



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Normalizzazione

Basi di Dati

Corso di Laurea in Informatica per il Management

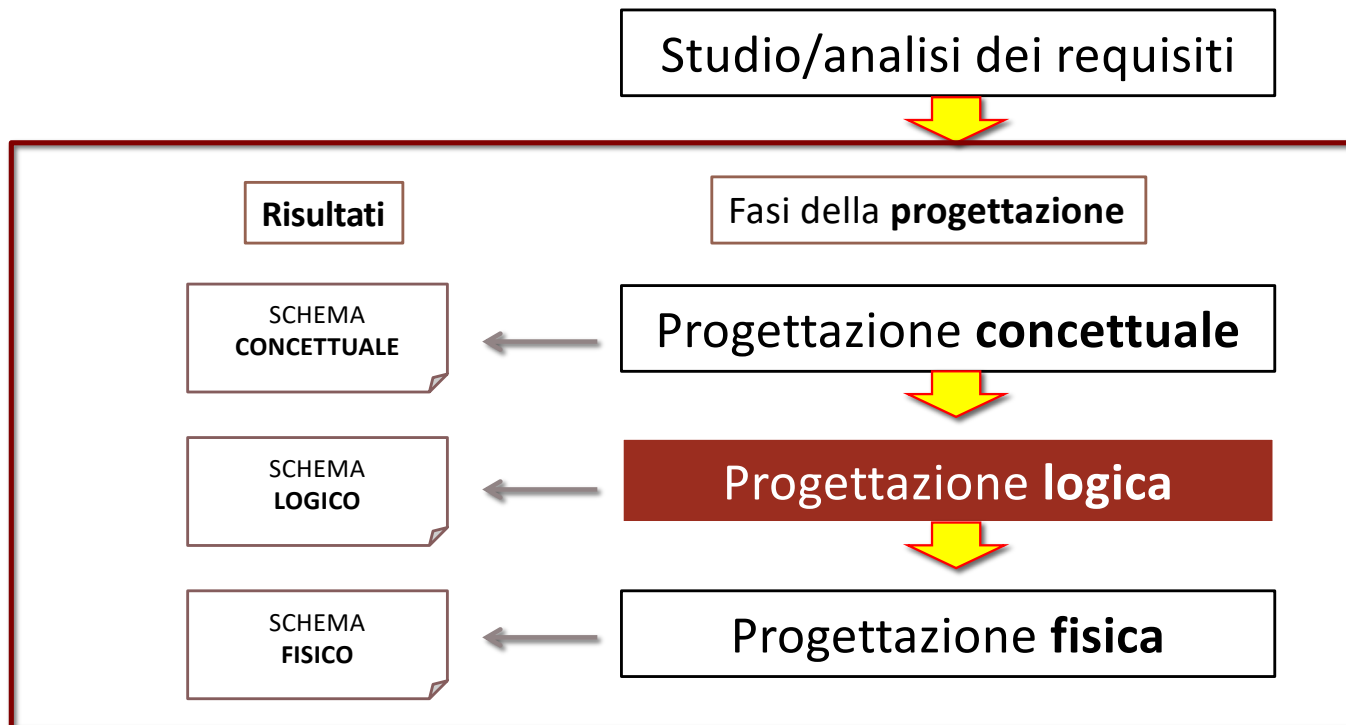
Alma Mater Studiorum - Università di Bologna

Prof. Marco Di Felice

Dipartimento di Informatica – Scienza e Ingegneria

marco.difelice3@unibo.it

Progettazione logica



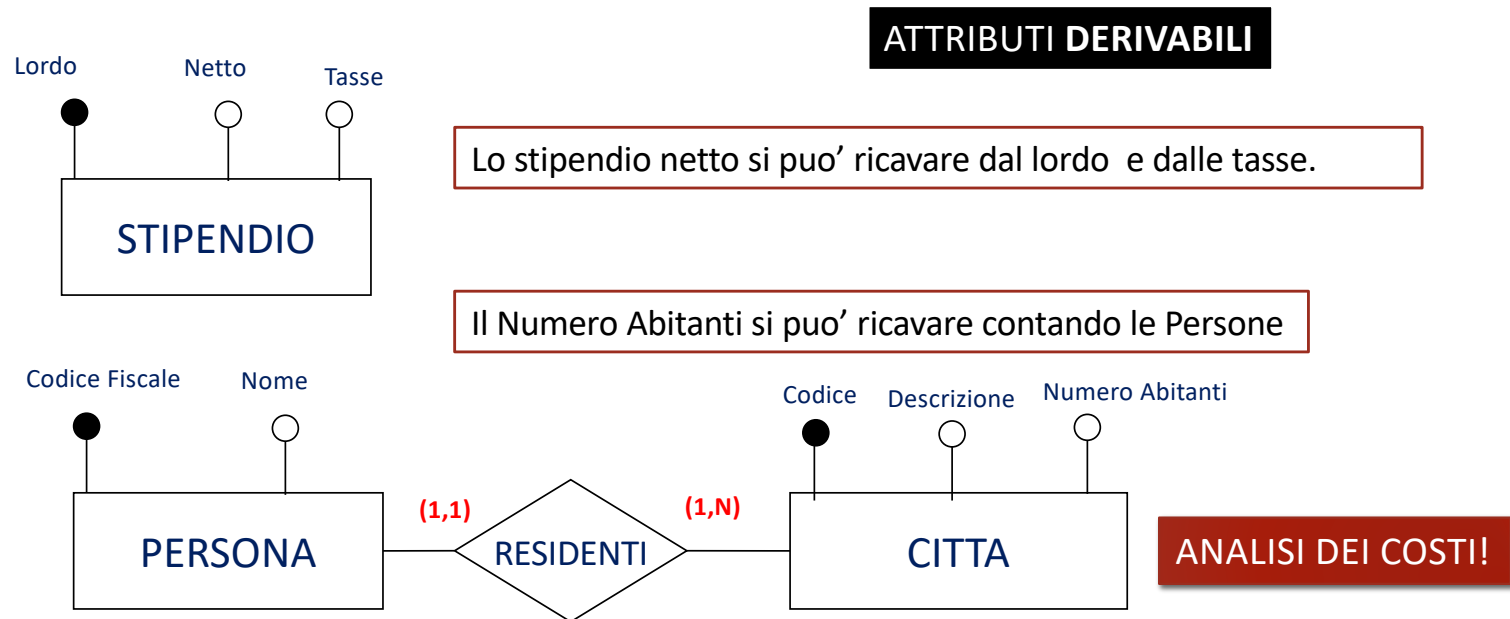
Normalizzazione dei dati

Le **ridondanze** sui dati possono essere di due tipi:

- **Ridondanza concettuale** → non ci sono duplicazioni dello stesso dato, ma sono memorizzate informazioni che possono essere ricavate da altre già contenute nel DB.
- **Ridondanza fisica** → esistono duplicazioni sui dati, che possono generare anomalie nelle operazioni sui dati ...

