



# Progettazione e sviluppo di applicazioni web

Azienda Speciale di Formazione “Scuola Paolo Borsa”  
Ing. Masciadri Andrea <masciadri.andrea@gmail.com>

---

# Design di una base di dati

# Ingredienti dei modelli concettuali

- **ASTRAZIONI**: capacità di evidenziare caratteristiche comuni ad insiemi di oggetti.
- Tre **ASTRAZIONI** di base per la rappresentazione della conoscenza:
  - classificazione;
  - aggregazione;
  - generalizzazione.

# Classificazione

- Capacità di definire classi di oggetti o fatti del mondo reale:
  - LIBRO;
  - BICICLETTA;
  - PERSONA;
  - APPARTAMENTO .
- Per ogni classe esiste un implicito “test di appartenenza” che consente di dire se un oggetto o fatto del mondo reale è una istanza della classe.

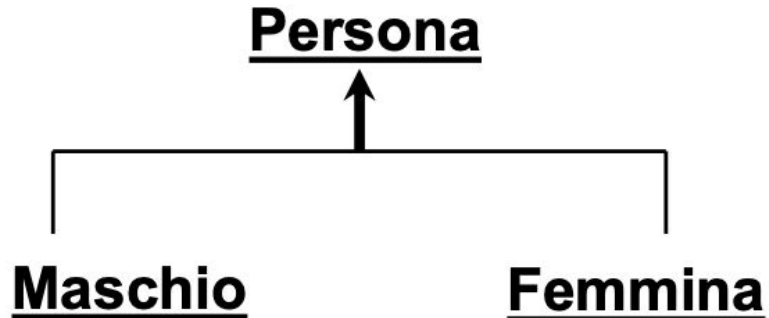
# Aggregazione

- Costruzione di una classe complessa aggregando classi più semplici (componenti):
  - BICICLETTA;
  - RUOTE;
  - PEDALI;
  - MANUBRIO.
- Per ogni componente si indica quante istanze sono presenti in una istanza della classe aggregata.

# Generalizzazione

Stabilisce legami di sottoinsieme fra classi:

- FEMMINA < PERSONA
- MASCHIO < PERSONA



# Modello Entità-Relazione

- Il modello Entity-Relationship "ER" (Peter P. Chen 1976) si è affermato come standard industriale di buona parte delle metodologie e degli strumenti per il progetto concettuale di basi di dati.
- Attenzione: Relationship = Associazione (però poi si dice informalmente “relazione”)

# Entità

- Rappresenta una classe di oggetti (es. automobili, impiegati, studenti) o di fatti (es. conti correnti, corsi universitari).
- Devono essere oggetti rilevanti per la applicazione.
- Ogni entità è caratterizzata da un nome.



# Simbologia grafica per rappresentare entità



**nome  
dell'entità**

**esempio:**

**studente**

# Relazione (o associazione)

- Rappresenta una aggregazione di entità di interesse per l'applicazione.
- Ogni istanza di una associazione è una ennupla tra istanze di entità (es. legame tra un automobile e il suo proprietario).
- Ogni associazione è caratterizzata da un nome.

# Simbologia grafica per rappresentare associazioni



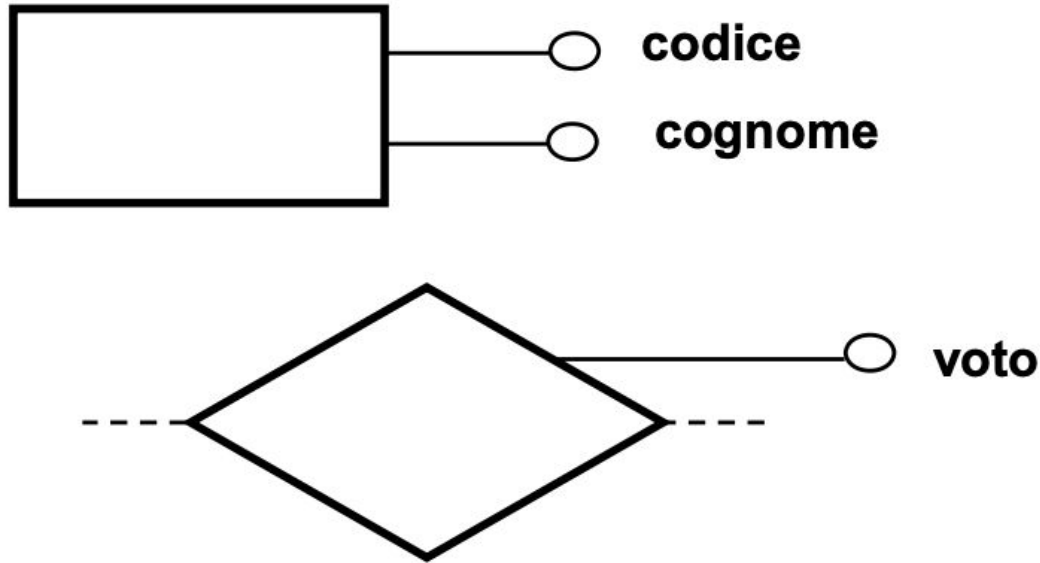
**esempio:**



# Attributi

- Rappresentano caratteristiche delle entità e delle associazioni di interesse per l'applicazione.
- Ogni attributo è caratterizzato da un nome.

