

- Attività di sviluppo di sistemi informativi
 - Formulazione e definizione delle problematiche alle quali il sistema informativo deve rispondere
- Attività iterativa
 - Creazione di un modello “completo” a partire da modelli semplici

“Ciclo dell'innovazione”



Il ciclo di progettazione

- **Analisi: cosa?**
 - interazione con gli utenti (*knowledge extraction*)
 - studio di eventuali DB preesistenti
 - modellazione
- **Progettazione: come?**
 - definizione del flusso di dati
 - selezione degli strumenti *hardware* e *software*
 - definizione delle funzioni del *software*
 - automazione

- **Analisi dei requisiti**
 - progettazione concettuale
 - raccolta, definizione ed esame delle specifiche tecniche
 - descrizione dei dati secondo un modello astratto
 - dizionario dei dati
- **Progettazione logica**
 - schema logico della base di dati
 - definizione delle relazioni e dei rapporti tra di esse
 - schema di riferimento per utenti, programmatori e amministratori
- **Progettazione fisica**
 - definizione delle caratteristiche fisiche del sistema

Modello concettuale

- Vengono spesso utilizzati fin dalle prime fasi del ciclo produttivo di un DB
- Utili a comunicare informazioni anche a non specialisti (*knowledge extraction*)
- Molto più espressivi e intuitivi di un modello logico

- A livello **estensionale**:
 - enfasi sui dati (istanze), molto variabile nel tempo
- A livello **intensionale**:
 - enfasi sullo **schema** della struttura dei dati, stabile nel tempo
 - alto livello di astrazione

Modello *Entity-Relationship*

- È il più diffuso paradigma per la produzione di modelli concettuali (Chen, 1976)
- Ne esistono numerose versioni (diverse simbologie)...

I "blocchi da costruzione":

- entità
- relazioni
- attributi
- identificatori
- generalizzazioni e sottoinsiemi
- ...

Entità

- Classe di oggetti appartenenti al *miniworld*
- Collezione omogenea
- Ogni entità ha un **nome** univoco
 - il nome deve essere **espressivo**
 - notazioni e convenzioni (uso del singolare...)
- Istanza di un'entità
 - ciascun elemento della classe

Rappresentazione grafica

Esemplare

Specie

Studente

Corso

Relazione (*relationship*)

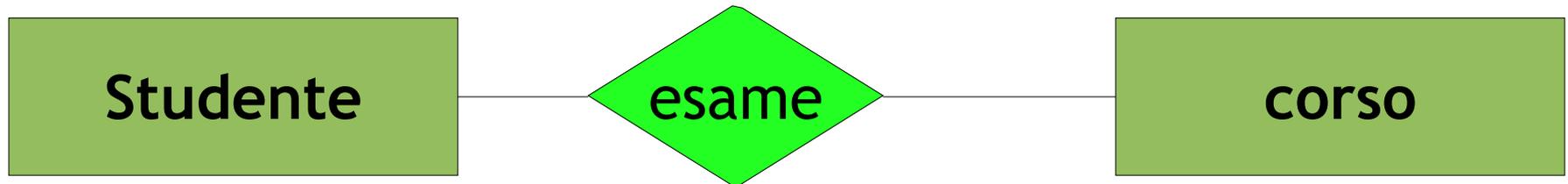
Spesso non tradotto, in italiano '*relation*' e '*relationship*' generano ambiguità.

Relazione = *relation*

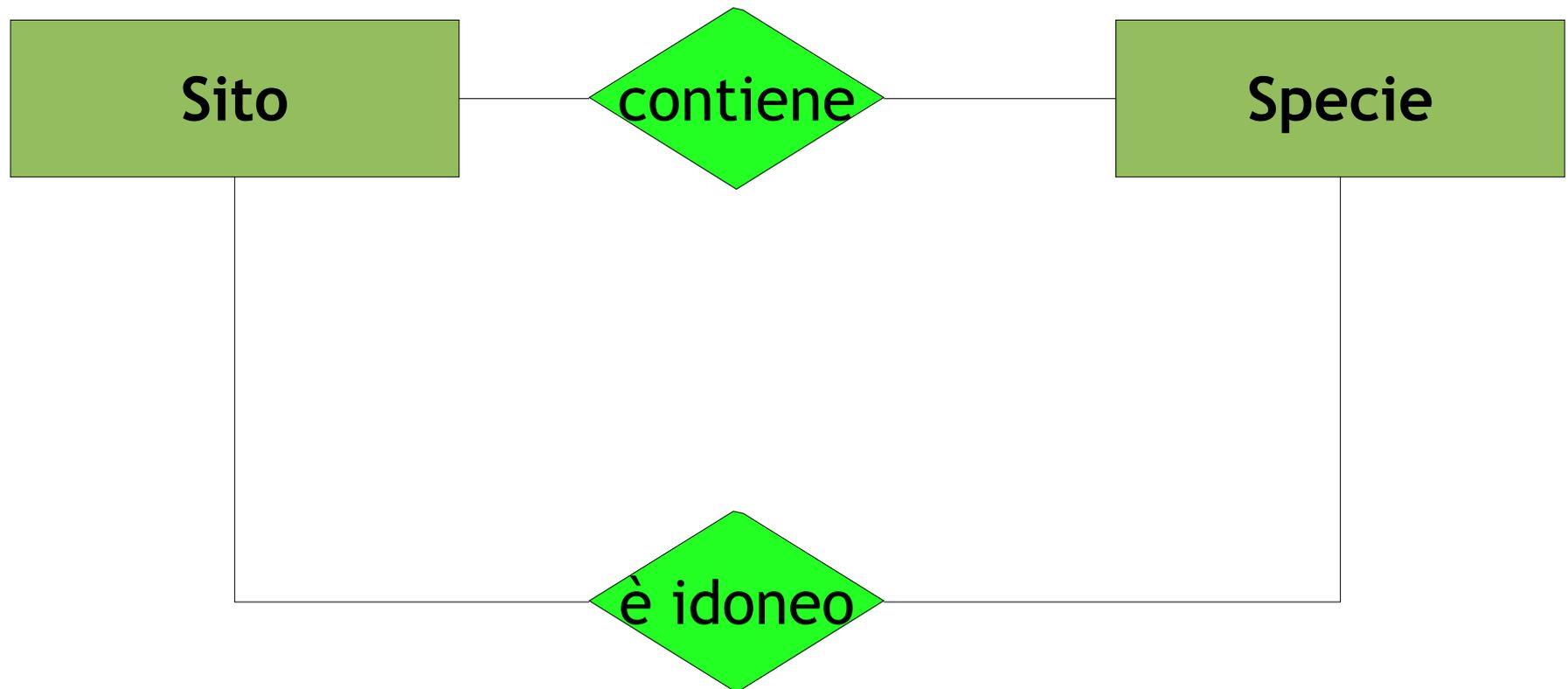
Associazione, correlazione = *relationship*

