



# Sistemi informativi

## Unità 5 Progettazione di basi di dati

# Progettazione di basi di dati

- Modello Entità-Relazione
- Progettazione concettuale
- Progettazione logica
- Normalizzazione



# Progettazione di basi di dati

## Modello Entità - Relazione

# Modello Entità-Relazione

- Ciclo di vita di un sistema informativo
- Progettazione di basi di dati
- Entità e relazioni
- Attributi
- Identificatori
- Generalizzazione
- Documentazione di schemi E-R
- UML ed E-R



# Modello Entità-Relazione

Ciclo di vita di un sistema  
informativo

# Progettazione di basi di dati

- La progettazione di una base di dati è una delle attività del processo di sviluppo di un sistema informativo
- va inquadrata nel contesto più ampio di ciclo di vita di un sistema informativo

# Ciclo di vita di un sistema informativo

Studio di fattibilità

# Ciclo di vita di un sistema informativo

## ➤ Studio di fattibilità

- determinazione dei costi delle diverse alternative e delle priorità di realizzazione delle componenti del sistema



# Ciclo di vita di un sistema informativo

Studio di fattibilità



Raccolta e analisi  
dei requisiti

# Ciclo di vita di un sistema informativo

- Raccolta e analisi dei requisiti
  - definizione delle proprietà e delle funzionalità del sistema informativo
  - richiede interazione con l'utente
  - produce una descrizione completa, ma informale del sistema da realizzare

# Ciclo di vita di un sistema informativo



# Ciclo di vita di un sistema informativo

## ➤ Progettazione

- suddivisa in progettazione dei dati e delle applicazioni
- produce descrizioni formali

# Ciclo di vita di un sistema informativo



# Ciclo di vita di un sistema informativo

## ➤ Implementazione

- realizzazione del sistema informativo secondo le caratteristiche definite nella fase di progettazione

# Ciclo di vita di un sistema informativo



# Ciclo di vita di un sistema informativo

## ➤ Validazione e testing

- verifica del corretto funzionamento e della qualità del sistema informativo
- può portare a modifiche dei requisiti o revisione del progetto



# Ciclo di vita di un sistema informativo



# Ciclo di vita di un sistema informativo

## ➤ Funzionamento

- operatività del sistema
- richiede operazioni di gestione e manutenzione

# Ciclo di vita di un sistema informativo

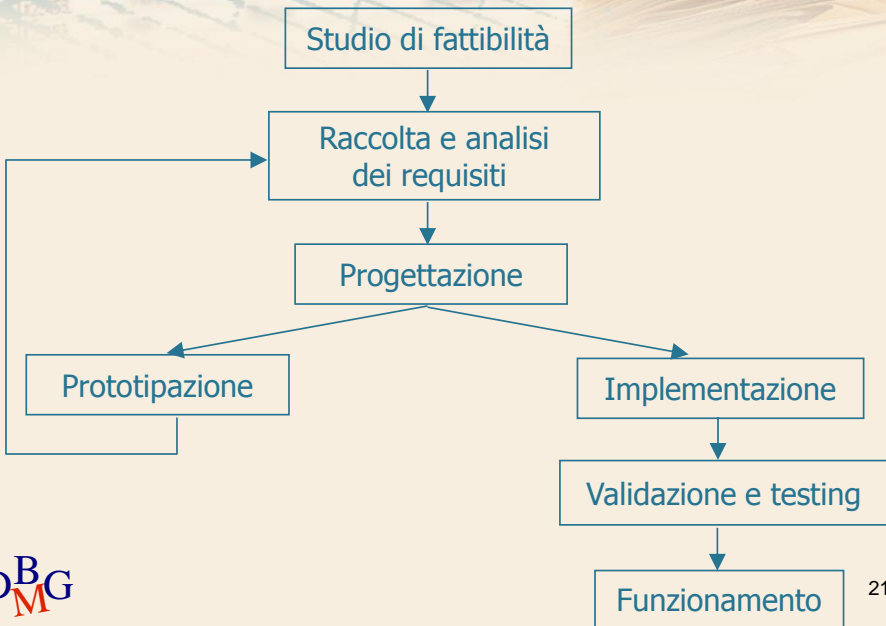


# Ciclo di vita di un sistema informativo

## ➤ Prototipazione

- realizzazione rapida di una versione semplificata del sistema per valutarne le caratteristiche
- può portare a modifiche dei requisiti o revisione del progetto

# Ciclo di vita di un sistema informativo



# Ciclo di vita di un sistema informativo





# Modello Entità-Relazione

Progettazione di basi di dati

# Progettazione di una base di dati

- La base di dati costituisce un componente importante del sistema complessivo
- Metodologia di progettazione basata sui dati
  - la progettazione della base di dati precede la progettazione delle applicazioni che la utilizzano
  - attenzione maggiore alla fase di progettazione rispetto alle altre fasi



# Metodologia di progettazione

- Una metodologia di progettazione consiste in
- decomposizione dell'attività di progetto in passi successivi indipendenti tra loro
  - strategie da seguire nei vari passi e criteri per la scelta delle strategie
  - modelli di riferimento per descrivere i dati d'ingresso e di uscita delle varie fasi

# Metodologia di progettazione: Esempio

- Preparazione atletica
  - decomposizione dell'attività
    1. forma fisica
    - 2a. potenziamento
    - 2b. velocità

# Metodologia di progettazione: Esempio

## ➤ Preparazione atletica

- decomposizione dell'attività
- strategie da seguire nei vari passi
  1. A) dieta alimentare  
B) esercizi per ridurre la percentuale di grasso
  - 2a. A) esercizi con pesi  
B) esercizi di resistenza

# Metodologia di progettazione: Esempio

## ➤ Preparazione atletica

- decomposizione dell'attività
- strategie da seguire nei vari passi
- modelli di riferimento per descrivere i dati d'ingresso e di uscita delle varie fasi
  1. dati d'ingresso: peso attuale, % di grasso corporeo  
dati di uscita: modello della struttura corporea della persona in forma
  - 2a. dati di ingresso: modello di persona in forma  
dati di uscita: modello della struttura corporea dell'atleta medio

# Proprietà della metodologia

## ➤ Generalità

- possibilità di utilizzo indipendentemente dal problema e dagli strumenti a disposizione

## ➤ Qualità del risultato

- in termini di correttezza, completezza ed efficienza rispetto alle risorse utilizzate

## ➤ Facilità d'uso

- sia delle strategie che dei modelli di riferimento

# Progettazione basata sui dati

➤ Per le basi di dati, metodologia basata sulla separazione delle decisioni

- *cosa* rappresentare nella base di dati
  - progettazione concettuale
- *come* rappresentarlo
  - progettazione logica e fisica

# Fasi della progettazione di basi di dati

Requisiti  
applicazione



## Requisiti applicazione

- Specifiche informali della realtà di interesse
  - proprietà dell'applicazione
  - funzionalità dell'applicazione



# Fasi della progettazione di basi di dati

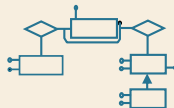
Requisiti  
applicazione



Progettazione  
concettuale



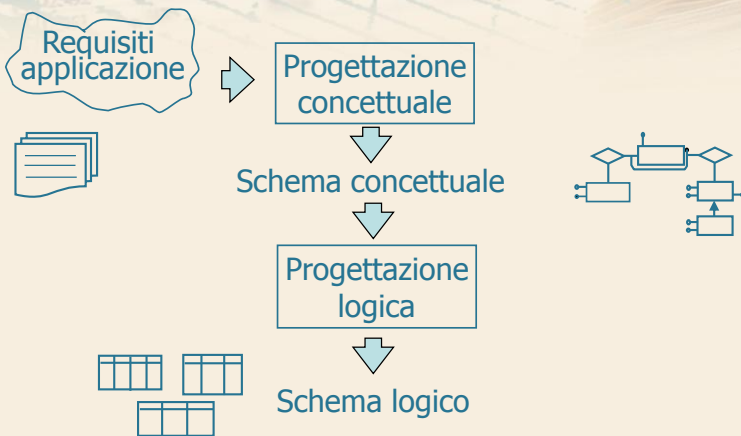
Schema concettuale



# Progettazione concettuale

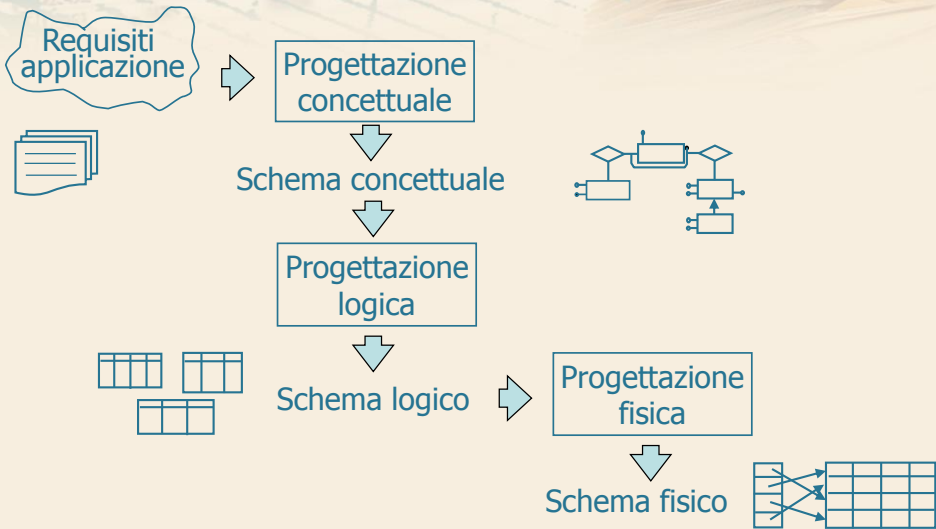
- Rappresentazione delle specifiche informali sotto forma di *schema concettuale*
- descrizione formale e completa, che fa riferimento ad un modello concettuale
  - indipendenza dagli aspetti implementativi (modello dei dati)
  - obiettivo è la rappresentazione del *contenuto informativo* della base di dati

# Fasi della progettazione di basi di dati



- Traduzione dello schema concettuale nello schema logico
- fa riferimento al modello logico dei dati prescelto
  - si usano criteri di ottimizzazione delle operazioni da fare sui dati
  - qualità dello schema verificata mediante tecniche formali (normalizzazione)

