

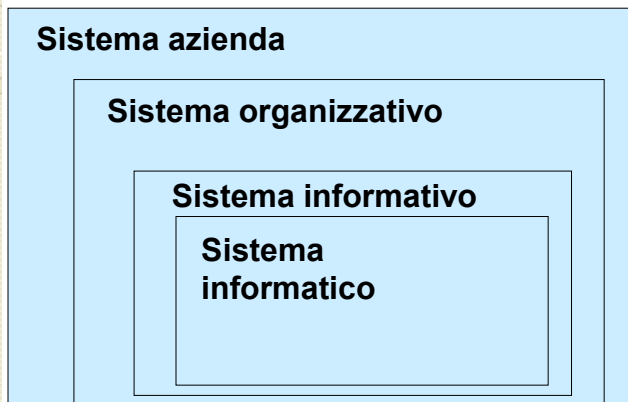
# Progettazione di basi di dati

## Tratto da:

Atzeni, Ceri, Fraternali, Paraboschi, Torlone  
Basi di dati *Quinta edizione*  
McGraw-Hill Education, 2018  
Capitolo 6: *Metodologie e modelli per il progetto*

1

## Sistema Informatico



- Ogni organizzazione è dotata di un **sistema informativo** che gestisce le sue informazioni di interesse
- Porzione automatizzata del sistema informativo: **Sistema informatico**
  - la parte del sistema informativo che gestisce informazioni con tecnologia informatica

2

2

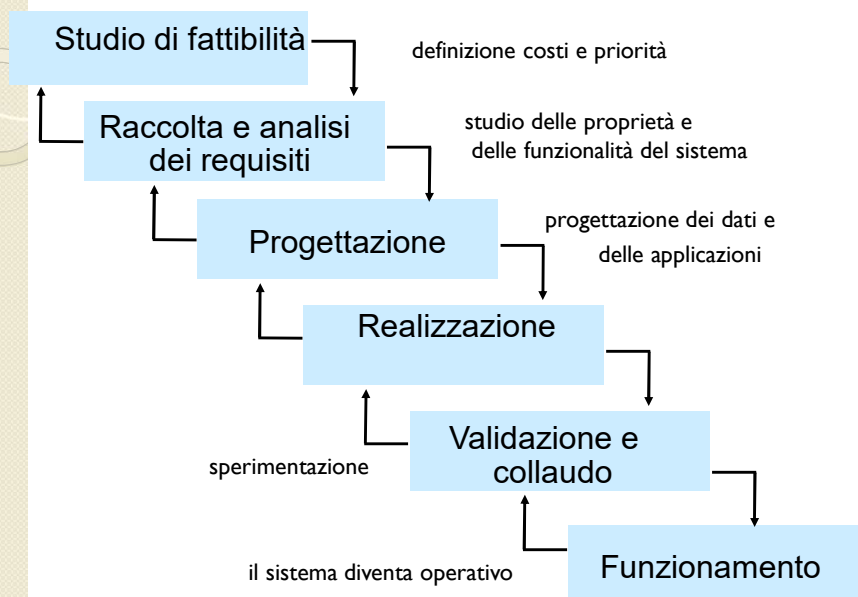
## Progettazione di basi di dati

- È una delle attività del processo di sviluppo dei sistemi informativi
- va quindi inquadrata in un contesto più generale:
- **il ciclo di vita dei sistemi informativi:**
  - Insieme e sequenzializzazione delle attività svolte da analisti, progettisti, utenti, nello sviluppo e nell'uso dei sistemi informativi
  - attività iterativa, quindi **ciclo**

3

3

## Fasi (tecniche) del ciclo di vita



4

4

La progettazione di un sistema informativo riguarda due aspetti:

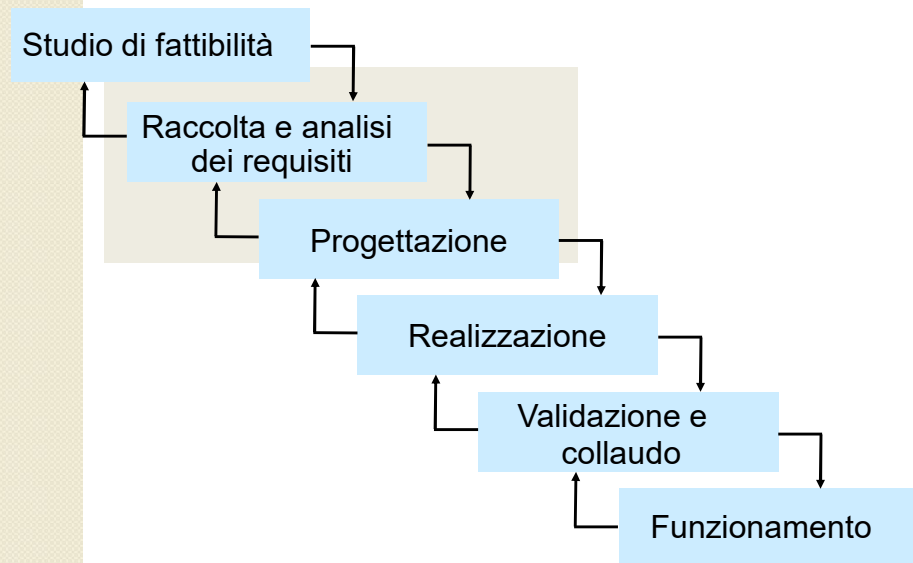
- ▶ **progettazione dei dati**
- progettazione delle applicazioni**

**Ma:**

- ▶ i dati hanno un ruolo centrale
  - i dati sono più stabili

5

5



6

6

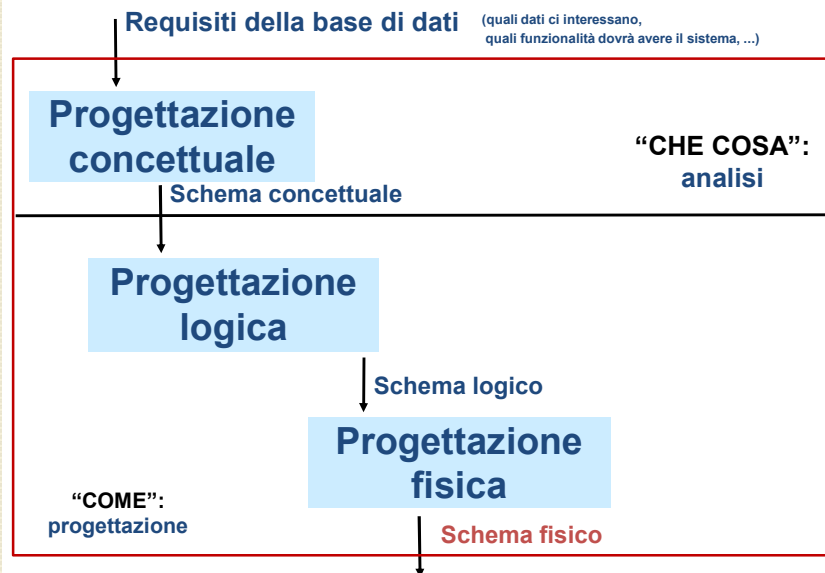
## Metodologie di progettazione di BD

- Per garantire prodotti di buona qualità è opportuno seguire una
  - **metodologia di progetto**, con:
    - articolazione delle attività in fasi (decomposizione dell'intera attività di progetto)
    - criteri di scelta in caso di alternative
    - modelli di riferimento per descrivere i dati di input e output delle varie fasi
    - generalità e facilità d'uso

