Progettazione Concettuale - Il Modello Entità - Relazione

Giuseppe Della Penna

Università degli Studi di L'Aquila
giuseppe.dellapenna@univaq.it
http://people.disim.univaq.it/dellapenna

> *Questo documento si basa sulle slide del corso di Laboratorio di Basi di Dati, riorganizzate per una migliore l'esperienza di lettura. Non è un libro di testo completo o un manuale tecnico, e deve essere utilizzato insieme a tutti gli altri materiali didattici del corso. Si prega di segnalare eventuali errori o omissioni all'autore.*

1. Le Fasi di Progettazione di un Database

La progettazione di un database passa per tre fasi a cascata:

- Concettuale
- Logica
- Fisica

A monte della progettazione concettuale c'è la raccolta e l'analisi dei requisiti, mentre a valle di quella fisica c'è un DBMS che implementa tutti i requisiti raccolti all'inizio.

2. Progettazione Concettuale

Nella progettazione concettuale si cerca di creare un *modello del dominio da rappresentare* nel database che ne *catturi tutti gli elementi più importanti in maniera chiara*.

Condizione importante di questa fase è il rimanere il più possibile aderenti alla realtà, in modo da poter verificare facilmente se il modello è corretto rispetto al dominio, anche discutendone col committente. In altre parole, il modello *non deve in alcun modo essere influenzato da scelte tecniche relative al DBMS* che verrà utilizzato (se noto a priori) e al suo modello di memorizzazione dei dati.

Si usano a questo scopo preferibilmente modelli grafici, più semplici da visualizzare.

3. I Diagrammi Entità-Relazione

I diagrammi Entità-Relazione (*ER*) sono uno degli strumenti più utilizzati per la progettazione concettuale di una base di dati. Si tratta di un formalismo grafico che permette di rappresentare gli *elementi* facenti parte del dominio che si vuole modellare nel database e le loro *interconnessioni*.

Naturalmente ER non è l'unico formalismo disponibile a questo scopo: ad esempio, molti preferiscono usare i *class diagram* di UML a questo scopo. Inoltre, esistono varie sintassi per gli stessi diagrammi ER: noi ci focalizzeremo sulla sintassi "classica", ma vedremo negli esempi anche sintassi alternative utili per semplificare alcuni aspetti dei diagrammi.

I diagrammi ER definiscono i seguenti elementi

- Entità
- Attributi
- Relazioni

Eventuali dettagli del dominio non esprimibili con questi elementi devo essere annotati a margine del diagramma, per poter essere poi incorporati nelle successive fasi di modellazione.

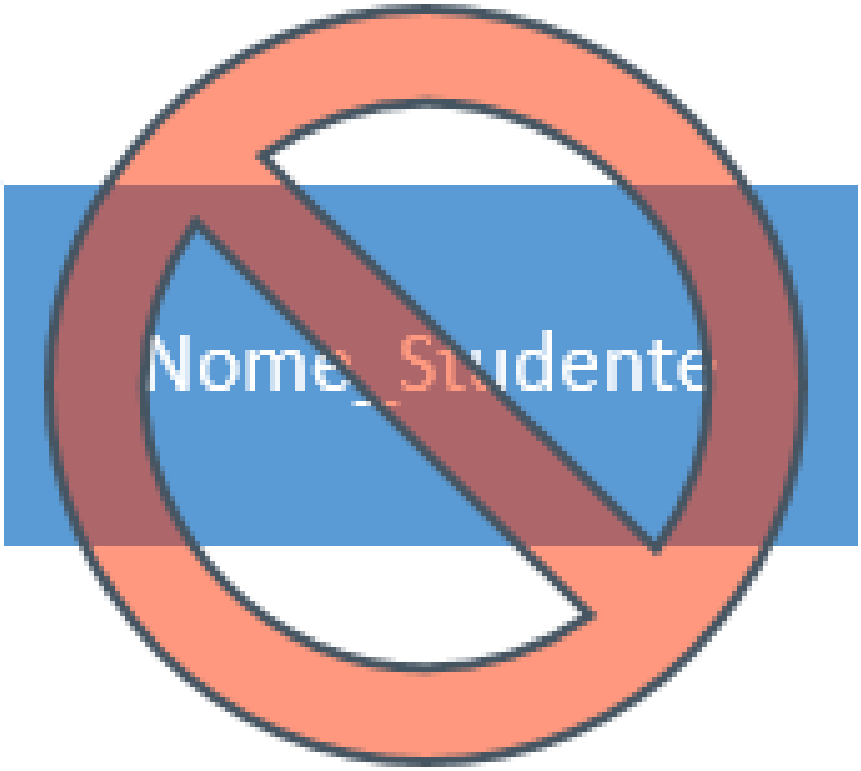
3.1. Entità

Ogni entità rappresenta una **classe di oggetti nel dominio** dell'applicazione. Ad esempio, l'entità *studente* conterrà tante istanze quanti sono gli studenti da rappresentare nella base di dati.

Le entità sono sempre *oggetti autonomi*, non facenti parte (aggregati) ad altri elementi.

Le entità si rappresentano come rettangoli con bordo continuo contenenti il nome dell'entità stessa, generalmente un sostantivo.

