

Standard di documentazione  
Modellazione dati

Versione 1.00

## Sommario

<b>Premessa .....</b>	<b>3</b>
<b>Schema Concettuale.....</b>	<b>4</b>
Oggetti della progettazione concettuale.....	4
Generazione di uno schema concettuale di un sistema applicativo .....	8
<b>Schema Logico .....</b>	<b>12</b>
Analisi delle ridondanze .....	16
Eliminazione delle gerarchie .....	17
Accorpamento o partizione di entità e di associazioni.....	17
Scelta degli identificatori principali.....	18
Traduzione al modello logico .....	19
Tecniche di normalizzazione .....	24

## **Premessa**

Questo documento descrive il processo di progettazione di una base dati relazionale, secondo una strategia che prevede, a partire dall'analisi dei requisiti, la progettazione di uno schema concettuale (detto anche schema entità/relazioni), le fasi di semplificazione e traduzione che portano alla costituzione dello schema logico, l'uso delle tecniche di normalizzazione come verifica degli schemi detti.

Scopi fondamentali del documento sono: giungere ad un linguaggio comune in materia, definire la documentazione da produrre per ogni fase del processo di produzione, condividere le definizioni di base e l'uso di alcuni oggetti fondamentali di tale processo come lo schema concettuale, il glossario, lo schema logico, i prodotti case ecc.

Il documento non vuole essere un testo di formazione per i temi trattati, e si rimanda, per una trattazione esaustiva, ai riferimenti di letteratura tecnica elencati in bibliografia.

## Schema Concettuale

### Oggetti della progettazione concettuale

Lo schema concettuale rappresenta la base dati di un progetto applicativo ed è prodotto nella prima fase di definizione del progetto.

Tale rappresentazione è indipendente dall'ambiente tecnologico scelto per la realizzazione della base dati e persegue i seguenti obiettivi:

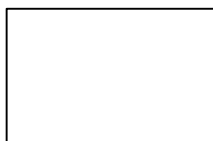
1. rappresentare gli oggetti della realtà del progetto;
2. rendere verificabile l'interpretazione da parte dell'utente con una modalità semplice, rapida e senza equivoci;
3. rendere possibile l'individuazione degli errori logici di progettazione nelle fasi alte di analisi.

In uno schema concettuale sono rappresentabili i seguenti oggetti:

#### Entità:

Ente astratto che modella concetti della realtà di interesse da rappresentare, ottenuto per aggregazione di proprietà atomiche o composte (attributo).

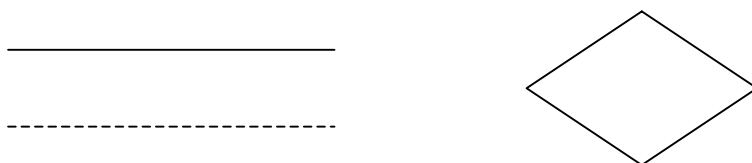
In generale viene rappresentata con un rettangolo.



#### Associazione:

Corrispondenza tra entità.

Graficamente può essere rappresentato con una linea continua o tratteggiata o tramite un rombo.

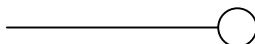


Al fine di non creare confusione tra i termini associazione e relazione si è riservato il primo per la progettazione concettuale e il secondo per la progettazione logica.

Attributo:

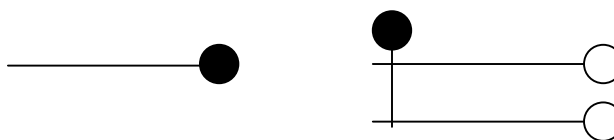
Descrive le proprietà elementari di entità o associazioni che sono di interesse ai fini dell'applicazione.

La sua rappresentazione grafica può essere:

Identificatore di entità/ associazione

Un attributo (insieme di attributi) si dice chiave o identificatore per l'entità/associazione a cui partecipa se è definita una funzione univoca tra l'insieme dei valori dell'attributo (nupla di valori) e le occorrenze dell'entità/associazione. Un identificatore può essere atomico o composto. Una entità/associazione può avere più di un identificatore, questi si dicono chiavi candidate (si dicono attributi primi quegli attributi che appartengono alla formazione di almeno una chiave candidata).

La rappresentazione grafica può essere:

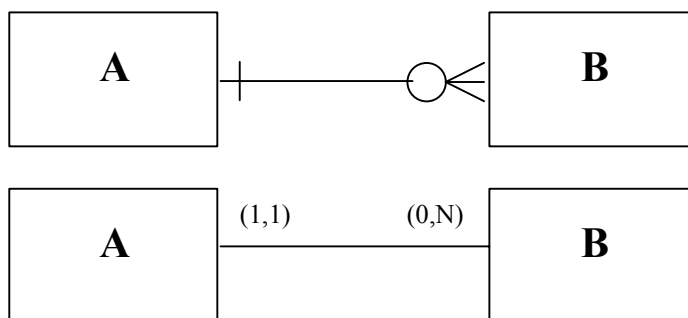


Nel caso in cui l'entità (figlia) è dipendente da un'altra entità (padre), tale dipendenza si manifesta con la presenza nell'entità dipendente di uno o più attributi dell'entità padre. Tali attributi sono detti esterni. Le tipologie di dipendenza possono essere per identificazione o per esistenza a seconda dell'appartenenza o meno degli attributi dell'entità padre alla chiave dell'entità figlia.

Molteplicità:

Si definisce molteplicità dell'entità il numero minimo e massimo di occorrenze con cui ciascuna entità partecipa all'associazione. Il valore minimo indica obbligatorietà se uguale a 1, facoltatività se uguale a 0.

In generale può essere rappresentato sullo schema mettendo tra parentesi tonde i valori di minimo e massimo o tramite simbologia grafica del tipo:



La molteplicità dell'entità A è rappresentata accanto all'entità B e viceversa la molteplicità dell'entità B è rappresentata accanto all'entità A.

Considerando solo i valori massimi delle occorrenze delle due entità che partecipano all'associazione, la stessa si dirà (1,1), (1,n), (n,m).

Si definisce molteplicità dell'attributo i numeri minimo e massimo di valori dell'attributo associati a ogni occorrenza di entità o associazione. La molteplicità di un attributo è pari a (1,1) se a ogni occorrenza dell'entità viene associato un solo valore dell'attributo. Il valore per un certo attributo può essere però nullo oppure possono esistere diversi valori di un certo attributo per una occorrenza dell'entità. In questo caso la molteplicità dell'attributo sarà (0,n).

#### Entità debole o attributiva:

Un'entità è debole se dipende funzionalmente da un'altra entità.

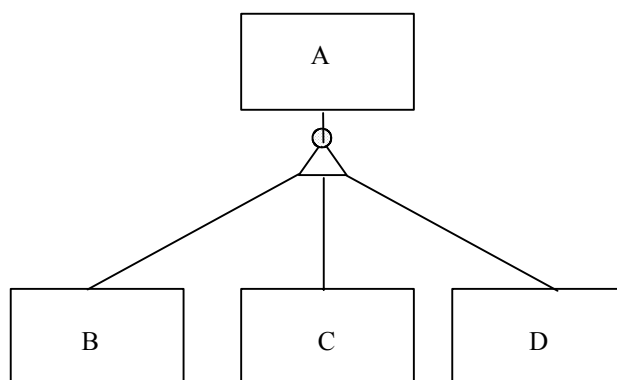
In generale può essere rappresentata con una linea tratteggiata o con simboli quali:



#### Gerarchia:

La gerarchia è una struttura che rappresenta la derivazione di alcune entità (figlie) da un'entità (padre), attraverso successivi livelli di specializzazione. La gerarchia è ottimizzata se le occorrenze dell'entità padre sono date dall'unione delle occorrenze delle entità figlie e queste ultime non presentano tra loro occorrenze in comune.

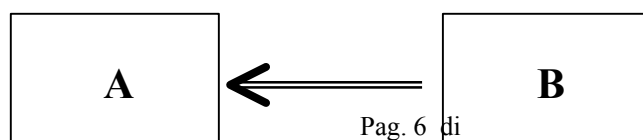
La sua rappresentazione grafica può essere:



#### Sottoentità:

Sottoinsieme delle occorrenze di una entità

In generale può essere rappresentato nel seguente modo (B sottoentità di A):



Gli elementi (o enti) sopra descritti possono essere rappresentati graficamente in vario modo tramite modelli di riferimento disponibili in letteratura (Bachman, Chen, Martin, IDEF1X, Merise, ecc.)

