

Modello Relazionale

Modelli Logici

- Tre modelli logici **tradizionali**
 - gerarchico, reticolare, **relazionale**
- Più **recenti**
 - a oggetti (poco diffuso), basato su XML, **NoSQL**
- Gerarchico e reticolare
 - utilizzano **riferimenti espliciti** (puntatori) fra record
- Relazionale
 - è basato su **valori**
 - anche i **riferimenti** fra dati in strutture (relazioni) diverse sono rappresentati per mezzo dei **valori** stessi

Esempio di Modello Relazionale

Studenti

Matricola	Cognome	Nome	Nascita
6554	Rossi	Mario	05/12/78
8765	Neri	Paolo	03/11/76
9283	Verdi	Luisa	12/11/79
3456	Rossi	Maria	01/02/79

Esami

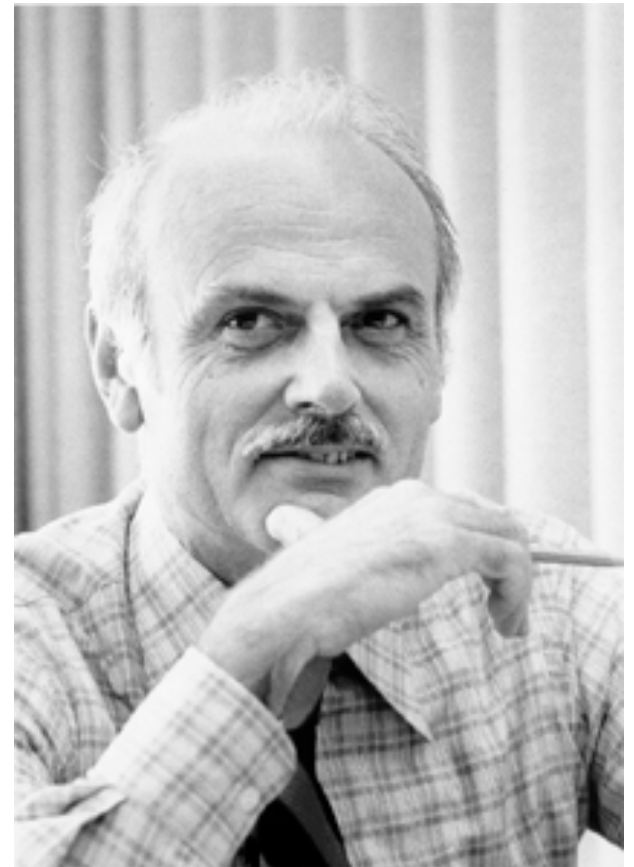
Studente	Voto	Corso
3456	30	4
3456	24	2
9283	28	1
6554	26	1

Corsi

Codice	Titolo	Docente
1	Analisi	Bruni
2	Chimica	Peri
4	Chimica	Fino

Modello Relazionale

- Proposto da **E. F. Codd** nel **1970** per favorire l'indipendenza dei dati
- Disponibile in DBMS reali nel **1981** (non è facile implementare l'indipendenza con efficienza e affidabilità)
- Si basa sul **concetto matematico di relazione** (con una variante)
- Le relazioni hanno naturale rappresentazione per mezzo di **tabelle**



Edward Frank Codd
1923 - 2003
Turing Award 1981

Turing Award

- L'**A.M. Turing Award** (in italiano, premio Turing) è un premio, assegnato annualmente dalla **Association for Computing Machinery** (ACM), a una personalità che eccelle per i contributi di natura tecnica offerti alla comunità informatica, in particolare per progressi che siano duraturi e di elevata importanza tecnica
- Il premio viene spesso anche chiamato “**premio Nobel dell’informatica**” ed è intitolato al matematico inglese **Alan Mathison Turing** (1912-1954), in riconoscimento del suo contributo unico e originale alla nascita delle attività di calcolo mediante dispositivi automatici