- Nesso logico tra due (o più) entità
- Classe di "fatti di interesse"
- Ennupla di istanze di entità
 - un'istanza per ciascuna entità implicata nella *relationship*
- Insieme di ennuple omogenee
 - per tipo (entità di origine)
 - per significato (rispetto al modello logico)

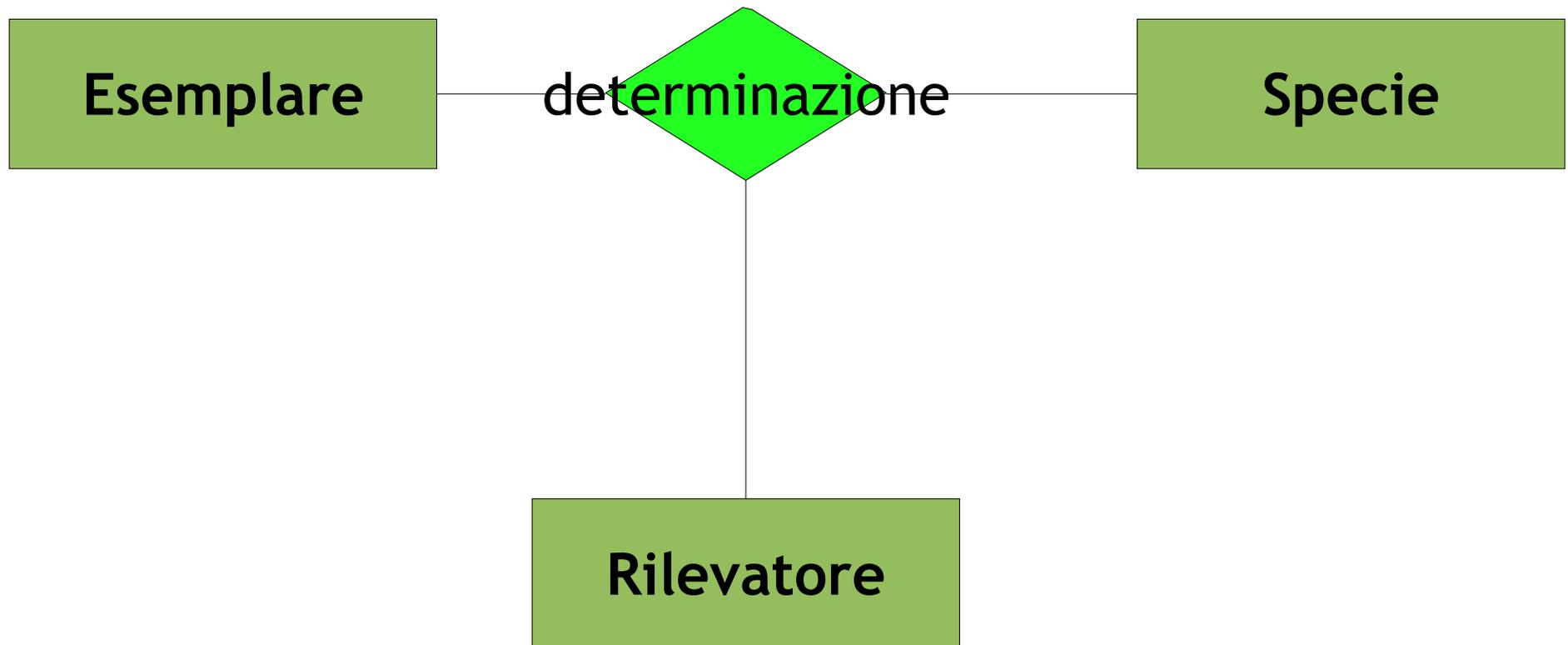
Rappresentazione grafica



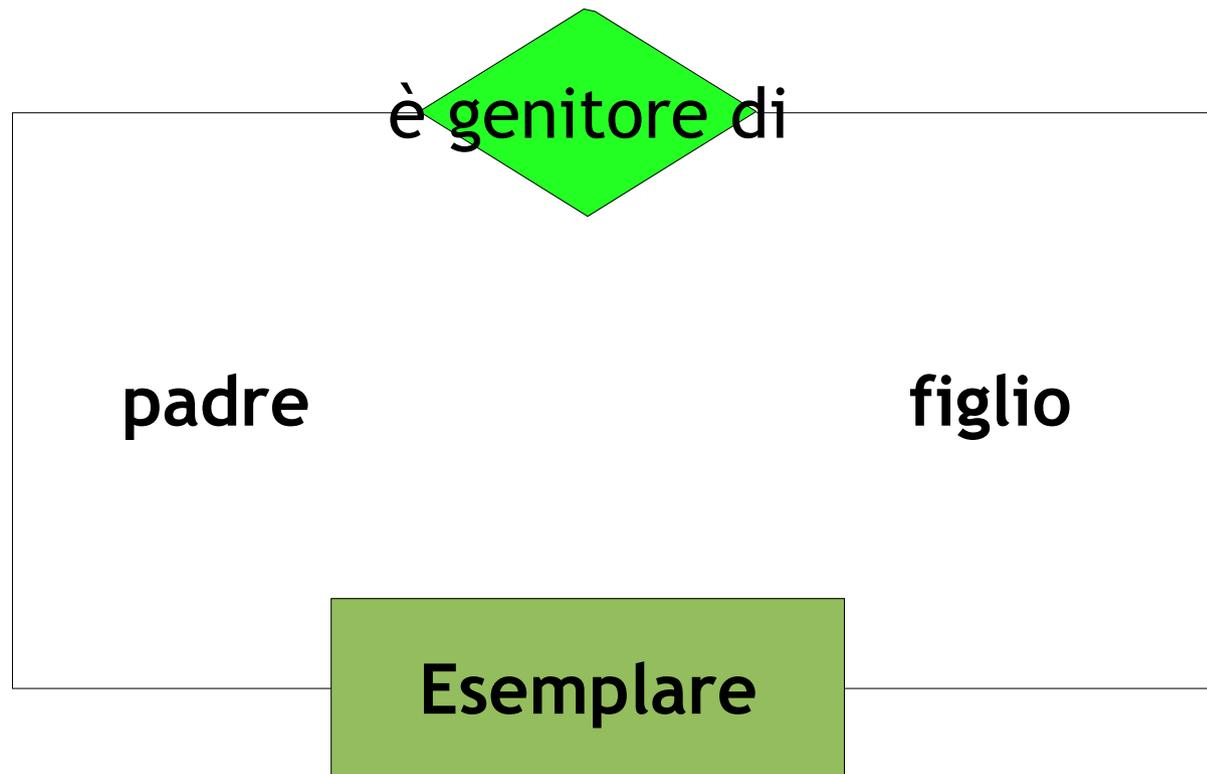