Normalizzazione dei dati

Esempi di **ridondanze concettuali** che possono presentarsi già nel diagramma E-R...



Normalizzazione dei dati

Le **ridondanze** sui dati possono essere di due tipi:

- **Ridondanza concettuale** → non ci sono duplicazioni dello stesso dato, ma sono memorizzate informazioni che possono essere ricavate da altre già contenute nel DB.
- **Ridondanza fisica** → esistono duplicazioni sui dati, che possono generare anomalie nelle operazioni sui dati ...

Normalizzazione dei dati

D. Cosa c'è di strano in questa tabella?

<u>Docente</u>	Livello	Salario	Dipartimento	Direttore	<u>Corso</u>
Rossi	4	15000	Fisica	Neri	Mat. Discreta
Rossi	4	15000	Chimica	Rossini	Analisi I
Bianchi	3	10000	Informatica	Viola	Basi di Dati
Neri	4	15000	Informatica	Viola	Programmazione
Neri	4	15000	Matematica	Bruni	Inf. di base
Rossi	3	15000	Matematica	Bruni	Geometria

Normalizzazione dei dati

D. Cosa c'è di strano in questa tabella?

<u>Docente</u>	Livello	Salario	Dipartimento	Direttore	<u>Corso</u>
Rossi	4	15000	Fisica	Neri	Mat. Discreta
Rossi	4	15000	Chimica	Rossini	Analisi I
Bianchi	3	10000	Informatica	Viola	Basi di Dati
Neri	4	15000	Informatica	Viola	Programmazione
Neri	4	15000	Matematica	Bruni	Inf. di base
Rossi	3	15000	Matematica	Bruni	Geometria

R. Lo stipendio di ciascun docente è ripetuto in tutte le tuple relative → **Ridondanze sui dati!**

Normalizzazione dei dati

D. Cosa c'è di strano in questa tabella?

<u>Docente</u>	Livello	Salario	Dipartimento	Direttore	<u>Corso</u>
Rossi	4	15000	Fisica	Neri	Mat. Discreta
Rossi	4	15000	Chimica	Rossini	Analisi I
Bianchi	3	10000	Informatica	Viola	Basi di Dati
Neri	4	15000	Informatica	Viola	Programmazione
Neri	4	15000	Matematica	Bruni	Inf. di base
Rossi	3	15000	Matematica	Bruni	Geometria

A. Il direttore di un dipartimento è ripetuto in tutte le tuple relative → **Ridondanze sui dati!**

Normalizzazione dei dati

D. Cosa c'è di strano in questa tabella?

<u>Docente</u>	Livello	Salario	Dipartimento	Direttore	<u>Corso</u>
Rossi	4	15000	Fisica	Neri	Mat. Discreta
Rossi	4	15000	Chimica	Rossini	Analisi I
Bianchi	3	10000	Informatica	Viola	Basi di Dati
Neri	4	15000	Informatica	Viola	Programmazione
Neri	4	15000	Matematica	Bruni	Inf. di base
Rossi	3	15000	Matematica	Bruni	Geometria

- **Anomalia di aggiornamento** → se varia lo stipendio, devo modificare tutte le tuple del docente!

Normalizzazione dei dati

Q. Cosa c'è di strano in questa tabella?

<u>Docente</u>	Livello	Salario	Dipartimento	Direttore	<u>Corso</u>
Rossi	4	15000	Fisica	Neri	Mat. Discreta
Rossi	4	15000	Chimica	Rossini	Analisi I
Bianchi	3	10000	Informatica	Viola	Basi di Dati
Neri	4	15000	Informatica	Viola	Programmazione
Neri	4	15000	Matematica	Bruni	Inf. di base
Rossi	3	15000	Matematica	Bruni	Geometria

- **Anomalia di cancellazione** → Se un docente non ha corsi, dobbiamo eliminare tutti i suoi dati ...

Normalizzazione dei dati

<u>Docente</u>	Livello	Salario	Dipartimento	Direttore	<u>Corso</u>
----------------	---------	---------	--------------	-----------	--------------

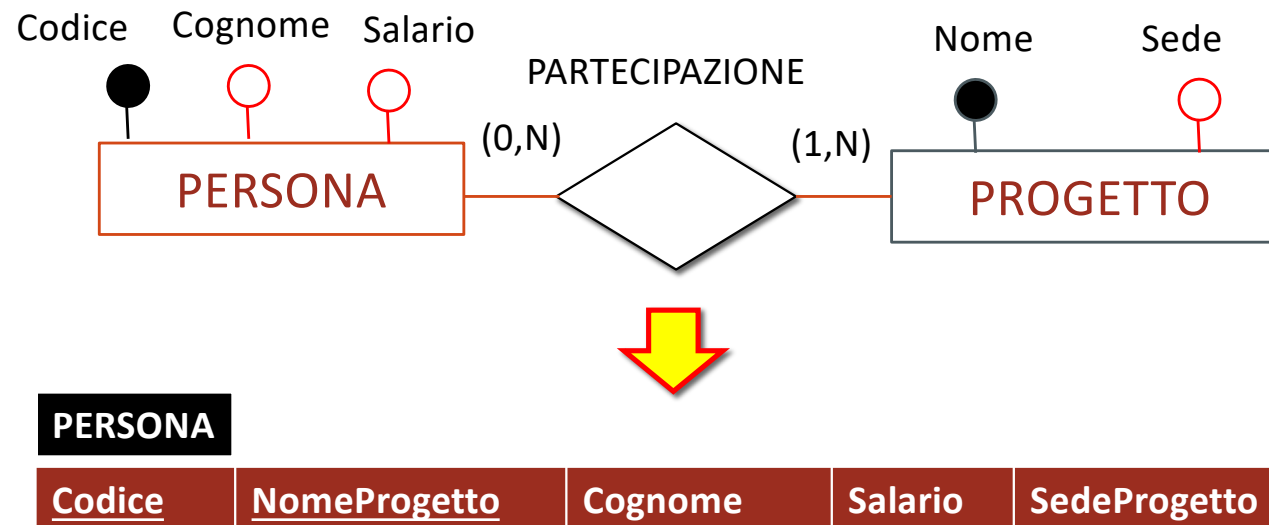
- **V1.** Ogni dipartimento ha un solo direttore.
- **V2.** Ogni docente ha un solo stipendio (anche se ha più corsi).
- **V3.** Lo stipendio dipende dal livello e non dal dipartimento o dal corso tenuto!