Association for
Computing
Machinery
acm.org

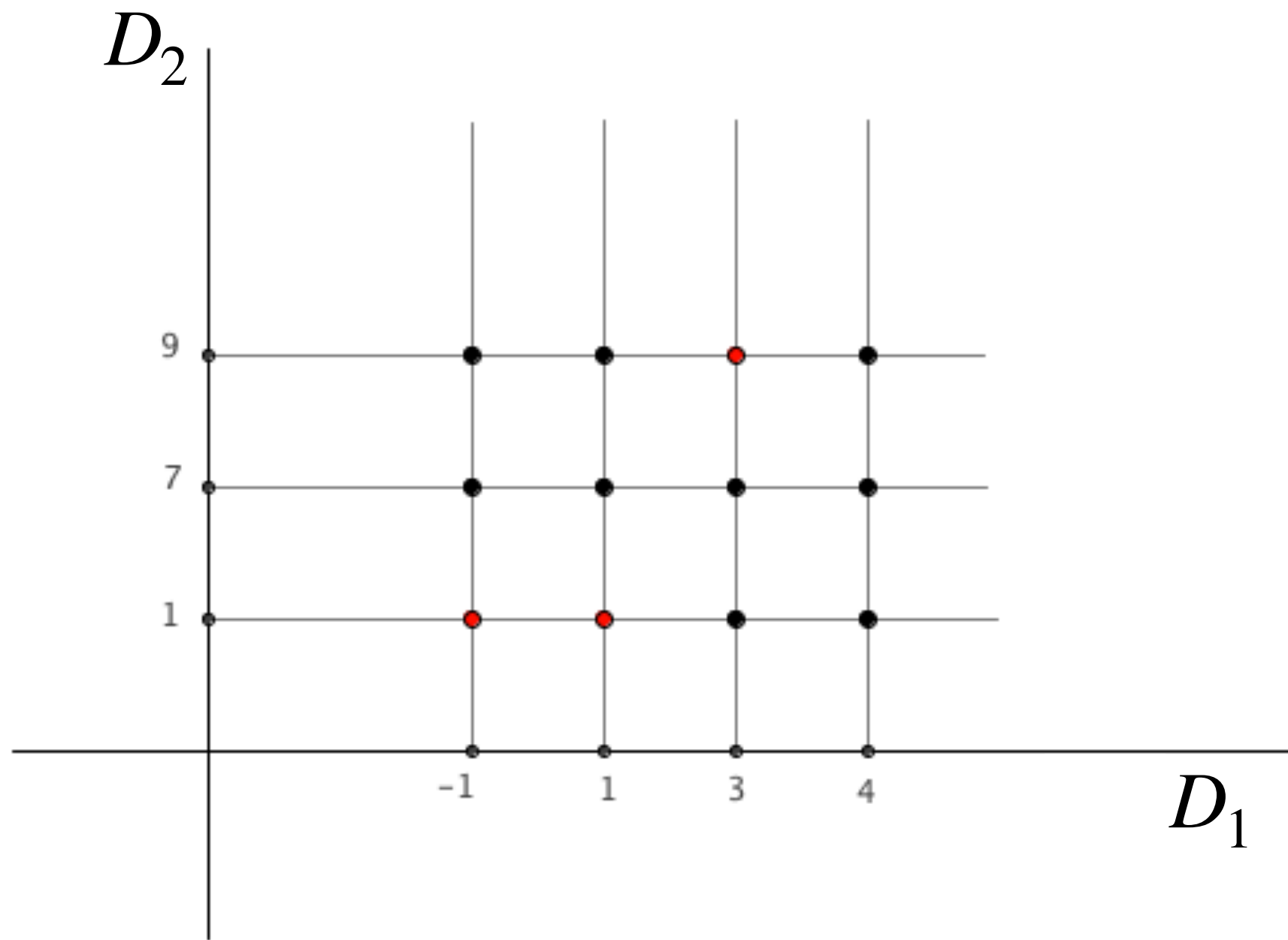
Alcuni vincitori

Anno	Vincitore	Nazione	Note
1971	John McCarthy	USA	Padre dell'AI
1972	Edsger Dijkstra	NL	Algoritmo del cammino minimo
1975	Donald Knuth	USA	Arte della programmazione
1977	John Backus	USA	FORTRAN
1981	Edgar Codd	USA	Database relazionali
1998	Jim Gray	USA	Database transazionali
2004	V. Cerf, R. Kahn	USA	TCP/IP
2014	M. Stonebaker	USA	Database moderni
2016	Tim Berners Lee	UK	World Wide Web
2018	Bengio, Hinton, LeCun	CA	Padrini dall'AI