Uno schema concettuale deve rispettare i seguenti requisiti affinché possa garantire un livello di qualità adeguato:

- 1) **Completezza**, lo schema deve rappresentare tutte e sole le proprietà dei concetti della realtà di interesse descritte nei requisiti utente;
- 2) **Correttezza**, le strutture del modello devono essere utilizzate nello schema in modo conforme al significato con cui sono definite nel modello;
- 3) **Minimalità**, ogni aspetto della realtà d'interesse deve essere rappresentato una sola volta nello schema;
- 4) **Leggibilità**, lo schema deve essere rappresentativo della realtà di interesse in modo comprensibile ed autoesplicativo;
- 5) **Consistenza**, lo schema concettuale si dice consistente se verifica tutti i vincoli definiti su di esso.

## Glossario

Parte integrante dello schema concettuale è il **glossario**, inteso come raccolta delle definizioni delle entità, degli attributi e delle associazioni coinvolte.

Nel glossario alle entità si deve associare:

1. Nome dell'entità.
2. Descrizione dell'entità. Deve contenere una descrizione di ciò che rappresenta: definizione, contesto d'uso, utilizzo, eventuali rimandi ad oggetti appartenenti ad altre aree di business. La terminologia da usare deve essere significativa per l'utente.
3. Partecipazione a gerarchia. Nel caso in cui l'entità partecipi ad una gerarchia va indicato:
  - il nome delle eventuali entità figlie,
  - il nome dell'eventuale entità padre,
  - la tipologia della gerarchia.

Per ogni entità elencare gli attributi e per ognuno di essi fornire:

1. Nome dell'attributo.
2. Descrizione completa dell'attributo, eventuali vincoli, fonti di alimentazione, eventuali sinonimi e alias. Se importante corredare la documentazione relativa alla descrizione degli attributi con la storia dei cambiamenti. La storia di alcuni cambiamenti/raffinamenti va annotata con data e nome dell'autore e va esposta in ordine cronologico inverso. Se importante aggiungere i riferimenti alle maschere di inserimento e il sorgente dei dati.
3. Indicare se l'attributo fa parte della chiave dell'entità a cui appartiene.

Le associazioni sono descritte da:

1. Ruolo o verso dell'associazione,
2. Nome entità,
3. Nome dell'associazione,
4. Cardinalità minima e massima,
5. Nome entità associata,
6. Descrizione dell'associazione e degli eventuali vincoli presenti.

#### **Generazione di uno schema concettuale di un sistema applicativo**

Il processo per ottenere uno schema concettuale deve seguire la strategia mista (Bottom up / Top down)<sup>1</sup>.

E' senza dubbio la strategia più usata. Si procede per approssimazioni successive dove passo dopo passo si definiscono tutti gli enti utili. Viene rappresentata la realtà contestuale aggregando e raffinando fino a completare lo schema concettuale.

Si descrivono i passi relativi alla strategia mista:

1. Individuazione di tutte le "funzioni" (processi di lavoro, casi d'uso) previste dal progetto e di cui si deve progettare la base dati.
2. Realizzazione di uno schema concettuale per ognuna di esse in modo da individuare tutte le entità e associazioni utili al progetto secondo una delle strategie suddette;
3. Realizzazione di un glossario che affianchi ogni schema concettuale realizzato al passo precedente
4. Integrazione di tutti gli schemi concettuali realizzati (schema concettuale globale);
5. Integrazione dei glossari con eventuali rettifiche dei nomi relativamente a sinonimie e omonimie;
6. Verifica di completezza, potrà essere effettuata attraverso il riesame delle funzioni individuate durante la fase di analisi dei requisiti Utente controllando la fattibilità delle singole funzioni sullo schema concettuale ottenuto dopo la fase di integrazione;
7. Verifica di consistenza, controllare che tutti i vincoli individuati e rappresentati graficamente nello schema concettuale descrivono correttamente le regole relative al significato dei dati.
8. Verifica di correttezza, tutti gli enti individuati nello schema concettuale siano rappresentati nel modello scelto.

I punti 2 e 3 prevedono all'interno i seguenti passi:

---

<sup>1</sup>

##### **Bottom up**

Si individuano tutte le caratteristiche atomiche della realtà in analisi e si procede per aggregazioni successive in un processo di astrazione continuo definendo tutti gli enti utili al processo applicativo (entità, attributi, associazioni, cardinalità, aggregazioni e sottoinsiemi). Tale strategia risulta idonea per progetti in cui i livelli di raffinamento degli enti utili si è già realizzato (ad esempio progetti di reverse-engineering ).

##### **Top down**

Si individuano inizialmente le macro entità con gli attributi più evidenti e le associazioni più immediate e poi per raffinazioni successive si ottiene lo schema concettuale finale. Tale strategia risulta idonea per progetti i cui requisiti risultano poco definiti.



### Identificazione dei dati

Le informazioni necessarie all'identificazione dei dati sono costituite da tutti i documenti raccolti nella fase di analisi dei requisiti utente. Questi sono generalmente moduli cartacei usati nei processi lavorativi (es: modulo di iscrizione, modulo presenza ecc.), documenti scritti in linguaggio naturale (verbali riunione), archivi magnetici o cartacei preesistenti (documentazione di strutture records, schede lavoro ecc.)

Il primo passo per la creazione dello schema concettuale dei dati consiste nel riesaminare questi documenti e nel compilare un elenco di tutti i dati che il sistema deve trattare.

Dopo aver compilato l'elenco vanno individuate le entità e le associazioni con i relativi attributi.

### Riesame delle entità

Per ciascuna entità vanno analizzate le seguenti informazioni:

- Descrizione dell'entità nell'ambito dell'applicazione
- I processi di lavoro che creano, modificano, usano e rimuovono l'entità
- Verifica delle chiavi candidate
- Ogni altra entità che potrebbe interagire con essa o dipendere da essa (entità debole)
- Le regole applicative e i vincoli riguardanti l'entità
- Gli attributi dell'entità e relativi domini e vincoli (in questo caso sono regole di restrizione del dominio dell'attributo).

### Definizione delle associazioni

Occorre stabilire le associazioni tra le entità individuate e riportarle nello schema Entità/Associazioni. Successivamente vanno analizzate le associazioni in maggior dettaglio. Per ciascuna associazione bisogna determinare:

- La molteplicità dell'associazione: rappresenta il numero di occorrenze con cui ciascuna entità partecipa all'associazione, viene definita con un valore minimo e un valore massimo. La obbligatorietà dell'associazione è determinata dal fatto che ogni occorrenza dell'entità partecipa all'associazione. L'opzionalità dell'associazione è determinata dal fatto che possono esistere occorrenze dell'entità che non partecipano all'associazione. Se l'associazione è opzionale rispetto a tutte le entità allora l'associazione si dice opzionale.
- Gli attributi dell'associazione, loro domini e vincoli (in questo caso sono regole di restrizione del dominio dell'attributo).
- I vincoli dell'associazione: regole scritte in linguaggio naturale che per semplicità dovranno far parte delle descrizioni di glossario

Lo scopo del passo di integrazione, richiamato nei punti 4 e 5, è di costruire, a partire da un insieme di schemi concettuali parziali, uno schema concettuale unico.

Il passo di integrazione, in linea generale può essere applicato ad un numero qualunque di schemi. E' consigliabile integrare due schemi concettuali alla volta ad evitare che la complessità non cresca eccessivamente ed induca in errore.

Per quanto riguarda l'ordine degli schemi da integrare può essere utile considerare un grado di importanza del singolo schema rispetto agli altri, questo per ottenere una migliore convergenza e stabilità nella costruzione dello schema parziale integrato anticipando l'integrazione di schemi con maggiore rilevanza.

In caso di conflitti tra le diverse rappresentazioni di schemi disporre di una soluzione prioritaria, cioè associata allo schema di maggiore rilevanza.

