

Lezione XXX – Ricapitolazione dei concetti base

Prof.ssa Maria De Marsico
demarsico@di.uniroma1.it



SAPIENZA
UNIVERSITÀ DI ROMA



- Nonostante l'apparente complessità di alcuni passaggi, TUTTO il corso di Basi di Dati – Modulo 1 si base su un numero di concetti di base che si può contare sulle dita di una mano!



- Proviamo ad andare oltre e alla base dei concetti.
- A cosa serve una base di dati? A memorizzare **informazioni** su **oggetti** di interesse per una certa applicazione.
- Ad ogni tipo di oggetto corrisponde uno **schema** e un gruppo di oggetti dello stesso tipo costituisce una **istanza**

- **Schema** = insieme di attributi con una etichetta e un tipo
- In pratica = le informazioni rilevanti **per una certa applicazione** che riguardano un **certo tipo di oggetto**

- ~~Esempio: oggetto = automobile~~

- schema1 = (marca, modello, cilindrata)
- schema2 = (marca, modello, cilindrata, colore)

- Per alcune applicazioni il colore è rilevante, per altre no, quindi si userà lo schema 1 o il 2 di conseguenza

- **Istanza**= insieme di tuple; ognuna assegna un valore **ad ognuno** degli attributi dello schema
- In pratica = le informazioni sugli oggetti rilevanti **per una certa applicazione**

- Esempio: oggetto = automobile

- Istanza di automobile (schema1) =

FIAT	Tipo	1000
FIAT	Panda	600
Citroen	Picasso	1200

- Istanza di automobile (schema2) =

FIAT	Tipo	1000	giallo
FIAT	Panda	600	rosso
Citroen	Picasso	1200	blu

- In testa alle colonne dell'istanza troviamo le etichette degli attributi
- Possiamo avere istanze vuote (nessuna tupla)

- Esempio: oggetto = automobile
- Istanza di automobile (schema1) =

MARCA	MODELLO	CILINDRATA
FIAT	Tipo	1000
FIAT	Panda	600
Citroen	Picasso	1200

- Istanza di automobile (schema2) =

MARCA	MODELLO	CILINDRATA	COLORE
FIAT	Tipo	1000	giallo
FIAT	Panda	600	rosso
Citroen	Picasso	1200	blu



- Le dipendenze funzionali **si definiscono sugli schemi** (ancora prima di inserire le istanze ...) e **vengono soddisfatte** dalle istanze
- Una dipendenza funzionale lega i valori di due insieme di attributi ...
- Complicato? No, il legame non è «matematico» ma discende dalla realtà che stiamo modellando, cioè viene **individuato** da come sono configurati gli oggetti nell'applicazione



- Esempio:
- Due marche **non producono** automobili dello stesso modello (almeno per ora e nella realtà delle applicazioni attuali), quindi in una istanza **se** trovo due tuple con lo stesso modello **allora** devo anche avere la stessa marca.
- Come esprimo questo fatto? Definendo sullo schema la dipendenza
- modello → marca (la freccia si legge «determina»)

- Quando una istanza soddisfa una dipendenza?
- Quando non viola il vincolo espresso dalla dipendenza
- Esempio: abbiamo individuato la dipendenza modello \rightarrow marca

- Istanza1

MARCA	MODELLO	CILINDRATA	COLORE
FIAT	Tipo	1000	giallo
FIAT	Panda	600	rosso
Citroen	Picasso	1200	blu

OK

- Istanza2

MARCA	MODELLO	CILINDRATA	COLORE
FIAT	Tipo	1000	giallo
Citroen	Tipo	600	rosso
Citroen	Picasso	1200	blu

NO

Istanze legali



SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Una istanza costruita su uno schema è **legale** se soddisfa **TUTTE** le dipendenze funzionali definite sullo schema stesso
- Una stessa istanza può essere legale rispetto ad un certo insieme di dipendenze e non esserlo rispetto ad un altro
- Esempio: se abbiamo anche che (modello, cilindrata) → colore vuol dire che nella nostra realtà un certo modello con una certa cilindrata viene prodotto sempre con lo stesso colore, e quindi l'istanza