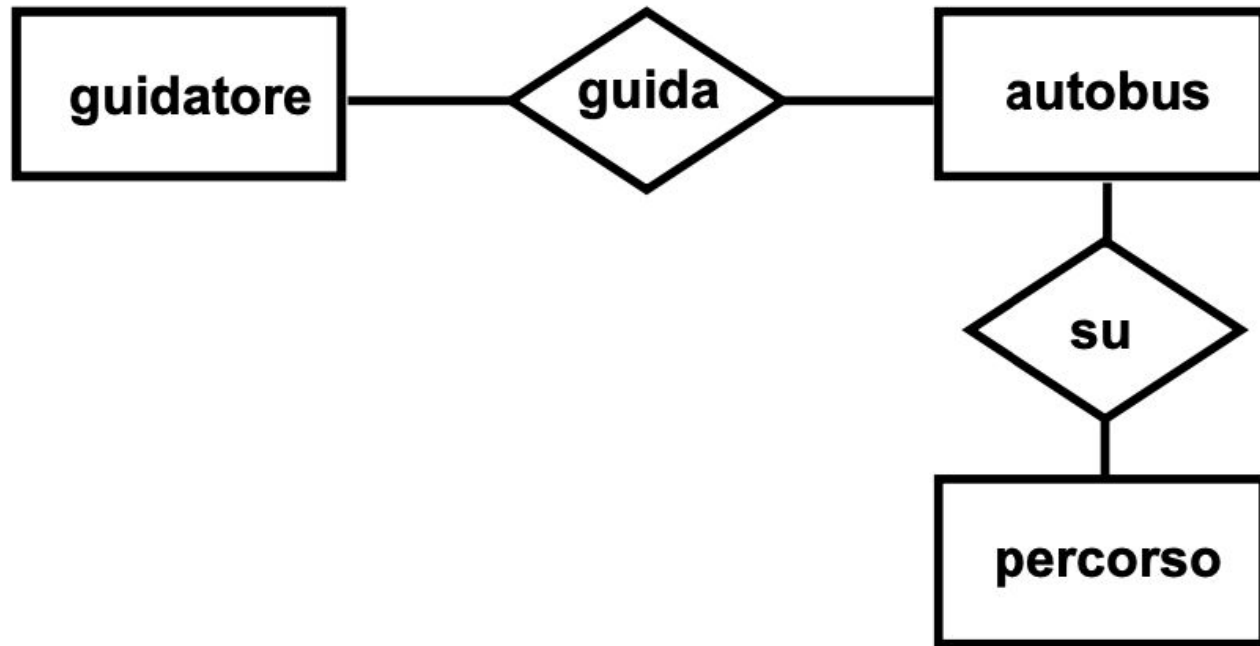
# Simbologia grafica per rappresentare attributi



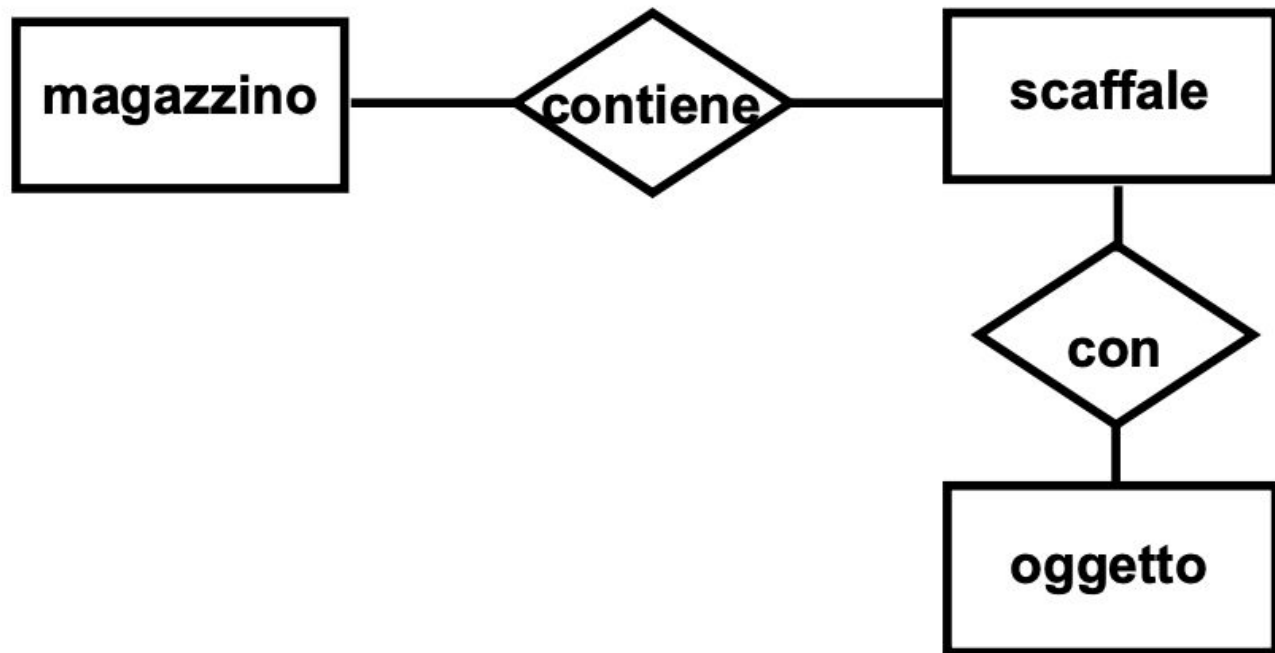
# Linee guida per il progetto

- Se il concetto è significativo per il contesto applicativo: entità.
- Se il concetto è descrivibile tramite un dato elementare: attributo.
- Se il concetto definisce un legame tra entità: associazione.