Relazioni multiple



Relazioni ternarie



Relazioni ricorsive



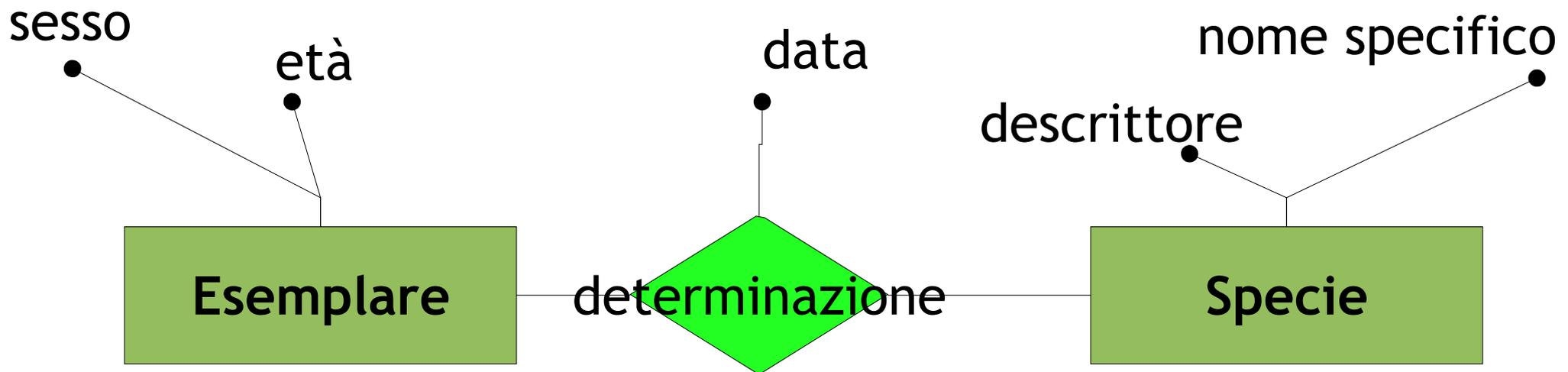
Relationship come relazione

- Le istanze di una *relationship* sono a loro volta una relazione
 - vale sempre il concetto di sottoinsieme del prodotto cartesiano degli insiemi di istanze implicate nella relazione
 - non possono esistere ennuple di ennuple ripetute

