

# 11. Il modello dei dati relazionale

## progettazione logica

database

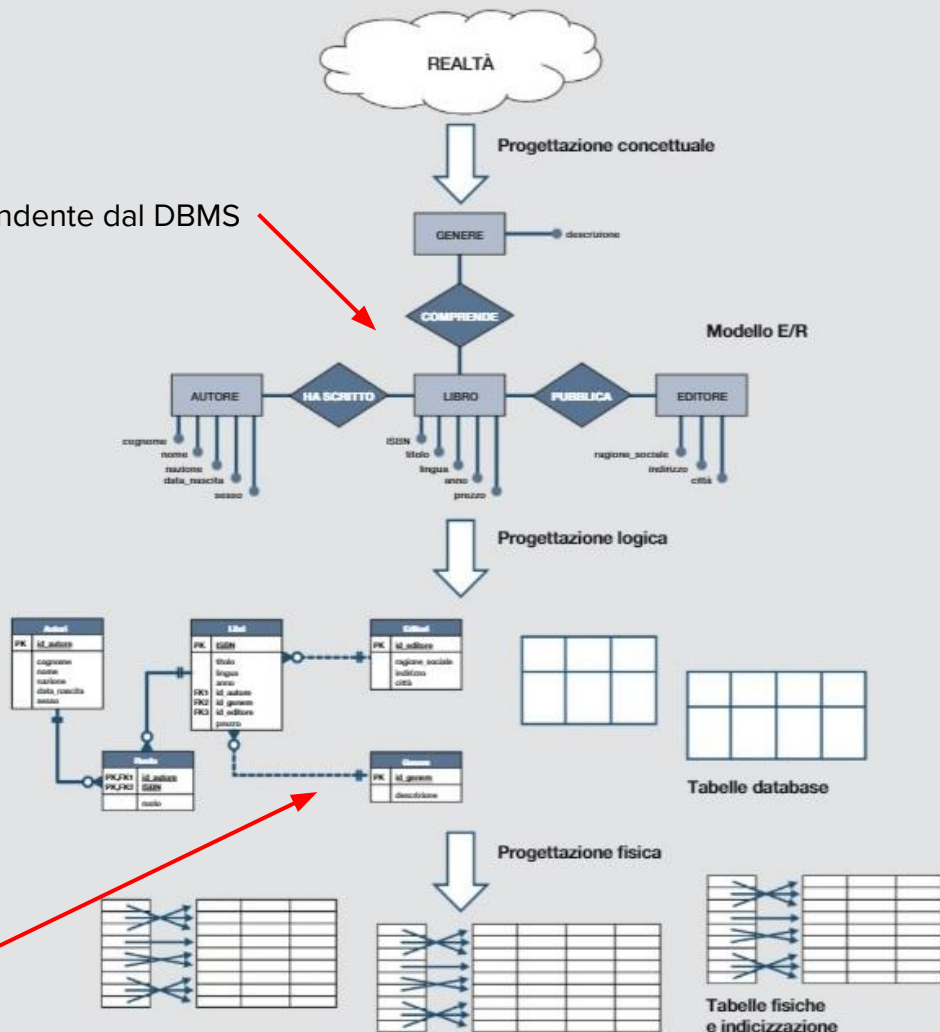
# il modello dei dati relazionale

Le fasi del processo di progettazione di un database possono essere riassunte come mostrato in figura:

- **Analisi dei requisiti;**
- **Progettazione concettuale:**  
produzione dello *Schema E/R*;
- **Progettazione logica:**  
produzione del *progetto del DB*;
- **Progettazione fisica:** realizzazione delle *tabelle*.

Dipendente dal DBMS

Indipendente dal DBMS



## il modello dei dati relazionale - progettazione logica

Durante la precedente fase di **Progettazione Concettuale**, abbiamo realizzato un modello concettuale (rappresentazione semplificata) di una determinata situazione reale, schematizzata graficamente in un diagramma E/R, al fine di evidenziarne le entità caratterizzanti e le associazioni tra esse.

Lo scopo fondamentale è quello di mettere in evidenza gli **aspetti semantici** dei dati di interesse, ragionando sul significato che essi hanno all'interno del sistema informativo.

