



Basi di dati

Maurizio Lenzerini

***Dipartimento di Informatica e Sistemistica “Antonio Ruberti”
Università di Roma “La Sapienza”***

Anno Accademico 2024/2025

<http://www.dis.uniroma1.it/~lenzerin/home/?q=node/44>



5. La progettazione concettuale

5.1 introduzione alla progettazione di basi di dati

1. **introduzione alla
progettazione di basi di dati**
2. modello ER
3. metodologia per la
progettazione concettuale
4. esempio



Inquadramento del progetto di basi di dati

- Il progetto di basi di dati ha lo scopo di produrre lo schema logico e lo schema fisico della basi di dati, secondo le indicazioni raccolte durante la fase di raccolta dei requisiti
- Il progetto di basi di dati viene svolto **nell'ambito** della più ampia attività di progetto del sistema informatico



Fasi del ciclo di vita di un'applicazione software

1. Studio di fattibilità e raccolta dei requisiti

- valutare costi e benefici
- pianificare le attività e le risorse del progetto
- raccogliere i **requisiti**

2. Analisi

- si occupa del **cosa**
- descrivere il dominio dell'applicazione e specificare le funzioni delle varie componenti: lo **schema concettuale**

3. Progetto e realizzazione

- si occupa del **come**
- definire l'architettura del programma
- scrivere il **codice del programma** e produrre la **documentazione**

4. Verifica

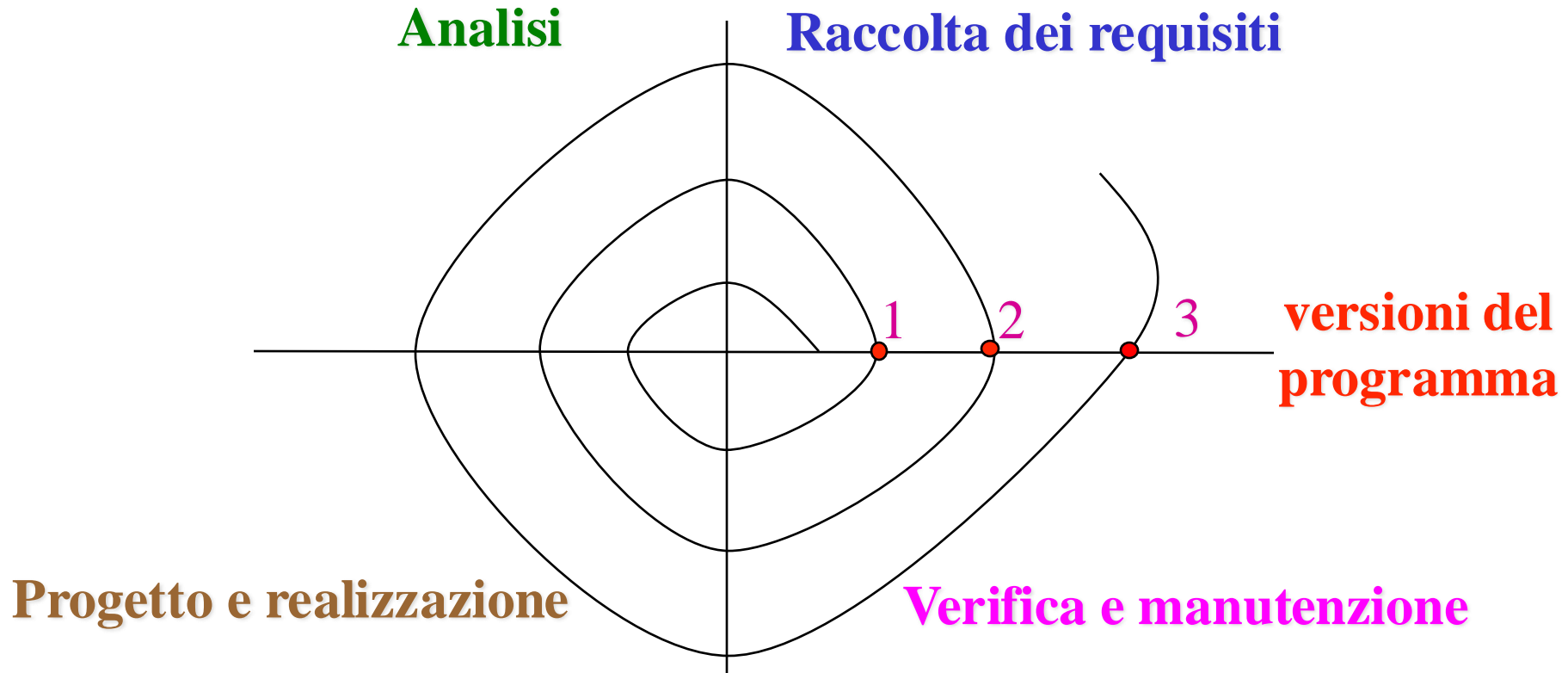
- Il programma svolge correttamente, completamente, efficientemente il compito per cui è stato sviluppato?

5. Manutenzione

- Correzione e aggiornamento del programma



Ciclo di vita (modello a spirale)





Progetto complessivo del sistema informatico

Studio di fattibilità e raccolta dei requisiti



Analisi

**Parte di analisi relativa alla
base di dati**



Progetto e realizzazione

**Parte di progetto e realizzazione
relativa alla base di dati**



Verifica e manutenzione



Progetto della basi di dati

Requisiti della base di dati

Progettazione
concettuale

Schema concettuale

COSA
(Analisi)

Progettazione
logica

• DBMS

Schema logico

COME
(Progetto e
realizzazione)

Progettazione
fisica / tuning

• DBMS

• Carico applicativo

Schema fisico



Modelli dei dati adottati in questo corso

- Schema concettuale:
 - Modello Entità-Relazione (ER)
- Schema logico:
 - Modello relazionale
 - DDL: SQL-standard
- Schema fisico:
 - Modello relazionale + strutture fisiche
 - DDL: SQL supportato dal DBMS scelto



5. La progettazione concettuale

5.2 modello ER

1. introduzione alla progettazione di basi di dati
- 2. modello ER**
3. metodologia per la progettazione concettuale
4. esempio



Il modello Entità-Relazione (ER) è un modello di dati

Un **modello (concettuale) di dati** è un linguaggio con il quale si esprimono schemi concettuali dei dati. Come ogni linguaggio, un modello dei dati è caratterizzato dai costrutti ammessi. Ogni costrutto ha un impatto sia a livello intensionale (**schema concettuale**), sia a livello estensionale (**istanze** dello schema). Per definire precisamente un modello di dati, occorre specificare per ogni costrutto

- il suo significato intuitivo, che determina il modo con cui si utilizza (**pragmatica**)
- la forma che assume a livello intensionale (**sintassi**)
- l'impatto sul livello estensionale (**semantica**)

Nel modello ER la struttura dello schema concettuale è descritta in forma grafica (diagramma dello schema)



Differenza tra schemi e istanze nei modelli di dati

- Lo **schema** descrive la struttura, cioè l'aspetto **intensionale** (ad esempio, nel modello relazionale, l'intestazione delle tabelle)
- Ad ogni schema corrispondono più **istanze dello schema** (aspetto **estensionale**), anche se, istante per istante, solo una è quella significativa (ad esempio, nel modello relazionale l'istanza corrente è formata dalle tuple che popolano le tabelle).
- Se S è uno schema che descrive la struttura di un universo di interesse, una **istanza di S** rappresenta lo stato (fotografia) di questo universo in un certo istante, ovvero la collezione di dati che vuole descrivere quello stato dell'universo.



Costrutti fondamentali del modello Entità-Relazione

- Entità
- Attributi di entità
- Relazioni
- Ruoli
- Attributi di relazione
- IS-A e Generalizzazioni
- Vincoli di identificazione
- Vincoli di cardinalità
- Altri vincoli



ER e UML

I costrutti del modello ER sono analoghi a quelli usati in altri modelli concettuali, ad esempio i diagrammi delle classi di UML

ER	UML
<ul style="list-style-type: none">• Entità• Relazioni• Attributi• Cardinalità• IS-A e Generalizzazioni• Identificatori• -----	<ul style="list-style-type: none">• Classi• Associazioni/aggregazioni• Attributi• Cardinalità• IS-A e Generalizzazioni• -----• Operazioni

Il modello ER è specializzato alla rappresentazione dei dati



La nozione di entità

Una entità è una **classe di oggetti** (fatti, persone, cose) che sono di interesse per l'applicazione, che hanno esistenza autonoma, e che hanno proprietà comuni

Esempi:

- ▶ **impiegato**
- ▶ **vendita**
- ▶ **dipartimento**
- ▶ **ordine**
- ▶ **città**
- ▶ **studente**

Sintassi: rappresentazione grafica di entità

Ogni **entità** ha nome che la identifica in modo univoco nello schema (non ci possono essere due entità diverse con lo stesso nome), ed è rappresentata da un rettangolo nel diagramma che descrive lo schema stesso

Esempi:



Impiegato

Dipartimento

Città

Vendita



Semantica delle entità

- A livello **estensionale** un'entità è costituita da un insieme di oggetti, che sono chiamati le sue **istanze** (detti anche esemplari, o occorrenze)
- Ciò significa che, se in uno schema **S** è definita una entità **E**, in ogni istanza **I** dello schema **S**, alla entità **E** è associato un insieme di oggetti (che viene denotato $istanze(I, E)$)
 $\{e_1, e_2, e_3, \dots\}$
che viene detto anche l'estensione di **E** nella istanza **I** dello schema **S**
- **Osservazione importante:** una istanza di entità non è un valore che rappresenta una proprietà di un oggetto, ma è l'oggetto stesso (spesso rappresentato da un suo surrogato, object identifier)



Entità e istanze d'entità

- Una entità (nello schema concettuale *S*) denota un insieme (o classe) di oggetti “simili” (quelle che poi diventeranno le sue istanze nelle istanze dello schema)
- Una istanza dell'entità (in una istanza di *S*) denota un oggetto dell'insieme rappresentato dall'entità

Nota: nello schema concettuale si rappresentano tipicamente le entità, non le singole istanze (“astrazione”)



La nozione di attributo di entità

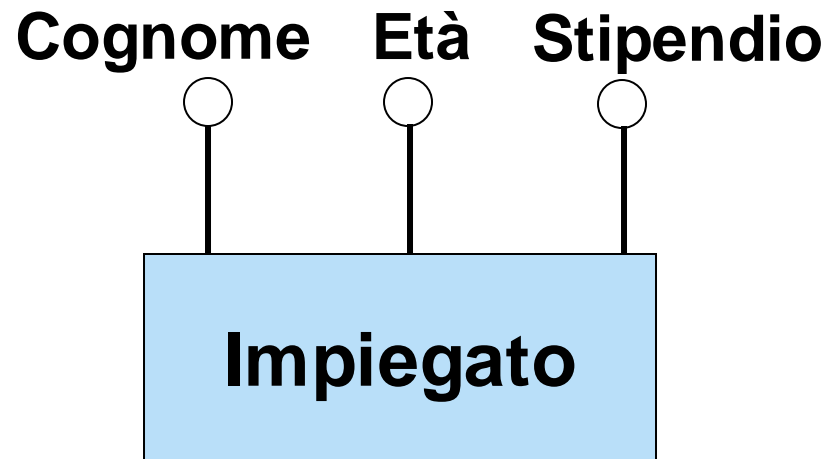
- Un attributo di entità è una **proprietà locale** di un'entità, di interesse ai fini dell'universo di interesse nell'applicazione
- Un attributo associa ad ogni istanza di entità un valore appartenente ad un **insieme di valori**, detto **dominio** dell'attributo (tipicamente, interi, caratteri, stringhe, ecc.)
- Si definisce un attributo per l'entità E quando si vuole rappresentare una proprietà **locale** delle istanze dell'entità E. Una proprietà di un oggetto si dice locale quando in ogni istanza dello schema il valore di tale proprietà dipende **solamente** dall'oggetto stesso, e **non ha alcun rapporto** con altri elementi dell'istanza dello schema



Sintassi: rappresentazione grafica di attributo

Ogni attributo di entità ha un nome che lo identifica in modo univoco nell'ambito della entità (attributi diversi devono avere nomi diversi), ed è rappresentato da un cerchio collegato alla entità per cui è rilevante.

Esempi:



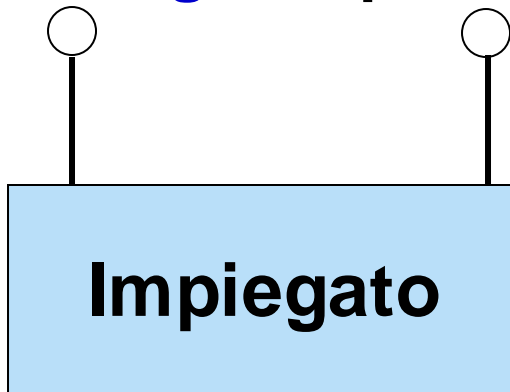


Sintassi: rappresentazione grafica di dominio

Il dominio associato ad un attributo viene generalmente tralasciato nella rappresentazione grafica (ma si indica nel dizionario dei dati – si veda in seguito).

Se si indica nel diagramma, la notazione è:

Cognome/stringa Stipendio/intero



Semantica degli attributi di entità

Se **I** è una istanza di uno schema **S** dove **A** è un attributo dell'entità **E** su un dominio **D**, $istanze(I, A)$, è un insieme di coppie (x,y) tali che x appartiene a $istanze(I, E)$, y è in **D**, ed esiste una ed una sola coppia in $istanze(I, A)$ per ogni x in $istanze(I, E)$. In altre parole, a livello **estensionale**, ovvero in ogni istanza **I** dello schema **S**, l'attributo **A** è una funzione totale

$$A : istanze(I, E) \rightarrow D$$

cioè, in ogni istanza **I** dello schema **S**, l'attributo **A** è una funzione che associa ad **ogni** oggetto che è istanza di **E** in **I** un valore appartenente a **D**.

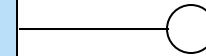


Esercizio 1: attributi di entità

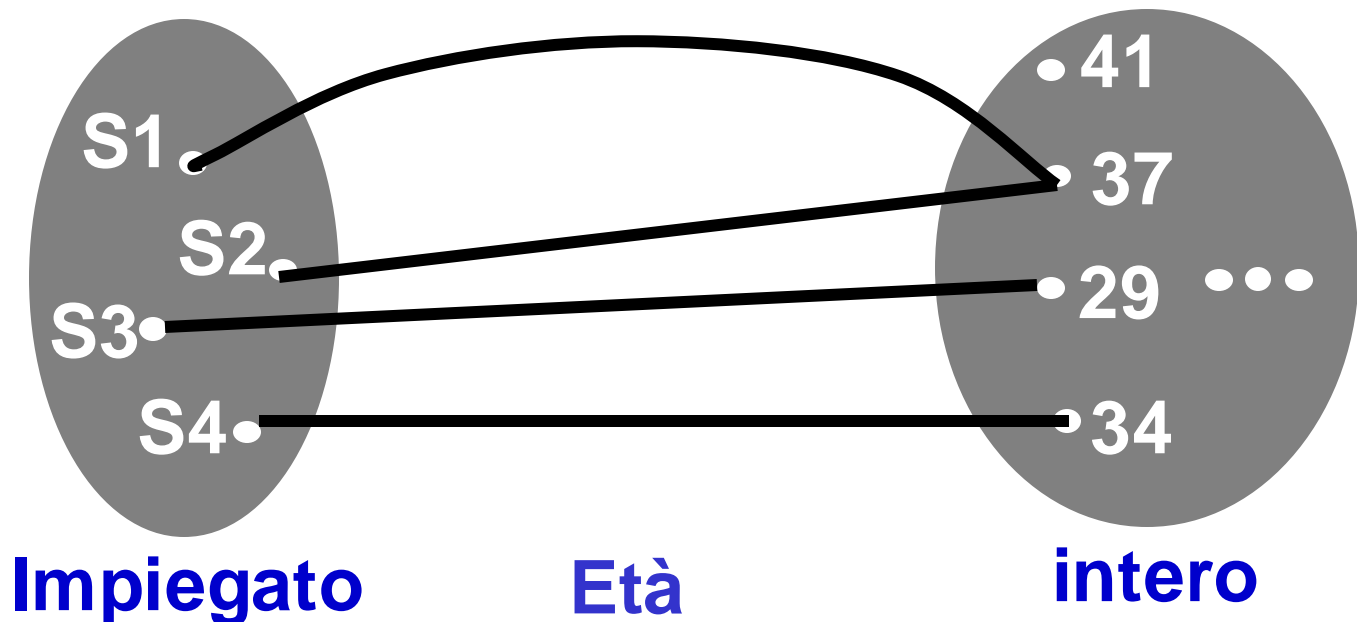
Dato lo schema:



Età/intero



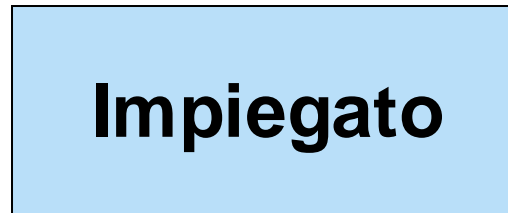
La seguente è
una sua istanza ?



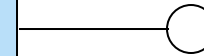


Esercizio 1: soluzione

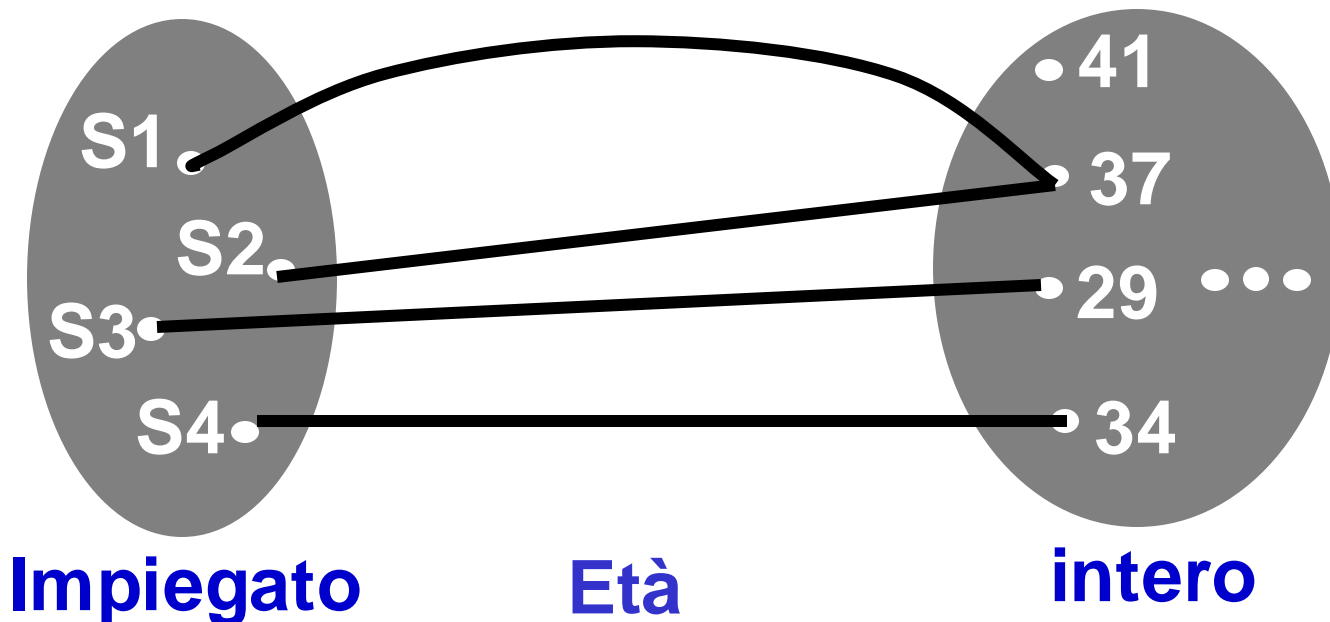
Dato lo schema:



Età/intero



La seguente è
una sua istanza ? **SI'!**



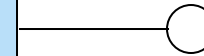


Esercizio 2a: attributi di entità

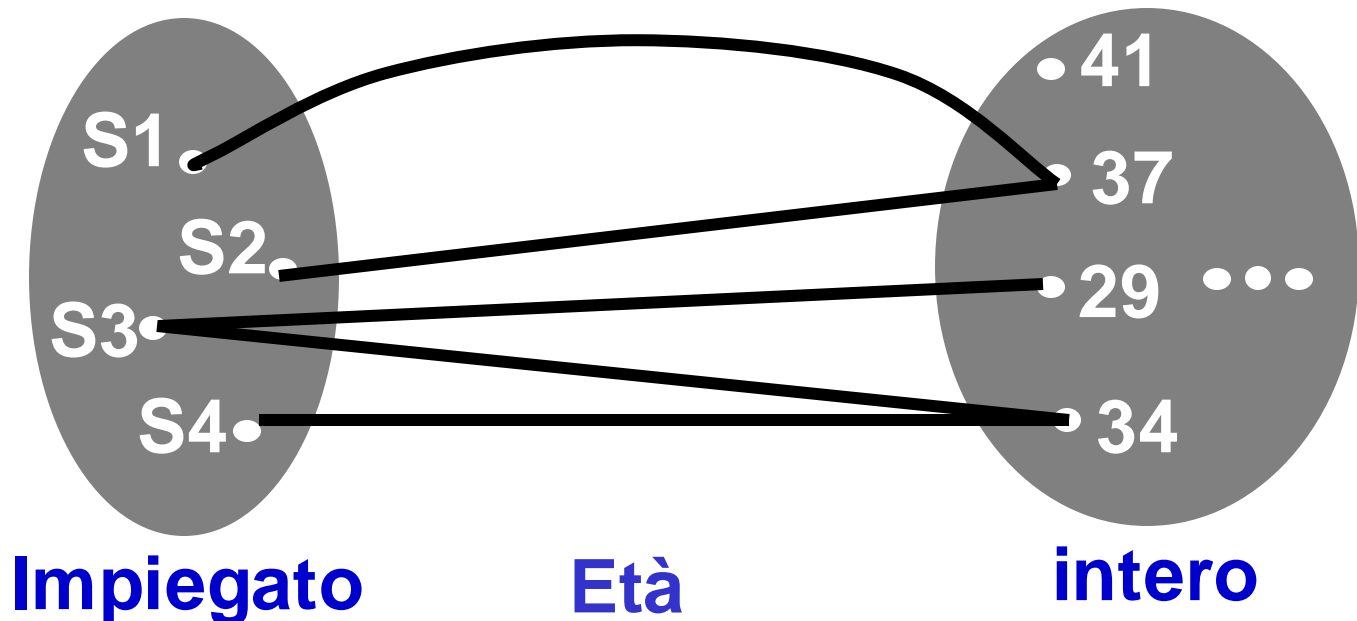
Dato lo schema:



Età/intero



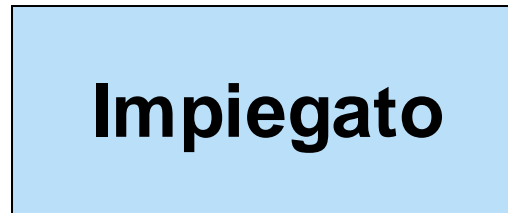
La seguente è
una sua istanza ?



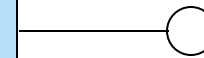


Esercizio 2a: soluzione

Dato lo schema:

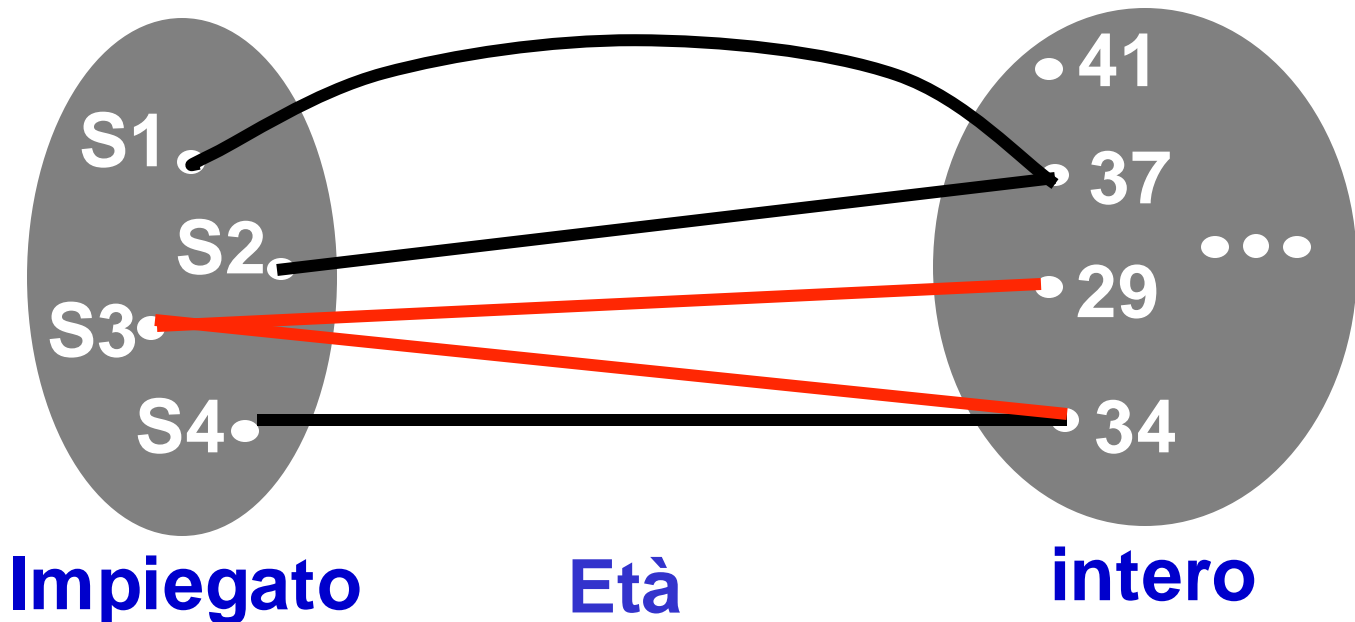


Età/intero



La seguente è
una sua istanza ?

No, l'attributo Età associa due valori a S3!
Ricordarsi che un attributo è una funzione.



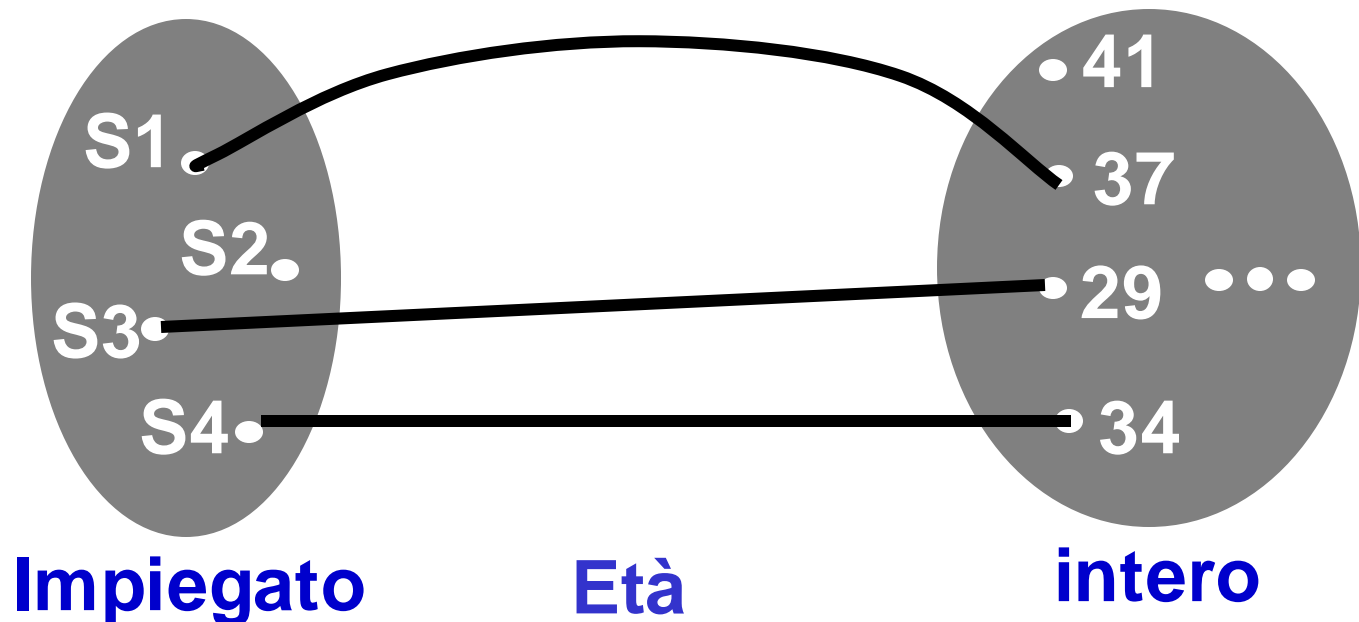


Esercizio 2b: attributi di entità

Dato lo schema:



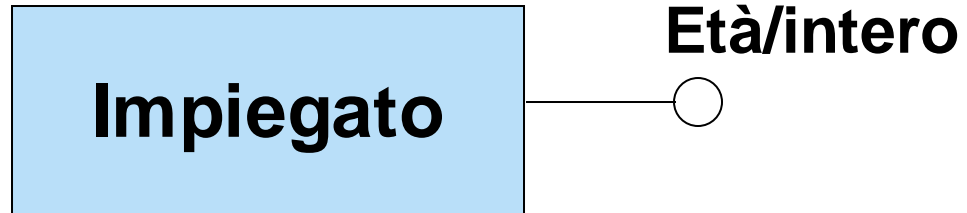
La seguente è
una sua istanza ?





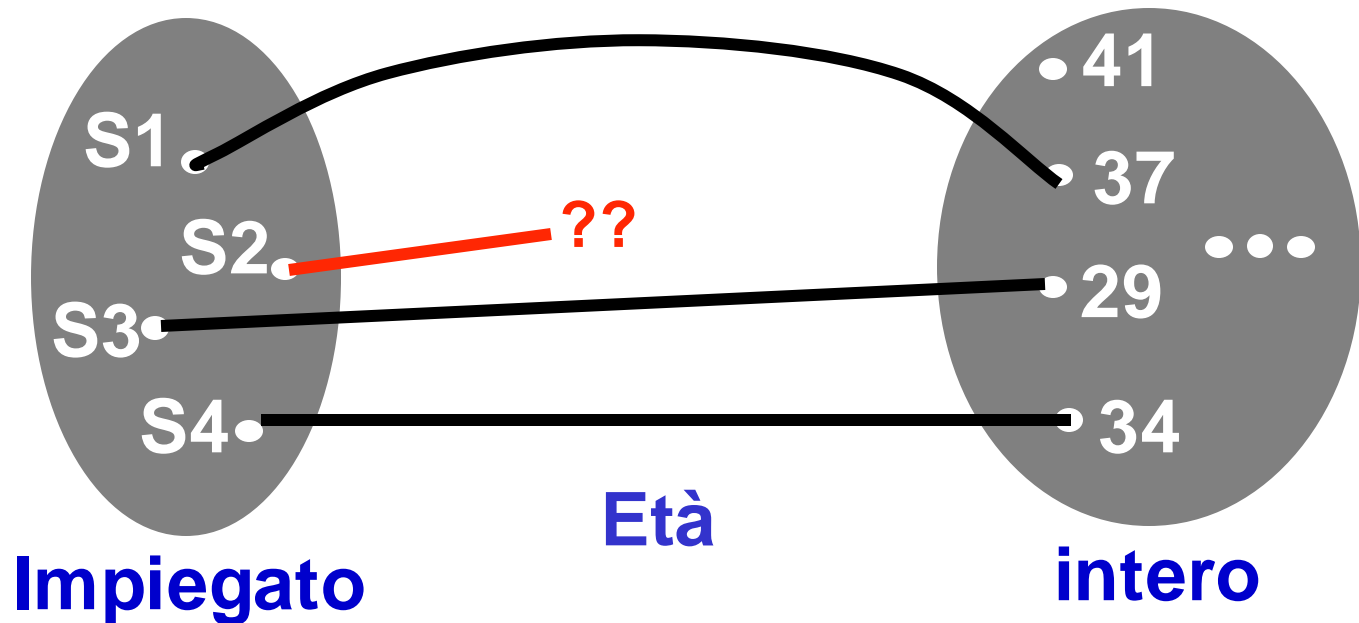
Esercizio 2b: soluzione

Dato lo schema:



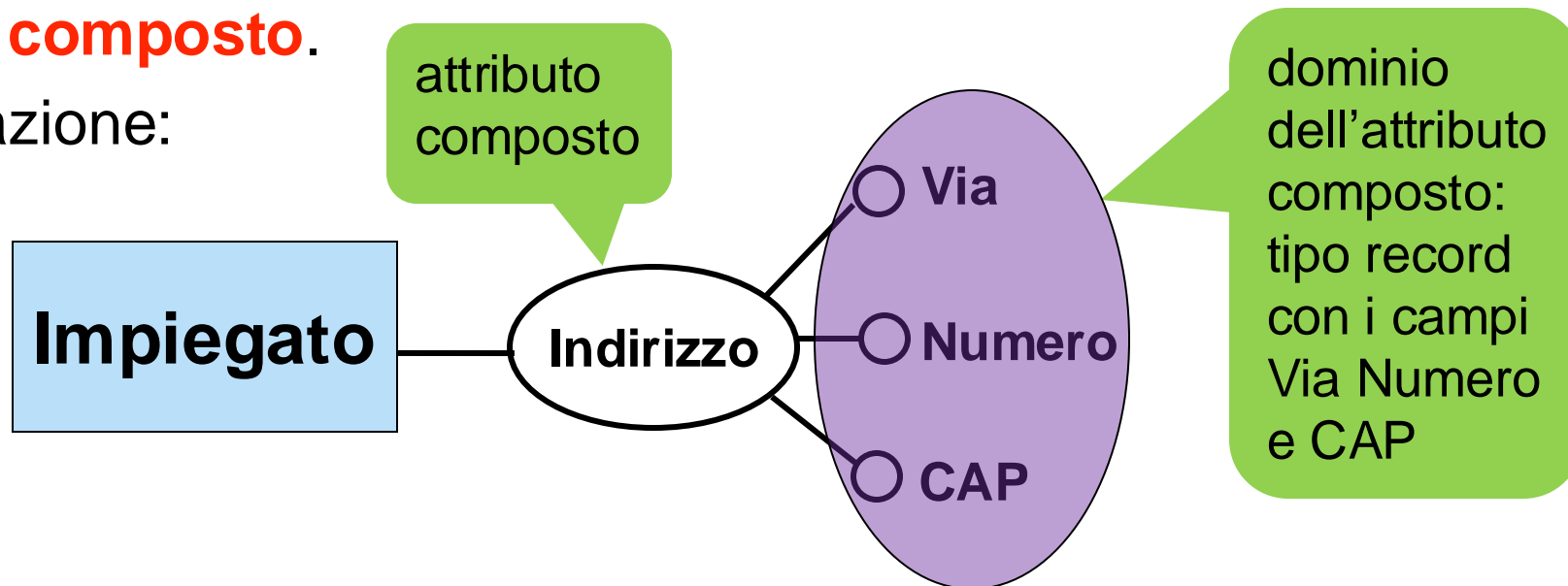
La seguente è
una sua istanza ?

No, manca il valore di età per S2! Ricordarsi
che un attributo è una funzione totale



Attributi composti

- Un attributo può anche essere definito su un dominio complesso. Di particolare interesse è il caso di dominio di tipo “record”. Un attributo il cui dominio è il tipo record si dice **composto**.
- Notazione:



Nell'esempio qui sopra, “Indirizzo” è un attributo composto, perché il suo dominio è di tipo record, con campi “Via”, “Numero” e “CAP”.



La nozione di relazione

Una **relazione** (o associazione) si definisce su due o più entità, e rappresenta un **legame** fra tali entità. Il numero di entità coinvolte in una relazione determina il suo **grado**

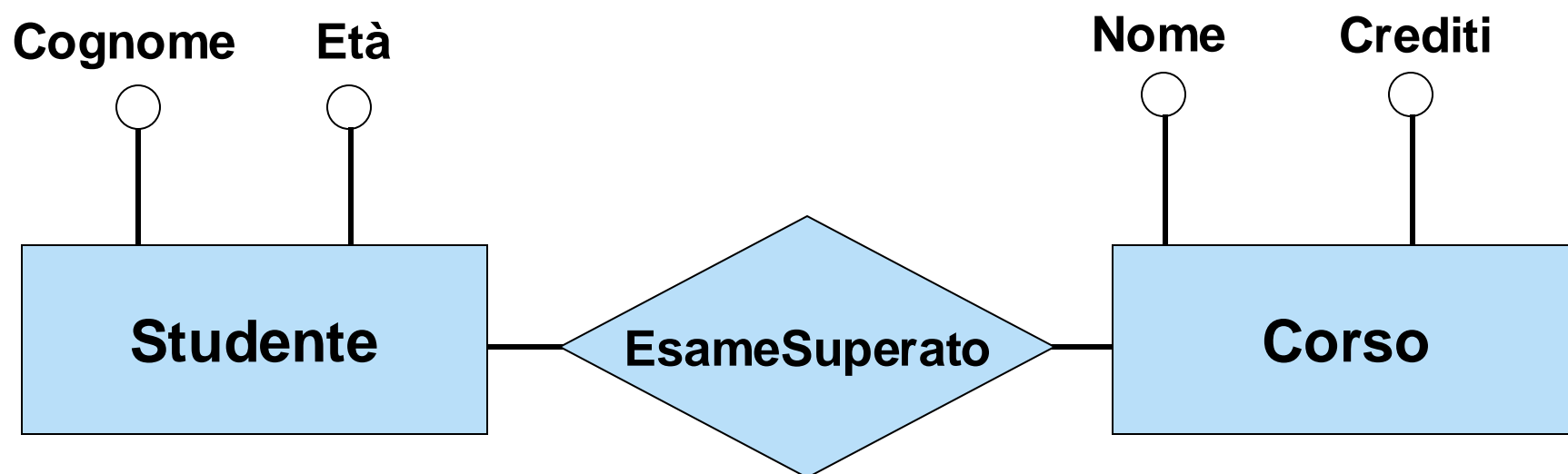
Esempi

- ▶ **Residenza** (tra Persona e Città)
- ▶ **Afferenza** (tra Impiegato e Dipartimento)

Sintassi: rappresentazione grafica di relazione

Ogni **relazione** ha un nome che la identifica in modo univoco nello schema, ed è rappresentata nel diagramma che descrive lo schema da un rombo collegato alle entità sulle quali è definita la relazione

Esempio di relazione di grado 2 (relazione binaria tra Studente e Corso):



Introduzione alla semantica delle relazioni

- A livello **estensionale** una relazione binaria **R** tra le entità **E** ed **F** ha come istanze coppie etichettate della forma **(E:x,F:y)**, tali che **x** è una istanza di **E**, ed **y** è una istanza di **F**. Ogni coppia etichettata è appunto detta **istanza** della relazione **R**
- Quindi, se in uno schema **S** è definita una relazione **R** sulle entità **E** ed **F**, in ogni istanza **I** dello schema **S**, alla relazione **R** è associato un insieme di coppie etichettate, denotato da *istanze(I,R)*:

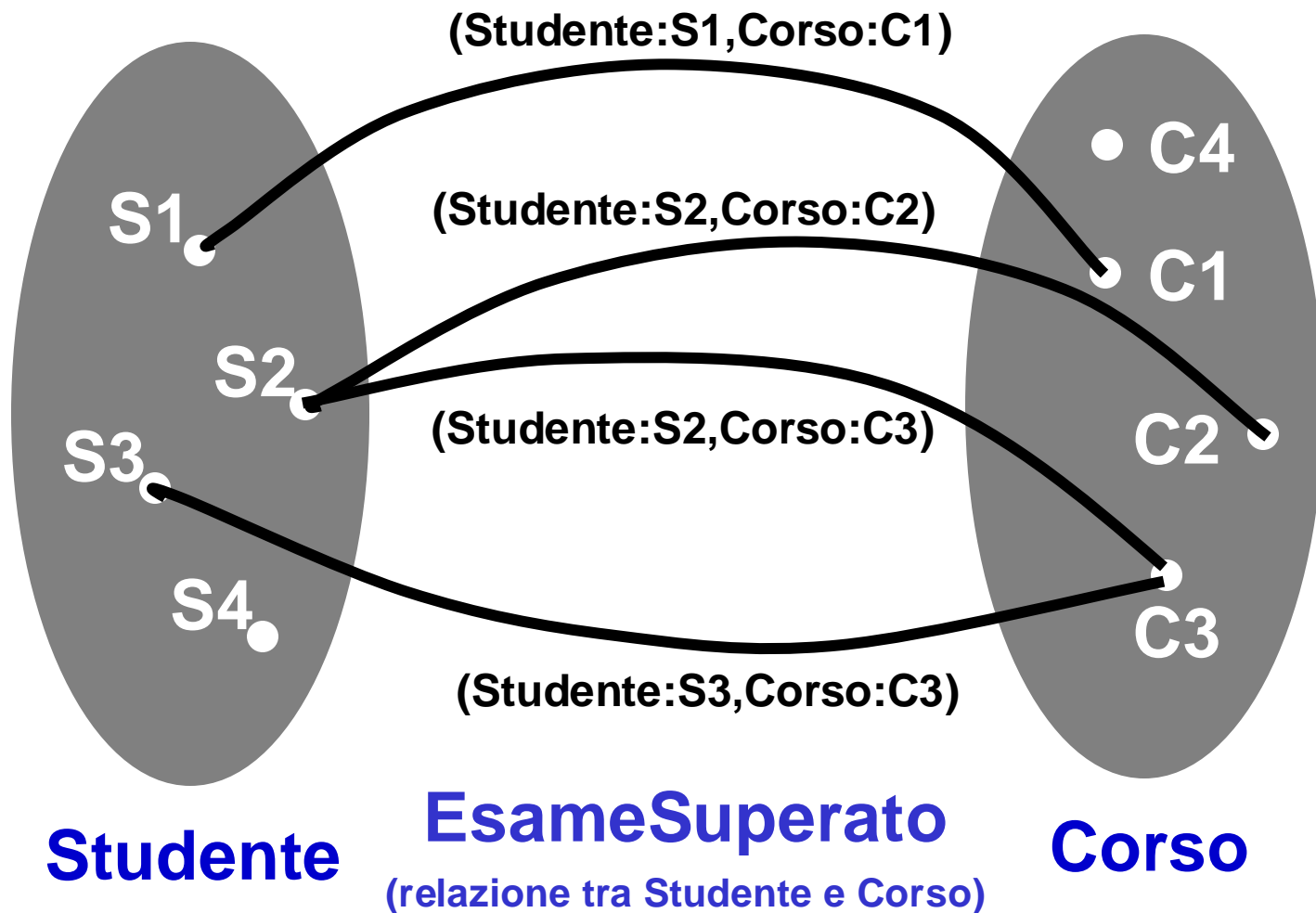
$$\{(E:x1, F:y1), (E:x2, F:y2), (E:x3, F:y3), \dots\}$$

che viene detto l'estensione di **R** nella istanza **I** dello schema **S**

- In altre parole, una relazione nel modello ER è, dal punto di vista della semantica, una relazione matematica. In ogni istanza **I** dello schema **S** si ha: $istanze(I,R) \subseteq istanze(I,E) \times istanze(I,F)$, ma con l'avvertenza che ogni coppia che fa parte di *istanze(I,E)* è etichettata, ovvero ogni componente *d* della coppia viene associata all'entità **C** di cui è istanza usando la notazione **C:d**.

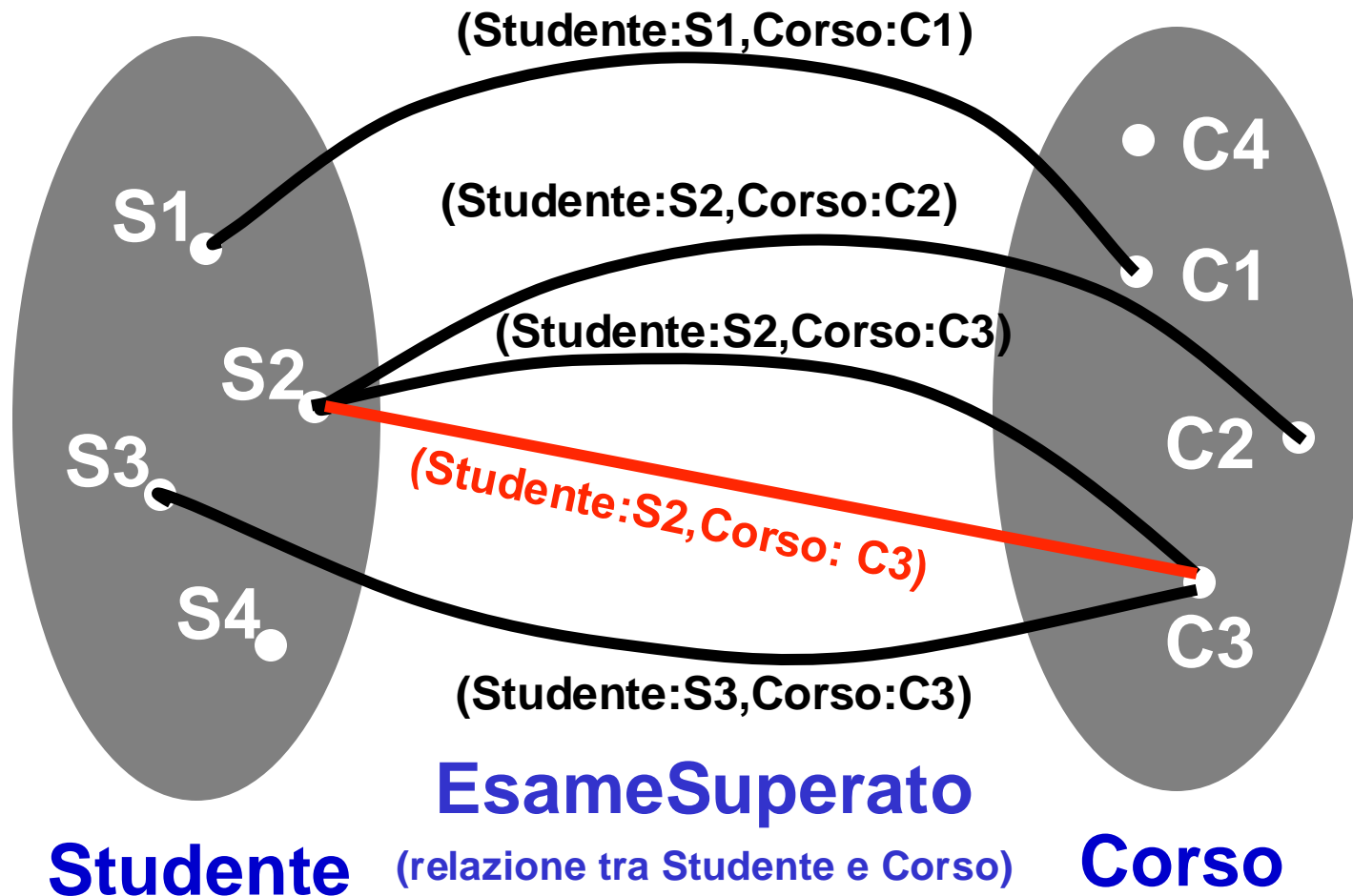


Relazione a livello estensionale



Osservazione importante

Dalla semantica delle relazioni segue immediatamente che **non** possono esistere due istanze della stessa relazione che coinvolgono le stesse istanze di entità.

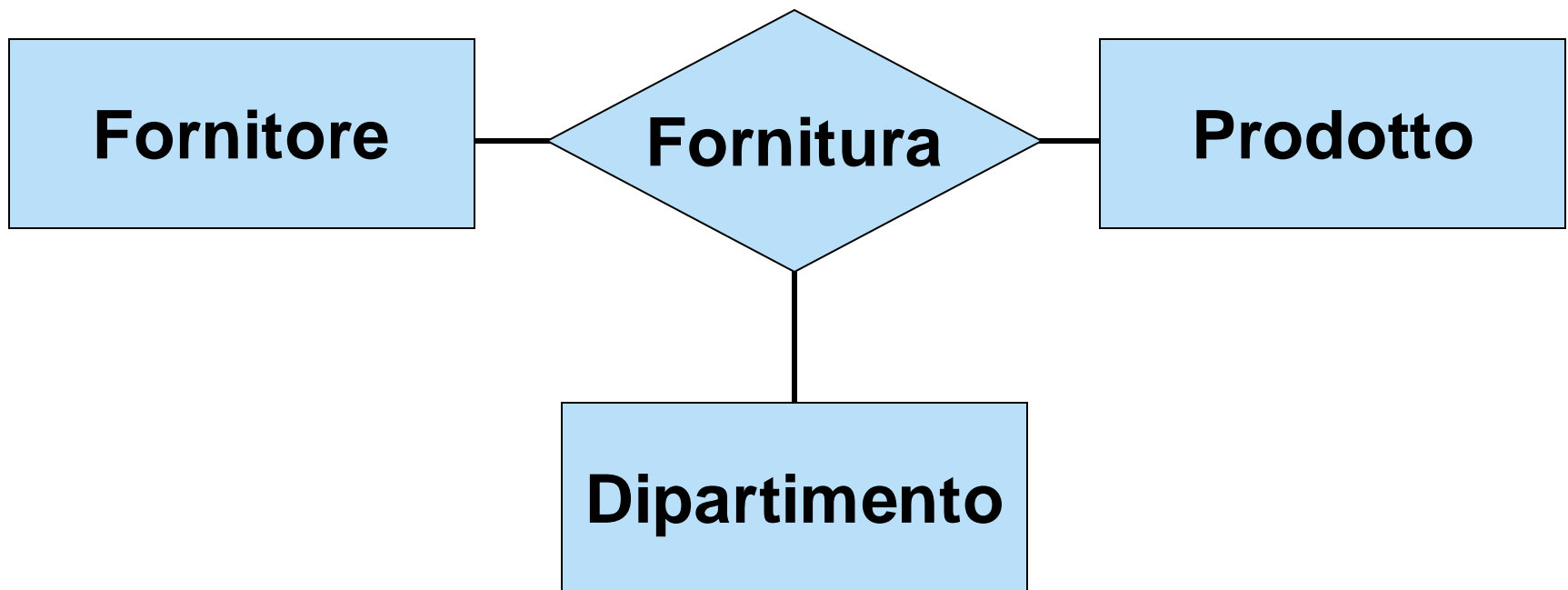




Relazioni n-arie (grado maggiore di 2)

Una relazione di grado maggiore di 2 si dice n-aria.

Esempio (relazione Fornitura tra Fornitore, Dipartimento e Prodotto):



Semantica delle relazioni (prima versione)

- A livello **estensionale** (ovvero in ogni istanza **I** dello schema **S**) una relazione **R** tra le entità **E₁, E₂, ..., E_n**, che per il momento assumiamo tutte diverse tra loro, è costituita da un insieme di tuple (o n-ple) etichettate

$$(E_1:x_1, E_2:x_2, \dots, E_n:x_n)$$

tali che **x₁** è una istanza di **E₁** in **I**, **x₂** è una istanza di **E₂** in **I**, ..., **x_n** è una istanza di **E_n** in **I**. Ogni tupla è detta **istanza** della relazione **R** nella istanza **I** dello schema **S**

- Quindi, in ogni istanza **I** dello schema **S** si ha che è un sottoinsieme del prodotto cartesiano tra i vari insiemi *istanze(I, E₁)*, ... , *istanze(I, E_n)*

$$istanze(I, R) \subseteq istanze(I, E_1) \times \dots \times istanze(I, E_n)$$

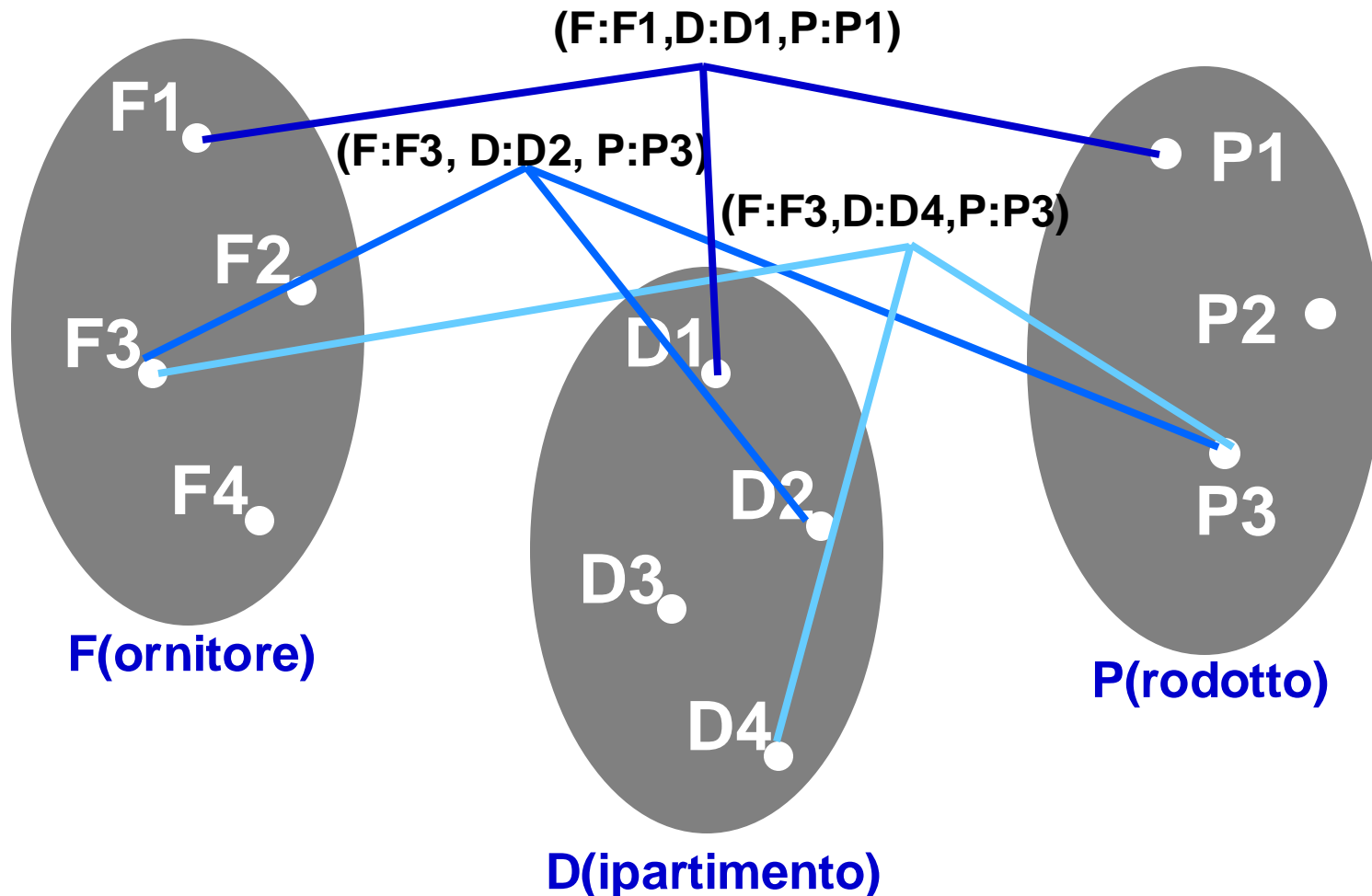
ma con l'avvertenza che le tuple in questo insieme sono etichettate



Relazione n-aria a livello estensionale

Relazione Fornitura

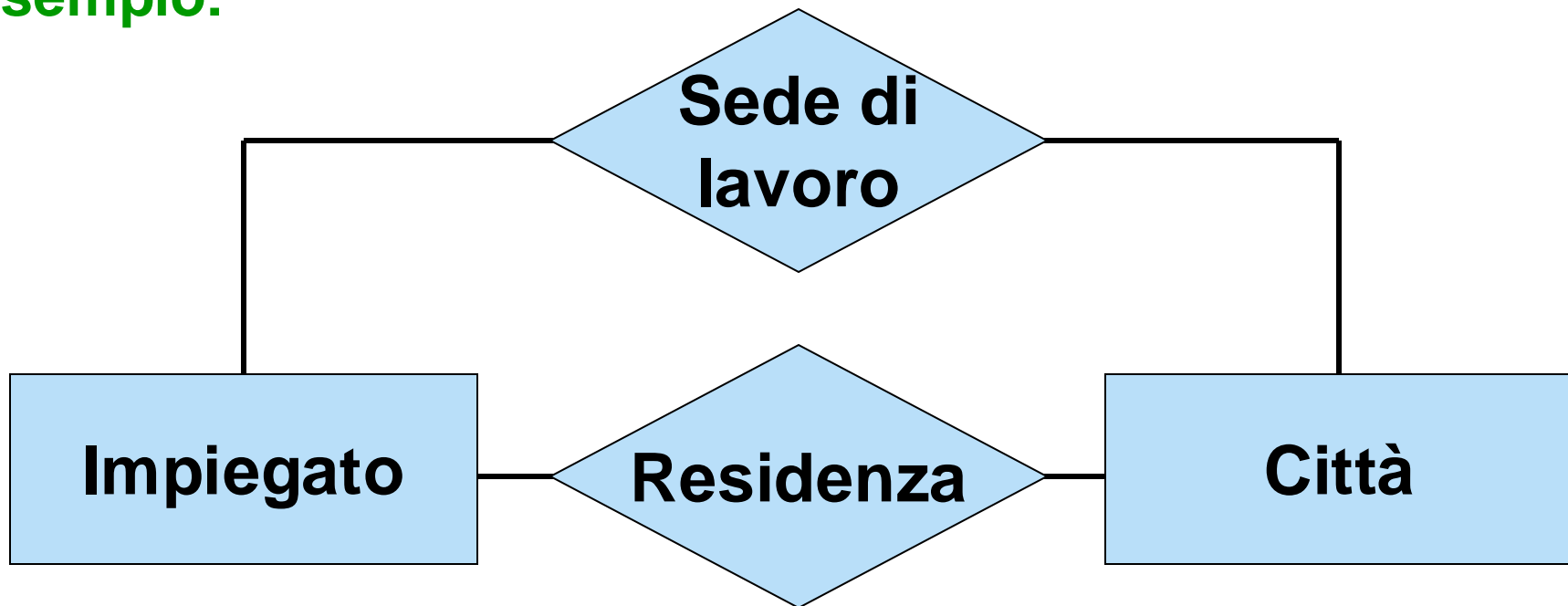
(tra F(ornitore), D(ipartimento) e P(rodotta))





Ovviamente lo stesso insieme di entità può essere coinvolto in più relazioni

Esempio:



“Sede di Lavoro” e “Residenza” rappresentano due legami diversi tra le stesse entità “Impiegato” e “Città”



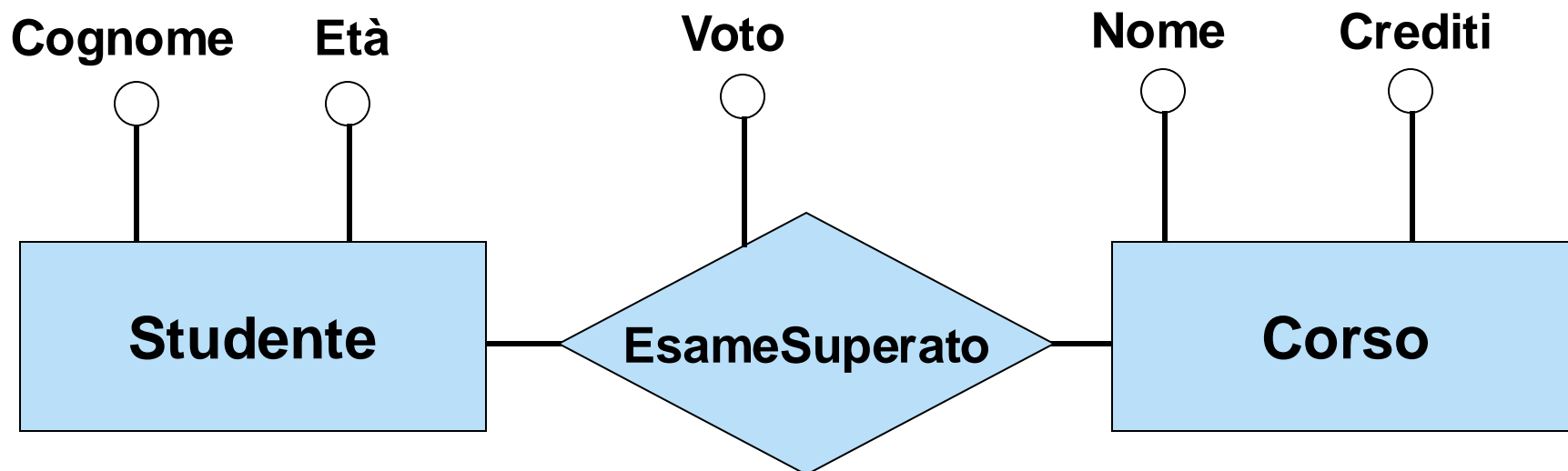
La nozione di attributo di relazione

- Un attributo di relazione è una proprietà locale di una relazione, di interesse ai fini dell'applicazione
- Un attributo della relazione R tra le entità E_1, E_2, \dots, E_n modella una proprietà **non** di E_1 , **non** di E_2, \dots , **non** di E_n , ma del **legame** tra E_1, E_2, \dots, E_n rappresentato da R
- Analogamente al caso di attributo di entità, un attributo di relazione associa ad ogni istanza di relazione un valore appartenente ad un insieme detto **dominio** dell'attributo
- **Attenzione**: l'attributo non fa parte della natura intrinseca della relazione, e perciò non potrà distinguere istanze diverse della stessa relazione

Sintassi: rappresentazione grafica di attributo di relazione

Ogni attributo di relazione ha un nome che lo identifica in modo univoco nell'ambito della relazione, ed è rappresentato da un cerchio collegato alla relazione a cui appartiene.

Esempio



Semantica degli attributi di relazione

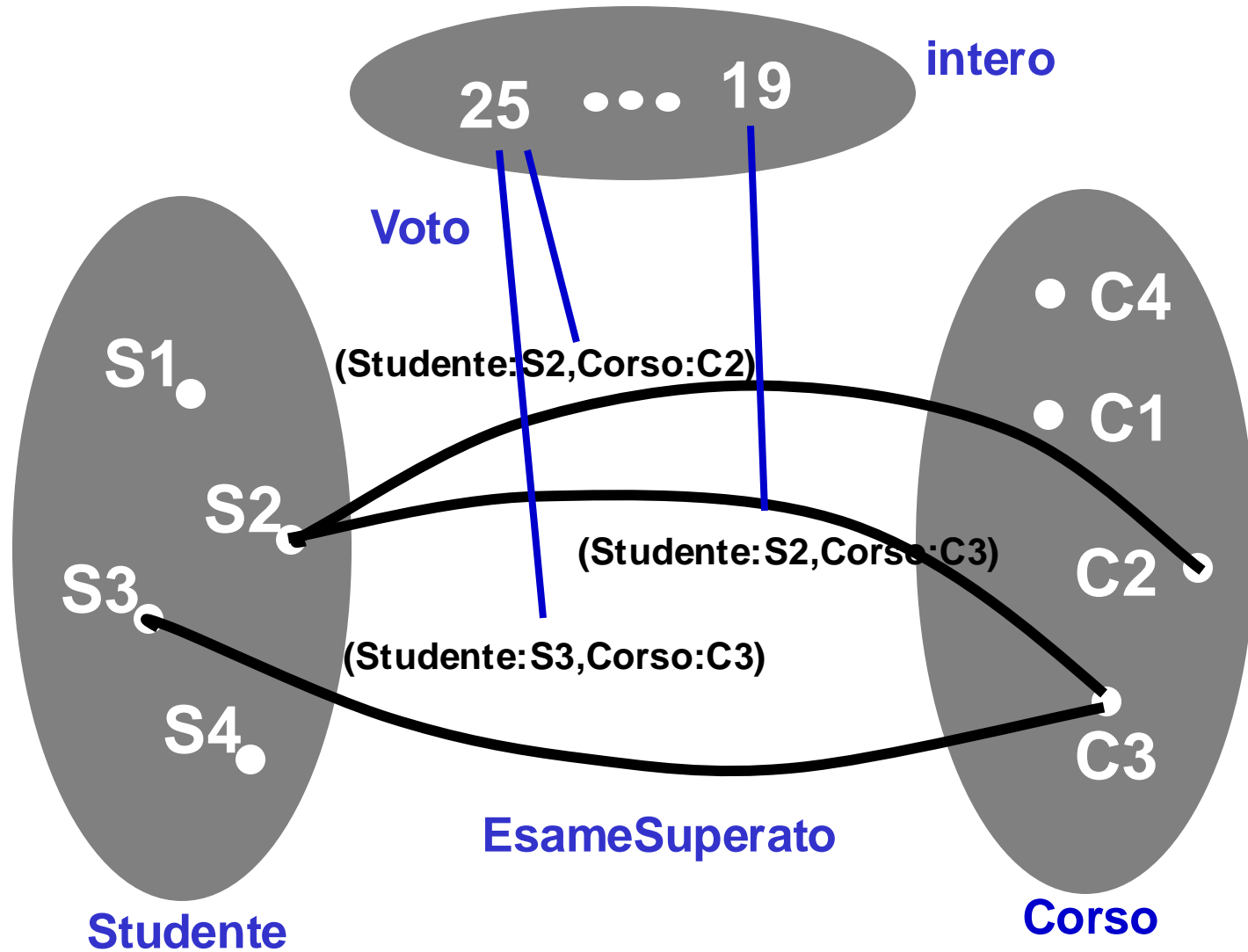
Se **I** è una istanza di uno schema **S** dove **A** è un attributo della relazione **R** su un dominio **D**, allora $istanze(I, A)$ è un insieme di coppie (x,y) tali che x è in $istanze(I, R)$, y è in **D**, ed esiste una ed una sola coppia in $istanze(I, A)$ per ogni x in $istanze(I, R)$. In altre parole, a livello **estensionale** (ovvero in ogni istanza **I** dello schema **S**) l'attributo **A** è una funzione totale

$$A : istanze(I, R) \rightarrow D$$

cioè, in ogni **I** che è una istanza di **S**, l'attributo **A** è una funzione che associa ad **ogni** tupla che è istanza di **R** in **I** un valore appartenente a **D**.



