

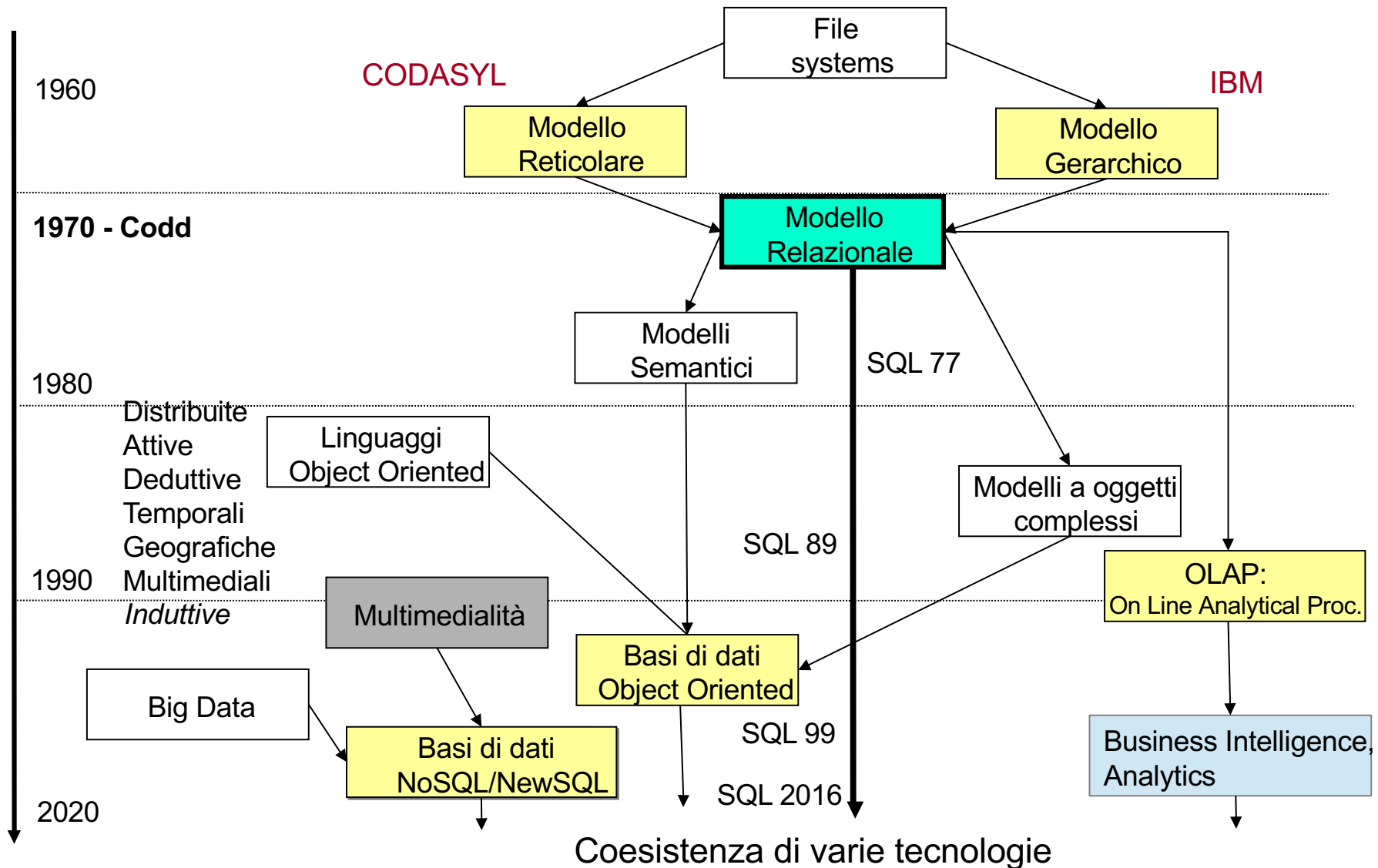
Basi di Dati

Il modello relazionale

Contenuti - Roadmap

Lab (progettazione)	Corso di Teoria	Lab (SQL)
<ul style="list-style-type: none">• Metodologie e modello Entità Associazioni• Progettazione concettuale e logica	<ul style="list-style-type: none">• Modello Relazionale• Algebra relazionale• Ottimizzazione logica• Calcolo relazionale• La normalizzazione• Metodi di accesso e indici• Gestione della concorrenza• Gestione del ripristino	<ul style="list-style-type: none">• Linguaggio SQL

Evoluzione delle basi di dati



I modelli logici dei dati

- Tre modelli logici tradizionali
 - gerarchico
 - reticolare
 - relazionale
- Più recenti
 - a oggetti (poco diffuso)
 - basato su XML ("complementare" a quello relazionale)
 - NoSQL (document-based, colonnari, graph-based, RDF, ...)