# Esempio: gestione viaggi

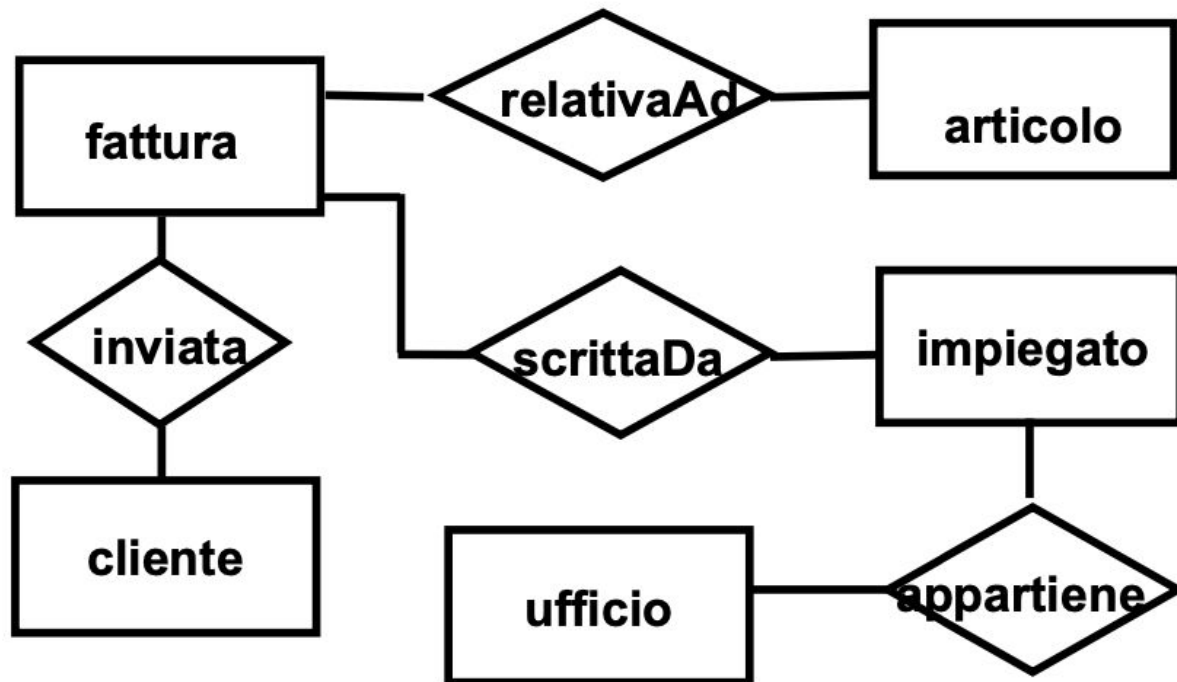


# Esempio: gestione magazzino





# Esempio: gestione fatture



# Cardinalità delle associazioni

- Per cardinalità si intende un vincolo sul numero di istanze di associazione cui ciascuna istanza di entità deve partecipare.
- È una coppia (MIN-CARD, MAX-CARD):
  - MIN-CARD = 0 (opzionale)  
= 1 (obbligatoria)
  - MAX-CARD = 1 (uno)  
= N (molti)
- In base alla sola cardinalità massima si hanno associazioni uno-uno, uno-molti, molti-molti.

# Associazione 1:1

- un reparto deve essere diretto da uno e un solo direttore (1,1)
- un direttore deve dirigere uno ed un solo reparto (1,1)



# Associazione 1:1 con opzionalità

- un reparto può essere diretto da uno e uno solo direttore (0,1)
- un direttore deve dirigere uno ed un solo reparto (1,1)



# Associazione 1:N

- un ripiano può contenere molti oggetti (0,n)
- un oggetto può essere contenuto al più su un ripiano (0,1)



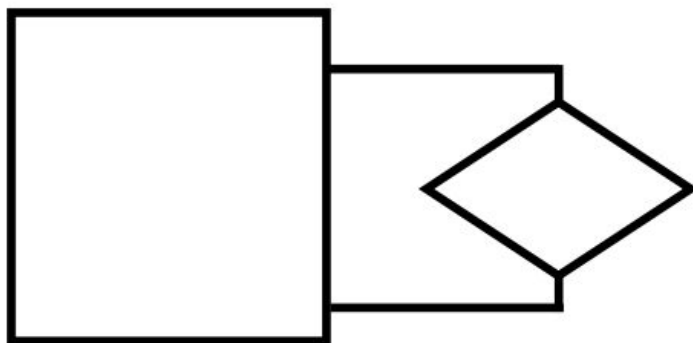
# Associazione N:M

- un progetto può essere fatto da molti ingegneri (0,n)
- un ingegnere deve partecipare ad uno o più progetti (1,m)

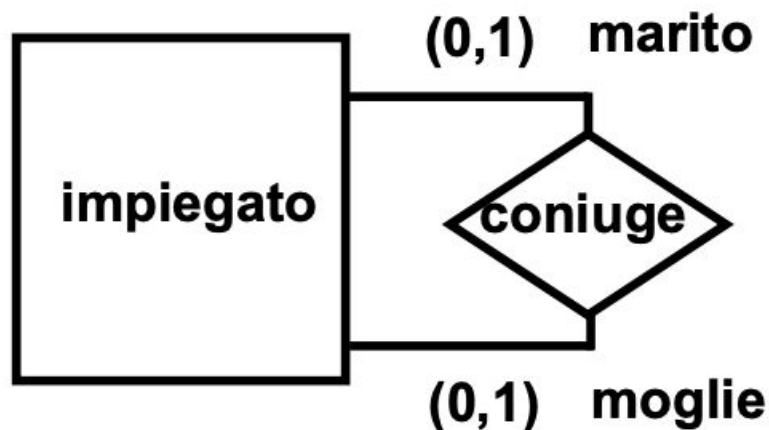


# Auto-associazioni

- Associazioni aventi come partecipanti istanze provenienti dalla stessa entità (chiamate anche “ad anello”).



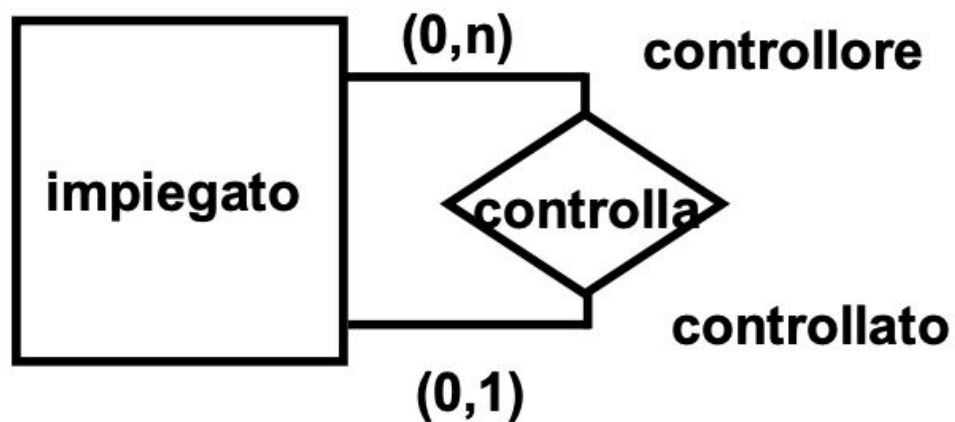
# Auto-associazioni 1:1



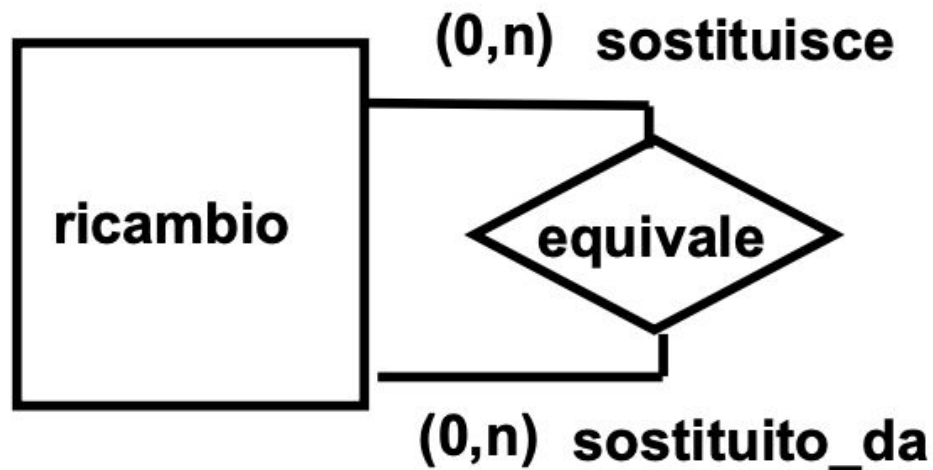
può essere riportato il “ruolo” sul ramo



