

UNIVERSITÀ DEGLI STUDI DI PADOVA

FACOLTÀ DI SCIENZE STATISTICHE

CORSO DI LAUREA IN STATISTICA E GESTIONE DELLE IMPRESE



TESI DI LAUREA

**DATABASE E DATAWAREHOUSE:  
ASPETTI TEORICI E OPERATIVI IN  
UN'APPLICAZIONE PER I RISCHI DI MERCATO**

RELATORE: CH.MO. PROF. MICHELE BONOLLO

LAUREANDO: GIORGIO BOZIO

A.A. 2003-2004

*A Valentina  
per il suo aiuto e incoraggiamento  
e ai miei genitori  
per il supporto e la fiducia accordatami*

# INDICE

<b>INTRODUZIONE.....</b>	<b>V</b>
--------------------------	----------

## **CAPITOLO 1**

<b>MODELLI DATI.....</b>	<b>1</b>
--------------------------	----------

1.1 IL DATABASE RELAZIONALE .....	1
1.1.1 <i>Il modello relazionale</i> .....	2
1.1.2 <i>Terminologia</i> .....	5
1.1.3 <i>Il modello dati</i> .....	6
1.1.4 <i>Lo schema E-R</i> .....	11
1.2 LA STRUTTURA DEL DATABASE.....	12
1.2.1 <i>Ridondanze e anomalie</i> .....	13
1.2.2 <i>Chiavi Candidate e Chiave Primarie</i> .....	14
1.2.3 <i>Chiavi esterne e integrità referenziale</i> .....	15
1.2.4 <i>Dipendenze funzionali</i> .....	17
1.3 LA NORMALIZZAZIONE .....	18
1.3.1 <i>La prima forma normale (1NF)</i> .....	18
1.3.2 <i>Seconda forma normale (2NF)</i> .....	20
1.3.3 <i>Terza Forma Normale (3NF)</i> .....	22
1.3.4 <i>Altre forme normali: la forma normale di Boyce/Codd</i> .....	24
1.4 LA DENORMALIZZAZIONE ED I DATAWAREHOUSE.....	25
1.4.1 <i>Definizione</i> .....	25
1.4.2 <i>Come denormalizzare</i> .....	27

## **CAPITOLO 2**

<b>IL VAR.....</b>	<b>31</b>
--------------------	-----------

2.1 IL RISCHIO DI MERCATO .....	31
2.2 COS'È IL VALUE AT RISK (VAR).....	32

2.3 MODELLI NON-PARAMETRICI .....	35
2.3.1 <i>La simulazione storica</i> .....	35
2.3.2 <i>Il metodo Monte Carlo</i> .....	36
2.4 I MODELLI PARAMETRICI .....	37
2.4.1 <i>Le assunzioni del Var</i> .....	38
2.4.2 <i>I rendimenti e la regola della radice</i> .....	39
2.4.3 <i>Il metodo portfolio-normal</i> .....	40
2.4.4 <i>Il metodo asset-normal</i> .....	41
2.4.5 <i>Il metodo delta-normal</i> .....	44
2.5 IMPIEGO DEL VAR PER LA DETERMINAZIONE DEGLI ACCANTONAMENTI DI CAPITALE .....	46

### **CAPITOLO 3**

#### **UN MODELLO DATI PER IL RISCHIO DI MERCATO ..... 49**

3.1 UN MODELLO PER IL RISCHIO DI MERCATO .....	49
3.1.1 <i>Definizione della realtà da modellare: il rischio di mercato</i> .....	49
3.1.2 <i>Gli input necessari ai processi di calcolo</i> .....	51
3.1.3 <i>I requisiti richiesti</i> .....	52
3.1.4 <i>Le fonti di dati</i> .....	53
3.2 LE ENTITÀ .....	54
3.2.1 <i>Notazione e metodo di esposizione</i> .....	54
3.2.2 <i>Gli strumenti finanziari</i> .....	55
3.2.3 <i>I portafogli</i> .....	57
3.2.4 <i>I fattori di rischio</i> .....	61
3.2.5 <i>I prezzi</i> .....	63
3.2.6 <i>Lo Storico VaR</i> .....	65
3.2.2 <i>Considerazioni conclusive: il modello dati complessivo</i> .....	66

#### **BIBLIOGRAFIA ..... 71**

# INTRODUZIONE

L'avvento della New Economy porta a definire quella attuale come l'era dell'informazione: il mondo economico si rende sempre più conto che vince sul mercato globale chi meglio sa accumulare e gestire l'informazione. Di conseguenza il valore dell'azienda non è più determinato solo dagli asset fisici ma soprattutto dalle informazioni che possiede. I progressi della tecnologia informatica ed in particolare dei database hanno permesso di gestire grandi quantità di dati da cui ottenere sempre maggiori informazioni.

Uno degli ambiti nel quale la gestione dei dati ha assunto una rilevanza centrale è quello della gestione dei rischi di mercato. In particolare l'approccio "Value at Risk" alla stima del rischio di mercato presuppone la disponibilità e l'utilizzo di una grande mole di dati. Per questo motivo, e per l'importanza che queste misurazioni hanno assunto negli ultimi anni, la progettazione di una base di dati a supporto di questi calcoli risulta di particolare interesse. L'obiettivo del presente lavoro è appunto la definizione di un modello di dati con queste finalità, focalizzando sia le questioni teoriche sia gli aspetti operativi.

La presente trattazione ha origine dalla esperienza professionale dell'autore nel campo dell'Information Technology e dalla collaborazione con Engineering S.p.A. nella realizzazione di un prototipo per la gestione dei rischi di mercato sviluppato in MS Access. Ho cercato di unire le mie consolidate conoscenze di teoria e pratica dei Database con le nuove conoscenze acquisite riguardo il rischio di mercato, in particolare in riferimento all'approccio parametrico alla stima del VaR e alle strutture di dati necessarie ai suoi processi di calcolo.

Nella stesura del testo si è scelto di adottare un approccio pragmatico, privilegiando gli esempi pratici, ma mantenendo, allo stesso tempo, una solida base teorica. Infatti, sia nella teoria dei database che nella teoria del VaR sono presenti concetti altamente formalizzati e l'utilizzo di esempi pratici ne semplifica notevolmente la spiegazione.

Nel Capitolo 1, dopo una definizione del termine “database” si espongono le caratteristiche dei database relazionali. In seguito alla presentazione dei requisiti di integrità dei dati imposti dall’utilizzo operativo dei database relazionali si definiscono le Forme Normali che aiutano, ma non assicurano, il loro ottenimento.

Vengono infine illustrate le caratteristiche di utilizzo dei dati che giustificano la convenienza delle tecniche di denormalizzazione dei database.

Nel Capitolo 2 si espongono le tematiche della gestione del rischio di mercato e della sua stima mediante la misurazione del VaR, con un accento particolare sui metodi parametrici. Inoltre si definiscono i parametri imposti dagli accordi di Basilea per la determinazione del capitale di accantonamento.

Il Capitolo 3 descrive, data la base teorica fornita dai primi due capitoli, un modello dati a supporto di un applicativo che gestisca il rischio di mercato. Si definiscono i requisiti e le regole di funzionamento della gestione di portafoglio, si descrivono le funzionalità richieste dall’analisi del VaR e dal suo utilizzo, e si rappresentano le tabelle e le principali associazioni che ne implementano le strutture di dati.

# CAPITOLO 1

## MODELLI DATI

### 1.1 Il Database relazionale

Nello svolgimento di qualunque attività, sono fondamentali la disponibilità di informazioni e la capacità di gestirle in modo efficace. Ed è proprio con questo scopo che la tecnologia dei database è oggi largamente utilizzata. Nel corso degli ultimi 40 anni l'utilizzo dei database ha avuto uno sviluppo esponenziale. Si è passati dal suo uso in grandi computer limitato ad un impiego da parte di poche persone molto specializzate alla situazione attuale che vede installato su ogni PC Workstation un qualche applicativo che si basa su database. Persino navigare in Internet significa sempre più avere a che fare con siti Web che spesso basano il loro funzionamento su di un database. Gli stessi motori di ricerca non sono altro che enormi database di catalogazione di pagine web.

Il termine database è perciò diventato di largo utilizzo, alle volte impropriamente. Urge pertanto, prima di iniziare a parlarne nei dettagli, una sua definizione.

Un database è una collezione di dati associati tra loro, conservati in modo persistente in file. I *dati* sono rappresentazioni di fatti registrabili che hanno un significato implicito.

La precedente definizione è alquanto generale e permetterebbe di considerare anche questo stesso testo, essendo un insieme di dati collegati tra loro, come un database. Solitamente però il termine *database* ha un significato più ristretto. Implicitamente il concetto di *database* porta con sé le seguenti proprietà:

- un database è concepito per modellare un qualche aspetto del mondo reale, che alcuni autori chiamano *miniworld*. Modifiche al *miniworld* sono rispecchiate nel database;

- l'insieme dei dati che compone un database è logicamente coerente e possiede un significato. Un assortimento casuale di dati non può essere considerato un database;
- un database è disegnato, costruito e popolato di dati per uno scopo ben definito. Un gruppo di utilizzatori è designato al suo uso, essi sono interessati alle applicazioni per le quali il database è stato concepito.

I meccanismi e i dettagli operativi necessari alla registrazione fisica dei dati sono astratti dagli utenti tramite un DBMS (*database management system*) che è un insieme di programmi che permette di creare, mantenere e modificare un database.

Esistono diverse modalità tramite le quali un DBMS può implementare queste funzionalità ma, negli anni, la tipologia che è risultata di gran lunga più popolare è quella che si basa sul modello dati relazionale.

### ***1.1.1 Il modello relazionale***

Il modello relazionale organizza i dati in un database tramite una collezione di relazioni. Informalmente ogni *relazione* è rappresentata da una tabella. Formalmente, una relazione è un insieme di definizioni di predicati, come per esempio “Mario è di Udine”, dove “Mario” è il soggetto e “è di Udine” è il predicato. I dati “Mario” e “Udine” sono collegati tra loro da una *relazione* (nel senso che un dato è relativo all'altro).

Il modello relazionale fu definito per la prima volta da E. F. Codd (1970) e successivamente esteso da altri autori. Lo scopo di Codd fu di superare le limitazioni dei modelli dell'epoca che non permettevano di realizzare la proprietà di indipendenza tra le strutture fisiche dei dati e la loro descrizione logica.

Il modello relazionale è basato sul concetto di relazione, che ha solide fondamenta nella teoria degli insiemi. Tuttavia, nonostante derivi da una formalizzazione molto precisa e apparentemente complessa, è ricondotto ad alcuni concetti di natura più semplice ed intuitiva che ne hanno decretato il successo.

Nel definire il modello relazionale, Codd fornì 13 regole per la visualizzazione della relazione nella forma di tabella, numerate da 0 a 12:



0. Perché un sistema si possa qualificare come un database relazionale esso deve usare le sue funzioni relazionali per gestire il database
1. Regola dell'informazione: tutta l'informazione nel database deve essere rappresentata in un unico modo, nominalmente, da valori posizionati in colonne nelle righe delle tabelle.
2. Regola della garanzia di accesso: ogni valore memorizzato in un DataBase relazionale deve essere accessibile indicando il nome della tabella in cui si trova, il nome della colonna e il valore della chiave primaria che definisce la riga in cui è memorizzato.
3. Gestione sistematica dei valori nulli (NULL): il DBMS (il sistema di gestione del database) deve supportare una rappresentazione dei valori mancanti e delle informazioni inapplicabili che sia sistematica, distinta da altri valori regolari e indipendente dal tipo di dato. (Codd pensava a due distinte rappresentazioni, *token*, per i valori mancanti e le informazioni inapplicabili, mentre poi l'Sql le ha unificate)
4. Dizionario basato sul modello relazionale: la descrizione logica del DataBase relazionale deve essere rappresentata allo stesso modo di un dato ordinario, in modo che gli strumenti del sistema di gestione del DataBase relazionale si possano usare per gestire la descrizione stessa del DataBase.
5. Linguaggio dei dati: un sistema di gestione di DataBase relazionale può offrire molti tipi di linguaggi di descrizione dei dati e di accesso al DataBase. Comunque ci deve essere almeno un linguaggio che (a) ha una sintassi lineare; (b) può essere utilizzato sia interattivamente che all'interno di programmi; (c) supporta la definizione dei dati e delle viste, la loro manipolazione, i vincoli di integrità dei dati, la gestione delle informazioni riguardanti le autorizzazioni e le operazioni di gestione delle transazioni.

6. Aggiornamento delle viste: deve essere possibile al sistema di gestione del DataBase aggiornare qualunque vista che possa essere definita usando combinazioni di tabelle base che, in teoria sono aggiornabili a loro volta. (L'aggiornamento delle viste è un problema complesso).
7. Inserimento, Aggiornamento e Cancellazione : qualunque operando che descriva il risultato di una singola operazione di lettura deve poter essere applicato ad una singola operazione di inserzione, aggiornamento o cancellazione.
8. Indipendenza fisica dei dati : le modifiche che vengono apportati alla struttura fisica di memorizzazione e ai meccanismi di accesso non devono comportare cambiamenti nei programmi applicativi.
9. Indipendenza logica dei dati: i cambiamenti effettuati sulle tabelle che non modificano nessuno dei dati già memorizzati non devono comportare cambiamenti ai programmi applicativi.
10. Regole di Integrità: le regole che si applicano all'integrità di entità e all'integrità referenziale devono essere esprimibili nel linguaggio dei dati implementato dal sistema di gestione del DataBase e non con i comandi codificati nei programmi applicativi.
11. Distribuzione del DataBase: il linguaggio dei dati implementato dal sistema di gestione del DataBase deve offrire la possibilità di distribuire il DataBase senza richiedere modifiche ai programmi applicativi. Questa opportunità deve essere fornita indipendentemente dalla capacità del DBMS di supportare i DataBase distribuiti.
12. Non-sovrversione: se il sistema di gestione del DataBase offre strumenti che consentano ai programmi applicativi di operare sulle tabelle una riga alla volta, ad un programma applicativo che usa questo tipo di accesso al DataBase deve

essere impedito di infrangere le regole di integrità di entità o integrità referenziale definite per quello stesso DataBase.

La lista appena riportata identifica le 13 regole originarie, che nella seconda versione del modello relazionale divennero ben 333. Codd specificò, inoltre, altri elementi necessari, che non verranno trattati in questo elaborato (tra questi, 9 funzioni strutturali, 3 funzioni di integrità e 18 funzioni di manipolazione).

### 1.1.2 Terminologia

La figura di seguito mostra una relazione, sotto forma di tabella:

NomeFornitore:NomeAzienda	Prodotto:NomeProdotto	PrezzoUnitario:Valuta
Golia	Caramelle Mou Golia	€ 14,00
Cooperativa "La Calabria"	Cereali Carbini	€ 21,00
Formaggi Fortini s.r.l.	Mozzarella di Giovanni	€ 34,80
G'day, Mate	Manjimup Dried Apples	€ 53,00
Mayumi's	Tofu	€ 23,25
New England Selected Tea	The verde al limone	€ 9,65
Delizia d'America	Salsa piccante al peperoncino	€ 21,05
Pasta Buttini s.r.l.	Ravioli Angelo	€ 19,50

Figura 1.1

Ogni riga di dati è una **tupla** (traslitterazione del termine inglese *tuple*, abbreviazione di *n-tuple*).

Le **intestazioni** delle colonne sono composte dai nomi degli **attributi** e dal **dominio**, che rappresenta la “tipologia” del dato.

Il **corpo** della relazione consiste in un insieme non ordinato di zero o più tuple. Questa affermazione porta con sé alcuni concetti rilevanti, primo fra tutti il fatto che le relazioni non hanno nessun ordine intrinseco. Secondariamente, una relazione con zero tuple, ossia una relazione vuota, è ancora una relazione. In terzo luogo, una relazione è un insieme. Gli elementi di un insieme sono per definizione identificabili univocamente. Perciò, per poter qualificare una tabella come relazione, ogni riga della stessa deve essere univocamente identificabile e la tabella non deve contenere righe doppie.

E' necessario sottolineare che alcuni dei termini che abbiamo utilizzato fanno parte della teoria delle basi di dati e non si riscontrano poi nei sistemi di gestione di basi di dati attualmente in uso. Relazione, tupla, attributo, sono termini che nelle implementazioni di basi di dati sono sostituiti, rispettivamente, da tabella, riga e campo.

### **1.1.3 Il modello dati**

Un **modello dati** è un insieme di concetti per organizzare e descrivere la struttura dei dati nel database e perciò, in qualche modo, per rappresentare formalmente e logicamente un insieme di soggetti e processi del mondo reale.

Il livello più astratto del processo di disegno di un database è il *modello dati*, la descrizione concettuale dello spazio di un problema. I modelli dati sono espressi in termini di **entità**, **attributi**, **domini** e **associazioni**. Il resto del presente paragrafo discuterà ciascuno di questi elementi.

#### **Entità**

Non è facile definire formalmente il termine entità, ma, intuitivamente, il concetto è alquanto chiaro: un'entità è tutto ciò di cui il sistema deve registrare informazioni.

Quando si inizia a disegnare un modello dati, compilare una lista iniziale delle entità è semplice. Quando si parla dello spazio del problema, la maggior parte dei nomi e dei verbi che vengono usati è candidato ad essere una entità. Per comprendere meglio tale concetto possono essere analizzate le seguenti tre frasi:

- "I clienti comprano prodotti"
- "I dipendenti vendono prodotti"
- "I fornitori vendono prodotti"

Le parole "Clienti", "Prodotti", "Dipendenti" e "Fornitori" sono tutte entità.

Anche i verbi "comprare" e "vendere" sono entità, ma con alcuni distinguo e chiarimenti. Infatti, nelle diverse frasi che abbiamo riferito, i due verbi in questione si riferiscono allo stesso evento poiché la vendita di un venditore nel verso contrario diventa acquisto di un cliente. Mentre può anche accadere l'opposto, e cioè che il termine "vendita" si riferisca a due eventi diversi, la vendita del dipendente e la vendita del fornitore.

Nella definizione delle entità occorre pertanto individuare le eventuali entità duplicate e le entità distinte che stanno mascherando quella che è in realtà una sola entità.

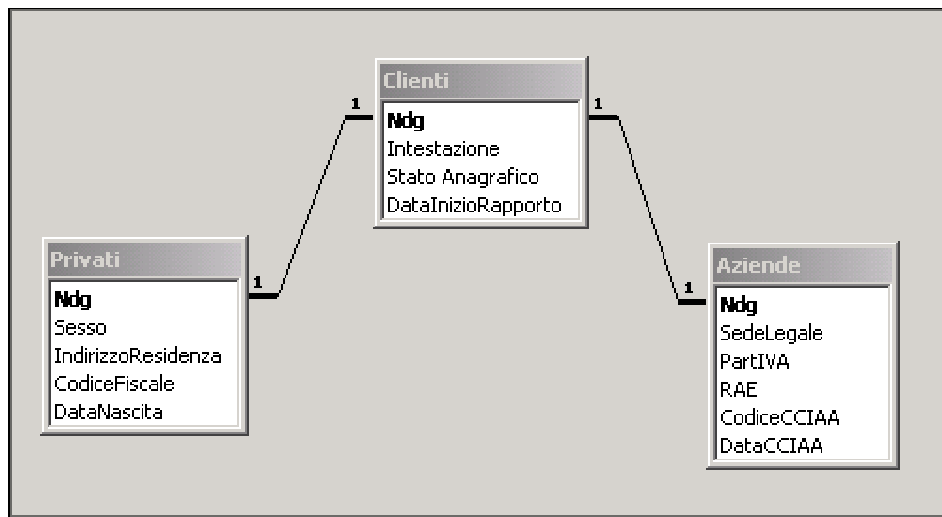


Figura 1.2

Un concetto che aiuta nella definizione del modello dati in questo caso è quello delle sotto-entità, anche definito da alcuni *modello is-a*. Facendo riferimento all'esempio in figura 1.2, vediamo che sono state progettate tre tabelle per rappresentare, nell'ambito di un database per una banca, l'entità "Clienti", suddivisa per gli attributi specifici, nelle sottoentità "Privati" e "Aziende". Hanno infatti in comune alcune caratteristiche che li definiscono come "clienti" ma possiedono al tempo stesso delle peculiarità che ne integrano la definizione e che richiedono la loro modellizzazione nella forma di due sotto-entità. Le caratteristiche comuni ai due tipi di cliente saranno perciò parte dell'entità generale "Clienti", mentre nelle due sotto-entità saranno presenti solo le caratteristiche peculiari.

