



BASI DI DATI 2

DATA WAREHOUSE

Supporto alle decisioni aziendali

- La maggior parte delle aziende dispone di enormi basi di dati contenenti dati di tipo operativo:
 - Queste basi di dati costituiscono una potenziale miniera di informazioni utili.
- I sistemi per il supporto alle decisioni permettono di:
 - Analizzare lo stato dell'azienda.
 - Prendere decisioni *rapide e migliori*.

Supporto alle decisioni aziendali (2)

- ✓ Analisi e previsione dell'evoluzione della domanda.
- ✓ Individuazione di aree critiche.
- ✓ Chiarezza dei conti e trasparenza finanziaria:
 - ▣ reporting, pratiche antifrode e antiriciclaggio.
- ✓ Definizione e realizzazione di strategie vincenti:
 - ▣ Contenimento di costi e aumento di profitti.

Business Intelligence

- *Disciplina di supporto alla decisione strategica aziendale.*
- **Obiettivo:** trasformazione dei dati aziendali in informazioni fruibili:
 - a diversi livelli di dettaglio;
 - per applicazioni di analisi.
- Tipologia di utenza eterogenea.
- Necessaria un'adeguata infrastruttura hardware e software di supporto.

Ambiti applicativi

- *Industrie manifatturiere*: gestione ordini e spedizioni, supporto clienti.
- *Distribuzione*: profilo utenti, gestione magazzino.
- *Servizi finanziari*: analisi acquisti (carta di credito).
- *Assicurazioni*: analisi richieste indennizzo, riconoscimento frodi.
- *Telecomunicazioni*: analisi delle chiamate, riconoscimento frodi.
- *Servizi pubblici*: analisi dell'utilizzo.
- *Sanità*: analisi dei risultati.

Knowledge Discovery

- **Dati:** insieme di informazioni contenute in una base di dati o data warehouse.

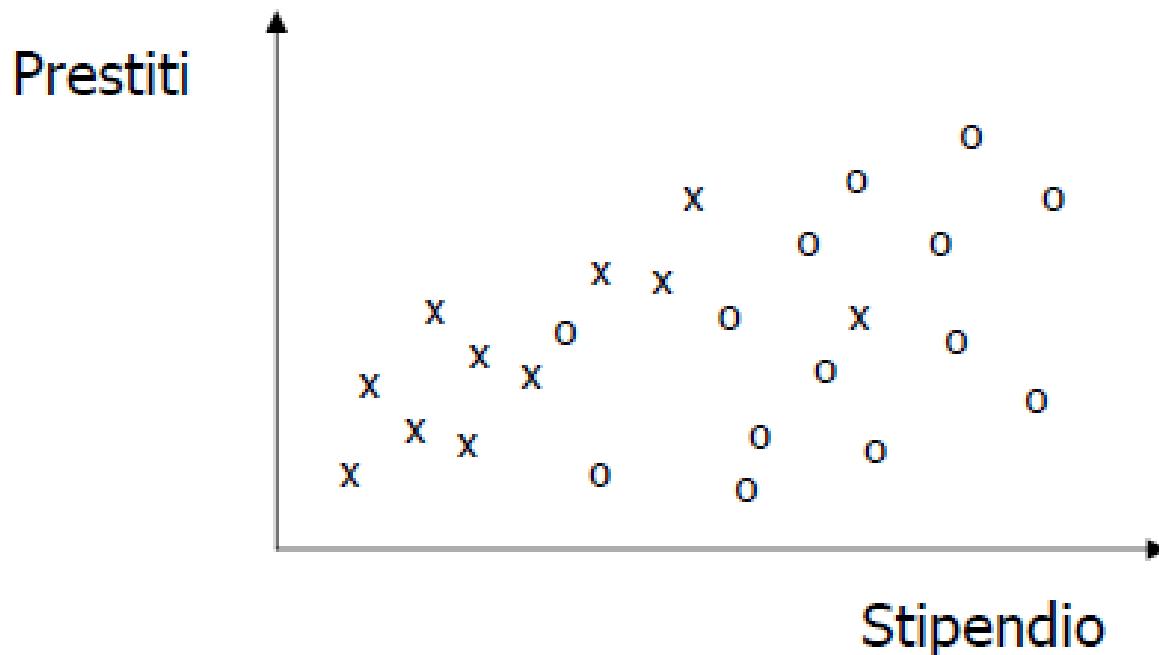
- **Pattern:** espressione in un linguaggio opportuno che descrive in modo succinto le informazioni estratte dai dati:
 - regolarità;
 - informazione di alto livello.

Knowledge Discovery (2)

- Processo di estrazione dai dati di pattern:
 - validi
 - precedentemente ignoti (novità)
 - potenzialmente utili (utilità)
 - comprensibili

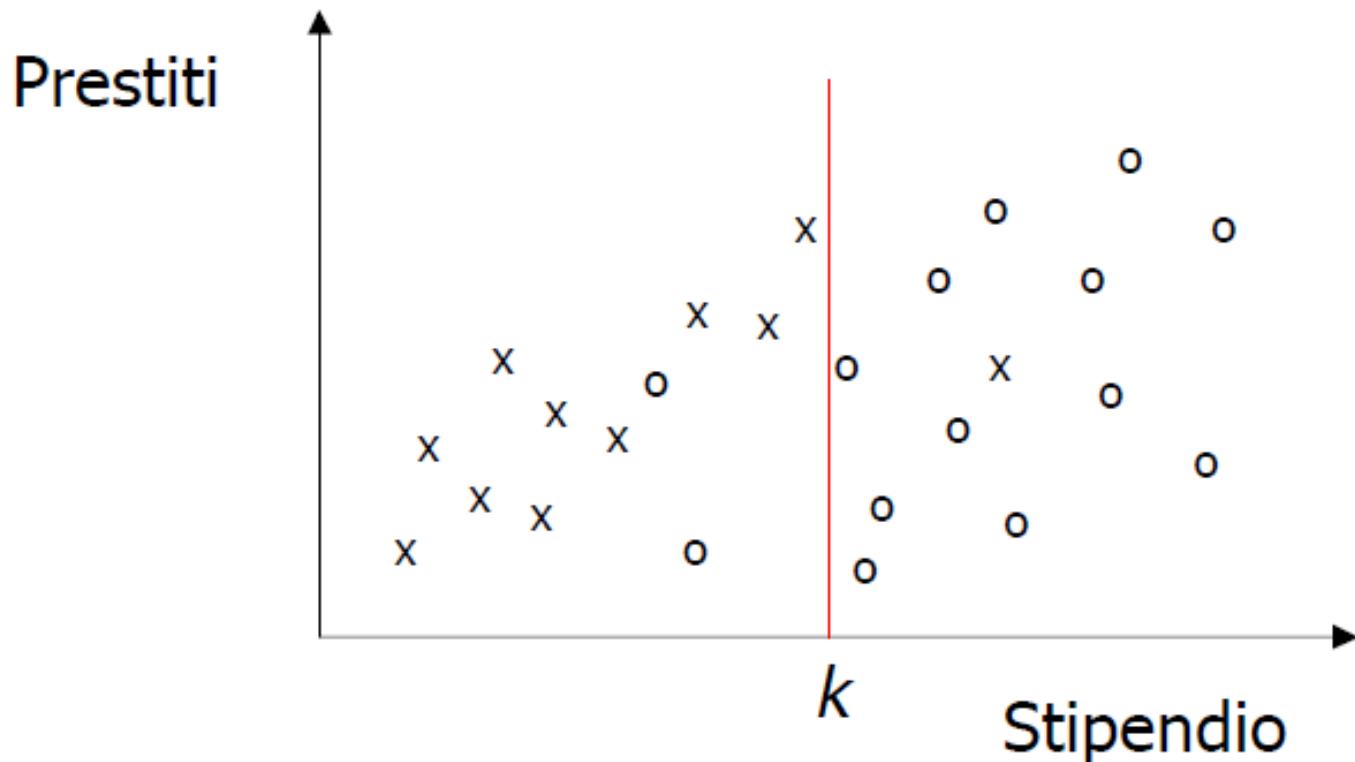
Esempio

- Clienti di una banca che hanno contratto un prestito:
 - x: clienti che hanno mancato la restituzione di rate.
 - o: clienti che hanno rispettato le scadenze.



Esempio (2)

- If $\text{stipendio} < k\text{\euro}$ then mancati pagamenti.

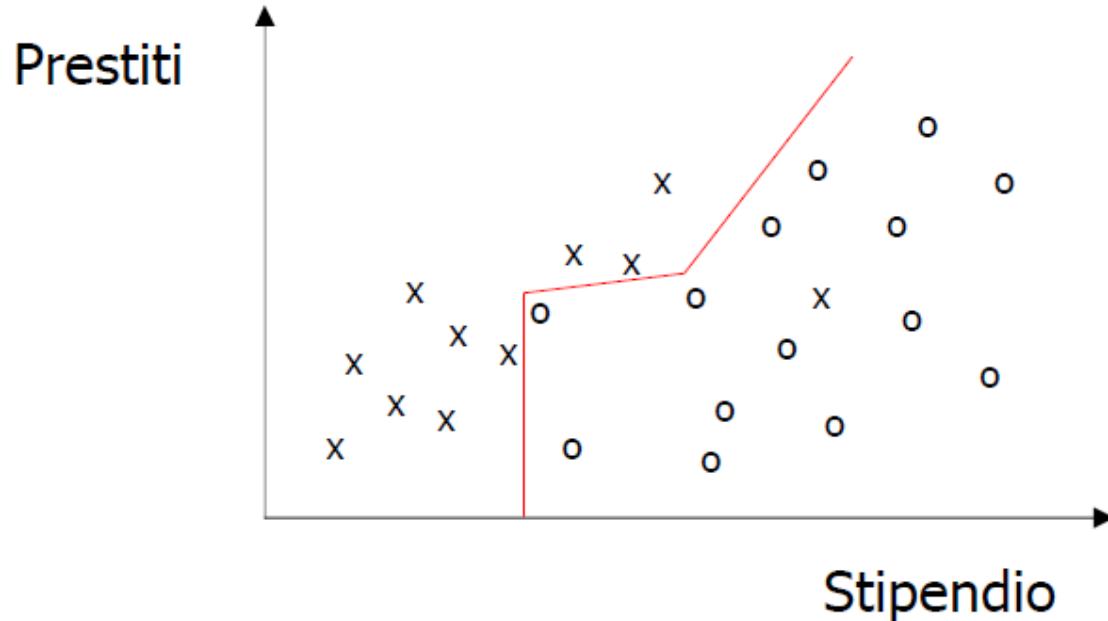


Caratteristiche dei pattern

- **Validità:** i pattern scoperti devono essere validi su nuovi dati con un certo grado di certezza.
 - **Es:** Lo spostamento a destra del valore di **k** porta riduzione del grado di certezza.
- **Novità:** misurata rispetto a variazioni dei dati o della conoscenza estratta.
- **Utilità:**
 - **Es:** un aumento di profitto atteso dalla banca associato alla regola estratta.
- **Comprendibilità:** misure di tipo:
 - sintattico (numero di bit del pattern);
 - semantico.

Esempio (3)

- Usando le tecniche di *data mining* (i cui requisiti sono la qualità delle informazioni estratte ed i criteri diversificati di estrazione):
 - Clustering
 - Alberi di decisione
 - ...



Elaborazione dei dati

- La modalità tradizionale di uso dei DBMS è caratterizzata da:
 - istantanea del valore corrente dei dati;
 - dati dettagliati, rappresentazione relazionale;
 - operazioni strutturate e ripetitive;
 - accesso in lettura o aggiornamento di pochi record;
 - transazioni brevi;
 - isolamento, affidabilità e integrità sono critici;
 - dimensione della base di dati » 100Mb-Gb.

Analisi dei dati

- L'elaborazione dei dati per il supporto alle decisioni è caratterizzata da:
 - dati di tipo “storico”;
 - dati consolidati e integrati;
 - applicazioni ad hoc;
 - accesso in lettura a milioni di record;
 - interrogazioni di tipo complesso;
 - consistenza dei dati prima e dopo le operazioni di caricamento periodico;
 - dimensione della base di dati » 100Gb-Tb;

Data Warehousing

- Il Data Warehousing:
 - ▣ Una collezione di metodi, tecnologie e strumenti di ausilio al “*lavoratore della conoscenza*”
(knowledge worker: *dirigente, amministratore, gestore, analista, ...*)
per condurre analisi dei dati finalizzate all’attuazione di processi decisionali e al miglioramento del patrimonio informativo.

Perché il Data Warehousing?

- Alcune esigenze che hanno decretato la nascita del data warehousing:
 - ❖ “Abbiamo montagne di dati ma non possiamo accedervi!”.
 - ❖ “Come è possibile che persone che svolgono lo stesso ruolo presentino risultati sostanzialmente diversi?”.
 - ❖ “Vogliamo selezionare, raggruppare e manipolare i dati in ogni modo possibile!”.
 - ❖ “Mostratemi solo ciò che è importante!”.
 - ❖ “Tutti sanno che alcuni dati non sono corretti!”

Data Warehouse

- Un **Data Warehouse (DW)** è una base di dati per il supporto alle decisioni, che è mantenuta *separatamente dalle basi di dati operative* dell'azienda.
- I dati devono essere:
 - orientati ai soggetti di interesse;
 - integrati e consistenti;
 - dipendenti dal tempo, non volatili;
 - utilizzati per il supporto alle decisioni aziendali.

Data Warehouse (2)

- In particolare, un DW è una collezione di dati di supporto al processo decisionale che presenta le seguenti caratteristiche:
 - Orientata ai soggetti di interesse:
 - Si incentra sui concetti di interesse dell'azienda (clienti, prodotti, vendite, ...).
 - Integrata e consistente:
 - Il DW si appoggia su più fonti eterogenee di dati.
 - Rappresentativa dell'evoluzione temporale e non volatile:
 - Aggiornato ad intervalli regolari.
 - Permette analisi che spaziano sulla prospettiva di alcuni anni.

Data Warehouse (3)

- Se si tengono in considerazione alcune problematiche nel campo dei sistemi informativi:
 - Esigenza di accedere in maniera efficiente a grandi moli di dati.
 - Esigenza di utilizzare l'informazione per scopi strategici e decisionali da parte delle aziende.
 - Esigenza di separare l'elaborazione di tipo analitico (*OLAP: On-Line Analytical Processing*) da quella di tipo transazionale (*OLTP: On-Line Transactional Processing*).
- Come soluzione a tali problematiche nascono i *Sistemi di Supporto alle Decisioni*.
- Tra tali sistemi concentriamo l'attenzione sui *Sistemi di Data Warehousing*.

Differenze tra DB operazionali e DW

	DB operazionali	DW
Utenti	Migliaia	Centinaia
Carico di lavoro	Transazioni predefinite	Interrogazioni di analisi ad hoc
Accesso	Centinaia di record in lettura e scrittura	Milioni di record per lo più in lettura
Scopo	Dipende dall'applicazione	Supporto alle decisioni
Dati	Elementari	Di sintesi

Differenze tra DB operazionali e DW (2)

Integrazione dei dati	Per applicazione	Per soggetto
Qualità	In termini di integrità	In termini di consistenza
Copertura temporale	Solo dati correnti	Dati correnti e storici
Aggiornamenti	Continui	Periodici
Modello	Normalizzato	Denormalizzato, Multidimensionale
Ottimizzazione	Per accessi OLTP su una frazione del DB	Per accessi OLAP su gran parte del DB
Sviluppo	A cascata	Iterativo

Perché dati separati?

- **Prestazioni:**
 - Ricerche complesse riducono le prestazioni delle transazioni operative.
 - Metodi di accesso diversi a livello fisico.
- **Gestione dei dati:**
 - Informazioni mancanti (storico).
 - Consolidamento dei dati.
 - Qualità dei dati (problema di inconsistenze).