Le attività di integrazione sono:

1. Analisi dei conflitti di nome - Vengono analizzati i nomi assegnati agli elementi nei due schemi ed individuate le possibili incoerenze di omonimia o sinonimia tra loro. L'attività si conclude con l'individuazione di un nome univoco.
2. Individuazione dei conflitti di rappresentazione - Vengono analizzate le diverse rappresentazioni esistenti nei due schemi, si verifica la loro compatibilità e si sceglie una rappresentazione comune.
3. Fusione degli schemi - Risolti i conflitti di rappresentazione si procede con la fusione degli schemi sovrapponendo gli enti comuni ottenendo un interschema.
4. Analisi delle proprietà dell'interschema - Le proprietà interschema sono proprietà che legano classi di oggetti di diversi schemi e vanno sovrapposte nello schema integrato.
5. Verifica ristrutturazione - Sono volte a migliorare la leggibilità dello schema integrato.
6. Verifica di ridondanza - Vanno individuate ed eliminate possibili ridondanze presenti nello schema integrato.

Uno schema concettuale è utile per due aspetti fondamentali, il primo come strumento di ausilio al dialogo tra utente e analista; il secondo come uso prettamente tecnico che ne fa l'analista per giungere alla definizione dello schema logico.

In entrambi i casi si può giungere a vari livelli di approfondimento.

Ad un primo livello di complessità, anche detto schema concettuale scheletro, le entità possono essere semplicemente abbozzate o risultare isolate e quindi non avere collegamenti con altre entità. In questo caso si ha l'obiettivo di individuare per grandi linee un disegno della realtà d'interesse.

Per approfondimenti successivi si possono avere livelli sempre più precisi dello schema fino a ritenerlo conclusivo.

In generale si fa uso di schemi concettuali scheletro o di livello comunque alto di astrazione nelle fasi di analisi dei requisiti. In queste fasi è importante condurre un'attività di riconoscimento tra le entità individuate concettualmente e le informazioni esistenti nella base dati del sistema informatico in essere per capire quali possano essere riutilizzate (condivise o dare vita a forniture interne), quali debbono essere acquisite ex novo e quali debbano essere integrate. In questo caso si suggerisce l'uso documentativo di una semplice tabella che mappi un'entità del concettuale con un oggetto fisicamente esistente nella base informativa (tab.T1).

**Tabella T1 - CARATTERISTICHE DEI DATI DI ORIGINE**

NOME ENTITA'	DESCRIZIONE	NOME ELEMENTO BASE DATI	AREA FUNZ. DI PROVENIENZA	TIPO BASE DATI	NOTE

- ☐ NOME ENTITA': nome della entità definita nello schema
- ☐ DESCRIZIONE : descrizione del contenuto dell'entità considerata
- ☐ NOME ELEMENTO BASE DATI: oggetto fisico (tabella, file, ecc.) appartenente alla base informativa a cui si fa riferimento (se applicabile)
- ☐ AREA FUNZ. DI PROVENIENZA: struttura organizzativa di riferimento
- ☐ TIPO BASE DATI: tipo di organizzazione dati (DB2, VSAM, ORACLE.....)
- ☐ NOTE: ad esempio inserire il riferimento a documenti pertinenti

In fase più avanzata di analisi, in cui il concettuale è da considerarsi definitivo e tutti gli attributi sono stati individuati, si potrà fare uso di uno strumento documentativo (tab.T2) più preciso che approfondisce la tracciabilità tra lo schema concettuale finale e i dati di origine.

**Tabella T2 - MAPPING TRA IL MODELLO CONCETTUALE UTENTE E I DATI DI ORIGINE**

NOME ENTITA'	NOME ATTRIBUTO	FORMATO ATTRIBUTO	NOME FILE/TABELLA ORIGINE	NOME CAMPO FISICO ORIGINE	ALGORITMI DI TRASFORMAZIONE

- ☐ NOME ENTITA': nome della entità definita nello schema concettuale di riferimento
- ☐ NOME ATTRIBUTO : nome dell'attributo appartenente all'entità considerata
- ☐ FORMATO ATTRIBUTO: formato dell'attributo
- ☐ NOME FILE/TABELLA ORIGINE : oggetto fisico appartenente alla base informativa a cui si fa riferimento
- ☐ CAMPI DI INTERESSE: campo fisico che partecipa alla costituzione dell'attributo considerato
- ☐ DESCRIZIONE: descrizione del campo fisico considerato

**N.B.** nel caso un attributo sia composto dalla elaborazione di più di un campo fisico si inserisce una nuova riga in cui le prime due colonne saranno ripetitive delle precedenti o omesse, si scriverà un nuovo campo fisico e la descrizione correlata essa conterrà una indicazione sulla modalità di aggregazione al precedente campo.

Sul mercato sono presenti prodotti che offrono ausilio alla progettazione e documentazione di una base dati.

Gli strumenti attualmente adottati sono:

Erwin;

Cool\_biz;

Designer 2000;

In tutti e tre questi prodotti è possibile usare un formalismo di rappresentazione grafica riconducibile al modello definito da Martin. In Erwin, viene denominato "IE", in Cool\_biz e Designer 2000 è l'unico formalismo adottabile.

Nell'ambito di questo formalismo viene introdotto il concetto di entità associativa che rappresenta una associazione nel caso in cui questa:

- contenga uno o più attributi propri (obbligatoria);
- presenti una molteplicità  $n$  ad  $m$  tra due entità da essa connesse (facoltativa);
- colleghi tra loro più di due entità (obbligatoria);

Per ulteriori informazioni sui tool e sulle differenze di rappresentazione si rimanda alla documentazione associata ai singoli prodotti.

## Schema Logico

Nella prima parte di questo documento abbiamo definito le linee guida della progettazione concettuale di una base informativa anche detta schema “Entità - Relazioni” (E - R).

Con la progettazione concettuale abbiamo individuato: le entità, le associazioni che le correlano, le loro molteplicità, gli attributi (atomici o composti), gli attributi chiave, le strutture gerarchiche (is a), gli insiemi e sotto insiemi di dati. Tali informazioni hanno l’obiettivo di condividere con l’utente le caratteristiche, i significati, la semantica dei dati che dovranno essere gestiti.

In tale fase, per semplificare lo scambio di informazioni tra progettista ed utente, non si deve tenere conto di requisiti di tipo tecnico o di esigenze di tipo funzionale.

Lo schema fisico, invece, nell’ambito del sistema di gestione dati scelto (ORACLE, DB2, ...), dovrà rispondere a requisiti di sicurezza, coerenza, correttezza e caratteristiche prestazionali, richieste dall’utente nei requisiti non funzionali. Dalla progettazione concettuale a quella fisica il passo non è breve. Una banca dati fisica di tipo relazionale si compone di tabelle atte a contenere dati, tabelle di ausilio tecnico che conterranno indici atte a migliorare le prestazioni, inoltre dovrà essere gestita la distribuzione dei dati in settori distinti dei dischi fisici (table space) per rendere possibile l’uso di letture parallele.

Alcune strutture individuate nella progettazione concettuale non sono facilmente riconducibili a semplici relazioni, si ha dunque l’esigenza di progettare schemi di dati intermedi che ci semplifichino la progettazione fisica della banca dati.

La progettazione logica ha lo scopo di semplificare la realizzazione dello schema fisico e di definire in modo più efficiente tutte le informazioni contenute nello schema concettuale. In questa fase progettuale ci si occuperà di scegliere le strutture che ottimizzano lo schema tenendo conto degli *indicatori di prestazione* che influenzano i cosiddetti “costi di un’operazione”: il volume dei dati, degli spazi di memoria reale e di massa occorrenti nelle fasi di attivazione delle funzioni, necessità di denormalizzazioni, ecc.