7

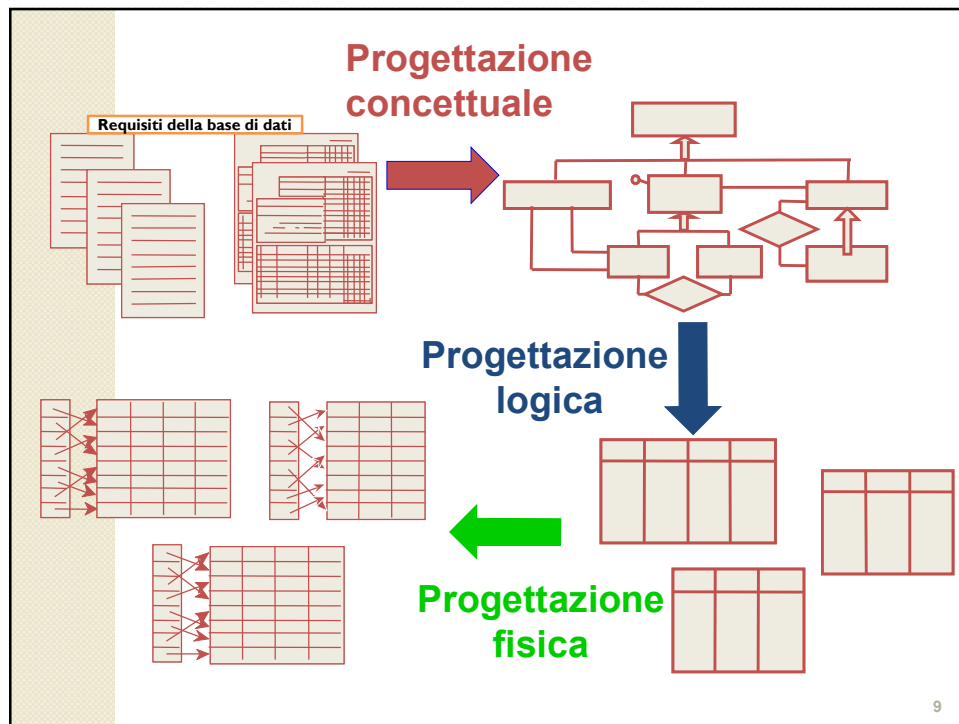
7

## Progettazione di una base di dati



8

8



9

## Modelli logici e modelli concettuali

- **modelli logici**: utilizzati nei DBMS esistenti per l'organizzazione dei dati
  - utilizzati dai programmi
  - indipendenti dalle strutture fisiche
 esempi: **relazionale**, reticolare, gerarchico, a oggetti
- **modelli concettuali**: permettono di rappresentare i dati in modo indipendente da ogni sistema
  - cercano di descrivere i concetti del mondo reale
  - sono utilizzati nelle fasi preliminari di progettazione
 il più noto è il modello **Entità-Relazione** (nel seguito verrà chiamato modello **Entity-Relationship** per non confondersi con la relazione del modello relazionale)

10

10

## Modelli concettuali, perché?

- servono per ragionare sulla realtà di interesse, indipendentemente dagli aspetti realizzativi
- permettono di rappresentare le classi di oggetti di interesse e le loro correlazioni
- prevedono efficaci rappresentazioni grafiche (utili anche per documentazione e comunicazione)

11

11

## Modello Entity-Relationship (Entità-Relazione)

- Il più diffuso modello concettuale
  - Ne esistono molte versioni,
  - (più o meno) diverse l'una dall'altra

12

12

## I costrutti del modello E-R

- Entità
- Relationship
- Attributo
- Identificatore
- Generalizzazione
- ....

13

13

## Entità

- Classe di oggetti (fatti, persone, cose) della realtà di interesse con proprietà comuni e con esistenza “autonoma”
- Esempi:
  - paziente, città, reparto, medico

**Paziente**

**Città**

**Reparto**

**Medico**

14

14

## Relationship

- Legame logico fra due o più entità, rilevante nell'applicazione di interesse
- Esempi:
  - Residenza (fra paziente e città)



15

15

## Entità: schema e istanza

- **Entità:**
  - classe di oggetti, persone, ... "omogenei"
- **Occorrenza** (o istanza) di entità:
  - elemento della classe (l'oggetto, la persona, ..., non i dati)
- nello schema concettuale rappresentiamo le **entità**, non le singole istanze ("astrazione")

16

16



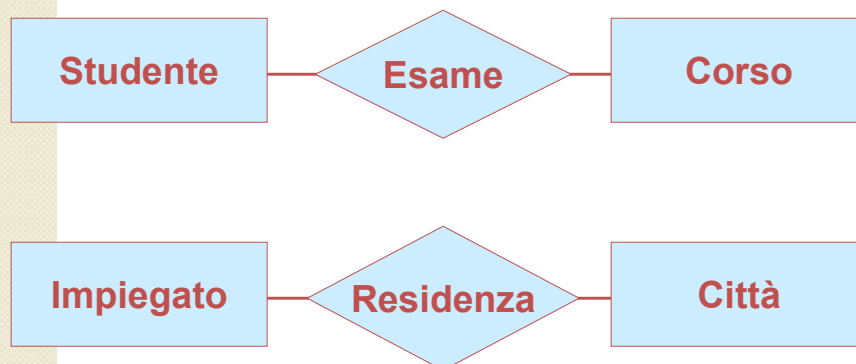
## Relationship

- Legame logico fra due o più entità, rilevante nell'applicazione di interesse
- Esempi:
  - Residenza (fra persona e città)
  - Frequenza (fra studente e corso)
- Chiamata anche:
  - relazione, correlazione, associazione