La fase successiva di **Progettazione Logica** è più interessata agli **aspetti funzionali** del database e ne costituisce di fatto il punto di partenza per l'effettiva realizzazione.

In questa seconda fase dunque forniremo il diagramma E/R appena prodotto, di un modello logico dei dati adeguato a formalizzare le operazioni che dovranno poi essere eseguite sul database stesso.

# il modello dei dati relazionale - progettazione logica

Esistono diversi modelli logici utilizzabili per la realizzazione di questa fase progettuale (tra questi ricordiamo il modello gerarchico e quello reticolare, proposti negli anni sessanta e implementati nei primi DBMS durante gli anni settanta e ottanta), ma attualmente quello di gran lunga più diffuso è senza dubbio il **Modello Relazionale**.

Introdotta nel 1970 dal ricercatore della IBM Edgar Codd, il modello dei dati relazionale si basa sul concetto di *relazione* matematica (rappresentata come tabella di valori), e poggia il suo fondamento logico nella teoria degli insiemi.

Forte del suo rigido formalismo matematico, il modello di dati relazionale ci consente di definire in modo preciso ed esaustivo:

- una *notazione* per descrivere le strutture dei dati;
- un insieme di *operazioni* per gestirli e realizzare le interrogazioni.

## il modello dei dati relazionale - progettazione logica

Il modello relazionale soddisfa il requisito di indipendenza dei dati relativamente all'architettura dei DBMS, dal momento che separa nettamente il livello fisico da quello logico.

In pratica gli utenti che accedono ai dati, così come i programmatori che sviluppano l'interfaccia del database, fanno riferimento solo al livello logico, vedono cioè i dati come tabelle (relazioni del modello relazionale) ed applicano ad esse gli operatori dell'algebra relazionale; non è necessario conoscere i dettagli delle strutture fisiche di memorizzazione e di gestione dei dati all'interno delle memorie di massa.

Usando un modello gerarchico o reticolare invece, bisogna fare esplicito riferimento alla struttura realizzativa sottostante, ad es. attraverso l'uso di puntatori.

# il modello dei dati relazionale - progettazione logica

Il database in figura, che mette in relazione ogni agente con i propri clienti, evidenzia questa caratteristica fondamentale del modello relazionale indicata spesso dicendo che tale modello logico è “*basato su valori*”.

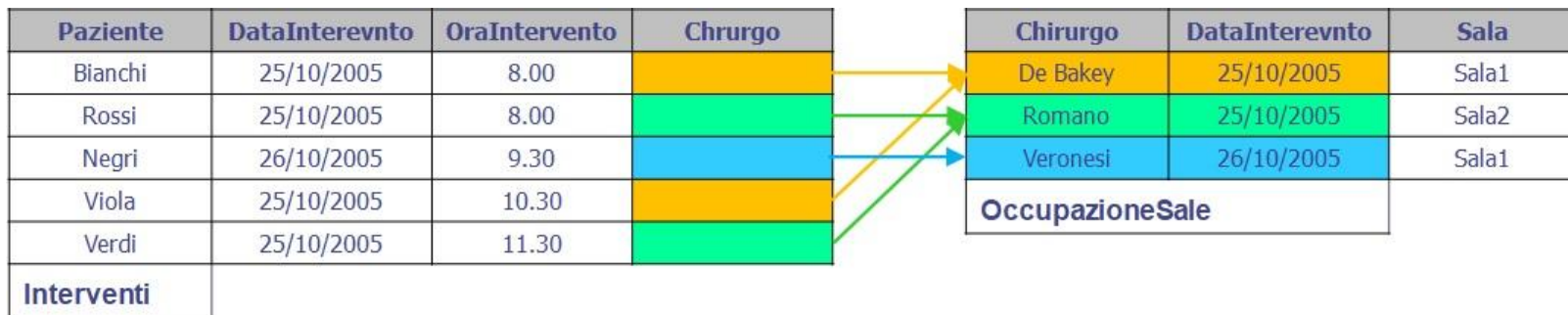
CodAgente	CodCli	Nominativo	Indirizzo	Prov
AK 0098	C 002	Antonia Rossi	Via Roma, 126	RM
AK 0098	A 011	Giacomo Alfieri	Via Napoli, 29	NA
JP 1123	A 004	Franco Pisori	Via Venezia, 38	VE
AK 0098	C 014	Maria Petrocelli	Via Bologna, 78	BO
PM 0753	B 003	Piero Giovanelli	Via Palermo, 23	NA
null	D 147	Gianfranco Verdi	Via Cosenza, 18	CS
JP 1123	A 012	Carla Vecchi	Via Salerno, 88	MI
CLIENTI				