> Un'entità rappresenta una classe di oggetti distinti ed autonomi all'interno del dominio da rappresentare



> Il nome (dello studente) non è un'entità, perché è parte dello studente e non ha senso da solo

3.2. Attributi

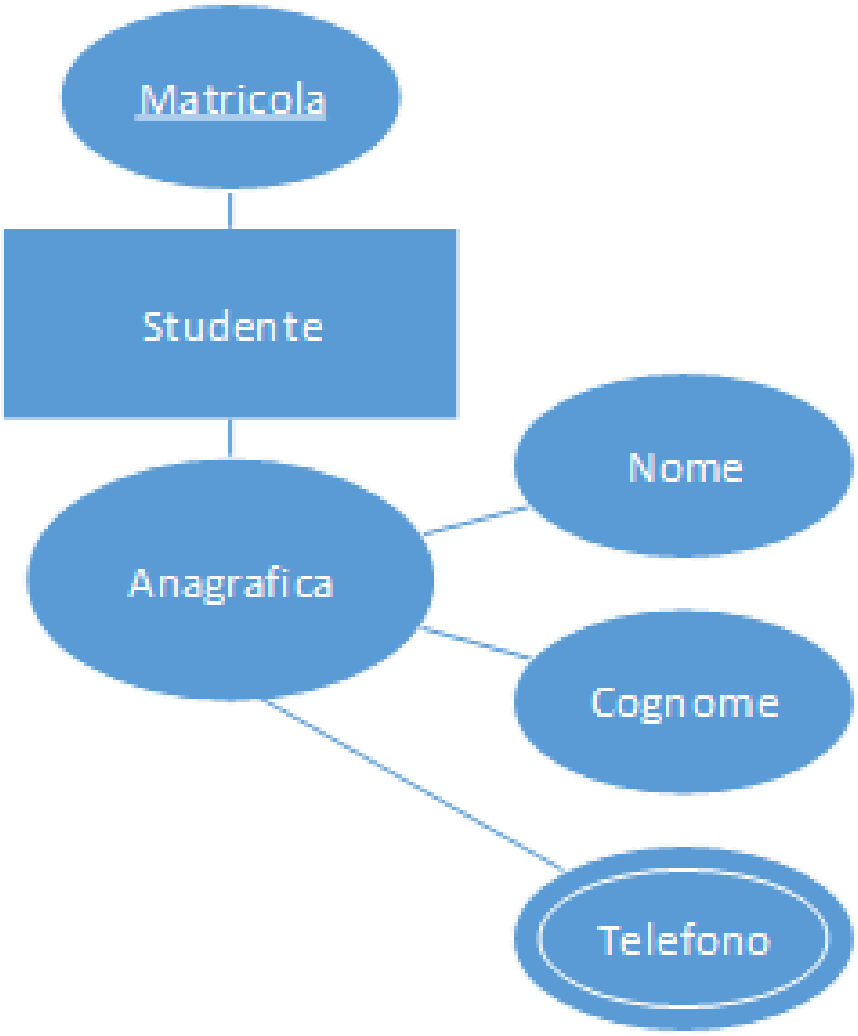
Ogni entità può avere un numero arbitrario di attributi che **ne definiscono il contenuto informativo**.

Ogni istanza di entità avrà, in generale, un insieme di valori differenti per i suoi attributi.

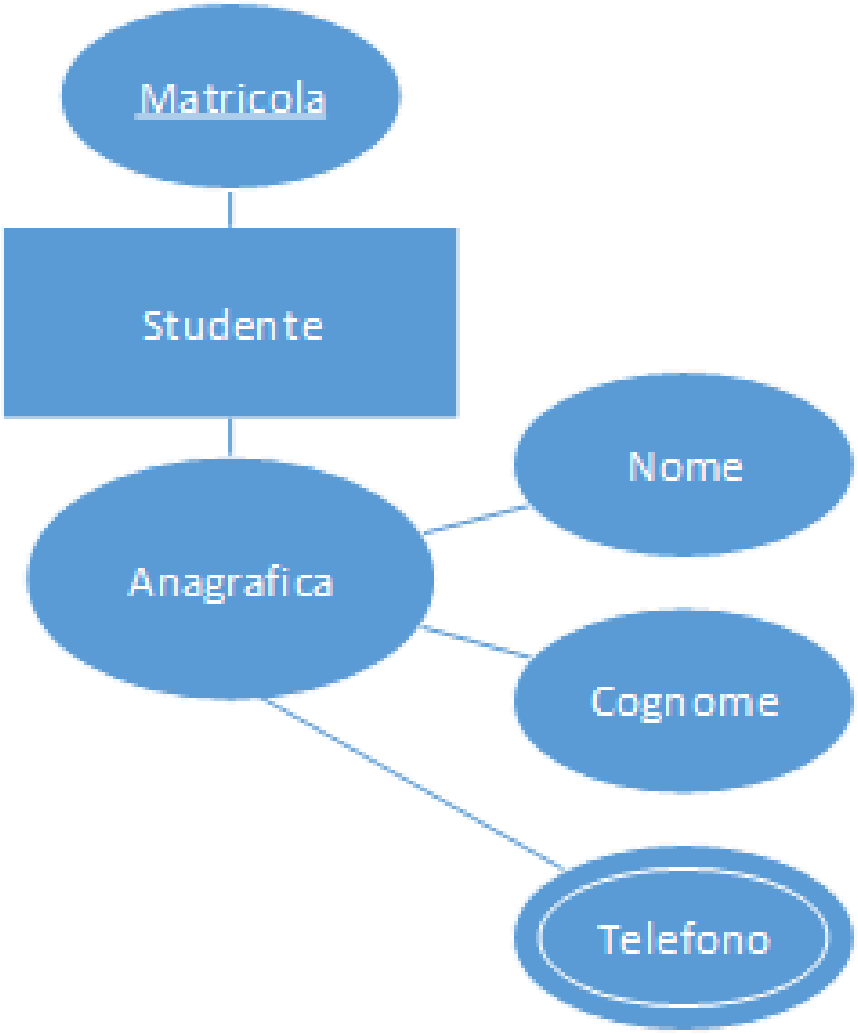
Gli attributi possono essere anche

- composti, cioè composti da sotto-attributi,
- multi valore, cioè contenere più di un valore.

Gli attributi si legano all'entità di appartenenza tramite il simbolo grafico illustrato.