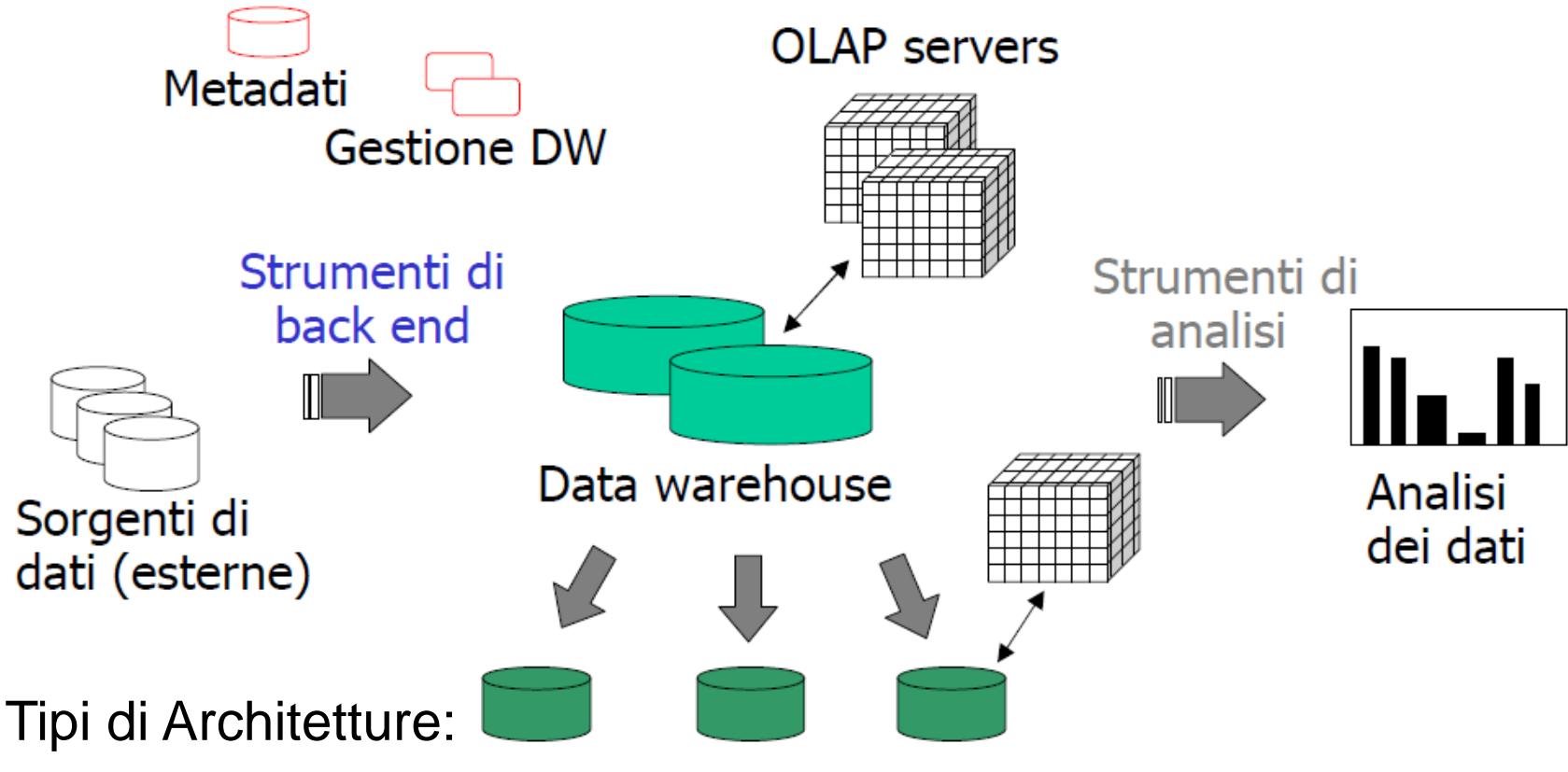
DW e Data mart

- **Data Warehouse aziendale:** contiene informazioni sul funzionamento di “tutta” l’azienda.
 - Processo di modellazione funzionale esteso.
 - Progettazione e realizzazione richiedono molto tempo.
- **Data mart:** sottoinsieme dipartimentale focalizzato su un settore prefissato.
 - Si hanno due possibilità:
 - alimentato dal DW primario (i.e., DW aziendale);
 - alimentato direttamente dalle sorgenti.
 - Realizzazione più rapida.
 - Richiede progettazione attenta, in modo da evitare problemi di integrazione in seguito.

Architetture dei DW

- **Caratteristiche architetturali:**
 - Separazione
 - Dell'elaborazione analitica da quella transazionale.
 - Scalabilità
 - Capacità di ridimensionamento a fronte della crescita del volume dei dati.
 - Estendibilità
 - Possibilità di integrare nuove applicazioni senza riprogettare il sistema.
 - Sicurezza
 - Controllo sugli accessi.
 - Amministrabilità
 - Semplicità nell'amministrazione dei dati.

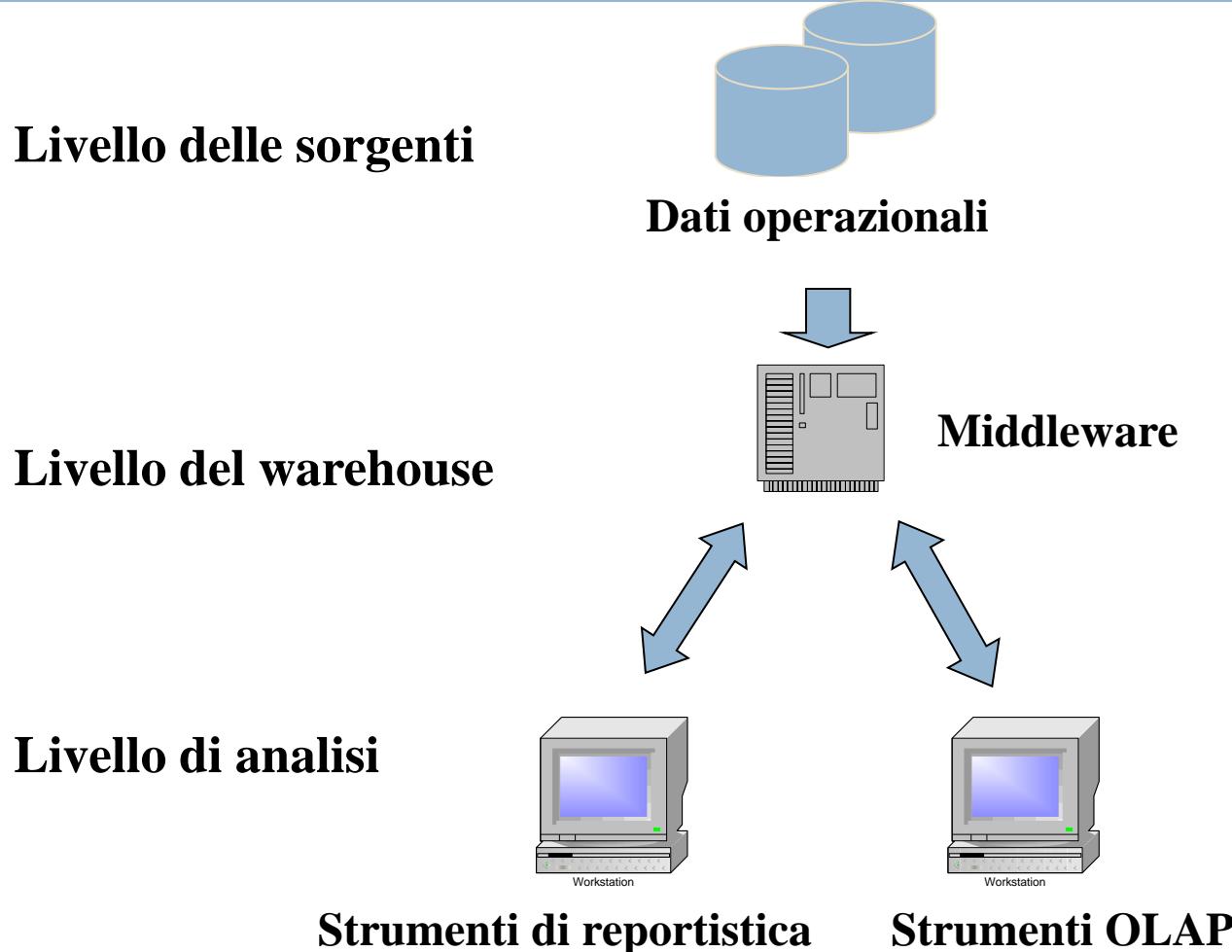
Elementi costitutivi di un DW



□ Tipi di Architetture:

1. Ad un livello
2. A due livelli
3. A tre livelli

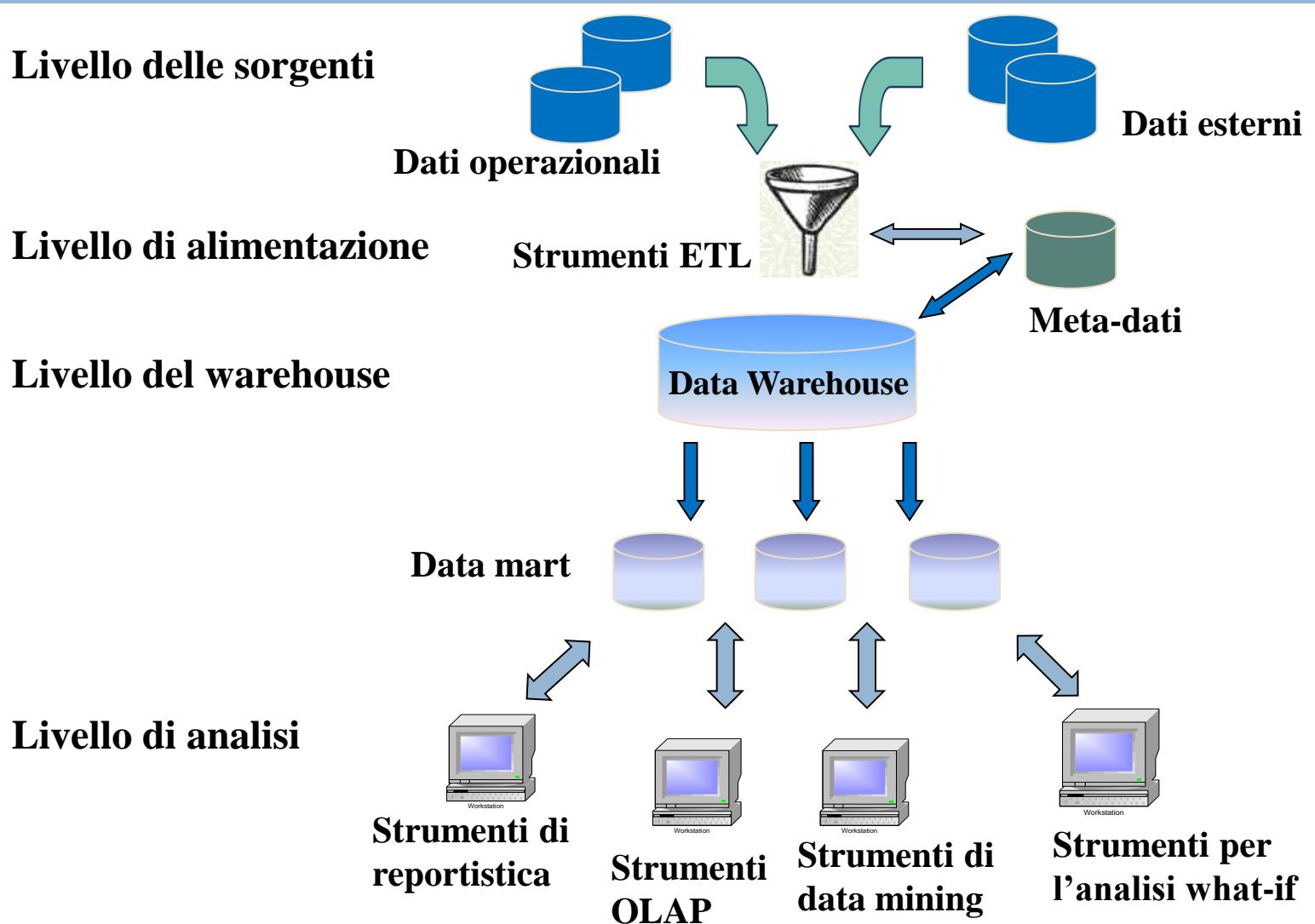
Architettura ad un livello



Architettura ad un livello (2)

- **Caratteristiche:**
 - DW virtuale:
 - Implementato come una vista multidimensionale.
 - Minimizzazione dei dati memorizzati.
- **Punti deboli:**
 - Non rispetta il requisito di separazione tra l'elaborazione analitica OLAP e quella transazionale OLTP.

Architettura a due livelli

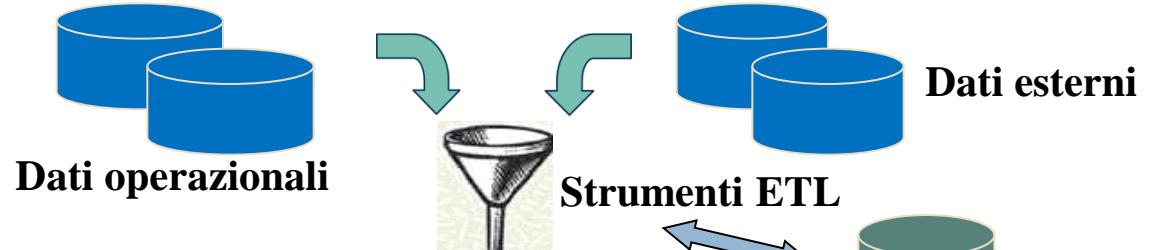


Architettura a due livelli (2)

- Livello delle sorgenti:
 - Fonti di dati eterogenei estratti dall'ambiente di produzione oppure provenienti da sistemi informativi esterni all'azienda.
- Livello dell'alimentazione:
 - I dati memorizzati nelle sorgenti vengono estratti e ripuliti tramite strumenti **ETL** (*Extraction, Transformation and Loading*).
- Livello del warehouse:
 - Le informazioni vengono raccolte in un DW centralizzato (primario).
 - Può essere consultato direttamente o utilizzato come sorgente per costruire i Data mart.
 - Il Data mart è un sottoinsieme o aggregazione dei dati presenti nel DW primario, contenente l'insieme delle informazioni rilevanti per una particolare area di business.
- Livello di analisi:
 - Permette la consultazione efficiente e flessibile dei dati integrati per fini di stesura di report, di analisi e di simulazione.

Architettura a tre livelli

Livello delle sorgenti



Livello di alimentazione

Caricamento

Dati riconciliati

Data Warehouse

Livello del data warehouse

Data mart



Livello di analisi

Strumenti di reportistica



Strumenti OLAP



Strumenti di data mining



Strumenti per l'analisi what-if



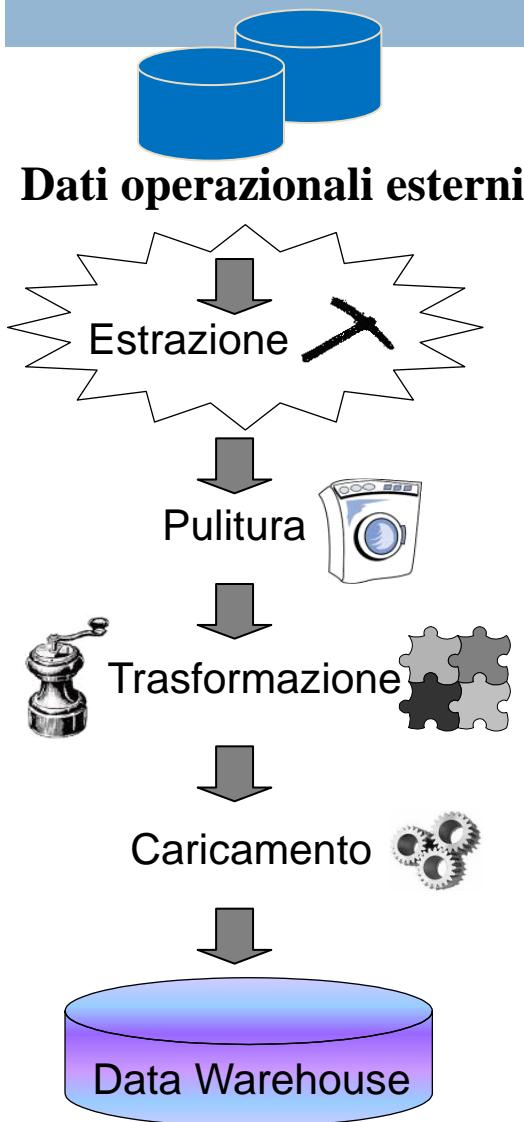
Architettura a tre livelli (2)

- Viene introdotto il livello dei dati riconciliati:
 - Materializza i dati operazionali ottenuti a valle del processo di integrazione e ripulitura dei dati sorgente.

Gli strumenti ETL

- Il ruolo degli **strumenti ETL** è quello di alimentare una sorgente di dati:
 - singola;
 - dettagliata;
 - esauriente;
 - di alta qualità;
- che possa a sua volta alimentare il DW.
- Le operazioni svolte dagli strumenti ETL vengono spesso definite con il termine di *riconciliazione*.