CodAgente	NomeAgente	CodZona
AK 0098	Carlo Bressi	AB 2
PM 0753	Pietro Conti	PP 5
JP 1123	Alba Sergi	KR 1
AGENTI		

I collegamenti fra i dati in tabelle diverse (in questo caso, tra *Agenti* e *Clienti*) sono rappresentati tramite l'uguaglianza dei valori contenuti nei campi in comune (il valore di *CodAgente* lega ogni agente con i suoi clienti) .

# il modello dei dati relazionale - progettazione logica

Va notato che altri modelli logici (reticolare e gerarchico) realizzano invece i riferimenti tra i dati in modo esplicito attraverso puntatori e vengono pertanto definiti modelli *"basati su record e puntatori"*.



# il modello dei dati relazionale - progettazione logica

Rispetto ad un sistema basato su record e puntatori, il modello relazionale, basato su valori, presenta diversi vantaggi:

- richiede di rappresentare solo ciò che è rilevante dal punto di vista dell'utente: i puntatori sono qualcosa di aggiunto, legato ad aspetti realizzativi;
- la rappresentazione logica dei dati (basata su valori) non fa riferimento alcuno a quella fisica, permettendo quindi di ottenere l'indipendenza fisica dei dati;
- essendo l'informazione contenuta nei valori, è semplice trasferire i dati da un contesto ad un altro: i puntatori invece hanno significato locale al singolo sistema.



## il modello dei dati relazionale - progettazione logica

**NB:** Il modello di dati che verrà adottato durante la progettazione logica dipende fortemente dal tipo di DBMS che verrà poi utilizzato per la realizzazione fisica del sistema informativo corrispondente.

**Un database si dice relazionale quando al livello di progettazione logica è basato appunto sul modello di dati relazionale.**

La maggior parte dei moderni DBMS sono basati sul modello dei dati relazionale; tra i più diffusi ricordiamo: Oracle DB, IBM DB2, Microsoft SQL-server e MySQL di recente acquisito dalla Oracle Corporation.

# il modello dei dati relazionale - progettazione logica

Il modello relazionale realizza il database fondamentalmente come una collezione di relazioni (rappresentate mediante tabelle), dove una relazione, dal punto di vista matematico può essere interpretata semplicemente come un sottoinsieme del prodotto cartesiano di  $n$  insiemi.

Siano dati gli insiemi  $A = \{1,2,3\}$  e  $B = \{a,b\}$ , il prodotto cartesiano  $A \times B$  è l'insieme di tutte le possibili coppie in cui il primo elemento appartiene ad  $A$ , mentre il secondo appartiene a  $B$ :

$$A \times B = \{ (1,a), (\mathbf{1,b}), (2,a), (2,b), (\mathbf{3,a}), (3,b) \}$$

per cui, ad es. l'insieme  $R = \{ (1,b), (3,a) \}$ , essendo un sottoinsieme di  $A \times B$  ( $R \subset A \times B$ ), è una relazione matematica sugli insiemi  $A$  e  $B$ .

