

09. Un esempio pratico

database

un esempio pratico

La società Alfaomega s.p.a. che gestisce una catena di grandi magazzini dislocati in diversi punti della città, vuole informatizzare il proprio sistema informativo.

Tutti i magazzini hanno gli stessi **reparti** e vendono gli stessi **prodotti**, inoltre, per ciascun reparto nei vari magazzini, esiste un unico **direttore** che ne è responsabile (così ad es. il sig. Paolino Paperino dirige il reparto “Articoli sportivi” in tutti i magazzini della città).

un esempio pratico

In modo particolare Alfaomega s.p.a. è interessata a:

- registrare gli ordini effettuati dai **clienti**, insieme con le principali informazioni anagrafiche dei clienti stessi;
- conoscere l'elenco dei **prodotti** di ciascun **reparto**, a prescindere dal magazzino di appartenenza;
- censire i prodotti disponibili nei diversi magazzini sparsi per la città registrandone informazioni quali il prezzo ed il nome del fornitore;
- per ciascun reparto avere un recapito telefonico del **direttore** responsabile.

Per ora non valutiamo quelle che potranno essere le richieste di informazioni sul DB (come invece dovremmo fare), analizziamo piuttosto i pochi requisiti informali che abbiamo per cercare elementi utili alla realizzazione del modello concettuale.

un esempio pratico

Intanto il cliente (con i suoi dati anagrafici), il prodotto (del quale dovrà essere registrato anche il magazzino di appartenenza, se vogliamo censire lo stesso prodotto stoccato in magazzini differenti), il reparto (con il nome ed un codice identificativo) ed il direttore del reparto (del quale dobbiamo avere a disposizione almeno un recapito telefonico), hanno tutti esistenza autonoma e presentano caratteristiche di interesse per il nostro sistema informativo.

un esempio pratico

Scegliamo dunque di rappresentare soggetti con altrettante entità sulle quali definiamo gli attributi come riportato di seguito:

- *Cliente*: codice fiscale, nominativo (composto da nome e cognome), indirizzo di residenza (completo di via, cap e città);
- *Prodotto*: codice prodotto, codice del magazzino di stoccaggio, prezzo, fornitore;
- *Reparto*: numero del reparto, denominazione/descrizione;
- *Direttore*: matricola, nominativo, telefono.

un esempio pratico

Per quel che riguarda gli ordinativi eseguiti dai clienti, siamo interessati a proprietà quali la data, il numero di registrazione, il costo totale e la quantità del prodotto ordinato.

L'idea concettuale di “*ordine sul determinato prodotto*” comunque di per sé non ha esistenza autonoma, dal momento che ha motivo di esistere solo se esiste il cliente che lo ha effettuato ed il prodotto cui si riferisce.

Per questo motivo, certamente può diventare una nuova entità (con attributi per la data di emissione, il numero di registrazione, il costo totale e la quantità ordinata), ma può anche essere tradotto in una associazione *molti-a-molti* tra *Cliente* e *Prodotto* (visto che ogni cliente può ordinare più prodotti così come ogni prodotto può essere inserito in molti ordini); tale associazione presenterebbe gli stessi attributi.