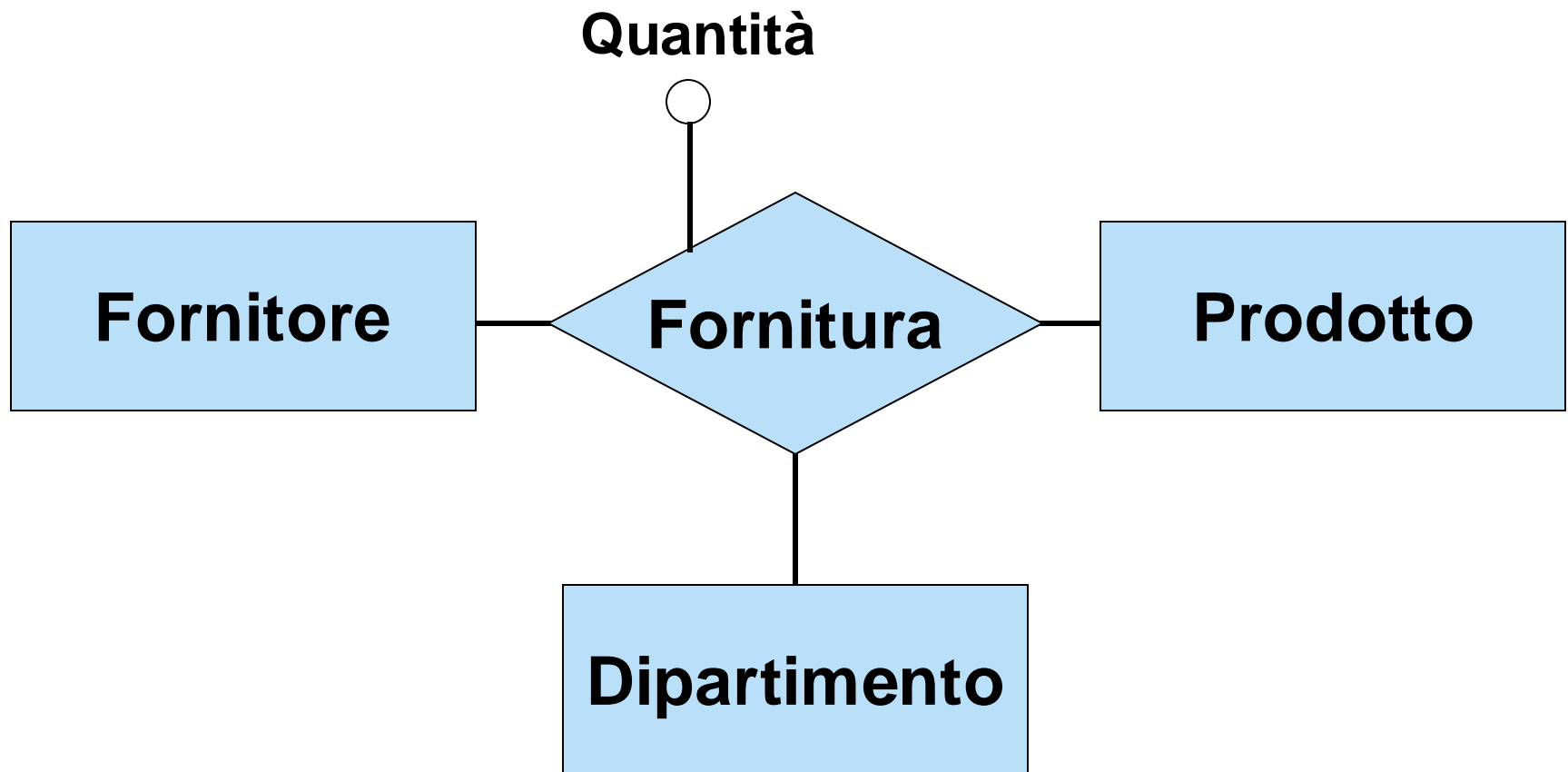
Attributi di relazione binaria: livello estensionale





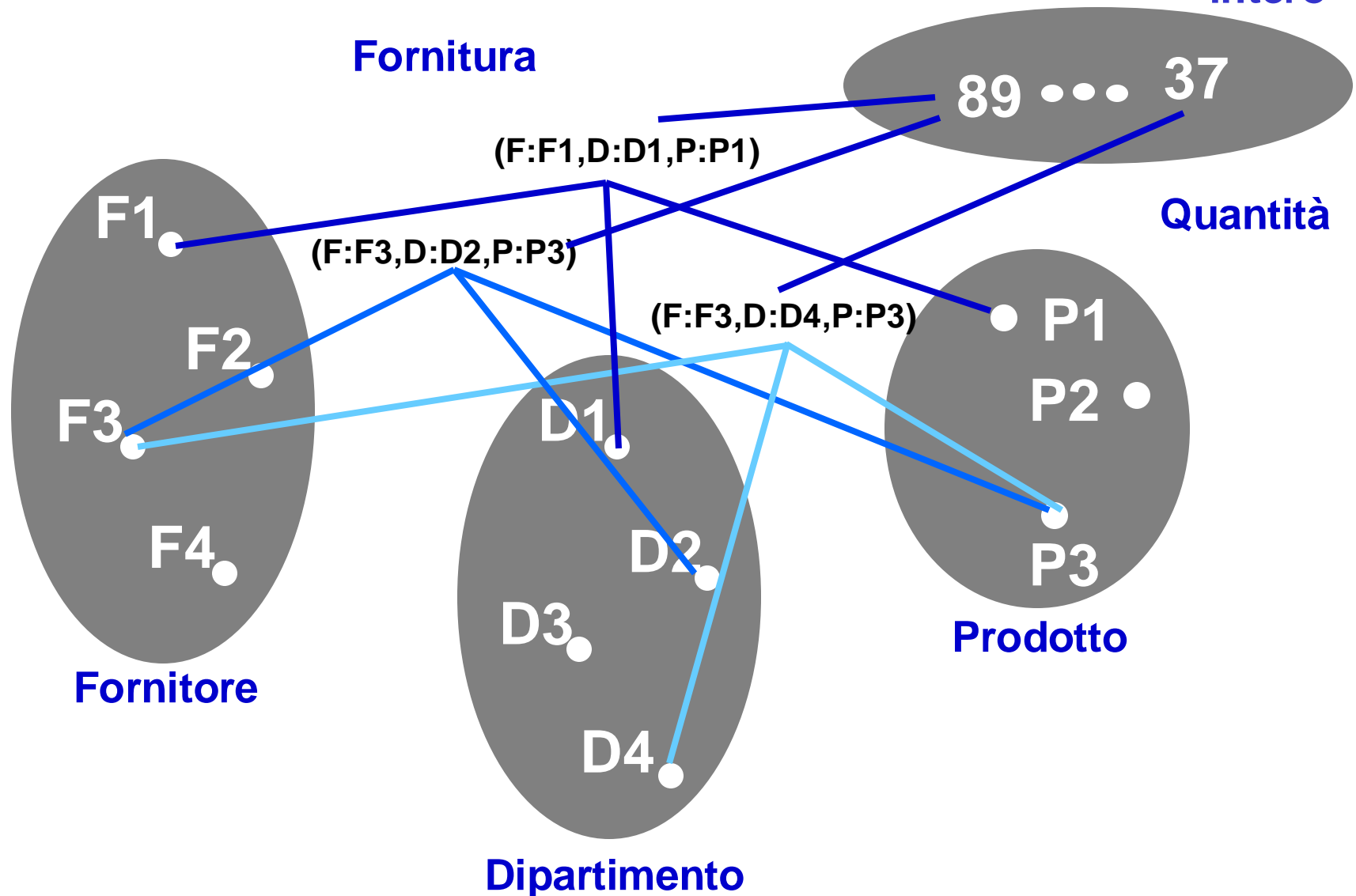
Relazioni n-arie con attributi

Esempio di relazione n-aria con attributo





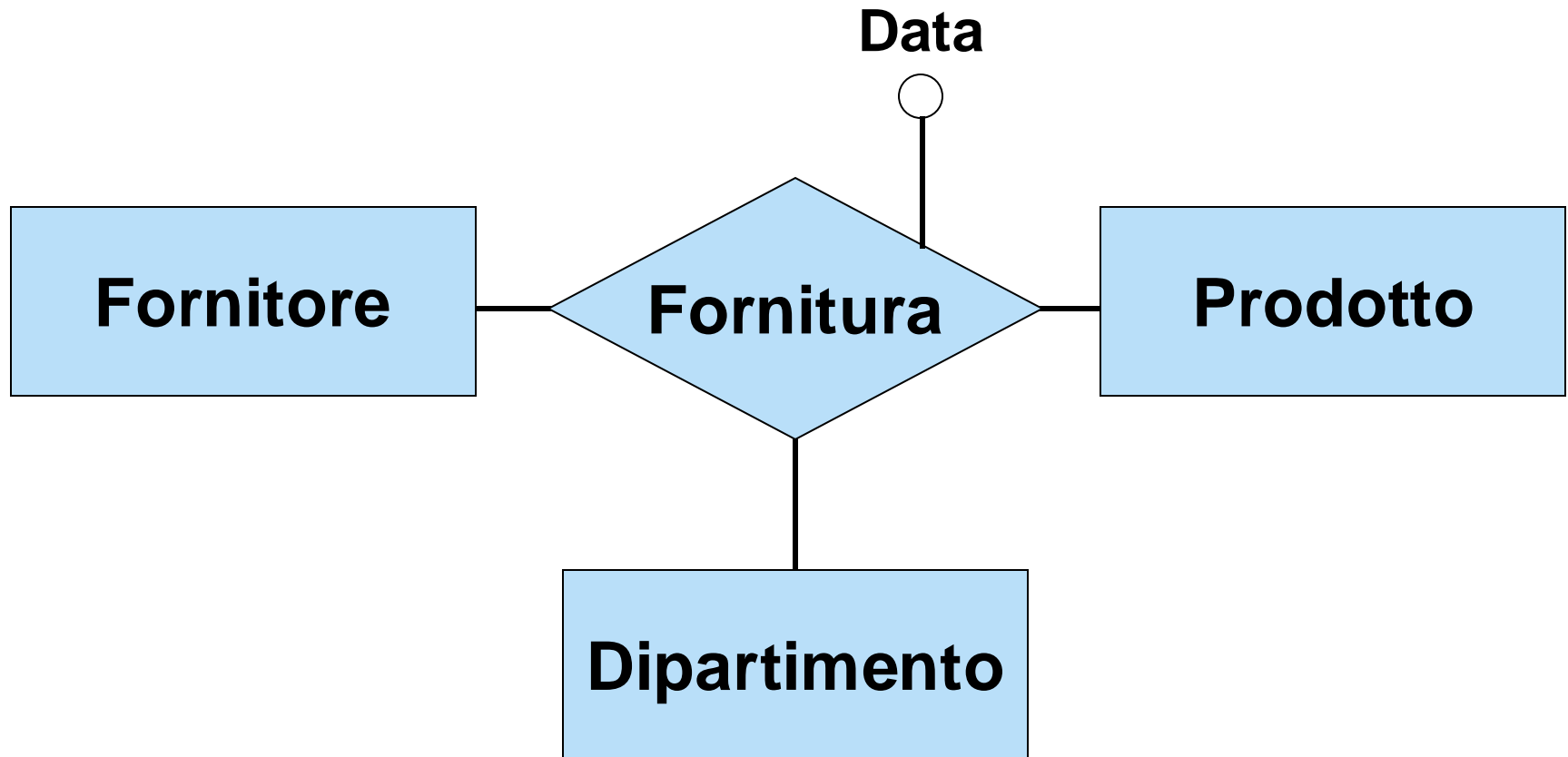
Attributi di relazione n-aria: livello estensionale intero





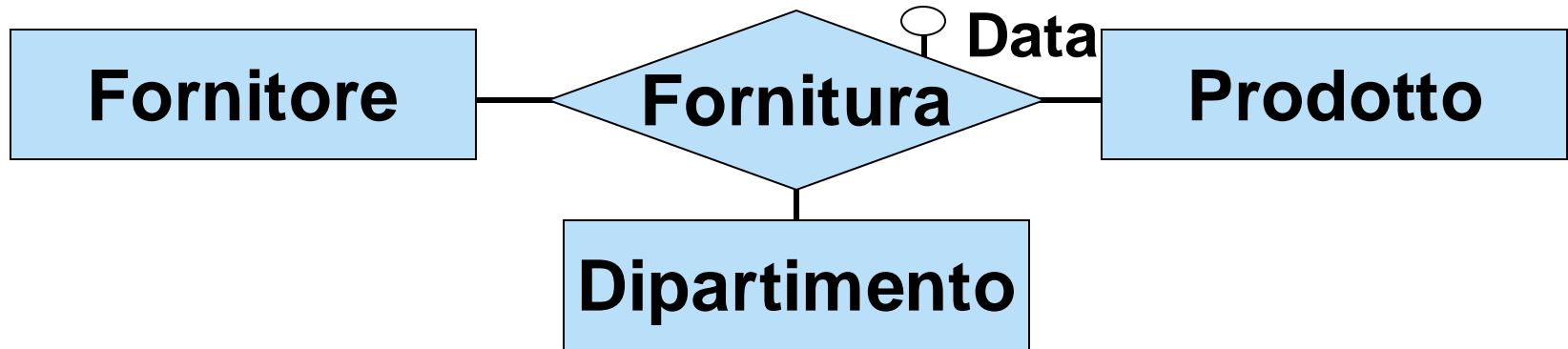
Esercizio 3: attributi di relazione

Se diverse forniture possono riguardare la stessa tripla $\langle \text{fornitore}, \text{prodotto}, \text{dipartimento} \rangle$ ma in diverse date, lo schema qui sotto è corretto?





Esercizio 3: soluzione



Lo schema è scorretto. Se diverse forniture (diciamo f_1 e f_2) riguardano la stessa tripla (diciamo $\langle t, p, m \rangle$) di $\langle \text{fornitore}, \text{prodotto}, \text{dipartimento} \rangle$ in date diverse (diciamo d_1 e d_2), vuol dire che in una istanza I dello schema dovremmo avere

$\text{istanze}(I, \text{Fornitura}) = \{ \langle \text{fornitore}:t, \text{prodotto}:p, \text{Dipartimento}:m \rangle, \langle \text{fornitore}:t, \text{prodotto}:p, \text{Dipartimento}:m \rangle \},$

con $\text{istanze}(I, \text{Date}) = \{ \langle \langle \text{fornitore}:t, \text{prodotto}:p, \text{Dipartimento}:m \rangle, d_1 \rangle, \langle \langle \text{fornitore}:t, \text{prodotto}:p, \text{Dipartimento}:m \rangle, d_2 \rangle \},$

ma **questo non è possibile**, perché implicherebbe che $\text{istanze}(I, \text{Fornitura})$ non è un insieme (infatti, conterrebbe lo stesso elemento più di una volta), contraddicendo la definizione di relazione. Vedremo più avanti quale schema corretto corrisponde alla situazione descritta.



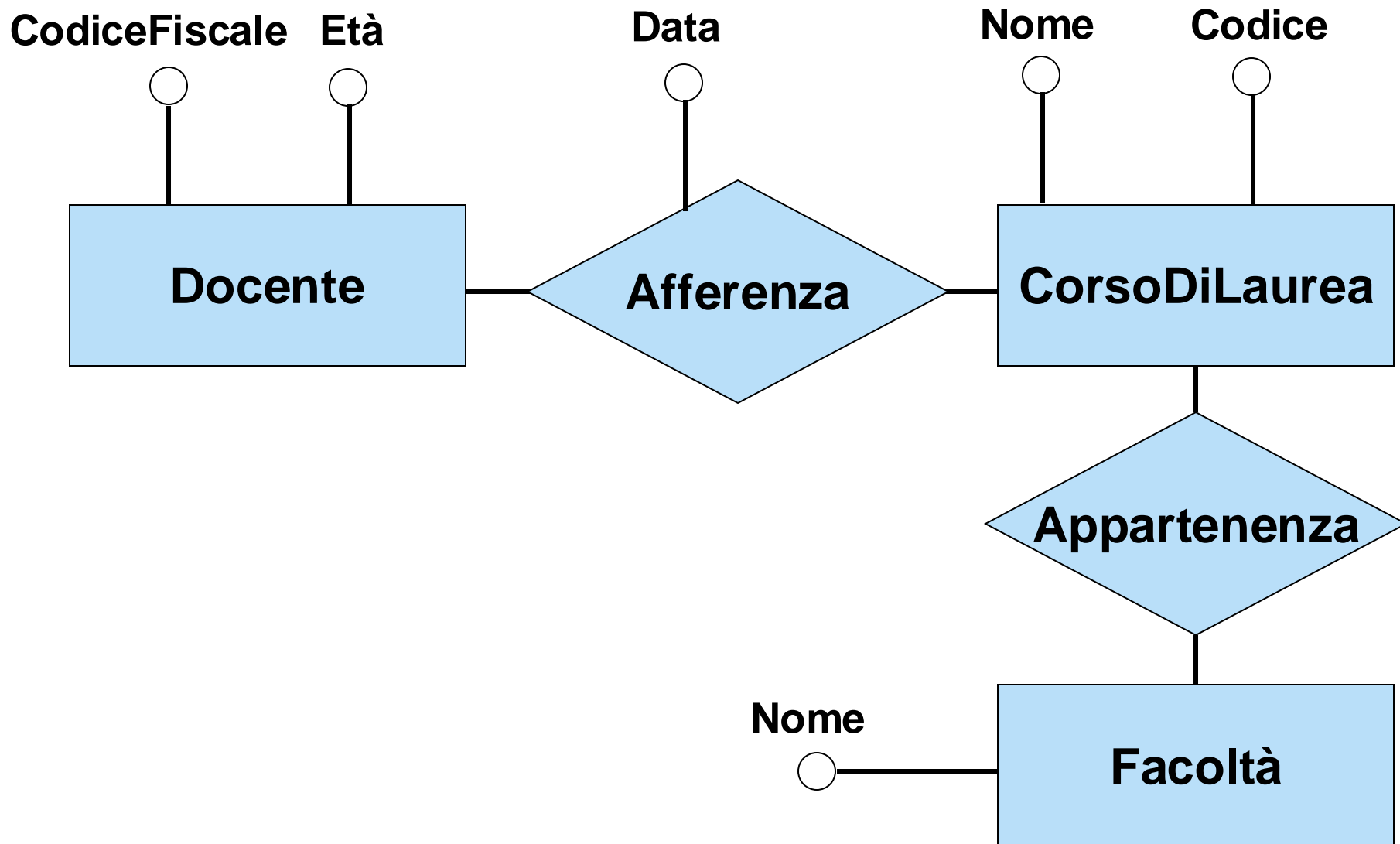
Esercizio 4: un semplice schema concettuale

Descrivere lo schema concettuale della seguente realtà:

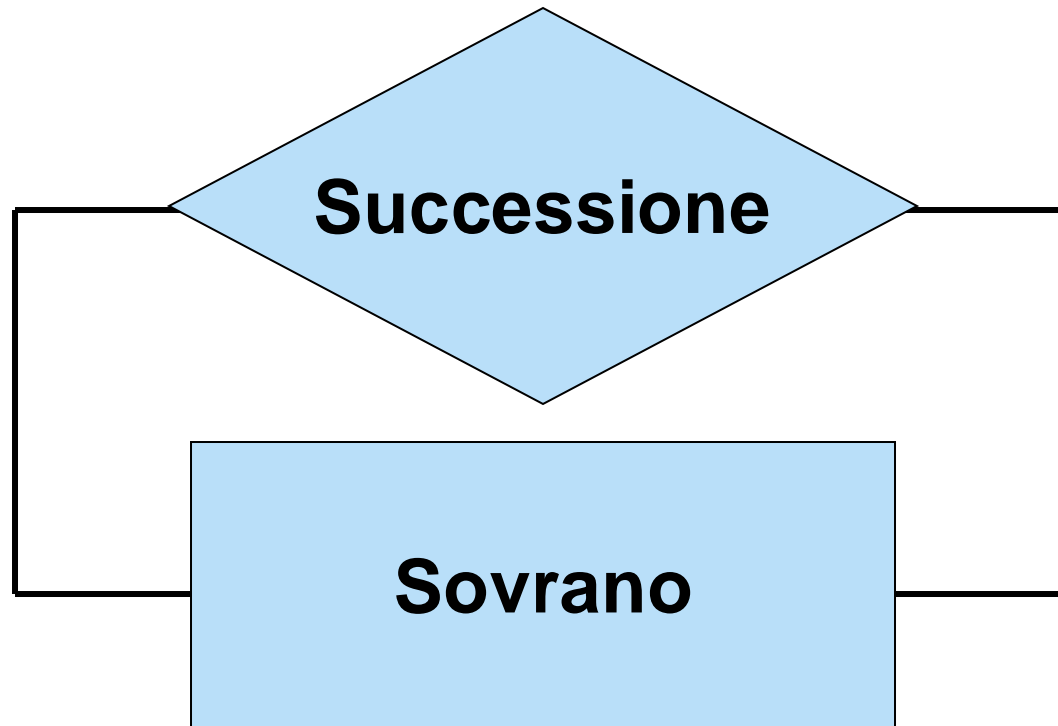
I docenti hanno un codice fiscale ed una età. I docenti operano nei corsi di laurea (si dice che afferiscono ai corsi di laurea). Interessa la data di afferenza dei docenti ai corsi di laurea. I corsi di laurea hanno un codice ed un nome, ed appartengono alle facoltà. Ogni facoltà ha un nome.



Esercizio 4: soluzione



Una relazione può coinvolgere due o più volte la stessa entità

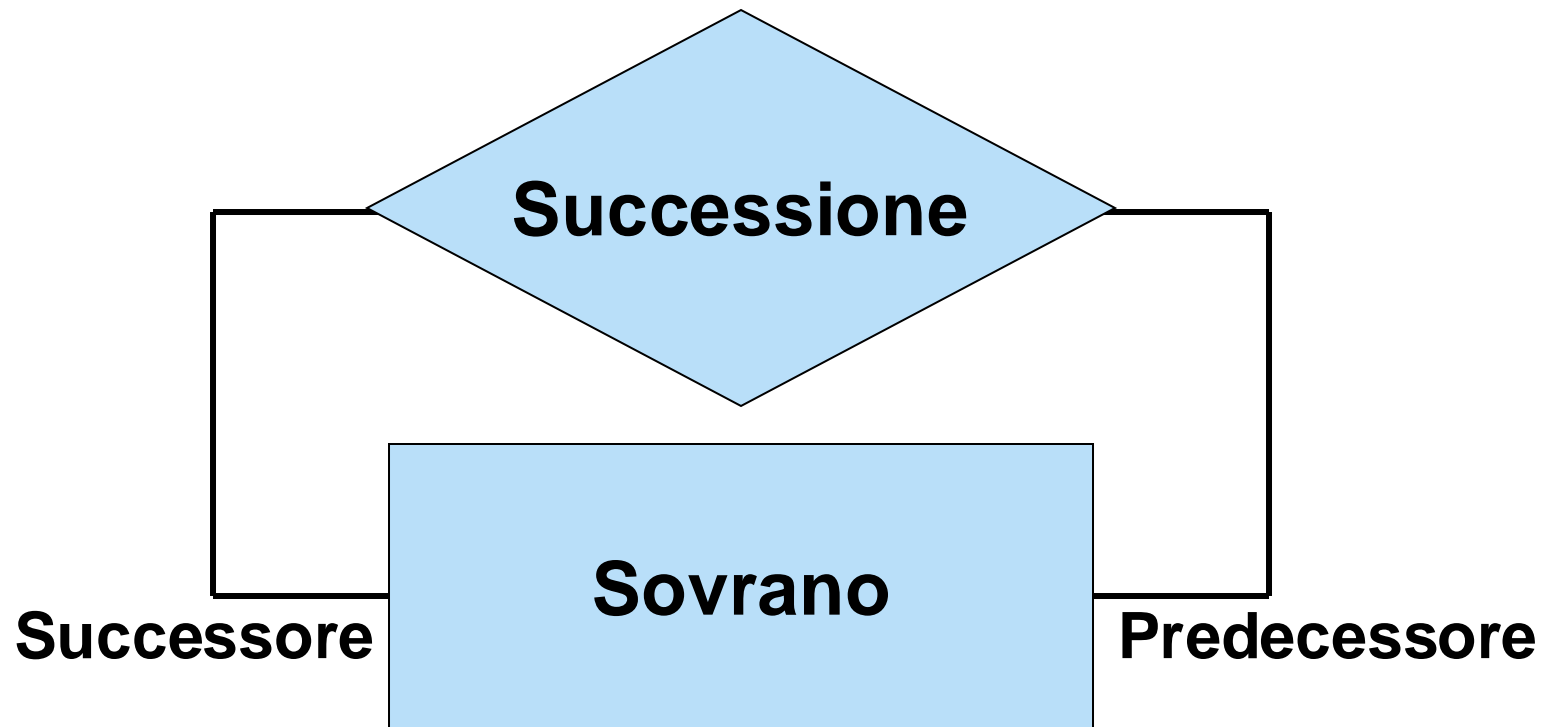


Problema: in una istanza di questo schema, data una coppia che è istanza di “Successione”, non si può individuare chi è il sovrano predecessore e chi il sovrano successore.



Relazioni sulla stessa entità

Nelle relazioni dove una stessa entità è coinvolta più volte è necessario aggiungere la specifica dei “**ruoli**”





La nozione di ruolo

- Si aggiunge quindi ad ogni relazione l'informazione che specifica il **ruolo** che ciascuna entità gioca nella relazione
- Se per una relazione **R** che coinvolge l'entità **E** (e altre entità) si stabilisce che il ruolo giocato da **E** in **R** è **U**, questo significa che in ogni istanza dello schema, per ogni tupla istanza della relazione **R** che coinvolge l'istanza **e** di **E**, il ruolo “giocato” da **e** nella tupla è **U**



Sintassi: rappresentazione grafica di ruolo

- La necessità del ruolo è evidente nel caso di relazioni che coinvolgono la stessa entità più volte, mentre sembrerebbe non necessaria negli altri casi. Tuttavia, per uniformità, **si richiede che, per ogni relazione R dello schema, per ogni entità E coinvolta in R (e, se E è coinvolta in R più di una volta, per ogni volta in cui è coinvolta), si specifichi il ruolo giocato da E in R**
- Graficamente, per ogni collegamento tra una relazione ed una entità coinvolta in essa, noi **specificheremo il nome del ruolo vicino al collegamento stesso**
- Se l'esplicita specifica del nome del ruolo manca per un collegamento tra R ed E, allora **si assume che il nome del ruolo coincida con il nome della entità E**
- Ovviamente, ogni collegamento tra R ed una entità coinvolta in R deve essere identificata univocamente dal nome di ruolo



Semantica dei ruoli e delle relazioni (versione finale)

- L'impatto dei ruoli sul livello **estensionale** è tale da modificare la semantica delle relazioni in modo che le etichette usate nelle tuple non siano più le entità, ma i ruoli.
- In particolare, in ogni istanza **I** dello schema **S**, una relazione **R** tra le entità **E₁, E₂, ..., E_n** (non necessariamente tutte diverse tra loro) con rispettivi ruoli **U₁, U₂, ..., U_n** (tutti distinti) è costituita da un insieme *istanze(I, R)* in cui ogni elemento è una **tupla (o n-pla) etichettata** **(U₁:x₁, U₂:x₂, ..., U_n:x_n)**

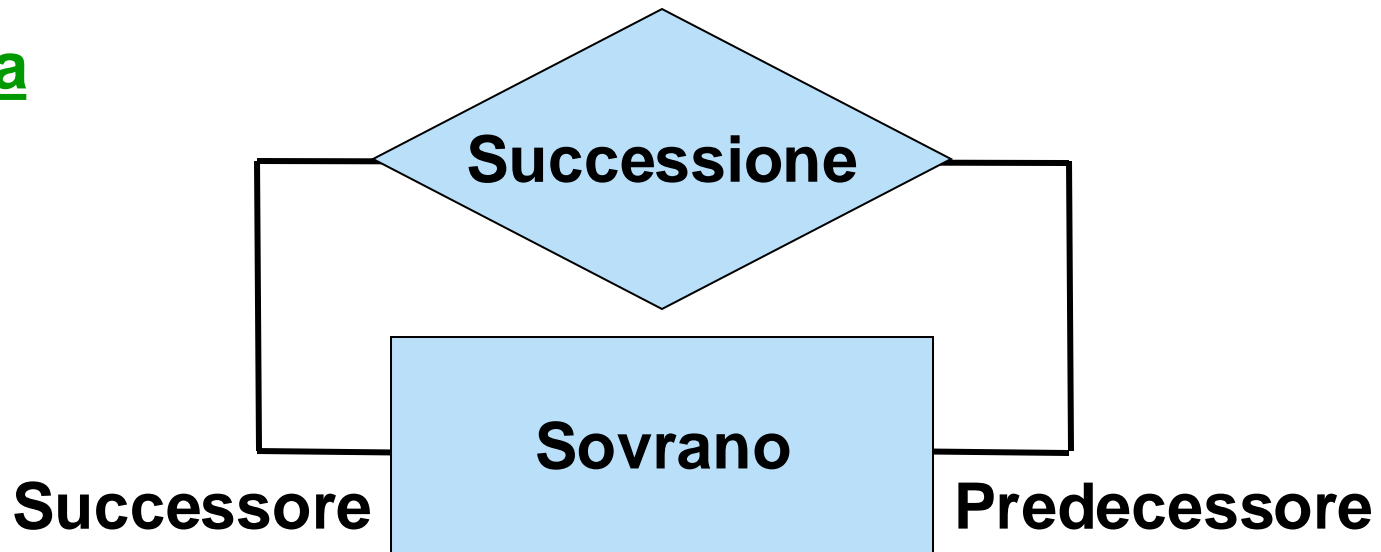
tale che **x₁** appartiene a *istanze(I, E₁)*, **x₂** appartiene a *istanze(I, E₂)*, ..., **x_n** appartiene a *istanze(I, E_n)*. Ogni n-pla etichettata è detta **istanza** della relazione **R** nella istanza **I** dello schema **S**

Nelle slide, quando ciò non genera ambiguità, ometteremo, per ragioni di spazio, le etichette nelle tuple etichettate



Semantica dei ruoli: esempio

Schema



Istanza I dello schema

$istanze(I, \text{Sovrano}) = \{ \text{romolo}, \text{numa}, \text{tulio}, \text{anco} \}$

$istanze(I, \text{Successione}) = \{ (\text{Precedessore:romolo}, \text{Successore:numa}),$
 $(\text{Precedessore:numa}, \text{Successore:tulio}),$
 $(\text{Precedessore:tulio}, \text{Successore:anco})$
 $\}$



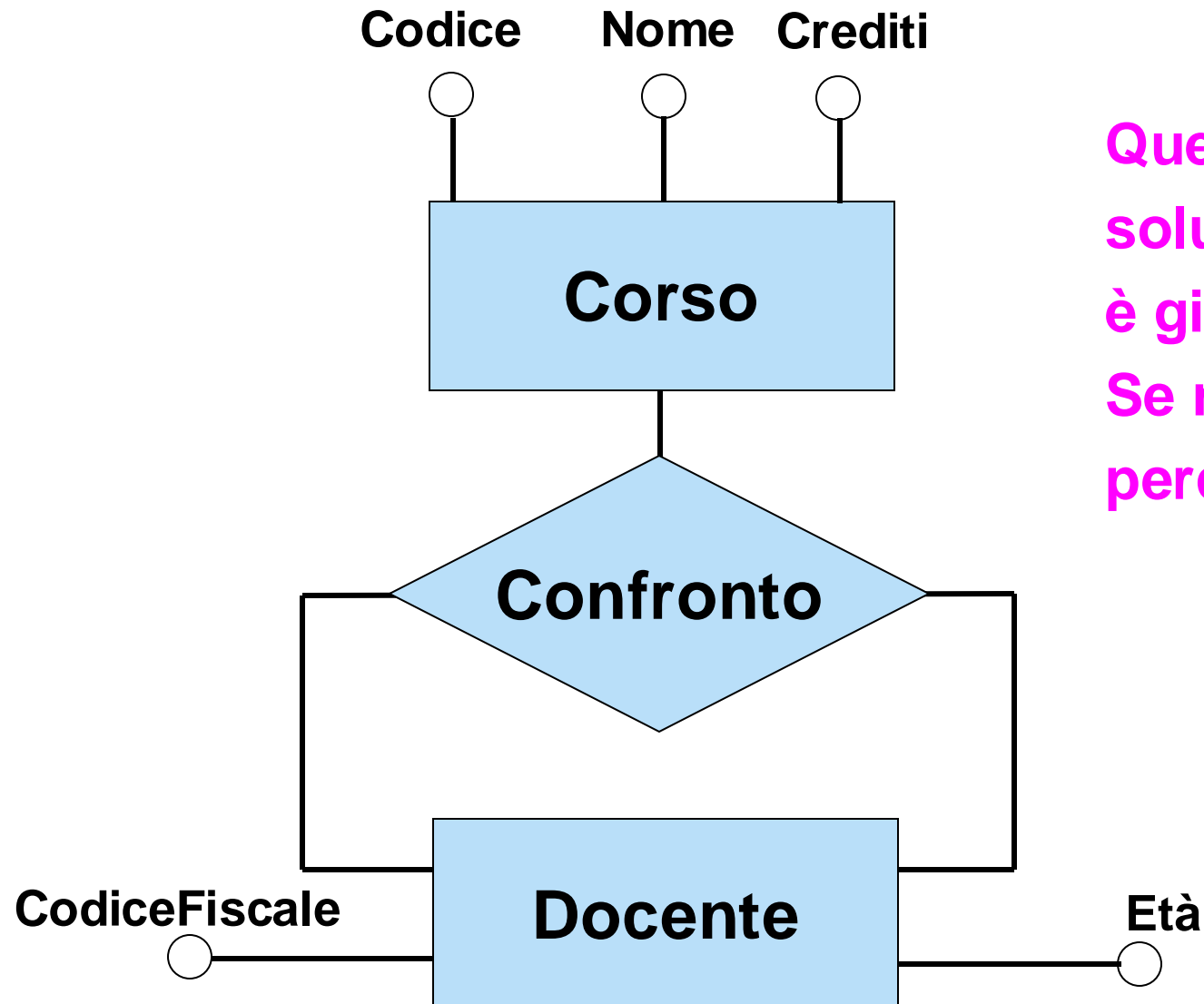
Esercizio 5: schema concettuale

Descrivere lo schema concettuale della seguente realtà:

I docenti hanno un codice fiscale ed una età. I corsi hanno un codice identificativo, un nome, ed il numero di crediti. Sfruttando i moduli di valutazione dei corsi e dei docenti da parte degli studenti, si vuole rappresentare nella base di dati l'informazione se un docente è migliore di un altro nell'insegnare un corso.



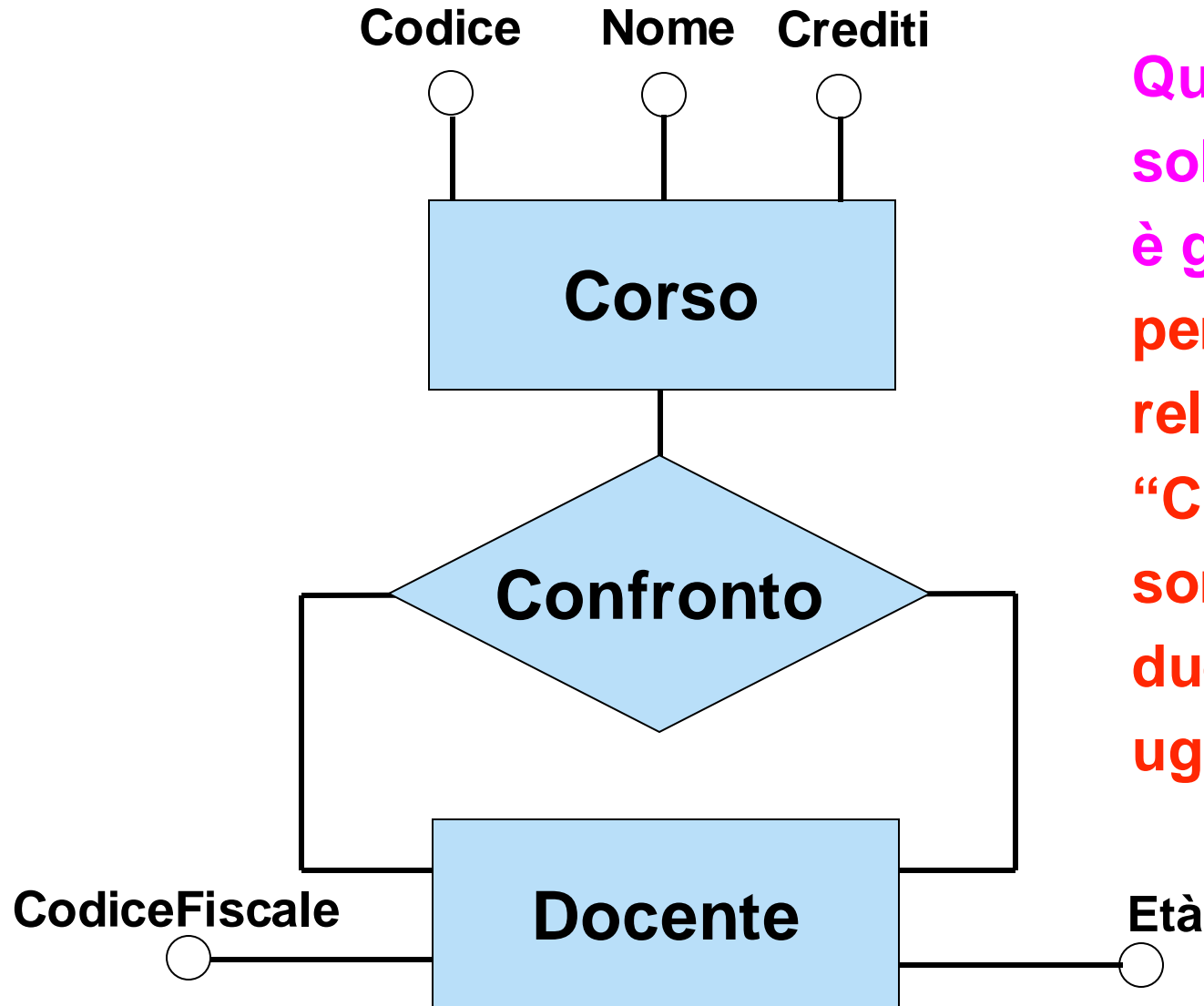
Esercizio 5: soluzione 1



**Questa
soluzione
è giusta?
Se no,
perché ?**



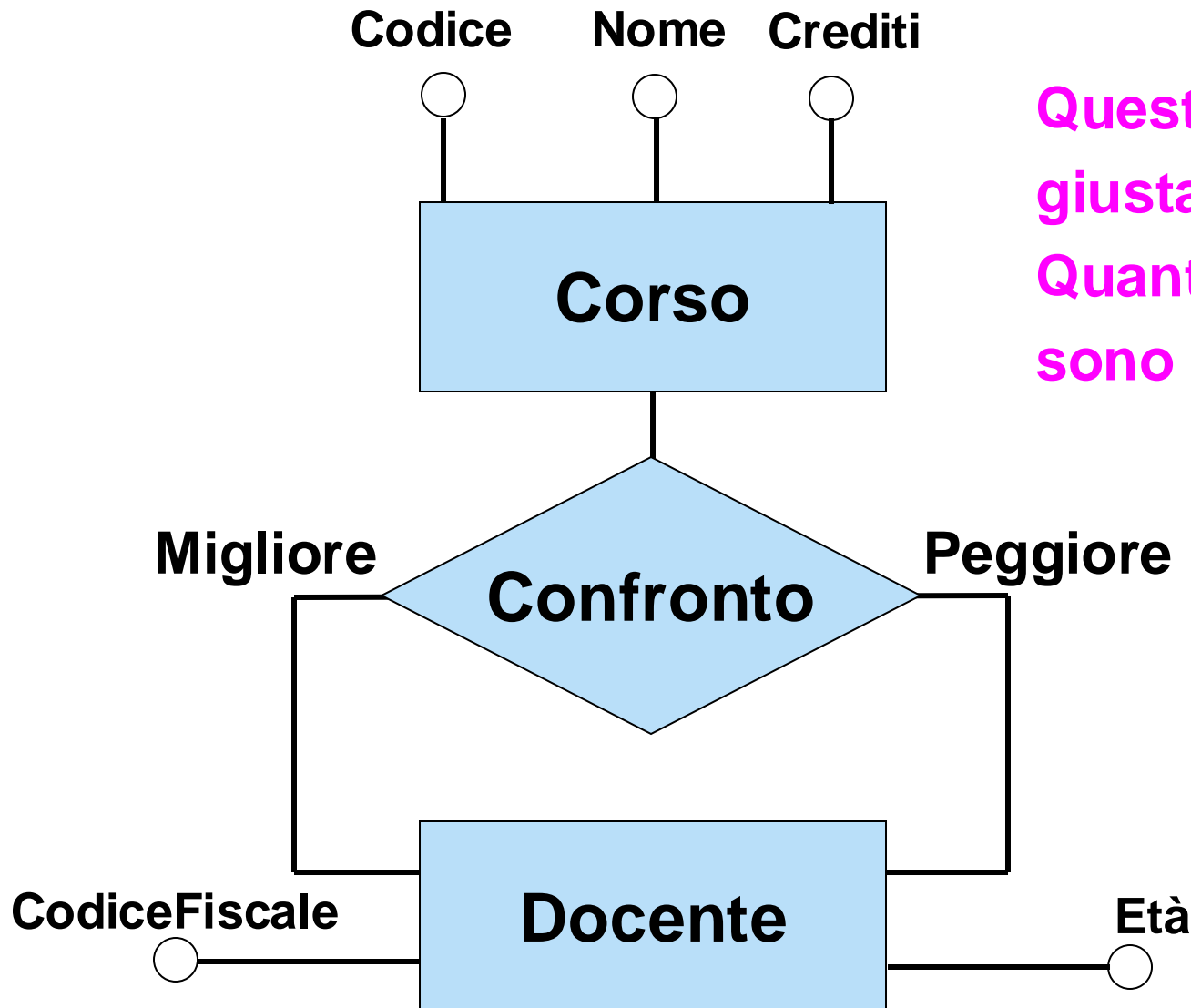
Esercizio 5: soluzione 1



**Questa
soluzione
è giusta? NO,
perché nella
relazione
“Confronto”
sono definiti
due ruoli
uguali**



Esercizio 5: soluzione 2

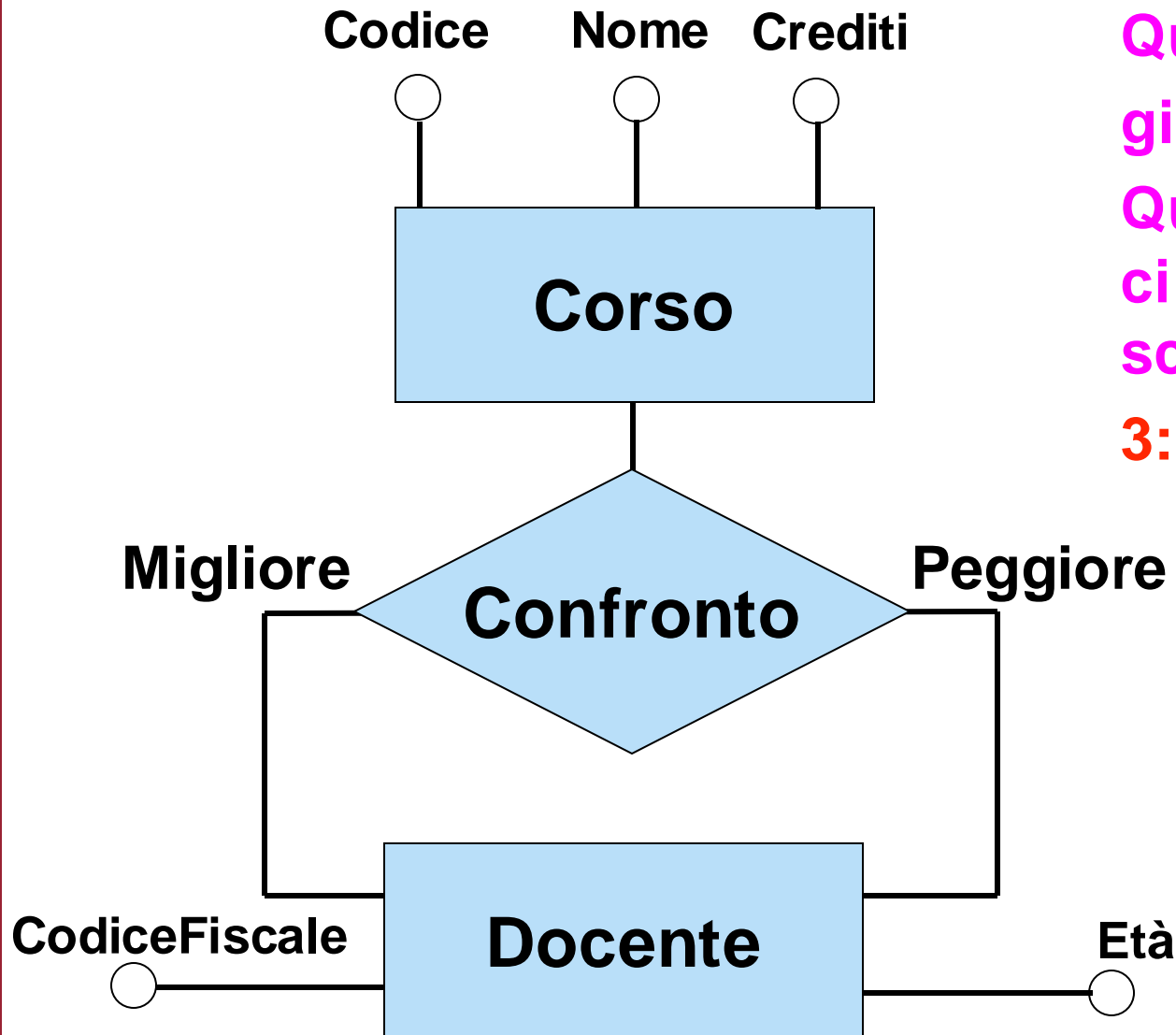


Questa soluzione è giusta?

Quanti e quali ruoli ci sono nello schema ?



Esercizio 5: soluzione 2



Questa soluzione è giusta? **SI'**

Quanti e quali ruoli ci sono nello schema ?

3: Corso, Migliore, Peggior



Esercizio 6: livello estensionale

Descrivere l'istanza dello schema concettuale precedente (soluzione 2) corrispondente ai seguenti dati (si ignorino gli attributi):

Docenti: Paolo, Mario, Grazia

Corsi: Basi di Dati, Logica, Algebra

Graduatorie:

Basi di dati

1. Mario
2. Grazia
3. Paolo

Logica

1. Grazia
2. Mario
3. Paolo

Algebra

1. Grazia
2. Paolo



Esercizio 6: soluzione

Istanza I dello schema corrispondente alla soluzione 2:

istanze(I,Docente): { Paolo, Mario, Grazia }

istanze(I,Corso): { BasiDiDati, Logica, Algebra }

istanze(I,Confronto):

{ (Corso:BasiDiDati, Migliore:Mario, Peggior: Grazia),
 (Corso:BasiDiDati, Migliore:Mario, Peggior: Paolo),
 (Corso:BasiDiDati, Migliore:Grazia, Peggior: Paolo),
 (Corso:Logica, Migliore:Grazia, Peggior: Mario),
 (Corso:Logica, Migliore:Grazia, Peggior: Paolo),
 (Corso:Logica, Migliore:Mario, Peggior: Paolo),
 (Corso:Algebra, Migliore:Grazia, Peggior: Paolo) }



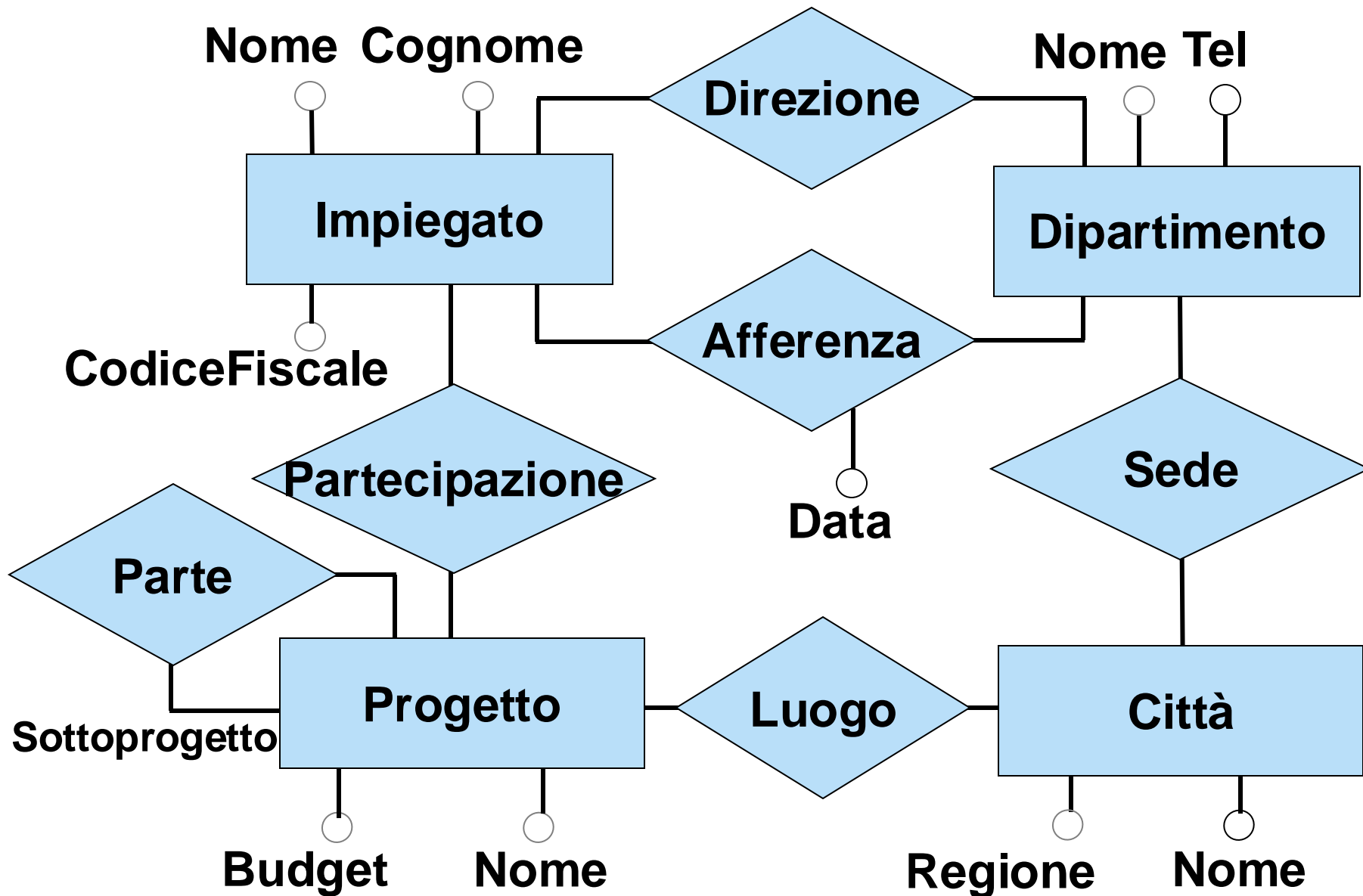
Esercizio 7: schema concettuale

Descrivere lo schema concettuale della seguente realtà:

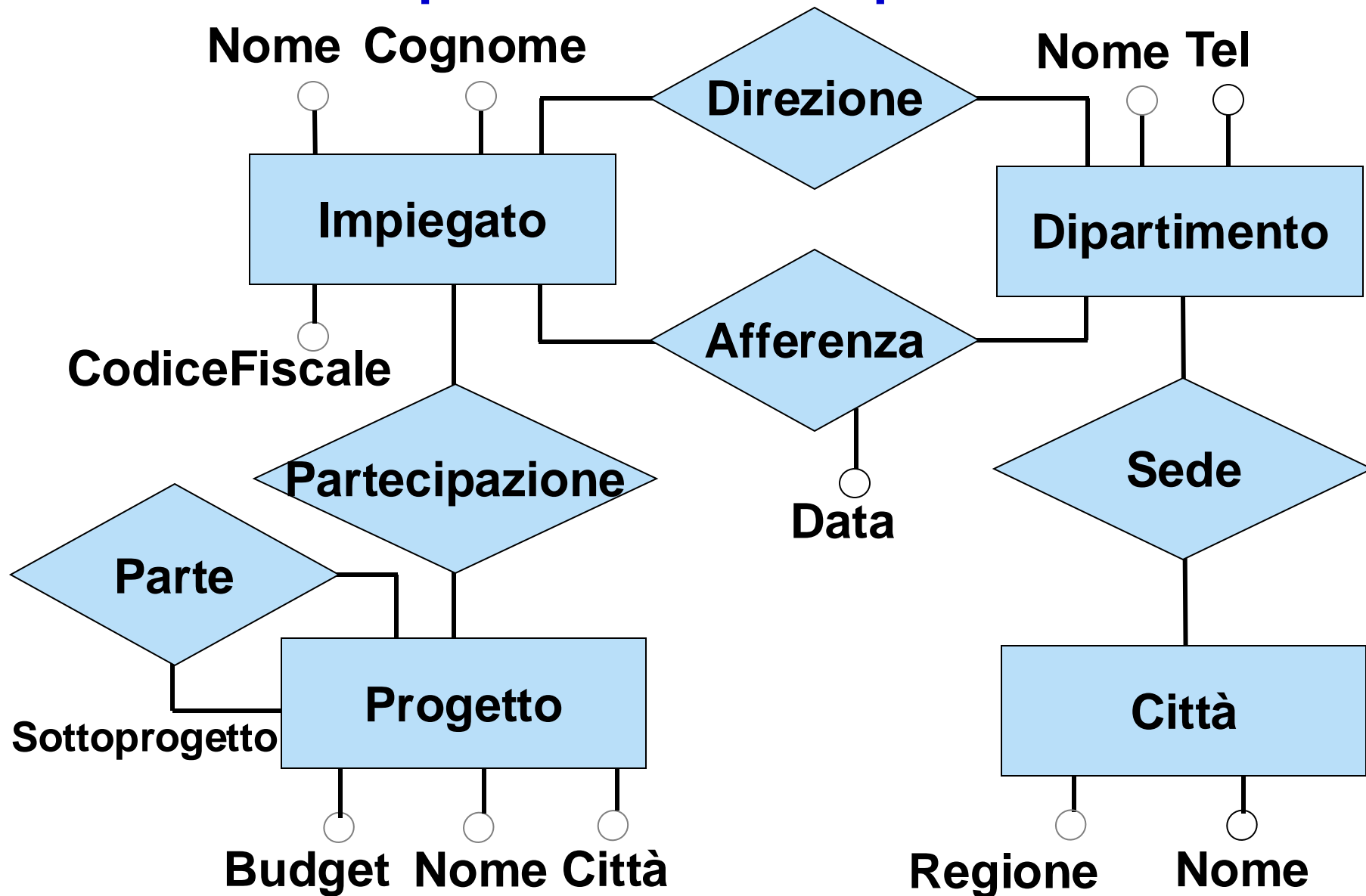
Degli impiegati interessa il codice fiscale, il nome, il cognome, i dipartimenti ai quali afferiscono (con la data di appartenenza), ed i progetti ai quali partecipano. Dei progetti interessa il nome, il budget, e la città in cui hanno luogo le corrispondenti attività. Alcuni progetti sono parti di altri progetti, e sono detti loro sottoprogetti. Dei dipartimenti interessa il nome, il numero di telefono, gli impiegati che li dirigono, e la città dove è localizzata la sede. Delle città interessa il nome e la regione.



Esercizio 7: soluzione



Esercizio 8: qual è l'errore in questo schema?





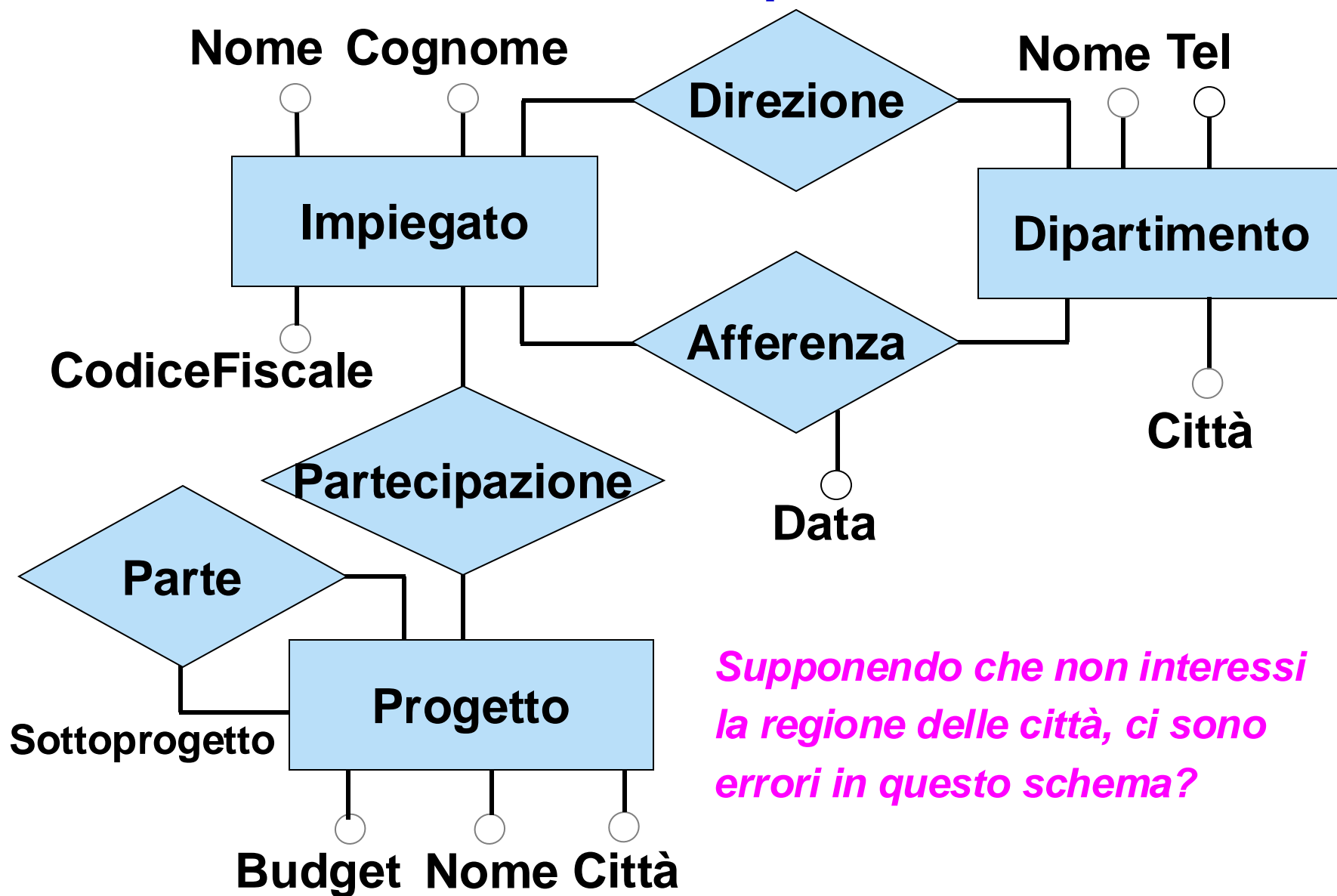
Esercizio 8: soluzione

L'errore sta nell'aver rappresentato il luogo di svolgimento di un progetto come un attributo. In realtà il luogo di svolgimento è una città, e questo viene perso nello schema.

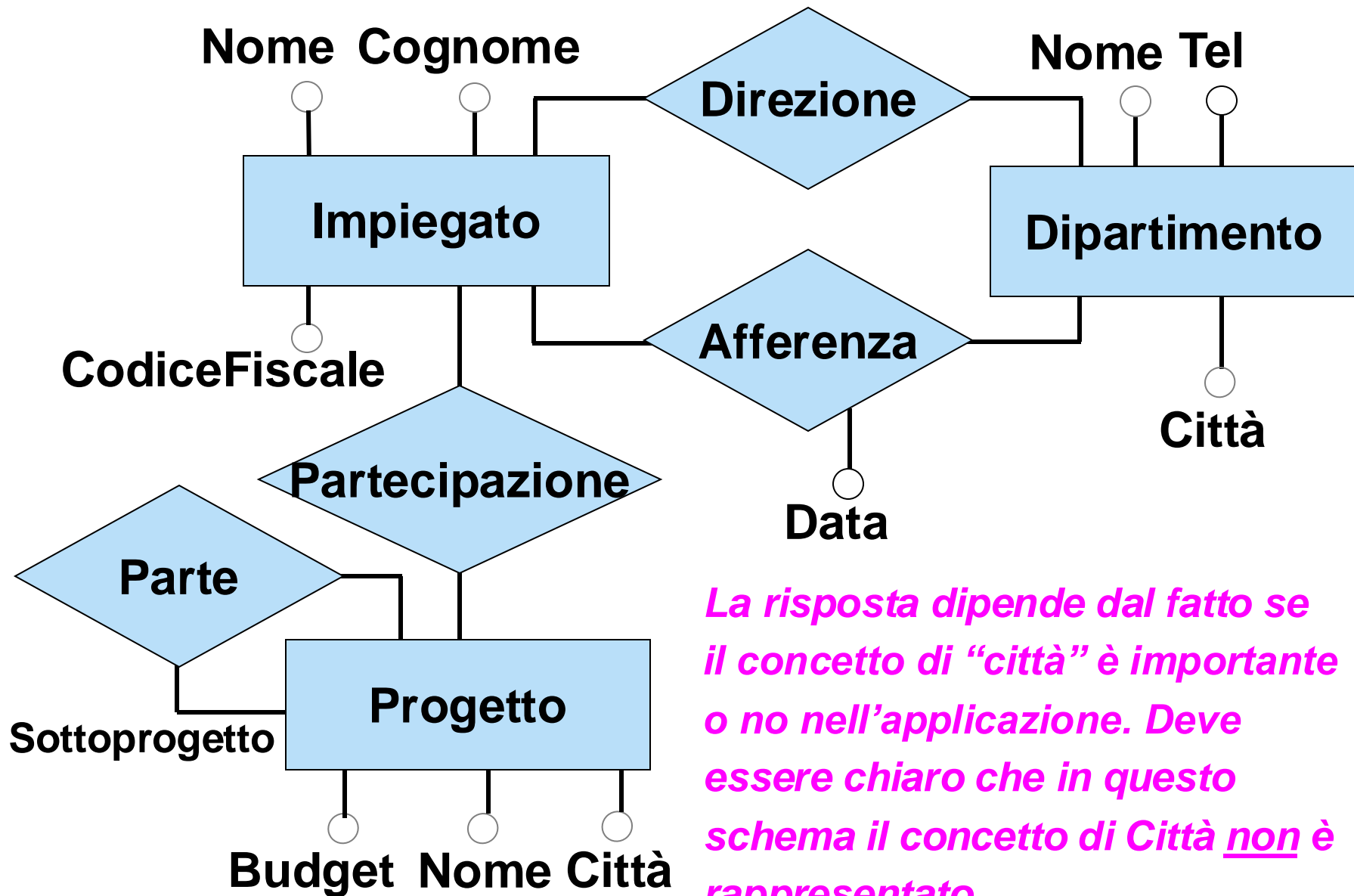
Infatti, il nome dell'attributo (Città) **non ha nulla a che vedere con la semantica** dello schema (che invece è caratterizzata dalle istanze dello schema): ciò che viene perso è che, in ogni istanza dello schema, il luogo di svolgimento di un progetto è una istanza dell'entità Città.

Detto in altro modo, nello schema il luogo di progetto è rappresentato come una proprietà locale di Progetto, perdendo così il legame con l'entità Città.

Esercizio 9: è corretto questo schema?



Esercizio 9: soluzione



La risposta dipende dal fatto se il concetto di “città” è importante o no nell’applicazione. Deve essere chiaro che in questo schema il concetto di Città non è rappresentato.



Scelta tra entità e attributo

Un concetto verrà modellato come

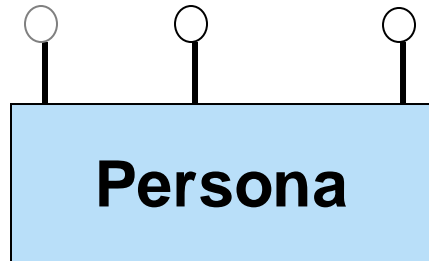
- una **entità**
 - se le sue istanze sono concettualmente significative indipendentemente da altre istanze
 - se ha o potrà avere delle proprietà indipendenti dagli altri concetti
 - se il concetto è importante nell'applicazione
- un **attributo** di una entità o relazione
 - se le sue istanze non sono concettualmente significative
 - se non ha senso considerare una sua istanza indipendentemente da altre istanze
 - se serve solo a rappresentare una proprietà locale di un altro concetto



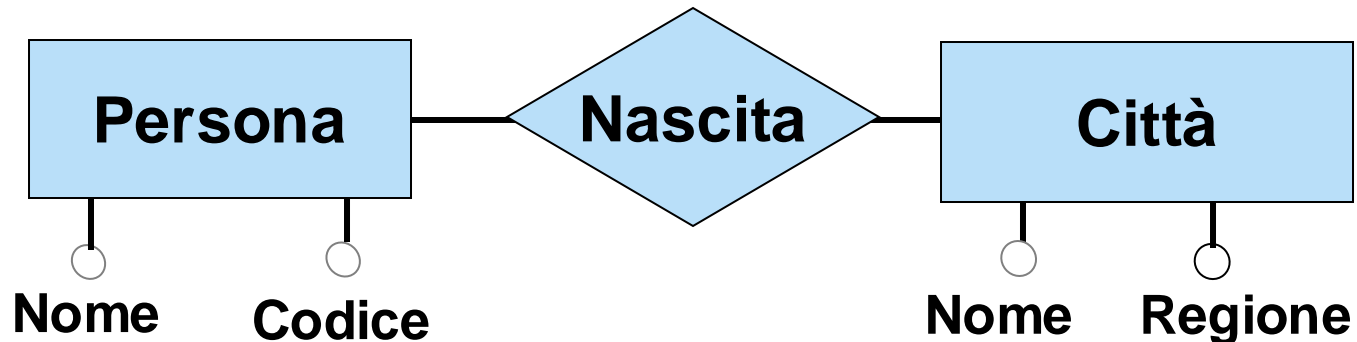
Scelta tra entità e attributo

Le scelte possono cambiare durante l'analisi. Esempio:

Nome Codice CittàNascita



Interessa anche la regione:
Città diventa una entità

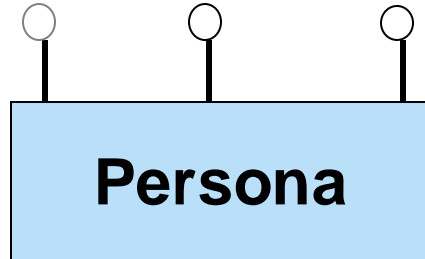




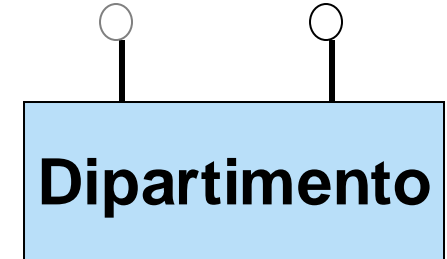
Scelta tra entità e attributo

Le scelte possono cambiare durante l'analisi. Esempio:

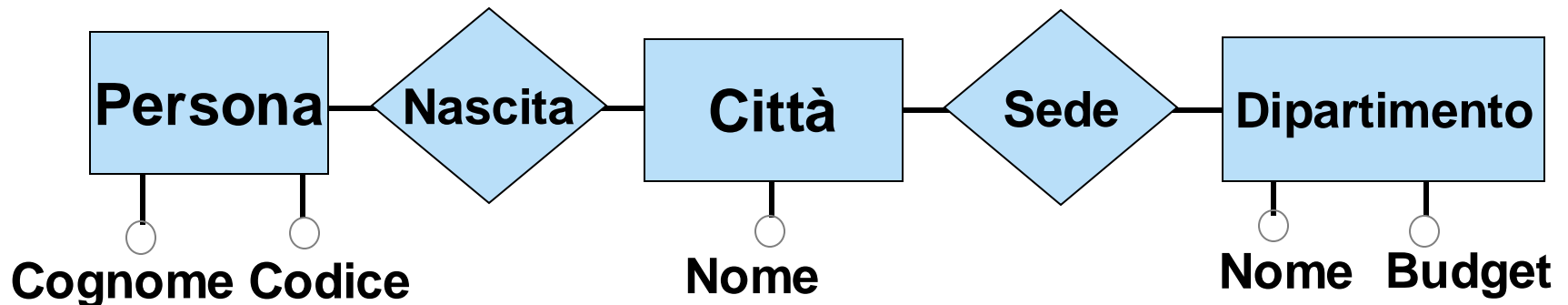
Cognome Codice CittàNascita



Nome Budget



Interessa anche la città in cui ha sede un dipartimento





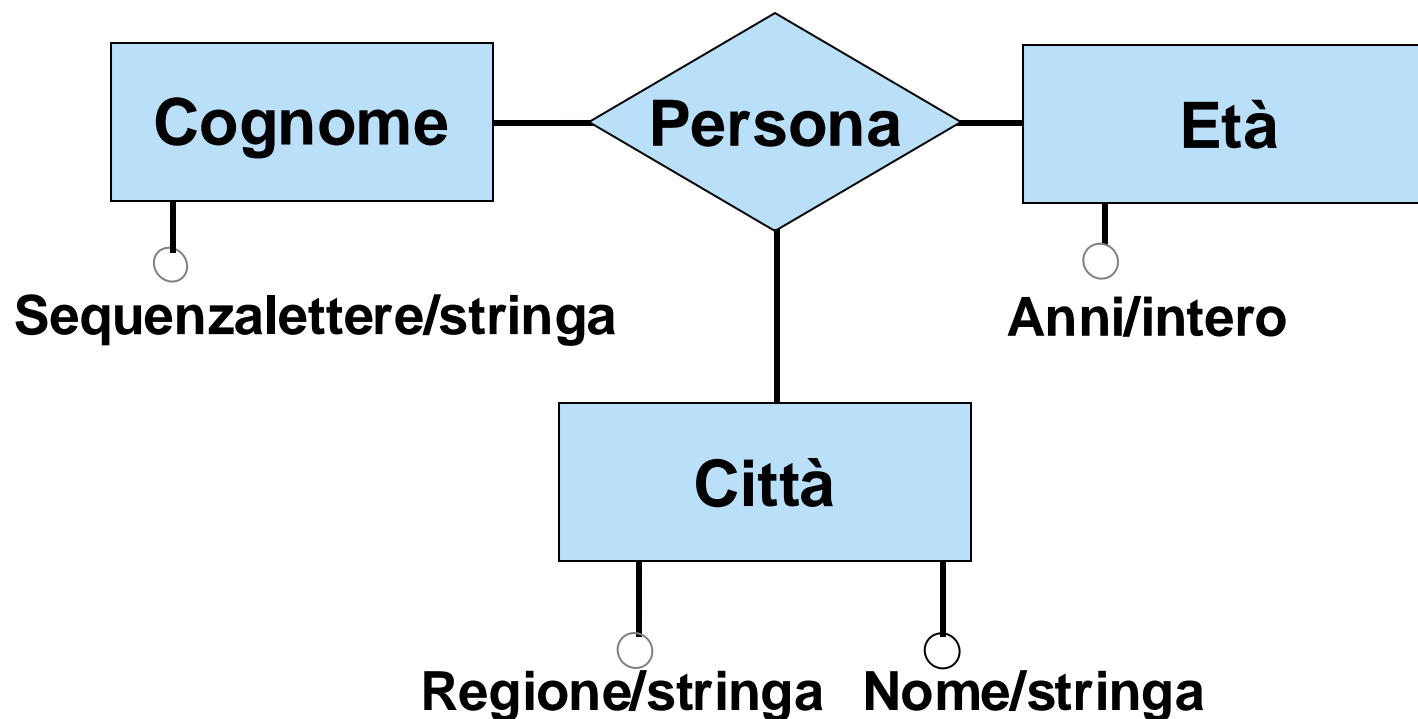
Scelta tra entità e relazione

Un concetto verrà modellato come

- una **entità**
 - se le sue istanze sono concettualmente significative indipendentemente da altre istanze
 - se ha o potrà avere delle proprietà indipendenti dagli altri concetti
 - se ha o potrà avere relazioni con altri concetti
- una **relazione**
 - se le sue istanze non sono concettualmente significative indipendentemente da altre istanze, cioè se le sue istanze rappresentano n-ple di altre istanze
 - se non ha senso pensare alla partecipazione delle sue istanze ad altre relazioni

Esercizio 10: scelta tra entità e relazione

In un contesto in cui occorre rappresentare le persone e le loro proprietà, questo schema è chiaramente sbagliato. Perché?





Esercizio 10: soluzione

- Nello schema, la classe delle persone non è modellato, nel senso che, a livello estensionale non esistono oggetti con significato autonomo che sono istanze di persone
- A livello estensionale ogni persona sarà quindi rappresentata da una tupla, che, per definizione, esiste solo perchè esistono le sue componenti. È come dire che una persona esiste solo perchè ha un cognome, una età, ed una città di nascita
- Modellando Cognome come entità, lo schema asserisce implicitamente che i cognomi sono oggetti la cui esistenza autonoma è di interesse per la nostra applicazione, e questo è chiaramente assurdo. Stessa considerazione vale per Età
- Infine, per la semantica delle relazioni, lo schema asserisce implicitamente che non esistono due persone con lo stesso cognome, la stessa età e la stessa città di nascita, e anche questo è assurdo.



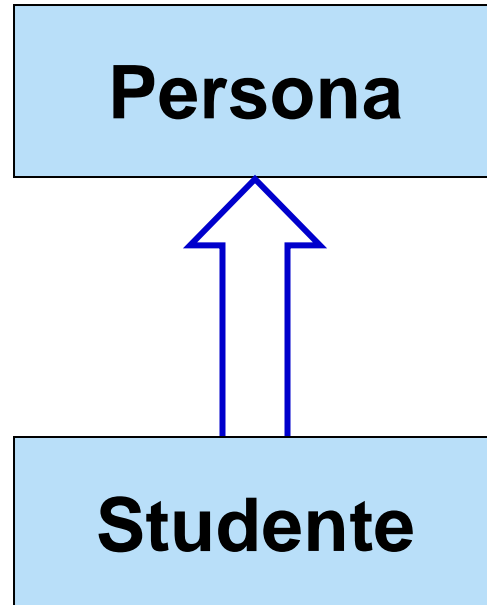
La nozione di relazione ISA tra entità

- Fino ad ora non abbiamo detto alcunché sul fatto se due entità possano o no avere istanze in comune. Tuttavia, negli esempi che abbiamo proposto era in qualche modo implicito il fatto che per ogni coppia di entità, non vi fossero istanze in comune tra esse.
- È facile verificare che, in molti contesti, può invece accadere che tra due classi rappresentate da due entità nello schema concettuale sussista la **relazione ISA** (o **relazione di sottoinsieme**), e cioè che ogni istanza di una sia anche istanza dell'altra.
- La relazione ISA nel modello ER si può definire tra due entità, che si dicono “entità padre” ed “entità figlia” (o sottoentità, cioè quella che rappresenta un sottoinsieme della entità padre)

Sintassi: rappresentazione grafica della relazione ISA tra entità

La relazione ISA si rappresenta nel diagramma dello schema concettuale mediante una freccia dalla sottoentità alla entità padre

Esempio

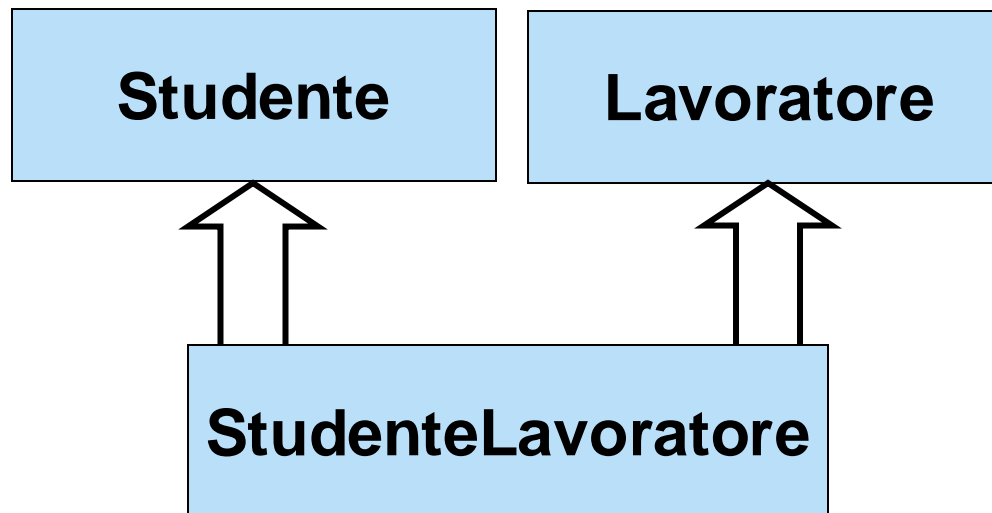


*Si dice che **Studente** è in relazione ISA con **Persona**, o, in alternativa, che **Studente ISA Persona***

Sintassi: regole sulla relazione ISA tra entità

ATTENZIONE: vige la regola che una entità può avere al **massimo una** entità padre. In altre parole, il modello ER **non** ammette **ereditarietà multipla**

Esempio



*Questo schema è
scorretto*

Semantica della relazione ISA tra entità

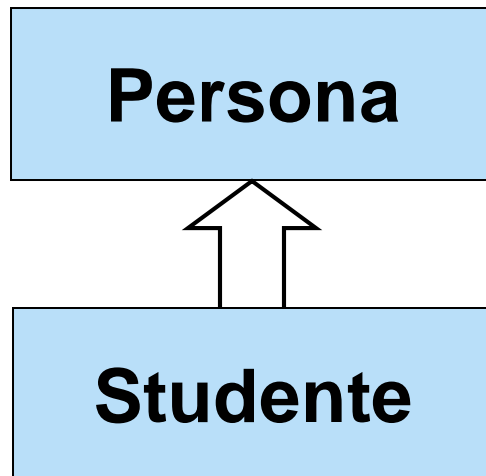
A livello estensionale la relazione ISA impone semplicemente che in ogni istanza **I** di uno schema **S** in cui **E₁** ISA **E₂**:

$$istanze(I, E_1) \subseteq istanze(I, E_2)$$

Si noti che dalla definizione segue che la relazione ISA è riflessiva e transitiva.