17

17

## Rappresentazione grafica di relationship



18

18

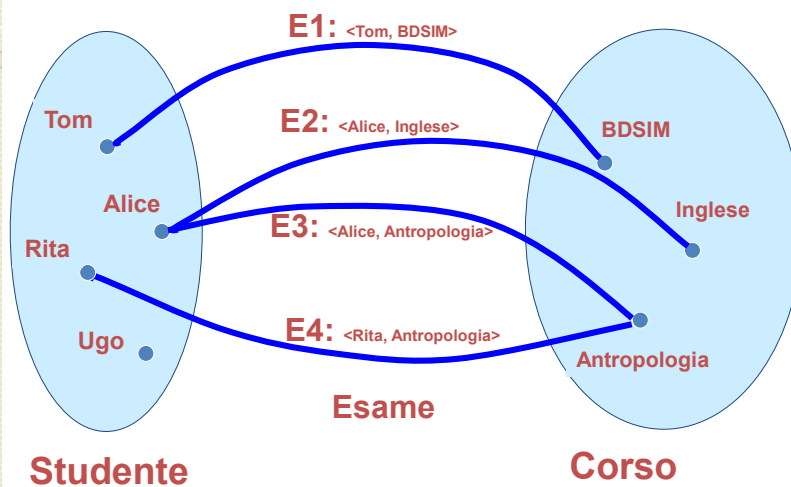
## Relationship, occorrenze

- Una occorrenza di una relationship binaria è coppia di occorrenze di entità, una per ciascuna entità coinvolta
- Una occorrenza di una relationship n-aria è una n-upla di occorrenze di entità, una per ciascuna entità coinvolta
- Nell'ambito di una relationship non ci possono essere occorrenze (coppie, ennuple) ripetute

19

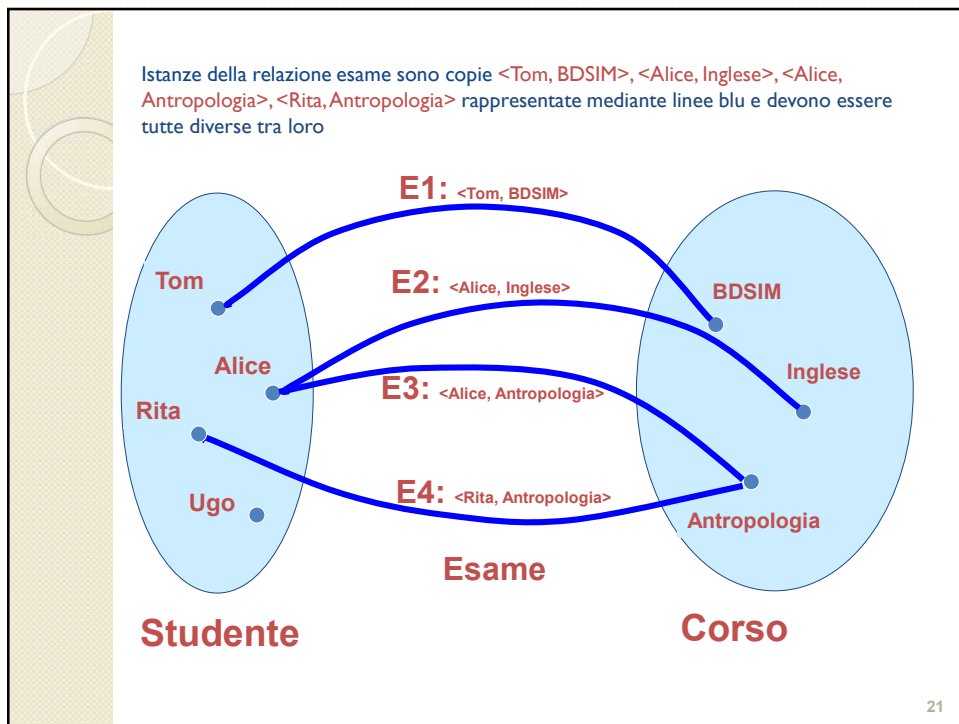
19

## Esempi di occorrenze

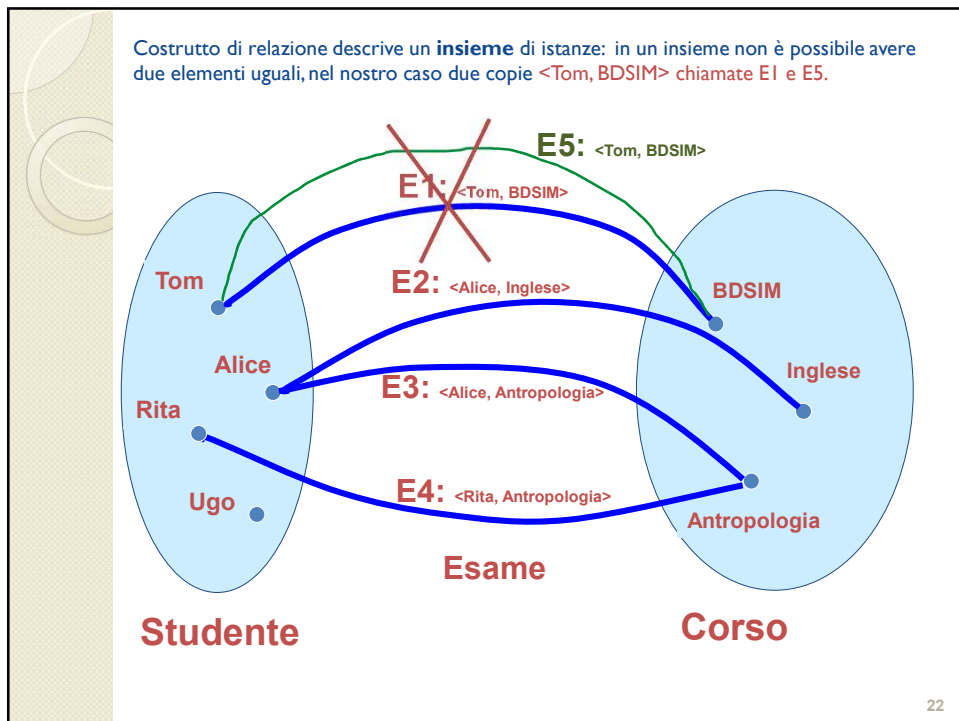


20

20

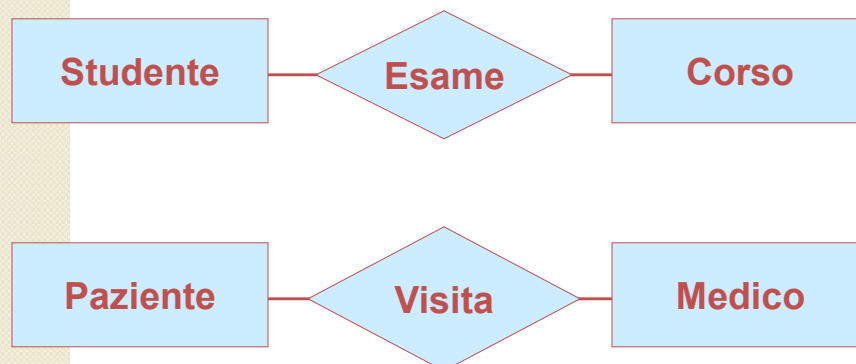


21



22

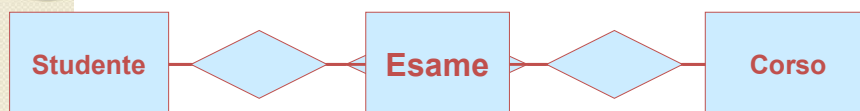
## Rappresentazioni corrette?



