

A2 - 3.Progettazione e normalizzazione di un DB relazionale - normalizzazione

pag. A47 - A55 (parte prima)

normalizzazione

Prendiamo in esame il seguente schema di relazione, del quale riportiamo anche un esempio di tabella, istanza di relazione dello schema dato:

Incarichi (impiegato, livello, progetto, budget, anno, ruolo, costo_orario);

La chiave primaria è formata dai due attributi *impiegato* e *progetto*.



| <u>impiegato</u> | <u>livello</u> | <u>progetto</u> | budget | anno | ruolo | costo_orario |
|------------------|----------------|-----------------|--------|------|--------------|--------------|
| Rossi | B1 | Alfa | 4 | 2011 | tecnico | 10 |
| Bianchi | C2 | Beta | 6 | 2012 | progettista | 25 |
| Bianchi | C2 | Delta | 6 | 2011 | progettista | 25 |
| Neri | D2 | Delta | 6 | 2011 | responsabile | 40 |
| Neri | D2 | Beta | 6 | 2012 | consulente | 30 |
| Neri | D2 | Alfa | 4 | 2011 | consulente | 30 |
| Verdi | D1 | Alfa | 4 | 2011 | responsabile | 40 |
| Verdi | D1 | Delta | 6 | 2011 | consulente | 30 |
| Grigi | C4 | Delta | 6 | 2011 | tecnico | 10 |
| Grigi | C4 | Beta | 6 | 2012 | responsabile | 40 |

normalizzazione

Lo schema proposto non risulta ben progettato in quanto, effettuando operazioni di inserimento, cancellazione ed aggiornamento, possono presentarsi anomalie sui dati registrati.

Vediamo più nel dettaglio le problematiche che possono presentarsi.

| impiegato | livello | progetto | budget | anno | ruolo | costo_orario |
|-----------|---------|----------|--------|------|--------------|--------------|
| Rossi | B1 | Alfa | 4 | 2011 | tecnico | 10 |
| Bianchi | C2 | Beta | 6 | 2012 | progettista | 25 |
| Bianchi | C2 | Delta | 6 | 2011 | progettista | 25 |
| Neri | D2 | Delta | 6 | 2011 | responsabile | 40 |
| Neri | D2 | Beta | 6 | 2012 | consulente | 30 |
| Neri | D2 | Alfa | 4 | 2011 | consulente | 30 |
| Verdi | D1 | Alfa | 4 | 2011 | responsabile | 40 |
| Verdi | D1 | Delta | 6 | 2011 | consulente | 30 |
| Grigi | C4 | Delta | 6 | 2011 | tecnico | 10 |
| Grigi | C4 | Beta | 6 | 2012 | responsabile | 40 |

normalizzazione

Anomalie di Inserimento: non è possibile inserire i dati di un impiegato (*impiegato*, *livello*) se questo non è coinvolto in almeno un progetto; non è possibile inserire un nuovo progetto con i relativi attributi (*progetto*, *budget*, *anno*) se non coinvolge almeno un impiegato.

Questi vincoli derivano dal fatto che gli attributi *impiegato* e *progetto* sono parte della chiave primaria (vincolo di *integrità sull'entità*).

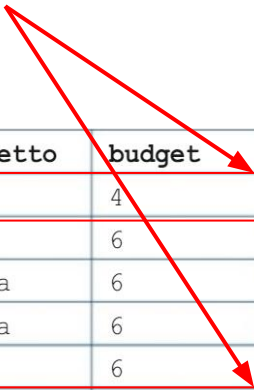


| impiegato | livello | progetto | budget | anno | ruolo | costo_orario |
|-----------|---------|----------|--------|------|--------------|--------------|
| Rossi | B1 | Alfa | 4 | 2011 | tecnico | 10 |
| Bianchi | C2 | Beta | 6 | 2012 | progettista | 25 |
| Bianchi | C2 | Delta | 6 | 2011 | progettista | 25 |
| Neri | D2 | Delta | 6 | 2011 | responsabile | 40 |
| Neri | D2 | Beta | 6 | 2012 | consulente | 30 |
| Neri | D2 | Alfa | 4 | 2011 | consulente | 30 |
| Verdi | D1 | Alfa | 4 | 2011 | responsabile | 40 |
| Verdi | D1 | Delta | 6 | 2011 | consulente | 30 |
| Grigi | C4 | Delta | 6 | 2011 | tecnico | 10 |
| Grigi | C4 | Beta | 6 | 2012 | responsabile | 40 |

normalizzazione

Anomalie di Cancellazione: la cancellazione della riga identificata dal valore della chiave primaria (Neri, Alfa) è diversa dalla cancellazione della riga identificata dal valore della chiave primaria (Rossi, Alfa).