Esempio

Schema



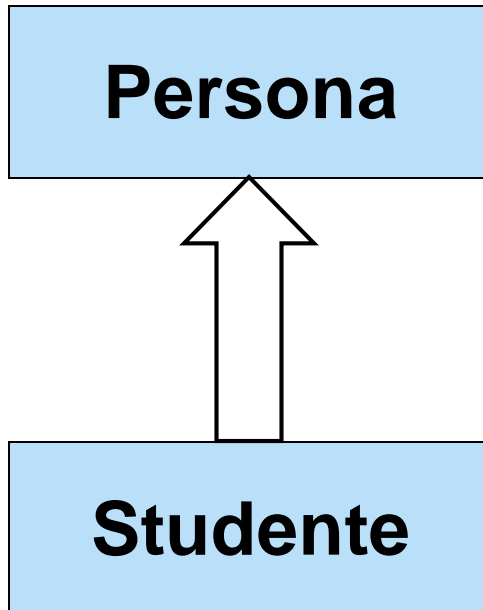
Istanza I:

$istanze(I, \text{Persona}) = \{ a, b, c, d \}$

$istanze(I, \text{Studente}) = \{ b, d \}$

Relazione ISA a livello estensionale

Schema



Istanza I1:

istanze(I1,Persona) = { a,b,c,d }

istanze(I1,Studente) = { b,d }

Istanza I2:

istanze(I2,Persona) = { a,b,c,d }

istanze(I2,Studente) = {a,b,c,d }

Questa non è un'istanza:

istanze(J,Persona) = { a }

istanze(J,Studente) = { a,b }

Istanza I3:

istanze(I3,Persona) = { a,b,c,d }

istanze(I3,Studente) = { }

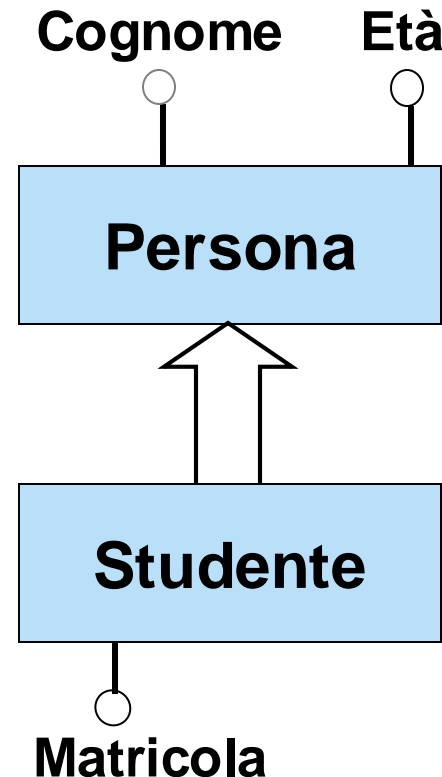
Ereditarietà su entità nel modello ER

Principio di ereditarietà: ogni proprietà dell'entità padre è anche una proprietà della sottoentità, e non si riporta esplicitamente nel diagramma. L'entità figlia può avere ovviamente ulteriori proprietà

Dal fatto che

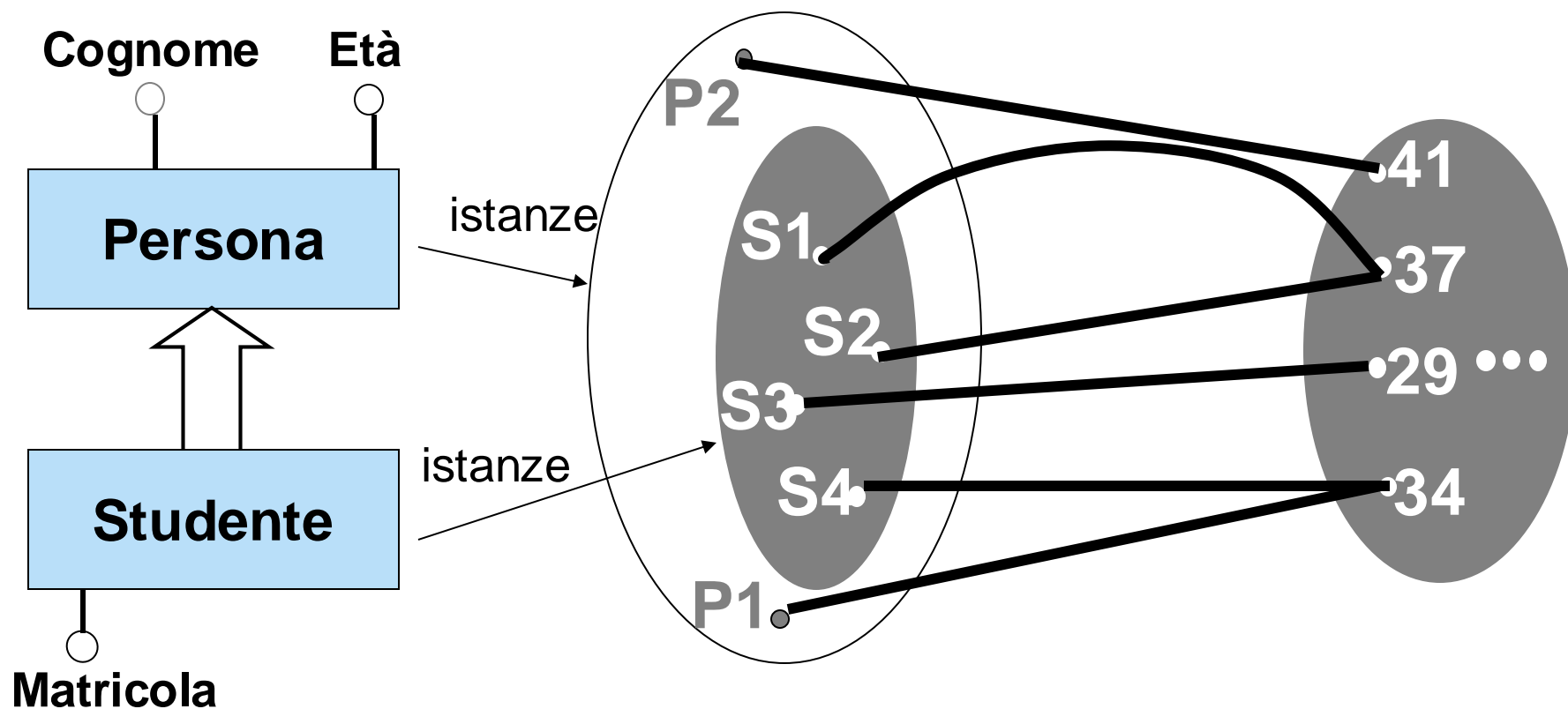
1. Ogni istanza di Persona ha una Età
2. Ogni istanza di Studente è istanza di Persona
- segue logicamente che
3. Ogni istanza di Studente ha una Età

*Ragionamento sillogistico
(cfr. opera di Aristotele più di
due millenni fa)*



**Cognome,
Età ereditati
da Persona,
Matricola
ulteriore
proprietà**

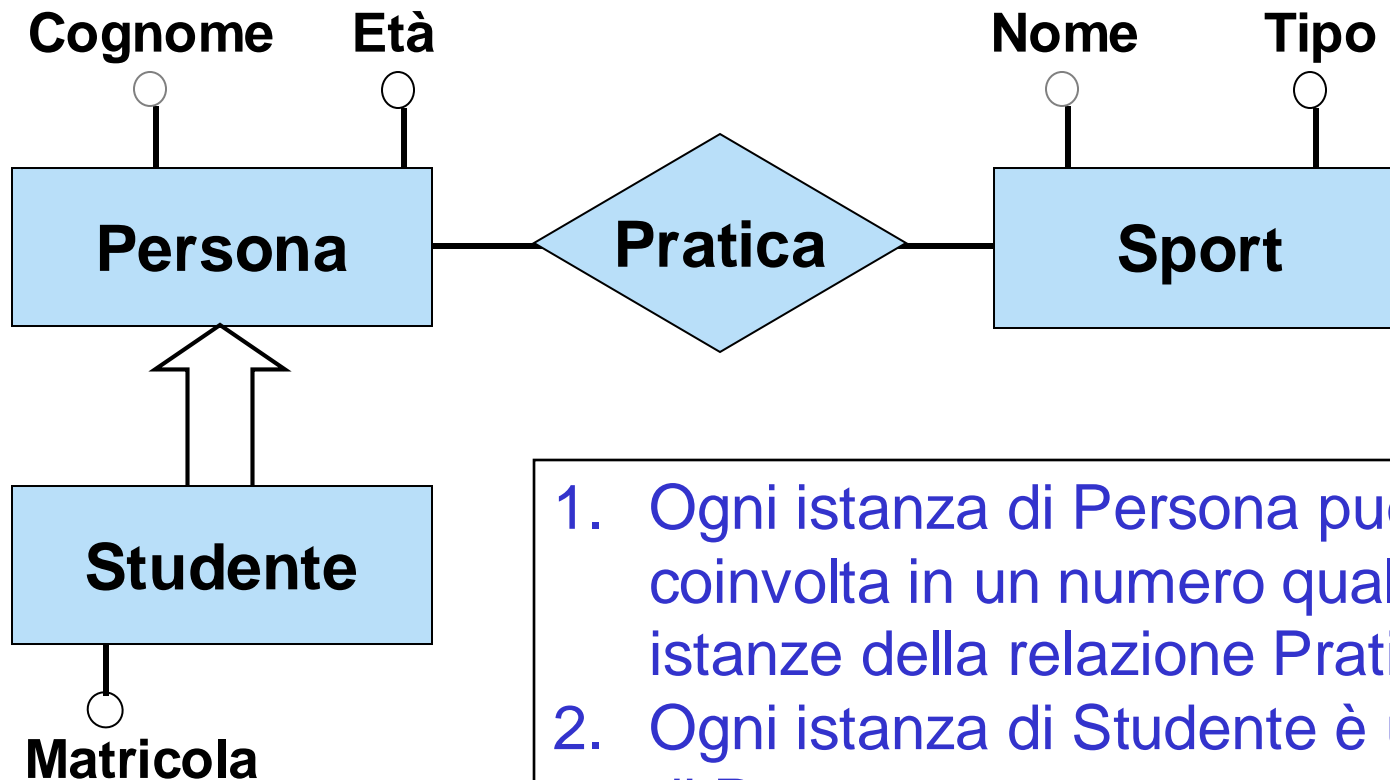
Ereditarietà: livello estensionale



*L'entità **Studente** eredita l'attributo **Età** dalla entità padre **Persona**. Ciò significa che ogni studente ha una età: infatti, se esistesse una istanza di **Studente** che non ha un valore per l'attributo **Età**, quella sarebbe una istanza di **Persona** senza un valore per **Età**, contraddicendo il significato dell'attributo **Età** (funzione totale) per **Persona***



Ereditarietà delle relazioni



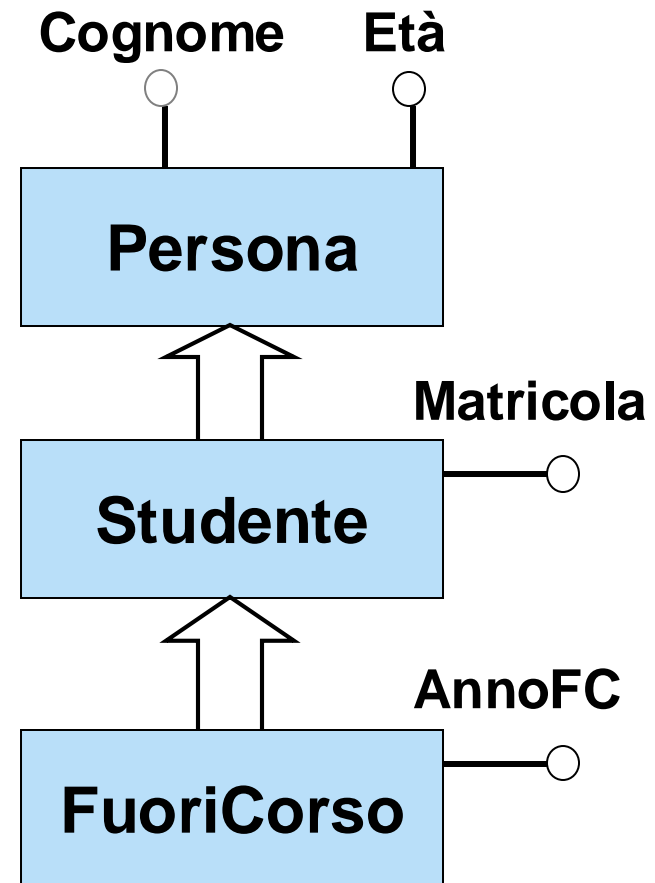
1. Ogni istanza di Persona può essere coinvolta in un numero qualunque di istanze della relazione Pratica
2. Ogni istanza di Studente è una istanza di Persona
- quindi**
3. Ogni istanza di Studente può essere coinvolta in un numero qualunque di istanze della relazione Pratica

Ereditarietà della relazione ISA: transitività

Principio di ereditarietà: anche la relazione ISA si eredita. Questo conferma che la relazione ISA è transitiva (oltre a essere riflessiva)

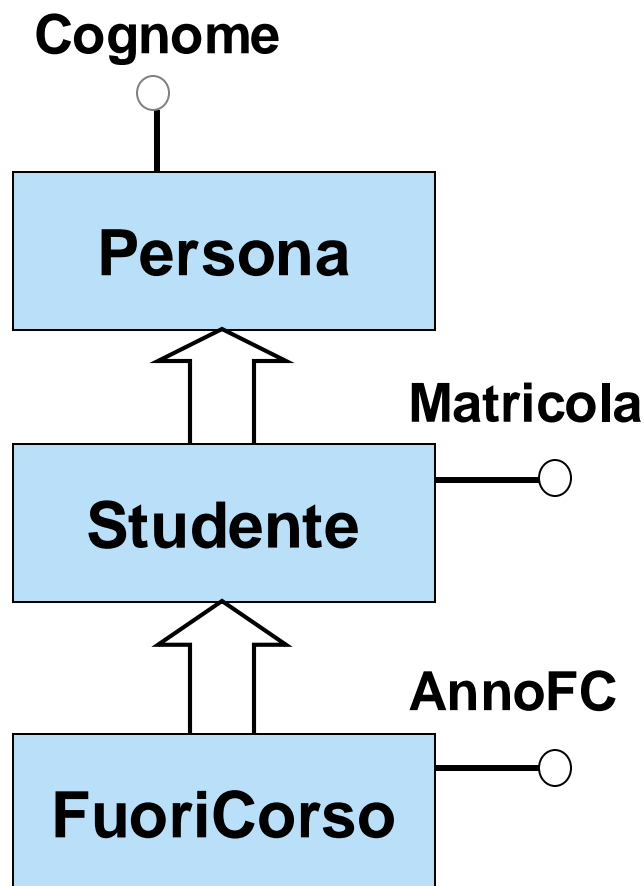
Dal fatto che

1. Ogni istanza di **Studente** è una istanza di **Persona**
 2. Ogni istanza di **FuoriCorso** è una istanza di **Studente**
- segue logicamente che**
3. Ogni istanza di **FuoriCorso** è una istanza di **Persona**



Esercizio 11: I1 è una istanza dello schema?

Schema



I1:

istanze(I1,Persona) = { a,b,d }

istanze(I1,Studente) = { b,d }

istanze(I1,FuoriCorso) = { b,d }

istanze(I1,Cognome) =
{ (a,'a'), (b,'b'), (d,'d') }

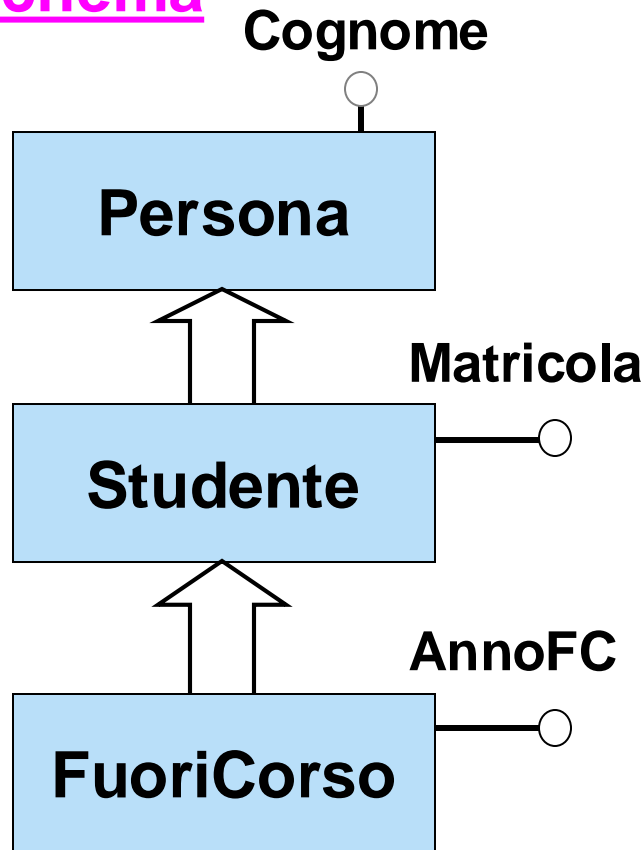
istanze(I1,Matricola) =
{ (b,'12'), (d,'34'), (a,'89') }

istanze(I1,AnnoFC) = { (b,2) }

Esercizio 11: soluzione

La risposta è NO, perchè: - (a,'89') in Matricola
- manca AnnoFC per d

Schema



I1:

istanze(I1,Persona) = { a,b,d }

istanze(I1,Studente) = { b,d }

istanze(I1,FuoriCorso) = { b,d }

*istanze(I1,Cognome) =
{ (a,'a'), (b,'b'), (d,'d') }*

*istanze(I1,Matricola) =
{ (b,'12'), (d,'34'), (a,'89') }*

istanze(I1,AnnoFC) = { (b,2) }



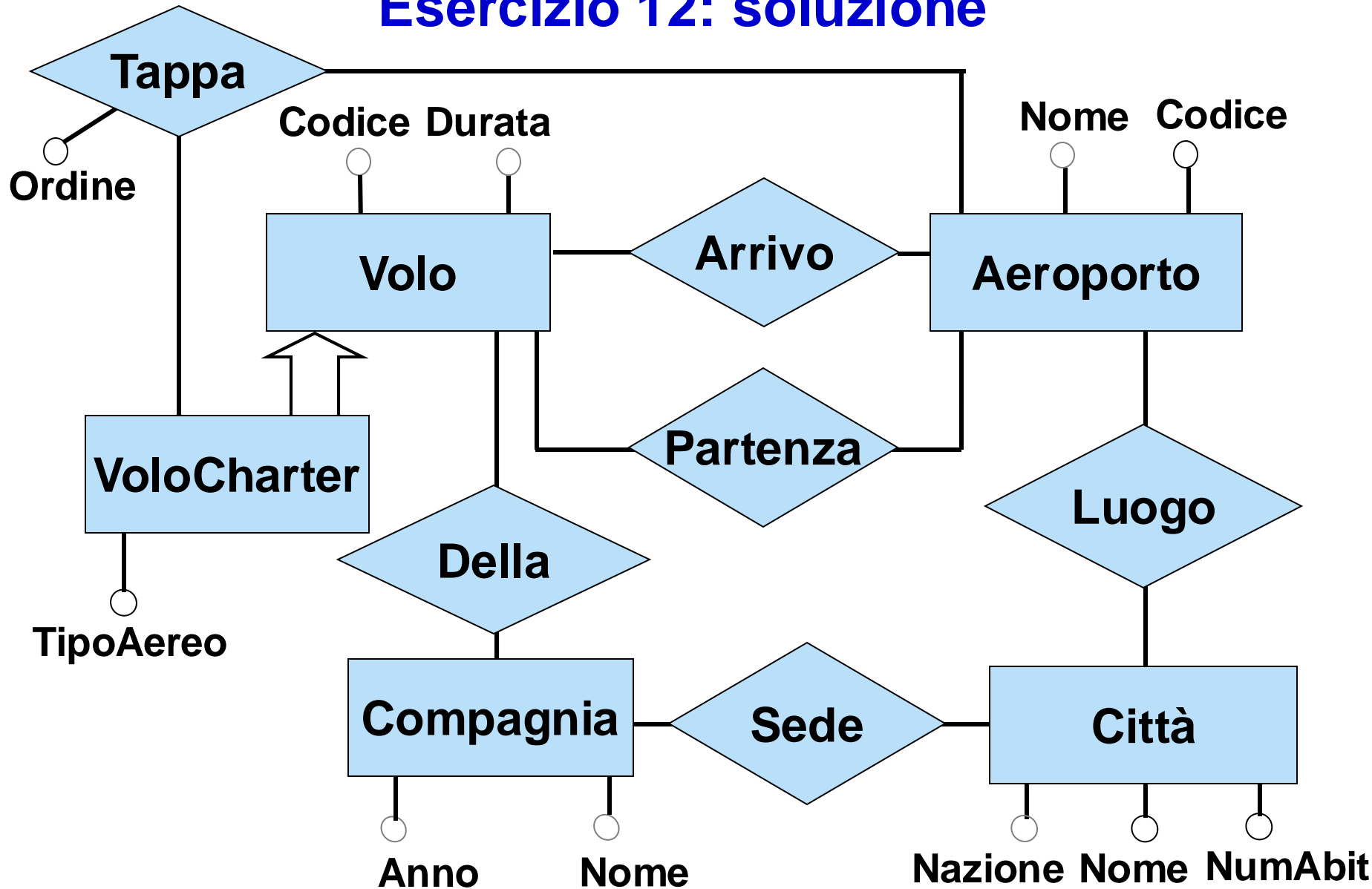
Esercizio 12: schema concettuale

Descrivere lo schema concettuale corrispondente ad un'applicazione riguardante voli aerei, per la quale valgono le seguenti specifiche.

Dei voli interessa: codice, durata in minuti, compagnia aerea, aeroporto di partenza e aeroporto di arrivo. Degli aeroporti interessa: codice, nome, città (con nome e numero di abitanti) e nazione. Delle compagnie aeree interessa il nome, l'anno di fondazione, e la città in cui ha sede la direzione. I voli charter sono particolari voli, che possono prevedere tappe intermedie in aeroporti. Delle tappe intermedie di un volo charter interessa l'ordine con cui esse si susseguono (ad esempio, il volo 124, che parte da "Milano Linate" e arriva a "Palermo Punta Raisi", prevede prima l'aeroporto di Bologna e poi quello di Napoli come tappe intermedie). Infine, dei voli charter interessa anche il tipo di aereomobile utilizzato per il volo.

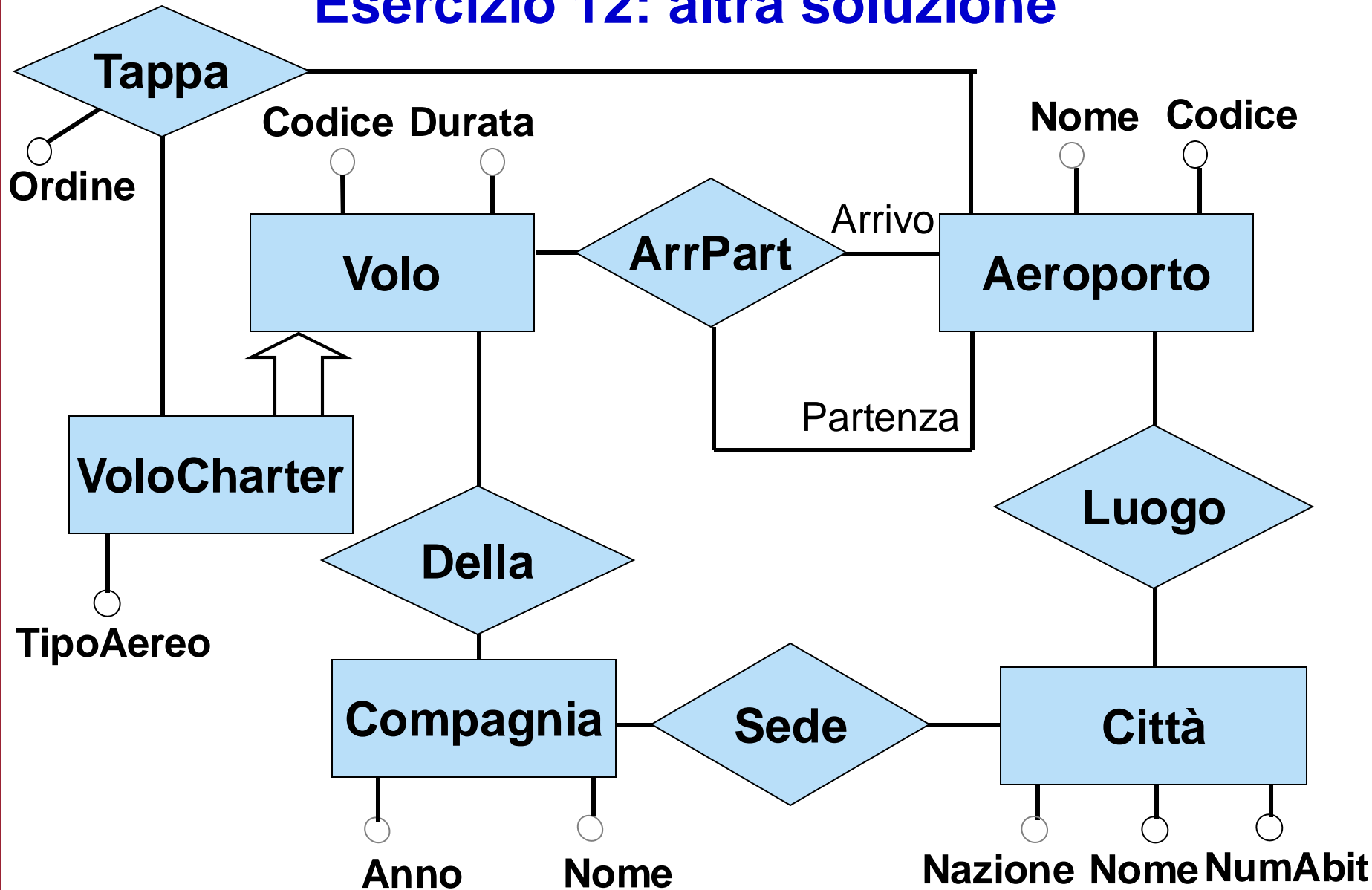


Esercizio 12: soluzione





Esercizio 12: altra soluzione

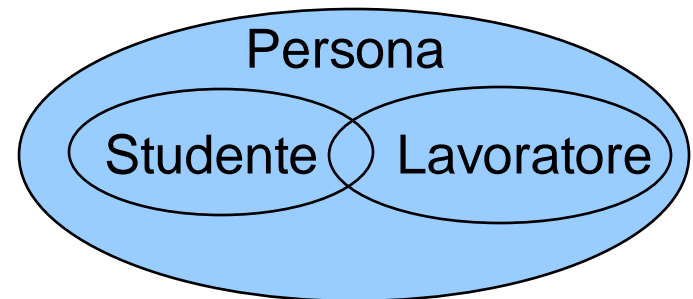
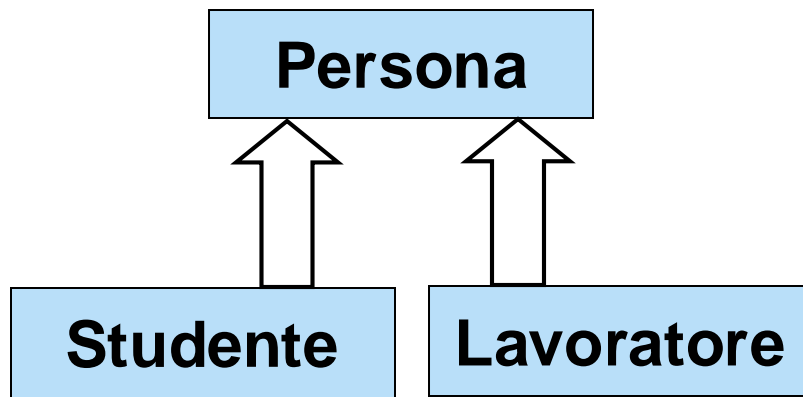




Diverse figlie di una entità

- Una entità può avere ovviamente diverse entità figlie
- Le istanze di due entità che sono figlie della stessa entità **possono avere istanze in comune**
- Questo significa che, al contrario di UML, nel modello ER uno stesso oggetto può essere istanza di diverse classi più specifiche

Esempio:





La nozione di generalizzazione tra entità

Finora, abbiamo considerato la relazione ISA che stabilisce che l'entità padre è più generale della sottoentità. Talvolta, però, l'entità padre può generalizzare diverse sottoentità rispetto ad un unico criterio. In questo caso si parla di **generalizzazione**.

Nella generalizzazione, le sottoentità hanno insiemi di istanze disgiunti a coppie (anche se in alcune varianti del modello ER, si può specificare se due sottoentità della stessa entità padre sono disgiunte o no).

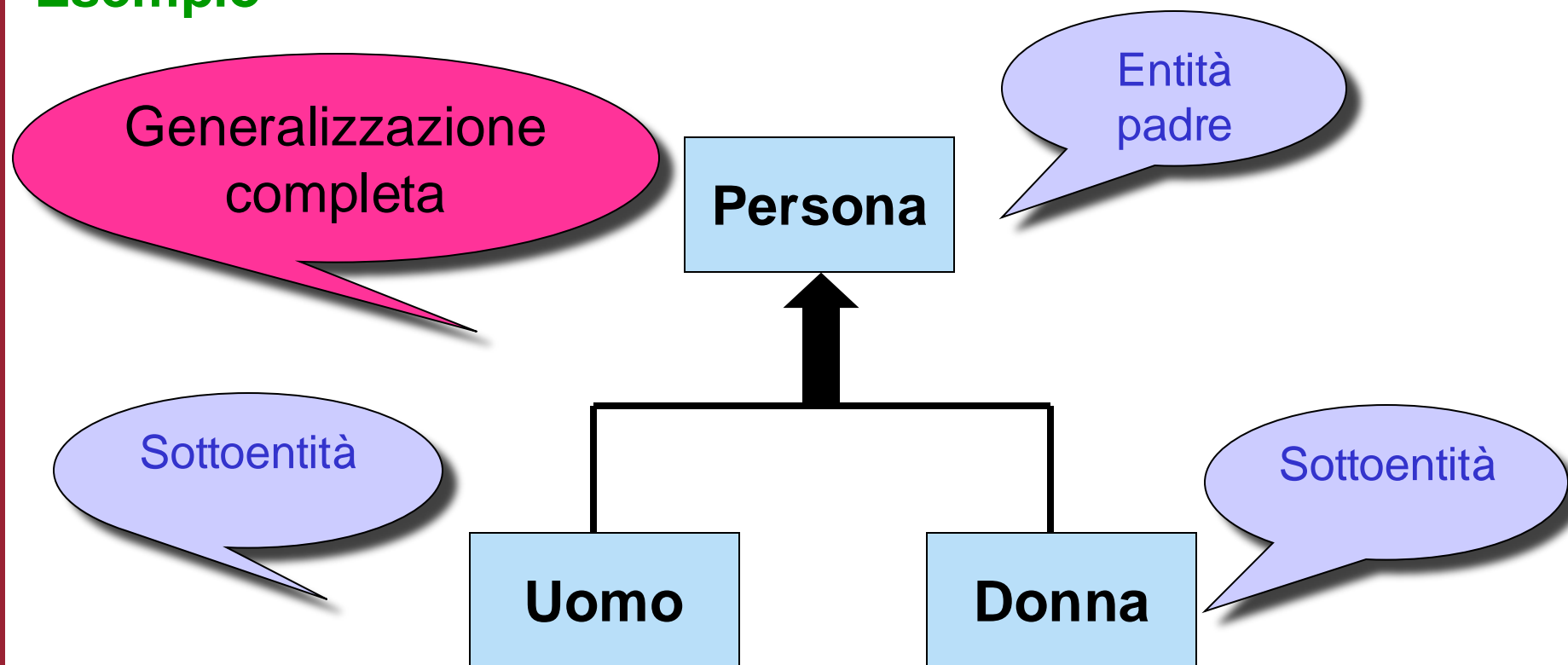
Una generalizzazione può essere di due tipi:

- **Completa**: l'unione delle istanze delle sottoentità è uguale all'insieme delle istanze dell'entità padre
- **Non completa**

Sintassi: rappresentazione della generalizzazione

La generalizzazione si indica collegando mediante un arco le sottoentità, e collegando con una freccia tale arco alla entità padre. La freccia è annerita se la generalizzazione è completa.

Esempio

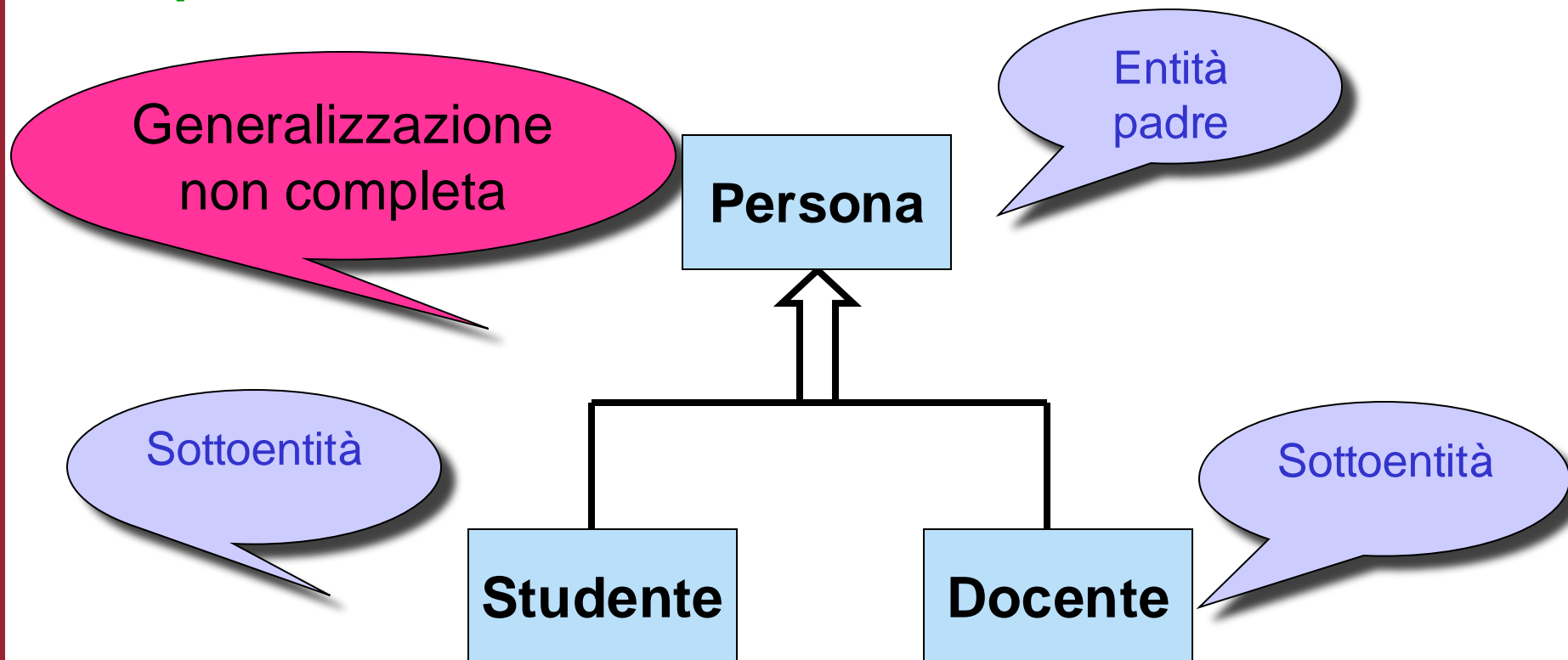




Sintassi: rappresentazione della generalizzazione

La freccia è non è annerita se la generalizzazione non è completa.

Esempio





Semantica della generalizzazione

Se in uno schema **S** è definita una generalizzazione tra una entità padre **F** e le sottoentità **E₁**, **E₂**, ..., **E_n**, in ogni istanza **I** dello schema **S** si ha:

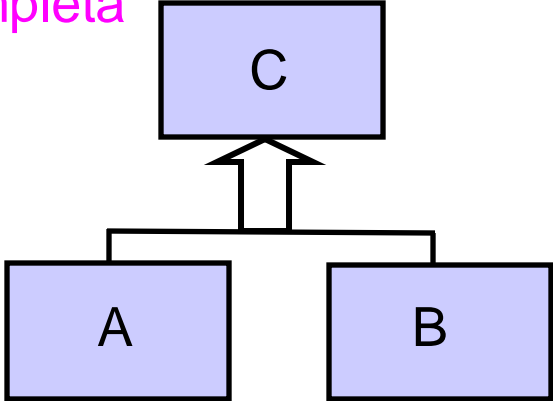
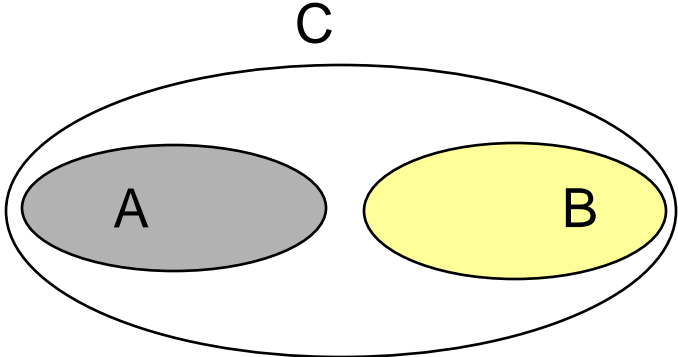
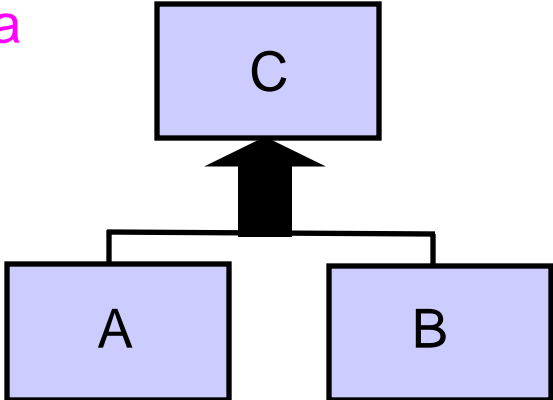
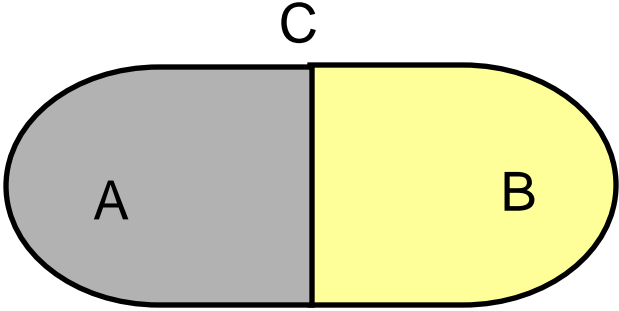
- $istanze(I, E_1) \subseteq istanze(I, F)$
- $istanze(I, E_2) \subseteq istanze(I, F)$
-
- $istanze(I, E_n) \subseteq istanze(I, F)$
- $istanze(I, E_i) \cap istanze(I, E_k) = \emptyset$, per ogni $1 \leq i, k \leq n, i \neq k$

In più, se la generalizzazione è completa, si ha che:

- $istanze(I, E_1) \cup \dots \cup istanze(I, E_n) = istanze(I, F)$

Si noti che ciò implica che in una generalizzazione ogni **E_i** è in relazione ISA con **F**, e che in una generalizzazione completa le sottoentità formano **partizione** dell'entità padre.

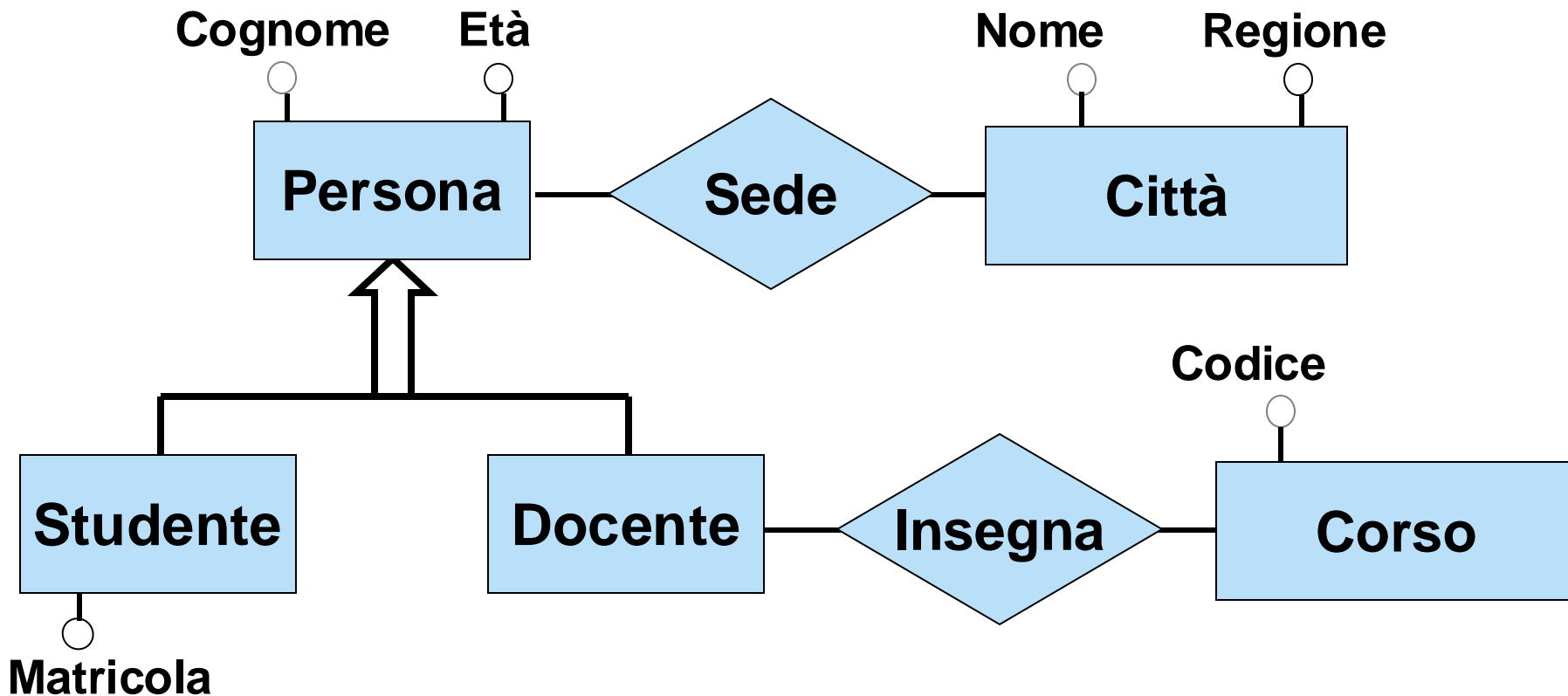
Generalizzazioni a livello estensionale

Livello intensionale	Livello estensionale
<p>non completa</p> 	
<p>completa</p> 	



Generalizzazioni ed ereditarietà

Il principio di ereditarietà vale anche per le generalizzazioni: ogni proprietà dell'entità padre è anche una proprietà della sottoentità, e non si riporta esplicitamente nel diagramma. L'entità figlia può avere ovviamente ulteriori proprietà.

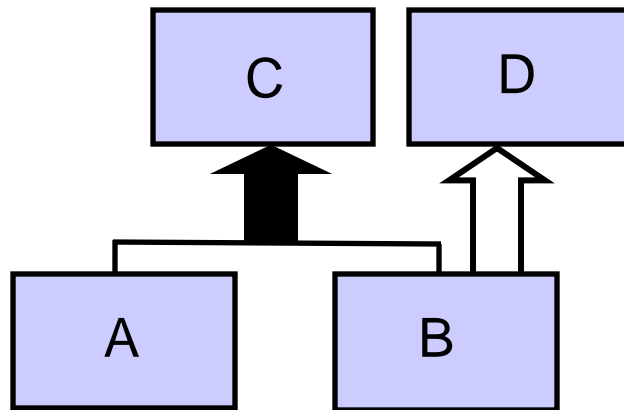




Regole sulla generalizzazione

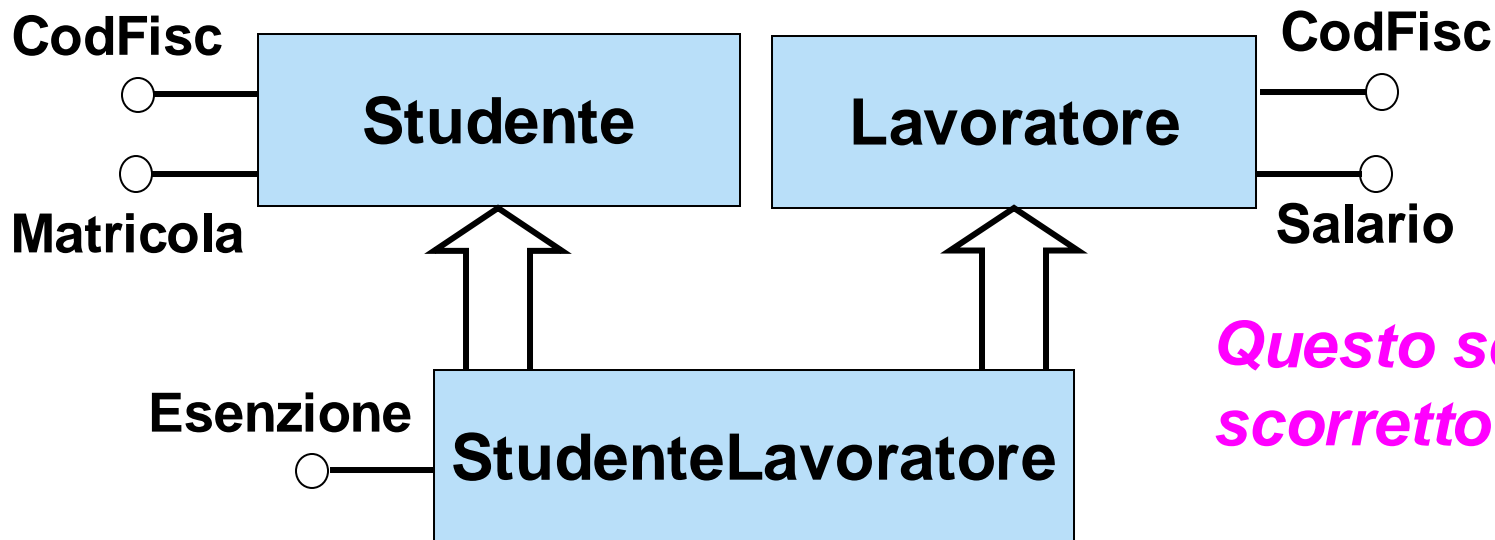
Attenzione: continua a valere la regola che una entità può avere al massimo una entità padre (tenendo conto sia della relazione ISA sia delle generalizzazioni)

Esempio:



Questo schema è scorretto

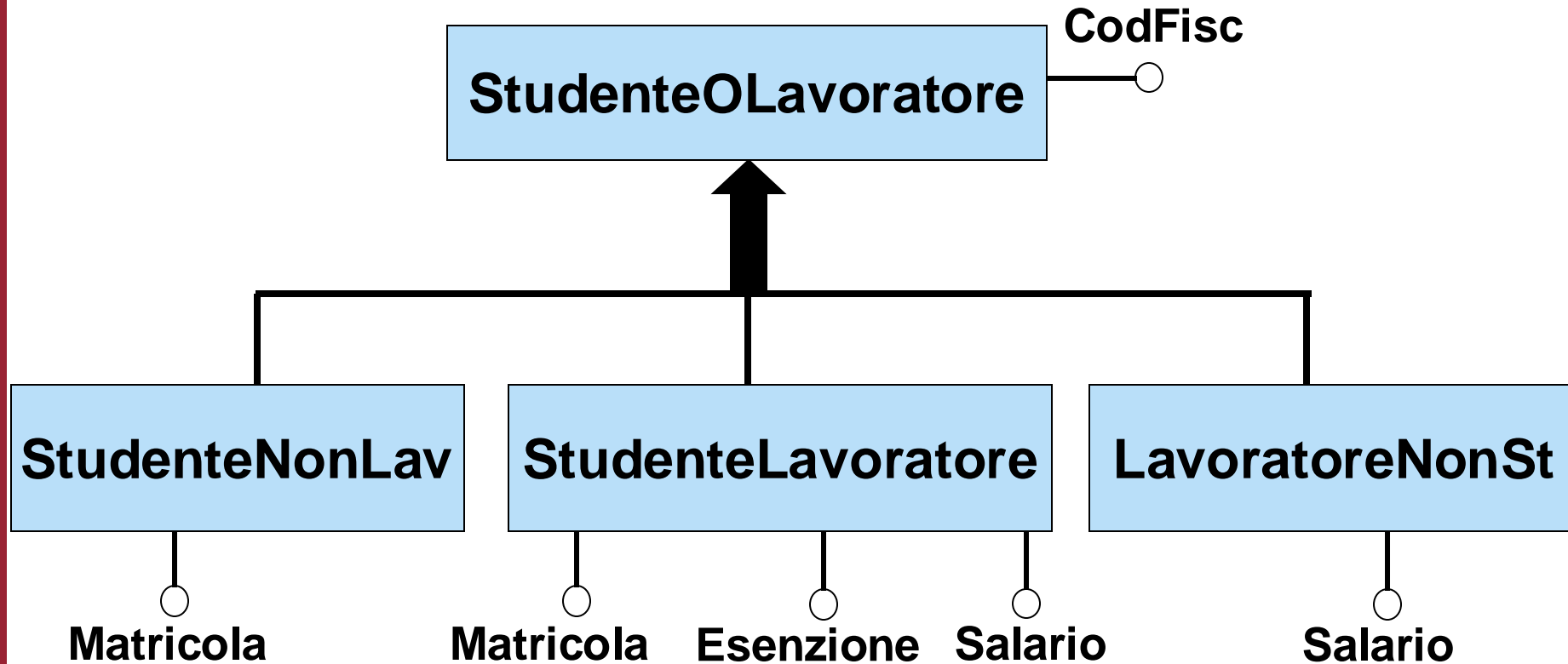
Esercizio 13: relazione ISA e generalizzazioni



Si può ristrutturare lo schema in modo che lo schema risultante sia corretto, e colga sostanzialmente la stessa semantica?



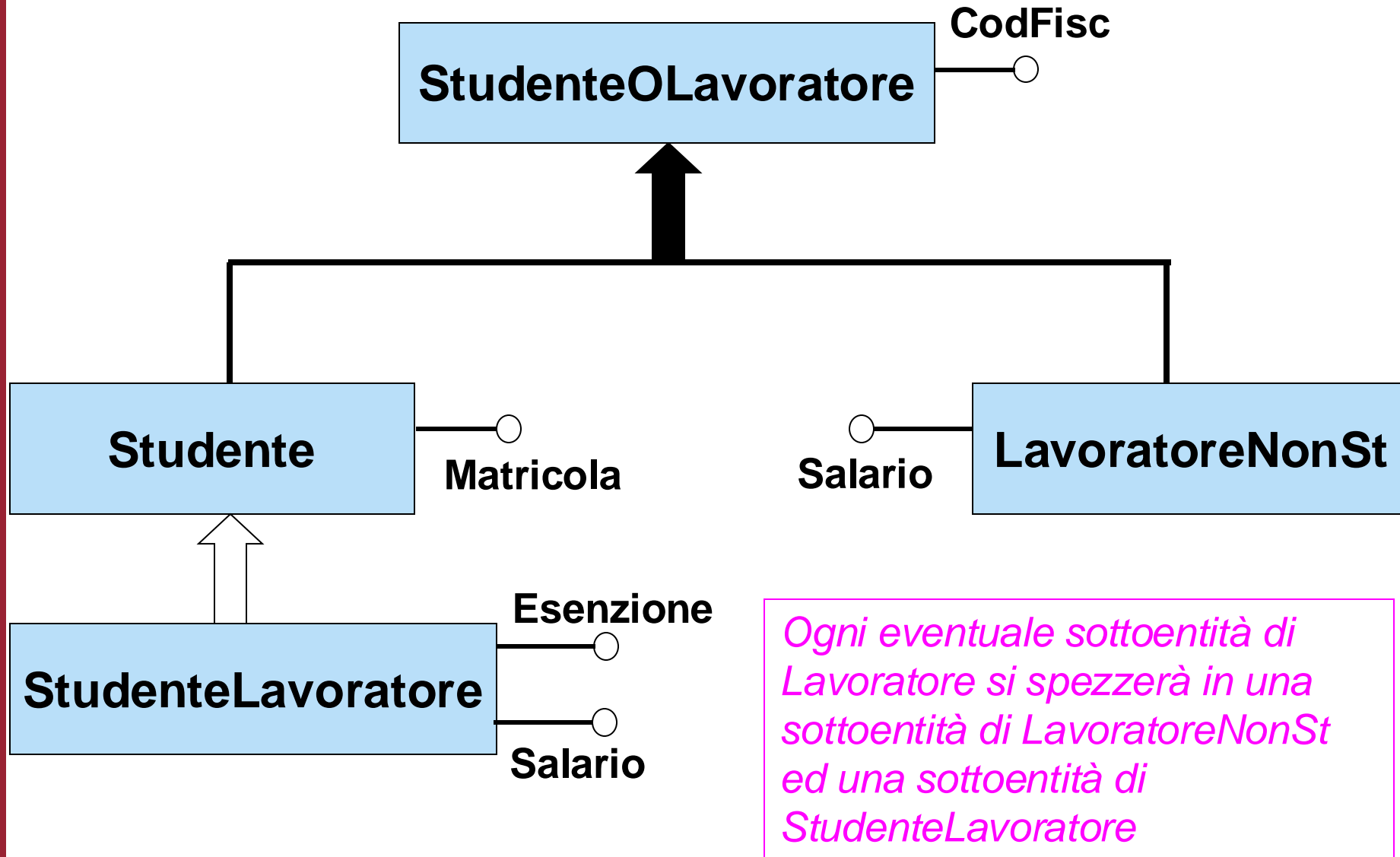
Esercizio 13: soluzione 1



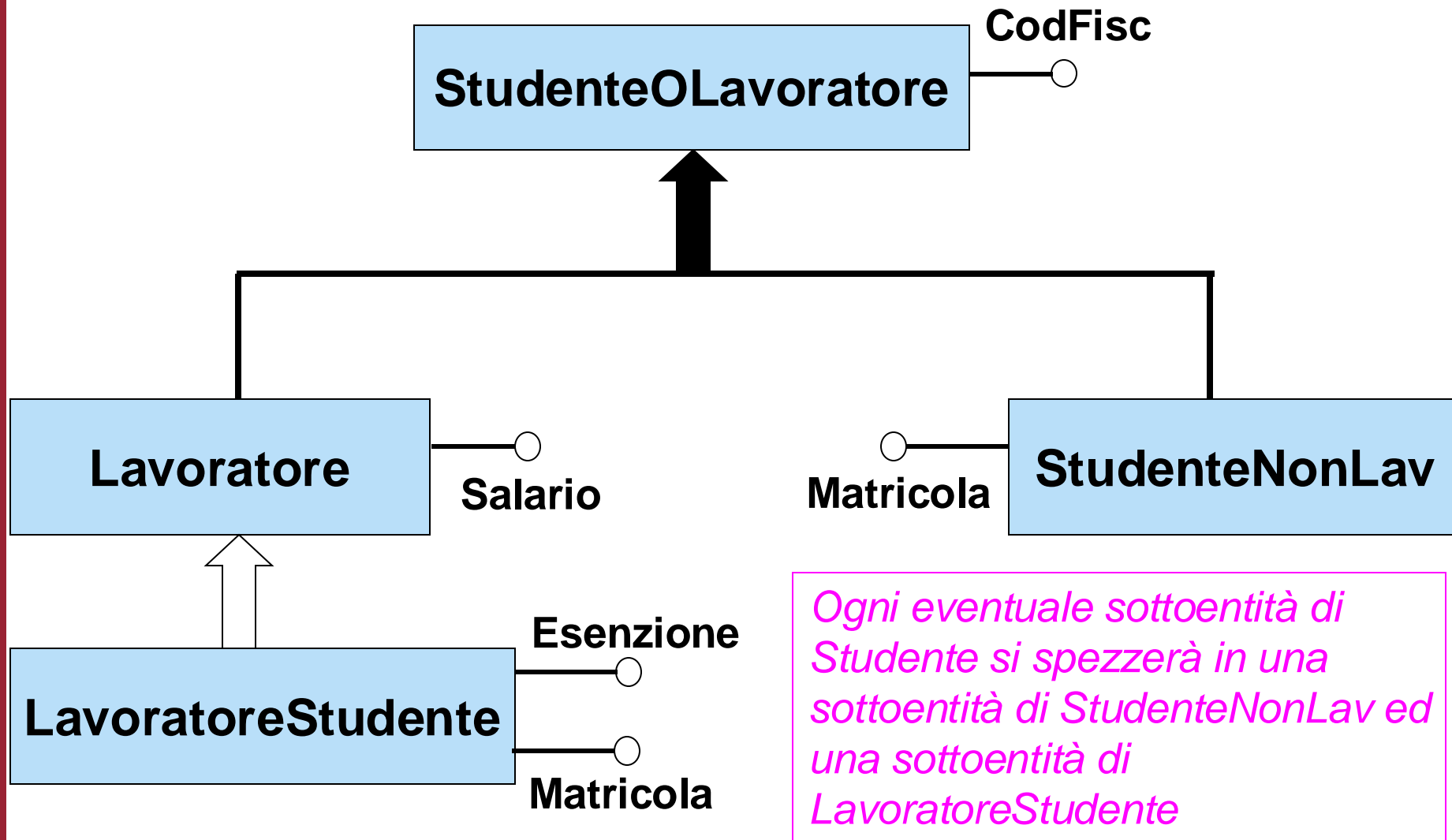
*Ogni eventuale sottoentità di **Studente** si spezzerà in una sottoentità di **StudenteNonLav** ed una sottoentità di **StudenteLavoratore**. Analogamente, ogni sottoentità di **Lavoratore** si spezzerà in una sottoentità di **LavoratoreNonSt** ed una sottoentità di **StudenteLavoratore***



Esercizio 13: soluzione 2

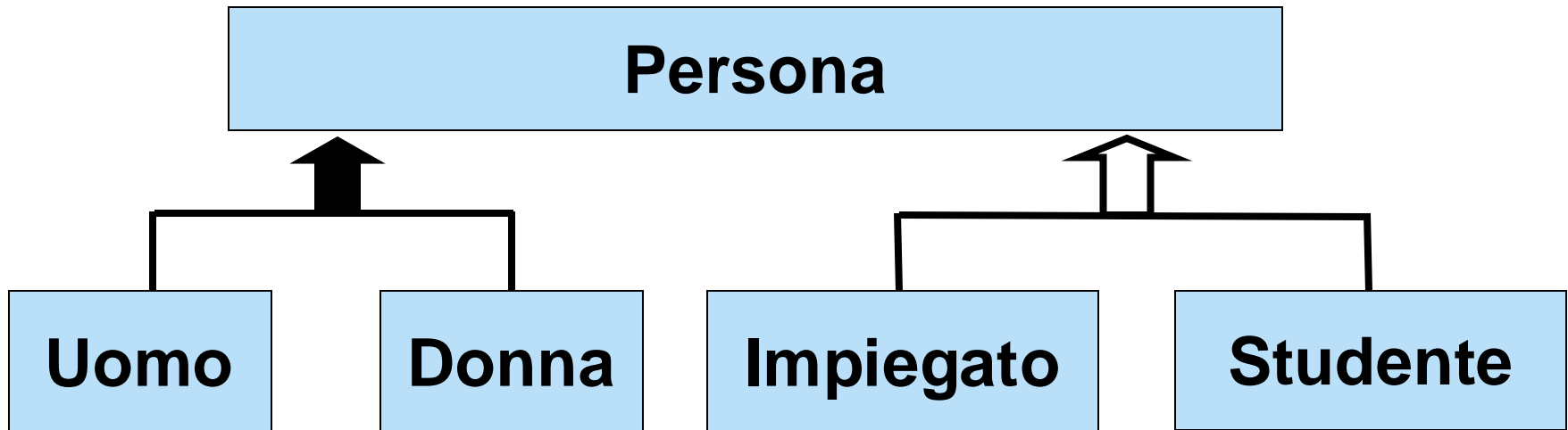


Esercizio 13: soluzione 3



Diverse generalizzazioni della stessa classe

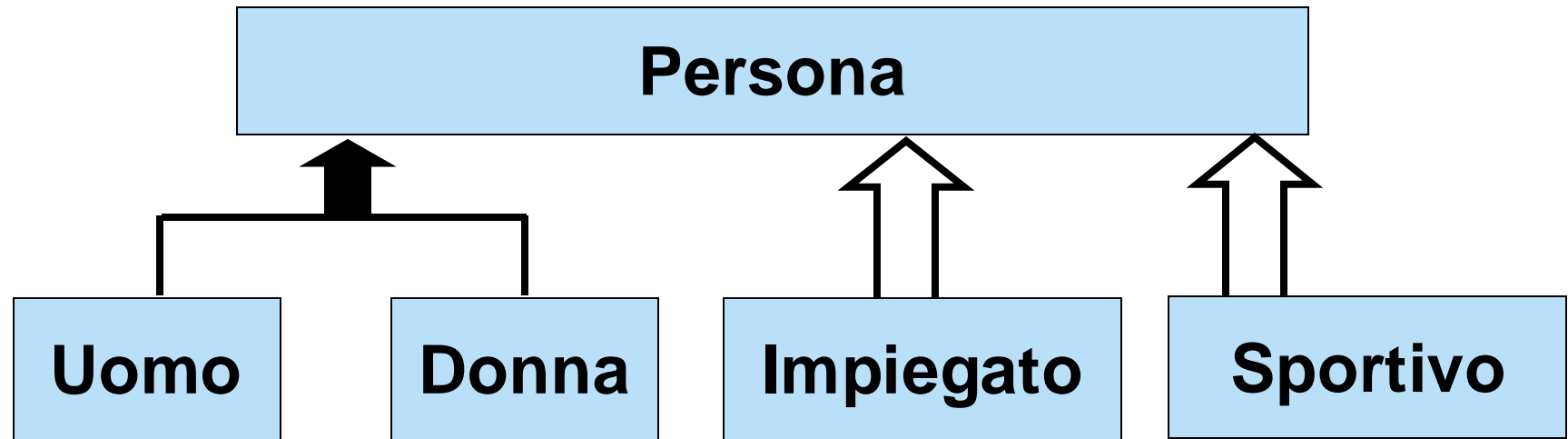
La stessa entità può essere padre in diverse generalizzazioni



Concettualmente, non c'è alcuna correlazione tra due generalizzazioni diverse, perchè rispondono a due criteri diversi di classificare le istanze della entità padre



Differenza tra due isa e una generalizzazione



Le due sottoclassi derivano da uno stesso criterio di classificazione delle istanze della entità padre

Le due sottoentità sono indipendenti, nel senso che il loro significato non deriva dallo stesso criterio di classificazione delle istanze della entità padre



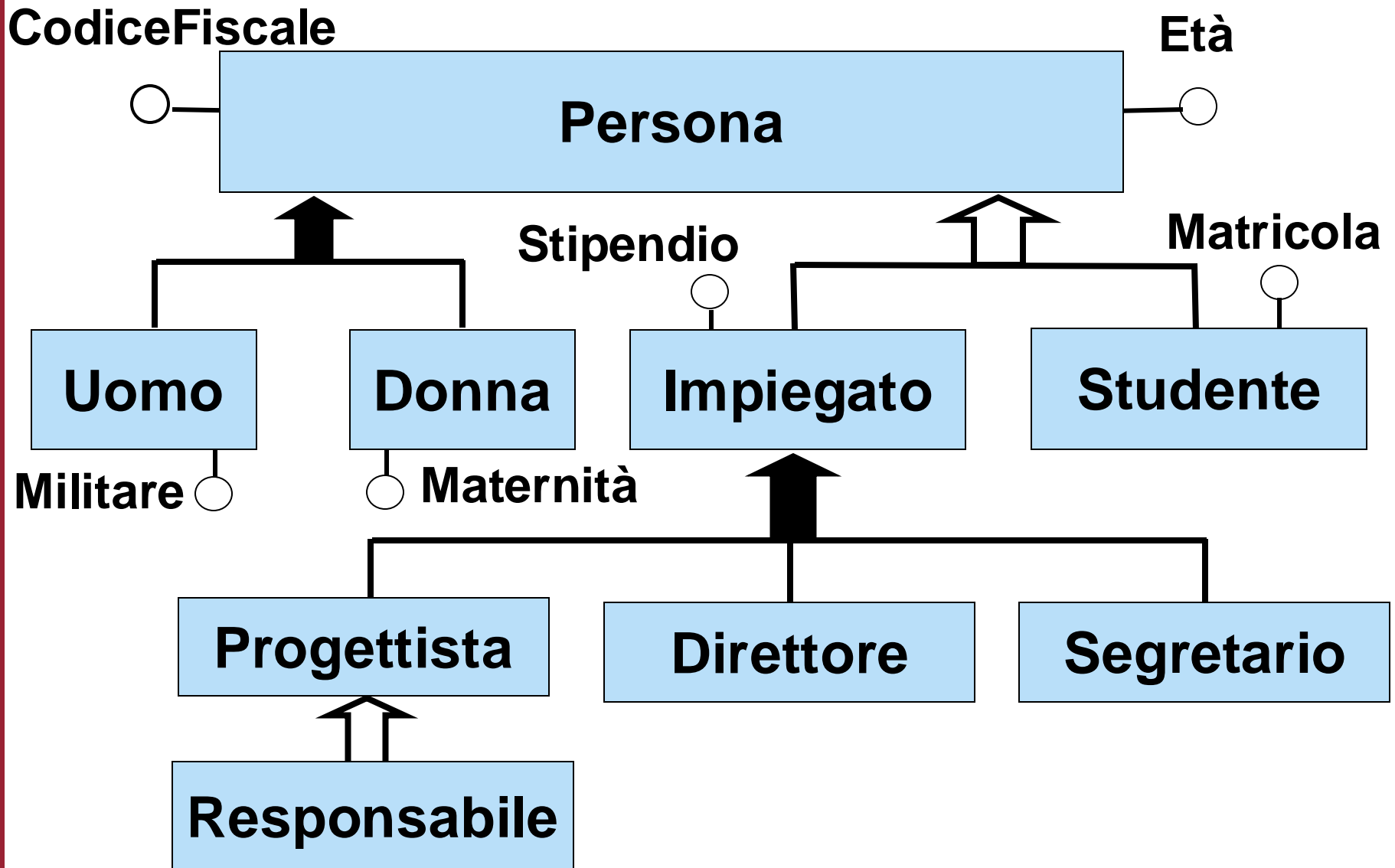
Esercizio 14: generalizzazione

Descrivere lo schema concettuale corrispondente alle seguenti specifiche.

Le persone hanno un codice fiscale ed una età. Gli uomini anche la posizione militare, e le donne anche il numero di maternità. Gli impiegati hanno lo stipendio e possono essere o segretari, o direttori o progettisti. Alcuni progettisti sono responsabili di progetto. Gli studenti (che non possono essere impiegati) hanno un numero di matricola. Esistono persone che non sono né impiegati né studenti.



Esercizio 14: soluzione





Grafo delle gerarchie

Ad ogni schema S si può associare un grafo, detto **grafo delle gerarchie associato ad S** , costruito come segue:

1. Si definisce nel grafo un nodo per ogni entità ed un nodo per ogni relazione di S
2. Si traccia un arco del grafo dal nodo n corrispondente alla entità E al nodo m (diverso da n) corrispondente alla entità F se F è padre di E in S secondo la relazione ISA o secondo la generalizzazione.
3. Si fa la stessa cosa per le relazioni (vedremo tra poco, infatti, che la ISA e la generalizzazione si possono stabilire anche tra relazioni)
4. Per ogni ciclo di G , si collassano i nodi che fanno parte del ciclo: in questo modo a tutte le entità (se il ciclo coinvolge entità) o relazioni (se il ciclo coinvolge relazioni) che facevano parte del ciclo corrisponde ora lo stesso nodo in G

Diremo che una entità (o relazione) G è **antenata** dell'entità (o relazione) E in uno schema S se esiste un cammino da E ad G nel grafo delle gerarchie associato ad S (si noti che ogni entità è antenata di se stessa)



Entità disgiunte

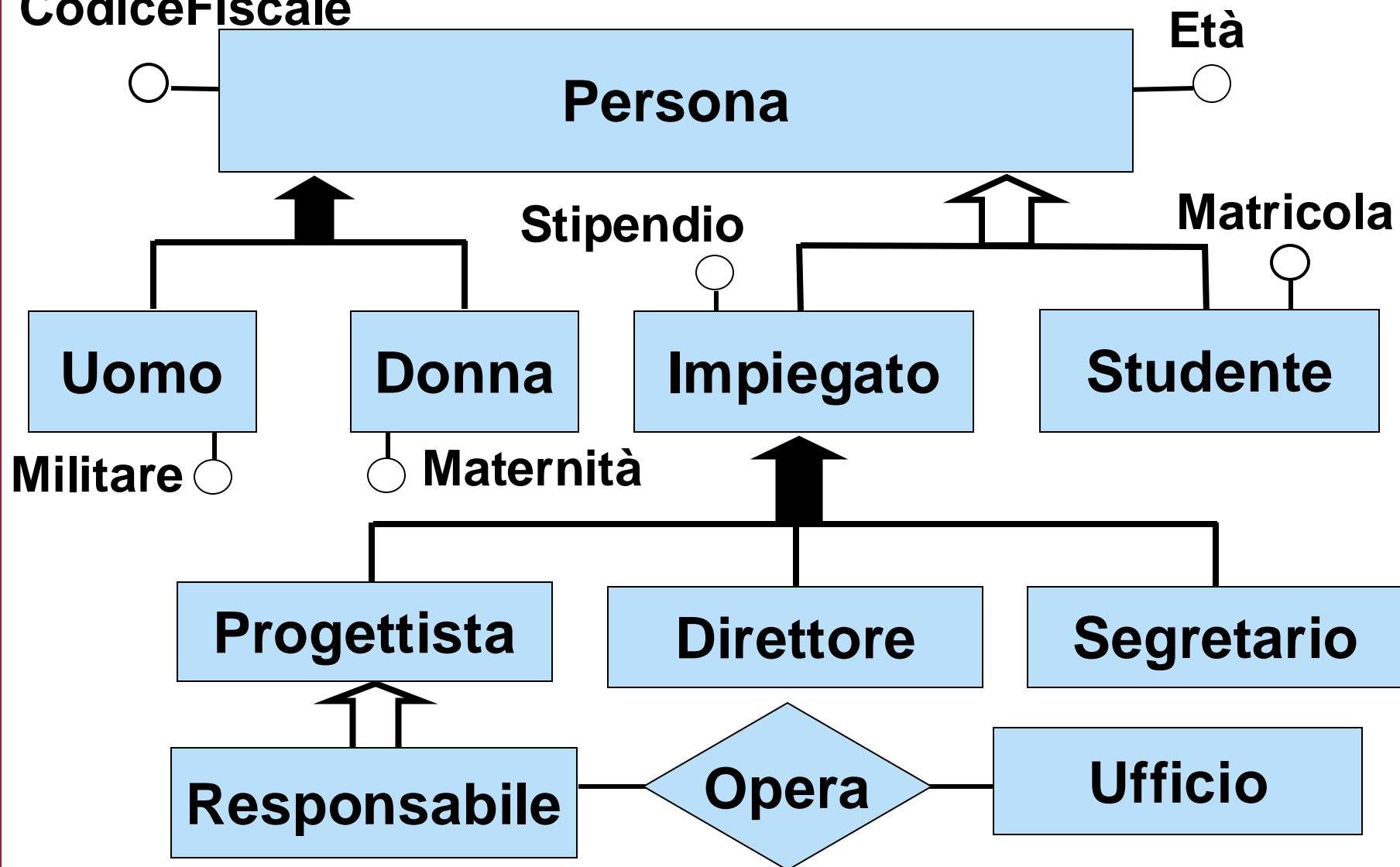
Da quanto detto sinora, non è ancora chiaro quando, date due entità di uno schema, esse siano disgiunte (cioè non hanno istanze comuni in tutte le istanze dello schema).