23

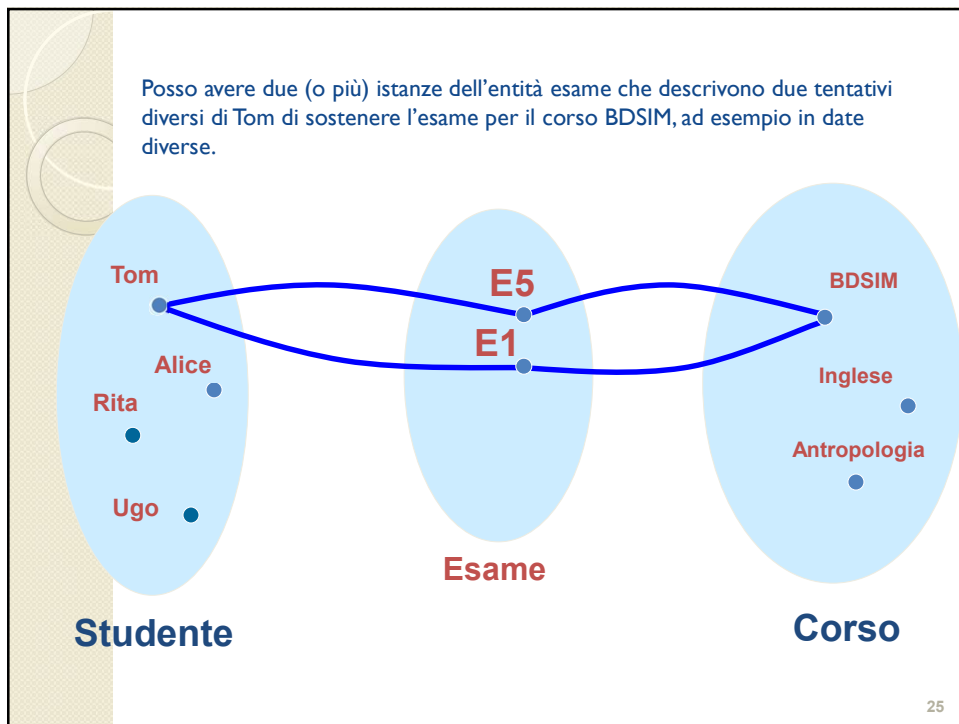
23

## "Promuoviamo" la relationship

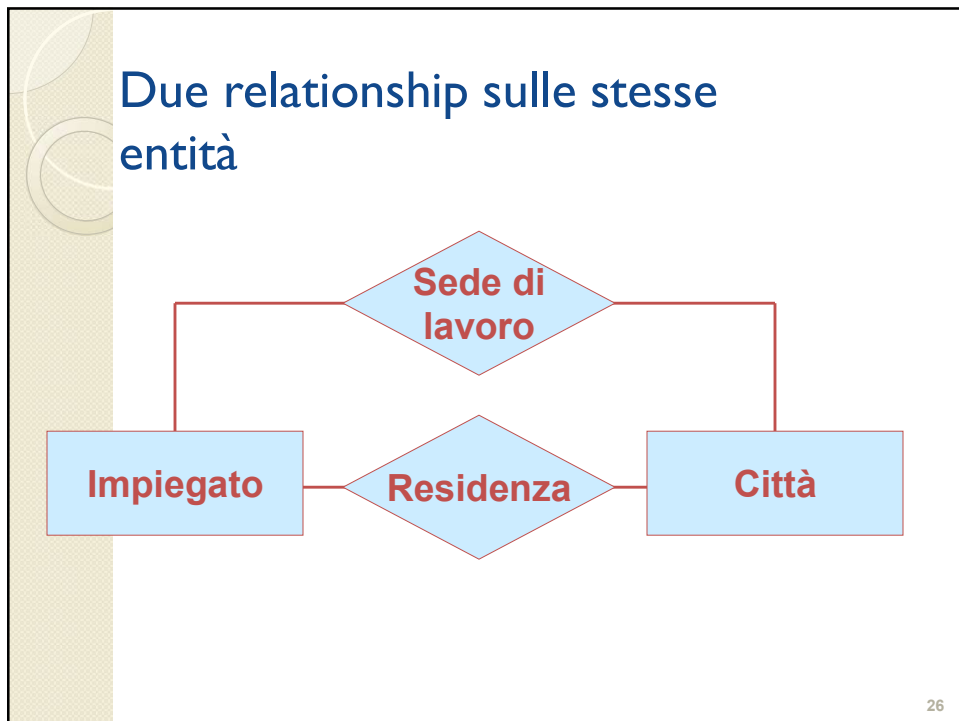


24

24

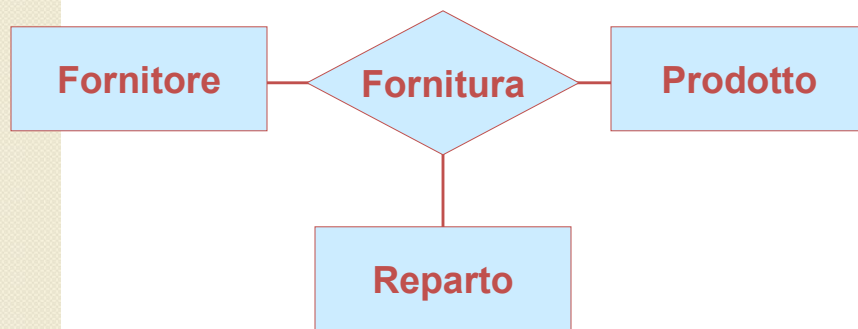


25



26

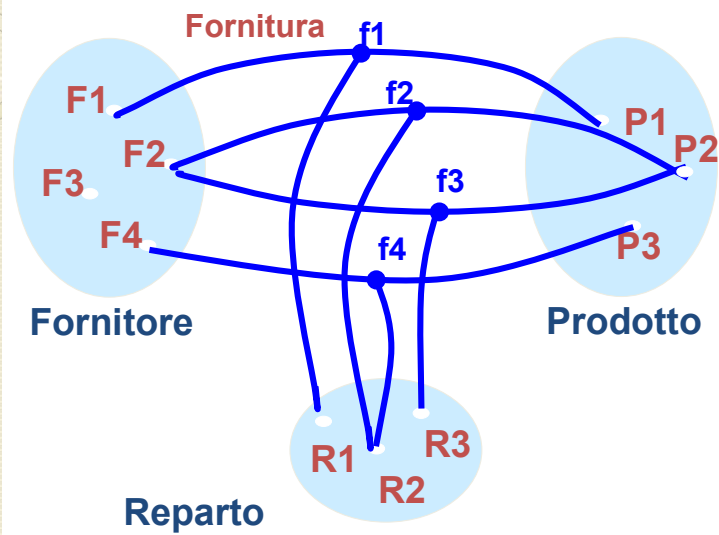
## Relationship n-aria



27

27

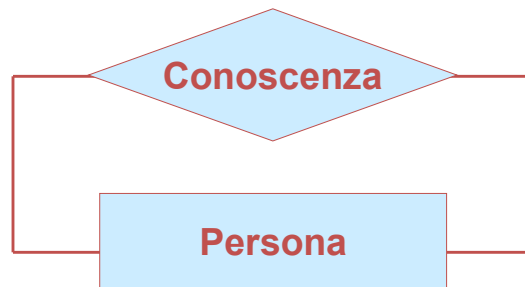
## Esempi di occorrenze



28

28

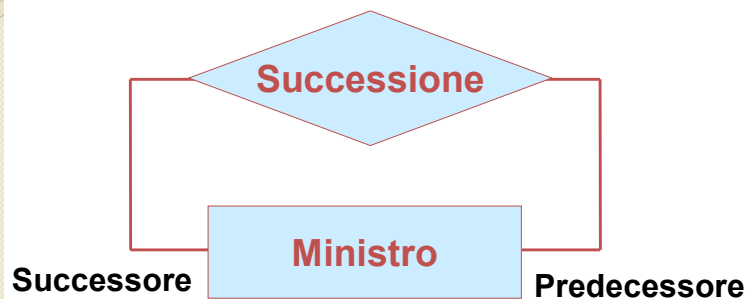
Relationship ricorsiva:  
coinvolge “due volte” la stessa entità



29

29

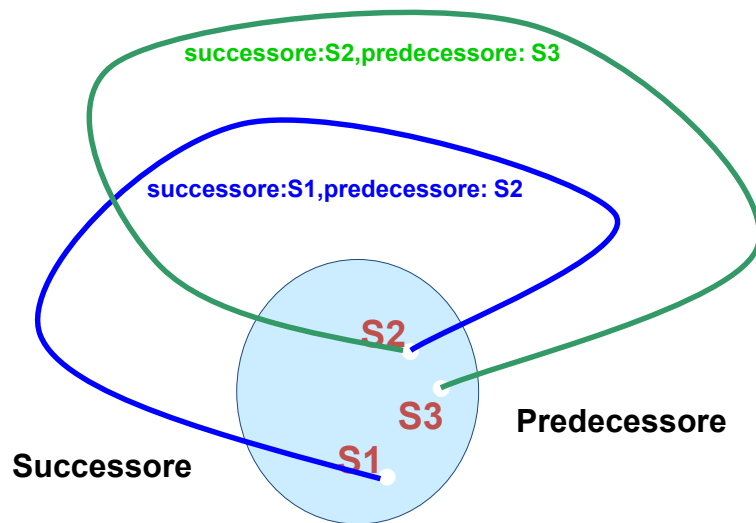
Relationship ricorsiva con “ruoli”



30

30