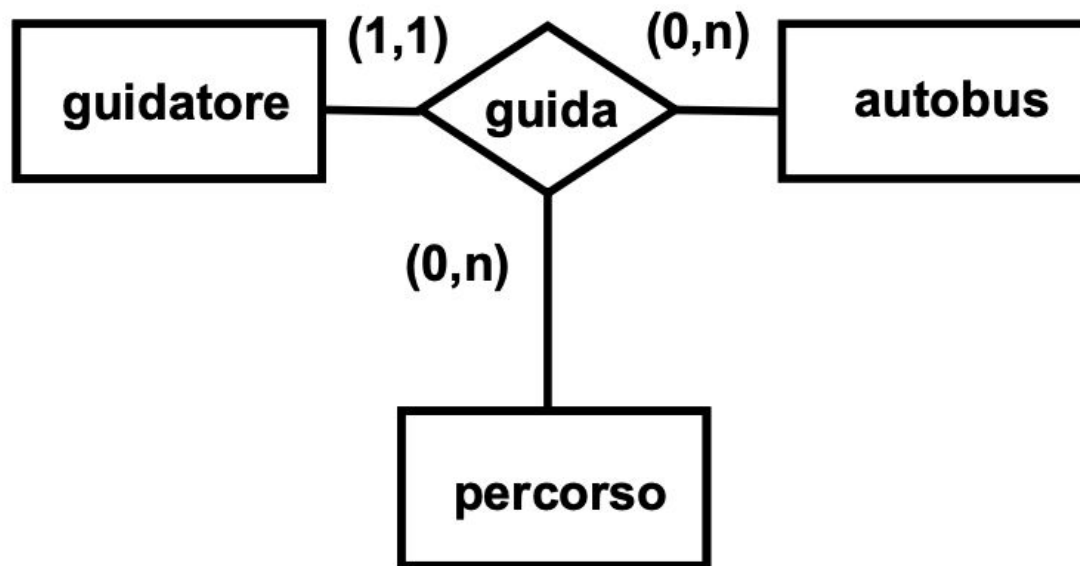
# Auto-associazioni 1:N



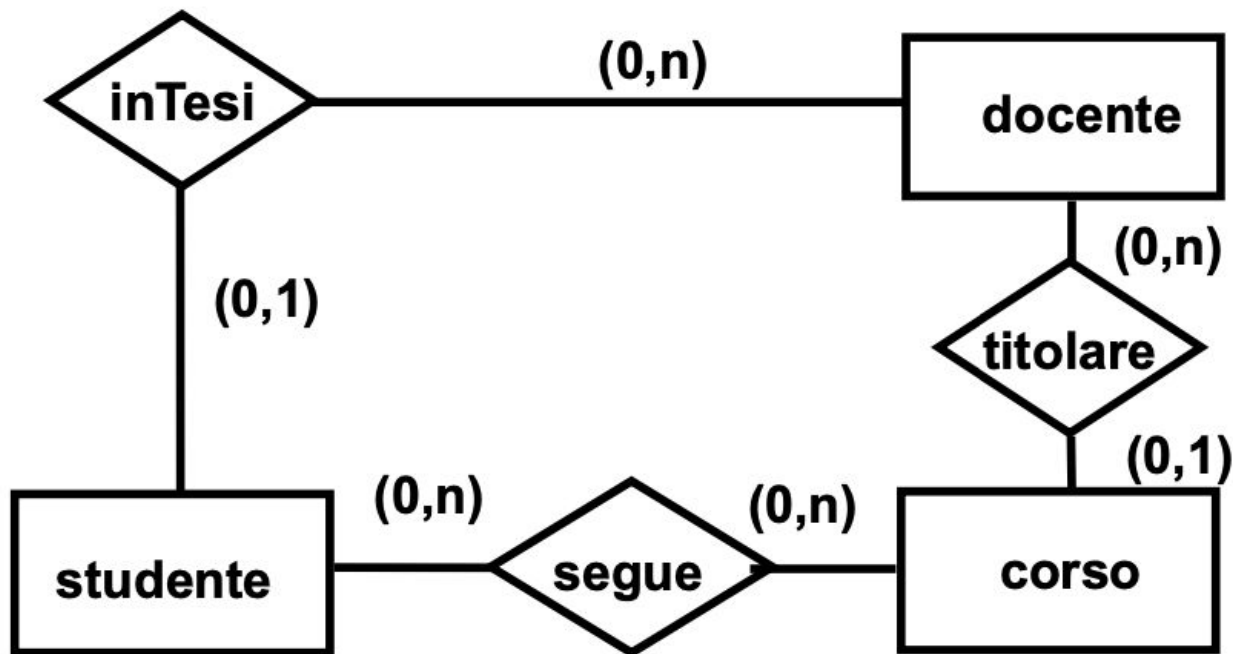
# Auto-associazioni N:M



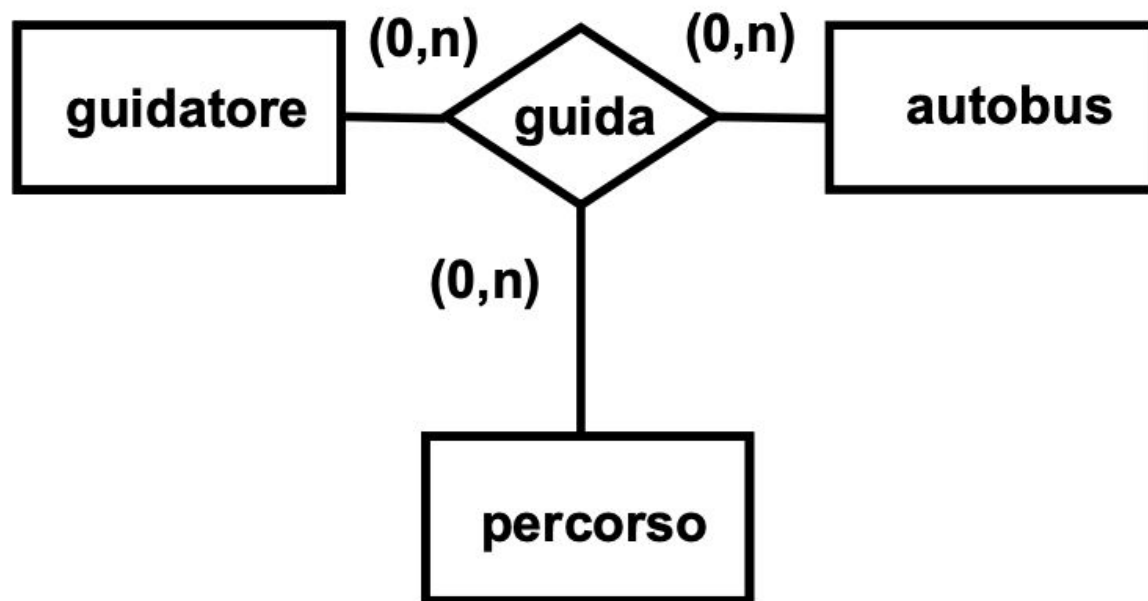
# Associazioni ternarie



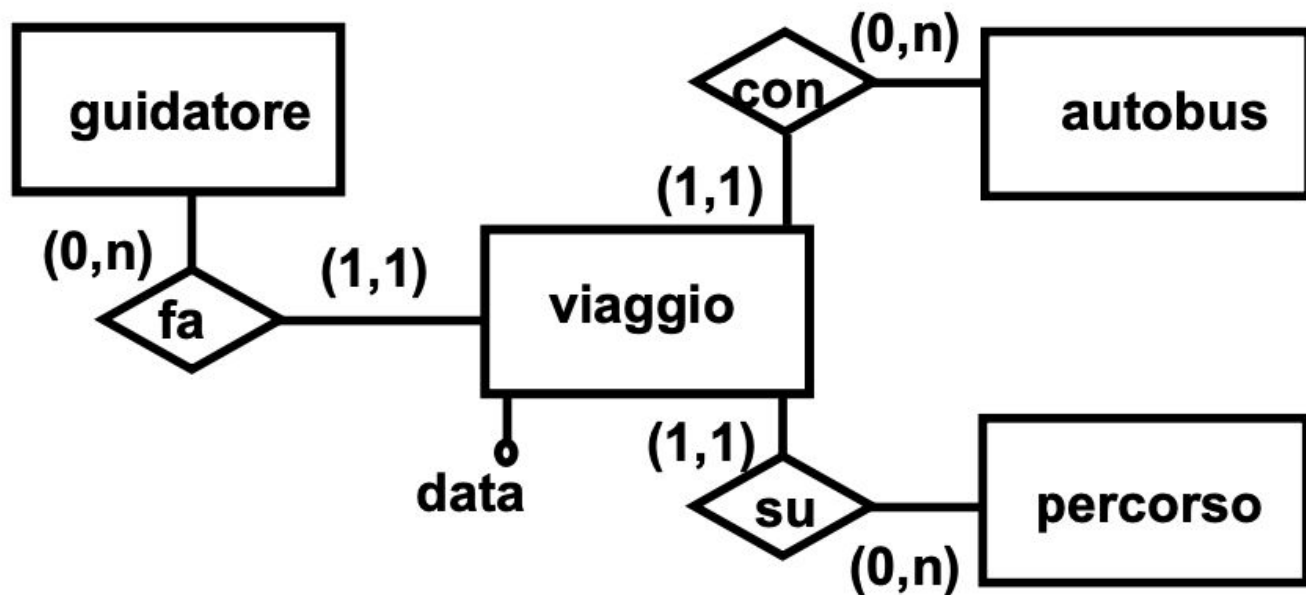
# Esempio: università



# Esempio: gestione autobus



# Esempio: gestione autobus



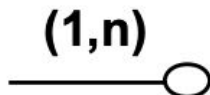
# Cardinalità degli attributi

Una prima classificazione:

- attributo scalare (semplice, ad un sol valore)
  - es.: matricola, cognome, voto



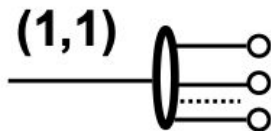