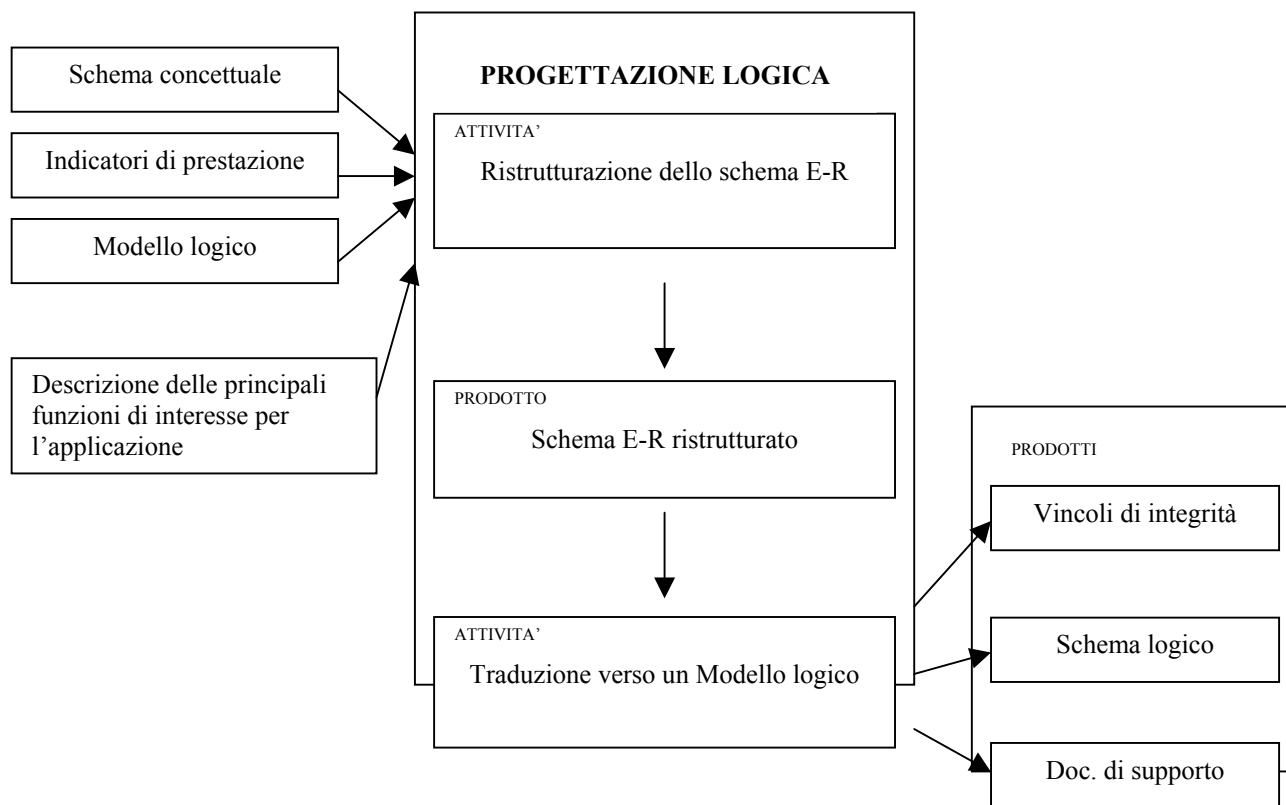
Il passaggio dallo schema concettuale a quello logico avviene attraverso una serie di passi da effettuare in sequenza:

1. analisi delle ridondanze;
2. eliminazione delle generalizzazioni (gerarchie);
3. accorpamento o partizionamento di entità ed associazioni;
4. scelta degli identificatori principali.

I passi elencati produrranno schemi intermedi che non potranno dirsi schemi concettuali anche se ne manterranno le sembianze a tutti gli effetti, in quanto terranno conto dell’architettura tecnico funzionale.

A chiusura di questa fase si otterrà uno schema concettuale ristrutturato che potrà tradursi facilmente in un insieme di relazioni che sarà il vero e proprio schema logico.

Prima di passare alla realizzazione effettiva dello schema fisico verrà fatto un controllo della qualità dello schema logico verificando le regole di normalizzazione e decidendone il livello opportuno.



Gli elementi di ingresso alla progettazione logica sono:

**Schema concettuale:** già definito nella fase di analisi

**Modello logico:** rappresentazione utilizzata da un sistema di gestione di base dati per descrivere i dati tenendo conto delle funzioni del sistema da realizzare.

I modelli in generale usati sono:

modello gerarchico (IMS);

modello reticolare;

altri modelli (Vsam, sequenziale, ecc.)

modello relazionale, di cui ci occuperemo nel presente documento.

Per schema logico di una base dati invece si intende la descrizione degli oggetti del mondo reale coinvolti in un'applicazione per mezzo del modello logico del sistema di gestione di basi di dati adottato.

**Indicatori di prestazione** :Il progettista, nella realizzazione dello schema logico, dovrà fare delle scelte che dipenderanno dalle funzioni che lo schema stesso dovrà supportare basandosi su indicatori di prestazione: volume dei dati, memoria reale disponibile, frequenza di esecuzione dell'operazione etc...

**Descrizione delle principali funzioni di interesse per l'applicazione**: sono le funzioni definite in fase di requisiti, in questo caso sarà possibile considerare il 20% delle funzioni dell'applicazione che come sappiamo dalla regola introdotta da Pareto individuano l'80% dei dati.

Dal processo di progettazione logica si otterranno:

- Schema concettuale (E-R) ristrutturato;
- Vincoli di integrità: regole semantiche che devono essere rispettate affinché la base dati sia una rappresentazione plausibile della realtà d'interesse;
- Schema logico: elenco delle relazioni e loro descrizione, per ogni relazione gli attributi che la compongono con la loro descrizione ed eventuale appartenenza alla chiave;
- Documentazione di supporto: documenti che supportino le scelte eseguite nella realizzazione dello schema logico, tavola dei volumi, tavola delle funzioni, tavola degli accessi per operazione e tavola degli accessi per attributo.

La definizione dei vincoli di integrità riguarda la valutazione delle singole funzioni individuando i seguenti indici di prestazione:

#### Caratteristiche delle funzioni

- il tipo batch o on-line;
- la frequenza (numero atteso, medio, minimo, massimo giornaliero o mensile);
- i dati coinvolti (entità o associazione ed eventualmente i singoli attributi).

#### Volume dei dati:

- il numero di occorrenze in ogni entità e associazione coinvolte;
- la dimensione di ogni occorrenza come somma delle dimensioni dei singoli attributi che ne fanno parte.

Tutte queste informazioni dovranno dare indicazioni al progettista nell'attività di ottimizzazione delle strutture da usare nello schema logico e dovranno dare risposta per ciascuna operazione alle seguenti domande:

quali sono i dati richiesti?

quanti sono i dati (numero di occorrenze)?

quale è la dimensione delle occorrenze richieste?

quali sono le modalità di aggregazione dei dati che ottimizzano il numero delle scritture, letture, modifiche e cancellazioni?

quali le aggregazioni all'interno dello stesso record o blocco fisico?

In questa fase del processo si utilizzeranno tavole dimensionali atte a rispondere ai problemi suddetti. Il formato delle tavole sarà:

## **TABELLA VOLUMI**

NOME ELEMENTO BASE DATI	VOLUME ATTUALE Sorgente	VOLUME INIZIALE Target	VOLUME A REGIME Target	TIPO DI PERIODICITA'	N° DI FORNITURE

Dove:

- ☐ *nome elemento base dati: oggetto fisico individuato nella base dati di origine*
- ☐ *volume attuale: numero di occorrenze attualmente presenti sulla base dati di origine*
- ☐ *volume iniziale: numero di occorrenze previste nella fornitura iniziale sulla base dati targhet*
- ☐ *volume a regime: numero di occorrenze previste sulla base dati targhet a regime*
- ☐ *tipo di periodicità: unità di misura temporale prevista "annuale, mensile, settimanale, giornaliera, a richiesta".*
- ☐ *n° di forniture: n° di forniture previste*

Può essere utile aggiungere altri tipi di colonne a questa tabella ad es: il volume del singolo invio, il volume annuale atteso, il volume massimo o altre caratteristiche ritenute importanti dal capo progetto.

## **TABELLA FUNZIONI**

FUNZIONE	TIPO	FREQUENZA Unità di tempo	DESCRIZIONE TEMPI DI ATTIVAZIONE	DESCRIZIONE FUNZIONE

Dove:

- ☐ *funzione*, si riporta una descrizione sintetica della funzione (ad esempio "CARICAMENTO");
- ☐ *tipo*, si riporta la tipologia di operazione (on line, batch, a richiesta);
- ☐ *frequenza*, si riporta il numero di funzioni svolte nell'unità di tempo specificata (annuale, mensile,...);
- ☐ *descrizione tempi di attivazione*, viene brevemente descritta la modalità di attivazione dell'operazione (ad esempio: se l'operazione è "a richiesta" andrà riportato l'evento scatenante);
- ☐ *descrizione funzione*, descrizione estesa della funzione.

Le funzioni qui rappresentate sono quelle (più significative) individuate nelle prime fasi di analisi, ed inoltre, eventualmente, nuove funzioni che si evidenziano proprio durante l'analisi dei dati.

Può essere utile aggiungere altri tipi di colonne a quelle qui previste ad es: la frequenza attesa, media, minima, massima di esecuzione di una operazione giornaliera o mensile o altre ritenute importanti dal capo progetto.