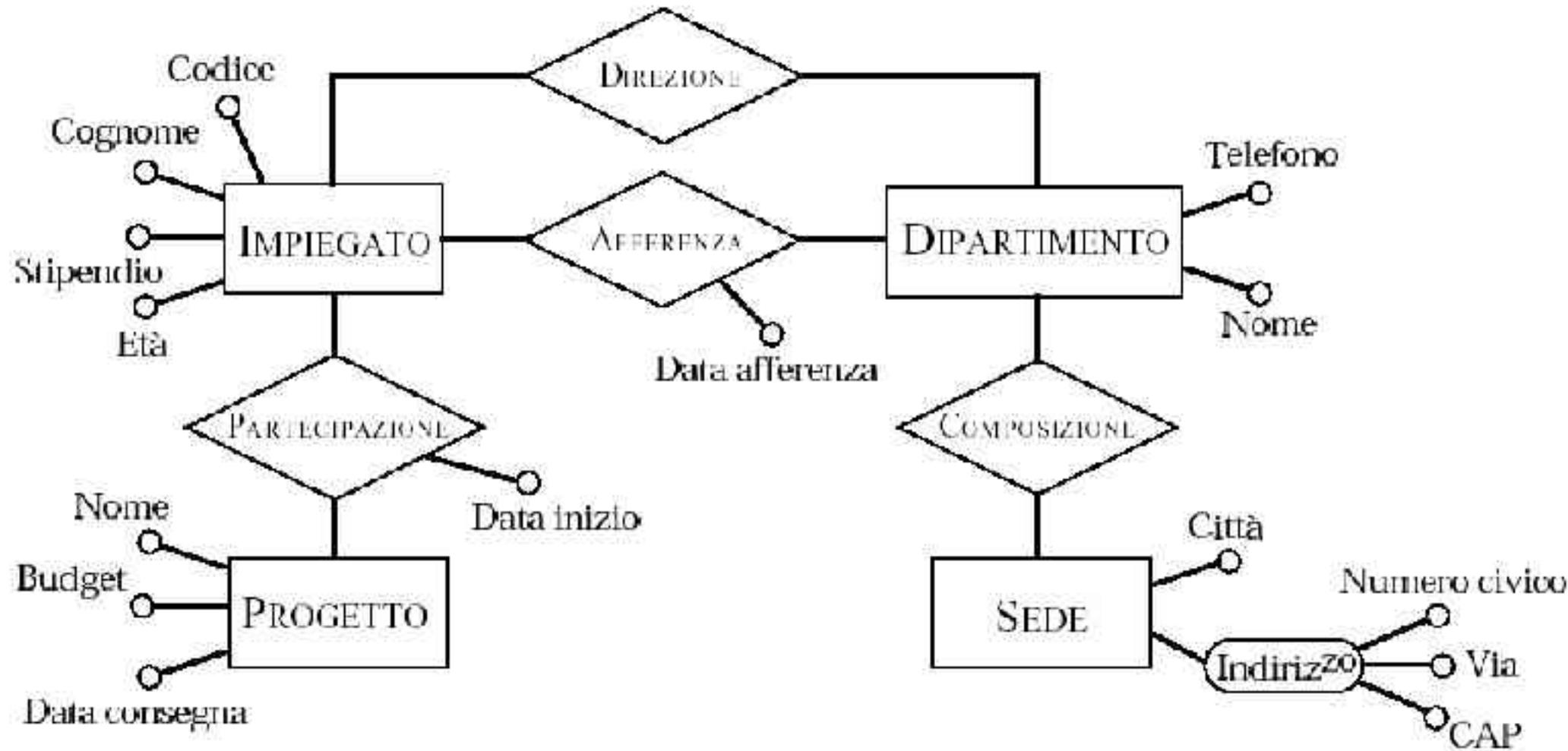
Attributo

- Proprietà elementare di una entità, o di una *relationship*
- Associa un valore a ciascuna istanza

Se una entità definisce il significato di una ennupla, i suoi attributi ne definiscono la struttura



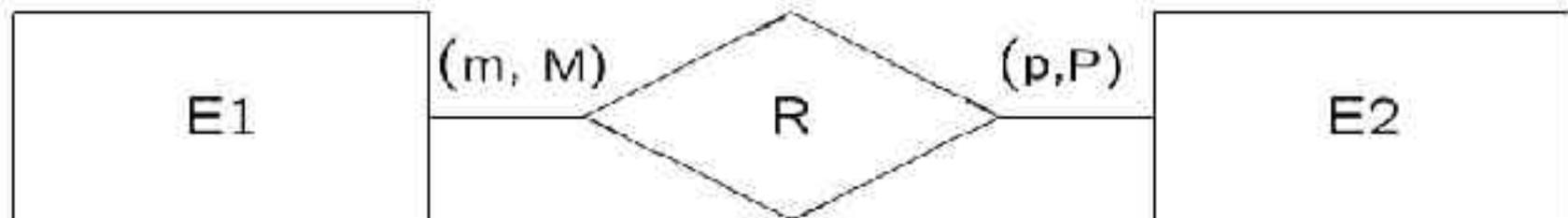
Un esempio (notazione "classica")



