

*Corso di Laurea in Ingegneria Gestionale*  
*Sapienza - Università di Roma*

# ***Corso di Basi di Dati***

## ***A.A. 2015/2016***

### **10 – Normalizzazione (cenni)**

**Tiziana Catarci, Andrea Marrella**

Ultimo aggiornamento : 22/02/2016

# Forme normali

---

- ▶ Una **forma normale** è una proprietà di una base di dati relazionale che ne garantisce la “qualità”, cioè l'assenza di determinati difetti.
- ▶ Quando una relazione non è normalizzata:
  - ▶ presenta **ridondanze**,
  - ▶ si presta a **comportamenti poco desiderabili durante le operazioni di aggiornamento**.
- ▶ Per gli schemi che non soddisfano una forma normale, è possibile applicare un procedimento, detto di **normalizzazione**, che permette di trasformare schemi non normalizzati in schemi che soddisfano una forma normale.
- ▶ La normalizzazione si utilizza prevalentemente come **tecnica di verifica** dei risultati della progettazione di una base di dati.

# Una relazione con anomalie

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Funzione
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

## ► Osservazioni:

- Lo stipendio per ciascun impiegato è unico ed è funzione del solo impiegato, indipendentemente dai progetti cui partecipa.
- Il bilancio per ciascun progetto è unico ed è funzione del solo progetto, indipendentemente dagli impiegati che vi partecipano.
- Ogni impiegato, in ciascun progetto cui partecipa, svolge una sola funzione, eventualmente diversa da progetto a progetto.

# Una relazione con anomalie

**Anomalia di cancellazione:** Se un impiegato interrompe la partecipazione a tutti i progetti senza lasciare l'azienda, dobbiamo cancellarlo definitivamente, a meno di ammettere valori nulli sulla chiave *Progetto*, il che è **inammissibile**.

**Anomalia di inserimento:** Un nuovo impiegato senza progetto non può essere inserito.

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Funzione
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

**Ridondanza:** Lo stipendio di ciascun impiegato è ripetuto in tutte le tuple relative.

**Anomalia di aggiornamento:** Se lo stipendio di un impiegato varia, è necessario andarne a modificare il valore in diverse tuple.

# Perché questi fenomeni indesiderabili?

---

- ▶ Abbiamo usato un'unica relazione per rappresentare informazioni eterogenee:
  - ▶ gli impiegati con i relativi stipendi;
  - ▶ i progetti con i relativi bilanci;
  - ▶ le partecipazioni degli impiegati ai progetti con le relative funzioni.
- ▶ Per studiare in maniera sistematica questi aspetti, è necessario introdurre un nuovo vincolo di integrità: la **dipendenza funzionale (DF)**. Essa descrive **legami di tipo funzionale** tra gli attributi di una relazione.

# Individuare le dipenden

Lo stipendio di ciascun impiegato è **unico** (anche se partecipa a più progetti).

Perciò, il valore dell'attributo *Impiegato* **determina** il valore dell'attributo *Stipendio*. Più precisamente, **esiste una funzione** che associa ad ogni elemento del dominio dell'attributo *Impiegato* un solo elemento del dominio dell'attributo *Stipendio*.

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Funzione
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

Il bilancio di ciascun progetto è **unico** e dipende solo dal progetto, indipendentemente dagli impiegati che vi partecipano.

Perciò, il valore dell'attributo *Progetto* **determina** il valore dell'attributo *Bilancio*.

# Dipendenze funzionali

---

- ▶ Il concetto di dipendenza funzionale può essere formalizzato come segue:
  - ▶ data una relazione  $r$  su uno schema  $R(X)$
  - ▶ e dati due sottoinsiemi non vuoti  $Y$  e  $Z$  di  $X$
  - ▶ esiste in  $r$  una **dipendenza funzionale (DF)** da  $Y$  a  $Z$  se, per ogni coppia di tuple  $t1$  e  $t2$  di  $r$  con gli stessi valori su  $Y$  (cioè tali che  $t1[Y]=t2[Y]$ ), risulta che  $t1$  e  $t2$  hanno gli stessi valori anche su  $Z$  (perciò,  $t1[Z]=t2[Z]$ ).
- ▶ Una dipendenza funzionale tra  $Y$  e  $Z$  viene generalmente indicata con la seguente notazione:  
 **$Y \rightarrow Z$**