> Un attributo rappresenta una proprietà di un'entità, ovvero un valore che caratterizza l'entità



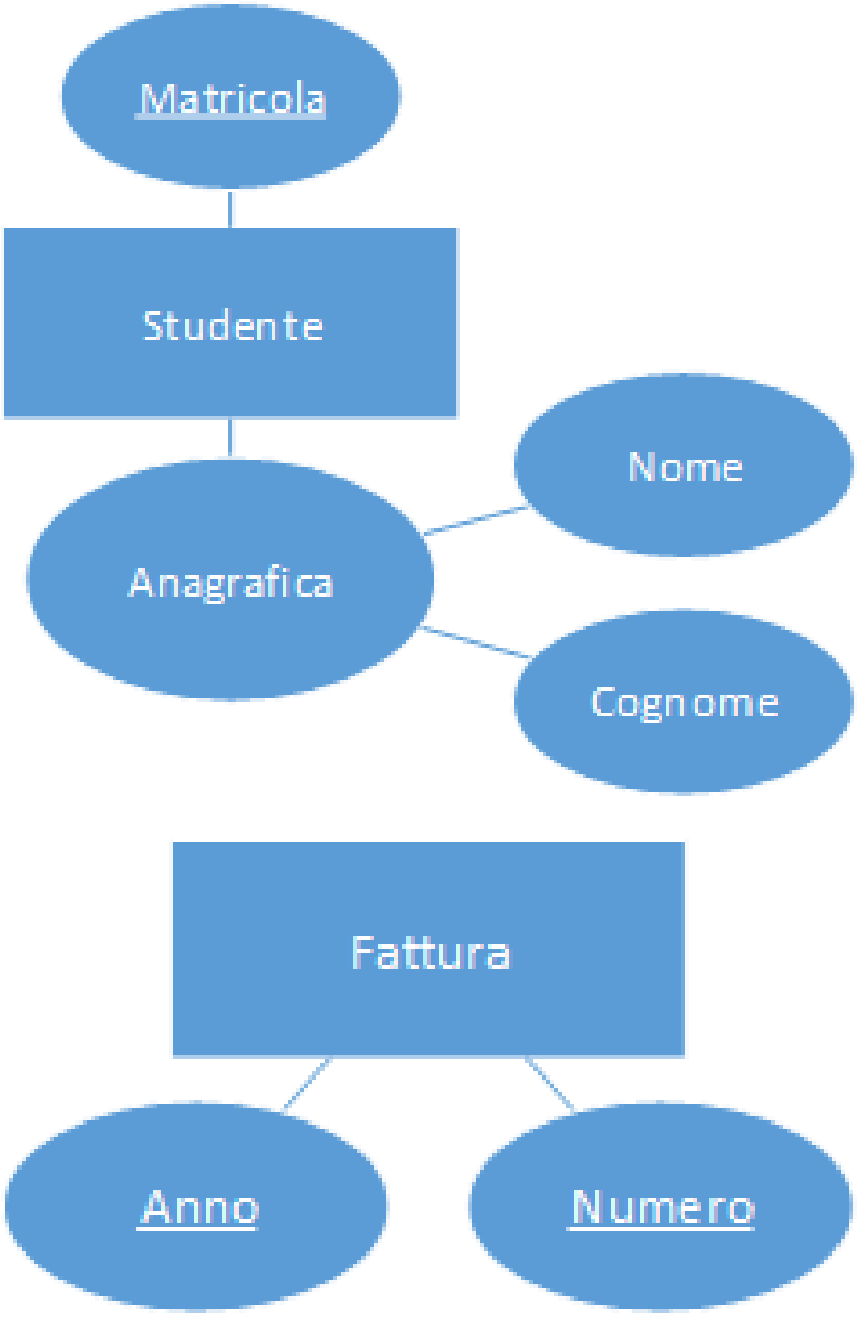
> Gli attributi composti sono spesso usati per raggruppare attributi correlati. Un attributo può anche ammettere più valori (più numeri di telefono per uno studente, nell'esempio)

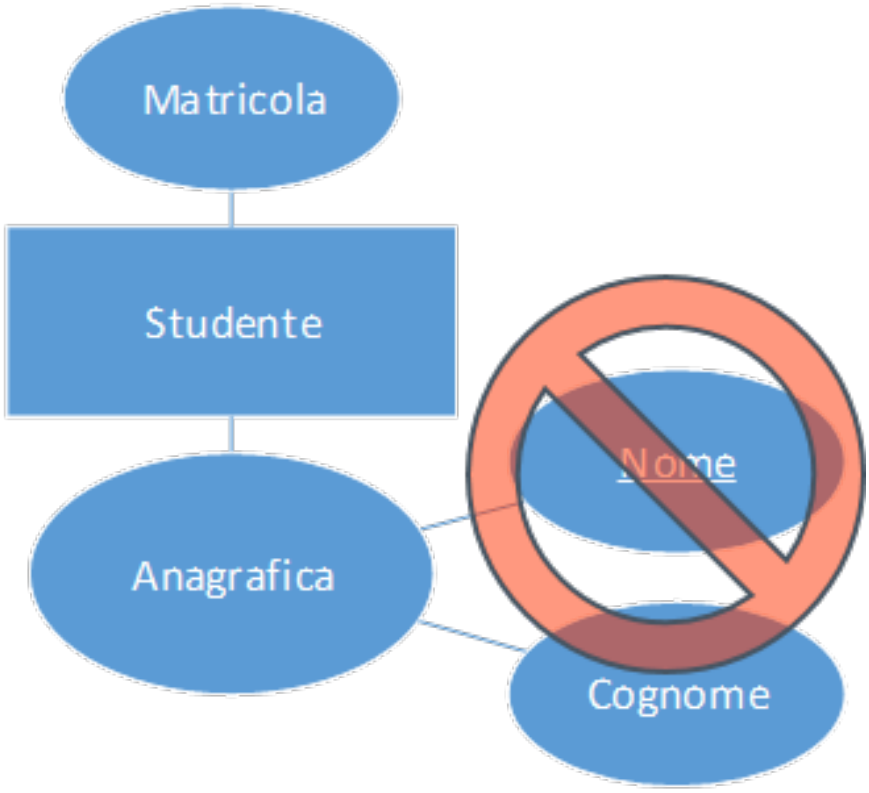
3.3. Chiavi

Per **distinguere tra loro le istanze** di un'entità (ad es., distinguere gli studenti), è necessario definire un attributo particolare, o un insieme di attributi, che siano diversi in ogni istanza. L'attributo o gli attributi così individuati costituiranno la **chiave primaria** dell'entità.