PROBLEMA: Abbiamo usato **un'unica tabella per rappresentare informazioni eterogenee!**

Normalizzazione dei dati

Da dove deriva una **ridondanza**?

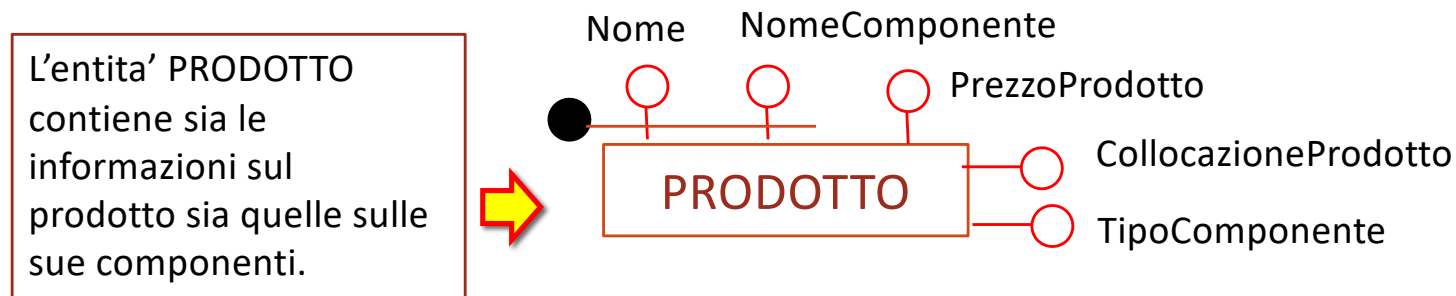
- Traduzioni non corrette nel modello logico relazionale...



Normalizzazione dei dati

Da dove deriva una **ridondanza**?

- Errori durante la progettazione concettuale ...



- Sarebbe stato meglio ristrutturare lo schema E-R, partizionando l'entità ed introducendo delle relazioni **uno-a-molti** o **molti-a-molti**!

Normalizzazione dei dati

- Per risolvere le anomalie viste fin qui, si introduce un nuovo concetto del modello relazionale: la **Dipendenza Funzionale (DF)**.

Data una tabella su uno schema $R(X)$ e due attributi Y e Z di X . Esiste la **dipendenza funzionale** $Y \rightarrow Z$ se per ogni coppia di tuple t_1 e t_2 di r con $t_1[Y] = t_2[Y]$, si ha anche che $t_1[Z] = t_2[Z]$.

Normalizzazione dei dati

- Per risolvere le anomalie viste fin qui, si introduce un nuovo concetto del modello relazionale: la **Dipendenza Funzionale (DF)**.

Data una tabella su uno schema $R(X)$ e due liste di attributi $Y=\{Y_0, Y_1, \dots, Y_n\}$ e $Z=\{Z_0, Z_1, \dots, Z_m\}$. Esiste la **dipendenza funzionale** $Y \rightarrow Z$ se per ogni coppia di tuple $t1$ e $t2$ di r con $t1[Y]=t2[Y]$, si ha anche che $t1[Z]=t2[Z]$.

Normalizzazione dei dati

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Sede	Ruolo
Rossi	20000	Marte	Roma	Tecnico
Verdi	35000	Giove	Bologna	Tecnico
Verdi	35000	Venere	Milano	Progettista
Neri	55000	Venere	Milano	Direttore
Neri	55000	Giove	Bologna	Direttore
Neri	55000	Marte	Roma	Tecnico
Bianchi	48000	Venere	Milano	Consulente

- **DF1: Impiegato → Stipendio**
Spiegazione: [Ogni impiegato ha un unico stipendio]

Normalizzazione dei dati

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Sede	Ruolo
Rossi	20000	Marte	Roma	Tecnico
Verdi	35000	Giove	Bologna	Tecnico
Verdi	35000	Venere	Milano	Progettista
Neri	55000	Venere	Milano	Direttore
Neri	55000	Giove	Bologna	Direttore
Neri	55000	Marte	Roma	Tecnico
Bianchi	48000	Venere	Milano	Consulente

- **DF2: Progetto → Sede**

Spiegazione: [Ogni progetto ha un'unica sede]

Normalizzazione dei dati

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Sede	Ruolo
Rossi	20000	Marte	Roma	Tecnico
Verdi	35000	Giove	Bologna	Tecnico
Verdi	35000	Venere	Milano	Progettista
Neri	55000	Venere	Milano	Direttore
Neri	55000	Giove	Bologna	Direttore
Neri	55000	Marte	Roma	Tecnico
Bianchi	48000	Venere	Milano	Consulente

DF3: Impiegato → Impiegato

DF ovvia ($Y \rightarrow Y$). Per definizione stessa di DF ...

Normalizzazione dei dati

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Sede	Ruolo
Rossi	20000	Marte	Roma	Tecnico
Verdi	35000	Giove	Bologna	Tecnico
Verdi	35000	Venere	Milano	Progettista
Neri	55000	Venere	Milano	Direttore
Neri	55000	Giove	Bologna	Direttore
Neri	55000	Marte	Roma	Tecnico
Bianchi	48000	Venere	Milano	Consulente

- **DF4: Impiegato Progetto → Ruolo**

Spiegazione: Un impiegato può coprire un solo ruolo per progetto!

Normalizzazione dei dati

- Le dipendenze funzionali sono definite **a livello di schema** e non a livello di istanza!

<u>Matricola</u>	Cognome	<u>Corso</u>	Voto
1244	Rossi	Basi di Dati	18
1567	Bianchi	Programmazione	22
1898	Bianchi	Analisi I	20
2040	Verdi	Programmazione	22
2121	Verdi	Basi di Dati	18
2678	Bruni	Analisi I	20

- Dipendenza funzionale **Corso** → **Voto**? NO!

Normalizzazione dei dati

Le dipendenze funzionali sono definite **a livello di schema** e non a livello di istanza!

<u>Matricola</u>	Studente	<u>Corso</u>	Docente	Voto
1244	Rossi	Basi di Dati	Di Felice	18
1567	Bianchi	Programmazione	Messina	22
1898	Bianchi	Analisi I	Mughetti	20
2040	Verdi	Programmazione	Messina	22
2121	Verdi	Basi di Dati	Di Felice	18
2678	Bruni	Analisi I	Mughetti	20

- Dipendenza funzionale **Corso** → **Docente** ? Può essere, occorre considerare le specifiche del sistema ...