Esempio di Relazione Matematica

- Insieme
 - $D_1 = \{a, b\}$
- Insieme
 - $D_2 = \{x, y, z\}$
- Prodotto cartesiano
 - $D_1 \times D_2 = \{(a, x), (a, y), (a, z), (b, x), (b, y), (b, z)\}$
- Relazione
 - $R = \{(a, x), (a, z), (b, y)\} \subseteq D_1 \times D_2$

Esempio di Relazione Matematica



$$R = \{(-1, 1), (1, 1), (3, 9)\} \subseteq D_1 \times D_2$$

Relazione Matematica

- Siano dati n **insiemi** (anche non distinti) D_1, D_2, \dots, D_n
- Definiamo il loro **prodotto cartesiano** $D_1 \times D_2 \times \dots \times D_n$ come l'insieme di tutte le n -uple (d_1, d_2, \dots, d_n) tali che $d_1 \in D_1, d_2 \in D_2, \dots, d_n \in D_n$
- Una **relazione** (matematica) è un sottoinsieme di $D_1 \times D_2 \times \dots \times D_n$
 - **Non c'è un ordinamento** tra la n -uple, dette anche **tuple**
 - **Non esistono n -uple uguali**
 - **Ogni n -upla è ordinata**: l' i -esimo valore d_i proviene dall' i -esimo insieme D_i
- Gli insiemi D_1, D_2, \dots, D_n sono i **domini** della relazione

Esempio di Relazione Matematica

- $Partite \subseteq \text{String} \times \text{String} \times \mathbb{N} \times \mathbb{N}$

Juventus	Lazio	3	1
Lazio	Milan	2	0
Juventus	Roma	0	2
Roma	Milan	0	1

- Ciascuno dei domini “ripetuti” ha due **ruoli** diversi, distinguibili attraverso la **posizione**
 - La **struttura** è **posizionale**

Struttura Non Posizionale

- A ciascun dominio si associa un **nome unico** nella tabella, detto **attributo**, che ne descrive il **ruolo**

Casa	Fuori	RetiCasa	RetiFuori
Juventus	Lazio	3	1
Lazio	Milan	2	0
Juventus	Roma	0	2
Roma	Milan	0	1

Il Modello Relazionale è basato sui valori

- I riferimenti fra dati in **relazioni diverse** sono rappresentati per mezzo di **valori** dei domini che compaiono nelle n -uple

Il Modello Relazionale è basato sui valori

Studenti

Matricola	Cognome	Nome	Nascita
6554	Rossi	Mario	05/12/78
8765	Neri	Paolo	03/11/76
9283	Verdi	Luisa	12/11/79
3456	Rossi	Maria	01/02/79

Esami

Studente	Voto	Corso
3456	30	4
3456	24	2
9283	28	1
6554	26	1

Corsi

Codice	Titolo	Docente
1	Analisi	Bruni
2	Chimica	Peri
4	Chimica	Fino

Il Modello Relazionale è basato sui valori

Studenti

Matricola	Cognome	Nome	Nascita
6554	Rossi	Mario	05/12/78
8765	Neri	Paolo	03/11/76
9283	Verdi	Luisa	12/11/79
3456	Rossi	Maria	01/02/79

Esami

Studente	Voto	Corso
3456	30	4
3456	24	2
9283	28	1
6554	26	1

Corsi

Codice	Titolo	Docente
1	Analisi	Bruni
2	Chimica	Peri
4	Chimica	Fino

Il Modello Relazionale è basato sui valori

Studenti

Matricola	Cognome	Nome	Nascita
6554	Rossi	Mario	05/12/78
8765	Neri	Paolo	03/11/76
9283	Verdi	Luisa	12/11/79
3456	Rossi	Maria	01/02/79

