

SQL



Renata Carriero

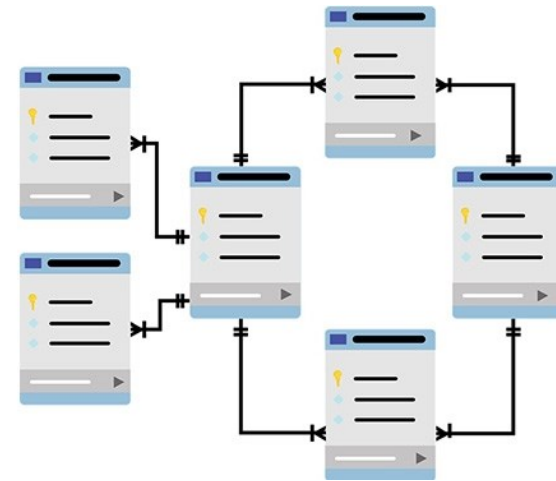
Renata.Carriero@icubed.it



Cos'è un DataBase (DB)?

È un archivio di dati strutturato in modo da razionalizzare la gestione e l'aggiornamento delle informazioni e da permettere lo svolgimento di ricerche complesse.

Database relazionale → archivio in cui le informazioni sono organizzate in tabelle legate (o meglio «in relazione»), che consentono ricerche e aggiornamenti incrociati.



Database (Base di Dati)

Database → struttura di **dati** organizzati secondo un modello.

Dato → informazione.

Un DB ha le seguenti caratteristiche:

- Usato per rappresentare/raccogliere dati d'interesse
- Condiviso tra diverse applicazioni software e più utenti
- Ogni dato è rappresentato solo una volta nella collezione

DBMS

Per accedere a uno o più database di usa il DBMS:

Database **M**anagement **S**ystem

Un set di software che permettono l'accesso, l'aggiornamento e eventuale recupero di dati.



<https://docs.microsoft.com/it-it/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15>

Modelli di database

1. Relational

Struttura tramite tabelle composte da campi e record.

Relazioni: interne alla tabella e tra diverse tabelle

-> Gestiti da RDBMS > Relational Database Management System

2. Object Oriented

Struttura tramite oggetti, usata soprattutto in ambito documentale (Json, XML..)

-> Gestiti da ODBMS / OODBMS

3. Object-Relational

Struttura mista

Database Relational Model

Un archivio è solitamente composto da dati **non omogenei** (ad esempio pensando ad un DB che raccoglie le info di una scuola, i dati –non omogenei- potrebbero essere Libri, Alunni, Professori, Voti, Assenze, ...).

- Ogni gruppo di dati **omogenei** viene registrato all'interno di uno stesso contenitore/struttura detta **tabella**.
- Il singolo elemento inserito in una tabella è detto **record**.
- Le proprietà che caratterizzano ogni singolo elemento della stessa tabella vengono definite **attributi/campi**.

In una rappresentazione tabellare:

- le **righe** rappresentano i **record**
- le **colonne** rappresentano i **campi**.

L'insieme delle descrizione dei campi (nome, dimensione, tipo ...) prende il nome di struttura della tabella.

TABELLA

ID_Libro	TitoloLibro	Categoria
2	Storia di BRESCIA	Storia
3	La Lombardia	Geografia
4	Secchiare info in allegria	Informatica

STRUTTURA

Nome campo	Tipo dati
ID_Libro	Numerico
TitoloLibro	Testo
Categoria	Testo

Database Relational Model

Altro esempio. Un archivio di dati anagrafici contiene le informazioni sulle persone, con Cognome, Nome, Data di nascita, Città, Telefono.

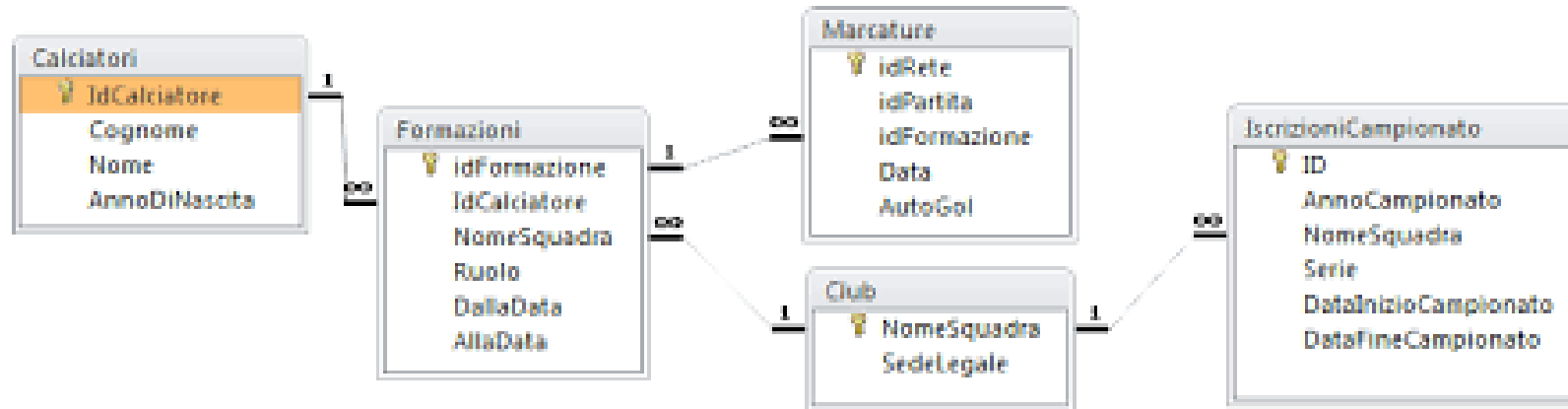
Cognome	Nome	Data di nascita	Città	Telefono
---------	------	-----------------	-------	----------

Ognuno di questi dati si chiama **campo** e l'insieme dei campi di una stessa riga forma il **record**, che si riferisce ad una singola persona. L'archivio è quindi un insieme di record.

