

Elementi di Informatica LB

Basi di Dati

Anno accademico 2007/2008

Prof. Stefano Contadini

Elementi di Informatica LB

La progettazione concettuale

titolo:

1. Introduzione alla progettazione
delle basi di dati

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 base di dati, secondo le indicazioni raccolte durante la fase di raccolta dei requisiti

Due possibilità di inquadramento:

1. In questo scenario, il progetto di basi di dati viene svolto **nell'ambito** della più ampia attività di progetto del sistema informatico
 2. In questo scenario, il progetto di basi di dati viene svolto come un **processo a sé stante**
-

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**
-

Ciclo di vita di un'applicazione software

3. Progetto e realizzazione

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

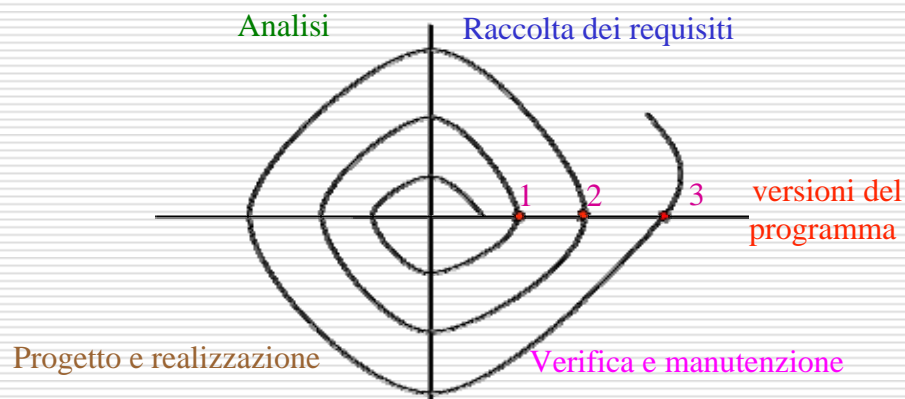
4. Verifica

- 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)



Scenario 1: progetto complessivo del sistema informatico

Studio di fattibilità e raccolta dei requisiti

Analisi

Parte di analisi relativa alla base di dati

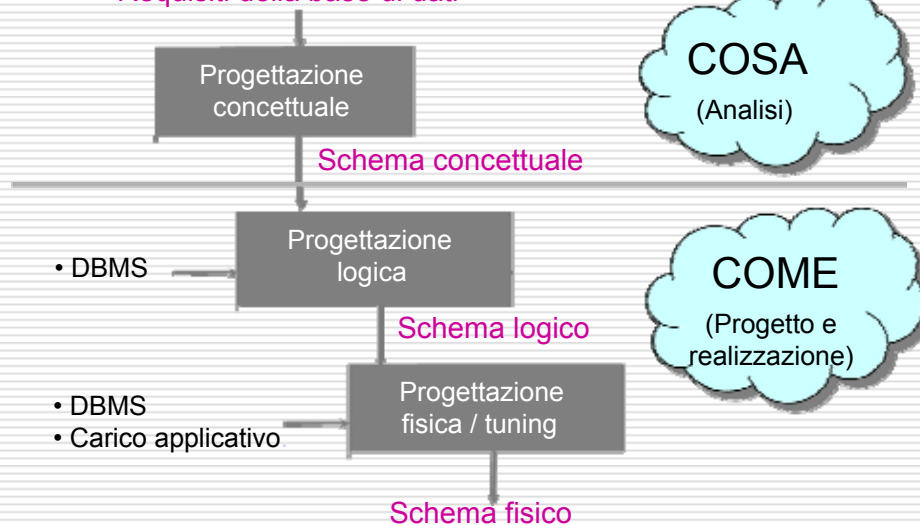
Progetto e realizzazione

Parte di progetto e realizzazione relativa alla base di dati

Verifica e manutenzione

Scenario 2: Progetto della basi di dati

Requisiti della base di dati



Confronto tra i due scenari

- In entrambi gli scenari, il processo di progettazione di basi di dati si compone delle seguenti fasi:
 1. Progettazione concettuale
 2. Progettazione logica
 3. Progettazione fisica / tuning
- Le differenze tra i due scenari sono:
 - Nel primo scenario la progettazione concettuale è parte della fase complessiva di analisi e si deve coordinare con gli altri aspetti, mentre nel secondo è una fase autonoma
 - Nel primo scenario le fasi di progettazione logica e progettazione fisica / tuning sono parti della fase complessiva di progetto e realizzazione, e si devono coordinare con gli altri aspetti, mentre nel secondo sono fasi autonome

Progettazione concettuale

- Rappresentare i dati della realtà d'interesse in termini di un modello (descrizione) formale, ad alto livello, indipendente dal DBMS
- Input:
 - Descrizione della realtà d'interesse: (per sapere quali dati rappresentare)
- Output:
 - Schema concettuale (rappresentazione dei dati in un modello di dati concettuale)
- Qualità:
 - correttezza e completezza della rappresentazione

Progettazione logica

- Rappresentare i dati della realtà d'interesse in termini dei **costrutti logici di una classe di DBMS** (quelli basati su un determinato modello logico)
- **Input:**
 - Schema concettuale (prodotto dalla progettazione concettuale) - DBMS scelto (per sapere quale modello logico di dati adottare)
- **Output:**
 - **Schema logico** (rappresentazione dei dati in un modello di dati logico)
- **Qualità:**
 - Correttezza (e possibilmente completezza) della rappresentazione - Efficienza della manipolazione dei dati

Progettazione fisica / tuning

- Rappresentare i dati della realtà di interesse attraverso le **strutture dati di uno specifico DBMS**
- **Input:**
 - Schema logico (prodotto della progettazione logica) - DBMS scelto (per sapere quali strutture dati fisiche utilizzare) - Previsioni del carico applicativo (per ottimizzare la base di dati rispetto ad esso)
- **Output:**
 - **Schema fisico** (Tipicamente nei DBMS attuali consiste in uno schema logico ottimizzato + alcune raffinamenti relativi all'effettiva implementazione su DBMS di detto schema)
- **Qualità:**
 - Efficienza rispetto al carico applicativo

Compromesso tra correttezza/completezza ed efficienza

- Si noti che passando dallo schema concettuale allo schema logico e poi allo schema fisico, stiamo **degradando la correttezza/completezza** (qualità semantiche).
 - D'altra parte in questi passaggi poniamo sempre **maggiore attenzione all'efficienza** (qualità computazionali).
-

Correttezza e completezza del sistema

- Si noti che la correttezza e completezza rispetto alle specifiche del sistema **basi di dati + applicazioni** è **irrinunciabile**.
 - Quindi, se la nostra rappresentazione dei dati **non è completa** (come è tipicamente il caso per lo schema logico e lo schema fisico) **oppure non garantisce la correttezza in tutti i casi** (come può accadere per lo schema fisico), allora saranno le **applicazioni** a garantire di manipolare i dati in modo da preservare correttezza e completezza.
-

Modelli di dati

I prodotti della varie fasi sono schemi basati su specifici **modelli di dati**:

- Schema concettuale → **modello concettuale**
- Schema logico → **modello logico (relazionale)**
- Schema fisico → **modello logico + parametri fisici**

Modelli dei dati adottati

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

La progettazione concettuale

titolo:

2. Modello Entità - Relazione

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

- Un **modello (concettuale) di 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** (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).
-

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
- hanno esistenza autonoma, e
- hanno proprietà comuni.

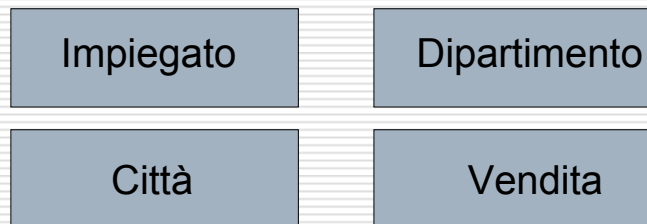
Esempi:

‡ impiegato	‡ vendita
‡ dipartimento	‡ ordine
‡ città	‡ studente

Sintassi: rappresentazione grafica di entità

- Ogni entità ha un **nome** che la identifica in modo univoco nello schema.
- È **rappresentata da un rettangolo** nel diagramma che descrive lo schema stesso.

Esempi:



Semantica delle entità

- A livello **estensionale** un'entità è costituita da un insieme di oggetti, che sono chiamati le sue **istanze**.
- 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**) o semplicemente istanze(**E**))
 $\{e_1, e_2, e_3, \dots\}$
che viene detto anche l'**estensione** di **E** nella istanza **I** di **S**.
- Una istanza di entità non è un valore che identifica un oggetto, ma è l'oggetto stesso.

Entità e istanze d'entità

- Una entità (nello schema concettuale S) denota un insieme (o classe) di oggetti "simili".
- Una istanza dell'entità (in una istanza di S) denota un oggetto dell'insieme rappresentato dall'entità.

Nota: nello schema concettuale rappresentiamo le entità, non le singole istanze ("astrazione").

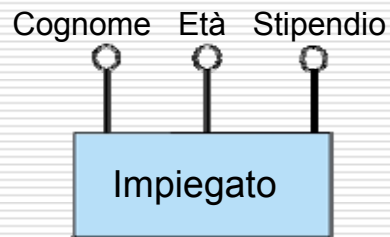
La nozione di attributo di entità

- Un attributo di entità è una **proprietà locale** di un'entità, di interesse ai fini dell'applicazione
- Un attributo associa ad ogni istanza di entità un valore appartenente ad un insieme 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.
- È rappresentato da un **cerchio collegato alla entità** a cui appartiene.

Esempi:

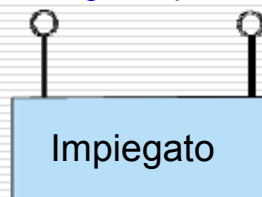


Sintassi: rappresentazione grafica di dominio

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

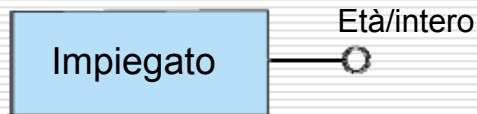
Se I è una istanza di uno schema S dove A è un attributo dell'entità E su un dominio D , $\text{istanze}(I,A)$, o semplicemente $\text{istanze}(A)$, è un insieme di coppie (x,y) tali che x è in $\text{istanze}(E)$, y è in D , ed esiste una ed una sola coppia in $\text{istanze}(A)$ per ogni x in $\text{istanze}(E)$.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

$$A : \text{istanze}(E) \rightarrow D$$

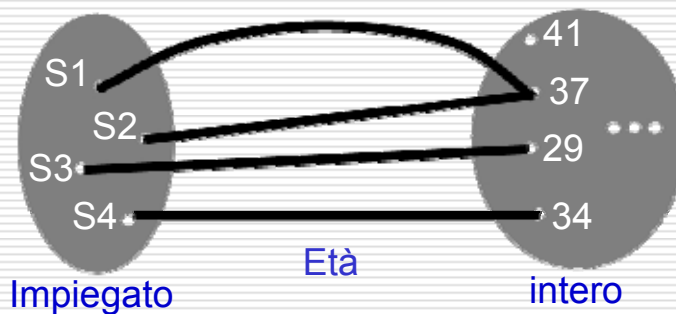
cioè una funzione che associa ad **ogni** istanza di E un valore appartenente a D .

Esercizio 1: attributi di entità

Dato lo schema:

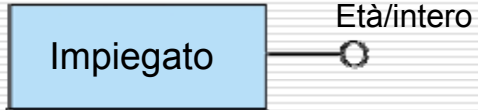


La seguente è una sua istanza ?

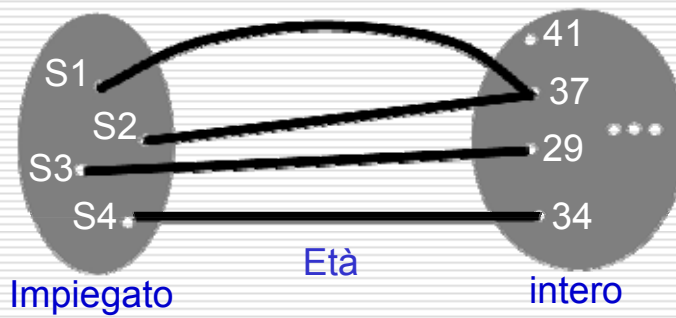


Esercizio 1: soluzione

Dato lo schema:

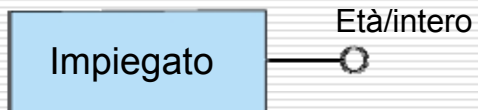


La seguente è una sua istanza ? **SI!**

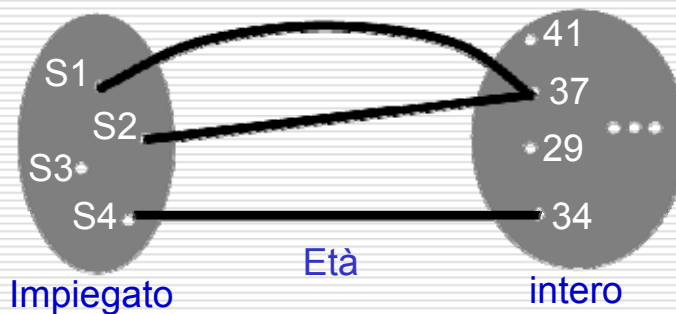


Esercizio 2: attributi di entità

Dato lo schema:

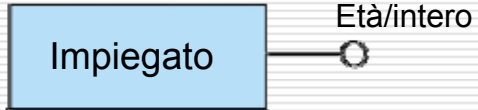


La seguente è una sua istanza ?



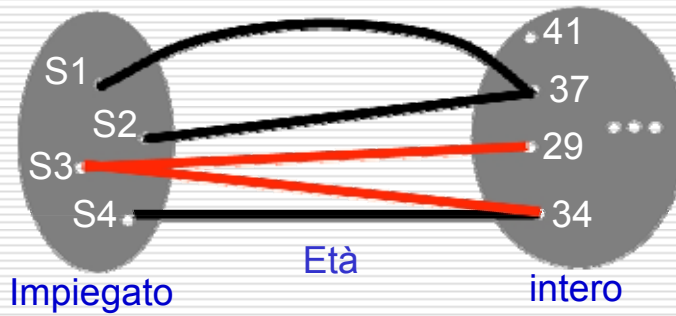
Esercizio 2: soluzione

Dato lo schema:



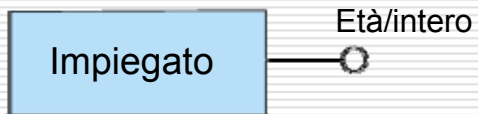
La seguente è una sua istanza ?

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

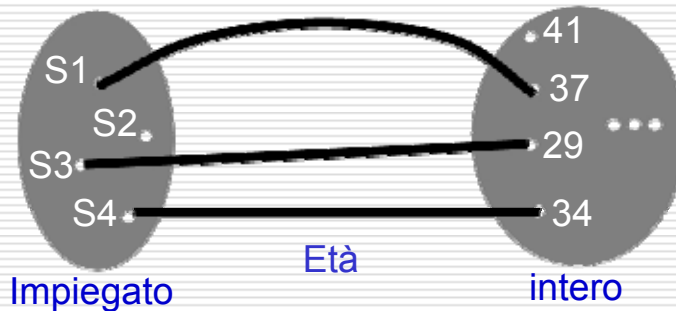


Esercizio 3: attributi di entità

Dato lo schema:

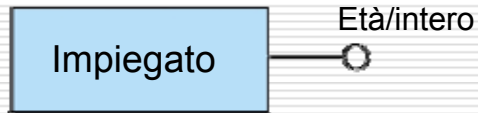


La seguente è una sua istanza ?



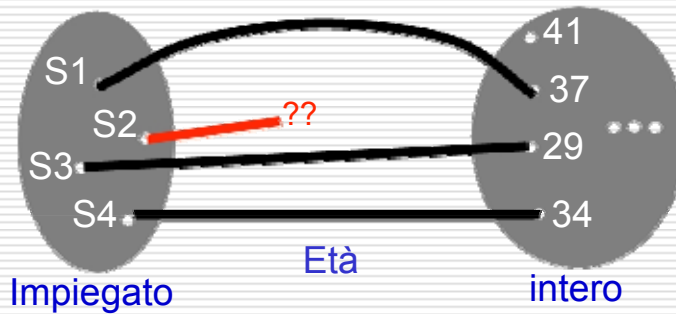
Esercizio 3: soluzione

Dato lo schema:



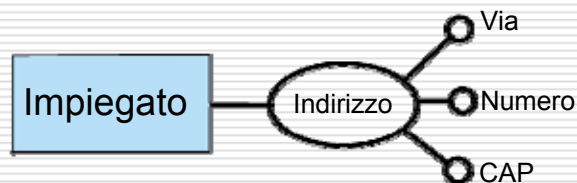
La seguente è una sua istanza ?