Normalizzazione dei dati

- Le dipendenze funzionali hanno sempre **un verso!**

<u>Matricola</u>	Studente	<u>Corso</u>	Docente	Voto
1244	Rossi	Basi di Dati	Di Felice	18
1567	Bianchi	Programmazione	Messina	22
1898	Bianchi	Analisi I	Mughetti	20
2040	Verdi	Programmazione	Messina	22
2121	Verdi	Basi di Dati	Di Felice	18
2678	Bruni	Analisi I	Mughetti	20
4354	Bruni	Sistemi Context-aware	Di Felice	28

○ Corso → Docente? **OK** Docente → Corso? **NO!**

Normalizzazione dei dati

- Le dipendenze funzionali sono una **generalizzazione del vincolo di chiave (e di superchiave)**.
- Data una tabella con schema $R(X)$, con superchiave K .
Esiste un vincolo di dipendenza funzionale tra K e qualsiasi attributo della tabella o combinazione degli stessi.

$$K \rightarrow X_1, \quad X_1 \subseteq X$$

Normalizzazione dei dati

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Sede	Ruolo
------------------	-----------	-----------------	------	-------

ESEMPIO. Impiegato, Progetto è una (super)**chiave della relazione** → non possono esistere due tuple con lo stesso valore della coppia <Impiegato, Progetto>!

DF₁: Impiegato Progetto → Stipendio

DF₂: Impiegato Progetto → Sede

DF₃: Impiegato Progetto → Ruolo

DF₄: Impiegato Progetto → Sede Ruolo

....

DF_n: Impiegato Progetto → Impiegato Stipendio Progetto Sede Ruolo



Normalizzazione dei dati

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Sede	Ruolo
Rossi	20000	Marte	Roma	Tecnico
Verdi	35000	Giove	Bologna	Tecnico
Verdi	35000	Venere	Milano	Progettista
Neri	55000	Venere	Milano	Direttore
Neri	55000	Giove	Bologna	Direttore
Neri	55000	Marte	Roma	Tecnico
Bianchi	48000	Venere	Milano	Consulente

- **DF1:** Impiegato → Stipendio
- **DF2:** Progetto → Sede
- **DF3:** Impiegato Progetto → Ruolo

Normalizzazione dei dati

Dipendenze funzionali “buone” e “cattive”.

- **DF1:** Impiegato → Stipendio 
- **DF2:** Progetto → Sede 

Ridondanza sui dati, possibili anomalie (aggiornamento, cancellazione, etc) nelle operazioni sui dati ...

- **DF3:** Impiegato Progetto → Ruolo 

Non determina ridondanze sui dati ...

Normalizzazione dei dati

Perchè DF3 non causa anomalie a differenza di DF1 e di DF2?

- **DF1:** Impiegato → Stipendio
- **DF2:** Progetto → Sede
- **DF3:** Impiegato Progetto → Ruolo

Motivo:

- **DF3** ha sulla sinistra una **(super)chiave**.
- **DF1** e **DF2** non contengono una (super)chiave.

Normalizzazione dei dati

FORMA NORMALE di BOYCE-CODD (FNBC)

Uno schema $R(X)$ si dice in **forma normale di Boyce e Codd** se **per ogni dipendenza funzionale (non ovvia) $Y \rightarrow Z$ definita su di esso, Y è una **superchiave** di $R(X)$.**

- Se una tabella è in FNBC, non presenta le anomalie e ridondanze viste fin qui.
- Se una tabella **NON** è in FNBC, bisogna trasformarla (**normalizzarla**) -se possibile- in FNBC.

Normalizzazione dei dati

Esempi di tabelle...

<u>Localita</u>	<u>Stato</u>	Abitanti
Roma	Italia	60000000
Cambridge	UK	50000
Cambridge	US	200000
Bologna	Italia	400000
NY	US	15000000

DF: Localita Stato → Abitanti
Rispetta la FNBC!

<u>Localita</u>	<u>Stato</u>	Prefisso
Roma	Italia	0039
Cambridge	US	001
Cambridge	UK	0044
Bologna	Italia	0039
NY	US	001

DF: Stato → Prefisso
NON rispetta la FNBC!

Normalizzazione dei dati

D. Come **normalizzare** una tabella?

R. Creare **tabelle separate** per ogni dipendenza funzionale

IMPIEGATO → STIPENDIO

<u>Impiegato</u>	Stipendio
Rossi	20000
Verdi	35000
Neri	55000
Bianchi	48000

IMPIEGATO, PROGETTO → RUOLO

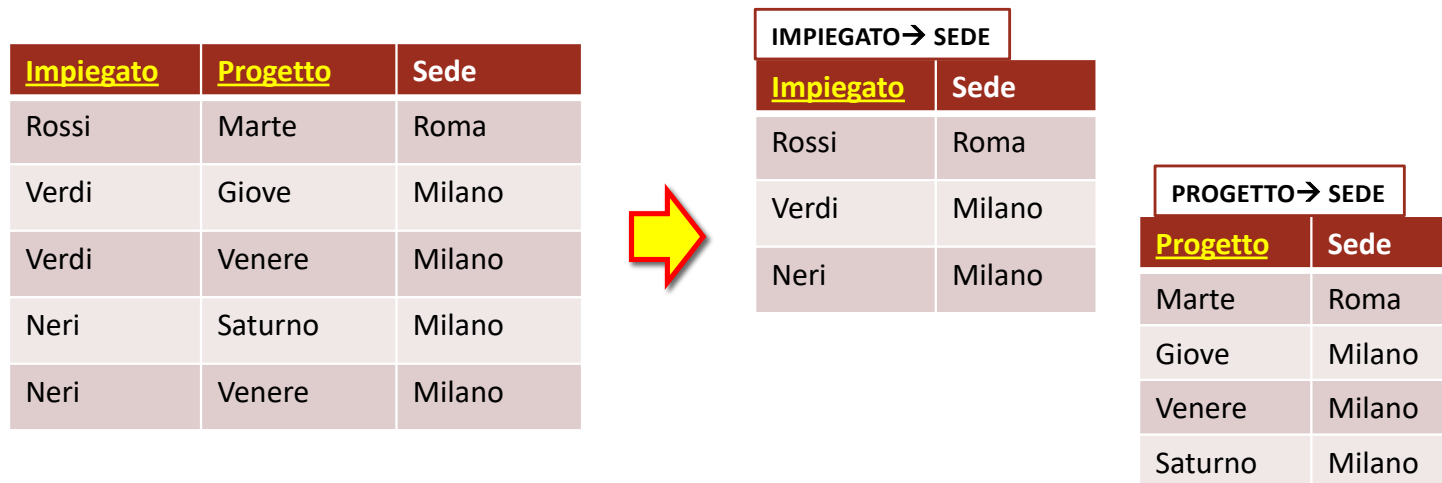
<u>Impiegato</u>	<u>Progetto</u>	Ruolo
Rossi	Marte	Tecnico
Verdi	Giove	Tecnico
Verdi	Venere	Progettista
Neri	Venere	Direttore
Neri	Giove	Direttore
Neri	Marte	Tecnico
Bianchi	Venere	Consulente