Gli strumenti ETL (2)



- Essi possono essere così classificati:
 1. Estrazione dei dati da sorgenti esterne.
 2. Pulizia dei dati (errori, dati mancanti o duplicati).
 3. Trasformazioni e conversioni di formato.
 4. Caricamento e refresh periodico.
- La riconciliazione avviene in due occasioni:
 - Quando il DW viene popolato la prima volta.
 - Periodicamente (aggiornamento del DW).

Riconciliazione (Estrazione)

- Consiste nell'estrazione di dati rilevanti dalle sorgenti.
- Tipi di estrazione:
 - **Estrazione statica:**
 - Effettuata quando il DW viene popolato per la prima volta.
 - Fotografia dei dati operazionali.
 - **Estrazione incrementale:**
 - Viene usata per l'aggiornamento periodico del DW.
 - Cattura solo i cambiamenti avvenuti nelle sorgenti dall'ultima estrazione.
 - Basata sul *log* del DBMS operazionale.
 - **Guidata dalle sorgenti:**
 - Consiste nel riscrivere le applicazioni operazionali per far sì che esse notifichino in modo asincrono le modifiche.
 - Oppure nell'implementare dei trigger nei DB operazionali, abbinati alle transazioni di modifica dei dati rilevanti.

Riconciliazione (Pulitura)

- Si occupa di migliorare la qualità dei dati, normalmente scarsa nelle sorgenti.
- Alcune tipologie di errori:
 - ▣ Dati duplicati:
 - Stesso paziente che compare più volte in un'anagrafica ospedaliera.
 - ▣ Dati mancanti:
 - Manca la professione di un cliente.
 - ▣ Inconsistenza tra valori logicamente associati:
 - Tra indirizzo, comune e il CAP.
 - ▣ Valori impossibili:
 - **Es:** 30/02/2016 o 29/02/2017.
 - ▣ Valori inconsistenti dovuti a diverse convenzioni
 - **V.** Risorgimento e **Via** Risorgimento.
 - ▣

Riconciliazione (Pulitura) (2)

- Funzionalità:
 - **Correzione ed omogeneizzazione:**
 - Uso di dizionari appositi per correggere gli errori di scrittura.
 - **Pulitura basata su regole:**
 - Applicazione di regole del dominio applicativo per stabilire le corrette corrispondenze tra valori.

Riconciliazione (Trasformazione)

- Converte i dati dal formato operazionale sorgente a quello del DW.
 - La corrispondenza con il livello sorgente è in genere complicata dalla presenza di fonti eterogenee.
- Funzionalità:
 - **Conversione e normalizzazione** (**da non confondere**):
 - Operano sia a livello di formato di memorizzazione sia a livello di unità di misura al fine di uniformare i dati.
 - **Matching:**
 - Stabilisce corrispondenze tra campi equivalenti in sorgenti diverse.
 - **Selezione:**
 - Riduce il numero di campi e di record rispetto alle sorgenti.

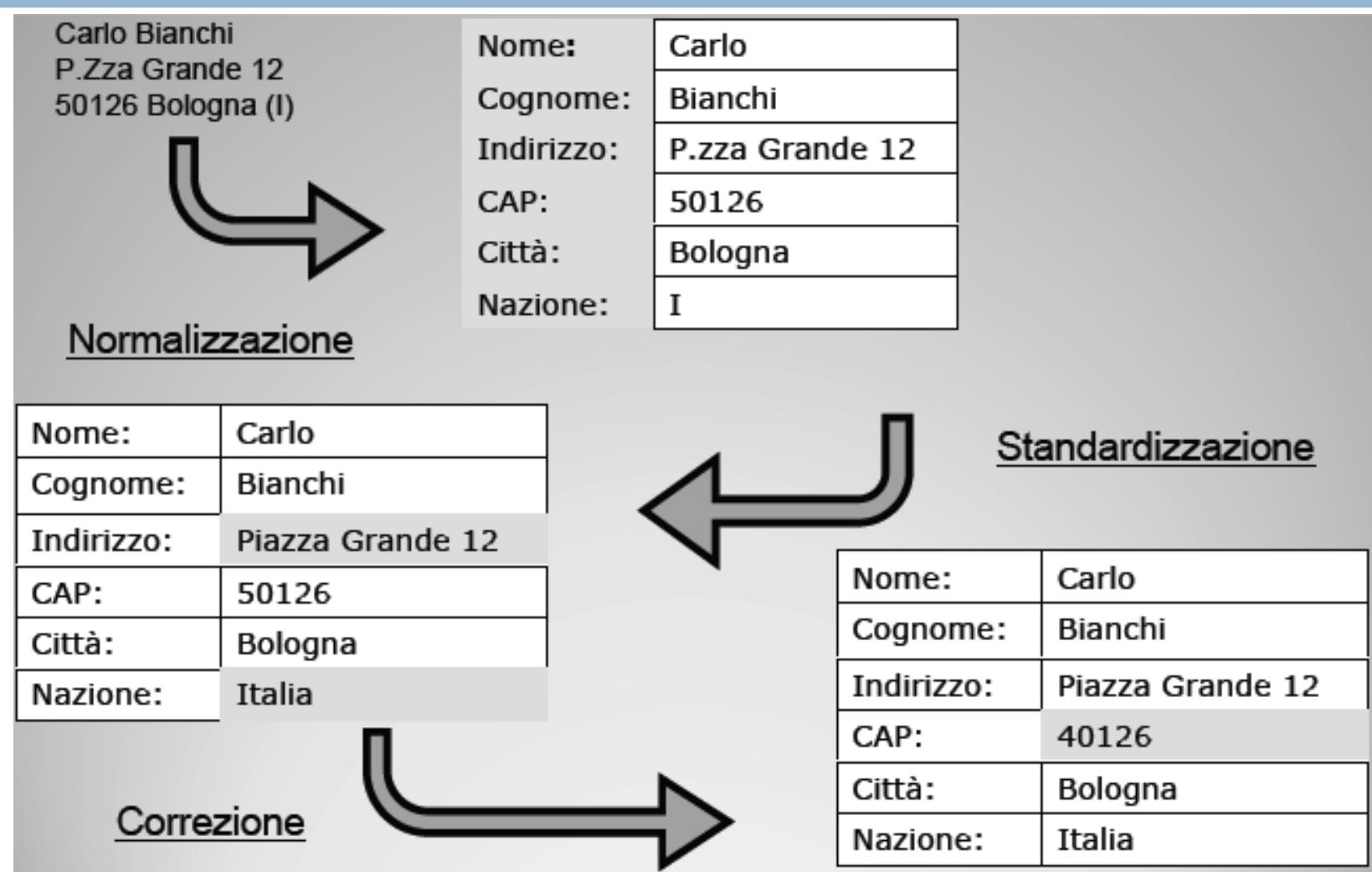
Riconciliazione (Trasformazione) (2)

- Per l'alimentazione del DW si hanno due differenze:
 - ▣ La normalizzazione (legata alle dipendenze funzionali) è sostituita dalla **denormalizzazione**:
 - I dati nel DW vengono denormalizzati con operazioni di JOIN.
 - ▣ Si introduce l'**aggregazione**:
 - Si realizza una sintesi dei dati nel DW.

Riconciliazione (Caricamento)

- Il caricamento dei dati avviene secondo due modalità:
 - **Refresh:**
 - I dati nel DW vengono riscritti integralmente.
 - Tecnica usata in abbinamento all'estrazione statica.
 - **Update:**
 - Nel DW vengono aggiunti solo i cambiamenti occorsi ai dati sorgente.
 - Tecnica usata in abbinamento all'estrazione incrementale.

Esempio di pulitura e trasformazione di un dato anagrafico





Struttura ed elaborazione dei dati

Il Modello Multidimensionale

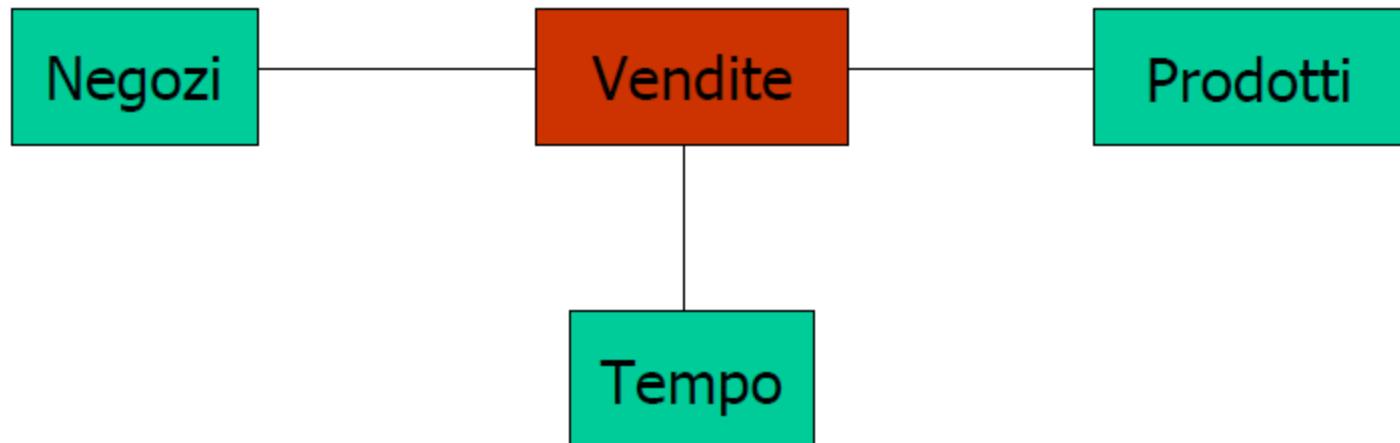
- Esigenza di rispondere in modo efficiente alle domande:
 - “*Quali incassi sono stati registrati l’anno passato per ciascuna regione e ciascuna categoria di prodotto?*”
 - “*Che correlazione esiste tra l’andamento dei titoli azionari dei produttori di PC e i profitti trimestrali degli ultimi 5 anni?*”
 - “*Quali sono gli ordini che massimizzano gli incassi?*”
 - ...
- È il fondamento per la rappresentazione e l’interrogazione dei dati nei DW.

Il Modello Multidimensionale (2)

- Gli oggetti che influenzano il processo decisionale sono **fatti** di un'organizzazione.
 - Es: vendite, spedizioni, ricoveri, interventi chirurgici.
- I **fatti** di interesse sono rappresentati in **cubi** in cui:
 - ogni cella contiene **misure** numeriche che quantificano il fatto da diversi punti di vista (**evento**);
 - ogni asse rappresenta una **dimensione** di interesse per l'analisi;
 - ogni dimensione può essere la radice di una **gerarchia** di attributi usati per aggregare i dati memorizzati nei cubi base.

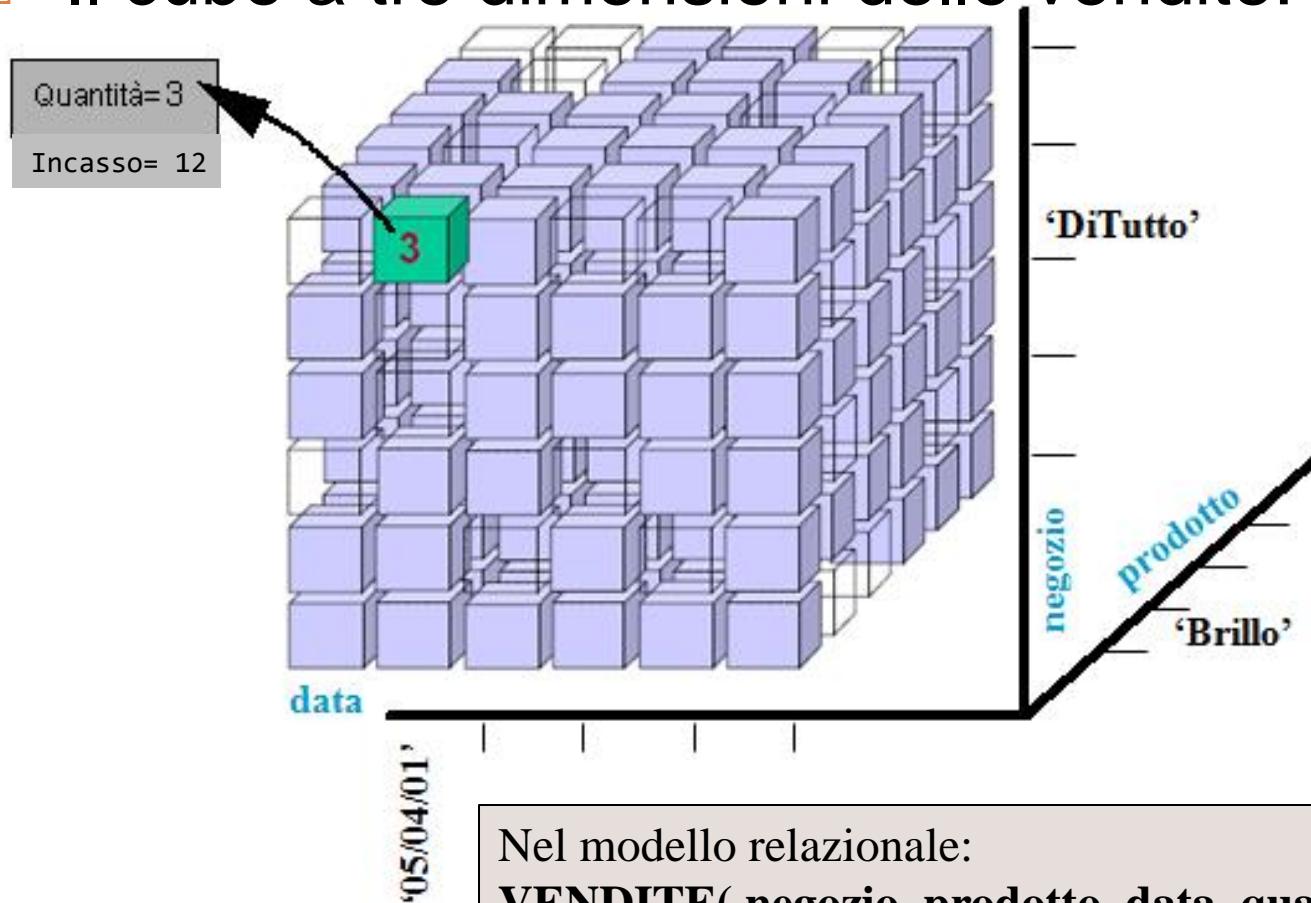
Esempio

- Data warehouse per l'analisi delle vendite di una catena di supermercati (*modello a stella*):



Il Modello Multidimensionale (3)

- Il cubo a tre dimensioni delle vendite:

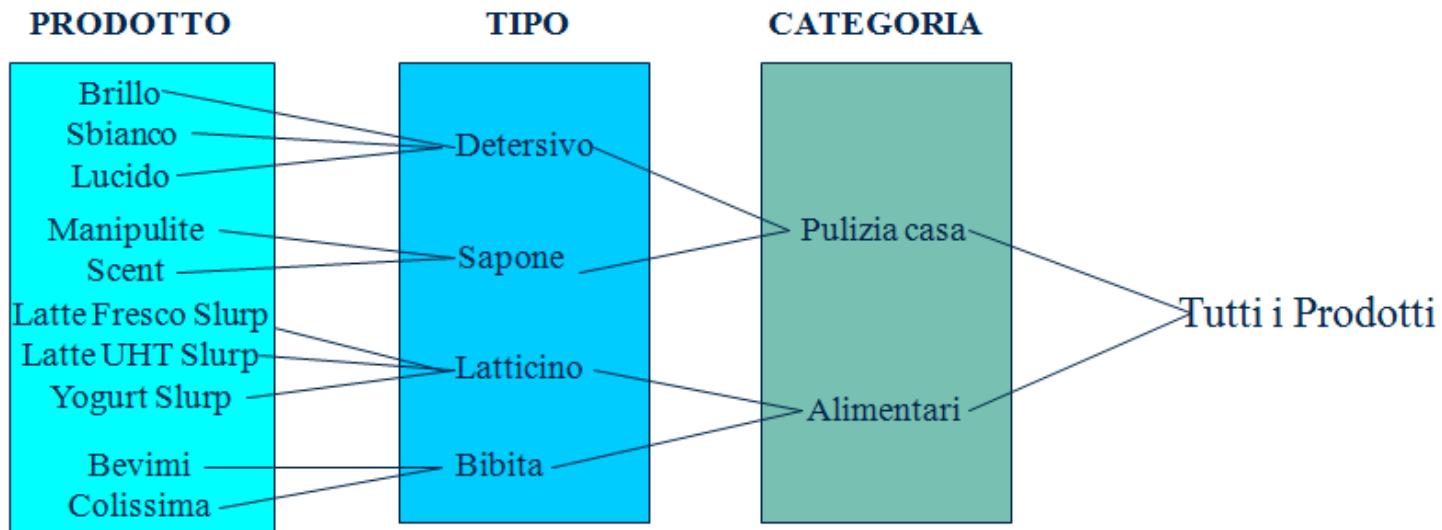


Size del DW

- Dimensione Tempo: 2 anni x 365 giorni.
- Dimensione Negozi: 300 negozi.
- Dimensione Prodotti: 30.000 prodotti, di cui 3.000 venduti ogni giorno in ogni negozio.
- Numero di celle nella tabella dei fatti:
 - $730 \times 300 \times 3000 = 657$ millioni
- Size delle tabelle dei fatti $\approx 20\text{GB}$