Nel primo caso si conservano sia i dati del progetto Alfa sia quelli dell'impiegato Neri, mentre nel secondo caso i dati dell'impiegato Rossi vanno perduti.



| impiegato | livello | progetto | budget | anno | ruolo | costo_orario |
|-----------|---------|----------|--------|------|--------------|--------------|
| Rossi | B1 | Alfa | 4 | 2011 | tecnico | 10 |
| Bianchi | C2 | Beta | 6 | 2012 | progettista | 25 |
| Bianchi | C2 | Delta | 6 | 2011 | progettista | 25 |
| Neri | D2 | Delta | 6 | 2011 | responsabile | 40 |
| Neri | D2 | Beta | 6 | 2012 | consulente | 30 |
| Neri | D2 | Alfa | 4 | 2011 | consulente | 30 |
| Verdi | D1 | Alfa | 4 | 2011 | responsabile | 40 |
| Verdi | D1 | Delta | 6 | 2011 | consulente | 30 |
| Grigi | C4 | Delta | 6 | 2011 | tecnico | 10 |
| Grigi | C4 | Beta | 6 | 2012 | responsabile | 40 |

normalizzazione

Anomalie di Aggiornamento: (*Inconsistenza Potenziale*): l'aggiornamento del livello di inquadramento dell'impiegato Rossi implica la modifica di una sola riga, mentre quello dell'impiegato Neri implica la modifica di tre righe.

Dover modificare tre righe per l'aggiornamento di un dato potrebbe potenzialmente rendere inconsistente il database, laddove i dati venissero aggiornati in una riga e non in un'altra (o aggiornati in modo differente).



| impiegato | livello | progetto | budget | anno | ruolo | costo_orario |
|-----------|---------|----------|--------|------|--------------|--------------|
| Rossi | B1 | Alfa | 4 | 2011 | tecnico | 10 |
| Bianchi | C2 | Beta | 6 | 2012 | progettista | 25 |
| Bianchi | C2 | Delta | 6 | 2011 | progettista | 25 |
| Neri | D2 | Delta | 6 | 2011 | responsabile | 40 |
| Neri | D2 | Beta | 6 | 2012 | consulente | 30 |
| Neri | D2 | Alfa | 4 | 2011 | consulente | 30 |
| Verdi | D1 | Alfa | 4 | 2011 | responsabile | 40 |
| Verdi | D1 | Delta | 6 | 2011 | consulente | 30 |
| Grigi | C4 | Delta | 6 | 2011 | tecnico | 10 |
| Grigi | C4 | Beta | 6 | 2012 | responsabile | 40 |

normalizzazione

Le anomalie evidenziate nell'esempio precedente sono una diretta conseguenza della presenza, nello stesso schema di relazione, di campi relativi sia agli impiegati, sia ai progetti (la chiave primaria infatti è formata dalla coppia di attributi: *impiegato* e *progetto*), oltre che della **ridondanza dei dati**, cioè del fatto che lo stesso dato è presente nella tabella replicato più volte.

normalizzazione

La **Normalizzazione** rappresenta un processo formalizzato di **analisi di ognuno degli schemi relazionali** (cioè di ogni tabella) di un determinato modello relazionale, basato sulle loro *dipendenze funzionali* e sulle *chiavi (primarie o candidate)* per:

1. minimizzare la **ridondanza**,
2. minimizzare, di conseguenza, le possibili **anomalie di inserimento, cancellazione e aggiornamento**.

Schemi di relazione inadeguati che non soddisfano certe condizioni (i cosiddetti test di forma normale), vengono decomposti in schemi di relazione più piccoli che superano i test e pertanto possiedono le proprietà desiderate.

