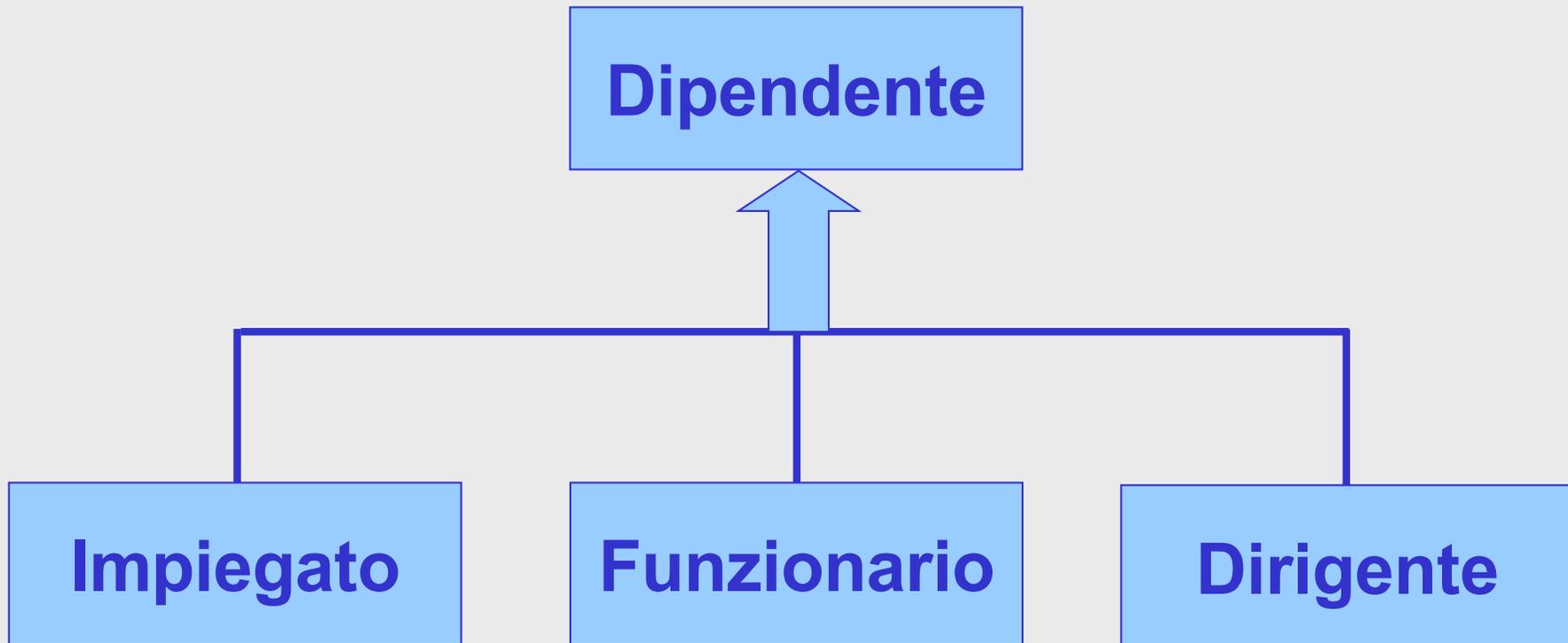


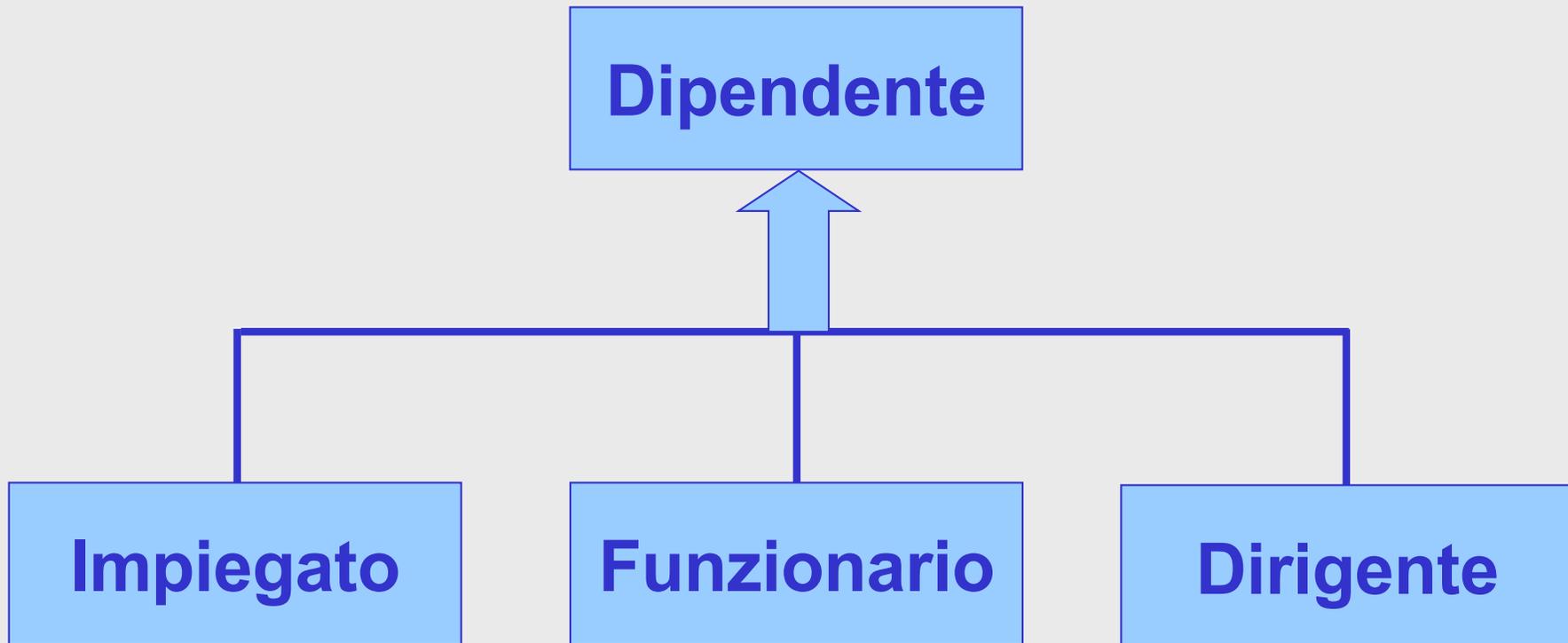
# Un ultimo costrutto, la generalizzazione