Le gerarchie

- A ciascuna dimensione del cubo è associata una gerarchia di livelli di aggregazione:

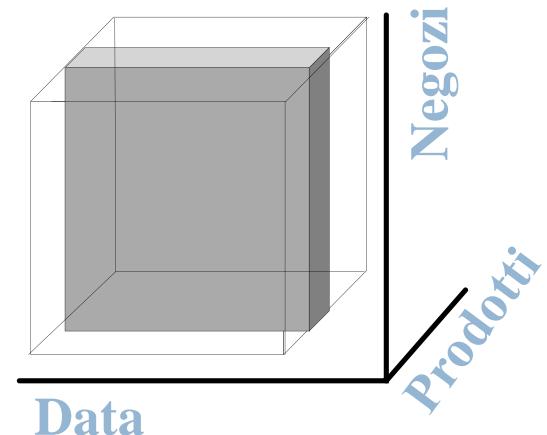


- Lo stesso vale per:
 - negozio → città → regione
 - data → mese → anno

Attributi dimensionali

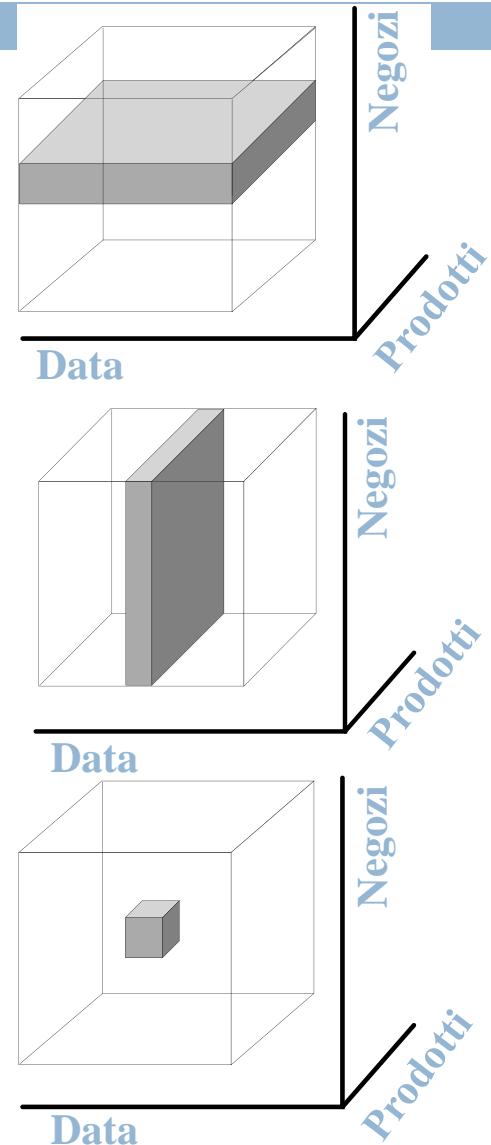
Restrizione

- Restringere i dati significa ritagliare una porzione dal cubo circoscrivendo il campo di analisi.
- La forma più semplice di restrizione è lo ***slicing***.
 - Si riduce la dimensionalità del cubo fissando un valore per uno o più dimensioni.
- **Esempio:**
 - ❖ Al manager di prodotto interessa la vendita di **un prodotto** in tutti i periodi e in tutti i negozi:



Restrizione (2)

- ❖ Al manager regionale interessa la vendita dei prodotti in tutti i periodi nei **propri negozi**:
- ❖ Al manager finanziario interessa la vendita dei prodotti in tutti i negozi relativamente ad **un determinato periodo**:
- ❖ Il manager strategico si concentra su una categoria di prodotti, un'area regionale ed un orizzonte temporale medio:



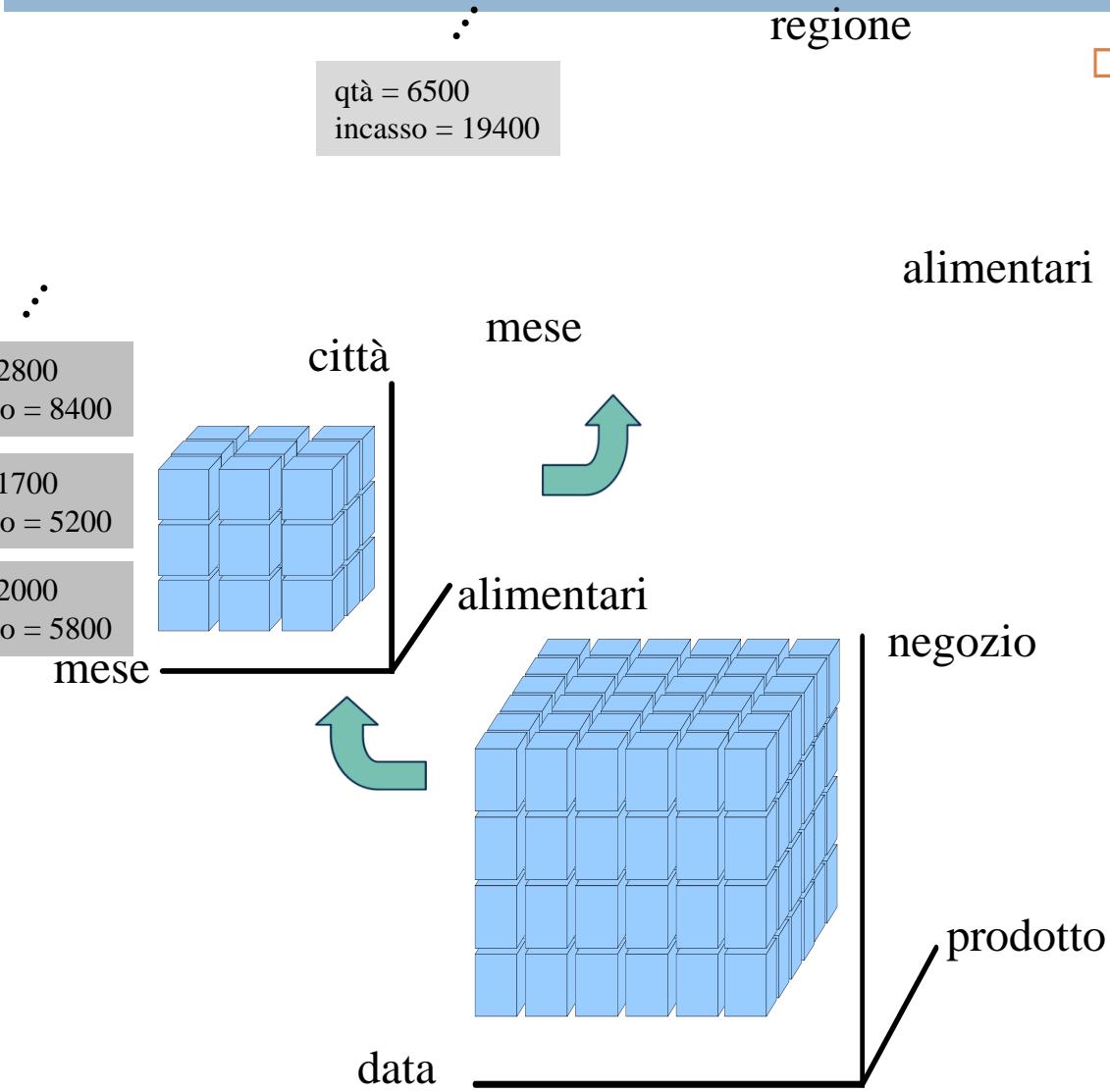
Eventi e aggregazione

- Un **evento primario** è una particolare occorrenza di un fatto, individuata da una ennupla caratterizzata da un valore per ciascuna dimensione.
 - Un esempio di evento primario potrebbe essere: *il ‘05/04/01’ nel negozio ‘DiTutto’ è stata venduta una quantità 3 con incasso 12 del prodotto ‘Brillo’.*
- Dato un insieme di attributi dimensionali (*pattern*), ciascuna ennupla di questi valori individua un **evento secondario** che aggrega tutti gli eventi primari corrispondenti.
 - A ciascun evento secondario è associato un valore per ciascuna misura, che riassume in sé tutti i valori della stessa misura negli eventi primari corrispondenti.

Eventi e aggregazione (2)

- Le gerarchie definiscono il modo in cui gli eventi primari possono essere aggregati e selezionati significativamente per il processo decisionale.
- La dimensione in cui una gerarchia ha radice ne definisce la *granularità* più fine di aggregazione.
 - Agli altri attributi dimensionali corrispondono granularità via via crescenti.

Aggregazione



- L'aggregazione richiede di definire un operatore adatto per comporre i valori delle misure che caratterizzano gli eventi primari in valori da abbinare a ciascun evento elementare.

Metadati

- I metadati sono dati usati per descrivere altri dati.
- Nel DW:
 - Indicano le sorgenti.
 - Descrivono la struttura dei dati nel DW.
 - Indicano il valore, l'uso e le funzioni dei dati memorizzati.
 - Descrivono come i dati sono alterati e trasformati.
 - Dati sulla struttura delle query ed esecuzione:
 - Codice SQL per le query.
 - Piano di esecuzione.

Metadati (2)

- Il contenitore dei metadati è strettamente collegato al DW:
 - Le applicazioni ne fanno un intenso uso sia sul lato dell'alimentazione che su quello dell'analisi.
- Kelly (1997) distingue i metadati in due categorie:
 - Metadati interni:
 - Descrivono sorgenti, trasformazioni e politiche di alimentazione, ...
 - Di interesse per l'amministratore.
 - Metadati esterni:
 - Descrivono le definizioni, quantità, unità di misura, aggregazioni significative, ...
 - Di interesse per gli utenti.

Accesso ai dati del DW

□ Reportistica:

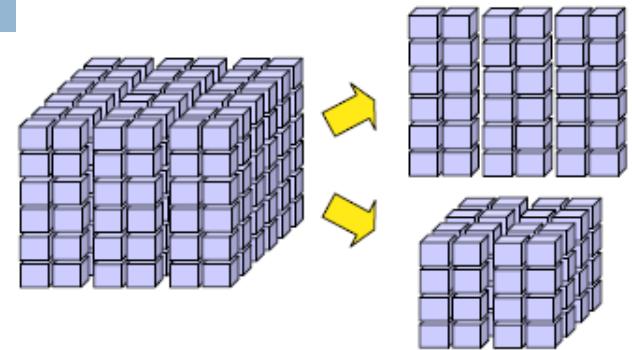
- Orientata ad utenti che devono accedere ad informazioni strutturate in modo pressoché invariabile nel tempo.
 - **Es:** azienda ospedaliera che deve periodicamente fornire agli uffici regionali rapporti mensili sui costi di ricovero sostenuti.
- Il progettista può formulare l'interrogazione e renderla disponibile nel tempo.
- Un **report** è definito da una interrogazione (**Es:** selezione ed aggregazione) e da una presentazione (forma tabellare o grafica).

□ OLAP:

- È la principale modalità di fruizione delle informazioni contenute in un DW.
- Consente agli utenti di esplorare interattivamente i dati sulla base del modello multidimensionale.
- L'utente è in grado di costruire interattivamente una sessione di analisi.

Operatori OLAP

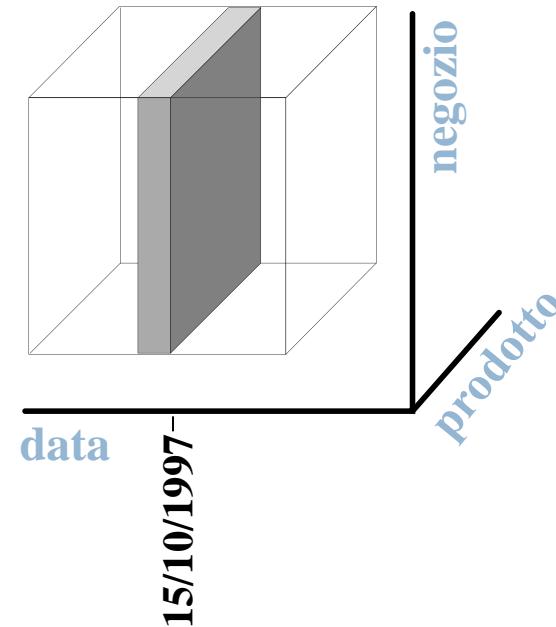
- Operatori di restrizione:
 - *Slice and Dice*
- Operatori di aggregazione:
 - *Roll-Up*
 - *Drill-Down*
 - *Drill-Across*
 - *Drill-Through*
- Operatore di pivoting.



Operatori di restrizione (Slicing)

- È il processo per cui si fissa un valore per almeno uno degli attributi dimensionali e si escludono dall'analisi tutti quegli eventi che non presentano tale valore.
- **Risultato:** un cubo dei fatti con un numero di dimensioni inferiore (almeno di uno) rispetto al cubo sorgente.

data = “15/10/1997”



Slicing

Category	Year	Metrics	Dollar Sales									
		Customer Region	North-East	Mid-Atlantic	South-East	Central	South	North-West	South-West	England	France	Germany
Electronics	1997		\$ 138	\$ 1.774	\$ 384	\$ 138	\$ 2.346	\$ 2.554	\$ 2.184	\$ 566	\$ 199	\$ 5
	1998	◀	\$ 1.184	\$ 4.529	\$ 1.892	\$ 7.232	\$ 651	\$ 9.488	\$ 476	\$ 2.683	\$ 462	\$ 7
Food	1997		\$ 759	\$ 682	\$ 729	\$ 262	\$ 588	\$ 469	\$ 807	\$ 156	\$ 615	\$ 1
	1998	◀	\$ 538	\$ 925	\$ 959	\$ 677	\$ 213	\$ 1.503	\$ 261	\$ 165	\$ 175	\$ 1
Gifts	1997		\$ 2.532	\$ 1.355	\$ 1.854	\$ 1.413	\$ 2.535	\$ 2.132	\$ 1.904	\$ 908	\$ 375	\$ 10
	1998	◀	\$ 1.955	\$ 2.785	\$ 2.800	\$ 2.695	\$ 1.813	\$ 2.844	\$ 1.778	\$ 1.158	\$ 717	\$ 6
Health & Beauty	1997		\$ 624	\$ 640	\$ 1.317	\$ 647	\$ 588	\$ 754	\$ 654	\$ 143	\$ 292	\$ 3
	1998	◀	\$ 611	\$ 887	\$ 566	\$ 382	\$ 499	\$ 1.162	\$ 1.044	\$ 273	\$ 72	
Household	1997		\$ 5.354	\$ 4.112	\$ 5.410	\$ 4.446	\$ 3.058	\$ 3.974	\$ 2.654	\$ 3.545	\$ 2.875	\$ 1.9
	1998	◀	\$ 5.787	\$ 5.320	\$ 5.416	\$ 6.812	\$ 4.334	\$ 5.008	\$ 7.588	\$ 2.139	\$ 3.649	\$ 2.7
Kid's Korner	1997		\$ 201	\$ 398	\$ 485	\$ 186	\$ 409	\$ 323	\$ 396	\$ 105	\$ 34	\$
	1998	◀	\$ 247	\$ 422	\$ 441	\$ 380	\$ 221	\$ 592	\$ 290	\$ 198	\$ 19	\$
Travel	1997		\$ 624	\$ 505	\$ 564	\$ 386	\$ 300	\$ 978	\$ 416	\$ 48	\$ 38	
	1998	◀	\$ 608	\$ 559	\$ 1.096	\$ 611	\$ 464	\$ 316	\$ 573	\$ 257	\$ 198	\$



Filter Details:
Year = 1998

Category	Year	Metrics	Dollar Sales									
		Customer Region	North-East	Mid-Atlantic	South-East	Central	South	North-West	South-West	England	France	Germany
Electronics	1998		\$ 1.184	\$ 4.529	\$ 1.892	\$ 7.232	\$ 651	\$ 9.488	\$ 476	\$ 2.683	\$ 462	\$ 702
Food	1998		\$ 538	\$ 925	\$ 959	\$ 677	\$ 213	\$ 1.503	\$ 261	\$ 165	\$ 175	\$ 100
Gifts	1998		\$ 1.955	\$ 2.785	\$ 2.800	\$ 2.695	\$ 1.813	\$ 2.844	\$ 1.778	\$ 1.158	\$ 717	\$ 686
Health & Beauty	1998		\$ 611	\$ 887	\$ 566	\$ 382	\$ 499	\$ 1.162	\$ 1.044	\$ 273	\$ 72	
Household	1998		\$ 5.787	\$ 5.320	\$ 5.416	\$ 6.812	\$ 4.334	\$ 5.008	\$ 7.588	\$ 2.139	\$ 3.649	\$ 2.791
Kid's Korner	1998		\$ 247	\$ 422	\$ 441	\$ 380	\$ 221	\$ 592	\$ 290	\$ 198	\$ 19	\$ 69
Travel	1998		\$ 608	\$ 559	\$ 1.096	\$ 611	\$ 464	\$ 316	\$ 573	\$ 257	\$ 198	\$ 55

Slicing sul predicato Year = '1998'.