# il modello dei dati relazionale - progettazione logica

Più in generale, siano dati  $n$  insiemi non necessariamente distinti (detti *domini* della relazione)  $D_1, D_2, \dots, D_n$ , si dice **relazione un sottoinsieme dell'insieme di tutte le ennuple  $d_1, d_2, \dots, d_n$  che si ottengono prendendo un elemento  $d_1$  dal primo dominio  $D_1$ ,  $d_2$  dal secondo dominio  $D_2$ , e così via.**

Siano dati ed es.  $D_1$  l'insieme dei nominativi degli studenti iscritti in un certo ateneo universitario,  $D_2$  l'insieme delle corrispondenti città di residenza,  $D_3$  l'insieme dei corsi di laurea previsti:

$D_1 = \text{Studente}$
Baldieri Marco
Criniti Annarita
Velasco Ester
Bitonti Piero

$D_2 = \text{CittàResidenza}$
Catanzaro
Soverato
Crotone

$D_3 = \text{CorsoLaurea}$
Scienze Infermieristiche
Fisioterapista
Logopedista

# il modello dei dati relazionale - progettazione logica

Il prodotto cartesiano  $D_1 \times D_2 \times D_3$ , è l'insieme generato dalle terne ottenute combinando in tutti i modi possibili gli elementi di ciascuno degli insiemi coinvolti:

$D_1 = \text{Studente}$	$D_2 = \text{CittàResidenza}$	$D_3 = \text{CorsoLaurea}$
Baldieri Marco	Catanzaro	Scienze Infermieristiche
Baldieri Marco	Catanzaro	Fisioterapista
Baldieri Marco	Catanzaro	Logopedista
Baldieri Marco	Soverato	Scienze Infermieristiche
Baldieri Marco	Soverato	Fisioterapista
Baldieri Marco	Soverato	Logopedista
Baldieri Marco	Crotone	Scienze Infermieristiche
Baldieri Marco	Crotone	Fisioterapista
Baldieri Marco	Crotone	Logopedista
Criniti Annarita	Catanzaro	Scienze Infermieristiche
Criniti Annarita	Catanzaro	Fisioterapista
Criniti Annarita	Catanzaro	Logopedista
...	...	...

Ogni riga del prodotto cartesiano si definisce *ennupla* (o ***tupla***).

# il modello dei dati relazionale - progettazione logica

La seguente relazione che mostra dati reali, cioè gli studenti effettivamente iscritti al primo anno, con la loro città di residenza e l'indicazione del corso di laurea cui sono realmente iscritti, è formata da soli 4 elementi (righe) compresi tra quelli del prodotto cartesiano  $D_1 \times D_2 \times D_3$ , vale a dire:

$D_1 = \text{Studente}$	$D_2 = \text{CittàResidenza}$	$D_3 = \text{CorsoLaurea}$
Baldieri Marco	Catanzaro	Logopedista
Criniti Annarita	Catanzaro	Fisioterapista
Velasco Ester	Crotone	Scienze Infermieristiche
Bitonti Piero	Soverato	Scienze Infermieristiche
<b>IscrittiPrimoAnno</b>		

$IscrittiPrimoAnno \subseteq \text{Studente} \times \text{CittàResidenza} \times \text{CorsoLaurea}$

# il modello dei dati relazionale - progettazione logica

Il numero delle colonne  $n$  si definisce **grado** della relazione, mentre il numero delle *ennuple* (dette genericamente *tuple*) si definisce la **cardinalità** della relazione, (da non confondere con la cardinalità della partecipazione di una associazione).

Gli insiemi  $D_i$  si chiamano **domini** e descrivono lo **Schema di Relazione** indicato come  $Rel(D_1, D_2, \dots, D_n)$ , dove  $Rel$  è il nome dello schema di relazione.

Rel (D <sub>1</sub> , D <sub>2</sub> , ... , D <sub>n</sub> )					Domini	cardinalità
	D <sub>1</sub>	D <sub>2</sub>	...	D <sub>n</sub>		
prima tupla	d <sub>1</sub> <sup>1</sup>	d <sub>2</sub> <sup>1</sup>	...	d <sub>n</sub> <sup>1</sup>		
seconda tupla	d <sub>1</sub> <sup>2</sup>	d <sub>2</sub> <sup>2</sup>	...	d <sub>n</sub> <sup>2</sup>		
	...	...	...	...		
i-ma tupla	d <sub>1</sub> <sup>i</sup>	d <sub>2</sub> <sup>i</sup>	...	d <sub>n</sub> <sup>i</sup>		
	...	...	...	...		
grado						

# il modello dei dati relazionale - progettazione logica

Uno stesso schema di relazione può definire relazioni (**istanze di relazione**) diverse .