# 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  $E$

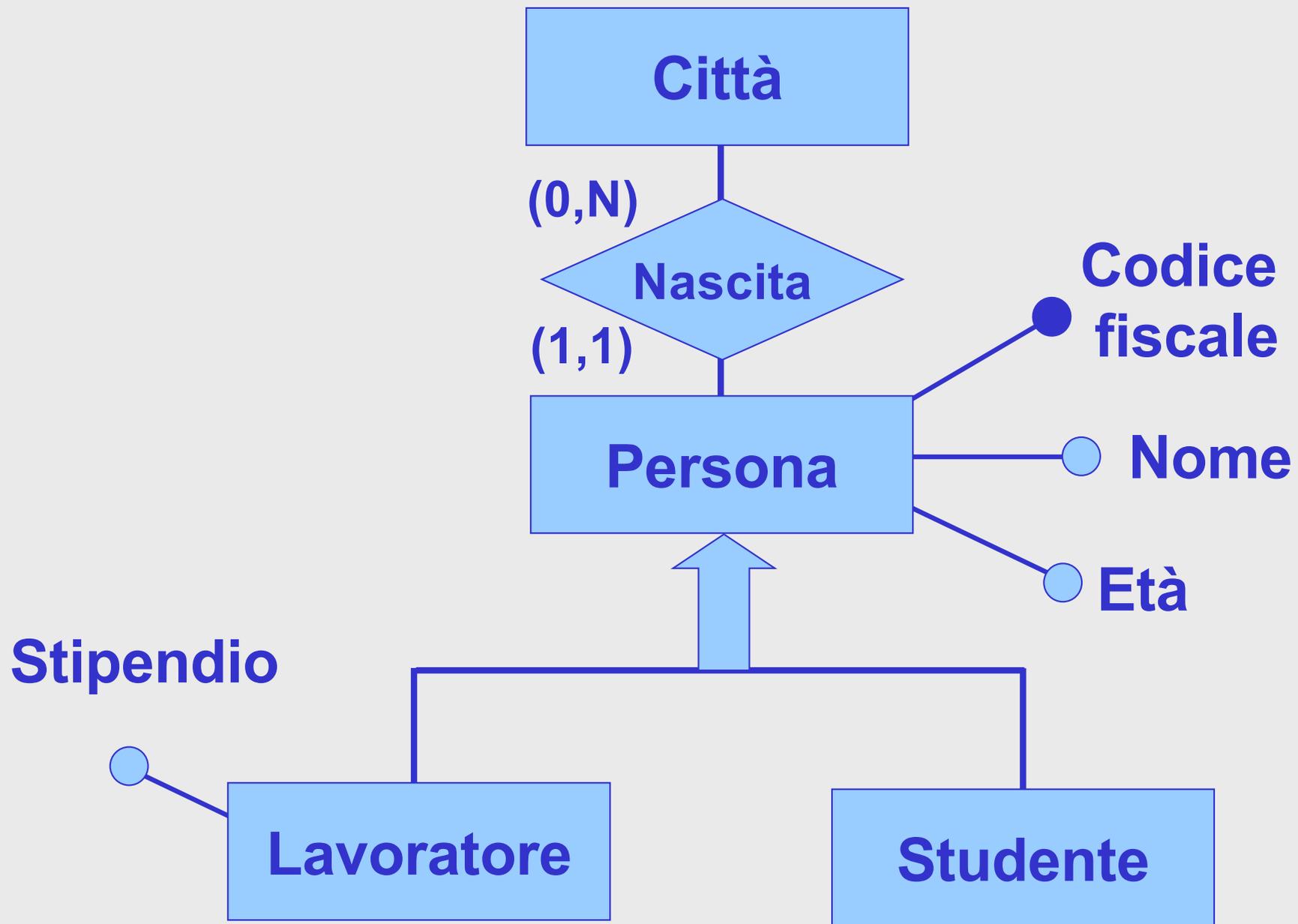
# Rappresentazione grafica



# Caratteristiche delle generalizzazioni

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

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

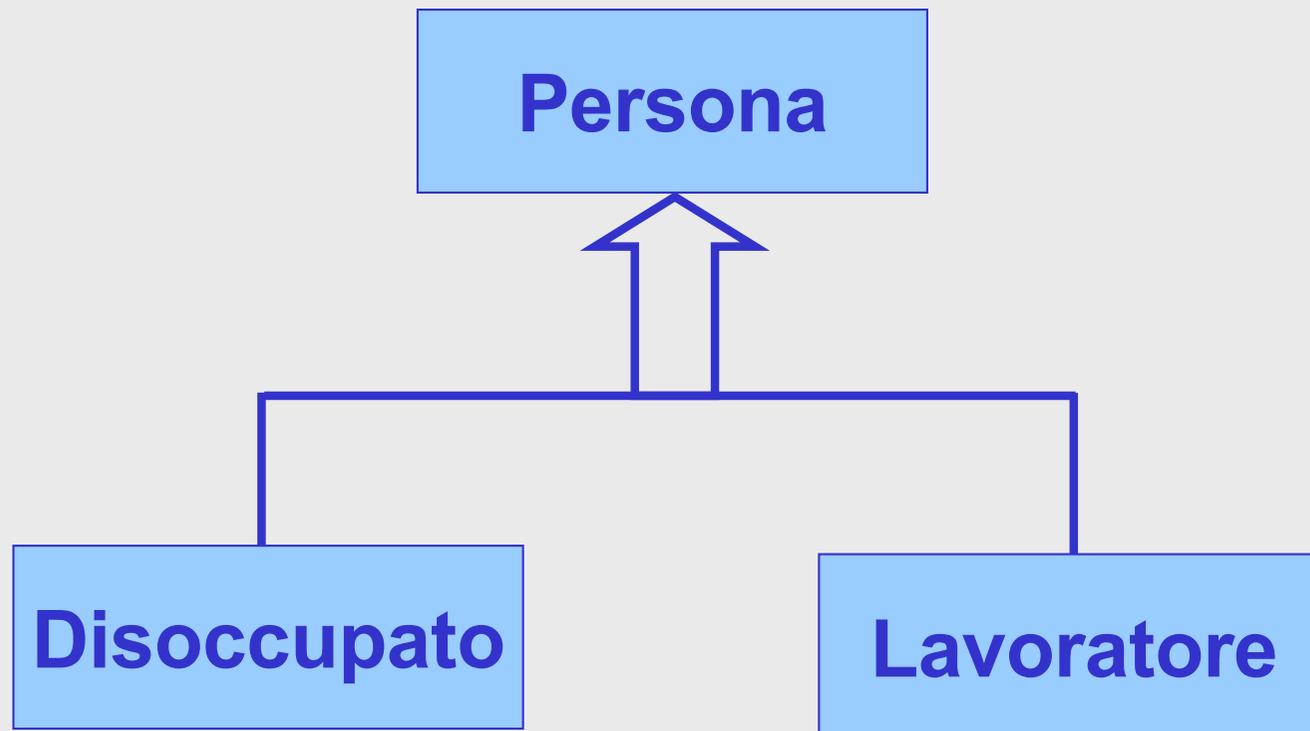


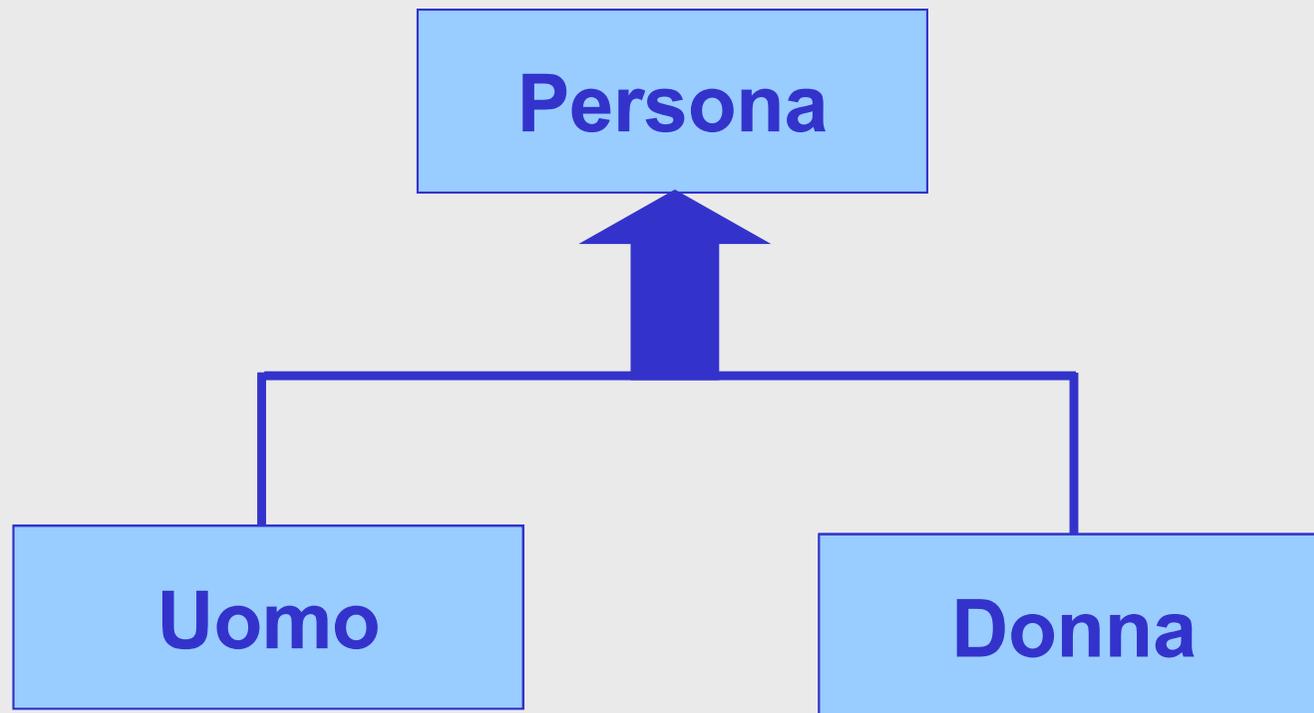
# Ereditarietà

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