Database Relational Model

Un DB composto da **diverse tabelle in relazione** tra loro si dice **Relazionale**.

Le relazioni tra le tabelle, permettono di manipolare i dati più facilmente e soprattutto evitano la ridondanza dei dati, ovvero la duplicazione delle informazioni che è inevitabile quando si opera con singole tabelle indipendenti.



Progettazione del Database

La costruzione di una base di dati deve essere preceduta da una fase di **progettazione** per definire le caratteristiche fondamentali della realtà che si vuole automatizzare e gli obiettivi che si vogliono raggiungere.

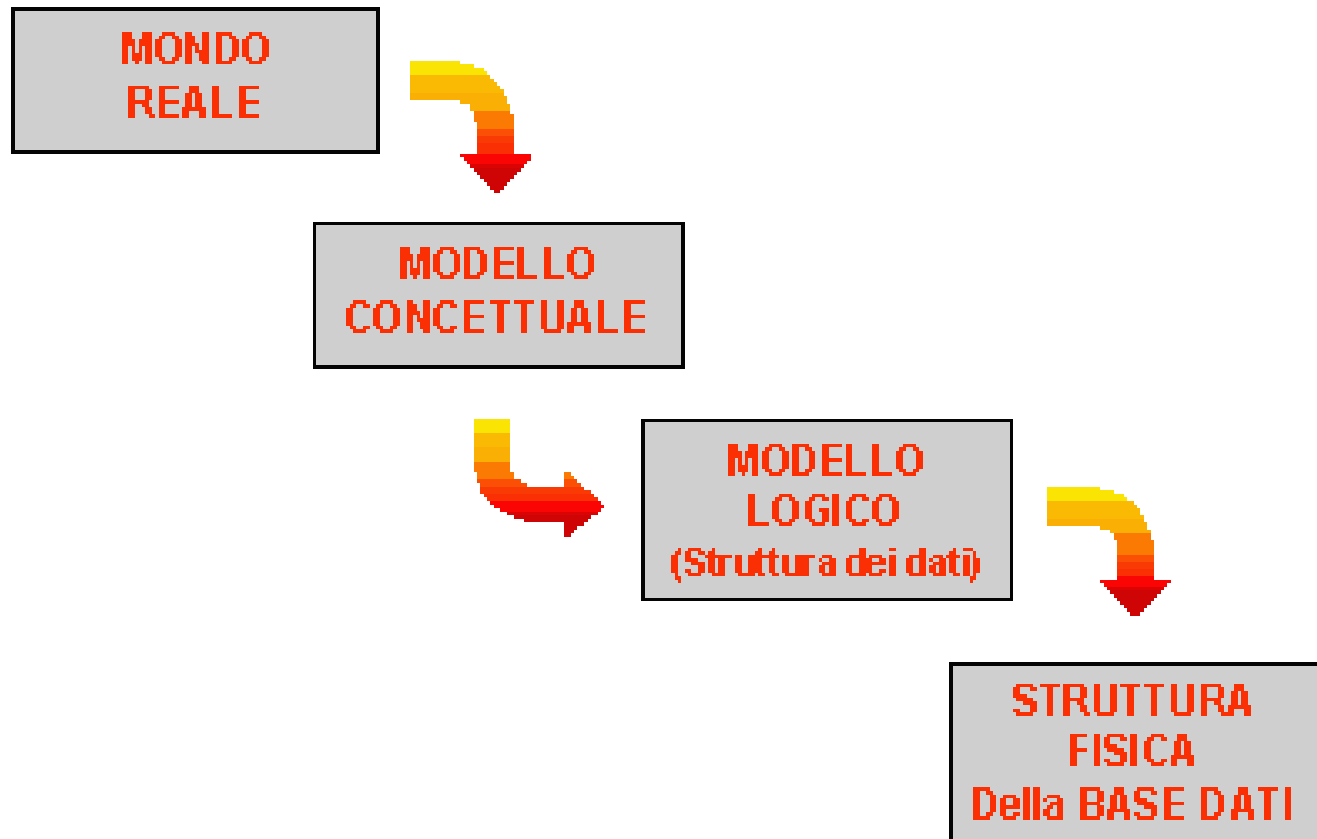
Viene dunque adottato il modello dei dati con cui procedere alla rappresentazione.

Il **modello** è un insieme di concetti che rappresenta formalmente la realtà da rappresentare attraverso una codifica interpretabile in modo automatizzato da un elaboratore.

Il suo scopo è quello di rendere efficiente la fase progettuale, per cui i modelli utilizzati dovranno essere standardizzati e condivisibili (uniformati) e saranno accompagnati da un protocollo di progettazione che ottimizzi la creazione e l'aggiornamento del progetto.

Modelli per la fase di Progettazione

- **Concettuale**
- **Logico** (relazionale)
- **Fisico**



Modello Concettuale

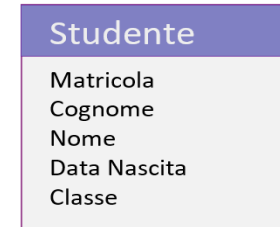
Osservando una realtà possiamo coglierne le **entità** utili per rappresentarne la gestione automatizzata.

Ciò si ottiene individuando gli elementi che la caratterizzano: ad esempio in una scuola gli studenti, i docenti, le materie, le prove degli studenti, ecc.

Ciascun'entità possiede degli **attributi**, ovvero le proprietà che la identificano e la caratterizzano.

Modello Concettuale

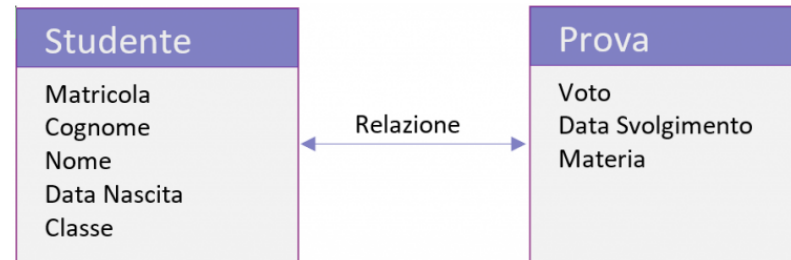
Per esempio, le proprietà (o attributi) dell'entità **Studente** sono la matricola, il cognome, il nome, la data di nascita, la classe.