In altri casi può capitare che in realtà questi due sottotipi di un'entità non possiedano aspetti distinti. In questo caso è corretto (e conveniente) specificare un attributo (TipoDiVendita) per distinguere semplicemente le due tipologie.

Le sotto-entità sono di solito mutuamente esclusive ma ciò non è sempre vero. Si pensi, ad esempio, ad un database di dipendenti di un'azienda, in cui all'entità comune "dipendenti" sono associate delle sotto-entità quali impiegati, venditori, componenti della squadra di calcio aziendale, con alcuni attributi comuni (nome, codice, telefono ecc.) e altri specifici. In questo caso uno stesso soggetto può essere definito nel modello

tramite due sotto-entità diverse, infatti non c'è nulla che vieti ad un ragioniere di giocare a calcio.

Le entità analizzate fino ad ora sono entità *concrete*, poiché rappresentano qualcosa di esistente nel mondo fisico. Esistono tuttavia anche delle entità *astratte*, che rappresentano relazioni (o associazioni) tra altre entità. Ad esempio, la figura 1.3 rappresenta l'entità Prestito di una biblioteca. Questa entità è astratta, nel senso che non rappresenta nessun soggetto fisico se non l'associazione tra un utente della biblioteca ed il libro preso in prestito.

CodiceLibro	CodiceLettore	DataPrestito
657-234	28374	12/12/2003
746-748	15234	07/12/2003
872-745	31093	23/12/2003

**Figura 1.3**

Alcune volte è sufficiente definire semplicemente l'esistenza di una tale associazione, mentre altre volte è utile aggiungere delle proprietà specifiche, come nell'esempio la data del prestito.

### **Attributi**

Gli attributi servono a tenere traccia di "fatti" riguardanti ciascuna entità. Determinare gli attributi che vanno inclusi nel modello dati è un processo che attiene alla semantica delle entità stesse. Gli attributi devono, perciò, dare significato alle entità e questo significato è determinato dal ruolo che le entità hanno all'interno dello spazio del problema.

Un classico esempio di quanto testé enunciato è la rappresentazione degli indirizzi. Tale rappresentazione dipende dal significato dell'indirizzo nel problema in esame e dall'utilizzo che si prevede di farne. Se l'indirizzo viene utilizzato solo per spedire posta, sarà sufficiente un singolo attributo che contenga l'intera stringa dell'indirizzo. Se invece esiste la necessità di analizzare i dati geograficamente o di spedire posta solo in determinate regioni o zone, allora sarà necessario separare le singole parti dell'indirizzo in distinti attributi.

Secondo Matisse, un dipinto è concluso quando nulla se ne può aggiungere né sottrarre. Il disegno delle entità segue la stessa regola. Ma come si fa a sapere quando si è raggiunto quel punto? Sfortunatamente non lo si sa mai per certo. Come per la scienza sperimentale, si può dire che un modello di dati ha dei difetti di design ma non che non ne sia sicuramente privo: non esistono modelli dati perfetti, ma solo non confutati.

Ci sono però delle strategie per evitare (o limitare) i difetti.

La prima strategia consiste nel progettare il modello senza aggiungere livelli di complessità che non siano necessari. Per raggiungere tale obiettivo è necessario individuare l'insieme delle domande alle quali deve rispondere il database. Nel caso di cui sopra, la domanda potrà essere: "Dove spedisco una lettera a questa persona?" oppure "In quale provincia vive?". A queste due domande corrisponde una struttura diversa dell'entità, con un'articolazione diversa degli attributi.

Sono da considerare non solo le domande attuali degli utenti ma anche quelle che prevedibilmente sorgeranno in futuro. Inoltre occorre prendere in considerazione i quesiti che gli utenti farebbero se solo sapessero che il sistema è in grado di fornire una risposta.

In questo processo è fondamentale gestire il trade-off tra l'aumento della flessibilità e l'aumento della complessità.

La seconda strategia riguarda la gestione delle eccezioni. Ciò significa in primo luogo identificarle e poi gestirne il maggior numero possibile senza rendere confuso il modello dati per gli utenti che dovranno utilizzarlo.

Tornando all'esempio della casa di moda, supponiamo che la nostra maison abbia tra i suoi clienti la Contessa Lucrezia Lante della Rovere. La tabella "Clienti", per gestire questo nome, che rappresenta senza dubbio un'eccezione, dovrebbe avere dei campi che gestiscano anche i titoli nobiliari. Anche considerando che diverse nobildonne siano clienti di alta moda, probabilmente una tale complicazione del modello dati non sarebbe ben accetta a chi dovrà poi usarlo.

## **Domini**

Come accennato precedentemente, il dominio di un attributo specifica la "tipologia" del dato e rappresenta un concetto diverso dal tipo di dato. La differenza tra "tipo di dato" e

“dominio” è che il primo è un concetto fisico mentre il secondo è un concetto logico. In particolare il dominio è l’insieme dei dati che un attributo può contenere.

Non è banale distinguere i due concetti, ma va tenuto ben presente che i “domini” e i “tipi di dati” si riferiscono a piani diversi: “Numero intero” è un tipo di dato, “Età” è un dominio. “Nome della Via” e “Cognome” sono entrambi delle stringhe ma chiaramente si riferiscono a domini diversi.

Il dominio contiene una definizione della validità del dato contenuto nell’attributo più specifica del tipo di dato. Ad esempio, il dominio “Provincia”, che rappresenta la sigla delle province italiane, in un database sarà definito come stringa di 2 caratteri, ma è noto che non corrisponde ad una qualsiasi stringa di 2 caratteri, bensì ad una delle sigle delle province italiane esistenti.

Si potrebbe perciò dire che un dominio è un tipo di dati con delle regole di validazione, anche se la differenza tra questa definizione e i domini è che questi ultimi riguardano la descrizione dei dati e non l’integrità degli stessi.

Dati due domini, se ha senso comparare attributi definiti su di essi, questi si dicono “type-compatible”.

Sebbene i DBMS commerciali non forniscano in genere supporto per i domini, essi sono molto importanti nella fase di progettazione concettuale del modello dati e se ne utilizza il concetto soprattutto per specificare quegli attributi che assumono dei valori all’interno di un insieme ben delimitato come, ad esempio, per l’attributo “sesso” i valori nell’insieme {m,f} oppure per l’attributo “stato civile” i valori in {“Coniugato”, “Celibe”}

### **Associazioni**

Oltre agli attributi di ogni entità, un modello dati deve specificare anche le associazioni tra entità. A livello concettuale, le associazioni rappresentano legami logici tra due o più entità. L’affermazione “I clienti acquistano prodotti” indica che esiste una associazione tra le tabelle “clienti” e “prodotti”.

Le entità che fanno parte di una associazione sono dette *partecipanti* e il numero di partecipanti determina il *grado* dell’associazione (mentre nel caso della singola relazione il grado rappresenta il numero degli attributi).



La maggior parte delle associazioni sono binarie, coinvolgono cioè due entità. Esistono tuttavia anche delle associazioni ternarie come ad esempio “i venditori vendono prodotti ai clienti”.

Un’entità è *ricorsiva* se è posta in associazione con se stessa. Un esempio calzante è il caso della associazione tra dipendenti e dirigenti, poiché ogni dipendente può avere altri dipendenti cui fa capo e lui stesso far capo ad un dirigente.

Altro aspetto importante nelle associazioni è la loro *Cardinalità*, che specifica, per ogni associazione, il numero minimo e massimo di occorrenze (righe) di associazione cui una occorrenza dell’entità può partecipare.

La cardinalità viene rappresentata in un modello dati suddividendo le associazioni in associazioni uno-ad-uno, uno-a-molti, multi-a-molti e poi classificando la partecipazione delle entità come parziale o totale. Le associazioni *uno-a-molti* sono le più comuni: “un ordine contiene più prodotti”, “un venditore procura molti ordini” sono entrambi esempi di associazioni con cardinalità uno-a-molti. Le associazioni di cardinalità *uno-a-uno* sono meno comuni e riguardano principalmente le associazioni di tipo entità-sottoentità (*is-a*), quali “una vendita è di pret-a-porter”, “una vendita è di alta moda”.

Sono abbastanza comuni le associazioni con cardinalità *molti-a-molti*, come ad esempio “un docente insegna a molti studenti” e “uno studente ha molti docenti”.

La partecipazione delle entità si dice *totale* se l’esistenza dell’entità è subordinata all’esistenza dell’associazione, per esempio la sottoentità “capo di alta moda” che specifica un’entità “prodotto” della casa di moda non esisterà se non esiste il corrispondente prodotto. Se invece l’esistenza dell’entità è indipendente dall’associazione, la partecipazione si dice *parziale*.

#### **1.1.4 Lo schema E-R**

Il modello Entità Relazione, che descrive i dati in termini di entità, attributi, e associazioni, venne introdotto da Peter Pin Shan Chen nel 1976.

Contestualmente, Chen propose una modalità di rappresentazione grafica tramite diagrammi del database relazionale, chiamata “Schema E/R” e diventata oggi di comune utilizzo, poiché è facile da comprendere e disegnare. Gli schemi E/R utilizzano rettangoli per descrivere le entità, ellissi per gli attributi e rombi per rappresentare le

associazioni, come illustrato nella figura 1.4. In pratica però spesso si omettono le ellissi degli attributi, preferendo indicarli a parte o sistemarli sotto forma di lista all'interno dei rettangoli delle entità.

La tipologia delle relazioni tra le entità (uno-a-uno, uno-a-molti o molti-a-molti) è rappresentata in vari modi. Alcuni utilizzano 1 e M o 1 e  $\infty$  per indicare cardinalità 1 o molti. Altri usano la tecnica "crow's foot" che rappresenta la cardinalità con una simbologia astratta.

D'ora in avanti, sarà utilizzata la notazione che posiziona 0,1 e N agli estremi delle linee che mettono in relazione due tabelle.

Alcuni DBMS comprendono già al loro interno modalità per rappresentare le associazioni (anche MS Access lo consente).

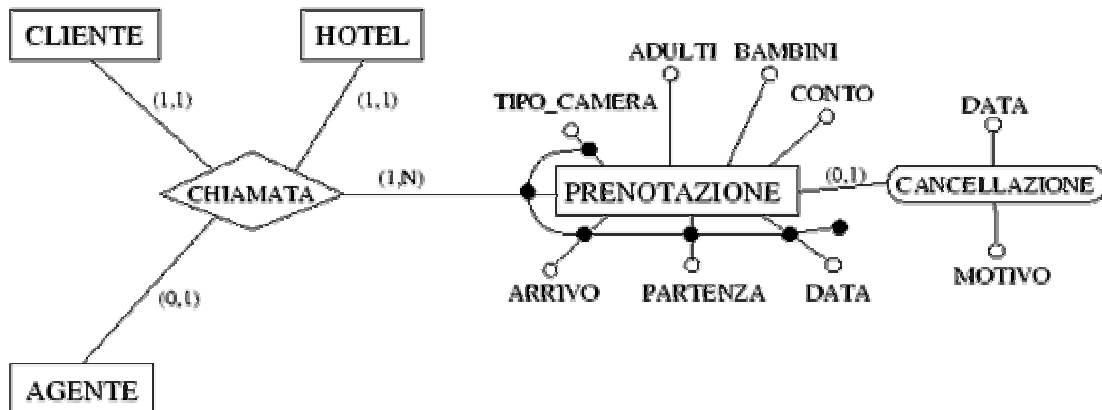


Figura 1.4 - Esempio di schema ER tratto da un modello dati per la gestione di un hotel

## 1.2 La struttura del database

Questo paragrafo discuterà del primo aspetto del processo di disegno di basi di dati: la struttura del database. Disegnando la base di dati si terranno in considerazione due obiettivi: rispondere alle domande che gli utenti potranno e minimizzare le anomalie e le ridondanze.

### 1.2.1 Ridondanze e anomalie

Per introdurre i concetti che sottendono il processo di normalizzazione, sarà utilizzato un esempio di modello dati.

Si consideri un'azienda che fornisce il servizio di noleggio di automobili. I dati di cui l'azienda avrà bisogno per gestire il servizio saranno quelli che associano ad ogni noleggio una riga di dati sviluppata come in figura 1.5. L'azienda utilizza esclusivamente la tabella dell'esempio per gestire i suoi dati.

Modello Auto	Costo/giorno	Targa	Cliente	Categoria	Inizio ultimo Noleggio
Fiat Punto	30	CA 345 HG	Rossi Mario	Utilitarie	12/10/2003
Fiat Punto	30	BZ 394 GF	Doni Enzo	Utilitarie	14/10/2003
Alfa 156	60	BW 293 BN	Bianchi Dino	Medie	10/10/2003
Bmw 730d	75	CG 394 GH	Verdi Silvio	Lusso	11/10/2003

Figura 1.5

I problemi che verosimilmente potrebbero insorgere nella gestione di questi dati sono sostanzialmente quattro:

1. il costo giornaliero, che non dipende dal noleggio ma dal modello di macchina, viene ripetuto per tutte le righe relative allo stesso modello. Si tratta in una **ridondanza**;
2. se cambia il costo giornaliero di un modello, questo va cambiato in tutte le righe corrispondenti. Questa è una **anomalia di aggiornamento**;
3. se un'auto temporaneamente non è noleggiata essa non appare tra i dati disponibili azienda. Questa si dice **anomalia di cancellazione**;
4. se l'azienda acquista una nuova auto questa non compare fino a che non viene noleggiata, generando una **anomalia di inserimento**.

Questi problemi sorgono perché si è tentato di rappresentare con una singola tabella statica informazioni eterogenee. In una singola rappresentazione sono inglobati dati riguardanti le entità auto, modelli, categorie, clienti e noleggi.

Per evitare questi problemi, si decompone la relazione eterogenea avendo cura di non perdere informazione in questo processo. Questa tecnica viene chiamata

“decomposizione senza perdita” (“Lossless Decomposition”). La figura 1.6 illustra la decomposizione della tabella unica del precedente esempio in tre tabelle.

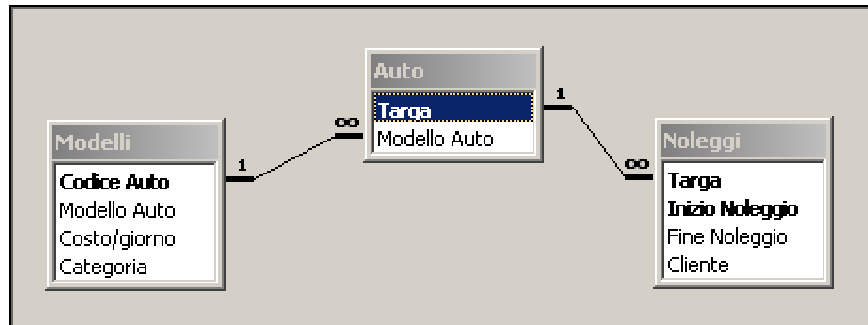


Figura 1.6

### 1.2.2 Chiavi Candidate e Chiave Primarie

Precedentemente si è detto che il corpo di una relazione è definito come un insieme di zero o più tuple ed è stato rimarcato il fatto che per definizione i componenti sono unici. Questa proprietà implica che ci debba essere una combinazione di attributi che identifichi univocamente le tuple e questo insieme di uno o più attributi viene chiamato *Chiave Candidata*.

Per ogni relazione possono esistere una o più chiavi candidate ma ognuna di queste deve identificare univocamente le tuple. Pertanto la loro definizione implica la conoscenza della semantica dei dati. Il fatto che una particolare combinazione di attributi, per un particolare insieme di tuple, capiti di essere valorizzata con valori diversi per ogni tupla, non è indicativa.

Un requisito fondamentale perché una combinazione di attributi sia una delle chiavi candidate è che non siano compresi attributi non necessari all’identificazione univoca della tupla. In altre parole, la chiave deve essere irriducibile.

Per una relazione ci può essere più di una chiave candidata ma poi nelle implementazioni pratiche una di queste chiavi è designata come **Chiave Primaria (PK)**.

Nella figura 1.7 è riportata una tabella la cui chiave candidata (e chiave primaria) è IDCategoria, mentre anche se la combinazione dei due campi IDCategoria e

NomeCategoria è univoca, non è chiave candidata dato che NomeCategoria non è un attributo necessario alla definizione univoca delle righe (tuple).

IDCategoria	NomeCategoria	Descrizione
1	Bevande	Soft drinks, caffè, tè, birra
2	Condimenti	Salse dolci e salate, sale, pepe, olio, aceto, creme spalmabili
3	Pasticceria	Dessert, canditi, paste
4	Prodotti caseari	Formaggi
5	Cereali	Pane, cracker, pasta e cereali
6	Carne/Pollame	Carni confezionate
7	Frutta secca	Arachidi, mandorle, pistacchi, nocciole
8	Frutti di mare	Alghe e pesce

Figura 1.7

### 1.2.3 Chiavi esterne e integrità referenziale

La definizione di chiave primaria permette di completare il concetto di associazione tra tabelle fornendo lo strumento per stabilire il legame tra due relazioni.

L'esempio in figura 1.8 rappresenta il concetto di associazione e introduce una notazione leggermente diversa da quella dello schema ER. Infatti gli attributi non sono raffigurati come oggetti separati dato che, a livello di definizione delle associazioni, ciò che si vuole rappresentare non è la composizione interna delle relazioni, quanto le associazione tra di esse.

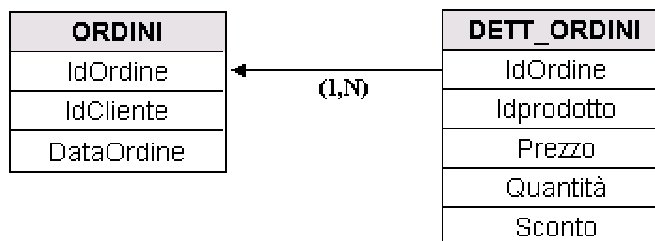


Figura 1.8

Questo tipo di rappresentazione risulta più sintetico dello schema ER interpretato in senso stretto e si ritrova nei software DBMS (anche in MS Access) non solo come mera raffigurazione, ma come parte stessa del database. Non essendo una rappresentazione

astratta del database, occorre creare fisicamente le tabelle prima di definire un diagramma ed è inevitabile il pericolo di implementare il database prima di averlo progettato. Di conseguenza, si corre il rischio di dover correggere eventuali errori del progetto in fase avanzata con difficoltà e dispendio di tempo maggiori.