MARCA	MODELLO	CILINDRATA	COLORE
FIAT	Tipo	1000	giallo
FIAT	Panda	600	rosso
Citroen	Picasso	1200	blu
FIAT	Tipo	1000	verde

- soddisfa la dipendenza modello → marca ma NON (modello, cilindrata) → colore
- (abbiamo due tuple con uguali modello e cilindrata, ma diverso colore) quindi non è legale



- Una istanza VUOTA è SEMPRE legale (non possiamo violare nessuna dipendenza) e lo stesso vale per una istanza con una sola tupla
- Per avere una istanza che non soddisfa una o più dipendenze il numero minimo (e sufficiente) di tuple è 2 (proprio quello che troviamo nella dimostrazione dei teoremi!)

- L'insieme di dipendenze definite (individuate in base alla realtà) su uno schema viene convenzionalmente denotato con la lettera F
- Dagli esempi a lezione abbiamo visto che se una istanza soddisfa tutte le dipendenze di F (è legale) allora **ne soddisfa molte di più anche se non sono state individuate esplicitamente**
- Esempio, se una istanza soddisfa modello \rightarrow marca e anche (modello, cilindrata) \rightarrow colore, allora intuitivamente ci aspettiamo che soddisferà anche (modello, cilindrata, marca) \rightarrow colore
- Le dipendenze in F^+ (chiusura di F) **hanno la proprietà di essere soddisfatte da ogni istanza legale dello schema**
- Sono importanti perché se modifichiamo F in maniera corretta F^+ non cambia!
- Due insiemi con la stessa chiusura sono equivalenti

- Vista l'importanza di F^+ , vorremmo essere in grado di capire «cosa c'è dentro».
- Possiamo farlo? Sì, indirettamente, perché scopriamo che la chiusura di Armstrong, denotata con F^A , che si calcola applicando ricorsivamente ad F gli assiomi omonimi, **è proprio UGUALE ad F^+**
- Lo scopriamo dimostrando un teorema che ci dice che **tutte le dipendenze di F^A hanno la proprietà di essere soddisfatte da ogni istanza legale, e quindi sono in F^+ , e che non può esistere una dipendenza in F^+ che non stia in F^A (che cioè non può essere ottenuta applicando gli assiomi di Armstrong)**
- **ATTENZIONE:** questi due insiemi sono proprio UGUALI, cioè contengono le stesse dipendenze, mentre due insiemi equivalenti sono diversi (non tutte le dipendenze di uno sono anche nell'altro) ma hanno la stessa chiusura

- La terza forma normale (3NF) è una proprietà dello **schema** di relazione (quindi per verificarla non ci occorrono le istanze!)
- Che si fa se lo schema non è 3NF? Si può **sempre** decomporre (vale per questa forma normale, ma non per tutte, ad esempio non vale per la forma normale di Boyce-Codd)

- La decomposizione di uno schema effettuata per avere o mantenere la 3NF deve soddisfare 3 condizioni:
 1. Ogni sottoschema deve essere 3NF
 2. Devono essere preservate le dipendenze (insiemi equivalenti, cioè con uguale chiusura)
 3. Devono essere preservati i dati; perdita = tuple estranee all'informazione originale
- Abbiamo un algoritmo che produce una decomposizione con le 3 proprietà, e inoltre due algoritmi per verificare se una decomposizione (non ottenuta dall'algoritmo di decomposizione!) soddisfa le proprietà 2 e 3



- Concetti di base:
 - Schema e Istanza
 - Dipendenze funzionali e istanze legali
 - Chiusura di F (F^+ ed F^A)
 - 3NF
 - Decomposizioni e loro proprietà
- Come sono collegati? Guardiamo una semplice mappa concettuale