L'entità **Prova** ha come attributi il voto, la data di svolgimento, la materia a cui si riferisce.



Tra le entità si stabiliscono inoltre delle **relazioni** o associazioni, cioè dei collegamenti. Per conoscere a quale studente si riferiscono le prove fissiamo un collegamento tra l'entità Prova e l'entità Studente.



Definendo le entità, gli attributi e le relazioni si costruisce il modello concettuale della realtà osservata.

Modello Logico

Dal modello precedente si passa al modello logico (o relazionale).

- Ogni **entità** del modello concettuale diventa una **tabella**.
- Gli **attributi** diventano i titoli delle colonne e andranno a formare il tracciato record, cioè l'insieme di tutti gli identificatori dei **campi della tabella**.

Studenti

Matricola	Cognome	Nome	DataNascita	Classe	Tel
0001	Rossi	Laura	15/04/2002	2A	320.5564332
0002	Verdi	Maria	12/08/2001	2A	333.9887001
0003	Bianco	Giorgio	06/01/2002	2A	349.5435672
0004	Neri	Luca	21/12/2001	2A	348.1267887

Prove

ID	Voto	Data	Materia
V001	10	24/03/2018	ITALIANO
V002	9	23/07/2017	MATEMATICA
V003	7	16/01/2018	FRANCESE
V004	9	20/11/2017	INGLESE

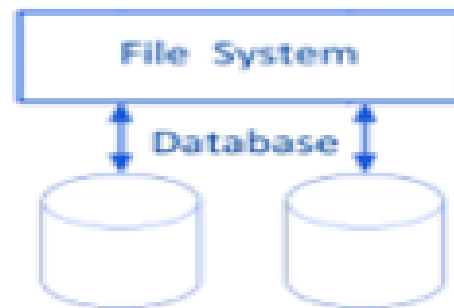
Le righe (o record) contengono i dati che si riferiscono a uno specifico esemplare (o istanza) dell'entità.

Ad esempio, la prima riga della tabella Studenti rappresenta lo studente Laura Rossi.

Modello Fisico

Infine, il modello fisico individua il supporto fisico di memorizzazione da utilizzare per l'archiviazione dei dati (cd-rom, hard-disk,...).

La progettazione fisica coincide con l'associazione della struttura logica ad una struttura fisica per la memorizzazione di massa.

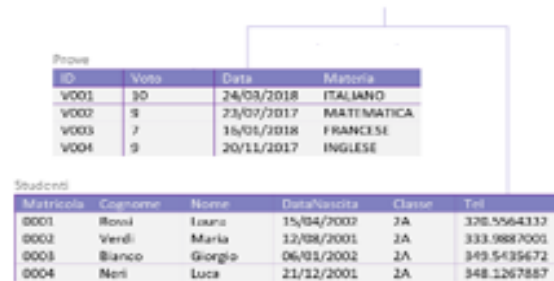


Progettazione del Database

SCHEMA CONCETTUALE



SCHEMA LOGICO



SCHEMA FISICO



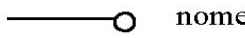


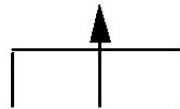


Modello Entità-Relazione

Il modello Entità-Relazione (E-R) è un **modello concettuale** di dati, e come tale fornisce una serie di strutture (costrutti), atte a descrivere la realtà in una maniera facile da comprendere

Rappresentazione concettuale della struttura dei dati

- Costrutti hanno una rappresentazione con diagramma
- Modello più leggibile e comprensibile

STRUTTURA DI CLASSIFICAZIONE	SIMBOLO
ENTITA'	
RELAZIONE	
ATTRIBUTO SEMPLICE	
ATTRIBUTO AGGREGATO	
SOTTOINSIEME	
GENERALIZZAZIONE	

● IL PALLINO PIENO INDICA UNA CARATTERISTICA O ATTRIBUTO CHE NON ACCETTA DUPLICATI

○ IL PALLINO VUOTO INDICA UNA CARATTERISTICA O ATTRIBUTO CHE ACCETTA DUPLICATI

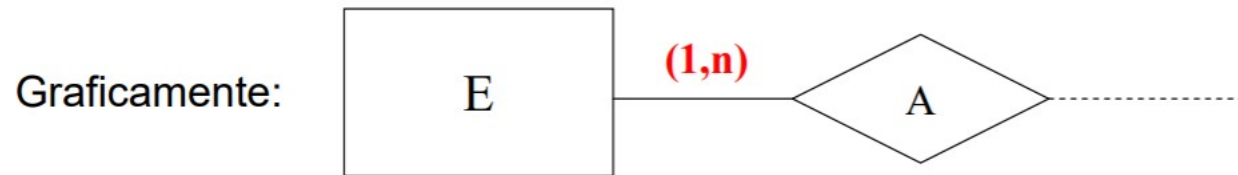


CARATTERISTICA O ATTRIBUTO CHE PUÒ ASSUMERE PIÙ VALORI, COME AD ESEMPIO I NUMERI DI TELEFONO

Cardinalità di Relazione

Per ogni entità partecipante ad una relazione viene specificata una **cardinalità di relazione**.

Essa è una **coppia di numeri naturali** che specifica il **numero minimo e massimo** di istanze di relazione → (min-card,max-card)



Ad esempio, se i vincoli di cardinalità per un'entità E relativamente a un'associazione A sono (1,n) questo significa:

- ogni istanza di E partecipa almeno ad una istanza di A → min-card = 1
- ogni istanza di E può partecipare a più istanze di A → max-card = n

N.B. Con la costante n si indica un numero generico maggiore di 1 quando la cardinalità non è nota con precisione.