A questo punto possiamo dare la regola generale. Sia S uno schema:

1. Si assume che le entità corrispondenti a nodi del grafo delle gerarchie associato ad S che non hanno successori siano in realtà sottoentità di una generalizzazione completa la cui entità padre è una entità speciale detta **TOP** (si noti che se una sola entità non ha successori nel grafo delle gerarchie, allora tale entità è equivalente a **TOP**). Ne segue che diverse entità figlie di **TOP** sono disgiunte a coppie, e che in ogni istanza dello schema, **TOP** rappresenta tutti gli oggetti che sono istanze di qualche entità
2. È facile verificare che, sulla base dell'assunzione 1. due entità E ed F diverse tra loro sono **disgiunte** in uno schema S se e solo se esistono due entità G ed H (non necessariamente diverse da E ed F) nello schema S tali che:
 1. G ed H sono diverse sottoentità della stessa entità padre in una generalizzazione (e quindi sono disgiunte)
 2. G è antenata di E , ed H è antenata di F

Esercizio 15: elencare tutte le coppie di entità disgiunte in questo schema

CodiceFiscale





Esercizio 15: soluzione

Coppie di entità disgiunte:

- Ufficio e ognuna delle altre entità
- Uomo e Donna
- Impiegato e Studente
- Progettista e Studente
- Segretario e Studente
- Direttore e Studente
- Responsabile e Studente
- Progettista e Direttore
- Progettista e Segretario
- Direttore e Segretario
- Responsabile e Direttore
- Responsabile e Segretario



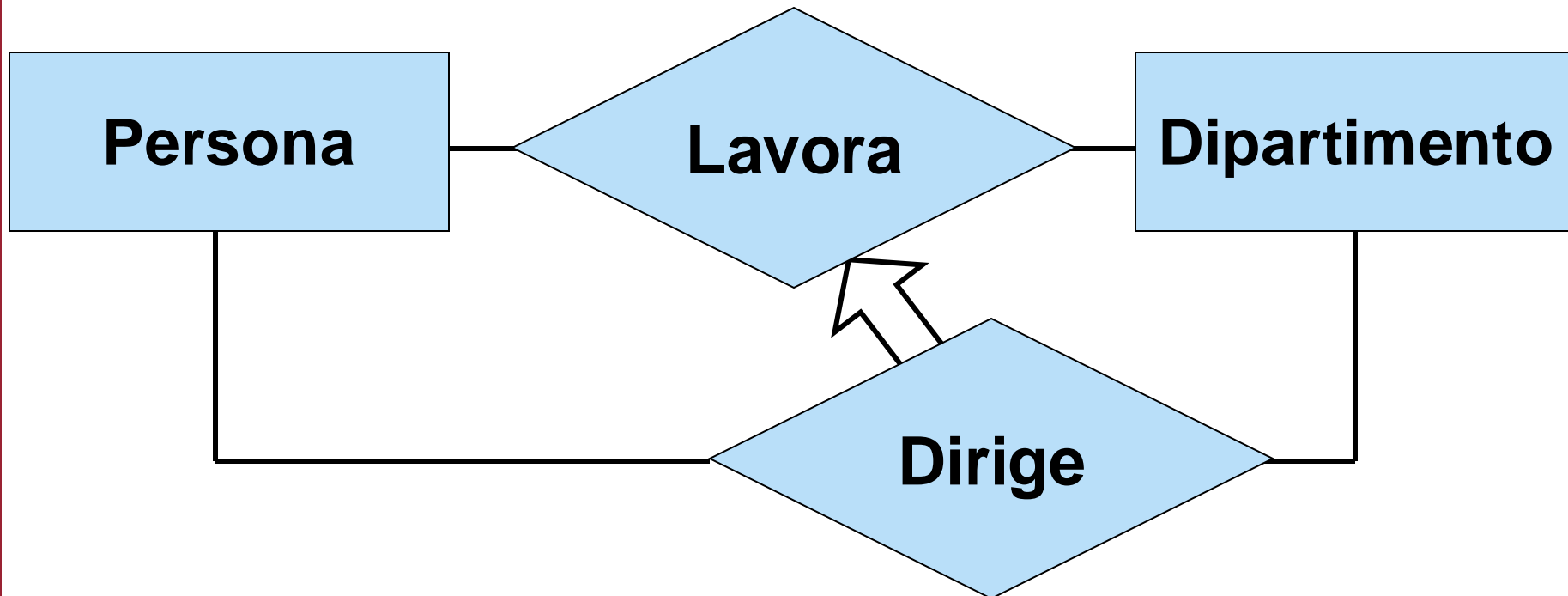
Relazione ISA e generalizzazioni tra relazioni

- La relazione ISA può essere definita anche tra relazioni, ma con la seguente **regola**: si può definire la relazione ISA tra la relazione R (relazione figlia) e la relazione Q (relazione padre) in uno schema S solo se R e Q
 - hanno lo stesso insieme di ruoli (e quindi anche uguale grado)
 - per ogni ruolo U, l'entità corrispondente ad U in R è in relazione ISA (al limite transitivamente) con l'entità corrispondente ad U in Q
- La semantica non cambia rispetto al caso della relazione ISA tra entità: se in uno schema **S** è definita la relazione ISA tra **R** e **Q** (**R** ISA **Q**, dove **R** e **Q** sono due relazioni con lo stesso grado e gli stessi ruoli), allora in ogni istanza **I** dello schema **S**, si ha che $istanze(I, R) \subseteq istanze(I, Q)$
- Analogamente, si possono definire generalizzazioni tra relazioni, seguendo le regole sopra descritte per le relazioni figlie rispetto alla relazione padre nella generalizzazione



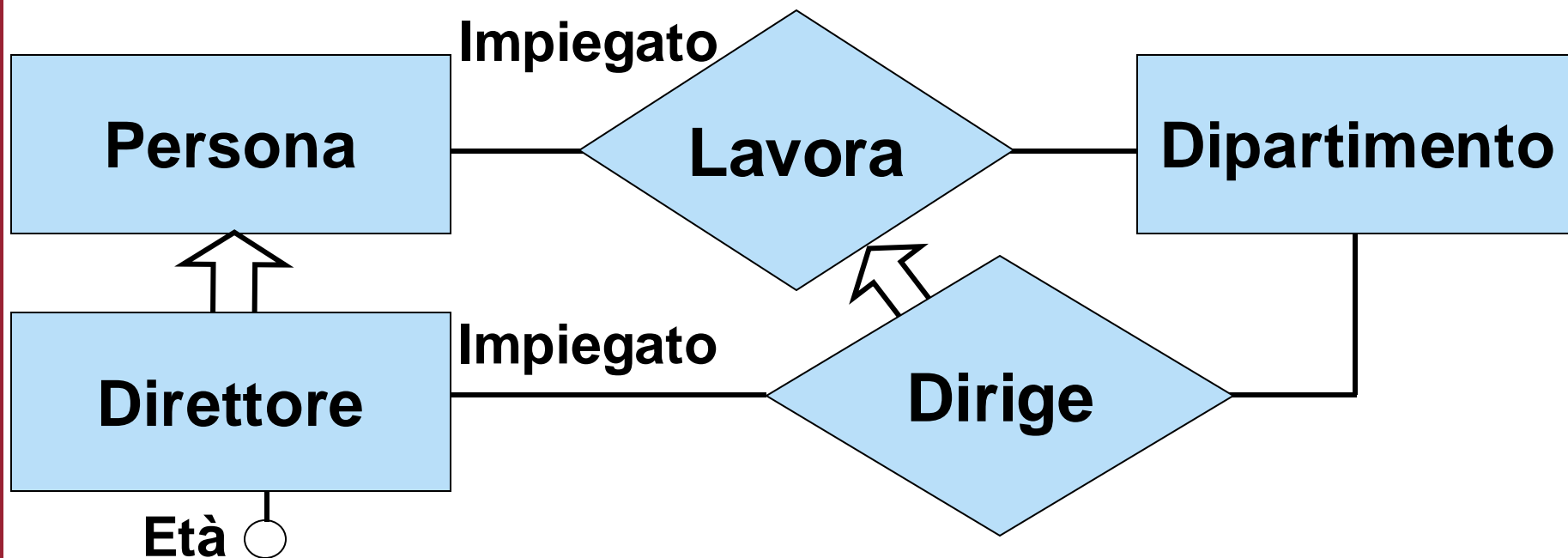
Esempio di relazione ISA tra relazioni

Il direttore di un dipartimento deve lavorare in quel dipartimento



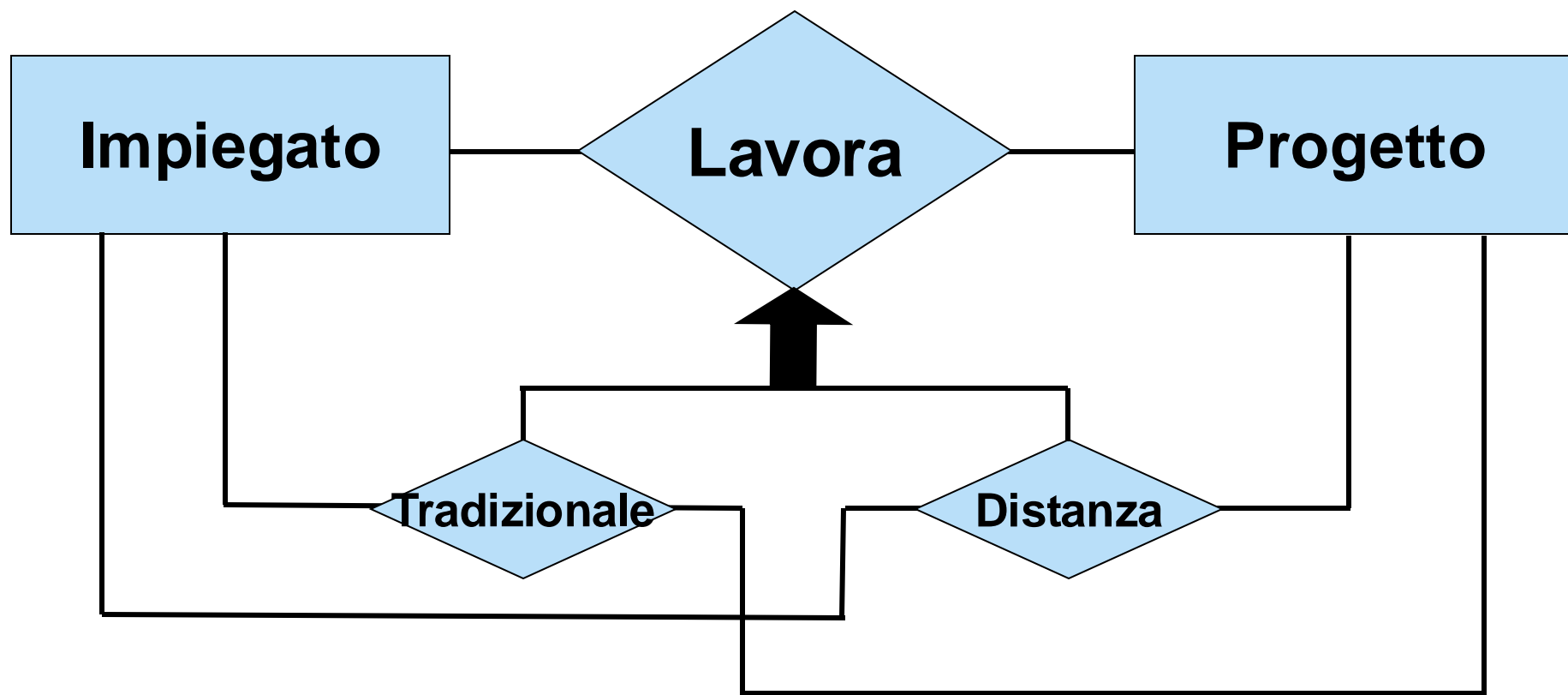
Esempio di relazione ISA tra relazioni

Il direttore di un dipartimento deve lavorare in quel dipartimento, ed ha proprietà aggiuntive (ad esempio l'età) rispetto agli impiegati



Esempio di generalizzazione di relazioni

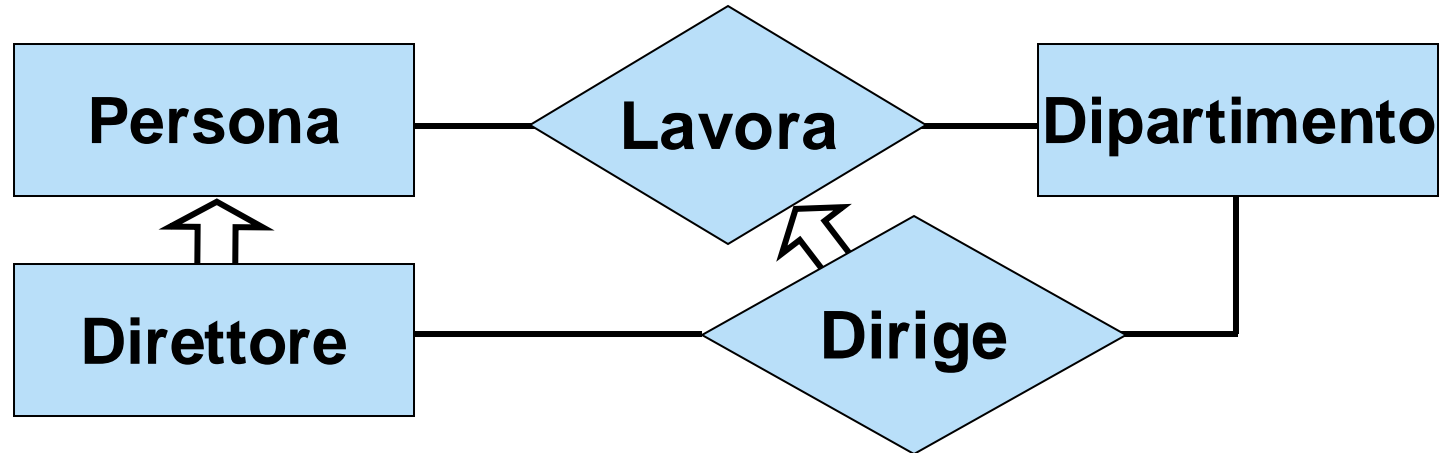
Ogni impiegato lavora ad un progetto o con modalità tradizionale o con modalità “a distanza”: definiamo una generalizzazione della relazione Lavora



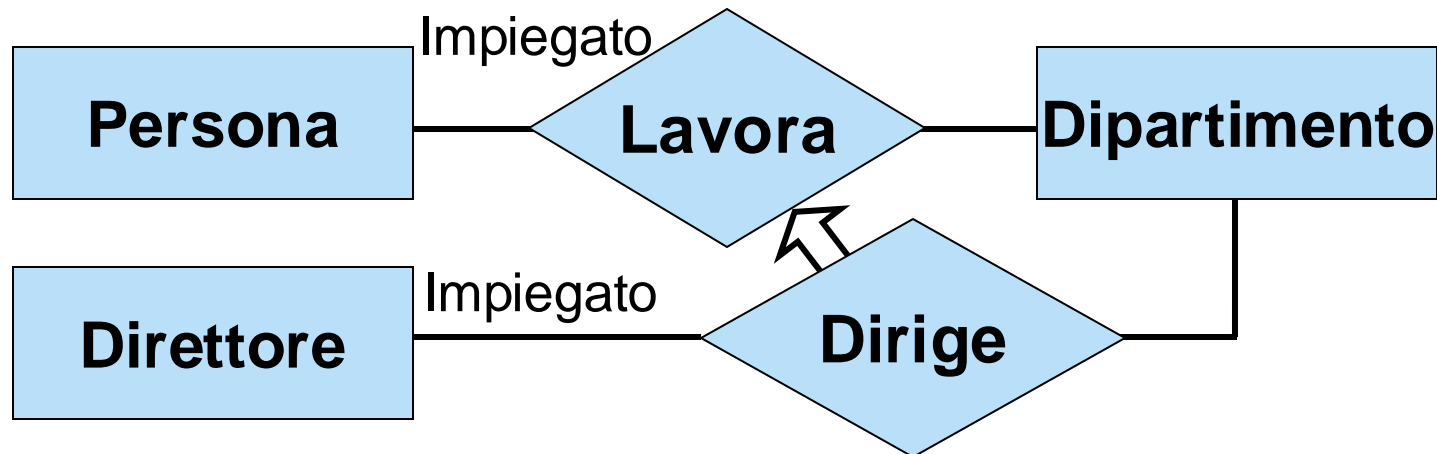


Esempi di schemi non corretti

*Ruoli diversi
nelle due
relazioni*



*Ruoli uguali
ma entità non
in relazione
ISA*





Relazioni disgiunte

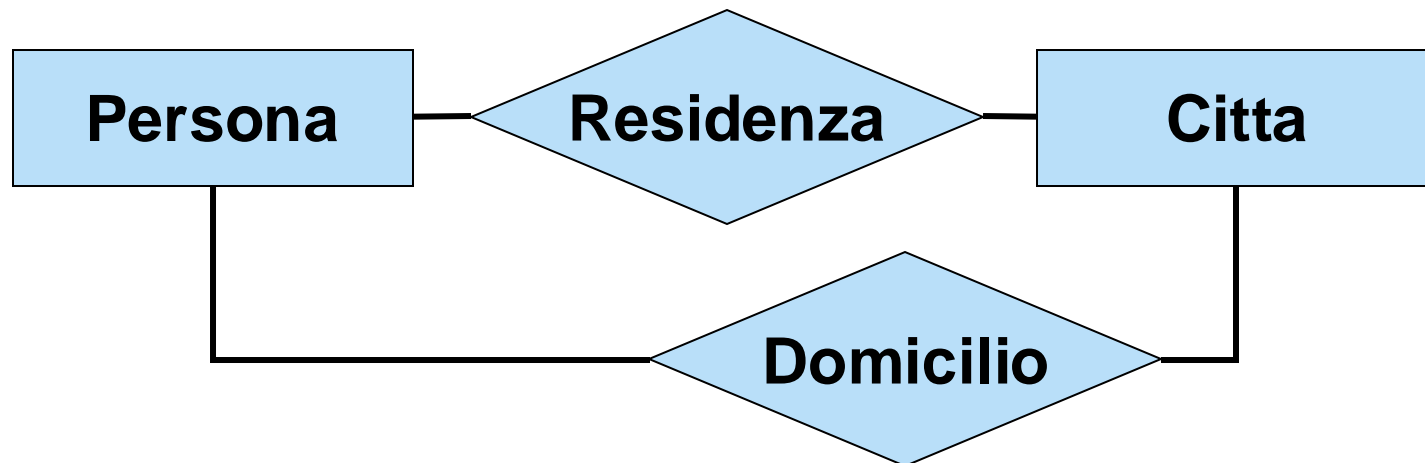
Per le relazioni valgono osservazioni simili (ma non uguali!) a quelle che abbiamo fatto sulle entità.

- In ogni schema si assume che le relazioni dello stesso grado k e con gli stessi ruoli U_1, U_2, \dots, U_k che non hanno padre nel grafo delle gerarchie associato allo schema siano in realtà sottorelazioni (e **non** figlie di una generalizzazione, come accadeva per le entità) in una ISA la cui relazione padre è una relazione speciale detta **TOP _{U_1, U_2, \dots, U_k}** (ne segue che, al contrario del caso delle entità, tali relazioni non sono disgiunte a coppie).
- Due relazioni P ed R con lo stesso grado e gli stessi ruoli sono **disgiunte** in uno schema S se esistono due relazioni Q e T (non necessariamente diverse da P ed R) nello schema S tali che:
 - Q e T sono due diverse figlie della stessa relazione padre in una generalizzazione
 - Q è antenata di P , e T è antenata di R

Relazioni disgiunte

Da quanto detto segue che, al contrario del caso delle entità, due relazioni (dello stesso grado e con gli stessi ruoli) che non hanno padre nello schema **non sono disgiunte**.

Esempio: Residenza e Domicilio non sono disgiunte





Vincoli di integrità nel modello ER

Ricordiamo che un **vincolo di integrità** è una regola che si esprime sullo schema (livello intensionale), e che specifica una condizione che deve valere **per ogni istanza** dello schema (livello estensionale).

Classificazione dei vincoli di integrità nel modello ER:

- Vincoli di cardinalità sulle relazioni
- Vincoli di cardinalità sugli attributi
- Vincoli di identificazione di entità e di relazioni
- Altri vincoli (esterni)



Vincoli di cardinalità sulle relazioni

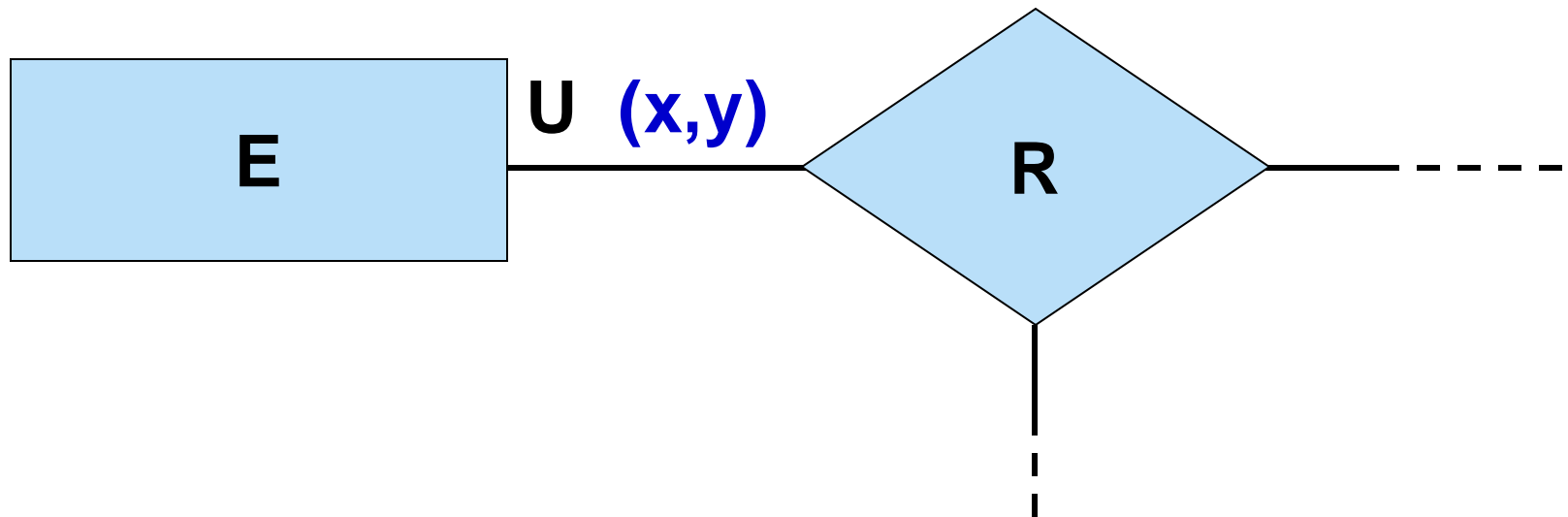
Un **vincolo di cardinalità** si associa ad un ruolo U (corrispondente ad una entità E) in una relazione R, ed impone un limite minimo ed un limite massimo di istanze della relazione a cui può partecipare ogni istanza dell'entità E nel ruolo U.

Serve a caratterizzare meglio il significato di una relazione.

Sintassi dei vincoli di cardinalità

Un vincolo di cardinalità si esprime mediante **una coppia (x,y) associata al ruolo U della relazione R**, dove:

- **x** è la cardinalità minima, cioè un **intero ≥ 0**
- **y** è la cardinalità massima, che è
 - “**n**”, oppure
 - un **intero positivo $\geq x$**





Semantica dei vincoli di cardinalità

Se in uno schema **S** è definito un vincolo di cardinalità (x,y) associato ad un ruolo **U** (corrispondente ad una entità **E**) in una relazione **R**, allora in ogni istanza **I** dello schema **S**, per ogni **e** in $istanze(I,E)$, il numero di istanze di **R** che in **I** hanno **e** come componente nel ruolo **U** è:

- maggiore o uguale a x
- minore o uguale a y (se y è diverso da “n”)

Attenzione: il vincolo di cardinalità si definisce su un ruolo di una relazione, ma in realtà stabilisce una condizione che deve valere per tutte le istanze della entità a cui è associato il vincolo stesso (tramite il ruolo)



Esempio di vincoli di cardinalità

- Ad ogni impiegato sono assegnati da 1 a 5 incarichi
- Ogni incarico è assegnato ad al più 50 impiegati



Istanza I:

$istanze(I, Impiegato) = \{ a, b, c \}$

$istanze(I, Incarico) = \{ x, y, v, w, z \}$

$istanze(I, Assegnazione) = \{ (Impiegato:a, Incarico:w),$
 $(Impiegato:b, Incarico:v),$
 $(Impiegato:b, Incarico:w),$
 $(Impiegato:c, Incarico:y),$
 $(Impiegato:c, Incarico:w),$
 $(Impiegato:c, Incarico:z) \}$

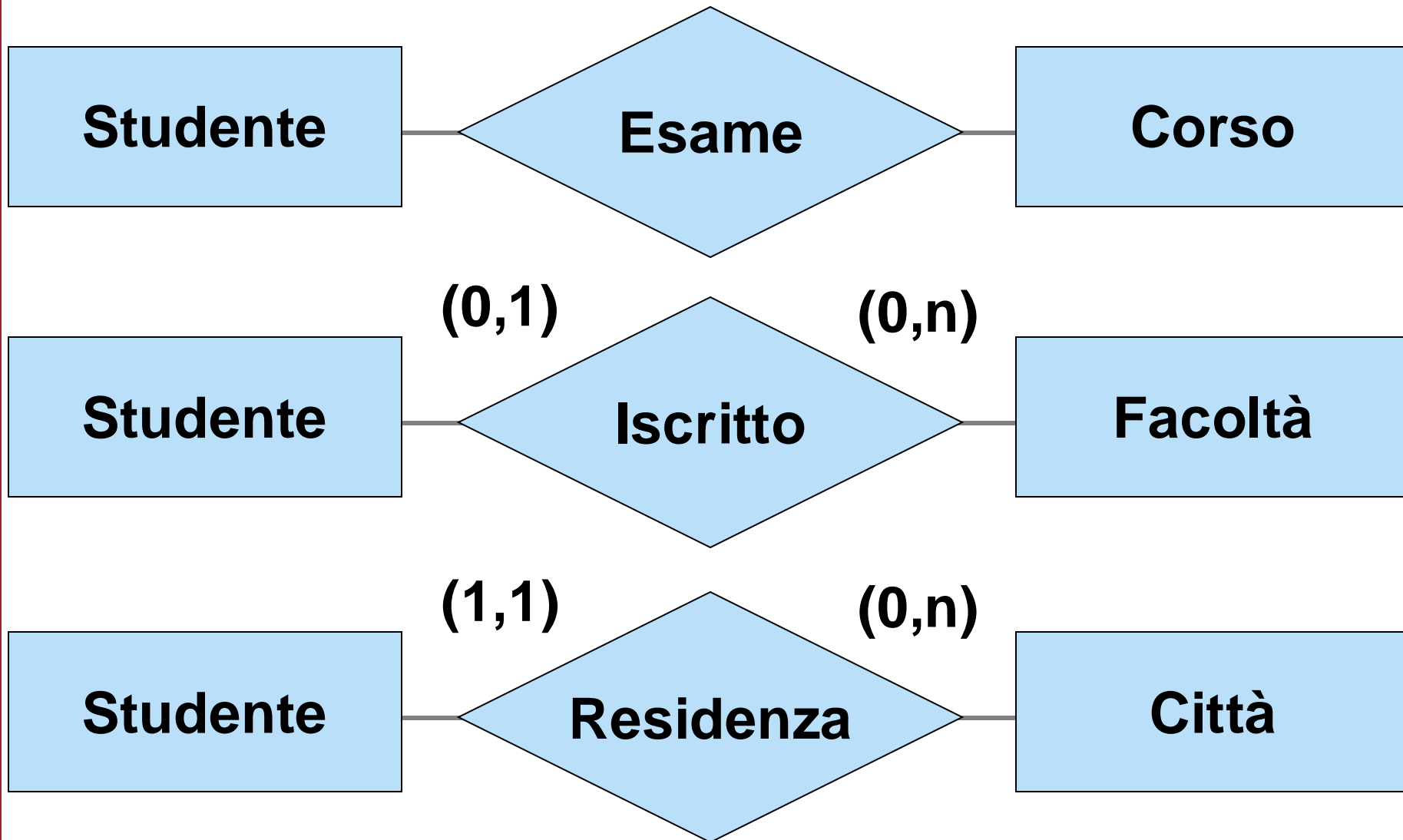


Cardinalità maggiormente utilizzate

- Le cardinalità di interesse sono tipicamente tre: 0, 1, n.
- Per la **cardinalità minima**:
 - 0 significa “partecipazione opzionale”
 - 1 significa “partecipazione obbligatoria”
- Per la **cardinalità massima**:
 - 1 significa: “l’entità partecipa al più una volta sola alla relazione”
 - n significa: “l’entità partecipa un numero qualsiasi di volte alla relazione”
- La **mancaanza del vincolo di cardinalità** è equivalente alla coppia (0,n)



Esempi di vincoli di cardinalità





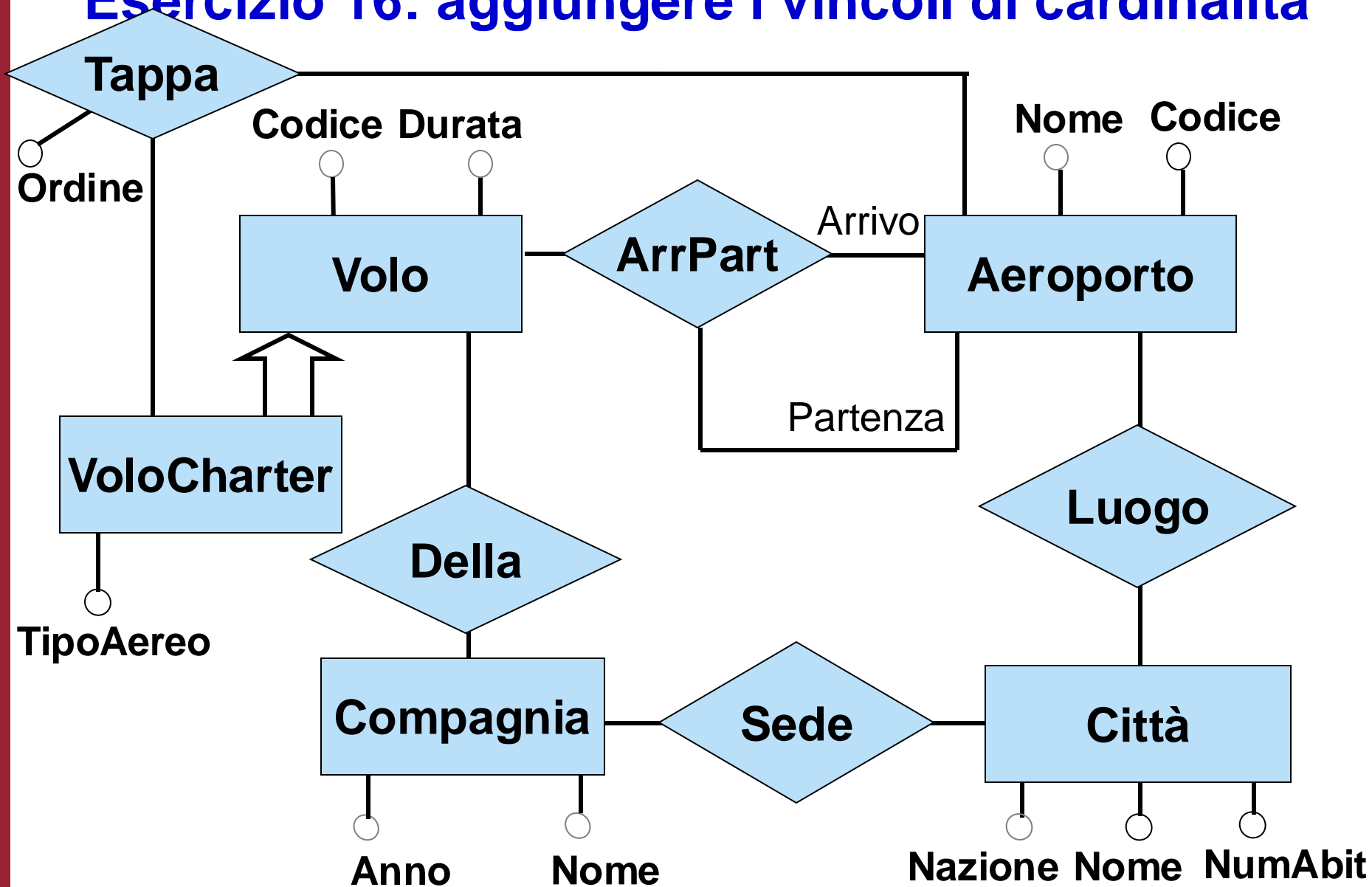
Classificazione di relazioni binarie rispetto alla cardinalità

Esiste una classificazione delle relazioni binarie che si adotta spesso riferendosi alla cardinalità massime. Infatti, con riferimento alle cardinalità **massime**, le relazioni binarie possono essere di tre tipi:

- **uno a uno** (esempio: relazione **vendita** tra fattura e ordine)
- **uno a molti** (esempio: relazione **contiene** tra provincia e comune)
- **molti a molti** (esempio, relazione **esame** tra studente e corso)

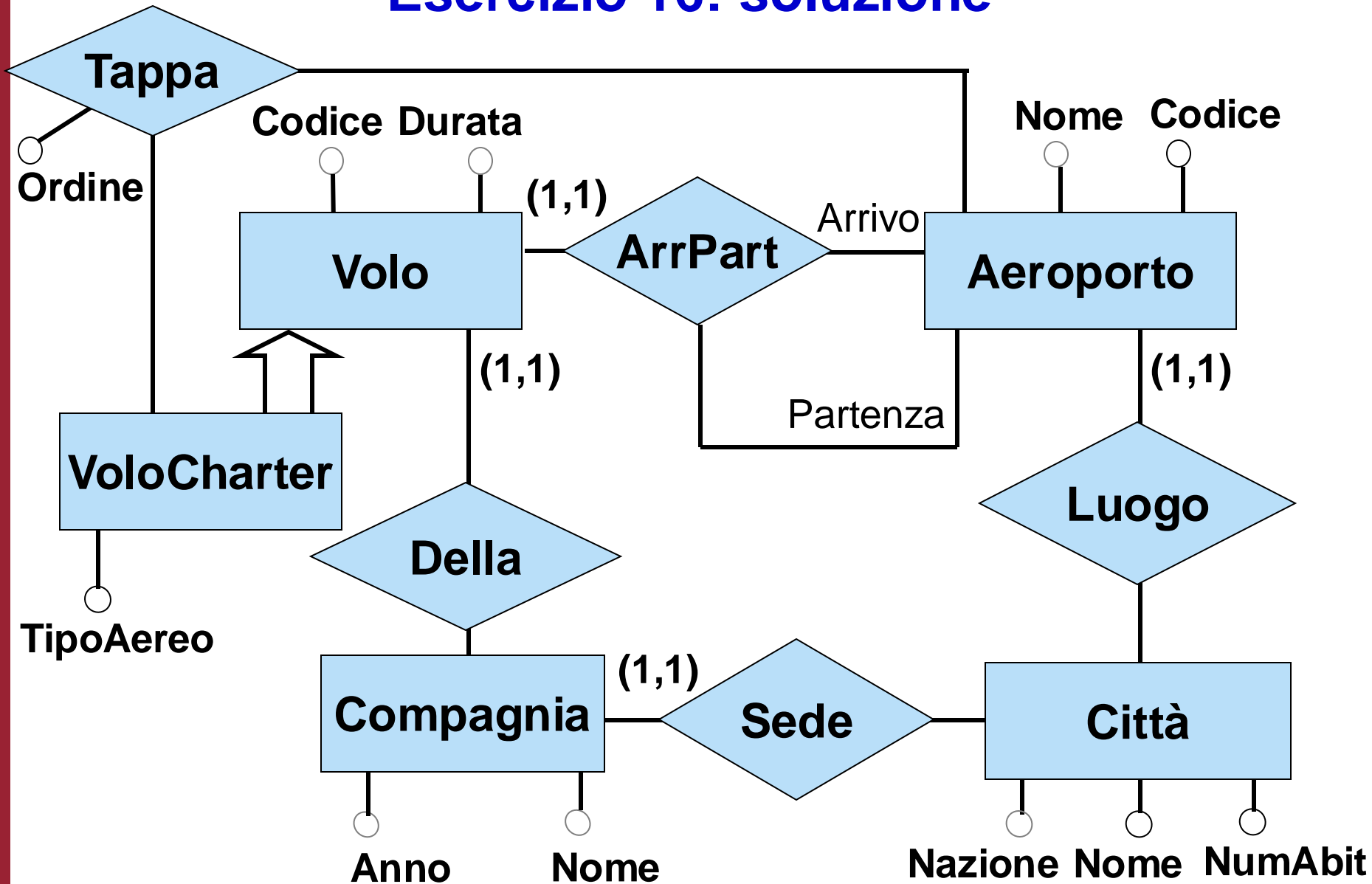


Esercizio 16: aggiungere i vincoli di cardinalità

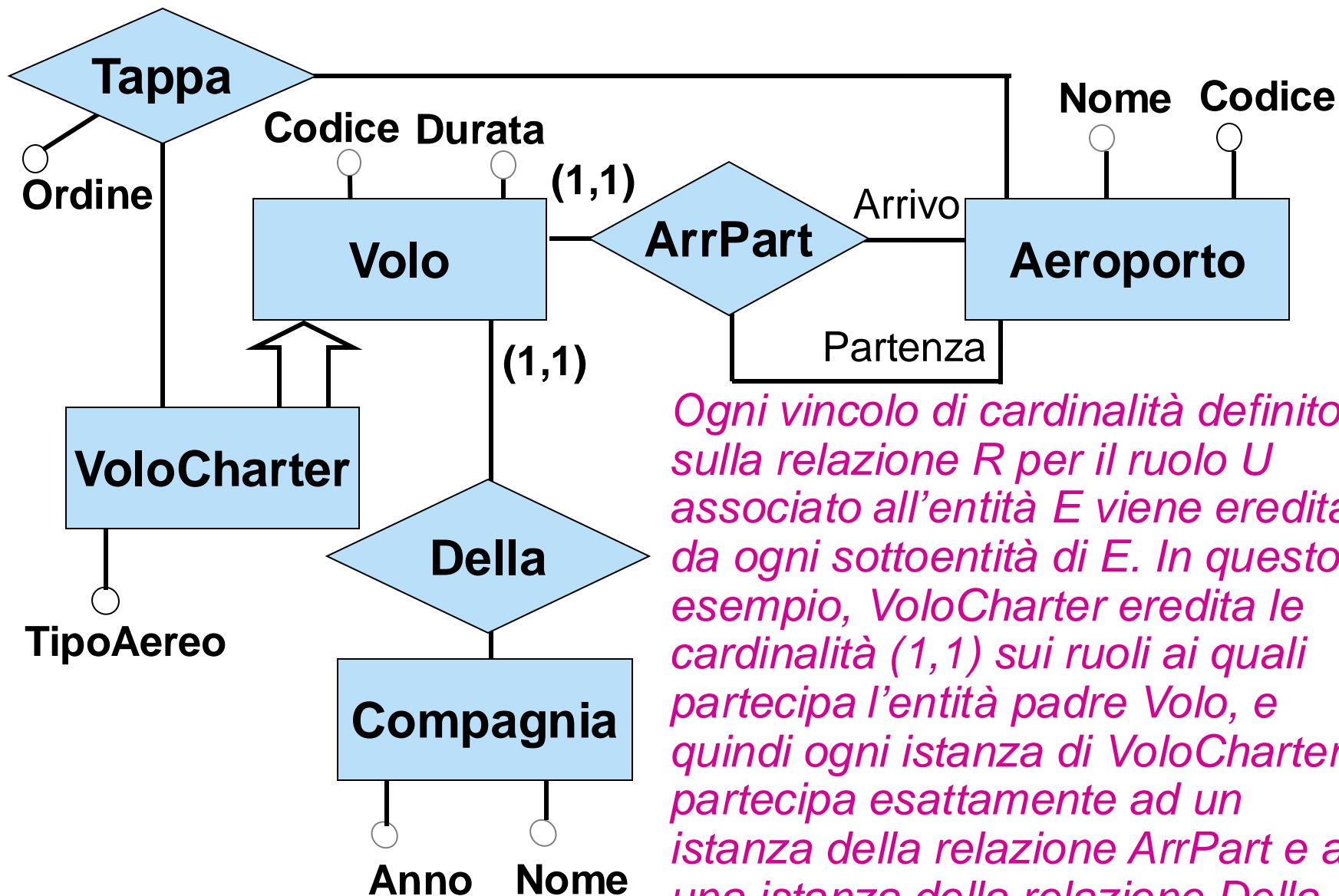




Esercizio 16: soluzione



Vincoli di cardinalità ed ereditarietà su entità



*Ogni vincolo di cardinalità definito sulla relazione R per il ruolo U associato all'entità E viene ereditato da ogni sottoentità di E . In questo esempio, **VoloCharter** eredita le cardinalità (1,1) sui ruoli ai quali partecipa l'entità padre **Volo**, e quindi ogni istanza di **VoloCharter** partecipa esattamente ad un'istanza della relazione **ArrPart** e ad una istanza della relazione **Della**.*



Vincoli di cardinalità massima ed ereditarietà su relazioni

Studiamo adesso come i vincoli di cardinalità definiti su una relazione R si propagano sulle relazioni che sono figlie di R o padri di R .

Approfondendo questo aspetto ci renderemo conto che può capitare che sullo stesso ruolo della stessa relazione vengano definiti (esplicitamente o implicitamente, ovvero a fronte della propagazione) diverse coppie di vincoli di cardinalità.

Se ad un ruolo U di una relazione R sono associate più coppie

$$(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$$

diremo che i vincoli di cardinalità **più stringenti** che valgono per U in R sono dati dalla coppia

$$(x, y)$$

dove

- x è il massimo tra x_1, x_2, \dots, x_n e
- y è il minimo tra y_1, y_2, \dots, y_n

Ovviamente, uno schema si può sempre scrivere indicando, per ogni ruolo, una sola coppia di vincoli di cardinalità, quelli più stringenti.

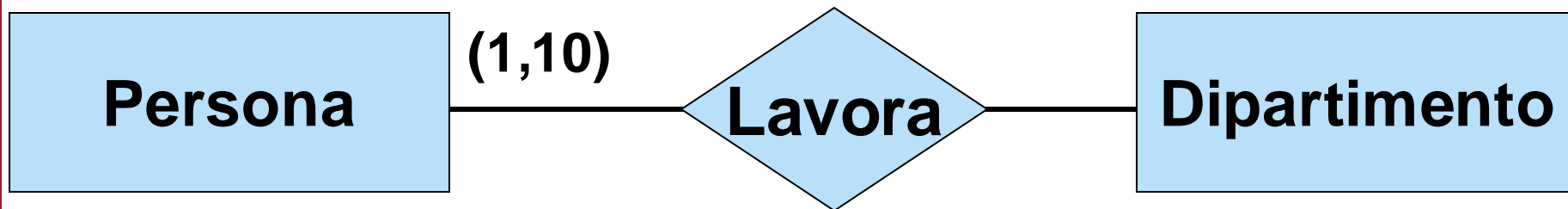


Esempio di vincoli più stringenti

Nello schema seguente allo stesso ruolo vengono associate più coppie (due, in particolare) di vincoli di cardinalità:



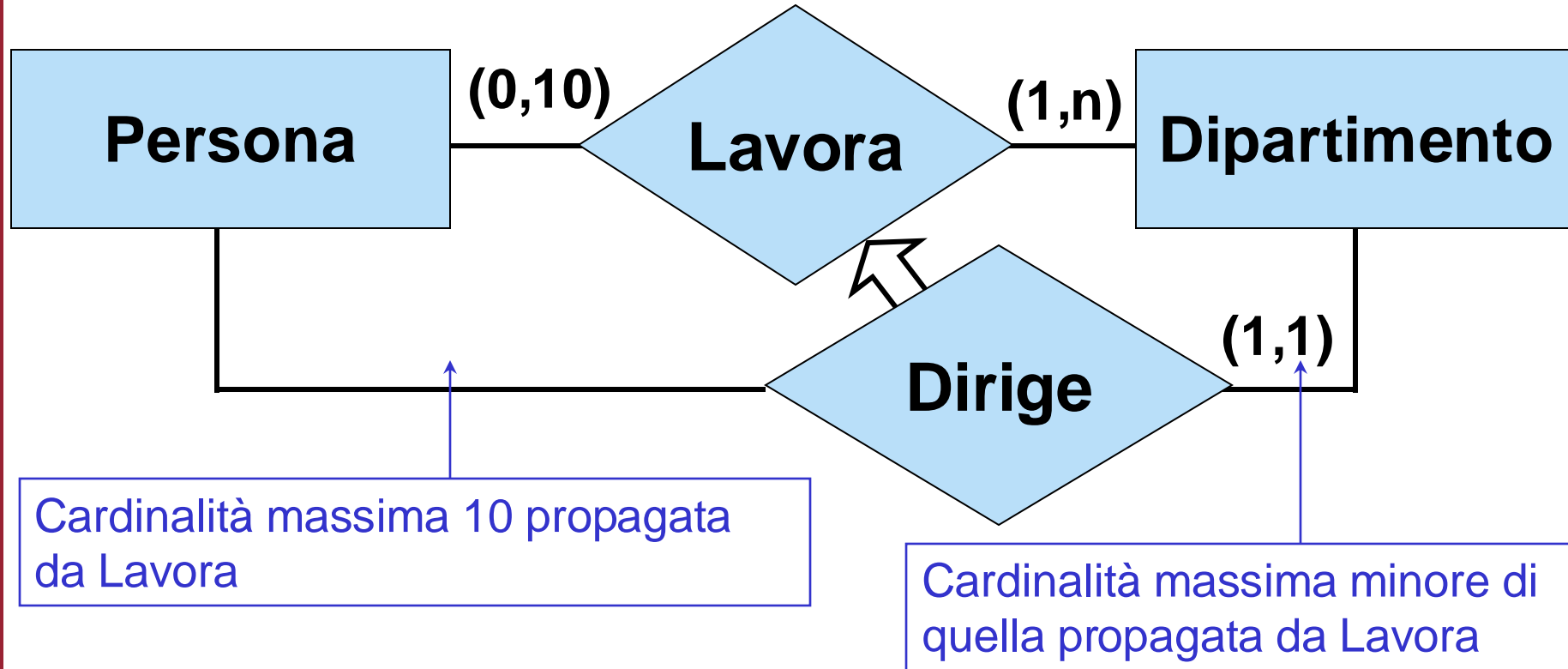
Vediamo ora uno schema equivalente a quello precedente in cui abbiamo indicato per il ruolo **Persona** la coppia di vincoli più stringenti.





Vincoli di cardinalità massima ed ereditarietà su relazioni

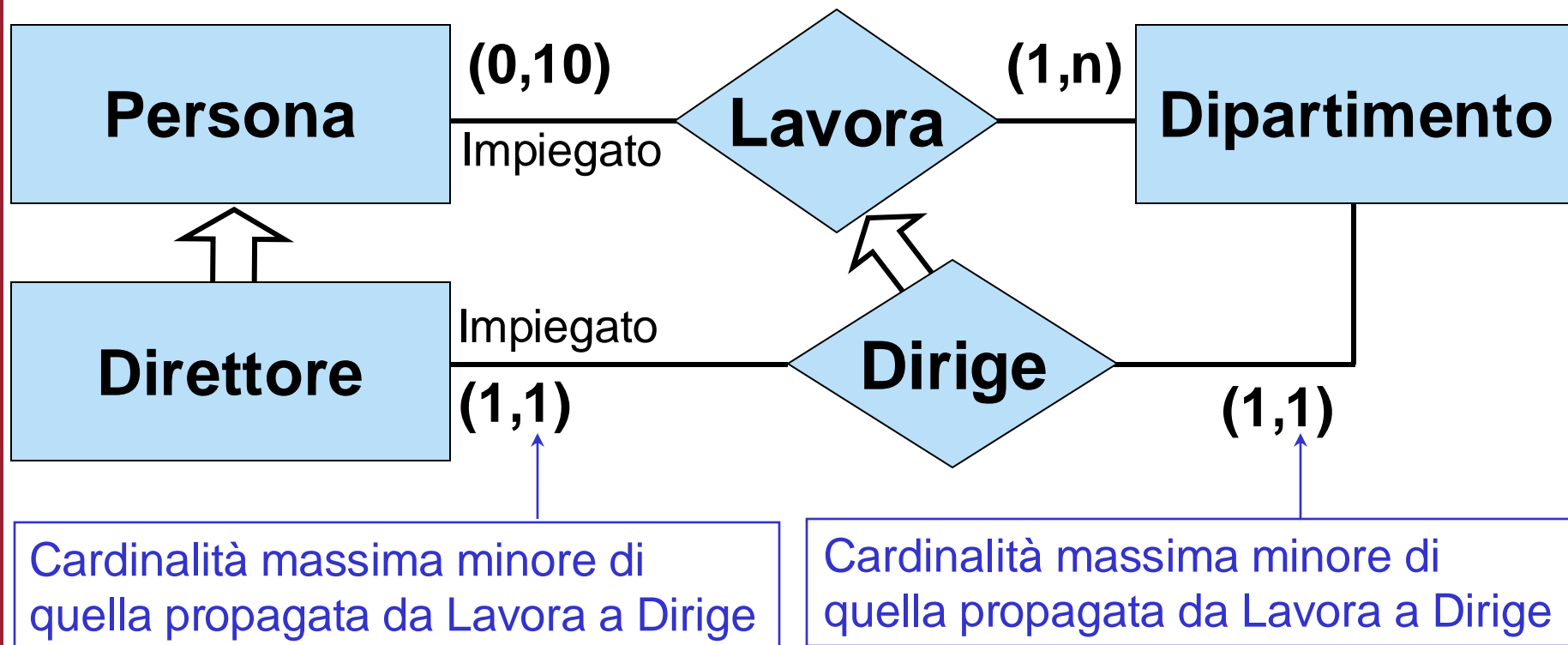
Se Q ISA R , ogni vincolo di cardinalità **massima** definito sulla relazione R (per il ruolo U) viene propagato (seguendo l'ereditarietà) sulla relazione Q (per lo stesso ruolo U), che può però avere cardinalità massima più stringente di quella propagata, ovvero **cardinalità massima minore**.





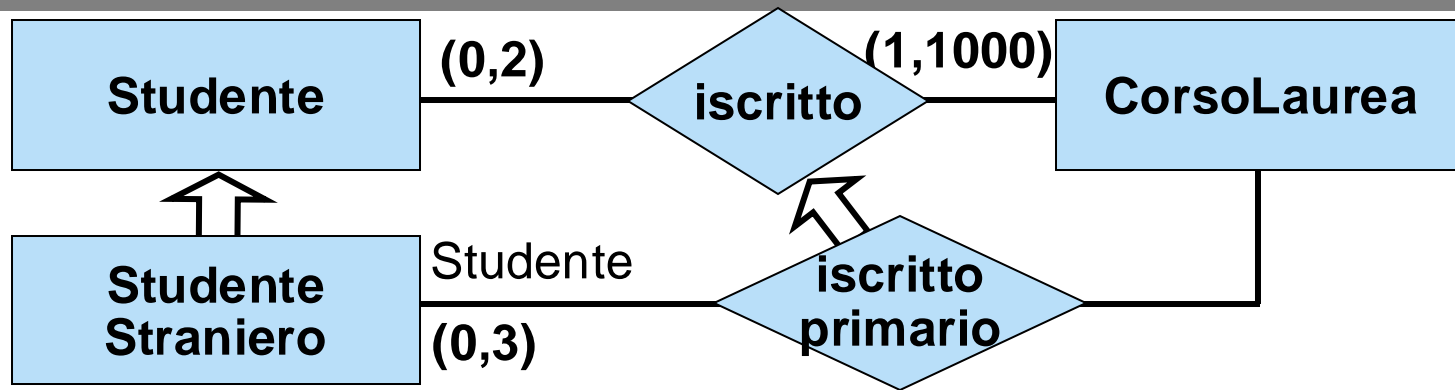
Vincoli di cardinalità massima ed ereditarietà su relazioni

La regola sulla propagazione della cardinalità **massima** vale anche se l'entità corrispondente al ruolo U nella relazione figlia è diversa dalla (cioè è una sottoentità della) entità corrispondente al ruolo U nella relazione padre

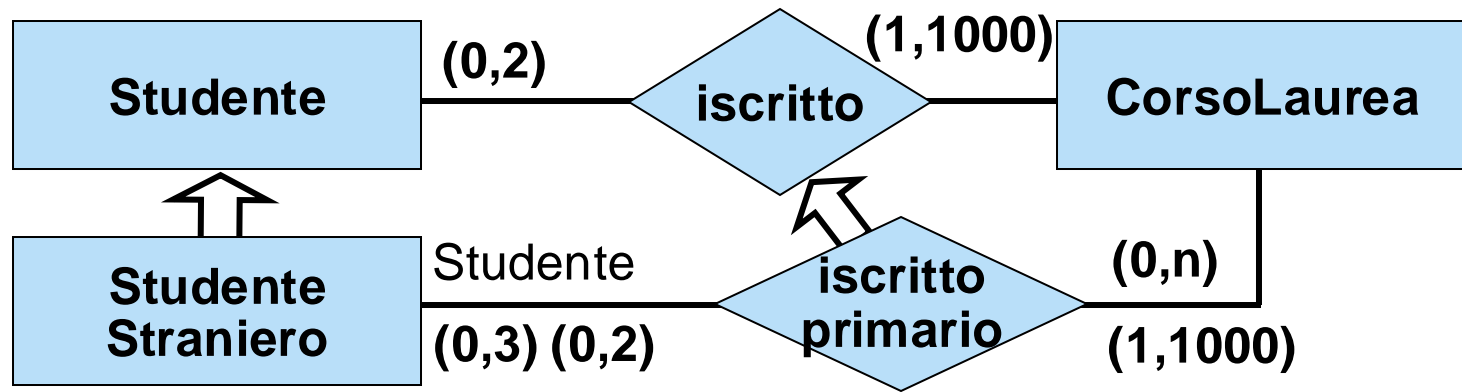




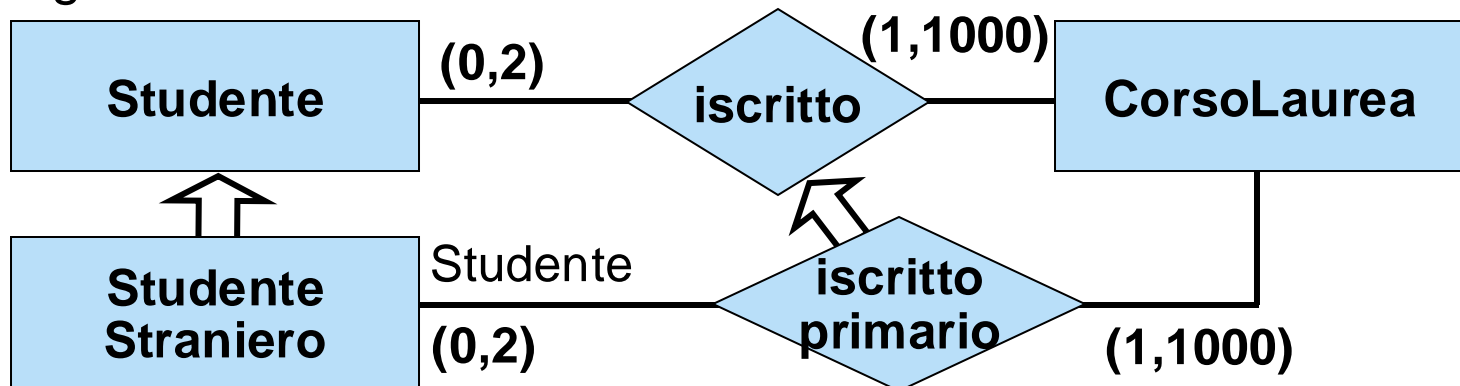
Esempio



Propagazione:

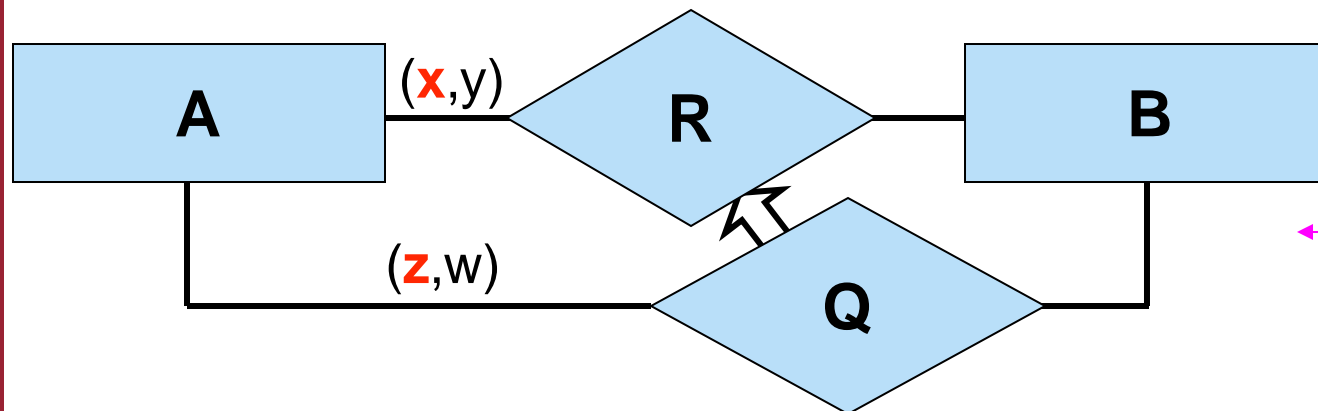


Vincoli più stringenti:

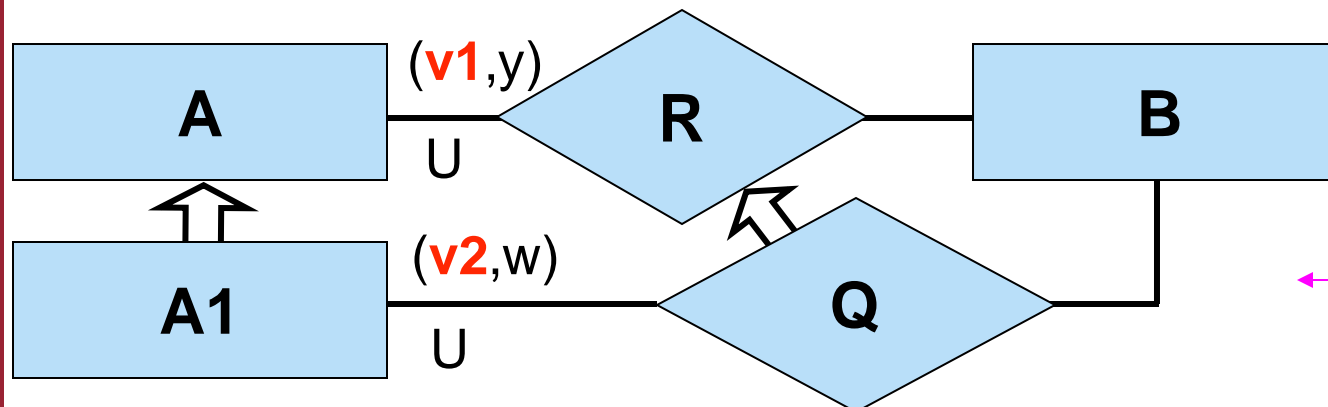


Vincoli di cardinalità minima ed ereditarietà su relazioni

Se Q ISA R , il rapporto tra i vincoli di cardinalità **minima** su Q ed R sono più complessi rispetto al caso dei vincoli di cardinalità massima. Nel primo caso qui sotto la propagazione è "al contrario". Nel secondo caso non c'è alcuna propagazione.



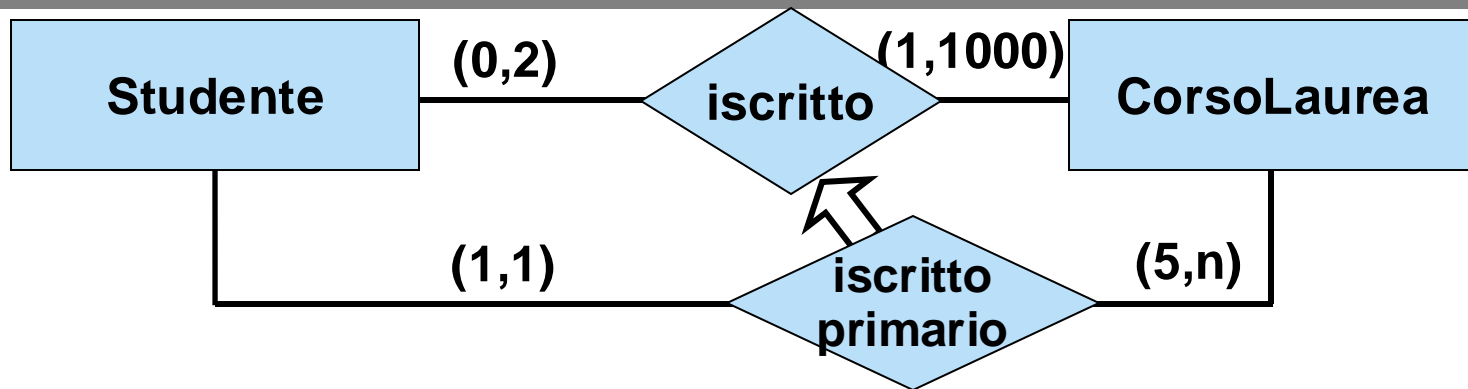
La cardinalità minima z nel ruolo A di Q si propaga su R . Quindi, se x è minore di z , si può sostituire x con z ed ottenere uno schema equivalente più accurato, nel senso che esso mostra il vincolo più stringente



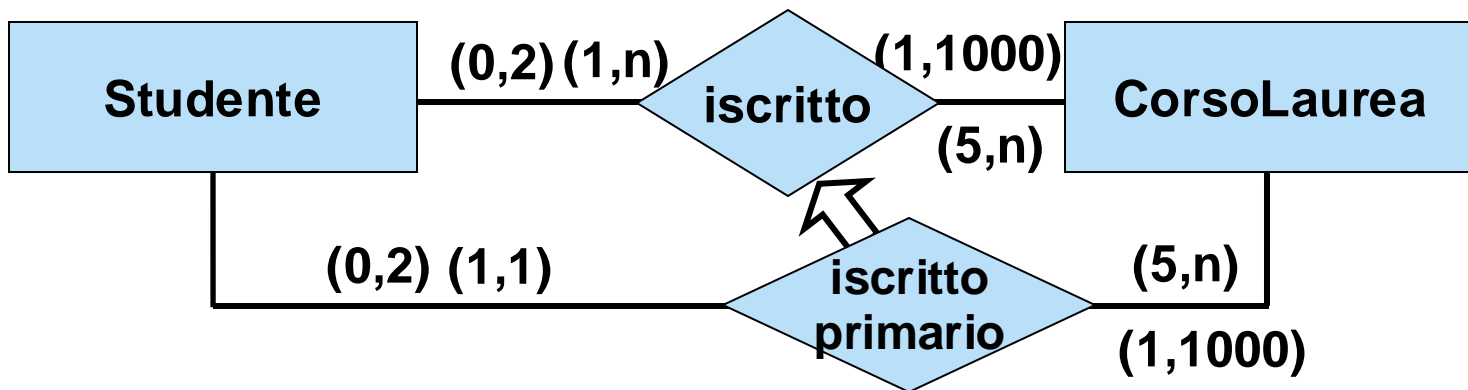
$v1$ e $v2$ sono totalmente scorrelati



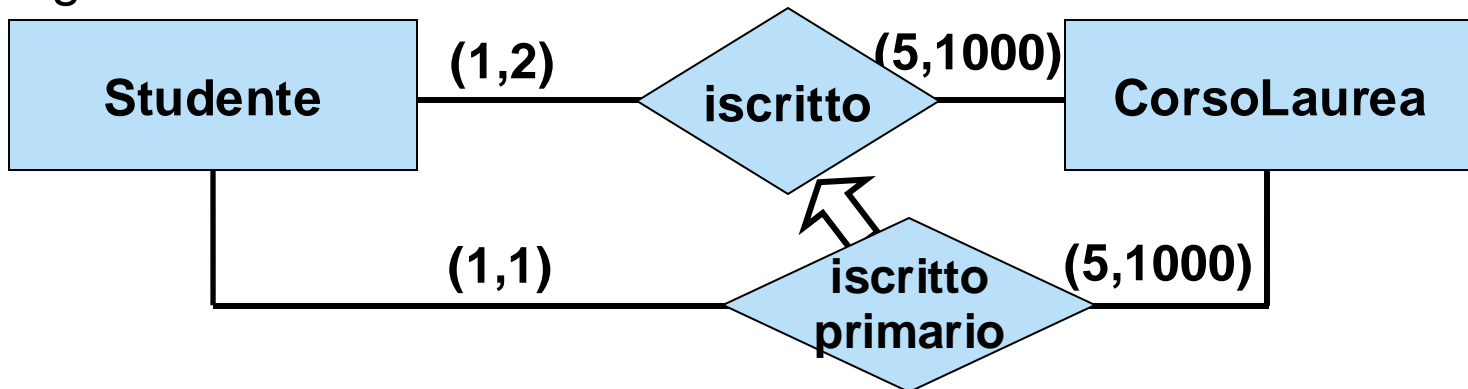
Esempio



Propagazione:

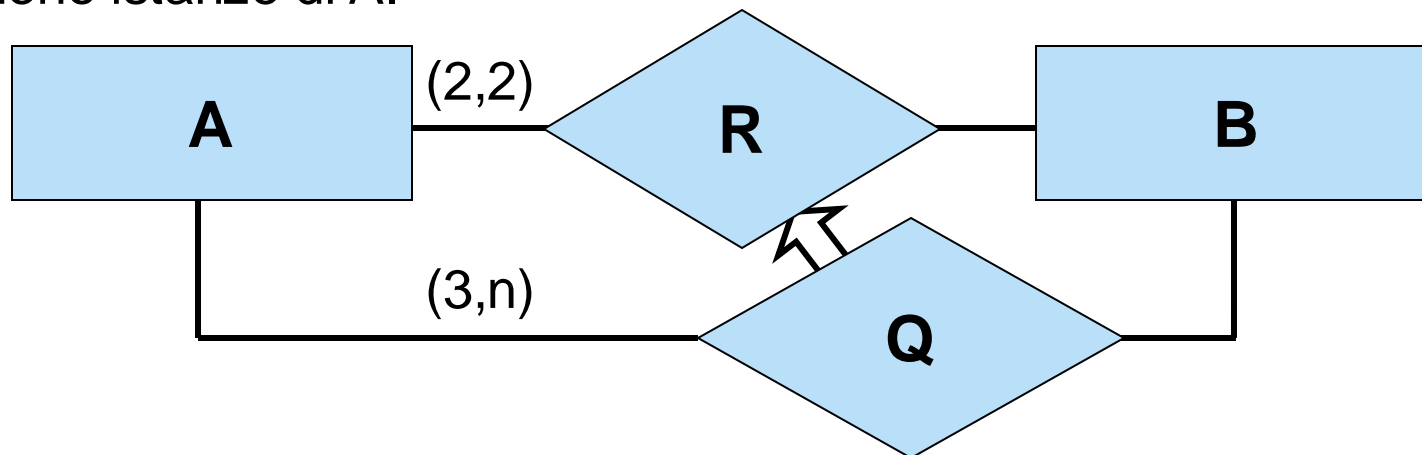


Vincoli più stringenti:



Vincoli di cardinalità ed ereditarietà su relazioni

La regola, già introdotta, che asserisce che per un certo ruolo U di una relazione R la **cardinalità minima associata ad U deve essere minore o uguale alla cardinalità massima associata ad U** rimane valida anche nel caso in cui i vincoli di cardinalità vengono propagati. Ad esempio, il seguente schema S non segue questa regola, perché nel ruolo A di R i vincoli di cardinalità stringenti sono $(3,2)$: infatti la cardinalità minima è propagata da quella di Q e la cardinalità massima è quella esplicitamente dichiarata. La regola è importante, perché serve ad evitare che l'insieme delle istanze di una relazione o di una entità sia sempre vuoto: infatti non è difficile dimostrare che in ogni istanza dello schema S mostrato qui sotto non possono esistere istanze della relazione R (e quindi anche di Q) e perciò, vista la partecipazione obbligatoria di A ad R , non possono esistere nemmeno istanze di A .

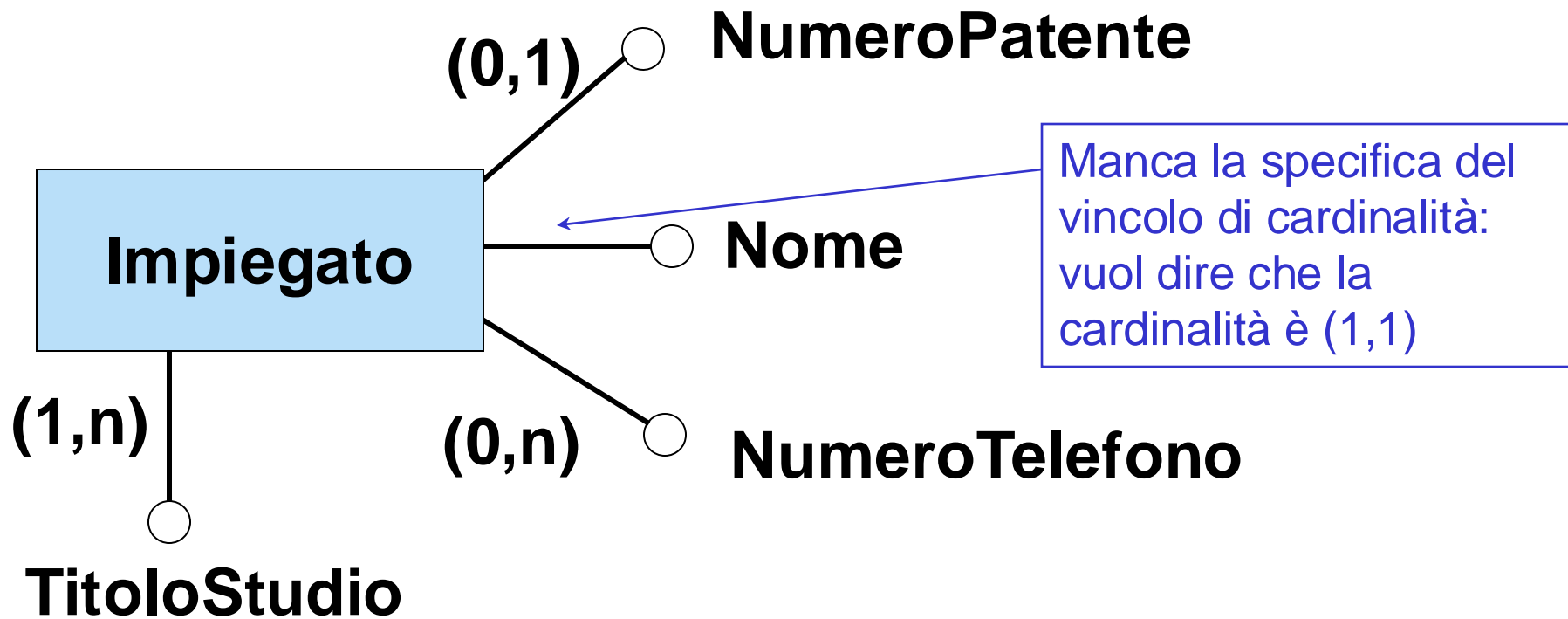




Vincoli di cardinalità sugli attributi

- È possibile definire vincoli di cardinalità anche sugli attributi, con due scopi:
 - indicare **opzionalità**
 - indicare **attributi multivalore**
- La semantica di un attributo con cardinalità diverso da $(1,1)$ è diversa dal caso di cardinalità $(1,1)$:
 - Quando la cardinalità massima di un attributo non è 1, l'attributo si dice multivalore, e non rappresenta più una funzione ma una **relazione**; le cardinalità stabiliscono il numero minimo e massimo di valori che l'attributo associa ad ogni istanza dell'entità
 - Quando la cardinalità minima di un attributo è 0, la relazione (o la funzione, se la cardinalità massima è uguale a 1) rappresentata dall'attributo **non è più totale**
- Se la specifica del vincolo manca, si intende che la cardinalità dell'attributo sia $(1,1)$, e la semantica è quella usuale

Rappresentazione grafica dei vincoli di cardinalità sugli attributi





Osservazione sulla semantica degli attributi

Se in uno schema **S** un attributo **A** è definito in n entità E_1, \dots, E_n , rispettivamente con domini D_1, \dots, D_n , e con cardinalità $(x_1, y_1), \dots, (x_n, y_n)$, in ogni istanza **I** dello schema **S**, **A** è una relazione (o una funzione se tutte le cardinalità massime y_1, \dots, y_n sono pari a 1) del tipo

$$\text{istanze}(\mathbf{I}, \mathbf{A}) \subseteq (\text{istanze}(\mathbf{I}, E_1) \cup \dots \cup \text{istanze}(\mathbf{I}, E_n)) \times (D_1 \cup \dots \cup D_n)$$

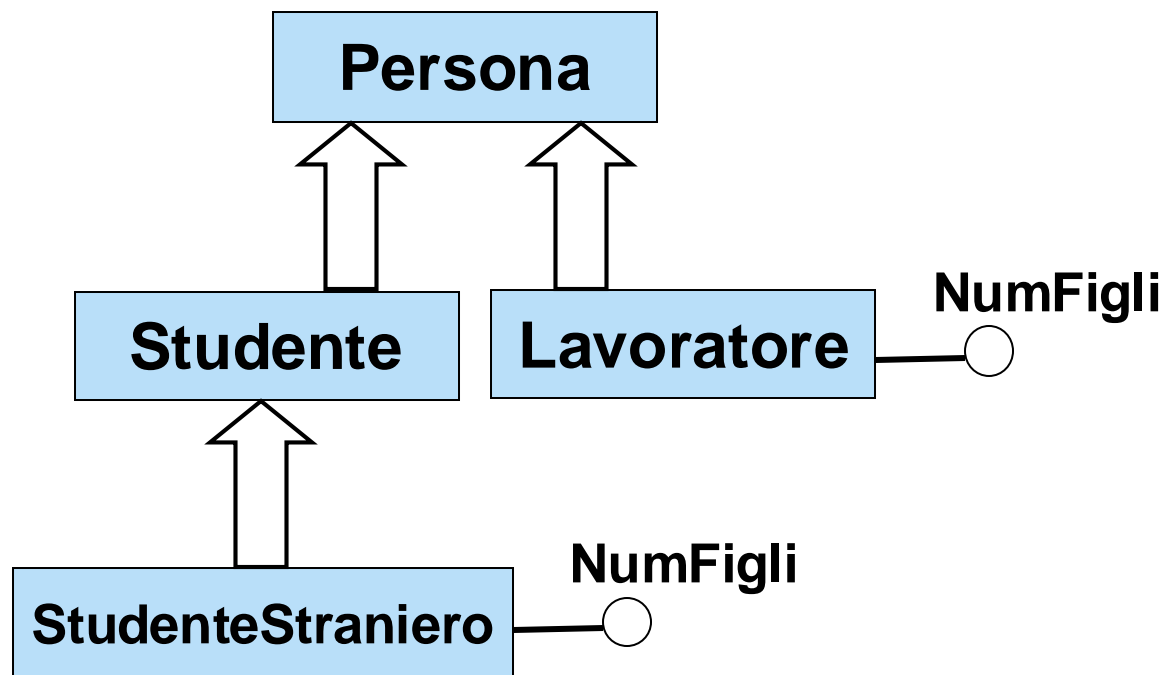
e tale che, per ogni **i** vale la seguente condizione:

- se $a \in \text{istanze}(\mathbf{I}, E_i)$, allora il numero di coppie (a, b) in $\text{istanze}(\mathbf{I}, \mathbf{A})$ è soggetto ai vincoli di cardinalità (x_i, y_i) , e per ogni $(a, b) \in \text{istanze}(\mathbf{I}, \mathbf{A})$, si ha che $b \in D_i$

Da ciò segue che occorre porre attenzione ai nomi assegnati agli attributi: quando si assegna lo stesso nome a due attributi di entità diverse, si deve ricordare che questi, a livello estensionale, rappresenteranno un'unica relazione (o funzione).