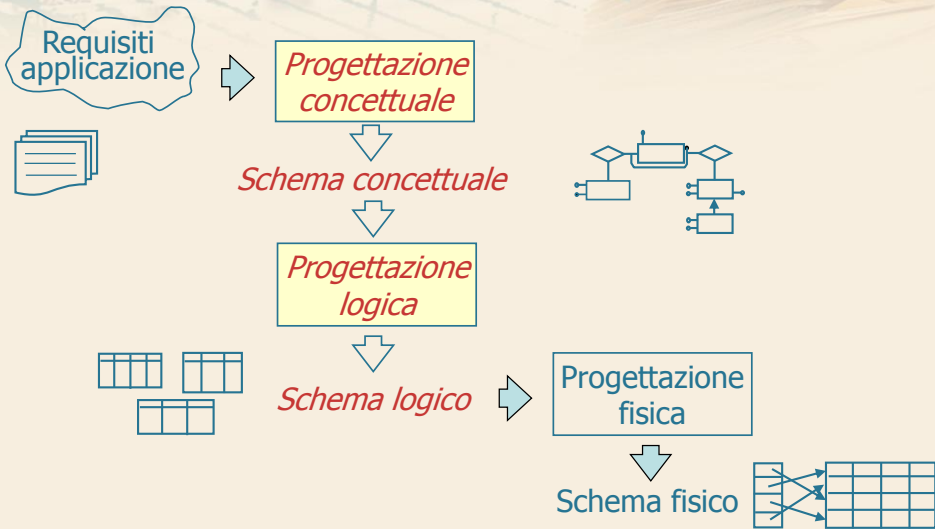
# Fasi della progettazione di basi di dati



## Progettazione fisica

- Specifica dei parametri fisici di memorizzazione dei dati (organizzazione dei file e degli indici)
  - produce un modello fisico, che dipende dal DBMS prescelto

# Fasi della progettazione di basi di dati





# Modello Entità-Relazione

Entità e relazioni



# Il modello E-R (Entity-Relationship)

- È il modello concettuale più diffuso
- Fornisce costrutti per descrivere le specifiche sulla struttura dei dati
  - in modo semplice e comprensibile
  - con un formalismo grafico
  - in modo indipendente dal modello dei dati, che può essere scelto in seguito
- Ne esistono numerose varianti

# Costrutti principali del modello E-R

- Entità
- Relazioni
- Attributi
- Identificatori
- Generalizzazioni e sottoinsiemi

Nome entità

- Rappresenta classi di oggetti del mondo reale (persone, cose, eventi, ...), che hanno
  - proprietà comuni
  - esistenza autonoma
- Esempi: dipendente, studente, articolo
- Un'occorrenza di un'entità è un oggetto della classe che l'entità rappresenta



Nome relazione

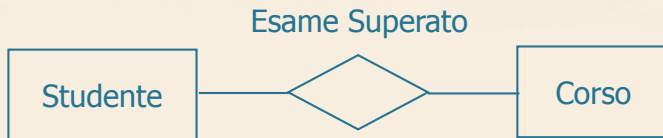
- Rappresenta un legame logico tra due o più entità
- Esempi: esame tra studente e corso, residenza tra persona e comune
- Da non confondere con la relazione del modello relazionale
  - a volte indicata con il termine associazione

# Esempi di relazioni

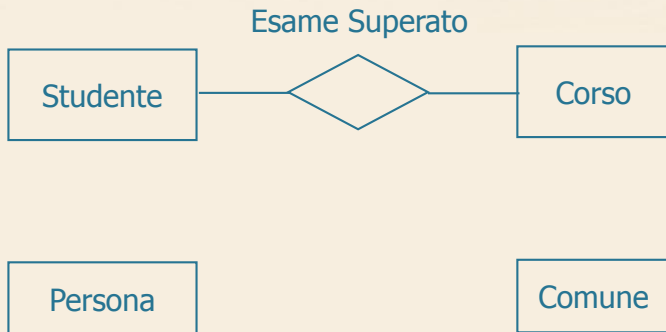
Studente

Corso

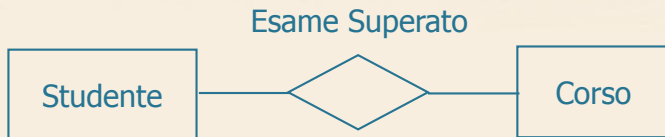
# Esempi di relazioni



# Esempi di relazioni

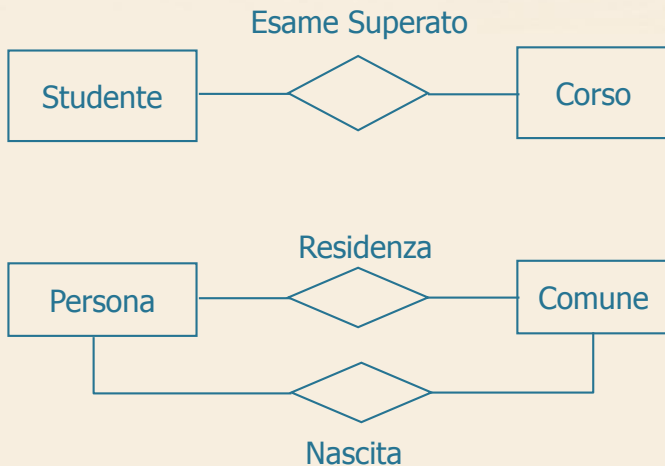


# Esempi di relazioni



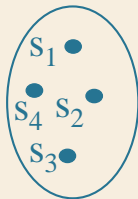


# Esempi di relazioni

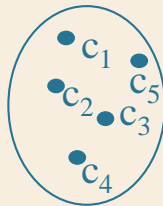


# Occorrenze di una relazione

Studente

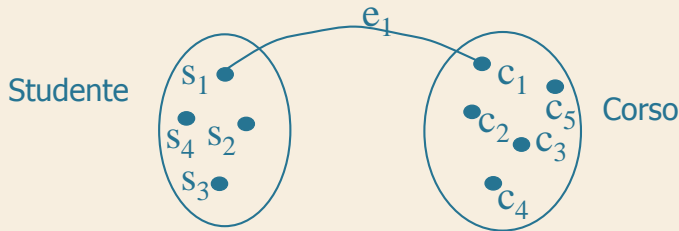


Corso



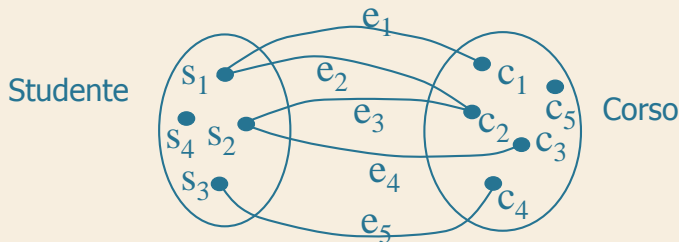
# Occorrenze di una relazione

- Un'occorrenza di una relazione è una n-upla (coppia nel caso di relazione binaria) costituita da occorrenze di entità, una per ciascuna delle entità coinvolte



## Occorrenze di una relazione

- Un'occorrenza di una relazione è una n-upla (coppia nel caso di relazione binaria) costituita da occorrenze di entità, una per ciascuna delle entità coinvolte
- Non vi possono essere n-uple identiche

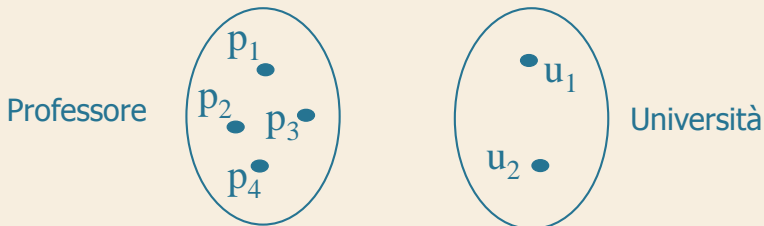


# Cardinalità delle relazioni binarie

- Sono specificate per ogni entità che partecipa ad una relazione
- Descrivono numero minimo e massimo di occorrenze di una relazione a cui può partecipare una occorrenza di un'entità
  - minimo assume i valori
    - 0 (partecipazione opzionale)
    - 1 (partecipazione obbligatoria)
  - massimo varia tra
    - 1 (al più una occorrenza)
    - N (numero arbitrario di occorrenze)

# Cardinalità delle relazioni binarie

➤ Corrispondenza 1 a 1



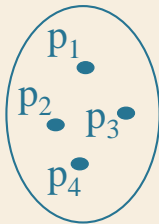
# Cardinalità delle relazioni binarie

➤ Corrispondenza 1 a 1

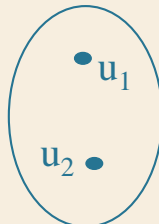
Professore

Università

Professore

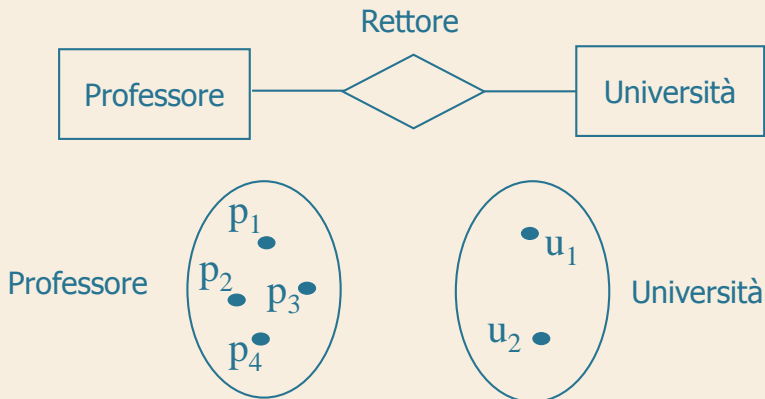


Università



# Cardinalità delle relazioni binarie

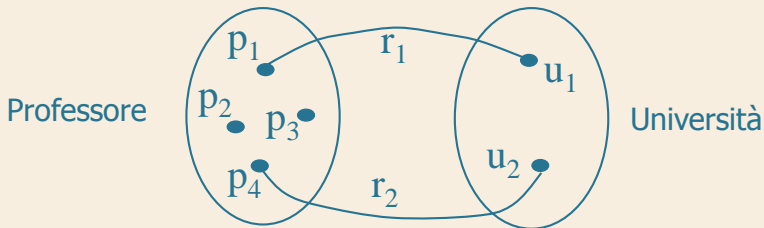
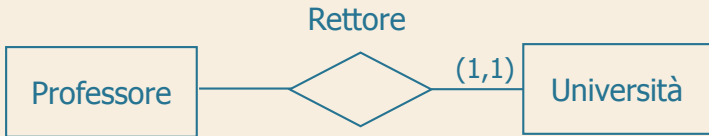
➤ Corrispondenza 1 a 1