Si pensi, nel caso in figura 1.8, ad una riga della tabella DETT\_ORDINI, in cui sono registrati i singoli prodotti facenti parte di un acquisto, che non abbia, in corrispondenza dell'attributo IdOrdine, un corrispondente valore nell'attributo IdOrdine nella tabella ORDINI. In questo caso si dice che esiste una violazione dell'integrità referenziale che identifica la regola di business secondo cui ogni ordine ha un cliente che lo esegue ed una data di acquisto. L'attributo IdOrdine della tabella ORDINI è chiave primaria, mentre l'attributo IdOrdine della tabella DETT\_ORDINI si dice chiave esterna.

Formalmente, si ha un vincolo di *integrità referenziale* tra due relazioni R1 e R2 quando un insieme di attributi in R1 definito *chiave esterna* FK (foreign key) soddisfa le seguenti due regole:

1. gli attributi in FK in R1 hanno lo stesso dominio degli attributi della chiave primaria PK in R2. Si dice allora che gli attributi di FK si riferiscono alla relazione R2;
2. un valore di FK in una tupla di R1 si ritrova come valore di PK per una tupla di R2. Si dirà, allora, che la tupla di R1 si riferisce alla tupla di R2.

Se per una riga di una tabella in cui abbiamo definito una FK, al suo valore non corrisponde alcuna riga della tabella associata con un uguale valore di PK, allora avremo che un vincolo di integrità è stato violato.

La figura 1.9 mostra cosa succederebbe in un database con le due tabelle in figura 1.8 se venisse violata l'integrità referenziale.

IdOrdine	IdCliente	DataOrdine
122	345	11/10/2003
123	234	10/10/2003
137	164	12/10/2003
138	147	14/10/2003

IdOrdine	IdProdotto	Prezzo	Quantità	Sconto
122	873	€18,00	1	0%
123	273	€12,20	2	0%
125	263	€10,35	3	33%
125	283	€13,70	1	0%
137	823	€11,70	2	0%
138	283	€15,30	1	0%

**Figura 1.9**

Alle righe di DETT\_ORDINI corrispondenti ai prodotti acquistati tramite l'ordine con IdOrdine uguale a 125 non corrisponde una riga in ORDINI e ciò priva delle informazioni riguardo la data di acquisto ed il cliente.

#### ***1.2.4 Dipendenze funzionali***

Sorge la necessità di dividere i dati conservando nelle varie rappresentazioni i **vincoli** tra gli attributi, che possano riflettere, nel nostro modello dati, i legami **funzionali** tra gli attributi stessi.

Nell'esempio di figura 1.5 è chiaro che esiste un legame funzionale tra il modello di auto ed il suo costo giornaliero. Più precisamente, si può affermare che il modello d'auto determina il suo costo di noleggio e cioè che ad ogni modello corrisponde un costo.

La notazione formale della dipendenza funzionale è la seguente:

modello auto → costo noleggio

e possiamo dire che il costo del noleggio è determinato in *funzione* del modello di auto.

La dipendenza funzionale è, probabilmente, il concetto più importante nella definizione di uno schema relazionale.

### 1.3 La Normalizzazione

Il processo di decomposizione delle relazioni tra i dati effettuato per evitare i problemi esposti nel nell'esempio di figura 1.9 assume il nome di normalizzazione. Fu E.F.Codd a coniare nel 1970 il termine *relazioni normalizzate* prendendolo a prestito dal gergo della politica internazionale di quei tempi.

La normalizzazione si estrinseca attraverso una serie di test che verificano se un modello relazionale soddisfa quelle che vengono chiamate **Forme Normali**. Una branca della matematica chiamata algebra relazionale si occupa della mappatura tra insiemi definita dalla logica formale. Come in una relazione algebrica, ci sono diverse forme della stessa dichiarazione di relazione, ma le forme normali delle relazioni sono senza dubbio quelle la cui dichiarazione formale è più chiaramente definita.

Le forme normali sono un tentativo di assicurarsi che non si distruggano dati veri o non si creino dati falsi nel database. Uno dei modi di evitare errori è di rappresentare un fatto solo una volta nel database, dato che se un fatto appare più di una volta, una delle sue istanze è probabilmente erronea.

Esistono tre forme normali principali che sono considerate la base per poter definire una base di dati come normalizzata; esistono poi altre tre forme normali che trattano alcuni casi speciali ma che normalmente nei casi pratici vengono raramente soddisfatte.

Vedremo, in questo e nei paragrafi successivi, come non sempre sia necessario né utile soddisfare pienamente tutte queste regole, tenendo presente quanto detto riguardo le motivazioni che ne hanno fatto sorgere la necessità e cioè i problemi di ridondanza, aggiornamento, cancellazione e inserimento.

#### 1.3.1 La prima forma normale (1NF)

Una relazione è nella *prima forma normale* se i domini sui quali i suoi attributi sono definiti sono scalari. Questo è allo stesso tempo il più semplice e il più difficile concetto nella modellizzazione dei dati. Il principio è lineare: ogni attributo di una riga deve contenere un singolo valore. Nella relazione definita nella figura seguente (figura 1.10), l'attributo Articoli contiene manifestamente valori multipli e per questo non soddisfa la prima forma normale.



IDOrdine	IDCliente	DataOrdine	Articoli	TotaleOrdine
1	ALROB	12/11/2003	3 Zucchero Fini, 1 The Star	€ 89,70
2	BVFRE	18/12/2003	4 Mozzarella di Giovanni	€ 139,20
3	TOSGR	21/12/2003	3 Ravioli Angelo, 6 Tofu	€ 198,06

Figura 1.10

Non sempre, tuttavia, il rispetto della prima forma normale è così facilmente individuabile.

Ad esempio nel considerare in un modello il dato relativo all'indirizzo, ci si trova di fronte al problema di decidere se un indirizzo espresso nella sua forma estesa che considera via, città, cap sia da considerarsi scalare o meno, in quanto si potrebbe essere interessati al trattamento dei dati anche nelle singole parti dell'indirizzo.

Per lo stesso motivo, anche le date rappresentano un dominio problematico, poiché anche in questo caso potrebbe essere necessario utilizzare le singole parti di una data.

In entrambi i casi sarà la logica di business e la semantica del modello a indicare se l'attributo sia da considerarsi scalare o meno. L'atomicità del dato è relativa al suo utilizzo.

Un altro tipo di valore non scalare da considerare quando si controlla se una relazione è in prima forma normale è il *repeating group*. La figura 1.11 mostra una relazione di Ordine di acquisto. Qualcuno, in questo caso, ha deciso che un cliente non possa acquistare più di 5 articoli. C'è da chiedersi se si sia considerata l'opinione del direttore delle vendite... questa è senza dubbio una limitazione artificiale imposta dal sistema, di certo non dall'attività commerciale.

IDOrdine	IDCliente	Articolo1	Qtà1	Articolo2	Qtà2	Articolo3	Qtà3	Articolo4	Qtà4	Articolo5	Qtà5
1	BRCEN	Salsa piccante Peperoncino Fiery	4	Tofu	5	Ravioli Angelo	3		0		0
2	GHYRT	Condimento Peperlizia	2		0		0		0		0

Figura 1.11

Un altro esempio di gruppo ripetuto è presentato nella figura 1.12. Non si tratta in tal caso di un errore lampante e molti sistemi efficaci sono stati implementati in questo modo. Ma è solo una variante di quanto visto nel caso precedente ed ha le medesime

problematiche. Si immagina l'interrogazione per calcolare quali prodotti hanno superato il target di più del 10% in uno qualsiasi dei mesi del primo quadrimestre.

Prodotto	Anno	ObiettivoGen	RealizzatoGer	ObiettivoFeb	RealizzatoFeb	ObiettivoMar	RealizzatoMar
Sciroppo Fabbri	1999	€ 1.100,00	€ 1.400,00	€ 0,00	€ 0,00	€ 0,00	€ 0,00
Condimento Peperlizia	1999	€ 3.000,00	€ 1.500,00	€ 0,00	€ 0,00	€ 0,00	€ 0,00
Mayonese Creek	1999	€ 2.000,00	€ 6.500,00	€ 0,00	€ 0,00	€ 0,00	€ 0,00
Pasta Annesa	1999	€ 7.000,00	€ 7.500,00	€ 0,00	€ 0,00	€ 0,00	€ 0,00
Capperi Saclà	1999	€ 1.500,00	€ 2.500,00	€ 0,00	€ 0,00	€ 0,00	€ 0,00

Figura 1.12

### 1.3.2 Seconda forma normale (2NF)

Una relazione è nella seconda forma normale se soddisfa la prima forma normale e in aggiunta tutti suoi attributi dipendono da tutta la chiave candidata. La chiave nella figura 1.13, ad esempio, è {NomeProdotto, NomeFornitore}, ma il campo TelefonoFornitore dipende solo da NomeFornitore e non dall'intera chiave.

NomeProdotto	NomeFornitore	NomeCategoria	TelefonoFornitore
Sciroppo Fabbri	Fabbri	Bevande	06 555 878 92
Mayonese Creek	Crios Group	Condimenti	081 8975996
Pasta Annesa	Annes srl	Cereali	0432 7995632
Capperi Saclà	Saclà	Condimenti	02 6687563
Pistacchi Alger	Alger SpA	Frutta secca	0429 781598

Figura 1.13 -Tutti gli attributi in una relazione devono dipendere da tutta la chiave.

Abbiamo già visto che la struttura di cui si sta parlando produce ridondanza e può, a volte, comportare fastidiosi problemi di manutenzione dei dati. Un migliore modello dati sarebbe quello esposto in figura 1.14 in cui le due entità Prodotti e Fornitori non vengono rappresentate in una singola relazione bensì in due tabelle distinte.

In questo modo non solo si evitano ridondanze, ma si ha anche la possibilità di registrare maggiori informazioni sulle entità in maniera indipendente l'una dall'altra. Sarà ad esempio possibile avere delle informazioni su di un fornitore senza per questo dover avere dei suoi prodotti nel database.

### Relazione Prodotti

IDProdotto	NomeProdotto	NomeCategoria
1	Capperi Saclà	Condimenti
2	Mayonese Creek	Condimenti
3	Pasta Annesa	Cereali
4	Pistacchi Alger	Frutta secca
5	Sciroppo Fabbri	Bevande

### Relazione Fornitori

IDFornitore	NomeFornitore	TelefonoFornitore
1	Fabbri	06 555 878 92
2	Crios Group	081 8975996
3	Annes srl	0432 7995632
4	Saclà	02 6687563
5	Alger SpA	0429 781598
6	Berilli SpA	02 87962115
7	Pirti	011 48772659
8	Oas srl	081 25169987
9	Californiana	049 887799652

**Figura 1.14** - Queste due tabelle soddisfano la seconda forma normale.

Un altro caso tipico di violazione della seconda forma normale è il pensare che una condizione data da una situazione contingente sia da interpretare come una condizione generalmente valida. Un esempio è presentato in figura 1.15, dove si ipotizza che ogni fornitore abbia un solo indirizzo. Non è detto però che tale ipotesi sia sempre verificata.

IDFornitore	Indirizzo	Località	Provincia	CAP
1	via Alcide de Gasperi, 16	Ostia	RM	00050
2	via Consalvo, 101C	Napoli	NA	80134
3	via Caccia, 15	Campoformido	UD	33030
4	viale Liguria, 105	Corsico	MI	20094
5	via Piemonte, 25	Monselice	PD	35043
6	via Carlo Forlanini, 2	Milano	MI	20133
7	via Ianza, 34	Grugliasco	TO	10095
8	via B. Longo, 25	Napoli	NA	80147
9	via Piazzola 32/A	Piazzola sul Brenta	PD	35016
10	viale delle Tre Venezie, 58	Ponte Giurino	BG	24030
11	via Codalunga, 68	Serrazzano	PI	56040
12	via Trebaseleghe, 4	Cadoneghe	PD	35010
13	via Raffaello, 2	Castelfranco Veneto	TV	31033
14	via Gambarare, 47	Taglio di Po	RO	45019
15	via Zuccherificio, 15	Este	PD	35042
16	via dei Canneti, 115	Giaveno	TO	10094

**Figura 1.15** - I fornitori possono avere più indirizzi.

### 1.3.3 Terza Forma Normale (3NF)

Una relazione soddisfa la terza forma normale se soddisfa la seconda forma e oltre a questo se tutti gli attributi che non sono chiave sono vicendevolmente indipendenti.

Si consideri ad esempio un'azienda che ha un singolo agente di vendita per ogni regione. Data la relazione mostrata nella figura 1.16, esiste una dipendenza tra la Provincia e l'Agente di vendita, ma nessuno di questi due attributi è chiave.

IDOrdine	NomeAzienda	Provincia	AgenteVendita
10253	Carni Panari	TO	Gianna Levi
10268	Ristoranti Grosselle	RM	Lucia Davolio
10290	Arcibaldoni	NA	Laura Collina
10347	E&G Commerce	NA	Laura Collina
10375	Magazzini Colonia	PD	Michele Sulmona
10386	E&G Commerce	NA	Laura Collina
10389	Star	MI	Margherita Piccoli
10394	Magazzini Colonia	PD	Michele Sulmona
10415	Magazzini Colonia	PD	Michele Sulmona
10423	Gourmet	NA	Laura Collina
10528	Grandi Laghi	PD	Michele Sulmona
10785	Ristoranti Grosselle	RM	Lucia Davolio
10925	Carni Panari	TO	Gianna Levi
10981	Carni Panari	TO	Gianna Levi
11052	Carni Panari	TO	Gianna Levi
11061	Grandi Laghi	PD	Michele Sulmona

**Figura 1.16** - Anche se dipendono l'una dall'altra, né la Provincia né l'Agente di vendita, sono chiave candidata

E' possibile cadere in eccessiva pedanteria nell'applicazione della terza forma normale, poiché per la maggior parte degli indirizzi il CAP può essere determinato sulla base del Comune, per cui la relazione rappresentata in figura 1.17 non è strettamente in terza forma normale.

NomeAzienda	Indirizzo	Località	Provincia	CAP
Moreno Antonio	via Caccia, 15	Campoformido	UD	33030
Nico Velo e figli	via Raffaello, 2	Castelfranco Veneto	TV	31033
Varem SpA	viale Liguria, 105	Corsico	MI	20094
La Frutteria	via Zuccherificio, 15	Este	PD	35042
Fratelli Volpin	via lanza, 34	Grugliasco	TO	10095
Ditta Camillo Benso	via Piemonte, 25	Monselice	PD	35043
Ditta Pasquale Nascibeni	via B.Longo, 25	Napoli	NA	80147
Blauen srl	via Alcide de Gasperi, 16	Ostia	RM	00050
Anna del Giudice Ditta	viale delle Tre Venezie, 58	Ponte Giurino	BG	24030
Eliomotor	via Codalunga, 68	Serrazzano	PI	56040

**Figura 1.17** - Questa relazione non soddisfa del tutto la terza forma normale.

Le due relazioni raffigurate in figura 1.18 sono tecnicamente più corrette ma il beneficio di questo modello è molto limitato e, come vedremo più avanti<sup>1</sup>, alle volte controproducente.

NomeAzienda	Indirizzo	Località	Provincia
Blauen srl	via Alcide de Gasperi, 16	Ostia	RM
Moreno Antonio	via Caccia, 15	Campoformido	UD
Varem SpA	viale Liguria, 105	Corsico	MI
Ditta Camillo Benso	via Piemonte, 25	Monselice	PD
Fratelli Volpin	via lanza, 34	Grugliasco	TO
Ditta Pasquale Nascibeni	via B.Longo, 25	Napoli	NA
Anna del Giudice Ditta	viale delle Tre Venezie, 58	Ponte Giurino	BG
Eliomotor	via Codalunga, 68	Serrazzano	PI
Nico Velo e figli	via Raffaello, 2	Castelfranco Veneto	TV
La Frutteria	via Zuccherificio, 15	Este	PD

Località	Provincia	CAP
Ostia	RM	00050
Campoformido	UD	33030
Corsico	MI	20094
Monselice	PD	35043
Grugliasco	TO	10095
Napoli	NA	80147
Ponte Giurino	BG	24030
Serrazzano	PI	56040
Castelfranco Veneto	TV	31033
Este	PD	35042

**Figura 1.18** - Queste due relazioni soddisfano la terza forma normale.

<sup>1</sup> Si veda il paragrafo 1.4.

Come nelle altre decisioni da prendere nello sviluppo di un modello dati, dovremo essere guidati dalla semantica dei dati. La rilevanza di un'entità nel modello ne determina l'opportunità di svilupparla come relazione indipendente, soprattutto considerando la frequenza di aggiornamento del dato o comunque se si è sicuri che ci siano dei concreti vantaggi. Nel caso in esame, dal momento che il CAP per una città non cambia frequentemente, si potrebbe decidere per un modello meno aderente alla terza forma normale.

#### ***1.3.4 Altre forme normali: la forma normale di Boyce/Codd***

Le tre forme normali appena enunciate furono incluse nella formulazione originaria della teoria relazionale di Codd e nella maggior parte dei modelli dati sono più che sufficienti ad assicurarne l'integrità e la correttezza. Il processo è riassumibile con il concetto che gli attributi devono dipendere "dalla chiave, tutta la chiave, nient'altro che la chiave".

Esistono tuttavia altre forme normali, in particolare quella di Boyce /Codd (BCNF). Si tratta di una variante della terza forma normale che considera il caso speciale di relazioni con chiavi candidate multiple.

Una tabella soddisfa la forma normale di Boyce-Codd se e solo se ogni dipendenza funzionale è basata su una chiave candidata di quella stessa tabella.

La BCNF viene violata quando in una relazione sono presenti più dipendenze funzionali e almeno una di queste è basata su un attributo che non è chiave candidata. Quello che capita più facilmente è che l'attributo su cui si basa la dipendenza è parte della chiave primaria ma non la esprime totalmente. Un esempio è riportato nella figura 1.19. Si tratta di un modello dati in cui ad alcuni agenti venditori sono assegnati dei prodotti dai loro manager.

<b>Id_Manager</b>	<b>Codice_Prodotto</b>	<b>Id_Agente</b>
M-04	P-03	S-14
M-19	P-03	S-07
M-19	P-45	S-02
M-19	P-63	S-02

**Figura 1.19**

Un agente ha un solo manager e un manager può affidare un prodotto ad uno solo dei suoi venditori, mentre uno stesso prodotto può essere affidato a più venditori da manager diversi. Si noti la presenza di un problema di aggiornamento nel caso in cui un agente cambi il suo manager di riferimento senza cambiare i prodotti che tratta. Il cambio di manager per l'agente con codice S-02 comporterebbe la modifica di due righe nella tabella in figura 1.19. Quello cui ci si trova di fronte a causa della non conformità alla BCNF, è infatti una duplicazione dell'informazione che un venditore fa capo ad un manager.

La soluzione a questo tipo di anomalia si ottiene suddividendo questa tabella in due tabelle "Prodotti-Venditori" e "Venditori-Manager", attribuendo a ogni dipendenza funzionale una relativa tabella. Si soddisferebbero così le regole di business, tranne quella che limita l'attribuzione di un venditore per prodotto per ogni Manager. Questo vincolo dovrà giocoforza essere gestito tramite codice procedurale.

Esiste una lunga disputa tra i professionisti nell'implementazione di database relazionali e i teorici dei modelli ER su quale livello di aderenza ai criteri di normalizzazione debba essere considerato sufficiente per poter definire un modello relazione normalizzato o anche pienamente normalizzato.

