

Schema (logico) relazionale del database della sede centrale

In linea di massima, il database può essere costituito dalle quattordici tabelle di base appresso descritte; le prime otto modellano entità, le altre rappresentano associazioni multi-a-molti.

- **PRODOTTO** (CodiceProdotto, Descrizione, Immagine, PrezzoUnitarioAListino, QuantitàAMagazzino1, QuantitàAMagazzino2)

Questa tabella (*visibile a chiunque*, anche via *web*) raccoglie i dati relativi a tutti i prodotti che possono essere venduti nella catena di supermercati, e servirà sia come catalogo e listino dei prezzi per i clienti, sia come registro delle disponibilità nei due magazzini della sede centrale (gli ultimi due attributi, di tipo intero non negativo: anch'essi possono rivelarsi utili ai clienti che desiderano acquistare via *web*, i quali dovranno quindi rivolgersi alla sede centrale: v. oltre).

La descrizione può essere semplicemente una stringa di caratteri oppure può contenere un riferimento a un *text file* in cui siano riportate tutte le indicazioni utili al cliente; l'eventuale immagine può essere memorizzata come campo BLOB, estraibile con una semplice *query SQL*. Come prezzo unitario s'intende quello attuale, e per semplicità si assume correntemente valido per tutti i punti vendita.

- **PUNTO_VENDITA** (CodicePV, Località, CodiceDipendenteResponsabile*)
- **DIPENDENTE** (CodiceDipendente, Cognome, Nome, ..., CodicePVLavoro*)

Ciascuna di queste due tabelle ha un attributo che costituisce una chiave esterna verso l'altra tabella (e che quindi modella un'associazione uno-a-molti), col vincolo: "se un dipendente è il responsabile di un punto vendita, allora quel dipendente lavora presso quel punto vendita"; formalmente:

$$(x, _, y) \in \text{PUNTO_VENDITA} \Rightarrow (y, _, _, \dots, x) \in \text{DIPENDENTE}.$$

Si conviene di usare un codice particolare per la sede centrale oppure, ammesso che non possa essere diversamente interpretabile, di assegnare il valore nullo all'attributo *CodicePVLavoro* in quelle *tuple* che si riferiscono ai dipendenti che lavorano nella sede centrale.

Eventuali ulteriori attributi di **DIPENDENTE** possono riguardare gli altri dati anagrafici, il numero di telefono, la mansione, la data di inizio del rapporto di lavoro, la retribuzione eccetera.

- **CLIENTE** (CodiceCliente, Cognome, Nome, ...)

Si assume necessaria la registrazione per quei clienti che acquistano via *web*. Eventuali altri attributi di **CLIENTE** sono l'indirizzo (al quale potrà essere spedita la merce), il numero di telefono e l'*e-mail*.

- **FORNITORE** (CodiceFornitore, NomeDitta, ...)

Eventuali ulteriori attributi di **FORNITORE** sono l'indirizzo, i numeri di telefono e di *fax*, l'*e-mail*, il tempo medio di consegna della merce ordinata eccetera.

- **ORDINE_A_FORNITORE** (CodiceOrdine, CodiceFornitore*, DataOrdine, DataEvasione, DataPagamento)

S'intende fatto dalla sede centrale. Ovviamente, $\text{DataOrdine} \leq \text{DataEvasione}$ (se quest'ultima non ha valore nullo) e $\text{DataOrdine} \leq \text{DataPagamento}$ (se quest'ultima non ha valore nullo). Potrebbe essere necessario aggiungere un attributo per registrare le spese di spedizione sostenute dalla sede centrale; eventuali altri per tener conto del tempo e del mezzo di consegna.

A ciascuno degli ordini fatti dalla sede centrale a un fornitore dovrà corrispondere almeno un dettaglio (ossia, almeno una *tuple* nella tabella **DETTAGLIO_ORDINE_A_FORNITORE**: v. oltre).

- **RICHIESTA_DA_PV** (CodiceRichiesta, CodicePV*, DataRichiesta, DataEvasione)

S'intende fatta alla sede centrale. Ovviamente, $\text{DataRichiesta} \leq \text{DataEvasione}$ (se quest'ultima non ha valore nullo). Potrebbe essere necessario aggiungere altri attributi, per tener conto del mezzo di consegna o per registrare il codice del dipendente della sede centrale che ha provveduto a evadere la richiesta e il codice del dipendente del punto vendita al quale è stata consegnata la merce.

A ciascuna richiesta fatta da un punto vendita alla sede centrale dovrà corrispondere almeno un dettaglio (ossia, almeno una *tuple* nella tabella **DETTAGLIO_RICHIESTA_DA_PV**: v. oltre).