Cardinalità di Relazione



Cardinalità di relazione (Persone,Proprietà) $\rightarrow (0,n)$

min-card = 0: esistono persone che non posseggono alcuna automobile

max-card = n: ogni persona può essere proprietaria di molte (n) automobili

Cardinalità di relazione (Automobili,Proprietà) $\rightarrow (0,1)$

min-card = 0: esistono automobili non possedute da alcuna persona

max-card = 1: ogni automobile può avere al più un proprietario

Tipi di associazione: terminologia

Nel caso di un'associazione binaria A tra due entità $E1$ ed $E2$ (non necessariamente distinte), si dice che:

- A è **uno a uno** se le cardinalità massime di entrambe le entità rispetto ad A sono 1
- A è **uno a molti** se $\max\text{-card}(E1,A) = 1$ e $\max\text{-card}(E2,A) = n$, o viceversa
- A è **molti a molti** se $\max\text{-card}(E1,A) = n$ e $\max\text{-card}(E2,A) = n$

Si dice inoltre che:

La partecipazione di $E1$ in A è *opzionale* se $\min\text{-card}(E1,A) = 0$

La partecipazione di $E1$ in A è *obbligatoria* (o totale) se $\min\text{-card}(E1,A) = 1$

Relazione uno a uno

La relazione **uno a uno** è detta anche **biunivoca** perché ad ogni elemento della prima entità, fa corrispondere un solo, specifico, elemento dell'entità collegata.

Esempi:

- A ciascun marito, corrisponde una sola e specifica moglie.
- A ciascuna persona corrisponde una sola carta di identità.

NB. Ha senso parlare di relazione 1:1 solo se entrambe le entità collegate sono entità a tutti gli effetti.

Altrimenti la relazione «non esiste» ma si traduce nell'inserire un attributo in più nell'entità di partenza.

Ad esempio: ad ogni persona corrisponde un solo Codice Fiscale. Il codice fiscale non è un'entità vera e propria quindi diventa un attributo dell'entità persona.

Relazione uno a molti

La relazione **uno a molti** fa corrispondere:

- a ciascun elemento della prima entità, **uno o più elementi** della seconda entità.
- ad ogni elemento della seconda entità, **un solo e specifico elemento** della prima entità.

Esempio ***Studente - Valutazione***:

- per ogni studente possiamo avere più valutazioni (voto di Storia di novembre; voto di Matematica di ottobre; voto di Italiano di gennaio; voto di Italiano di febbraio...),
- a ciascuna valutazione (personale), corrisponde il solo e specifico Studente che l'ha presa.

Relazione molti a molti

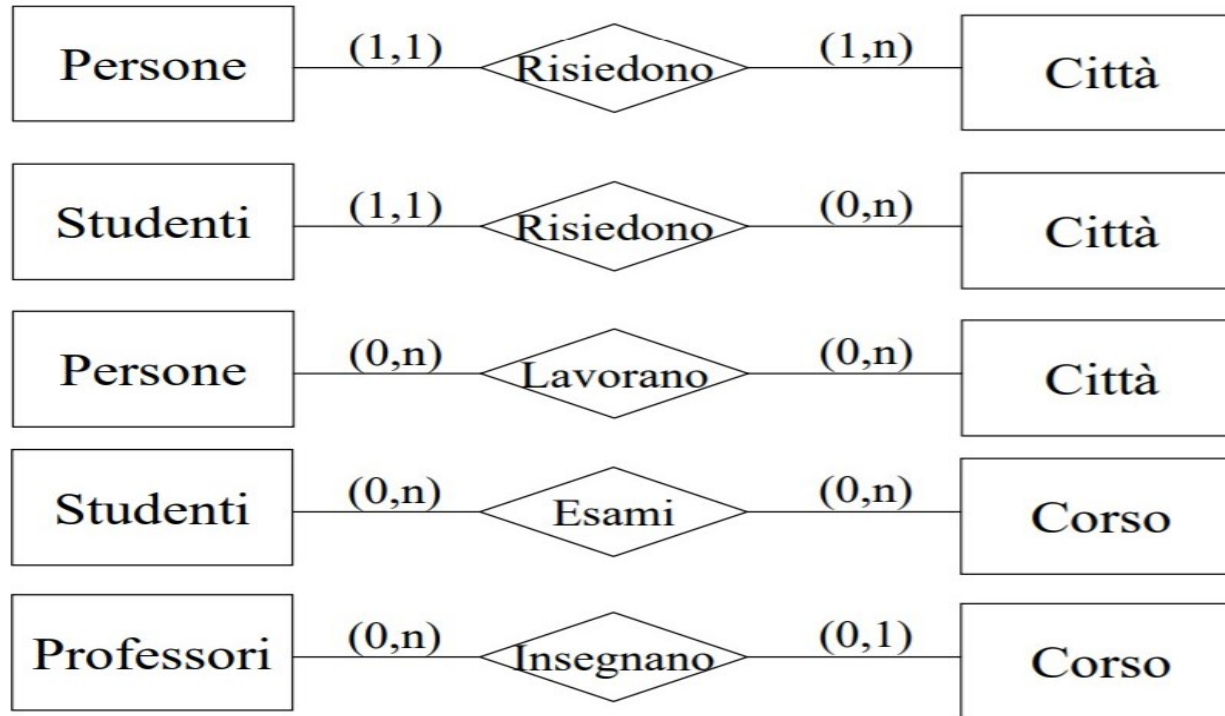
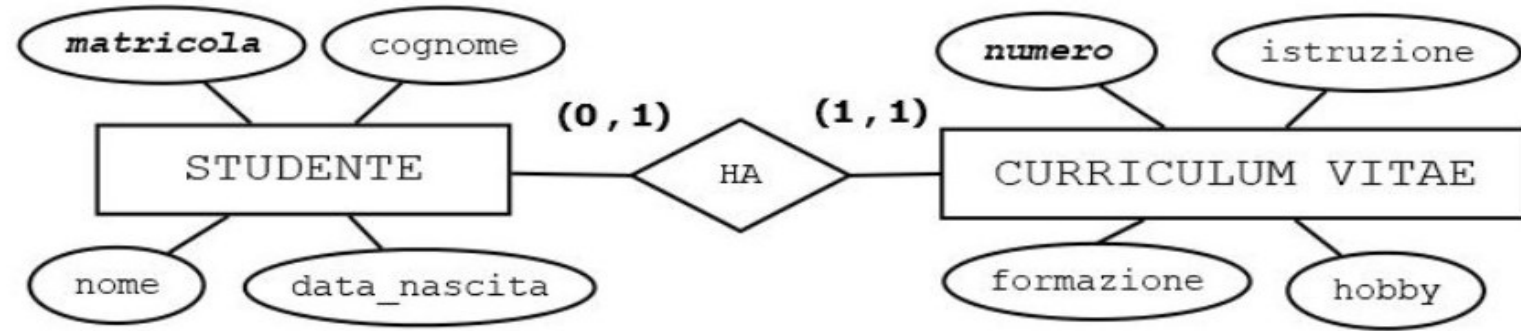
La relazione **molti a molti** invece fa corrispondere:

- ad un elemento della prima entità, tanti elementi della seconda entità;
- a ciascun elemento della seconda entità, fanno capo tanti elementi della prima entità.

Ad esempio: Ogni studente ha più Docenti e ogni Docente ha più Studenti.

NOTA: Poiché la relazione di tipo molti a molti è riconducibile, attraverso un artificio, ad una combinazione di relazioni uno a molti, ci focalizzeremo soprattutto sulle relazioni 1 a molti!

Esempi



Attributi: vincoli di cardinalità

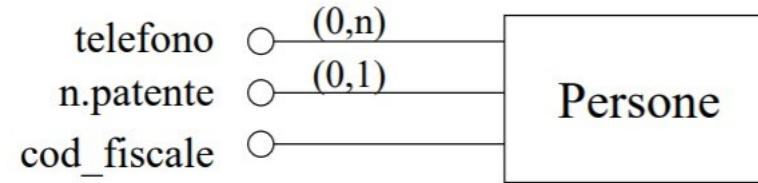
Anche per gli attributi è possibile specificare il numero minimo e massimo di valori dell'attributo che possono essere associati ad un'istanza della corrispondente associazione o entità

Graficamente si può indicare la coppia (min-card,max-card) sulla linea che congiunge l'attributo all'associazione/entità, o affianco al nome dell'attributo

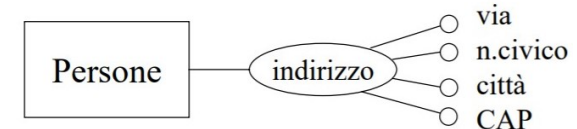
- se non si indica niente il valore di **default è (1,1)**

Si parla di attributi:

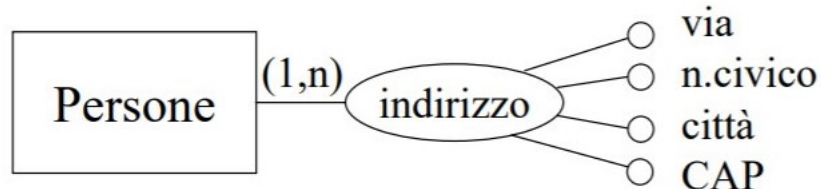
- **opzionali**: se la cardinalità minima è 0 (es. n. patente)
- **monovalore**: se la cardinalità massima è 1 (es. cod_fiscale)
- **multivalore** (o ripetuti): se la cardinalità massima è n (es. telefono)



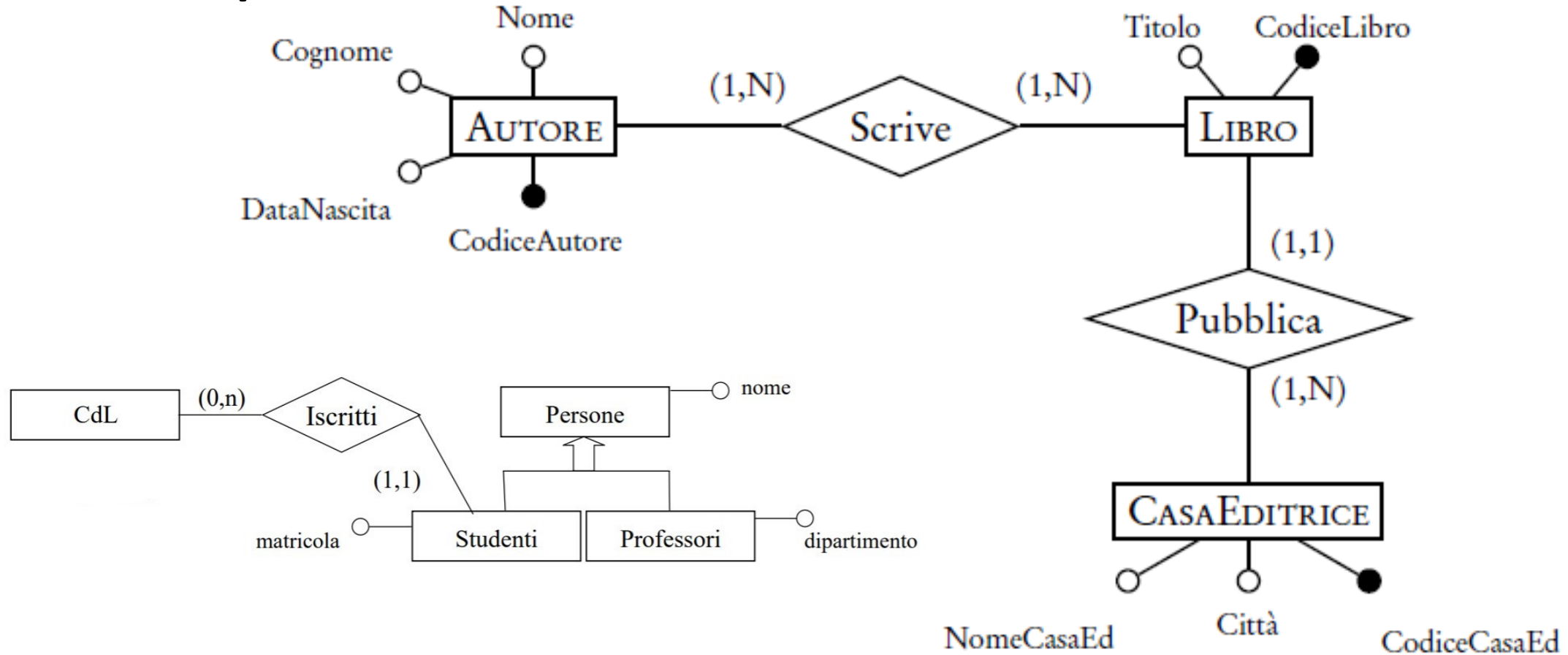
Nel caso di presenza di più attributi multivalore, la creazione di un **attributo composto** può rendersi necessaria per evitare ambiguità.