Ad es. le tabelle mostrate di seguito e relative agli iscritti al primo anno di corso, magari per annate diverse, rappresentano relazioni diverse:

Studente	CittàResidenza	CorsoLaurea
Scolieri Claudia	Catanzaro	Logopedista
Minniti Roberto	Cosenza	Fisioterapista
Casella Antonio	Crotone	Scienze Infermieristiche
Rispoli Andrea	Catanzaro	Scienze Infermieristiche
Genni Ester	Soverato	Fisioterapista
Merli Piero	Catanzaro	Scienze Infermieristiche
IscrittiPrimoAnno		

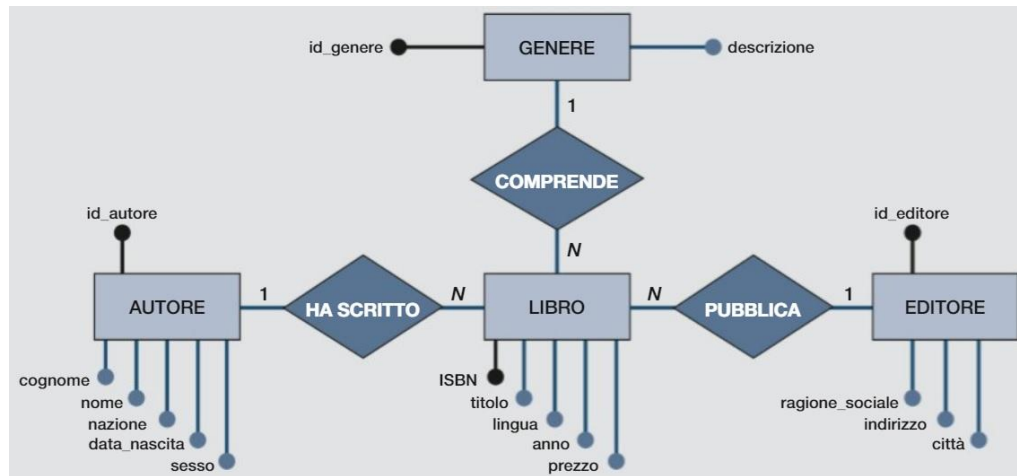
Studente	CittàResidenza	CorsoLaurea
Baldieri Marco	Catanzaro	Scienze Infermieristiche
Criniti Annarita	Soverato	Fisioterapista
Velasco Ester	Crotone	Scienze Infermieristiche
Bitonti Piero	Catanzaro	Scienze Infermieristiche
IscrittiPrimoAnno		

ma entrambe sono istanze appartenenti all'insieme delle relazioni definite dal seguente schema di relazione: *IscrittiPrimoAnno(Studente, CittàResidenza, CorsoLaurea)*.

# il modello dei dati relazionale - progettazione logica

Possiamo definire uno **Schema di Database** un insieme di schemi di relazione che sono relativi ad uno stesso database.

Ad es. il DB relativo al diagramma E/R della biblioteca mostrato di lato, comprende quattro schemi di relazione; lo schema di database è il seguente:



*Autore*(id\_autore, cognome, nome, data\_nascita, sesso)

*Libro*(ISBN, titolo, lingua, anno, prezzo, **id\_autore**, **id\_editore**, **id\_genere**)

*Editore*(id\_editore, ragione\_sociale, indirizzo, città)

*Genere*(id\_genere, descrizione)



## il modello dei dati relazionale - progettazione logica

Con riferimento ad uno schema di database, definiamo allora un **database** (o base di dati) come una **istanza di uno schema di database**, cioè un insieme di relazioni definite sugli schemi di relazione facenti parte dello schema di database in questione (e derivanti da uno specifico diagramma E/R).

Su uno stesso schema di database, possiamo definire diverse istanze di database, cioè diversi database reali dove ogni tabella è una istanza di relazione definita su uno degli schemi di relazione del nostro schema di database.