Esami

Studente	Voto	Corso
3456	30	4
3456	24	2
9283	28	1
6554	26	1

Corsi

Codice	Titolo	Docente
1	Analisi	Bruni
2	Chimica	Peri
4	Chimica	Fino

Definizioni

- **Schema di relazione:**

- un simbolo R detto **nome della relazione** e un insieme di (nomi di) **attributi** $X = \{A_1, \dots, A_n\}$, solitamente indicato con $R(X)$
- A ciascun attributo è associato un dominio

- **Schema di base di dati:**

- un **insieme di schemi di relazione** con nomi diversi, solitamente indicato come $\mathbf{R} = \{R_1(X_1), \dots, R_m(X_m)\}$

Definizioni

- Una ***n*-upla** o **tupla** su un insieme di attributi X è una **funzione** che associa a ciascun **attributo** $A \in X$ un **elemento**, o **valore**, nel **dominio** di A
- Il simbolo $t[A]$ denota il valore della *n*-upla t sull'attributo A
- Il simbolo $t[Y]$, con $Y \subseteq X$, denota il valore della *n*-upla t sull'insieme di attributi Y

Definizioni

- **(Istanza di) relazione** su uno schema $R(X)$:
 - insieme r di n -uple su X
- **(Istanza di) base di dati**: su uno schema $\mathbf{R} = \{R_1(X_1), \dots, R_m(X_m)\}$:
 - insieme di relazioni $\mathbf{r} = \{r_1, \dots, r_m\}$ dove ogni r_i è una relazione sullo schema $R_i(X_i)$

Esempio 1

Studenti

Matricola	Cognome	Nome	Nascita
6554	Rossi	Mario	05/12/78
8765	Neri	Paolo	03/11/76
9283	Verdi	Luisa	12/11/79
3456	Rossi	Maria	01/02/79

Esami

Studente	Voto	Corso
3456	30	4
3456	24	2
9283	28	1
6554	26	1

Corsi

Codice	Titolo	Docente
1	Analisi	Bruni
2	Chimica	Peri
4	Chimica	Fino

Esempio 1

- $X_1 = \{ \text{Matricola, Cognome, Nome, Nascita} \}$
- $X_2 = \{ \text{Studente, Voto, Corso} \}$
- $X_3 = \{ \text{Codice, Titolo, Docente} \}$
- $R_1 = \text{Studenti}$
- $R_2 = \text{Esami}$
- $R_3 = \text{Corsi}$

$$\begin{aligned} \mathbf{R} &= \{ R_1(X_1), R_2(X_2), R_3(X_3) \} = \\ &\quad \{ \text{Studenti}(\text{Matricola, Cognome, Nome, Nascita}), \\ &= \quad \text{Esami}(\text{Studente, Voto, Corso}), \\ &\quad \text{Corsi}(\text{Codice, Docente, Titolo}) \} \end{aligned}$$

Esempio 2

- Relazione su singolo attributo:
Studenti

Matricola	Cognome	Nome	Nascita
6554	Rossi	Mario	05/12/78
8765	Neri	Paolo	03/11/76
9283	Verdi	Luisa	12/11/79
3456	Rossi	Maria	01/02/79

Studenti Lavoratori

Matricola
6554
3456

Esempio 3

- Strutture nidificate:

<i>Da Filippo</i> <i>Via Roma 2, Roma</i>		
<i>Ricevuta Fiscale</i> <i>1235 del 12/10/2017</i>		
3	Coperti	3,00
2	Antipasti	6,20
3	Primi	12,00
2	Bistecche	18,00
<i>Totale</i>		39,20

<i>Da Filippo</i> <i>Via Roma 2, Roma</i>		
<i>Ricevuta Fiscale</i> <i>1240 del 13/10/2017</i>		
2	Coperti	2,00
2	Antipasti	7,00
2	Primi	8,00
2	Orate	20,00
2	Caffè	2,00
<i>Totale</i>		39,00

Esempio 3

- Strutture nidificate:

Ricevute

Numero	Data	Qtà	Descrizione	Importo	Totale
1235	12/10/2017	3	Coperti	3,00	39,20
		2	Antipasti	6,20	
		3	Primi	12,00	
		2	Bistecche	18,00	
1247	13/10/2017	2	Coperti	2,00	39,00
		

- Ma i valori devono essere semplici, non relazioni!