# Tipi di generalizzazioni

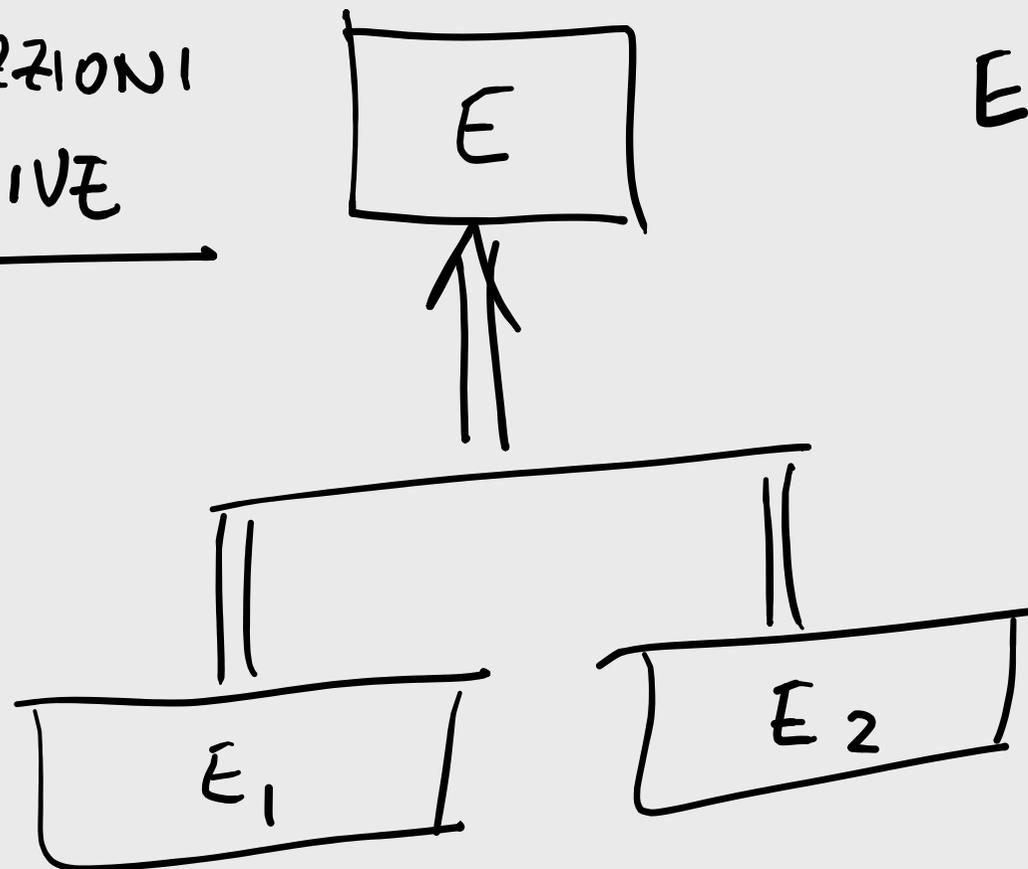
- **totale** se ogni occorrenza dell'entità genitore è occorrenza di almeno una delle entità figlie, altrimenti è **parziale**
- **esclusiva** se ogni occorrenza dell'entità genitore è occorrenza di al più una delle entità figlie, altrimenti è **sovrapposta**
- consideriamo (senza perdita di generalità)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





