[x] = V$ . Scegliamo due valori  $W_1$  e  $W_2$  diversi tra loro e per ogni attributo  $y \notin S$  poniamo  $t_1[y] = W_1$  e  $t_2[y] = W_2$ . Siccome  $D \notin S$ , abbiamo  $t_1[D] \neq t_2[D]$  e quindi r soddisfa il vincolo intra-relazionale. Per quanto riguarda la condizione di superchiave per  $\{A,C\}$ , valgono le seguenti osservazioni: se  $C \notin S$ , allora abbiamo  $t_1[C] \neq t_2[C]$ , altrimenti  $C \in S$  e allora  $A \notin S$  e quindi abbiamo  $t_1[A] \neq t_2[A]$ . In entrambi i casi r soddisfa la condizione di superchiave per  $\{A,C\}$ .