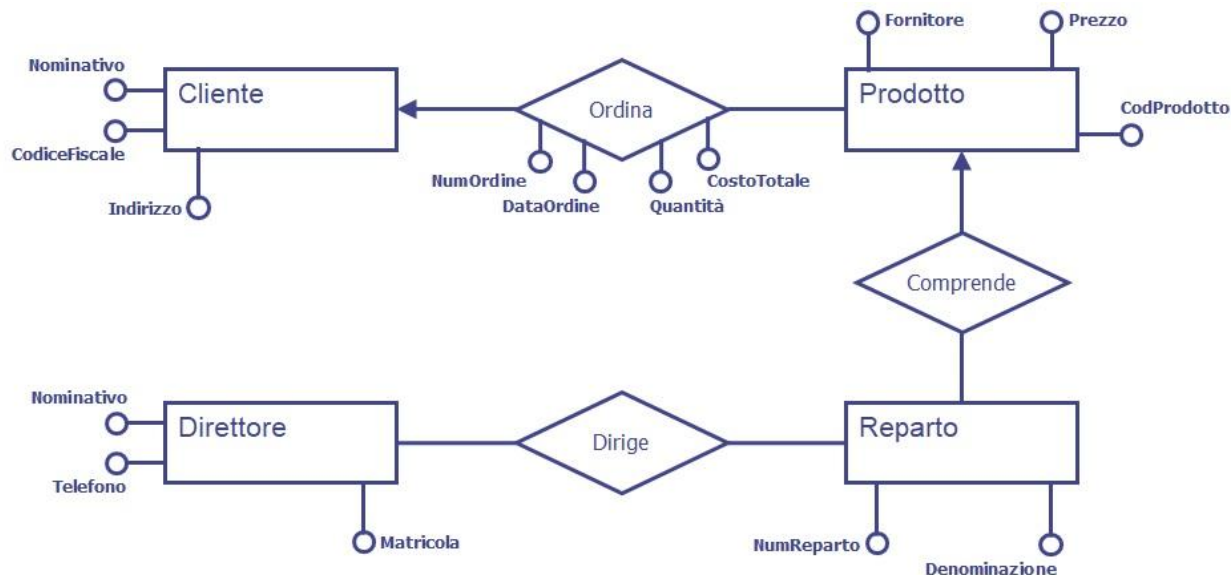
un esempio pratico

La direzione di un reparto invece, così come la collocazione dei prodotti nei diversi reparti, non solo non hanno esistenza autonoma, ma neanche proprietà interessanti.

Semplicemente legano ciascun direttore al proprio reparto di competenza e ciascun reparto all'insieme dei prodotti che questo contiene, pertanto possono essere rappresentate come altrettante associazioni.

un esempio pratico

Una prima bozza semplificata del modello concettuale può allora essere riassunta nel seguente diagramma E/R:



un esempio pratico

Relativamente alla cardinalità e tipologia delle associazioni che legano le entità riportate, possiamo puntualizzare che:

- ogni direttore può dirigere un solo reparto, così come ciascun reparto può avere un solo direttore; inoltre non possono esserci direttori senza un reparto da dirigere o reparti senza un direttore responsabile per quel reparto (per entrambi la cardinalità è 1:1), per cui l'associazione tra *Direttore* e *Reparto* è di tipo *uno-a-uno* e obbligatoria da entrambi i lati;
- Ogni reparto comprende da 0 (nel caso di nuovo reparto ancora da allestire) a più prodotti (cardinalità 0:N), ma ogni prodotto può appartenere ad un solo reparto, (cardinalità 1:1) per cui l'associazione è di tipo *uno-a-molti* e opzionale dal lato del reparto e obbligatoria dal lato del prodotto.

un esempio pratico

Relativamente alla cardinalità e tipologia delle associazioni che legano le entità riportate, possiamo puntualizzare che:

- ogni cliente infine può fare ordini che comprendono da 1 a più prodotti (quindi cardinalità 1:N), mentre ogni prodotto può far parte di più ordini, ma può anche non essere stato ancora ordinato da nessun cliente (cardinalità 0:N) per cui l'associazione tra *Cliente* e *Prodotto* è di tipo *molti-a-molti*, obbligatoria dal lato del cliente e opzionale dal lato del prodotto.

un esempio pratico

Notiamo inoltre che:

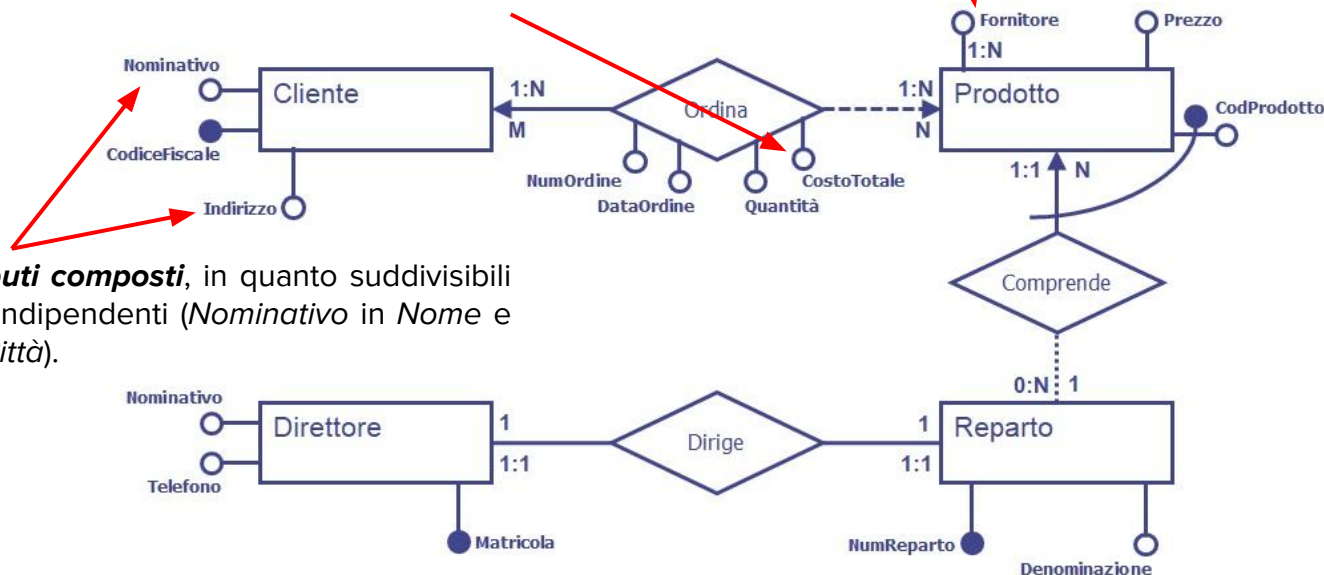
- il codice fiscale è attributo chiave per l'entità *Cliente*, così come la matricola per l'entità *Direttore* e il numero del reparto per l'entità *Reparto*;
- ogni prodotto invece può essere individuato univocamente solo dalla coppia di attributi codice prodotto e codice magazzino, dato che ciascun prodotto potrebbe trovarsi in più magazzini (identificatore esterno);
- il nominativo comprende nome e cognome del cliente, così come l'indirizzo comprende via, cap e città di residenza (attributi composti);
- se ammettiamo che ciascun prodotto possa essere fornito da diversi fornitori, l'attributo corrispondente potrebbe assumere più valori per lo stesso prodotto (attributo multivalore).