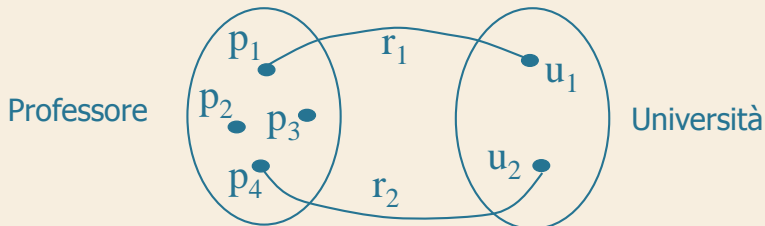
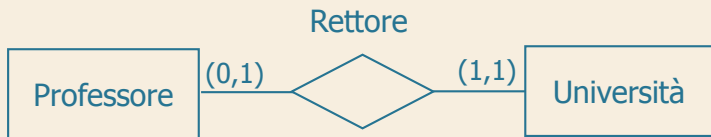
# Cardinalità delle relazioni binarie

➤ Corrispondenza 1 a 1



# Cardinalità delle relazioni binarie

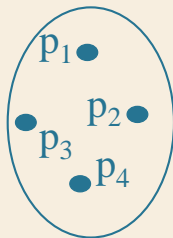
➤ Corrispondenza 1 a 1



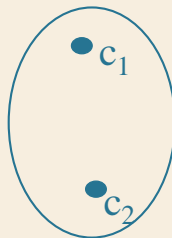
# Cardinalità delle relazioni binarie

➤ Corrispondenza 1 a N

Persona



Comune



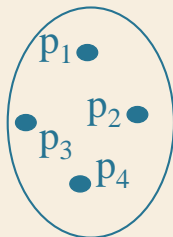
# Cardinalità delle relazioni binarie

➤ Corrispondenza 1 a N

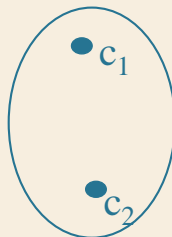
Persona

Comune

Persona

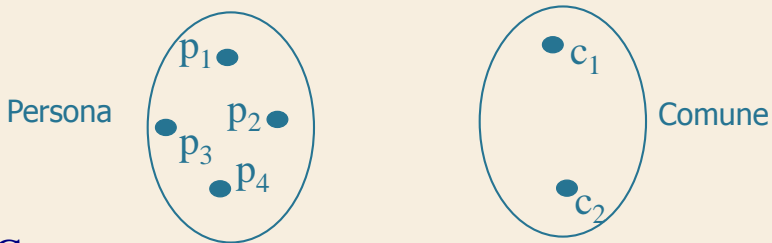


Comune



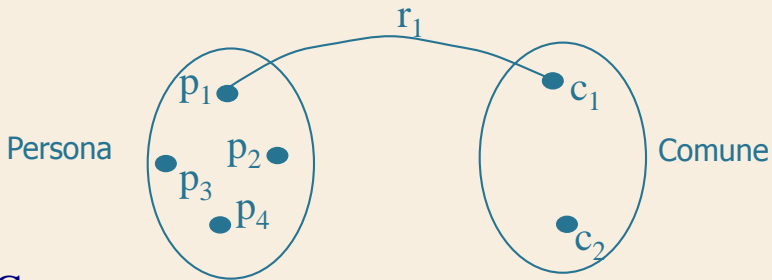
# Cardinalità delle relazioni binarie

➤ Corrispondenza 1 a N



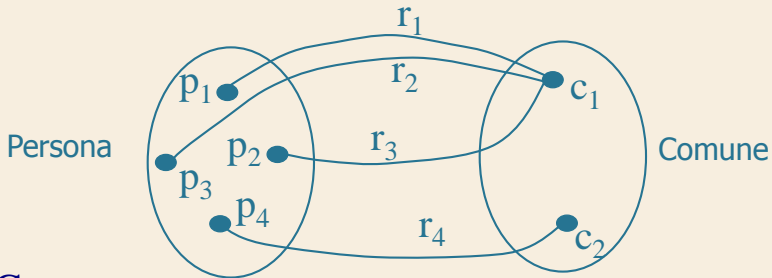
# Cardinalità delle relazioni binarie

➤ Corrispondenza 1 a N



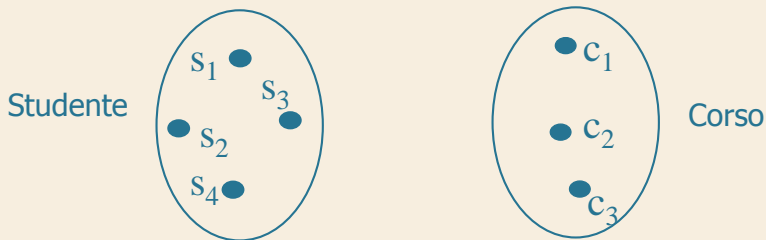
# Cardinalità delle relazioni binarie

➤ Corrispondenza 1 a N



# Cardinalità delle relazioni binarie

➤ Corrispondenza molti a molti





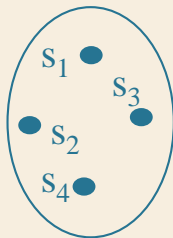
# Cardinalità delle relazioni binarie

➤ Corrispondenza molti a molti

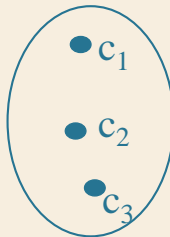
Studente

Corso

Studente

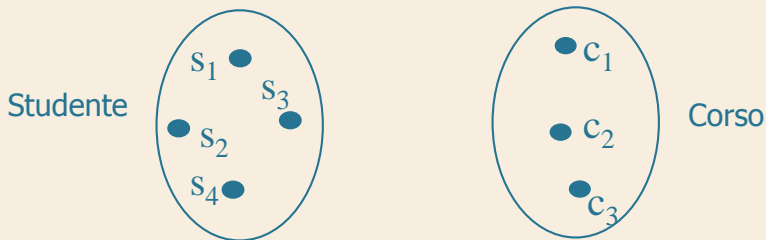
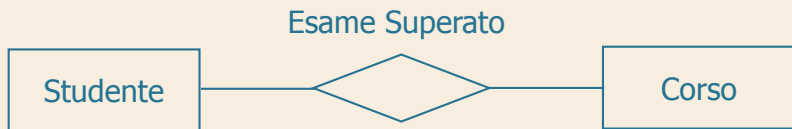


Corso



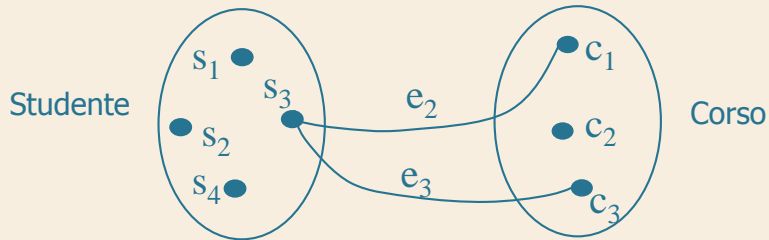
# Cardinalità delle relazioni binarie

➤ Corrispondenza molti a molti



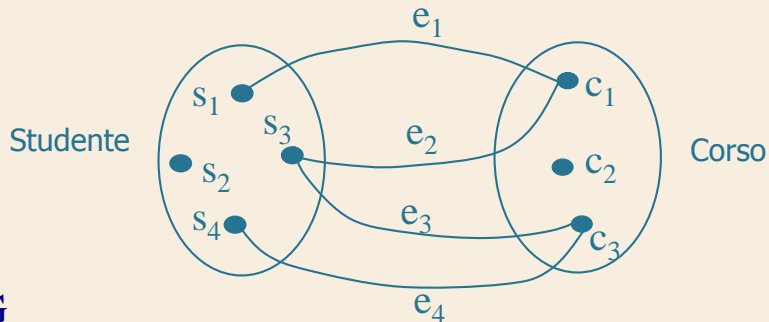
# Cardinalità delle relazioni binarie

➤ Corrispondenza molti a molti

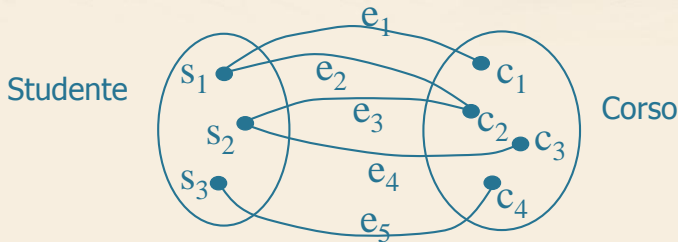


# Cardinalità delle relazioni binarie

➤ Corrispondenza molti a molti



## Limite di una relazione binaria



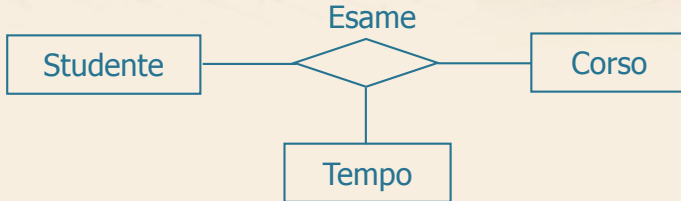
➤ Non è possibile che uno studente sostenga due volte lo stesso esame

## Relazione ternaria

- Uno studente può ripetere lo stesso esame in tempi diversi
- Esempio di istanza di esame

$s_1$	$c_1$	$t_1$
$s_1$	$c_1$	$t_2$
...		

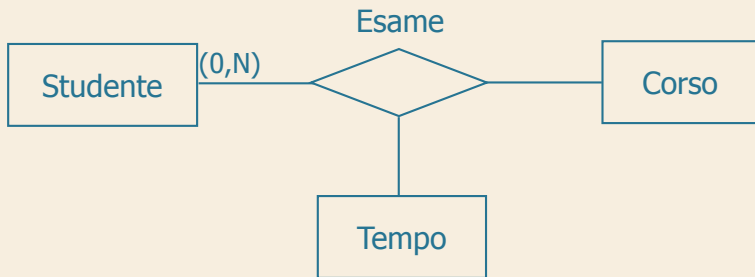
## Relazione ternaria



- Uno studente può ripetere lo stesso esame in tempi diversi
- Esempio di istanza di esame

$s_1$	$c_1$	$t_1$
$s_1$	$c_1$	$t_2$
...		