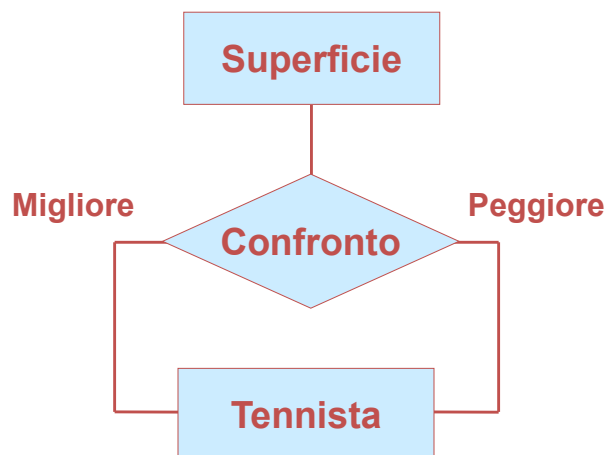
## Esempi di occorrenze



31

31

## Relationship ternaria ricorsiva

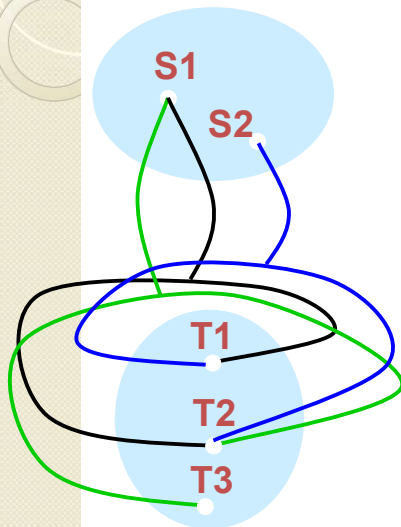


32

32



## Esempi di occorrenze



T1 è migliore di T2 su S2

T2 è migliore di T1 su S1

T3 è migliore di T2 su S1

33

33

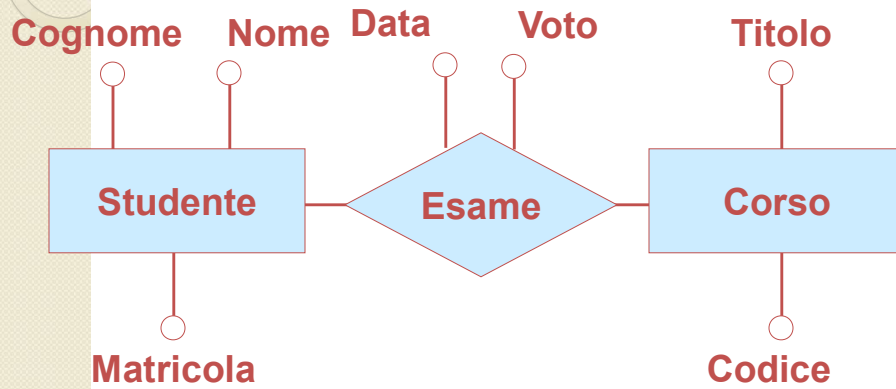
## Attributo

- Proprietà elementare di un'entità o di una relationship, di interesse ai fini dell'applicazione
- Associa ad ogni occorrenza di entità o relationship un valore appartenente a un insieme detto **dominio** dell'attributo