PROGETTO → SEDE

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

Normalizzazione dei dati

D. Tutte le decomposizioni vanno bene?



- **DF1. Impiegato → Sede** (Ogni impiegato lavora in una sola sede)
- **DF2. Progetto → Sede** (Ogni progetto ha la stessa sede)

Normalizzazione dei dati

D. Tutte le decomposizioni vanno bene?

<u>Impiegato</u>	Sede		<u>Progetto</u>	Sede		<u>Impiegato</u>	<u>Progetto</u>	Sede
Rossi	Roma		Marte	Roma		Rossi	Marte	Roma
Verdi	Milano	▷◁	Giove	Milano	=	Verdi	Giove	Milano
Neri	Milano		Venere	Milano		Verdi	Venere	Milano
			Saturno	Milano		Verdi	Saturno	Milano
						Neri	Venere	Milano
						Neri	Giove	Milano
						Neri	Saturno	Milano

- Se combino le due tabelle della decomposizione tramite operatore di join, non ottengo la tabella di partenza! (**decomposizione con perdita/aggiunta**)

ESEMPIO DI TUPLA SPURIA

Normalizzazione dei dati

DECOMPOSIZIONE SENZA PERDITA

Uno schema $R(X)$ si **decompone senza perdita** negli schemi $R1(X1)$ ed $R2(X2)$ se, per ogni possibile istanza r di $R(X)$, **il join naturale delle di $X1$ ed $X2$ produce la tabella di partenza.**

$$\pi_{X1}(r) \bowtie \pi_{X2}(r) = r$$

- In caso di **decomposizione con perdite/aggiunte**, possono generarsi delle **tuple spurie dopo il join.**

Normalizzazione dei dati

Anche se una decomposizione è senza aggiunte, può comunque presentare dei problemi di **conservazione delle dipendenze ...**

<u>Impiegato</u>	<u>Progetto</u>	<u>Sede</u>
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Venere	Milano
Neri	Saturno	Milano



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

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

- Con questa decomposizione, non ho tuple spurie ...

Normalizzazione dei dati

- Anche se una decomposizione è senza perdite, può comunque presentare dei problemi di **conservazione delle dipendenze ...**

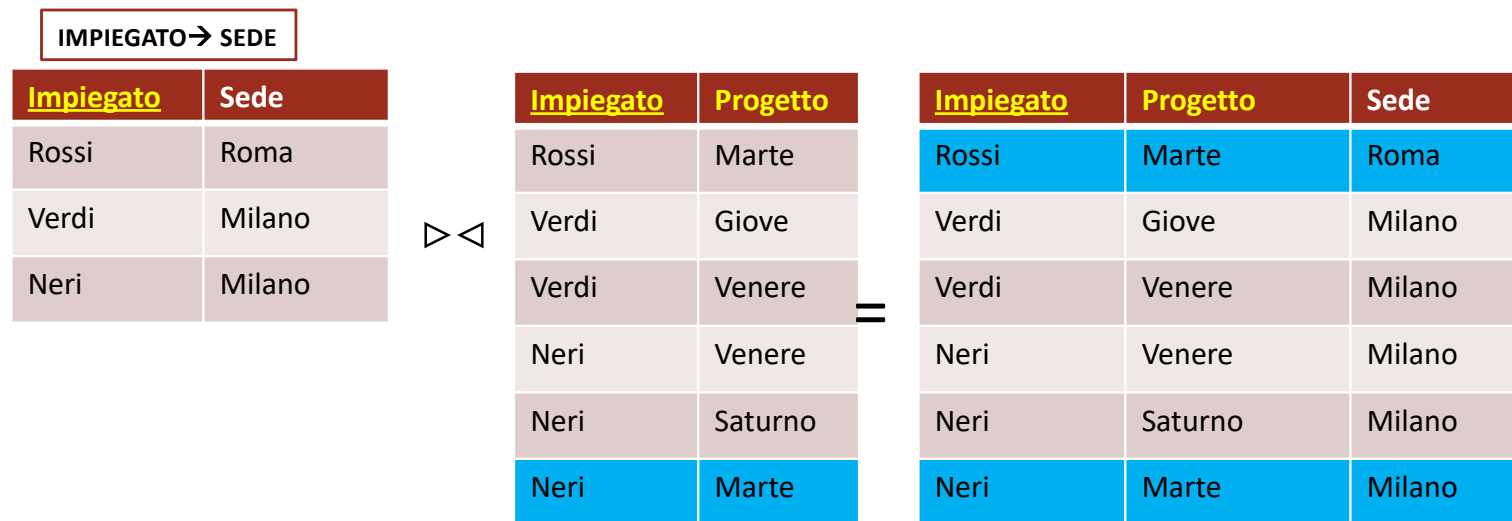
IMPIEGATO → SEDE	
<u>Impiegato</u>	Sede
Rossi	Roma
Verdi	Milano
Neri	Milano

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

D. Che accade se aggiungo l'impiegato Neri al progetto Marte?

Normalizzazione dei dati

- Anche se una decomposizione è senza perdite, può comunque presentare dei problemi di **conservazione delle dipendenze ...**



Violazione del vincolo di dipendenza **Progetto → Sede**

Normalizzazione dei dati

D. Tutte le decomposizioni vanno bene?

R. NO! Le decomposizione deve soddisfare **tre proprietà**:

- **Soddisfacimento della FNBC:** ogni tabella **deve essere in FNBC.**
- **Decomposizione senza perdita:** il join delle tabelle decomposte **deve produrre la relazione originaria.**
- **Conservazione delle dipendenze:** il join delle tabelle decomposte **deve rispettare tutte le DF dello schema originario.**

Normalizzazione dei dati

D. Data una relazione non in FNBC, è **sempre possibile** ottenere una **decomposizione in FNBC**?

R. NO! consideriamo un **controesempio** ...

Dirigente	<u>Progetto</u>	<u>Sede</u>
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Marte	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

- **DF1.** Progetto Sede → Dirigente
- **DF2.** Dirigente → Sede

PROBLEMA: DF1 coinvolge tutti gli attributi, nessuna decomposizione può preservare la dipendenza!

Normalizzazione dei dati

Per risolvere casi come quello precedente, si introduce **una nuova definizione di forma normale** **meno restrittiva della forma di Boyce e Codd...**

TERZA FORMA NORMALE (TFN)

Una tabella r è in **terza forma normale** se per ogni dipendenza funzionale $X \rightarrow A$ dello schema, **ALMENO UNA** delle seguenti condizioni è verificata:

- X è una superchiave di r
- A appartiene ad almeno una chiave K di r

Normalizzazione dei dati

- La tabella considerata fin qui rispetta la **TFN**!