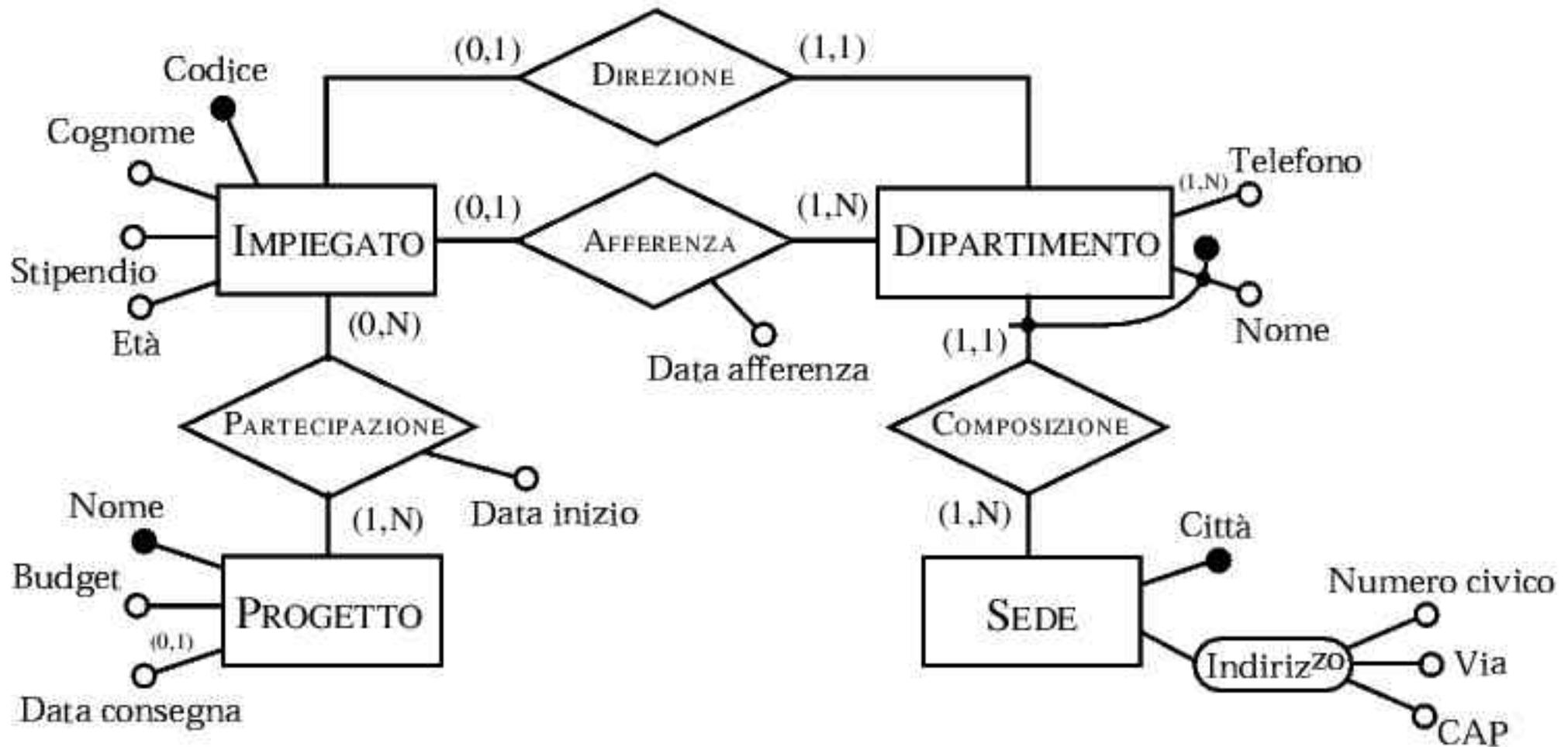
Cardinalità

Una *relationship* pone in corrispondenza un numero definito di istanze: occorrerebbe specificare quante!

- Le cardinalità minime e massime vanno esplicitate per entrambi i lati di una *relationship*
- Valori tipici:
 - 0: *relationship* opzionale
 - 1: uno e uno solo
 - N: più di uno
- 3 tipi possibili di *relationship*: 1:1; 1:N; N:N



Un esempio

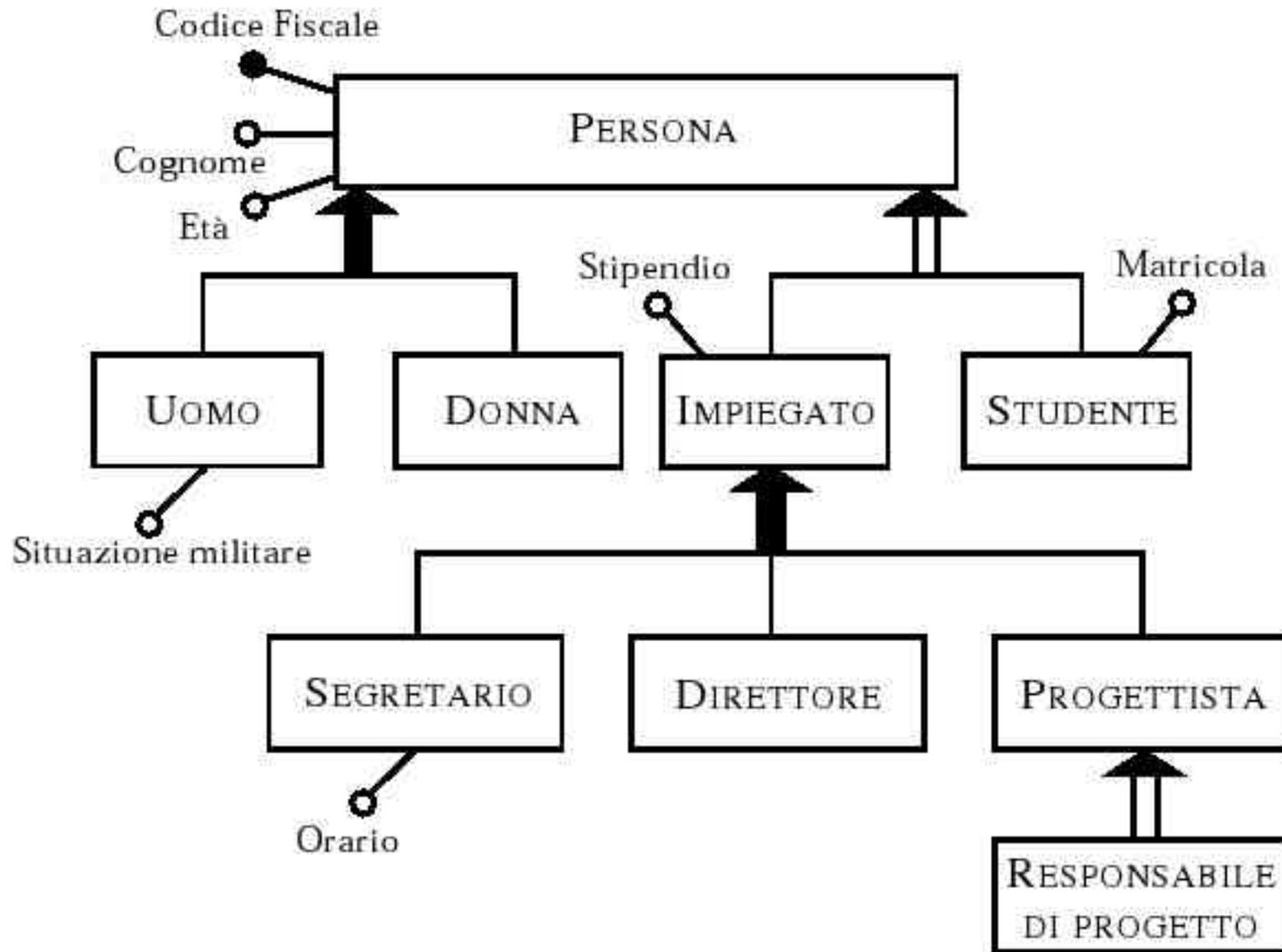


Consente di definire “casi generali” e “casi specifici” mediante gerarchie di entità