Modelli logici, caratteristiche

- 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

Il modello relazionale

- Proposto da E. F. Codd nel 1970 per favorire l'indipendenza dei dati
- Disponibile in DBMS commerciali 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 una naturale rappresentazione per mezzo di tabelle

Relazione: tre accezioni

- **relazione matematica**: come nella teoria degli insiemi
- **relazione** secondo il modello relazionale dei dati
- **relazione** (dall'inglese **relationship**) che rappresenta una classe di fatti, nel modello Entity-Relationship; tradotto anche con **associazione**

Relazione matematica, esempio

- $D_1 = \{a, b\}$
- $D_2 = \{x, y, z\}$
- prodotto cartesiano $D_1 \times D_2$

a	x
a	y
a	z
b	x
b	y
b	z

- una relazione $r \subseteq D_1 \times D_2$
- $r = \{(a, x), (a, z), (b, y)\}$

a	x
a	z
b	y

Relazione matematica

- D_1, \dots, D_n (n insiemi anche non distinti)
- **prodotto cartesiano** $D_1 \times \dots \times D_n$:
 - l'insieme di tutte le n -uple (d_1, \dots, d_n) tali che $d_1 \in D_1, \dots, d_n \in D_n$
- **relazione matematica** su D_1, \dots, D_n :
 - un sottoinsieme di $D_1 \times \dots \times D_n$.
- D_1, \dots, D_n sono i **domini** della relazione

Relazione matematica, proprietà

- una relazione matematica è un insieme di n-uple ordinate:
 - (d_1, \dots, d_n) tali che $d_1 \in D_1, \dots, d_n \in D_n$
- una relazione è un insieme, quindi:
 - non c'è ordinamento fra le n-uple
 - le n-uple sono distinte
 - ciascuna n-upla è ordinata: l'i-esimo valore proviene dall'i-esimo dominio

Relazione matematica, esempio

Partite* \subseteq *string* \times *string* \times *int* \times *int

Torino	Lazio	3	1
Lazio	Milan	2	0
Juve	Roma	0	2
Roma	Milan	0	1

- Ciascuno dei domini string e int ha due ruoli diversi, distinguibili attraverso la posizione:
 - La struttura è **posizionale**

Struttura non posizionale

- Nel modello relazionale usiamo un concetto di relazione diverso da quello matematico
- Una relazione è ancora un insieme di n-uple (chiamate **tuple**), ma...
- Invece di basarci sulla posizione, a ciascun dominio associamo un nome unico nella tabella (**attributo**) che ne descrive il "ruolo"

Casa	Ospite	RetiCasa	RetiOspite
Juve	Lazio	3	1
Lazio	Milan	2	0
Juve	Roma	0	2
Roma	Milan	0	1

Struttura non posizionale, 2

Casa	Ospite	RetiCasa	RetiOspite
Torino	Lazio	3	1
Lazio	Milan	2	0
Juve	Roma	0	2
Roma	Milan	0	1

Tabelle e relazioni

- In una tabella che rappresenta una relazione
 - l'ordinamento tra le righe è irrilevante
 - l'ordinamento tra le colonne è irrilevante
- Una tabella rappresenta una relazione se
 - le righe sono diverse fra loro
 - le intestazioni delle colonne sono diverse tra loro
 - ogni colonna ha valori omogenei

Il modello è basato su valori

- Vedremo quando parleremo di normalizzazione perché occorre memorizzare i dati in relazioni differenti
- I riferimenti fra dati in relazioni diverse sono rappresentati per mezzo di valori dei domini che compaiono nelle tuple
- Altri modelli (sia quelli "storici", reticolare e gerarchico, sia quello a oggetti) prevedono riferimenti espliciti, gestiti dal sistema