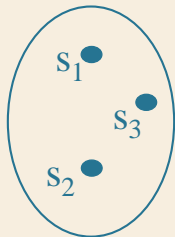
# Cardinalità delle relazioni ternarie



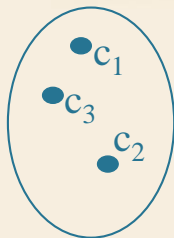


# Occorrenze di una relazione ternaria

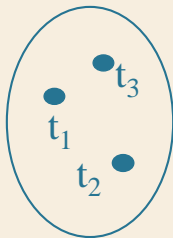
Studente



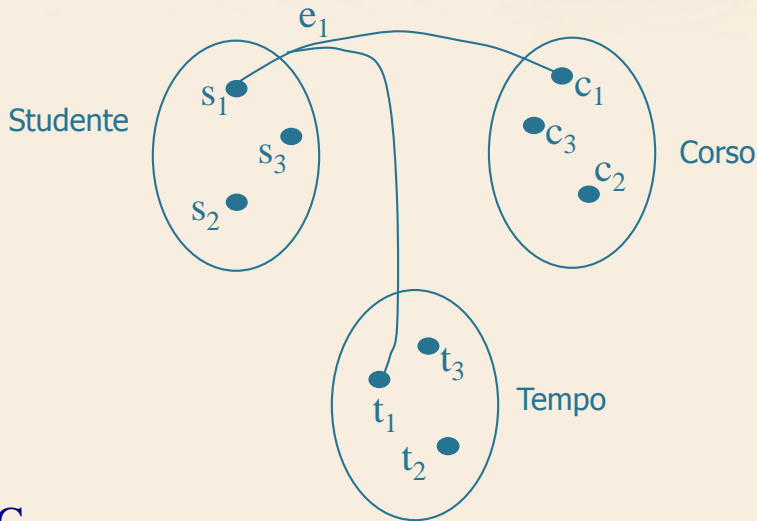
Corso



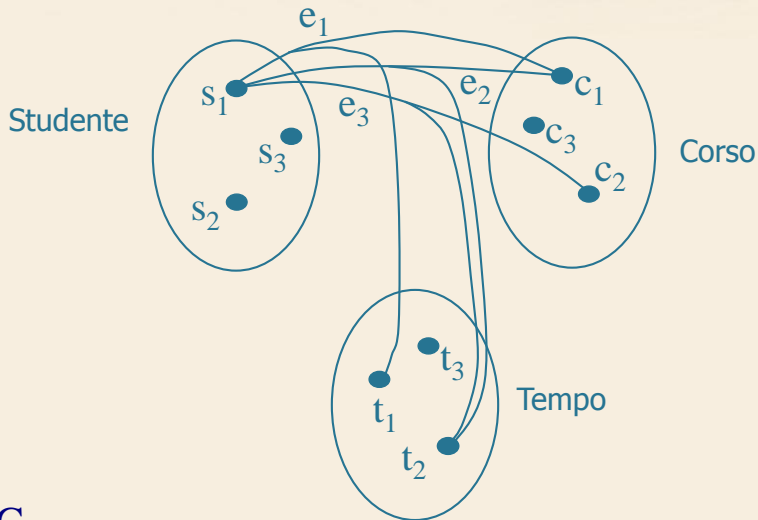
Tempo



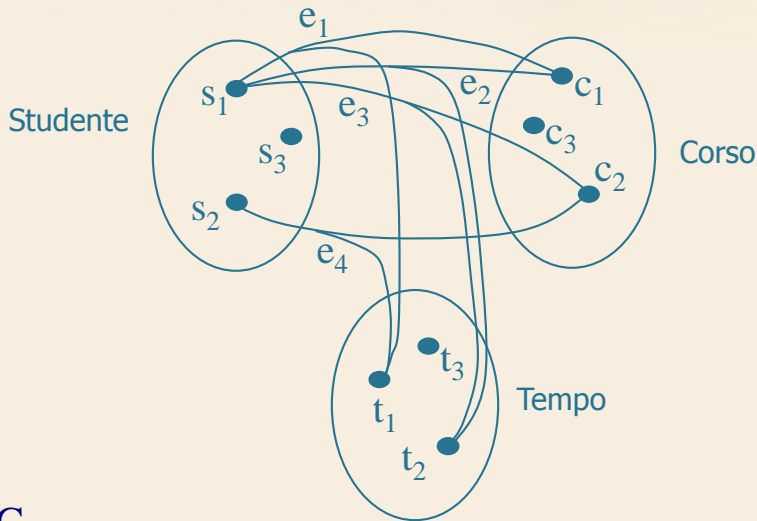
# Occorrenze di una relazione ternaria



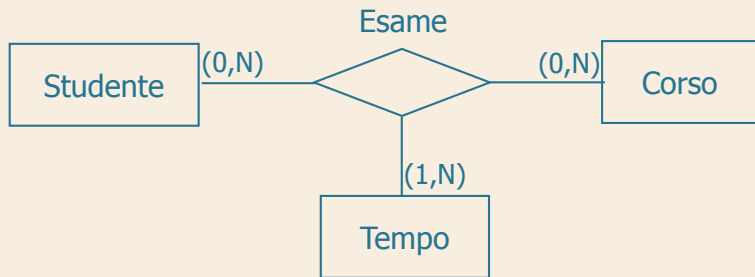
# Occorrenze di una relazione ternaria



# Occorrenze di una relazione ternaria

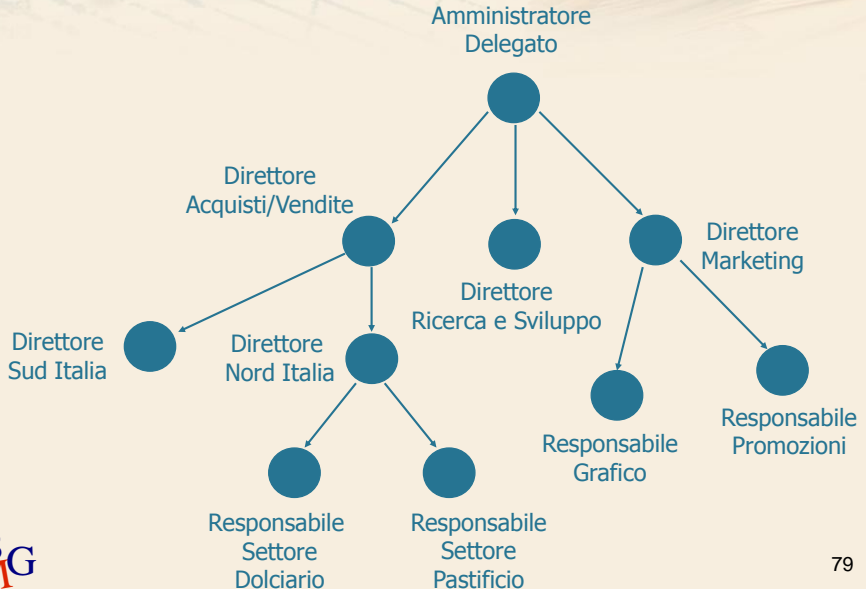


# Cardinalità delle relazioni ternarie



- Le cardinalità minime raramente sono 1 per tutte le entità coinvolte in una relazione
- Le cardinalità massime di una relazione  $n$ -aria sono (praticamente) sempre  $N$ 
  - se la partecipazione di un'entità  $E$  ha cardinalità massima 1, è possibile eliminare la relazione  $n$ -aria e legare l'entità  $E$  con le altre mediante relazioni binarie

# Relazione ricorsiva



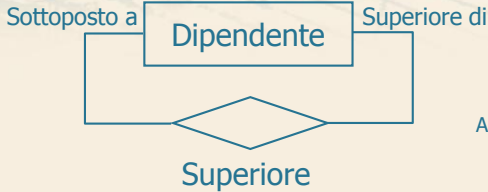
# Relazione ricorsiva

Dipendente





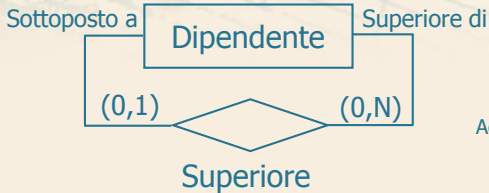
# Relazione ricorsiva



➤ Relazione di un'entità con se stessa

➤ Se la relazione non è simmetrica, occorre definire due ruoli dell'entità

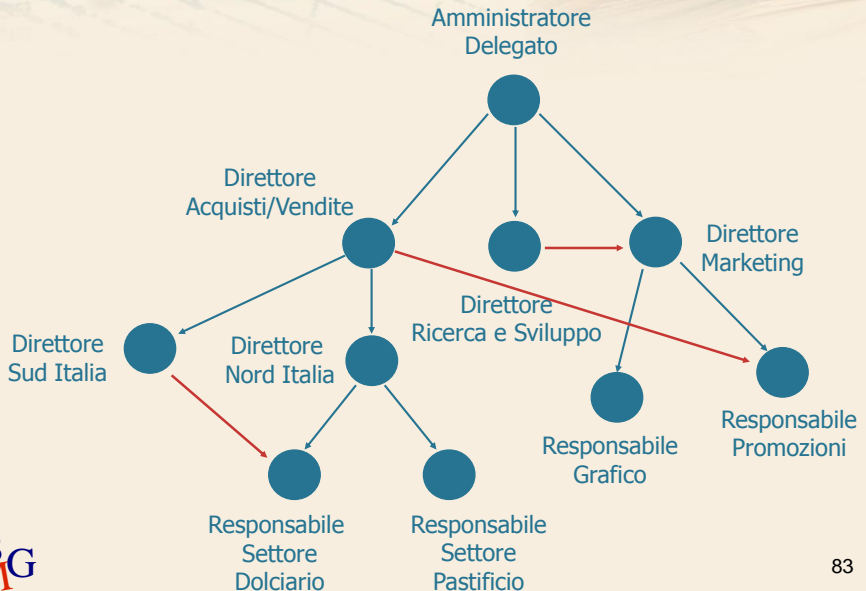
# Relazione ricorsiva



➤ Relazione di un'entità con se stessa

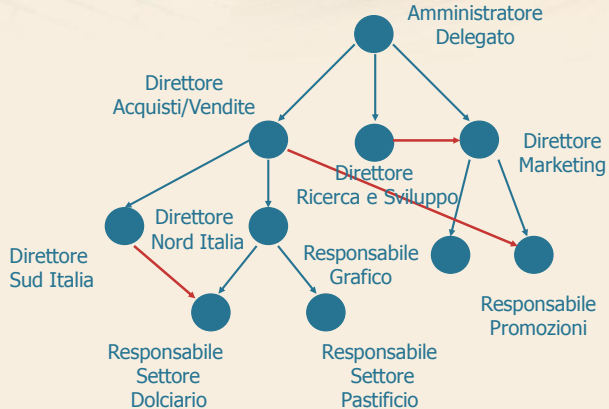
➤ Se la relazione non è simmetrica, occorre definire due ruoli dell'entità

# Relazione ricorsiva

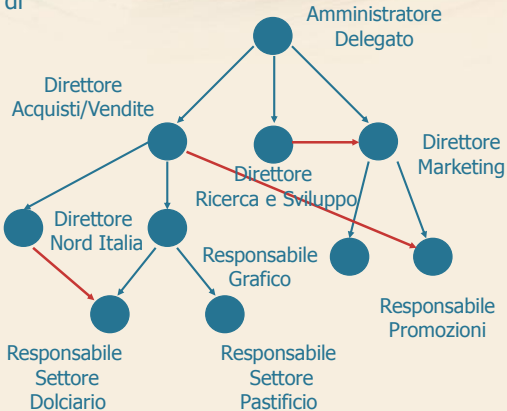
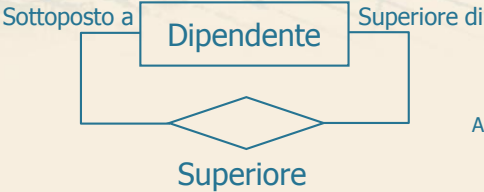