Ogni entità deve avere la sua chiave primaria. Tuttavia, alcune entità potrebbero non contenere attributi sufficienti per costituire una chiave (*entità deboli*).

Gli attributi chiave si indicano graficamente sottolineandone il nome.


Matricola
Studente
Nome
Anagrafica
Cognome
Fattura
Anno
Numero



> L'attributo chiave deve avere un valore diverso in tutte le istanze dell'entità!

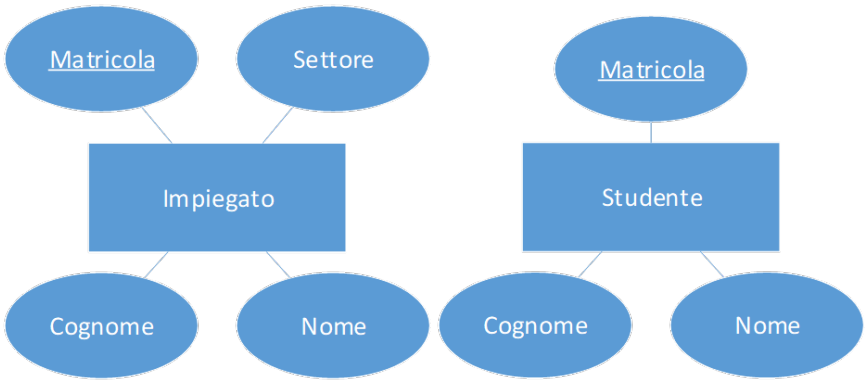
3.4. Generalizzazioni

A volte può essere utile decomporre concettualmente un'entità in una gerarchia di entità con diverso livello di dettaglio. Questo permette di caratterizzare meglio i singoli componenti della gerarchia.
Si parla in questo caso di **gerarchia di generalizzazione** tra le entità. La generalizzazione può essere **parziale** o **totale**.

Le generalizzazioni sono un'estensione dei diagrammi ER standard, sebbene molto diffusa, e si rappresenta graficamente usando una notazione presa in prestito da UML.

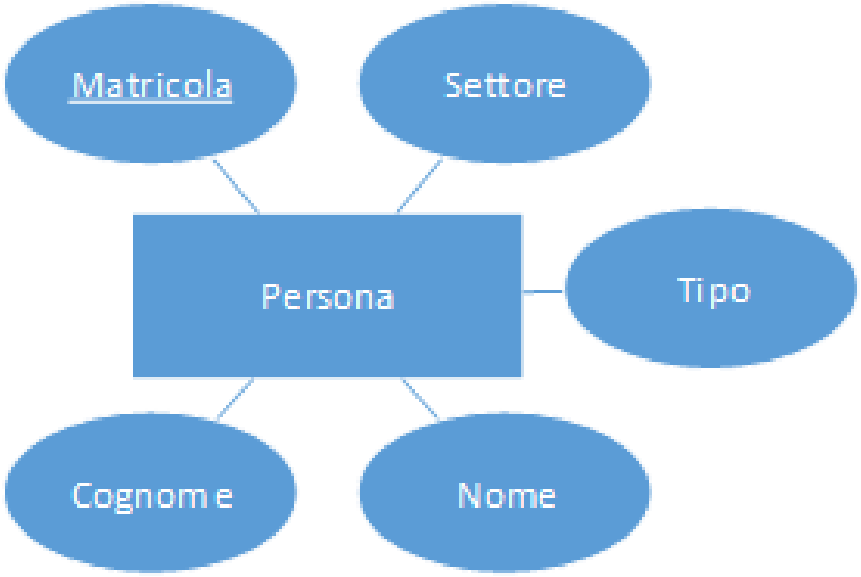
Supponiamo di avere nel nostro DB i dati relativi a tutti gli studenti e gli impiegati dell'Università. Studenti e impiegati avranno attributi differenti, ma anche un gruppo di attributi comuni, ad esempio quelli anagrafici.

Qual è il modo migliore di rappresentare queste due entità?



> Questa soluzione duplica gli attributi comuni a impiegato e studente. Graficamente, aumenta la

> complessità del diagramma.

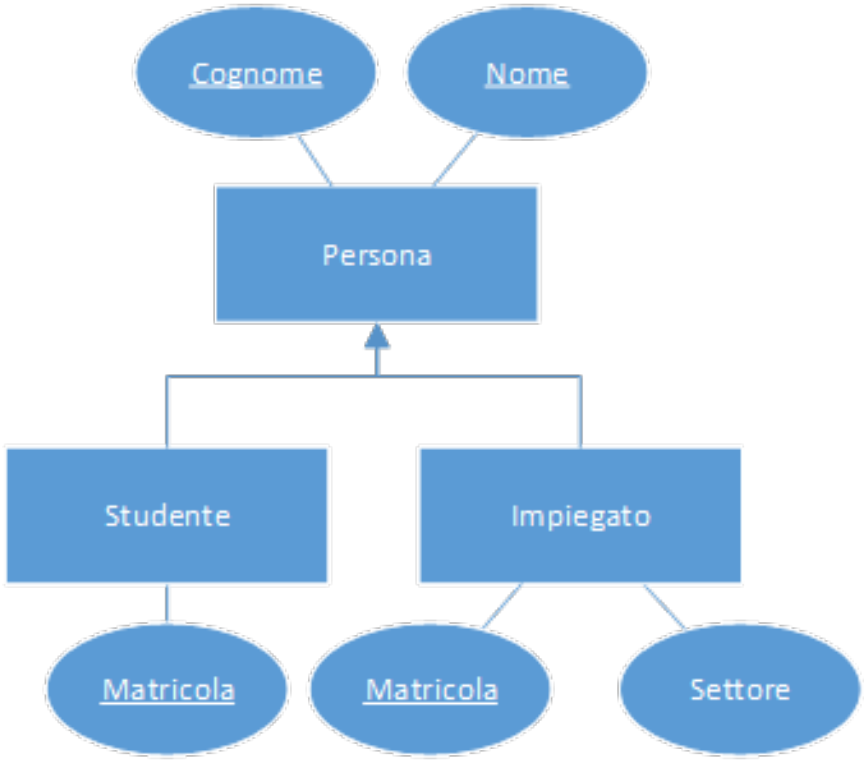


> Qui abbiamo astratto il concetto generale di persona ed abbiamo assegnato a essa tutti gli attributi di impiegato e studente. Ma è corretto?