# Individuare le dipendenze funzionali

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Funzione
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

Due dipendenze funzionali facilmente individuabili:

**Impiegato → Stipendio**

**Progetto → Bilancio**

Si noti che le anomalie precedentemente identificate **sono legate proprio alla presenza di dipendenze funzionali.**



# Dipendenze funzionali banali

---

- ▶ Nella relazione precedente è verificata la seguente DF:  
**Impiegato Stipendio  $\rightarrow$  Stipendio**
- ▶ in quanto due tuple con gli stessi valori sulla coppia di attributi **Impiegato** e **Stipendio** hanno ovviamente lo stesso valore sull'attributo **Stipendio**, che è uno dei due. Si tratta di una dipendenza funzionale **banale** (sempre soddisfatta).
- ▶ Una DF  **$Y \rightarrow A$**  è **banale** se **A** appartiene a **Y**.
- ▶ Una DF  **$Y \rightarrow Z$**  è **non banale** se nessun attributo in **Z** appartiene a **Y**.
- ▶ D'ora in poi ci riferiremo sempre a dipendenze funzionali non banali.

# Dipendenze Funzionali e chiavi

---

- ▶ Se prendiamo una chiave **K** di una relazione **R** si può facilmente verificare che esiste una DF tra **K** e ogni altro attributo dello schema di **R**.
- ▶ Infatti, per la definizione stessa di vincolo di chiave, non possono esistere due tuple con gli stessi valori su **K**, e quindi una DF che ha **K** al primo membro sarà sempre soddisfatta.
- ▶ Perciò, esisterà sempre una DF tra una chiave di una relazione e **tutti gli attributi dello schema della relazione** (esclusi quelli della chiave stessa).

# Dipendenze Funzionali e chiavi

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Funzione
Rossi	20	Marte	2	Tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

- Dato che gli attributi **Impiegato** e **Progetto** formano una chiave:

**Impiegato Progetto → Funzione**

Analogamente, le seguenti DF non banali saranno automaticamente verificate:

**Impiegato Progetto → Stipendio; Impiegato Progetto → Bilancio; Impiegato Progetto →**  
**~~Stipendio Funzione; Impiegato Progetto → Stipendio Bilancio ; Impiegato Progetto →~~**

- **Bilancio Funzione; Impiegato Progetto → Stipendio Bilancio Funzione**

# Dipendenze Funzionali e chiavi

---

- ▶ Abbiamo affermato che esisterà sempre una DF tra una chiave di una relazione e **tutti gli attributi dello schema della relazione** (esclusi quelli della chiave stessa), perciò:

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Funzione
------------------	-----------	-----------------	----------	----------

**Impiegato Progetto → Stipendio Bilancio Funzione**

- ▶ Possiamo quindi concludere dicendo che il vincolo di dipendenza funzionale **generalizza** il vincolo di chiave.
- ▶ Più precisamente, una DF  $Y \rightarrow Z$  su uno schema  $R(X)$  **degenera in un vincolo di chiave** se l'unione fra  $Y$  e  $Z$  è pari a  $X$ . In tal caso, infatti,  $Y$  è (super)chiave per lo schema  $R(X)$ .

# Dipendenze Funzionali e anomalie

---

- ▶ Si può notare che nelle prime due DF:

**Impiegato  $\rightarrow$  Stipendio**

**Progetto  $\rightarrow$  Bilancio**

**la prima componente non corrisponde a chiave causando anomalie.**

- ▶ Al contrario, la terza DF:

**Impiegato Progetto  $\rightarrow$  Funzione**

**corrisponde ad una chiave e non causa anomalie.**

- ▶ La differenza risiede nel fatto che **Impiegato** e **Progetto** formano una superchiave (nello specifico, la coppia di attributi è l'unica chiave della relazione), che – per definizione – non può contenere due tuple uguali. Perciò, la terza DF non genera mai ridondanze.
- ▶ Le anomalie sono perciò causate dalle DF  **$Y \rightarrow Z$**  tali che  **$Y$**  non contiene una chiave.

