

La colonna *AreaGeografica* è la parte della chiave primaria *Agente*, da cui dipende, possono quindi essere staccate per formare una nuova tabella *Agenti* con i dati degli agenti. Per completare la tabella *Agenti* aggiungiamo la colonna *Codice*, con i codici degli agenti, che definisce la sua chiave primaria. La tabella *Agenti* è in relazione, uno a molti, con la tabella *Vendite* mediante la chiave esterna *Agente* delle *Vendite* che memorizzerà il codice degli agenti che hanno fatto una vendita. In modo analogo, la dipendenza *Cliente* → *IndirizzoCliente* genera una nuova tabella dei *Clienti*, mentre da *Prodotto* → *PrezzoListino* nasce la tabella dei *Prodotti*. Lo schema relazionale in 2NF diventa quindi quello che segue.

Schema logico relazionale in 1NF



Schema logico relazionale in 2NF



Le colonne della tabella *Vendite* sono state anche rinominate eliminando il nome "Vendita" o "Venduta" che ormai è inutile (ad esempio *DataVendita* diventa semplicemente *Data* e *QuantitaVendita* diventa *Quantita*). La sua chiave primaria diventa quella semplice *Codice*, che non include più le chiavi esterne. Osserviamo infine che nello schema relazionale in 2NF compare una relazione molti a molti: un prodotto compare in molte vendite e una vendita contiene molti prodotti; questa relazione deve essere eliminata, come vedremo, usando la 3NF.



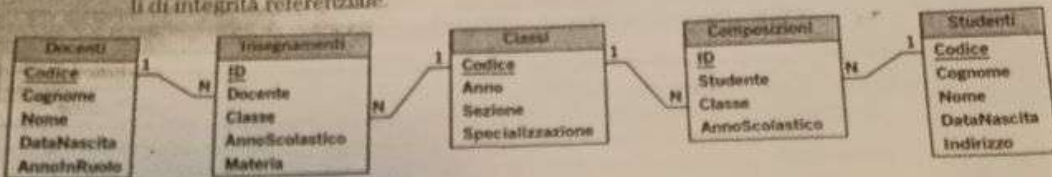
Una relazione molti a molti tra due tabelle comporta ancora delle ridondanze nei dati memorizzati. Possiamo accorgerci di queste ridondanze provando a inserire alcuni dati nella tabella *Vendite* in 2NF. Nei dati, per maggior chiarezza, abbiamo ancora usato le denominazioni di agenti e clienti e la descrizione dei prodotti invece che i loro codici.

Codice	Agente (denominaz.)	Cliente (denominaz.)	Prodotto (descriz.)	Data	Prezzo	PerSconto (in%)	Quantita	Importo totale
V-901	Rossi	Bianchi HW	Mouse	14/04/2008	11,15	7,0	250	2592,38
V-901	Rossi	Bianchi HW	Hard disk	14/04/2008	53,35	3,0	70	3622,47
V-901	Rossi	Bianchi HW	Video LCD	14/04/2008	266,00	5,0	15	3790,50
V-901	Rossi	Bianchi HW	Mouse	16/04/2008	11,15	5,0	190	2012,58
V-902	Verdi	Neri Spa	Mouse	16/04/2008	266,00	7,0	35	8658,30
V-902	Verdi	Neri Spa	Video LCD	16/04/2008	266,00	7,0
...

Il prezzo che compare in tabella è quello di vendita che deriva dal prezzo di listino che è già stato scontato dall'azienda. Sul prezzo di vendita però gli agenti hanno a loro disposizione un ulteriore sconto (che dipende dal cliente e dalla quantità venduta) che è memorizzato nella colonna *PerSconto*. Nella tabella non viene indicata l'IVA che dovrà essere aggiunta al momento dell'emissione della fattura. Una relazione molti a molti tra due tabelle deve essere trasformata, applicando la 2NF o la 3NF, introducendo una nuova **tabella intermedia che è legata alle altre due tabelle da due relazioni uno a molti**.

Esempio

Ripresentiamo lo schema del database della scuola e completiamolo elencando i suoi vincoli di integrità referenziale.



- V1. La chiave esterna *Docente* della tabella *Insegnamenti* è in relazione con la tabella *Docenti* mediante la chiave primaria *Codice*.
- V2. La chiave esterna *Classe* della tabella *Insegnamenti* è in relazione con la tabella *Classi* mediante la chiave primaria *Codice*.
- V3. La chiave esterna *Classe* della tabella *Composizioni* è in relazione con la tabella *Classi* mediante la chiave primaria *Codice*.
- V4. La chiave esterna *Studente* della tabella *Composizioni* è in relazione con la tabella *Studenti* mediante la chiave primaria *Codice*.