Esempio 3

- Strutture nidificate:

Ricevute

Numero	Data	Qtà	Descrizione	Importo	Totale
1235	12/10/2017	3	Coperti	3,00	39,20
		2	Antipasti	6,20	
		3	Primi	12,00	
		2	Bistecche	18,00	
1247	13/10/2017	2	Coperti	2,00	39,00
		

- Ma i valori devono essere semplici, non relazioni!

Esempio 3

Ricevute

Numero	Data	Totale
1235	12/10/2017	39,20
1247	13/10/2017	39,00

Dettaglio

Numero	Qtà	Descrizione	Importo
1235	3	Coperti	3,00
1235	2	Antipasti	6,20
1235	3	Primi	12,00
1235	2	Bistecche	18,00
1247	2	Coperti	2,00
1247

Esempio 3

Esempio 3

- Abbiamo rappresentato davvero tutti gli aspetti delle ricevute?

Esempio 3

- Abbiamo rappresentato davvero tutti gli aspetti delle ricevute?
- Dipende da cosa ci interessa!

Esempio 3

- Abbiamo rappresentato davvero tutti gli aspetti delle ricevute?
 - Dipende da cosa ci interessa!
- Possono esistere linee ripetute in una ricevuta?

Esempio 3

- Abbiamo rappresentato davvero tutti gli aspetti delle ricevute?
 - Dipende da cosa ci interessa!
- Possono esistere linee ripetute in una ricevuta?
 - Se aggiungiamo altri piatti già presi, sì!

Esempio 3

- Abbiamo rappresentato davvero tutti gli aspetti delle ricevute?
 - Dipende da cosa ci interessa!
- Possono esistere linee ripetute in una ricevuta?
 - Se aggiungiamo altri piatti già presi, sì!
- L'ordine delle linee è rilevante?

Esempio 3

- Abbiamo rappresentato davvero tutti gli aspetti delle ricevute?
 - Dipende da cosa ci interessa!
- Possono esistere linee ripetute in una ricevuta?
 - Se aggiungiamo altri piatti già presi, sì!
- L'ordine delle linee è rilevante?
 - Prima gli antipasti, poi i primi, i secondi, ...

Esempio 3

- Abbiamo rappresentato davvero tutti gli aspetti delle ricevute?
 - Dipende da cosa ci interessa!
- Possono esistere linee ripetute in una ricevuta?
 - Se aggiungiamo altri piatti già presi, sì!
- L'ordine delle linee è rilevante?
 - Prima gli antipasti, poi i primi, i secondi, ...
- Sono possibili rappresentazioni diverse!

Esempio 3

Ricevute

Numero	Data	Totale
1235	12/10/2017	39,20
1247	13/10/2017	39,00

Dettaglio

Numero	Riga	Qtà	Descrizione	Importo
1235	1	3	Coperti	3,00
1235	2	2	Antipasti	6,20
1235	3	3	Primi	12,00
1235	4	2	Bistecche	18,00
1247	1	2	Coperti	2,00
1247

Informazione Incompleta

- Il modello relazionale impone ai dati una struttura rigida
 - Le informazioni sono rappresentate da n -uple il cui formato deve corrispondere esattamente agli schemi di relazione
- Ma i dati disponibili possono non corrispondere al formato previsto

Informazione Incompleta

Persone

Nome	Secondo Nome	Cognome
Franklin	Delano	Roosevelt
Winston		Churchill
Charles		De Gaulle
Josip		Stalin