Dirigente	<u>Progetto</u>	<u>Sede</u>
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Marte	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

- **DF1.** Progetto Sede → Dirigente

- **DF2.** Dirigente → Sede

DF1: Progetto Sede è una chiave →
Condizione 1 soddisfatta!

DF2: Sede è parte di una chiave →
Condizione 2 soddisfatta!

Se è già in TFN, **non è necessaria alcuna normalizzazione!**
Putroppo le ridondanze sui dati restano ...

Normalizzazione dei dati

CONFRONTO TRA TFN e FNBC

(SVANTAGGI) La TFN è **meno restrittiva** della FNBC

- **Tollera alcune ridondanze** ed anomalie sui dati.
- **Certifica meno lo qualità** dello schema ottenuto.

(VANTAGGI) La TFN è **sempre ottenibile**, qualsiasi sia la tabella

- **COME? Algoritmo di normalizzazione** in TFN!

Normalizzazione dei dati

ALGORITMO DI NORMALIZZAZIONE IN TERZA FORMA NORMALE (TFN)

Normalizzazione dei dati

TERZA FORMA NORMALE (TFN)

Una tabella r è in **terza forma normale** se per ogni dipendenza funzionale $X \rightarrow A$ (non banale) dello schema, **almeno una delle seguenti condizioni è verificata:**

- X è una superchiave di r
- A appartiene ad almeno una chiave K di r

Normalizzazione dei dati

DIPENDENZA FUNZIONALE BANALE

Una dipendenza funzionale $X \rightarrow Y$ si dice **banale** se Y è contenuto in X .

ESEMPI:

- Impiegato Progetto \rightarrow Impiegato
- Impiegato Progetto Sede \rightarrow Impiegato Progetto

Questo genere di dipendenze funzionali non ci interessano, e non le conseriamo come tali nel resto della trattazione ...

Normalizzazione dei dati

Data una relazione r con schema $R(X)$ non in TFN, **normalizzare in TFN** vuol dire: decomporre r nelle relazioni r_1, r_2, \dots, r_n , garantendo che:

- Ogni r_i ($1 \leq i \leq n$) è in TFN.
- La decomposizione è senza perdite. $r_1 \bowtie r_2 \bowtie \dots \bowtie r_n = r$
- La decomposizione conserva tutte le dipendenze F definite sullo schema $R(X)$ di partenza.

Normalizzazione dei dati

Ad esempio, data la relazione: **R(ACDGMPRS)**, con dipendenze funzionali:

$F = \{M \rightarrow RSDG, MS \rightarrow CD, G \rightarrow R, D \rightarrow S, S \rightarrow D, MPD \rightarrow AM\}$

Qual è la sua decomposizione in 3FN?

In molti casi, la decomposizione **non è così intuitiva** ...

Normalizzazione dei dati

IDEA alla base dell'algoritmo di normalizzazione:

- **Semplificare l'insieme di dipendenze F** , rimuovendo quelle non necessarie, e trasformando ogni dipendenza in modo che nella parte destra compaia un singolo attributo.
- **Raggruppare gli attributi coinvolti nelle stesse dipendenze**, e costruire le tabelle corrispondenti.
- **Assicurarsi che almeno una delle tabella prodotta contenga la chiave** della tabella originaria.

Normalizzazione dei dati

IMPLICAZIONE FUNZIONALE

Dato un insieme di dipendenze funzionali F , ed una dipendenza funzionale f , diremo che **F implica f** se **ogni tabella che soddisfa F soddisfa anche f** .

F : {Impiegato \rightarrow Livello, Livello \rightarrow Stipendio}

f : Impiegato \rightarrow Stipendio

In questo caso, **F implica f** ? **SI!**

Dim. Devo dimostrare che se in una tabella sono vere entrambe le dipendenze funzionali di F , allora vale anche la dipendenza funzionale f ...

Normalizzazione dei dati

IMPLICAZIONE FUNZIONALE

Dato un insieme di dipendenze funzionali F , ed una dipendenza funzionale f , diremo che **F implica f** se **ogni tabella che soddisfa F soddisfa anche f** .

F : {Impiegato \rightarrow Livello, Impiegato \rightarrow Stipendio}

f : Livello \rightarrow Stipendio

In questo caso, **F implica f ? NO!**

<u>Impiegato</u>	Livello	Stipendio
Neri	4	13400
Rossi	4	15000

Normalizzazione dei dati

CHIUSURA DI UNA DIPENDENZA FUNZIONALE

Dato uno schema $R(U)$, con un insieme di dipendenze F . Sia X un insieme di attributi contenuti in U . Si definisce la **chiusura di X rispetto ad F** (X_F^+) come l'insieme degli attributi che dipendono funzionalmente da X :

$$X_F^+ = \{A \mid A \in U \text{ e } F \text{ implica } X \rightarrow A\}$$

Normalizzazione dei dati

Esempio (facile). Siano:

$R=(ABCDE)$

$F=\{A \rightarrow B, A \rightarrow C\}$

Vogliamo conoscere la chiusura di A: A^+_F

$A^+_F=\{B,C\}$

Normalizzazione dei dati

Esempio (facile). Siano:

$R=(ABCDE)$

$F=\{A \rightarrow B, A \rightarrow C, C \rightarrow D\}$

Vogliamo conoscere la chiusura di A: A^+_F

$A^+_F=\{B,C,D\}$

Normalizzazione dei dati

INPUT: **X** (attributi) e **F** (dipendenze)

OUTPUT: La chiusura di **X** rispetto ad **F**: X_F^+

1) $X_F^+ = X$.

2) Per ogni dipendenza $f: Y \rightarrow A$ in **F**

$$\text{Se } (Y \subseteq X_F^+) \wedge (A \notin X_F^+) \Rightarrow X_F^+ = X_F^+ \cup \{A\}$$

3) Ripeti il passo 2 finchè non è possibile aggiungere nuovi elementi in X_F^+ .