# Forma Normale di Boyce e Codd

---

- ▶ Una relazione  $r$  è in forma normale di Boyce e Codd (**BCNF**) se, per ogni dipendenza funzionale (non banale)  $X \rightarrow Y$  definita su di essa,  $X$  contiene una chiave  $K$  di  $r$ .
- ▶ La BCNF richiede che i concetti in una relazione siano omogenei (solo proprietà direttamente associate alla chiave).
- ▶ Se una relazione  $r$  non soddisfa la BCNF, la rimpiazziamo con altre relazioni che soddisfano la BCNF, **decomponendo**  $r$  sulla base delle dipendenze funzionali, al fine di separare i concetti.

# Procedura intuitiva di normalizzazione

---

- ▶ Non valida in generale, ma solo nei ‘casi semplici’:
- ▶ Per ogni dipendenza  $\mathbf{X} \rightarrow \mathbf{Y}$  (con diverso primo membro) che viola la BCNF, definire una relazione su  $\mathbf{XY}$  ed eliminare  $\mathbf{Y}$  dalla relazione originaria.
- ▶ **Purtroppo con questa tecnica non sempre le decomposizioni sono di buona qualità!**

# Un esempio di decomposizione

Le tre relazioni, ottenute decomponendo la relazione originaria secondo le DF, sono in BCNF.

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Funzione
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

**Progetto → Bilancio**

<u>Progetto</u>	Bilancio
Marte	2
Giove	15
Venere	15

Si noti che abbiamo decomposto la relazione originaria in modo che a ciascuna dipendenza corrisponda una diversa relazione la cui **chiave** è proprio il **primo membro della dipendenza stessa**.

**Impiegato → Stipendio**

<u>Impiegato</u>	Stipendio
Rossi	20
Verdi	35
Neri	55
Mori	48
Bianchi	48

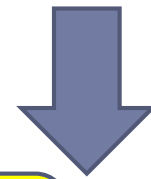


# Decomposizione senza perdita

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

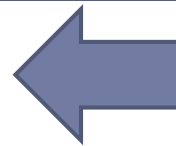
**Impiegato → Sede**  
**Progetto → Sede**

Decomponiamo la relazione in base alle dipendenze funzionali, secondo la tecnica vista precedentemente, proiettando la relazione originaria sugli attributi Impiegato e Sede e Progetto e Sede, risp.



La relazione ricostruita contiene **tuple aggiuntive** (dette **spurie**) rispetto alla relazione originaria. Ad esempio, risulta che Verdi lavora a Milano sul progetto Saturno, ma – guardando la relazione originaria – Verdi effettivamente non lavora a tale progetto.

<u>Impiegato</u>	<u>Sede</u>
Rossi	Roma
Verdi	Milano
Neri	Milano



Proviamo a ricostruire le informazioni sulla base della partecipazione degli impiegati ai progetti attraverso un join naturale, utilizzando l'attributo **Sede**, che è l'unico attributo comune alle due relazioni.

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Saturno	Milano
Neri	Venere	Milano
Verdi	Saturno	Milano
Neri	Giove	Milano

<u>Progetto</u>	<u>Sede</u>
Marte	Roma
Giove	Milano
Venere	Milano
Saturno	Milano

# Decomposizione senza perdita

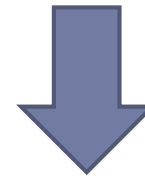
---

- ▶ Una relazione  $r$  si scompone senza perdita su  $X_1$  e  $X_2$  se il join delle proiezioni di  $r$  su  $X_1$  e  $X_2$  è uguale a  $r$  stessa (cioè non contiene tuple spurie).
  - ▶ Si tratta di un **requisito irrinunciabile** il fatto che una decomposizione sia effettuata senza perdita.
- ▶ E' possibile individuare una condizione che garantisce la decomposizione senza perdita di una relazione:
  - ▶ Sia  $r$  una relazione su un insieme di attributi  $X$  e siano  $X_1$  e  $X_2$  due sottoinsiemi di  $X$  tali che  $X = X_1 \cup X_2$
  - ▶ Inoltre, sia  $X_0 = X_1 \cap X_2$  ;
  - ▶ Allora,  $r$  si **scompone senza perdita** su  $X_1$  e  $X_2$  se soddisfa la dipendenza funzionale  $X_0 \rightarrow X_1$  oppure  $X_0 \rightarrow X_2$
- ▶ In altre parole, la decomposizione senza perdita è **garantita** se gli attributi comuni nelle relazioni decomposte **contengono una chiave** per almeno una delle relazioni decomposte.