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:



“Indirizzo” è un attributo composto, e il suo dominio è un 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**.

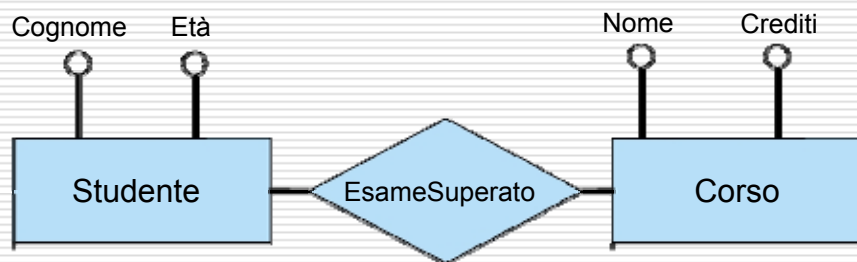
Esempi:

- ‡ Residenza (tra Persona e Città)
- ‡ Afferenza (tra Impiegato e Dipartimento)

Sintassi: rappresentazione grafica di relazione

- Ogni relazione ha **nome** che la identifica in modo univoco nello schema.
- 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):

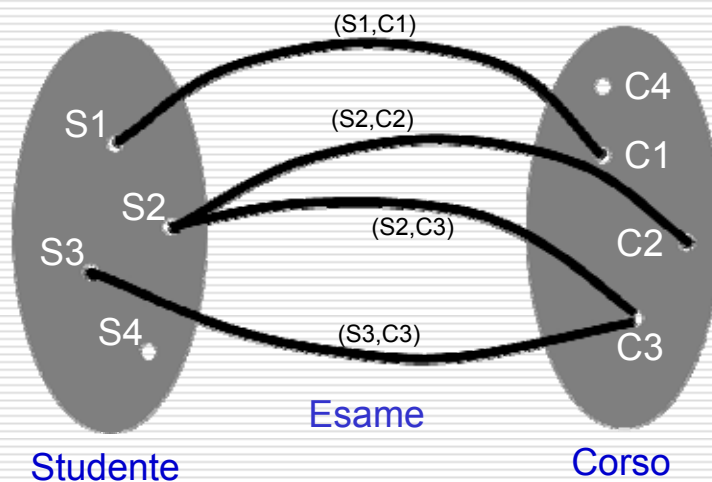


Semantica delle relazioni binarie (prima versione)

- A livello **estensionale** una relazione **R** tra le entità **E** ed **F** è costituita da un insieme di coppie (x,y) , tali che **x** è una istanza di **E**, ed **y** è una istanza di **F**. Ogni coppia è detta **istanza** della relazione **R**
- Ciò significa che, 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 (denotato da $\text{istanze}(I,R)$ o semplicemente $\text{istanze}(R)$)
 $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots\}$
che viene detto anche l'**estensione** di **R** nella istanza **I** di **S**
- In altre parole, una relazione nel modello ER è, dal punto di vista della semantica, una **relazione matematica**. In ogni istanza dello schema si ha:

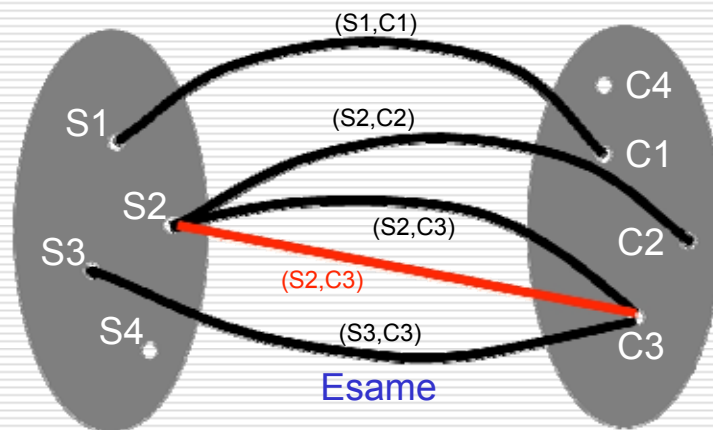
$$\text{istanze}(R) \subseteq \text{istanze}(E) \times \text{istanze}(F)$$

Relazione a livello estensionale



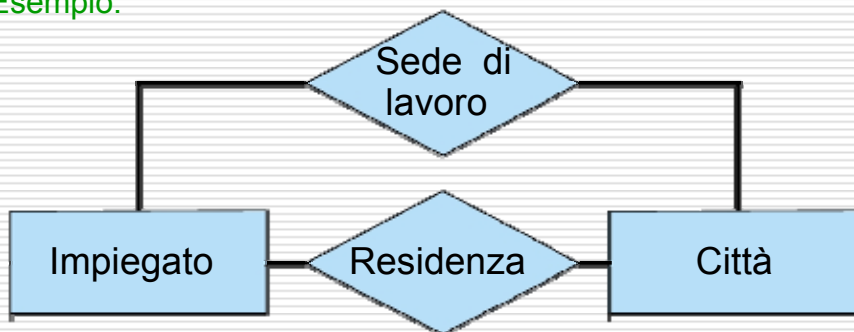
Osservazione importante

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



Due entità possono essere coinvolte in più relazioni

Esempio:

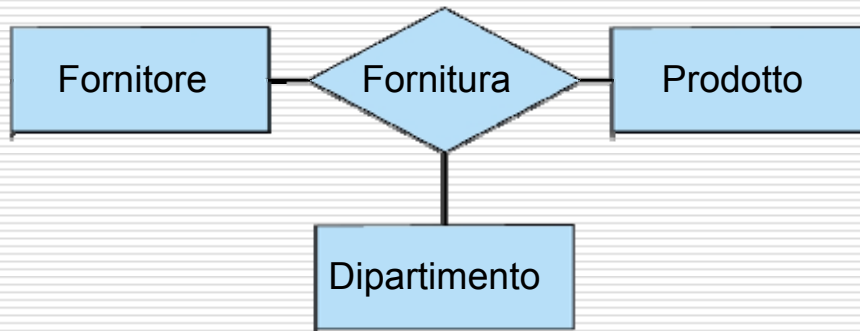


“Sede di Lavoro” e “Residenza” rappresentano due legami diversi tra le stesse entità “Impiegato” e “Città”.

Relazioni n-arie (di grado maggiore di 2)

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

Esempio:

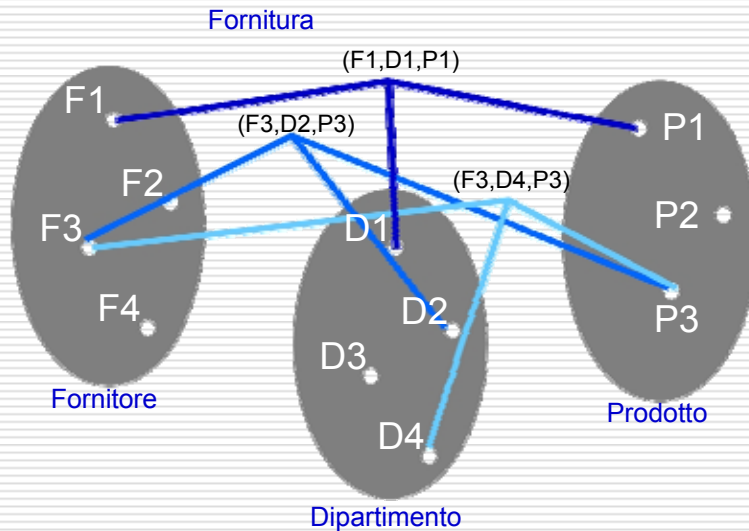


Semantica delle relazioni n-arie (prima versione)

- A livello **estensionale** una relazione **R** tra le entità E_1, E_2, \dots, E_n è costituita da un insieme di n-ple (o tuple) (x_1, x_2, \dots, x_n) , tali che x_1 è una istanza di E_1 , x_2 è una istanza di E_2, \dots, x_n è una istanza di E_n . Ogni n-ple è detta **istanza** della relazione **R**.
- Quindi, in ogni istanza dello schema si ha:

$$\text{istanze}(\mathbf{R}) \subseteq \text{istanze}(\mathbf{E}_1) \times \dots \times \text{istanze}(\mathbf{E}_n)$$

Relazione n-aria a livello estensionale



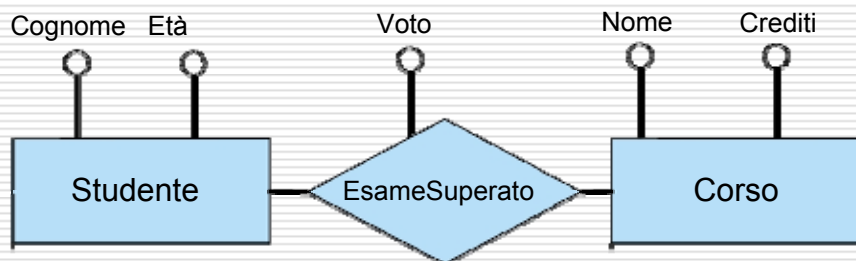
La nozione di attributo di relazione

- 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 .
- Un attributo associa ad ogni istanza di relazione un valore appartenente ad un insieme detto **dominio** dell'attributo.

Sintassi: rappresentazione grafica di attributo di relazione

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

Esempio:



Semantica degli attributi di relazione

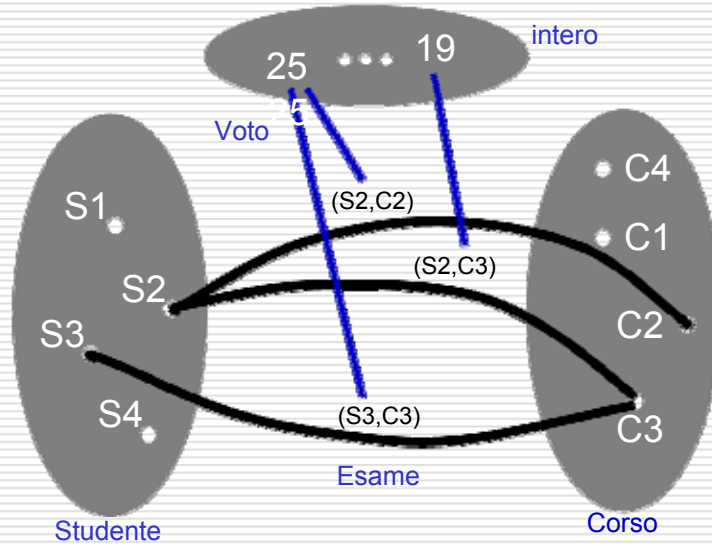
Se I   una istanza di uno schema S dove A   un attributo della relazione R su un dominio D , $istanze(I,A)$   un insieme di coppie (x,y) tali che x   in $istanze(I,R)$, y   in D , ed esiste una ed una sola coppia in $istanze(I,A)$ per ogni x in $istanze(I,R)$.

In altre parole, a livello **estensionale** l'attributo A   una **funzione totale**

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

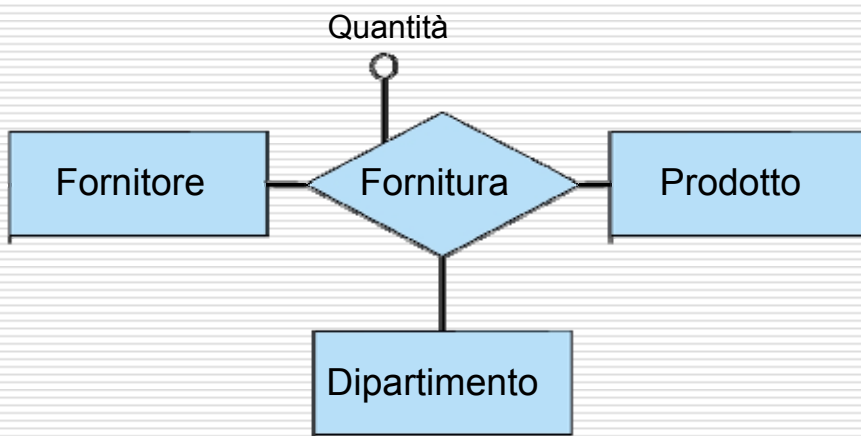
cio  una funzione che associa ad **ogni** istanza di R un valore appartenente a D .

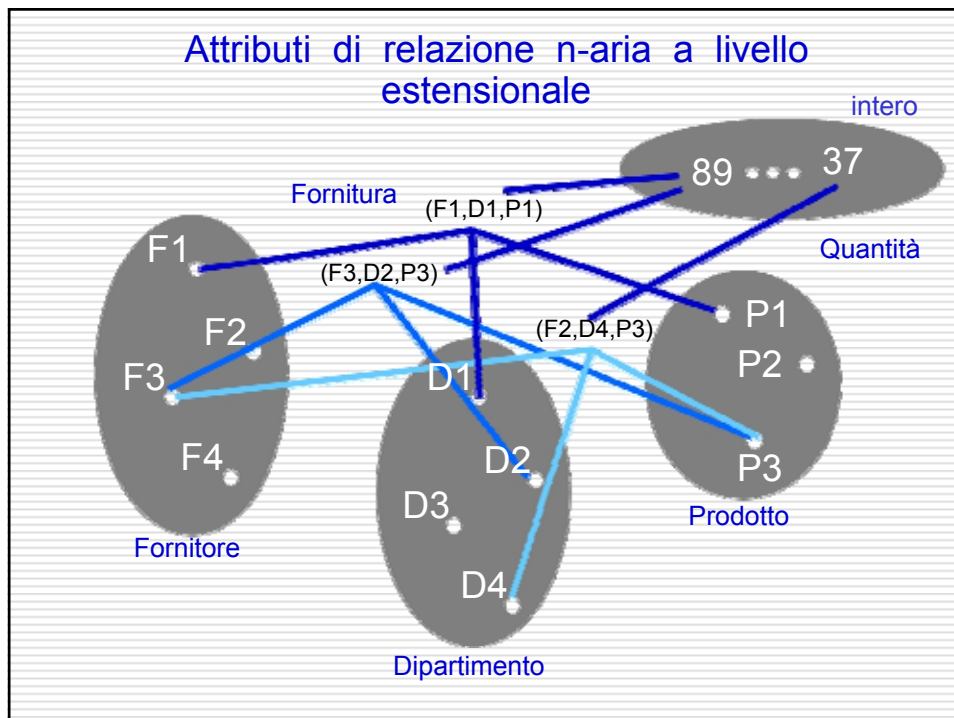
Attributi di relazione binaria a livello estensionale



Relazioni n-arie con attributi

Esempio di relazione n-aria con attributo:



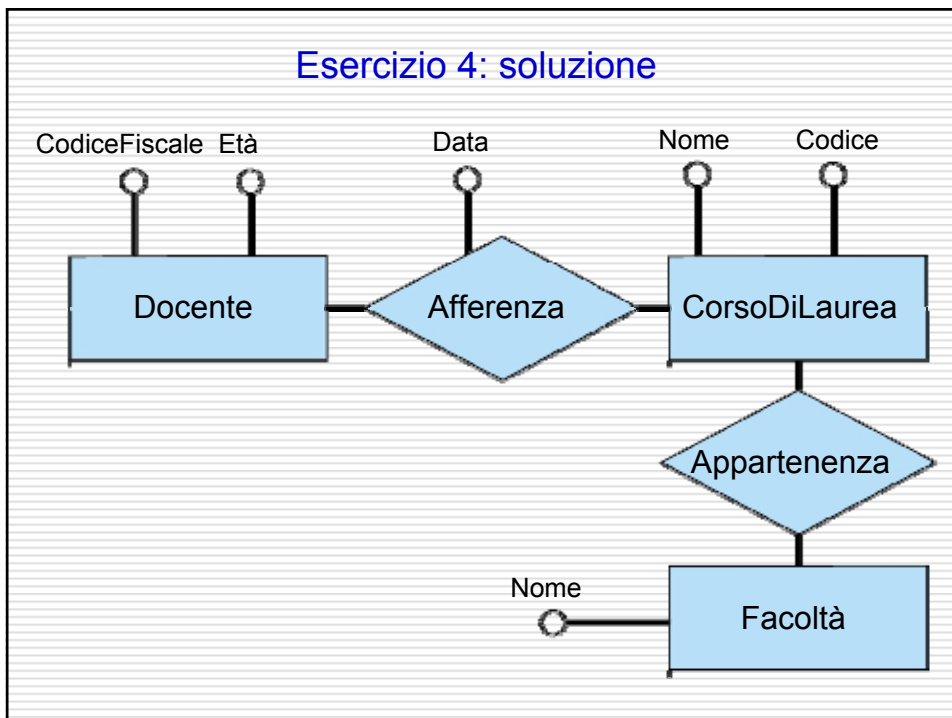


Esercizio 4: un semplice schema concettuale

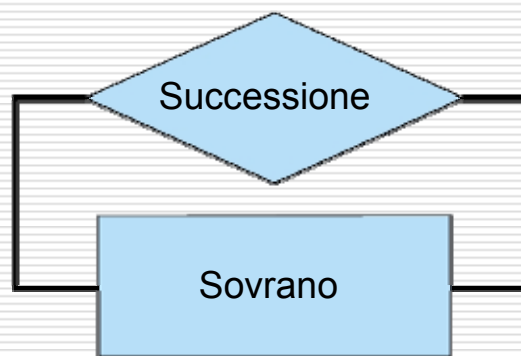
Descrivere lo schema concettuale della seguente realtà:

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



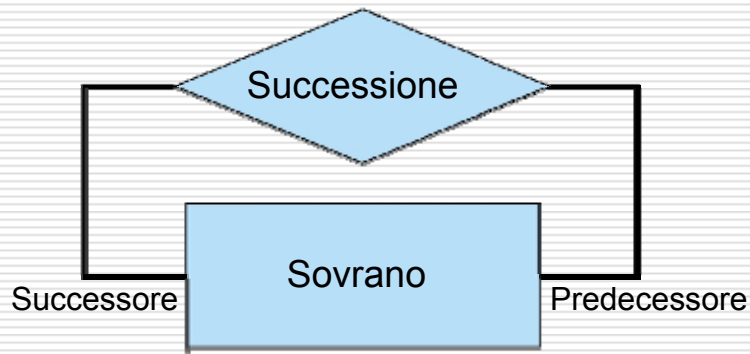
Una relazione pu  coinvolgere due o pi  volte la stessa entit 



Problema: in una istanza di questo schema, data una coppia che   istanza di "Successione", non si pu  individuare chi   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

- È possibile aggiungere ad una relazione l'informazione che specifica il ruolo che una 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 specifica del ruolo è necessaria nel caso di relazioni che coinvolgono la stessa entità più volte, mentre non sarebbe necessaria negli altri casi.
- Tuttavia, per uniformità, **noi richiediamo 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 identificato 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.
- A livello **estensionale** una relazione **R** tra le entità **E₁, E₂, ..., E_n** (non necessariamente tutte distinte) con rispettivi ruoli **U₁, U₂, ..., U_n** (tutti distinti) è costituita da un insieme di **n-ple (o tuple) etichettate**

$$(U_1:x_1, U_2:x_2, \dots, U_n:x_n)$$

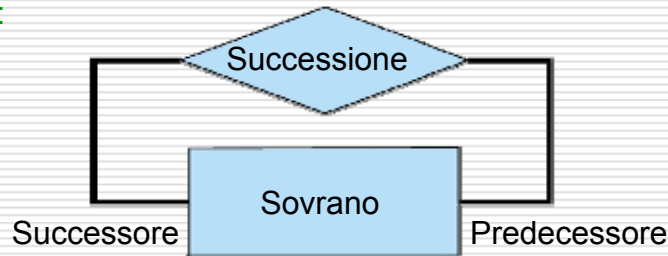
tali che **x₁** è una istanza di **E₁**, **x₂** è una istanza di **E₂**, ..., **x_n** è una istanza di **E_n**.

- Ogni n-ple etichettata è detta **istanza** della relazione **R**.

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

Semantica dei ruoli: esempio

Schema:



Istanza dello schema:

istanze(Sovrano) = { romolo,numa,tullo,anco }

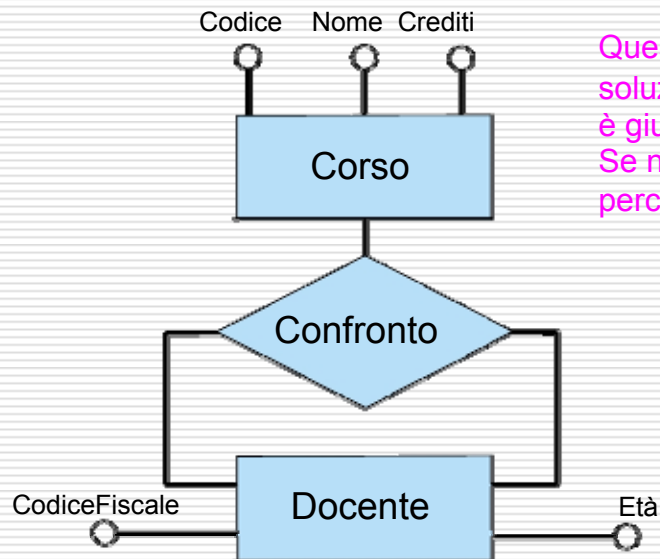
istanze(Successione) = { (Predecessore:romolo,Successore:numa),
(Predecessore:numa,Successore:tullo),
(Predecessore:tullo,Successore:anco) }

Esercizio 5: schema concettuale

Descrivere lo schema concettuale della seguente realtà:

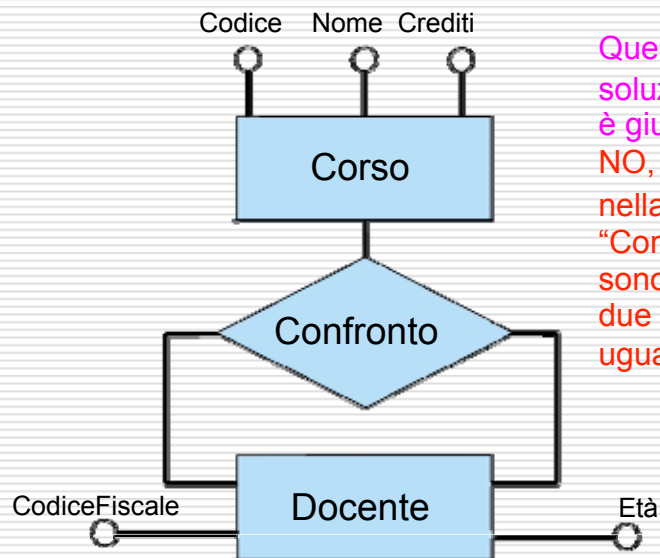
I docenti hanno un codice fiscale ed una età. I corsi hanno un codice identificativo, un nome, ed il numero di crediti. Sfruttando i moduli di valutazione dei corsi e dei docenti da parte degli studenti, si vuole rappresentare nella base di dati l'informazione se un docente è migliore di un altro nell'insegnare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

Esercizio 5: soluzione 1



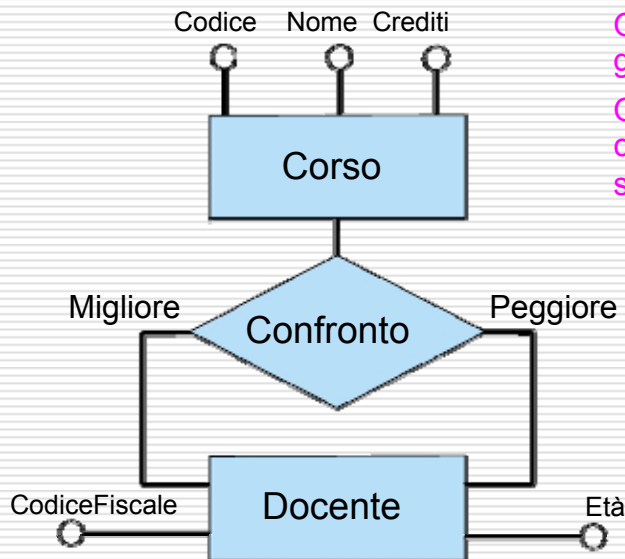
Questa soluzione è giusta? Se no, perché ?

Esercizio 5: soluzione 1



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

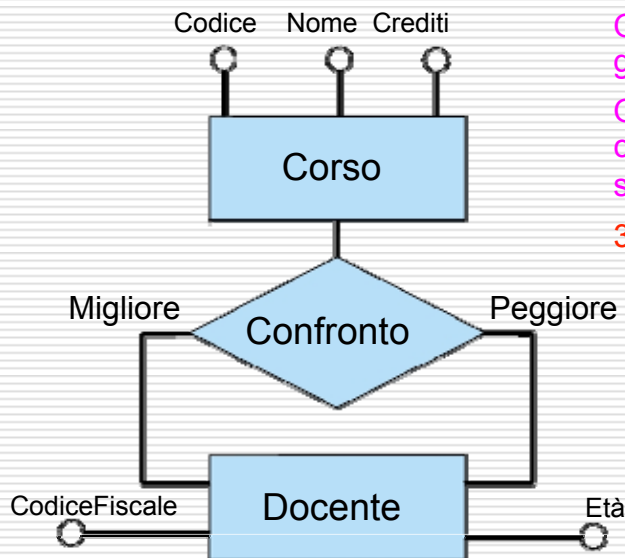
Esercizio 5: soluzione 2



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:

<u>Basi di dati</u>
1. Mario
2. Grazia
3. Paolo

<u>Logica</u>
1. Grazia
2. Mario
3. Paolo

<u>Algebra</u>
1. Grazia
2. Paolo

Esercizio 6: soluzione

Istanza I:

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

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

istanze(Confronto):

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

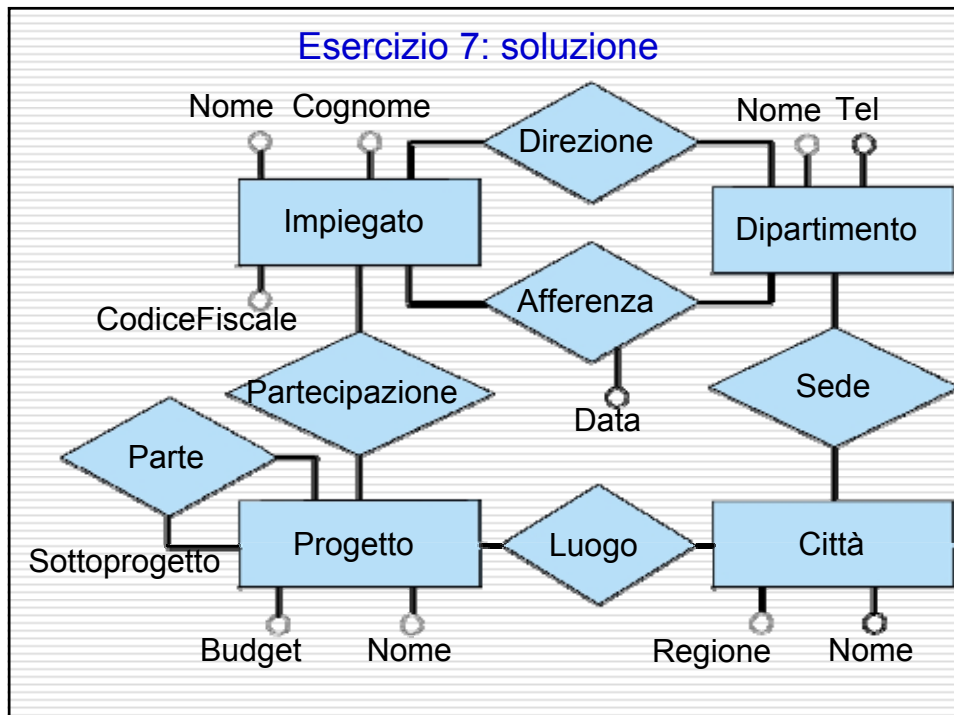
Esercizio 7: schema concettuale

Descrivere lo schema concettuale della seguente realtà:

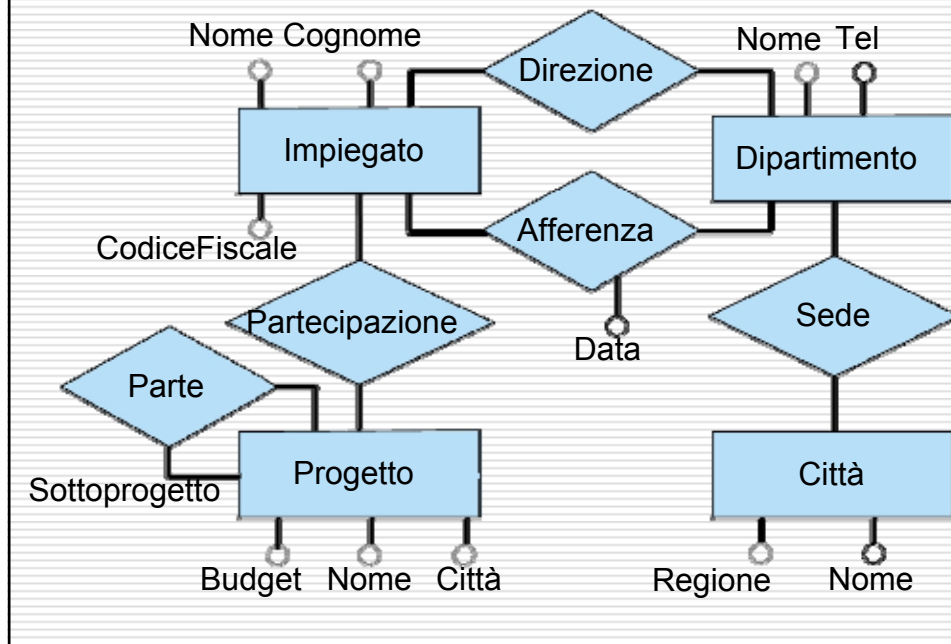
Degli impiegati interessa il codice fiscale, il nome, il cognome, i dipartimenti ai quali afferiscono (con la data di afferenza), ed i progetti ai quali partecipano. Dei progetti interessa il nome, il budget, e la città in cui hanno luogo le corrispondenti attività.

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?



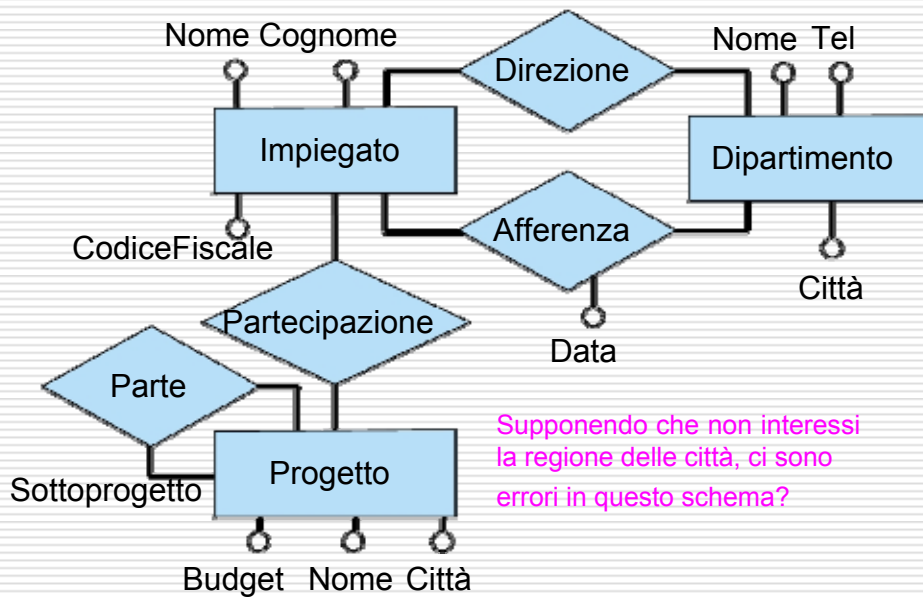
Esercizio 8: soluzione

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.

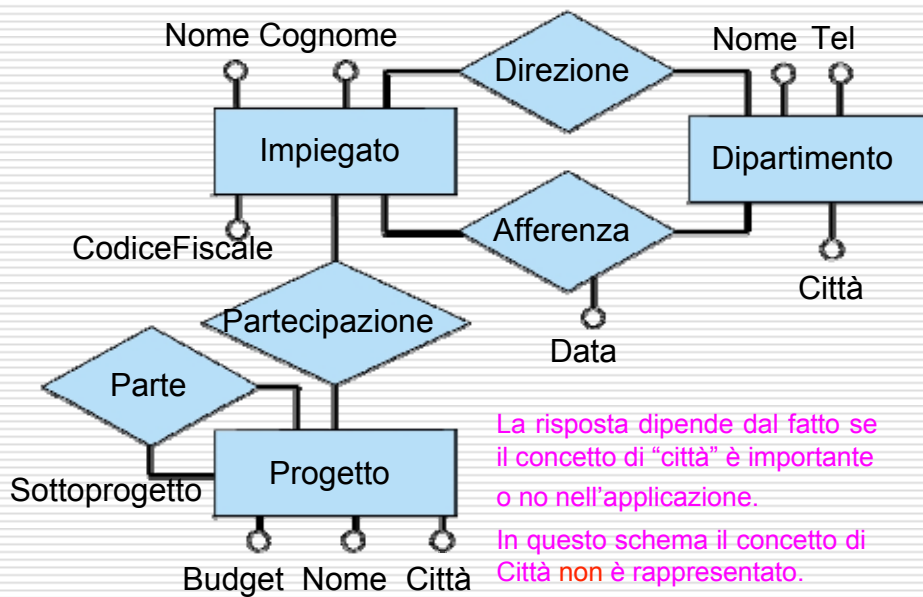
Infatti, **il nome dell'attributo (Città) non ha nulla a che vedere con la semantica** dello schema (che invece è caratterizzata dall'istanze dello schema): ciò che viene perso è che, in ogni istanza dello schema, il luogo di svolgimento di un progetto è una istanza dell'entità Città.