Totali

Partendo dall'esempio precedente, l'uso della generalizzazione permette di rendere più leggibile il diagramma e mostra con chiarezza il rapporto esistente tra le entità persona, impiegato e studente.

Poiché per noi le persone sono solo impiegati o studenti, la generalizzazione si dice *totale* (freccia *piena*).

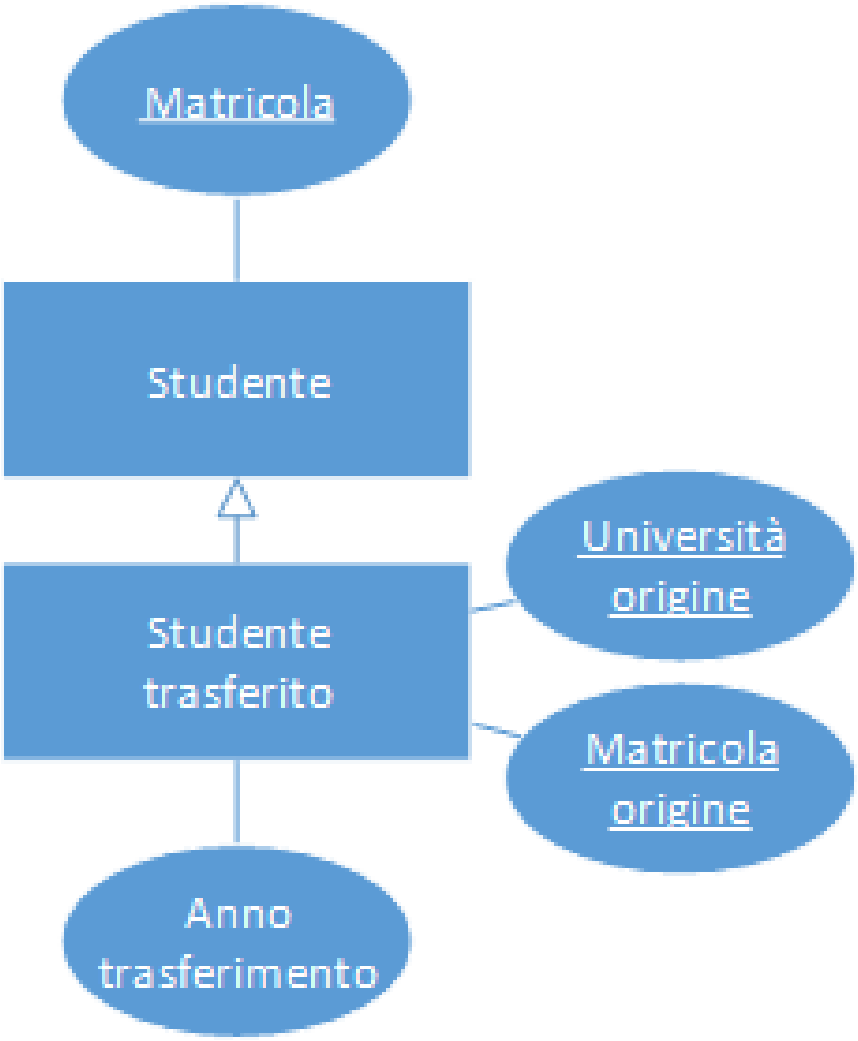


> I due numeri di matricola hanno semantica diversa, quindi è pericoloso "fonderli" in persona.

> Persona ha la sua chiave, come pure le sue sotto-entità. Notare che la coppia (nome,cognome) potrebbe non essere una chiave univoca.

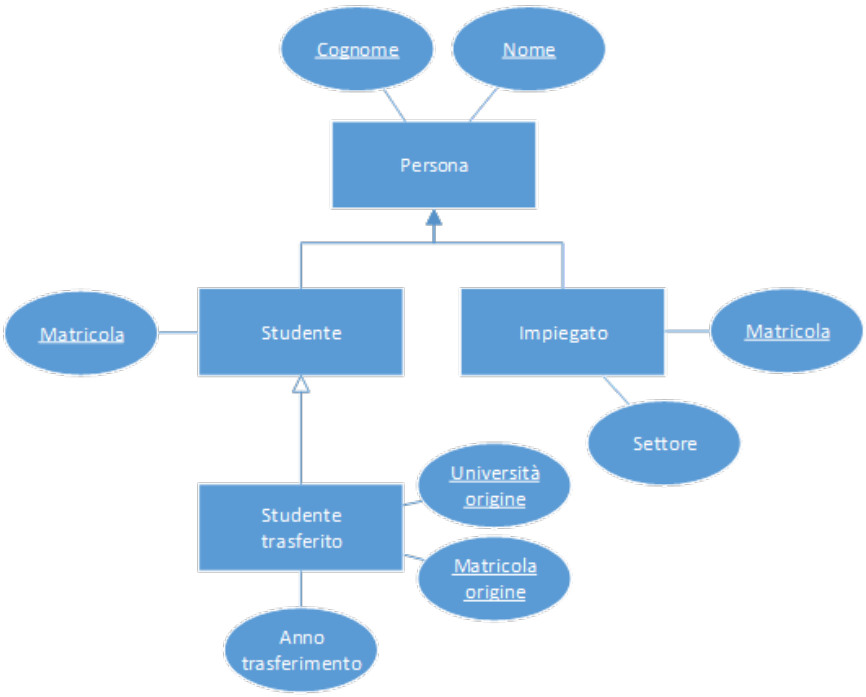
Parziali

In questo caso, invece, la generalizzazione è *parziale* (freccia *vuota*): uno studente *può essere* uno studente trasferito da altra Università. In altre parole, le istanze di studente non devono necessariamente appartenere anche a una delle entità più "dettagliate" (qui ne abbiamo una sola).