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1978
8765	Neri	Paolo	03/11/1976
9283	Verdi	Luisa	12/11/1979
3456	Rossi	Maria	01/02/1978

esami

Studente	Voto	Corso
3456	30	04
3456	24	02
9283	28	01
6554	26	01

corsi

Codice	Titolo	Docente
01	Analisi	Mario
02	Chimica	Bruni
04	Chimica	Verdi

Struttura basata su valori: vantaggi

- Indipendenza dalle strutture fisiche, che possono cambiare dinamicamente
- Si rappresenta solo ciò che è rilevante dal punto di vista dell'applicazione
- L'utente finale vede gli stessi dati dei programmatori
- I dati sono portabili più facilmente da un sistema a un altro

Definizioni

- Schema di relazione:

un nome R con un insieme di attributi A_1, \dots, A_n :

$$R(A_1, \dots, A_n)$$

- Ad esempio:

Studenti(Matricola, Nome, Cognome, DataNascita)

Definizioni

- Schema di base di dati:

insieme di schemi di relazione:

$$R = \{R_1(X_1), \dots, R_k(X_k)\}$$

- Ad esempio:

{Studenti(Matricola, Nome, Cognome, DataNascita),
Esami(Studiante, Voto, Corso),
Corsi(Codice, Titolo, Docente)}

Definizioni

- Una **tupla** su un insieme di attributi (A_1, \dots, A_n) associa a ciascun attributo A_i un valore del dominio di A_i
- $t[A_i]$ denota il valore della tupla t sull'attributo A_i
- Ad es. sulla tupla $t=('3456', 30, '04')$ $t[\text{Voto}] = 30$

Definizioni

- (Istanza di) relazione
su uno schema di relazione $R(X)$ dove X sono attributi:
insieme r di tuple su X
- (Istanza di) base di dati
su uno schema di base di dati $\{R_1(X_1), \dots, R_h(X_h)\}$:
insieme di relazioni $\{r_1, \dots, r_h\}$ (con r_i relazione su R_i)

Esempio

<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 (dati)

DA FILIPPO VIA ROMA 2, ROMA		
RICEVUTA FISCALE 1235 DEL 12/10/2017		
3	Coperti	3,00
2	Antipasti	6,20
3	Primi	12,00
2	Bistecche	18,00
TOTALE		39,20

DA FILIPPO VIA ROMA 2, ROMA		
RICEVUTA FISCALE 1240 DEL 13/10/2017		
2	Coperti	2,00
2	Antipasti	7,00
2	Primi	8,00
2	Orate	20,00
2	Caffè	2,00
TOTALE		39,00

Esempio (dati nidificati)

<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>		<i>39,20</i>

<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>		<i>39,00</i>

Strutture nidificate

Ricevute

Numero	Data	Piatti			Totale
1235	12/10/2017	Qtà	Descrizione	Importo	39,20
		3	Coperti	3,00	
		2	Antipasti	6,20	
		3	Primi	12,00	
		2	Bistecche	18,00	
1240	13/10/2017	Qtà	Descrizione	Importo	39,00
		2	Coperti	2,00	
		

- Ma nel modello relazionale i valori devono essere semplici, non relazioni! (Questa restrizione si chiama, per motivi storici, **Prima forma normale**)

Relazioni che rappresentano strutture nidificate

Ricevute

Numero	Data	Totale
1235	12/10/2017	39,20
1240	13/10/2017	39,00

Dettaglio

Ricevuta	Qtà	Descrizione	Importo
1235	3	Coperti	3,00
1235	2	Antipasti	6,20
1235	3	Primi	12,00
1235	2	Bistecche	18,00
1240	2	Coperti	2,00
...

Strutture nidificate, riflessione

- Abbiamo rappresentato veramente tutti gli aspetti delle ricevute?
- Dipende da che cosa ci interessa!
 - Possono esistere linee ripetute in una ricevuta?
 - L'ordine delle righe è rilevante?
- Sono possibili rappresentazioni diverse

Rappresentazione alternativa per strutture nidificate

Ricevute

Numero	Data	Totale
1235	12/10/2017	39,20
1240	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
1240	1	2	Coperti	2,00
...

Informazione incompleta

- Il modello relazionale impone ai dati una struttura rigida:
 - le informazioni sono rappresentate per mezzo di tuple
 - solo alcuni formati di tuple sono ammessi: quelli che corrispondono agli schemi di relazione
- I dati disponibili possono non corrispondere al formato previsto

Informazione incompleta: motivazioni

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

Informazione incompleta: soluzioni?