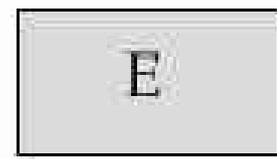
- Più entità E_1, E_2, \dots, E_n vengono poste in relazione con un'entità E , che le ricomprende come casi particolari
- Ogni proprietà di E vale anche per E_1, E_2, \dots, E_n
- Ogni istanza di E_i è anche un'istanza di E
- Ogni istanza di E è, al più, un istanza di E_i
- E_1, E_2, \dots, E_n sono *sottotipi* (o *specializzazioni*) di E , e ne ereditano tutte le proprietà

“Uomo” e “Donna” sono due casi particolari di “persona”

Gerarchia di generalizzazione



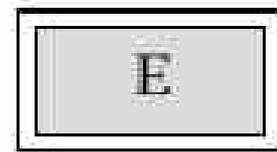
Notazione classica



Entity Set



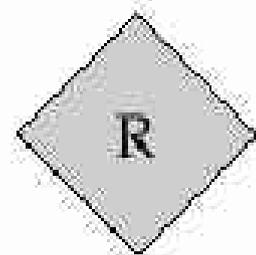
Attribute



Weak Entity Set



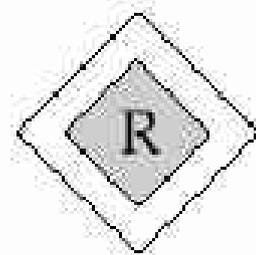
Multivalued Attribute



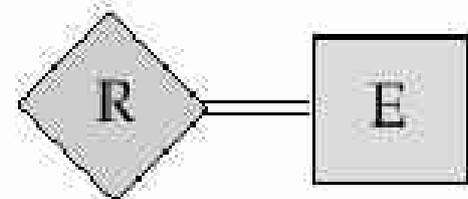
Relationship Set



Derived Attribute



Identifying Relationship Set for Weak Entity Set



Total Participation of Entity Set in Relationship

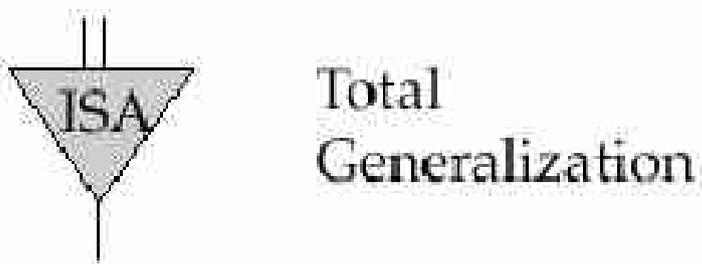
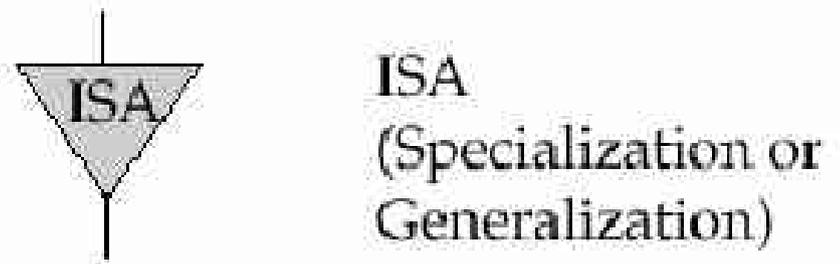
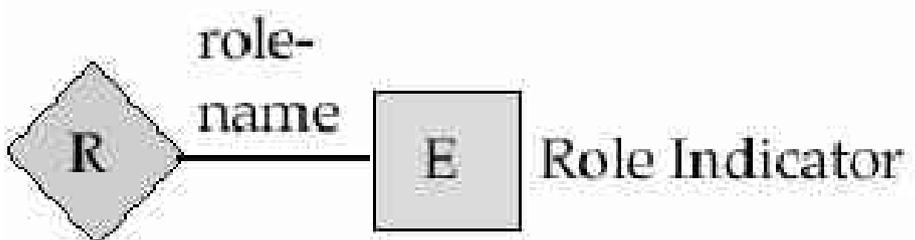
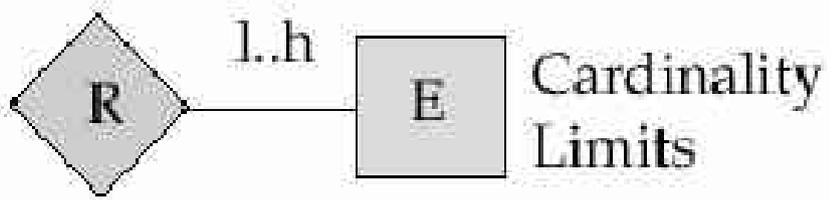
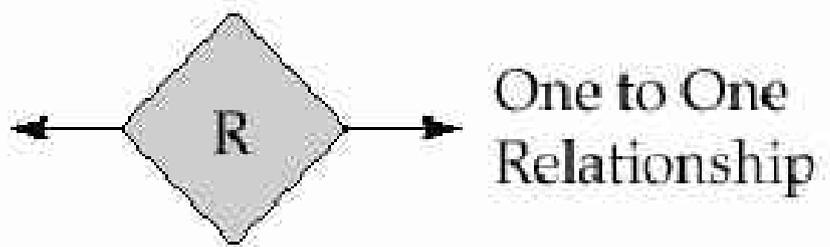
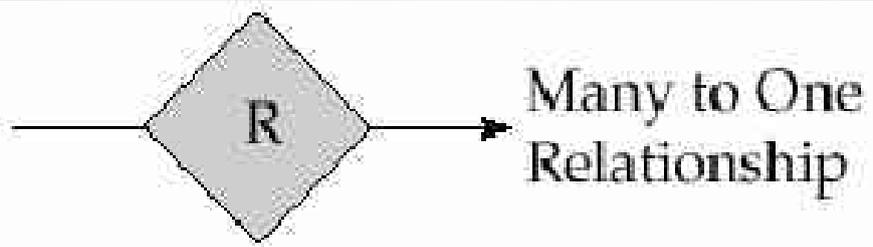
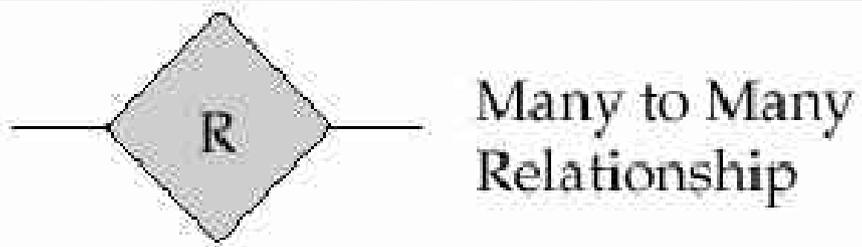