Alla fine **ogni tabella derivata da questi schemi corrisponde ad un singolo oggetto della realtà di interesse**.

normalizzazione

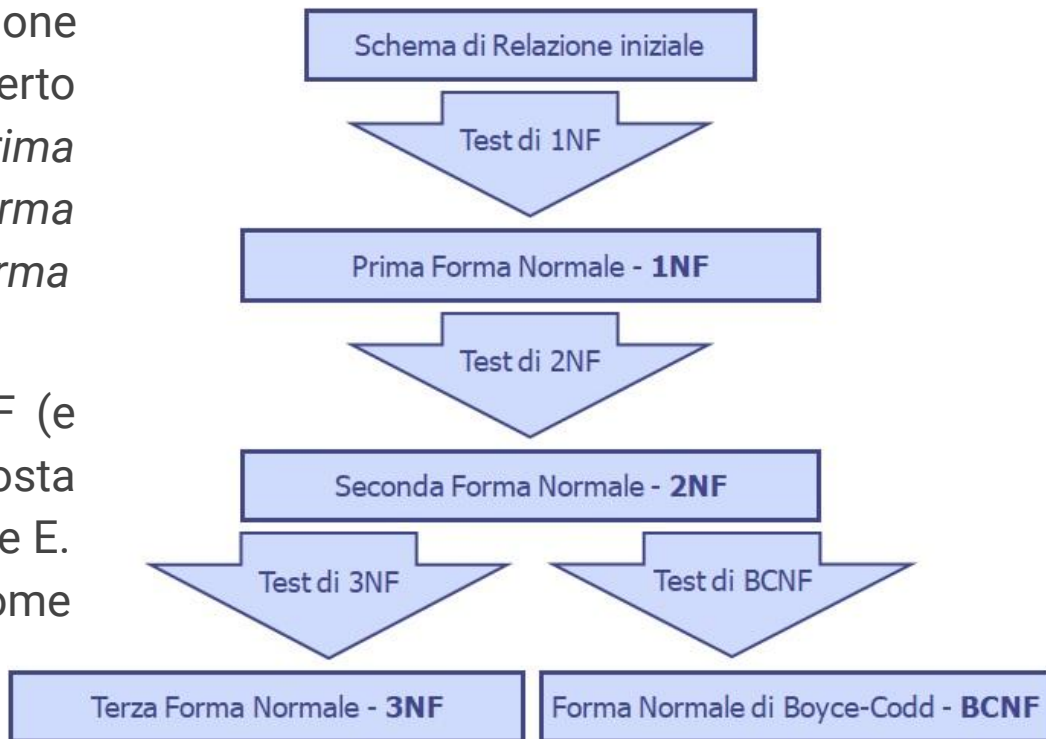
La **normalizzazione** va intesa dunque come **tecnica di verifica** dei risultati della progettazione logica (cioè del modello relazionale), **non costituisce in alcun modo una metodologia di progettazione**.

Il processo di normalizzazione, così come è stato inizialmente proposto da E. F. Codd, avviene a livelli crescenti di normalizzazione a cui corrispondono diverse *forme normali* dello schema di relazione considerato, ciascuna con caratteristiche (e test di forma normale da verificare) ben definite.

normalizzazione

Partendo da uno schema di relazione iniziale rappresentativo di un certo modello relazionale, si passa dalla *prima forma normale (1NF)* alla *seconda forma normale (2NF)*, quindi alla *terza forma normale (3NF)*.

Una definizione più restrittiva di 3NF (e alternativa ad essa) è stata proposta solo successivamente da R. F. Boyce e E. F. Codd, ed è conosciuta appunto come **BCNF** (*Boyce Codd Normal Form*).



normalizzazione

Sia dato uno schema di relazione $Rel(A_1, A_2, \dots, A_n)$, una **dipendenza funzionale** tra due insiemi di attributi X e Y dello schema Rel , indicata con $X \rightarrow Y$, specifica un vincolo sulle tuple di qualunque istanza di relazione derivata dallo schema Rel ; si dice in questo caso che **X determina funzionalmente Y** , o anche che **X è un determinante per Y** .

Quando risulta ad es. $A_i \rightarrow A_j$ (dove i e j $\in [1, 2, \dots, n]$) significa che **per ogni coppia di tuple appartenenti ad una relazione qualunque derivata dallo schema Rel , se queste hanno gli stessi valori per l'attributo A_i devono necessariamente avere gli stessi valori anche per l'attributo A_j** ; quindi il valore assunto dall'attributo A_j viene *univocamente* (o *funzionalmente*) determinato dal valore assunto dall'attributo A_i .