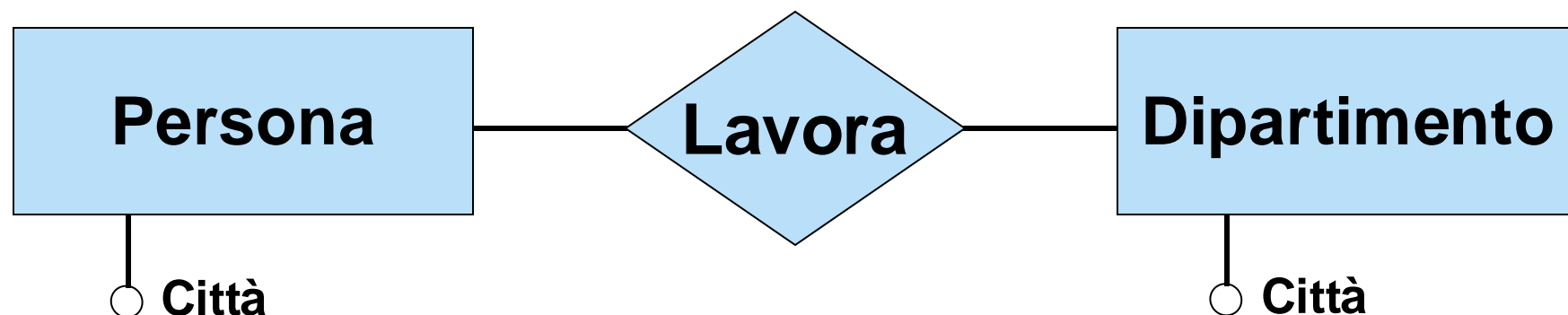
Questa osservazione vale anche per uno stesso attributo definito su relazioni diverse.

Esempio di attributo definito su più entità



La condizione descritta in precedenza impone che l'attributo NumFigli nelle due entità Lavoratore e StudenteStraniero rappresenti in realtà la stessa funzione. Ne segue che, se uno stesso oggetto è istanza sia di Lavoratore sia di StudenteStraniero, esso ha un unico valore per l'attributo NumFigli

Attributo definito su più entità disgiunte



Come abbiamo visto in precedenza, l'attributo **Città** nelle due entità **Persona** e **Dipartimento** rappresenta in realtà la stessa funzione. Tuttavia, **Persona** e **Dipartimento** sono entità disgiunte, e quindi è possibile considerare la funzione come l'unione disgiunta di due funzioni distinte, una definita sulle istanze di **Persona** ed una definita sulle istanze di **Dipartimento**.



Vincoli di identificazione di entità

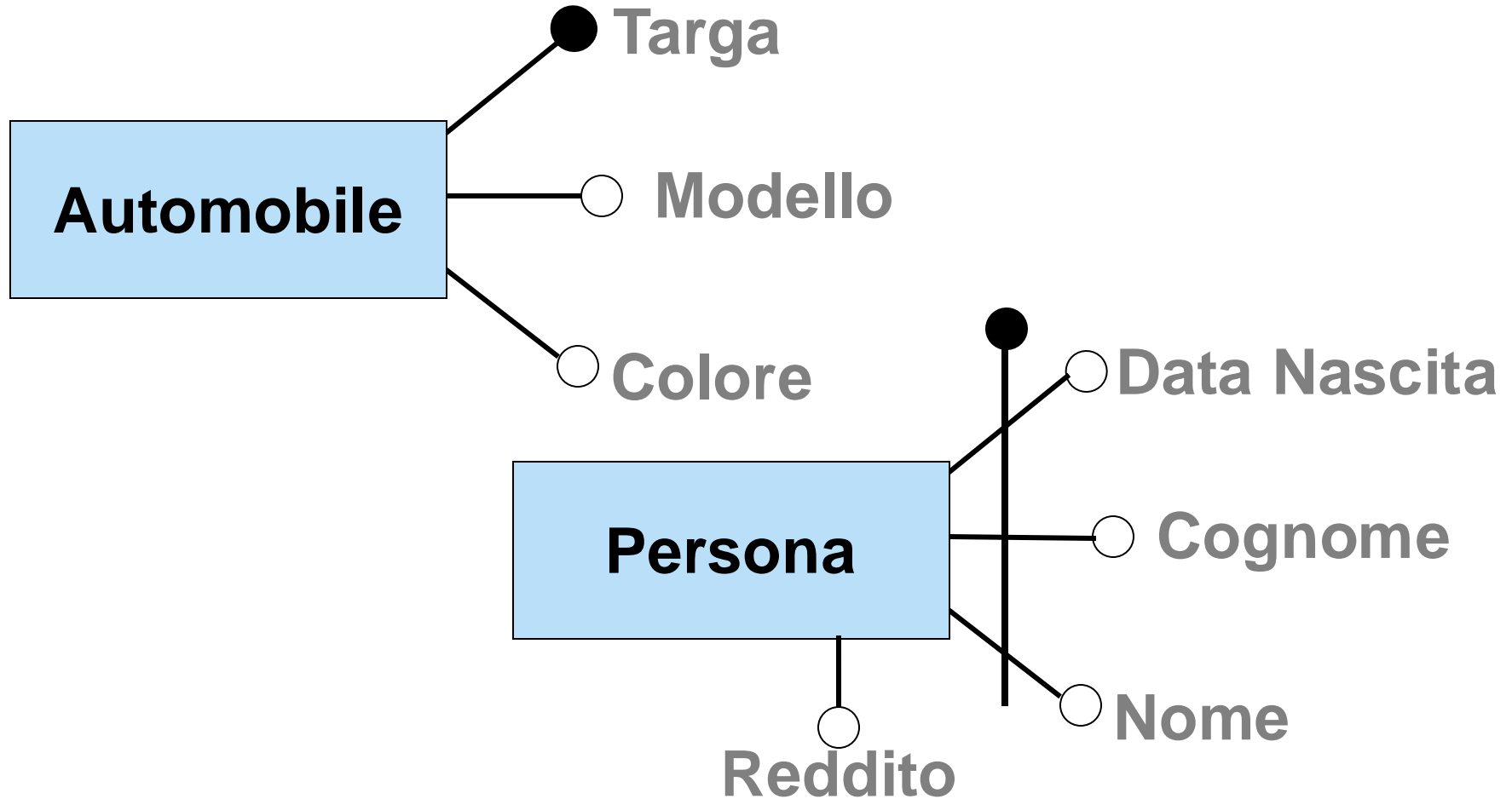
- Un identificatore di una entità è un insieme di proprietà (attributi o relazioni) che permettono di identificare univocamente le istanze di un'entità. In altre parole, in ogni livello estensionale, non esistono due istanze dell'entità che hanno lo stesso valore per tutte le proprietà che formano l'identificatore.
- Un **vincolo di identificazione** per una entità E definisce un identificatore per E
- Su ogni entità si può definire un numero qualunque di vincoli di identificazione (tipicamente almeno uno, anche se è possibile che un'entità non abbia alcun identificatore)
- Il fatto che per una entità E possa non essere definito alcun identificatore non contraddice il fatto che ogni istanza di E è identificabile in modo univoco: infatti ogni oggetto (e quindi anche ogni istanza di E) ha implicitamente un «object identifier» per distinguerlo da tutti gli altri oggetti



Sintassi dei vincoli di identificazione di entità

- Un identificatore di una entità E può essere
 - **Interno**, ossia formato solo da attributi di E
 - **Esterno**, ossia formato da attributi di E e da ruoli di relazioni che coinvolgono E, oppure solo da ruoli di relazioni che coinvolgono Econ **la regola che tutti gli attributi e tutti i ruoli di relazione che concorrono ad un identificatore di entità devono avere cardinalità (1,1)**
- Notazione per gli identificatori interni:
 - Se l'identificatore è formato da un solo attributo, si annerisce il corrispondente pallino.
 - Se l'identificatore è formato da più attributi, si uniscono tali attributi con una linea che termina con pallino annerito.
- Notazione per gli identificatori esterni:
 - Se l'identificatore è formato da attributi e relazioni (o meglio ruoli), esso si indica unendo gli attributi ed i ruoli con una linea che termina con pallino annerito.

Esempi di identificatori interni di entità





Semantica dei vincoli di identificazione di entità

Se in uno schema **S** è definito un vincolo di identificazione che specifica un identificatore per l'entità **E** formato

- dagli attributi **A₁, A₂, ..., A_k** (tutti di cardinalità (1,1))
- dalle relazioni **R₁** (ruolo **U₁**), **R₂** (ruolo **U₂**), ..., **R_h** (ruolo **U_h**) (tutti ruoli ai quali è associata la cardinalità (1,1))

allora in ogni istanza **I** dello schema **S**, prese due diverse istanze qualunque **e₁** ed **e₂** in *istanze(I, E)*, esse differiscono nel valore di almeno un **A_i** o nella partecipazione con ruolo **U_i** in almeno una **R_i**.

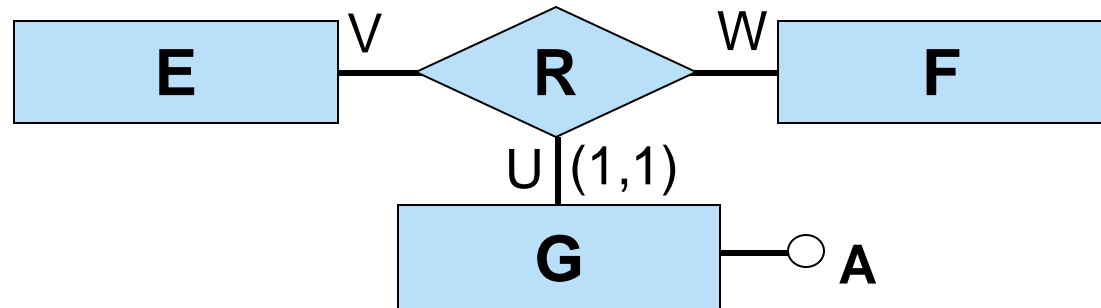
Quando diciamo che **e₁** ed **e₂** in *istanze(I, E)* differiscono nella partecipazione con ruolo **U_i** in almeno una **R_i** intendiamo che l'istanza di **R_i** a cui partecipa **e₁** e l'istanza di **R_i** a cui partecipa **e₂** non solo hanno diverse istanze di **E** nel ruolo **U_i**, ma differiscono anche in almeno un ruolo **U_j** diverso da **U_i**.



Istanze che differiscono nella partecipazione ad una relazione

Cosa vuol dire che in una istanza dello schema due istanze della stessa entità differiscono per il valore di un attributo oppure per la partecipazione con un ruolo in una relazione? Illustriamolo con un esempio.

Schema S



Istanza I di S:

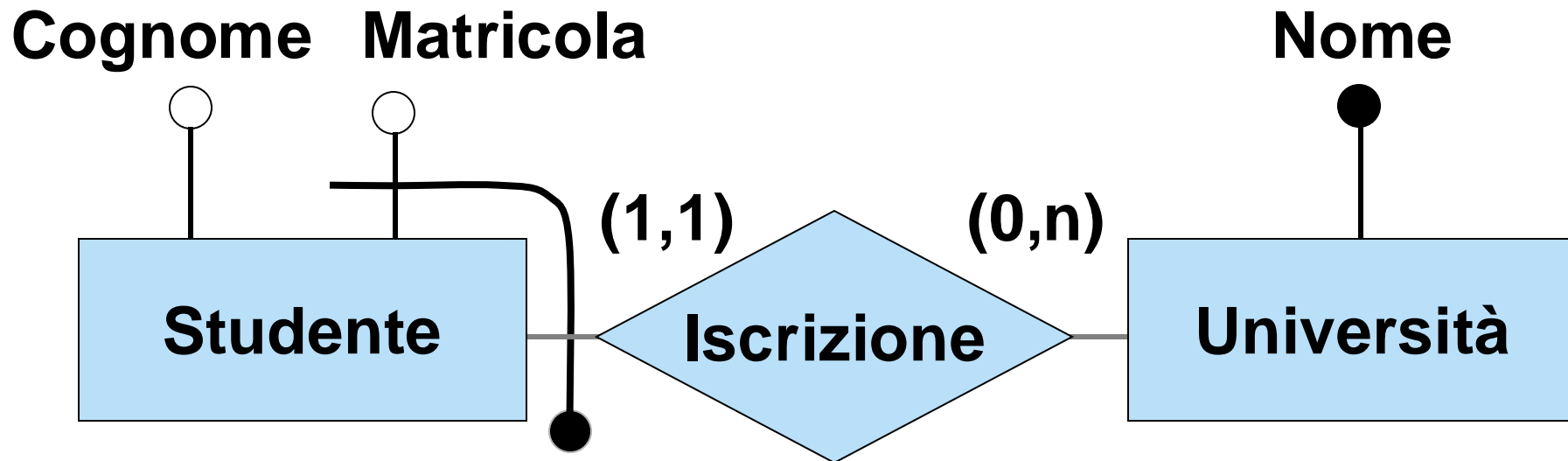
Istanze(I,E) = {a,b,c} Istanze(I,F) = {n,p,s}

Istanze(I,G) = {d,g,m} Istanze(I,A) = {<d,1>, <g,2>, <m,2> }

Istanze(I,R) = { <U:d,V:a,W:n>, <U:g,V:a,W:n>, <U:m,V:a,W:p> }

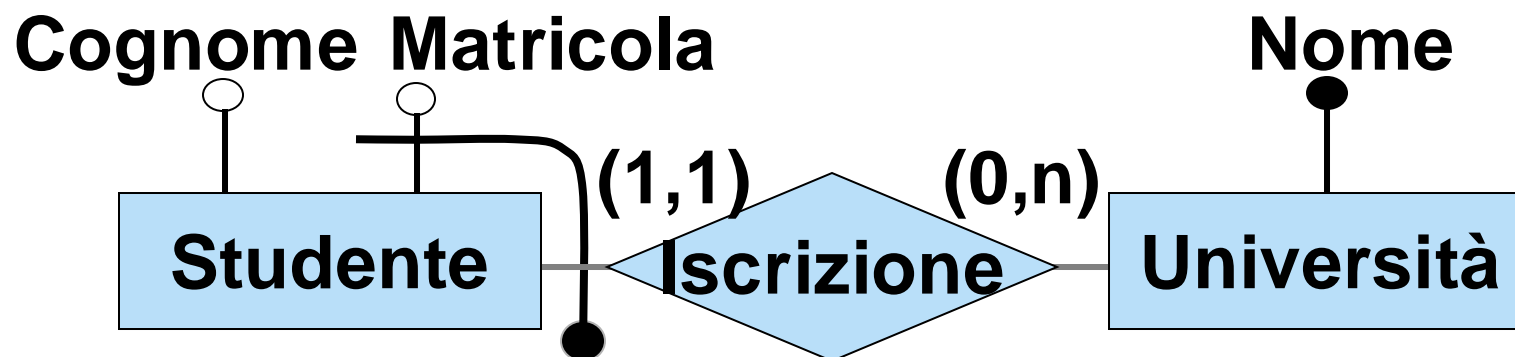
Nella istanza I di S, d e g differiscono nel valore di A, ma **non** differiscono nella partecipazione con ruolo U in R. Al contrario, g ed m **non** differiscono nel valore di A, ma differiscono nella partecipazione nel ruolo U in R, perché nelle relative istanze di R alle quali partecipano troviamo istanze diverse nel ruolo W (nella istanza alla quale partecipa g troviamo n, mentre nella istanza alla quale partecipa m troviamo p). Le istanze d ed m differiscono in entrambi gli aspetti, cioè sia nel valore di A sia nell partecipazione con ruolo U in R

Esempio di identificatore esterno di entità



Nota: Un'entità che ha un identificatore esterno viene detta **entità debole**

Identificatore esterno di entità: livello estensionale



Istanza I:

istanze(I, Studente) = { a, b, c }

istanze(I, Università) = { v, z }

istanze(I, Matricola) = { (a, 12), (b, 24), (c, 12) }

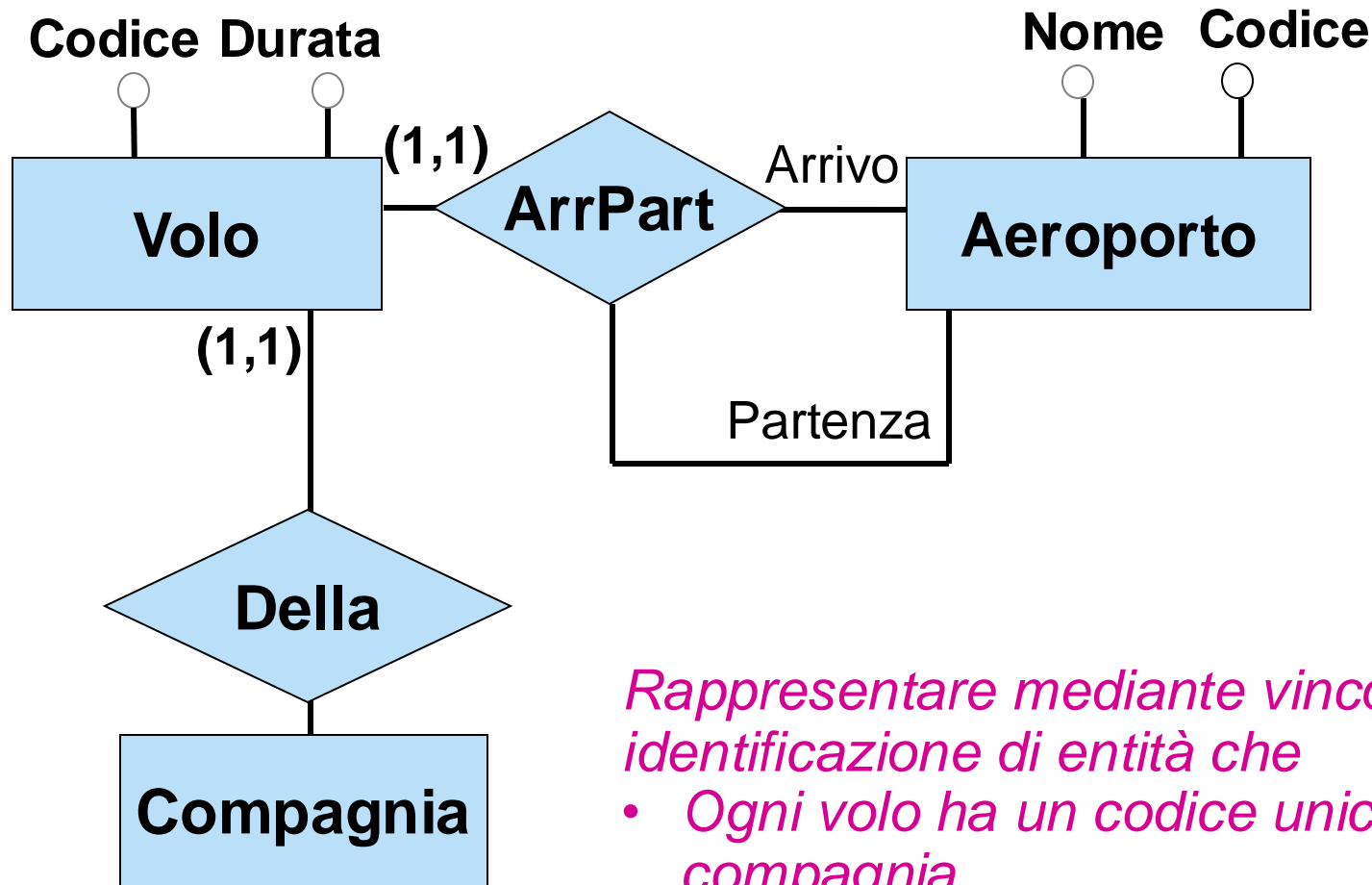
istanze(I, Cognome) = { (a, 'Rossi'), (b, 'Finzi'), (c, 'Rossi') }

istanze(I, Nome) = { (v, 'Viterbo'), (z, 'Camerino') }

istanze(I, Iscrizione) = { (Studente:a, Università:v),
(Studente:b, Università:v),
(Studente:c, Università:z) }

Per ogni coppia di istanze di Studente, tali istanze differiscono o nel valore di Matricola o nella partecipazione a Iscrizione nel ruolo Studente

Esercizio 17: vincoli di identificazione di entità

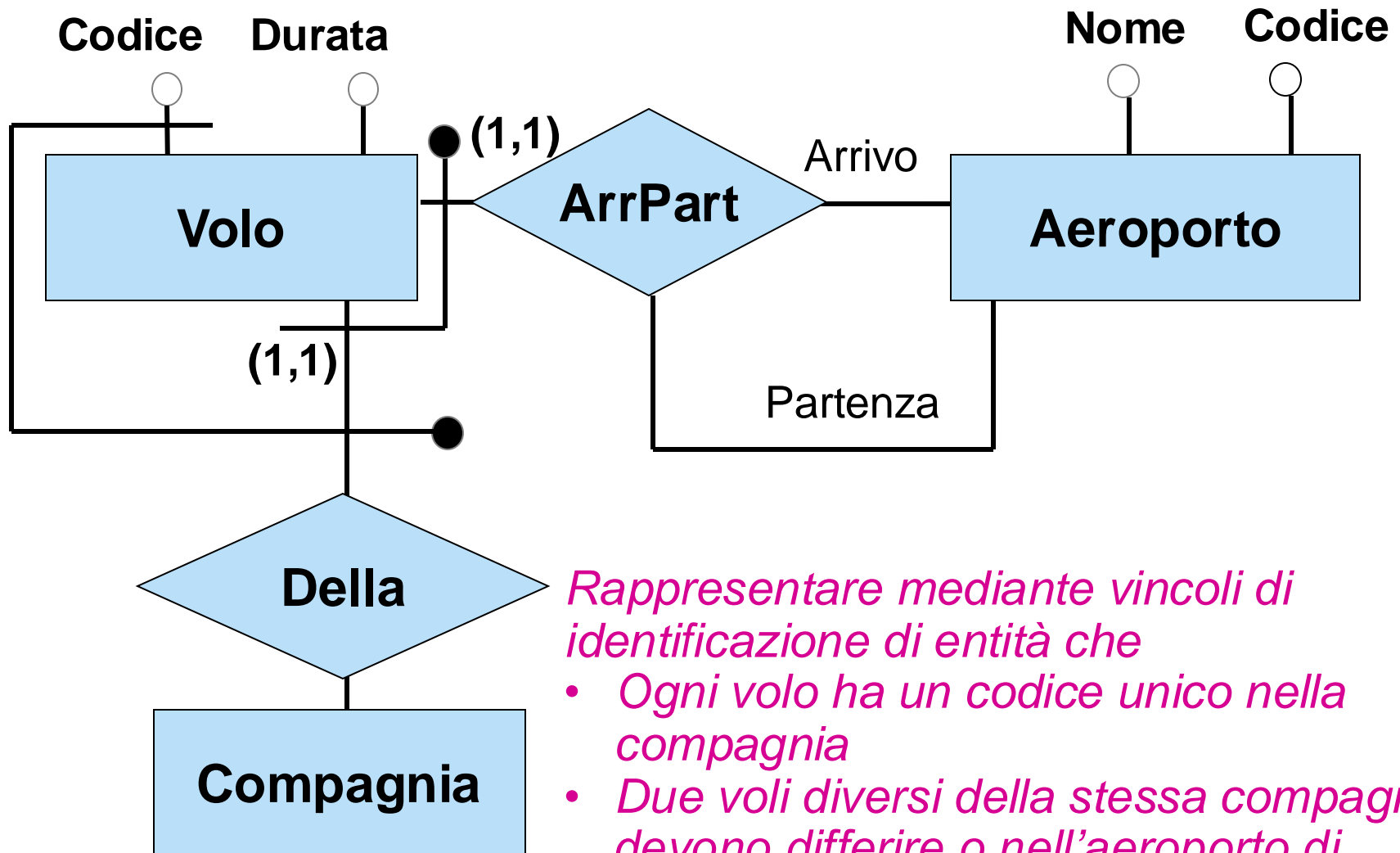


Rappresentare mediante vincoli di identificazione di entità che

- Ogni volo ha un codice unico nella compagnia*
- Due voli diversi della stessa compagnia devono differire nell'aeroporto di arrivo o nell'aeroporto di partenza*



Esercizio 17: soluzione

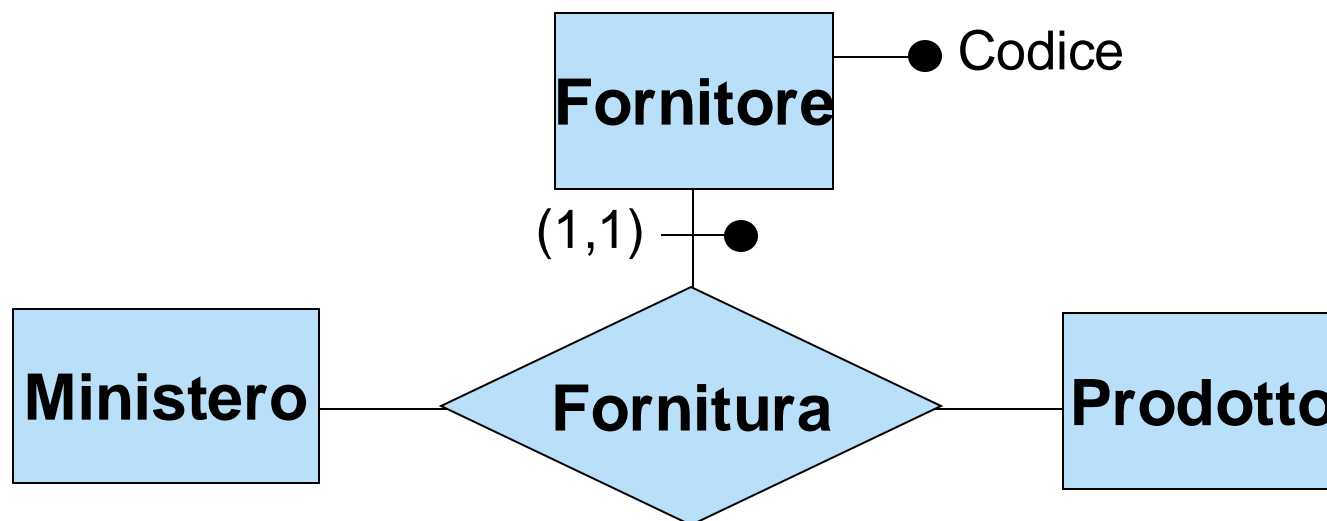


Rappresentare mediante vincoli di identificazione di entità che

- Ogni volo ha un codice unico nella compagnia*
- Due voli diversi della stessa compagnia devono differire o nell'aeroporto di arrivo o nell'aeroporto di partenza*

Esempio di identificazione esterna di entità

I fornitori accreditati forniscono prodotti ai ministeri, con il vincolo che un fornitore partecipi esattamente ad una fornitura. Un prodotto non può essere fornito allo stesso ministero da due fornitori diversi. Questo vuol dire che due fornitori diversi non possono partecipare a due istanze della relazione Fornitura che hanno la stessa coppia (Ministero, Prodotto), e questo a sua volta vuol dire che la relazione Fornitura, nel ruolo Fornitore, è un identificatore dell'entità Fornitore.

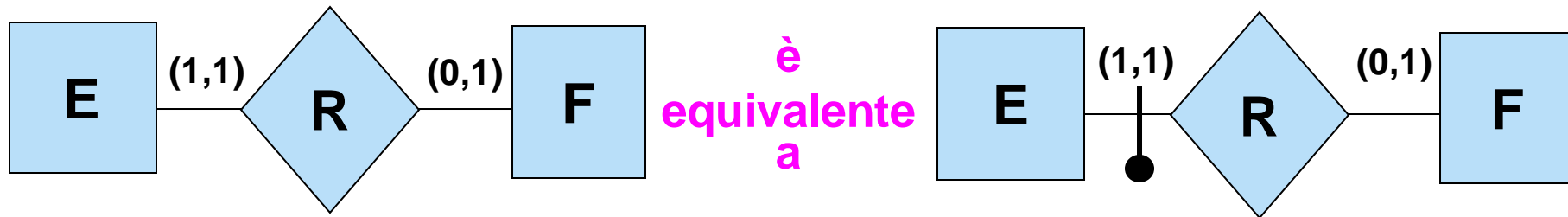


L'esempio mostra che un identificatore esterno di entità può anche non comprendere attributi, e può coinvolgere una sola relazione attraverso un unico ruolo.



Osservazione sull'identificazione esterna di entità

Sia E una entità che partecipa ad una relazione R nel ruolo U con cardinalità $(1,1)$, e supponiamo che almeno uno dei ruoli di R diversi da U abbia cardinalità massima 1. È facile verificare che questa situazione implica che E è identificata esternamente da R tramite U . Tale identificatore esterno può essere esplicitamente indicato, ma non è necessario perché è già implicito nella cardinalità $(1,1)$ del ruolo U e nella cardinalità massima di U pari ad 1. Ad esempio:



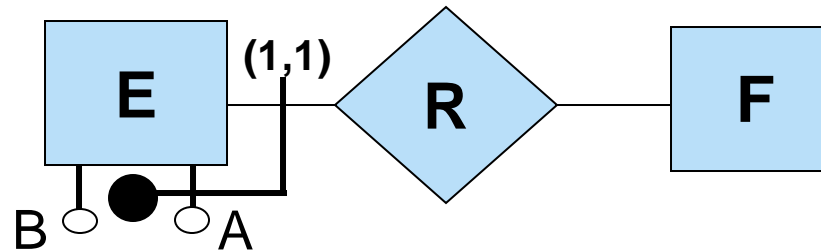
Infatti, se in una istanza dello schema non valesse il vincolo di identificazione esterno per E , allora esisterebbero due istanze di E connesse con una stessa istanza f di F tramite R , e questo violerebbe il vincolo di cardinalità massima di F in R .



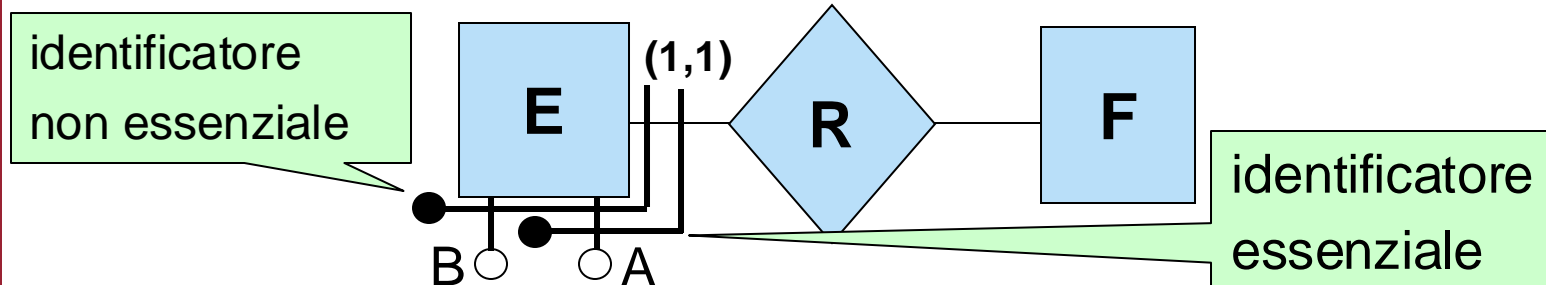
Identificatori “essenziali”

Un identificatore dell'entità E formato dall'insieme W di proprietà (attributi o ruoli) si dice **essenziale** se ogni insieme ottenuto da W togliendo un elemento non è più un identificatore per E.

Ad esempio, supponendo che non esistano due istanze di E con lo stesso valore per A e la stessa partecipazione ad R, il seguente identificatore è essenziale per E:

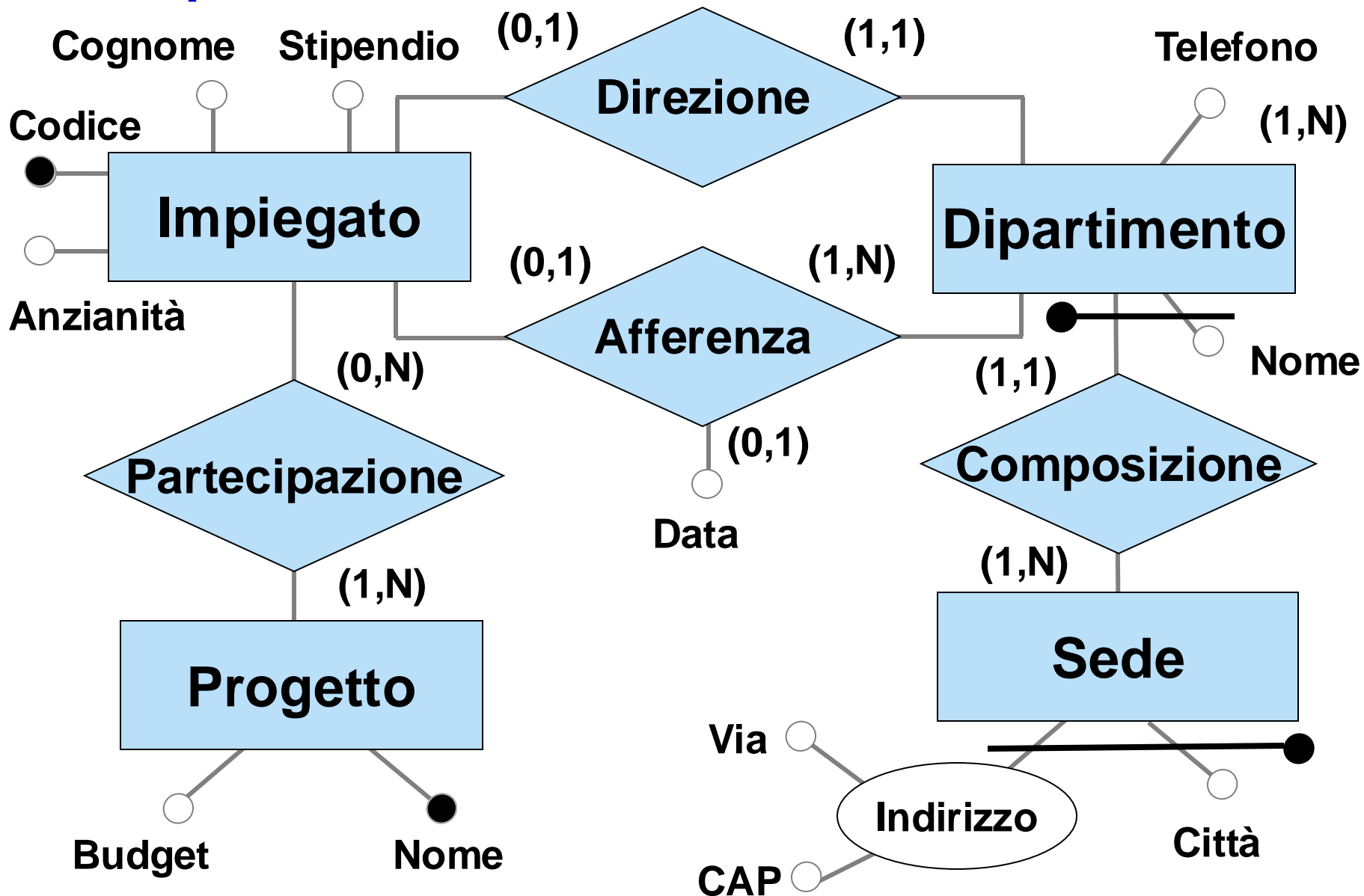


Al contrario, l'identificatore che include anche l'attributo B non è ovviamente essenziale:





Esempio di vincoli di cardinalità e identificazione





Vincoli di identificazione di relazione

Anche per le relazioni si possono definire vincoli di identificazione.

- Un identificatore di una relazione è un insieme di proprietà (attributi di cardinalità (1,1) o ruoli) che permettono di identificare univocamente le istanze di una relazione, nel senso che non possono esistere due istanze della relazione che hanno lo stesso valore per tutte le proprietà che formano l'identificatore.
- Un **vincolo di identificazione** per una relazione R definisce un identificatore per R.
- Per ogni relazione si può dichiarare un numero qualunque di vincoli di identificazione.

La definizione precedente implica che **su ogni relazione è implicitamente definito un identificatore, detto appunto implicito**, formato da tutti ruoli che partecipano alla relazione. Infatti, la semantica della relazione stabilisce che non possono esistere due istanze della relazione con le stesse componenti nei vari ruoli. Si noti che questo vincolo di identificazione implicito, per prassi, non si indica nello schema concettuale.

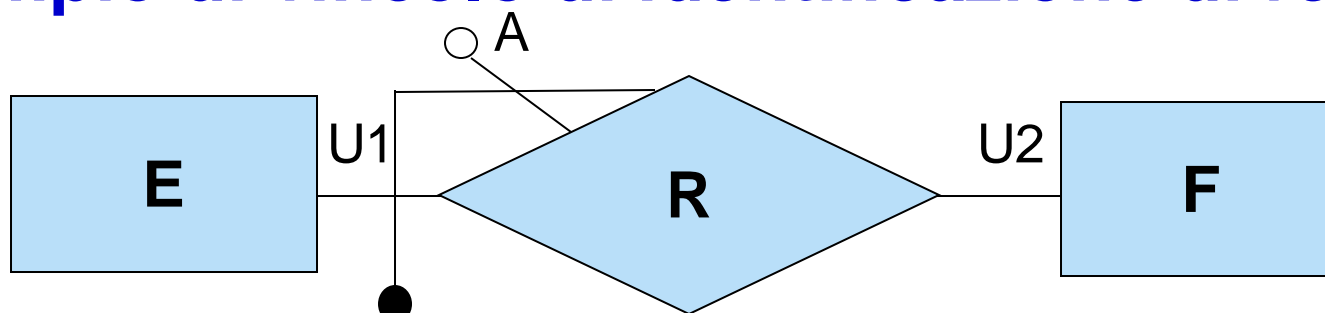


Sintassi dei vincoli di identificazione di relazione

La sintassi per gli identificatori di relazione è analoga a quella per gli identificatori di entità, ma con una variante: **la linea che indica un identificatore di relazione è collegata al rombo corrispondente alla relazione stessa**. Più precisamente:

- Se l'identificatore è formato da un solo attributo (di cardinalità $(1,1)$) della relazione, si annerisce il corrispondente pallino dell'attributo di relazione.
- Se l'identificatore è formato da più attributi della relazione (tutti di cardinalità $(1,1)$), si uniscono tali attributi con una linea che parte dalla relazione e termina con un pallino annerito.
- Se l'identificatore è formato da attributi (di cardinalità $(1,1)$) e ruoli oppure solo da ruoli, esso si indica unendo gli attributi ed i ruoli che formano l'identificatore con una linea che parte dalla relazione e termina con un pallino annerito.
- Gli identificatori impliciti tipicamente non si indicano nello schema; quelli derivati (vedi dopo) si possono indicare o no.

Esempio di vincolo di identificazione di relazione



Nello schema S sopra descritto è presente un vincolo di identificazione di relazione, indicato con una linea che parte dalla relazione R e termina con un pallino. La linea unisce l'attributo A (di cardinalità (1,1) ed il ruolo U1, e ciò significa che A ed U1 formano un identificatore per R.

Si noti la **differenza di notazione tra l'identificatore di entità e l'identificatore di relazione**: quello nella figura qui sopra è chiaramente un identificatore di relazione, perché la linea che termina in un pallino nero parte dalla relazione R (al contrario, le linee degli identificatori di entità uniscono semplicemente le proprietà che formano l'identificatore, senza partire dal simbolo di entità).



Semantica dei vincoli di identificazione di relazione

Se in uno schema **S** è definito un vincolo di identificazione che specifica un identificatore per la relazione **R** formato

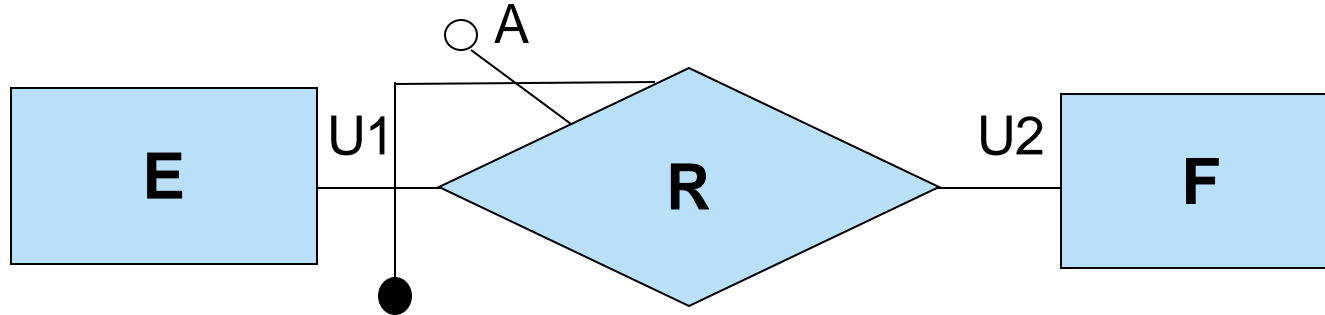
- dagli attributi **A₁**, **A₂**, ..., **A_k** (di cardinalità (1,1))
- dai ruoli **U₁**, ..., **U_h**,

allora in ogni istanza **I** dello schema **S**, prese due diverse istanze **r₁** ed **r₂** in *istanze(I, R)*, esse differiscono per il valore di almeno un **A_i** o per almeno una istanza di entità nei ruoli **U₁**, ..., **U_h**

Si noti che il concetto di **identificatore essenziale** si applica anche al caso di identificatore di relazione

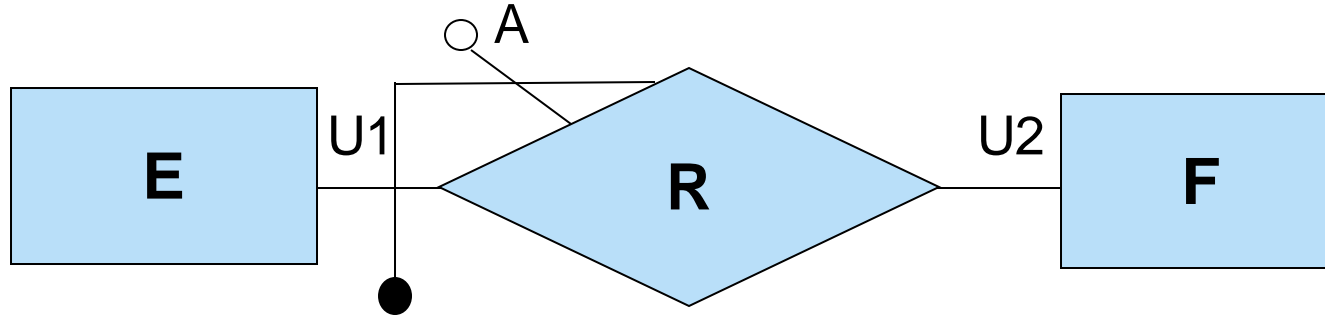


Esempio di vincolo di identificazione di relazione



Nello schema S sopra descritto, la semantica del vincolo di identificazione di relazione si può descrivere nel seguente modo: non possono esistere due istanze di R che sono associate allo stesso valore di A ed hanno nella componente corrispondente al ruolo U1 la stessa istanza di entità.

Esempio di vincolo di identificazione di relazione



Riferendoci ancora allo schema S , è facile verificare che la seguente istanza I di S

$Istanze(I, E) = \{ e1, e2 \}$

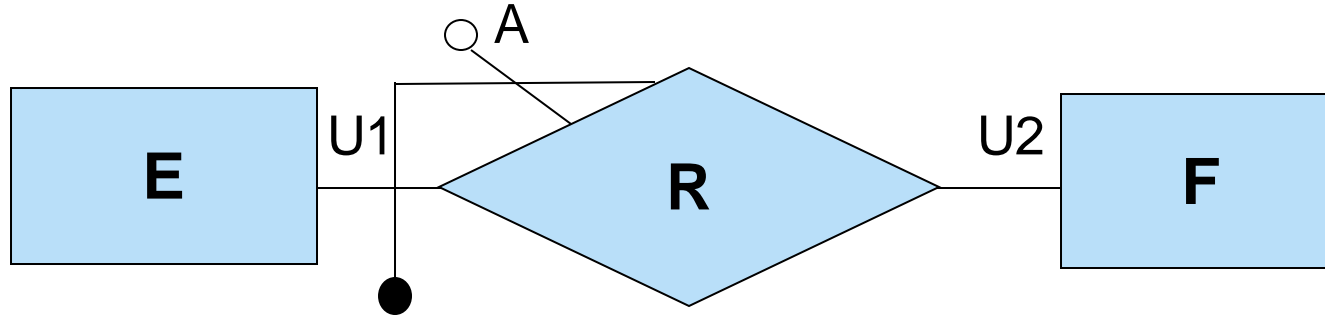
$Istanze(I, F) = \{ f1, f2 \}$

$Istanze(I, R) = \{ \langle U1:e1, U2:f1 \rangle, \langle U1:e1, U2:f2 \rangle, \langle U1:e2, U2:f2 \rangle \}$

$Istanze(I, A) = \{ \langle \langle U1:e1, U2:f1 \rangle, v1 \rangle, \langle \langle U1:e1, U2:f2 \rangle, v1 \rangle, \langle \langle U1:e2, U2:f2 \rangle, v2 \rangle \}$

non è legale, perché le due istanze $\langle U1:e1, U2:f1 \rangle$ e $\langle U1:e1, U2:f2 \rangle$ di R hanno lo stesso valore (ossia $v1$) per l'attributo A , ed hanno la stessa istanza di entità (ossia $e1$) nel ruolo $U1$.

Esempio di vincolo di identificazione di relazione



Al contrario, la seguente istanza J di S è legale:

Istanze(I,E) = { e1,e2 }

Istanze(I,F) = { f1,f2 }

Istanze(I,R) = { <U1:e1,U2:f1>, <U1:e1,U2:f2>, <U1:e2,U2:f2> }

Istanze(I,A) = { <<U1:e1,U2:f1>,v1>, <<U1:e1,U2:f2>,v2>,
<<U1:e2,U2:f2>,v2> }

perché, prese due qualunque istanze di R, esse differiscono o per il valore dell'attributo A, o per l'istanza di entità nel ruolo U1.



Vincoli derivati di identificazione di relazione

Ci sono due casi in cui vincoli di identificazione o di cardinalità definiti su entità implicano vincoli di identificazione su relazioni, che, in questo caso, vengono detti **vincoli derivati di identificazione di relazione**.

1. Il primo caso è quello in cui una entità E partecipa ad una relazione R nel ruolo U con vincolo di cardinalità massima uguale ad 1. Questo implica un vincolo di identificazione nel ruolo U della relazione R , vincolo derivato dalla cardinalità massima.
2. Il secondo caso è quello in cui una entità E ha un identificatore che coincide con una relazione (nel ruolo U), ovvero E partecipa ad una relazione R nel ruolo U e tale ruolo costituisce un identificatore per l'entità E (quindi U ha cardinalità $(1,1)$). Questo implica un vincolo di identificazione derivato sulla relazione R .

Nel seguito analizziamo separatamente i due casi. Si noti comunque che vale anche il contrario: vincoli di identificazione su relazioni possono anche implicare vincoli definiti su entità.



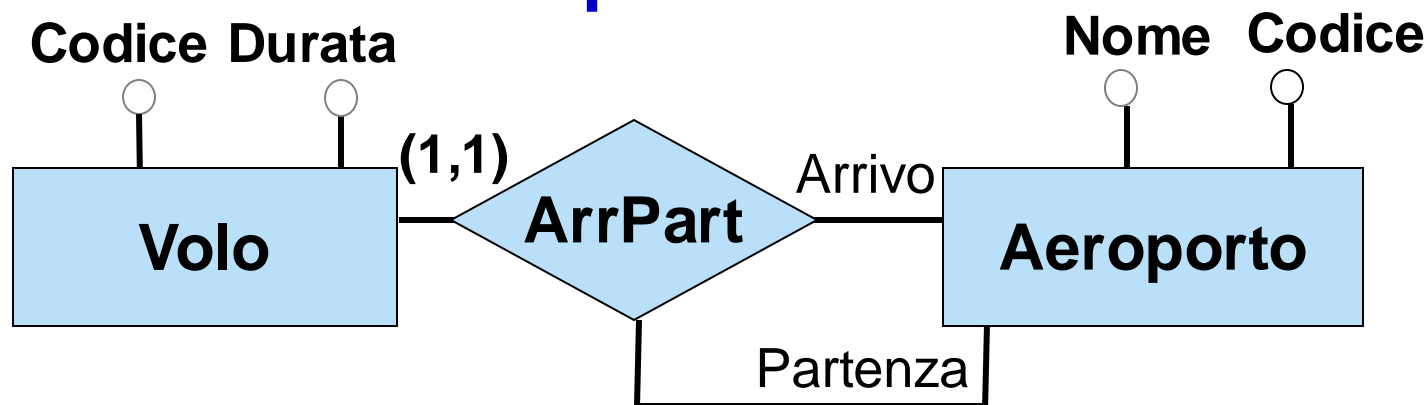
Vincoli derivati di identificazione di relazione

1. Il primo caso è quello in cui una entità E partecipa ad una relazione R nel ruolo U con vincolo di cardinalità massima uguale ad 1.

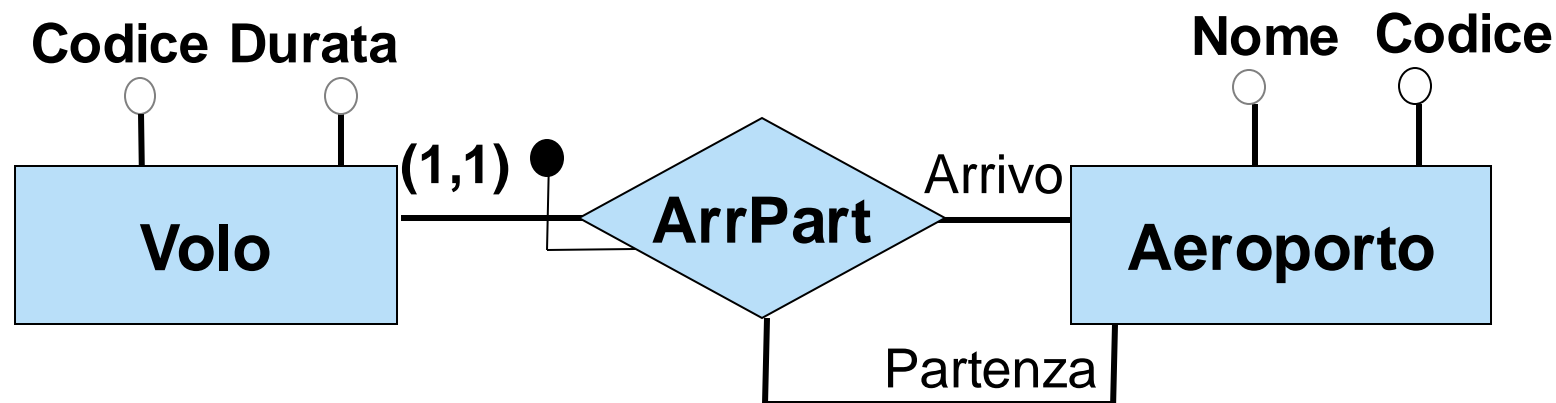
Se l'entità E partecipa ad una relazione R nel ruolo U ed il vincolo di cardinalità massima associata a questo ruolo è 1, allora ogni istanza di E partecipa al massimo una volta ad R nel ruolo U , e quindi non possono esistere due istanze di R che hanno la stessa istanza di E nella componente U . **Ciò implica che esiste un identificatore di relazione derivato: U è infatti un identificatore della relazione R .** Ovviamente, vale anche il contrario: se un identificatore di relazione è formato da un solo ruolo, allora la cardinalità massima associata a questo ruolo è 1.

Quindi nel seguito, **possiamo anche omettere di indicare uno dei due vincoli**: in particolare, se vogliamo privilegiare (come spesso accade) l'indicazione della cardinalità massima sul ruolo U , allora possiamo omettere di indicare esplicitamente l'identificatore di relazione costituito da U , che è derivato dalla cardinalità di U .

Esempio del caso 1



Nello schema qui sopra, ogni istanza di Volo partecipa al massimo ad una istanza di ArrPart, e questo implica che non esistono due istanze di ArrPart che coincidono nella componente corrispondente al ruolo Volo. In altre parole, Volo è un identificatore derivato per la relazione ArrPart. Abbiamo quindi omesso di indicare esplicitamente l'identificatore di relazione. Se volessimo comunque farlo, non sarebbe un errore ed otterremmo:





Vincoli derivati di identificazione di relazione

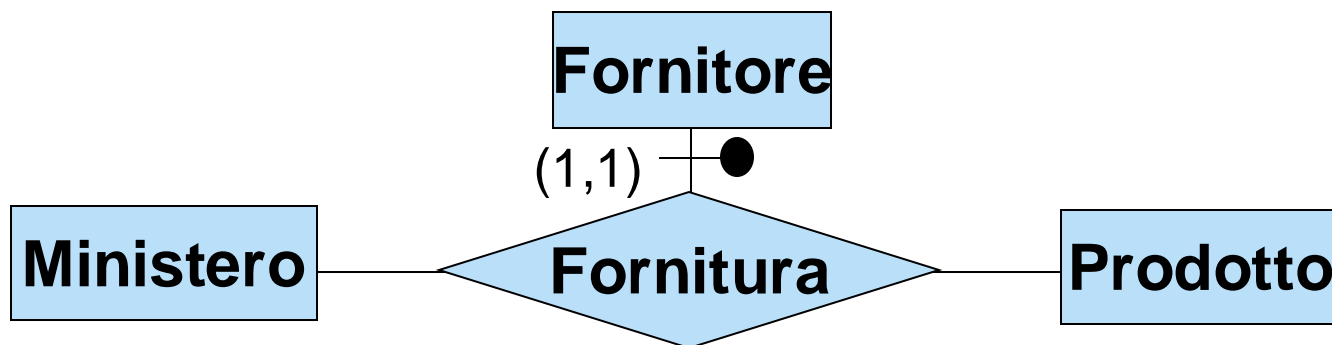
2. Il secondo caso è quello in cui una entità E partecipa ad una relazione R nel ruolo U , e tale ruolo costituisce un identificatore per l'entità E (il che implica che la cardinalità di U in R è $(1,1)$).

Se la relazione R ha ruoli U_1, U_2, \dots, U_h , e se l'entità E partecipa ad R nel ruolo U_1 (evidentemente con cardinalità $(1,1)$), e se tale ruolo costituisce (da solo) un identificatore per E , allora non esistono due istanze di E che partecipano a due istanze di R che hanno le stesse istanze di entità nei ruoli U_2, \dots, U_h . **Ciò implica che non esistono due diverse istanze di R che coincidono nei ruoli U_2, \dots, U_h , ossia che differiscono solo nel ruolo U_1 .**

Questo a sua volta implica che U_2, \dots, U_h formano un identificatore derivato della relazione R . È facile verificare che vale anche la condizione opposta: se U_2, \dots, U_h formano un identificatore della relazione R e se la cardinalità del ruolo U_1 è $(1,1)$, allora la relazione R in quel ruolo U_1 costituisce un identificatore per l'entità E .

Quindi nel seguito, **possiamo anche omettere di indicare uno dei due vincoli di identificazione**: in particolare, se vogliamo privilegiare l'indicazione del vincolo di identificazione di entità costituito da U_1 , allora, per una relazione con h ruoli, possiamo omettere di indicare esplicitamente il suo identificatore derivato costituito da $(h-1)$ ruoli (diciamo i ruoli U_2, \dots, U_h), perché questa condizione è implicitamente rappresentata dal fatto che U_1 è un identificatore dell'entità E , dove E è l'entità corrispondente al ruolo U_1 .

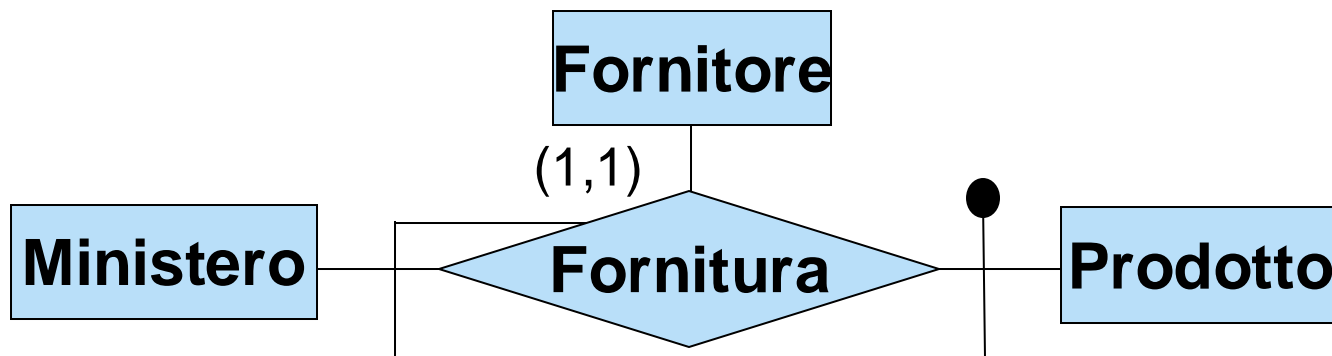
Esempio del caso 2



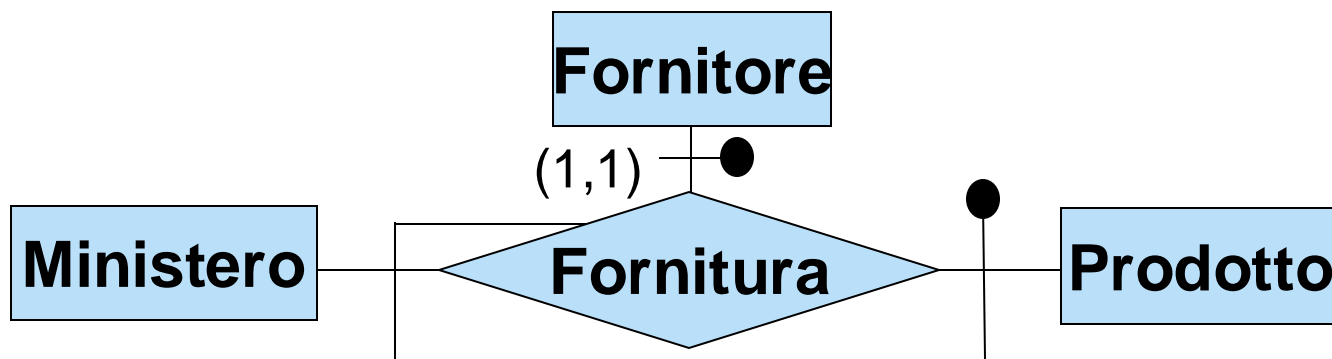
Lo schema qui sopra indica che il ruolo Fornitore è un identificatore dell'entità Fornitore, come già illustrato nell'esempio di pagina 147. Come abbiamo appena visto nella slide precedente, questo implica che la coppia (Ministero,Prodotto) è un identificatore derivato per la relazione Fornitura. Si noti che nel diagramma abbiamo omissso di indicare esplicitamente tale identificatore derivato della relazione.

Esempio del caso 2

Se invece decidessimo di omettere l'indicazione esplicita dell'identificatore di entità, lo schema diventerebbe



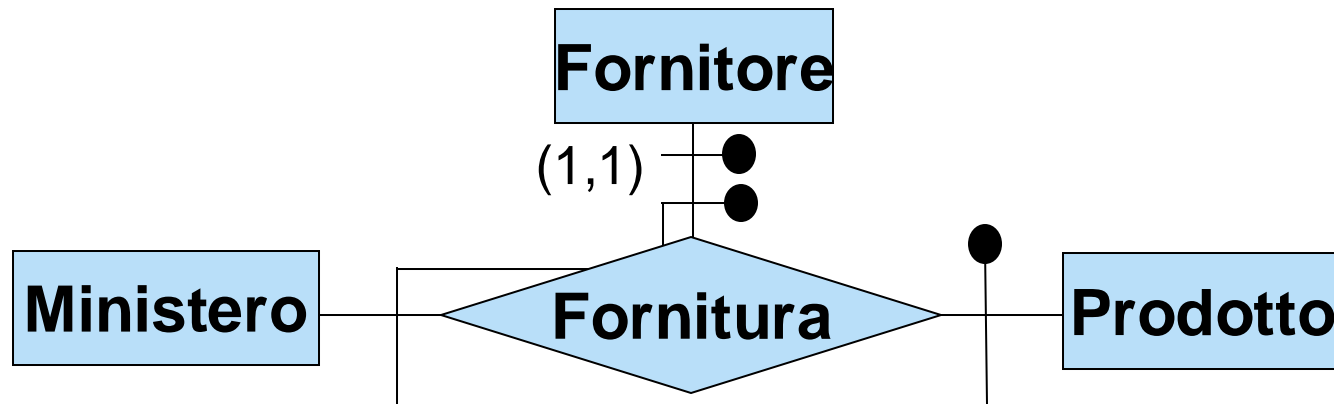
Infine, possiamo anche decidere di lasciare l'indicazione esplicita di entrambi gli identificatori, ed ottenere quindi il seguente schema:





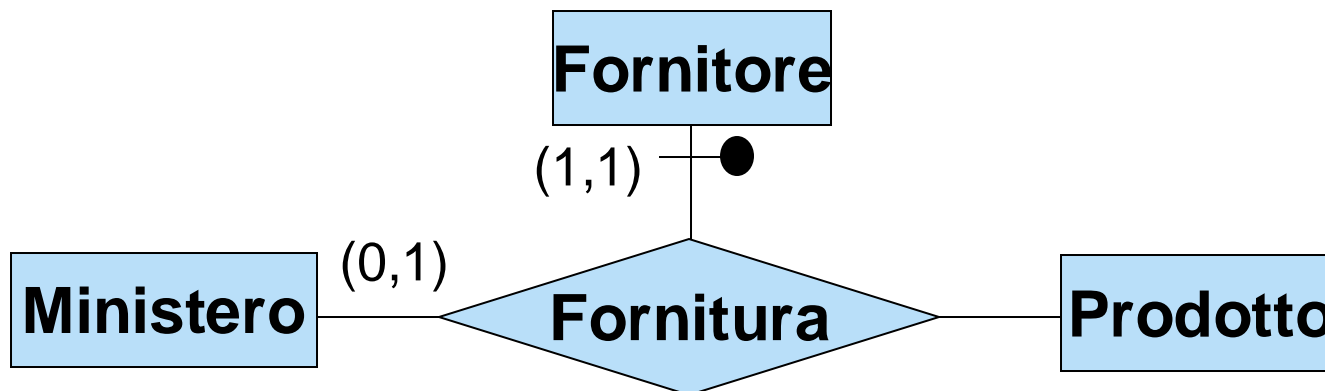
Esempio del caso 2

Se poi, partendo dallo schema di pagina 162, volessimo indicare proprio tutti gli identificatori, (ovviamente non quello implicito, ossia formato da tutti i ruoli della relazione, che non si indica mai e che comunque sarebbe non essenziale), allora lo schema diventerebbe:

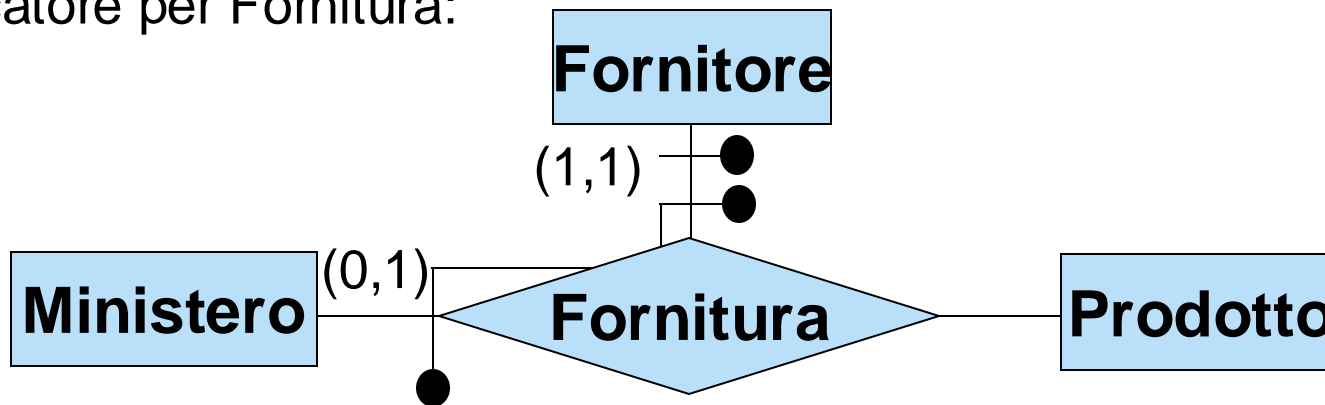


Esempio del caso 2

Supponiamo adesso che lo schema sia il seguente, dove la cardinalità massima di Ministero in Fornitura è 1:



In questo caso, non ha senso indicare l'identificatore di relazione formato da Ministero e Prodotto, perché non è essenziale, visto che la cardinalità massima pari ad 1 su Ministero implica che Ministero stesso è un identificatore per Fornitura:

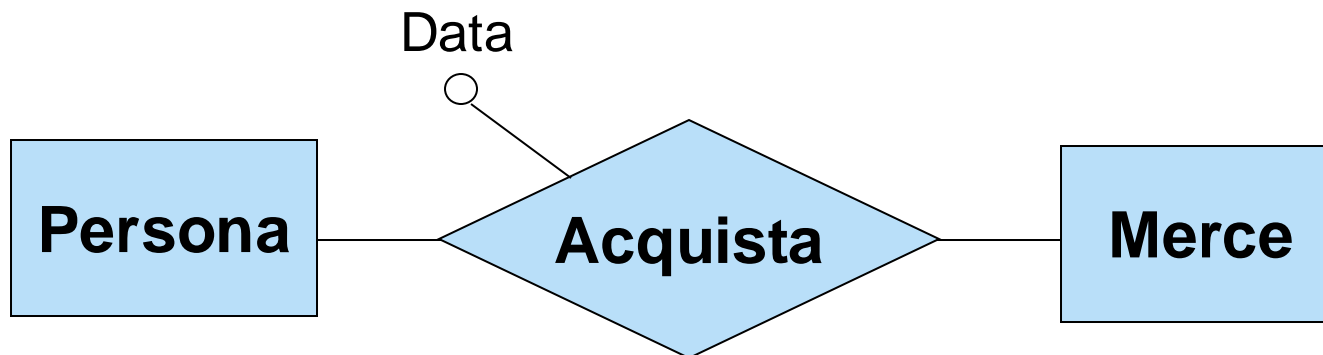




Esercizio 18: vincoli di identificazione di relazione

Supponiamo che nella nostra applicazione una persona non possa effettuare più di un acquisto nella stessa data.

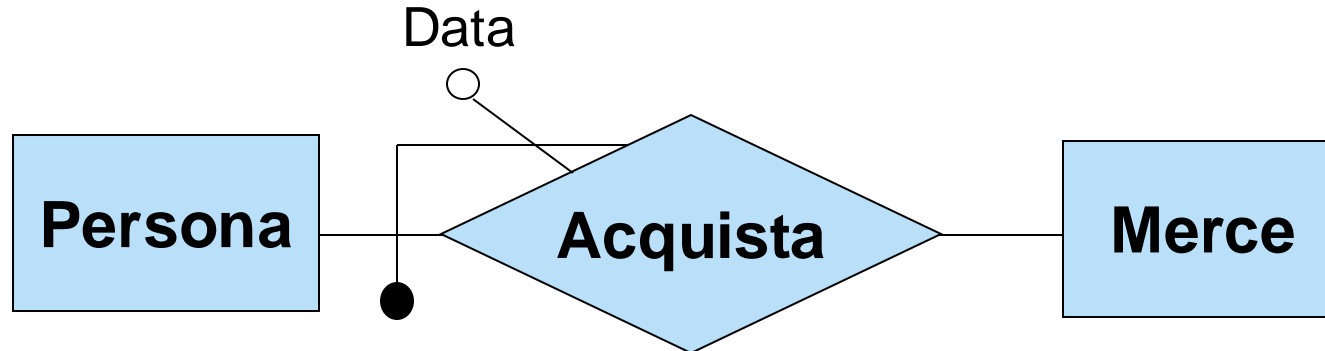
Assumendo di aver definito la relazione Acquista tra Persona e Merce, con attributo Data, come si può esprimere questa condizione?