Si assume già da ora che ciascun punto vendita, per le vendite al dettaglio ai propri clienti, possa richiedere alla sede centrale l'approvvigionamento di qualsiasi prodotto (a catalogo).

- VENDITA_VIA_WEB (CodiceVendita, CodiceCliente*, DataPagamento, DataEvasione)

S'intende fatta dalla sede centrale a un cliente (registrato). Se si richiede il pagamento anticipato, allora $\text{DataPagamento} \leq \text{DataEvasione}$ (se quest'ultima non ha valore nullo)... Potrebbe essere necessario aggiungere un attributo per registrare le spese di spedizione rimborsate dal cliente, eventualmente un altro per tener conto del mezzo di consegna.

A ciascuna vendita fatta via *web* dalla sede centrale a un cliente (registrato) dovrà corrispondere almeno un dettaglio (ossia, almeno una *tupla* nella tabella DETTAGLIO_VENDITA_VIA_WEB: v. oltre). Le successive tabelle modellano le necessarie associazioni multi-a-molti, con i rispettivi attributi.

- DISPONIBILITÀ_PV (CodicePV*, CodiceProdotto*, Quantità)

Il terzo attributo indica il numero di pezzi del prodotto disponibili nel magazzino del punto vendita; ovviamente è di tipo intero non negativo, e le *tuple* che hanno questo attributo a valore zero è come se non fossero presenti in tabella!

Qualora il prezzo unitario di un dato prodotto potesse variare da un punto vendita a un altro (fermo restando che il valore registrato nella tabella PRODOTTO rimarrebbe valido per le vendite fatte via *web* dalla sede centrale), allora questa tabella dovrebbe avere in più la colonna relativa al prezzo particolare...

- FORNISCE (CodiceFornitore*, CodiceProdotto*, CostoUnitarioAttuale)

Eventuali ulteriori attributi di FORNISCE potrebbero riguardare i tempi (e i mezzi) specifici di consegna, gli sconti praticati dal fornitore per (intervalli di) quantità del prodotto eccetera.

- DETTAGLIO_ORDINE_A_FORNITORE (CodiceOrdine*, CodiceProdotto*, Quantità, CostoUnitario)

Ovviamente, come costo unitario s'intende il prezzo praticato dal fornitore alla data dell'ordine.

- DETTAGLIO_RICHIESTA_DA_PV (CodiceRichiesta*, CodiceProdotto*, Quantità)
- DETTAGLIO_VENDITA_VIA_WEB (CodiceVendita*, CodiceProdotto*, Quantità, PrezzoUnitario)
- VENDITA_AL_DETTAGLIO (CodicePV*, Tempo, CodiceProdotto*, Quantità, PrezzoUnitario, CodiceCliente*)

Nell'associazione rappresentata da quest'ultima tabella è coinvolta, oltre a PUNTO_VENDITA e a PRODOTTO, anche l'entità “debole” Tempo (di tipo datetime): in effetti, ciascun gruppo di *tuple* con stesso valore di CodicePV e stesso valore di Tempo può corrispondere a uno “scontrino” emesso da una cassa di quel punto vendita (ammettendo di serializzare la registrazione degli “scontrini” provenienti dalle casse di ciascun dato punto vendita). L'ultimo attributo può assumere valore nullo (lo assumerà certamente se il cliente che acquista al dettaglio non è registrato nella tabella CLIENTE).

Nell'ottica di un database distribuito, questa tabella sarebbe sostituita da una tabella locale a ogni punto vendita (senza la prima colonna) e da un consuntivo centrale che raccolga (magari giorno per giorno) i dati relativi a ciascun prodotto (con quantità totale e ricavo complessivo) venduto al dettaglio in qualche punto vendita.

Da ciascuno dei punti vendita dovrebbero essere *non accessibili*: i “dati sensibili” di DIPENDENTE e di CLIENTE; le tabelle FORNITORE, ORDINE_A_FORNITORE, DETTAGLIO_ORDINE_A_FORNITORE, FORNISCE, VENDITA_VIA_WEB e DETTAGLIO_VENDITA_VIA_WEB, per intero; infine, sia le richieste fatte alla sede centrale (con i relativi dettagli) sia le vendite al dettaglio, relative agli altri punti vendita. Si noti che in questo database non vi sono ridondanze di informazione (salvo forse qualche vincolo – quale? – sulle quantità dei prodotti in gioco, ammesso di mantenere nel database tutti i “dati storici”!), e comunque non si potranno verificare anomalie, purché i vincoli di integrità descritti siano rispettati e siano progettate opportune procedure per l'inserimento, la modifica e la cancellazione dei dati.