Su più livelli

Ovviamente le generalizzazioni possono anche comporsi tra loro...



3.5. Relazioni

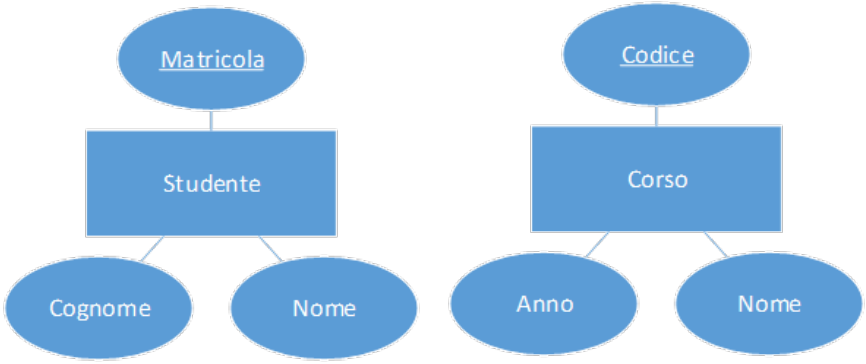
Finora abbiamo mappato sul diagramma ER:

- gli oggetti (entità) del nostro dominio
- i dati (attributi) che li caratterizzano

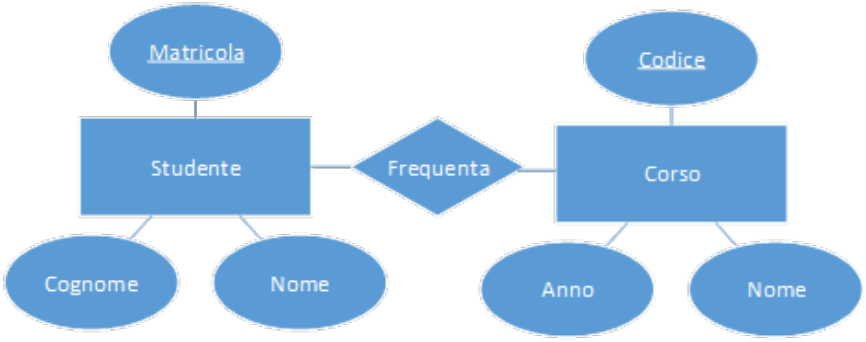
Tuttavia, un altro aspetto importante da esprimere è il modo in cui questi oggetti sono in relazione tra loro. Ad esempio, se rappresentiamo le entità *studente* e *corso*, vorremmo poter esprimere il fatto che gli studenti frequentano i corsi.

Le relazioni tra entità si rappresentano graficamente con un rombo (la relazione vera e propria) che connette le due entità correlate. Il nome della relazione è solitamente un verbo, ma in alcuni casi può essere anche un sostantivo.

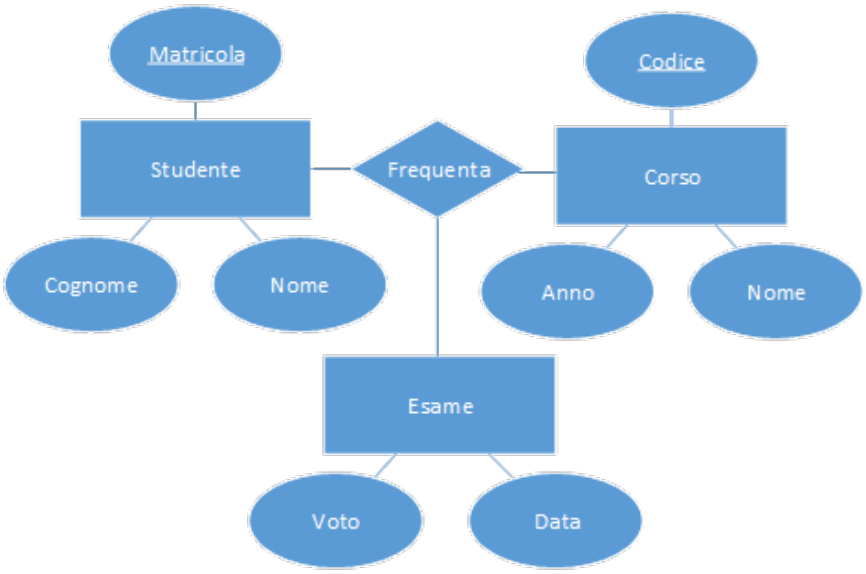
La relazione qui rappresentata è *binaria*, ed è la più comune.