Un approccio empirico al problema sta nel considerare, nella definizione del modello dati, tutte le forme che abbiamo enunciato e tenerne poi conto nell'implementazione dello stesso, non in quanto affermazioni di principio, quanto come utili strumenti per analizzare e considerare tutte le problematiche che potranno affiorare nell'utilizzo della base dati che ne scaturirà.

## **1.4 La Denormalizzazione ed i Datawarehouse**

### ***1.4.1 Definizione***

Come accennato in precedenza, non sempre un database normalizzato è utile.

Abbiamo detto che le forme normali servono ad evitare tutti i problemi che sorgono nella gestione e nell'aggiornamento di dati, ossia nelle funzionalità primarie richieste nell'utilizzo operativo di un database.

Cosa succede invece se dei dati non interessa minimamente la loro aggiornabilità e la loro integrità viene data per acquisita al momento dell'immissione nel database?

Esistono infatti dei casi in cui queste condizioni si verificano, in particolare quando sui dati vengono effettuate solo operazioni di lettura ed estrazione. Mentre un database normalizzato è funzionale alla gestione operativa dei processi, la denormalizzazione è finalizzata al loro monitoraggio ed analisi.

Normalizzazione e denormalizzazione sono entrambe tecniche accettabili nei loro rispettivi contesti, e precisamente nel contesto operativo e gestionale la normalizzazione, in quello di analisi ed estrazione di informazioni la denormalizzazione.

Se si vuole sapere quanto hanno venduto lo scorso mese le filiali dei supermercati Tizio in Nord Italia o quale prodotto ha preferito acquistare la fascia di clientela con istruzione medio-alta, se si intende usare il database per estrarre informazioni o per controllare l'andamento storico di un dato o indicatore, allora, dato che non saranno necessarie operazioni di aggiornamento, né di inserimento, né di cancellazione ma solo di estrazione, la problematica dell'integrità dei dati e delle anomalie perde di interesse e ne acquista invece tutta un'altra serie di questioni.

Si vorrà, infatti, che l'estrazione dei dati sia il più possibile facile, immediata, efficiente, completa e veloce.

Un database con questi requisiti dovrà avere le seguenti caratteristiche:

- i dati sono integri e consistenti prima di essere immessi nel database
- i dati sono archiviati e mai cancellati, mantengono perciò informazione temporale
- i dati sono raggruppati e integrati tra loro. I dati possono provenire da fonti diverse e vanno resi omogenei prima dell'inserimento
- i dati sono registrati in formati che siano convenienti per l'estrazione e l'analisi
- i dati sono solo per lettura.

I database concepiti per questi scopi solitamente affiancano i database aziendali operativi e sono utilizzati da software specifici per l'analisi e il supporto alle decisioni aziendali (DSS).

Tali database sono chiamati DataWarehouse, che appunto significa "deposito di dati", proprio a sottolineare il fatto che si tratta di una collezione di dati accumulati nel tempo.



### 1.4.2 Come denormalizzare

Denormalizzare significa ammettere che un dato (atomico, calcolato o raggruppamento di altri dati) sia istanziato più volte nel database.

Questo dato duplicato può essere il risultato di un raggruppamento di più tabelle che contengono dati collegati tra loro. Oppure il risultato di un calcolo su campi di una tabella.

Normalizzare induce a decomporre le tabelle, preservandone le dipendenze funzionali, per evitare problemi di integrità a seguito di modifiche dei dati. Il processo di denormalizzazione si svolge, al contrario, tramite l'accorpamento di diverse tabelle in un'unica entità che solitamente rappresenta un soggetto sul quale si vogliono svolgere analisi.

Nell'esempio del noleggio di paragrafo 1.2.1, si è decomposta la tabella globale che registrava i noleggi in tre tabelle distinte per ovviare ai problemi che sorgevano nell'aggiornare, aggiungere e modificare i dati. Se invece vogliamo analizzare, magari ogni mese, l'andamento dei noleggi, si troverà conveniente creare una denormalizzazione (datawarehouse) dei dati ottenendo, dalle tre tabelle normalizzate, una singola tabella congegnata come in figura 1.21.

Modelli				Noleggi			
Codice Auto	Modello Auto	Costo/giorno	Categoria	Targa	Inizio Noleggio	Fine Noleggio	Cliente
fpnt02	Fiat Punto	30	Utilitarie	CA345HG	19/10/2003	23/10/2003	Bianchi Dino
				CA345HG	12/10/2003	14/10/2003	Rossi Mario
				BZ394GF	14/10/2003	18/10/2003	Doni Enzo
				BZ394GF	22/10/2003	27/10/2003	Corso Gianna
				BZ394GF	10/10/2003	13/10/2003	Fanna Andrea

Figura 1.20

NoleggiDenormalizzati							
Targa	InizioNoleggio	FineNoleggio	Modello Auto	CostoGiorno	Cliente	Categoria	GiorniDiNoleggio
CA345HG	19/10/2003	23/10/2003	Fiat Punto	30	Bianchi Dino	Utilitarie	4
CA345HG	12/10/2003	14/10/2003	Fiat Punto	30	Rossi Mario	Utilitarie	2
BZ394GF	14/10/2003	18/10/2003	Fiat Punto	30	Doni Enzo	Utilitarie	4
BZ394GF	22/10/2003	27/10/2003	Fiat Punto	30	Corso Gianna	Utilitarie	5
BZ394GF	10/10/2003	13/10/2003	Fiat Punto	30	Fanna Andrea	Utilitarie	3

Figura 1.21

In figura 1.20 è rappresentato il database normalizzato, mentre la figura 1.21 rappresenta gli stessi dati, ma organizzati in una singola tabella denormalizzata. Si può

effettuare una stima dello spazio occupato considerando il numero di campi presenti nelle due rappresentazioni dei dati. La versione normalizzata consta di 28 campi, quella denormalizzata occupa 35 campi, che arrivano a 40 aggiungendo anche il campo calcolato GiorniDiNoleggio. L'esempio mostra come il processo di denormalizzazione diminuisce il numero di tabelle accorpandole, e di converso, aumenta il numero di campi necessari e perciò lo spazio occupato.

Avendo a disposizione, su meno tabelle, i dati di interesse, estrarre le informazioni risulta non solo più agevole ma anche meno impegnativo per il DBMS e ciò rende possibile effettuare estrazioni che prendono in considerazione moli di dati notevoli ottimizzando i tempi di esecuzione.

Il grafico di figura 1.22 riassume le differenze tra database normalizzati e denormalizzati.

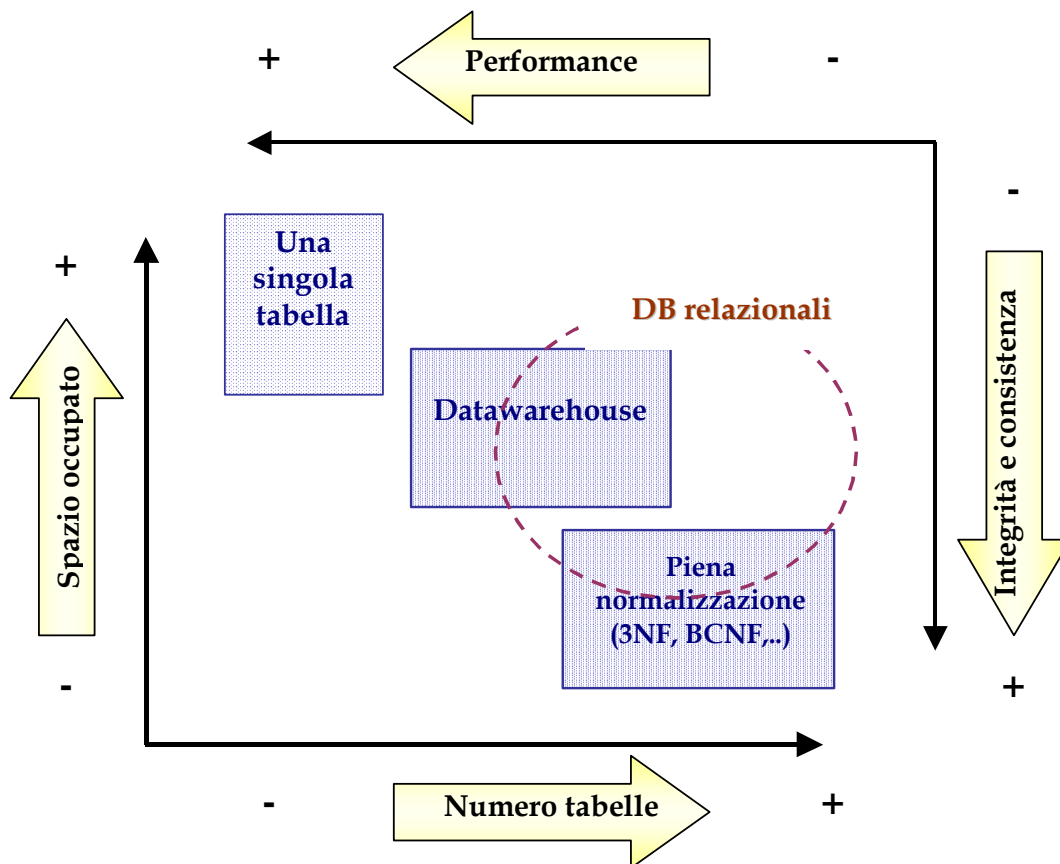


Figura 1.22

Tutti i dati nel database denormalizzato avranno una precisa indicazione temporale, dato che la funzionalità di archiviazione è primaria per gli scopi per i quali viene solitamente creato un database denormalizzato (datawarehouse).

Occorre chiarire però, che il processo di denormalizzazione presuppone obbligatoriamente la realizzazione di un database normalizzato che contenga i dati sui quali creare, come derivazione, accorpamento e archiviazione, una collezione di tabelle denormalizzate.

Il processo che porta alla creazione di un datawarehouse, o comunque ad una collezione di tabelle denormalizzate, si può riassumere in alcuni sotto processi, elencati di seguito.

1. Selezione dei dati di interesse:  
il modello logico è concepito per includere solo i dati che servono a supportare le funzionalità richieste.
2. Aggiunta dell'aspetto temporale alla chiave delle relazioni (tabelle):  
la funzionalità di archiviazione dei dati è un aspetto chiave
3. Aggiunta dei dati derivati e riassuntivi:  
sono da aggiungere i dati di particolare e frequente consultazione
4. Aggiustamento del livello di granularità:  
è importante determinare fino a che livello vanno specificati i dati e dove fermarsi nel raggruppamento.
5. Unione di più tabelle:  
quando dati (cioè soggetti ed eventi di quello che abbiamo chiamato *problem space*) di interesse si articolano su più tabelle, spesso conviene, per obiettivi di analisi e lettura, unire e denormalizzare le stesse in un'unica relazione.



# CAPITOLO 2

## IL VAR

### 2.1 Il rischio di mercato

Da alcuni anni i mercati finanziari sono dominati da incertezza e turbolenza. Gli strumenti finanziari sono sempre più numerosi, hanno differenti caratteristiche e sono espressione di molti mercati. Questa situazione, che è andata via via complicandosi, ha esposto i partecipanti al mercato finanziario a sempre maggiori rischi. Di conseguenza l'esigenza di strumenti che possano misurare il rischio di mercato è divenuta ancora più rilevante. Quest'obiettivo è stato raggiunto solo recentemente.

Alla diffusione della misurazione del rischio di mercato ha contribuito la sua adozione come parametro di riferimento da parte delle istituzioni di controllo e regolazione. Nell'aprile 1995, il Comitato di Basilea propose di permettere alle banche di calcolare i requisiti di capitale per far fronte al rischio di mercato utilizzando modelli sviluppati all'interno dell'istituto che facciano uso di alcuni parametri indicati dal comitato stesso. Poco dopo, anche negli Stati Uniti, nel giugno 1995 la US Federal Reserve e nel dicembre dello stesso anno la SEC (Security Exchange Commission), proposero un approccio simile con l'uso del VaR come possibile strumento per la determinazione del capitale richiesto per accantonamento.

La banca commerciale americana JP Morgan diede una grossa spinta alla diffusione del VaR sviluppando una metodologia ad architettura aperta (piuttosto che proprietaria), chiamata RiskMetrics che rapidamente divenne lo standard di riferimento nella misurazione del rischio.

Fu proprio Tennis Weatherston, al tempo a capo di JP Morgan, a porre la domanda alla base del VaR: "Quanto possiamo perdere domani sul nostro portafoglio titoli?". Si può

notare che si tratta di una misurazione del rischio e non una questione di gestione del portafoglio. Si vuol sapere quanto il portafoglio può perdere se la giornata di domani risulterà particolarmente cattiva dal punto di vista dei mercati.

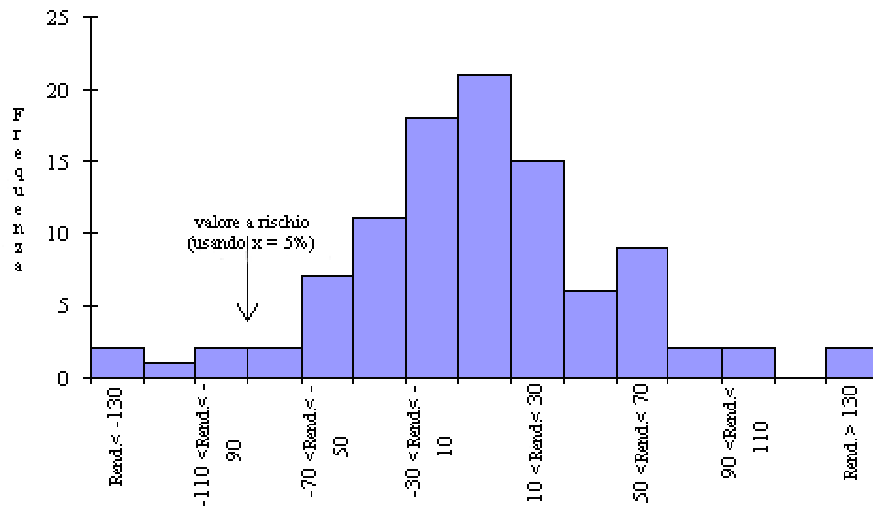
Il VaR è concepito per dare una risposta di tipo statistico-probabilistico alla domanda di cui sopra dando un'interpretazione probabilistica al concetto di "cattiva giornata" che viene definita come la percentuale  $x$  dei peggiori giorni di contrattazione.

Pur basandosi su presupposti della teoria statistica, il VaR, rispondendo chiaramente ad una domanda così semplice, si presta ad essere utilizzato come agevole strumento della misurazione di rischio specialmente da parte delle strutture dirigenziali.

## **2.2 Cos'è il Value at Risk (VaR)**

Value at Risk è una misurazione statistica, singola, riassuntiva delle possibili perdite di un portafoglio titoli. Ci si riferisce a perdite che risultano da movimenti "normali" del mercato. Perdite maggiori del VaR sono subite solo con una probabilità (piccola) specificata. Come risultato delle semplificazioni utilizzate nel processo di calcolo adoperato per la sua computazione, il VaR incorpora in una unica cifra il rischio associato ad un portafoglio composto da titoli diversi, quest'unico numero verrà poi utilizzato nei consigli di amministrazione, nei report interni ed esterni per le istituzioni di controllo. Se si assumono alcune semplici nozioni di statistica, il concetto di VaR risulta semplice da comprendere.

Si supponga di avere oggi un portafoglio di 1 milione di euro basato sul paniere dei titoli Mib30, vogliamo sapere quanto possiamo perdere domani e che la distribuzione dei possibili rendimenti sia quella in figura 2.1.



**Figura 2.1**

Il grafico indica che la probabilità che la perdita sia superiore a 130.000 euro è il 2%, la probabilità che sia tra 110.000 e 130.000 è l'1%, e la probabilità che la perdita sia tra 90.000 e 110.000 è del 2%. Sommando queste probabilità si ottiene che la probabilità che la perdita sia superiore a circa 90.000 euro è del 5%.

Se si considera eccezionale o “anormale” una perdita che si verifica con una frequenza minore del 5% allora la cifra di 90.000 euro sarà il discriminante oltre il quale una perdita non è più considerata ordinaria. Usando questa probabilità del 5% come soglia per la definizione di perdita dovuta a movimenti usuali del mercato, il VaR è approssimativamente pari a 90.000 Euro.

La probabilità del 5% usata come soglia nel nostro esempio, non è sempre valida ma viene invece scelta da chi calcola il VaR a seconda del tipo di profilo di rischio considerato accettabile. Se si fosse scelto il 2% come soglia nell'esempio precedente, allora il VaR che ne sarebbe risultato sarebbe stato di € 130.000, dato che la distribuzione indicava che la perdita supera questo valore solo il 2% delle volte.

Allo stesso modo è il risultato di una scelta la considerazione del periodo di riferimento, che nel caso in esempio è stato di un giorno. Si sarebbero potuti scegliere periodi di 5 giorni o 10 giorni per costruire la distribuzione di profitti e perdite. Periodi più lunghi,

come ci si può aspettare, significano perdite e profitti maggiori e perciò una distribuzione più ampia la cui conseguenza è un valore del VaR più elevato.

Una definizione dei concetti esposti con l'ausilio dell'esempio è la seguente: data una probabilità del  $x$  percento e un periodo di  $t$  giorni (orizzonte temporale), il VaR di una entità è la perdita che si prevede verrà sostenuta con una probabilità  $x\%$  durante i successivi  $t$  giorni. In maniera meno rigorosa si può anche dire che il VaR rappresenta la perdita che ci si aspetta di superare nell' $x\%$  dei successivi periodi di  $t$  giorni. Nel caso di esempio si prevede di avere una perdita superiore a 90.000 € in 5 giorni su 100. Se l'orizzonte temporale fosse stato di 5 giorni (una settimana lavorativa), allora si sarebbe avuta una perdita attesa di € 90.000 per 5 settimane ogni 100.

Il periodo  $t$  preso in esame nel calcolo del VaR solitamente dipende proprio dalla frequenza con la quale gli elementi del portafoglio sono posseduti. Di conseguenza se il trading è giornaliero, come nel caso di istituzioni finanziarie, allora l'orizzonte temporale scelto è di solito di 1 giorno mentre per gli investitori istituzionali e le società non finanziarie il periodo sarà maggiore.

La definizione del VaR implica, tra i suoi assunti, che il portafoglio non subisca modifiche nell'orizzonte temporale  $t$ . Questa assunzione non è sempre verificata poi nella pratica, specialmente quando l'orizzonte temporale si dilata.

Per quanto riguarda la scelta della probabilità  $x$ , questa non ha un legame così diretto con la definizione del VaR ed è il frutto di valutazioni attinenti le politiche di gestione di rischio.

Nell'interpretazione del significato delle misurazioni di VaR, è molto importante tenere presenti entrambi i parametri  $x$  e  $t$  utilizzati, dato che al variare di questi si ottengono cifre significativamente differenti.