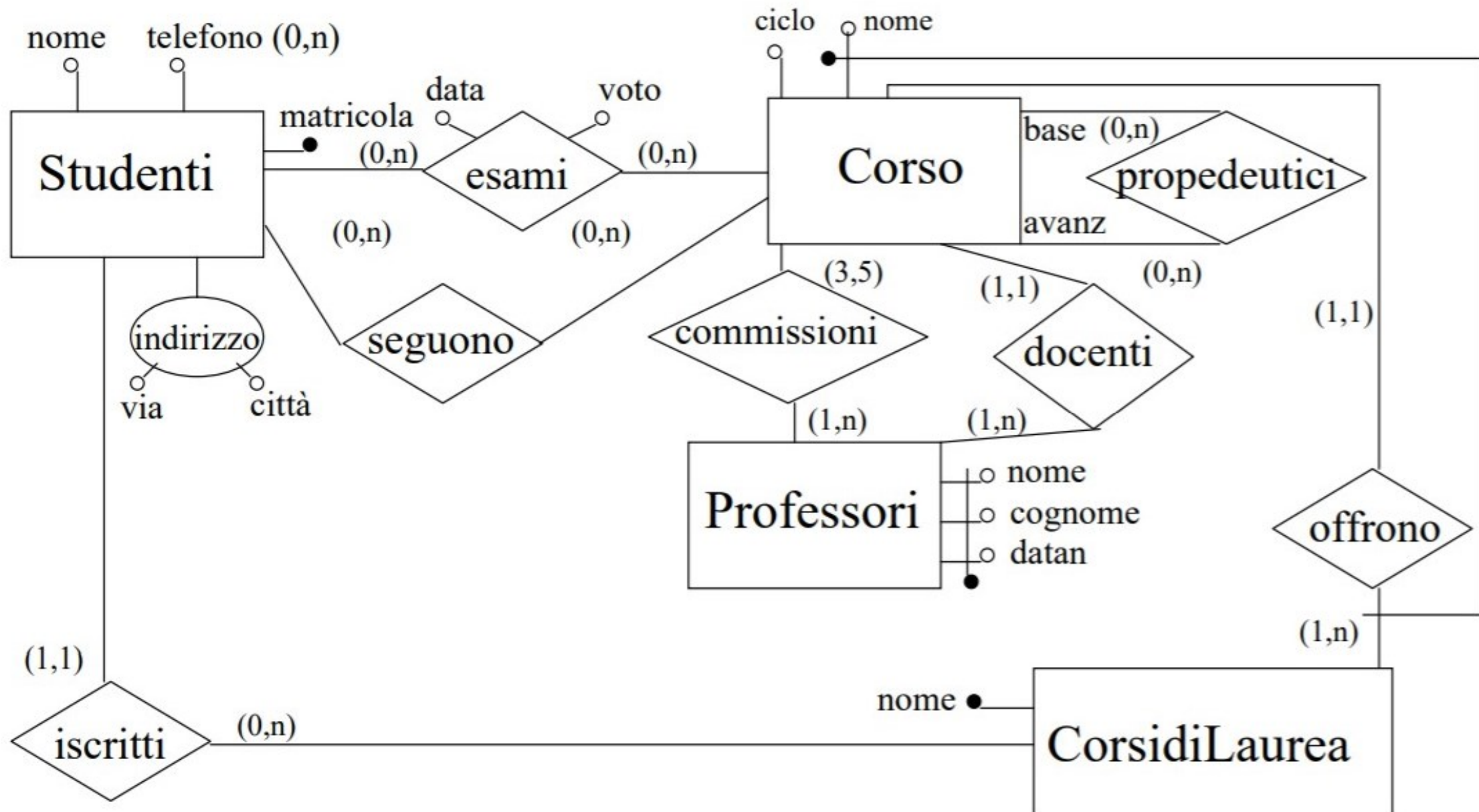


Ad esempio, se una persona ha più indirizzi



Esempi

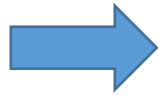




Chiavi e Integrità Referenziale

Tutte le tabelle hanno un campo

Chiave Primaria (PK: Primary Key)



codice alfanumerico o un
numero identificativo (ID)
per distinguere ciascuna riga
all'interno della tabella.

La chiave primaria di una tabella è un campo (obbligatorio) del tracciato record i cui valori identificano **univocamente** ciascun singolo record della tabella, in modo che non possano esistere due o più record della tabella con la stessa chiave primaria.

(Es. Per l'entità Studente la PK è Matricola)

Per stabilire poi i **collegamenti tra le tabelle** occorre aggiungere le **Chiavi Esterne!**

La **Chiave Esterna (FK: Foreign Key)** di una tabella è un campo del tracciato record che può ammettere valori duplicati, ma che invece è chiave primaria di un'altra tabella alla quale ci si vuole relazionare logicamente.

I record di due tabelle si mettono in relazione attraverso la coppia di campi chiave primaria/chave esterna.