- Non conviene usare valori del dominio (0, stringa nulla, “99”, ...):
 - potrebbero non esistere valori “non utilizzati”
 - valori “non utilizzati” potrebbero diventare in futuro significativi
 - in fase di utilizzo (nei programmi) sarebbe necessario ogni volta tenere conto del “significato” di questi valori

Informazione incompleta nel modello relazionale

- Il modello relazionale adotta una tecnica rudimentale ma efficace, un valore “extra”:
 - **valore nullo**: denota l’assenza di un valore del dominio (ma non è un valore del dominio)
- $t[A]$, per ogni attributo A , è un valore del dominio $\text{dom}(A)$ oppure il valore nullo (che indichiamo qui con **NULL**)
- Si possono (e devono) imporre restrizioni sulla presenza di valori nulli

Tipi di valore nullo

- (Almeno) tre casi differenti
 - valore **sconosciuto** (es. numero telefono)
 - valore **inesistente** (es. numero permesso di soggiorno per un cittadino UE)
 - valore **senza informazione** (es. numero passaporto)
- I DBMS non distinguono tra questi tre tipi di valore nullo

Alcuni valori nulli non ammissibili

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1995
9283	Verdi	Luisa	12/11/1993
<i>NULL</i>	Rossi	Maria	01/02/1994

esami

Studente	Voto	Corso
<i>NULL</i>	30	<i>NULL</i>
<i>NULL</i>	24	02
9283	28	01

corsi

Codice	Titolo	Docente
01	Analisi	Mario
02	<i>NULL</i>	<i>NULL</i>
04	Chimica	Verdi

Vincoli di integrità

- Esistono istanze di basi di dati in cui non è sufficiente rispettare i domini degli attributi per rappresentare informazioni ammissibili per l'applicazione di interesse

Una base di dati "scorretta"

Esami	Studente	Voto	Lode	Corso
	276545	32	false	01
	276545	30	true	02
	787643	27	true	03
	739430	24	false	04

Studenti	Matricola	Cognome	Nome
	276545	Rossi	Mario
	787643	Neri	Piero
	787643	Bianchi	Luca

Vincolo di integrità

- Proprietà che deve essere soddisfatta dalle istanze che rappresentano informazioni corrette per l'applicazione
- Un vincolo è un un **predicato** logico che associa a ogni istanza un valore di verità
- A ogni modifica dell'istanza della base di dati (inserimento, cancellazione, aggiornamento di tuple), il DBMS verifica tutti i vincoli
- Se un vincolo non risulta **vero**, il DBMS rifiuta la modifica

Vincoli di integrità, perché?

- Descrizione più accurata della realtà
- Contributo alla “qualità dei dati”
- Utili nella progettazione (lo vedremo)
- Usati dai DBMS nell'esecuzione delle interrogazioni

Vincoli di integrità, nota

- alcuni tipi di vincoli (ma non tutti) sono supportati dai DBMS:
 - possiamo quindi specificare questi vincoli nella nostra base di dati e il DBMS ne impedirà la violazione
- per i vincoli non supportati dal DBMS, la responsabilità della verifica è dell'utente o del programmatore

Tipi di vincoli

- vincoli intrarelazionali
 - vincoli su valori (o di dominio)
 - vincoli di tupla
- vincoli interrelazionali

Esami	Studente	Voto	Lode	Corso
	276545	32	False	01
	276545	30	True	02
	787643	27	True	03
	739430	24	False	04

Studenti	Matricola	Cognome	Nome
	276545	Rossi	Mario
	787643	Neri	Piero
	787643	Bianchi	Luca

Vincoli di tupla

- Esprimono condizioni sui valori di ciascuna tupla, indipendentemente dalle altre tuple
- Caso particolare:
 - Vincoli di dominio: coinvolgono un solo attributo

Sintassi ed esempi

- Una possibile sintassi:
 - espressione booleana di atomi che confrontano valori di attributo o espressioni aritmetiche su di essi

$\text{Voto} \geq 18 \text{ AND } \text{Voto} \leq 30$

$\text{NOT } (\text{Lode} = \text{True}) \text{ OR } \text{Voto} = 30$

Se... allora...