34

34

## Attributi, rappresentazione grafica



35

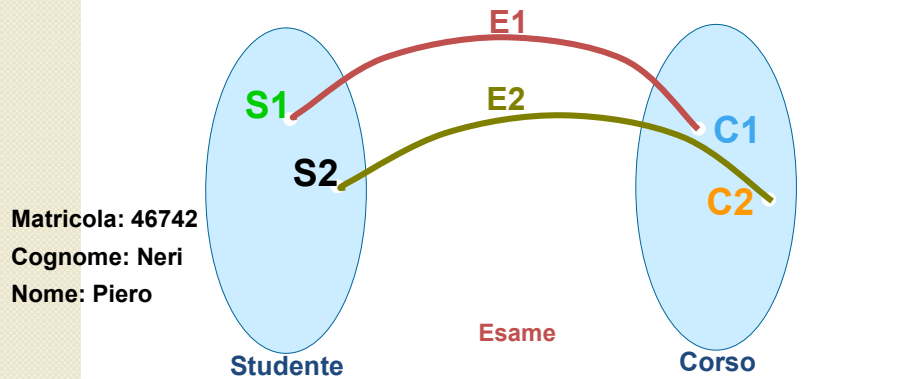
35

## Esempi di occorrenze

Matricola: 34567  
Cognome: Rossi  
Nome: Mario

Data: 25/07/2022  
Voto: 26

Codice: Inf205  
Titolo: Basi di dati



36

36

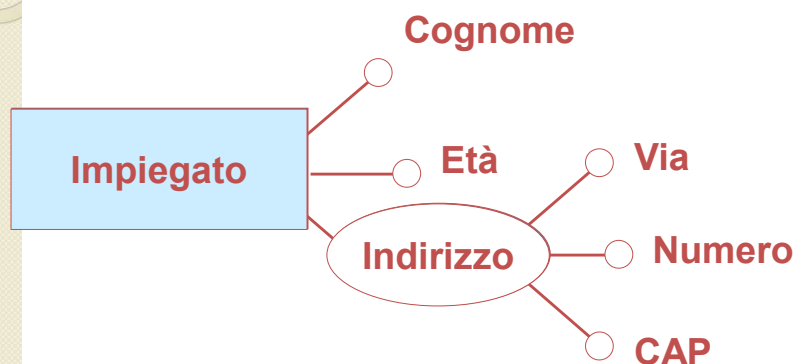
## Attributi composti

- Raggruppano attributi di una medesima entità o relationship che presentano affinità nel loro significato o uso
- Esempio:
  - Via, Numero civico e CAP formano un Indirizzo

37

37

## Rappresentazione grafica



38

38

## Altri costrutti del modello E-R

- **Cardinalità**
  - di relationship
  - di attributo
- **Identificatore**
  - interno
  - esterno
- **Generalizzazione**

39

39

## Cardinalità di relationship

- Coppia di valori associati a ogni entità che partecipa a una relationship
- specificano il numero minimo e massimo di occorrenze delle relationship cui ciascuna occorrenza di una entità può partecipare

40

40

## Esempio di cardinalità



41

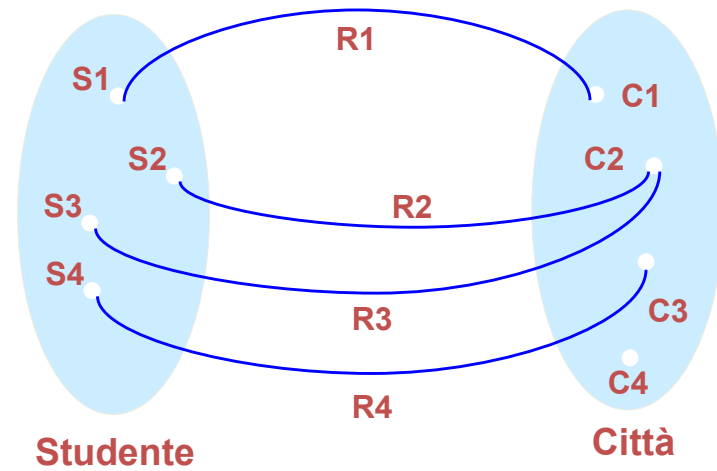
41

- per semplicità usiamo solo tre simboli:
- 0 e 1 per la cardinalità minima:
  - 0 = “partecipazione **opzionale**”
  - 1 = “partecipazione **obbligatoria**”
- 1 e “N” per la massima:
  - “N” non pone alcun limite

42

42

## Occorrenze di Residenza



43

43

## Cardinalità di Residenza



44

44

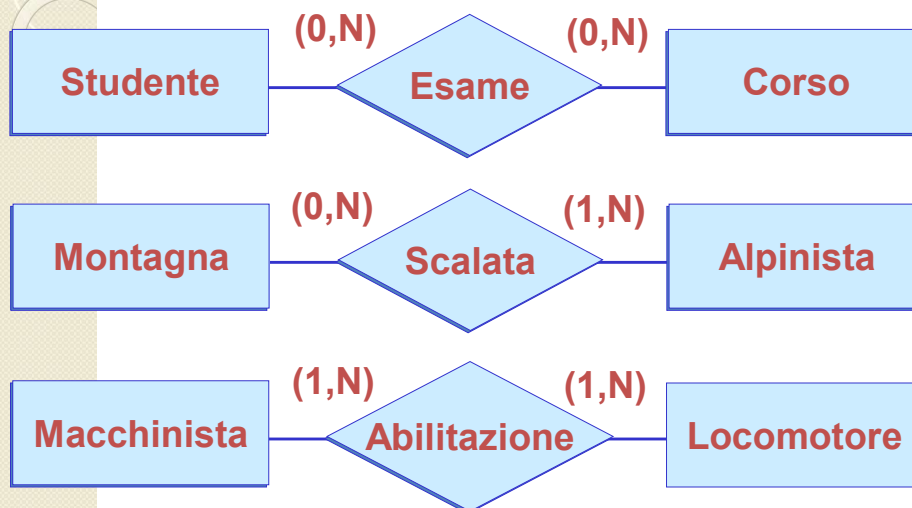
## Tipi di relationship

- Con riferimento alle cardinalità **massime**, abbiamo relationship:
  - uno a uno
  - uno a molti
  - molti a molti

45

45

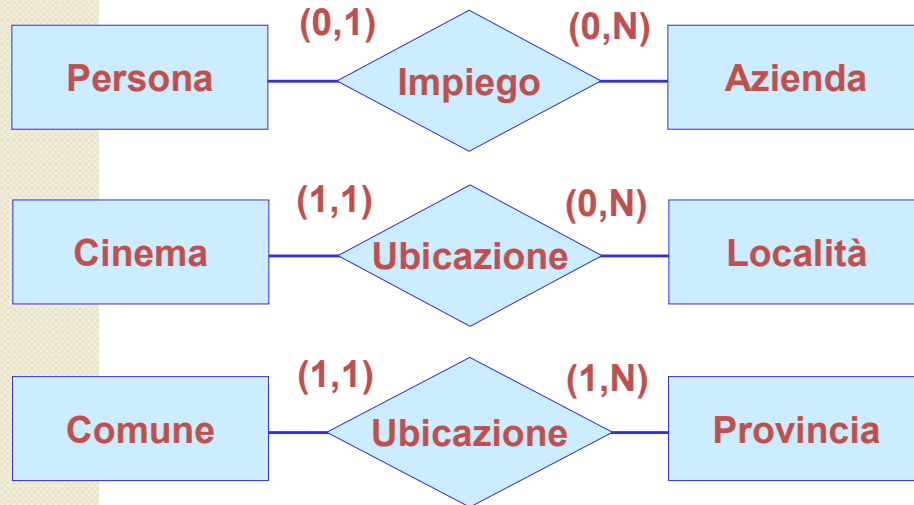
## Relationship “molti a molti”