- attributo multiplo (sono ammessi n valori)
  - es.: qualifica, titolo, specialità



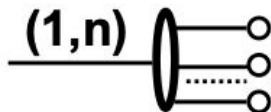
- il simbolo  $(n,m)$  esprime la cardinalità dell'attributo.

# Attributi composti

- Attributo composto
  - es.: data (gg,mm,aaaa), indirizzo (via, numero civico, città, provincia, cap)



- Attributo multiplo composto
  - es.: telefono (stato, città, numero)





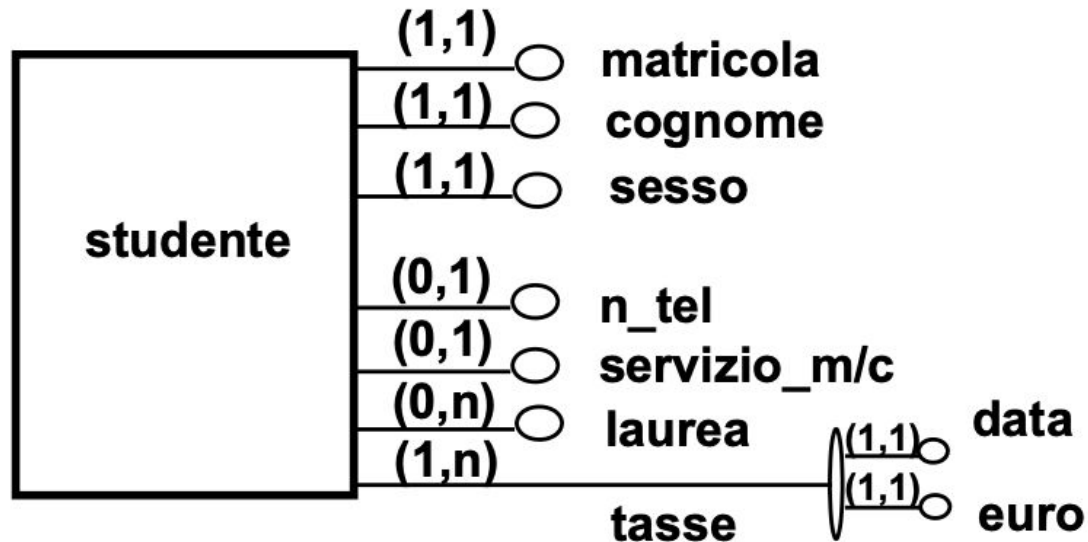
# Opzionalità

- Attributo opzionale (è ammessa la “non esistenza del valore”).
  - es.: tel., qualifica, targa