Esercizio 18: soluzione

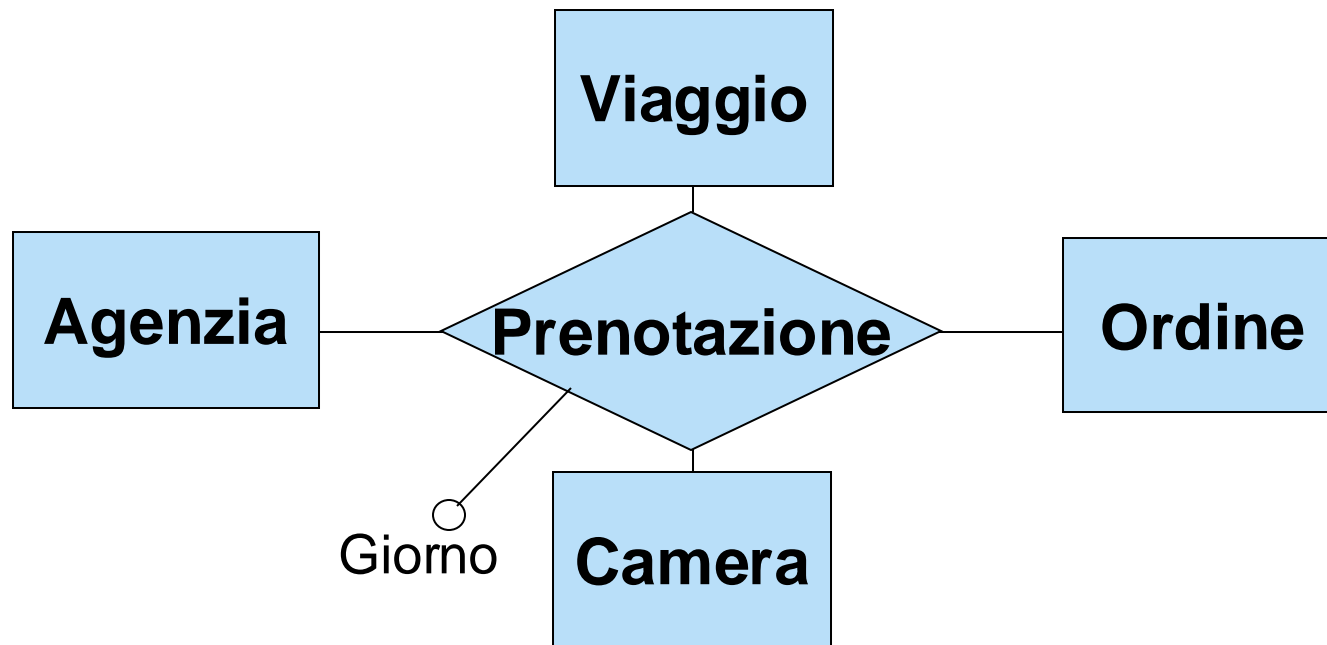
La condizione che una persona non può effettuare più di un acquisto nella stessa data si può esprimere mediante un identificatore di relazione, nel modo seguente:





Esercizio 19: vincoli di identificazione di relazione

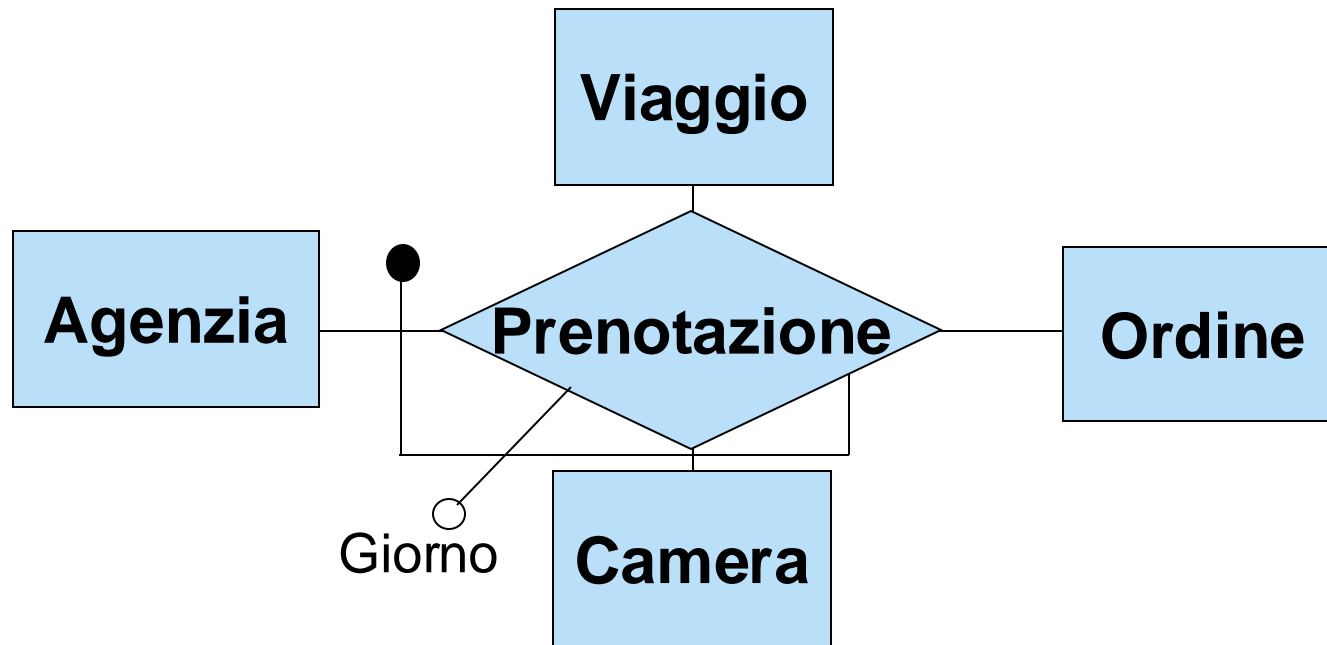
Nello schema qui sotto, la relazione Prenotazione lega l'agenzia e le camere degli alberghi prenotate dalla agenzia per i viaggi organizzati mediante ordini. Di ogni prenotazione interessa il giorno in cui è effettuata. Supponiamo che viga la regola che, nello stesso giorno, una camera d'albergo non può essere prenotata più volte dalla stessa agenzia. Come si può esprimere questa condizione?





Esercizio 19: soluzione

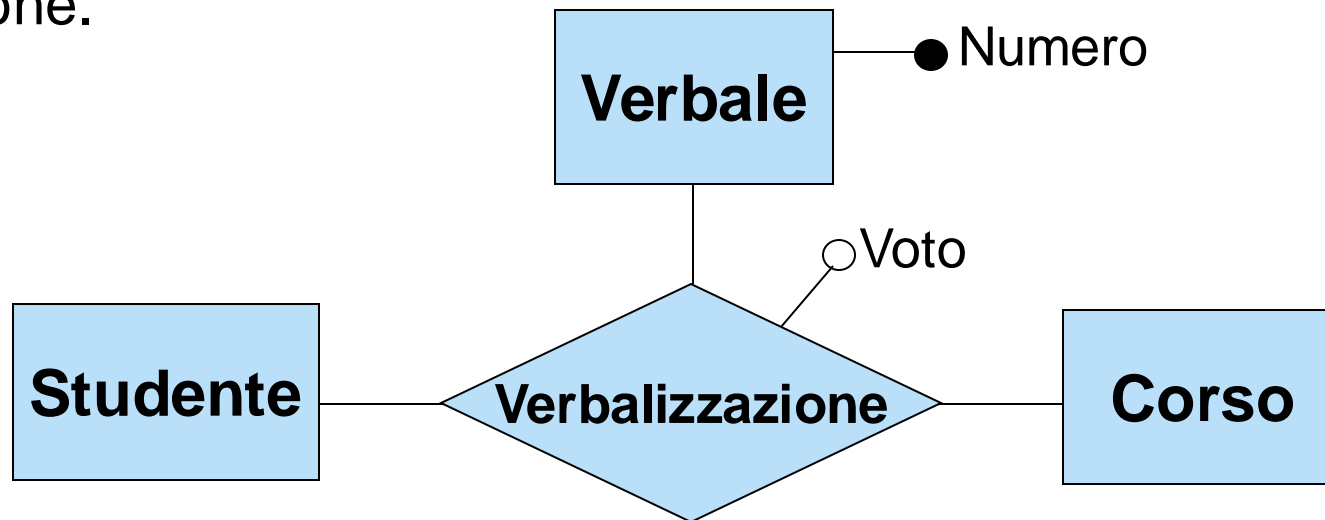
La condizione che nello stesso giorno una camera d'albergo non può essere prenotata più volte alla stessa agenzia, si può esprimere mediante il seguente vincolo di identificazione di relazione:





Esercizio 20: vincoli di identificazione di relazione

La verbalizzazione di un esame coinvolge lo studente che ha superato l'esame (con un certo voto), il corso, ed il verbale d'esame (con il numero identificativo del verbale). Lo schema seguente rappresenta il concetto di verbalizzazione mediante una relazione.



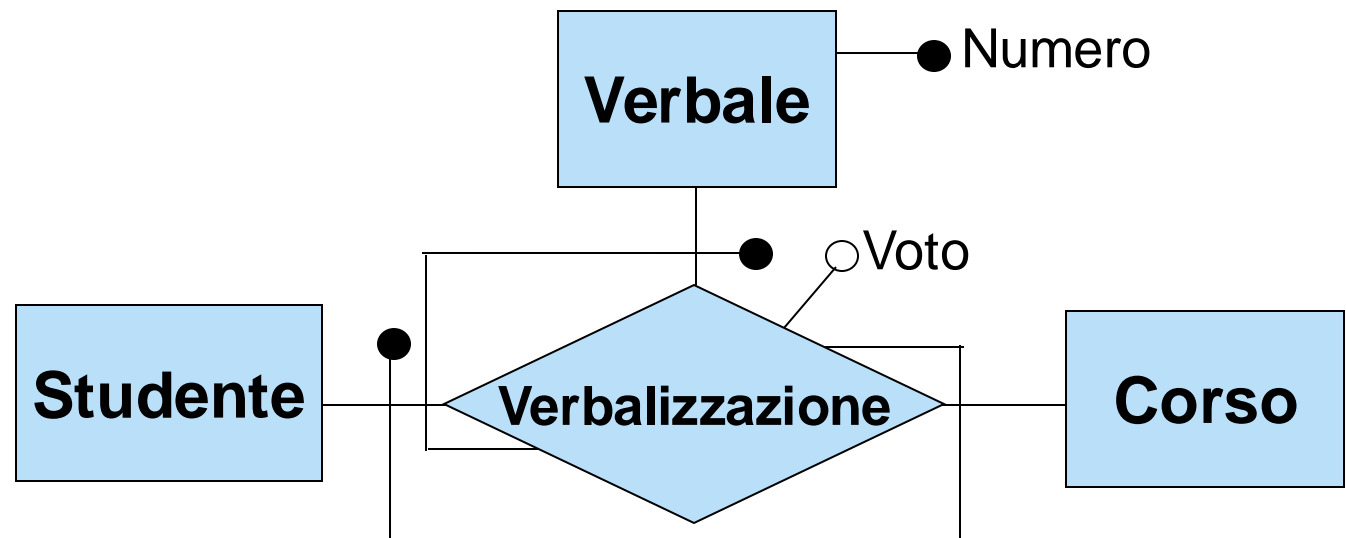
Aggiungere allo schema i vincoli che modellano le seguenti regole:

1. Uno studente non può comparire in più verbalizzazioni con lo stesso verbale
2. Uno studente non può verbalizzare più volte lo stesso corso.



Esercizio 20: soluzione

1. La prima regola implica che non ci possono essere due istanze della relazione Verbalizzazione che coincidono nella coppia (Studente,Verbale): Studente e Verbale formano quindi un identificatore per la relazione Verbalizzazione.
2. Analogamente, la seconda regola impone che non ci possono essere due istanze della relazione Verbalizzazione che coincidono nella coppia (Studente,Corso): Studente e Corso formano quindi un altro identificatore per la relazione Verbalizzazione.





Identificatori derivati di relazioni: riassumiamo

Per determinare gli identificatori derivati (cioè quelli che sono implicati da altri vincoli) di R non già indicati nello schema si devono seguire le seguenti indicazioni:

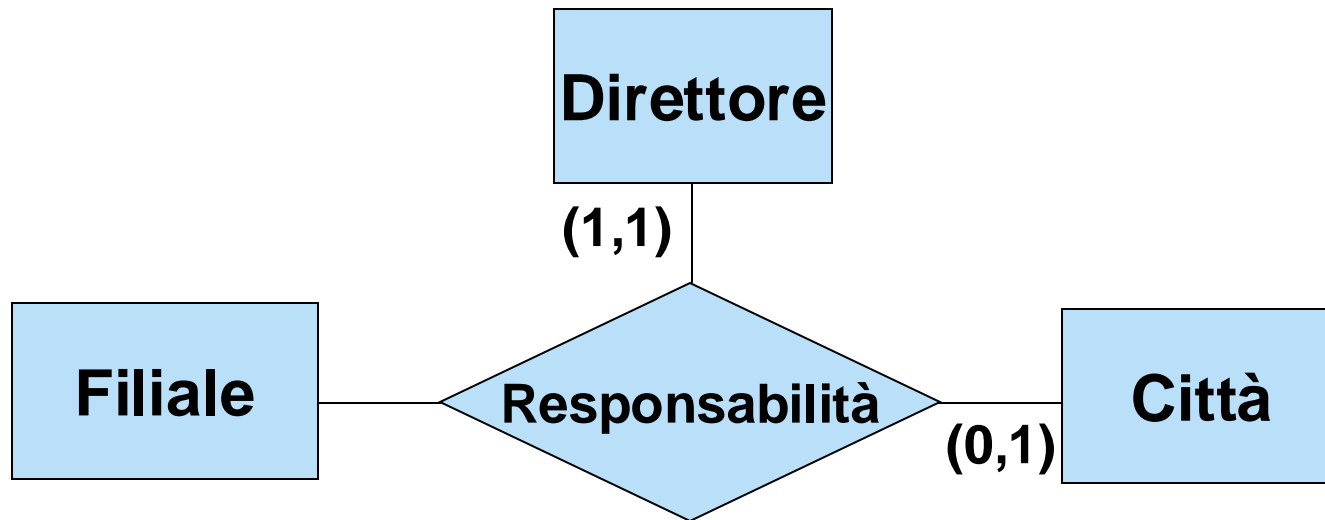
- Se R ha almeno un ruolo con cardinalità massima pari ad uno, o almeno un ruolo che è identificatore di entità, allora
 - ogni ruolo con cardinalità massima pari ad uno è identificatore derivato di R (ed essendo il vincolo derivato, la sua dichiarazione può essere omessa nel diagramma)
 - per ogni ruolo U che è identificatore di entità, la combinazione di ruoli formata da tutti i ruoli di R escluso U è identificatore derivato di R (ed essendo il vincolo derivato, la sua dichiarazione può essere omessa nel diagramma)

Esiste poi l'identificatore implicito di R , formato da tutti i ruoli di R . Come abbiamo detto, poiché questo vincolo discende direttamente dalla semantica della nozione di relazione, per prassi non si indica.



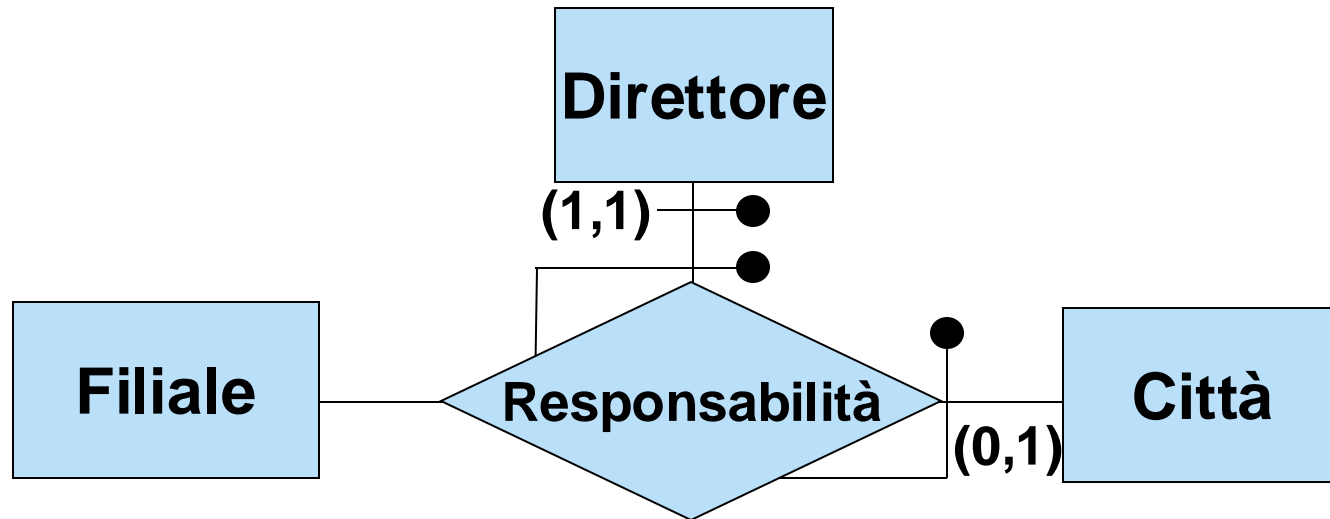
Esercizio 21: vincoli di identificazione di relazione

Lo schema qui sotto rappresenta una situazione in cui un direttore è responsabile delle vendite effettuate da una filiale di banca per una certa città. Ogni direttore ha esattamente una responsabilità, ed ogni città è coinvolta al massimo in una responsabilità. Analizzare lo schema e rendere espliciti tutti i vincoli di identificazione essenziali che derivano da altri vincoli.





Esercizio 21: soluzione



- Che il ruolo Direttore sia un identificatore della relazione Responsabilità deriva dal fatto che la cardinalità massima di Direttore nella relazione è 1.
- Che il ruolo Città sia un identificatore della relazione Responsabilità deriva dal fatto che la cardinalità massima di Città nella relazione è 1.
- Che la relazione Responsabilità, nel ruolo Direttore, sia un identificatore dell'entità Direttore deriva da due fatti:
 - la cardinalità massima del ruolo Città nella relazione è 1,
 - l'entità Direttore partecipa alla relazione, nel ruolo Direttore, con cardinalità (1,1)



Vincoli derivati di identificazione di entità

Abbiamo appena discusso i casi in cui i vincoli di identificazione di relazione vengono derivati da altri vincoli (di cardinalità ed identificazione di entità).

Ci chiediamo ora se vale anche il contrario: **è possibile derivare vincoli di identificazione di entità sulla base di vincoli di identificazione di relazione?**

La risposta è **positiva** e si ottiene generalizzando un'osservazione già riportata nella slide 161:

Sia R una relazione con ruoli U_1, U_2, \dots, U_h , e sia E l'entità che partecipa ad R nel ruolo U_1 con cardinalità $(1,1)$. **Se $\{U_{i_1}, \dots, U_{i_k}\}$ è un qualunque sottoinsieme di $\{U_1, U_2, \dots, U_h\}$ che non contiene U_1 ed è un identificatore di R , allora U_1 è un identificatore per E .**

Per dimostrare questa proprietà supponiamo che esistano due istanze e, f di E che hanno la stessa partecipazione ad R nel ruolo U_1 . È immediato verificare che questo significa che esistono due istanze di R con questa forma: $\langle U_1:e, U_2:g_2, \dots, U_h:g_h \rangle$ e $\langle U_1:f, U_2:g_2, \dots, U_h:g_h \rangle$. Ma questo implica che tali due istanze di R coincidono nei ruoli U_{i_1}, \dots, U_{i_k} , contraddicendo il fatto che U_{i_1}, \dots, U_{i_k} sia un identificatore di R .



Vincoli non esprimibili nel diagramma ER

- Gli schemi ER permettono di cogliere la maggior parte delle interrelazioni tra i dati del dominio d'interesse
- Tuttavia alcune interrelazioni non possono essere colte direttamente da uno schema ER
- Tali interrelazioni vanno in ogni caso tenute presenti attraverso delle asserzioni aggiuntive dette **vincoli esterni al diagramma**, o semplicemente **vincoli esterni**

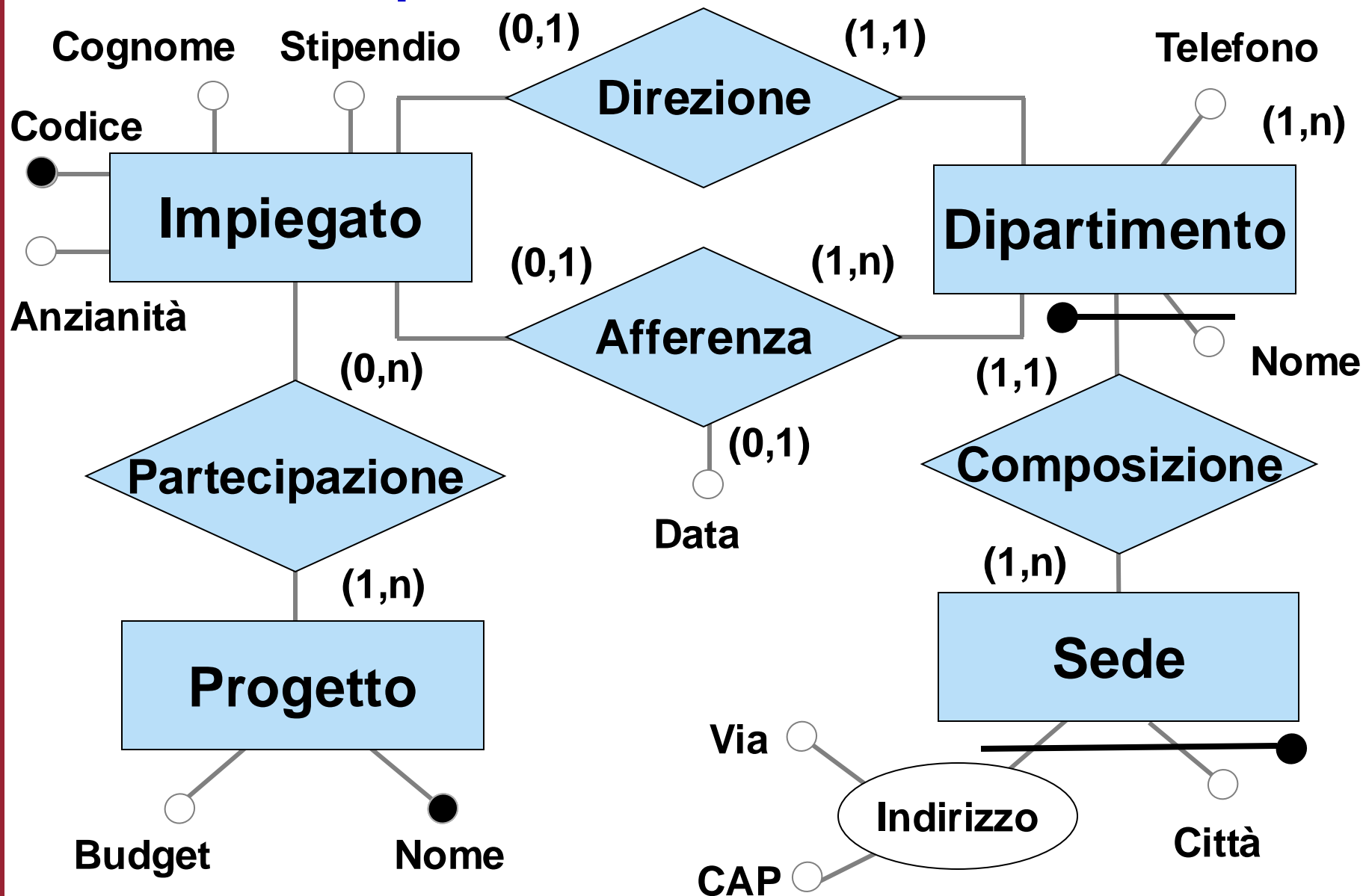


Vincoli non esprimibili nel diagramma ER

- Come rappresentiamo tali vincoli?
- Attraverso formalismi opportuni (es, in logica matematica)
- Attraverso delle asserzioni in linguaggio naturale (che devono essere il più possibile precise e non ambigue)



Esempio di schema concettuale ...



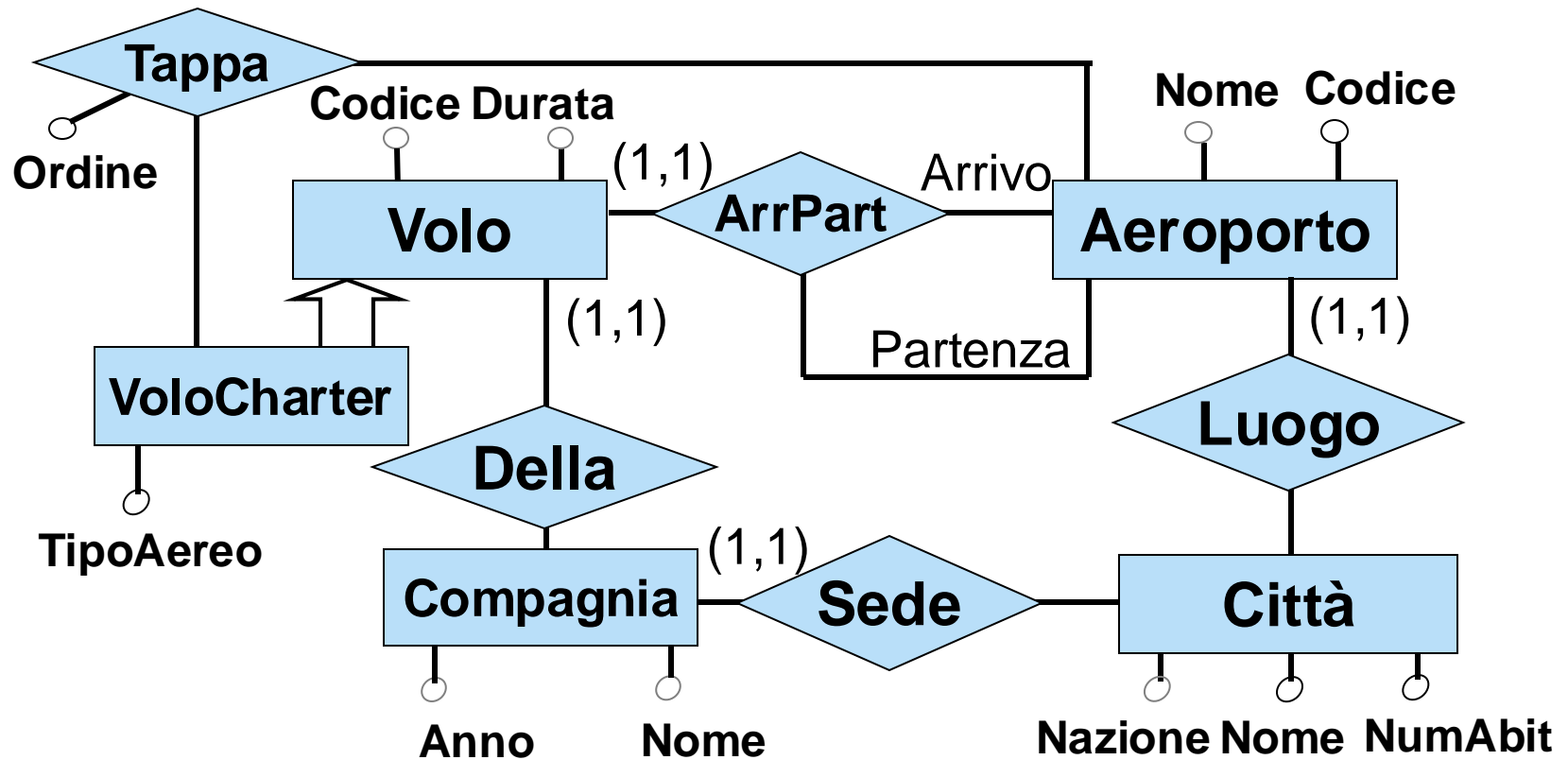


... con vincoli esterni

Vincoli di integrità esterni
(1) Il direttore di un dipartimento deve afferire a tale dipartimento da almeno 5 anni
(2) Un impiegato non deve avere uno stipendio maggiore del direttore del dipartimento al quale afferisce
(3) Un dipartimento con sede a Roma deve essere diretto da un impiegato con più di dieci anni di anzianità
(4) Un impiegato non può partecipare ad un numero di progetti maggiore di due volte il numero di dipartimenti ai quali afferisce



Vincolo esterno sullo schema dell'esercizio 12



Vincolo esterno: Per ogni v in **VoloCharter**, se $(v, a_1), \dots, (v, a_n)$ sono tutte le coppie in **Tappa** alle quali v partecipa, e se o_1, \dots, o_n , sono i valori assegnati a tali coppie dall'attributo **Ordine**, allora per ogni $i=1, \dots, n$, esiste un o_j tale che $o_j = i$.

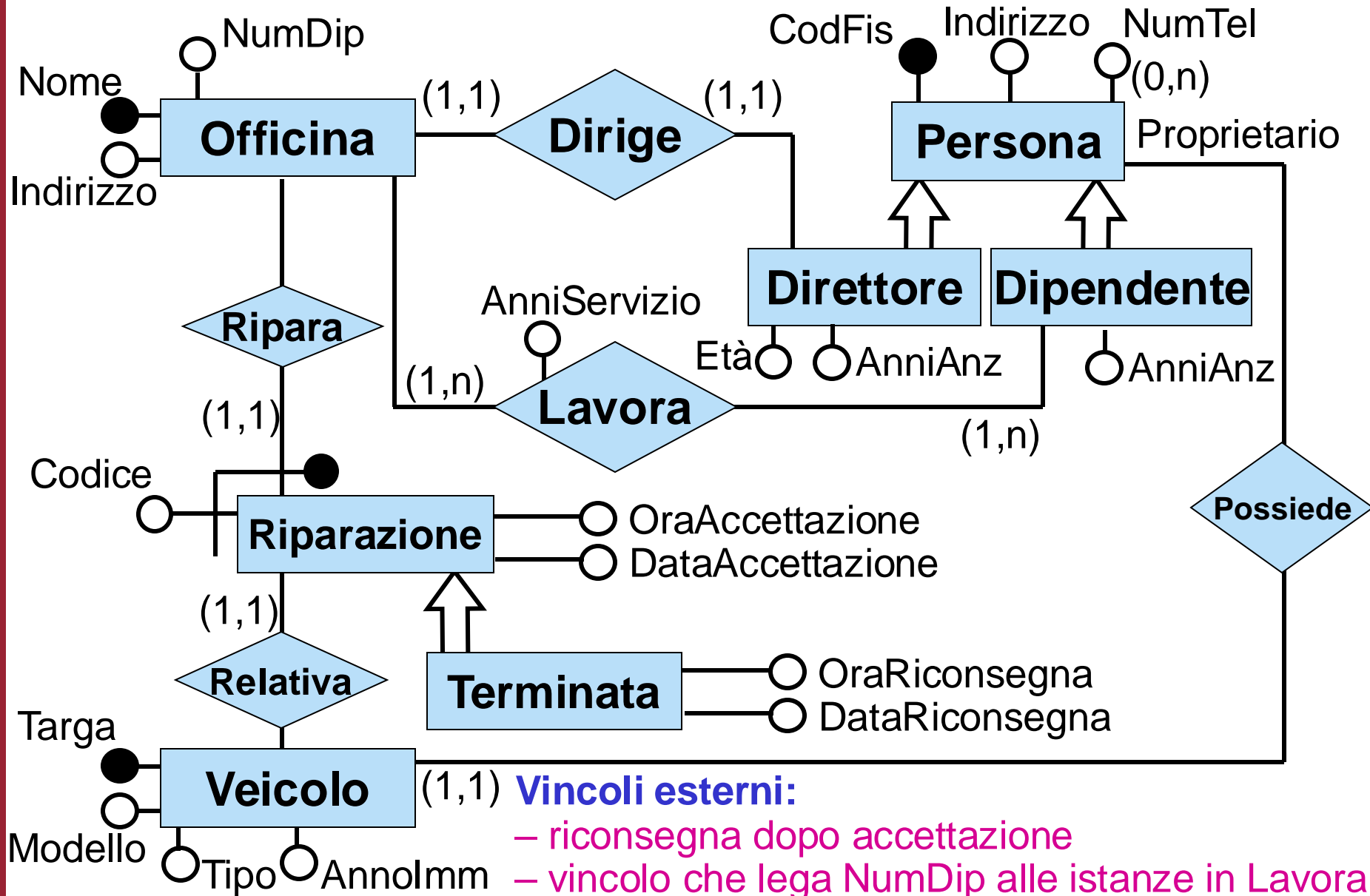


Esercizio 22: progettazione concettuale

Descrivere lo schema concettuale corrispondente ad un'applicazione riguardante un insieme di officine, facendo riferimento alle seguenti specifiche.

Delle officine interessano: nome (che identifica le officine), indirizzo, numero di dipendenti (maggiore di 0), dipendenti (con l'informazione su quanti anni di servizio), e direttore. Si noti che ogni officina ha uno ed un solo direttore, ed un direttore dirige una ed una sola officina. Dei dipendenti e dei direttori interessano: codice fiscale, indirizzo, numeri di telefono, e anni di anzianità. Dei direttori interessa anche l'età. Si noti che un direttore non è necessariamente un dipendente di officina (ma può esserlo). Ogni riparazione è effettuata da una ed una sola officina, e riguarda uno ed un solo veicolo. Di ogni riparazione interessano: codice (univoco nell'ambito dell'officina), ora e data di accettazione del veicolo, e, nel caso di riparazione terminata, ora e data di riconsegna del veicolo. Dei veicoli interessano: modello, tipo, targa, anno di immatricolazione, e proprietario. Ogni veicolo ha uno ed un solo proprietario. Dei proprietari di veicoli interessano: codice fiscale, indirizzo, e numeri di telefono.

Esercizio 22: soluzione





Documentazione associata agli schemi ER

Oltre al diagramma ER, lo schema concettuale è anche costituito dal cosiddetto **dizionario dei dati** che riassume alcuni aspetti dello schema che servono a documentarne gli elementi

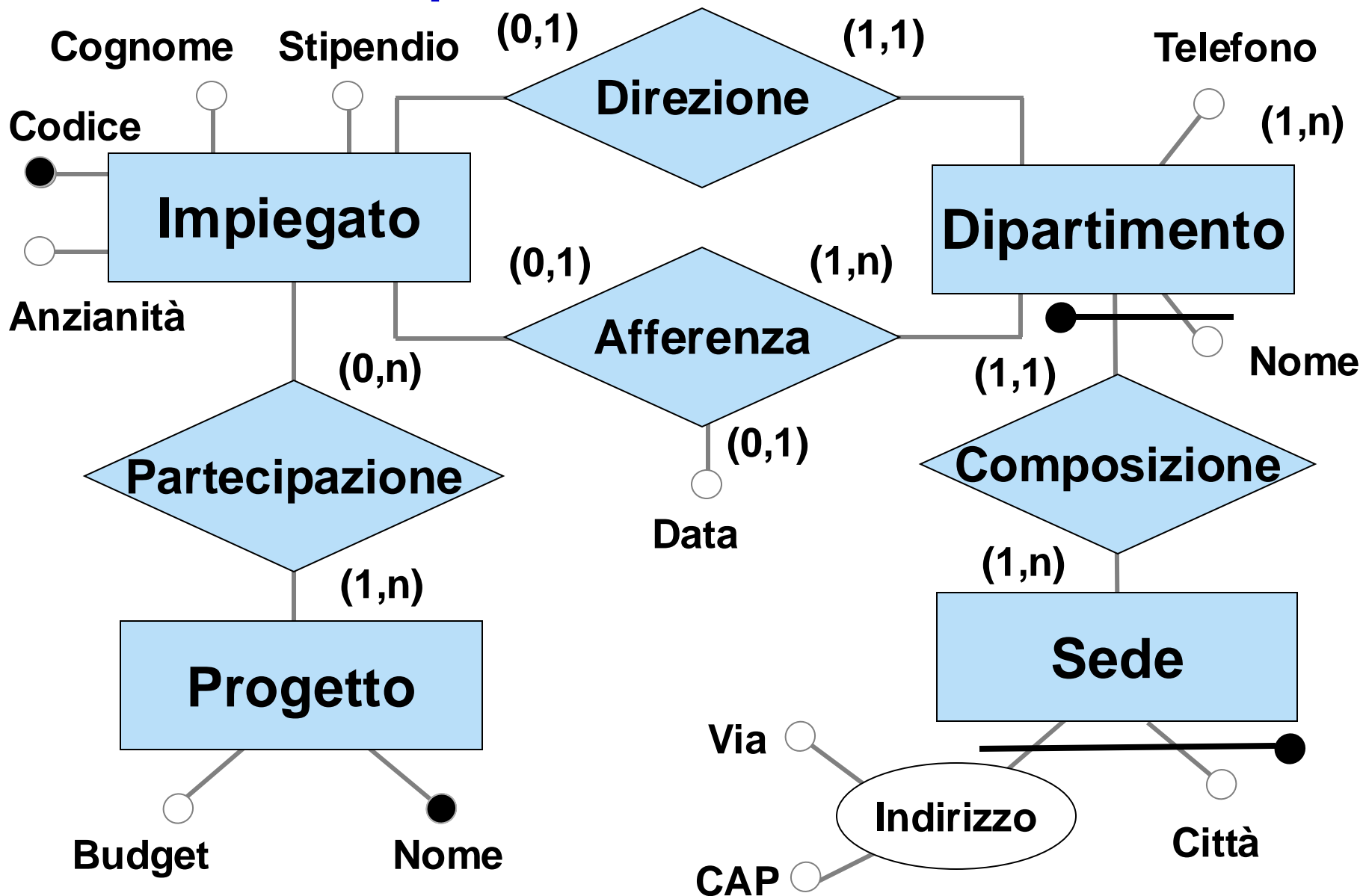
Il dizionario dei dati è formato dalle tabelle di

- Entità
- Relazioni
- Attributi
- Vincoli esterni

La tabella davvero cruciale è quella dei vincoli esterni, che non sono espressi nel diagramma e che, se non riportati nel dizionario dei dati, non farebbero parte dello schema concettuale.



Esempio di schema concettuale





Dizionario dei dati: entità

Entità	Descrizione	Attributi	Identificatori
Impiegato	Dipendente dell'azienda	Codice Cognome Stipendio Anzianità	{ Codice }
Progetto	Progetti aziendali	Nome Budget	{ Nome }
Dipartimento	Struttura aziendale	Nome Telefono	{ Nome, Sede }
Sede	Sede dell'azienda	Città Indirizzo (Via, CAP)	{ Città, Indirizzo }



Dizionario dei dati: relazioni

Relazione	Descrizione	Componenti	Attributi	Identificatori
Direzione	Direzione di un dipartimento	Impiegato, Dipartimento		
Afferenza	Afferenza ad un dipartimento	Impiegato, Dipartimento	Data	
Partecipazione	Partecipazione ad un progetto	Impiegato, Progetto		
Composizione	Composizione dell'Azienda	Dipartimento, Sede		



Dizionario dei dati: attributi

Attributo	Entità/Relazione	Dominio	Descrizione
Codice	Impiegato	Intero	Codice identificativo di impiegati
Cognome	Impiegato	Stringa	Cognome di impiegato
Stipendio	Impiegato	Reale	Stipendio di impiegato
Nome	Progetto	Stringa	Nome del progetto
...



Dizionario dei dati: vincoli esterni

Vincoli di integrità esterni

- | |
|--|
| (1) Il direttore di un dipartimento deve afferire a tale dipartimento |
| (2) Un impiegato non deve avere uno stipendio maggiore del direttore del dipartimento al quale afferisce |
| (3) Un dipartimento con sede a Roma deve essere diretto da un impiegato con più di dieci anni di anzianità |
| (4) Un impiegato che non afferisce ad alcun dipartimento non deve partecipare ad alcun progetto |



5. La progettazione concettuale

5.3 metodologia per la progettazione concettuale

1. introduzione alla progettazione di basi di dati
2. modello Entità-Relazione
3. **metodologia per la progettazione concettuale**
4. esempio



Attività nella progettazione concettuale

La progettazione concettuale è preceduta dalla fase di raccolta (o acquisizione) dei requisiti

A partire dai requisiti raccolti, la progettazione concettuale comprende le seguenti **attività**:

1. strutturazione e organizzazione dei requisiti raccolti
2. costruzione del glossario (opzionale)
3. costruzione dello schema concettuale (diagramma e dizionario dei dati)
4. controllo di qualità

Queste attività non sono necessariamente eseguite in serie, ma sono tipicamente interconnesse



Fase 1: Strutturazione e organizzazione dei requisiti

I requisiti raccolti devono venire strutturati e organizzati per gli scopi dell'analisi.

Alcune regole generali per l'organizzazione di termini e concetti:

- riorganizzare le frasi per concetti
- costruire un glossario dei termini
- individuare omonimi e sinonimi e unificare i termini
- rendere esplicito il riferimento fra termini



Esempio di organizzazione dei requisiti: società di formazione

Si vuole realizzare una base di dati per una società che eroga corsi, di cui vogliamo rappresentare i dati dei partecipanti ai corsi e dei docenti. Per gli studenti (circa 5000), identificati da un codice, si vuole memorizzare il codice fiscale, il cognome, l'età, il sesso, il luogo di nascita, il nome dei loro attuali datori di lavoro, i posti dove hanno lavorato in precedenza insieme al periodo, l'indirizzo e il numero di telefono, i corsi che hanno frequentato (i corsi sono in tutto circa 200) e il giudizio finale.

Rappresentiamo anche i seminari che stanno attualmente frequentando e, per ogni giorno, i luoghi e le ore dove sono tenute le lezioni. I corsi hanno un codice, un titolo e possono avere varie edizioni con date di inizio e fine e numero di partecipanti. Se gli studenti sono liberi professionisti, vogliamo conoscere l'area di interesse e, se lo possiedono, il titolo. Per quelli che lavorano alle dipendenze di altri, vogliamo conoscere invece il loro livello e la posizione ricoperta.

Per gli insegnanti (circa 300), rappresentiamo il cognome, l'età, il posto dove sono nati, il nome del corso che insegnano, quelli che hanno insegnato nel passato e quelli che possono insegnare. Rappresentiamo anche tutti i loro recapiti telefonici. I docenti possono essere dipendenti interni della società o collaboratori esterni.



Strutturazione dei requisiti in gruppi di frasi omogenee (1)

Frasi di carattere generale:

Si vuole realizzare una base di dati per una società che eroga corsi, di cui vogliamo rappresentare i dati dei partecipanti ai corsi e dei docenti.

Frasi relative ai partecipanti:

Per i partecipanti (circa 5000), identificati da un codice, rappresentiamo il codice fiscale, il cognome, l'età, il sesso, la città di nascita, i nomi dei loro attuali datori di lavoro e di quelli precedenti (insieme alle date di inizio e fine rapporto), le edizioni dei corsi che stanno attualmente frequentando e quelli che hanno frequentato nel passato, con la relativa votazione finale in decimi.



Strutturazione dei requisiti in gruppi di frasi omogenee (2)

Frase relative ai datori di lavoro:

Relativamente ai datori di lavoro presenti e passati dei partecipanti, rappresentiamo il nome, l'indirizzo e il numero di telefono.

Frase relative ai corsi:

Per i corsi (circa 200), rappresentiamo il titolo e il codice, le varie edizioni con date di inizio e fine e, per ogni edizione, rappresentiamo il numero di partecipanti e il giorno della settimana, le aule e le ore dove sono tenute le lezioni.



Strutturazione dei requisiti in gruppi di frasi omogenee (3)

Frase relative a tipi specifici di partecipanti:

Per i partecipanti che sono liberi professionisti, rappresentiamo l'area di interesse e, se lo possiedono, il titolo professionale. Per i partecipanti che sono dipendenti, rappresentiamo invece il loro livello e la posizione ricoperta.

Frase relative ai docenti:

Per i docenti (circa 300), rappresentiamo il cognome, l'età, la città di nascita, tutti i numeri di telefono, il titolo del corso che insegnano, di quelli che hanno insegnato nel passato e di quelli che possono insegnare. I docenti possono essere dipendenti interni della società di formazione o collaboratori esterni.



Fase 2: Costruzione del glossario (opzionale)

Termine	Descrizione	Sinonimi	Collegamenti
Partecipante	Persona che partecipa ai corsi	Studente	Corso, Società
Docente	Docente dei corsi. Può essere esterno	Insegnante	Corso
Corso	Corso organizzato dalla società. Può avere più edizioni.	Seminario	Docente
Società	Ente presso cui i partecipanti lavorano o hanno lavorato	Posti	Partecipante



Fase 3: Costruzione dello schema concettuale

- Si parte dalla specifica dei requisiti raccolti, strutturati e organizzati, e dall'eventuale glossario
- Si fa uso del **modello Entità-Relazione** e si costruisce il diagramma e il dizionario dei dati
- Si utilizzano opportune **strategie** per sviluppare lo schema concettuale
- Nello sviluppare il progetto si tiene conto dei **fattori di qualità** che uno schema concettuale deve possedere



Fase 3: Costruzione dello schema concettuale – principi di utilizzo del modello ER

1. Scelta dei **costrutti** più adatti a rappresentare i concetti della specifica: entità, relazioni, ruoli o attributi?
2. Scelta dei **pattern di modellazione**, ovvero di quelle modalità di modellazione dei fenomeni che ricorrono di frequente nell'attività di progettazione.
3. Scelta delle operazioni di **trasformazione dello schema**; infatti, è spesso necessario durante la progettazione concettuale effettuare trasformazioni che preservino l'equivalenza.



3.1 Scelta dei costrutti

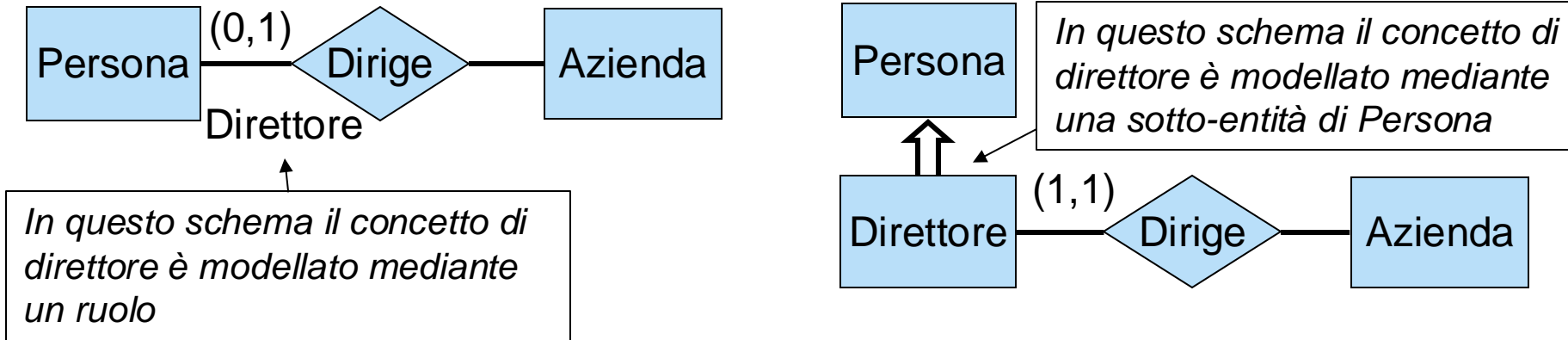
Quale costrutto ER va utilizzato per rappresentare un concetto presente nelle specifiche?

Bisogna basarsi sulle significato dei costrutti del modello ER. Abbiamo già discusso questo aspetto. Qui riassumiamo i principi fondamentali già discussi

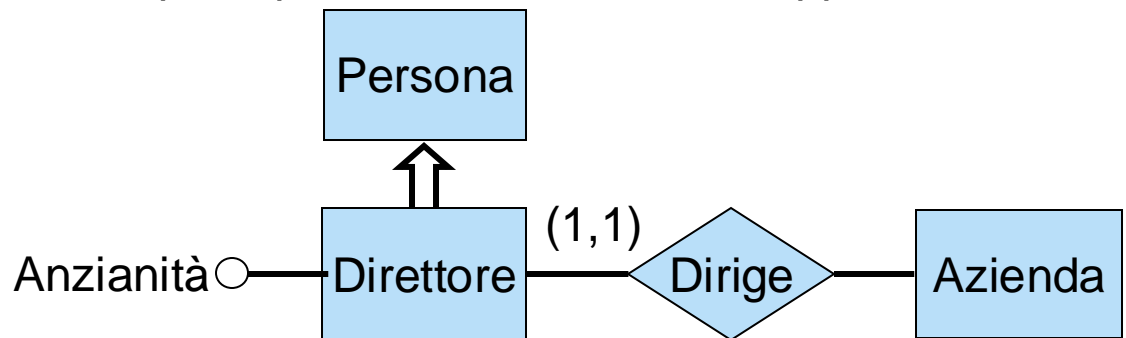
- se il concetto ha **proprietà significative** e descrive oggetti con **esistenza autonoma**:
 - **entità**
- se il concetto è una **proprietà locale di un altro** e non ha proprietà a sua volta:
 - **attributo**
- se il concetto **correla due o più** concetti:
 - **relazione**
- se il concetto induce la presenza di **casi particolari** rispetto ad un altro concetto :
 - **is-a** o **generalizzazione**

3.1 Scelta dei costrutti: ruolo vs entità

Tra le possibili scelte da effettuare c'è anche quella tra ruolo ed entità. Consideriamo i seguenti due schemi:



I due schemi sono in effetti equivalenti, nel senso che gli oggetti che soddisfano il concetto di direttore nel primo schema sono le istanze di Persona che partecipano alla relazione Dirige nel ruolo Direttore, mentre nel secondo schema sono le istanze dell'entità Direttore. Ovviamente, se dovessimo attribuire proprietà aggiuntive a Direttore (ad esempio, Anzianità), allora dovremmo optare per la seconda scelta e rappresentare il concetto mediante entità:





3.2 Scelta dei pattern di modellazione

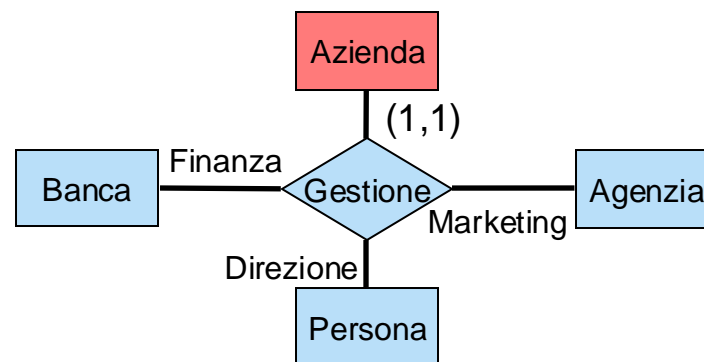
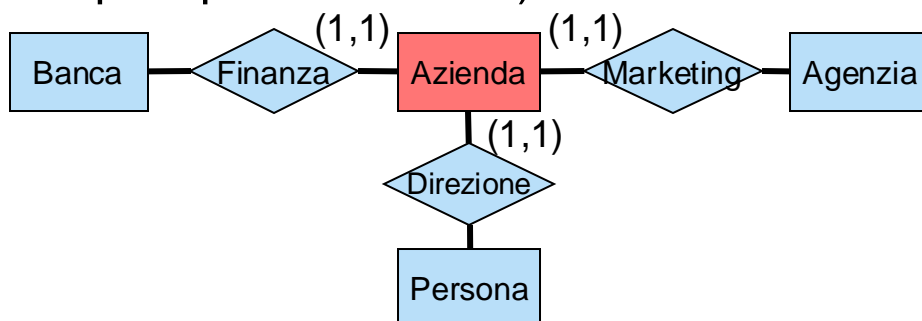
In molte situazioni durante la progettazione concettuale ci troviamo di fronte a scelte tra diversi pattern da utilizzare. Un pattern è una modalità di modellazione di fenomeni che ricorrono di frequente nell'attività di progettazione. Ci concentriamo qui su due pattern importanti.

3.2.1 Scelta tra relazioni binarie e relazioni n-arie. Quando un insieme di N proprietà che attribuiamo ad una entità condividono la stessa molteplicità $(1,1)$ dobbiamo decidere se modellarle come singole relazioni binari indipendenti l'una dall'altra, oppure accorparle e descriverle mediante un'unica relazione di grado $N+1$.

3.2.2. Rappresentazione di aspetti dinamici. Se le istanze di una entità o di una relazione hanno proprietà che cambiano nel tempo ed interessa il loro valore in diversi stati, allora dobbiamo rappresentare l'entità o la relazione in modo cosiddetto "storicizzato". Esistono, a questo scopo, specifici pattern che aiutano a modellare correttamente proprietà di questo tipo, proprietà che sono dette "dinamiche".

3.2.1 Scelta dei pattern di modellazione: relazioni binarie vs relazioni n-arie

Se una entità è coinvolta in N diverse relazioni binarie alle quali partecipa con cardinalità $(1,1)$, è possibile “accorpare” queste relazioni in un’unica relazione n -aria, il cui grado è $N+1$, sempre che tale relazione sia concettualmente significativa. Mostriamo la situazione per il caso di $N=3$ (ma il ragionamento vale per qualsiasi $N > 1$).

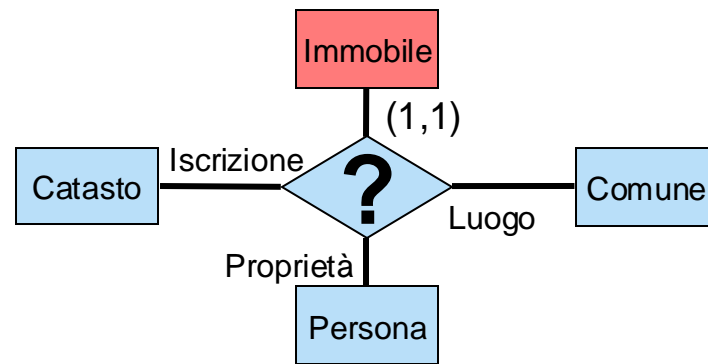
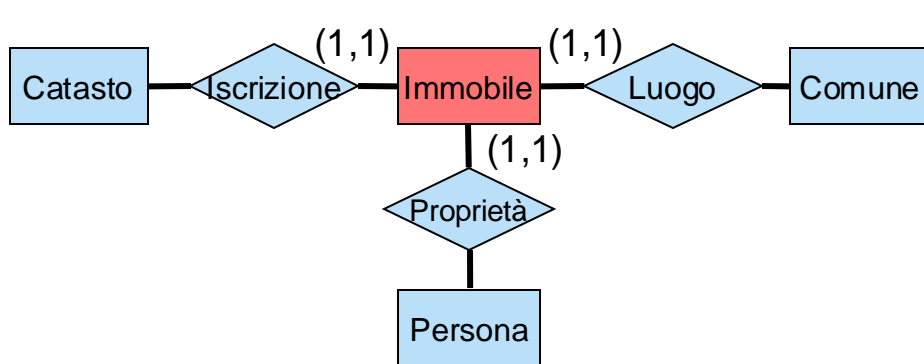


Nello schema a sinistra le proprietà di Azienda sono state modellate mediante tre relazioni indipendenti, tutte di cardinalità $(1,1)$. Esse specificano la banca che gestisce la finanza dell'azienda, la persona che gestisce la direzione dell'azienda e l'agenzia che ne gestisce il marketing. Nello schema a destra le tre proprietà sono accorpate in un'unica relazione, che modella la gestione dell'azienda. Si noti come le tre relazioni originarie sono ora modellate come ruoli nello schema. Si può dimostrare che i due schemi sono equivalenti dal punto di vista semantico, nel senso che per ogni istanza dello schema a sinistra esiste una corrispondente (isomorfa) istanza dello schema a destra, e viceversa.

3.2.1 Scelta dei pattern di modellazione: relazioni binarie vs relazioni n-arie

L'esempio precedente dimostra che N relazioni binarie alle quali una entità E partecipa con cardinalità (1,1) si possono accorpare in un'unica relazione di grado N+1 alla quale E partecipa con cardinalità (1,1) in modo che lo schema risultante sia equivalente semanticamente a quello originario.

Questo però non significa che i due schemi siano equivalenti dal punto di vista pragmatico. Ovvero, questo non significa che l'accorpamento sia sempre ragionevole dal punto di vista della modellazione. Lo schema risultante, infatti, potrebbe essere scorretto dal punto di vista dell'adeguatezza dello schema stesso a rappresentare in modo concettualmente ragionevole la realtà da modellare. In particolare, questo succede se accorpriamo relazioni concettualmente disomogenee tra loro, come nell'esempio seguente.

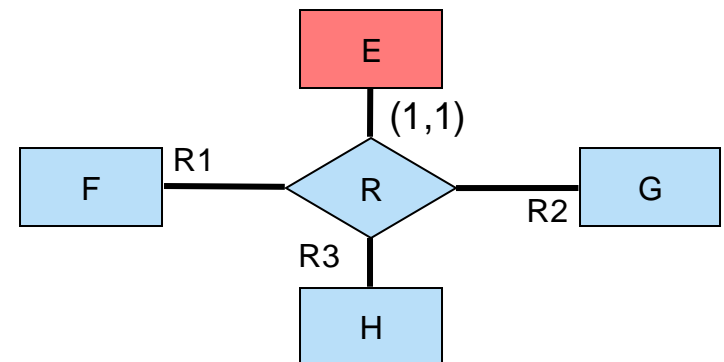
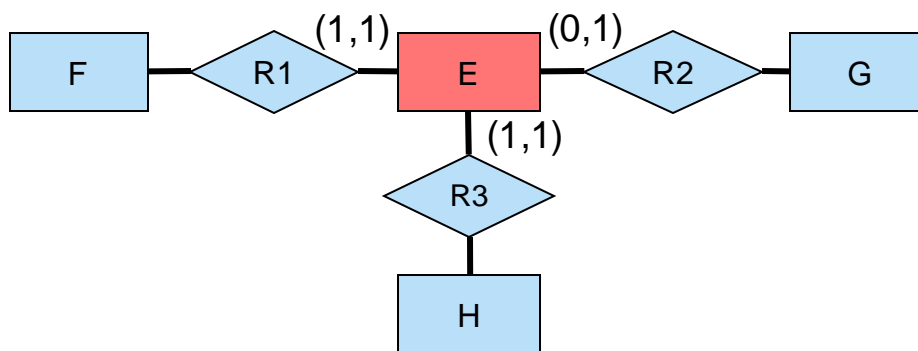


In questo caso, sotto il profilo pragmatico, è scorretto accorpare le tre relazioni in una relazione di grado 4 che, dal punto di vista concettuale, non avrebbe senso.



Esercizio 23: relazioni binarie vs relazioni n-arie

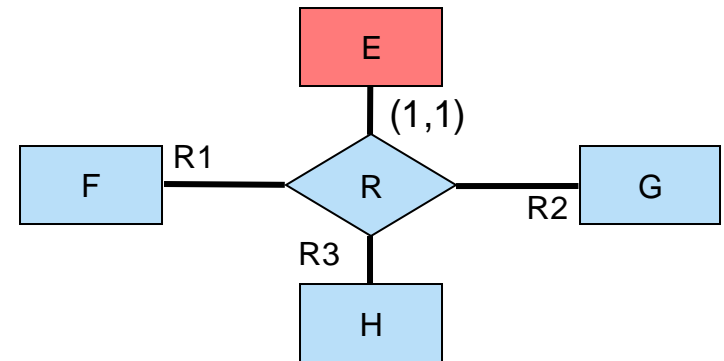
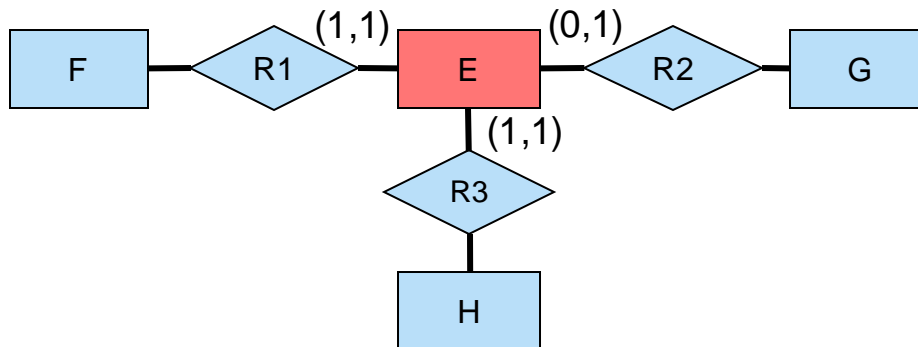
Si considerino i due schemi seguenti, e si dica se essi sono equivalenti o no dal punto di vista della semantica.





Esercizio 23: soluzione

Si considerino i due schemi seguenti, e si dica se essi sono equivalenti o no dal punto di vista della semantica.



È facile vedere che i due schemi non sono equivalenti semanticamente. Infatti, esistono istanze dello schema a sinistra in cui vi è una istanza di **E** che non è collegata ad alcuna istanza di **G** tramite **R2**. Al contrario, in ogni istanza dello schema di destra, ogni istanza di **E** è necessariamente collegata ad una istanza **r** della relazione **R** e quindi ad una istanza di **G** (quella che partecipa nel ruolo **R2** ad **r**). Si noti che il problema non si risolverebbe con la cardinalità $(0,1)$ per **E** in **R** nello schema a destra, perché a quel punto esisterebbero istanze di **E** non collegate ad alcuna istanza di **F** e **G**, cosa che nello schema a sinistra non può succedere. Ed ovviamente non si risolverebbe neppure con la cardinalità $(0,1)$ nel ruolo **R2**, perché tale cardinalità parla della partecipazione delle istanze di **G** ad **R**.



3.2.2 Scelta dei pattern di modellazione: rappresentazione di aspetti dinamici

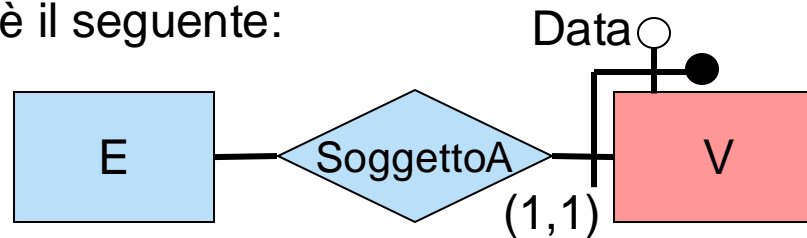
Quando modelliamo un dominio è spesso necessario rappresentare sia gli aspetti statici degli oggetti e delle relazioni rilevanti, ovvero quegli aspetti che non cambiano durante l'evoluzione del sistema informativo, sia gli aspetti dinamici, ovvero quelli che cambiano le proprietà durante tale evoluzione.

Tra i tanti aspetti dinamici che possono presentarsi, tratteremo di due tipi di aspetti dinamici che possono emergere nella modellazione concettuale e che sono forse tra i più rilevanti:

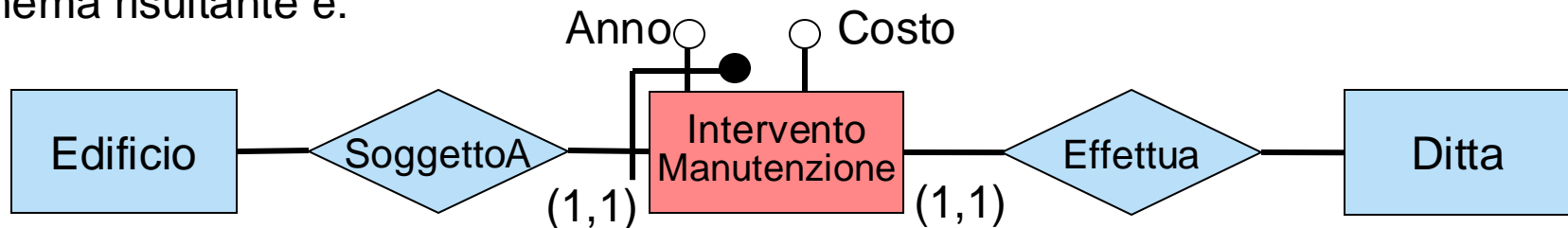
1. Rappresentazione di **eventi** che coinvolgono le istanze delle entità o delle relazioni modellate.
2. Rappresentazione “**storicizzata**” di entità o relazioni. Si tratta della rappresentazione di proprietà di istanze di una entità che cambiano durante l'evoluzione del sistema informativo e delle quali interessa il valore in diversi stati. Lo stesso fenomeno può anche riguardare le istanze di una relazione: in questo caso è sufficiente trasformare la relazione in entità (si veda più avanti) ed applicare il metodo che descriviamo per le entità.

3.2.2.1 Scelta dei pattern di modellazione: rappresentazione di eventi

Una categoria V di eventi che riguarda una entità E si può modellare mediante un'entità V connessa da una relazione (che chiamiamo "SoggettoA") ad E , in modo che ogni istanza di tale entità V rappresenti il verificarsi di un evento di tale categoria. Si assume che non possano verificarsi due o più eventi di tipo V che riguardano la stessa istanze di E in una stessa, determinata finestra temporale (in questo caso la data). Il pattern che ne deriva è il seguente:



Ogni istanza di V rappresenta un evento di tipo V al quale è soggetta una istanza di E . Ovviamente, se la granularità della finestra temporale è diversa (ad esempio, non più di un evento al mese), l'attributo Data cambierà di conseguenza. Inoltre, se l'evento ha proprietà aggiuntive, queste si modelleranno come attributi e/o relazioni dell'entità V . Consideriamo come esempio gli interventi di manutenzione di edifici. Assumiamo che sullo stesso edificio non possa essere effettuato più di un intervento all'anno e che ci interessi anche il costo di ogni intervento e la ditta che lo effettua. Lo schema risultante è:

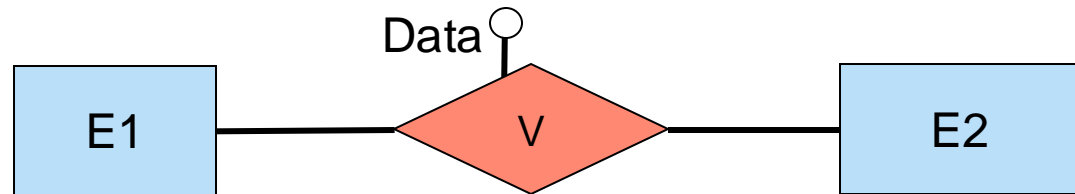


3.2.2.1 Scelta dei pattern di modellazione: rappresentazione di eventi

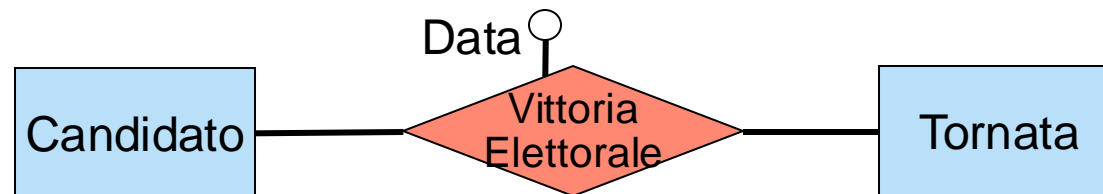
Generalizziamo le considerazioni precedenti al caso in cui la categoria V di eventi riguardi N entità (con $N > 1$), distinguendo tra due situazioni:

- 1) la prima situazione è quella in cui, date N istanze o_1, o_2, \dots, o_N , ciascuna di una delle N entità, si può verificare al massimo un evento della categoria V che riguarda o_1, o_2, \dots, o_N ;

Questa situazione può essere modellata facendo corrispondere alla categoria V una relazione tra le N entità. Ad esempio, se le due entità sono $E1$ ed $E2$ e la categoria di eventi è V , lo schema sarà:

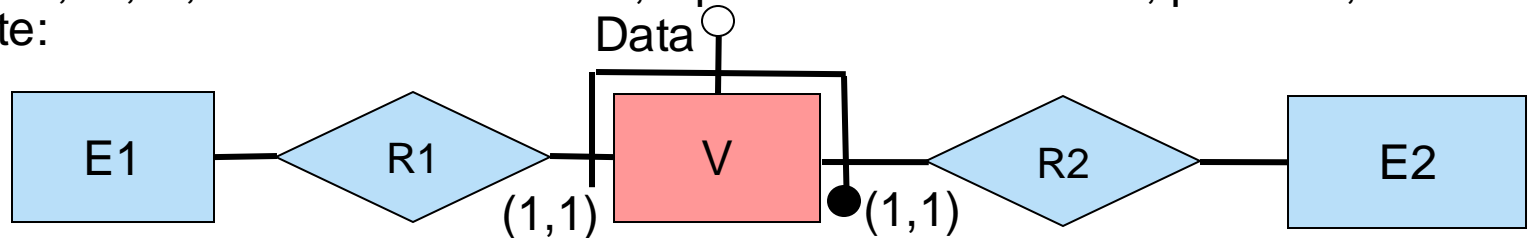


Si noti che in questo caso il verificarsi di un evento di categoria V è rappresentato da una istanza della relazione V e la data dell'evento è rappresentato dall'appropriato attributo di V . Ad esempio, se un candidato **può essere eletto al massimo una volta in una tornata elettorale**, allora l'evento che rappresenta la vittoria di una persona alle elezioni di una tornata si può modellare come segue:

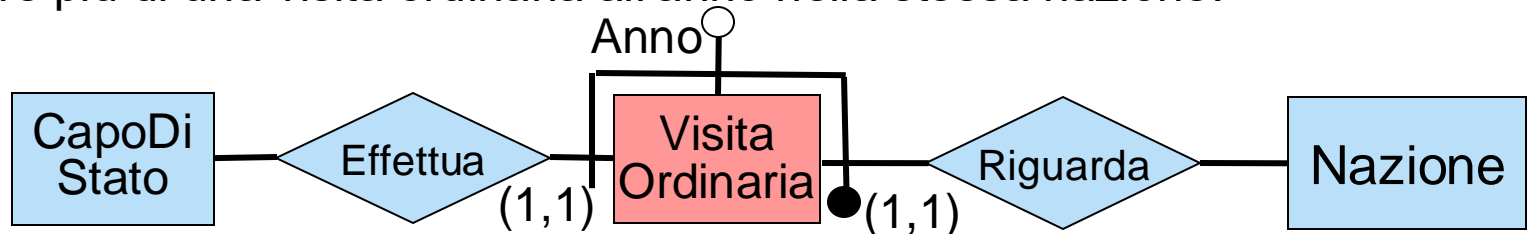


3.2.2.1 Scelta dei pattern di modellazione: rappresentazione di eventi

2) La seconda situazione è quella in cui, date N istanze o_1, o_2, \dots, o_N , ciascuna di una delle N entità, si possa verificare un numero qualunque di eventi della categoria V che riguardano o_1, o_2, \dots, o_N . In questo caso dobbiamo modellare la categoria V tramite un'entità V connessa opportunamente alle entità coinvolte, in modo che ogni istanza di tale entità rappresenti il verificarsi di un evento. Se si assume che non possano verificarsi due eventi di tipo V che riguardano la stessa combinazione di istanze o_1, o_2, \dots, o_N nella stessa data, il pattern che ne deriva, per $N=2$, è il seguente:

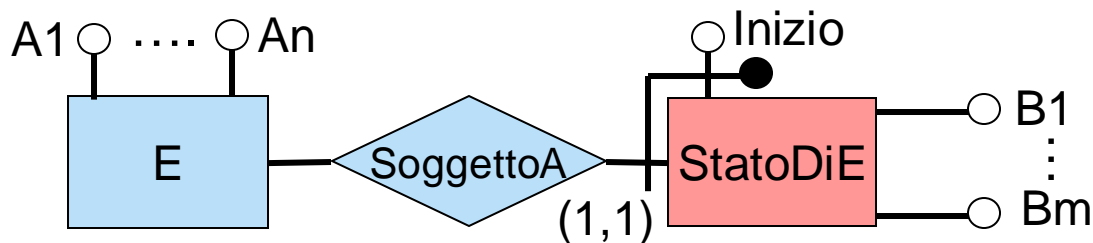


Ogni istanza di V rappresenta un evento di tipo V al quale sono soggette una istanza di $E1$ ed una istanza di $E2$. Ancora una volta, ovviamente, se la granularità del tempo è diversa (ad esempio, non più di due eventi al mese), l'attributo **Data** cambierà di conseguenza, e se l'evento ha proprietà aggiuntive, queste si modelleranno come attributi e/o relazioni dell'entità V . Consideriamo come esempio le visite ordinarie che i capi di stato effettuano nelle nazioni, assumendo che un capo di stato non possa effettuare più di una visita ordinaria all'anno nella stessa nazione:



3.2.2.2 Scelta dei pattern di modellazione: rappresentazione “storicizzata” di entità

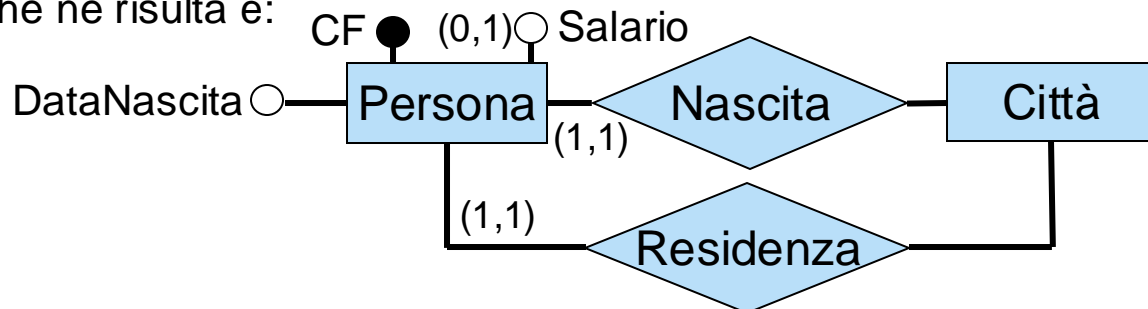
Quando per le istanze di una entità cambiano alcune proprietà e a noi interessa tenere traccia della storia di questi cambiamenti, si può fare ricorso alla nozione di “storicizzazione” dell’entità, che consiste nel modellare il concetto di “stato” dell’entità stessa. Gli stati delle istanze di una entità E si possono modellare mediante un’entità “StatodiE” connessa da una relazione (che chiamiamo “HaStato”) ad E, in modo che ogni istanza di StatodiE rappresenti appunto uno stato di una delle istanze di E. All’entità StatodiE associamo un tempo o condizione di inizio validità e tutte le proprietà di E che possono cambiare, in modo che ogni istanza di StatodiE porti i valori di tali proprietà (anche quelli che non sono cambiati). Quando il valore di qualche proprietà di una istanza di E cambia, questo verrà segnalato dalla esistenza di una nuova istanza di StatodiE, al quale attribuiremo il valore (nuovo o vecchio) delle varie proprietà. Il pattern che ne deriva è il seguente:



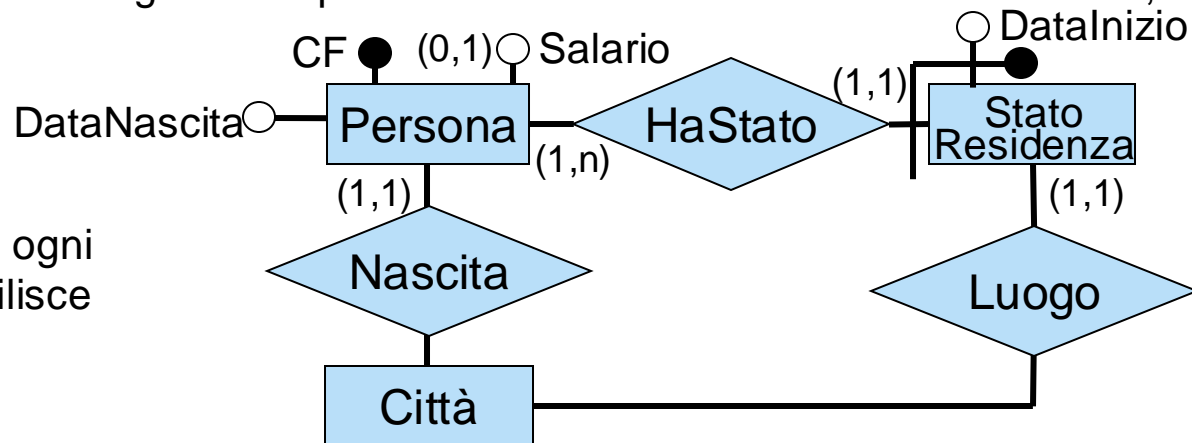
Si noti che le proprietà A_1, \dots, A_n sono le proprietà di E che non cambiano (o la cui evoluzione non ci interessa). Talvolta si usa il termine **rigide** per indicare tali proprietà. Le proprietà B_1, \dots, B_m sono invece le proprietà di E che ci interessa storicizzare, ovvero la cui evoluzione ci interessa tracciare. Esse non sono attribuite ad E, ma a StatodiE. L’attributo “Inizio” rappresenta il tempo (ad esempio una data) o un qualunque valore (anche un numero progressivo, ad esempio) che segnala l’inizio di validità dello stato.