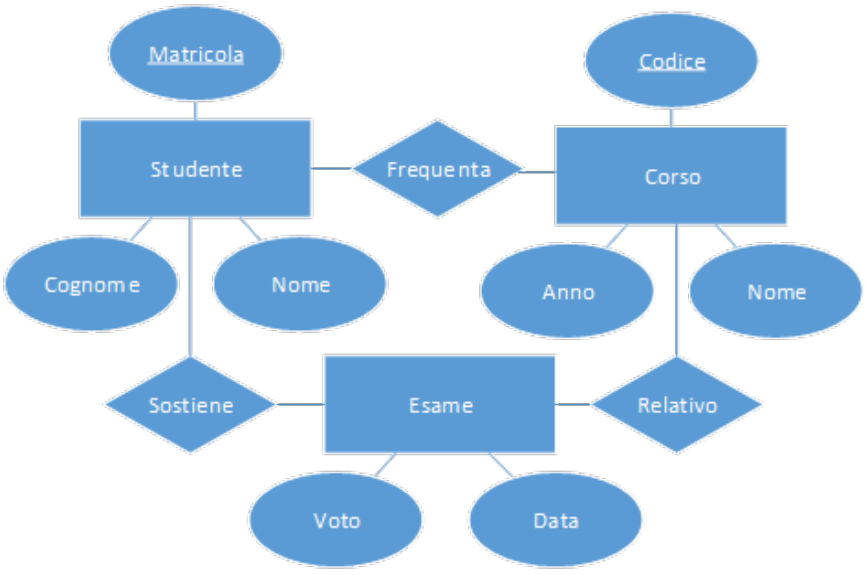
> ...gli studenti frequentano i corsi...



Ternarie (e non binarie)



> Questa relazione ternaria ("lo studente frequenta un corso per cui sostiene un esame") è corretta, ma difficile da interpretare



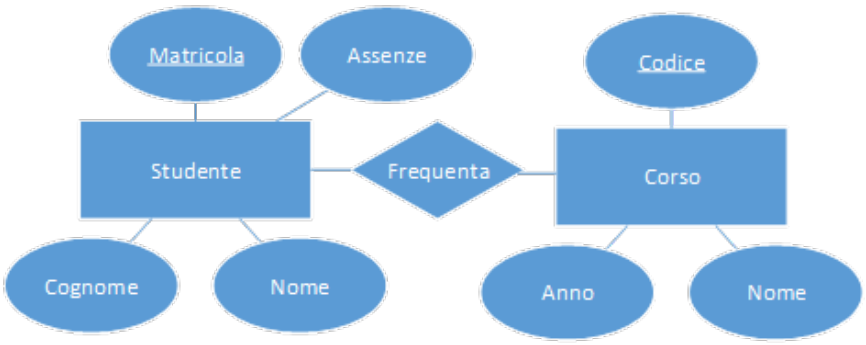
> "Esplodendo" la relazione ternaria, si ottiene un diagramma molto più leggibile

Attributi

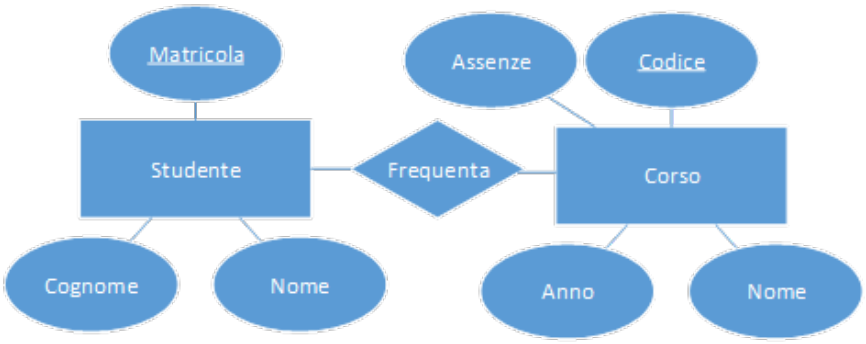
Una relazione può avere anche degli attributi, indicati graficamente con lo stesso simbolo usato per gli attributi delle entità.

Il significato di un attributo su una relazione è quello di una caratteristica che non riguarda le entità coinvolte, ma il fatto stesso di essere in relazione tra loro.

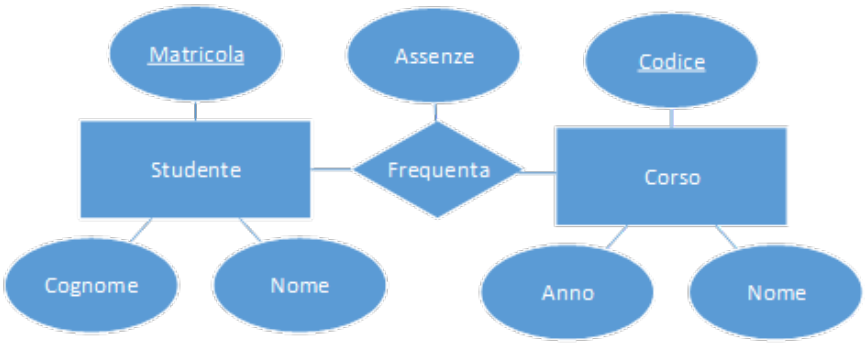
Ad esempio: come inseriamo nel nostro schema relazionale studenti-corsi il numero di assenze a lezione?



> Se inseriamo le assenze sullo studente, queste varranno per tutti corsi che frequenta!



> Se inseriamo le assenze sul corso, queste varranno per tutti gli studenti che lo frequentano!



> Così le assenze sono quelle di uno studente per un certo corso.

Cardinalità

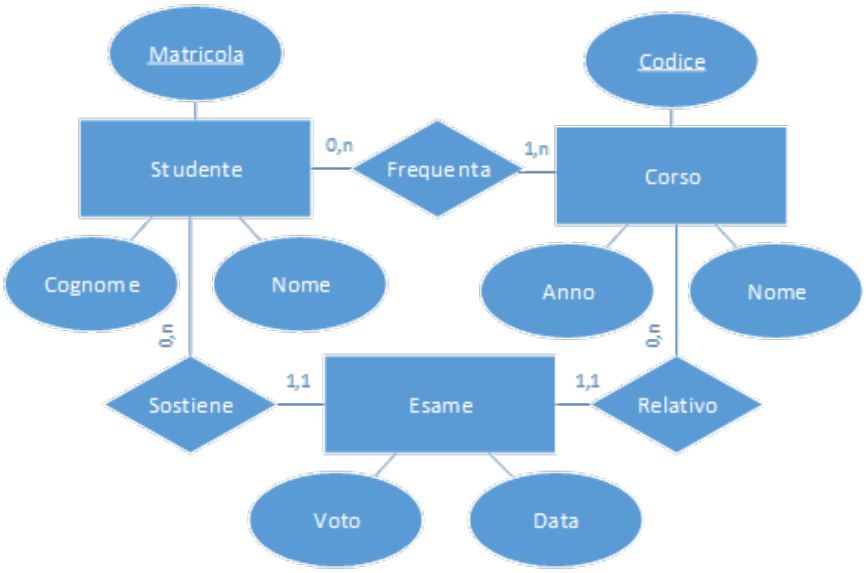
Le relazioni hanno sempre dei *vincoli di cardinalità sulla partecipazione delle entità*, che indicano in che numero le entità coinvolte possono parteciparvi.