# Relazione ricorsiva

Dipendente

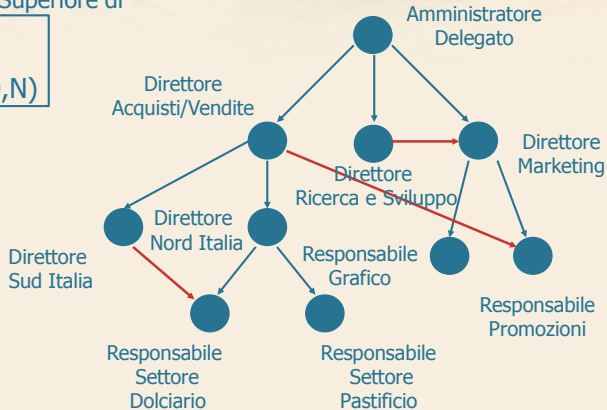
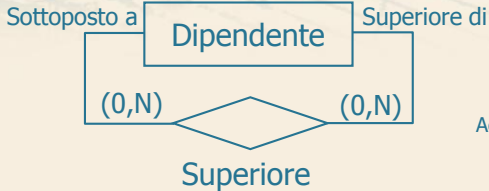


## Relazione ricorsiva



⇒ Un sottoposto potrebbe avere più superiori

# Relazione ricorsiva



Un sottoposto potrebbe avere più superiori



# Modello Entità-Relazione

Attributi

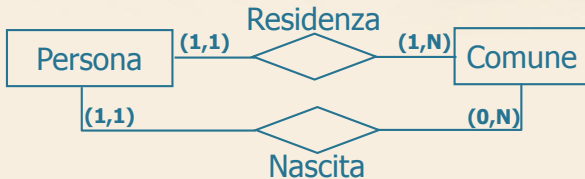


Nome attributo

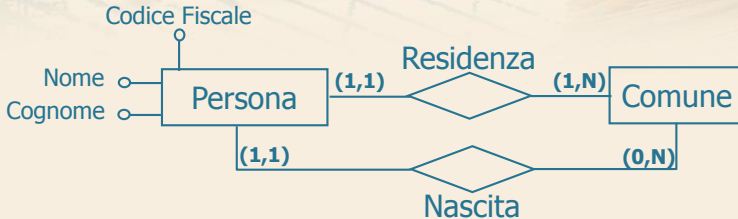
- Descrive una proprietà elementare di un'entità o di una relazione
- Esempi
  - cognome, nome, matricola sono attributi che descrivono l'entità studente
  - voto è un attributo che descrive la relazione esame
- Ogni attributo è caratterizzato dal *dominio* l'insieme dei valori ammissibili per l'attributo



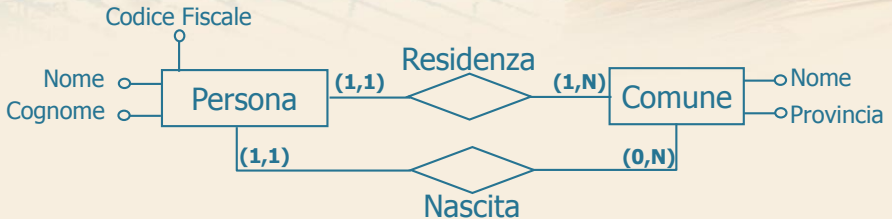
## Esempi di attributi



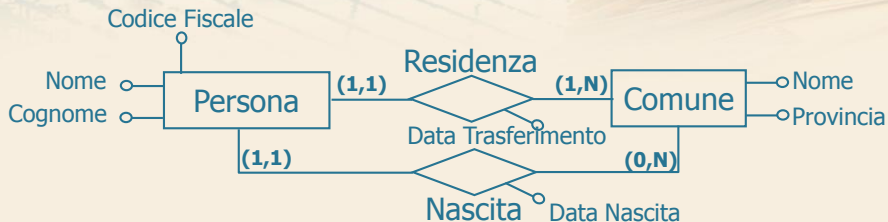
# Esempi di attributi



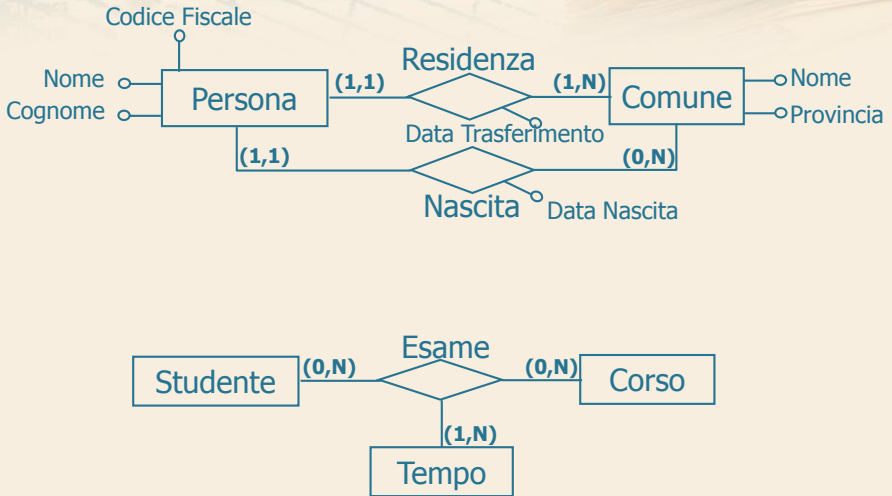
# Esempi di attributi



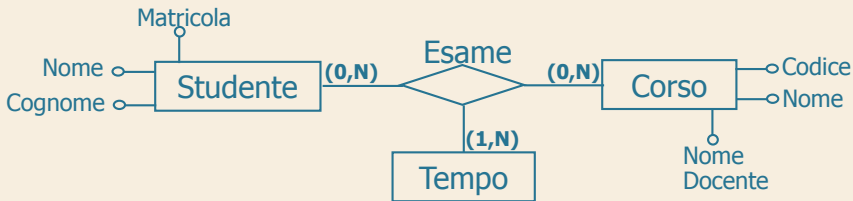
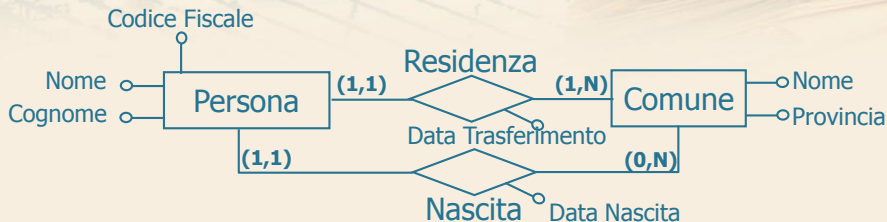
## Esempi di attributi



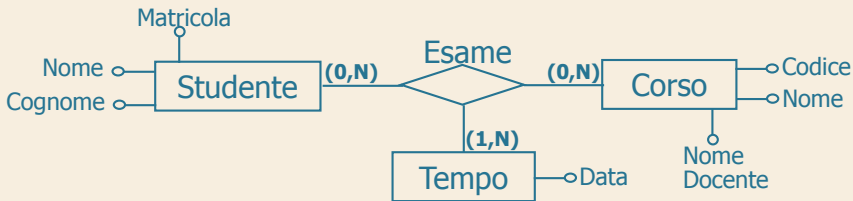
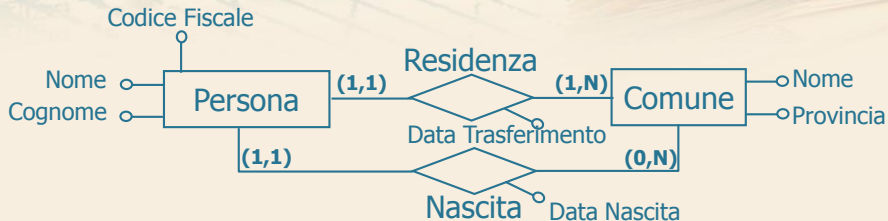
# Esempi di attributi



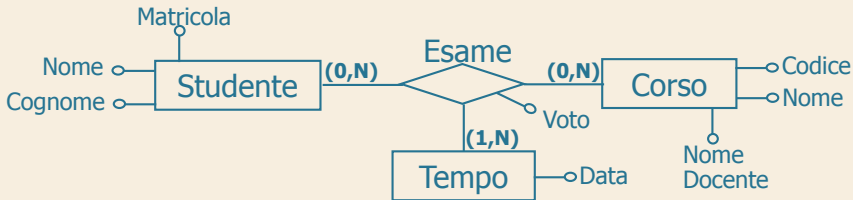
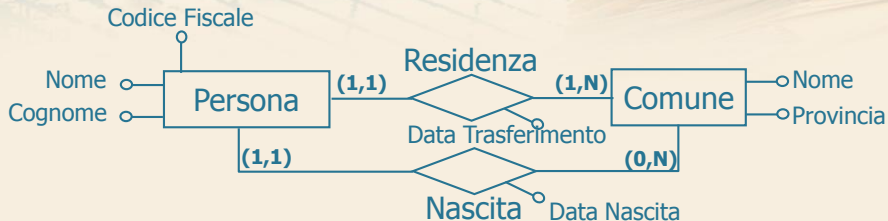
# Esempi di attributi



# Esempi di attributi



# Esempi di attributi





## Attributo composto



➤ Raggruppamento di attributi affini per significato o per uso

# Attributo composto



➤ Raggruppamento di attributi affini per significato o per uso

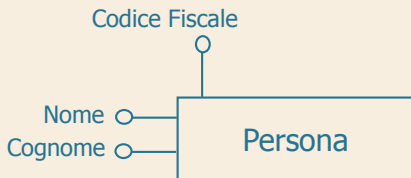
➤ Esempio



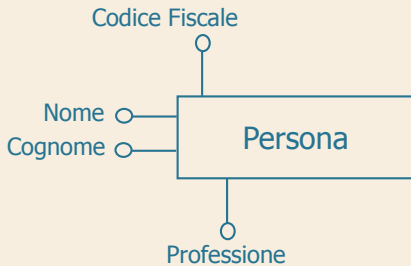
## Cardinalità di un attributo

- Può essere specificata per gli attributi di entità o relazioni
- Descrive numero minimo e massimo di valori dell'attributo associati ad una occorrenza di un'entità o di una relazione
  - se è omessa corrisponde ad (1,1)
  - minima 0 corrisponde ad attributo che ammette il valore nullo
  - massima N corrisponde ad attributo che può assumere più di un valore per la stessa occorrenza (attributo multivalore)