Per calcolare il VaR, due sono gli approcci principali: l'approccio Varianza-Covarianza, utilizzato dai **modelli parametrici** e l'approccio di simulazione, utilizzato dai **modelli**



**non-parametrici.** Lo schema che segue rappresenta schematicamente i modelli maggiormente utilizzati.

- Modelli non-parametrici (Approccio di simulazione)
  - Simulazione storica
  - Metodo Monte Carlo
- Modelli parametrici (Approccio Varianza-Covarianza)
  - Portfolio-Normal
  - Asset-Normal
  - Delta-Normal

## **2.3 Modelli non-parametrici**

### ***2.3.1 La simulazione storica***

La simulazione storica è un approccio non parametrico che calcola il VaR non ricorrendo ad ipotesi probabilistiche sulle variabili in oggetto ma basandosi sulla distribuzione storica (osservata) dei rendimenti. Il metodo procede lungo le seguenti fasi. Dato un portafoglio di azioni del quale si vuole calcolare il VaR giornaliero, si raccolgono i rendimenti storici (per esempio, relativi all'ultimo anno) delle azioni in portafoglio. Partendo dal prezzo odierno, si ipotizza che nel giorno successivo si realizzi, per ciascuna azione, il primo rendimento osservato nella serie storica. Si calcola quindi il valore che assumerebbe il portafoglio in base a questi ipotetici valori delle azioni che lo compongono: questo è il primo scenario. Dopodiché, si ripete la procedura partendo sempre dal prezzo di oggi e ipotizzando che nel giorno successivo si realizzi, per ciascuna azione, il secondo rendimento osservato nella serie storica; si valuta il portafoglio in base a questi valori e si definisce così il secondo scenario. E così via. Una volta arrivati all'ultimo giorno del campione, si sarà ottenuta una serie di percorsi di rendimenti giornalieri che determineranno ciascuno un ipotetico valore del portafoglio. Il VaR al 99% (95%) sarà costituito dal 1° (5°) percentile della distribuzione di questi rendimenti simulati. Per calcolare il VaR a 10 giorni, si segue la stessa procedura, utilizzando stringhe di 10 giorni dei rendimenti storici osservati. Questa tecnica di calcolo del VaR consente di riflettere la distribuzione empirica dei rendimenti, preservando al tempo stesso la correlazione tra le singole azioni che

compongono il portafoglio. Il limite principale di questa tecnica è costituito dal fatto che il numero di percorsi simulati è vincolato al numero di osservazioni disponibili (alla serie storica dei rendimenti).

Un'alternativa che permette di svincolare la simulazione dal numero di osservazioni contenute nel campione storico è costituita dall'utilizzo del metodo *bootstrap*, che consiste nell'estrarre casualmente i dati dalla serie storica con reinserimento, anziché estrarre i dati in ordine cronologico senza reinserimento. Non viene quindi mantenuto l'ordine temporale e si può avere il numero di osservazioni desiderate nel campione, per effetto del reinserimento. Sono molteplici i vantaggi: sono già implicate le volatilità storiche e le correlazioni, si riescono a formare code più spesse rispetto ai modelli normali e, soprattutto, si può ottenere un campione grande quanto si vuole.

### **2.3.2 Il metodo Monte Carlo**

L'idea di base del metodo Monte Carlo consiste nella simulazione ripetitiva di percorsi (paths) dei prezzi degli strumenti cui si è interessati. Con un sufficiente numero di osservazioni la distribuzione che ne deriverà convergerà alla distribuzione "vera" del prezzo dell'attività e quindi del portafoglio, in base alla quale sarà possibile calcolare il VaR. Il processo di simulazione avviene seguendo precisi passaggi:

1. scelta del modello per il prezzo dell'attività in oggetto;
2. stima dei parametri per la simulazione con i dati storici disponibili;
3. costruzione di sentieri di prezzi fittizi simulando diversi valori per le variabili casuali incluse nel modello di prezzo;
4. ogni serie di numeri casuali produce un ipotetico valore finale per ciascuno strumento desiderato, e quindi un valore finale per il portafoglio detenuto;
5. calcolo del VaR sulla base della distribuzione simulata.

Questo metodo è molto potente perché può virtualmente gestire qualsiasi tipo di portafoglio, permettendo di calcolare il VaR in situazioni non lineari dove gli approcci normali non sono molto indicati, o di determinare prezzi che dipendono da più variabili

stocastiche<sup>2</sup>. È inoltre adeguato nelle situazioni in cui i parametri variano nel tempo, o le distribuzioni non sono normali (con presenza di curtosi e skewness), e per derivati più complessi dove non esistono soluzioni analitiche (sono un esempio le opzioni americane). L'uso del Metodo Monte Carlo per gestire il rischio dei derivati è stato proposto da Boyle nel 1977 ed è ora diventato molto comune per le istituzioni finanziarie con grosse posizioni in derivati.

## 2.4 I modelli parametrici

I modelli parametrici si basano sul metodo varianza-covarianza, che stima il VaR tramite le varianze e covarianze dei titoli costituenti un portafoglio.

La misura del rischio è concepita come funzione della sua volatilità e, perciò, della sua deviazione standard. L'assunzione della normalità della distribuzione dei rendimenti di un portafoglio titoli permette di esprimere questa volatilità in termini monetari.

Per chiarire come il calcolo della deviazione standard dei rendimenti del portafoglio sia alla base dell'approccio varianza-covarianza, si riprenda l'esempio del paniere di titoli Mib30. Assumendo ancora l'orizzonte temporale di 1 giorno e una probabilità pari al 5%, si può raffigurare la distribuzione dei rendimenti tramite la figura 2.2.

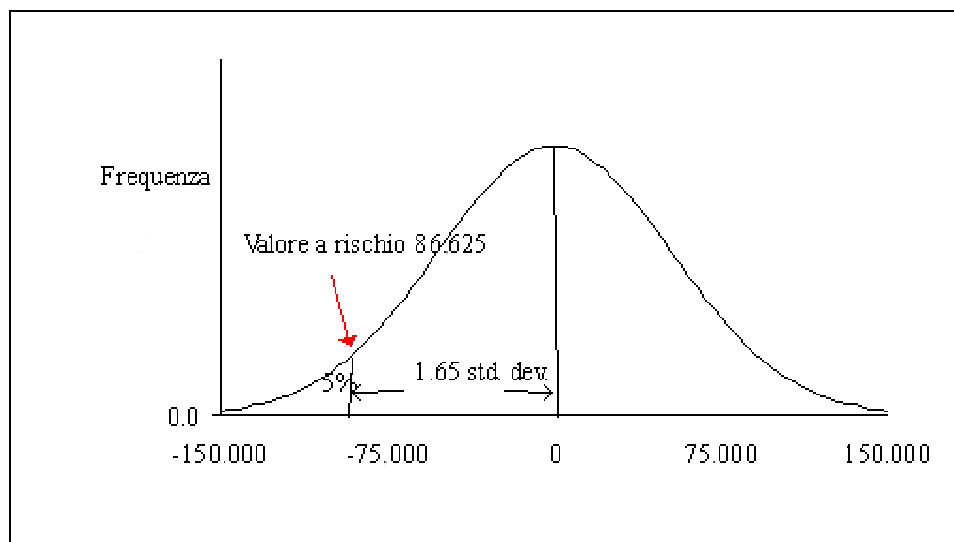


Figura 2.2

<sup>2</sup> Come nel caso delle opzioni asiatiche.

La distribuzione ha media zero, cosa che appare ragionevole dato il breve orizzonte temporale. La deviazione standard, che rappresenta la misura dello scostamento o dispersione della distribuzione, è all'incirca di € 52.500. Una proprietà della distribuzione Normale è che risultati uguali o inferiori a 1,65 volte la deviazione standard hanno una probabilità di accadere del 5%. Perciò, nel nostro esempio:

$$\begin{aligned}\text{VaR} &= 1,65 \times (\text{deviazione standard}) \\ &= 1,65 \times 52.500 \\ &= € 86.625\end{aligned}$$

L'interpretazione di questo risultato porta a sostenere che si avranno perdite uguali o superiori a € 86.625 nel 5% dei giorni. L'assunzione della normalità della distribuzione dei rendimenti permette dunque, come precedentemente affermato, di arrivare ad una quantificazione monetaria.

#### ***2.4.1 Le assunzioni del Var***

Nell'applicazione del metodo varianza covarianza, il calcolo del VaR è generalmente soggetto alla verifica di una serie di assunti statistici. Queste assunzioni non sono sempre perfettamente applicabili alla realtà analizzata ma, dal momento che il VaR rappresenta una misurazione indicativa e non assoluta, vengono considerate accettabili approssimazioni. Le assunzioni o requisiti sono i seguenti:

1. Stazionarietà:  
un rendimento dell'1% può verificarsi con la stessa probabilità in qualsiasi momento. Un altro modo di esprimere questo concetto è dire che le fluttuazioni giornaliere dei prezzi sono indipendenti tra loro (random walk)
2. Non negatività:  
il valore degli strumenti finanziari non può essere negativo.
3. Consistenza nel tempo:  
tutti gli assunti relativi ad un singolo periodo di tempo sono validi anche per periodi multipli. Ciò significa che quello che vale per 1 giorno vale proporzionalmente anche per t giorni. Vedremo che ciò assume particolare significato relativamente al calcolo dei rendimenti.
4. Normalità della distribuzione:

i rendimenti seguono una distribuzione normale a media zero. Questo assunto è il più controverso dato che l'evidenza empirica mostra che la maggior parte degli strumenti finanziari non sono distribuiti normalmente. In particolare quella che si registra è una tendenza delle distribuzioni dei rendimenti ad avere delle code più consistenti di quelle di una distribuzione normale portando di conseguenza ad una propensione del VaR a sottostimare il rischio.

#### **2.4.2 I rendimenti e la regola della radice**

Data l'importanza che rivestono la distribuzione e la volatilità dei rendimenti nel calcolo del VaR, vale la pena approfondire le modalità per il loro ottenimento.

Innanzitutto, esistono due modi per calcolare il rendimento relativo dal periodo t al periodo t+1 per il prezzo s:

- Rendimento semplice

$$R_{t,t+1} = \frac{(s_{t+1} - s_t)}{s_t}$$

- Rendimento composto

$$r_{t,t+1} = \ln\left(\frac{s_{t+1}}{s_t}\right)$$

Il rendimento semplice viola la consistenza temporale dato che calcolando il rendimento tra t e t+2 come:

$$1 + R_{t,t+2} = (1 + R_{t,t+1})(1 + R_{t+1,t+2})$$

e assumendo che i rendimenti dei singoli periodi  $1 + R_{t,t+1}$  e  $1 + R_{t+1,t+2}$  siano distribuiti normalmente, nulla si può dire riguardo alla distribuzione di  $1 + R_{t,t+2}$  essendo prodotto di due variabili normali.

Al contrario, nel caso dei rendimenti composti, il rendimento per due periodi è la somma dei due rendimenti singoli:

$$r_{t,t+2} = r_{t,t+1} + r_{t+1,t+2}.$$

Per una nota regola statistica, una variabile risultante dalla somma di due variabili normali a media zero è distribuita secondo una normale a media zero e varianza che è somma delle due variabili addendi. Perciò data  $\sigma^2$  la varianza del singolo periodo,  $2\sigma^2$  sarà quella per due periodi da cui deriva che la deviazione standard sia  $\sqrt{2}\sigma$ . Generalizzando per un periodo t, la varianza sarà pari a  $t\sigma^2$  e la deviazione standard sarà  $\sqrt{t}\sigma$ . Quest'ultimo termine costituisce, come vedremo, una parte della formula di calcolo del VaR.

### 2.4.3 Il metodo portfolio-normal

Il metodo *portfolio-normal* è il più semplice approccio e calcola il VaR come un multiplo della deviazione standard dei rendimenti aggregati del portafoglio.

Data la normalità dei rendimenti del portafoglio  $\Delta P$ :

$$\Delta P \sim N\left(\mu_p, \sigma_p\right)$$

il VaR è così esprimibile:

$$VaR_p = N^{-1}(\alpha)V\sigma_p\sqrt{t},$$

dove

$\alpha$  = livello di confidenza

$\sigma_p$  = volatilità del portafoglio

$t$  = orizzonte temporale

$V$  = valore del portafoglio.

$N^{-1}(\alpha)$  è la costante che fornisce il valore del quantile della distribuzione normale corrispondente alla probabilità  $\alpha$  del VaR richiesta.

Nel caso del modello *portfolio-normal*, pertanto, si prende in considerazione la volatilità del portafoglio nel suo insieme, senza scomposizioni.

Al fine di individuare quali siano i dati di input richiesti dai processi di calcolo del VaR tramite metodo portfolio-normal è possibile suddividerli nella determinazione del **valore del portafoglio** e della sua **volatilità**, come nella seguente tabella:

	Calcoli necessari	Input necessari
Valore del portafoglio	È dato come valore unico, dal momento che consideriamo il portafoglio nel suo insieme e perciò non necessità di calcoli	Valore totale del portafoglio
Volatilità portafoglio	Varianza del portafoglio $\sigma_p$	Storico dei rendimenti del portafoglio

**Tabella 2.1**

#### **2.4.4 Il metodo asset-normal**

Considerare il rendimento aggregato del portafoglio è una semplificazione che si può ottenere solo se del portafoglio si ha, o si può dedurre, la serie dei prezzi a composizione degli strumenti finanziari fissa. Se si devono calcolare VaR di molti portafogli ad elevata variabilità, ottenerne la serie di prezzi a composizione fissa significa in realtà non una semplificazione ma un appesantimento notevole in fase di calcolo.

Risulta quindi vantaggioso scomporre il calcolo considerando separatamente gli N singoli strumenti finanziari facenti parte del portafoglio sapendo che la serie **P** dei rendimenti del portafoglio nel periodo 1..T può essere espressa sotto forma della matrice **r** NxT dei rendimenti dei componenti e del vettore  $\omega$  dei pesi dei singoli strumenti nel portafoglio.

$$\begin{bmatrix} P_1 \\ P_2 \\ \cdot \\ \cdot \\ P_t \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & \cdot & \cdot & r_{1N} \\ r_{21} & r_{22} & \cdot & \cdot & r_{2N} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ r_{t1} & r_{t2} & \cdot & \cdot & r_{tN} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \cdot \\ \cdot \\ \omega_N \end{bmatrix},$$

ossia  $P = r\omega$ .

Nell'approccio asset-normal si considera, perciò, la volatilità dei rendimenti del portafoglio sotto forma di matrice di varianze e covarianze dei rispettivi rendimenti. Date le assunzioni precedenti avremo che la distribuzione dei rendimenti del portafoglio si può esprimere come:

$$\Delta P \sim N(\mu, \Sigma).$$

Il VaR è dato allora da:

$$VaR_p = N^{-1}(\alpha) V \sqrt{t} \sqrt{\omega \Sigma \omega},$$

dove  $\Sigma$  è la matrice NxN delle varianze e covarianze dei rendimenti:

$$\Sigma = (\sigma_{i,j})_{i,j=1,\dots,n}$$

con

$$\sigma_{i,j} = \frac{\sum_{k=1}^T r_{i,t_k} \cdot r_{j,t_k}}{T} - \bar{r}_i \cdot \bar{r}_j \quad \bar{r}_i \text{ rendimento medio del titolo, } \bar{r}_i = \frac{\sum_{k=1}^T r_{i,t_k}}{T}.$$

La varianza della distribuzione è:

$$\sigma_p^2 = \omega \Sigma \omega$$



che sviluppato è uguale a 
$$\begin{bmatrix} \omega_1 & \omega_2 & \cdot & \cdot & \omega_N \end{bmatrix} \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdot & \cdot & \sigma_{1N} \\ \sigma_{21} & \sigma_{22} & \cdot & \cdot & \sigma_{2N} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \sigma_{t1} & \sigma_{t2} & \cdot & \cdot & \sigma_{tN} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \cdot \\ \cdot \\ \omega_N \end{bmatrix},$$

e il valore del portafoglio V è dato da:

$$V = \sum_{i=1}^N p_i \cdot \omega_i$$

dove

$p_i$  = prezzo dello strumento i al momento della stima del VaR

$\omega_i$  = peso dello strumento i al momento della stima del VaR

La tabella che segue schematizza i calcoli presupposti dal modello Asset-Normal e i dati necessari:

	Calcoli necessari	Input necessari
Valore del portafoglio	Somma dei prodotti tra i prezzi e i pesi dei titoli del portafoglio: $V = \sum_{i=1}^N p_i \cdot \omega_i$	<ul style="list-style-type: none"> <li>▪ Pesi dei titoli</li> <li>▪ Prezzi dei titoli</li> </ul>
Volatilità portafoglio	Prodotto tra il vettore trasposto dei pesi dei titoli, la matrice varianze-covarianze e il vettore dei pesi stesso: $\sigma_p^2 = \omega' \Sigma \omega$	Storico dei rendimenti dei titoli

**Tabella 2.2**

Quando il numero N degli asset che compongono il portafoglio diventa particolarmente alto, allora anche questo approccio di calcolo risulta eccessivamente oneroso dato che

occorre calcolare  $N * (N - 1) / 2 + N$  varianze e covarianze. Ad esempio, supponendo  $N = 100$ , il calcolo richiesto è di 5050 parametri di cui 4950 covarianze.

#### **2.4.5 Il metodo delta-normal**

Nel caso gli strumenti finanziari siano molti, diventa preferibile concentrarsi sui fattori che ne influenzano l'andamento.

Ogni strumento finanziario sarà infatti correlato con uno o più fattori di rischio di mercato. Nel caso di strumenti quotati, il fattore di rischio è prima di tutto dato dal mercato di quotazione, che solitamente è caratterizzato da forte correlazione con i titoli che lo compongono.

Prendere in considerazione i fattori di rischio piuttosto che i singoli elementi del portafoglio, porta ad una consistente semplificazione nei processi di calcolo, senza determinare significative perdite di autorevolezza del VaR.

La dipendenza da un numero limitato di fattori di rischio è implicita nei modelli non-parametrici, mentre è esplicitata nei modelli Varianza-Covarianza che ne traggono un significativo beneficio in termini di snellimento del carico computazionale.

La procedura che permette l'utilizzo dei fattori di rischio è detta *mappatura del rischio*. Ad esempio, un'obbligazione pubblica a 12 mesi si può *mappare* sulla quotazione del tasso a 12 mesi moltiplicato per la sensibilità del bond rispetto a quel tasso.

Ricorrendo ad alcuni fattori di rischio, la dinamica del prezzo del portafoglio può essere espressa come il cambiamento del fattore moltiplicato per il fattore delta:

$$\Delta P = \delta' \Delta F$$

dove  $\delta$  è il vettore  $M \times 1$  dei  $\delta_i$  rispetto a ciascuno dei fattori  $F_i$  che compongono il vettore  $F$ :

$$\delta_i = \sum_{j \in I_i} \omega_j \partial A_j / \partial F_i,$$

con  $\omega_j$  e  $A_j$  che rappresentano, rispettivamente, la quantità ed il prezzo dello strumento finanziario  $j$  del portafoglio "mappato" dal fattore di rischio.

Il VaR è dunque calcolato come:

$$VaR = N^{-1}(\alpha)\sqrt{t}V\sigma_{\delta},$$

e, poiché la deviazione standard, calcolata come funzione dei delta del portafoglio espressi sotto forma di vettore  $M \times 1$ , è  $\sigma_{\delta} = \sqrt{\delta \Sigma \delta}$ , si ottiene:

$$VaR = N^{-1}(\alpha)\sqrt{t}V\sqrt{\delta \Sigma \delta}$$

con  $\Sigma$  matrice di covarianze  $M \times M$  dei fattori di rischio.