GENERALIZZAZIONI  
ESCLUSIVE

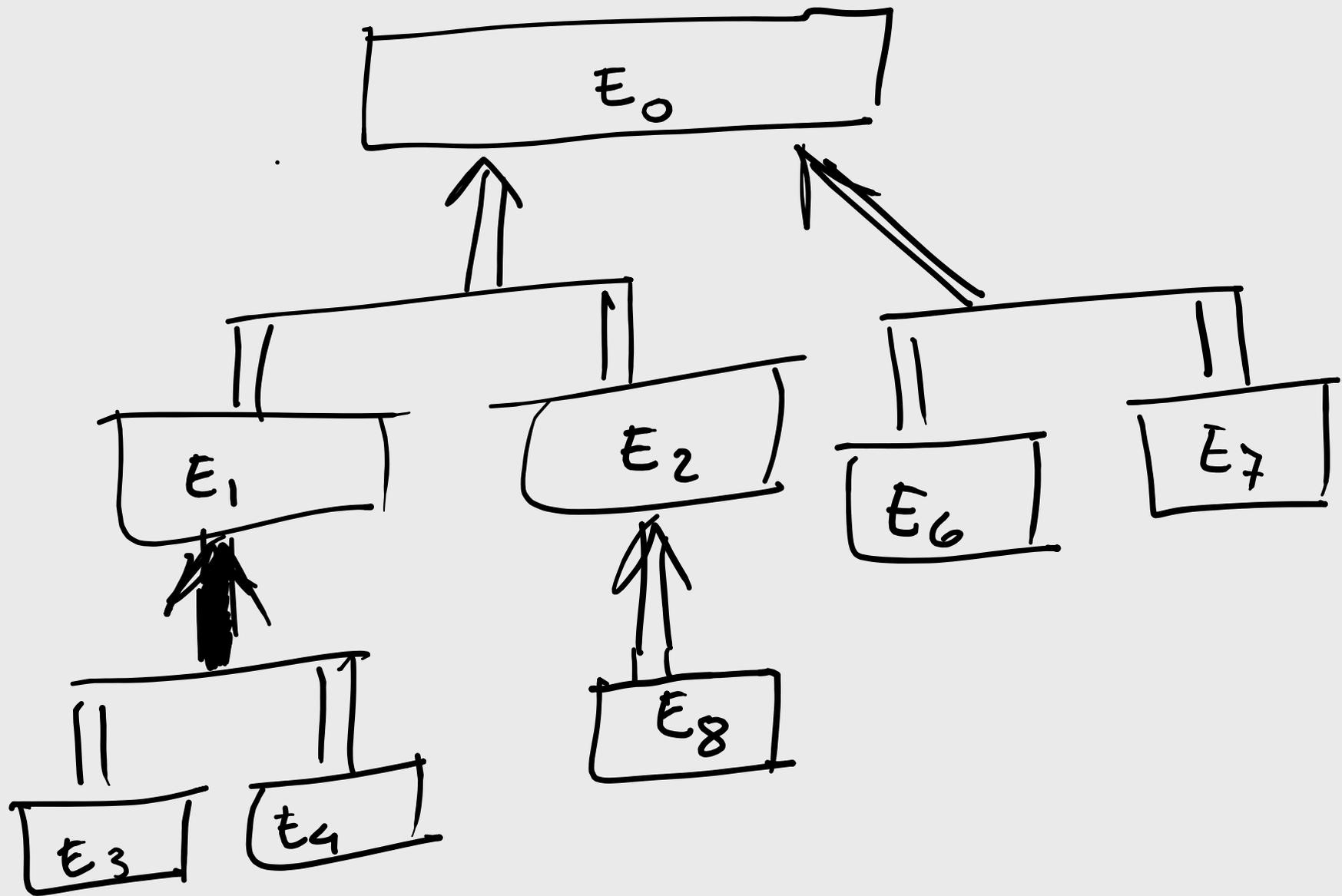
---



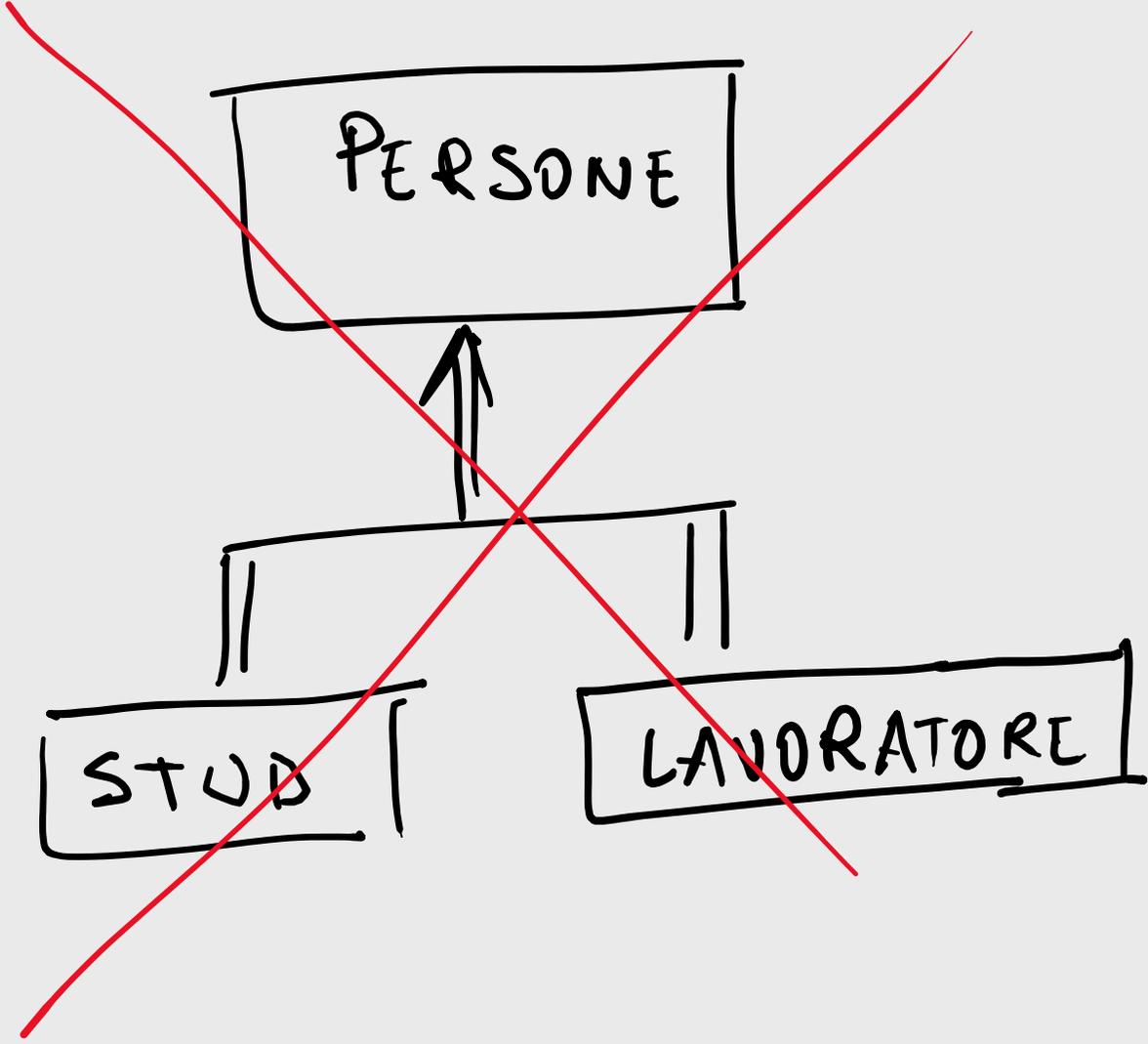
$$E_1 \wedge E_2 = \emptyset$$

## Altre proprietà

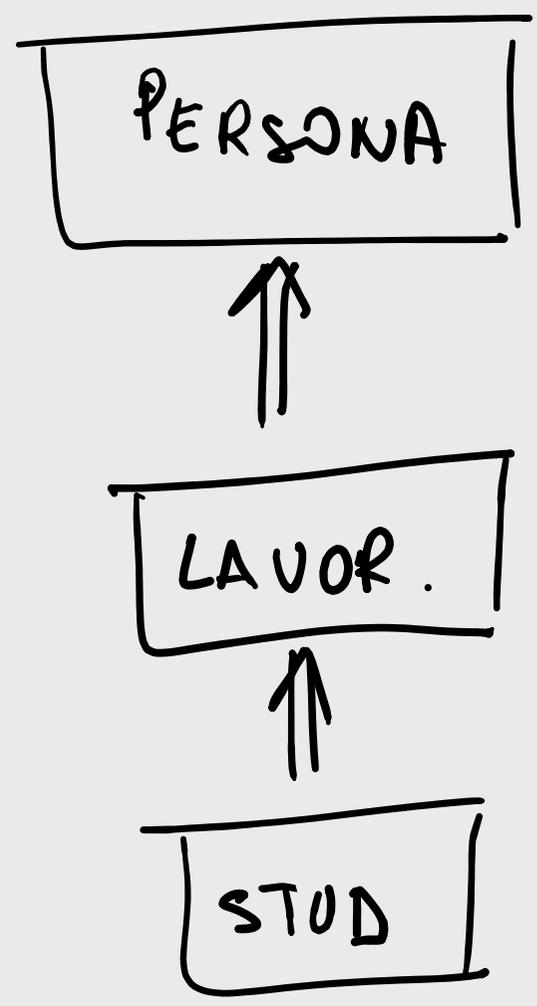
- possono esistere gerarchie a più livelli e multiple generalizzazioni allo stesso livello
- un'entità può essere inclusa in più gerarchie, come genitore e/o come figlia
- se una generalizzazione ha solo un' entità figlia si parla di **sottoinsieme**
- alcune configurazioni non hanno senso – vediamo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



?