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



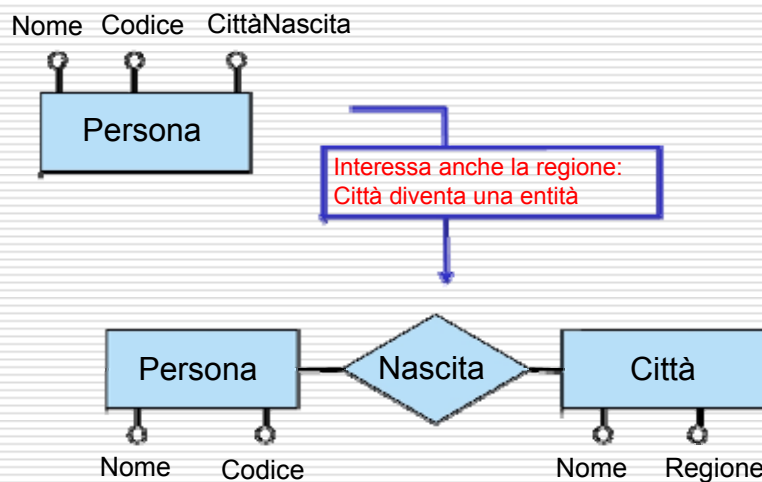
Scelta tra entità e attributo

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

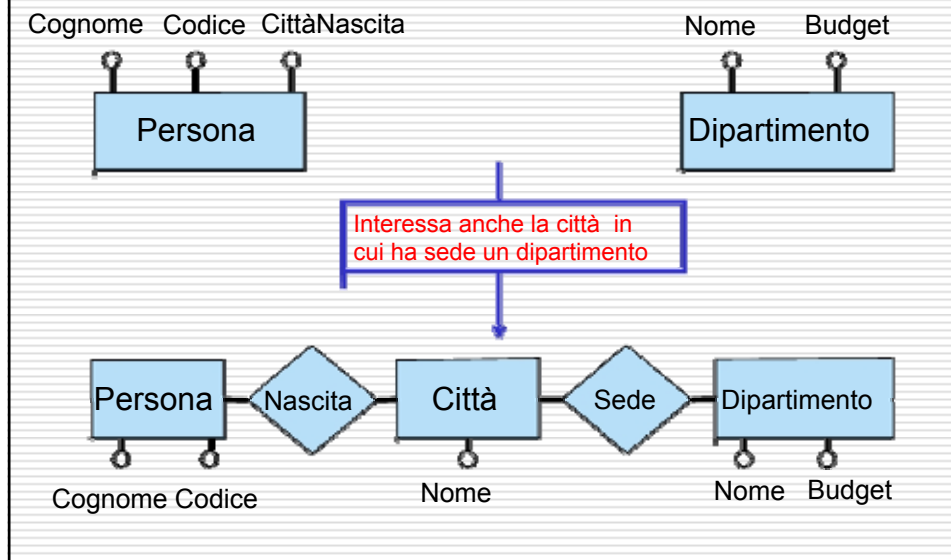
Scelta tra attributo ed entità

Le scelte possono cambiare durante l'analisi. **Esempio:**



Scelta tra attributo ed entità

Le scelte possono cambiare durante l'analisi. **Esempio:**



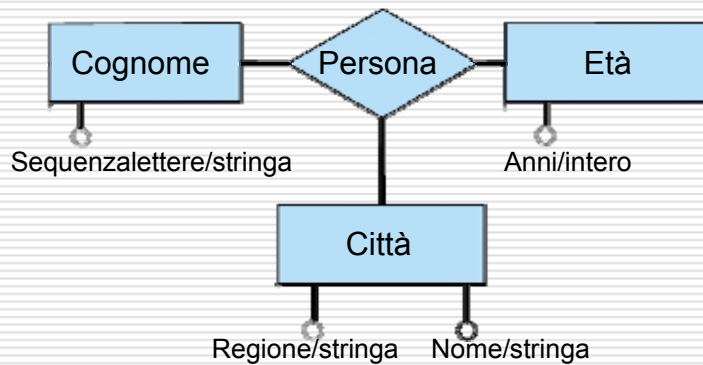
Scelta tra entità e relazione

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 modellata, 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. E' 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.

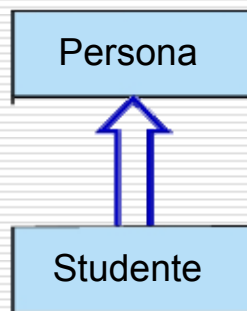
La nozione di relazione ISA

- Fino ad ora non abbiamo detto nulla sul fatto se due entità possano o no avere istanze in comune.
- E' facile verificare che, in molti contesti, può 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

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

Esempio:

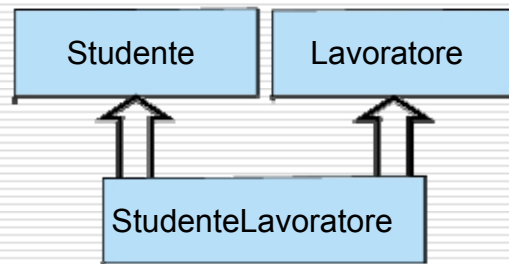


Si dice che Studente è in relazione ISA con Persona, o, in alternativa, che Studente ISA Persona

Sintassi: regole sulla relazione ISA

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

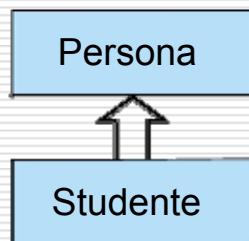
A livello estensionale la relazione ISA impone semplicemente che, in ogni istanza di uno schema in cui E_1 ISA E_2 :

$$\text{istanze}(E_1) \subseteq \text{istanze}(E_2)$$

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

Esempio:

Schema

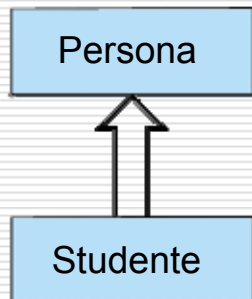


Istanza I:

$$\begin{aligned} \text{istanze}(\text{Persona}) &= \{ a, b, c, d \} \\ \text{istanze}(\text{Studente}) &= \{ b, d \} \end{aligned}$$

Relazione ISA a livello estensionale

Schema



Istanza I1:

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

Istanza I2:

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

Questa non è un'istanza:

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

Istanza I3:

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

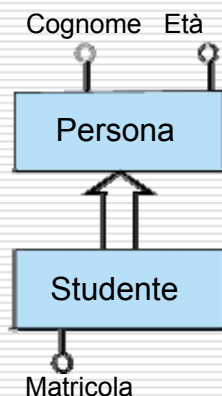
Ereditarietà nel modello ER

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

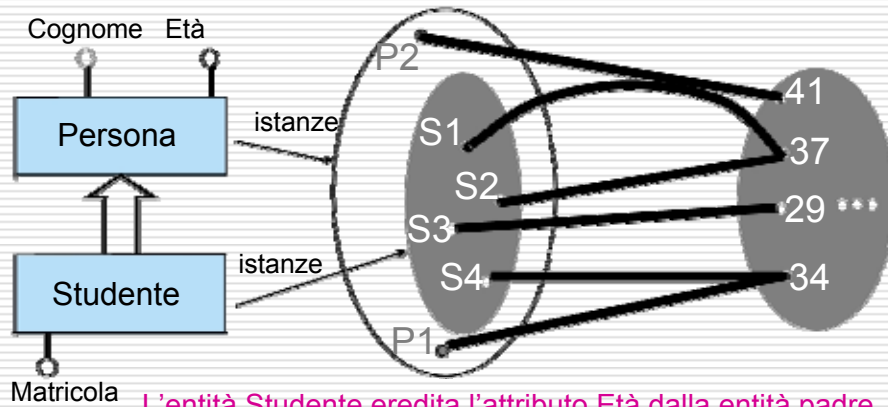
1. Ogni istanza di Persona ha una Età
 2. Ogni istanza di Studente è istanza di Persona
- segue logicamente che**
3. Ogni istanza di Studente ha una Età

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



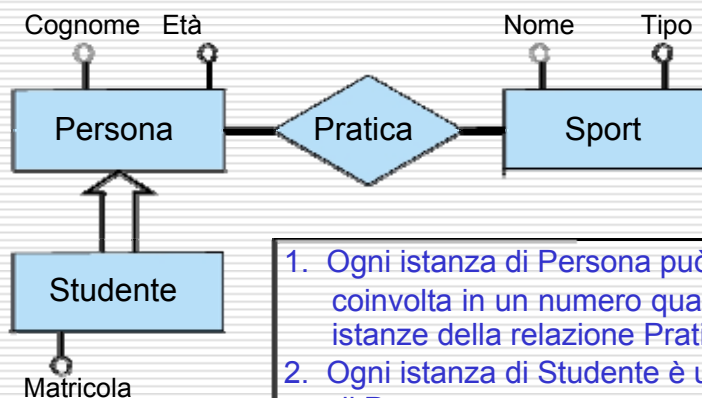
Cognome, Età ereditati da Persona, Matricola ulteriore proprietà

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



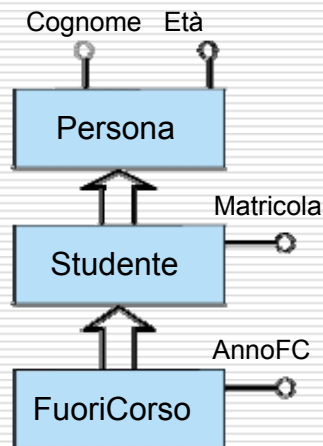
1. Ogni istanza di **Persona** può essere coinvolta in un numero qualunque di istanze della relazione **Pratica**
 2. Ogni istanza di **Studente** è una istanza di **Persona**
- quindi
3. Ogni istanza di **Studente** può essere coinvolta in un numero qualunque di istanze della relazione **Pratica**

Ereditarietà della relazione ISA: transitività

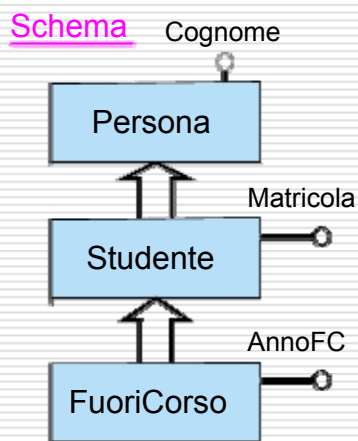
Principio di ereditarietà: anche la relazione ISA si eredita. Questo conferma che la relazione ISA è transitiva (oltre a essere riflessiva)

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?

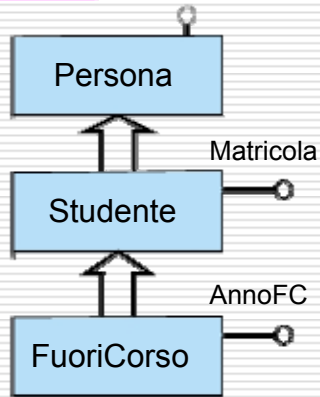


I1:

```
istanze(Persona) = { a,b,d }
istanze(Studente) = { b,d }
istanze(FuoriCorso) = { b,d }
istanze(Cognome) =
  { (a,'a'), (b,'b'), (d,'d') }
istanze(Matricola) =
  { (b,'12'), (d,'34'), (a,'89') }
istanze(AnnoFC) = { (b,2) }
```

Esercizio 11: soluzione

Schema Cognome



I1:

```
istanze(Persona) = { a,b,d }
istanze(Studente) = { b,d }
istanze(FuoriCorso) = { b,d }
istanze(Cognome) =
    { (a,'a'), (b,'b'), (d,'d') }
istanze(Matricola) =
    { (b,'12'), (d,'34'), (a,'89') }
istanze(AnnoFC) = { (b,2) }
```

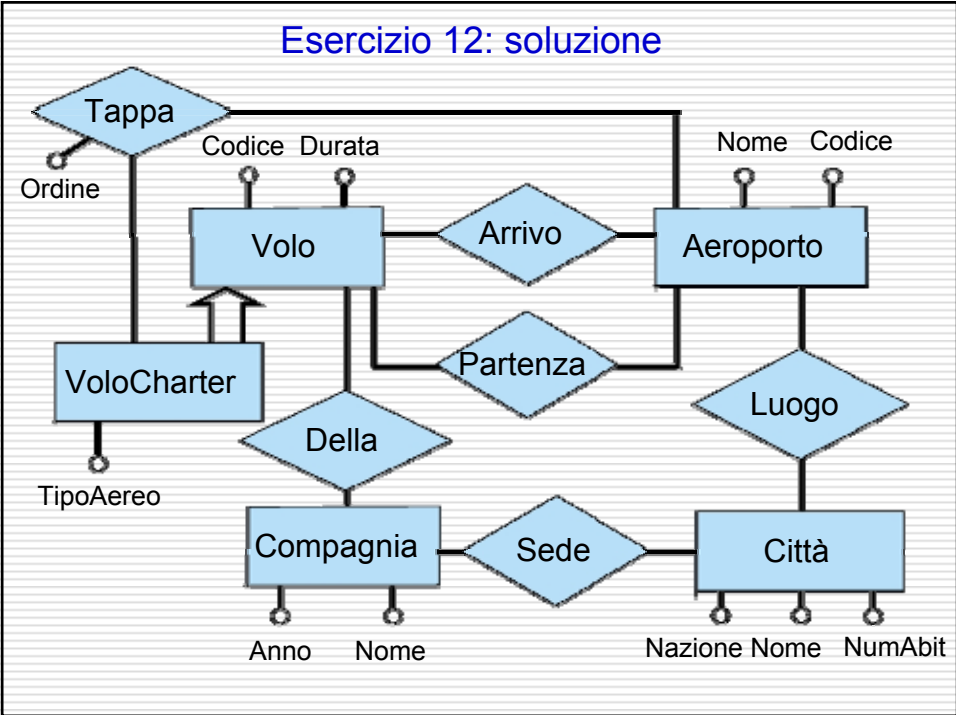
La risposta è NO, perchè: - (a,'89') in Matricola
 - manca AnnoFC per d

Esercizio 12: schema concettuale

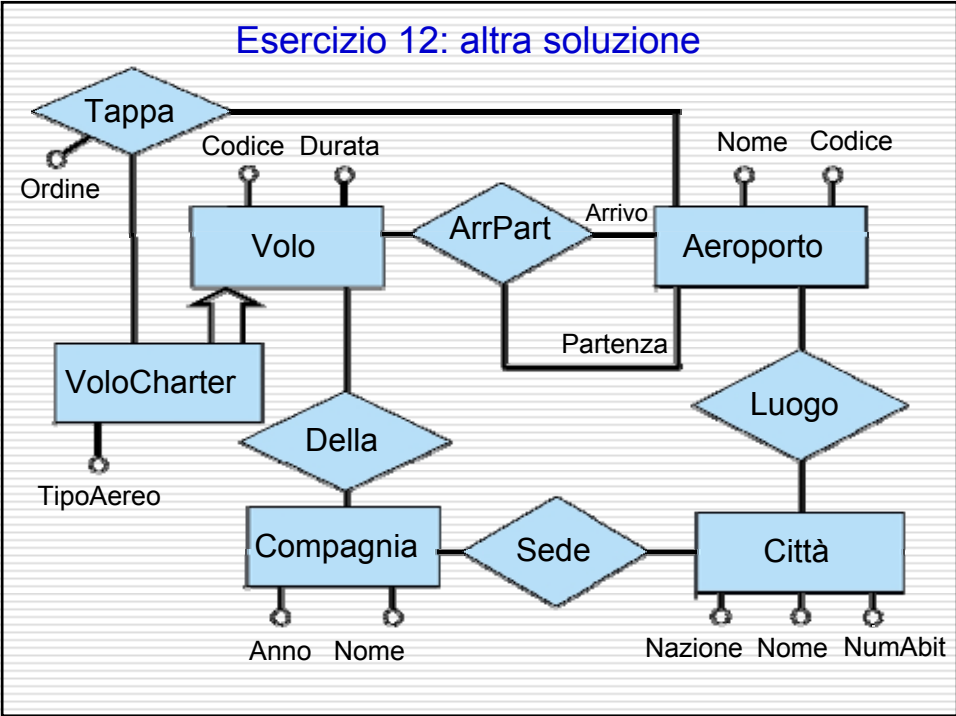
Descrivere lo schema concettuale corrispondente ad un'applicazione riguardante voli aerei, per la quale valgono le seguenti specifiche.