- Si può rappresentare l'implicazione con gli operatori booleani
- "Se piove prendo l'ombrello"
 - Se non piove posso prenderlo o non prenderlo!
 - NOT piove OR prendo l'ombrello

Vincoli di tupla, altro esempio

Stipendi	Impiegato	Lordo	Ritenute	Netto
	Rossi	55.000	12.500	42.500
	Neri	45.000	10.000	35.000
	Bruni	47.000	11.000	36.000

$$\text{Lordo} = \text{Ritenute} + \text{Netto}$$

Vincoli di tupla, violazione

Stipendi	Impiegato	Lordo	Ritenute	Netto
	Rossi	55.000	12.500	42.500
	Neri	45.000	10.000	35.000
	Bruni	50.000	11.000	36.000

$$\text{Lordo} = \text{Ritenute} + \text{Netto}$$

Vincoli di chiave

- **Superchiave**: insieme di attributi usato per identificare univocamente le tuple di una relazione
- Più formalmente, un insieme di attributi K è detto **superchiave** di una relazione r se e solo se r non può contenere due tuple che hanno valori uguali su K , cioè se r non può contenere due tuple distinte t_1 e t_2 con $t_1[K] = t_2[K]$

Vincoli di chiave

- Quali sono le superchiavi?

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- {Matricola} è una superchiave

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- {Matricola, CorsoDiLaurea} è una superchiave

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- {Matricola} è una superchiave

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- {Matricola, CF, Cognome, Nome, DataDiNascita, CorsoDiLaurea} è una superchiave

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- {Nome, CorsoDiLaurea} non è una superchiave

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- {Nome, DataDiNascita} sembra una superchiave in questa particolare istanza di Studenti, ma il concetto di superchiave (e più in generale di vincolo) riguarda tutte le *possibili* tuple di una relazione e non soltanto quelle esistenti in un dato momento

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- Dato che una relazione è un insieme di tuple e che in un insieme non possono esistere due elementi uguali, in una relazione non possono esistere due tuple identiche: quindi per qualunque relazione *l'intero insieme degli attributi è sempre banalmente una superchiave*
- Quindi è utile restringere il concetto di superchiave...

Vincoli di chiave

- K è una **chiave (candidata)** di r se e solo se è una superchiave **minimale** di r
Minimale = non contiene una superchiave, cioè se si toglie un attributo da K , K non è più superchiave
- Di conseguenza, ogni chiave è una superchiave
- L'insieme di tutti gli attributi di una relazione, pur essendo una superchiave, non è detto che sia una chiave, infatti potrebbe essere una superchiave non minimale

Vincoli di chiave

- Quali sono le chiavi?

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- {Matricola} è una chiave

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- {Matricola, CorsoDiLaurea} è una superchiave, ma non è una chiave

Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- {CF} è una chiave

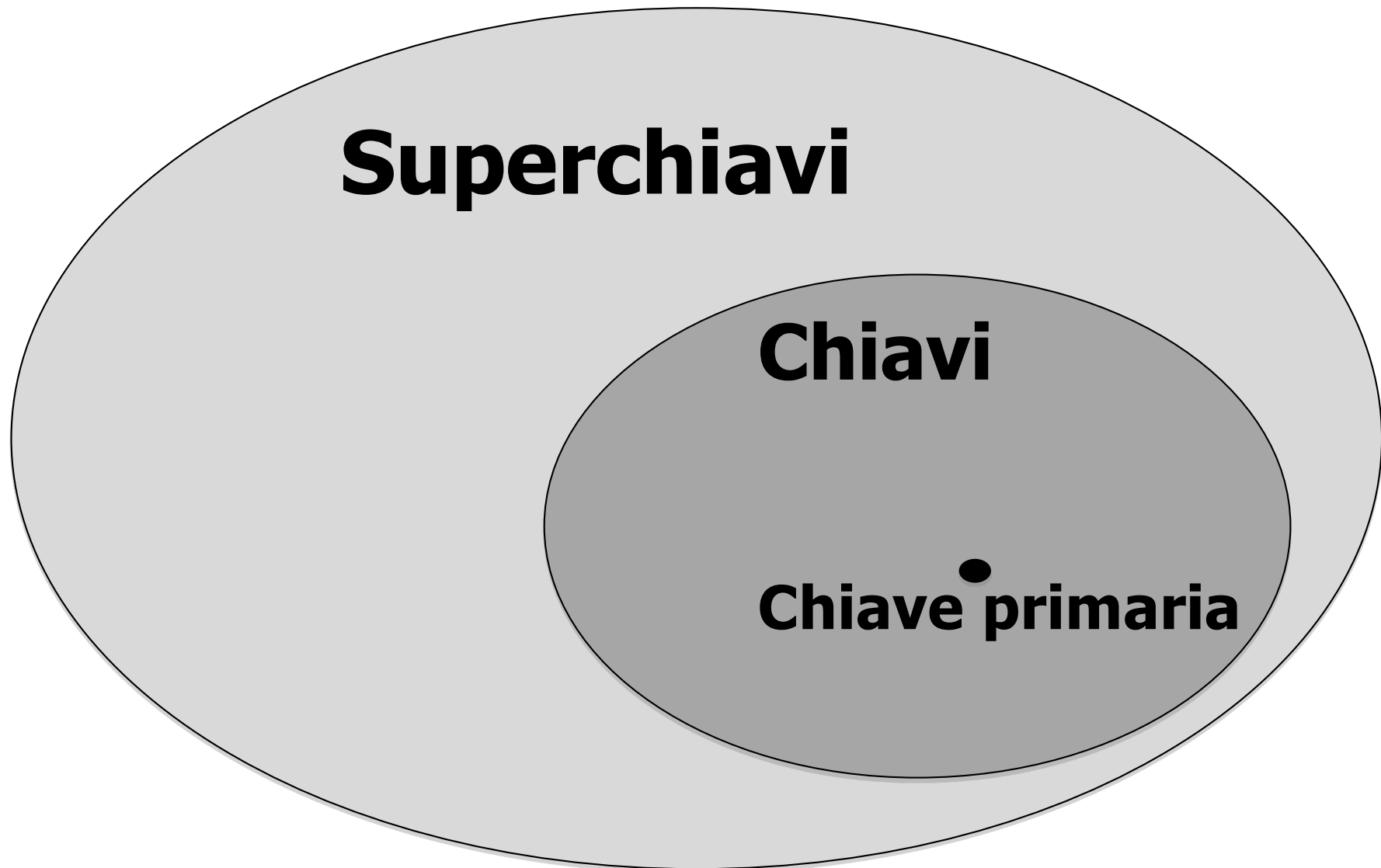
Studenti

Matricola	CF	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
4328	RSSL...	Rossi	Laura	29/04/79	Informatica
6328	RSSD...	Rossi	Dario	29/04/79	Informatica
4766	RSSC...	Rossi	Chiara	01/05/81	Fisica
4856	NRIL...	Neri	Luca	01/05/81	Economia
5536	NRIL...	Neri	Luca	05/03/78	Economia

Vincoli di chiave

- Una **chiave primaria** di r è una particolare chiave di r scelta dal progettista del DB come modo preferito per identificare univocamente le tuple di r
- Ogni relazione ha una e una sola chiave primaria, mentre può avere più di una chiave

Vincoli di chiave



Víncoli di chiave

- N.B.: Il concetto di chiave è relativo alla relazione: uno stesso insieme di attributi potrebbe essere chiave di una relazione, ma non esserlo di un'altra
- Per es., l'attributo Matricola è chiave della relazione Studenti, ma non della relazione Esami

Esempi di relazioni

Studenti

Matricola	Nome	Cognome
654321	Mario	Rossi
321654	Franca	Verde
123456	Marta	Bianco
456123	Livia	Nero
135246	Marta	Bianco

Esami

Corsi

Codice	Nome
0010	Inglese
0021	Psicologia clinica
0001	Informatica
0121	Diritto

Studente	Corso	Voto	Lode	Data
123456	0010	19	No	1/2/2010
654321	0021	30	No	3/2/2010
456123	0001	24	No	2/2/2010
321654	0121	30	Sì	7/2/2010
123456	0001	21	No	9/2/2010
135246	0010	25	No	1/2/2010

Importanza delle chiavi

- L'esistenza delle chiavi garantisce la possibilità di accedere a ciascun dato della base di dati
- Quindi ogni relazione deve possedere una chiave primaria
- Le chiavi permettono di correlare i dati in relazioni diverse: il modello relazionale è basato su valori
- I DBMS permettono di creare tabelle senza chiave primaria ma queste non rappresenterebbero informazioni «sensate»