46

46

## Relationship “uno a molti”



47

47

## Due avvertenze

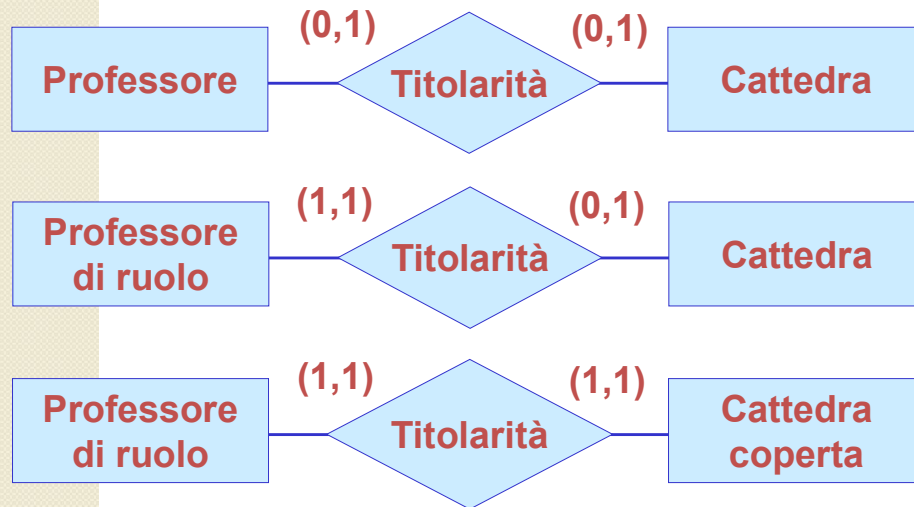
- Attenzione al "verso" nelle relationship uno a molti
- le relationship obbligatorie-obbligatorie sono molto rare

48

48



### Relationship “uno a uno”



49

49

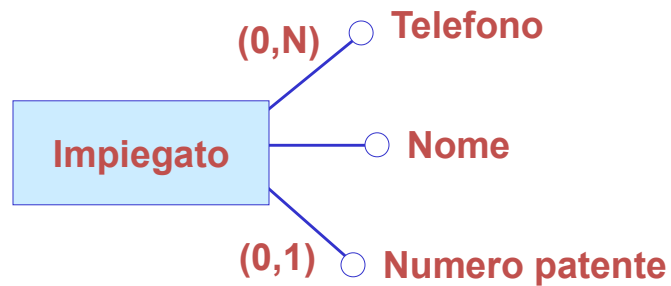
### Cardinalità di attributi

- E' possibile associare delle cardinalità anche agli attributi, con due scopi:
  - indicare opzionalità ("informazione incompleta")
  - indicare attributi multivalore

50

50

## Rappresentazione grafica



51

51

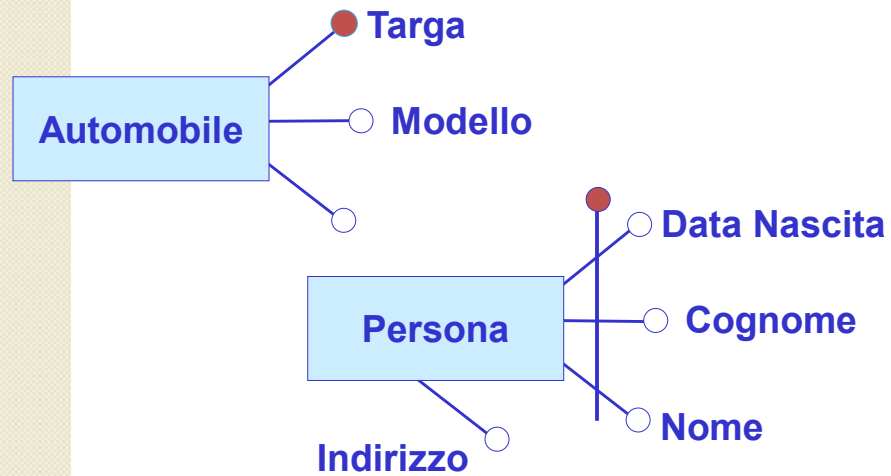
## Identificatore di una entità

- “strumento” per l'identificazione univoca delle occorrenze di un'entità
- costituito da:
  - attributi dell'entità
    - **identificatore interno**
  - (attributi +) entità esterne attraverso relationship
    - **identificatore esterno**

52

52

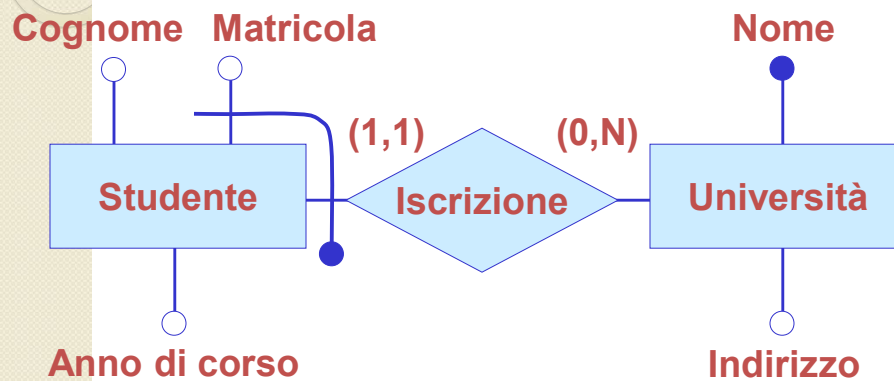
## Identificatori interni



53

53

## Identificatore esterno



54

54

## Alcune osservazioni

- ogni entità deve possedere almeno un identificatore, ma può averne in generale più di uno
- una identificazione esterna è possibile solo attraverso una relationship a cui l'entità da identificare partecipa con cardinalità (1,1)
- perché non parliamo degli identificatori delle relationship?

55

55

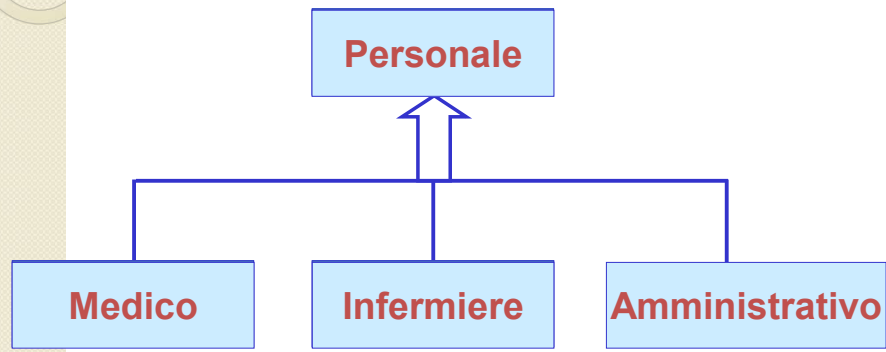
## Generalizzazione

- mette in relazione una o più entità  $E_1, E_2, \dots, E_n$  con una entità  $E$ , che le comprende come casi particolari
  - $E$  è **generalizzazione** di  $E_1, E_2, \dots, E_n$
  - $E_1, E_2, \dots, E_n$  sono **specializzazioni** (o sottotipi) di  $E$