4 La normalizzazione

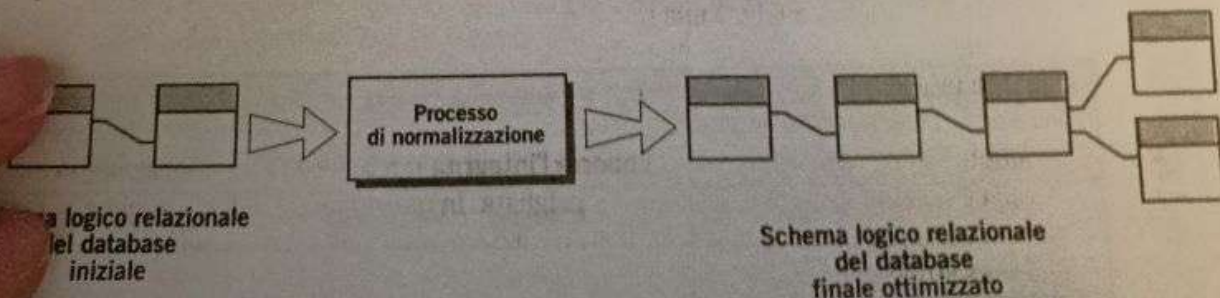
La normalizzazione (*normalization*) definisce un insieme di regole il cui obiettivo è quello di arrivare a uno schema logico relazionale corretto ovvero che rispetta le proprietà di un database. L'insieme delle regole della normalizzazione sono raggruppate in **forme normali** (*normal form*); ogni forma normale è ricordata mediante un numero intero, scritto anche come cifra, seguito dall'abbreviazione **NF** (iniziali di *Normal Form*). Ad esempio, la prima forma normale può essere ricordata come **1NF**. Le forme normali devono essere **rispettate nell'ordine**, a partire dalla **1NF**.



Uno schema relazionale che soddisfa una forma normale si dice che è in questa forma normale. Se un programmatore dice: "lo schema relazionale è nella **2NF**", significa che lo schema logico proposto soddisfa tutte le regole della **1NF** e anche le regole della **2NF**.

Le forme normali originali sono tre: la **1NF**, la **2NF** e la **3NF**, che sono state proposte dallo stesso Codd. La **3NF** è stata rivisitata e modificata in una forma più precisa dallo studioso Boyce; la forma alternativa alla **3NF** è ricordata come forma normale di Boyce e Codd e schematizzata con la sigla **BCNF** (*Boyce-Codd Normal Form*). Alle prime tre forme normali ne sono state aggiunte altre due, la **4NF** e la **5NF**, il cui utilizzo è poco frequente.

Le forme normali sono state proposte in anni in cui la fase di progetto non era ancora ben consolidata. Senza una fase di progettazione, le forme normali erano applicate nell'ordine (**1NF**, **2NF** e **3NF**) per trasformare uno schema logico iniziale con "errori" in uno schema relazionale ottimizzato.



sono presenti alcune di quelle che com-
dono, laddove presenti

112

una chiave dipendente solo da una parte della chiave primaria composta. La dipendenza che nella 2NF è la **dipendenza funzionale** (functional dependency) della teoria relazionale che viene usata in questo modo per rappresentare che i dati della **ColonnaA** sono associati a dati determinati, in ogni istante, a quelli della **ColonnaB** della medesima tabella.

ColonnaA → ColonnaB

Se la **ColonnaB** dipende (funzionalmente) dalla **ColonnaA**, viceversa si dice che i dati della **ColonnaA** "determinano" quelli della **ColonnaB** oppure che la **ColonnaA** è una **colonna determinante** (determinant) per la **ColonnaB**. Per semplicità abbiamo trasformato una sola colonna, tra **ColonnaA** e **ColonnaB** possono rappresentare gruppi di più colonne.

Per trasformare lo schema relazionale di una tabella che non è nella 2NF dobbiamo seguire i seguenti **passi operativi**:

1. Indicare le colonne (una o più) che dipendono da una parte della chiave primaria composta.
2. Creare una nuova tabella con queste colonne e la parte della chiave primaria da cui dipendono.
3. Mettere in relazione la nuova tabella usando la parte della chiave primaria come chiave estera.