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



La nozione di generalizzazione tra entità

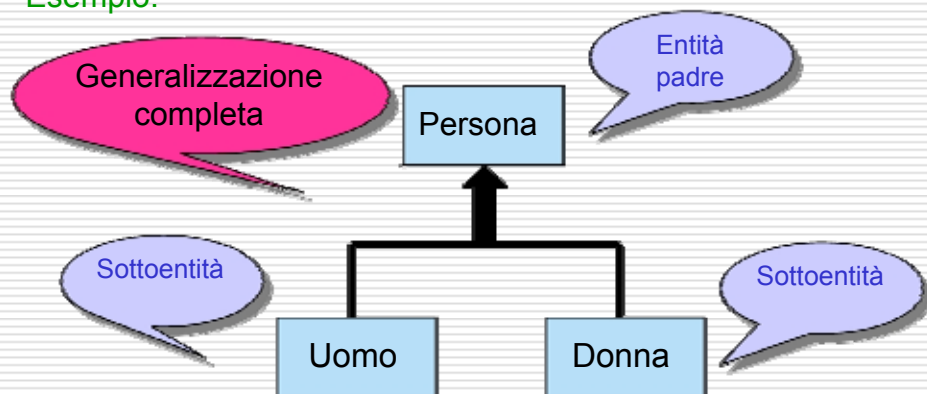
- Finora, abbiamo considerato la relazione ISA che stabilisce che l'entità padre è più generale della sottoentità. • Talvolta, però, l'entità padre può generalizzare diverse sottoentità rispetto ad un unico criterio. In questo caso si parla di **generalizzazione**.
- Nella generalizzazione, le **sottoentità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

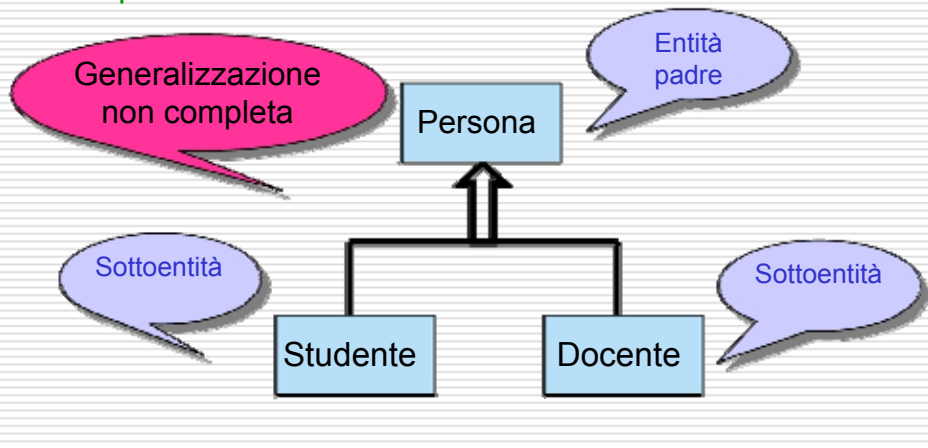
Esempio:



Sintassi: rappresentazione della generalizzazione

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

Esempio:



Semantica della generalizzazione

Se in uno schema \mathcal{S} è definita una generalizzazione tra una entità padre F e le sottoentità E_1, E_2, \dots, E_n , in ogni istanza I dello schema \mathcal{S} si ha:

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

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

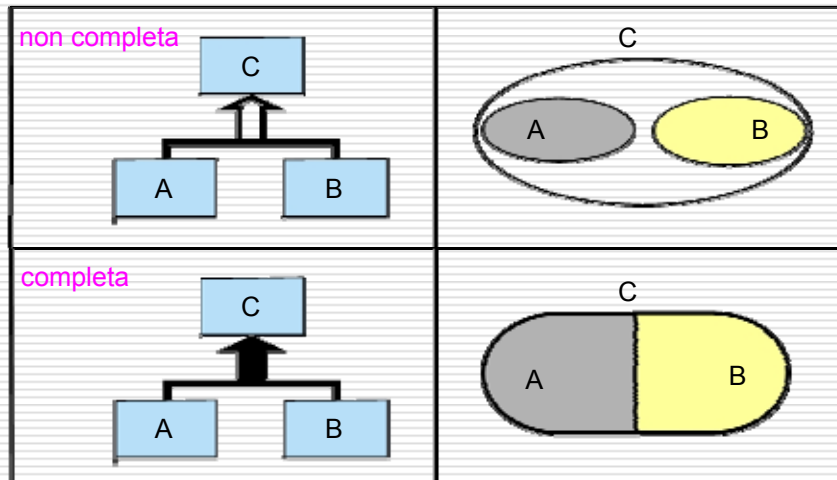
- $istanze(E_1) \cup \dots \cup istanze(E_n) = istanze(F)$

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

Generalizzazioni a livello estensionale

Livello intensionale

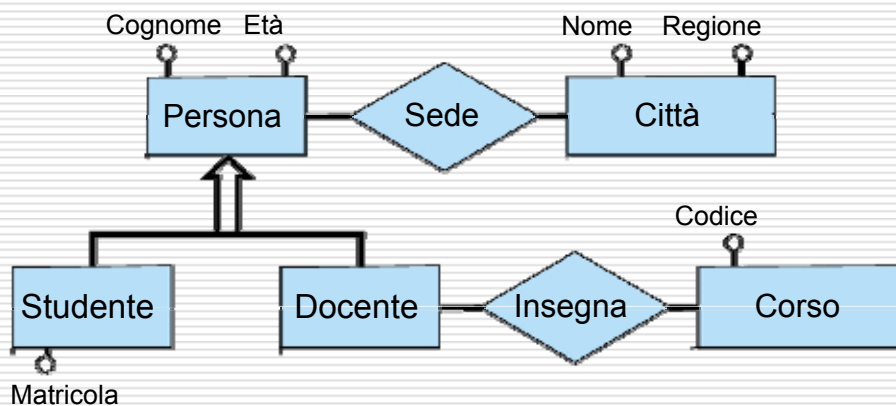
Livello estensionale



Generalizzazioni ed ereditarietà

Il principio di ereditarietà vale anche per le generalizzazioni:

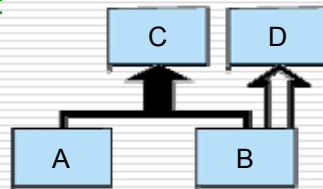
- ogni proprietà dell'entità padre è anche una proprietà della sottoentità, e non si riporta esplicitamente nel diagramma
- l'entità figlia può avere ovviamente ulteriori proprietà.



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)

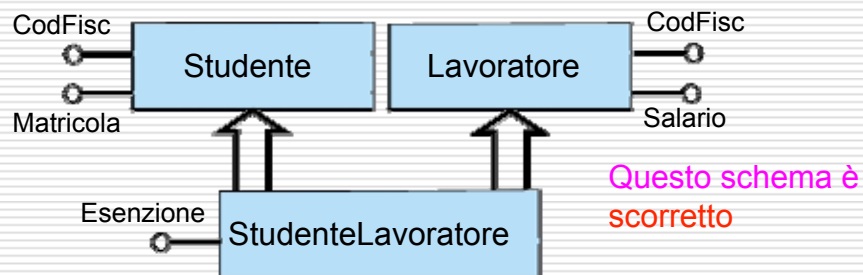
Esempio:



Questo schema è scorretto

Dato uno schema S, si può costruire un grafo che ha le entità di S come nodi ed ha un arco da una entità E ad una entità F se F è padre di E secondo la relazione ISA o secondo la generalizzazione. Diremo che una entità G è **antenata** di E in uno schema se esiste un cammino da E ad G nel grafo associato.

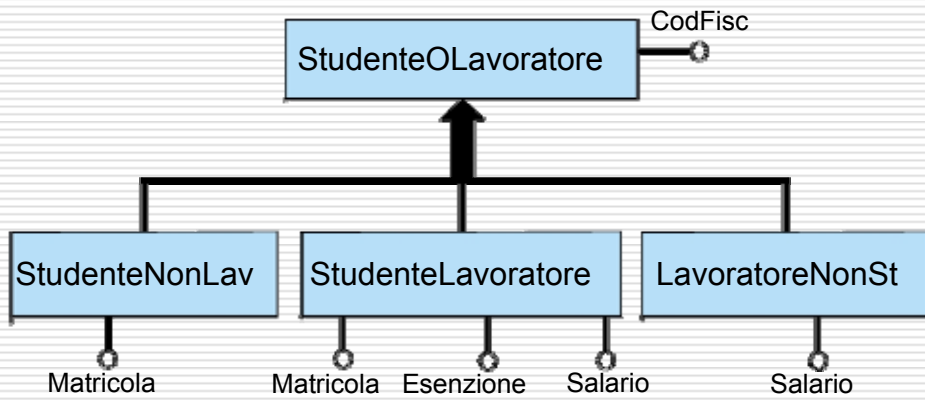
Esercizio 13: relazione ISA e generalizzazioni



Questo schema è scorretto

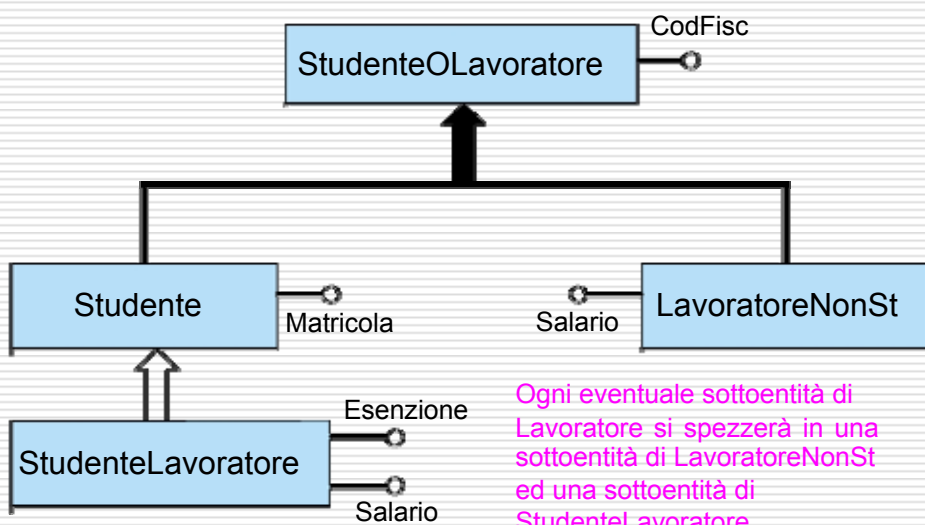
Si può ristrutturare lo schema in modo che lo schema risultante sia corretto, e colga sostanzialmente la stessa semantica?

Esercizio 13: soluzione 1



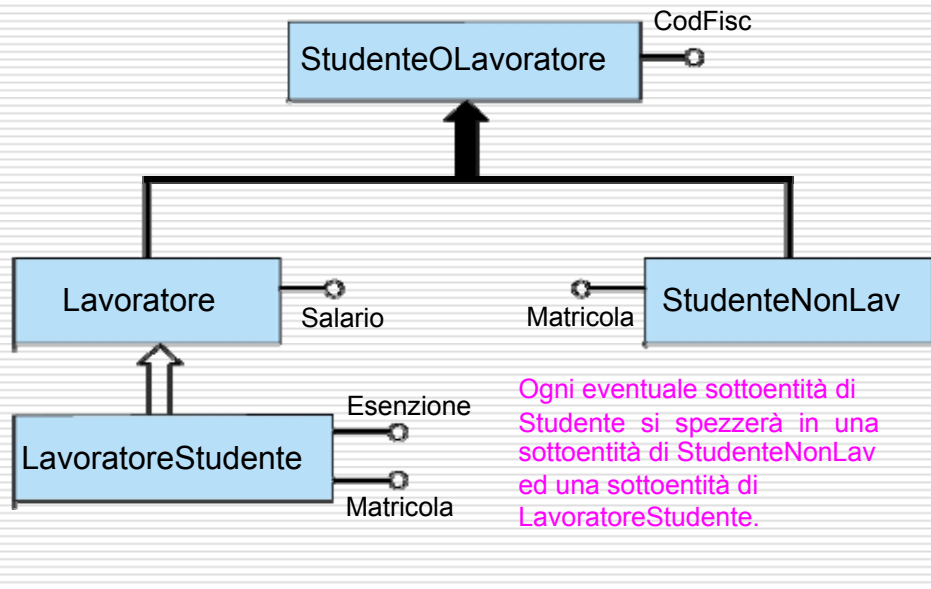
Ogni eventuale sottoentità di Studente si spezzerà in una sottoentità di StudenteNonLav ed una sottoentità di StudenteLavoratore. Analogamente, ogni sottoentità di Lavoratore si spezzerà in una sottoentità di LavoratoreNonSt ed una sottoentità di StudenteLavoratore.

Esercizio 13: soluzione 2



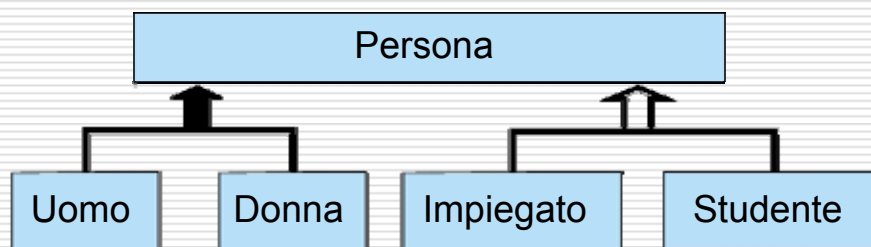
Ogni eventuale sottoentità di Lavoratore si spezzerà in una sottoentità di LavoratoreNonSt ed una sottoentità di StudenteLavoratore.

Esercizio 13: soluzione 3



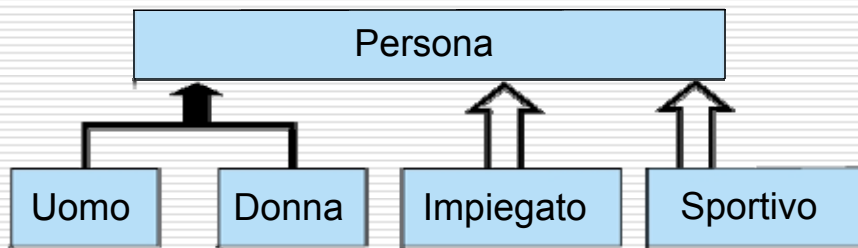
Diverse generalizzazioni della stessa entità

La stessa entità può essere padre in diverse generalizzazioni.



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

Differenza tra due isa e una generalizzazione



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

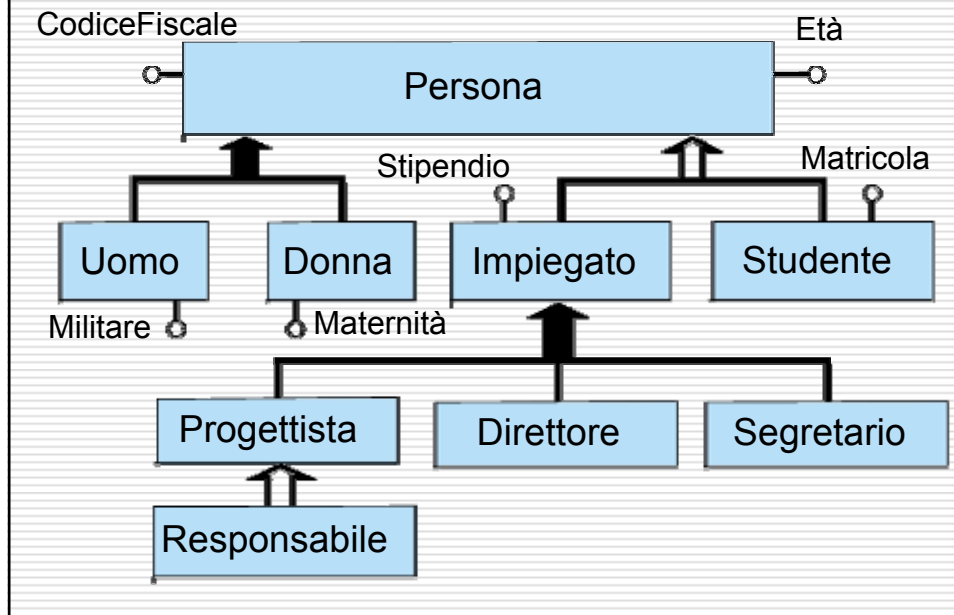
Le due sottoentità sono indipendenti, nel senso che il loro significato non deriva dallo stesso criterio di classificazione delle istanze della entità padre.