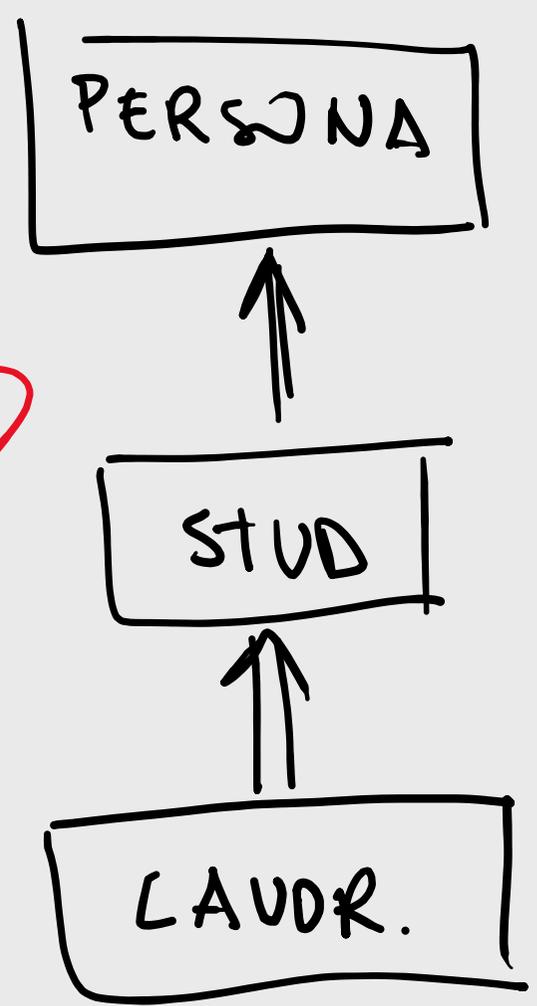


?