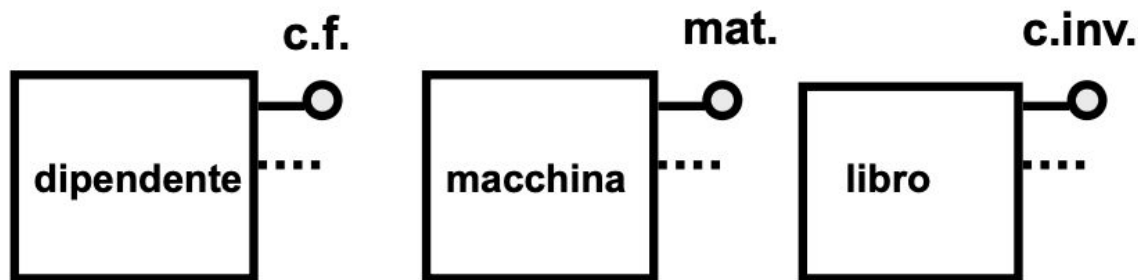
# Esempio

Esempio:



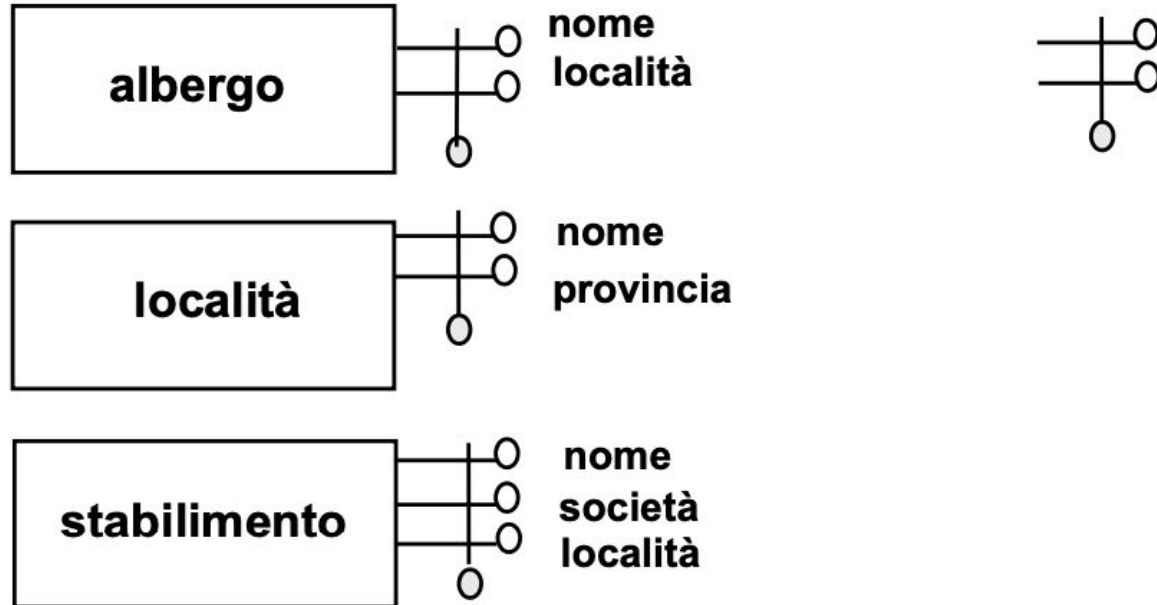
# Identificatore

- Un identificatore:
  - caratterizza in modo univoco ciascuna singola istanza di entità
  - non è modificabile (in generale...)
  - ha il simbolo 



# Identificatori composti

- L'identificatore di un'entità può essere composto.

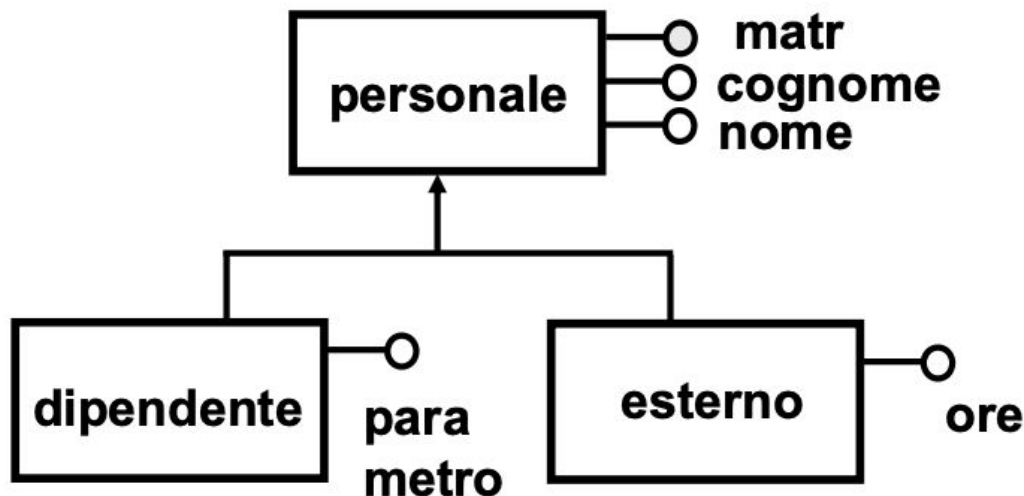


# Gerarchie di generalizzazione

- Una gerarchia di generalizzazione è un legame logico tra un'entità padre  $E$  ed alcune entità figlie  $E_1 E_2 \dots E_n$  dove:
  - $E$  è la generalizzazione di  $E_1 E_2 \dots E_n$
  - $E_1 E_2 \dots E_n$  sono specializzazioni di  $E$tale per cui:
  - ogni istanza di  $E_k$  è anche istanza di  $E$
  - una istanza di  $E$  può essere una istanza di  $E_k$
- Le entità figlio ereditano le proprietà (attributi, relazioni, identificatori) dell'entità padre.

# Esempio di gerarchia

- Un'azienda si avvale dell'opera di professionisti esterni, quindi il suo personale si suddivide in esterni e dipendenti.