Esercizio 13: 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 responsabile di progetto. Gli studenti (che non possono essere impiegati) hanno un numero di matricola. Esistono persone che non sono né impiegati né studenti.

Esercizio 13: soluzione



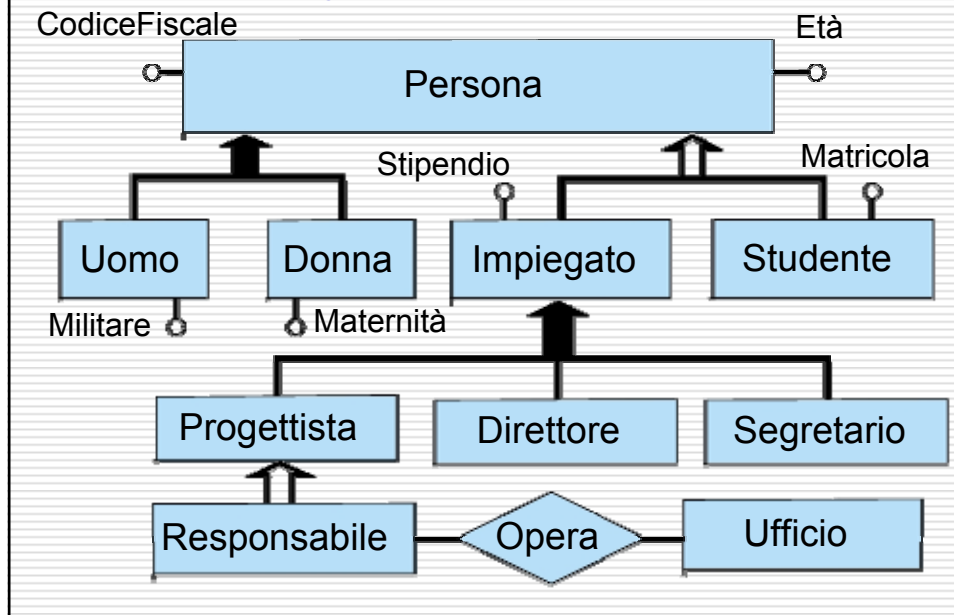
Entità disgiunte

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

A questo proposito, valgono queste osservazioni:

- In ogni schema si assume che le entità che non hanno padre nello schema siano in realtà sottoentità di una generalizzazione completa la cui entità padre è una entità speciale detta **TOP** (ne segue che tali entità sono disgiunte a coppie, e che in ogni istanza dello schema, **TOP** rappresenta tutti gli oggetti che sono istanze di qualche entità).
- E' facile verificare che due entità **E** ed **F** sono **disgiunte** in uno schema **S** se esistono due entità **G** ed **H** (non necessariamente diverse da **E** ed **F**) nello schema **S** tali che:
 - **G** ed **H** sono sottoentità della stessa entità padre in una generalizzazione
 - **G** è antenata di **E**, ed **H** è antenata di **F**

Esercizio 14: elencare tutte le coppie di entità disgiunte in questo schema



Esercizio 14: soluzione

Coppie di entità disgiunte:

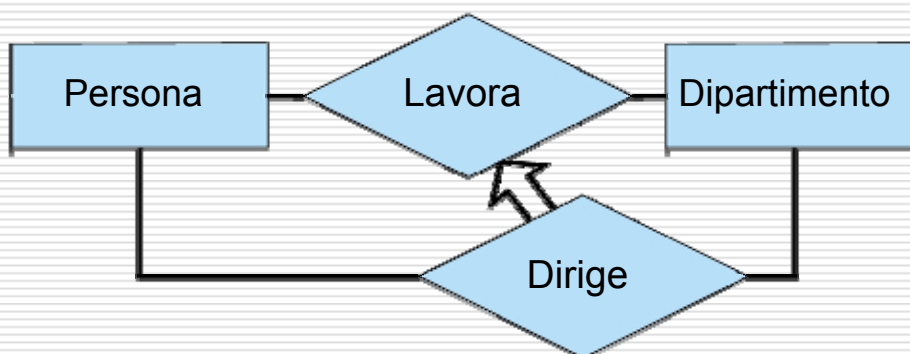
1. Uomo e Donna
2. Impiegato e Studente
3. Progettista e Studente
4. Direttore e Studente
5. Segretario e Studente
6. Responsabile e Studente
7. Progettista e Direttore
8. Progettista e Segretario
9. Direttore e Segretario
10. Responsabile e Direttore
11. Responsabile e Segretario
12. Ufficio e ognuna delle altre entità

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 grado
 - hanno gli stessi ruoli
 - per ogni ruolo U, l'entità corrispondente ad U in Q è un'entità antenata dell'entità corrispondente ad U in R.
- 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, $istanze(I,R) \subseteq istanze(I,Q)$.
- Analogamente, si possono definire generalizzazioni tra relazioni.

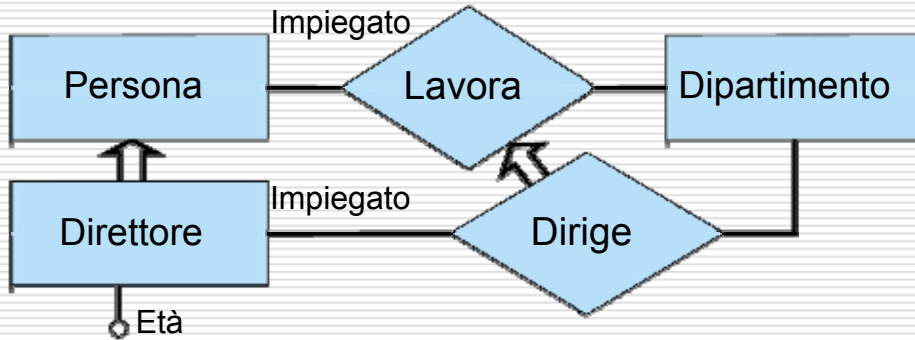
Esempio di relazione ISA tra relazioni

Il direttore di un dipartimento deve lavorare in quel dipartimento.



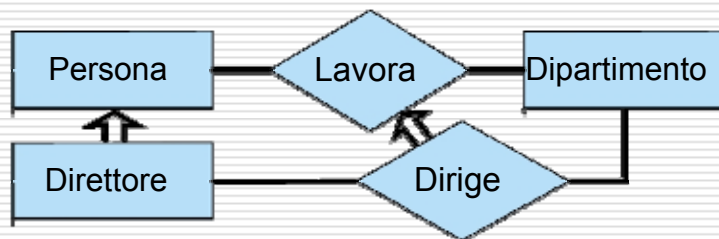
Esempio di relazione ISA tra relazioni

Il direttore di un dipartimento deve lavorare in quel dipartimento, ed ha proprietà aggiuntive rispetto agli impiegati.

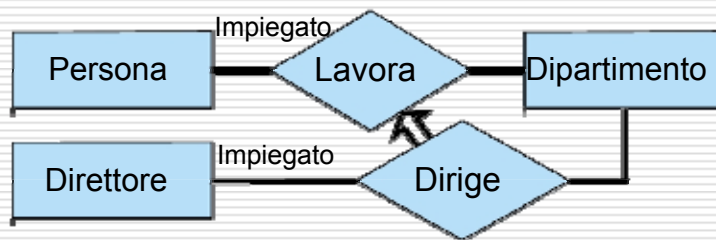


Esempi di schemi non corretti

Ruoli diversi
nelle due
relazioni



Ruoli uguali
ma entità non
in relazione
ISA



Relazione disgiunte

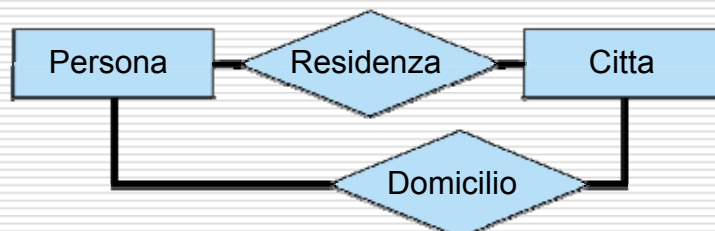
Per le relazioni valgono osservazioni simili a quelle che abbiamo fatto sulle entità:

- In ogni schema si assume che le relazioni dello stesso grado e con gli stessi ruoli U_1, U_2, \dots, U_k che non hanno padre nello schema siano in realtà sottoentità in una ISA la cui relazione padre è una relazione speciale detta $TOP_{U_1, U_2, \dots, U_k}$.
- Due relazioni P e Q (dello stesso grado e con gli stessi ruoli) sono **disgiunte** in uno schema S se esistono due relazioni R e T (non necessariamente diverse da P e Q) in S tali che:
 - R e T sono relazioni figlie della stessa relazione padre in una generalizzazione
 - R è antenata di P , e T è antenata di Q

Relazione disgiunte (cont.)

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



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 $(x_1, y_1), \dots, (x_n, y_n)$ - vedi dopo) in ogni istanza dello schema S , A è una relazione (o una funzione se la cardinalità massima è sempre 1) del tipo

$$\text{Istanze}(A) \subseteq (\text{Istanze}(E_1) \cup \dots \cup \text{Istanze}(E_n)) \times (D_1 \cup \dots \cup D_n)$$

e tale che, per ogni I vale la seguente condizione:

- se $a \in \text{Istanze}(E_i)$, allora il numero di coppie (a, b) in $\text{Istanze}(A)$ è soggetto ai vincoli di cardinalità (x_i, y_i) , e per ogni $(a, b) \in \text{Istanze}(A)$, 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.

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

Vincoli di integrità nel modello ER

Un **vincolo di integrità** è una regola che si esprime sullo schema (livello intensionale), ma 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 d'entità
- altri vincoli (esterni)

Vincoli di cardinalità sulle relazioni

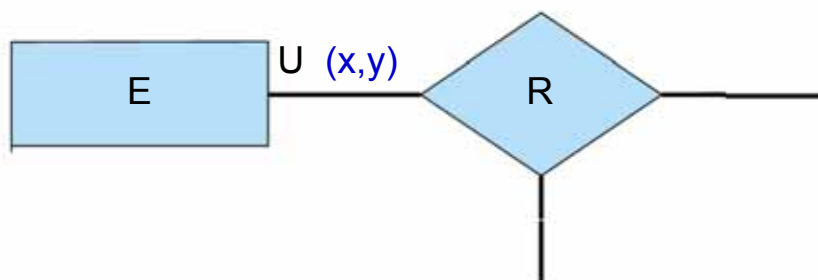
Un **vincolo di cardinalità** si associa ad un ruolo U (corrispondente ad una entità E) in una relazione R, ed impone un limite minimo ed un limite massimo di istanze della relazione a cui può partecipare ogni istanza dell'entità E nel ruolo U.

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(E), il numero di istanze di R che 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 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 vincolo di cardinalità

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



Istanza:

istanze(Impiegato) = { a,b,c }

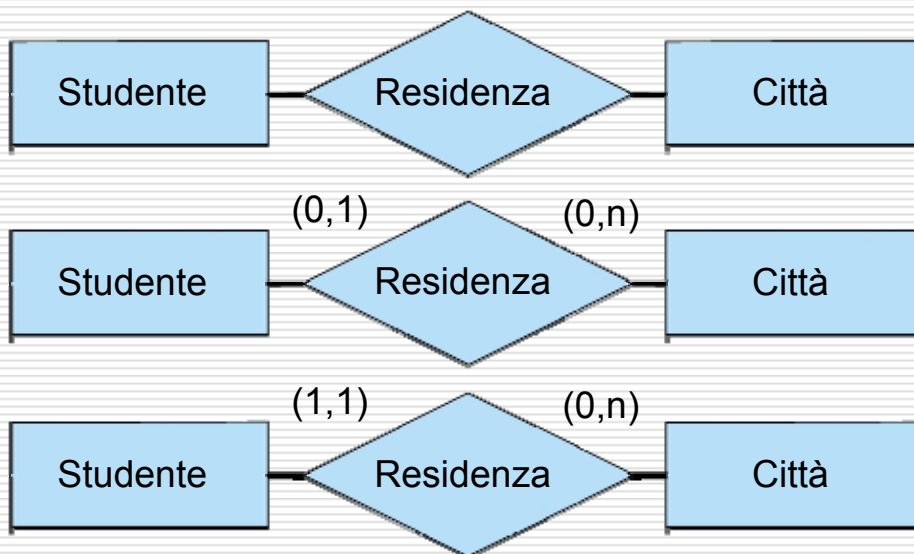
istanze(Incarico) = { x,y,v,w,z }

istanze(Assegnazione) = { (a,w), (b,v), (b,w), (c,y), (c,w), (c,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 **mancaza del vincolo di cardinalità** è equivalente alla coppia (0,n).

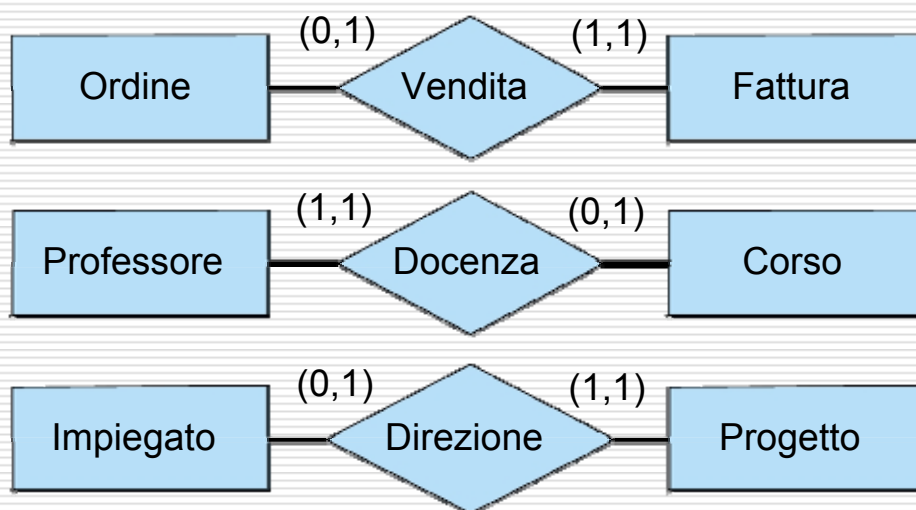
Esempi di vincolo di cardinalità



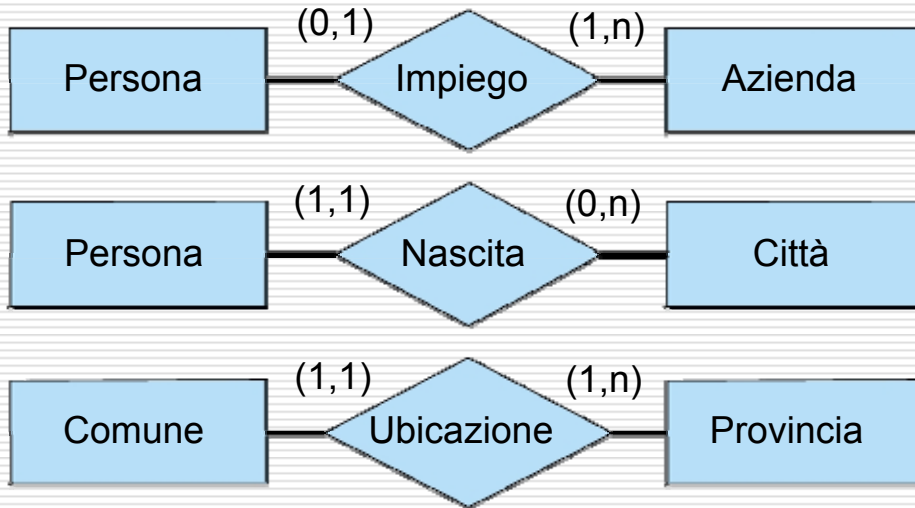
Classificazione di relazioni binarie

- Discutiamo la cardinalità massime sulle relazioni binarie.
 - Con riferimento alle cardinalità **massime**, le relazioni binarie possono essere di tipo:
 - uno a uno
 - uno a molti
 - molti a molti
-