NB: Questo **non implica** che se due tuple hanno valori differenti per l'attributo A_i devono necessariamente avere valori differenti anche per l'attributo A_j .

normalizzazione

Facciamo un esempio pratico; sia dato il seguente schema di relazione:

Studiante (*Matricola*, *Nominativo*, *DataNascita*, *Residenza*, *Età*, *CorsoLaurea*, *AnnoCorso*)

Esiste di certo la seguente dipendenza funzionale: $DataNascita \rightarrow Et\grave{a}$; appare chiaro infatti come per qualunque relazione derivata da questo schema (cioè qualunque sia la tabella di studenti che riporta sulle intestazioni di colonna gli attributi dello schema dato), se due studenti qualsiasi hanno lo stesso valore per il campo *DataNascita*, necessariamente devono avere lo stesso valore anche per il campo *Età*, dal momento che il valore assunto dall'attributo *Età* viene univocamente determinato dal valore assunto dall'attributo *DataNascita*.

Questo non implica, ovviamente, che due studenti possano avere la stessa età, pur essendo nati in giorni differenti.

normalizzazione

Vediamo altri esempi di dipendenze funzionali per i seguenti schemi di relazione:

Contribuente (CodFiscale, *Nominativo*, *AnnoNascita*, *Età*, *Residenza*, *Imponibile*, *Tassa*);

risulta: **Imponibile** → **Tassa**.

Ricovero (CodPaziente, DataRicovero, *Reparto*, *Diagnosi*, *Anamnesi*, *Note*);

risulta: **Diagnosi** → **Reparto**.

Immobile (CodImmobile, *Comune*, *MQuadri*, *Indirizzo*, *NomProprietario*, *IMU*);

risulta: (**Comune**, **MQuadri**) → **IMU**.

Dato che ogni comune può determinare l'aliquota ICI di un immobile in base alla grandezza dell'immobile stesso.

Incarichi (impiegato, *livello*, progetto, *budget*, *anno*, *ruolo*, *costo_orario*);

risultano verificate:

impiegato → **livello**: (ogni impiegato ha un proprio livello di inquadramento);

progetto → **budget** e **progetto** → **anno**: (ogni progetto ha un unico budget e un unico anno di riferimento);

(**impiegato**, **progetto**) → **ruolo**: (ad ogni impiegato per un certo progetto è assegnato un ruolo specifico);

ruolo → **costo_orario**: (per ogni ruolo è previsto un determinato costo orario).

normalizzazione

Appare chiaro che **il fatto che in uno schema di relazione risulti $A_i \rightarrow A_j$ non implica necessariamente che risulti anche il contrario $A_j \rightarrow A_i$.**

Consideriamo ad es. lo schema *Studente* visto prima.

Non è vero che $Età \rightarrow DataNascita$, anche se potrebbe esserci qualche tabella derivata da questo schema in cui effettivamente esistono due studenti nati lo stesso giorno, quindi con la stessa età anagrafica e la stessa data di nascita.

Analogamente, per lo schema *Ricovero* non si può certo dire che $Reparto \rightarrow Diagnosi$, visto che nello stesso reparto possono essere ricoverati pazienti con patologie (e dunque anche diagnosi) differenti, anche se queste prevedono metodologie di analisi o di cura analoghi (per cui è giustificato il ricovero nello stesso reparto).

normalizzazione

Quella di dipendenza funzionale è dunque una proprietà caratteristica dello schema di relazione determinata sulla base del significato degli attributi dello schema stesso.

Come già visto per la chiave primaria, non può essere semplicemente ricavata dai dati registrati in una determinata tabella di quello schema, ma deve essere dedotta dal progettista del database dopo un'attenta analisi della semantica degli attributi previsti per lo schema di relazione in questione.

Anche se a prima vista si può pensare che *Testo* → *Insegnamento*, non è possibile confermare tale dipendenza per ogni istanza di relazione dello stesso schema.