3.2.2.2 Scelta dei pattern di modellazione: rappresentazione “storicizzata” di entità

Mostriamo un esempio del pattern di modellazione appena descritto. Concentriamoci sull'entità Persona e sulle seguenti proprietà: codice fiscale (CF, identificativo), data di nascita, città di nascita, salario e città di residenza. Codice fiscale, data di nascita e città di nascita sono chiaramente proprietà rigide delle persone, nel senso che non possono cambiare. Assumiamo di essere interessati solo alla città di residenza attuale e al salario attuale, se esiste, delle persone. Lo schema concettuale che ne risulta è:



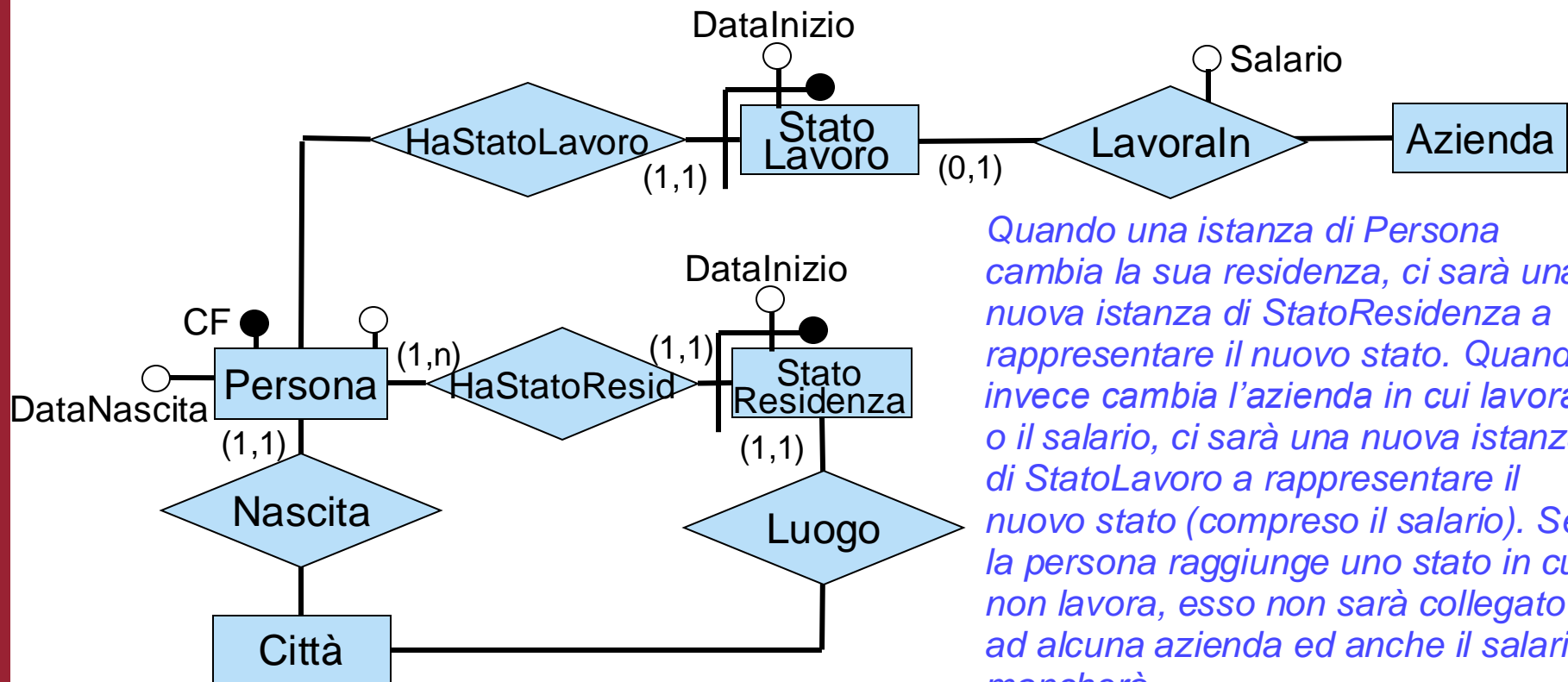
Assumiamo ora che i requisiti cambino in modo che nella nostra applicazione sia necessario tenere traccia di tutte le città in cui le persone hanno risieduto, con la data di inizio che caratterizza il periodo di residenza. Seguendo il pattern di modellazione di entità storicizzata, lo schema diventa il seguente:



Si noti che la data associata ad ogni istanza di StatoResidenza stabilisce l'ordine con cui si susseguono gli stati per quella istanza.

3.2.2.2 Scelta dei pattern di modellazione: rappresentazione “storicizzata” di entità

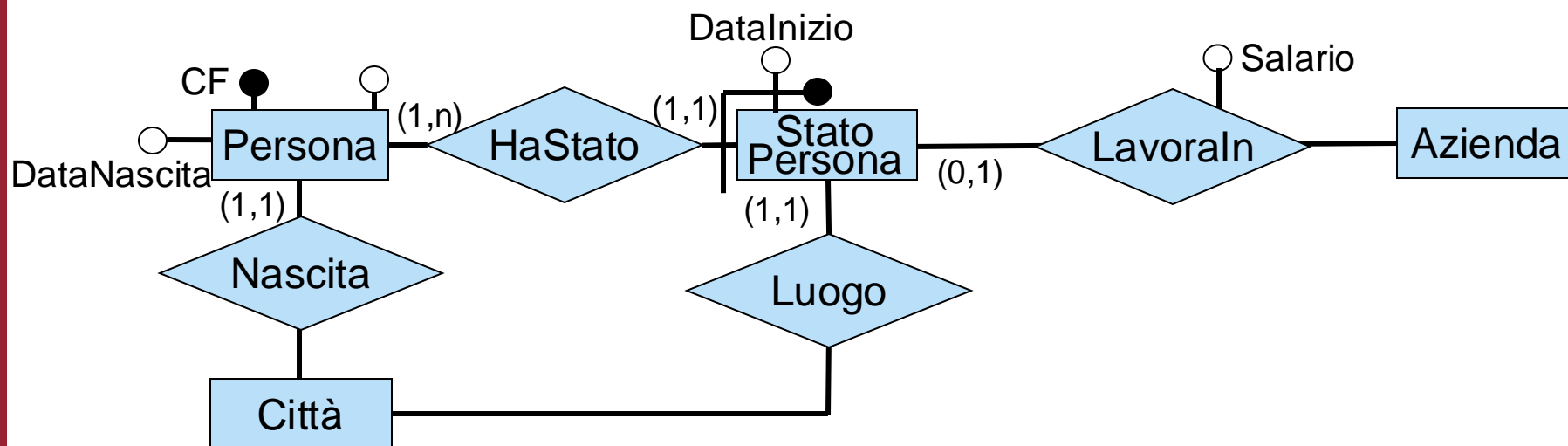
Assumiamo ora che i requisiti cambino ulteriormente: dobbiamo tenere traccia anche di tutti i vari salari di cui ha goduto la persona nel corso del tempo, insieme ai rapporti di lavoro con le aziende nei quali è stato coinvolto. In altre parole, dobbiamo storicizzare “salario” e dobbiamo anche storicizzare una nuova relazione “lavora in” con azienda. Una prima possibilità è aggiungere una nuova entità che riguarda lo stato della persona rispetto al lavoro, applicando il pattern di modellazione alle proprietà relative al lavoro. Lo schema risultante è:



Quando una istanza di Persona cambia la sua residenza, ci sarà una nuova istanza di StatoResidenza a rappresentare il nuovo stato. Quando invece cambia l'azienda in cui lavora o il salario, ci sarà una nuova istanza di StatoLavoro a rappresentare il nuovo stato (compreso il salario). Se la persona raggiunge uno stato in cui non lavora, esso non sarà collegato ad alcuna azienda ed anche il salario mancherà.

3.2.2.2 Scelta dei pattern di modellazione: rappresentazione “storicizzata” di entità

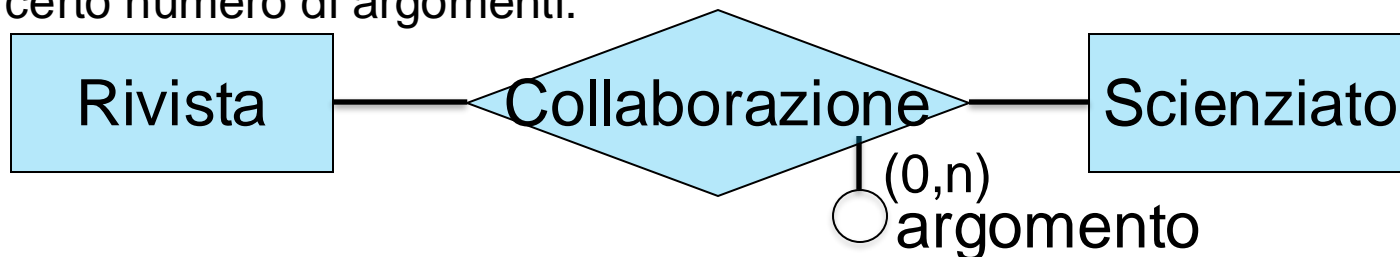
Una seconda possibilità è scegliere di rappresentare gli stati di una persona mediante un'unica entità “StatoPersona” che ingloba tutte le proprietà che possono cambiare e delle quali vogliamo tenere traccia. Lo schema risultante è il seguente:



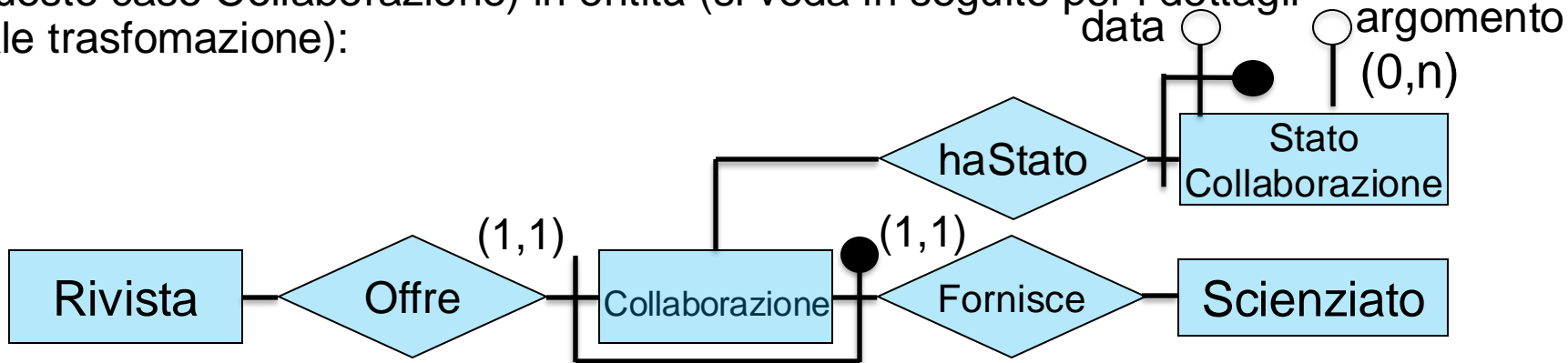
In questa soluzione, appena una delle proprietà storicizzate di **Persona** cambia per una certa istanza, vi sarà una nuova istanza di **StatoPersona** che rappresenta il nuovo stato di tale istanza. Ed in ogni stato memorizziamo i valori di tutte le proprietà rilevanti storicizzate. Si noti che nelle soluzioni discusse i valori attuali delle proprietà di una certa istanza di **Persona** sono quelli corrispondenti allo stato della istanza con la data più recente.

3.2.2.3 Scelta dei pattern di modellazione: rappresentazione “storicizzata” di relazione

La nozione di storicizzazione si può applicare non solo alle entità, ma anche alle relazioni. Consideriamo la relazione di collaborazione tra riviste e scienziati, ciascuna su un certo numero di argomenti.



Se i requisiti si modificano in modo che sia necessario registrare i cambiamenti delle collaborazioni per quanto riguarda gli argomenti, possiamo decidere di modellare gli stati delle collaborazioni, promuovendo la relazione ad entità (reificazione) ed imponendo che non si possano avere due stati diversi della nuova entità nello stesso giorno per la stessa coppia <rivista,scienziato>. Lo schema risultante può essere il seguente, ottenuto utilizzando un opportuno metodo per trasformare una relazione (in questo caso Collaborazione) in entità (si veda in seguito per i dettagli su tale trasformazione):





3.2.2 Scelta dei pattern di modellazione: eventi vs stati

Abbiamo parlato di proprietà dinamiche che si possono esprimere in termini di eventi e di stati. Come facciamo a distinguere e poi scegliere tra questi due tipi di evoluzioni del sistema informativo?

Non sempre è facile decidere se rappresentare una certa condizione dinamica come evento o come stato, ma la seguente indicazione potrebbe essere utile nella pratica.

1. Se l'evoluzione che vogliamo rappresentare è basata sul verificarsi di un certo fenomeno (evento, appunto) in un certo **tempo** o in un certo **contesto**, e questo fenomeno è di interesse per se nella nostra applicazione, allora ci dobbiamo orientare verso la scelta di rappresentare l'evoluzione in termini di eventi.
2. Se l'evoluzione che vogliamo rappresentare è basata invece sul succedersi di cambiamenti che non siamo necessariamente interessati a vedere come accadimenti, ma dei quali siamo invece interessati a rappresentarne **gli effetti (stati risultanti)**, allora possiamo orientarci verso la scelta di rappresentare l'evoluzione in termini di sequenze di stati.

Spesso, comunque, la stessa evoluzione può essere rappresentata correttamente in entrambi i modi e la scelta dipende da fattori soggettivi.



3.3 Operazioni di trasformazione dello schema

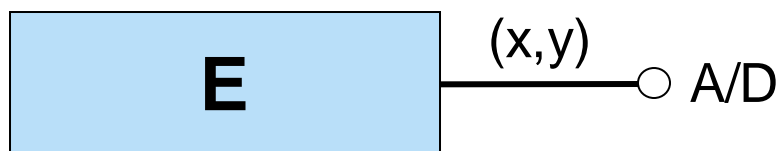
Indipendentemente dalla strategia adottata, è spesso necessario durante la progettazione concettuale effettuare trasformazioni che **preservino l'equivalenza**. A livello intuitivo, una trasformazione dallo schema S allo schema T preserva l'equivalenza se esiste una funzione che associa ad ogni istanza di S una istanza di T, ed una funzione che associa ad ogni istanza di T una di S.

Le più comuni sono di tipo «reificazione» (trasformazione in entità):

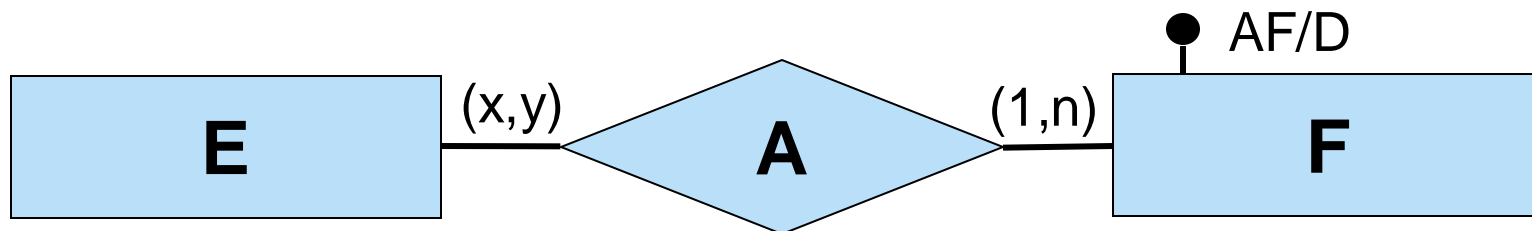
- a) Trasformazione di un attributo semplice di un'entità in una relazione (e del corrispondente dominio in entità)
- b) Trasformazione di un attributo composto di un'entità in un'altra entità connessa alla prima da una relazione
- c) Trasformazione di una relazione in una entità
- d) Trasformazione di un attributo di una relazione R in una relazione (e del corrispondente dominio in entità). Si noti che questa trasformazione può implicare anche la trasformazione di R in entità

3.3 Trasformazioni: caso a1

Trasformazione di un attributo semplice A di un'entità in una relazione (e del corrispondente dominio in entità).



Nel caso in cui E sia l'unica entità con attributo A, si trasforma in:



La cardinalità (1,n) sulla partecipazione di F alla relazione A denota che interessano solo le istanze di F che rappresentano valori dell'attributo A effettivamente assunti dalle istanze di E nello schema originario.

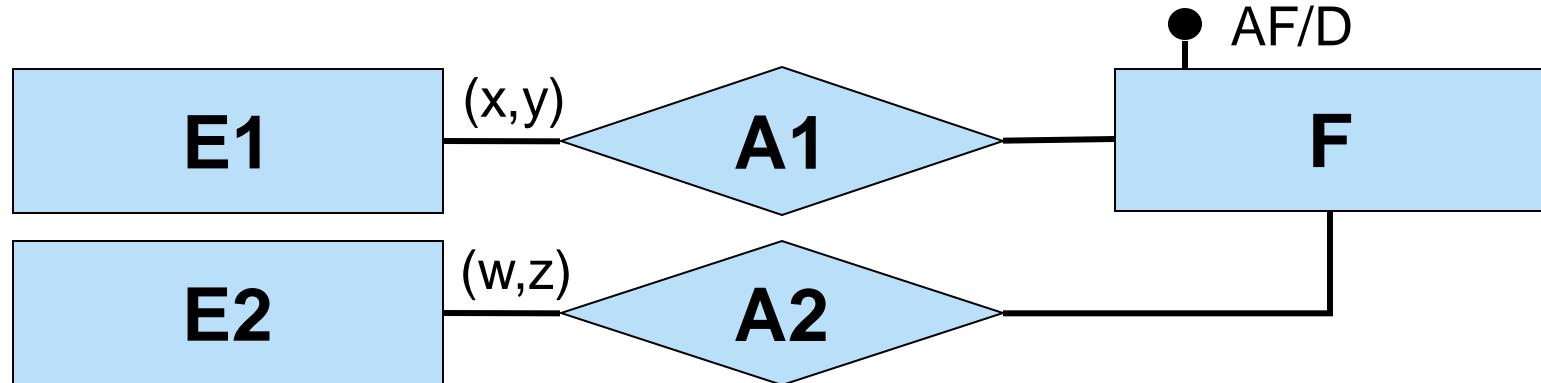


3.3 Trasformazioni: caso a2

Trasformazione di un attributo semplice A di un'entità in una relazione (e del corrispondente dominio in entità).



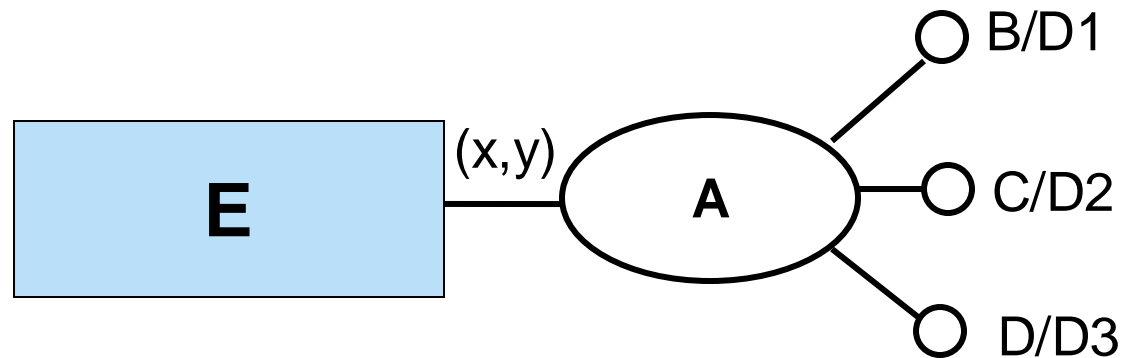
Nel caso in cui vi siano più entità con attributo A e con stesso dominio D associato ad A, si trasforma in:



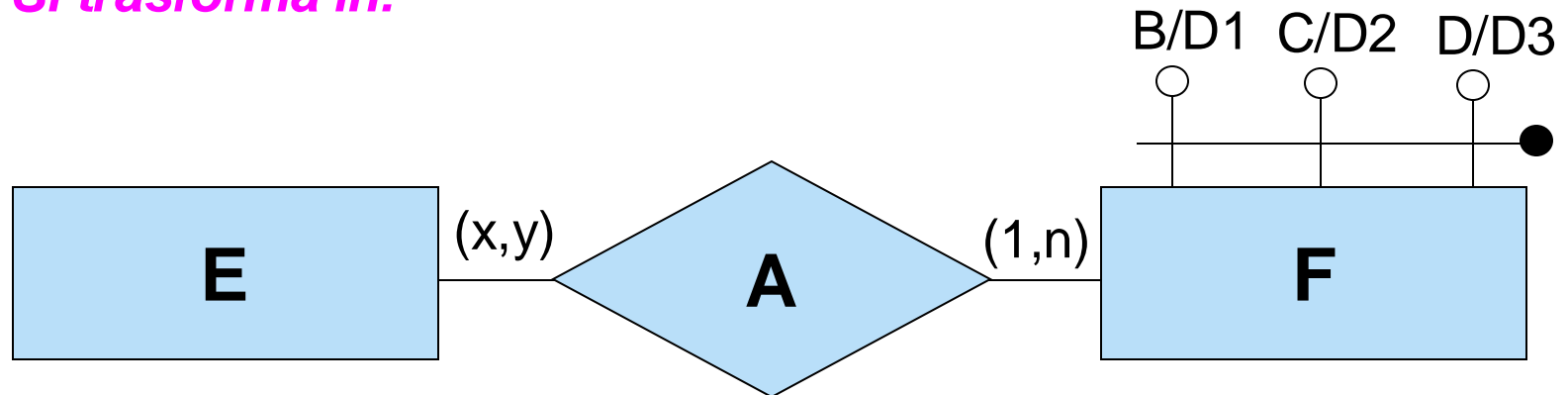
È necessario aggiungere un **vincolo esterno** di cardinalità (1,n) sulla partecipazione di F all'unione delle relazioni A1 e A2.

3.3 Trasformazioni: caso b

Trasformazione di un attributo A composto di un'entità in un'altra entità connessa alla prima da una relazione.

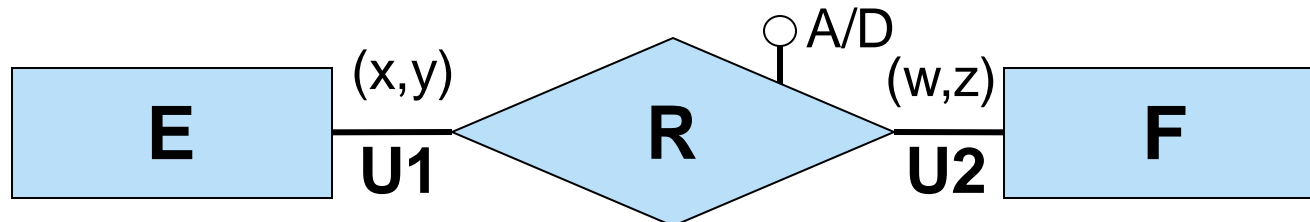


Si trasforma in:

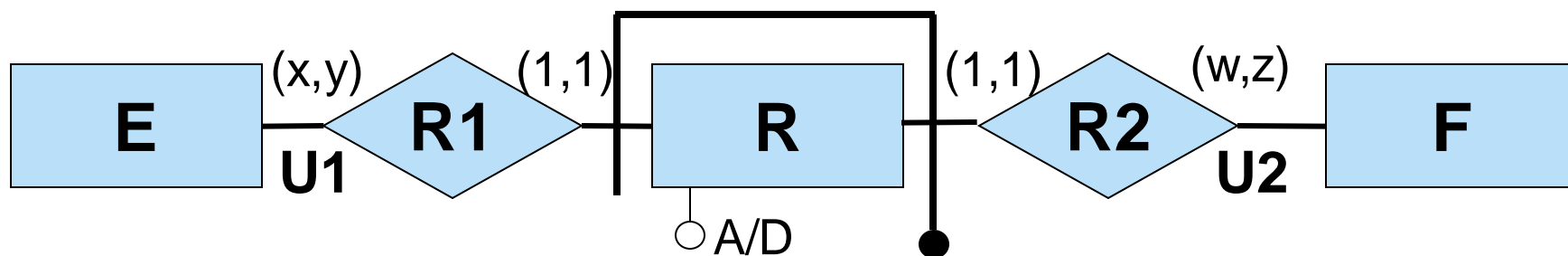


3.3 Trasformazioni: caso c1 (a)

Trasformazione di una relazione R in una entità.

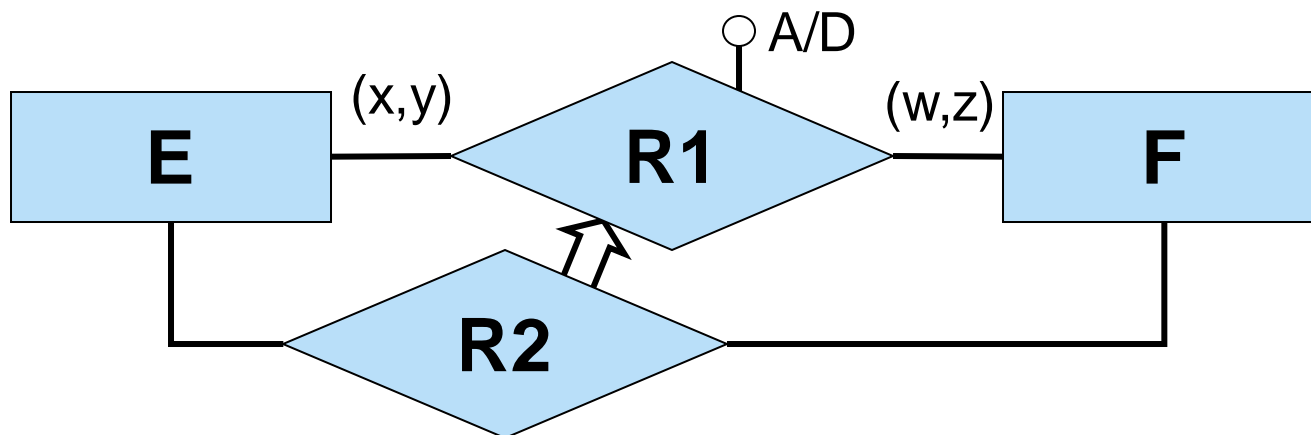


Secondo la strategia che chiameremo c1, se la relazione connette N entità, essa diventa una entità collegata alle N entità mediante altrettante relazioni binarie con cardinalità (1,1). Ad esempio, se la relazione R connette le due entità E ed F, allora secondo la strategia c1 si trasforma così (si noti l'identificatore della nuova entità, che riflette il fatto che nello schema originario non esistono due tuple di R che connettono la stessa coppia di istanze di E ed F):

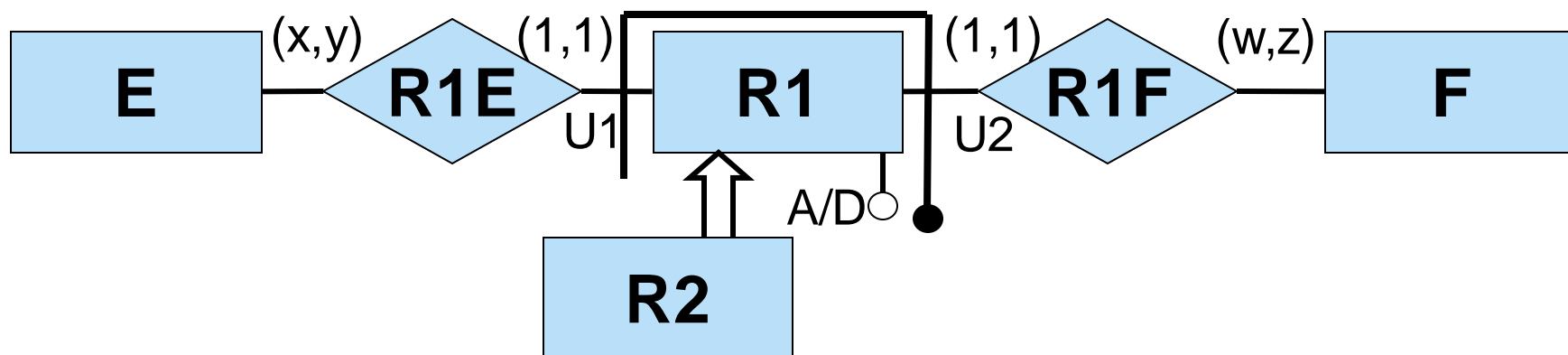


3.3 Trasformazioni: caso c1 (b)

Nella trasformazione di una relazione R in una entità, anche ogni relazione in ISA con R deve essere trasformata in entità.

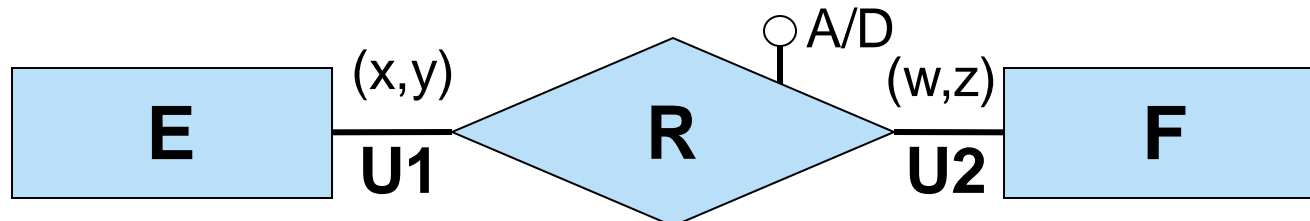


Secondo la strategia c1 si trasforma in:

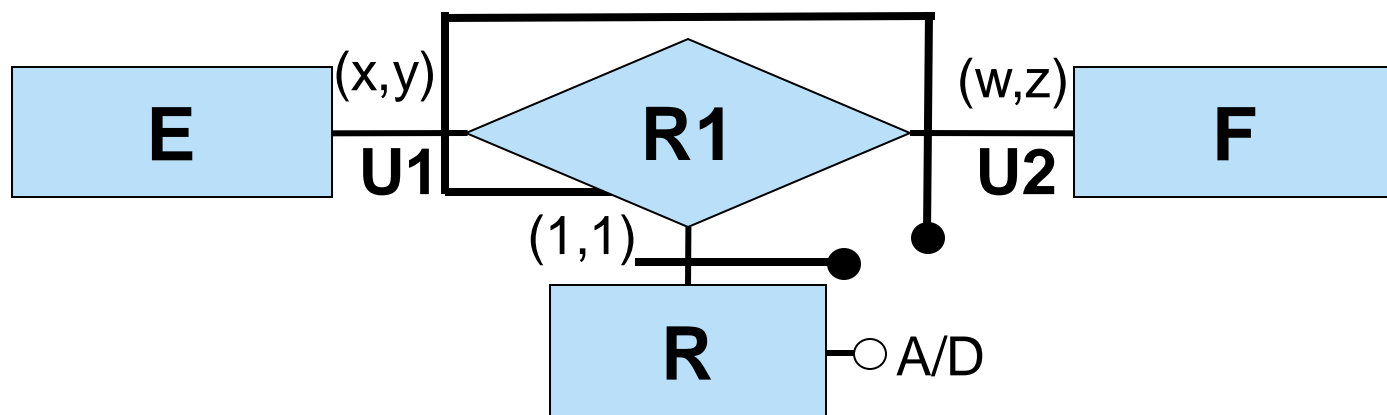


3.3 Trasformazioni: caso c2 (a)

Trasformazione di una relazione in una entità.

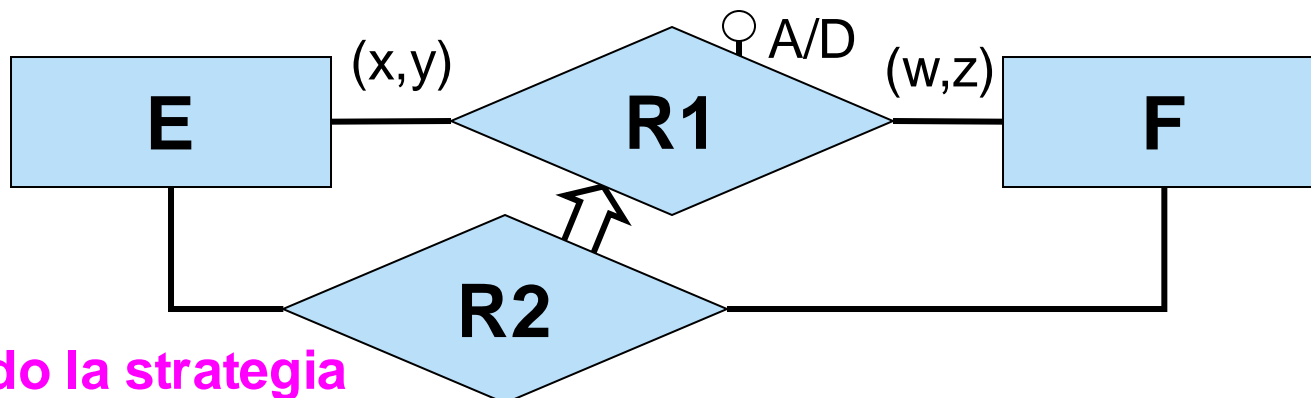


Secondo la strategia che chiameremo c2, se la relazione R connette N entità, essa diventa una entità R collegata, con cardinalità $(1,1)$, ad una nuova relazione che connette sia le N entità sia la nuova entità R . Ad esempio, se la relazione R connette le due entità E ed F , allora secondo la strategia c2 si trasforma così (si noti l'identificatore della nuova entità e quello della nuova relazione):

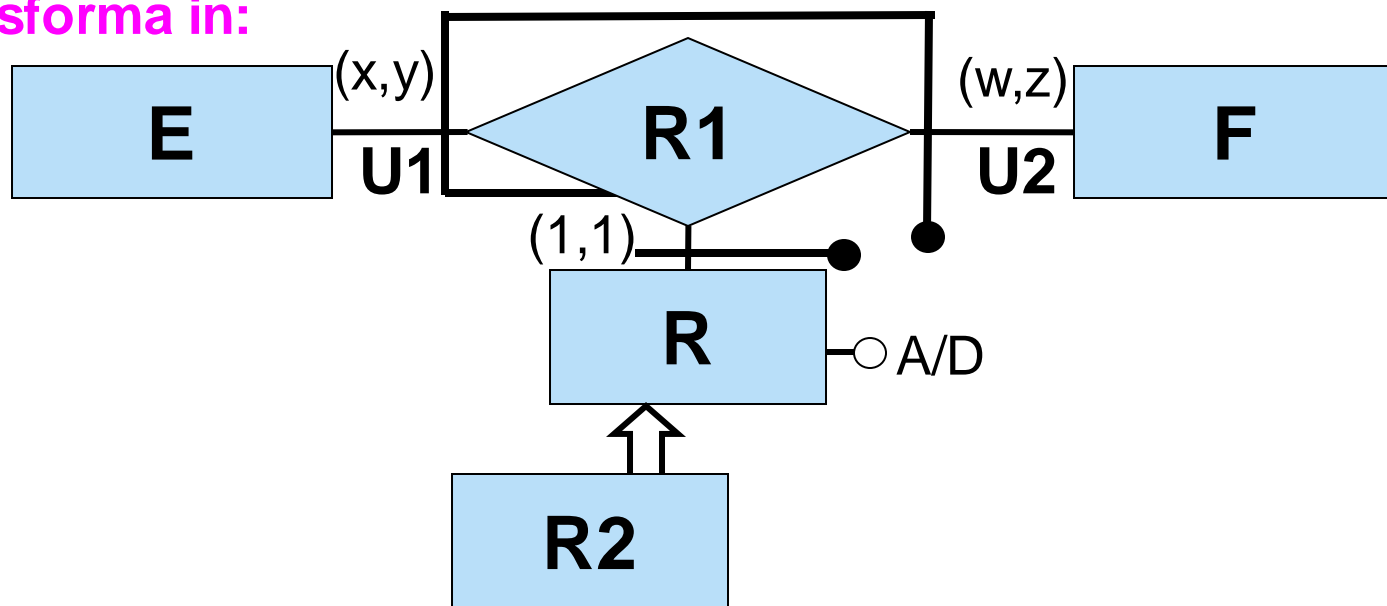


3.3 Trasformazioni: caso c2 (b)

Nella trasformazione di una relazione R in una entità, anche ogni relazione in ISA con R deve essere trasformata in entità.



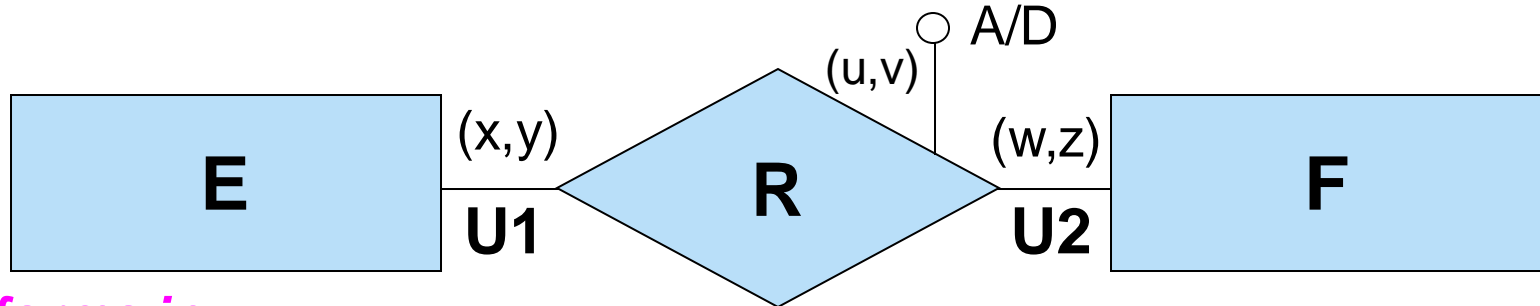
Secondo la strategia
c2 si trasforma in:



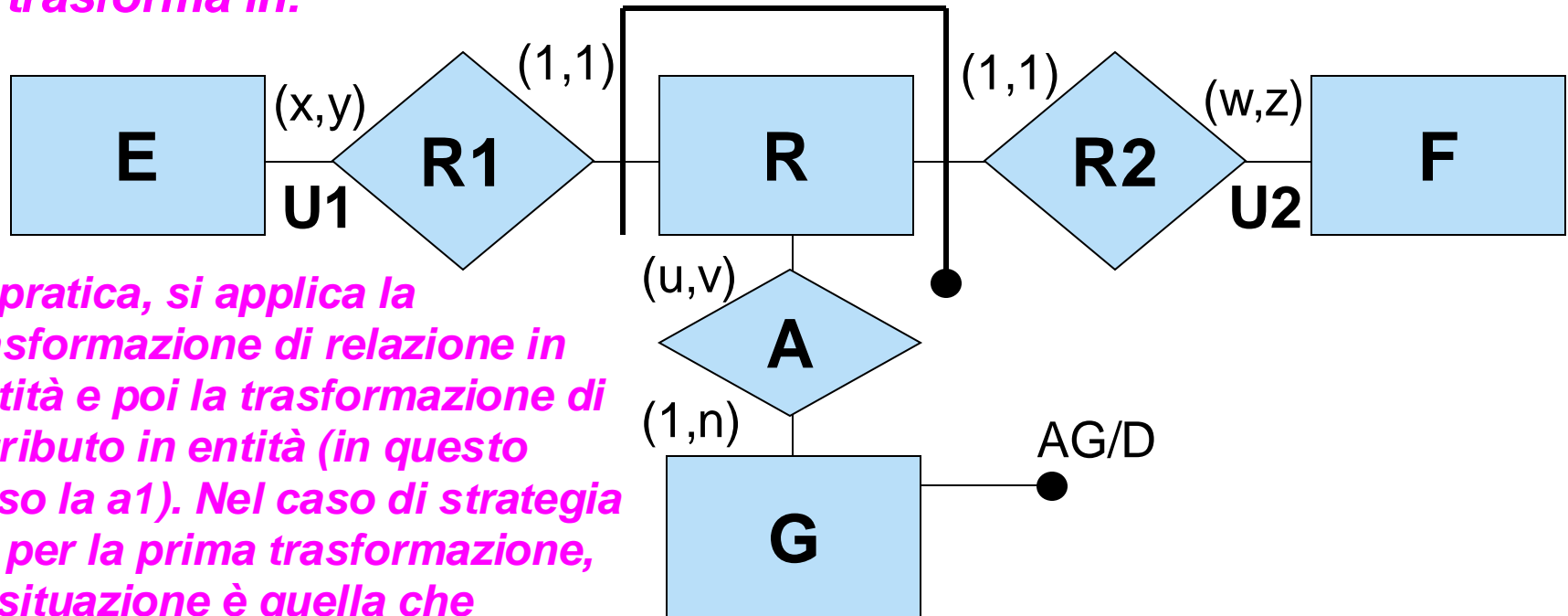


3.3 Trasformazioni: caso d1

Trasformazione di un attributo semplice A di una relazione R in una relazione (e del corrispondente dominio in entità).



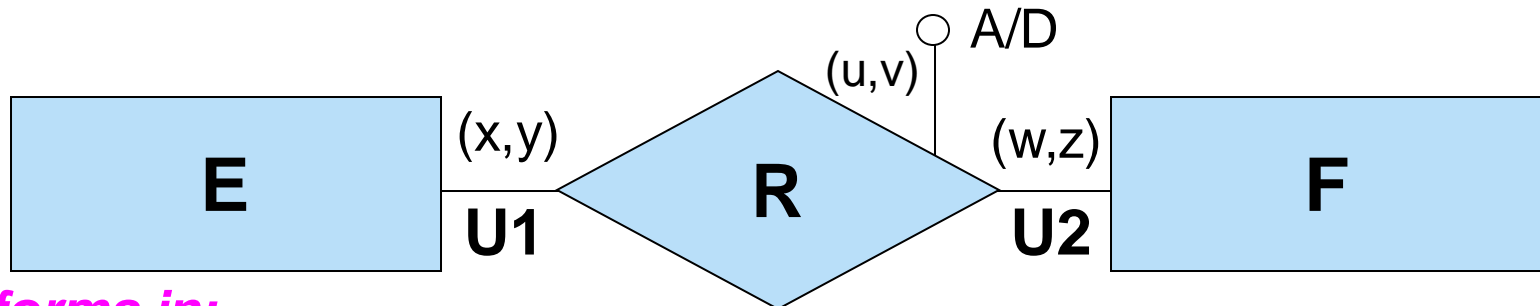
Si trasforma in:



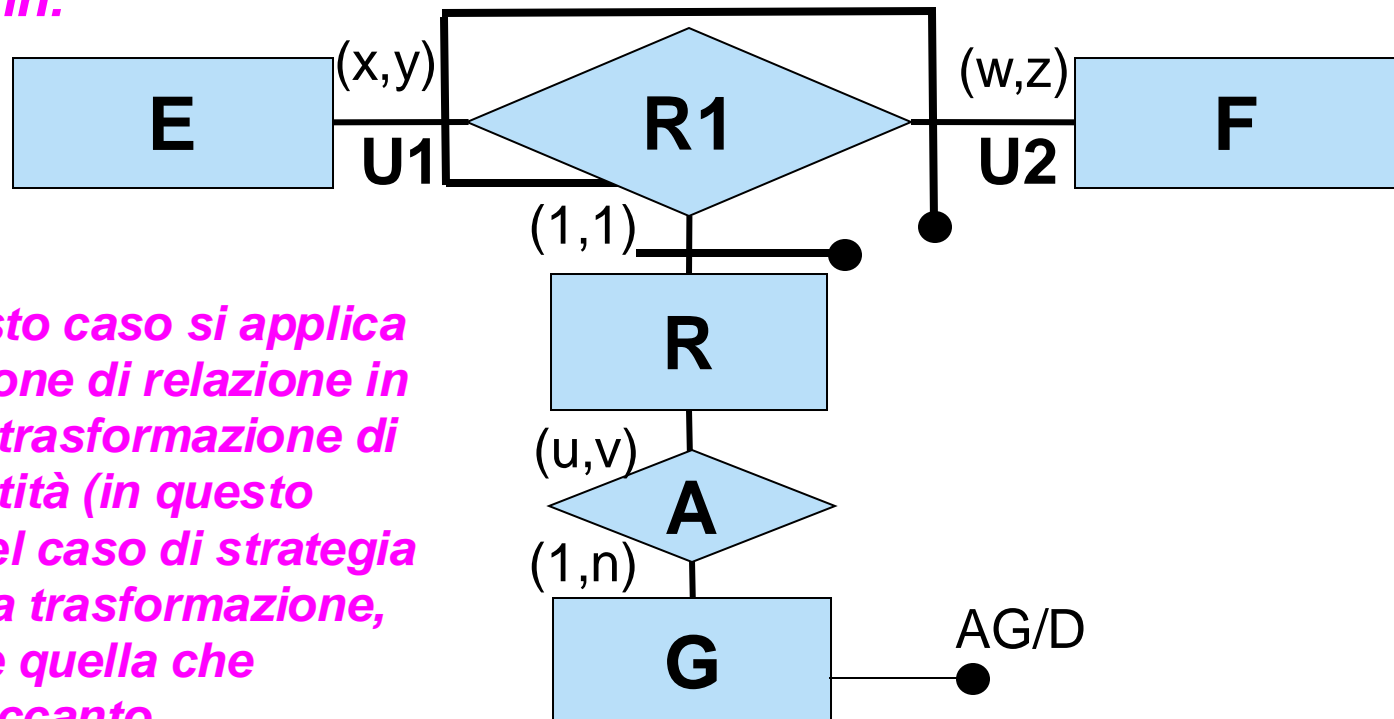
In pratica, si applica la trasformazione di relazione in entità e poi la trasformazione di attributo in entità (in questo caso la a1). Nel caso di strategia c1 per la prima trasformazione, la situazione è quella che vediamo qui accanto.

3.3 Trasformazioni: caso d2

Trasformazione di un attributo semplice A di una relazione R in una relazione (e del corrispondente dominio in entità).



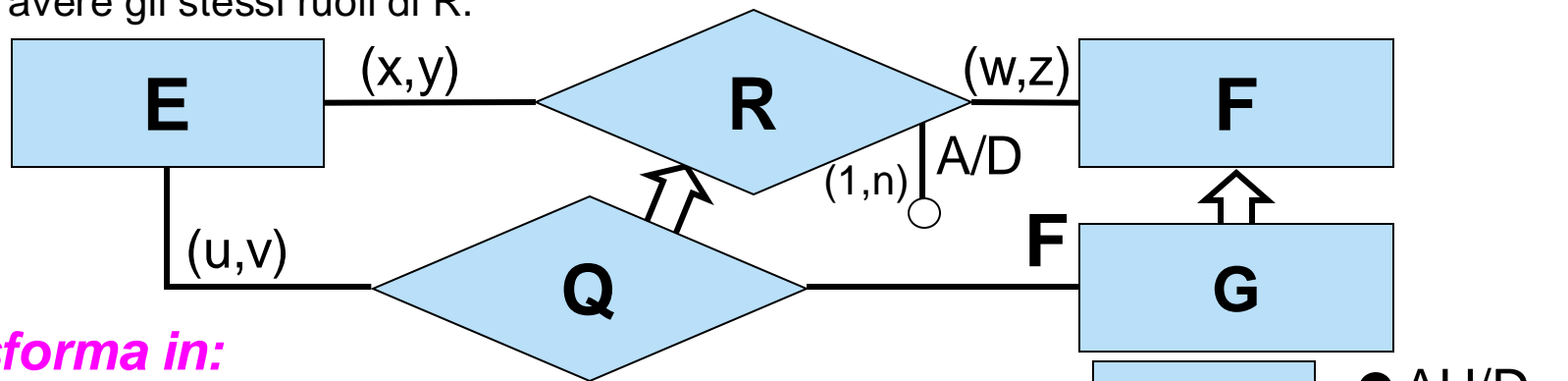
Si trasforma in:



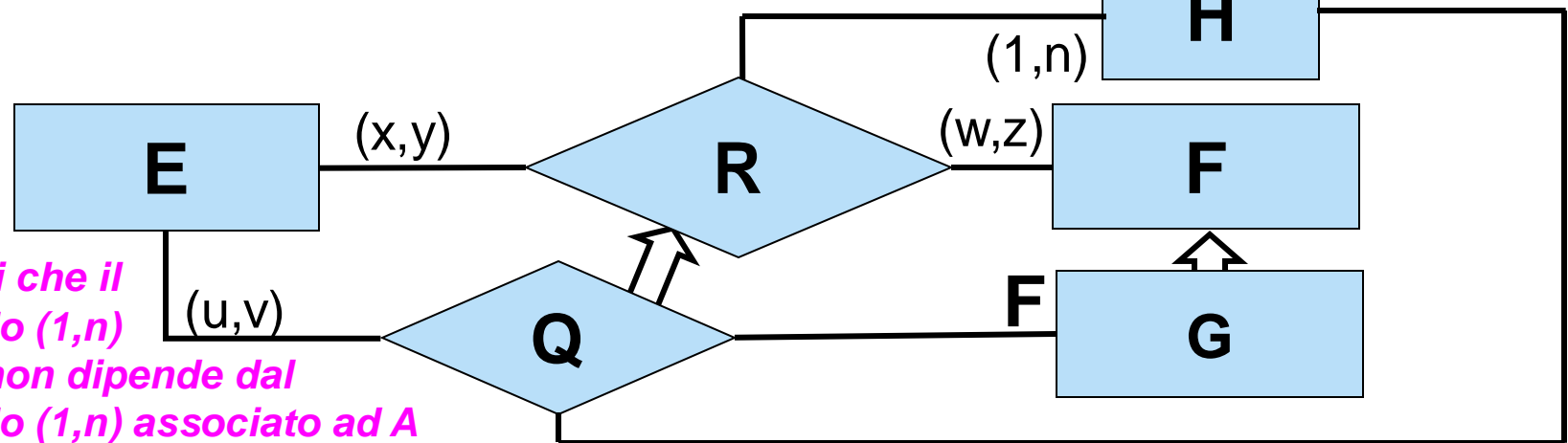
Anche in questo caso si applica la trasformazione di relazione in entità e poi la trasformazione di attributo in entità (in questo caso la a1). Nel caso di strategia c2 per la prima trasformazione, la situazione è quella che vediamo qui accanto.

3.3 Trasformazioni: caso e1

Se la cardinalità (u,v) dell'attributo è $(1,n)$, se la cardinalità massima dei ruoli di R è n (ossia, entrambi y e z sono n) e se R non ha altri attributi oltre ad A ed ha come identificatore quello implicito, allora si può effettuare una diversa trasformazione: si trasforma l'attributo multivalore di R in un'entità che viene connessa alla relazione R (che quindi aumenta la arietà). L'identificatore della nuova relazione è di nuovo quello implicito (ma adesso con un ruolo in più). Anche eventuali relazioni in ISA con R devono essere connesse alla nuova entità, perché devono avere gli stessi ruoli di R .



Si trasforma in:

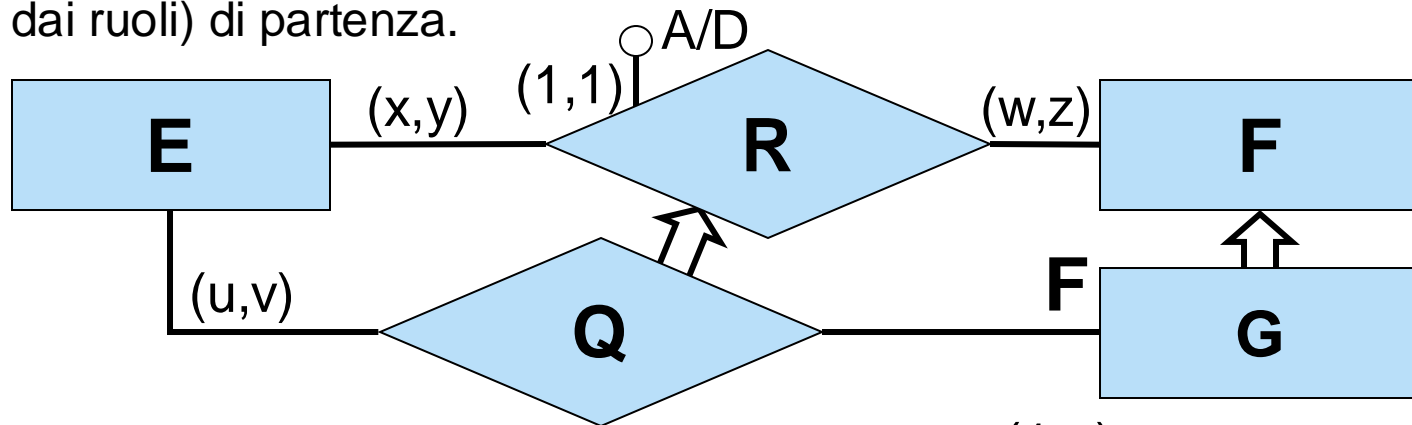


Si noti che il vincolo $(1,n)$ su H non dipende dal vincolo $(1,n)$ associato ad A

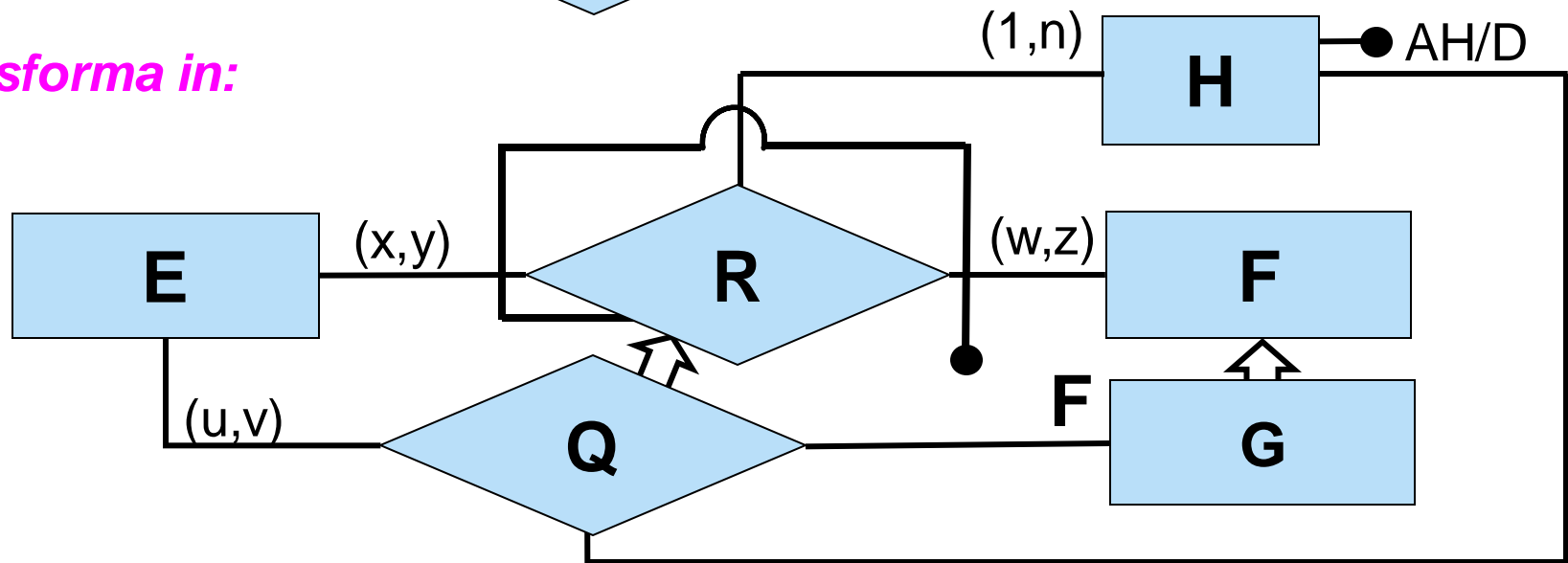


3.3 Trasformazioni: caso e2

Se la cardinalità (u,v) dell'attributo è $(1,1)$, allora si può effettuare una trasformazione analoga al caso e1, in cui però, visto che la cardinalità massima dell'attributo è 1, l'identificatore della relazione R non è quello implicito, ma è formato dalle entità (o meglio dai ruoli) di partenza.



Si trasforma in:





Fase 4: Controllo di qualità dello schema concettuale

Cosa guida nelle scelte di progetto da fare per arrivare allo schema ER finale?

Fattori di qualità:

- **Correttezza: adeguatezza e completezza**

- La realtà d'interesse deve essere colta in modo adeguato e completo

- **Minimalità**

- Evitare quanto possibile di rappresentare più volte la stessa proprietà, e comunque documentare eventuale ridondanze (*nota che le ridondanze, se non riconosciute, possono compromettere la correttezza dello schema*)

- **Leggibilità**

- Progettare uno schema che sia di facile e diretta interpretazione (*es. evitare vincoli esterni, se è possibile farlo senza perdere la correttezza e senza pregiudicare la semplicità di interpretazione del diagramma*)



4.1 La correttezza

Adeguatezza

- **Rispetto al modello:** rispettare le regole del modello dei dati
- **Rispetto alle specifiche:** descrivere in modo fedele la realtà, secondo quanto descritto nelle specifiche

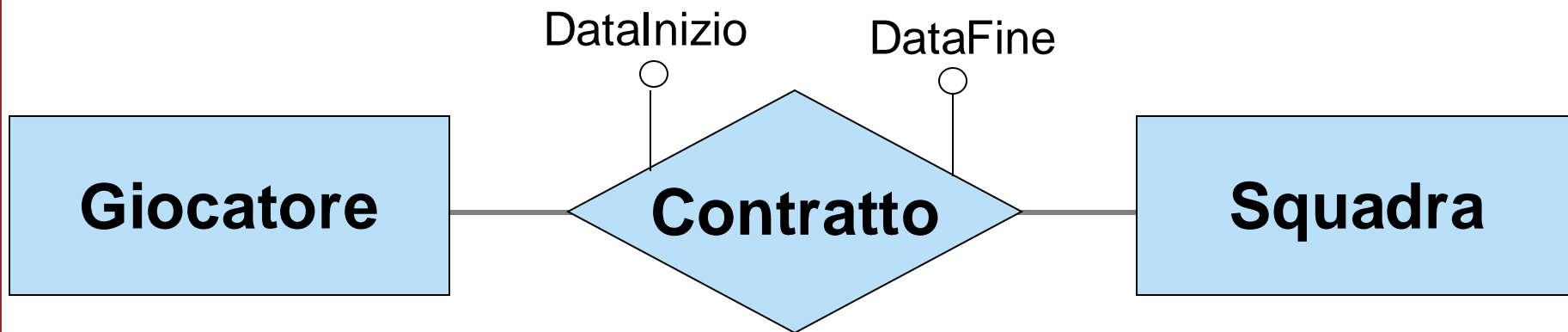
Completezza

- la realtà d'interesse deve essere colta in modo completo



Esercizio 24: correttezza

Si vogliono rappresentare le squadre in cui un giocatore milita attualmente, ed ha militato nel passato, con data inizio e data fine del contratto

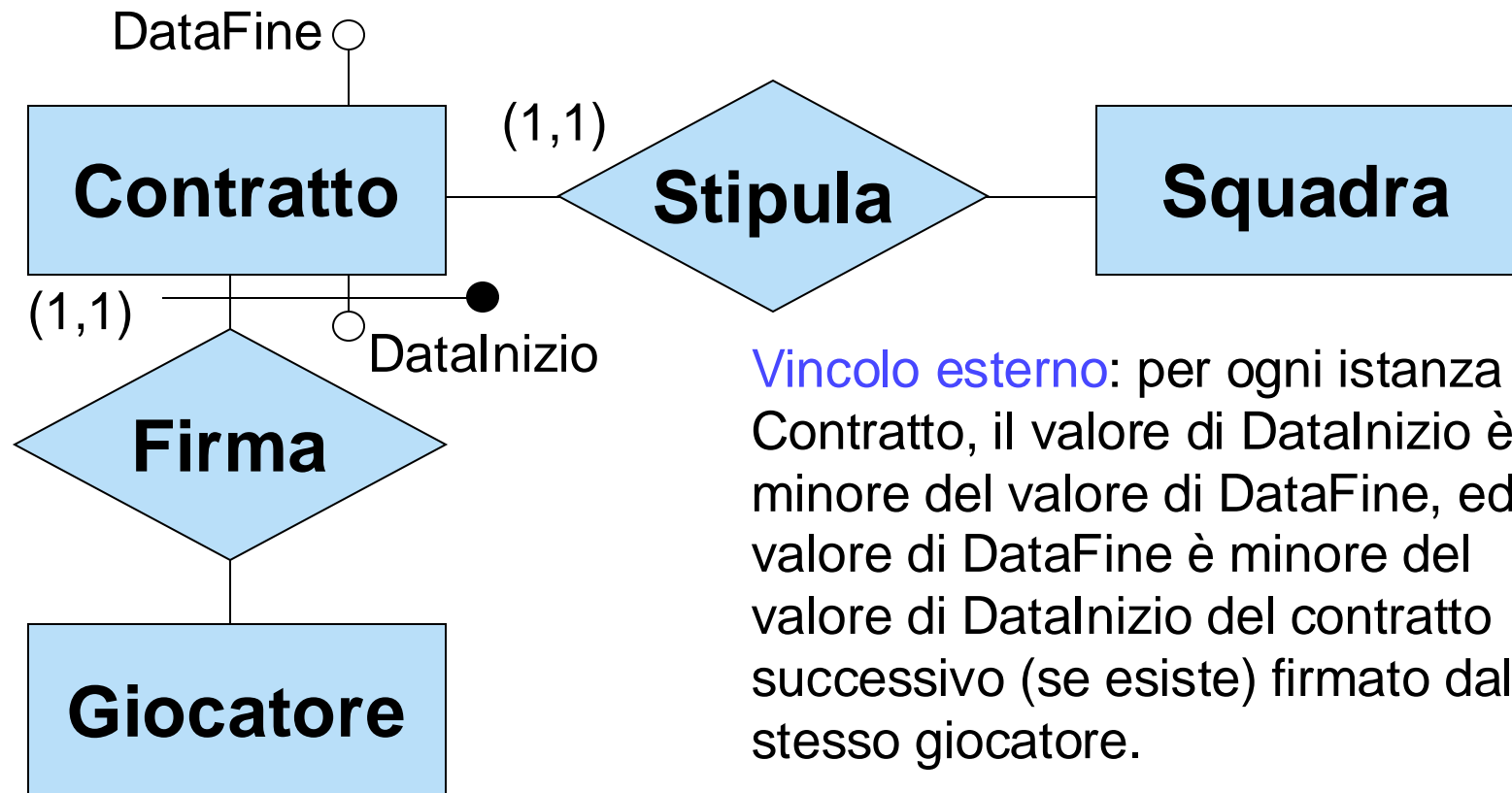


È corretto lo schema?



Esercizio 24: soluzione

Lo schema precedente non è corretto perché non ammette che un giocatore possa firmare due contratti (in epoche diverse) per la stessa squadra. Lo schema corretto è il seguente, assumendo che nessun giocatore firmi più di un contratto nella stessa data:



Vincolo esterno: per ogni istanza di Contratto, il valore di DataInizio è minore del valore di DataFine, ed il valore di DataFine è minore del valore di DataInizio del contratto successivo (se esiste) firmato dallo stesso giocatore.



4.2 La minimalità

Evitare quanto più possibile di rappresentare più volte la stessa proprietà: lasciare le ridondanze nello schema solo quando esse sono rilevanti dal punto di vista concettuale

- Ridondanze intensionali (in genere da evitare): si può “semplificare” lo schema preservando l’equivalenza?

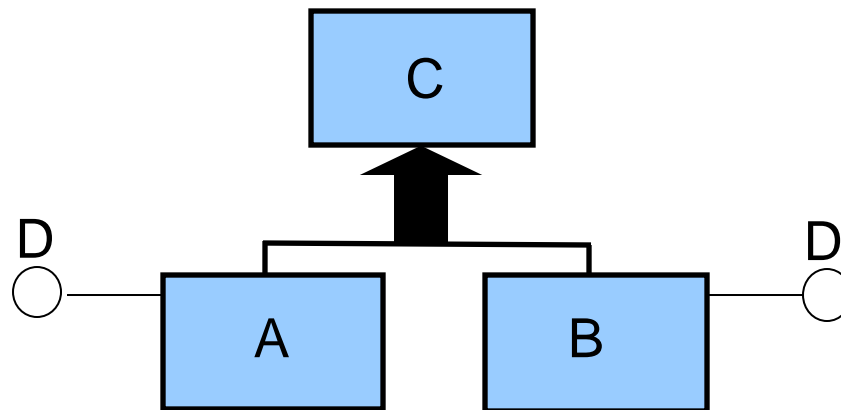
Esempio importante: evitare cicli nella relazione ISA o evitare di indicare identificatori inessenziali

- Ridondanze estensionali: nelle istanze dello schema la stessa proprietà estensionale è rappresentata più volte, implicitamente o esplicitamente

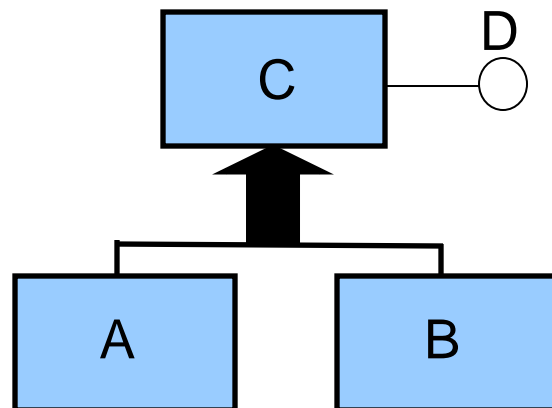
Se lo schema contiene ridondanze estensionale, occorre documentarle: se una ridondanza estensionale è rilevante a livello concettuale, occorre comunque esplicitarla, in genere mediante opportuni **vincoli di integrità**

4.2 La minimalità: ridondanza intensionale

Esempio di ridondanza intensionale (si noti che la generalizzazione è completa)



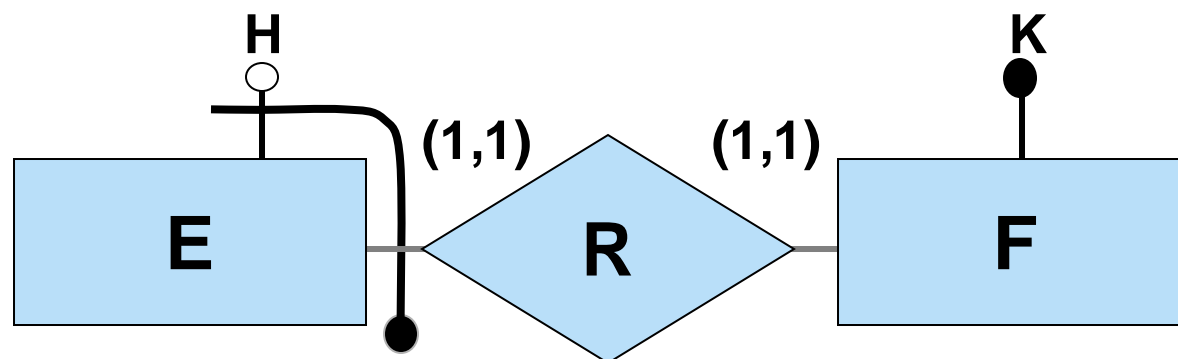
Risoluzione della ridondanza



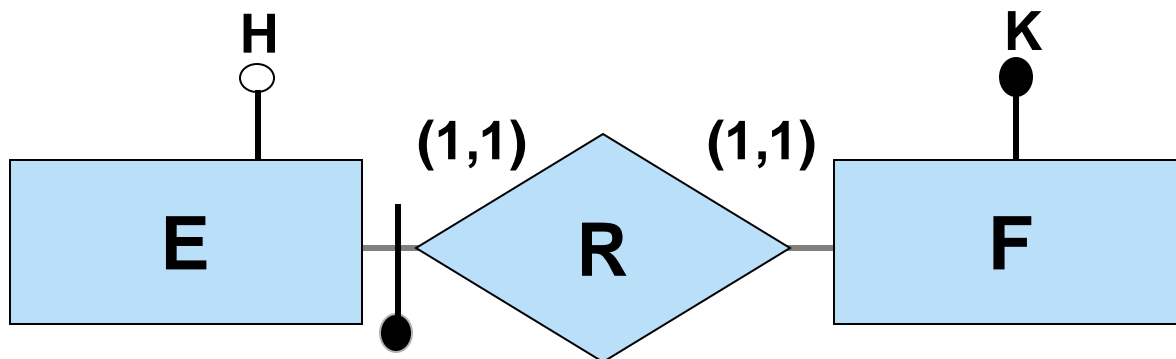
4.2 La minimalità: ridondanza intensionale

Un esempio di ridondanza intensionale è rappresentato da un identificatore inessenziale di una entità, cioè costituito da un soprainsieme delle proprietà che formano un altro identificatore della stessa entità.