# Decomposizione senza perdita

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

**Impiegato → Sede**  
**Progetto → Sede**



Decomponiamo la relazione originaria **r** sfruttando solamente la **DF** **Impiegato → Sede**, tenendo conto che nella relazione decomposta Impiegato è **chiave** per la proiezione di **r** sugli attributi Impiegato e Sede.

Impiegato	Sede
Rossi	Roma
Verdi	Milano
Neri	Milano



Proviamo a ricostruire le informazioni attraverso un join naturale, utilizzando l'attributo **Impiegato**, che è l'unico attributo comune alle due relazioni.

Impiegato	Progetto
Rossi	Marte
Verdi	Giove
Verdi	Venere
Neri	Saturno
Neri	Venere

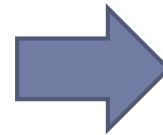
La relazione ricostruita è identica alla relazione originaria.

E' importante notare che la tecnica appena analizzata è **sufficiente** ma **non necessaria** per la decomposizione senza perdita. Esistono infatti istanze di relazioni che **non soddisfano nessuna delle due dipendenze**, ma al tempo stesso si decompongono senza perdita.

# Un ulteriore problema

Impiegato	Sede
Rossi	Roma
Verdi	Milano
Neri	Milano

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Saturno	Milano
Neri	Venere	Milano



**Impiegato → Sede**  
**Progetto → Sede**

Impiegato	Progetto
Rossi	Marte
Verdi	Giove
Verdi	Venere
Neri	Saturno
Neri	Venere

- ▶ Nonostante la decomposizione ottenuta sia senza perdita, essa presenta un nuovo inconveniente...
- ▶ Supponiamo di voler inserire una nuova tupla che specifica la partecipazione dell'impiegato Neri, che opera a Milano, al progetto Marte.

Impiegato	Progetto
Rossi	Marte
Verdi	Giove
Verdi	Venere
Neri	Saturno
Neri	Venere
<b>Neri</b>	<b>Marte</b>

- ▶ Sulla relazione originaria un tale aggiornamento verrebbe individuato come **illecito**, perché porterebbe ad una violazione della dipendenza **Progetto → Sede**.
- ▶ Sulla relazione decomposta **ImpiegatoProgetto**, invece, non è possibile definire alcuna dipendenza funzionale e perciò rilevare alcuna violazione di dipendenze.

# Conservazione delle dipendenze

---

- ▶ Una decomposizione **conserva le dipendenze** se ciascuna delle dipendenze funzionali dello schema originario **coinvolge attributi che compaiono tutti insieme in uno degli schemi decomposti**.
- ▶ In questo modo, è possibile garantire, sullo schema decomposto, il soddisfacimento degli stessi vincoli garantiti dallo schema originario.
- ▶ La DF **Progetto → Sede** non è conservata nella decomposizione dell'esempio precedente.
- ▶ La decomposizione a pag. 16 consente di conservare le dipendenze. Infatti, sia **Progetto → Bilancio** che **Impiegato → Stipendio** compaiono tutti insieme in uno degli schemi decomposti.

# Qualità delle decomposizioni

---

- ▶ Una decomposizione dovrebbe sempre soddisfare:
  - ▶ la **decomposizione senza perdita**, che garantisce la ricostruzione delle informazioni originarie (cioè senza informazioni spurie) a partire da quelle rappresentate nelle relazioni decomposte;
  - ▶ la **conservazione delle dipendenze**, che garantisce il mantenimento dei vincoli di integrità originari, con le relazioni decomposte che hanno la stessa capacità della relazione originaria di rappresentare i vincoli di integrità e quindi di rilevare aggiornamenti illeciti.
  - ▶ Ovviamente sono possibili sulle relazioni decomposte ulteriori aggiornamenti, legati ai singoli concetti rappresentati in ciascuna di esse, che non hanno un corrispettivo sulla relazione originaria, senza però corrispondere a violazione dei vincoli.
- ▶ Dato uno schema che viola una forma normale, l'attività di **normalizzazione** è quindi volta a ottenere una decomposizione che sia *senza perdita*, *conservi le dipendenze* e che *contenga relazioni in forma normale*. La decomposizione a pag. 16 presenta tutte e tre le qualità.

# Limitazioni della forma normale di Boyce e Codd

- ▶ Talvolta non è possibile raggiungere una buona decomposizione in forma normale di Boyce e Codd.