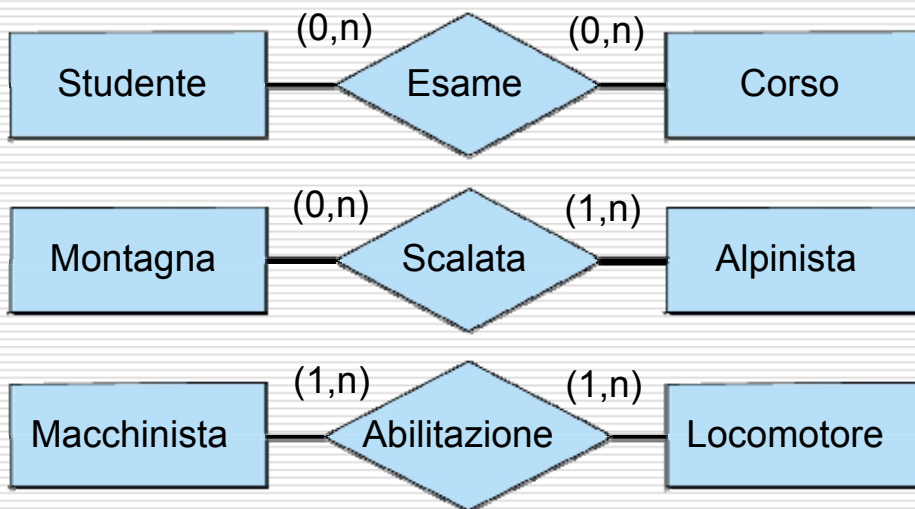
Relazioni binarie "uno a uno"



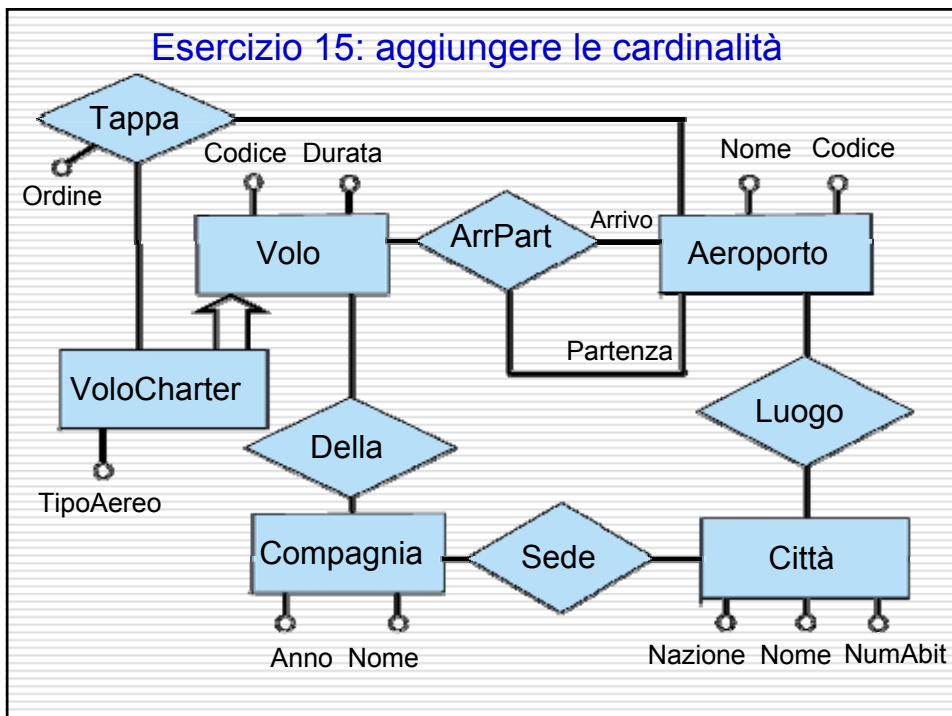
Relazioni binarie "uno a molti"



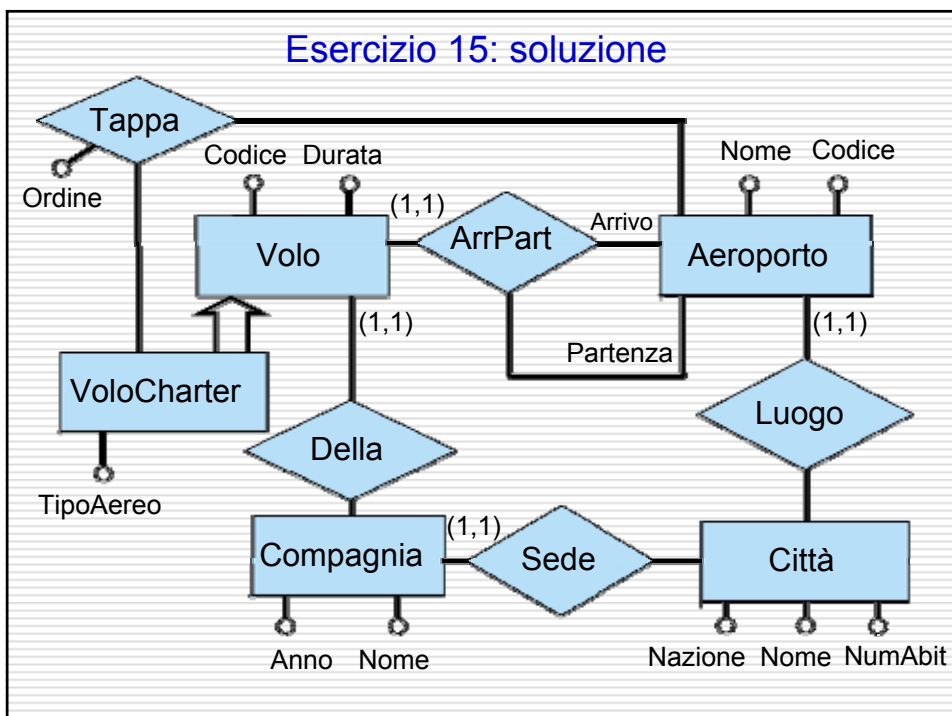
Relazioni binarie "molti a molti"



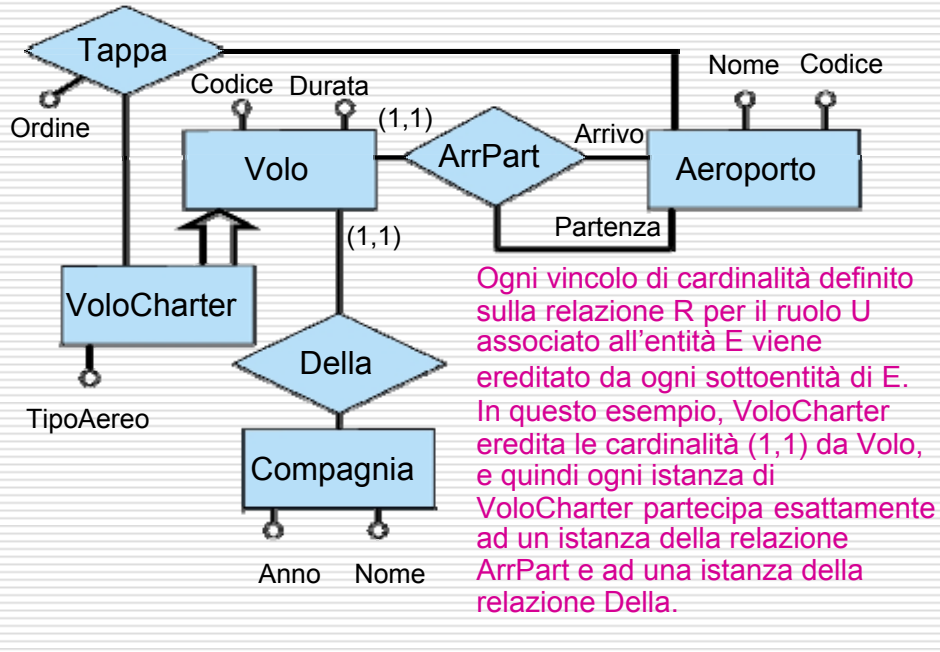
Esercizio 15: aggiungere le cardinalità



Esercizio 15: soluzione

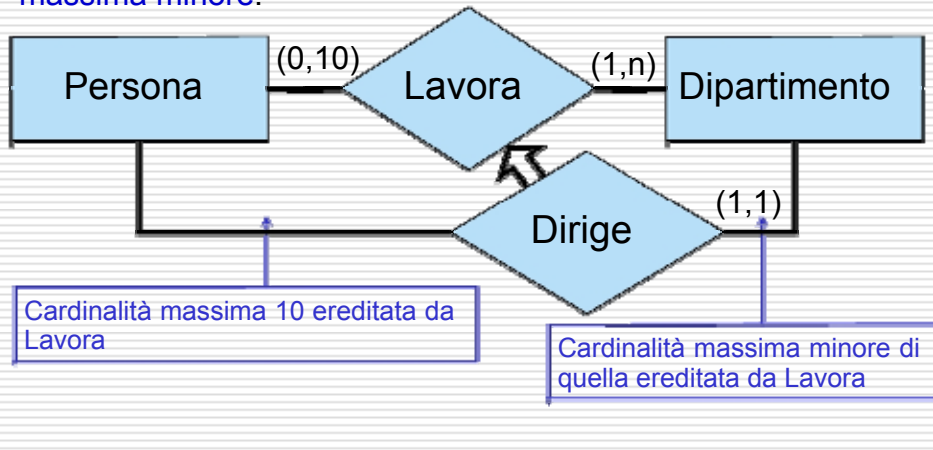


Vincoli di cardinalità ed ereditarietà su entità



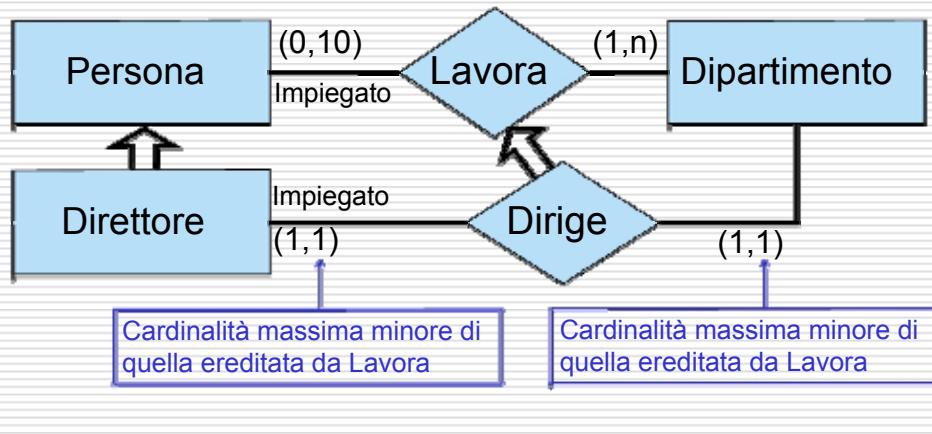
Vincoli di cardinalità ed ereditarietà su relazioni

Se Q ISA R, ogni vincolo di cardinalità **massima** definito sulla relazione R (per il ruolo U) viene ereditato dalla relazione Q (per lo stesso ruolo U), che può però avere cardinalità massima più stringente di quella ereditata, ovvero **cardinalità massima minore**.



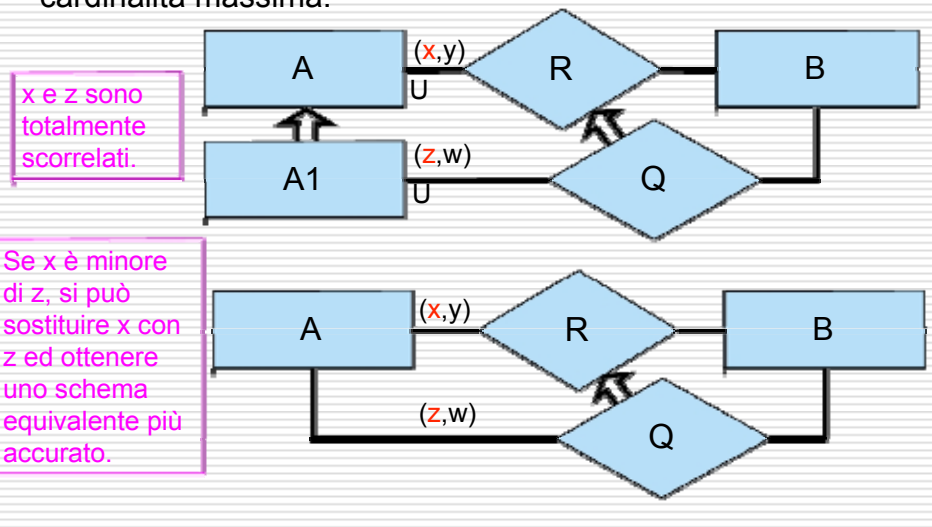
Vincoli di cardinalità ed ereditarietà su relazioni

La regola sulla ereditarietà 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.



Vincoli di cardinalità ed ereditarietà su relazioni

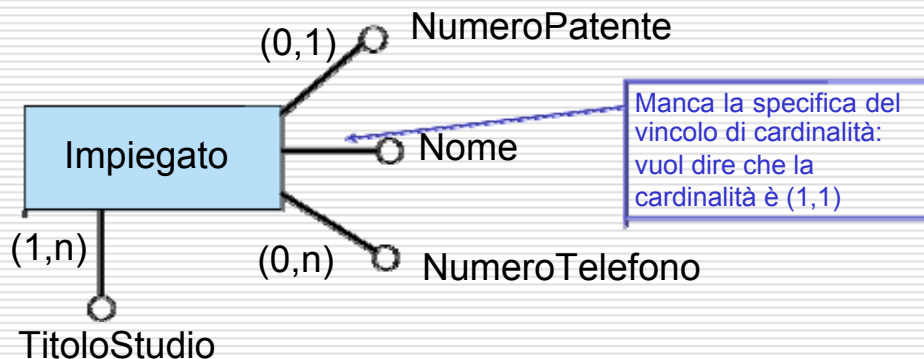
Se $Q \text{ ISA } R$, il rapporto tra i vincoli di cardinalità minima su Q ed R sono più complessi rispetto al caso dei vincoli di cardinalità massima.



Vincoli di cardinalità sugli attributi

- E' possibile definire vincoli di cardinalità anche sugli attributi, con due scopi:
 - indicare **opzionalità**
 - indicare **attributi multivalore**
- Le cardinalità stabiliscono il numero minimo e massimo di valori che l'attributo associa ad ogni istanza dell'entità.
- La semantica di un attributo con cardinalità diversa 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**.
 - 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



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 non esistono due istanze di una data entità che assumono 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).

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 E con 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 gli 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), si indica unendo gli attributi ed i ruoli con una linea che termina con pallino annerito.

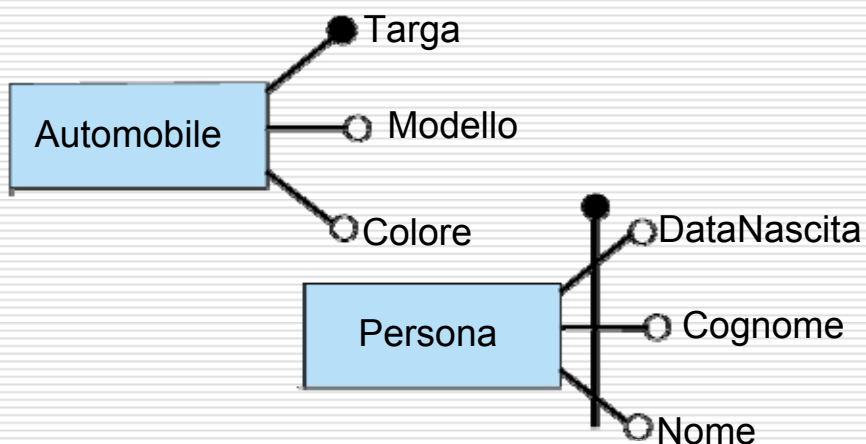
Semantica dei vincoli di identificazione di entità

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

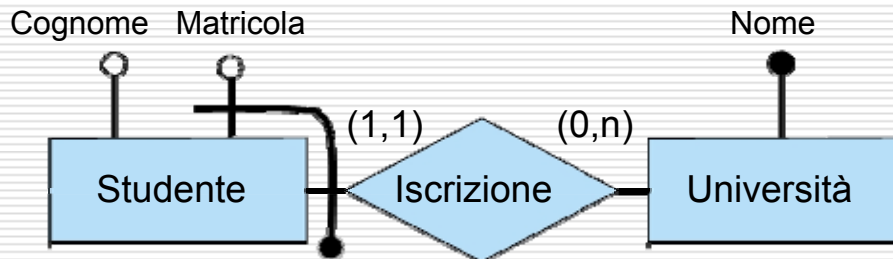
- dagli attributi A_1, A_2, \dots, A_k
- dalle relazioni R_1 (ruolo U_1), R_2 (ruolo U_2), ..., R_h (ruolo U_h)

allora in ogni istanza I dello schema S , prese due diverse istanze qualunque e_1 ed e_2 in istanze (E), esse differiscono per il valore di almeno un A_i o per la partecipazione con ruolo U_i in almeno una R_i .