Chiavi e Integrità Referenziale

Quando si mettono in relazione le tabelle, è possibile applicare all'associazione una particolare proprietà detta **integrità referenziale** che permette di rendere più forte il legame tra i record delle tabelle collegate.

L'**integrità referenziale** è una regola applicata ai valori che può assumere la chiave esterna, in modo da assicurare che i valori che questa assumerà siano sempre riferiti a quelli del campo chiave primaria in relazione.

In altre parole, l'integrità referenziale impone che **ogni inserimento di un valore della chiave esterna debba avere un valore corrispondente della chiave primaria associata** nella relazione.

Esempio Pk e Fk

La tabella dei Prodotti ha due campi aggiuntivi che rappresentano i collegamenti al codice della categoria e al codice del fornitore.

Le **tabelle** saranno così definite:

Categorie: (ID, NomeCategoria, Descrizione)

Fornitori: (CodForn, NomeSocietà, Città, Telefono)

Prodotti: (CodProdotto, NomeProdotto, Prezzo, *CodFornitore*, *IDCategoria*)

dove le chiavi primarie vengono sottolineate e le chiavi esterne sono indicate in corsivo.



 ID	NomeCategoria	Descrizione
1	Bevande	Bibite analcoliche, tè, caffè, birra
2	Dolci	Pasticceria fresca, Biscotti
3	Salumeria	Affettati, Salami, Wrustel
4	Latticini	Formaggi



 CodProdotto	NomeProdotto	Prezzo	<i>CodFornitore</i>	<i>IDCategoria</i>
100	Tè verde	€ 5	3	1
220	Tiramisù	€ 6	2	2
314	Fontina	€ 12	1	4
514	Toma	€ 7	1	4



 CodForn	NomeSocietà	Città	Telefono
1	La Pastora	NA	320 5564332
2	Dolcezze	RM	333 9887001
3	Drinking	PA	349 5435672
4	Altissima	TO	348 1267887

Riassumendo:

Quali sono gli step per creare un modello concettuale?

1. Identificare tutte le entità del sistema. Un'entità dovrebbe apparire una sola volta in un particolare diagramma.
2. Aggiungere gli attributi per le entità.
3. Identificare le relazioni tra le entità. Collegarli utilizzando una linea e aggiungere un diamante al centro che descriva il rapporto.
4. Specificare le cardinalità di relazione







Riassumendo:

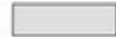
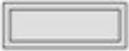









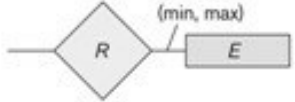
Il modello E/R è un **modello concettuale** molto utilizzato per la progettazione di basi di dati.

- Esistono molti **dialetti** E/R, che spesso si differenziano solo per la notazione grafica adottata
- I principali costrutti del modello sono **l'entità, l'associazione e l'attributo**, a cui si aggiungono **identificatori, gerarchie e vincoli di cardinalità**

N.B. L'espressività del modello E/R non è normalmente sufficiente in fase di progettazione, il che comporta la necessità di documentazione di supporto

Altro «dialetto» con altri costrutti (più «vicino» al Modello Logico)

	One
	Many
	One (and only one)
	Zero or one
	One or many
	Zero or many

Symbol	Meaning
	Entity
	Weak Entity
	Relationship
	Identifying Relationship
	Attribute
	Key Attribute
	Multivalued Attribute
	Composite Attribute
	Derived Attribute
	Total Participation of E_2 in R
	Cardinality Ratio 1: N for $E_1:E_2$ in R
	Structural Constraint (min, max) on Participation of E in R

Definitions:

entity something about which data is collected, stored, and maintained

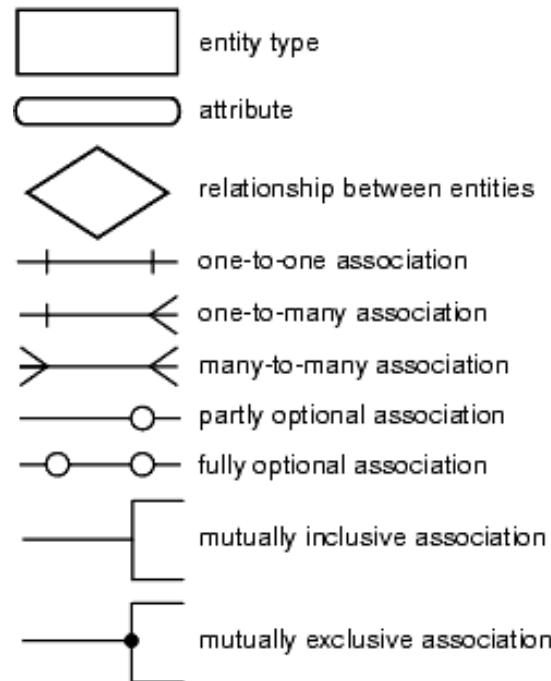
attribute a characteristic of an entity

relationship an association between entities

entity type a class of entities that have the same set of attributes

record an ordered set of attribute values that describe an instance of an entity type

Symbols:



Examples:

One A is associated with one B:



One A is associated with one or more B's:



One or more A's are associated with one or more B's:



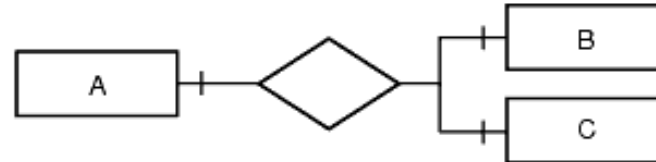
One A is associated with zero or one B:



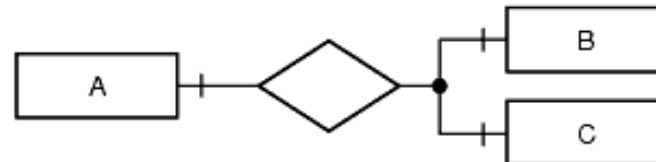
One A is associated with zero or more B's:



One A is associated with one B and one C:



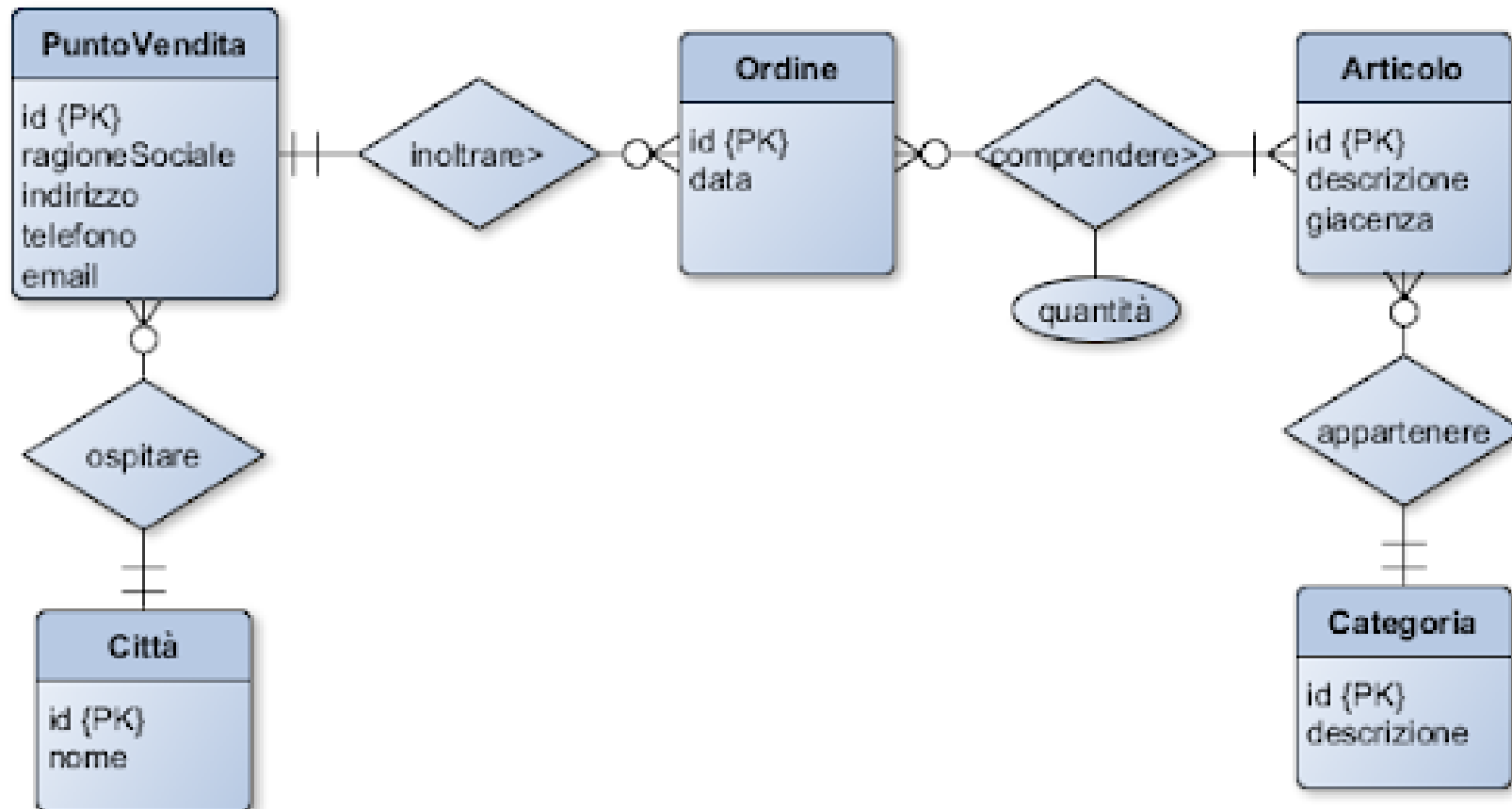
One A is associated with one B or one C (but not both):



Notazioni di cardinalità/facoltatività degli strumenti di modellazione più diffusi.

	<i>Notazione di Hoffer, Prescott e McFadden</i>	<i>Visible Analyst 7.4</i>	<i>Platinum ERwin 3.5.2</i>	<i>Microsoft Access 2000</i>	<i>Oracle Designer 6.0</i>
1:1		(Non disponibile senza cardinalità)	(Non disponibile senza cardinalità)		
1:M		(Non disponibile senza cardinalità)	(Non disponibile senza cardinalità)		
M:N		(Non disponibile senza cardinalità)		(Non consentita)	
Obbligatoria 1:1				(Nessun simbolo di facoltatività)	
Obbligatoria 1:M				(Nessun simbolo di facoltatività)	
Facoltativa 1:M				(Nessun simbolo di facoltatività)	

Esempio:



Da Modello Concettuale a Modello Logico

Nella teoria dei database relazionali è particolarmente importante il processo che permette di ottenere il modello logico a partire dal modello concettuale.

Nella progettazione di una base di dati relazionale il modello concettuale di riferimento è lo schema E/R (Entity Relationship), da cui otteniamo il modello logico relazionale.

Per intenderci, i database relazionali sono quelli che si interrogano con SQL.

Da Modello Concettuale a Modello Logico

- Ogni entità diventa una tabella
- Gli attributi dell'entità diventano colonne della tabella
- Le colonne ereditano le caratteristiche degli attributi
- La chiave primaria dell'entità diventa la chiave della tabella

Traduzione delle relazioni:

- Se l'associazione è **1 a N**, nel lato N si aggiunge una colonna, corrispondente alla chiave primaria del lato 1. Tale colonna è la chiave esterna della relazione.
- Se l'associazione è **1 a 1**, si può scegliere dove aggiungere la colonna (sempre individuata nella chiave primaria di una delle due tabelle).
- Se l'associazione è **N a N**, si aggiunge una terza tabella, che contiene le chiavi delle altre due tabelle (ed eventuali attributi riferiti a quella relazione)