Informazione Incompleta

- **Non** conviene **usare valori “particolari”** del **dominio** (0, stringa nulla, “99”, ...):
 - potrebbero non esistere valori “non utilizzati”
 - valori “non utilizzati” potrebbero diventare significativi
 - in fase di utilizzo (nei programmi) sarebbe necessario ogni volta tener conto del “significato” di questi valori
- Si usa un **valore distinto** aggiunto a **tutti i domini**
 - **valore nullo**: denota **l’assenza** di un valore del dominio
 - se i valori dell’attributo A appartengono al dominio D_A , $t[A]$ è un valore del dominio oppure il valore nullo **NULL**
- Si possono (e debbono) **imporre restrizioni** sulla presenza di valori nulli in una relazione

Tipi di valore nullo

- Almeno tre casi differenti:
 - valore **sconosciuto**
 - valore **inesistente**
 - valore **senza informazione**
- I DBMS non distinguono i tipi di valore nullo!

Troppi valori nulli

Studenti

Matricola	Cognome	Nome	Nascita
6554	Rossi	Mario	05/12/78
8765	Neri	Paolo	03/11/76
9283	Verdi	Luisa	12/11/79
NULL	Rossi	Maria	01/02/79

Esami

Studente	Voto	Corso
NULL	30	NULL
NULL	24	2
9283	28	1
6554	26	1

Corsi

Codice	Titolo	Docente
1	Analisi	Bruni
2	NULL	NULL
4	Chimica	Fino

Basi di dati “scorrette”

- Esistono istanze di basi di dati che, pur **sintatticamente corrette**, **non rappresentano informazioni possibili** per l'applicazione di interesse

Esami

Studente	Voto	Lode	Corso
276545	32		1
276545	30	e lode	2
787643	27	e lode	3
739430	24		4

Studenti

Matricola	Cognome	Nome
276545	Rossi	Mario
787643	Neri	Paolo
787643	Verdi	Luisa

Vincoli di Integrità

- Si devono associare alla base di dati delle **proprietà** che, se soddisfatte, **esprimono la sua “correttezza”** rispetto all'applicazione
- Queste proprietà si chiamano **vincoli di integrità**
 - permettono una descrizione più accurata della realtà
 - danno un contributo alla “qualità dei dati”
 - sono utili nella progettazione
 - sono usati dai DBMS nella esecuzione delle interrogazioni

Vincoli di Integrità

- I vincoli corrispondono a **proprietà** del mondo reale modellato dalla base di dati e interessano **tutte le istanze**
- I vincoli sono associati allo schema e si considerano **corrette** le sue istanze che soddisfano **tutti i vincoli**

Vincoli di Integrità

- Un vincolo di integrità è una **funzione Booleana** (**predicato**) che associa, a ogni istanza della base dati, il valore **vero** o **falso**
- Se il predicato associa **vero** allora la **proprietà è soddisfatta**
- Esistono due tipi di vincoli di integrità:
 - **intra-relazionali**
 - **inter-relazionali**

Vincoli Intrarelazionali

- Il suo soddisfacimento è definito rispetto a una **singola relazione** della base di dati
- Esempio:
 - **Vincolo di n -upla**: può essere valutato su ciascuna n -upla indipendentemente dalle altre
 - $(\text{Voto} = 30) \text{ OR NOT } (\text{Lode} = \text{"e lode"})$
 - **Vincolo di dominio**: vincolo di n -upla che coinvolge un solo attributo
 - $(\text{Voto} \geq 18) \text{ AND } (\text{Voto} \leq 30)$

Vincoli Interrelazionali

- Il suo soddisfacimento è definito rispetto a **più relazioni** della base di dati
- Esempio:
 - “Un numero di matricola può comparire nella relazione Esami solo se compare nella relazione Studenti”

Identificazione delle n -uple

Matricola	Cognome	Nome	Corso	Nascita
27655	Rossi	Mario	Ing. Inf.	05/12/78
78763	Rossi	Mario	Ing. Inf.	03/11/76
65432	Neri	Piero	Ing. Mecc.	12/11/79
87654	Neri	Mario	Ing. Inf.	03/11/76
67653	Rossi	Piero	Ing. Mecc.	05/12/78