Esempio di
ridondanza
intensionale



Risoluzione della
ridondanza

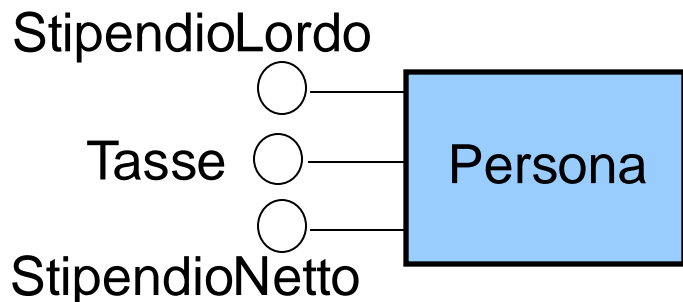


4.2 La minimalità: ridondanze estensionali

Due tipi importanti di ridondanze estensionali:

- il valore di un attributo si calcola sulla base di altre proprietà
- una relazione si ottiene da altre relazioni (per esempio da cammini di relazioni)

Esempio del primo tipo



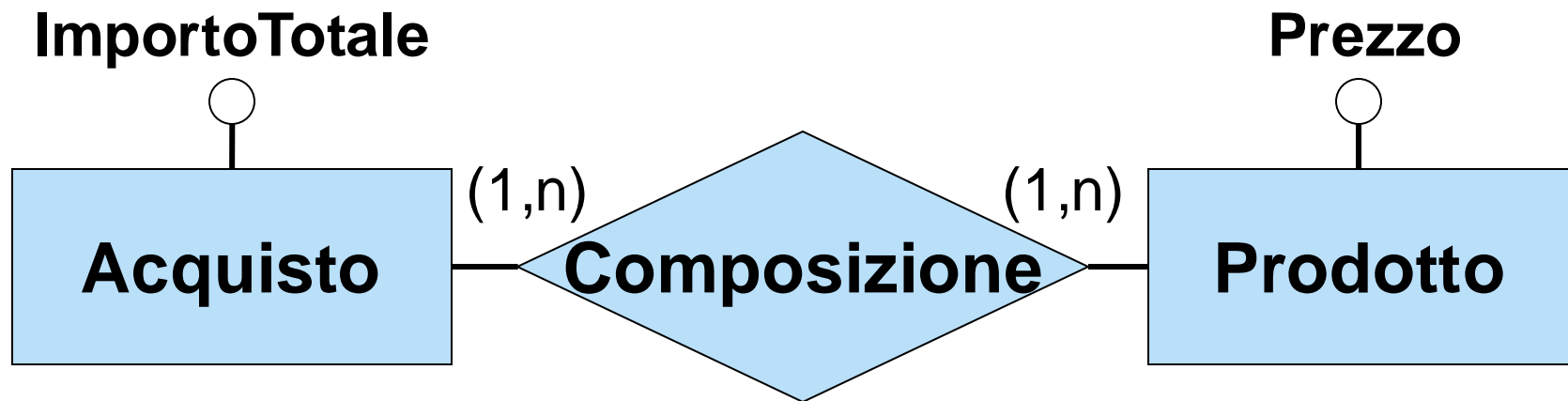
Vincolo esterno:

$$\text{StipendioNetto} = \text{StipendioLordo} - \text{Tasse}$$

4.2 La minimalità: ridondanze estensionali

Ulteriore esempio del primo tipo

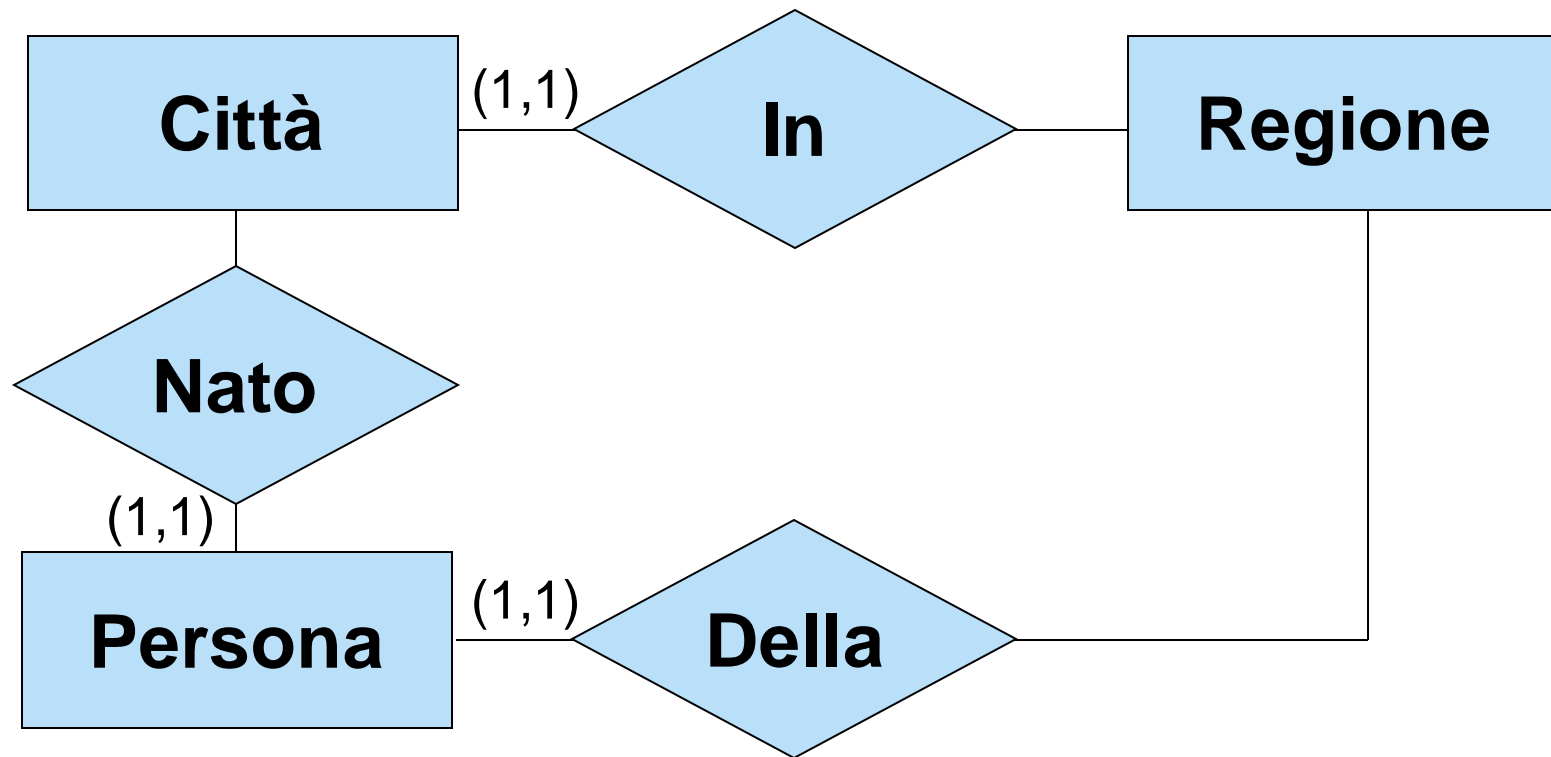
attributo calcolato sulla base degli attributi di un'altra entità



Vincolo esterno: per ogni istanza *a* di **Acquisto**, il valore di **ImportoTotale** è dato dalla somma dei valori di **Prezzo** per tutte le istanze di **Prodotto** connesse ad *a* tramite **Composizione**.

4.2 La minimalità: ridondanze estensionali

Esempio del secondo tipo

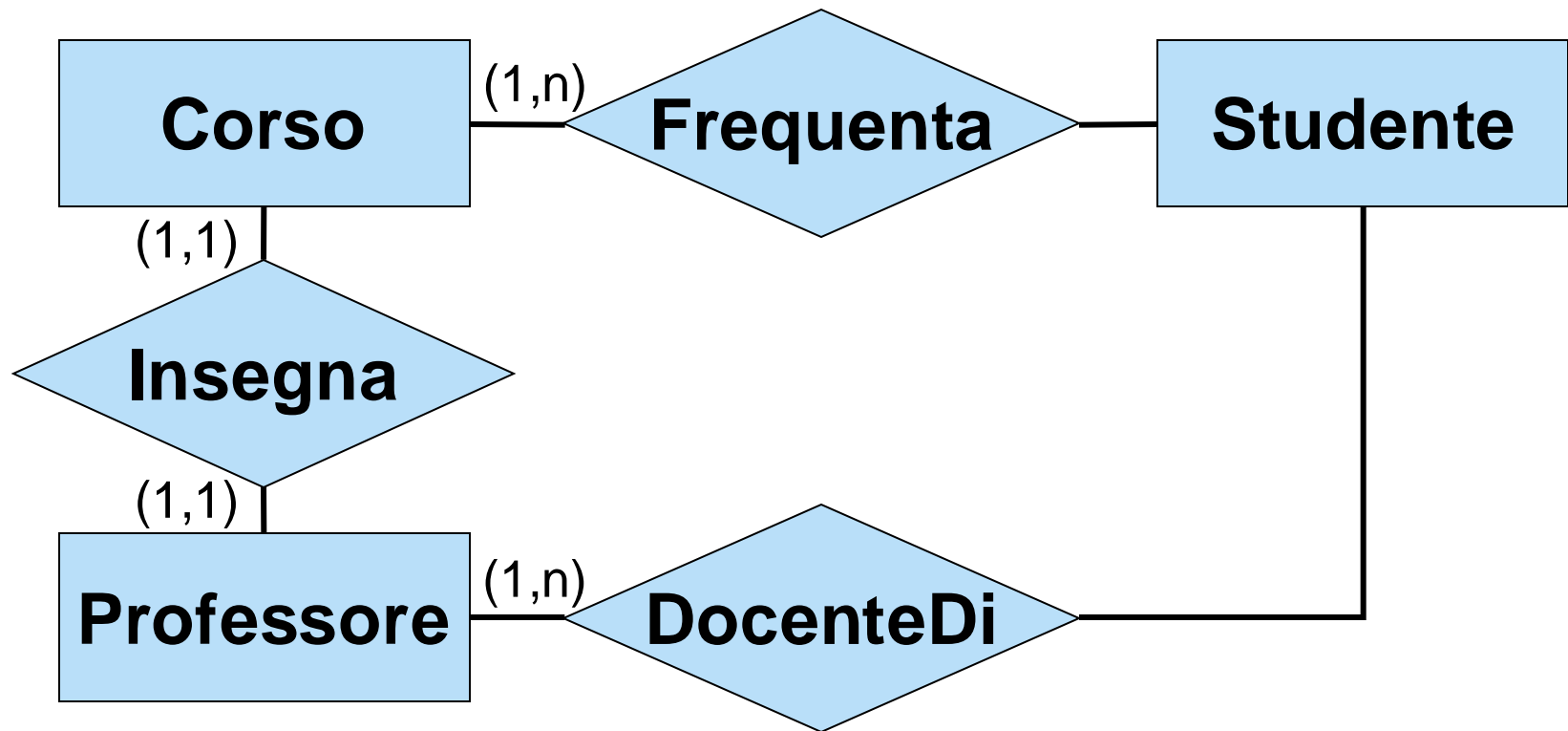


Vincolo esterno: per ogni istanza p di **Persona**, seguendo la relazione **Nato** e poi la relazione **In**, si ottiene la stessa istanza di **Regione** che si ottiene seguendo la relazione **Della** da p



4.2 La minimalità: ridondanze estensionali

Ulteriore esempio del secondo tipo



Vincolo esterno: per ogni istanza p di Professore, seguendo la relazione Insegna e poi la relazione Frequenta, si ottiene un'istanza di Studente che si ottiene anche seguendo la relazione DocenteDi da p .



4.3 La leggibilità

Oculata scelta dei concetti, dei loro nomi e delle strutture di rappresentazione scelte

Leggibilità del diagramma

- Produrre il diagramma secondo criteri “estetici naturali”, ad esempio cercando di produrre diagrammi che corrispondono a grafi planari, o comunque in cui il numero di incroci sia minimizzato

Leggibilità dei vincoli di integrità

- se possibile, scegliere lo schema concettuale che rappresenta un proprietà mediante vincoli espliciti e **non** mediante vincoli esterni

Produrre una buona documentazione (dizionario dei dati)



4. Controlli finali di qualità sullo schema ER

- È stata fatta una scelta oculata su come modellare i vari concetti?
 - se con attributi o con entità
 - se con entità o con relazioni
- Sono stati colti tutti gli aspetti importanti delle specifiche?
- Le generalizzazioni e le relazioni is-a sono corrette?
 - Non formano cicli?
 - Gli attributi sono associati alle entità giuste?
 - È corretto che le entità in una stessa generalizzazione siano disgiunte?
- Si possono applicare ulteriori generalizzazioni?



5. La progettazione concettuale

5.4 esempio

1. introduzione alla progettazione di basi di dati
2. modello Entità-Relazione
3. metodologia per la progettazione concettuale
- 4. esempio**



Base di dati corso di istruzione

Vogliamo memorizzare dati relativi ai partecipanti ad un corso. Per ciascuno, vogliamo ricordare nome, cognome, data di nascita, se è sposato, e, nel caso lo sia, il numero di figli.

Vogliamo, poi ricordare le città in cui risiedono e le città in cui sono nati, insieme al numero di abitanti. Per le città capoluogo di regione, vogliamo ricordare la regione.

Vogliamo poi sapere le lezioni che i partecipanti hanno frequentato, con i (o il), docenti che le hanno svolte (nome, cognome, e tipo di enti di provenienza), il corrispondente argomento ed il giorno in cui si sono svolte. Ad ogni lezione va associato un numero progressivo.

Relativamente ai docenti provenienti dall'Università, si vuole ricordare la Università da cui provengono e la materia che hanno li in affidamento.



Base di dati corso di istruzione

Vogliamo memorizzare dati relativi ai partecipanti ad un corso. Per ciascuno, vogliamo ricordare nome, cognome, data di nascita, se è sposato, e, nel caso lo sia, il numero di figli.

Vogliamo, poi ricordare le città in cui risiedono e le città in cui sono nati, insieme al numero di abitanti. Per le città capoluogo di regione, vogliamo ricordare la regione.

Vogliamo poi sapere le lezioni che i partecipanti hanno frequentato, con i (o il), docenti che le hanno svolte (nome, cognome, e tipo di enti di provenienza), il corrispondente argomento ed il giorno in cui si sono svolte. Ad ogni lezione va associato un numero progressivo.

Relativamente ai docenti provenienti dall'Università, si vuole ricordare la Università da cui provengono e la materia che hanno li in affidamento.



Base di dati corso di istruzione

Vogliamo memorizzare dati relativi ai partecipanti ad un corso. Per ciascuno, vogliamo ricordare nome, cognome, data di nascita, se è sposato, e, nel caso lo sia, il numero di figli.

Vogliamo, poi ricordare le città in cui risiedono e le città in cui sono nati, insieme al numero di abitanti. Per le città capoluogo di regione, vogliamo ricordare la regione.

Vogliamo poi sapere le lezioni che i partecipanti hanno frequentato, con i (o il), docenti che le hanno svolte (nome, cognome, e tipo di enti di provenienza), il corrispondente argomento ed il giorno in cui si sono svolte. Ad ogni lezione va associato un numero progressivo.

Relativamente ai docenti provenienti dall'Università, si vuole ricordare la Università da cui provengono e la materia che hanno li in affidamento.



Schema ER (1)

partecipante



Base di dati corso di istruzione

Vogliamo memorizzare dati relativi ai partecipanti ad un corso. Per ciascuno, vogliamo ricordare nome, cognome, data di nascita, se è sposato, e, nel caso lo sia, il numero di figli.

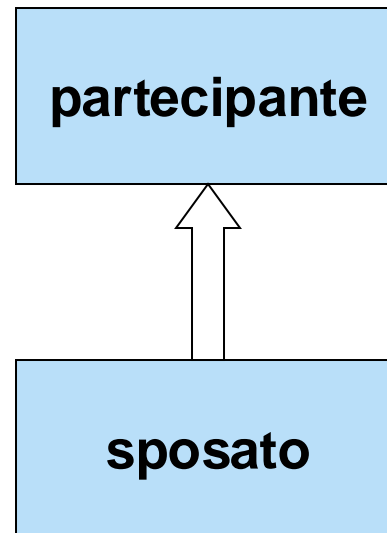
Vogliamo, poi ricordare le città in cui risiedono e le città in cui sono nati, insieme al numero di abitanti. Per le città capoluogo di regione, vogliamo ricordare la regione.

Vogliamo poi sapere le lezioni che i partecipanti hanno frequentato, con i (o il), docenti che le hanno svolte (nome, cognome, e tipo di enti di provenienza), il corrispondente argomento ed il giorno in cui si sono svolte. Ad ogni lezione va associato un numero progressivo.

Relativamente ai docenti provenienti dall'Università, si vuole ricordare la Università da cui provengono e la materia che hanno li in affidamento.



Schema ER (2)





Base di dati corso di istruzione

Vogliamo memorizzare dati relativi ai partecipanti ad un corso. Per ciascuno, vogliamo ricordare nome, cognome, data di nascita, se è sposato, e, nel caso lo sia, il numero di figli.

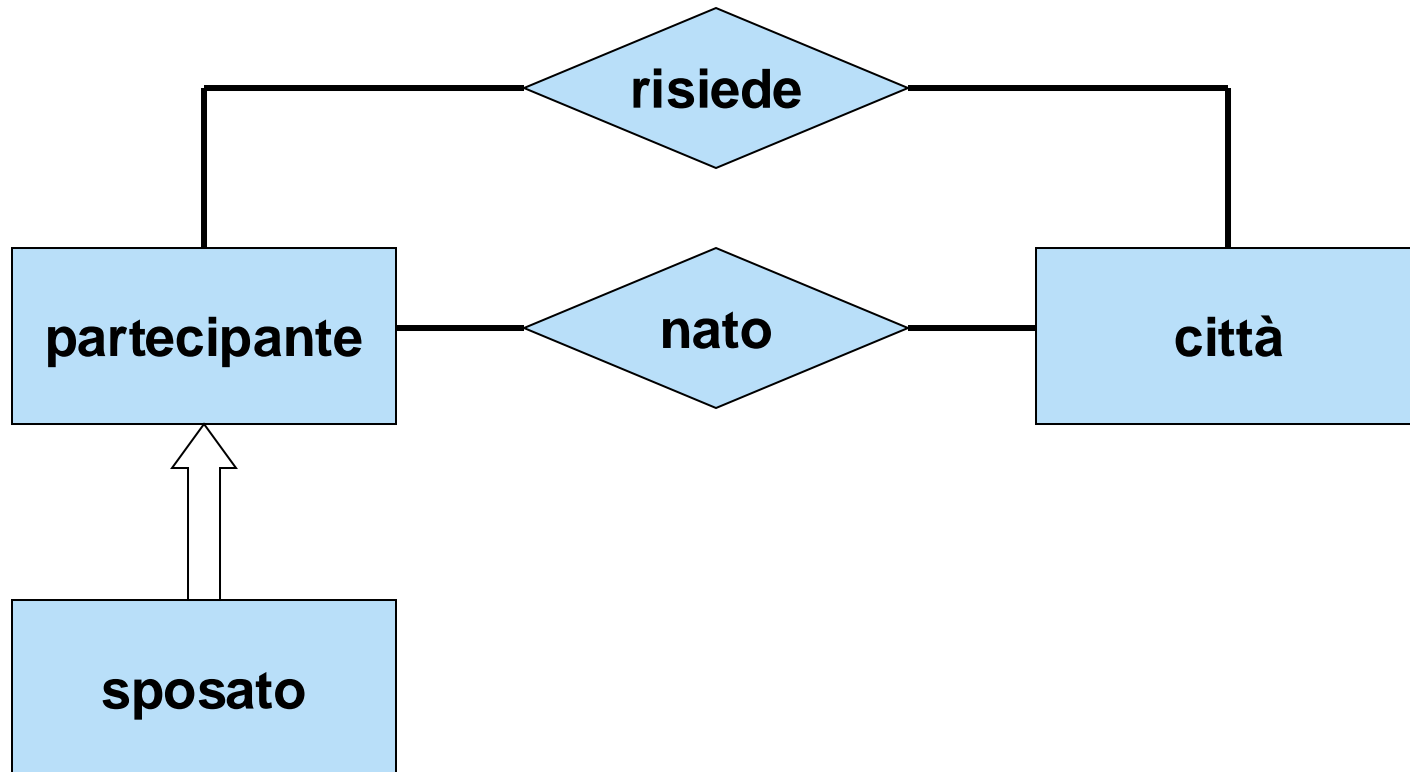
Vogliamo, poi ricordare le città in cui risiedono e le città in cui sono nati, insieme al numero di abitanti. Per le città capoluogo di regione, vogliamo ricordare la regione.

Vogliamo poi sapere le lezioni che i partecipanti hanno frequentato, con i (o il), docenti che le hanno svolte (nome, cognome, e tipo di enti di provenienza), il corrispondente argomento ed il giorno in cui si sono svolte. Ad ogni lezione va associato un numero progressivo.

Relativamente ai docenti provenienti dall'Università, si vuole ricordare la Università da cui provengono e la materia che hanno li in affidamento.



Schema ER (3)





Base di dati corso di istruzione

Vogliamo memorizzare dati relativi ai partecipanti ad un corso. Per ciascuno, vogliamo ricordare nome, cognome, data di nascita, se è sposato, e, nel caso lo sia, il numero di figli.

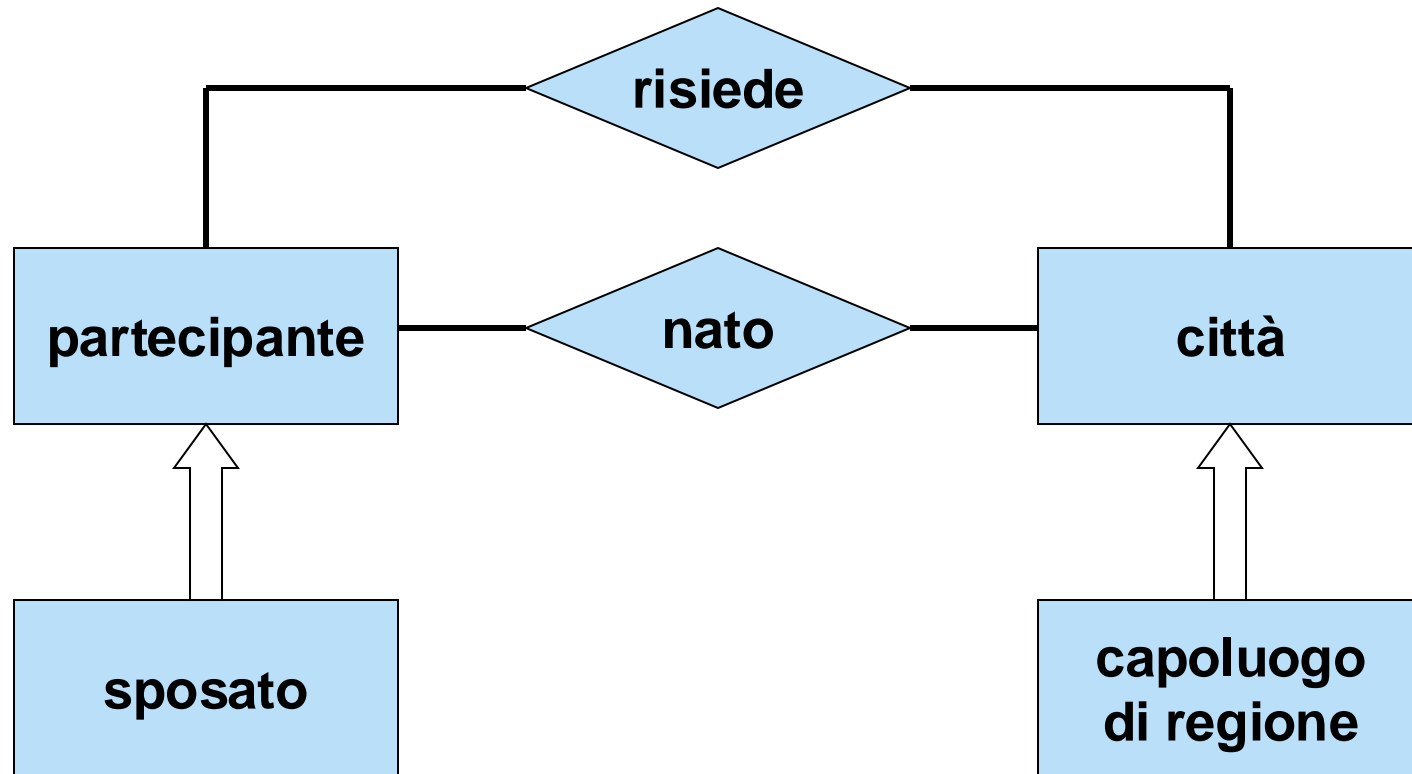
Vogliamo, poi ricordare le città in cui risiedono e le città in cui sono nati, insieme al numero di abitanti. Per le città capoluogo di regione, vogliamo ricordare la regione.

Vogliamo poi sapere le lezioni che i partecipanti hanno frequentato, con i (o il), docenti che le hanno svolte (nome, cognome, e tipo di enti di provenienza), il corrispondente argomento ed il giorno in cui si sono svolte. Ad ogni lezione va associato un numero progressivo.

Relativamente ai docenti provenienti dall'Università, si vuole ricordare la Università da cui provengono e la materia che hanno li in affidamento.



Schema ER (4)





Base di dati corso di istruzione

Vogliamo memorizzare dati relativi ai partecipanti ad un corso. Per ciascuno, vogliamo ricordare nome, cognome, data di nascita, se è sposato, e, nel caso lo sia, il numero di figli.

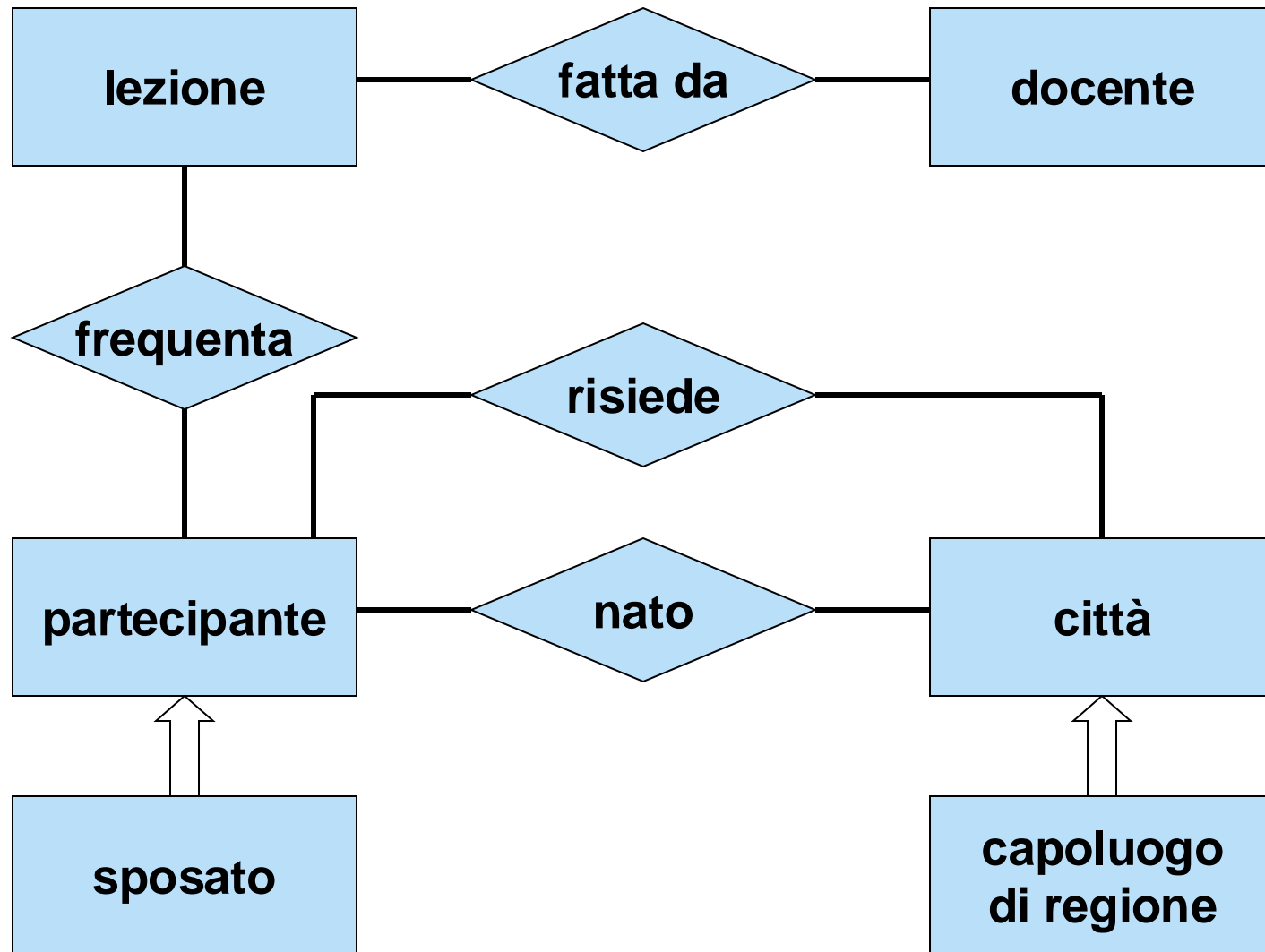
Vogliamo, poi ricordare le città in cui risiedono e le città in cui sono nati, insieme al numero di abitanti. Per le città capoluogo di regione, vogliamo ricordare la regione.

Vogliamo poi sapere le lezioni che i partecipanti hanno frequentato, con i (o il), docenti che le hanno svolte (nome, cognome, e tipo di enti di provenienza), il corrispondente argomento ed il giorno in cui si sono svolte. Ad ogni lezione va associato un numero progressivo.

Relativamente ai docenti provenienti dall'Università, si vuole ricordare la Università da cui provengono e la materia che hanno li in affidamento.



Schema ER (5)





Base di dati corso di istruzione

Vogliamo memorizzare dati relativi ai partecipanti ad un corso. Per ciascuno, vogliamo ricordare nome, cognome, data di nascita, se è sposato, e, nel caso lo sia, il numero di figli.

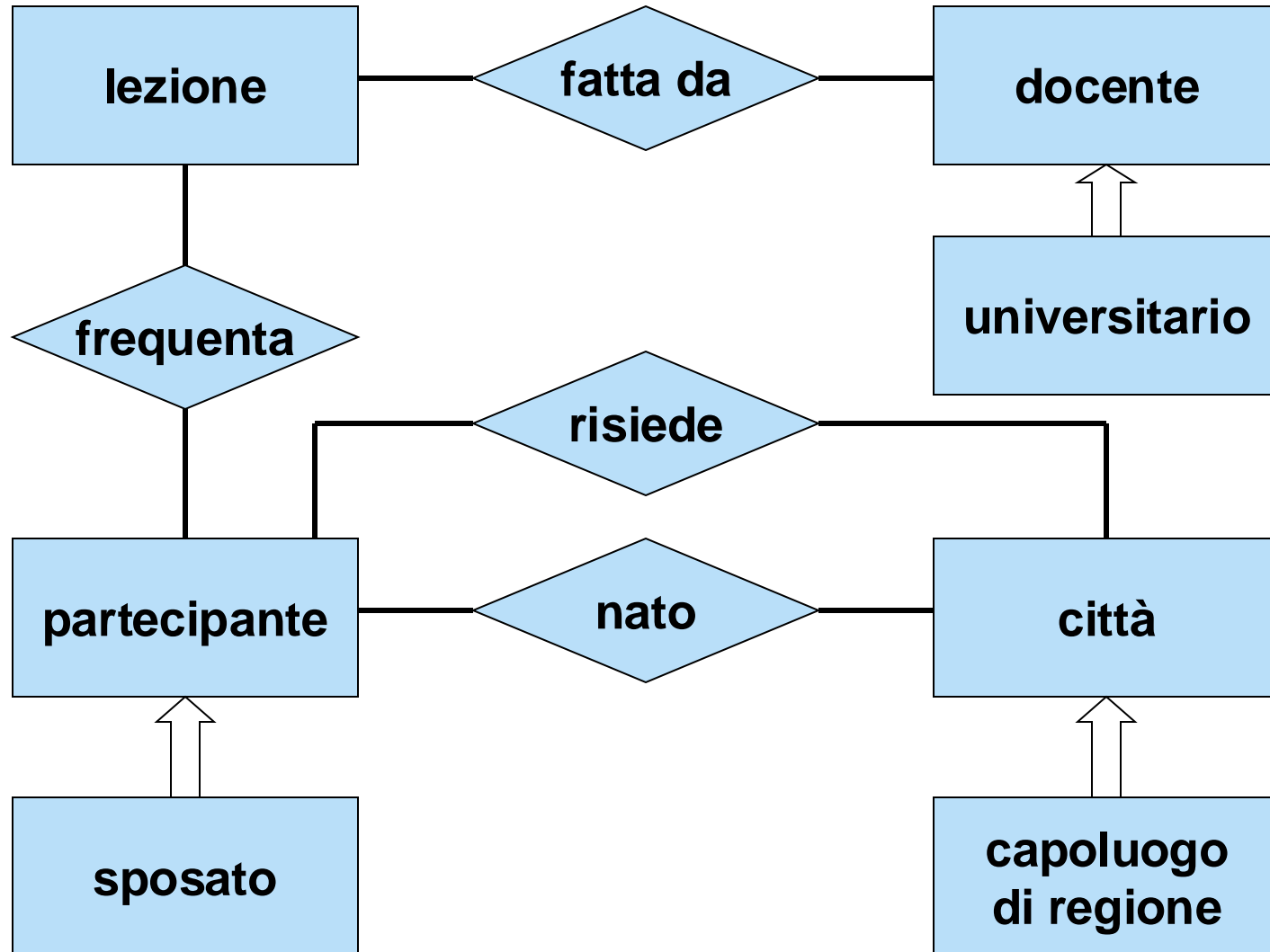
Vogliamo, poi ricordare le città in cui risiedono e le città in cui sono nati, insieme al numero di abitanti. Per le città capoluogo di regione, vogliamo ricordare la regione.

Vogliamo poi sapere le lezioni che i partecipanti hanno frequentato, con i (o il), docenti che le hanno svolte (nome, cognome, e tipo di enti di provenienza), il corrispondente argomento ed il giorno in cui si sono svolte. Ad ogni lezione va associato un numero progressivo.

Relativamente ai docenti provenienti dall'Università, si vuole ricordare la Università da cui provengono e la materia che hanno li in affidamento.



Schema ER (6)





Base di dati corso di istruzione

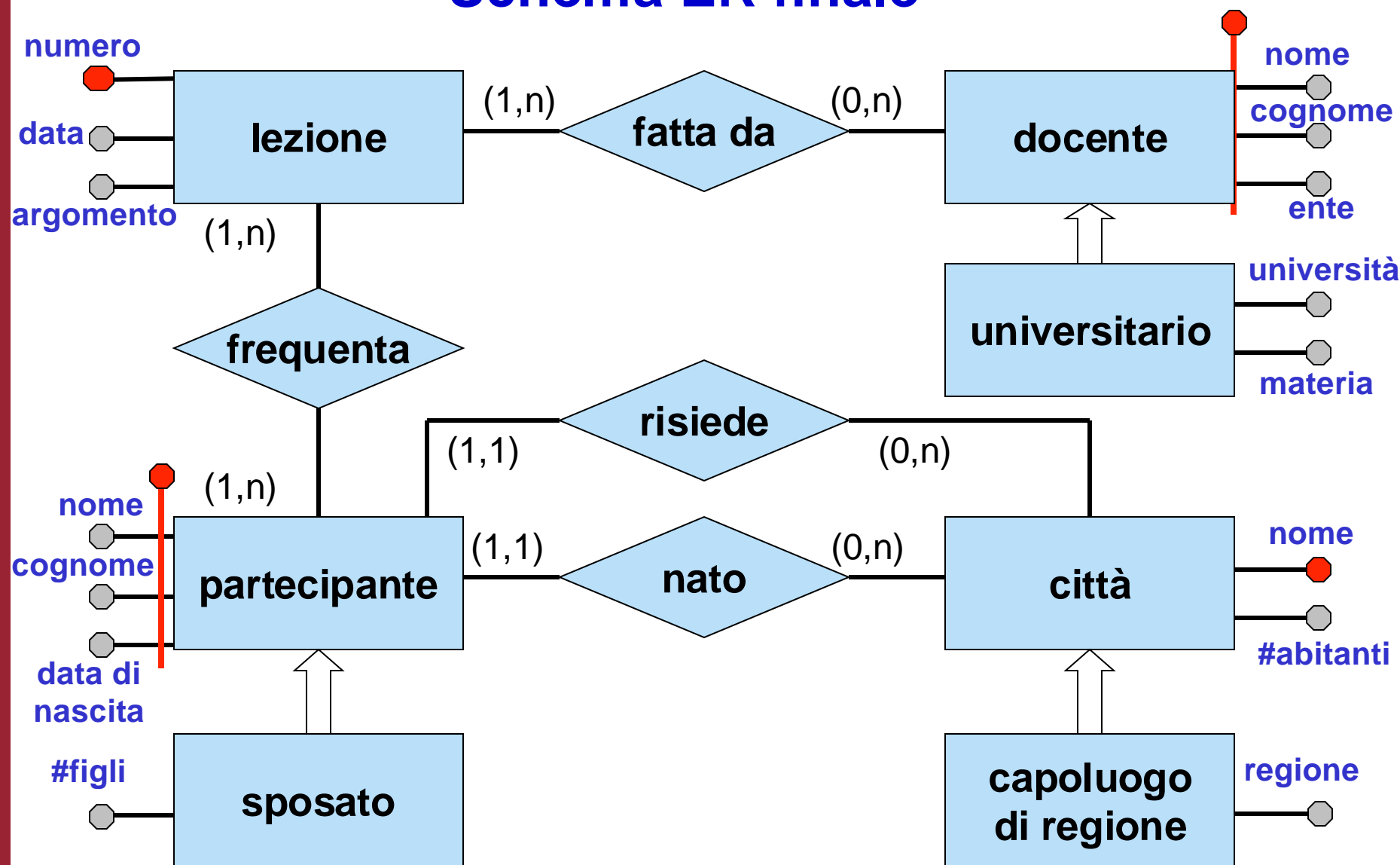
Vogliamo memorizzare dati relativi ai partecipanti ad un corso. Per ciascuno, vogliamo ricordare nome, cognome, data di nascita, se è sposato, e, nel caso lo sia, il numero di figli.

Vogliamo, poi ricordare le città in cui risiedono e le città in cui sono nati, insieme al numero di abitanti. Per le città capoluogo di regione, vogliamo ricordare la regione.

Vogliamo poi sapere le lezioni che i partecipanti hanno frequentato, con i (o il), docenti che le hanno svolte (nome, cognome, e tipo di enti di provenienza), il corrispondente argomento ed il giorno in cui si sono svolte. Ad ogni lezione va associato un numero progressivo.

Relativamente ai docenti provenienti dall'Università, si vuole ricordare la Università da cui provengono e la materia che hanno li in affidamento.

Schema ER finale





Vincoli esterni

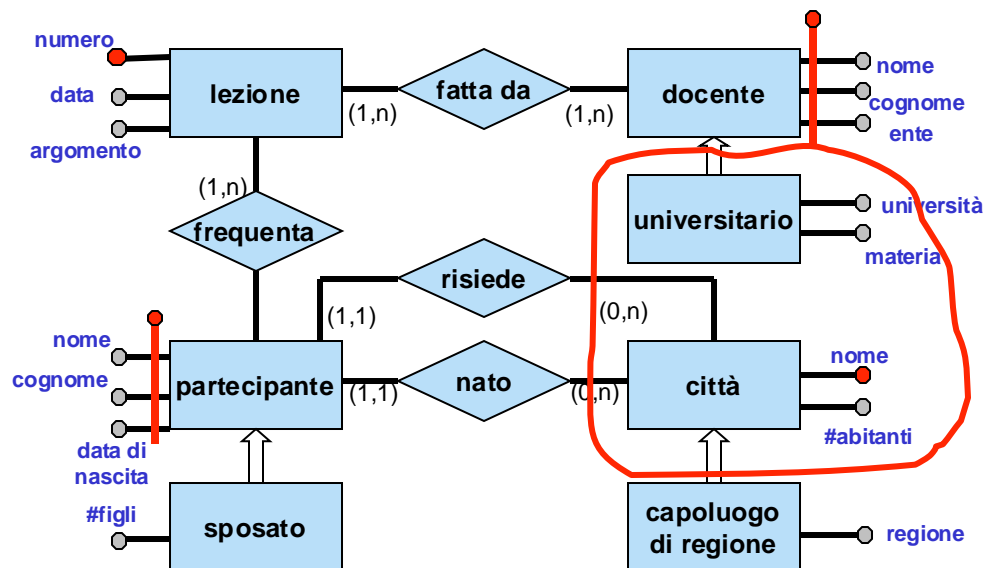
- Ogni istanza dell'entità città, o è la città in cui è nato un partecipante oppure è la città in cui risiede un partecipante.



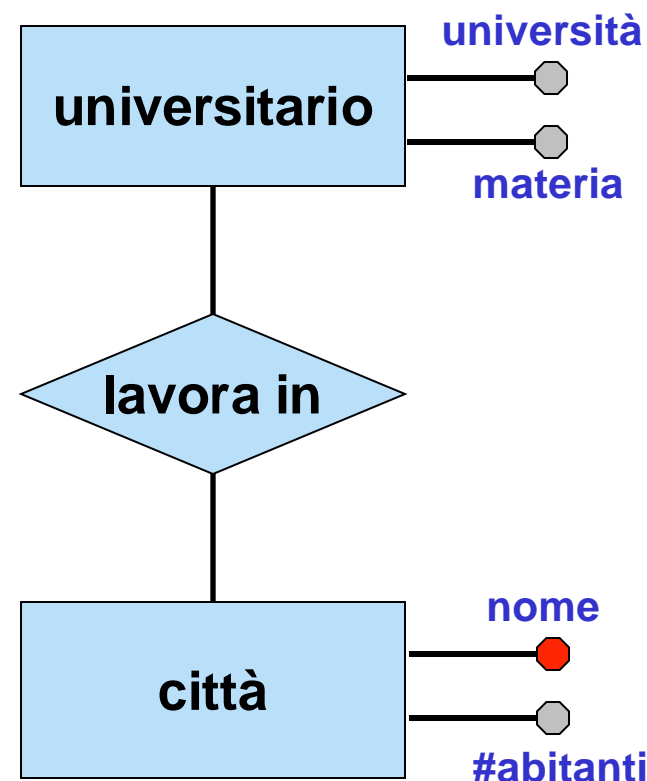
Esercizio 25: modifica dello schema 1

Per i docenti universitari vogliamo anche sapere la città in cui è ubicata l'Università in cui insegnano.

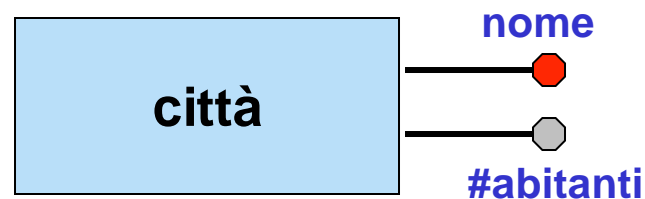
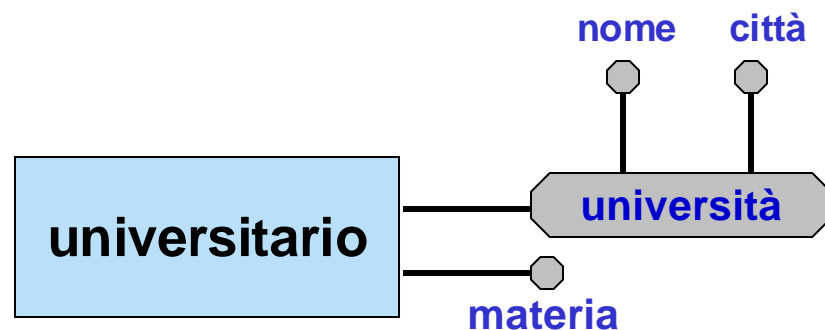
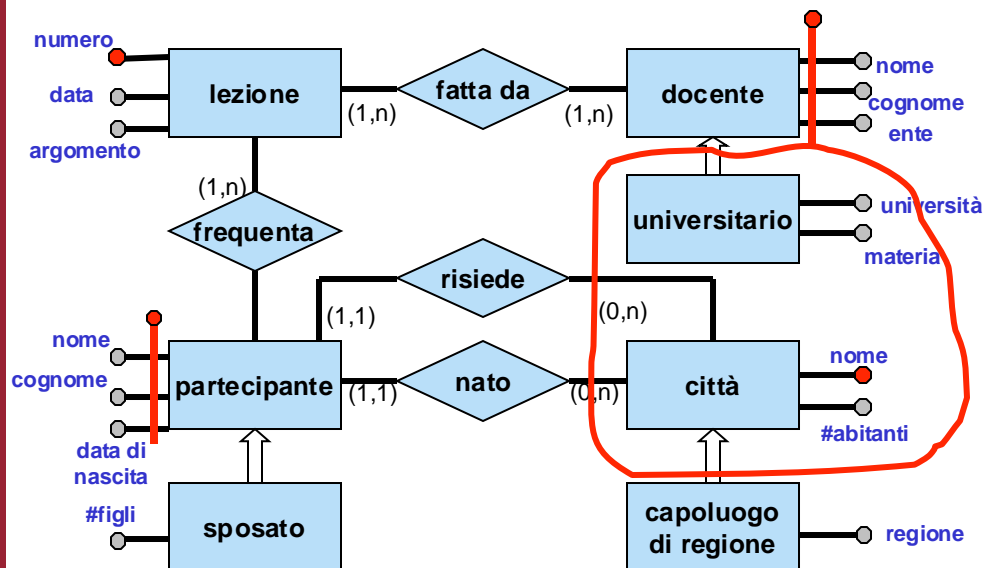
Esercizio 25: soluzione 1 [sbagliata!]



Per i docenti universitari vogliamo anche sapere la città in cui è ubicata l'Università in cui insegnano.

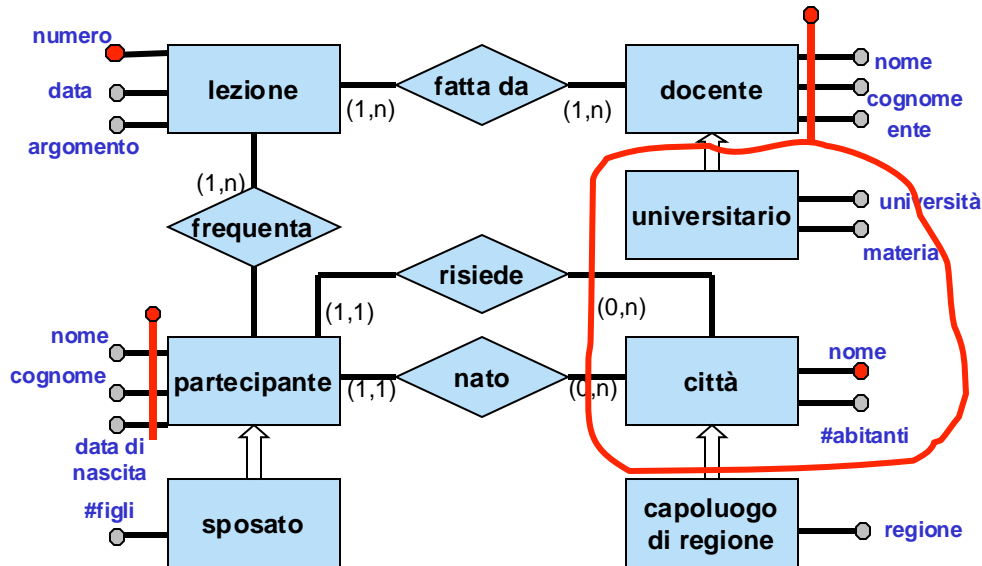


Esercizio 25: soluzione 2 [sbagliata!]

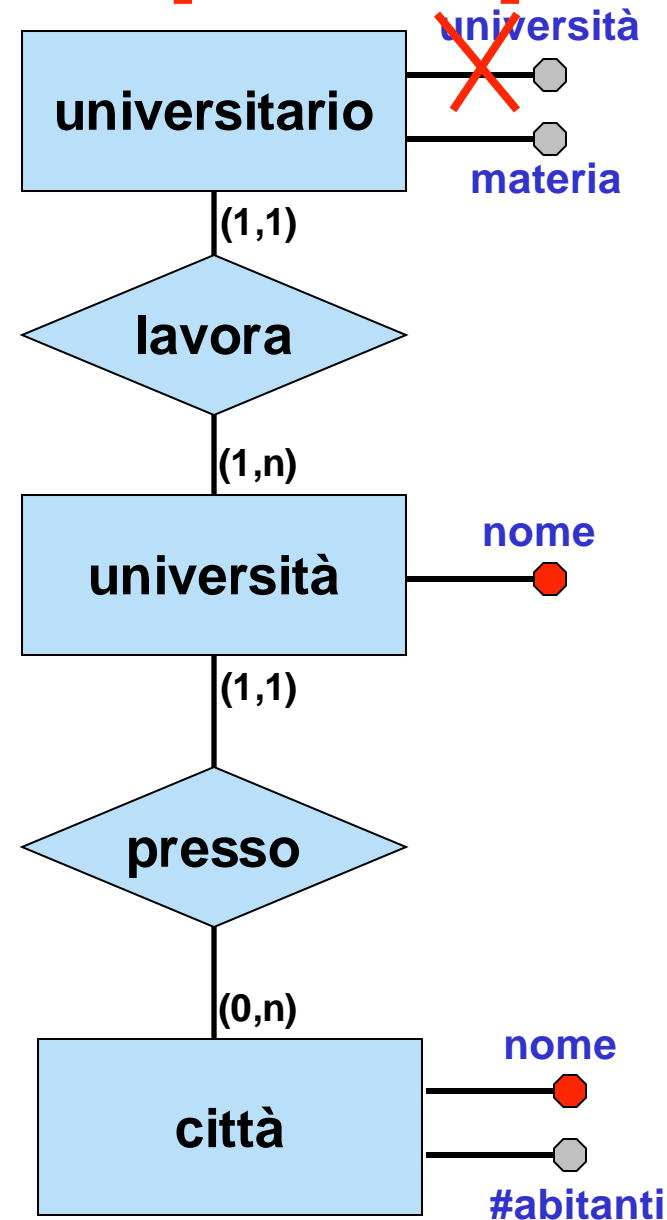


Per i docenti universitari vogliamo anche sapere la città in cui è ubicata l'Università in cui insegnano.

Esercizio 25: soluzione 3 [corretta!]



Per i docenti universitari vogliamo anche sapere la città in cui è ubicata l'Università in cui insegnano.





Esercizio 25: confronto tra le tre soluzioni

- La (2) non coglie il legame tra docente universitario e città (l'attributo città e l'entità città non sono correlate in alcun modo nel modello). **Si noti che questo è un errore!**
- La (1) e la (3) colgono il legame tra docente universitario e città.
- La (1) riferisce a docente universitario una proprietà (l'università) locale e non esplicita la dipendenza tra questa e la città.

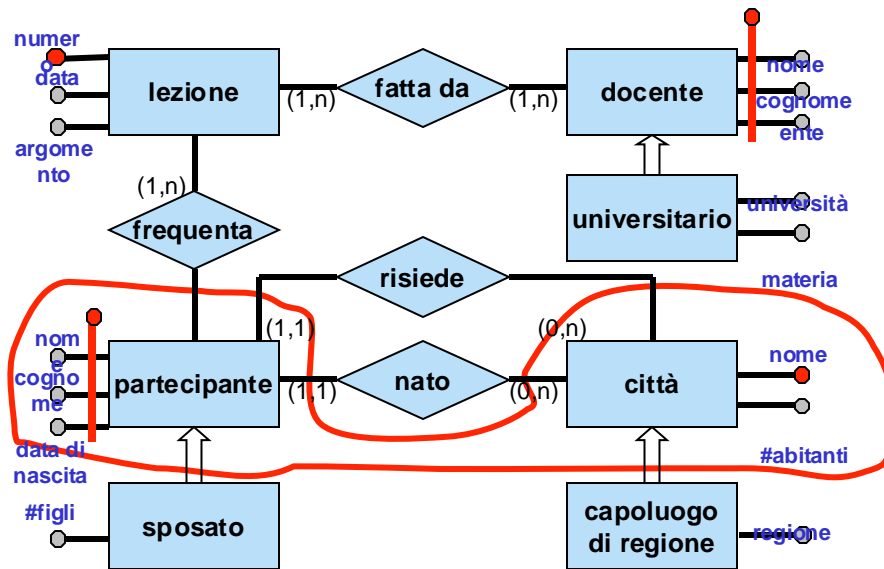
→ *La soluzione corretta è la (3).*



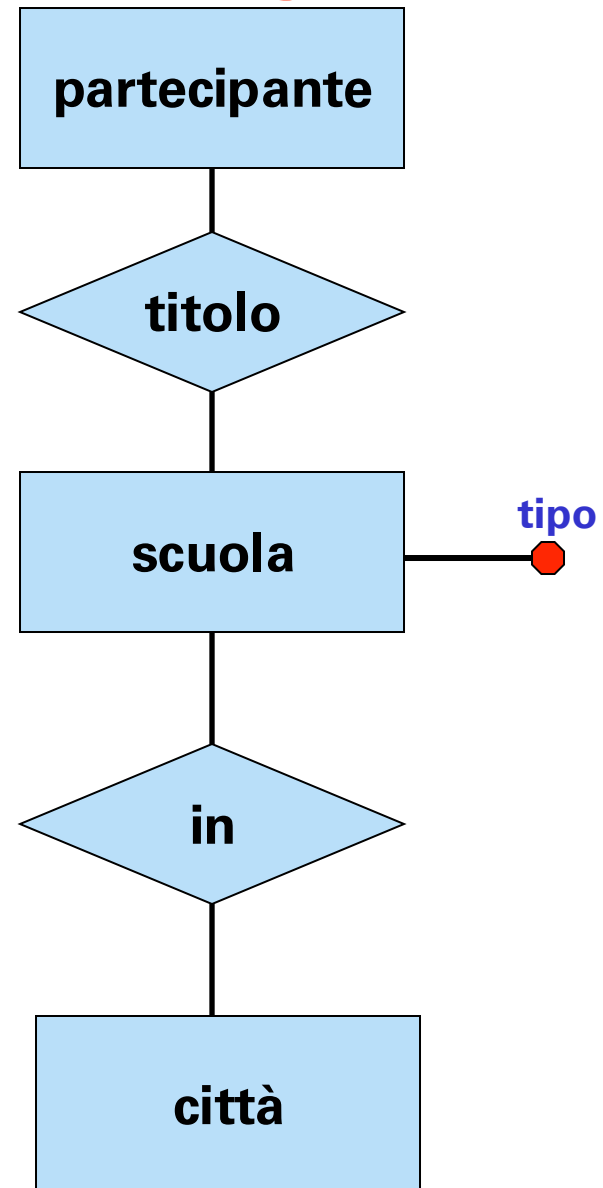
Esercizio 26: modifica dello schema 2

Si vuole anche sapere in quale tipo di scuola i partecipanti hanno ottenuto l'ultimo titolo di studio, e in quale città la scuola ha sede.

Esercizio 26: soluzione 1 [sbagliata!]



Si vuole anche sapere in quale tipo di scuola i partecipanti hanno ottenuto l'ultimo titolo di studio, e in quale città la scuola ha sede.



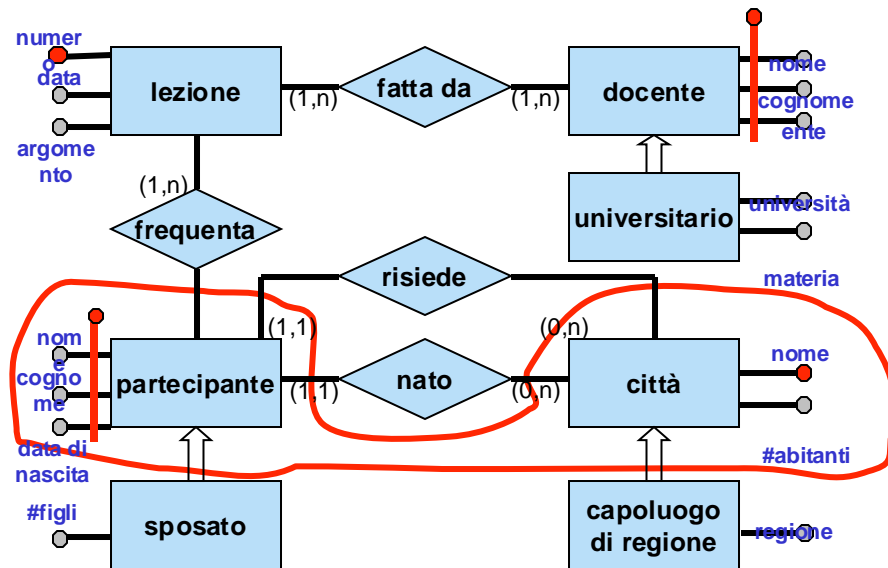


Esercizio 26: perché la soluzione è sbagliata?

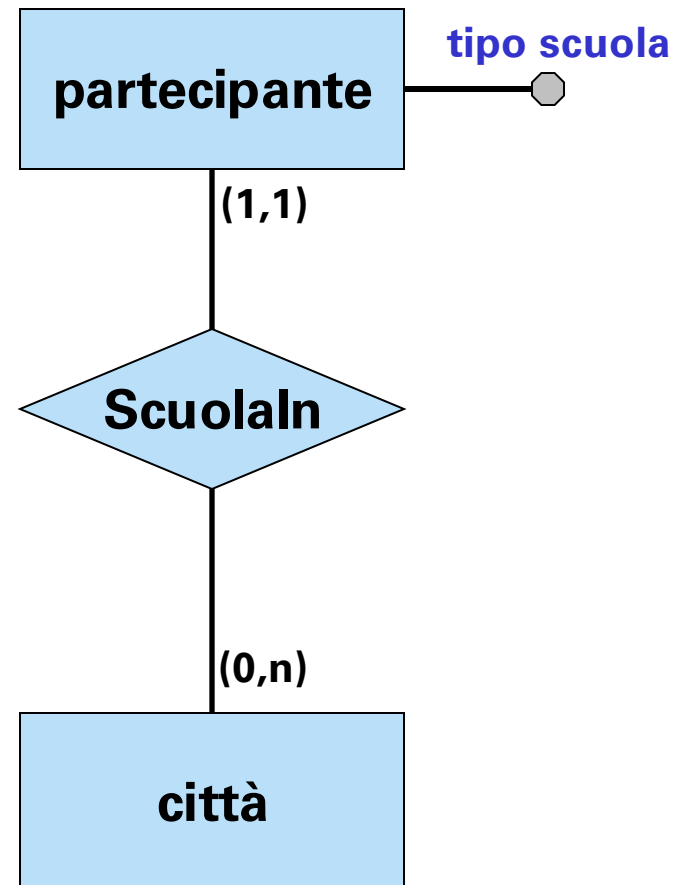
Perché noi vogliamo sapere il tipo di scuola frequentata dal partecipante e la città in cui la scuola frequentata è situata.

Nello schema invece viene rappresentato il tipo di scuola e le città in cui tale tipo di scuola è presente!

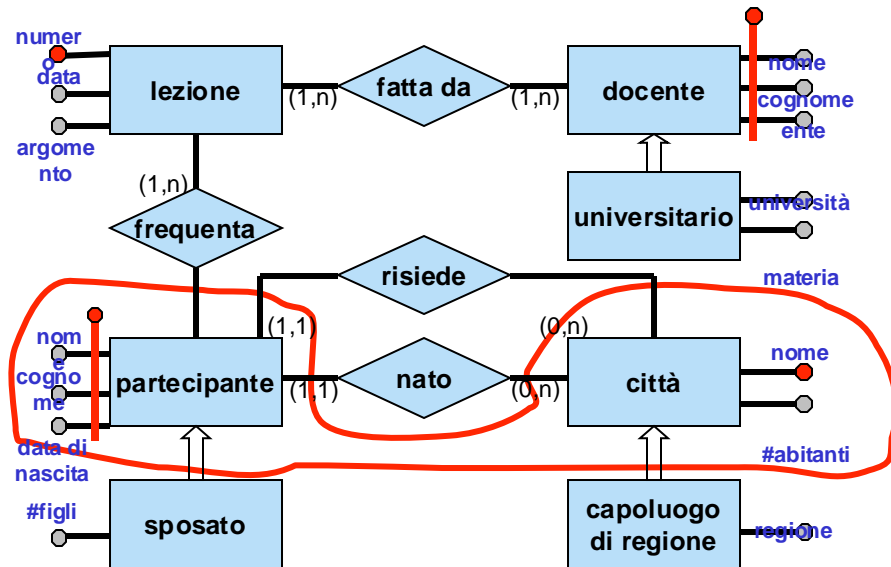
Esercizio 26: soluzione 2



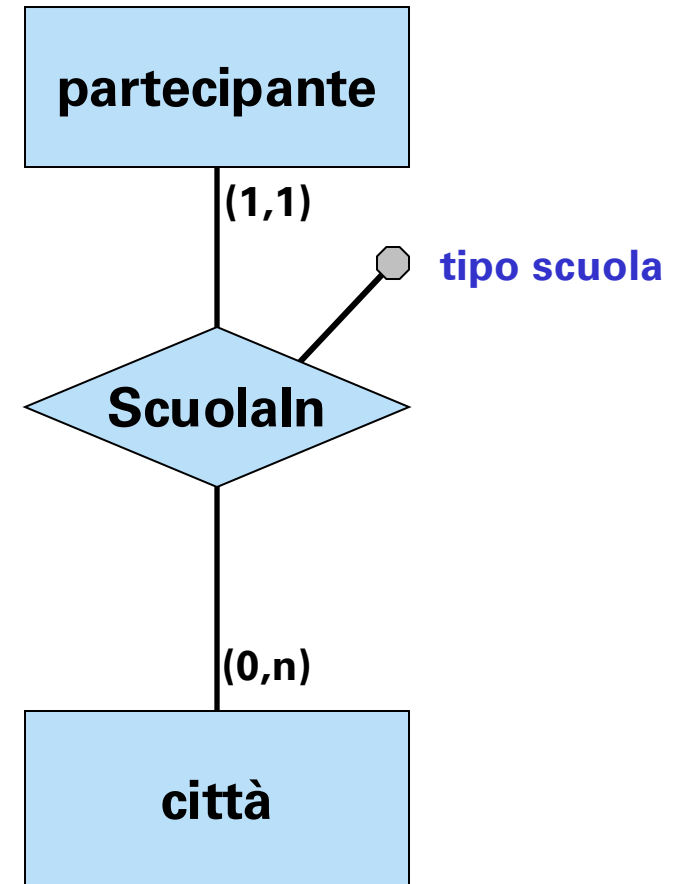
Si vuole anche sapere in quale tipo di scuola i partecipanti hanno ottenuto l'ultimo titolo di studio, e in quale città la scuola ha sede.



Esercizio 26: soluzione 3



Si vuole anche sapere in quale tipo di scuola i partecipanti hanno ottenuto l'ultimo titolo di studio, e in quale città la scuola ha sede.





Esercizio 26: confronto tra le tre soluzioni

1. La soluzione (1) non è corretta.
2. La soluzione (2) non coglie che TipoScuola e ScuolIn si riferiscono alla stessa scuola.
3. La soluzione (3) coglie questo legame ponendo TipoScuola come attributo della relazione ScuolIn.

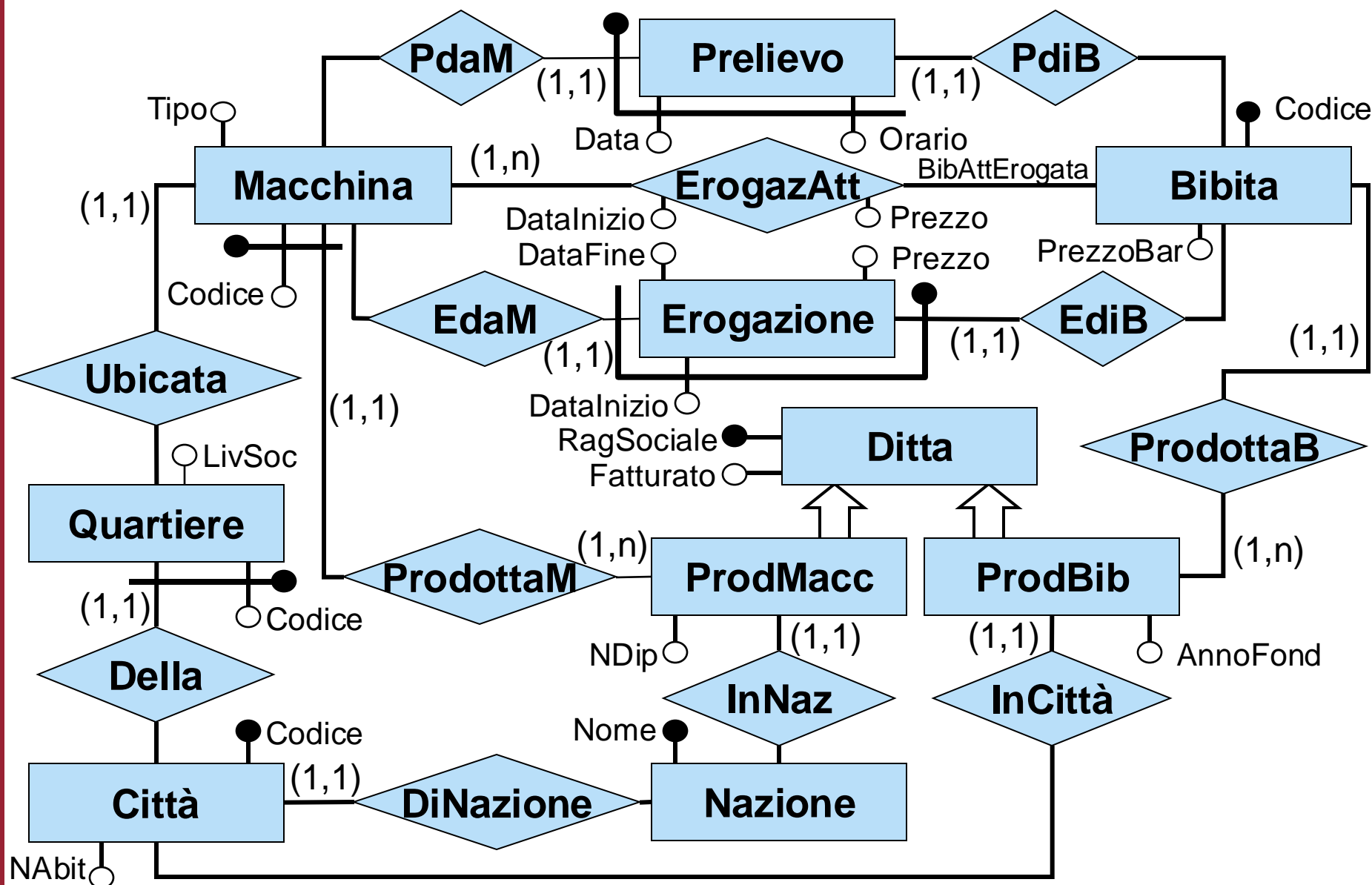
→ *La soluzione migliore è la (3).*



Esercizio 27: progettazione concettuale

Si richiede di progettare lo schema concettuale Entità-Relazione di un'applicazione relativa alle macchine che erogano bibite. Di ogni macchina erogatrice interessano il codice (unico fra quelle prodotte dalla stessa ditta produttrice), la ditta produttrice (ogni macchina erogatrice è prodotta da una ed una sola ditta), il tipo, e le bibite che la macchina eroga attualmente (ogni macchina erogatrice eroga attualmente almeno una bibita), con il relativo prezzo praticato, e la data in cui si è iniziato a praticare tale prezzo. Di ogni macchina interessano anche i prezzi eventualmente praticati precedentemente per le bibite attualmente erogate (con il relativo periodo specificato con data di inizio e data di fine), e le bibite che la macchina ha erogato e che non eroga più, sempre con i relativi prezzi nei vari periodi. Tutti i prezzi sono espressi in Euro. Ad esempio, la macchina erogatrice numero 5 della ditta "Pluto" eroga attualmente il Chinotto al prezzo di 35 Euro dal 1 marzo 2002, mentre lo erogava al prezzo di 30 Euro dal 1 gennaio 2001 al 28 febbraio 2002, e al prezzo di 25 Euro dal 1 gennaio del 1998 al 31 dicembre 2000. Inoltre, la stessa macchina erogatrice erogava il tea alla pesca, che adesso non eroga più, dal 1 gennaio 1995 al 31 dicembre del 1999 al prezzo di 25 Euro, e dal 1 aprile 2000 al 31 dicembre 2000 al prezzo di 28 Euro. Per ogni macchina erogatrice e per ogni bibita, interessa poi avere informazioni su ogni prelevamento effettuato dai clienti, con data e orario (espresso in ora, minuti e secondi) in cui è stato effettuato. Ogni prelevamento riguarda una ed una sola bibita, e si assume che le macchine erogatrici consentano al massimo un prelevamento al secondo. Di ogni macchina erogatrice interessa anche in quale quartiere di quale città è ubicata. Di ogni quartiere interessano il codice (unico all'interno della città) ed il livello sociale (numero intero positivo). Ogni città è identificata da un codice, e di ogni città interessano la nazione (della nazione interessa solo il nome) ed il numero di abitanti. Di ogni bibita interessano il codice (identificativo), il prezzo standard praticato al bar, e la ditta produttrice (ogni bibita è prodotta da una ed una sola ditta). Di ogni ditta produttrice di macchine erogatrici interessano la ragione sociale (identificativo), il fatturato, il numero di dipendenti, e la nazione in cui è situata la sede ufficiale (della nazione interessa solo il nome). Di ogni ditta produttrice di bibite interessano la ragione sociale (identificativo), il fatturato, l'anno di fondazione, e la città in cui è situata la sede ufficiale.

Esercizio 27: soluzione – schema ER



Esercizio 27: soluzione – vincoli esterni (1)

- Le date di inizio e di fine erogazione sono coerenti tra di loro:
per ogni $e \in \text{istanze}(\text{Erogazione})$: $\text{DataInizio}(e) \leq \text{DataFine}(e)$
- I periodi di erogazione di una stessa bibita da parte di una stessa macchina sono disgiunti a coppie:
per ogni $e_1, e_2 \in \text{istanze}(\text{Erogazione})$, $m \in \text{istanze}(\text{Macchina})$,
 $b \in \text{istanze}(\text{Bibita})$
se $(e_1, m), (e_2, m) \in \text{istanze}(\text{EdaM})$,
 $(e_1, b), (e_2, b) \in \text{istanze}(\text{EdiB})$, e
 $\text{DataInizio}(e_1) \leq \text{DataInizio}(e_2)$
allora $\text{DataFine}(e_1) \leq \text{DataInizio}(e_2)$

per ogni $e \in \text{istanze}(\text{Erogazione})$, $(m, b) \in \text{istanze}(\text{ErogazAtt})$,
se $(e, m) \in \text{istanze}(\text{EdaM})$ e $(e, b) \in \text{istanze}(\text{EdiB})$
allora $\text{DataFine}(e) \leq \text{DataInizio}((m, b))$

Esercizio 27: soluzione – vincoli esterni (2)

- I prelievi sono coerenti con le erogazioni:
per ogni $p \in \text{istanze}(\text{Prelievo})$, $m \in \text{istanze}(\text{Macchina})$,
 $b \in \text{istanze}(\text{Bibita})$
se $(p,m) \in \text{istanze}(\text{PdaM})$ e $(p,b) \in \text{istanze}(\text{PdiB})$
allora esiste $e \in \text{istanze}(\text{Erogazione})$
con $(e,m) \in \text{istanze}(\text{EdaM})$, $(e,b) \in \text{istanze}(\text{EdiB})$, e
 $\text{DataInizio}(e) \leq \text{Data}(p) \leq \text{DataFine}(e)$
oppure $(m,b) \in \text{istanze}(\text{ErogazAtt})$ e
 $\text{DataInizio}((m,b)) \leq \text{Data}(p)$
- Per le ditte che producono sia macchine che bibite, le informazioni su città e nazione della sede ufficiale sono coerenti:
per ogni $d \in \text{istanze}(\text{ProdMacc}) \cap \text{istanze}(\text{ProdBib})$
se $(d,c) \in \text{istanze}(\text{InCittà})$ e $(c,n) \in \text{istanze}(\text{DiNazione})$
allora $(d,n) \in \text{istanze}(\text{InNaz})$