Il vincolo di cardinalità, nella sua forma più generale, è espresso come una coppia (*min*, *max*) posta sulla linea che unisce la relazione all'entità.
Come cardinalità massima è possibile indicare l'infinito, rappresentato spesso con la lettera "n".

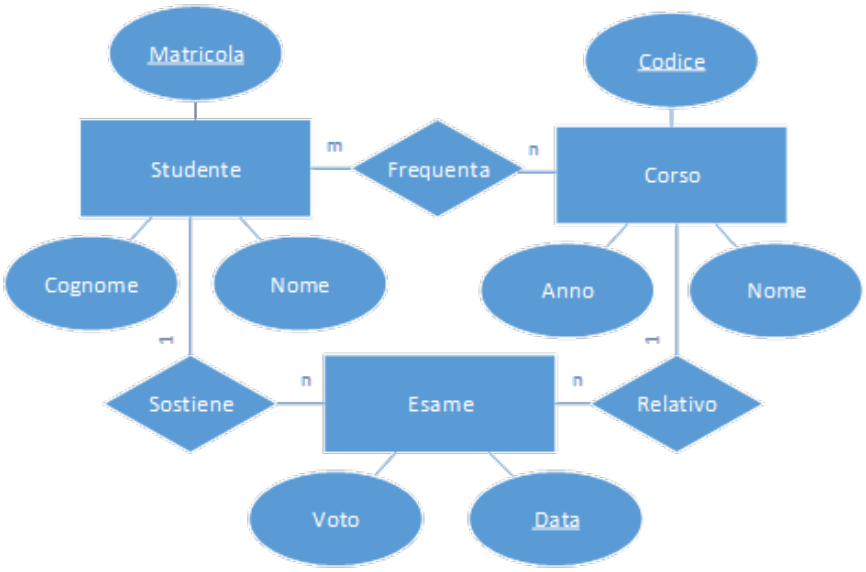
Esempi tipici di cardinalità sono `(1,1)` (*esattamente uno*), `(1,n)` (*almeno uno*), `(0,1)`

("*opzionalmente*"), `(0,n)` (*un numero qualsiasi*).

Spesso troviamo usati anche dei più semplici *rapporti di cardinalità* con la sintassi `1:1` (*uno a uno*), `1:n` (*uno a molti*) o `m:n` (*molti a molti*): in questi casi, ciascun numero rappresenta il vincolo di partecipazione massimo per un lato della relazione, mentre quello minimo è implicitamente sempre 1. Questa notazione è leggermente meno informativa rispetto a quella dei vincoli, quindi preferiremo sempre la prima nei nostri esempi.



> Uno studente può frequentare un numero qualsiasi di corsi e sostenere un numero qualsiasi di esami; un corso è frequentato da almeno uno studente ed ha associati un numero qualsiasi di esami. Un esame è relativo ad esattamente un corso ed è sostenuto da esattamente uno studente



> Questa versione, con rapporti di cardinalità, esprime sostanzialmente le stesse informazioni, ma con una precisione minore.

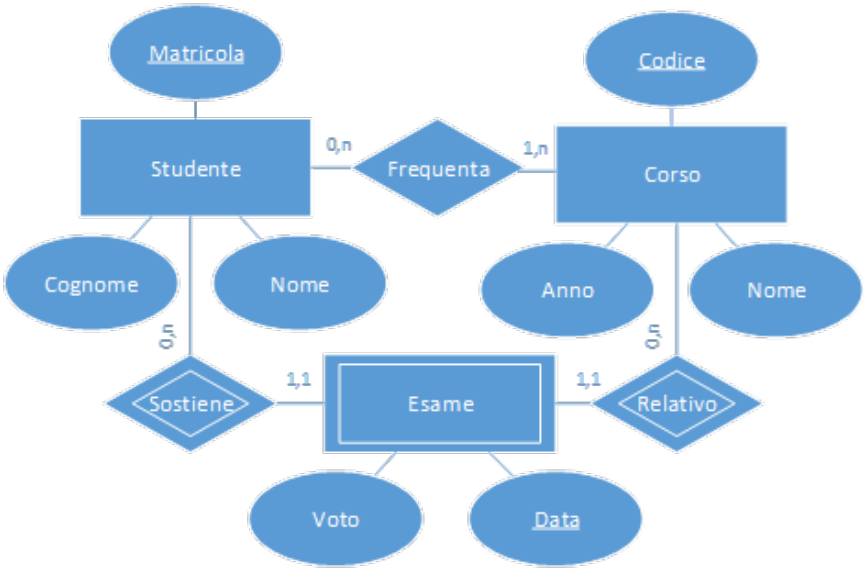
3.6. Entità Deboli

Un'entità debole non ha attributi sufficienti a formare una chiave, ma *è identificata dal fatto di essere*

in relazione con altre entità, quindi deve includere nella chiave una o più delle sue relazioni.

Nell'esempio, un esame è univocamente specificato dall'insieme della sua data, del corso di appartenenza e dello studente che lo ha sostenuto.

Per includere una relazione nella chiave, la sua cardinalità deve essere `(1,1)`.



Title: Il Modello Entità - Relazione
Author: Giuseppe Della Penna, *University of L'Aquila*
Version: 20240226