- Non ci sono due n -uple con lo stesso valore sull'attributo Matricola
- Non ci sono due ennuple uguali su tutti e tre gli attributi Cognome, Nome e Nascita

Chiave

- Insieme di attributi che identificano **univocamente** le n -uple
- Formalmente:
 - Un insieme K di attributi è una **superchiave** per una relazione r se r non contiene due n -uple distinte t_1 e t_2 con $t_1[K] = t_2[K]$
 - K è una **chiave** per r se è una superchiave minimale di r (cioè non contiene un'altra superchiave)

Esempio

Matricola	Cognome	Nome	Corso	Nascita
27655	Rossi	Mario	Ing. Inf.	05/12/78
78763	Rossi	Mario	Ing. Inf.	03/11/76
65432	Neri	Piero	Ing. Mecc.	12/11/79
87654	Neri	Mario	Ing. Inf.	03/11/76
67653	Rossi	Piero	Ing. Mecc.	05/12/78

- Matricola è una chiave:
 - è una superchiave
 - contiene un solo attributo, e quindi è minimale

Esempio

Matricola	Cognome	Nome	Corso	Nascita
27655	Rossi	Mario	Ing. Inf.	05/12/78
78763	Rossi	Mario	Ing. Inf.	03/11/76
65432	Neri	Piero	Ing. Mecc.	12/11/79
87654	Neri	Mario	Ing. Inf.	03/11/76
67653	Rossi	Piero	Ing. Mecc.	05/12/78

- Cognome, Nome, Nascita è un'altra chiave:
 - è una superchiave
 - è minimale
- Cognome e Nome non distinguono le n -uple 1 e 2
- Cognome e Nascita non distinguono le n -uple 1 e 5
- Nome e Nascita non distinguono le n -uple 2 e 4

Esistenza delle chiavi

- Una relazione contiene n -uple tutte diverse tra loro
 - Ricordiamoci che è un insieme
- Ogni relazione ha come superchiave l'insieme degli attributi su cui è definita
 - Quindi ha (almeno) una chiave

Importanza delle chiavi

- L'esistenza delle chiavi garantisce l'accessibilità a ciascun dato della base di dati
- Le chiavi permettono di correlare i dati in relazioni diverse
- **Le chiavi definite su schemi di relazioni sono un tipo di vincolo di integrità, detto vincolo di chiave**

Chiave Primaria

- Come ci comportiamo se una relazione ha più di una chiave?
- Scegliamo noi una delle chiavi e la dichiariamo **chiave primaria** (*primary key*)
 - Le altre chiavi sono chiamate **chiavi candidate**.
- Ogni relazione ha una chiave primaria
 - Gli attributi di una chiave primaria sono solitamente “sottolineati”

Chiavi e valori nulli

- In presenza di valori nulli, i valori della chiave non permettono
 - di identificare le n -uple
 - di realizzare facilmente i riferimenti da altre relazioni

Matricola	Cognome	Nome	Corso	Nascita
NULL	NULL	Mario	Ing. Inf.	05/12/78
78763	Rossi	Mario	Ing. Inf.	03/11/76
65432	Neri	Piero	Ing. Mecc.	12/11/79
87654	Neri	Mario	Ing. Inf.	NULL
NULL	Rossi	Piero	NULL	05/12/78

- La presenza di valori nulli nelle chiavi **deve essere limitata**

Dipendenze Funzionali

- I vincoli di chiave sono particolari tipi di vincoli, che fanno parte di una categoria più vasta: le **dipendenze funzionali**
- Formalmente:
 - Dati due insiemi di attributi X e Y sulla relazione R , si dice che X **determina** Y , e si scrive $X \rightarrow Y$, se e solo se:
 - date due n -uple distinte t_1 e t_2 , se $t_1[X] = t_2[X]$ allora $t_1[Y] = t_2[Y]$
- Le dipendenze funzionali possono essere usate per garantire opportune proprietà di una base di dati

Integrità Referenziale

- Informazioni in relazioni diverse posso essere correlate attraverso valori comuni
 - In particolare, i valori delle chiavi
- Data una relazione, un insieme di attributi della relazione che corrispondono a una chiave primaria di un'altra relazione è chiamato **chiave esterna** (*foreign key*)
 - Simili a un “puntatore logico”