Il processo che porta al calcolo del VaR tramite il modello Delta-Normal necessita di una serie di dati e parametri (e di alcuni calcoli intermedi):

	Calcoli necessari	Input necessari
Valore del portafoglio	Somma dei prodotti tra i prezzi e i pesi dei titoli del portafoglio: $V = \sum_{i=1}^N p_i \cdot \omega_i$	<ul style="list-style-type: none"> <li>▪ Pesi dei titoli</li> <li>▪ Prezzi dei titoli</li> </ul>
Volatilità portafoglio	Come nel metodo asset-normal ma con dati che si riferiscono ai fattori di rischio e pesi che, per ogni fattore, sono la sommatoria dei prodotti tra i pesi dei titoli “mappati” e la loro “sensibilità” al fattore. $\sigma_{\delta} = \sqrt{\delta \Sigma \delta}$ dove $\delta_i = \sum_{j \in I_i} \omega_j \partial A_j / \partial F_i$	<ul style="list-style-type: none"> <li>▪ <math>\delta_i</math>: sensibilità dei titoli</li> <li>▪ Storico dei rendimenti dei fattori di rischio</li> </ul>

**Tabella 2.3**

Si assume che il rapporto tra gli strumenti finanziari e i loro fattori di rischio sia lineare e questo non è sempre vero, soprattutto quando si ha a che fare con strumenti derivati, tuttavia per strumenti quotati l'approssimazione è accettabile.

## **2.5 Impiego del VaR per la determinazione degli accantonamenti di capitale**

L'accordo di Basilea del 1988 regola l'ammontare minimo di capitale da accantonare da parte delle banche per far fronte ai rischi derivanti dalla loro attività di credito.

Lo sviluppo dei mercati finanziari, l'incremento nell'importanza del rischio di trading e la maggiore complessità dei prodotti trattati, hanno indotto la revisione di quell'accordo al fine di contemplare una dotazione minima di patrimonio anche per i rischi derivanti dal mercato.

Nell'accordo del 1988 il capitale minimo era quantificato nell'8% degli impieghi nel settore privato indipendentemente dalla loro tipologia temporale e volatilità. Perciò, investimenti a basso rischio in titoli a breve termine, erano penalizzati, così come le erano le banche con investimenti globali e ben diversificati che, per questo motivo, proposero al Comitato di Basilea di considerare metodologie di calcolo del capitale richiesto che prendessero in considerazione i benefici della diversificazione.

Questi elementi spinsero il Comitato, nel gennaio del 1996, a emanare "L'Emendamento dell'Accordo sui requisiti patrimoniali per incorporarvi i rischi di mercato", che prevede due metodi alternativi per il calcolo della dotazione di capitale: un metodo *standard* e un metodo basato su modelli interni che utilizza il VaR.

In alternativa alla metodologia standardizzata, le banche possono calcolare i requisiti patrimoniali sulla base dei propri modelli interni, purché questi soddisfino talune condizioni e siano esplicitamente riconosciuti dall'Autorità di vigilanza nazionale. La Banca d'Italia ha introdotto nel nostro ordinamento quest'ultima possibilità con la pubblicazione nel febbraio 2000 del 13° aggiornamento delle "Istruzioni di vigilanza per le banche".

Le banche devono considerare nel calcolo del VaR un **orizzonte temporale di 10 giorni** usando almeno **12 mesi di dati** e applicare un **intervallo di confidenza del 99%**.

Il requisito patrimoniale da rispettare per le banche che utilizzano il VAR è rappresentato dal **maggior valore tra**:

- la stima attuale del VaR
- la media del VaR dei 60 giorni operativi precedenti, "moltiplicata per un fattore non inferiore 3, eventualmente maggiorata sulla base dei risultati dei test retrospettivi".

Inoltre per risultare idoneo al calcolo del requisito per il rischio specifico, il modello interno deve:

1. spiegare la variazione storica dei prezzi nel portafoglio;
2. riflettere la concentrazione del portafoglio;
3. resistere ad una situazione sfavorevole;
4. essere convalidato da test retrospettivi volti a verificare che il rischio specifico sia valutato in modo adeguato.

Se le banche non sono in grado di stimare un VAR comprensivo delle perdite derivanti da fattori di rischio specifico (rischio di evento e di inadempimento), il requisito patrimoniale richiesto dovrà includere una maggiorazione.

Nell'applicare il modello, le banche sono tenute a condurre quotidianamente dei test retrospettivi su un campione di 250 giorni lavorativi, verificando che le perdite maggiori al VaR stimato non superino l'1%. L'emendamento all'accordo di Basilea individua tre fasce in base al numero di casi in cui i risultati effettivi di negoziazione non sono coperti dalla misura del rischio (cosiddetti "scostamenti").

- Fascia Verde → Da 0 a 4 scostamenti
- Fascia Gialla → Da 5 a 9 scostamenti
- Fascia Rossa → 10 e più scostamenti

La frequenza degli scostamenti determina una maggiorazione del fattore di moltiplicazione del capitale richiesto che il supervisore stabilisce a sua discrezione e che comunque non è mai maggiore di 1.

L'emendamento all'accordo di Basilea permette di coprire con sufficiente sicurezza anche i portafogli più rischiosi, tuttavia i requisiti di backtesting imposti potrebbero influire anche pesantemente sul capitale da accantonare se i modelli interni alle istituzioni bancarie sottostimano il rischio di mercato.



# CAPITOLO 3

## Un modello dati per il rischio di mercato

### 3.1 Un modello per il rischio di mercato

L'intento di questo capitolo è proporre un possibile modello di dati che possa soddisfare le funzionalità richieste nella gestione dei rischi di mercato. In particolare, si vuole basare la gestione di tali rischi sull'approccio Delta-normal che è stato affrontato nel capitolo 2, dato che, come si è visto, tale metodo risulta il più conveniente nell'ottimizzazione dei processi di calcolo.

Progettare un database significa innanzitutto identificare i soggetti e i processi sottostanti alla realtà che si vuole rappresentare e organizzare i dati in modo che sia possibile derivare le informazioni di interesse per gli utenti del database.

Il processo di definizione di un modello di dati presuppone un'analisi dei soggetti coinvolti, delle associazioni tra di essi e delle regole che ne specificano il comportamento (*regole di business*).

#### ***3.1.1 Definizione della realtà da modellare: il rischio di mercato***

La realtà dei mercati finanziari e della negoziazione di portafogli titoli è la cornice in cui si svolgono i fatti oggetto del calcolo del rischio di mercato. Occorrerà perciò tenerne conto nella realizzazione della base di dati, in particolare nella progettazione delle tabelle anagrafiche e nella implementazione dei sistemi di gestione dei dati necessari ai processi di calcolo.

Sostanzialmente lo scopo del database è fornire i dati necessari alla misurazione del rischio di mercato tramite il calcolo del Valore a rischio, il VaR. Oggetto dei calcoli sono portafogli di titoli, per ognuno dei quali (e per aggregazioni degli stessi) si vuole permettere il calcolo del VaR. Un requisito importante che avrà il sistema è quello di dare all'utente la libertà di scegliere i parametri di calcolo. Ricordiamo, infatti, che la misura del VaR si ottiene dati i **parametri** “intervallo di confidenza”  $\alpha$  e “orizzonte temporale”  $t$ . L'output del VaR riassume con una singola valutazione monetaria il rischio del portafoglio in un determinato momento dati i parametri.

Nell'approccio che si è scelto, cioè il metodo Delta-Normal, per semplificare i calcoli della stima della volatilità del portafoglio si utilizzano i fattori di rischio di mercato, che approssimano i movimenti degli strumenti finanziari di cui è composto il portafoglio stesso. Questi fattori di rischio non sono altro che essi stessi degli strumenti finanziari quotati. Ad esempio nel caso di un titolo azionario quale le azioni Fiat, il mercato di riferimento è rappresentato dall'indice Mib30 di cui esiste quotazione.

Si ricorda che la formula del calcolo del VaR tramite il metodo Delta-Normal è:

$$VaR = N^{-1}(\alpha) \sqrt{t} V \sqrt{\delta \Sigma \delta}$$

con  $t$  orizzonte temporale,  $\alpha$  probabilità, entrambi parametri dati dall'utente,  $V$  valore del portafoglio e  $\delta \Sigma \delta$  volatilità calcolata sui fattori di rischio.

Ciò detto, si può schematizzare l'insieme dei soggetti coinvolti nella gestione del rischio di mercato con il seguente diagramma:



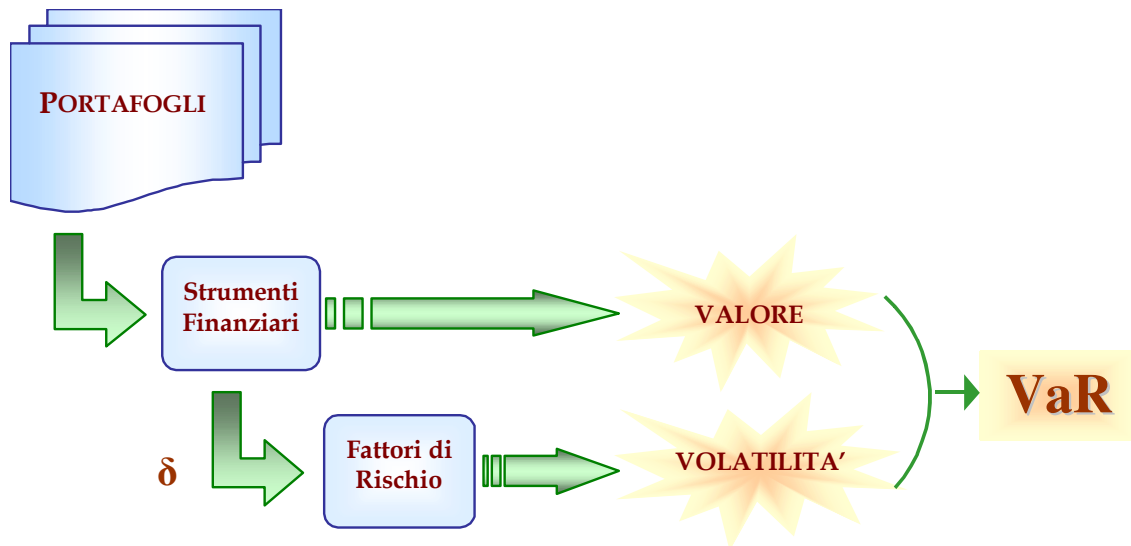


Figura 3.1

I portafogli sono il soggetto cui si riferiscono i calcoli del VaR e sono composti di **strumenti finanziari** che ne determinano il valore, come sommatoria dei singoli valori, e la volatilità tramite l'approssimazione dei **fattori di rischio**, che sono a loro volta strumenti finanziari.

### 3.1.2 Gli input necessari ai processi di calcolo

Il calcolo delle due componenti in cui abbiamo diviso il VaR, valore e volatilità, non è semplice come la loro sintetica definizione sembrerebbe indicare.

Il *valore* del portafoglio è senza dubbio il calcolo più agevole, dato che si tratta solo di sommare i valori dei singoli titoli alla data di cui si vuole calcolare il VaR. Precisamente se si vuole calcolare il VaR per domani, si dovrà prendere come valore del portafoglio quello di oggi. Per ogni strumento finanziario occorre dunque prendere il valore attuale o quello relativo al momento di cui si vuole calcolare il VaR. Se non si hanno i valori delle singole attività finanziarie, questi si possono ricavare conoscendo il prezzo e la quantità di ogni titolo tramite il prodotto

$$V = \sum_{i=1}^N p_i \cdot \omega_i.$$

La *volatilità* è senza dubbio più complessa da calcolare e proprio per la mole di calcoli richiesti si è arrivati alla formulazione del metodo Delta-Normal che, utilizzando i fattori di rischio, determina la volatilità del portafoglio come:

$$\sigma_{\delta} = \sqrt{\delta \Sigma \delta}$$

con  $\delta$  vettore di  $\delta_i$  con

$$\delta_i = \sum_{j \in I_i} \omega_j \partial A_j / \partial F_i$$

$\delta$  è pertanto un vettore in cui ciascun elemento è pari alla somma pesata delle singole sensibilità dei titoli mappati dal fattore di rischio  $i$ .

### ***3.1.3 I requisiti richiesti***

La base di dati che si sta progettando deve rispondere principalmente a due esigenze:

- l'analisi dei rischi di mercato per una gestione oculata delle posizioni in titoli di un portafoglio onde evitare pericoli quali l'imprevedibilità e l'instabilità dei mercati;
- l'utilizzo del VaR nella determinazione del capitale richiesto per accantonamento secondo le direttive degli accordi di Basilea.

Gli utenti del database devono poter censire i portafogli di titoli e poi gestirli modificandone la struttura a proprio piacimento. I portafogli potranno essere formati da attività finanziarie o da altri portafogli con una struttura che può svilupparsi su più livelli gerarchici. Di conseguenza, il database deve essere abbastanza flessibile per consentire il censimento dei molteplici strumenti finanziari che attualmente sono utilizzati nei mercati globali. Inoltre attività finanziarie così diverse potranno essere espresse in diverse divise e ciò richiede che il database contenga i dati necessari ad effettuare le conversioni valutarie per esprimere tutti i valori con una singola valuta.

Deve essere possibile per l'utilizzatore del sistema determinare per ogni strumento finanziario il suo fattore di rischio e il parametro  $\delta$  di sensibilità.

Per ottenere dal sistema il calcolo del VaR, l'utente, oltre a specificare su quale portafoglio effettuarlo, deve essere in grado di stabilire i parametri di input: intervallo di confidenza  $\alpha$  e orizzonte temporale  $t$ .

L'accordo di Basilea richiede, altresì, di disporre di dati storici riguardo la volatilità globale del portafoglio consistente di tutti i portafogli di pertinenza di una istituzione finanziaria. Il sistema dovrà dunque essere in grado di gestire più di una istituzione contemporaneamente, nel caso, ad esempio, che venga impiegato da una società di servizi che gestisca il rischio di mercato per un pool di istituti contabilmente separati. A questo scopo si predisporrà un portafoglio consistente di tutti i portafogli che fanno capo all'istituto di cui occorre determinare il capitale da accantonare secondo i parametri di Basilea.

#### ***3.1.4 Le fonti di dati***

I dati da utilizzare nel database potranno provenire da fonti diverse e il loro formato deve permetterne l'importazione nel database, se è il caso, anche utilizzando dei tool di importazione appositamente congegnati.

I dati devono essere integri e privi di errori che possono inficiare il risultato dei processi di calcolo. Va posta particolare attenzione alla gestione degli eventi societari che influenzano la consistenza delle serie storiche dei titoli. L'effetto di questi eventi è di modificare il prezzo del titolo a seguito di una decisione presa all'interno della stessa società, ad esempio quando, dato un prezzo particolarmente elevato per il titolo, la società, per facilitarne la circolazione, lo fraziona in più titoli. Per garantire la consistenza della serie storiche andrà applicata la stessa trasformazione del prezzo anche ai dati storici e questa attività può risultare di particolare difficoltà dato che richiede un monitoraggio continuo del mercato e non può essere automatizzata ma necessita dell'intervento di personale specializzato. Pertanto occorre utilizzare fonti di dati che garantiscano la gestione di tali eventi e perciò assicurino la consistenza delle serie di dati.

Tra le fonti di dati risultano di particolare utilità quelle che si riferiscono al sistema Risk-Metrics della banca d'affari JP Morgan, che oltre ad essere punto di riferimento nella realizzazione di modelli di calcolo del rischio, offre, in parallelo, anche un servizio di fornitura di dati. Tra i dati forniti da JP Morgan c'è anche la matrice di varianza-covarianze di una mole notevole di strumenti finanziari, aggiornata giornalmente. Questo tipo di dati, come abbiamo visto nel capitolo precedente, aumenta notevolmente all'aumentare degli strumenti finanziari del portafoglio di cui calcolare il VaR. Dato che

il calcolo diventa particolarmente oneroso, avere la possibilità di ottenere le informazioni da una fonte esterna può essere l'unico modo per rendere i processi di calcolo affrontabili dal sistema che gestisce la base di dati.

## **3.2 Le entità**

### ***3.2.1 Notazione e metodo di esposizione***

Nel seguito del capitolo si esporrà l'organizzazione della base di dati partendo dalle entità che ne compongono la struttura portante. Si utilizzerà, per l'esposizione delle tabelle e dei legami di associazione, la possibilità del software Access di raffigurare le relazioni tra le diverse tabelle di un database con un diagramma che richiama la funzionalità degli schemi ER partendo dalle tabelle e le associazioni esistenti. Le tabelle sono raffigurate da rettangoli la cui intestazione ne rappresenta il nome. All'interno è contenuta la lista dei campi relativi, con l'evidenza in grassetto dei campi che compongono la chiave primaria. Le linee che collegano le tabelle sono le associazioni. Le estremità delle linee si situano tra i campi di chiave esterna e primaria e sono etichettate con 1 e infinito ad indicare la cardinalità delle associazioni.

Quando, in un diagramma, una tabella è presente in più di una associazione essa sarà raffigurata più di una volta; ciò sta ad indicare quante istanze della tabella sono presenti nelle relazioni cui fa riferimento il diagramma.

Si è scelto di adottare nel testo la notazione che indica con il carattere grassetto le **tabelle** e con il corsivo i singoli *campi*.

### 3.2.2 Gli strumenti finanziari

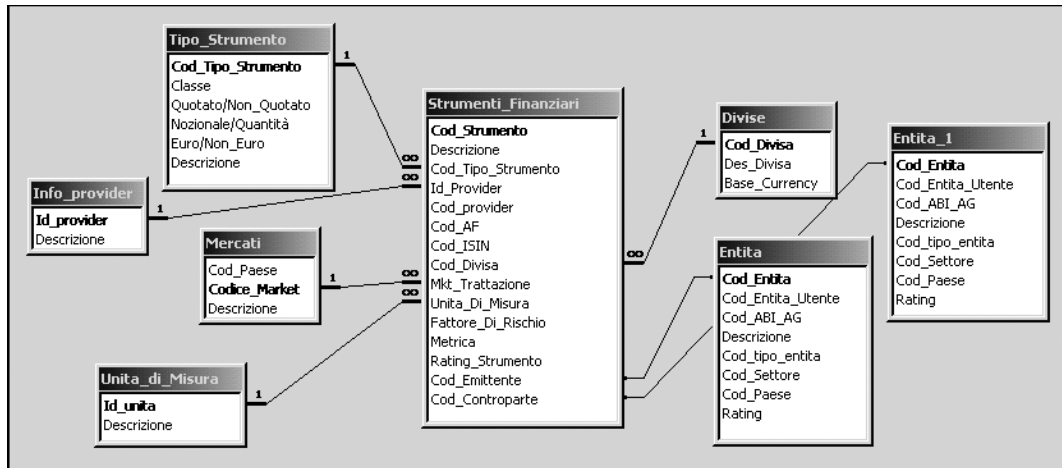


Figura 3.2

La tabella **Strumenti\_Finanziari** è la più articolata e ricca di attributi e associazioni. Nel diagramma qui sopra ne sono evidenziati gli attributi e le associazioni con le tabelle di specificazione degli attributi anagrafici.