Primary Key



Discriminating Attribute of Weak Entity Set

Notazione classica

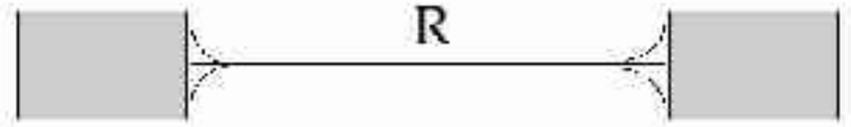


Notazione classica

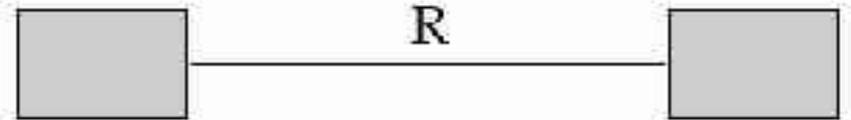
Entity set E with attributes A1, A2, A3 and primary key A1

E
A1
A2
A3

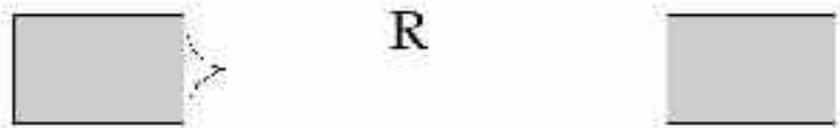
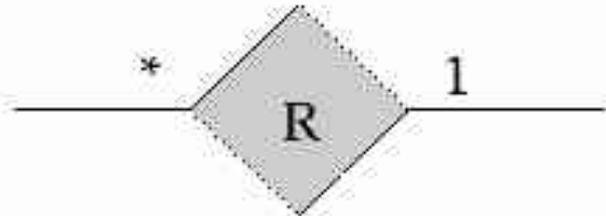
Many to Many Relationship