# Proprietà delle gerarchie

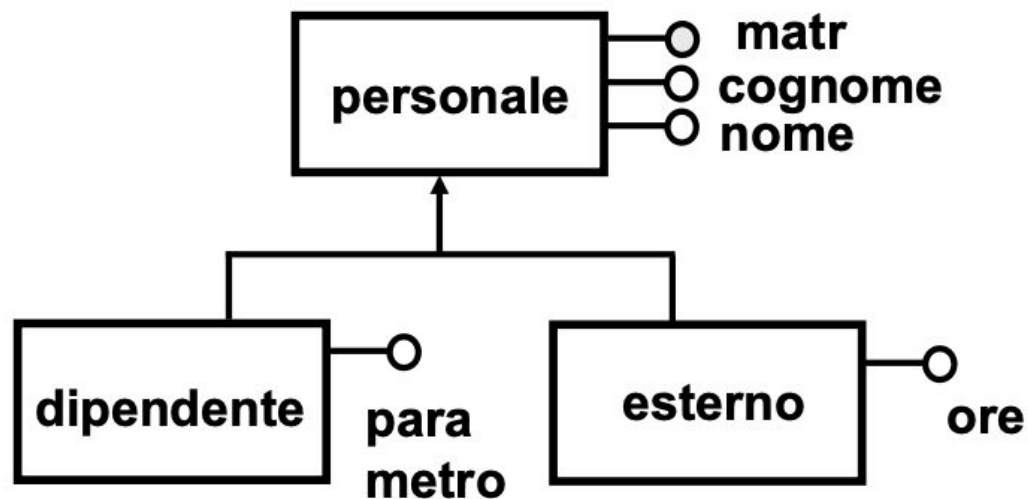
- $t$  sta per totale: ogni istanza dell'entità padre deve far parte di una delle entità figlie:
  - nell'esempio il personale si divide (completamente) in esterni e dipendenti.
- $p$  sta per parziale: le istanze dell'entità padre possono far parte di una delle entità figlie:
  - nell'esempio i cacciatori e pescatori sono un sottoinsieme dei cittadini.

# Proprietà delle gerarchie

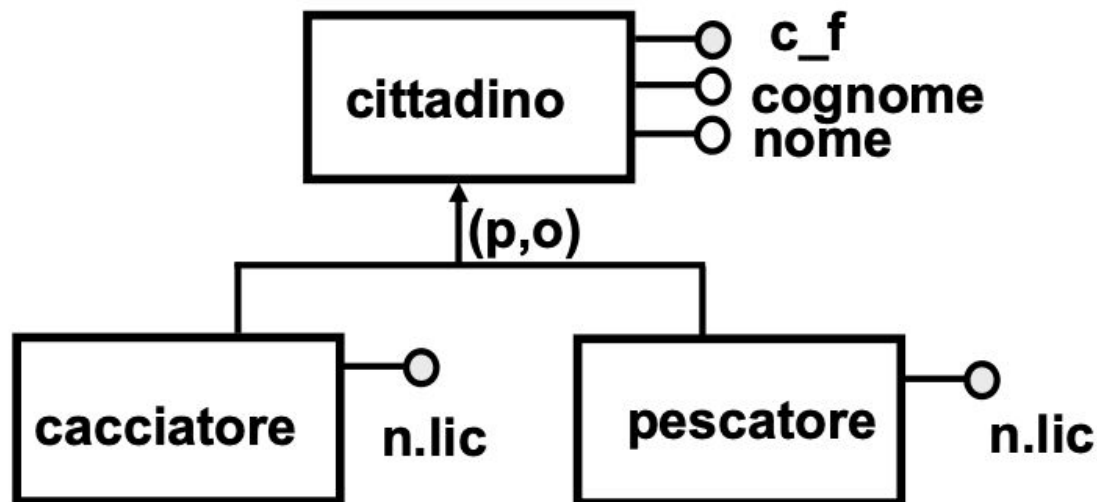
- **e** sta per esclusiva: ogni istanza dell'entità padre non può far parte di più di una delle entità figlie:
  - nell'esempio si esclude che una istanza di personale possa appartenere ad entrambe le sottoclassi.
- **o** sta per overlapping: ogni istanza dell'entità padre può far parte di più entità figlie:
  - nell'esempio un cittadino può essere al tempo stesso cacciatore e pescatore.



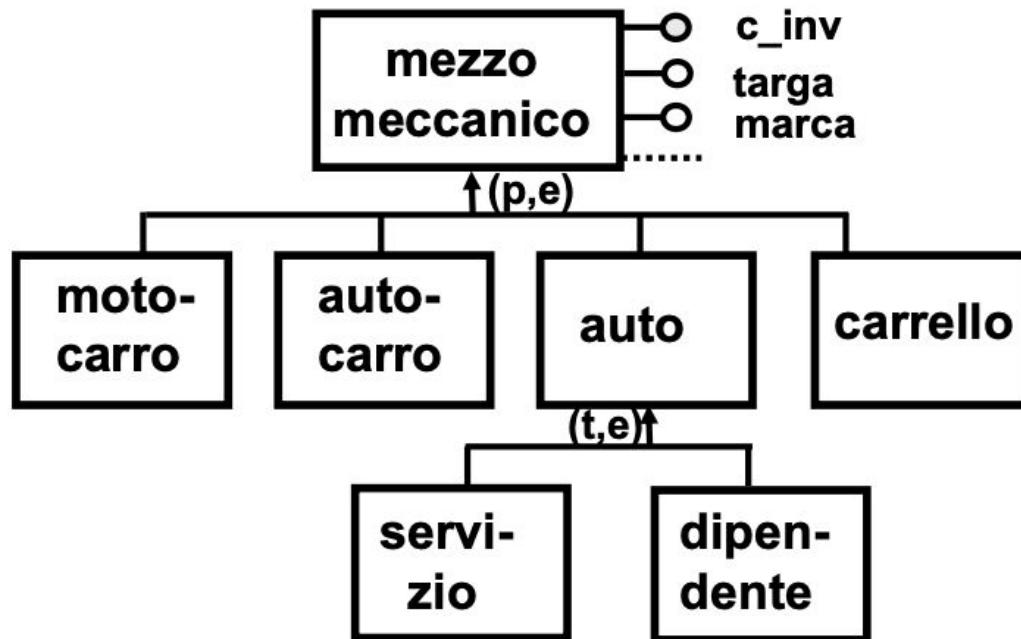
# Default: (t,e)