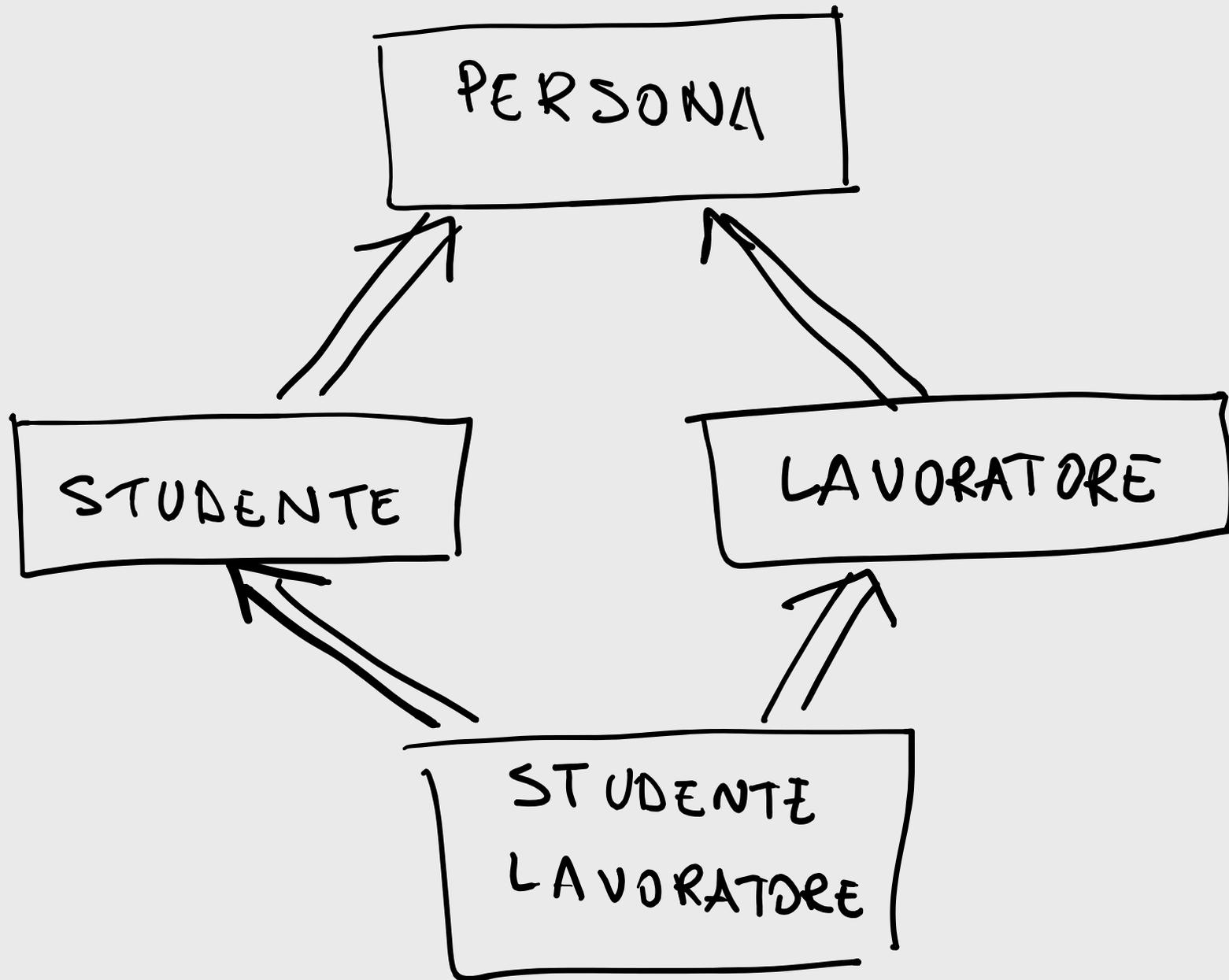


AZIENDA  
CON ANCHE  
LAVORATORI E STUDENTI

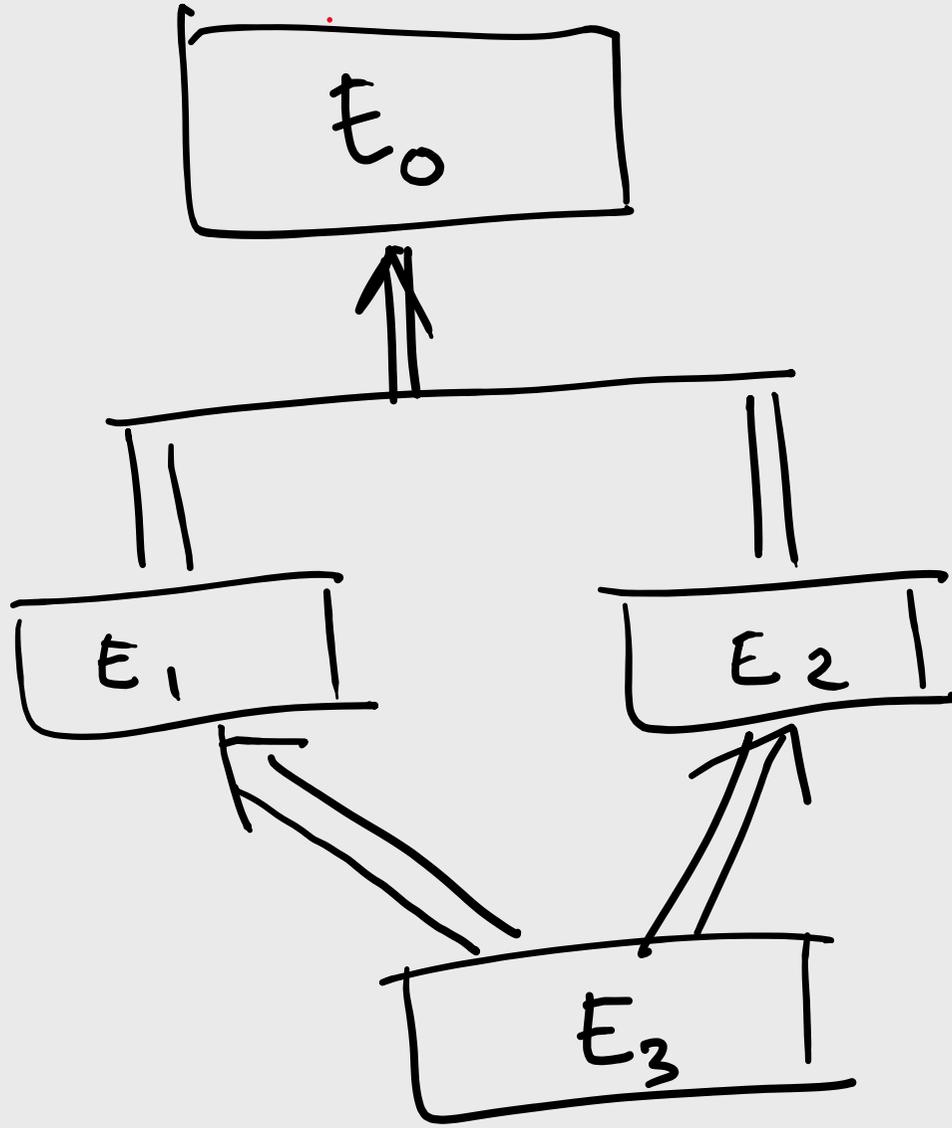