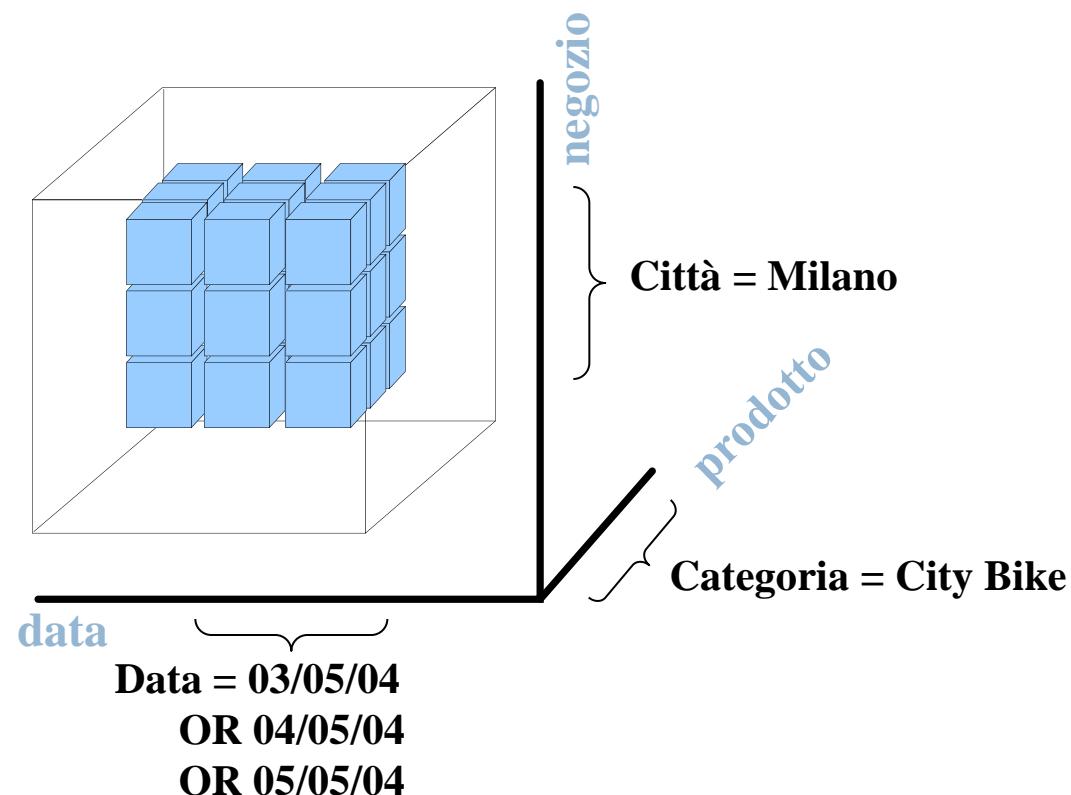
Operatori di restrizione (Dicing)

- Consiste nello stabilire per almeno una delle dimensioni di analisi un sottoinsieme di valori possibili per tale attributo e di escludere quei fatti che non sono associati a nessuno di tali valori.

Città = “Milano”

Categoria = “City Bike”

Data = “03/05/04 OR
04/05/04 OR
05/05/04”



Dicing

	Metrics Customer City	Dollar Sales												
Subcategory	Afton	Akron	Albon	Alcameda	Alka	Allagash	Alta	Altoola	Amestra	Amsterdam	Andersonville	Annap		
Audio						\$ 85								
Automotive	\$ 42	\$ 42				\$ 20		\$ 30						
Chocolate	\$ 30			\$ 50		\$ 25	\$ 30	\$ 22						
Christmas						\$ 7	\$ 26	\$ 15						
Classic Toys														\$ 38
Coffee														
Comfort														
Furniture														
Gadgets														
Games & Puzzles														
Gift Baskets														
Golf	\$ 25													
Hearth														
Jewelry	\$ 75													
Kitchen														
Lawn & Garden	\$ 75													
Learning	\$ 16													
Meat & Cheese														
Miscellaneous														
Natural Remedies	\$ 13													
Pets	\$ 215													
Plants & Flowers	\$ 65	\$ 65												
Safety & Security														
Skin Care														
Sleeping														
Toys & Accessories														

Selezione su un predicato complesso.

Filter Details:
 Category = Electronics
 AND
 Dollar Sales > 80
 AND
 Customer Region = North-West
 AND
 Year = 1997

	Metrics Customer City	Dollar Sales					
Subcategory	Alta	Armstrong	Avery Heights	Lane	Mt. Everest	San Francisco	
Audio		\$ 98		\$ 123	\$ 85		
Comfort		\$ 118			\$ 1,495		
Gadgets	\$ 199					\$ 199	

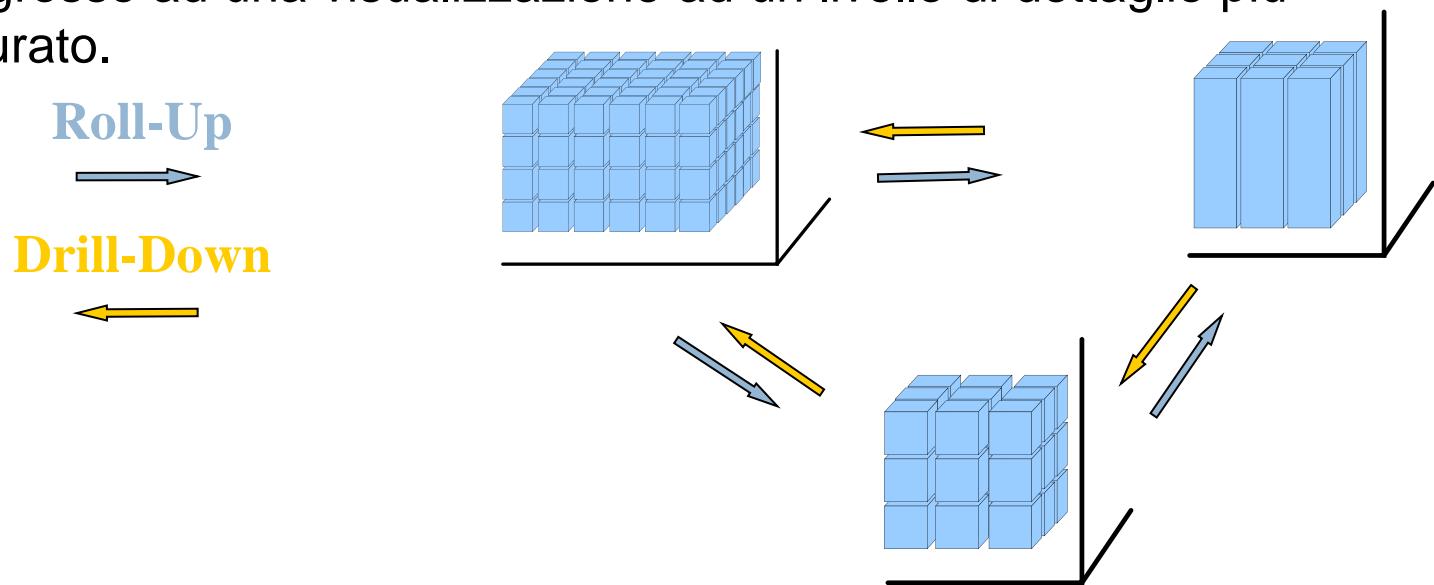
Operatori di aggregazione (Roll-Up e Drill-Down)

□ Roll-Up

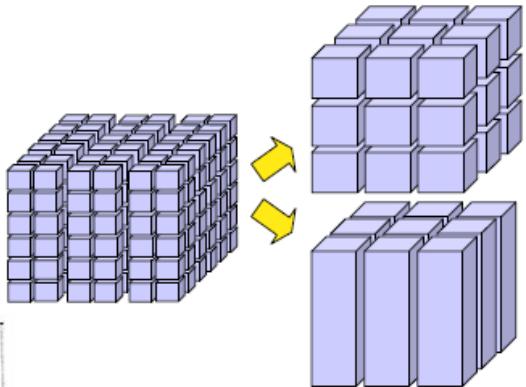
- Consiste nel passare da una visualizzazione ad un livello di dettaglio più fine ad una visualizzazione ad un livello di dettaglio meno accurato.

□ Drill-Down

- Consiste nel passare da una visualizzazione ad un livello di dettaglio più grosso ad una visualizzazione ad un livello di dettaglio più accurato.



Roll-Up



Customer Region	Metrics	Dollar Sales										
		North-East	Mid-Atlantic	South-East	Central	South	North-West	South-West	England	France	Germany	Canada
Month												
Jan 97		\$ 620	\$ 753	\$ 30	\$ 660	\$ 2,405	\$ 1,312	\$ 440	\$ 1,002	\$ 1,002	\$ 383	\$ 210
Feb 97		\$ 258	\$ 252	\$ 800	\$ 975	\$ 160	\$ 582	\$ 744	\$ 310	\$ 799	\$ 118	\$ 357
Mar 97		\$ 648	\$ 244	\$ 148	\$ 250	\$ 1,085	\$ 2,961	\$ 650	\$ 1,240	\$ 119	\$ 142	\$ 96
Apr 97		\$ 787	\$ 588	\$ 447	\$ 486	\$ 226	\$ 506	\$ 601	\$ 119	\$ 550	\$ 85	
May 97		\$ 1,350	\$ 245	\$ 936	\$ 159	\$ 664	\$ 626	\$ 107	\$ 135	\$ 200	\$ 177	\$ 230
Jun 97		\$ 842	\$ 582	\$ 1,281	\$ 937	\$ 240	\$ 774	\$ 176	\$ 1,139	\$ 652	\$ 254	\$ 745
Jul 97		\$ 652	\$ 690	\$ 486	\$ 1,293	\$ 605	\$ 303	\$ 818	\$ 103	\$ 124	\$ 173	\$ 66
Aug 97		\$ 1,783	\$ 304	\$ 1,032	\$ 170	\$ 398	\$ 356	\$ 432	\$ 190	\$ 241	\$ 407	\$ 259
Sep 97		\$ 581	\$ 778	\$ 3,558	\$ 587	\$ 440	\$ 1,652	\$ 1,071	\$ 315	\$ 210	\$ 202	
Oct 97		\$ 2,291	\$ 1,840	\$ 600	\$ 656	\$ 1,300	\$ 718	\$ 1,210	\$ 427	\$ 220	\$ 520	\$ 65
Nov 97		\$ 39	\$ 1,602	\$ 1,082	\$ 1,187	\$ 842	\$ 759	\$ 745	\$ 232	\$ 101	\$ 1,037	\$ 37
Dec 97		\$ 381	\$ 1,588	\$ 343	\$ 118	\$ 1,459	\$ 635	\$ 2,021	\$ 259	\$ 210	\$ 119	\$ 189
Jan 98		\$ 311	\$ 1,174	\$ 2,634	\$ 1,190	\$ 954	\$ 2,083	\$ 1,351	\$ 747	\$ 426	\$ 447	\$ 1,141
Feb 98		\$ 2,518	\$ 702	\$ 1,123	\$ 1,336	\$ 1,227	\$ 3,887	\$ 545	\$ 268	\$ 277	\$ 282	
Mar 98		\$ 2,459	\$ 1,523	\$ 1,178	\$ 4,708	\$ 1,420	\$ 3,514	\$ 1,948	\$ 1,705	\$ 276	\$ 1,168	\$ 63
Apr 98		\$ 407	\$ 841	\$ 524	\$ 712	\$ 133	\$ 2,486	\$ 49	\$ 390	\$ 1,298	\$ 221	\$ 46
May 98		\$ 667	\$ 1,721	\$ 440	\$ 148	\$ 80	\$ 1,310	\$ 303	\$ 104	\$ 657	\$ 65	
Jun 98		\$ 699	\$ 1,096	\$ 898	\$ 353	\$ 902	\$ 839		\$ 230	\$ 155	\$ 105	\$ 75
Jul 98		\$ 586	\$ 1,897	\$ 412	\$ 226	\$ 406	\$ 361	\$ 1,628	\$ 267	\$ 1,011	\$ 41	\$ 184
Aug 98		\$ 894	\$ 326	\$ 792	\$ 1,832	\$ 1,199	\$ 295	\$ 1,816	\$ 277	\$ 102	\$ 118	\$ 115
Sep 98		\$ 338	\$ 3,179	\$ 505	\$ 427	\$ 99	\$ 2,976	\$ 885	\$ 135	\$ 85	\$ 1,110	\$ 510
Oct 98		\$ 544	\$ 413	\$ 1,467	\$ 209	\$ 679	\$ 706	\$ 556	\$ 480	\$ 485	\$ 99	\$ 160
Nov 98		\$ 671	\$ 459	\$ 1,471	\$ 2,066	\$ 701	\$ 716	\$ 986	\$ 1,127	\$ 154	\$ 440	\$ 361
Dec 98		\$ 836	\$ 2,096	\$ 1,726	\$ 3,642	\$ 395	\$ 1,740	\$ 1,943	\$ 1,143	\$ 366	\$ 307	\$ 118

Roll-Up sulla gerarchia temporale.