Chiavi e valori nulli

- In presenza di valori nulli, i valori della chiave non permettono
 - di identificare le tuple
 - di realizzare facilmente riferimenti tra relazioni

Chiave e valori nulli

- Tupla 1: ha la chiave incompleta ({Matricola})

Matricola	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
NULL	Rossi	Dario	NULL	Informatica
4766	Rossi	Luca	01/05/81	Fisica
4856	Neri	Luca	NULL	NULL
NULL	Neri	Luca	05/03/78	Economia

Chiave e valori nulli

- Tuple 3 e 4: è sicuro che non siano lo stesso individuo?

Matricola	Cognome	Nome	DataDi Nascita	CorsoDi Laurea
NULL	Rossi	Dario	NULL	Informatica
4766	Rossi	Luca	01/05/81	Fisica
4856	Neri	Luca	NULL	NULL
NULL	Neri	Luca	05/03/78	Economia

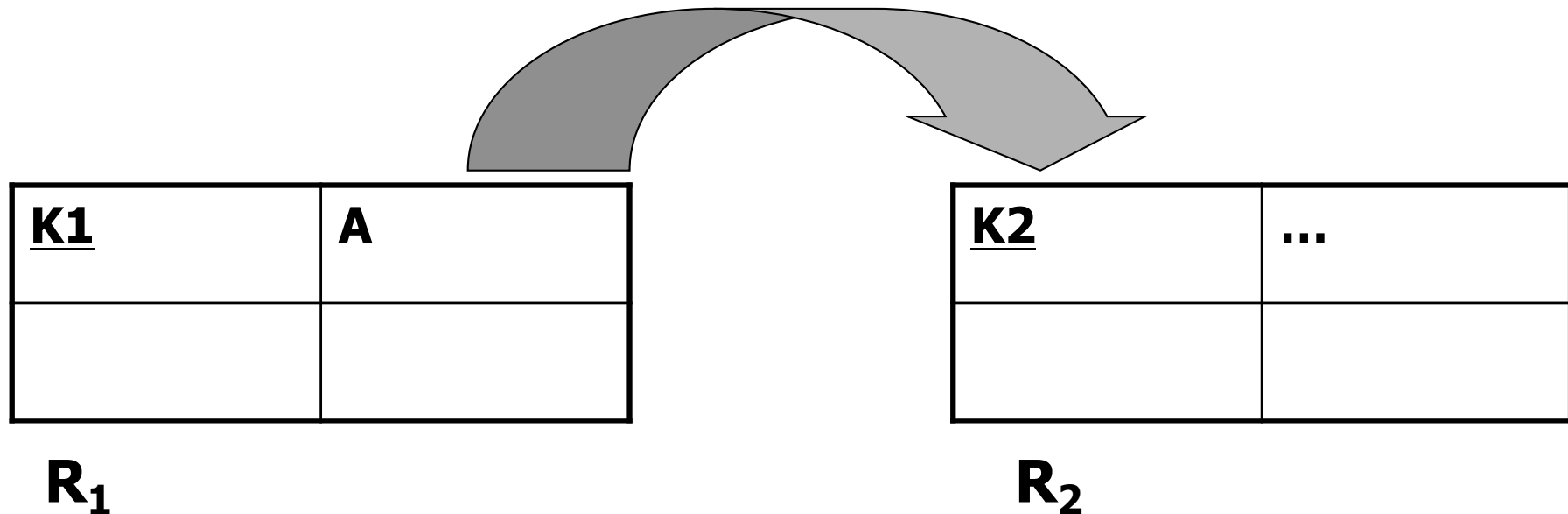
Chiave e valori nulli

- *Chiave primaria*: la chiave primaria non può assumere valori nulli
 - Graficamente viene rappresentata mediante sottolineatura
 - Usata per stabilire corrispondenza tra relazioni
 - Se nessun attributo della relazione può assumere il ruolo di chiave primaria, se ne definisce uno aggiuntivo a tale scopo
 - Es.
 - Studenti(Matricola,Cognome,Nome,DataNascita)
 - Corsi(Codice,Titolo,Docente)
 - Esami(Studente,Corso,Data,Voto,Lode)