## **TABELLA FUNZIONI/ACCESSI**

**OPERAZIONE** :< DESCRIZIONE >.....

NOME ENTITA' ACCEDUTA	NOME ATTRIBUTO ACCEDUTO	TIPO

Dove:

- ☐ *operazione*, si riporta una descrizione sintetica dell'operazione (ad esempio "CARICAMENTO");
- ☐ *nome entita' acceduta*, si riporta il/i nome/i dell'entità acceduta dall'operazione;
- ☐ *nome attributo appartenente alla entita' acceduta*, si riporta il/i nome/i dell'attributo acceduta dall'operazione;
- ☐ *tipo*, si riporta la tipologia di accesso dell'operazione (creazione, lettura, scrittura, cancellazione).

Questa tabella è conosciuta più in generale con l'acronimo inglese C,R,U,D: (Create =inserimento, Read = lettura, Update = aggiornamento, Delete = cancellazione) assumiamo questa abbreviazione perché più usato in letteratura.

In questo caso individuiamo a fronte di ogni entità o associazione il numero di accessi per operazione e il tipo di accesso (R read, W write). Le scritture sono normalmente pesate con un coefficiente superiore al valore attribuito alle semplici letture in quanto prevedono una quantità in più di funzioni intrinseche.

In questa tavola si correlano attributi e funzioni indicando per ogni operazione quale tipo di accesso viene eseguito e quale attributo in particolare viene trattato fornendo ulteriori opportunità di comprensione sulle strutture logiche da adottare.

### **Analisi delle ridondanze**

Definiamo ridondanza di un dato la sua ripetizione nell'ambito dello schema concettuale.

Le ridondanze possono essere prodotte da dati derivati o dalla presenza all'interno dello schema di cicli di entità ed associazioni.

I dati derivati possono essere ottenuti attraverso operazioni:

- su dati facenti parte della stessa occorrenza;
- su dati facenti parte di occorrenze di entità o associazioni diverse tra loro;
- dalla contabilizzazione delle occorrenze di una entità o associazione;
- determinate dall'uso di associazioni derivabili dalla composizione di altre associazioni (cicli).



L'uso dei dati derivati presenta vantaggi e svantaggi che dovranno essere valutati considerando gli indicatori prestazionali su detti.

Uno dei vantaggi della presenza di un dato derivato è la riduzione del numero di accessi per il calcolo del dato stesso quando richiesto e quindi una maggior efficienza.

Gli svantaggi sono una maggiore quantità di memoria occupata con un incremento:

- della complessità delle transazioni di aggiornamento;
- dei rischi di inconsistenza;
- del peso documentativo.

### **Eliminazione delle gerarchie**

Il modello logico non permette di rappresentare i costrutti di una gerarchia, si ha dunque bisogno di trasformarli in semplici entità ed associazioni per i quali la traduzione risulta più semplice.

Semplifichiamo la gerarchia a tre entità: P (entità padre), A (entità figlia) e B (entità figlia)

Esistono tre possibili casi fondamentali:

1. tutte le funzioni dell'applicazione nella gran parte dei casi accedono a tutte le entità di una gerarchia (entità padre ed entità figlie) usandone in generale tutti gli attributi, in questo caso si realizzerà una sola entità che riporta l'unione di tutti gli attributi delle entità suddette (P,A,B) unendo ad essi un attributo che dovrà indicare la tipologia di entità descritta nell'occorrenza. Lo schema ristrutturato riporterà una sola entità e le eventuali associazioni correlate alle entità figlie saranno correlate alla nuova entità con cardinalità minima a zero. Si otterrà un uso maggiore di valori nulli ma un numero minore di accessi.
2. Le operazioni dell'applicazione accedono in gran parte ad una delle entità figlie e raramente si fa uso di entrambe le entità; quindi a seconda delle funzioni eseguite, si ha bisogno dei dati dell'una o delle altre entità. In questo caso lo schema ristrutturato conterrà due entità che riporteranno in una, l'unione degli attributi delle entità (P,A) e nell'altra l'unione degli attributi delle entità (P,B). Le eventuali associazioni con l'entità padre si sdoppieranno puntando direttamente alle nuove entità costituite. Questa soluzione è possibile solo in caso di generalizzazione totale. Si ha, in generale un risparmio di memoria, inoltre gli attributi non assumono mai valori nulli.
3. Se la generalizzazione non fosse totale e le funzioni selezionassero anche occorrenze individuate solo nell'entità padre, la soluzione di ristrutturazione è: creare un'entità per ogni entità figlia, una per l'entità padre ed una associazione per ogni collegamento (Padre, figlia) con il vincolo che nessuna occorrenza possa essere individuata da più di una associazione.

I tre casi elencati non sono esaustivi né dei casi possibili né delle possibili soluzioni ma sono certamente i più usuali.

### **Accorpamento o partizione di entità e di associazioni**

Uno dei punti di vista importanti della progettazione delle banche dati è quello di associare le funzioni alle tipologie di accesso agli attributi e alle loro frequenze d'uso, cercando di caricare in memoria, nei momenti opportuni, solo ciò che serve. Si genera così un nuovo modo di aggregare gli attributi che confrontato con lo schema concettuale darà vita ad una serie di accorpamenti e partizioni.

Nel caso di banche dati relazionali avere delle tabelle con pochi attributi permette la lettura contemporanea di molte occorrenze rendendo molto più efficiente l'applicazione. D'altro canto

usare molte tabelle e correlarle durante una operazione significa eseguire operazioni di join che in genere non sono molto efficienti.

#### *Partizionamento verticale di una entità.*

Data una entità divisibile in due sotto insiemi di attributi e date due funzioni definite su di essi tali che si possano associare univocamente le funzioni ai sotto insiemi, allora l'entità potrà essere partizionata in due entità correlate da una associazione con molteplicità uno a uno. Si parla in questo caso di decomposizione verticale dell'entità, si avrà un risparmio di memoria reale durante il processo e in base a questo migliori prestazioni.

Se una entità contiene un attributo multivalore essa va trasformata perché non ci sono modi nell'ambito del modello logico di rappresentare una struttura simile. Dalla entità di origine si scompone in due entità una sarà uguale alla originaria a meno dell'attributo multivalore e l'altra si otterrà dall'entità originale unendone gli attributi chiave all'attributo multivalore.

La nuova entità conterrà come occorrenze i valori dell'attributo multivalore origine e verrà collegata all'altra entità ottenuta dalla scomposizione attraverso una associazione con molteplicità uno a molti.

#### *Partizionamento orizzontale di una entità.*

Data una entità divisibile in due sotto insiemi di occorrenze determinate da specifici valori di uno o più attributi e date due funzioni definite su di essi tali che si possano associare univocamente le funzioni ai sotto insiemi allora l'entità di origine potrà essere partizionata in due disgiunte. Si parla di decomposizione orizzontale dell'entità, in questo caso le due entità conterranno gli stessi attributi. Questo tipo di partizione provoca la duplicazione di tutte le associazioni a cui l'entità originaria partecipava.

#### *Accorpamento di entità.*

Durante la revisione dello schema concettuale ci si può rendere conto del fatto che due entità vengano accedute, durante gran parte delle funzioni, sempre contemporaneamente. Si ha dunque la esigenza di accorpare le due entità in una unica migliorando l'efficienza anche se c'è il rischio di generare attributi a valore nullo.

Un accorpamento può avvenire normalmente se le entità sono associate con molteplicità uno a uno, raramente in casi di molteplicità uno a molti mai nel caso di molteplicità molti a molti.

### **Scelta degli identificatori principali**

Un identificatore è un insieme di attributi che identificano univocamente le occorrenze di un'entità.

Un'entità può avere più di un identificatore.

Scegliere nell'insieme degli identificatori di una entità l'identificatore principale è una delle funzioni fondamentali in quanto tramite questi attributi si determinano i legami tra i dati in associazioni diverse.

Nella scelta vanno seguiti i seguenti criteri:

- gli identificatori principali non debbono contenere attributi a valori nulli;
- è conveniente scegliere in linea di principio l'identificatore composto da meno attributi possibile;
- è conveniente scegliere l'identificatore usato nel più alto numero di funzioni

In uno schema concettuale un identificatore di una entità può essere, come abbiamo già visto, esterno. Nello schema logico non è previsto questo tipo di identificatore, il problema si risolve creando nuovi attributi nella relazione che traduce l'entità rendendo interni gli identificatori esterni. Le chiavi esterne (FK) permettono di collegare tra loro occorrenze di relazioni diverse e costituiscono un modo, detto per valore, di realizzare delle associazioni.