Basta un solo controesempio per escludere una dipendenza; ad es. il prof. Verdi potrebbe decidere di utilizzare il testo di Elmasri per il corso di Strutture dati, il che dimostrerebbe l'infondatezza della precedente dipendenza funzionale.

| <u>Docente</u> | <u>Insegnamento</u> | <u>Testo</u> |
|----------------|---------------------|--------------|
| Bianchi | Strutture dati | Ullman |
| Bianchi | Gestione dati | Atzeni |
| Rossi | Sistemi e reti | Odom |
| Verdi | Strutture dati | Ullman |
| Insegna | | |



normalizzazione

Le dipendenze funzionali di uno schema di relazione rappresentano vincoli di integrità (o proprietà) dello schema stesso, nel senso che una qualsiasi istanza di relazione di quello schema non può che soddisfare tali dipendenze.

Riprendiamo ad es. lo schema visto all'inizio:

Incarichi (impiegato, livello, progetto, budget, anno, ruolo, costo_orario);

la proprietà “*Il livello salariale di ciascun impiegato è funzione del solo impiegato, indipendentemente dai progetti cui questi partecipa*”, implica il soddisfacimento della dipendenza funzionale:

impiegato \rightarrow livello.

Una qualsiasi istanza di relazione che non dovesse soddisfare tale proprietà, non sarebbe considerata corretta per lo schema dato.

normalizzazione

Sempre per lo stesso schema, la proprietà *“Il budget di ciascun progetto dipende dal solo progetto, indipendentemente dagli impiegati che vi partecipano”*, si traduce nella dipendenza funzionale:

progetto → **budget**.

Anche la proprietà *“In ogni progetto, ciascuno degli impiegati coinvolti può svolgere uno ed un solo ruolo”* può essere modellata per mezzo di una dipendenza funzionale:

(impiegato, progetto) → **ruolo**,

del resto questa dipendenza è anche conseguenza del fatto che gli attributi *impiegato* e *progetto* formano la chiave primaria dello schema di relazione.

Una eventuale tupla in cui ad es. l'impiegato Rossi collabora nel progetto Alfa sia come tecnico che come progettista non sarebbe valida perché violerebbe tale dipendenza.

normalizzazione

Vediamo ora qualche altra definizione utile.

Sia dato lo schema di relazione $Rel(A_1, A_2, \dots, A_n)$, su cui sia definita la seguente dipendenza funzionale: $X \rightarrow Y$; si parla di **dipendenza funzionale banale** se risulta: $Y \subseteq X$, ad es. $(Comune, MQuadri) \rightarrow Comune$.

Notiamo inoltre come in qualunque schema di relazione, **ogni chiave** (primaria o candidata che sia) **per definizione determina funzionalmente ciascun attributo (o insieme di attributi) dello schema stesso**.

Le dipendenze funzionali di questo tipo vengono indicate come **dipendenze funzionali fondamentali** per lo schema in questione.

normalizzazione

Come già visto in precedenza, dato uno schema di relazione $Rel(A_1, A_2, \dots, A_n)$, si definisce **superchiave** un qualunque insieme di attributi di Rel con la proprietà che nessuna coppia di tuple in una qualsiasi istanza di relazione derivata dallo schema dato possono assumere valori uguali per quegli attributi.

Una chiave K per lo stesso schema di relazione è una superchiave con la proprietà che la rimozione di un qualsiasi attributo da K fa sì che questa cessi di essere una superchiave; si tratta dunque di una superchiave minimale.

Se uno schema di relazione possiede più di una chiave, ognuna di esse è detta chiave **candidata**, mentre una sola di queste è arbitrariamente scelta come chiave **primaria**.

normalizzazione

Un attributo di uno schema di relazione si dice **attributo primo** (o *attributo chiave*) se fa parte di una qualche chiave (primaria o candidata), altrimenti se non è membro di nessuna chiave lo si definisce **attributo non primo** (o *attributo non chiave*).

Sia dato lo schema di relazione $Rel(A_1, A_2, \dots, A_n)$, supponiamo che esistano le seguenti dipendenze funzionali: $A_i \rightarrow A_k$ e $A_k \rightarrow A_j$, risulta allora anche: $A_i \rightarrow A_j$ e si dice che A_i *determina transitivamente* A_j ; in una situazione simile si può parlare di **dipendenza funzionale transitiva**.

Ad es. dato il seguente schema di relazione:

Contribuente (CodFiscale, Nominativo, AnnoNascita, Età, Residenza, Imponibile, Tassa);

risulta: **CodFiscale** \rightarrow **AnnoNascita**; **AnnoNascita** \rightarrow **Età** dunque **CodFiscale** \rightarrow **Età**.