Dirigente	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Marte	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

Supponiamo che siano definite le seguenti dipendenze:

**Dirigente** → **Sede** (ogni dirigente opera presso una sede)

**Progetto Sede** → **Dirigente** (per ogni sede, un progetto ha un solo responsabile).

- ▶ La relazione non è in BCNF, perché il primo membro della DF **Dirigente** → **Sede** non è superchiave.
- ▶ Allo stesso modo, si può notare come non sia possibile alcuna buona decomposizione di questa relazione: infatti, la DF **Progetto Sede** → **Dirigente** coinvolge tutti gli attributi e quindi nessuna decomposizione è in grado di conservarla.
- ▶ Perciò, talvolta la **BCNF non è raggiungibile**.

# Definizione di terza forma normale

---

- ▶ Per trattare casi come quello dell'esempio appena visto, si ricorre ad una forma normale meno restrittiva della BCNF (e ammette relazioni con alcune anomalie).
- ▶ Una relazione  $r$  è in terza forma normale se, per ogni DF (non banale)  $\mathbf{X} \rightarrow \mathbf{Y}$  definita su  $\mathbf{r}$ , è verificata almeno una delle seguenti condizioni:
  - ▶  $\mathbf{X}$  contiene una chiave  $\mathbf{K}$  di  $\mathbf{r}$
  - ▶ ogni attributo in  $\mathbf{Y}$  è contenuto in almeno una chiave di  $\mathbf{r}$



# Limitazioni della forma normale di Boyce e Codd

Dirigente	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Marte	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

**Dirigente  $\rightarrow$  Sede**

**Progetto Sede  $\rightarrow$  Dirigente**

- ▶ La relazione non soddisfa la BCNF, ma soddisfa la 3FN. Infatti, **Progetto Sede  $\rightarrow$  Dirigente** ha come primo membro

una chiave della relazione, mentre **Dirigente  $\rightarrow$  Sede**, pur non contenendo una chiave al primo membro, ha un unico attributo al secondo membro che fa parte della chiave **Progetto Sede**.

- ▶ Si osservi che la relazione **presenta una forma di ridondanza**. Ogni volta che un Dirigente compare in una tupla, viene ripetuta per esso la Sede in cui opera.
- ▶ Questa ridondanza viene però **tollerata** dalla 3FN, perché non sarebbe possibile una decomposizione in BCNF che elimini tale ridondanza e al tempo stesso conservi tutte le dipendenze.

# Decomposizione in terza forma normale

---

- ▶ La 3FN è meno restrittiva della forma normale di Boyce e Codd e quindi non offre le medesime garanzie di qualità per una relazione; ha però il vantaggio di essere **sempre ottenibile**.
- ▶ E' possibile infatti dimostrare che una qualunque relazione che non soddisfa la 3FN è certamente **decomponibile senza perdita** e con **conservazione delle dipendenze** in relazioni in 3FN.
- ▶ Una relazione che non soddisfa la 3FN si scompone in tante relazioni ottenute per proiezione sugli attributi corrispondenti alle dipendenze funzionali (con diverso primo membro) che violano la 3FN, con l'unica accortezza che alla fine **almeno una delle relazione decomposte contenga una chiave della relazione originaria**.

# Decomposizione in terza forma normale

Dirigente	Progetto	Stipendio
Rossi	Marte	30
Verdi	Giove	40
Verdi	Venere	40
Neri	Saturno	50
Neri	Venere	50

## Dirigente → Stipendio

- La relazione non soddisfa la 3FN. Infatti, l'unica DF **Dirigente → Stipendio** ha come primo membro un attributo non chiave, e come secondo membro un attributo che non partecipa a nessuna chiave della relazione.

- Una decomposizione in una relazione negli attributi **Dirigente Stipendio** e in un'altra sull'attributo **Progetto** violerebbe la proprietà di decomposizione senza perdita, in quanto le due relazioni non contengono un attributo in comune.
- Per garantire tale proprietà, dobbiamo invece definire la seconda relazione sugli attributi **Dirigente Progetto**, che formano una chiave della relazione originaria.
- Le due relazioni ottenute sono in 3FN!**

Dirigente	Stipendio
Rossi	30
Verdi	40
Neri	50

Dirigente	Progetto
Rossi	Marte
Verdi	Giove
Verdi	Venere
Neri	Saturno
Neri	Venere