#### Traduzione al modello logico

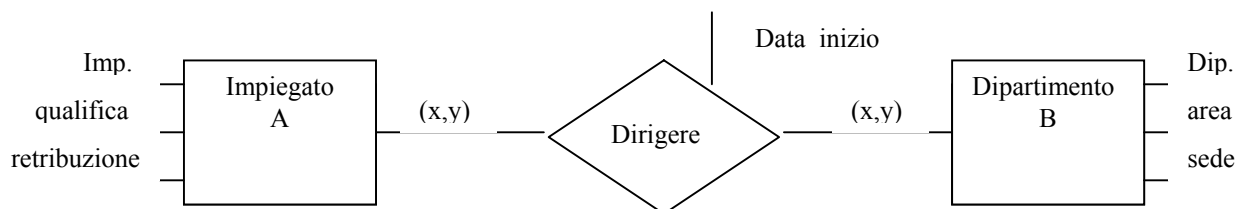
Con le attività precedentemente descritte abbiamo ottenuto un nuovo schema E-R ristrutturato che sarà facilmente traducibile ad uno schema logico.

Vediamo ora quali sono le regole generali di traduzione:

1. ogni entità diventa una relazione, mantiene gli stessi attributi e definisce “chiave” gli attributi che compongono l'identificatore principale (va comunque tenuta nota degli identificatori non principali per realizzare gli indici secondari in fase di progettazione fisica);
2. ogni associazione diventa una relazione (ad eccezione dei casi particolari successivamente descritti), manterrà gli attributi propri e definirà chiave l'unione delle chiavi delle entità coinvolte.

L'analisi delle molteplicità con cui le entità partecipano alle associazioni determina se e come le entità e le associazioni vengono tradotte in relazioni.

Consideriamo il caso generale rappresentato nella figura sottostante.



In questo tipo di costrutto applicando le regole generali precedentemente fornite, l'associazione deve essere tradotta in una relazione con una colonna per l'attributo e la chiave formata dalla unione delle chiavi delle due entità associate. Devono essere poi tradotte le entità A e B in altrettante relazioni.

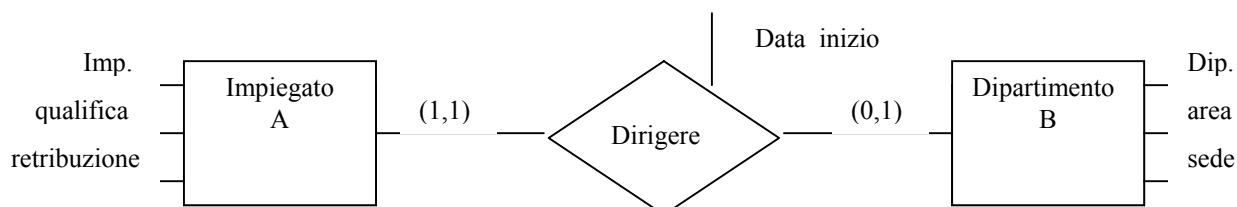
Lo schema logico che si ottiene è:

Impiegato (Imp., qualifica, retribuzione)  
Dipartimento (Dip., area, sede)  
Direzione (Imp., Dip., data\_inizio,.)

Analizziamo ora i casi particolari.

### Associazione binaria uno a uno:

Se una delle entità è obbligatoria (cardinalità 1,1) e l'altra è opzionale (cardinalità 0,1), lo schema concettuale ristrutturato sarà:



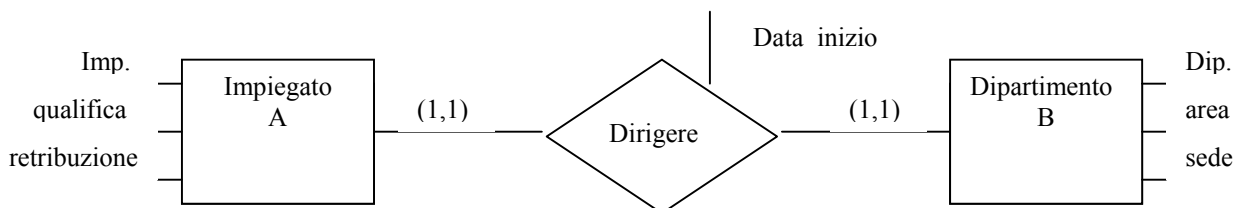
In questo tipo di costrutto applicando la regola, precedentemente data al secondo punto, l'associazione dovrebbe essere tradotta in una relazione con una colonna per attributo e la sua chiave dovrebbe essere formata dall'unione delle chiavi delle due entità associate. Dovrebbero essere poi tradotte le entità A e B. Data la cardinalità di B (obbligatoria), esiste una corrispondenza biunivoca tra le occorrenze di B e le eventuali occorrenze dell'associazione, si ritiene ottimale unire le due relazioni dove però la chiave di A non potrà far parte della chiave della nuova entità perché opzionale, il che potrebbe dare vita alla presenza di valori nulli nella chiave della nuova relazione. L'entità A verrà tradotta nei canoni della definizione data.

Conclusione: del costrutto formato da due entità e un'associazione vengono tradotte le due entità di cui una, quella con cardinalità (1,1) assorbe gli eventuali attributi della associazione e la chiave dell'altra entità viene catalogata tra gli attributi non chiave della nuova relazione.

Lo schema logico che si ottiene è:

Dipartimento(Dip., area, sede)  
Impiegato (Imp., qualifica, retribuzione, Dip., data\_inizio)

Se entrambi le entità sono obbligatorie lo schema concettuale ristrutturato sarà:



In questo tipo di costrutto, l'associazione può essere raccolta in una qualsiasi delle relazioni generate per raccogliere i dati delle entità.

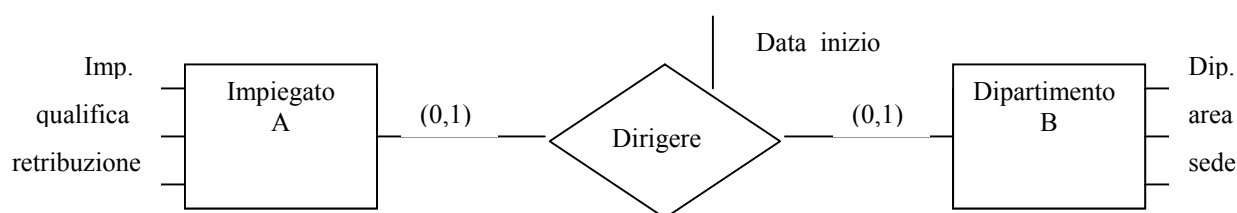
Data la obbligatorietà di entrambe le entità, esistono rispetto all'associazione due corrispondenze biunivoche, si può scegliere in quale delle due entità sia opportuno inserire i dati della associazione.

Vengono quindi tradotte le due entità di cui una a scelta assorbe gli eventuali attributi dell'associazione e in essa si introduce anche la chiave dell'altra entità tra gli attributi non chiave della nuova relazione.

Lo schema logico che si ottiene è:

Dipartimento(Dip. , area, sede)  
Impiegato (Imp. , qualifica, retribuzione, Dip., data\_inizio)

Se entrambi le entità sono opzionali lo schema concettuale ristrutturato sarà:



In questo tipo di costrutto, l'associazione può essere raccolta in una qualsiasi delle relazioni generate per raccogliere i dati delle entità.

Data la opzionalità di entrambe le entità, si può scegliere in quale delle due entità sia opportuno inserire i dati dell'associazione.

Vengono quindi tradotte le due entità di cui una a scelta assorbe gli eventuali attributi della associazione e in essa si introduce anche la chiave dell'altra entità tra gli attributi non chiave della nuova relazione.

In questo caso c'è una ulteriore soluzione, si può costituire la relazione che raccoglie i dati dell'associazione, unificando in una unica chiave gli identificatori primari delle entità che partecipano all'associazione e definendo una colonna per ogni attributo proprio dell'associazione, tale entità viene detta associativa.