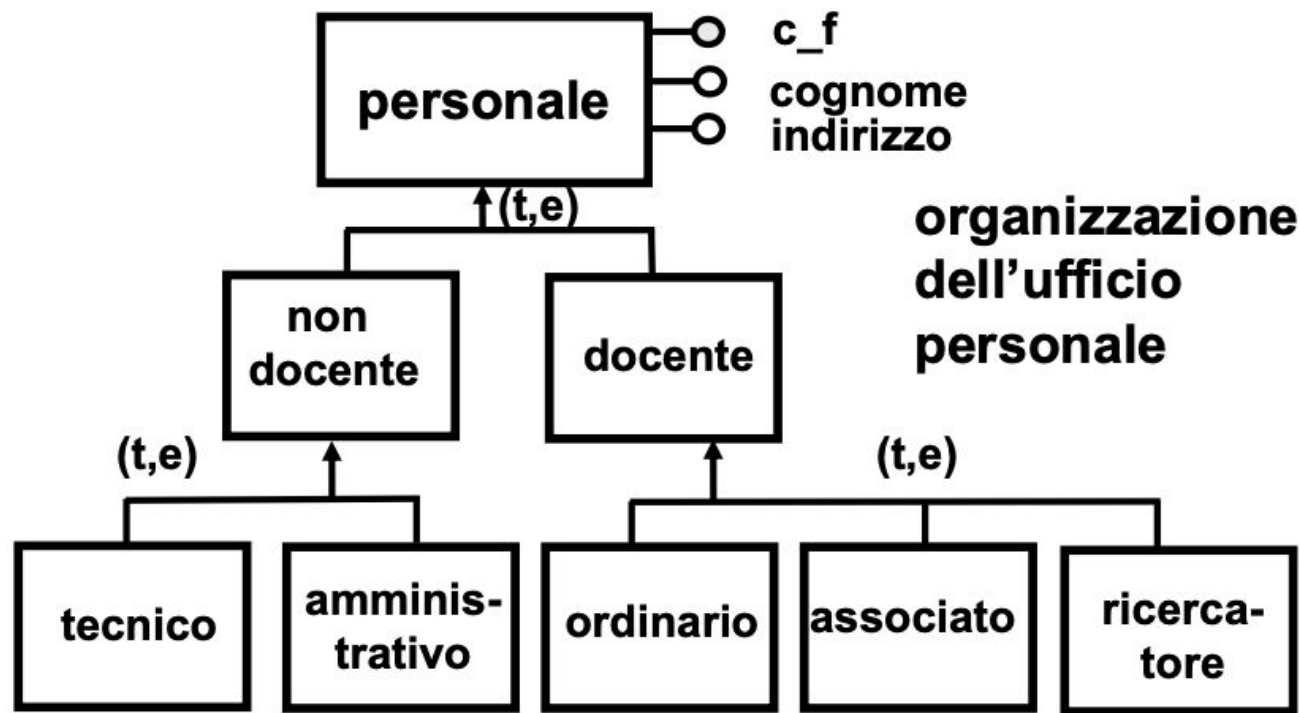
# Indicazione della proprietà



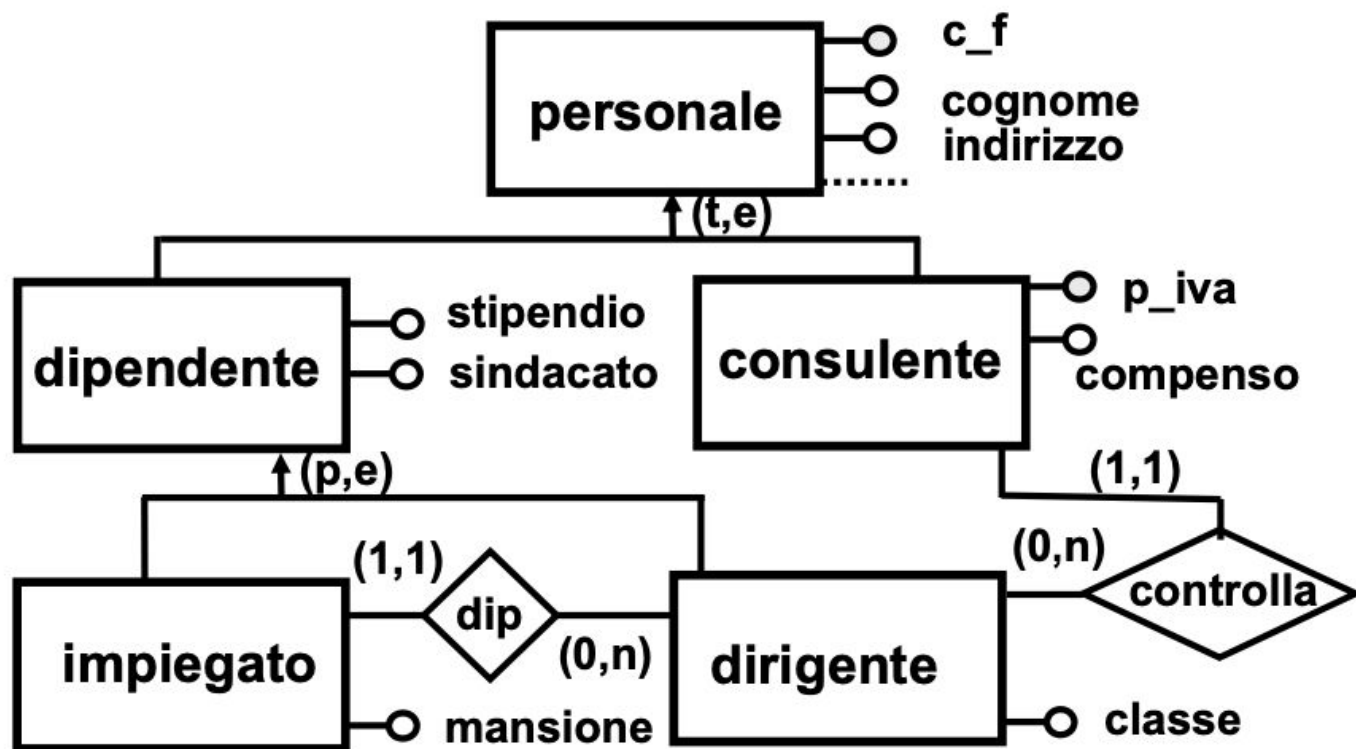
# Esempio: parco mezzi meccanici



# Esempio: università



# Esempio: personale d'azienda





**Domande?**

[masciadri.andrea@gmail.com](mailto:masciadri.andrea@gmail.com)

Ora provate voi..



## References

-