Customer Region	Metrics	Dollar Sales										
		North-East	Mid-Atlantic	South-East	Central	South	North-West	South-West	England	France	Germany	Canada
Quarter												
Q1 1997		\$ 1,526	\$ 1,249	\$ 978	\$ 1,885	\$ 3,650	\$ 4,855	\$ 1,834	\$ 2,552	\$ 1,920	\$ 643	\$ 663
Q2 1997		\$ 2,979	\$ 1,415	\$ 2,664	\$ 1,582	\$ 1,130	\$ 1,906	\$ 884	\$ 1,393	\$ 1,402	\$ 516	\$ 975
Q3 1997		\$ 3,016	\$ 1,772	\$ 5,076	\$ 2,050	\$ 1,443	\$ 2,311	\$ 2,321	\$ 508	\$ 575	\$ 782	\$ 325
Q4 1997		\$ 2,711	\$ 5,030	\$ 2,025	\$ 1,961	\$ 3,601	\$ 2,112	\$ 3,976	\$ 918	\$ 531	\$ 1,676	\$ 291
Q1 1998		\$ 5,288	\$ 3,399	\$ 4,935	\$ 9,174	\$ 3,601	\$ 9,484	\$ 3,844	\$ 2,720	\$ 979	\$ 1,897	\$ 1,204
Q2 1998		\$ 1,773	\$ 3,658	\$ 1,862	\$ 1,213	\$ 1,115	\$ 4,635	\$ 352	\$ 724	\$ 2,110	\$ 391	\$ 121
Q3 1998		\$ 1,818	\$ 5,402	\$ 1,709	\$ 2,485	\$ 1,704	\$ 3,632	\$ 4,329	\$ 679	\$ 1,198	\$ 1,269	\$ 809
Q4 1998		\$ 2,051	\$ 2,968	\$ 4,664	\$ 5,917	\$ 1,775	\$ 3,162	\$ 3,485	\$ 2,750	\$ 1,005	\$ 846	\$ 639

Roll-Up (2)

Category	Year	Metrics		Dollar Sales									
		Customer Region	North-East	Mid-Atlantic	South-East	Central	South	North-West	South-West	England	France	Germany	
Electronics	1997		\$ 138	\$ 1.774	\$ 384	\$ 138	\$ 2.346	\$ 2.554	\$ 2.184	\$ 566	\$ 199	\$ 1	
	1998		\$ 1.184	\$ 4.529	\$ 1.892	\$ 7.232	\$ 651	\$ 9.488	\$ 476	\$ 2.683	\$ 462	\$ 7	
Food	1997		\$ 759	\$ 682	\$ 729	\$ 262	\$ 588	\$ 469	\$ 807	\$ 156	\$ 615	\$ 1	
	1998		\$ 538	\$ 925	\$ 959	\$ 677	\$ 213	\$ 1.503	\$ 261	\$ 165	\$ 175	\$ 1	
Gifts	1997		\$ 2.532	\$ 1.355	\$ 1.854	\$ 1.413	\$ 2.535	\$ 2.132	\$ 1.904	\$ 908	\$ 375	\$ 10	
	1998		\$ 1.955	\$ 2.785	\$ 2.800	\$ 2.695	\$ 1.813	\$ 2.844	\$ 1.778	\$ 1.158	\$ 717	\$ 6	
Health & Beauty	1997		\$ 624	\$ 640	\$ 1.317	\$ 647	\$ 588	\$ 754	\$ 654	\$ 143	\$ 292	\$ 3	
	1998		\$ 611	\$ 887	\$ 566	\$ 382	\$ 499	\$ 1.162	\$ 1.044	\$ 273	\$ 72		
Household	1997		\$ 5.354	\$ 4.112	\$ 5.410	\$ 4.446	\$ 3.058	\$ 3.974	\$ 2.654	\$ 3.545	\$ 2.875	\$ 1.9	
	1998		\$ 5.787	\$ 5.320	\$ 5.416	\$ 6.812	\$ 4.334	\$ 5.008	\$ 7.588	\$ 2.139	\$ 3.649	\$ 2.7	
Kid's Korner	1997		\$ 201	\$ 398	\$ 485	\$ 186	\$ 409	\$ 323	\$ 396	\$ 105	\$ 34	\$	
	1998		\$ 247	\$ 422	\$ 441	\$ 380	\$ 221	\$ 592	\$ 290	\$ 198	\$ 19	\$	
Travel	1997		\$ 624	\$ 505	\$ 564	\$ 386	\$ 300	\$ 978	\$ 416	\$ 48	\$ 38		
	1998		\$ 608	\$ 559	\$ 1.096	\$ 611	\$ 464	\$ 316	\$ 573	\$ 257	\$ 198	\$	

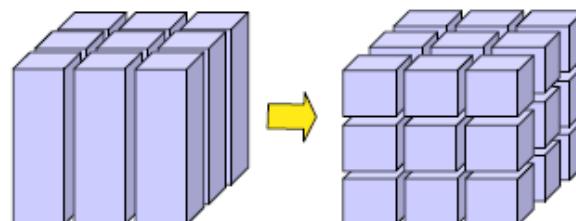
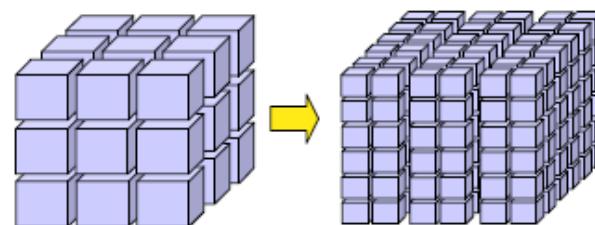


Category	Year	Metrics		Dollar Sales	
		Customer Region	North-East	Mid-Atlantic	South-East
Electronics	1997		\$ 10.616		
	1998		\$ 29.299		
Food	1997		\$ 5.300		
	1998		\$ 5.638		
Gifts	1997		\$ 16.315		
	1998		\$ 20.047		
Health & Beauty	1997		\$ 6.042		
	1998		\$ 5.665		
Household	1997		\$ 38.383		
	1998		\$ 50.391		
Kid's Korner	1997		\$ 2.559		
	1998		\$ 2.943		
Travel	1997		\$ 4.497		
	1998		\$ 4.792		

Roll-Up con eliminazione della gerarchia cliente.

Drill-Down

- I dettagli dei dati vengono aumentati:
 - ▣ Il dettaglio viene aumentato in una dimensione percorrendo la gerarchia.
 - **Es:** raggruppare per *città, mese* → raggruppare per *negozio, mese*
 - ▣ Aggiungendo una intera dimensione.
 - **Es:** raggruppare per *prodotto* → raggruppare per *prodotto, città*
- Di solito il Drill-Down opera su un sottoinsieme di dati prodotto da una precedente query.



Drill-Down (2)

	Metrics	Dollar Sales											
	Customer Region	North-East	Mid-Atlantic	South-East	Central	South	North-West	South-West	England	France	Germany	Canada	
Quarter													
Q1 1997		\$ 1.526	\$ 1.249	\$ 978	\$ 1.885	\$ 3.650	\$ 4.855	\$ 1.834	\$ 2.552	\$ 1.920	\$ 543	\$ 663	
Q2 1997		\$ 2.979	\$ 1.415	\$ 2.664	\$ 1.582	\$ 1.130	\$ 1.906	\$ 884	\$ 1.393	\$ 1.402	\$ 516	\$ 975	
Q3 1997		\$ 3.016	\$ 1.772	\$ 5.076	\$ 2.050	\$ 1.443	\$ 2.311	\$ 2.321	\$ 608	\$ 575	\$ 782	\$ 325	
Q4 1997		\$ 2.711	\$ 5.030	\$ 2.025	\$ 1.961	\$ 3.601	\$ 2.112	\$ 3.976	\$ 918	\$ 531	\$ 1.676	\$ 291	
Q1 1998		\$ 5.288	\$ 3.399	\$ 4.935	\$ 9.174	\$ 3.601	\$ 9.484	\$ 3.844	\$ 2.720	\$ 979	\$ 1.897	\$ 1.204	
Q2 1998		\$ 1.773	\$ 3.658	\$ 1.862	\$ 1.213	\$ 1.115	\$ 4.635	\$ 352	\$ 724	\$ 2.110	\$ 391	\$ 121	
Q3 1998		\$ 1.818	\$ 5.402	\$ 1.709	\$ 2.485	\$ 1.704	\$ 3.632	\$ 4.329	\$ 679	\$ 1.198	\$ 1.269	\$ 809	
Q4 1998		\$ 2.051	\$ 2.968	\$ 4.664	\$ 5.917	\$ 1.775	\$ 3.162	\$ 3.485	\$ 2.750	\$ 1.005	\$ 346	\$ 639	



	Metrics	Dollar Sales												
	Customer City	Arlin	San Pedro	Springfield	Chappel Hill	Scranburg	Pebble Beach	Martinsville	Maddon	Peoria	Pecos	Lake Barkley	Alcameda	Fingers Lake
Quarter														
Q1 1997		\$ 675										\$ 39		\$ 135
Q2 1997					\$ 203						\$ 53		\$ 252	\$ 63
Q3 1997					\$ 276								\$ 79	\$ 98
Q4 1997		\$ 215	\$ 124		\$ 140	\$ 174		\$ 113	\$ 45	\$ 192	\$ 348		\$ 30	\$ 119
Q1 1998										\$ 85				
Q2 1998										\$ 12	\$ 17			
Q3 1998		\$ 734						\$ 25	\$ 1.535					
Q4 1998								\$ 219	\$ 119	\$ 142				
											\$ 85	\$ 1.533		

Drill-Down sulla gerarchia del cliente.

Drill-Down (3)

Category	Metrics	Dollar Sales		
		Year	1997	1998
Electronics			\$ 10.616	\$ 29.299
Food			\$ 5.300	\$ 5.638
Gifts			\$ 16.315	\$ 20.047
Health & Beauty			\$ 6.042	\$ 5.665
Household			\$ 38.383	\$ 50.391
Kid's Korner			\$ 2.559	\$ 2.943
Travel			\$ 4.497	\$ 4.792



Category	Metrics	Customer Region	Dollar Sales											
			North-East		Mid-Atlantic		South-East		Central		South		North-West	
Year	1997	1998	1997	1998	1997	1998	1997	1998	1997	1998	1997	1998	1997	1998
Electronics			\$ 138	\$ 1.184	\$ 1.774	\$ 4.529	\$ 384	\$ 1.892	\$ 138	\$ 7.232	\$ 2.346	\$ 651	\$ 2.554	\$ 9.488
Food			\$ 759	\$ 538	\$ 682	\$ 925	\$ 729	\$ 959	\$ 262	\$ 677	\$ 588	\$ 213	\$ 469	\$ 1.503
Gifts			\$ 2.532	\$ 1.955	\$ 1.355	\$ 2.785	\$ 1.854	\$ 2.800	\$ 1.413	\$ 2.695	\$ 2.535	\$ 1.813	\$ 2.132	\$ 2.844
Health & Beauty			\$ 624	\$ 611	\$ 640	\$ 887	\$ 1.317	\$ 566	\$ 647	\$ 382	\$ 588	\$ 499	\$ 754	\$ 1.162
Household			\$ 5.354	\$ 5.787	\$ 4.112	\$ 5.320	\$ 5.410	\$ 5.416	\$ 4.446	\$ 6.812	\$ 3.058	\$ 4.334	\$ 3.974	\$ 5.008
Kid's Korner			\$ 201	\$ 247	\$ 398	\$ 422	\$ 495	\$ 441	\$ 186	\$ 380	\$ 409	\$ 221	\$ 323	\$ 592
Travel			\$ 624	\$ 608	\$ 505	\$ 559	\$ 564	\$ 1.096	\$ 386	\$ 611	\$ 300	\$ 464	\$ 978	\$ 316

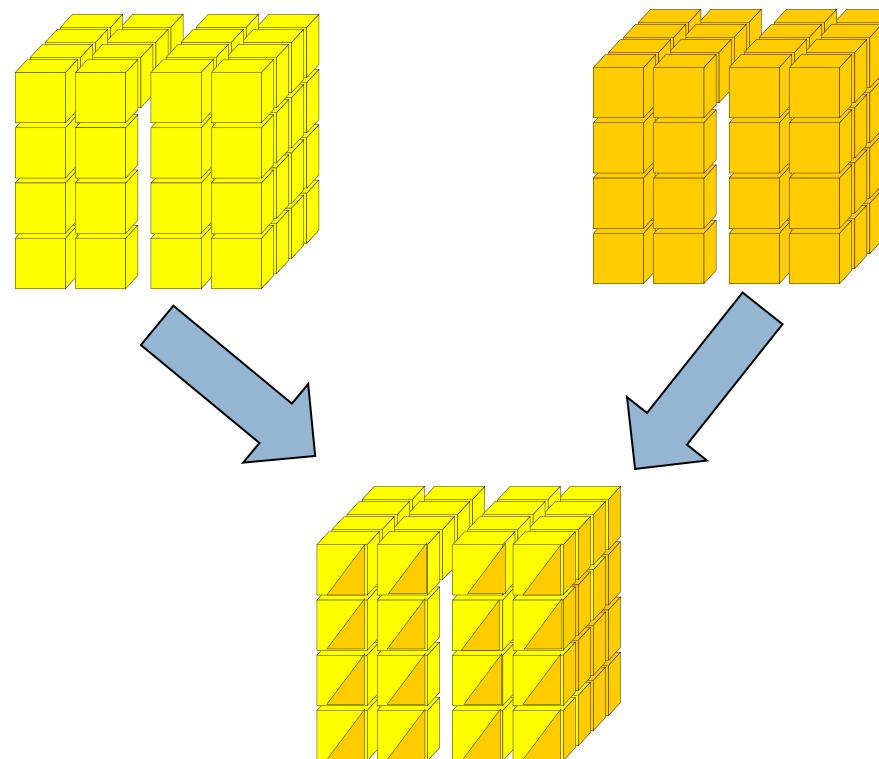
Drill-Down con aggiunta di una dimensione.

Operatori di aggregazione (Drill-Across)

- Il Drill-Across consiste nello stabilire un confronto tra due o più cubi correlati.
 - Per ottenere una visualizzazione comparata di due diverse misure.
 - Per il calcolo di misure derivate dai dati presenti sui cubi.
- Per poter effettuare queste operazioni su cubi presenti su Data mart differenti occorre che:
 - I due sistemi siano in qualche modo coordinati dai meta-dati presenti nel DW.

Operatori di aggregazione (Drill-Across) (2)

□ Esempio:

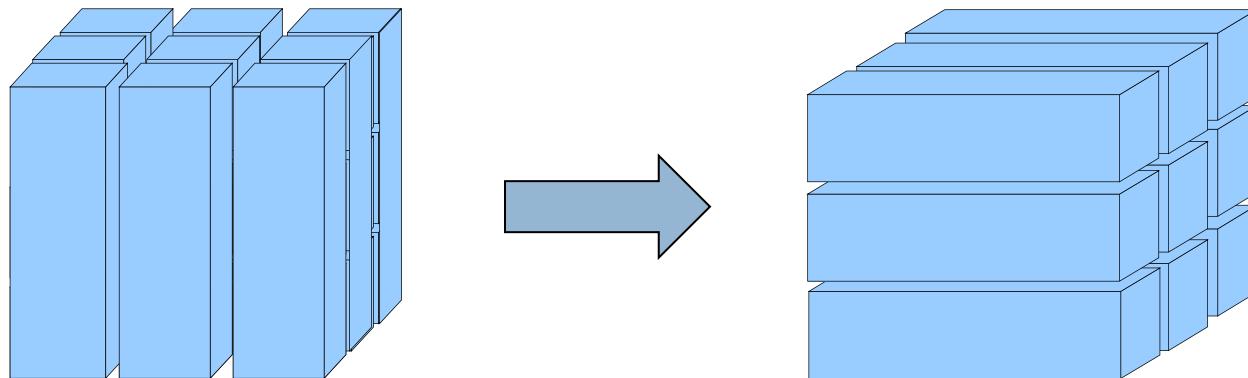


Drill-Through

- Consiste nel passaggio dai dati aggregati multidimensionalmente del DW ai dati operazionali presenti nelle sorgenti o nel livello riconciliato.

Pivoting

- L'operatore di pivoting consiste nel ruotare gli assi di visualizzazione del cubo dei fatti mantenendo invariato il livello di aggregazione ed il numero delle dimensioni:
 - ▣ Ciò incrementa la leggibilità delle stesse informazioni.



Pivoting (2)

Pivoting su una
tabella
bidimensionale.

Category	Year	Metrics		Dollar Sales	
		1997	1998	\$ 10.616	\$ 29.299
Food	1997	\$ 5.300			
	1998	\$ 5.638			
Gifts	1997	\$ 16.315			
	1998	\$ 20.047			
Health & Beauty	1997	\$ 6.042			
	1998	\$ 5.665			
Household	1997	\$ 38.383			
	1998	\$ 50.391			
Kid's Korner	1997	\$ 2.559			
	1998	\$ 2.943			
Travel	1997	\$ 4.497			
	1998	\$ 4.792			



Category	Year	Metrics		Dollar Sales	
		1997	1998	1997	1998
Electronics		\$ 10.616	\$ 29.299		
Food		\$ 5.300	\$ 5.638		
Gifts		\$ 16.315	\$ 20.047		
Health & Beauty		\$ 6.042	\$ 5.665		
Household		\$ 38.383	\$ 50.391		
Kid's Korner		\$ 2.559	\$ 2.943		
Travel		\$ 4.497	\$ 4.792		

Category	Year	Metrics		Dollar Sales									
		Customer Region	North-East	Mid-Atlantic	South-East	Central	South	North-West	South-West	England	France	Germany	
Electronics	1997		\$ 138	\$ 1.774	\$ 384	\$ 138	\$ 2.346	\$ 2.554	\$ 2.184	\$ 566	\$ 199	\$ 7	
	1998		\$ 1.184	\$ 4.529	\$ 1.892	\$ 7.232	\$ 651	\$ 9.488	\$ 476	\$ 2.683	\$ 462	\$ 1503	
Food	1997		\$ 759	\$ 682	\$ 729	\$ 262	\$ 588	\$ 469	\$ 807	\$ 156	\$ 615	\$ 1	
	1998		\$ 538	\$ 925	\$ 959	\$ 677	\$ 213	\$ 1.503	\$ 261	\$ 165	\$ 175	\$ 1	
Gifts	1997		\$ 2.532	\$ 1.355	\$ 1.854	\$ 1.413	\$ 2.535	\$ 2.132	\$ 1.904	\$ 908	\$ 375	\$ 1.0	
	1998		\$ 1.955	\$ 2.785	\$ 2.800	\$ 2.695	\$ 1.813	\$ 2.844	\$ 1.778	\$ 1.158	\$ 717	\$ 6	
Health & Beauty	1997		\$ 624	\$ 640	\$ 1.317	\$ 647	\$ 588	\$ 754	\$ 654	\$ 143	\$ 292	\$ 3	
	1998		\$ 611	\$ 887	\$ 566	\$ 382	\$ 499	\$ 1.162	\$ 1.044	\$ 273	\$ 72		
Household	1997		\$ 5.354	\$ 4.112	\$ 5.410	\$ 4.446	\$ 3.058	\$ 3.974	\$ 2.654	\$ 3.545	\$ 2.875	\$ 1.9	
	1998		\$ 5.787	\$ 5.320	\$ 5.416	\$ 6.812	\$ 4.334	\$ 5.008	\$ 7.588	\$ 2.139	\$ 3.649	\$ 2.7	
Kid's Korner	1997		\$ 201	\$ 398	\$ 485	\$ 186	\$ 409	\$ 323	\$ 396	\$ 105	\$ 34	\$	
	1998		\$ 247	\$ 422	\$ 441	\$ 380	\$ 221	\$ 592	\$ 290	\$ 198	\$ 19	\$	
Travel	1997		\$ 624	\$ 505	\$ 564	\$ 386	\$ 300	\$ 978	\$ 416	\$ 48	\$ 38		
	1998		\$ 608	\$ 559	\$ 1.096	\$ 611	\$ 464	\$ 318	\$ 573	\$ 257	\$ 198	\$	



Pivoting su una
tabella
tridimensionale.