Normalizzazione dei dati

Esempio. Siano:

$R=(ABCDE)$

$F=\{A \rightarrow B, BC \rightarrow D, B \rightarrow E, E \rightarrow C\}$

$A^+_F=\{A\}$

$A^+_F=\{A,B\}$ // con $f: A \rightarrow B$

$A^+_F=\{A,B,E\}$ // con $f: B \rightarrow E$

$A^+_F=\{A,B,E,C\}$ // con $f: E \rightarrow C$

$A^+_F=\{A,B,E,C,D\}$ // con $f: BC \rightarrow D$

Normalizzazione dei dati

D. Come verificare se F implica f: $X \rightarrow Y$?

1. Calcolare la **chiusura** X_F^+

1. Se **Y** appartiene ad X_F^+ , allora **F implica f**.

ESEMPIO

$F = \{A \rightarrow B, BC \rightarrow D, B \rightarrow E, E \rightarrow C\}$ f: $A \rightarrow E$

$A_F^+ = \{A, B, E, C, D\}$

.... Quindi F implica $A \rightarrow E$

Normalizzazione dei dati

Data una tabella con schema $R(U)$, l'algoritmo per determinare la chiusura X_F^+ può essere usato anche per verificare se X è una superchiave di R .

...COME?

Dato uno schema $R(U)$, con un insieme F di dipendenze funzionali, allora:

**un insieme di attributi K è una (super)chiave di $R(U)$ se:
 F implica $K \rightarrow U$.**

Normalizzazione dei dati

Dato uno schema $R(U)$, con un insieme F di dipendenze funzionali, allora:

**un insieme di attributi K è una (super)chiave di $R(U)$
se F implica $K \rightarrow U$.**

ESEMPIO

$R=(ABCDE)$ $F=\{A \rightarrow B, BC \rightarrow D, B \rightarrow E, E \rightarrow C\}$

Se A è una chiave allora F implica $A \rightarrow ABCDE$.

$A^+_F = \{A, B, E, C, D\}$ quindi A è una chiave!

Normalizzazione dei dati

INSIEMI DI DIPENDENZE EQUIVALENTI

Dati due insiemi di dipendenze funzionali F_1 ed F_2 , essi si dicono **equivalenti** se F_1 implica ciascuna dipendenza di F_2 e viceversa.

ESEMPIO

$$F = \{A \rightarrow B, AB \rightarrow C\} \quad F_1 = \{A \rightarrow B, A \rightarrow C\}$$

F e F_1 sono equivalenti! Occorre fare 4 verifiche ...

Normalizzazione dei dati

INSIEMI DI DIPENDENZE NON RIDONDANTI

Dato un insieme di dipendenze funzionali F definito su uno schema $R(U)$, esso si dice **non ridondante** se non esiste una dipendenza f di F tale che $F - \{f\}$ implica f .

ESEMPIO

$F = \{A \rightarrow B, AB \rightarrow C, A \rightarrow C\}$

F è ridondante perchè: $F - \{A \rightarrow C\}$ implica $A \rightarrow C$!

Normalizzazione dei dati

INSIEMI DI DIPENDENZE RIDOTTE

Dato un insieme di dipendenze funzionali F definito su uno schema $R(U)$, esso si dice **ridotto** se (1) non è ridondante, e (2) non è possibile ottenere un insieme F' equivalente eliminando attributi dai primi membri di una o più dipendenze di F .

ESEMPIO

$$F = \{A \rightarrow B, AB \rightarrow C\}$$

F **NON è ridotto** perchè B può essere eliminato da $AB \rightarrow C$ e si ottiene ancora un insieme F_2 equivalente ad F !

Normalizzazione dei dati

Dato uno schema $R(U)$ con insieme di dipendenze F , per trovare una **copertura ridotta di F** si procede in **tre passi**:

STEP 1. Sostituire F con F_1 , che ha tutti i **secondi membri composti da un singolo attributo**.

$M \rightarrow RSDG, MS \rightarrow CD, G \rightarrow R, D \rightarrow S, S \rightarrow D, MPD \rightarrow AM$

$F_1 = \{ M \rightarrow R, M \rightarrow S, M \rightarrow D, M \rightarrow G, MS \rightarrow C, MS \rightarrow D, G \rightarrow R, D \rightarrow S, S \rightarrow D, MPD \rightarrow A, MPD \rightarrow M \}$

Normalizzazione dei dati

STEP 2. Eliminare gli attributi estranei.

Supponiamo di avere $F=\{AB \rightarrow C, A \rightarrow B\}$, e calcoliamo A_F^+

$$A_F^+ = A$$

$$A_F^+ = AB \quad // \text{ da } A \rightarrow B$$

$$A_F^+ = ABC \quad // \text{ da } AB \rightarrow C$$

C dipende solo da A, quindi l'attributo B in $AB \rightarrow C$ puo' essere eliminato preservando l'uguaglianza!

$$F_1 = \{A \rightarrow C, A \rightarrow B\}$$

Normalizzazione dei dati

STEP 2. Eliminare gli attributi estranei.

D. In generale, se ho una dipendenza funzionale del tipo: **$AX \rightarrow B$** ,
come faccio a stabilire se l'attributo A può essere eliminato
preservando l'uguaglianza?

R. Si calcola **X^+** e si verifica se esso include **B**, nel qual caso A può
essere eliminato dalla dipendenza!

Normalizzazione dei dati

STEP 3. Eliminare le dipendenze non necessarie.

Supponiamo di avere $F = \{B \rightarrow C, B \rightarrow A, C \rightarrow A\}$:

$B \rightarrow A$ è **ridondante**, in quanto bastano le dipendenze $B \rightarrow C$, e $C \rightarrow A$ per capire che A dipende da B !

Formalmente, dovrei dimostrare che:

$F - \{B \rightarrow A\}$ implica $\{B \rightarrow A\}$ quindi, verificare che: $B^+_{F - \{B \rightarrow A\}}$ **contiene A ...**

Normalizzazione dei dati

STEP 3. Eliminare le ridondanze non necessarie.

D. In generale, **come posso stabilire se la dipendenza del tipo $X \rightarrow A$ è ridondante?**

R. Si elimina da F , si calcola $X^+_{F-\{X \rightarrow A\}}$, e **si verifica se tale insieme include ancora A** . Nel caso lo includa, si elimina la dipendenza funzionale $X \rightarrow A$.

Normalizzazione dei dati

Dati $R(U)$, ed un insieme di dipendenze F , **l'algoritmo di normalizzazione in terza forma normale** procede come segue:

STEP 1. Costruire una copertura ridotta F_1 di F .