Esempio

Infrazioni

<u>Codice</u>	Data	Vigile	Prov	Numero
34321	1/2/15	3987	MI	39548K
53524	4/3/15	3295	TO	E39548
64521	5/4/16	3295	PR	839548
73321	5/2/18	9345	PR	839548

Vigili

<u>Matricola</u>	Cognome	Nome
3987	Rossi	Luca
3295	Neri	Piero
9345	Neri	Mario
7543	Mori	Gino

Vincolo Integrità Referenziale

- Un **vincolo di integrità referenziale** fra gli attributi X di una relazione R_1 (chiave esterna di R_1) e un'altra relazione R_2 impone ai valori su X in R_1 di comparire come valori della chiave primaria di R_2
- **NOTA BENE:** in questo caso **l'ordine** degli attributi tra cui è stabilito il vincolo è **significativo**

Integrità Referenziale e Valori Nulli

- In presenza di valori nulli i vincoli possono essere resi meno restrittivi
- Il vincolo non è fra ogni valore degli attributi X di una relazione R_1 e la chiave primaria della relazione R_2 , ma tra i valori di X diversi da **NULL** e la chiave primaria di R_2

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
53524	Neri	XYZ
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
XYZ	2001	24	120
BOH	2001	24	150

Operazioni di Aggiornamento

- Operazioni di **inserimento**:
 - violazione dei vincoli intra-relazionali
 - violazione dell'integrità referenziale
- Operazioni di **cancellazione**:
 - violazione dell'integrità referenziale
- Operazioni di **modifica**:
 - $\text{modifica} = \text{cancellazione} + \text{inserimento}$

Reazione alla Violazione di Vincoli

- Cosa succede quando si tenta di compiere **un'operazione che viola un vincolo**, ad esempio si cerca di inserire nella base di dati un valore non consentito per quell'attributo
- Sono **possibili meccanismi** per il supporto alla **gestione delle violazioni** ("azioni compensative")

Azioni Compensative

- Esempio:
 - Viene eliminata una n -upla, causando una violazione
- Comportamento “standard”:
 - Rifiuto dell'operazione
- Azioni compensative:
 - Eliminazione in cascata
 - Introduzione di valori nulli

Esempio

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
53524	Neri	XYZ
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
XYZ	2001	24	120
BOH	2001	24	150

Esempio

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
53524	Neri	XYZ
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
BOH	2001	24	150

Esempio

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
BOH	2001	24	150

Esempio

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
BOH	2001	24	150

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
53524	Neri	XYZ
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
XYZ	2001	24	120
BOH	2001	24	150

Esempio

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
BOH	2001	24	150

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
53524	Neri	XYZ
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
BOH	2001	24	150

Esempio

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
BOH	2001	24	150

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
53524	Neri	NULL
64521	Verdi	NULL
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	2000	36	200
BOH	2001	24	150

Esercizio

- Date le seguenti tabelle:
- AUTORE(Nome, Cognome, DataNascita, Nazionalità)
- LIBRO(Titolo, NomeAutore, CognomeAutore, Lingua)
- Esiste un vincolo di integrità referenziale tra di esse?

Esercizio

- Supponendo di avere specificato una politica di eliminazione in cascata sulle modifiche e cancellazioni, spiegare qual è l'effetto dell'esecuzione dei seguenti aggiornamenti:
 - Cancella da AUTORE tutte le righe dove Cognome = 'Rossi'
 - Modifica LIBRO scrivendo NomeAutore = 'Umberto' in tutte le righe dove CognomeAutore = 'Eco'

Soluzione

- Il primo aggiornamento cancella dalla tabella AUTORE tutte le tuple con Cognome = 'Rossi'
- A causa della politica a cascata, anche tutte le n -uple di LIBRO con CognomeAutore = 'Rossi' vengono eliminate.
- Il secondo aggiornamento causa una violazione, a meno che la tabella AUTORE contenga già la n -upla "Umberto Eco"