L'applicazione della 2NF a una tabella permette di eliminare le colonne con dei dati ridondanti che vengono staccate e memorizzate in un'altra tabella. L'eliminazione dei dati ridondanti evita quindi anche le anomalie di aggiornamento.

ESEMPIO

Lo schema relazionale del database degli agenti di commercio in 1NF non rispetta la 2NF, perché molte colonne non chiave dipendono da una parte della chiave primaria composta (funzionalmente) da un **Agente**, l'**IndirizzoCliente** dipende da un **Cliente** e, infine, dipende così:

Agente → AreaGeografica
 Cliente → IndirizzoCliente
 Prodotto → PrezzoListino

Per evidenziare le dipendenze consideriamo i dati della tabella **Vendite** in 1NF e "stacciamo" le tre coppie di colonne dipendenti. Le colonne con le dipendenze sono anche riconoscibili perché sono quelle che **contengono dei dati ripetuti** oppure perché il loro distacco permette di **eliminare le ridondanze da altre colonne**.

Tabella Agenti

Agente (denominazione)	Area geografica
Rossi	Centro
Verdi	Sud

Tabella Clienti

Cliente (denominazione)	Indirizzo cliente
Bianchi HW	Via ...
Neri SpA	Piazza ...

Prodotto (descrizione)	Prezzo listino
Mouse	11,50
Hard disk	55,00
Video LCD	280,00

Tabella Prodotti

Codice vendita	Agente (denominazione)	Area geografica	Cliente (denominazione)	Indirizzo cliente	Prodotto (descrizione)	Prezzo listino	Data vendita
V-001	Rossi	Centro	Bianchi HW	Via ...	Mouse	11,50	14/04/2008
V-001	Rossi	Centro	Bianchi HW	Via ...	Hard disk	55,00	14/04/2008
V-001	Rossi	Centro	Bianchi HW	Via ...	Video LCD	280,00	14/04/2008
V-002	Verdi	Sud	Neri SpA	Piazza ...	Mouse	11,50	16/04/2008
V-002	Verdi	Sud	Neri SpA	Piazza ...	Video LCD	280,00	16/04/2008

Tabella Vendite

Terza forma normale

Una tabella si trova nella terza forma normale se rispetta la 2NF e tutte le colonne non chiave dipendono dalla chiave primaria.

La 3NF esige che una colonna non chiave possa dipendere da un'altra colonna non chiave, ma questa colonna non chiave deve dipendere solo dalla colonna o dalle colonne che formano la chiave primaria.

Alcune dipendenze possono essere eliminate **cancellando** dalla tabella la colonna che dipende dalle altre. Questa cancellazione si può fare quando i dati della colonna possono essere ricavati con un'elaborazione (ad esempio una somma, un prodotto ecc.) da quelli delle colonne non chiave. Negli altri casi, si applicano i **passi operativi** visti per la 2NF staccando le colonne che dipendono tra loro (e non dalla chiave primaria) che formeranno una nuova tabella.

Esempio

Nell'ultima versione dello schema logico in 2NF del database degli agenti di commercio, la tabella *Vendite* non rispetta la 3NF. Infatti, nella tabella *Vendite* possiamo riconoscere queste due dipendenze: nella prima, la colonna *ImportoTotale* dipende da *Prezzo*, *PerSconto* e *Quantità*, mentre nella seconda le colonne *Prezzo*, *PerSconto* e *Quantità* dipendono dalla colonna non chiave *Prodotto*.

Prezzo, *PerSconto*, *Quantità* → *ImportoTotale*
Prodotto → *Prezzo*, *PerSconto*, *Quantità*

La prima dipendenza si elimina **cancellando la colonna *ImportoTotale*** dalla tabella *Vendite*, perché i suoi dati si possono sempre ricavare da quelli delle colonne *Prezzo*, *PerSconto* e *Quantità* applicando questa espressione: $\text{Prezzo} \times \text{Quantità} \times (1 - \text{PerSconto}/100)$, per il calcolo dell'importo totale scontato. Per evidenziare la seconda dipendenza consideriamo i dati della tabella *Vendite* in 2NF e "stacciamo" le quattro colonne *Prodotto*, *Prezzo*, *PerSconto* e *Quantità*, che permettono di **eliminare i dati ripetuti** nelle colonne *Codice*, *Agente*, *Cliente* e *Data*.