$F = \{M \rightarrow RSDG, MS \rightarrow CD, G \rightarrow R, D \rightarrow S, S \rightarrow D, MPD \rightarrow AM\}$

\equiv

$F_1 = \{M \rightarrow D, M \rightarrow G, M \rightarrow C, G \rightarrow R, D \rightarrow S, S \rightarrow D, MP \rightarrow A\}$

Normalizzazione dei dati

STEP 2. Decomporre F_1 nei sottoinsiemi $F_1^{(1)}, F_1^{(2)}, \dots F_1^{(n)}$: ad ogni sottoinsieme appartengono dipendenze con gli stessi lati sinistri.

$$F_1^{(1)} = \{M \rightarrow D, M \rightarrow G, M \rightarrow C\}$$

$$F_1^{(2)} = \{G \rightarrow R\}$$

$$F_1^{(3)} = \{D \rightarrow S\}$$

$$F_1^{(4)} = \{S \rightarrow D\}$$

$$F_1^{(5)} = \{MP \rightarrow A\}$$

Normalizzazione dei dati

STEP 3. Se due o più lati sinistri delle dipendenze si implicano a vicenda, si fondono i relativi insiemi.

$$F_1^{(1)} = \{M \rightarrow D, M \rightarrow G, M \rightarrow C\}$$

$$F_1^{(2)} = \{G \rightarrow R\}$$

$$\begin{array}{l} F_1^{(3)} = \{D \rightarrow S\} \\ F_1^{(4)} = \{S \rightarrow D\} \end{array} \longrightarrow F_1^{(3)} = \{D \rightarrow S, S \rightarrow D\}$$

$$F_1^{(5)} = \{MP \rightarrow A\}$$

Normalizzazione dei dati

STEP 3. Trasformare ciascun $F_1^{(i)}$ in una tabella $R^{(i)}$ con gli attributi contenuti in ciascuna dipendenza.

Il lato sinistro diventa la chiave della relazione.

$F_1^{(1)} = \{M \rightarrow D, M \rightarrow G, M \rightarrow C\}$: $R^{(1)}(\underline{M}DGC)$

$F_1^{(2)} = \{G \rightarrow R\}$: $R^{(2)}(\underline{G}R)$

$F_1^{(3)} = \{D \rightarrow S, S \rightarrow D\}$: $R^{(3)}(\underline{S}D)$

$F_1^{(4)} = \{MP \rightarrow A\}$: $R^{(4)}(\underline{M}PA)$

Normalizzazione dei dati

STEP 5. Se nessuna relazione $R^{(i)}$ così' ottenuta contiene una chiave K di $R(U)$, **inserire una nuova tabella $R^{(n+1)}$** contenente gli attributi della chiave.

Nel nostro caso, la chiave è costituita da: (MP).

$R^{(1)}(\text{MDGC})$ $R^{(2)}(\text{GR})$ $R^{(3)}(\text{SD})$ **$R^{(4)}(\text{MPA})$**

$R^{(4)}(\text{MPA})$ contiene la chiave \rightarrow non c'è necessità di aggiungere altre tabelle!

Normalizzazione dei dati

In conclusione, data la relazione: **R(MGCRDSPA)**, con dipendenze funzionali:

$F = \{M \rightarrow RSDG, MS \rightarrow CD, G \rightarrow R, D \rightarrow S, S \rightarrow D, MPD \rightarrow AM\}$

La sua decomposizione in 3FN è la seguente:

$R^{(1)}(MDGC)$ $R^{(2)}(GR)$ $R^{(3)}(SD)$ $R^{(4)}(MPA)$

Normalizzazione dei dati

Esempio: $R(ABCDE)$ $F=\{C \rightarrow AB, BC \rightarrow DE, D \rightarrow B\}$

STEP (1.a) Ridurre F. (semplificare parte destra delle dipendenze)

$$F_1 = \{C \rightarrow A, C \rightarrow B, BC \rightarrow D, BC \rightarrow E, D \rightarrow B\}$$

STEP (1.b) Ridurre F. (semplificare parte sinistra delle dipendenze)

Calcolo $C^+_{F_1} = \{C, A, B, D, E\}$, e noto che include D ed E!

$$F_2 = \{C \rightarrow A, C \rightarrow B, C \rightarrow D, C \rightarrow E, D \rightarrow B\}$$

Normalizzazione dei dati

$$F_2 = \{C \rightarrow A, C \rightarrow B, C \rightarrow D, C \rightarrow E, D \rightarrow B\}$$

STEP (1.c) Ridurre F. (rimuovere dipendenze)

Che succede se elimino $C \rightarrow B$?

$$F_3 = \{C \rightarrow A, C \rightarrow D, C \rightarrow E, D \rightarrow B\}$$

STEP (2) Decomporre F (in insiemi di dipendenze con lo stesso lato sx)

$$F_{31} = \{C \rightarrow A, C \rightarrow D, C \rightarrow E\}$$

$$F_{32} = \{D \rightarrow B\}$$

Normalizzazione dei dati

$F_{31} = \{C \rightarrow A, C \rightarrow D, C \rightarrow E\}$

$F_{32} = \{D \rightarrow B\}$

STEP (3) Fondere gli insiemi. (le cui parti sinistre si implicano)

STEP (4) Costruire le relazioni associate.

$R_1(\underline{C}ADE)$

$R_2(\underline{D}B)$

STEP (5) Verificare esistenza della chiave.

$C^+ = \{A, B, C, D, E\} \rightarrow$ quindi C è una chiave dello schema r

Normalizzazione dei dati

Perchè si chiama **Terza Forma Normale (TFN)**?

- **Prima Forma Normale (PFN)** → si suppone sempre rispettata
- **Seconda Forma Normale (SFN)** → variante debole della TFN.

Procedendo per gradi, si dovrebbe **normalizzare in PFN, poi in SFN, e quindi in TFN.**

Normalizzazione dei dati

Una relazione r con schema $R(U)$ è in **Seconda Forma Normale (SFN)** quando **NON** presenta **dipendenze parziali**, della forma: $Y \rightarrow A$, dove:

- Y è un sottoinsieme **proprio** della chiave
- A è un qualsiasi sottoinsieme di $R(U)$

IMPIEGATO(Impiegato, Stipendio, Progetto, Budget)

Impiegato \rightarrow Stipendio

Progetto \rightarrow Budget

DIPENDENZA PARZIALE!

Non in SFN
Non in TFN

Normalizzazione dei dati

Una tabella con schema $R(U)$ è in **Seconda Forma Normale (SFN)** quando **NON** presenta **dipendenze parziali**, della forma: $Y \rightarrow A$, dove:

- Y è un sottoinsieme proprio della chiave
- A è un qualsiasi sottoinsieme di $R(U)$

IMPIEGATO(Impiegato, Categoria, Stipendio)

Impiegato \rightarrow Categoria

Categoria \rightarrow Stipendio

IN SFN
MA NON IN TFN!

Normalizzazione dei dati

- Una tabella con schema $R(U)$ è in **Quarta Forma Normale (4FN)** se non presenta dipendenze multivalore non banali diverse da una chiave della tabella. Es. $X \twoheadrightarrow Y$ $X \twoheadrightarrow Z$
- Una tabella con schema $R(U)$ è in **Quinta Forma Normale (5FN)** se non è possibile decomporre ulteriormente la tabella senza perdere informazioni.