Gli attributi sono i seguenti:

- **Cod\_strumento**: chiave primaria della tabella, verrà richiamato come chiave esterna da molte tabelle nel modello dati. Si tratta di un codice interno al database;
- **Descrizione**: testo descrittivo dell'attività finanziaria;
- **Cod\_Tipo\_Strumento**: chiave esterna alla tabella **Tipo\_Strumento** che specifica la categoria alla quale appartiene lo strumento finanziario;
- **Id\_provider**: specifica il provider, società che offre il servizio di fornitura di dati. Fa riferimento alla tabella **Info\_provider** che ne dà una descrizione;
- **Cod\_provider**: specificando anche il codice dello strumento utilizzato dal provider di dati, permette l'aggiornamento dei prezzi perché associa i prezzi scaricati al codice interno al sistema;
- **Mkt\_Trattazione**: identifica il mercato di trattazione del titolo facendo riferimento alla tabella **Mercati**;
- **Unita\_Di\_Misura**: specifica in quale unità di misura è espresso il valore dell'attività finanziaria. Non sempre si tratta di titoli che hanno un valore monetario, dato che può

trattarsi anche di tassi di interesse. Questo campo è chiave esterna della tabella **Unita\_di\_Misura** che ne dà una descrizione testuale;

- *Cod\_AF*: codice che identifica il titolo adoperato internamente all'istituzione finanziaria che utilizza la base di dati;
- *Cod\_ISIN*: codice standard internazionale di codifica dell'attività finanziaria;
- *Cod\_Divisa*: specifica la valuta in cui è espresso il titolo tramite l'associazione con la tabella **Divise**;
- *Codice\_Emittente*: è un campo opzionale che rappresenta il soggetto che ha emesso il titolo;
- *Codice\_Controparte*: rappresenta il soggetto controparte nei derivati e come il campo precedente è opzionale e legato alla tabella **Entita**.

La tabella **Entita**, cui sono collegati *Codice\_emittente* e *Codice\_controparte* censisce, attraverso l'intero modello, tutti quei soggetti che partecipano, con diversi ruoli, alla realtà in esame.

Questi attributi non sono tuttavia sufficienti a rappresentare tutte le caratteristiche di tutti gli strumenti. Alcuni infatti, data la loro particolarità, esigono attributi specifici e perciò necessitano di essere censiti tramite il concetto di sotto-entità. Come abbiamo visto nel Capitolo 1 ad una entità corrispondono delle sotto-entità quando all'entità principale fanno capo degli attributi comuni, mentre a diverse sotto-entità corrispondono gli attributi specifici delle singole categorie. In questo caso avremo, pertanto, dei legami del tipo "is-a" (uno-a-uno) tra la tabella **Strumenti\_finanziari** e tabelle quali: **Bond**, **Derivati**, **Panieri** e **Cambi**, come mostrato nel diagramma di figura 3.3.

La tabella **Derivati** ha un doppio legame con la tabella **Strumenti\_Finanziari** dato che per censire ogni derivato occorre specificare lo strumento finanziario sottostante. Questa associazione è implementata dal campo *Cod\_Sottostante*, chiave esterna di **Strumenti\_Finanziari**.

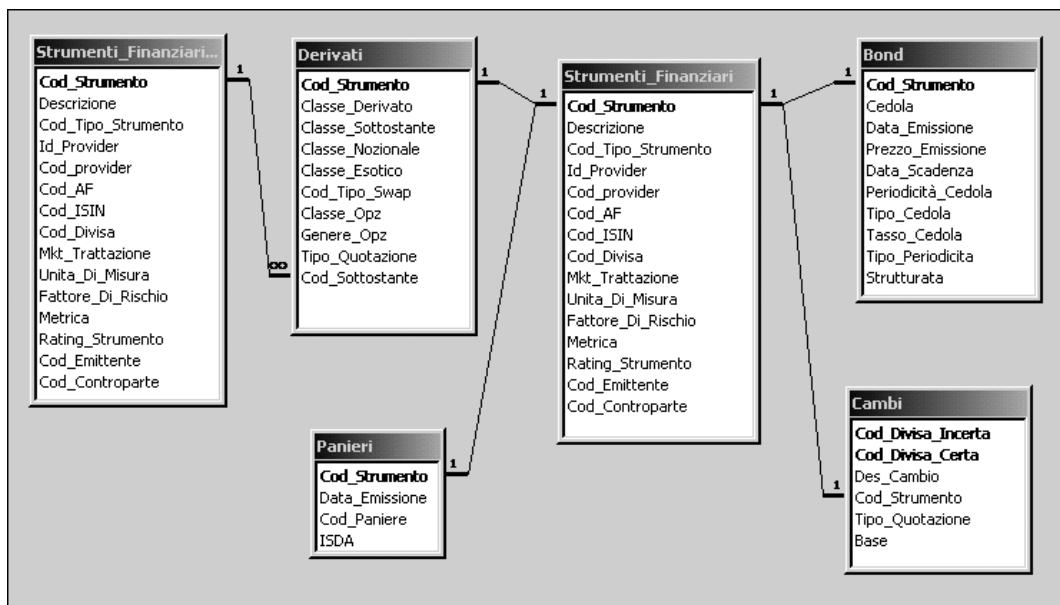


Figura 3.3

### 3.2.3 I portafogli

Nel Capitolo 2 si è definito un portafoglio come un insieme di attività finanziarie. Nell'affrontare i requisiti della base di dati per il VaR ci si rende conto che questa definizione non è sufficiente a coprire tutti i casi dell'utilizzo reale. Spesso, infatti, le istituzioni finanziarie che gestiscono titoli in portafogli vogliono poi aggregarli in portafogli di portafogli. Il diagramma di figura 3.4 oltre che a riportare i legami con le tabelle collegate alle specifiche dei campi, rappresenta la soluzione a questa esigenza: a tale scopo, nella tabella Portafogli è predisposto il campo *Cod\_Ptf\_Predecessore*.

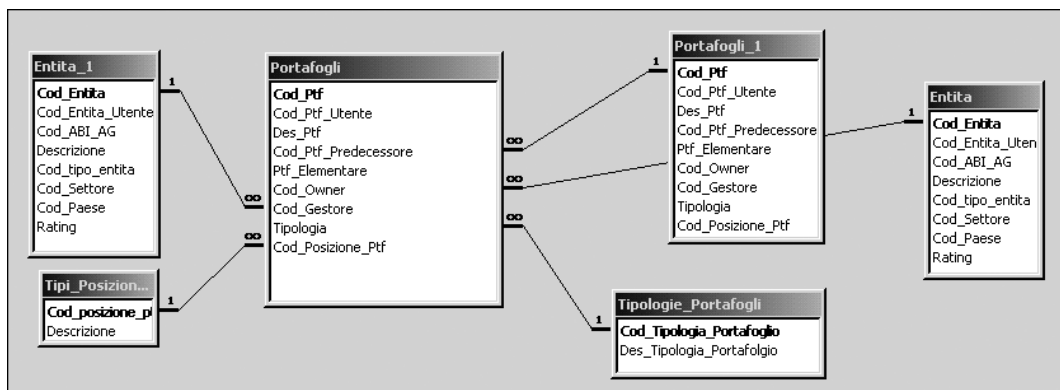


Figura 3.4

Gli attributi della tabella **Portafogli** sono i seguenti:

- *Cod\_Ptf*: chiave primaria, codice interno al database;
- *Cod\_Ptf\_Utente*: codice identificativo interno all'istituzione finanziaria. È unico e perciò chiave candidata;
- *Des\_Ptf*: testo di descrizione del portafogli;
- *Cod\_Ptf\_Predecessore*: implementa un'associazione ricorsiva tra la tabella e se stessa. Si vuole associare ad una riga della tabella Portafogli un'altra riga della stessa tabella tramite l'associazione tra la chiave esterna *Cod\_Ptf\_Predecessore* e la chiave primaria *Cod\_Ptf*.

Ci saranno due possibili valorizzazioni del campo:

- se si tratta di un portafoglio compreso in un altro portafoglio, in questo campo ci sarà il *Cod\_Ptf* del portafoglio che lo contiene;
- se il campo ha valore Null si ha un portafoglio che non è contenuto in nessun altro.

Il portafoglio contenuto può essere a suo volta portafoglio contenitore di altri e sarà il campo *Ptf\_Elementare* a specificare se si tratta di portafoglio che contiene invece dei titoli e non portafogli.

- *Ptf\_Elementare*: campo binario, assume i valori Si/No e specifica se si tratta di un portafoglio elementare;
- *Cod\_Owner*: contiene il riferimento al soggetto che possiede il portafoglio ed è chiave esterna della tabella **Entita**;
- *Cod\_Gestore*: contiene il riferimento al soggetto che gestisce il portafoglio ed è chiave esterna della tabella **Entita**;



- *Tipologia*: specifica la tipologia di portafoglio;
- *Cod\_Posizione\_Ptf*: specifica il tipo di posizione a bilancio.

Nell'intermediazione finanziaria, i valori dei titoli detenuti nei singoli portafogli sono specificati tramite i saldi che ne forniscono l'importo monetario.

Le associazioni del diagramma 3.5 consentono di specificare la composizione del singolo portafoglio. La tabella **Saldi**, unendo **Portafogli** con **Strumenti\_Finanziari**, permette di indicare le istanze dei titoli che fanno parte di un portafoglio e, tramite il campo *Data\_Rif\_Saldo*, li dota anche della dimensione temporale, utile in sede di analisi storica e di backtesting.

La tabella Saldi non solo identifica i componenti dei portafogli ma ne specifica anche la consistenza monetaria tramite il campo *Saldo*. Questo permette di conoscere il valore del portafoglio, che è un elemento importante nel calcolo del VaR. È da notare che il dato monetario fornito dal contenuto del campo *Saldo* ha già valorizzato il prodotto tra peso e prezzo del titolo, la cui sommatoria, come visto nel Capitolo 2, determina il valore del portafoglio. Basterà, perciò, sommare i saldi relativi ad un portafoglio per ottenerne il valore.

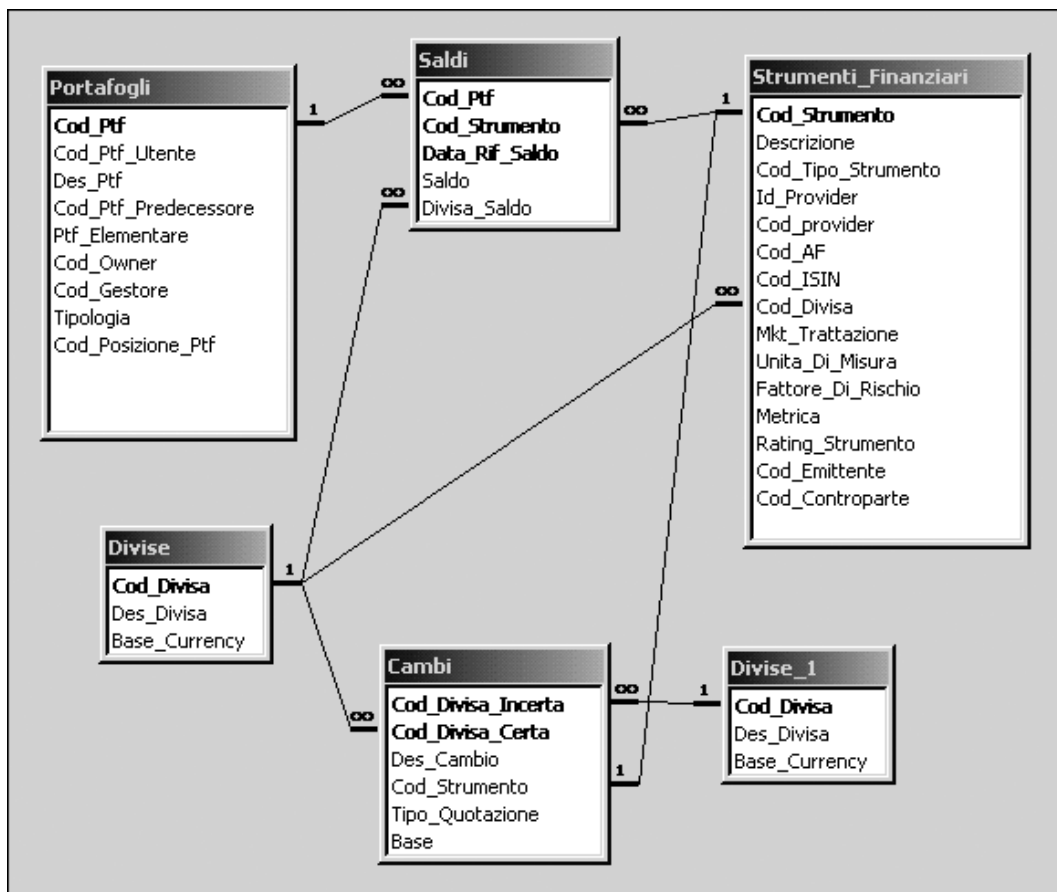


Figura 3.5

Il diagramma di figura 3.5 raffigura le tabelle e associazioni necessarie alla determinazione del valore del portafoglio quando questo è portafoglio elementare, cioè quando contiene strumenti finanziari e non altri portafogli. Nel caso, invece, di un portafoglio di portafogli si dovrà ripercorrerne ricorsivamente la struttura fino ad individuare i portafogli elementari che lo compongono e calcolarne la somma dei valori. Il calcolo del valore del singolo portafoglio si riassume nella somma dei saldi cui si riferisce la tabella saldi ed è perciò di grande importanza. Di seguito se ne schematizza la struttura ed il significato dei campi:

- *Cod\_Ptf*: determina l'appartenenza del saldo ad un portafoglio essendo chiave esterna dell'associazione con la tabella **Portafogli**;
- *Cod\_Strumento*: chiave esterna dell'associazione con la tabella **Strumenti\_Finanziari**, determina il titolo a cui si riferisce il saldo;

- *Data\_Rif\_Saldo*: specifica la data cui si riferisce il saldo e fa parte della chiave primaria della tabella. Permette di storicizzare i dati sui saldi;
- *Saldo*: è l'espressione monetaria del valore del titolo indicato da *Cod\_Strumento* presente nel portafoglio indicato da *Cod\_Ptf*;
- *Divisa\_Saldo*: determina la divisa in cui è espresso il saldo e, attraverso l'associazione con la tabella **Divise**, di cui è chiave esterna, determina il tasso di cambio da applicare al saldo. La definizione del tasso di cambio è data dal prezzo collegato allo strumento finanziario cui fa riferimento il cambio tra la divisa del saldo e la divisa di base del sistema.

#### **3.2.4 I fattori di rischio**

L'approccio di calcolo al VaR Delta-Normal richiede che le attività finanziarie presenti nei portafogli siano mappate dai fattori di rischio che ne approssimano la sensibilità ai movimenti di mercato. Nella realtà della gestione titoli, con la presenza di una moltitudine di portafogli diversi, sorge la necessità di specificare i fattori di rischio non solo a livello di strumento finanziario ma puntualizzando anche diversi fattori a livello di singolo portafoglio. A seconda del portafoglio, una stessa attività finanziaria si può considerare mappata da diversi fattori. Ad esempio un titolo Fiat può essere mappato dall'indice Mib30 in un portafoglio, mentre in un altro portafoglio si può decidere di mapparlo con un indice dell'industria automobilistica globale.

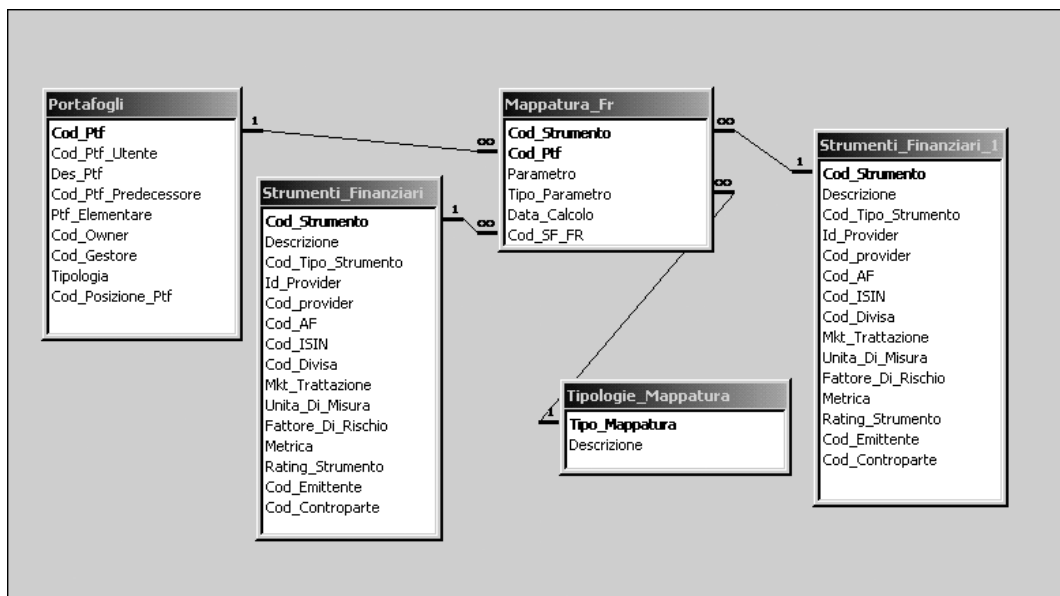


Figura 3.6

Come si vede nel diagramma di figura 3.6, la tabella **Mappatura\_Fr** gestisce la mappatura degli strumenti finanziari con i loro rispettivi fattori di rischio. Si è scelto di considerare gli stessi fattori di rischio come attività finanziarie censite con **Strumenti\_Finanziari**. Di qui la necessità di avere in **Mappatura\_Fr** due chiavi esterne che facciano entrambe riferimento a **Strumenti\_Finanziari**.

Gli attributi sono:

- *Cod\_Strumento*, che identifica lo strumento mappato;
- *Cod\_Ptf*: identifica il portafoglio in cui è presente il titolo mappato e, insieme a *Cod\_strumento*, ne rappresenta la chiave primaria;
- *Parametro*: contiene il valore del parametro. Nell’approccio delta-normal sarà appunto il  $\delta$  che specifica la sensibilità rispetto al fattore di rischio;
- *Tipo\_Parametro*: nella presente implementazione questo sarà sempre  $\delta$  ma questa struttura permette la specificazione di parametri di tipo diverso usati in altri metodi di calcolo specificati nella tabella **Tipologie\_Mappature** di cui è chiave esterna;
- *Data\_Calcolo*: la data in cui è stato calcolato il parametro;
- *Cod\_SF\_FR*: chiave esterna con la tabella **Strumenti\_Finanziari**, identifica lo strumento finanziario da considerare fattore di rischio;

### 3.2.5 I prezzi

Si è visto affrontando i saldi che questi ci permettono di determinare il valore dei portafogli. Il valore di cui abbiamo ancora bisogno nel calcolo del VaR è quello della volatilità. Nel caso Delta-Normal, è necessaria la matrice varianze-covarianze dei rendimenti degli strumenti finanziari che sono fattore di rischio dei titoli dei portafogli. Per ottenere la matrice dei rendimenti occorre la serie storica degli stessi. I rendimenti si ottengono a partire dai prezzi e il diagramma 3.7 espone la tabella e le associazioni che registrano i dati dei prezzi. Come si è già visto, trattando dei saldi si è fatto ricorso alle tabelle **Divise** e **Cambi** per rendere possibile la conversione dei prezzi nella divisa di base. Si è scelto di separare in due tabelle distinte i prezzi dei titoli che hanno una definizione monetaria (nella tabella **Prezzi**) e quelli che fanno riferimento a strumenti finanziari espressi con un tasso, come i tassi di cambio o di interesse (nella tabella **Prezzi\_Tassi**). Questi due prezzi infatti hanno attributi diversi, rispecchiati dalla diversità nei campi delle due tabelle. Rimane comunque possibile accorpate i due tipi di dato in una singola tabella semplificando la struttura della base di dati.