Vincoli di integrità referenziale

- *Vincoli di integrità referenziale* sono detti anche *vincoli di foreign key* o *vincoli di chiave esterna*
- Garantiscono la correttezza dei riferimenti tra tabelle
- Un vincolo di integrità referenziale fra un insieme di attributi X di una relazione R_1 e una relazione R_2 è **soddisfatto** se e solo se i valori su X di *ciascuna tupla di R_1* (eccetto NULL) compaiono come *valori della chiave (di solito primaria) di R_2* . In altre parole: R_1 non può contenere valori per X che non esistono in R_2 (eccetto NULL)

Vincoli di integrità referenziale



Vincoli di integrità referenziale

- Esempio:

Studenti(Matricola, Cognome, Nome, DataNascita)

Corsi(Codice, Titolo, Docente)

Esami(Studente, Corso, Data, Voto, Lode)



- Primo vincolo di integrità referenziale
 - Indicato come freccia che va dall'attributo Studente all'attributo Matricola
 - oppure scriviamo «Esami(Studente) referencia Studenti(Matricola)»
 - I valori dell'attributo Studente nella relazione Esami devono comparire come valori di Matricola nella relazione Studenti
 - Secondo vincolo di integrità referenziale
 - Indicato come freccia che va dall'attributo Corso all'attributo Codice
 - oppure scriviamo «Esami(Corso) referencia Corsi(Codice)»
 - I valori dell'attributo Corso nella relazione Esami devono comparire come valori di Codice nella relazione Corsi
 - Notare che ogni vincolo ha un «verso», cioè percorrendolo nel senso opposto cambia significato
-

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

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

Auto

<u>Prov</u>	<u>Numero</u>	Cognome	Nome
MI	39548K	Rossi	Mario
TO	E39548	Rossi	Mario
PR	839548	Neri	Luca

Violazione di vincolo di integrità referenziale

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

Auto

<u>Prov</u>	<u>Numero</u>	Cognome	Nome
MI	E39548	Rossi	Mario
TO	F34268	Rossi	Mario
PR	839548	Neri	Luca

Quali sono i vincoli corretti?

- 1) Infrazioni(Prov) referencia Auto(Prov), Infrazioni(Numero) referencia Auto(Numero)
 - 2) Infrazioni(Prov,Numero) referencia Auto(Prov,Numero)?
-

Vincoli di integrità referenziale: commenti

- Giocano un ruolo **fondamentale** nel concetto di “**modello basato su valori**”
- Quindi, quando si scrive uno schema, è obbligatorio rappresentare anche tali vincoli
- In presenza di valori nulli i vincoli possono essere resi meno restrittivi
- Sono possibili meccanismi per il supporto alla loro gestione ("azioni" compensative a seguito di violazioni)



Es



Integrità referenziale e valori nulli

Impiegati

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

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	01/2017	36	200
XYZ	07/2016	24	120
BOH	09/2016	24	150

Azioni compensative

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

Azioni compensative: eliminazione in cascata

Impiegati

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

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	01/2017	36	200
XYZ	07/2016	24	120
BOH	09/2016	24	150

Vincolo di integrità referenziale:

Impiegati(Progetto) referencia Progetti(Codice)

Azioni compensative: eliminazione in cascata

Impiegati

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

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	01/2017	36	200
XYZ	07/2016	24	120
BOH	09/2016	24	150

Vincolo di integrità referenziale:

Impiegati(Progetto) referencia Progetti(Codice)

Azioni compensative: eliminazione in cascata

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	01/2017	36	200
XYZ	07/2016	24	120
BOH	09/2016	24	150

Vincolo di integrità referenziale:

Impiegati(Progetto) referencia Progetti(Codice)

Azioni compensative: introduzione di valori nulli

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	01/2017	36	200
XYZ	07/2016	24	120
BOH	09/2016	24	150

Vincolo di integrità referenziale:

Impiegati(Progetto) referencia Progetti(Codice)

Azioni compensative: introduzione di valori nulli

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	01/2017	36	200
XYZ	07/2016	24	120
BOH	09/2016	24	150

Vincolo di integrità referenziale:

Impiegati(Progetto) referencia Progetti(Codice)

Azioni compensative: introduzione di valori nulli

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	01/2017	36	200
XYZ	07/2016	24	120
BOH	09/2016	24	150

Vincolo di integrità referenziale:

Impiegati(Progetto) referencia Progetti(Codice)