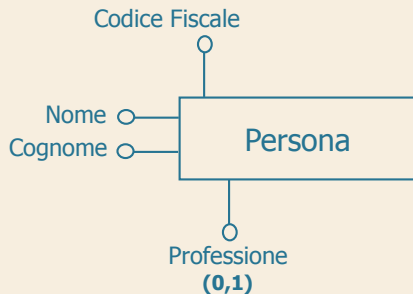
# Cardinalità di un attributo



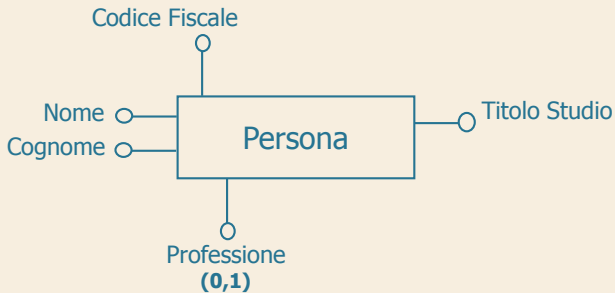
# Cardinalità di un attributo



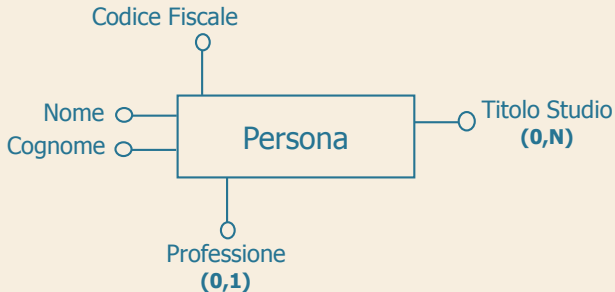
# Cardinalità di un attributo



# Cardinalità di un attributo



# Cardinalità di un attributo







# Modello Entità-Relazione

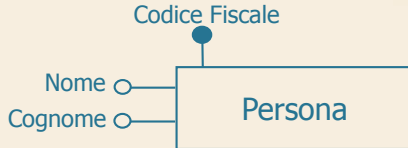
Identificatori

# Identificatore

- È specificato per ogni entità
- Descrive i concetti (attributi e/o entità) dello schema che permettono di individuare in modo univoco le occorrenze delle entità
  - ogni entità deve avere almeno un identificatore
  - può esistere più di un identificatore appropriato per un'entità

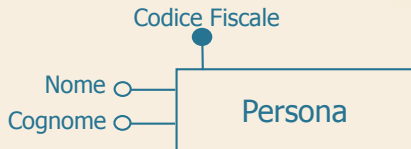
# Identificatore interno

➤ Semplice: costituito da un solo attributo

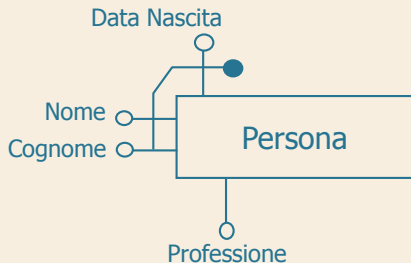


# Identificatore interno

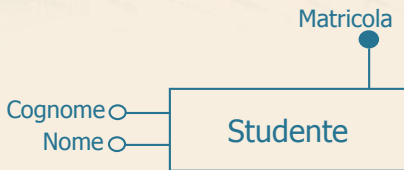
➤ Semplice: costituito da un solo attributo



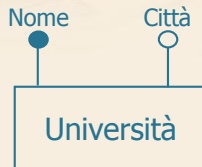
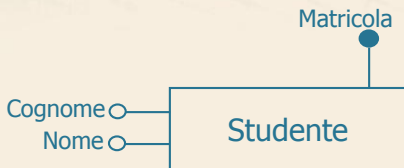
➤ Composto: costituito da più attributi



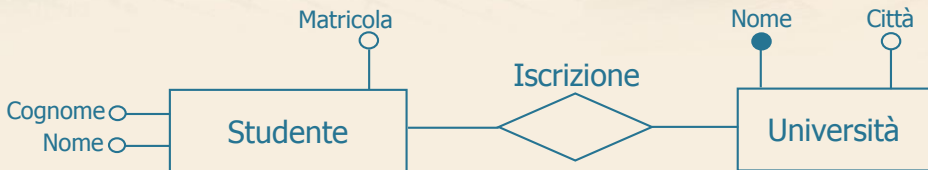
# Identificatore esterno



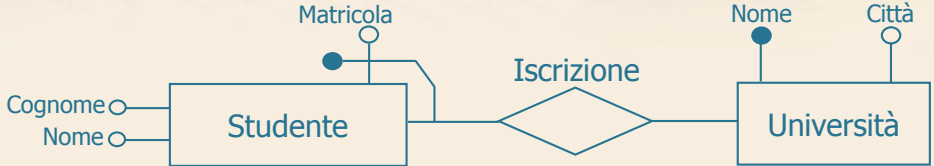
# Identificatore esterno



# Identificatore esterno



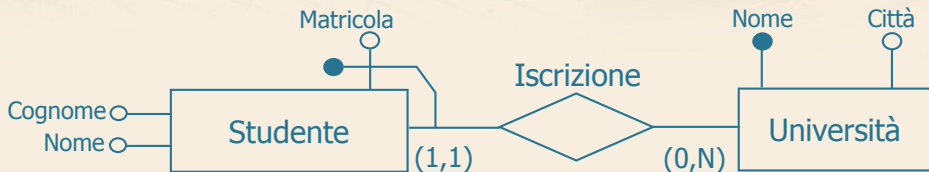
## Identificatore esterno



➤ L'entità che non dispone internamente di attributi sufficienti per definire un identificatore è denominata *entità debole*

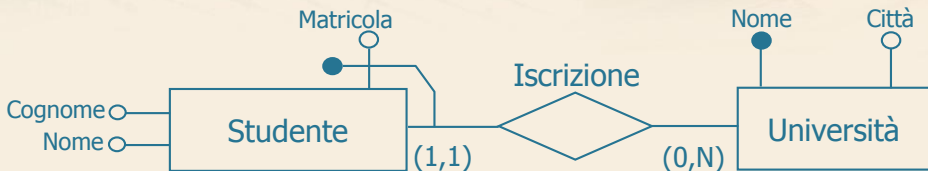


## Identificatore esterno



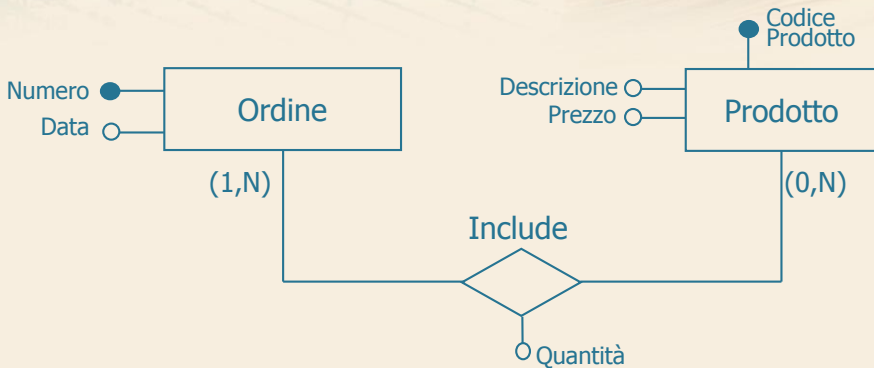
➤ L'entità che non dispone internamente di attributi sufficienti per definire un identificatore è denominata *entità debole*

## Identificatore esterno

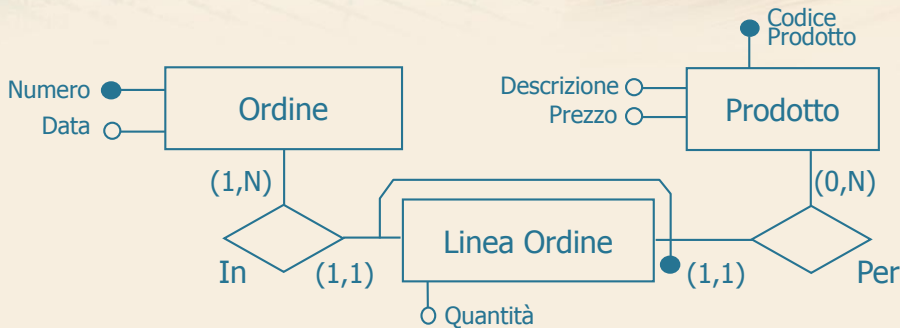


- L'entità che non dispone internamente di attributi sufficienti per definire un identificatore è denominata *entità debole*
- L'entità debole deve partecipare con cardinalità **(1,1)** in ognuna delle relazioni che forniscono parte dell'identificatore

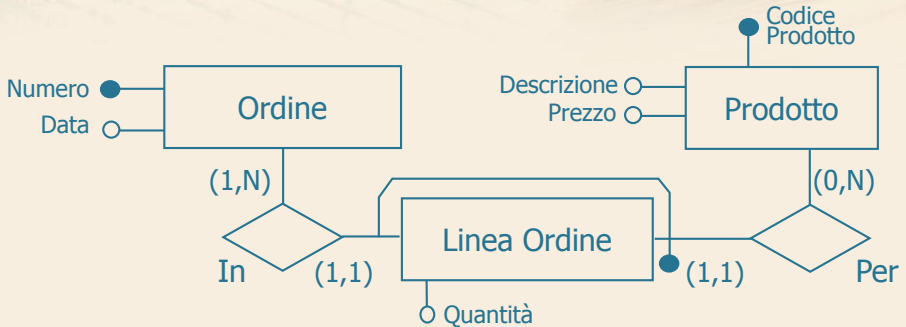
# Identificatore esterno



# Identificatore esterno

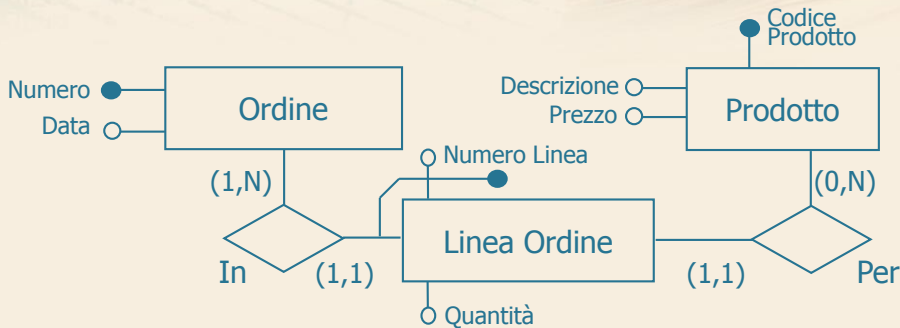


## Identificatore esterno



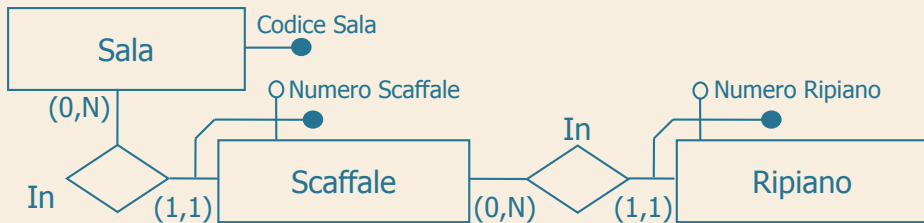
➤ È possibile rappresentare nello stesso ordine più linee ordine per lo stesso prodotto?