Category	Year	Metrics		Dollar Sales									
		Customer Region	North-East	Mid-Atlantic	South-East	Central	South	North-West	South-West	England	France	Germany	
			1997	1998	1997	1998	1997	1998	1997	1998	1997	1998	
Electronics			\$ 138	\$ 1.184	\$ 1.774	\$ 4.529	\$ 384	\$ 1.992	\$ 138	\$ 7.232	\$ 2.346	\$ 651	\$ 2.554
Food			\$ 759	\$ 538	\$ 682	\$ 925	\$ 729	\$ 959	\$ 262	\$ 677	\$ 588	\$ 213	\$ 1.503
Gifts			\$ 2.532	\$ 1.955	\$ 1.355	\$ 2.785	\$ 1.854	\$ 2.800	\$ 1.413	\$ 2.695	\$ 2.535	\$ 1.813	\$ 2.132
Health & Beauty			\$ 624	\$ 611	\$ 640	\$ 887	\$ 1.317	\$ 566	\$ 647	\$ 382	\$ 588	\$ 499	\$ 754
Household			\$ 5.354	\$ 5.787	\$ 4.112	\$ 5.320	\$ 5.410	\$ 5.416	\$ 4.446	\$ 6.812	\$ 3.058	\$ 4.334	\$ 3.974
Kid's Korner			\$ 201	\$ 247	\$ 398	\$ 422	\$ 485	\$ 441	\$ 186	\$ 380	\$ 409	\$ 221	\$ 323
Travel			\$ 624	\$ 608	\$ 505	\$ 559	\$ 564	\$ 1.096	\$ 386	\$ 611	\$ 464	\$ 978	\$ 316

Approcci all'implementazione di un DW

□ ROLAP: Relational OLAP

- Implementazione su DBMS relazionali.
- Necessarie tipologie specifiche di schemi che permettono di traslare il modello multidimensionale su attributi, relazioni e vincoli di integrità:
 - Ruolo svolto dallo *star schema* (**schema a stella**).
- Denormalizzazione (violazione consapevole della 3NF).
- Ridondanza materializzando tabelle derivate (viste).
- Basse prestazioni dovute a costose operazioni di JOIN su tabelle di elevate dimensioni.

□ MOLAP: Multidimensional OLAP

- Implementazione su DBMS multidimensionali.
- Modello *ad hoc* ed accesso di tipo posizionale.
- Le operazioni multidimensionali sono realizzabili in modo semplice e naturale senza ricorrere alle operazioni di JOIN.
- Ottime prestazioni.

Altri aspetti da considerare

- Qualità
 - ▣ La qualità di un processo misura la sua aderenza agli obiettivi degli utenti.
- Sicurezza
- Evoluzione
 - ▣ A livello dei *dati* e di *schema*.

Qualità

- I fattori che caratterizzano la qualità dei dati:
 - *Accuratezza*: conformità tra il valore memorizzato e quello reale.
 - *Attualità*: il dato memorizzato non è obsoleto.
 - *Completezza*: non mancano informazioni.
 - *Consistenza*: la rappresentazione dei dati è uniforme.
 - *Disponibilità*: i dati sono facilmente disponibili all'utente.
 - *Tracciabilità*: è possibile risalire alla fonte di ciascun dato.
 - *Chiarezza*: i dati sono facilmente interpretabili.

Sicurezza ed Evoluzione

- Sicurezza:
 - Controllo delle autorizzazioni.
 - Controllo sulla mole di dati trasferiti dalle sorgenti.
- Evoluzione (*delle informazioni nel DW nel tempo*):
 - A livello dei *dati*.
 - Aggiunta di nuove categorie di dati.
 - Possibilità di cambiare la categoria di un dato.
 - A livello di *schema*.
 - Se mutano i requisiti dell'utente.
 - Se variano le sorgenti di dati.



Ciclo di vita dei sistemi di data warehousing

Ciclo di vita

- Vedremo alcune metodologie per la gestione dell'intero ciclo di vita dei sistemi di *data warehousing*:
 - Definizione delle fasi di progettazione di un data-mart.

Fattori di rischio (1)

- Rischi legati alla gestione del progetto:
 - ▣ Scarsa disponibilità a condividere informazioni tra reparti aziendali.
 - ▣ Incapacità del progettista di presentare una convincente valutazione dei costi e dei benefici.
- Rischi legati alle tecnologie:
 - ▣ Rapida evoluzione delle tecnologie.
 - ▣ Mancanza di standard riconosciuti e testati.
 - ▣ Scarsa scalabilità delle architetture in termini di volumi dati e numero di utenti.
 - ▣ Assenza di estensibilità per accogliere nuove tecnologie.
 - ▣ Gestione inefficiente dell'interscambio di metadati tra componenti.

Fattori di rischio (2)

- Rischi legati ai dati ed alla progettazione:
 - Risultati di scarso valore, causati da sorgenti instabili ed inaffidabili.
 - Specifica inaccurata dei requisiti.
 - Inaccuratezza dei primi prototipi, il che mina la fiducia degli utenti nell'intero progetto.
- Rischi legati all'organizzazione:
 - Incapacità di coinvolgere attivamente l'utente finale nel progetto.
 - Difficoltà a sfruttare i risultati ottenuti a causa di inerzia organizzativa.

Approccio Top-Down

- Adottare un approccio Top-Down significa:
 - Analizzare i bisogni globali dell'intera azienda.
 - Pianificare lo sviluppo del DW.
 - Progettarlo e realizzarlo nella sua interezza.

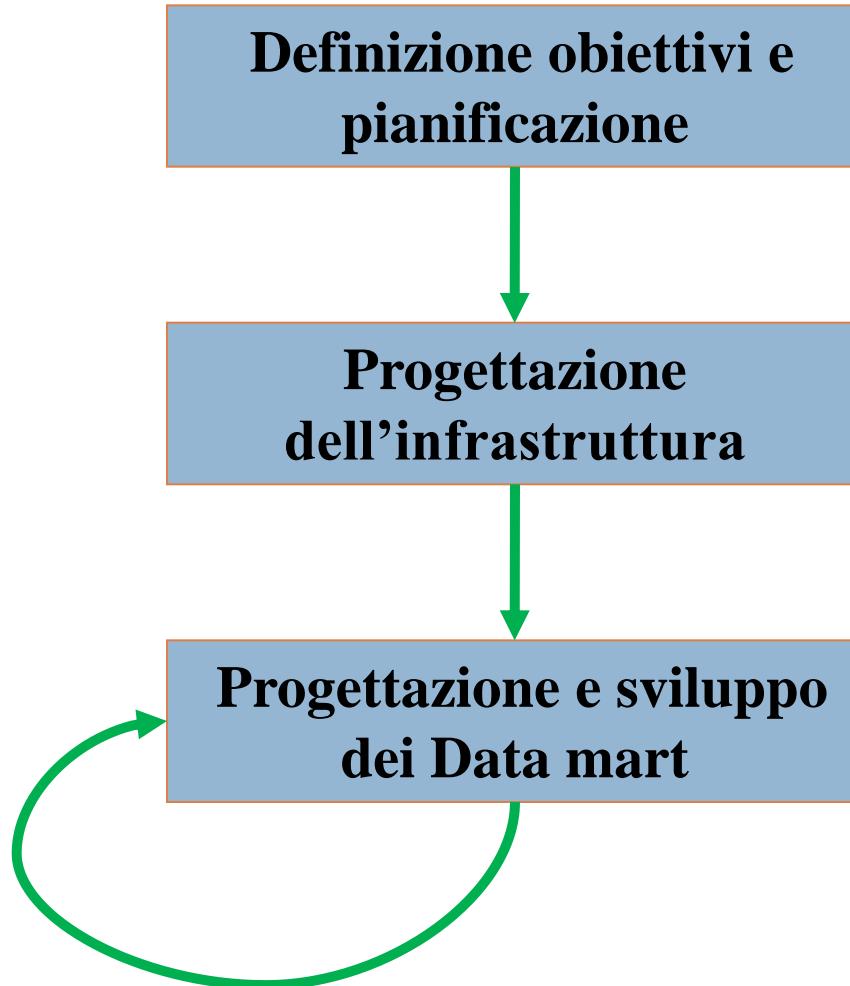
Top-Down: Vantaggi e Svantaggi

- **Vantaggi**
 - Visione globale dell'obiettivo.
 - DW consistente e ben integrato.
- **Svantaggi**
 - Tempi lunghi di realizzazione.
 - Complessità nell'analisi e nella riconciliazione di tutte le fonti.
 - Impossibilità di prevedere le esigenze particolari delle diverse aree aziendali.
 - La non breve consegna non permette di verificare l'utilità del progetto.

Approccio Bottom-Up

- DW costruito in modo incrementale.
- Data mart concentrati su una specifica area di interesse.
- È più facile costruire prototipi.
- È importante riporre la massima attenzione sulla scelta del primo Data mart:
 - Deve essere quello più strategico per l'azienda.
 - Deve ricoprire un ruolo centrale per l'intero DW.
 - Deve poggiare su fonti dati disponibili e consistenti.

Bottom-Up: Ciclo di vita



Bottom-Up: Ciclo di vita (2)

- Definizione degli obiettivi e pianificazione:
 - Individuazione obiettivi e confini del sistema.
 - Stima delle dimensioni.
 - Valutazione dei costi e del valore aggiunto.
 - Scelta dell'approccio per la costruzione.
 - Analisi dei rischi e delle aspettative.
 - Studio delle competenze gruppo di lavoro.
- Progettazione dell'infrastruttura:
 - Scelte architetturali.
 - Scelte degli strumenti.
- Progettazione e sviluppo dei Data mart.

Business Dimensional Lifecycle (BDL)

- Ciclo di vita per la progettazione, lo sviluppo e l'attuazione dei sistemi di data warehousing.
- Proposto da *Kimball* (1998):

Gestione progetto

Pianificazione

Definizione dei requisiti

Progetto dell'architettura

Selezione e installazione prodotti

TECNOLOGIA

Modellazione dimensionale

Progettazione fisica

DATI

Progettazione e sviluppo alimentazione

APPLICAZIONI

Specific applicazioni utente

Sviluppo applicazioni utente

Attuazione

Manutenzione

Fasi BDL (1)

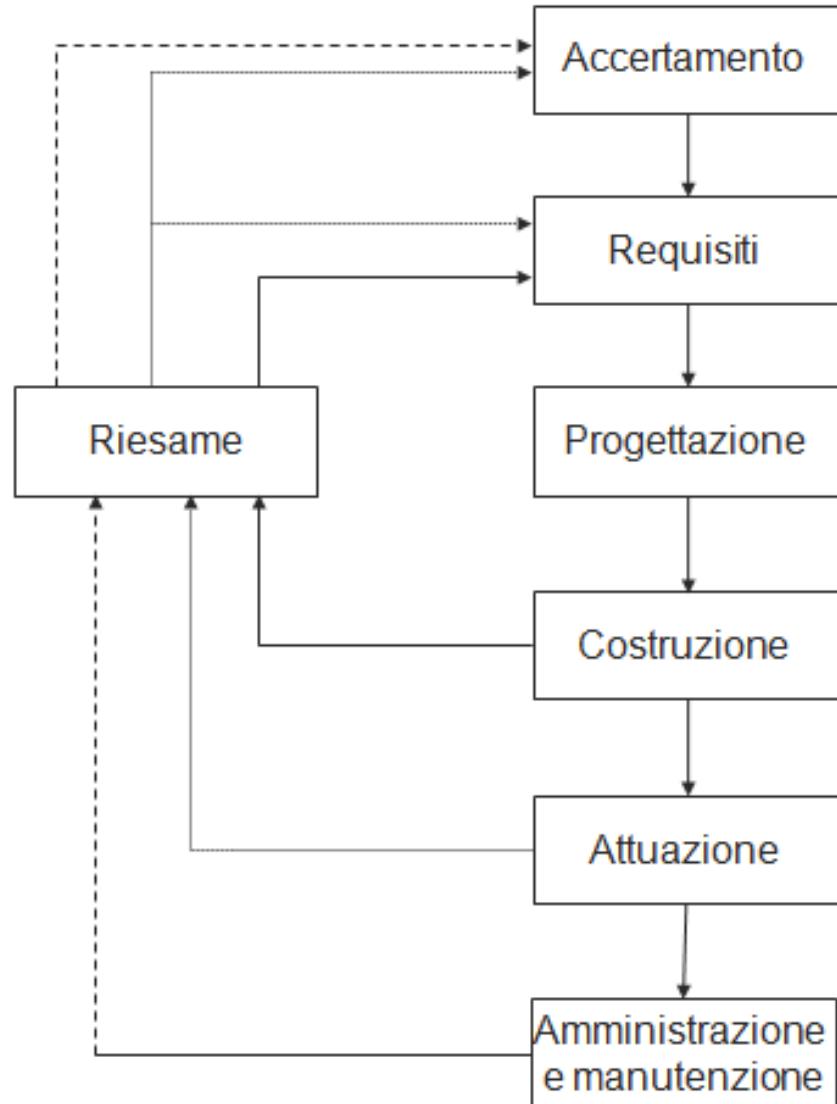
- Pianificazione:
 - Scopi e confini del sistema.
 - Valutazione impatti organizzativi.
 - Stima costi e benefici.
 - Allocazione delle risorse necessarie.
- Definizione dei requisiti:
 - Massima utilità e redditività del sistema.
 - Catturare i fattori chiave e trasformarli in specifiche.
 - Divisa in:
 - Fase dei dati.
 - Fase della tecnologia.
 - Fase delle applicazioni.

Fasi BDL (2)

- Attuazione:
 - ▣ Comporta l'effettivo avviamento del sistema sviluppato.
- Manutenzione:
 - ▣ Assicura il supporto e la formazione degli utenti.
- Gestione del progetto:
 - ▣ Occupa tutte le fasi del ciclo di vita.
 - ▣ Permette di mantenere le diverse attività sincronizzate.

Rapid Warehousing Methodology (RWM)

- Metodologia iterativa e evolutiva.
- Suddivide grossi progetti in sottoprogetti meno rischiosi (*build*).
- Ogni sottoprogetto riprende l'ambiente di quello precedente, estendendolo con nuove funzionalità.



Fasi RWM (1)

- Accertamento (pianificazione di Kimball):
 - Fattibilità del progetto da parte dell'azienda.
 - Scopi.
 - Rischi e benefici.
- Requisiti (applicazioni utente di Kimball):
 - Specifiche di analisi.
 - Specifiche di progetto.
 - Specifiche di architettura.

Fasi RWM (2)

- Progettazione:
 - Progetto logico e fisico dei dati.
 - Progetto dell'alimentazione.
 - Selezione degli strumenti d'implementazione.
- Costruzione:
 - Implementazione e popolazione del DW.
 - Sviluppo e collaudo delle applicazioni front-end.
- Attuazione:
 - Il sistema viene consegnato e avviato.
 - Gli utenti sono adeguatamente addestrati.