Prodotto (descrizione)
Mouse
Hard disk
Video LCD
Mouse
Video LCD
...

Tabella
Dettagli

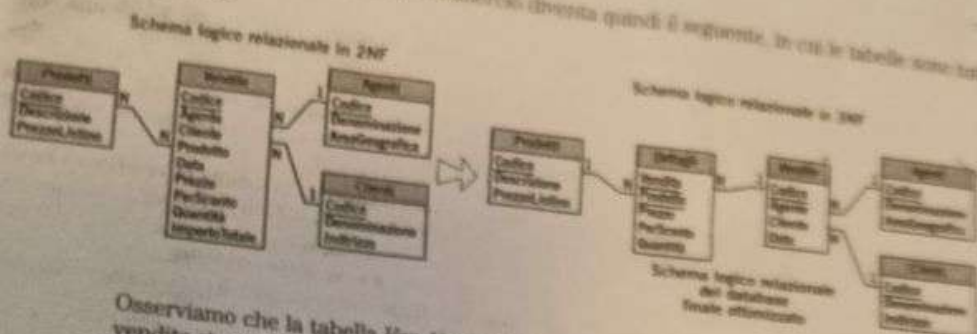
Prezzo	PerSconto (in %)	Quantità
11,15		
53,35	7,0	250
266,00	3,0	70
11,15	5,0	15
266,00	5,0	190
	7,0	35
...		

Codice	Agente (denominazione)	Cliente (denominazione)	Prodotto (descrizione)	Data	Prezzo	PerSconto (in %)	Quantità	Importo totale
V-901	Rossi	Bianchi HW	Mouse	14/04/2008	11,15	7,0	250	2500,00
V-901	Rossi	Bianchi HW	Hard disk	14/04/2008	53,35	3,0	70	3600,47
V-901	Rossi	Bianchi HW	Video LCD	16/04/2008	266,00	5,0	15	3700,50
V-902	Verdi	Neri Spa	Mouse	16/04/2008	11,15	5,0	190	2012,58
V-902	Verdi	Neri Spa	Video LCD	16/04/2008	266,00	7,0	35	8600,00
...								

Tabella *Vendite*

Per eliminare la seconda dipendenza creiamo una nuova tabella che contiene per ogni *Prodotto* venduto il *Prezzo*, la percentuale di sconto *PerSconto* e la *Quantità* e quindi i suoi dettagli di vendita. La nuova tabella *Dettagli* è in relazione uno a molti sia con la tabella *Vendite* sia con quella *Prodotti*, rispettivamente, con le chiavi esterne *Vendita* (con il codice delle vendite) e *Prodotto* (con il codice dei prodotti). Lo schema logico relazionale finale ottimizzato

del database degli agenti di commercio diventa quindi il seguente. In cui le tabelle sono tutte in 3NF.



Osserviamo che la tabella *Vendite* in 3NF memorizza soltanto le due righe relative alle due vendite rispetto alle cinque righe (con dati duplicati) della tabella in 1NF e in 2NF.

Codice	Agente (denominazione)	Cliente (denominazione)	Data
V-901	Rossi	Bianchi HW	14/04/2008
V-902	Verdi	Neri Spa	16/04/2008



Nel processo di normalizzazione, i programmatori partono da uno schema logico iniziale in cui le tabelle possono avere molte colonne e le chiavi primarie non sono ancora ben definite. Per soddisfare la 2NF e la 3NF, i programmatori cercano di evidenziare nelle tabelle tutte le colonne che dipendono da una parte della chiave primaria (non in 2NF) oppure che dipendono da altre colonne che non fanno parte della chiave primaria (non in 3NF). Le colonne che fanno parte delle dipendenze devono essere staccate per creare nuove tabelle poste in relazione con le altre. In questa attività, è molto utile ipotizzare dei dati campione da inserire nelle tabelle che non rispettano la 2NF o la 3NF, perché tutte le colonne da "staccare" sono quelle che contengono dei dati ripetuti oppure permettono di eliminare le ridondanze da altre colonne.

La forma normale di Boyce-Codd

La BCNF è una forma alternativa alla 3NF più adatta a essere applicata a tabelle che contengono molte colonne chiave composte che si sovrappongono ovvero contengono colonne in comune.

Una tabella si trova nella **forma normale di Boyce-Codd** se rispetta la 2NF e ogni determinante è una chiave candidata.