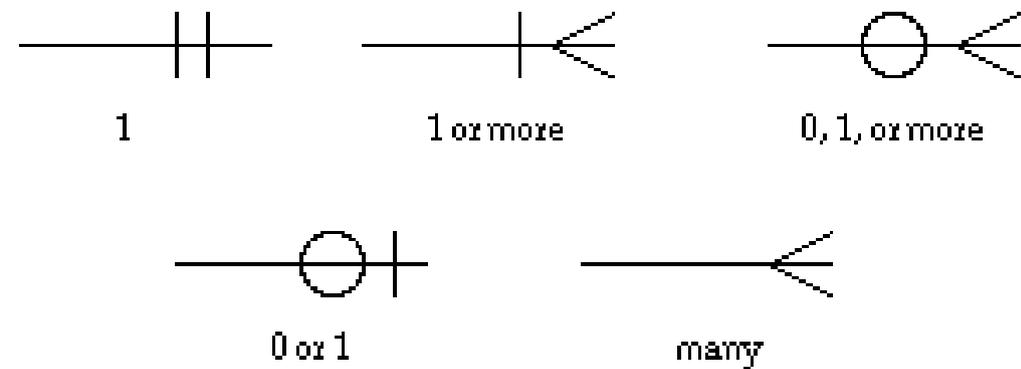
One to One Relationship



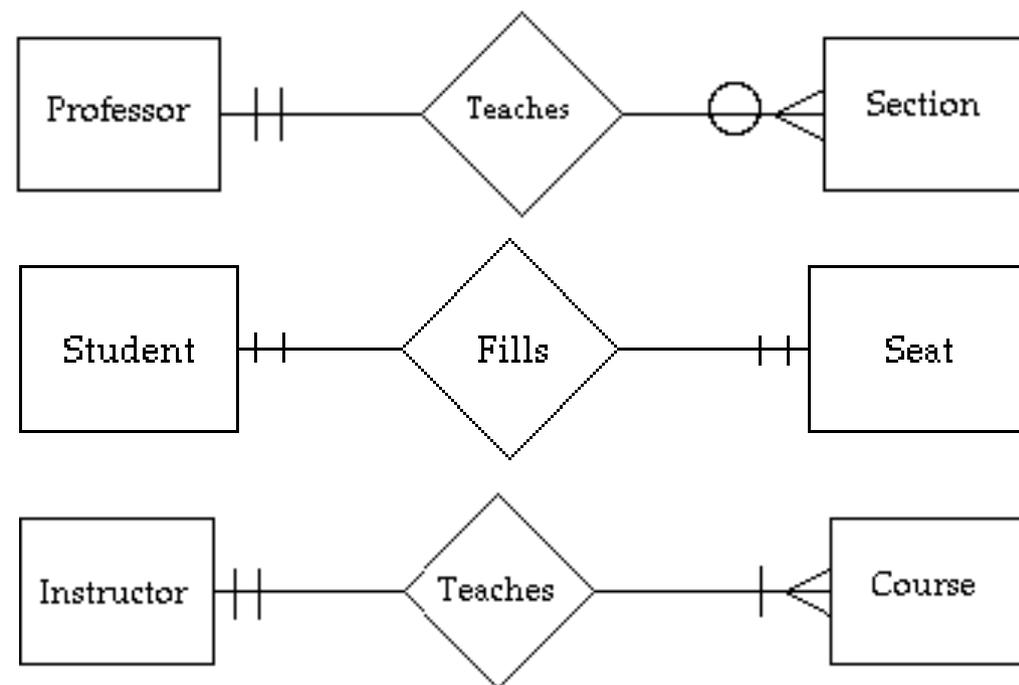
Many to One Relationship



Differenti notazioni



“Crow's foot”

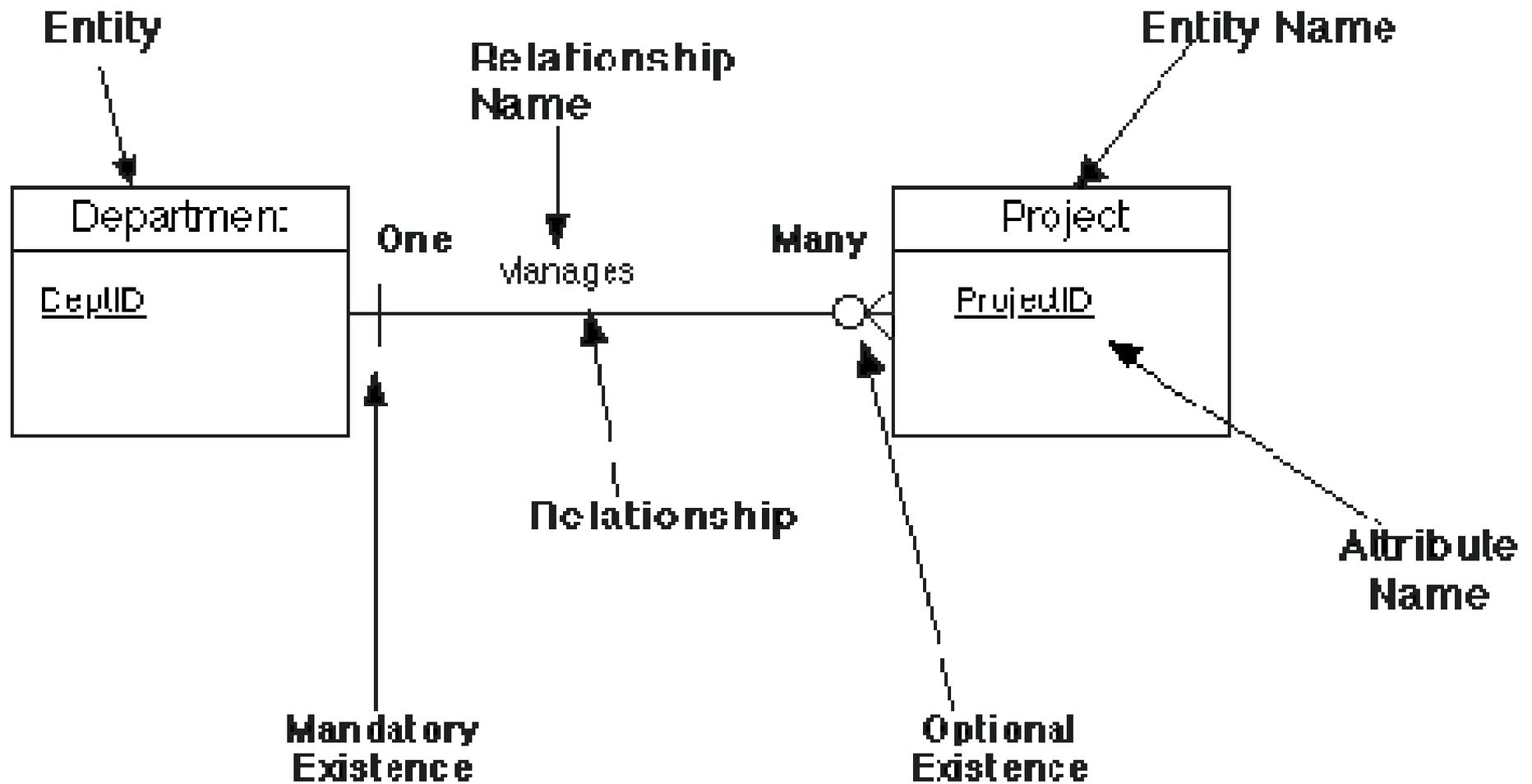


Un docente insegna zero, uno o più moduli

Uno studente occupa una e una sola sedia

Un istruttore insegna in uno o più corsi

Simbologia alternativa



In pratica...