# Identificatore esterno

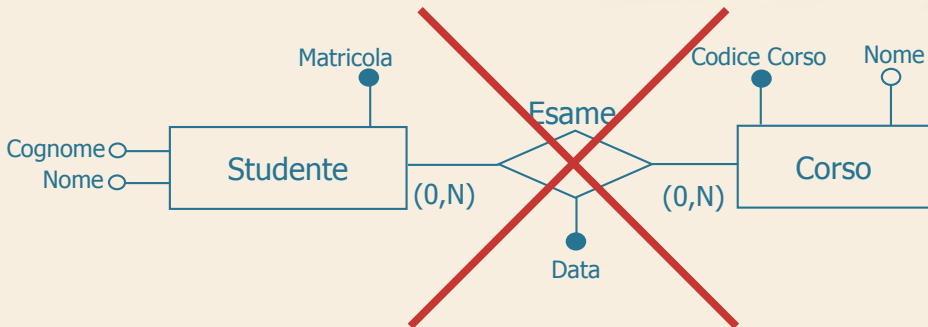


## Osservazioni

- Un identificatore esterno può coinvolgere un'entità a sua volta identificata esternamente
- non si devono generare cicli di identificazione



➤ Le relazioni *non* hanno identificatori



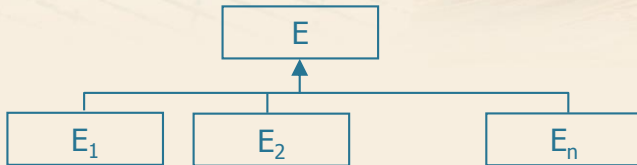




# Modello Entità-Relazione

Generalizzazione

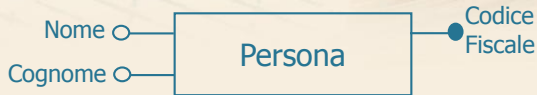
# Generalizzazione



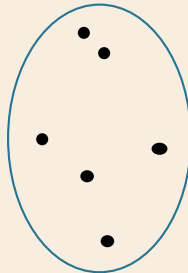
➤ Descrive un collegamento logico tra un'entità  $E$ , e una o più entità  $E_1, E_2, \dots, E_n$ , in cui  $E$  comprende come casi particolari  $E_1, E_2, \dots, E_n$

- $E$ , detta entità padre, è una generalizzazione di  $E_1, E_2, \dots, E_n$
- $E_1, E_2, \dots, E_n$ , dette entità figlie, sono una specializzazione di  $E$

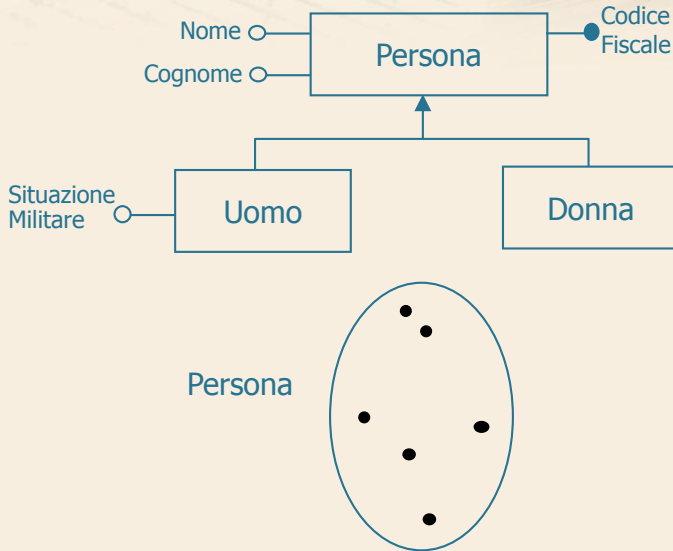
# Generalizzazione: esempio



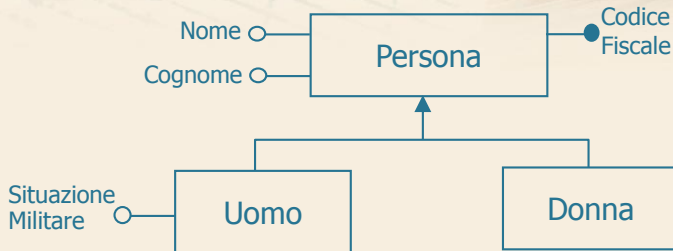
Persona



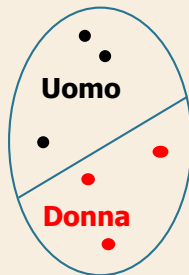
# Generalizzazione: esempio



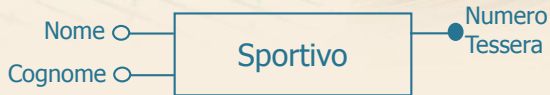
# Generalizzazione: esempio



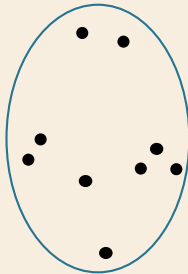
Persona



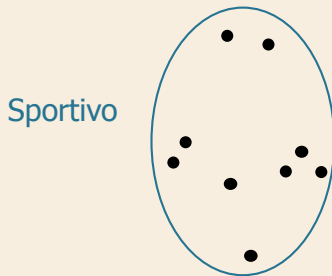
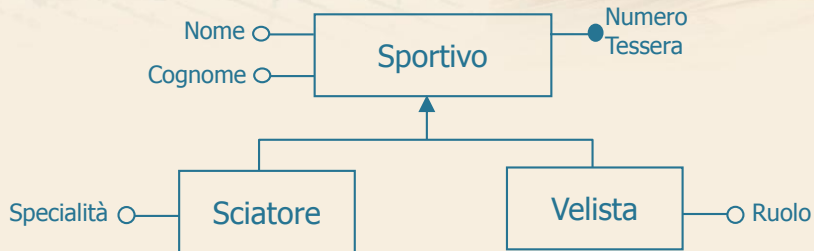
# Generalizzazione: esempio



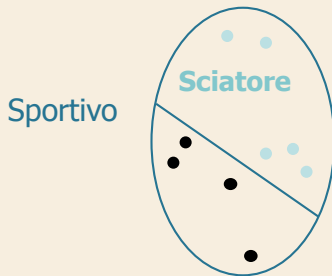
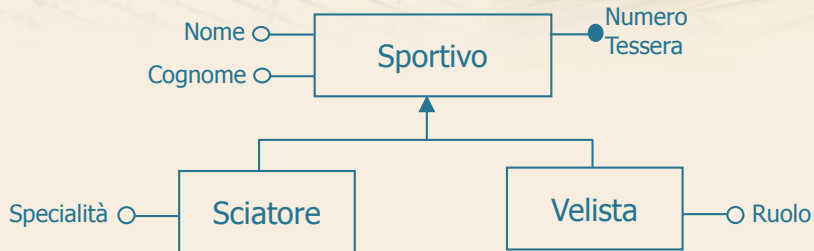
Sportivo



# Generalizzazione: esempio

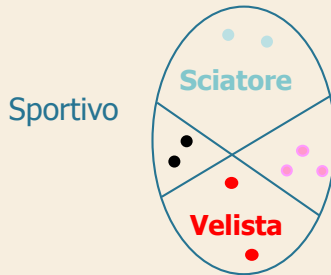
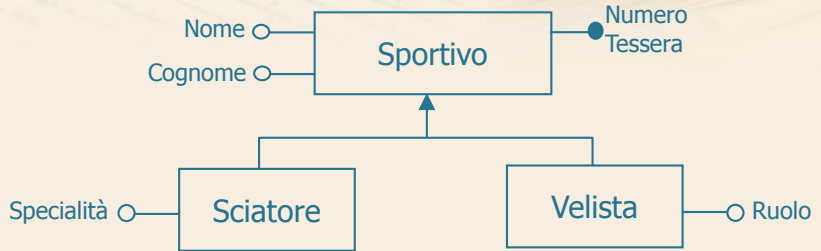


# Generalizzazione: esempio

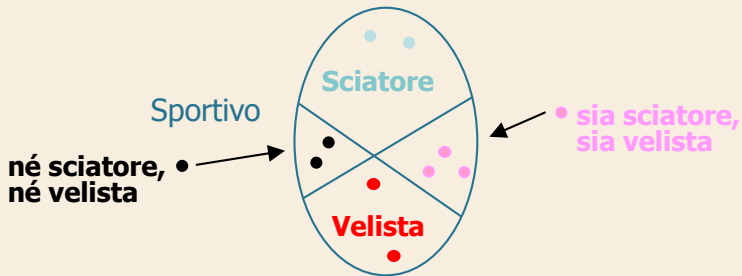
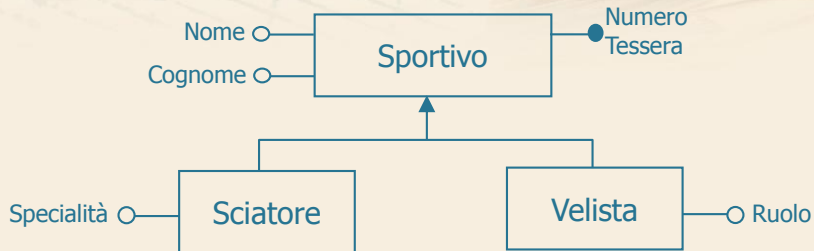