Per "chiave candidata" si intende che ogni determinante può diventare una chiave ("è candidato a"). Ricordiamo che un determinante è una colonna o più colonne il cui valore determina quello di altre colonne (funzionalmente dipendenti) e possiamo ricordarlo così:

Determinante → ColonneDipendenti

Per trasformare una tabella in forma BCNF, si devono estrarre sia la colonna determinante sia quelle dipendenti che formano una nuova tabella in relazione con quella originaria.



La normalizzazione ha un'importante applicazione nella ristrutturazione dei dati organizzati in una **matrice** (array bidimensionale). Ricordiamo che una matrice organizza i dati nelle righe e colonne di una tabella che non può essere tradotta direttamente in un'unica tabella dello schema logico relazionale.

Questa tabella non rispetta la 1NF, perché le colonne *Prodotti* e *DataVendita* non contengono valori elementari bensì insiemi di dati organizzati in un vettore e in un record. Per trasformare lo schema della tabella in un altro che rispetti la 1NF trasformiamo i vettori *Agenti*, *Clienti* e *Prodotti* in tre colonne con gli stessi nomi ma scritti al singolare. I campi della colonna *DataVendita* possono essere trasformati in tre colonne autonome, *Giorno*, *Mese* e *Anno*, che costituiscono il record.

L'alternativa più conveniente è quella di eliminare i tre campi e prevedere per la colonna *Data* un dominio che memorizzi le date nel formato *giorno/mese/anno*, che concatena i tre dati parziali in uno solo. Lo schema relazionale modificato (ma ancora da completare!) che rispetta la 1NF diventa quindi quello a fianco.

Come chiave primaria composta della tabella *Vendite* scegliamo inizialmente quella formata dalle colonne *CodiceVendita*, *Agente*, *Cliente* e *Prodotto*. La scelta della chiave primaria in questi primi schemi logici può non essere quella definitiva.

Schema logico relazionale in 1NF

Vendite	
CodiceVendita	
Agente	
AreaGeografica	
Cliente	
IndirizzoCliente	
Prodotto	
PrezzoListino	
DataVendita	
PrezzoVendita	
PerScontoVendita	
QuantitàVenduta	
ImportoTotaleVendita	



Una tabella nella 1NF può avere ancora dei dati duplicati (ridondanze) e presentare anomalie nelle operazioni di aggiornamento. Consideriamo i dati in alcune colonne della tabella *Vendite* che rispetta la 1NF.

Codice vendita	Agente (denominazione)	Area geografica	Cliente (denominazione)	Indirizzo cliente	Prodotto (descrizione)	Prezzo listino	Data vendita
V-901	Rossi	Centro	Bianchi HW	Via...	Mouse	11,50	14/04/2008
V-901	Rossi	Centro	Bianchi HW	Via...	Hard disk	55,00	14/04/2008
V-901	Rossi	Centro	Bianchi HW	Via...	Video LCD	280,00	14/04/2008
V-902	Verdi	Sud	Neri Spa	Piazza...	Mouse	11,50	16/04/2008
V-902	Verdi	Sud	Neri Spa	Piazza...	Video LCD	280,00	16/04/2008
...

Le cinque righe della tabella memorizzano in realtà soltanto due vendite! Molti dei dati della tabella precedente sono **ridondanti**, perché ripetuti (o duplicati) più volte e inutilmente. Esempi di questi dati sono quelli della denominazione dell'agente, dell'area geografica, della denominazione del cliente e del suo indirizzo e, infine, della data di vendita. La tabella "soffre" quindi di **anomalie di aggiornamento**. Supponiamo di dover modificare la denominazione di un agente da *Rossi* a *Rossi Snc*. L'utente si aspetta di eseguire questa modifica una sola volta nel database, mentre nella tabella precedente questa modifica deve essere effettuata in tutte le righe (nell'esempio sono le prime tre) in cui compare una vendita fatta da questo agente! Anche un'operazione di inserimento di un nuovo agente o la sua eliminazione comporta la ripetizione inutile della stessa operazione in più righe. Per eliminare i dati ridondanti e le anomalie di aggiornamento si deve dividere la tabella precedente in altre tabelle che rispettano la 2NF e poi la 3NF.

Seconda forma normale

Una tabella si trova nella **seconda forma normale** se rispetta la 1NF e ogni colonna non chiave dipende da tutte le colonne della chiave primaria composta.

Una **colonna non chiave** (*non-key column*) è una colonna di una tabella che però non fa parte di una di quelle che formano la chiave primaria composta. La 2NF vieta che una colonna