?



UNIVERSITA'  
CON ANCHE  
STUDENTI E  
LAVORATORI

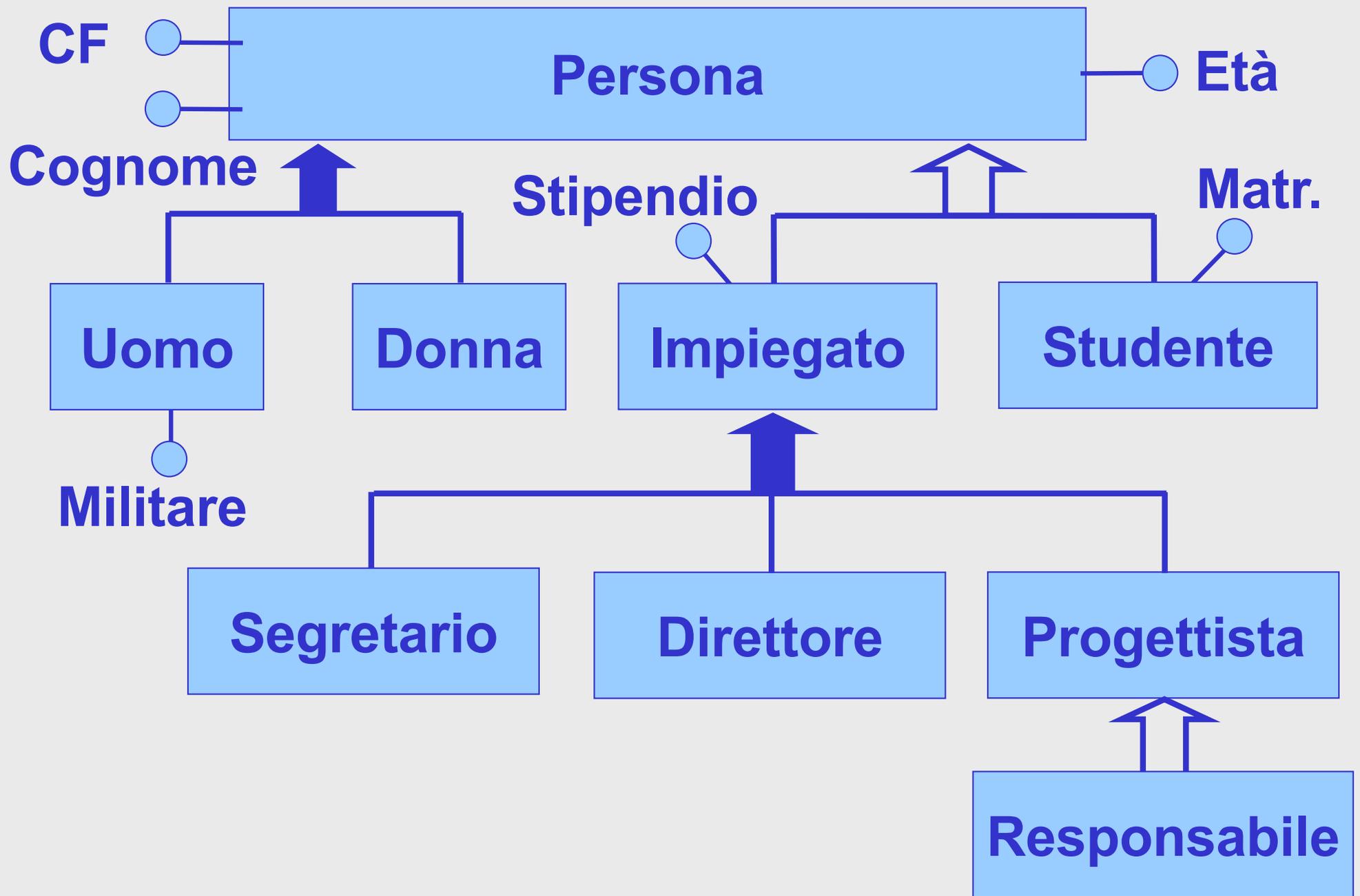


?