I possibili schemi relazionali sono:

Soluzione A)

Dipartimento(Dip. , area, sede)  
Impiegato (Imp. , qualifica, retribuzione, Dip., data\_inizio)

Soluzione B)

Dipartimento(Dip. , area, sede, Imp., data\_inizio,)  
Impiegato(Imp. qualifica, retribuzione)

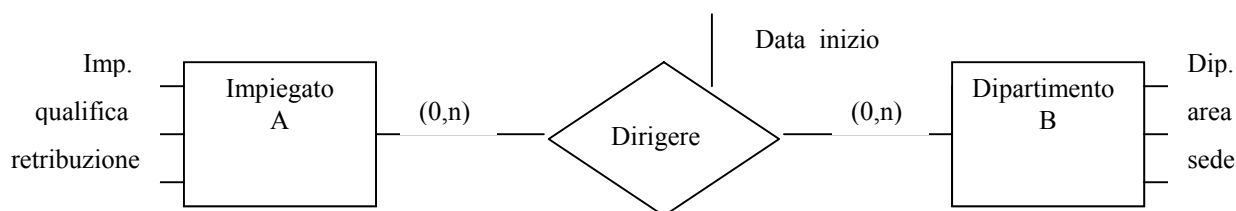
Soluzione C)

Dipartimento (Dip. , area, sede)  
Impiegato (Imp. , qualifica, retribuzione)  
Direzione (Imp., Dip., data\_inizio,)

In quest'ultima soluzione i due attributi Imp, Dip sono chiavi candidate di cui va scelta una delle due. Si determinano tabelle più piccole ottimizzando la fase di lettura e non si verificano valori nulli. Ciò implica però una tabella in più generando una maggiore complessità in fase di manutenzione.

### Associazione binaria molti a molti

Entrambe le entità partecipano all'associazione con cardinalità massima (n), lo schema concettuale ristrutturato sarà:



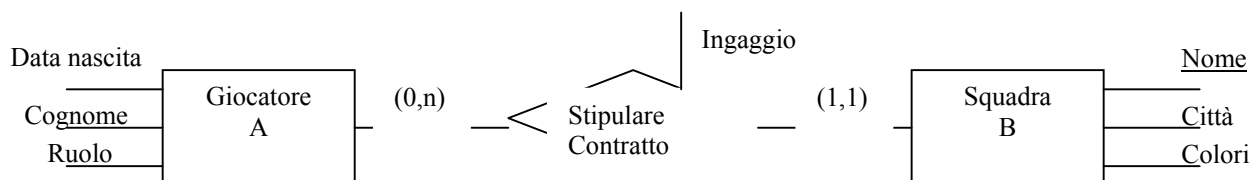
In questo tipo di costrutto non sono possibili ottimizzazioni e si applicano le regole generali precedentemente fornite. L'associazione deve essere tradotta in una relazione con una colonna per l'attributo e la chiave formata dalla unione delle chiavi delle due entità associate, tale entità viene detta associativa. Devono essere poi tradotte le entità A e B in altrettante relazioni.

Lo schema logico che si ottiene è:

Dipartimento (Dip, area, sede)  
Impiegato (Imp., qualifica, retribuzione)  
Direzione (Imp., Dip., data\_inizio,.)

### Associazione binaria uno a molti

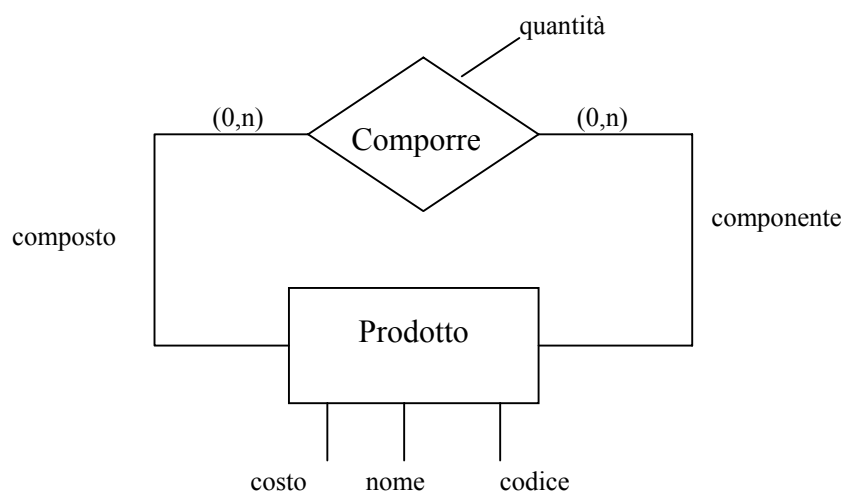
Vale il discorso già fatto per le cardinalità (1,1) e (0,n).



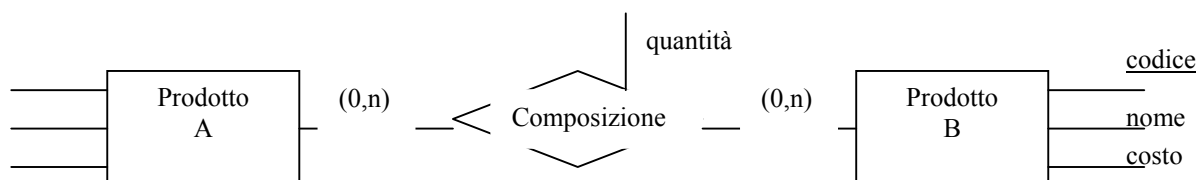
Data la cardinalità di B (obbligatoria), esiste una corrispondenza biunivoca tra le occorrenze di B e le occorrenze dell'associazione, si ritiene ottimale unire le due relazioni dove però la chiave di A non potrà far parte della chiave della nuova entità perché opzionale, il che potrebbe dare vita alla presenza di valori nulli nella chiave della nuova relazione. L'entità A verrà tradotta nei canoni della definizione data.

Conclusione: del costrutto formato da due entità e un'associazione vengono tradotte le due entità di cui una, quella con cardinalità (1,1) assorbe gli eventuali attributi dell'associazione e la chiave dell'altra entità viene catalogata tra gli attributi non chiave della nuova relazione.

### Associazione ricorsiva



Lo schema può essere rappresentato anche nel seguente modo:



In questo tipo di costrutto non sono possibili ottimizzazioni e si applicano le regole generali precedentemente fornite. L'associazione deve essere tradotta in una relazione con una colonna per l'attributo e la chiave formata dall'unione delle chiavi delle due entità associate. Devono essere poi tradotte le entità A e B in altrettante relazioni ma dato che A e B coincidono abbiamo un'unica relazione.

Lo schema logico che si ottiene è:

Prodotto (Codice , nome, costo)  
Composizione (ProdottoA.Codice, ProdottoB.Codice, quantità)

In questo caso, per dare nomi distinti agli attributi chiave dell'entità "prodotto" che formano la chiave della relazione "Composizione", si può fare ricorso all'uso di alias che dovendo essere usati due volte nella stessa entità potrebbero dare vita a confusione da cui l'uso di ProdottoA.Codice ProdottoB.Codice. Altre soluzioni implicano la nomenclatura del verso della associazione un concetto che però noi non abbiamo introdotto.

Normalmente lo schema logico ha anche una rappresentazione grafica dove sono comunque evidenziati i legami tra le relazioni. Nei tool di modellazione dati indicati in precedenza, a livello logico, la relazione è chiamata tabella e i legami tra le tabelle sono dette relazioni.

Parte integrante dello schema logico è il **glossario**, inteso come raccolta delle definizioni delle relazioni e delle colonne, al suo interno vengono riportati i collegamenti al concettuale.

Nel glossario alla relazione si deve associare:

1. Nome della relazione.
2. Descrizione della relazione. Deve contenere una descrizione minuziosa di ciò che rappresenta: definizione, contesto d'uso, utilizzo, eventuali rimandi ad oggetti appartenenti ad altre aree di business, descrizione di eventuali vincoli d'integrità.
3. Nome dell'entità o dell'associazione corrispondente nel concettuale.

Per ogni relazione elencare gli attributi e per ognuno di essi fornire:

1. Nome dell'attributo.
2. Descrizione completa dell'attributo, eventuali vincoli e fonti di alimentazione. Se importante corredare la documentazione relativa alla descrizione degli attributi con la storia dei cambiamenti. La storia di alcuni cambiamenti/raffinamenti va annotata con data e nome dell'autore e va esposta in ordine cronologico inverso. Se importante aggiungere i riferimenti alle maschere di inserimento e il sorgente dei dati.
3. Dominio.
4. Formato.
5. Nome dell'attributo corrispondente nel concettuale.
6. Indicare se è chiave della relazione.