# il modello dei dati relazionale - progettazione logica

Una relazione (dal punto di vista matematico) è un insieme di *tuple* (ennuple); quindi in una relazione, vista come sottoinsieme del prodotto cartesiano di più domini:

- **non viene definito alcun ordinamento fra le sue tuple**, visto che gli elementi di un insieme non hanno nessun ordine in particolare;
- **le tuple sono tutte diverse tra loro** dal momento che in un insieme non possono esistere elementi uguali;
- **ciascuna tupla al proprio interno è ordinata** in quanto l'*i*-esimo valore appartiene all'*i*-esimo dominio (se scambiassimo la posizione dei domini di una relazione, ne modificheremmo il significato, ottenendo una relazione differente derivata da uno schema di relazione differente); in effetti il prodotto cartesiano non gode della proprietà commutativa.

## il modello dei dati relazionale - progettazione logica

Quest'ultima proprietà risponde ad un principio matematico di posizionalità, che poco si addice però alle nostre esigenze di utilizzare le relazioni per la memorizzazione dei dati.

In ambito informatico si tende invece a privilegiare notazioni non posizionali in cui una relazione viene vista sostanzialmente come un insieme di righe omogenee, cioè definite sugli stessi campi, ad ognuno dei quali associamo un nome detto appunto *campo* (o **attributo**) ed assegnamo per ciascuna riga uno specifico valore del dominio corrispondente.

In generale si parla allora di **relazione con attributi** facendo riferimento ad un insieme di righe omogenee nelle quali ogni attributo prende valori da uno specifico dominio.

Una *relazione con attributi* può essere rappresentata graficamente in forma tabellare.

# il modello dei dati relazionale - progettazione logica

Diversamente dalla definizione insiemistica di relazione vista prima, **una relazione con attributi presenta le seguenti proprietà:**

- Possiede tante colonne quanti sono i domini (cioè il *grado* della relazione) e tante righe quante sono le tuple (la *cardinalità*);
- Le colonne sono identificate da attributi distinti e ognuna di esse corrisponde a un dominio della relazione R (l'insieme degli attributi di una tale relazione R è indicato con il termine *schema di relazione* e rappresenta l'*aspetto intensionale* dei dati);
- Ogni riga corrisponde ad una tupla: non vi possono essere due o più righe uguali (il contenuto delle righe della tabella rappresenta l'*aspetto estensionale* dei dati); vedremo comunque che questa proprietà viene superata;
- L'ordine delle righe e delle colonne non è più significativo.

## il modello dei dati relazionale - progettazione logica

A proposito di quest'ultima affermazione, ricordiamo che una relazione, in quanto prodotto cartesiano di un insieme di domini, è tale che variando l'ordine dei domini coinvolti, otteniamo una relazione differente, appartenente ad uno schema di relazione differente, visto che il prodotto cartesiano non è commutativo.

Quindi le due tuple: (*Genni Ester, Catanzaro, Fisioterapista*) e (*Genni Ester, Fisioterapista, Catanzaro*) sono diverse, in quanto derivano da due schema di relazione diversi:

*IscrittiPrimoAnno(Studiante, CittàResidenza, CorsoLaurea);*

*IscrittiPrimoAnno(Studiante, CorsoLaurea, CittàResidenza).*

Cionondimeno, il contenuto informativo di una relazione con attributi, cioè di una tabella usata per la memorizzazione dei dati di un archivio, non cambia modificando l'ordine con il quale vengono riportate le sue colonne.

## il modello dei dati relazionale - progettazione logica

Appare subito evidente la somiglianza tra i record di un archivio (realizzato mediante una tabella) come descritto in precedenza, e le tuple di una relazione, tra i campi del record (nel *tracciato del record*) e i domini della relazione (nello *schema di relazione*).

Per questo motivo nella terminologia corrente dei database relazionali si usano indifferentemente i termini *record* e *campo* al posto di *tupla* (o di riga) e *dominio*, mentre il termine *tabella* viene assunto come sinonimo di *relazione*.

Da questo momento in poi, dove non vi fosse rischio di confusione, potremo parlare indifferentemente di *tabella*, *campo* e *record* (o riga) al posto di *relazione*, *dominio* e *tupla*.

**Il modello relazionale di un DB è costituito dunque da un insieme di relazioni/tabelle.**