56

56

## Rappresentazione grafica



57

57

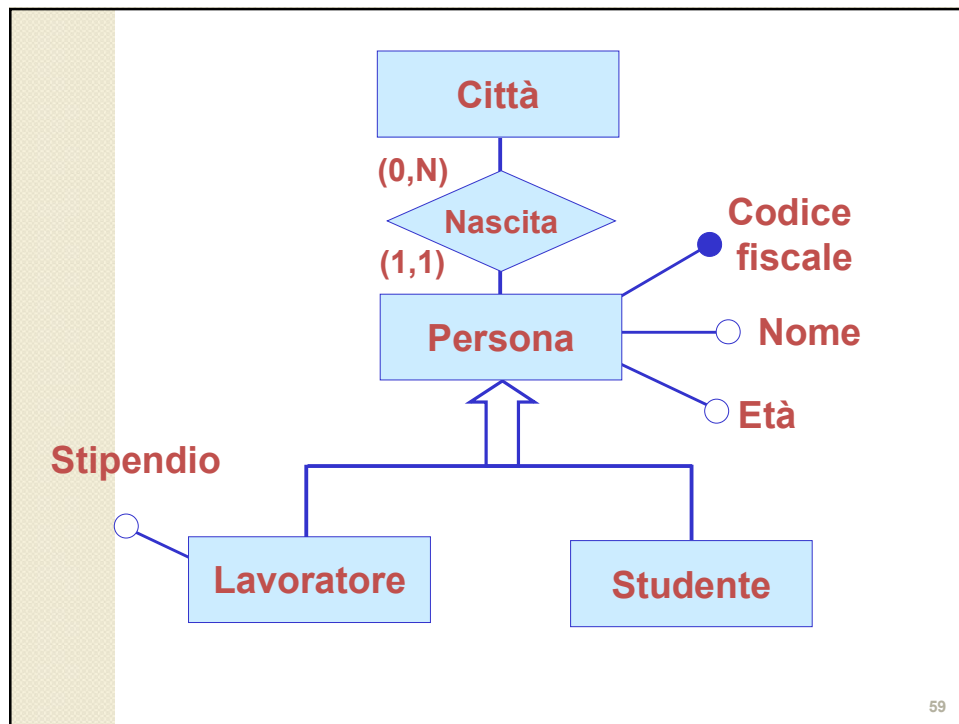
## Proprietà delle generalizzazioni

Se E (genitore) è generalizzazione di  $E_1, E_2, \dots, E_n$  (figlie):

- ogni proprietà di E è significativa per  $E_1, E_2, \dots, E_n$
- ogni occorrenza di  $E_1, E_2, \dots, E_n$  è occorrenza anche di E

58

58



59

## Ereditarietà

- tutte le proprietà (attributi, relationship, altre generalizzazioni) dell'entità genitore vengono **ereditate** dalle entità figlie e non rappresentate esplicitamente

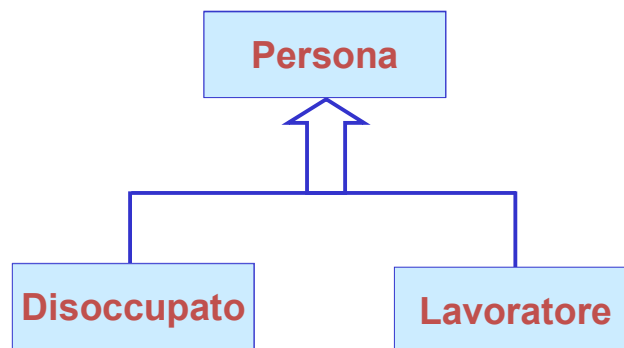
60

## Tipi di generalizzazioni

- **totale** se ogni occorrenza dell'entità genitore è occorrenza di almeno una delle entità figlie, altrimenti è **parziale**
- **esclusiva** se ogni occorrenza dell'entità genitore è occorrenza di al più una delle entità figlie, altrimenti è **sovrapposta**
- consideriamo (senza perdita di generalità) solo generalizzazioni esclusive e distinguiamo fra totali e parziali

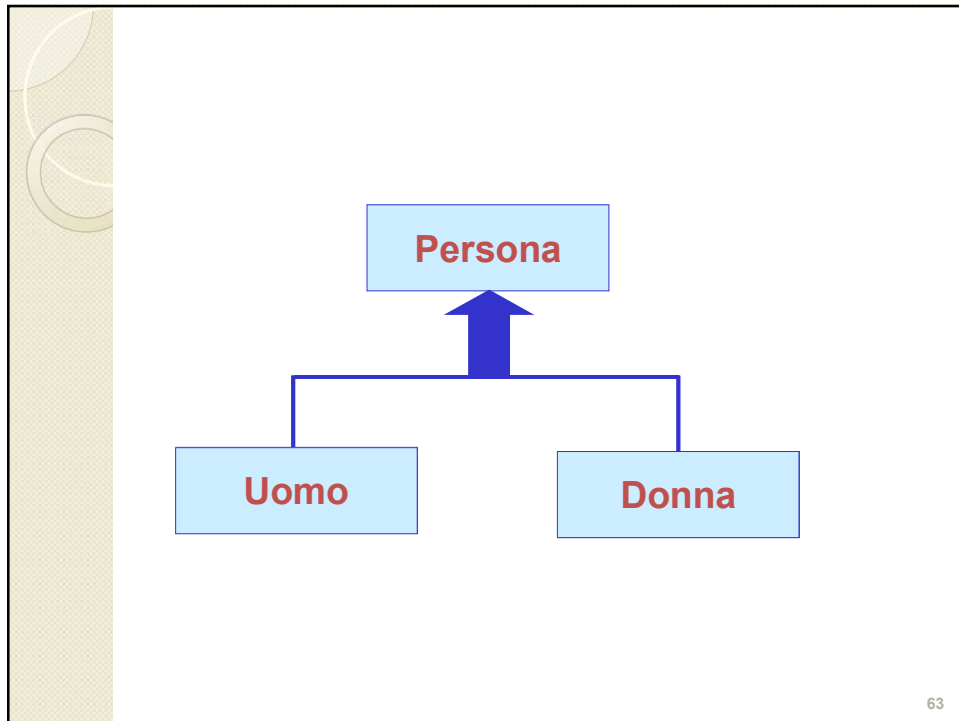
61

61



62

62



63

## Altre proprietà

- possono esistere gerarchie a più livelli e multiple generalizzazioni allo stesso livello
- un'entità può essere inclusa in più gerarchie, come genitore e/o come figlia
- se una generalizzazione ha solo un'entità figlia si parla di **sottoinsieme**
- alcune configurazioni non hanno senso
- il genitore di una generalizzazione totale può non avere identificatore, purché ...

64



## Esercizio

- Le persone hanno CF, cognome ed età; gli uomini anche la posizione militare; gli impiegati hanno lo stipendio e possono essere segretari, direttori o progettisti (un progettista può essere anche responsabile di progetto); gli studenti (che non possono essere impiegati) un numero di matricola; esistono persone che non sono né impiegati né studenti (ma i dettagli non ci interessano)

65

65

66

66