### Tecniche di normalizzazione

Dopo la fase di traduzione, descritta nelle pagine precedenti, individuando e separando in entità tutti i concetti utili e appartenenti alla realtà di riferimento, si può affermare che uno schema logico è di buona qualità se non si riscontrano in esso né ridondanze di informazioni né anomalie che possano provocare inconsistenze della base dati che si sta progettando.

Se l'analisi non è stata condotta correttamente può accadere che più concetti distinti sono latenti in una entità progettata per altri scopi, possono provocare **ridondanza** dei valori assunti dai loro attributi nelle ennuple che formano la istanza di una relazione, in questo caso una operazione di aggiornamento del valore di uno di questi attributi dovrà essere fatta ovunque tale valore si ripeta, provocando un'anomalia detta di **aggiornamento**. Questa anomalia mette a rischio la consistenza della base informativa quando, per qualunque motivo, le operazioni di aggiornamento non fossero andate a buon fine.

Se nelle fasi di analisi un concetto non è stato individuato e separato, reso indipendente, esso è presente con i suoi attributi in una entità definita opportunamente per rappresentare un altro concetto che in essa sarà principale, con degli attributi chiave che lo individuano univocamente e ogni ennupla viene univocamente individuata dagli attributi chiave di quest'ultimo, il concetto inespresso è dipendente dal concetto principale.



Queste anomalie consistono nell'impossibilità di trattare direttamente dati relativi a concetti che non siano stati esplicitati, resi indipendenti, e siano rimasti legati al trattamento di altri concetti considerati principali (espressi concettualmente da una entità e logicamente da una relazione).

In questi casi si potranno verificare due tipi di anomalie:

**di cancellazione** nel caso in cui tutte le ennuple relative ai valori di un certo attributo (o insieme di attributi) chiave venissero cancellate si provocherebbe anche la cancellazione di tutti i valori degli attributi relativi al concetto dipendente, provocando una perdita di informazioni;

**di inserimento** in quanto non sarebbe possibile registrare sulla base dati una informazione relativa al concetto dipendente fino a quando non si abbia la possibilità di inserire un ennupla relativa al concetto principale che in se contenga dati correlati al concetto dipendente.

Ad evitare queste problematiche sono state definite delle regole, dette di normalizzazione, utili ad individuare e risolvere i problemi esposti. Di queste regole daremo una semplice elencazione rimandandone l'approfondimento alla lettura dei testi (1) (3).

Prima di proseguire nella elencazione detta definiamo una dipendenza funzionale come una funzione univoca tra gli elementi di due insiemi, nel nostro caso data una relazione  $R(A)$ , con  $A$  insieme di attributi di  $R$ , diremo  $X$  e  $Y$  due sotto insiemi, in generale disgiunti, di  $A$  tali che ad ogni ennupla di  $X$  sia univocamente associata una ennupla di  $Y$ , si dice allora che  $X$  determina  $Y$  o che  $Y$  dipende funzionalmente da  $X$  e si scrive  $X \rightarrow Y$ .

Una dipendenza multi funzionale è l'associazione non univoca tra gli elementi di due insiemi.

La prima forma normale, considerata fondamentale nelle verifiche di qualità di schemi logici, ha l'obiettivo di ridurre la struttura del singolo attributo ad una funzione univoca tra il dominio (insieme dei valori assumibili) e il valore assunto nella ennupla ed inoltre elimina eventuali attributi strutturati.

**Prima forma normale:** uno schema di relazione  $R$  si dice in prima forma normale (1NF) se ogni attributo che la costituisce è semplice (non è né strutturato né multi valore);

**Seconda forma normale :** uno schema di relazione  $R$  si dice in seconda (2NF) se è in 1NF e tutti i suoi attributi che non appartengono a nessuna chiave della relazione (non primi) dipendono funzionalmente e completamente dagli attributi di ogni chiave in  $R$

**Terza forma normale :** uno schema di relazione è in terza forma normale 3NF se è in 2NF e ogni suo attributo non primo è dipendente in modalità non transitiva da ogni chiave di  $R$ , cioè non ci sono campi non chiave che dipendano da campi non chiave.

**Boyce-Codd :** uno schema di relazione  $R$  è in forma normale di Boyce-Codd (BCNF) se e solo se ogniqualevolta esista una dipendenza funzionale non banale  $A \rightarrow B$  in  $R$  allora  $A$  è una super chiave per  $R$ .

**Quarta forma normale** uno schema di relazione è in quarta forma normale (4NF) se è in 3NF e non ci sono attributi a valori multipli indipendenti.

La mancata verifica di una qualunque delle forme normali enunciate potrà essere risolta applicando un processo di scomposizione della relazione di origine  $R$  in sotto relazioni tali che:

- la unione degli insiemi di attributi appartenenti ad ognuna delle sotto relazioni individuate dovrà essere equivalente all'insieme degli attributi presenti nella relazione di origine; cioè dovranno essere mantenuti sempre tutti gli attributi della relazione di origine;

- dovrà essere sempre possibile ricomporre precisamente la relazione di origine attraverso un join naturale tra le relazioni ottenute dalla scomposizione, in modo da ricostruire le correlazioni tra i concetti individuati senza aggiungere informazioni spurie che implicherebbero una perdita di consistenza della base dati ( si riesce a rispettare questa caratteristica “**decomposizione senza perdita**” se gli attributi comuni tra le relazioni ottenute dalla scomposizione sono chiave in almeno una tra le nuove relazioni);
- dovranno rimanere validi i vincoli definiti sulla relazione di origine (si riesce a rispettare questa caratteristica “**conservazione delle dipendenze**” se tutte le dipendenze funzionali o vincoli d'integrità individuati nella relazione di origine vengono rappresentati nelle relazioni ottenute dalla scomposizione, cioè gli attributi relativi al vincolo siano contemporaneamente presenti in almeno una relazione ottenuta dalla scomposizione).

Il processo di normalizzazione deve essere applicato con cautela, in quanto, tale processo genera schemi non soggetti alle anomalie viste ma che possono rendere più inefficiente l'esecuzione delle interrogazioni, per questo motivo, pur ritenendo auspicabile portare almeno alla terza forma normale uno schema dati, riteniamo tuttavia che non sia possibile stabilire un livello di normalizzazione standard, lasciando al capo progetto la libertà di tali scelte.

Quando nella prima parte del documento abbiamo parlato della molteplicità e poi di vincoli d'integrità abbiamo definito in quali modalità una entità entra a far parte di un'associazione.

Possiamo ora riconoscere in queste definizioni il concetto di dipendenza funzionale e comprendere meglio come le fasi di analisi concettuale si possono ritrovare nelle regole di normalizzazione e come queste si leghino alle fasi di traduzione al logico convalidandone le regole di scomposizione senza perdita che ne garantisce l'uso delle foreign key.

Le regole di normalizzazione si applicano sugli insiemi di attributi che formano le relazioni senza tenere conto delle loro caratteristiche fisiche, questo implica che si possa fare una serie di ragionamenti egualmente validi sia in fase di realizzazione del concettuale sia in fase di realizzazione dello schema logico.

Esistono varie modalità di procedere alla progettazione usando le regole di normalizzazione, noi riteniamo più completo il processo indicato, operando con le tecniche di normalizzazione solo in fase di verifica in entrambi le fasi di progettazione concettuale e di progettazione logica.

In conclusione ogni relazione dello schema logico diventerà una tabella nello schema fisico ottenendo quella semplificazione nella progettazione motivo della definizione di questo processo di lavorazione. Si genera quindi una corrispondenza biunivoca (almeno iniziale) tra le relazioni e le tabelle che è alla base di una confusione nell'uso dei termini stessi nell'ambito logico e fisico.

Ritenendo comunque minimale la differenza tra i due concetti agli scopi lavorativi, li considereremo l'uno sinonimo dell'altro. Si potrà verificare come in letteratura è più usato il termine relazione per il logico e tabella per il fisico viceversa nel contesto operativo/pratico viene usato il termine tabella che assume entrambi i significati.

*Bibliografia*

- 1) Atzeni, Batini, De Antonellis “La teoria relazionale dei dati” Boringhieri
- 2) Atzeni, Ceri, Paraboschi, Torlone “Basi dati” McGraw-Hill
- 3) Jeffrey D.Ullman “Basi di dati e basi di conoscenza” Jackson Libri
- 4) C.Batini, G. De Petra, M. Lenzerini, G. Santucci “ La progettazione concettuale dei dati” Franco angeli