# Generalizzazione: esempio



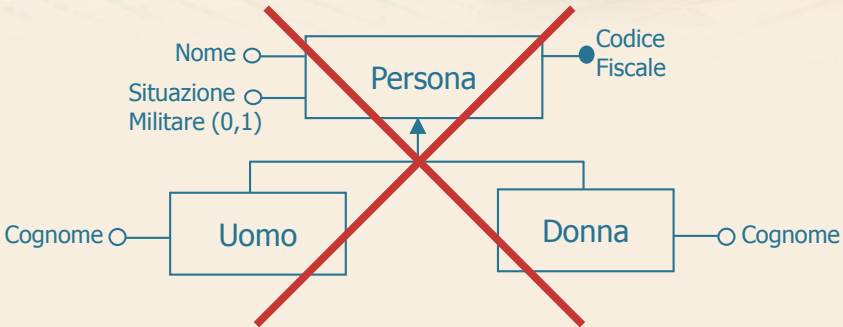
# Generalizzazione: esempio



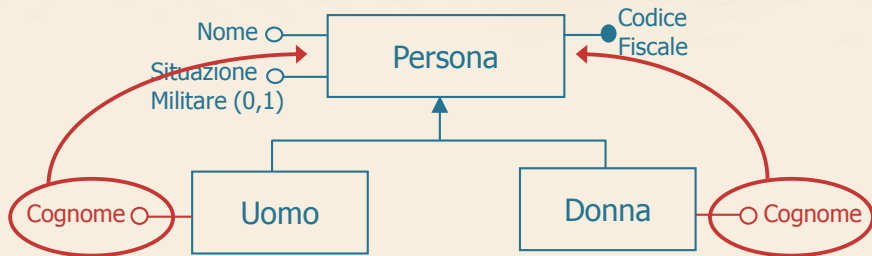
## Generalizzazione: proprietà

- Ogni occorrenza di un'entità figlia è anche un'occorrenza dell'entità padre
- Ogni proprietà dell'entità padre (attributi, identificatori, relazioni, altre generalizzazioni) è anche una proprietà di ogni entità figlia
  - proprietà nota come *ereditarietà*
- Un'entità può essere coinvolta in più generalizzazioni diverse

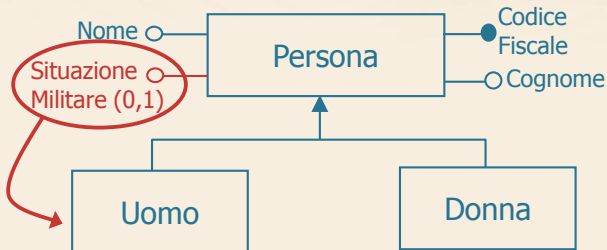
## Generalizzazione: esempio non corretto



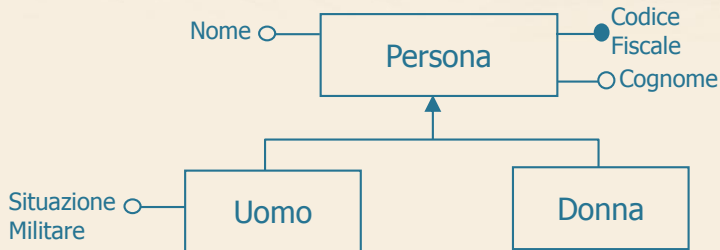
## Generalizzazione: esempio non corretto



## Generalizzazione: esempio non corretto



## Generalizzazione: esempio corretto



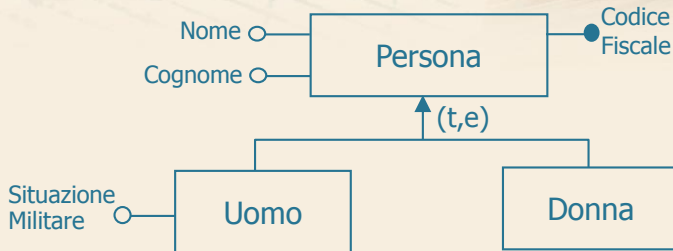
# Generalizzazione: proprietà

## ➤ Caratteristiche ortogonali

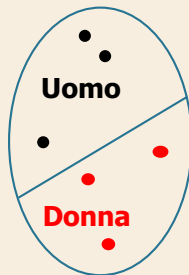
- generalizzazione *totale* se ogni occorrenza dell'entità padre è un'occorrenza di almeno una delle entità figlie, *parziale* altrimenti
- *esclusiva* se ogni occorrenza dell'entità padre è al più un'occorrenza di una delle entità figlie, *sovrapposta* altrimenti



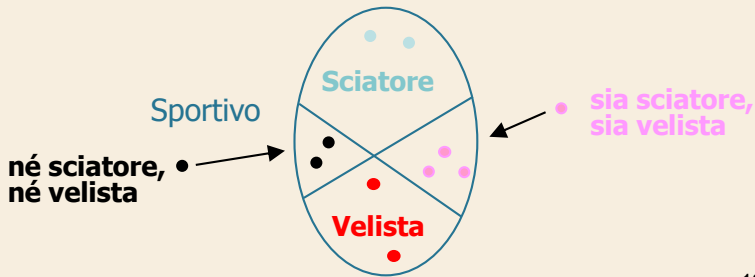
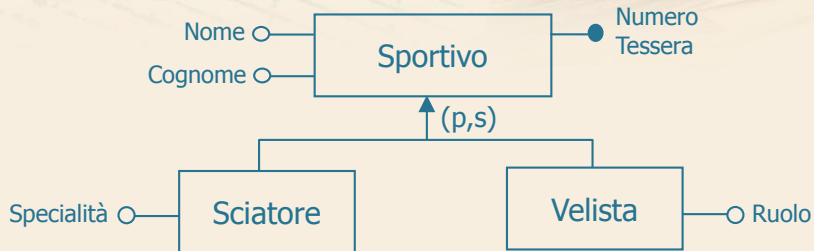
# Generalizzazione: esempio



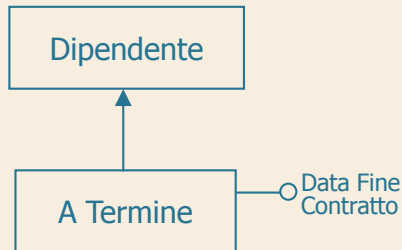
Persona



# Generalizzazione: esempio



- Caso particolare di generalizzazione con una sola entità figlia
- la generalizzazione è sempre parziale ed esclusiva

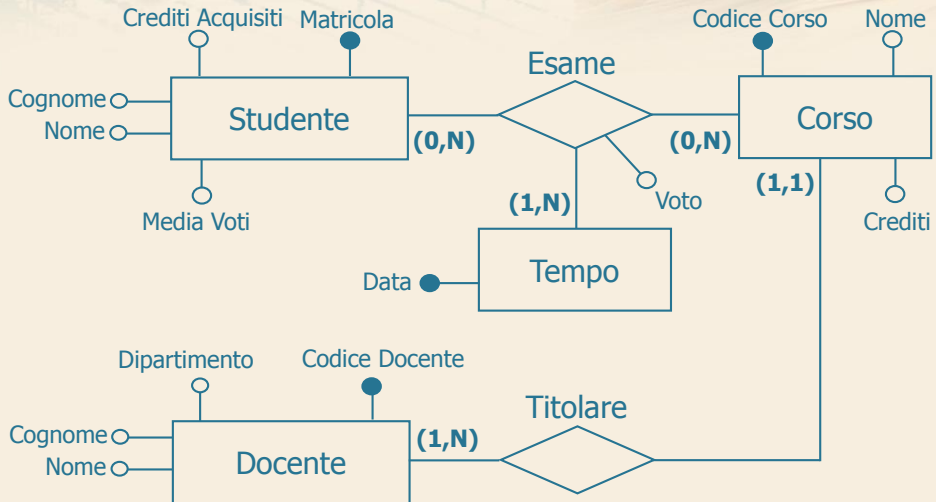




# Modello Entità-Relazione

Documentazione di schemi E-R

# Documentazione di schemi E-R



# Documentazione di schemi E-R

## ➤ Dizionario dei dati

- permette di arricchire lo schema E-R con descrizioni in linguaggio naturale di entità, relazioni e attributi

## Dizionario dei dati: esempio

Entità	Descrizione	Attributi	Identificatore
Studente	Studente dell'università	Matricola, Cognome, Nome, Crediti acquisiti, Media voti	Matricola
Docente	Docente dell'università	Codice docente, Dipartimento, Cognome, Nome	Codice docente
Corso	Corsi offerti dall'università	Codice corso, Nome, Crediti	Codice corso
Tempo	Date in cui sono stati sostenuti esami	Data	Data

## Dizionario dei dati: esempio

Relazione	Descrizione	Entità coinvolte	Attributi
Esame	Associa uno studente agli esami che ha sostenuto e memorizza il voto conseguito	Studente (0,N), Corso (0,N), Tempo (1,N)	Voto
Titolare	Associa ogni corso al suo docente titolare	Corso (1,1), Docente (0,N)	



# Documentazione di schemi E-R

## ➤ Dizionario dei dati

- permette di arricchire lo schema E-R con descrizioni in linguaggio naturale di entità, relazioni e attributi

## ➤ Vincoli d'integrità sui dati

- non sempre possono essere indicati esplicitamente in uno schema E-R
- possono essere descritti in linguaggio naturale

# Vincoli d'integrità sui dati: esempio

## Vincoli d'integrità

RV1	Il voto di un esame può assumere esclusivamente valori compresi tra 0 e 30
RV2	Ogni studente non può superare due volte con esito positivo lo stesso esame
RV3	Uno studente non può sostenere più di tre volte l'esame relativo allo stesso corso nell'arco dello stesso anno accademico

# Documentazione di schemi E-R

## ➤ Dizionario dei dati

- permette di arricchire lo schema E-R con descrizioni in linguaggio naturale di entità, relazioni e attributi

## ➤ Vincoli d'integrità sui dati

- non sempre possono essere indicati esplicitamente in uno schema E-R
- possono essere descritti in linguaggio naturale

## ➤ Regole di derivazione dei dati

- permettono di esplicitare che un concetto dello schema può essere ottenuto (mediante inferenza o calcolo aritmetico) da altri concetti dello schema

# Regole di derivazione dei dati: esempio

## Regole di derivazione

RD1	Il numero di crediti acquisiti da uno studente si ottiene sommando il numero di crediti dei corsi per cui lo studente ha superato l'esame
RD2	La media voti di uno studente si ottiene calcolando la media dei voti degli esami superati dallo studente



# Modello Entità-Relazione

UML ed E-R

## ➤ UML (Unified Modeling Language)

- modellazione di un'applicazione software
  - aspetti strutturali e comportamentali (dati, operazioni, processi e architetture)
- formalismo ricco
  - diagramma delle classi, degli attori, di sequenza, di comunicazione, degli stati, ...

## ➤ E-R

- modellazione di una base di dati
  - aspetti strutturali di un'applicazione
- costrutti funzionali alla modellazione di basi di dati

## ➤ Principali differenze di UML rispetto ad ER

- assenza di notazione standard per definire gli identificatori
- possibilità di aggiungere note per commentare i diagrammi
- possibilità di indicare il verso di navigazione di una associazione (non rilevante nella progettazione di una base di dati)

- Formalismi diversi
- Il diagramma delle classi di un'applicazione è diverso dallo schema E-R della base di dati
- Il diagramma delle classi, anche se progettato per uso diverso, può essere adattato per la descrizione del progetto concettuale di una base di dati