Esempi di identificatori interni

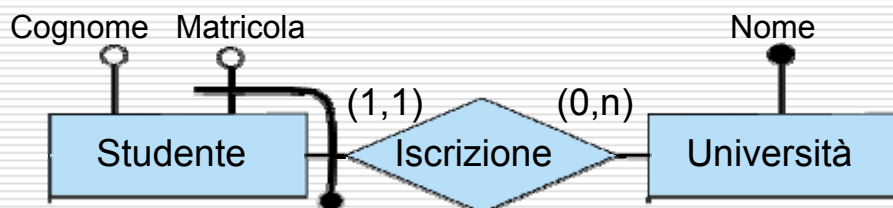


Esempio di identificatore esterno



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

Identificatore esterno: livello estensionale



Istanza I:

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

istanze(Università) = { v,z }

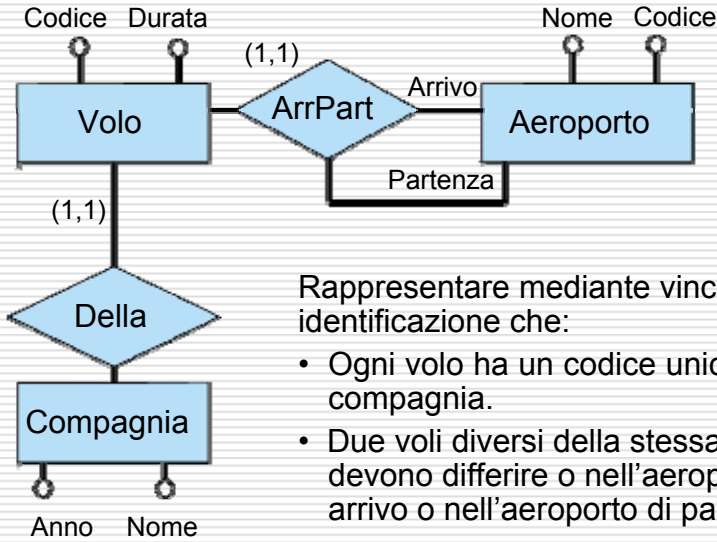
istanze(Matricola) = { (a,12), (b,24), (c,12) }

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

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

istanze(Iscrizione) = { (a,v), (b,v), (c,z) }

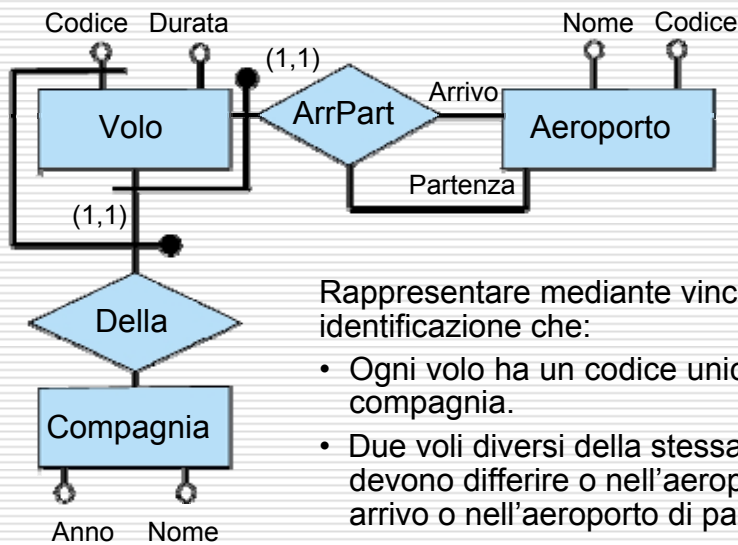
Esercizio 16: vincoli di identificazione



Rappresentare mediante vincoli di identificazione 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.

Esercizio 16: soluzione

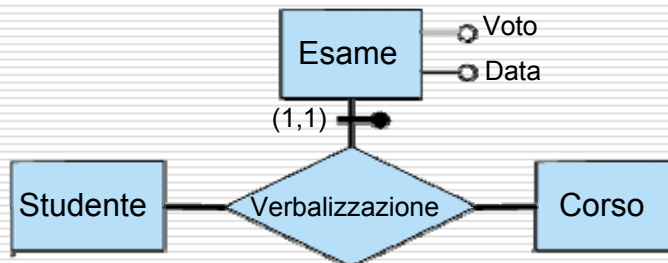


Rappresentare mediante vincoli di identificazione che:

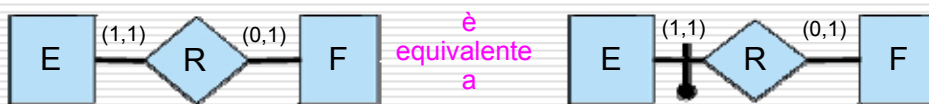
- Ogni volo ha un codice unico nella compagnia.
- Due voli diversi della stessa compagnia devono differire o nell'aeroporto di arrivo o nell'aeroporto di partenza.

Esempi di identificazione esterna

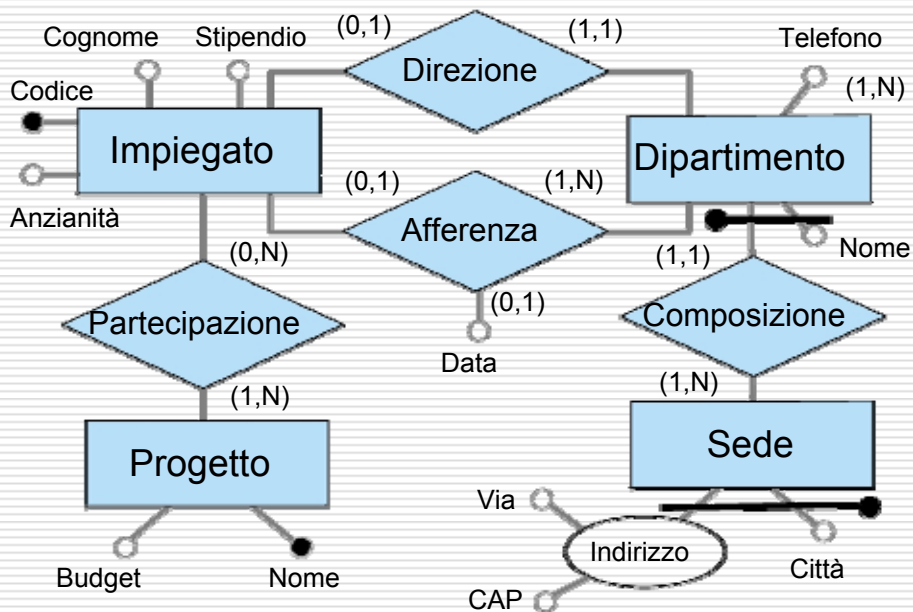
Un identificatore esterno può anche non comprendere attributi.



Una entità che partecipa con ruolo U e con cardinalità (1,1) ad una relazione R nella quale almeno un altro ruolo ha cardinalità massima 1, è implicitamente identificata esternamente da R tramite U. **Esempio:**



Esempio di vincoli di cardinalità e di identificaz.



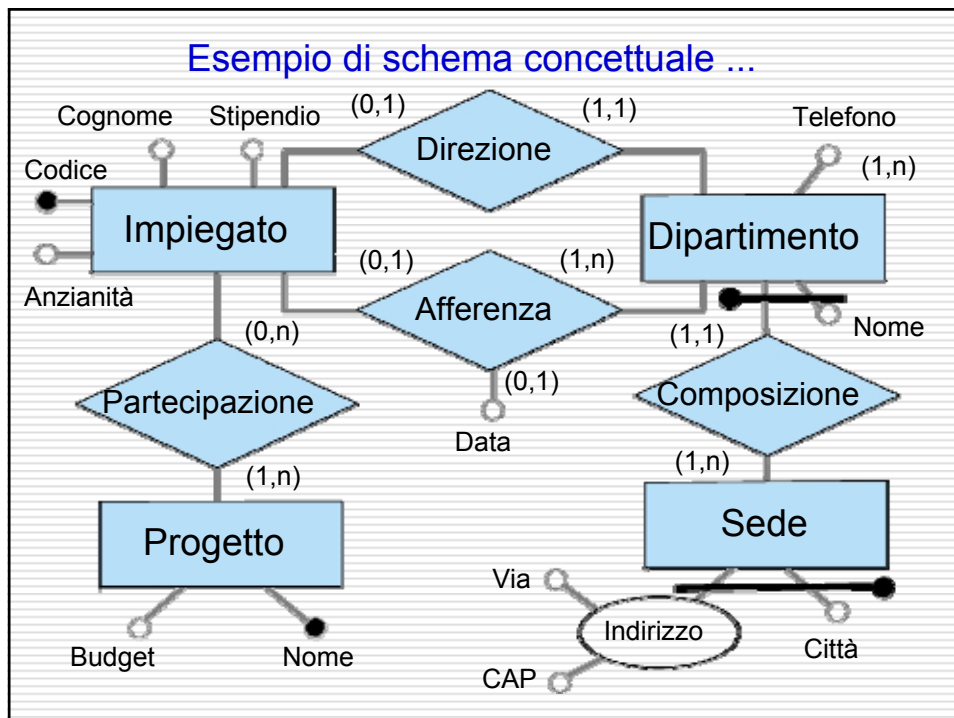
Vincoli non esprimibili nel diagramma ER

- 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?

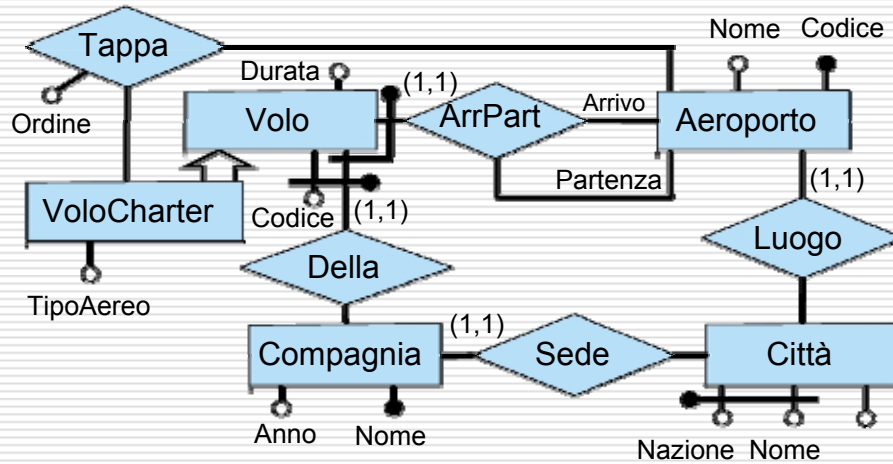
- attraverso formalismi opportuni (**es, in logica matematica**)
- attraverso delle asserzioni in linguaggio naturale (**che devono essere il più possibile precise e non ambigue**)



...con vincoli 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.

Vincoli sullo schema dell'esercizio 12



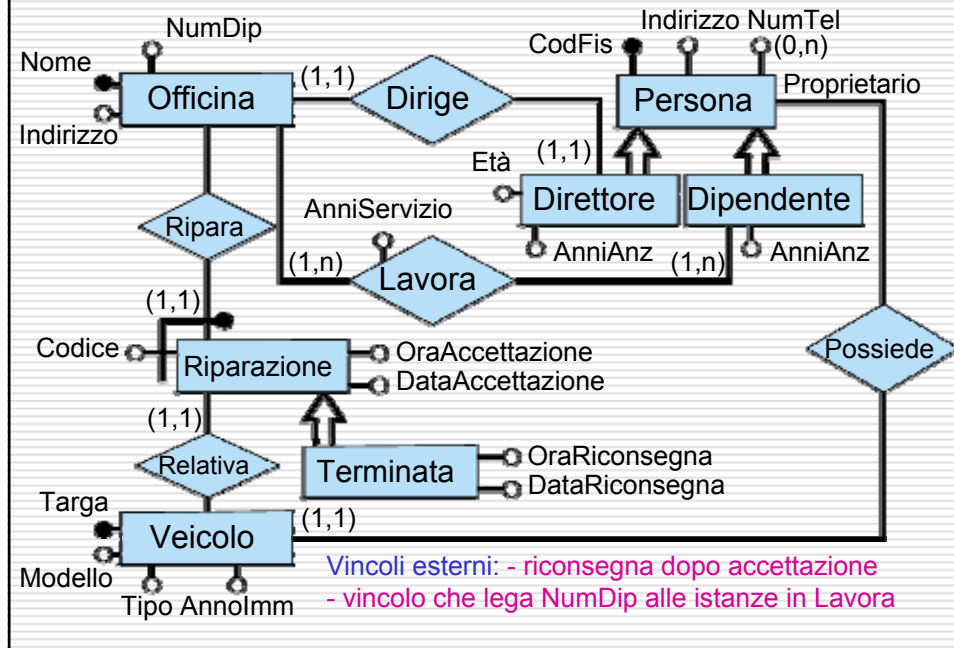
NumAbit **Vincolo esterno:** per ogni v in VoloCharter, se $(v, a_1), \dots, (v, a_n)$ sono tutte le coppie in Tappa alle quali partecipa v , e se o_1, \dots, o_n sono i valori assegnati a tali coppie dall'attributo Ordine, allora per $i=1, \dots, n$ esiste un o_j tale che $o_j=i$.

Esercizio 17: progettazione concettuale

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

Delle officine interessano: nome, indirizzo, numero di dipendenti, dipendenti (almeno uno) con l'informazione su quanti anni di servizio, e direttore. Si noti che ogni officina ha uno ed un solo direttore ed ogni 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 17: soluzione

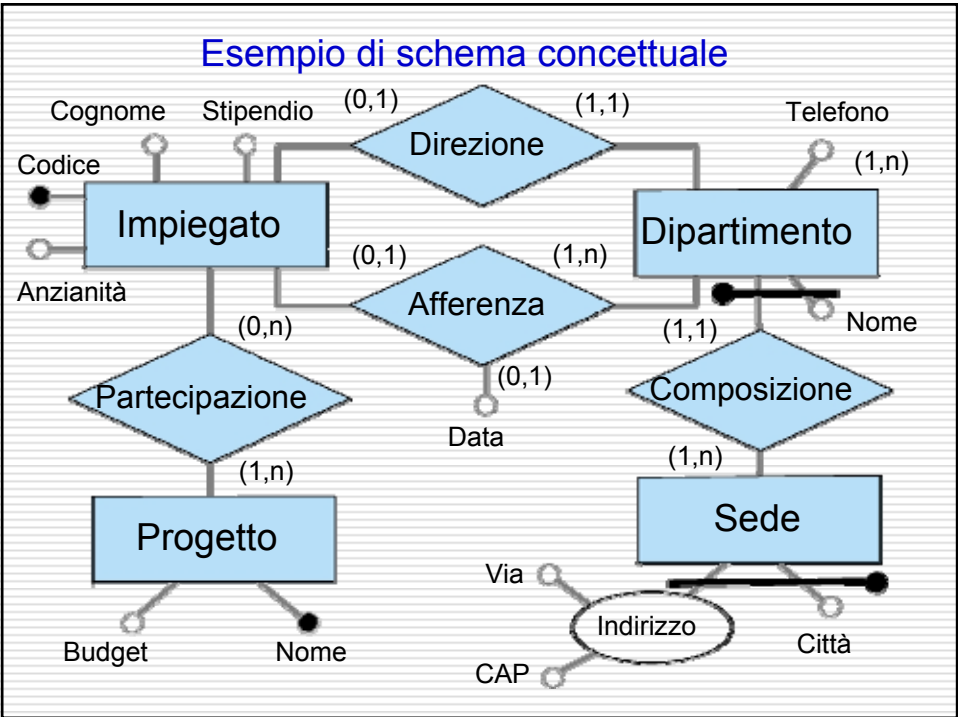


Documentazione associata agli schemi ER

Oltre al diagramma ER, lo schema concettuale è descritto dal cosiddetto **dizionario dei dati**.

Il **dizionario dei dati** è costituito dalle tabelle di:

- entità
- relazioni
- attributi (con i loro domini)
- vincoli esterni



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
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

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.