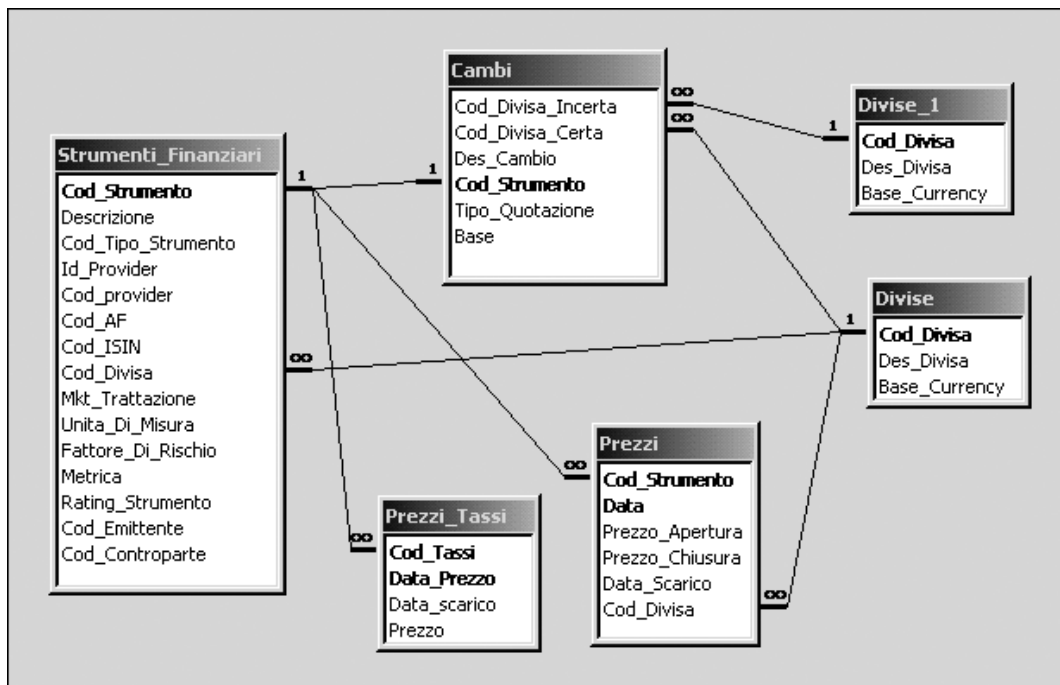


Figura 3.7

La tabella **Prezzi** contiene sia il prezzo di apertura che quello di chiusura per permettere di considerare come prezzo del giorno anche la media tra i due. Sarebbe stato possibile aggiungere anche il prezzo minimo e massimo di giornata ma si è considerata questa opzione come non necessaria ed eccessivamente onerosa dal punto di vista dello spazio occupato dalla tabella.

Gli attributi sono i seguenti:

- *Cod\_Strumento*: strumento a cui fa riferimento il prezzo;
- *Data*: data a cui si riferisce la rilevazione del prezzo;
- *Prezzo\_Apertura*;
- *Prezzo\_Chiusura*;
- *Data\_Scarico*: data in cui si è effettuata la registrazione dei dati;
- *Cod\_Divisa*: divisa in cui si esprime il prezzo.

La tabella **Prezzi** accumula i prezzi dei titoli e ne conserva almeno l'ultimo anno di quotazione. Nel caso di eventi societari che modificano il valore dei titoli occorrerà aggiornare tutti i prezzi dell'attività finanziaria.

Un altro approccio, più conveniente dal punto di vista computazionale, è utilizzare direttamente i dati sulle correlazioni forniti da una fonte di dati esterna, quale ad esempio data-metrics di JP Morgan e adottare come struttura di dati funzionale al calcolo della volatilità quella nel diagramma di figura 3.8 che assume le correlazioni presenti nel database come già calcolate.

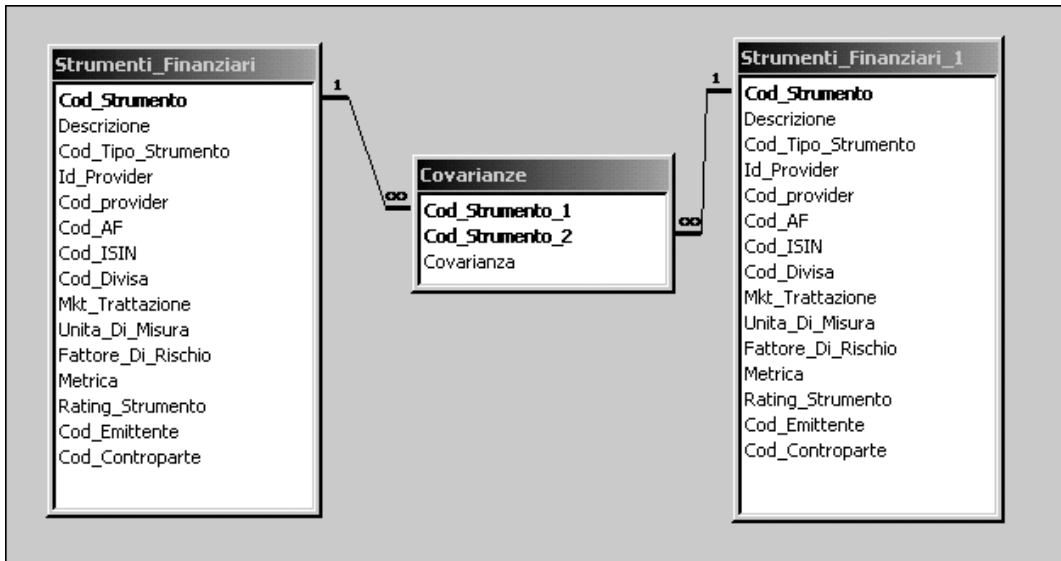


Figura 3.8

### 3.2.6 Lo Storico VaR

Si è visto nel Capitolo 2 che l'accordo di Basilea sulla determinazione del capitale di accantonamento a copertura del rischio di mercato impone all'istituto finanziario che scelga di utilizzare un modello interno di adoperare come parametro il maggiore tra il VaR attuale e la media degli ultimi 60 giorni, moltiplicata per un fattore non inferiore a 3. Questo fattore eventualmente andrà maggiorato di un valore tra 0 e 1 a seconda della bontà del modello interno misurata in base al backtesting che richiede 250 giorni di storico del VaR.



Figura 3.9

La storicizzazione dei VaR, che consente il backtesting secondo le disposizioni di Basilea, avviene con la tabella **Storico\_Var\_Ptf**, che contiene la serie storica di almeno un anno dei VaR ottenuti per ogni portafoglio.

Gli attributi sono:

- *Cod\_Ptf*: determina il portafoglio cui fa riferimento il VaR essendo chiave esterna della tabella **Portafogli**;
- *Data\_Rif\_Calcolo*: è la data per la quale è stato calcolato il VaR che fa parte della chiave primaria della tabella;
- *Valore\_VaR*: contiene il valore del VaR;
- *Cod\_Divisa*: indica la valuta in cui è espresso il VaR ed è chiave esterna della tabella **Divise**.

### 3.2.2 Considerazioni conclusive: il modello dati complessivo

Il diagramma 3.11 è del tipo DFD (Data Flow Diagram) e rappresenta una visione generale dei flussi di dati necessari per il calcolo del VaR. La figura 3.10 rappresenta una legenda per interpretare la simbologia dei diagrammi di tipo DFD.

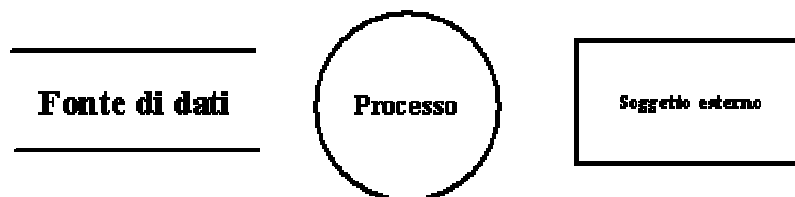


Figura 3.10

Due linee parallele rappresentano un *datastore* (deposito di dati), i cerchi indicano i processi e i rettangoli sono entità e soggetti esterni che interagiscono col sistema. Con delle frecce si rappresentano le interazioni tra questi elementi. Questo tipo di diagrammi è conveniente per rappresentare processi che utilizzano flussi di dati.



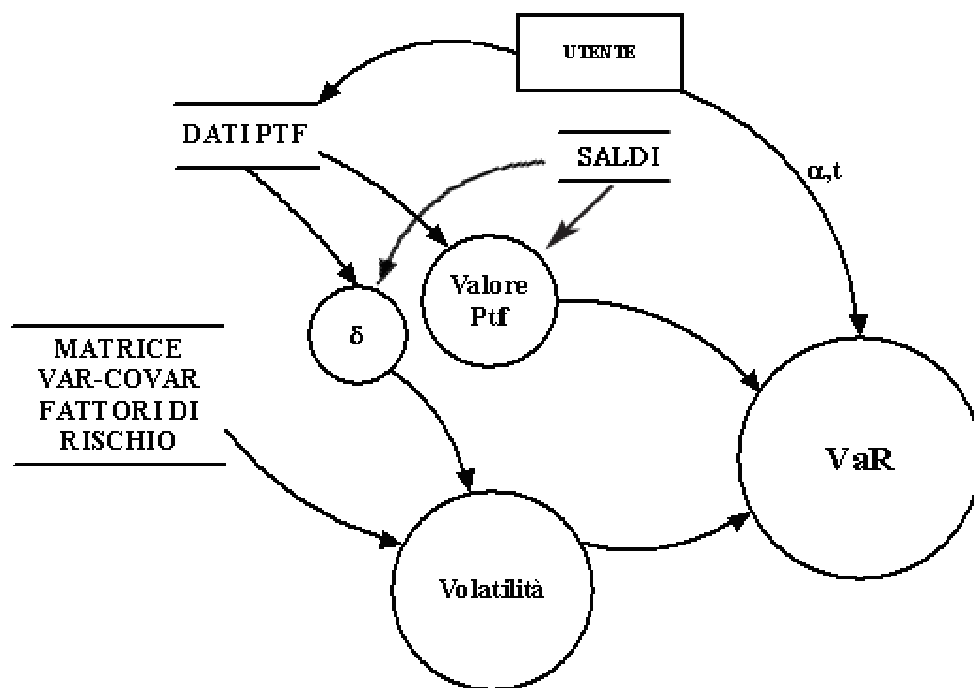


Figura 3.11

In figura 3.11 si ipotizza l'utilizzo di fonti di dati esterne per la matrice di varianze e covarianze e per i saldi degli strumenti finanziari e di un deposito di dati interno che contiene i dati dei portafogli e dei titoli che li compongono, alimentato dagli utenti del sistema. La presenza di un database che contenga i dati delle varianze e covarianze dei titoli che sono fattori di rischio è utile anche nel caso in cui, invece che ricavati esternamente, questi siano calcolati dal sistema. Questo perché i calcoli necessari per la stima delle covarianze sono talmente intensivi da non permetterne il calcolo in tempo reale al momento della stima del VaR quanto, piuttosto, giornalmente, in un momento nel quale il sistema non venga impiegato come ad esempio durante le ore notturne seguenti alla chiusura dei mercati americani.

I flussi di dati necessari al processo della stima del VaR sono riassumibili nel modo seguente:

- L'incrocio dei **dati del portafoglio** (composizione dei titoli presenti) con i **dati dei saldi** fornisce il *valore del portafoglio* di cui calcolare il VaR;

- i **dati del portafoglio** permettono di calcolare il *vettore*  $\delta$  dei pesi dei fattori di rischio;
- infine i **dati delle varianze e covarianze** insieme al *vettore*  $\delta$  danno una stima della *volatilità* del portafoglio.

La stima del VaR è poi ottenuta dal prodotto del valore del portafoglio e della volatilità stimato per i **dati scelti dall'utente**: l'orizzonte temporale  $t$  e l'intervallo di confidenza  $\alpha$ .

I *datastore* indicati nel diagramma DFD sono rappresentati, nel database discusso nel presente capitolo, dalle tabelle raffigurate nella figura 3.12

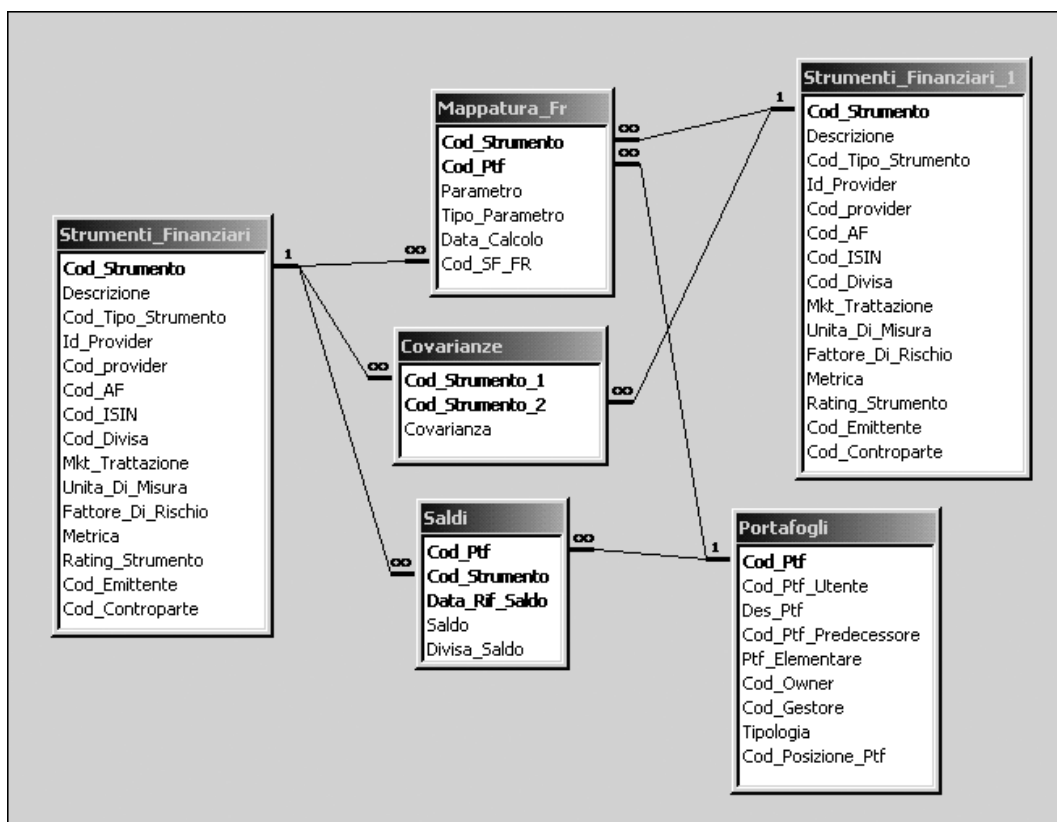


Figura 3.12

Fa parte di questo gruppo di tabelle principali la tabella **Portafogli**, che ne rappresenta il fulcro, dato che di essi si vuole misurare il rischio. I componenti dei portafogli sono identificati dalle righe della tabella **Saldi**, che ne registrano i valori e ne indicano le attività finanziarie componenti nella tabella **Strumenti\_Finanziari**. Questa tabella è

associata alla tabella **Mappatura\_Fr** che fornisce il riferimento al fattore di rischio e al  $\delta$  relativo. Dai fattori di rischio, rappresentati da strumenti finanziari, si ottiene la matrice di varianze e covarianze, attinte dalla tabella **Covarianze** che assieme al vettore dei  $\delta$  fornisce la misura della volatilità del portafoglio. La matrice si può ricavare direttamente da una fonte di dati esterna al sistema o, alternativamente, derivando prima i rendimenti dai prezzi dei fattori di rischio e poi calcolando, da questi, le varianze e covarianze.

Risulta evidente dalla panoramica proposta da questa trattazione che notevole è la quantità di dati coinvolti e, di riflesso, la mole di calcoli necessari alla misurazione e gestione dei rischi di mercato e ciò rende il compito delle strutture e dei sistemi preposti particolarmente delicato e sempre più strategico.



# BIBLIOGRAFIA

- ATZENI P., CERI S., PARABOSCHI S, TORLONE R. (1999), *Basi di dati*, McGraw-Hill
- BANCA D'ITALIA (2003), *Istruzioni di vigilanza per le banche*, Titolo IV
- BARONE E., BARONE-ADESI G., MASERA R. (1999), *Requisiti patrimoniali, adeguatezza del capitale e gestione del rischio*, SANPAOLO IMI
- BASEL COMMITTEE PUBLICATIONS NO. 22 (Gennaio 1996), *Supervisory framework for the use of "backtesting" in conjunction with the internal models approach to market risk capital requirements*
- BASEL COMMITTEE PUBLICATIONS NO. 23 (Gennaio 1996), *Overview of the amendment to the capital accord to incorporate market risks*
- BASEL COMMITTEE PUBLICATIONS NO. 24 (Gennaio 1996), *Amendment to the capital accord to incorporate market risks*
- CELKO J. (2000), *SQL For Smarties: Advanced SQL Programming*, Morgan Kaufmann
- DELANEY K. (2001), *Inside Microsoft SQL Server 2000*, Microsoft Press
- ELMASRI R., NAVATE S.B. (1994), *Fundamentals of Database Systems*, Addison Wesley
- ENGLE R.F. (1991), Statistical Models for financial volatility, *Financial Analysts Journal*, Jan/Feb, 72-78
- FRAIN J., MEEGAN C. (1996), *Market Risk: An introduction to the concept & analytics of Value-at-risk*, Central Bank of Ireland
- GREENSPUN P. (1998), *SQL for Web Nerds*, <http://philip.greenspun.com/sql>
- HENDRICKS D., HIRTLE BANK B. (1997), Capital Requirements for Market Risk: The Internal Models Approach, *FRBNY ECONOMIC POLICY REVIEW*
- HOBEIN G.A., SCHLATTER M., GÜGI P., *Value-at-Risk in portfolio management*, CSIM – CREDIT SUISSE INVESTMENT MODELS, April 1999
- HOLTON G.A. (2003), *Value-at-Risk Theory and Practice*, Academic Press
- INMON W.H. (1996), *Building the data warehouse*, J. Wiley & sons

JACKSON P., MAUDE D.J., PERRAUDIN W. (1998), *Bank capital and Value at Risk*, Bank of England

JOHNSTON T. (2002), A Tutorial on Basic Normalization, Part 7, [dataWarehouse.com](http://dataWarehouse.com)

JOHNSTON T. (2002), Modeling Matters: Logical Data Models, Denormalization and Conceptual Clarification, [dataWarehouse.com](http://dataWarehouse.com)

LINSMEIER T.J., PEARSON N.D. (1996), *Risk Measurement: An Introduction to Value at Risk*, University of Illinois at Urbana-Champaign

POOLET M.A. (2000), Responsible Denormalization, *SQL Server Magazine*, [www.sqlmag.com](http://www.sqlmag.com)

RIORDAN R. (1999), *Designing Relational Database Systems*, Microsoft Press

RISKMETRICS GROUP (1996), *RiskMetrics Technical Document*, J.P. Morgan/Reuters

SHARP J.K. (1998), De-normalized for Speed, *Journal of Conceptual Modeling*, [www.inconcept.com/JCM](http://www.inconcept.com/JCM)

SIMONS K. (2000), The Use of Value at Risk by Institutional Investors, *New England Economic Review*, November/December 2000 21-30

WIENER Z. (1997), *Introduction to VaR (Value-at-Risk) Risk Management and Regulation in Banking*, The Hebrew University of Jerusalem, Business School