un esempio pratico

L'attributo *Fornitore* è **multivalore**, in quanto ciascuna istanza dell'entità *Prodotto* può avere un numero diverso di valori per l'attributo in questione, cioè più fornitori.

NB: È bene eliminare gli attributi multivalore, in quanto non sono ammessi campi multivalore nel modello relazionale.

Diagramma E/R completo: *CostoTotale* è un **attributo calcolato** e può essere ricavato moltiplicando il valore dell'attributo *Quantità* (associazione *Ordina*) con quello dell'attributo *Prezzo* (entità *Prodotto*).



Nominativo e *Indirizzo* sono **attributi composti**, in quanto suddivisibili in parti più piccole con significati indipendenti (*Nominativo* in *Nome* e *Cognome*, *Indirizzo* in *Via*, *Cap* e *Città*).

un esempio pratico

Normalmente, dopo una prima stesura del diagramma E/R, questo deve essere rielaborato e corretto prima di passare alla fase successiva di progettazione logica; alcune azioni sono necessarie per preparare lo schema alle elaborazioni successive, altre sono invece orientate a migliorare le prestazioni generali.

Intanto, il modello relazionale non può gestire dati composti o multivalore, per cui lo schema E/R finale deve prevedere esclusivamente attributi semplici e a valore singolo.

Per questo motivo se ad es. siamo interessati a ricerche per città di residenza dei clienti, allora sostituiamo l'attributo composto *Indirizzo* con gli attributi semplici *Via*, *Cap* e *Città*; non siamo invece interessati a distinguere tra nome e cognome del cliente, per cui possiamo considerare l'attributo *Nominativo* come attributo semplice comprensivo di entrambi i dati.

un esempio pratico

Gli attributi multivalore possono essere sostituiti da una nuova entità che possiede almeno l'attributo in questione, ed è legata all'entità di appartenenza iniziale mediante un'associazione in genere di tipo *uno-a-molti*.

Così nel nostro caso possiamo introdurre una nuova entità chiamata *Fornitore* (del quale a questo punto possiamo anche prendere in considerazione attributi come *Nominativo* e *Telefono*) legata a *Prodotto* da un'associazione di tipo *uno-a-molti*, dato che ciascun prodotto può essere fornito da più fornitori.

Del resto è anche vero che ciascun fornitore può fornire più di un prodotto, dunque è più opportuno che l'associazione tra *Fornitore* e *Prodotto* sia di tipo *molti-a-molti* magari con l'aggiunta di attributi quali: *Quantità*, *DataFornitura* e *PrezzoFornitura*.

un esempio pratico

Per quanto riguarda gli attributi calcolati, un loro utilizzo vuol dire introdurre una sorta di ridondanza nello schema, dovuta alla presenza di dati che mediante opportune elaborazioni potrebbero essere altrimenti derivabili da altri dati comunque già registrati o da operazioni di conteggio sulle occorrenze di certe istanze di entità.

L'uso di attributi calcolati nello schema E/R comporta, da un lato la necessità di effettuare operazioni aggiuntive per mantenere tali dati sempre aggiornati, dall'altro la possibilità di avere un sistema più semplice da realizzare e soprattutto più snello che non deve ricalcolare i corrispondenti valori ad ogni aggiornamento dei dati.

La scelta se inserire o meno attributi calcolati va fatta valutando l'eventuale aumento del costo di esecuzione delle operazioni che coinvolgono questi campi.