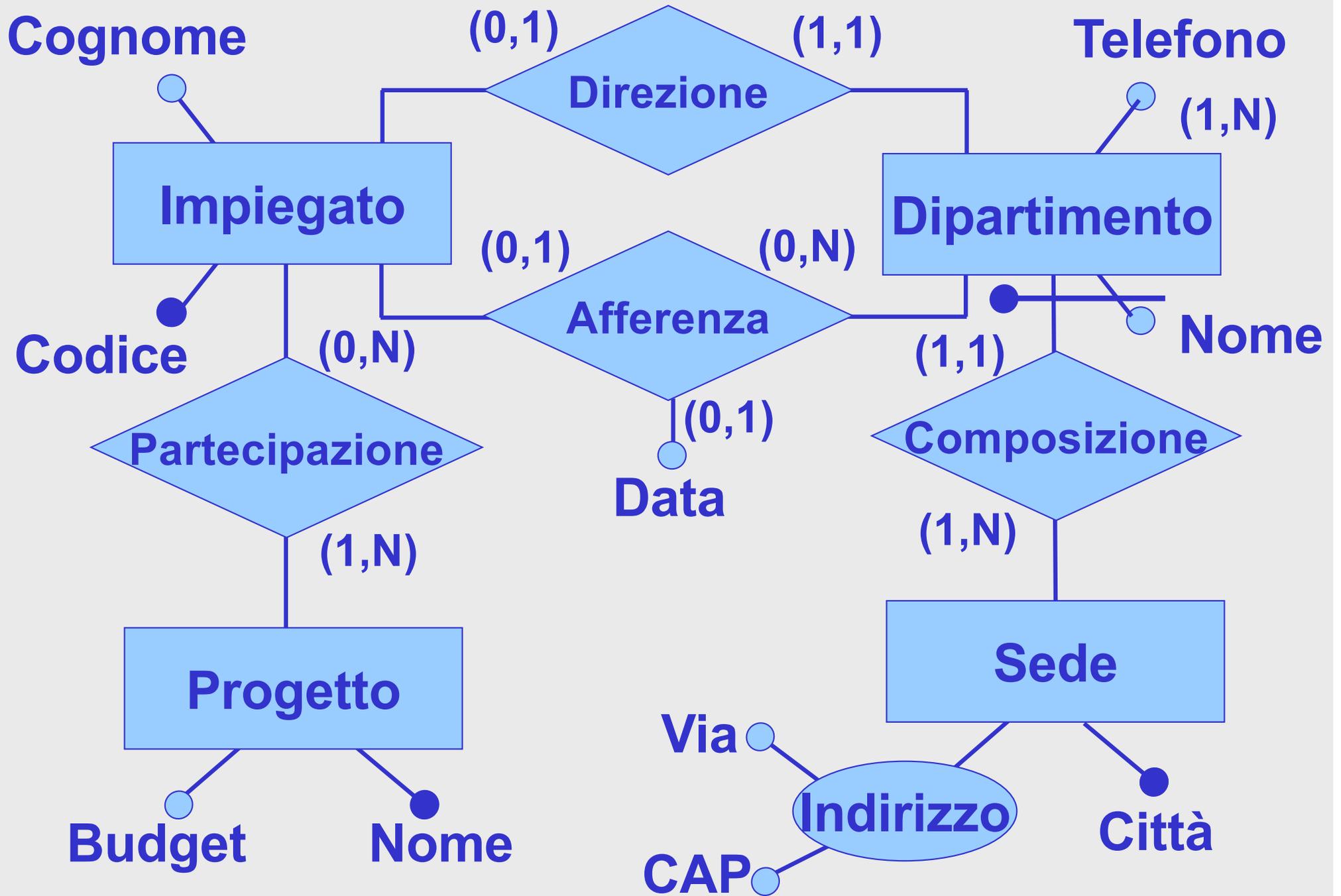
## Esercizio

- 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)



# Documentazione associata agli schemi concettuali

- dizionario dei dati
  - entità
  - relationship
- vincoli non esprimibili



## Dizionario dei dati (entità)

<b>Entità</b>	<b>Descrizione</b>	<b>Attributi</b>	<b>Identificatore</b>
Impiegato	Dipendente dell'azienda	Codice, Cognome, Stipendio	Codice
Progetto	Progetti aziendali	Nome, Budget	Nome
Dipartimento	Struttura aziendale	Nome, Telefono	Nome, Sede
Sede	Sede dell'azienda	Città, Indirizzo	Città

# Dizionario dei dati (relationship)

<b>Relazioni</b>	<b>Descrizione</b>	<b>Componenti</b>	<b>Attributi</b>
Direzione	Direzione di un dipartimento	Impiegato, Dipartimento	
Afferenza	Afferenza a un dipartimento	Impiegato, Dipartimento	Data
Partecipazione	Partecipazione a un progetto	Impiegato, Progetto	
Composizione	Composizione dell'azienda	Dipartimento, Sede	

# Vincoli non esprimibili

## Vincoli di integrità sui dati

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

# Modellazione dei dati in UML

- In alternativa al modello ER per la rappresentazione concettuale dei dati viene talvolta utilizzato il linguaggio/modello UML
- Si fa uso dei diagrammi delle classi
- Cambia la rappresentazione diagrammatica ma non l'approccio alla progettazione
- Non approfondiamo