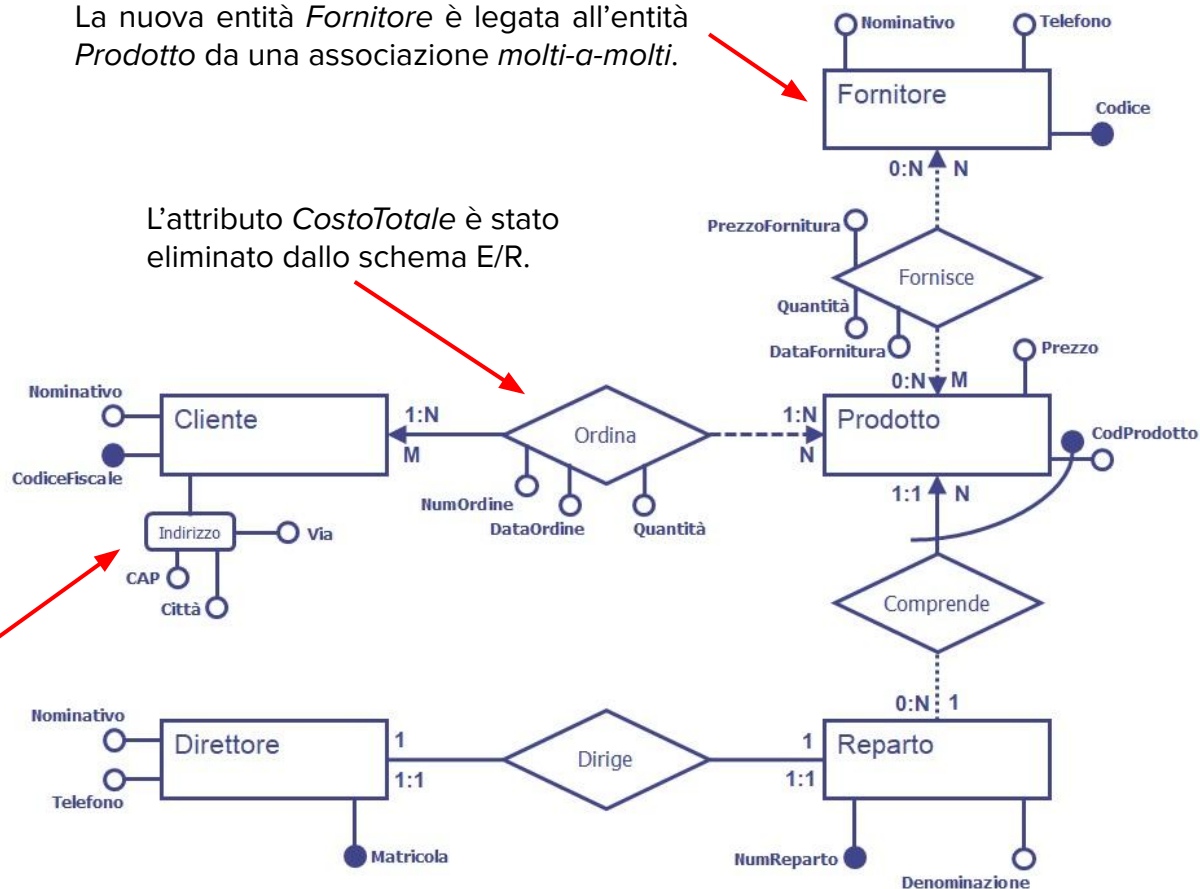
un esempio pratico

Nuovo diagramma E/R:

La nuova entità *Fornitore* è legata all'entità *Prodotto* da una associazione *molti-a-molti*.

L'attributo *CostoTotale* è stato eliminato dallo schema E/R.

L'attributo *Indirizzo* è stato sostituito dai tre attributi atomici *Via*, *Città* e *Cap*.



un esempio pratico

Uno schema E/R può risultare ridondante anche per la presenza di **associazioni derivabili** dalla composizione di altre associazioni in presenza di cicli.

Nel nostro caso ad es. una eventuale associazione *Acquista* tra le entità *Cliente* e *Reparto* (che naturalmente esiste, in quanto ciascun cliente fa acquisti nei vari reparti), sarebbe comoda (nel caso fossimo ad es. interessati a fare statistiche sulla tipologia di clienti che frequentano certi reparti), ma ridondante, dal momento che non darebbe alcuna informazione in più di quanto già non possano fare insieme le due associazioni *Ordine* e *Collocazione*.

In pratica se volessimo sapere ad es. quali sono i reparti più gettonati, basterebbe controllare i prodotti relativi agli ordini effettuati e quindi i reparti di appartenenza di tali prodotti.

un esempio pratico

Anche in questo caso la decisione se mantenere o meno associazioni derivabili dipende da motivazioni di carattere funzionale contrapposte a considerazioni atte a ridurre il carico operativo del sistema al fine di renderlo più efficiente.

Normalmente comunque si tende ad evitare la loro presenza nello schema E/R, se non strettamente necessario.