Fasi RWM (3)

- Amministrazione e manutenzione:
 - Presente durante tutta la vita del sistema.
 - Estensione delle funzionalità.
 - Ridimensionamento dell'architettura.
 - Controllo della qualità dei dati.
- Riesame:
 - Verifica dell'implementazione.
 - Accertamento che il sistema sia stato ben accettato dall'organizzazione.
 - Misura dei benefici effettivi.

Progettazione di un Data mart

- Analisi e riconciliazione delle fonti dati:
 - Analizzare e comprendere gli schemi delle sorgenti (*ricognizione*).
 - Normalizzazione (**da non confondere**).
 - Trasformare i dati per portare alla luce correlazioni utili precedentemente inespresse.
 - Valutare la qualità dei dati.
- Analisi dei requisiti:
 - Raccolta, filtro e documentazione dei requisiti.
 - Scelta dei fatti.
 - Granularità dei fatti.
 - Compromesso tra velocità e dettaglio.

Progettazione di un Data mart (2)

- Progettazione concettuale:
 - Può essere usato il DFM (*Dimensional Fact Model*).
 - Creazione degli schemi di fatto.
- Raffinamento del carico di lavoro e validazione dello schema concettuale:
 - Formulazione delle interrogazioni direttamente sullo schema concettuale.
 - Verifica che le interrogazioni siano effettivamente esprimibili.

Progettazione di un Data mart (3)

- Progettazione logica:
 - Scelta dell'implementazione:
 - ROLAP o MOLAP.
 - Schemi logici.
 - Materializzazione delle viste.
 - Frammentazione verticale e orizzontale.
- Progettazione dell'alimentazione:
 - Decisioni riguardanti il processo di alimentazione del livello riconciliato e del Data mart.

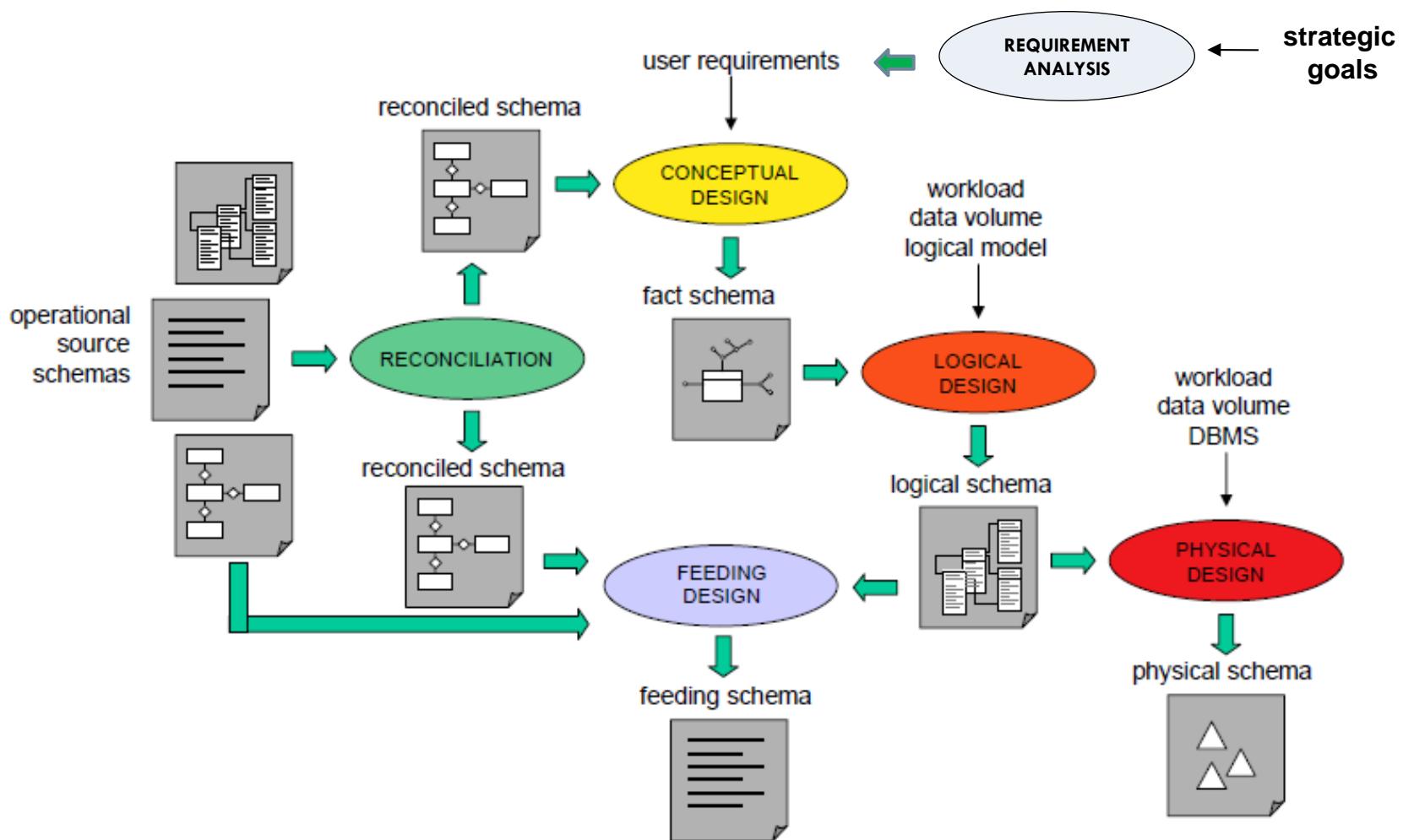
Progettazione di un Data mart (4)

- Progettazione fisica:
 - Scelta degli indici per ottimizzare le prestazioni.
 - Riferimento ad un particolare DBMS.
 - Carico di lavoro e volume dei dati.

Ciclo di vita

Fase	Ingresso	Uscita	Figure Coinvolte
Analisi e riconciliazione delle fonti	Schemi delle sorgenti	Schema riconciliato	Progettista; amministratore db operazionale;
Analisi dei requisiti	Obiettivi strategici	Specifiche dei requisiti; carico di lavoro preliminare	Progettista; utenti finali
Progettazione concettuale	Schema riconciliato; specifica dei requisiti	Schema di fatto	Progettista; utenti finali
Raffinamento carico di lavoro, validazione schema concettuale	Schemi di fatto; carico di lavoro preliminare	Carico di lavoro; schemi di fatto validati	Progettista; utenti finali
Progettazione Logica	Schemi di fatto; modello logico target; carico di lavoro	Schema logico del Data mart	Progettista
Progettazione dell'alimentazione	Schemi delle sorgenti; schema riconciliato; schema logico del Data mart.	Procedure di alimentazione	Progettista; amministratori db operazionale
Progettazione fisica	Schema logico del Data mart; DBMS target; carico di lavoro	Schema fisico del Data mart	Progettista

Le sette fasi della progettazione





Analisi e riconciliazione delle fonti dati

Analisi e riconciliazione delle fonti dati

- Questa fase richiede di definire e documentare lo schema del livello dei dati operazionali, a partire dal quale verrà alimentato il Data mart.

- Riceve in ingresso gli schemi delle sorgenti e produce un insieme di meta-dati che modellano lo schema riconciliato e le corrispondenze tra gli elementi di quest'ultimo e quelli del sistema operazionale.

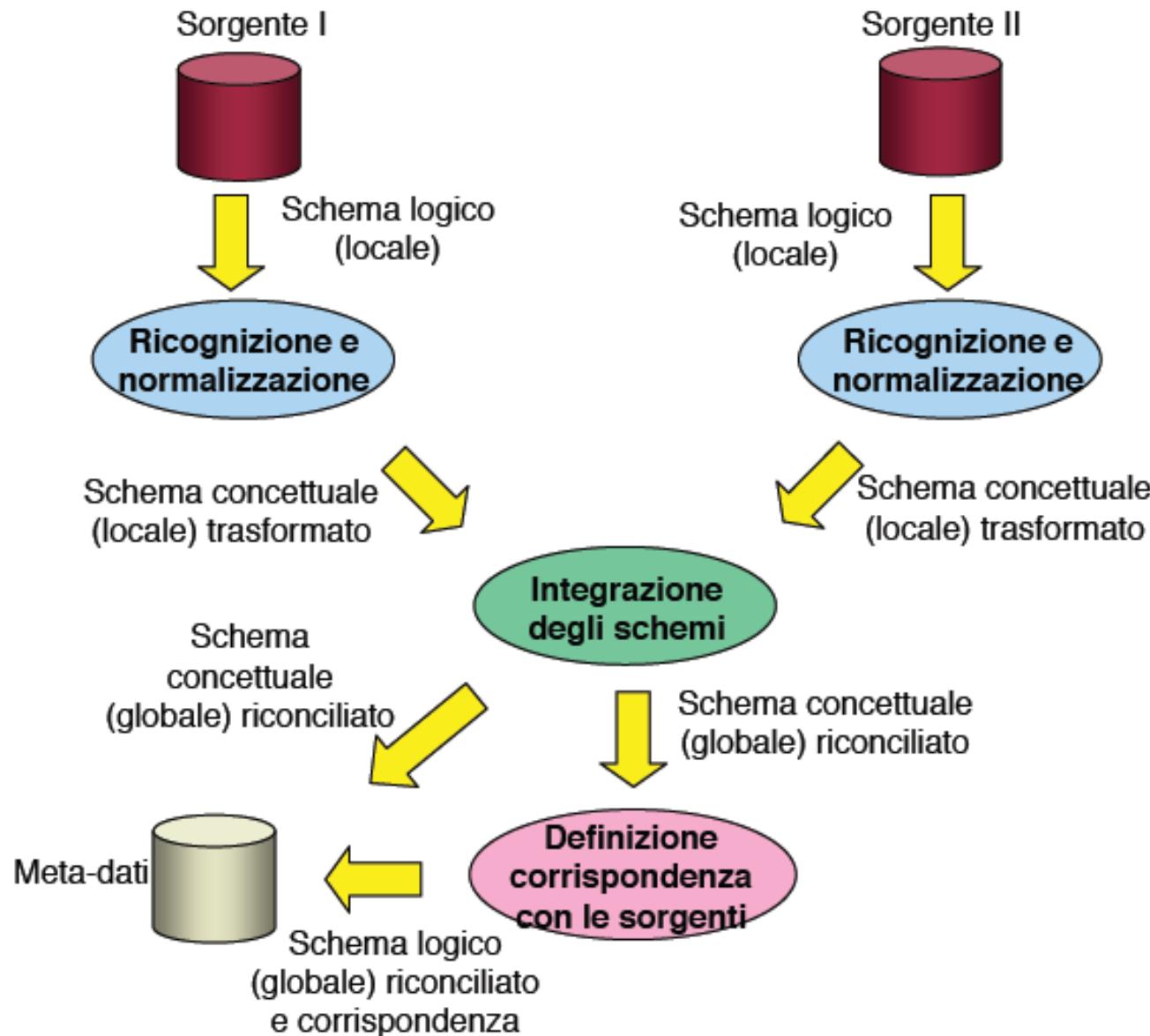
Figure coinvolte

- Oltre al progettista, in questa fase sono coinvolti gli amministratori dei database operazionali, che tipicamente sono gli unici in grado di attribuire un significato a schemi e tracciati record spesso incomprensibili ed inoltre, la loro conoscenza del dominio applicativo è indispensabile per normalizzare gli schemi.

Modello Architetturale

- Il modello architetturale di riferimento è il **modello a tre livelli**, nel quale si suppone che il livello di dati riconciliato esista.
- Tale soluzione viene preferita poiché l'alimentazione diretta del DW è un compito troppo complesso per essere eseguito in modo atomico.

Il Processo di Analisi e Riconciliazione



Descrizione delle Fasi

- La figura precedente dettaglia la fase di analisi e riconciliazione in caso di più sorgenti inconsistenti, di cui è noto il solo schema logico (*magari codificato con formalismi diversi*):
 - **ricognizione** e **normalizzazione** dei diversi schemi locali che produce un insieme di schemi concettuali, localmente consistenti e completi;
 - **integrazione** che produce uno schema concettuale globalmente consistente;
 - dallo schema ottenuto si effettua la progettazione logica dello schema riconciliato, per poi definirne la **corrispondenza** con gli schemi logici delle sorgenti.

Passi Progettuali: “Ricognizione e normalizzazione”

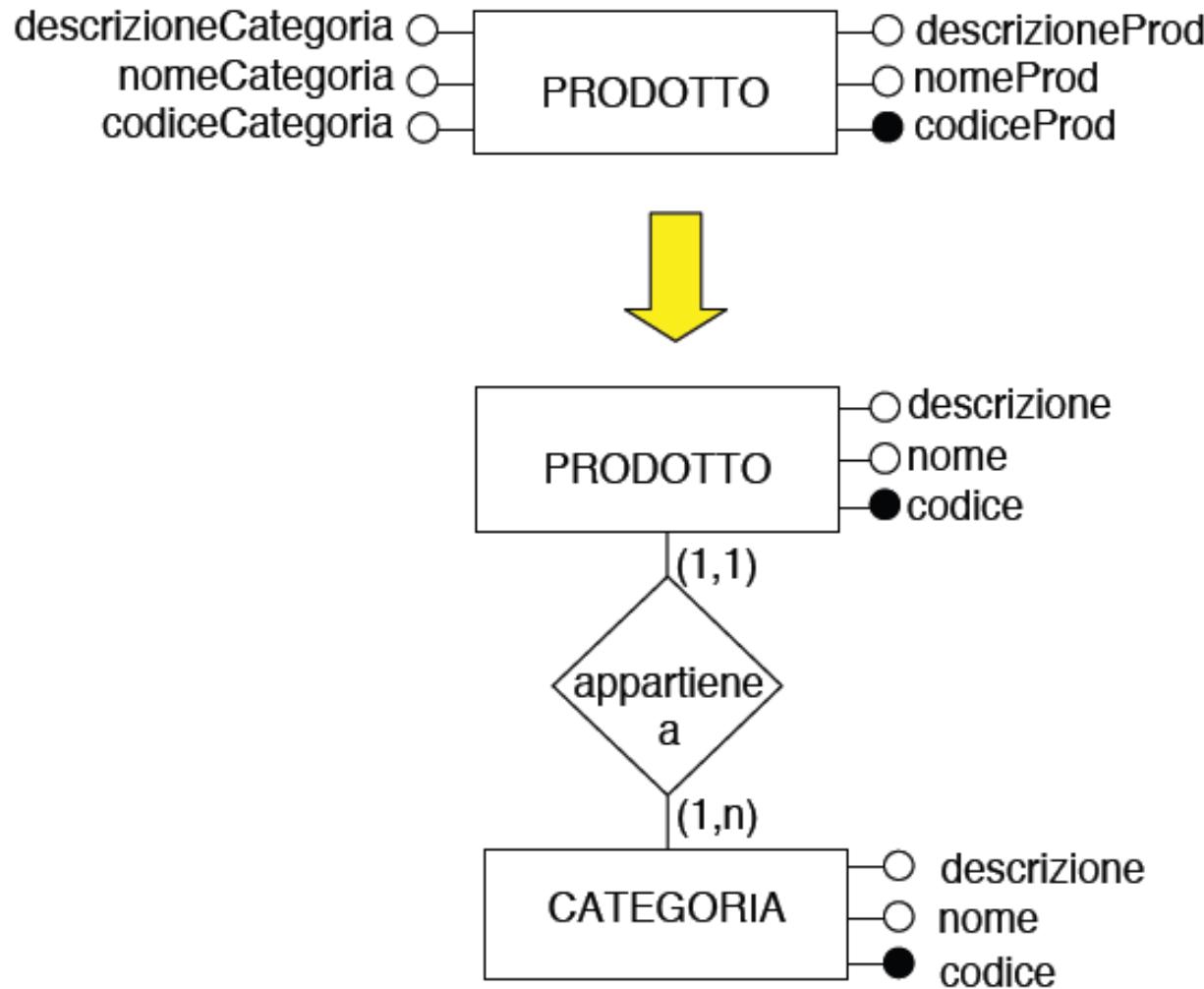
- Prima di procedere alla fase di progettazione concettuale il progettista deve acquisire una approfondita conoscenza delle sorgenti operazionali attraverso attività di:
 - **Ricognizione**, consiste in un esame approfondito degli schemi locali mirato alla piena comprensione del dominio applicativo;
 - **Normalizzazione**, mira a correggere gli schemi locali al fine di modellare in modo più accurato il dominio applicativo.
- Questa fase va svolta anche in presenza di una sola sorgente dati.
 - Nel caso in cui esistano più sorgenti, l'operazione dovrà essere ripetuta per ogni singolo schema locale.

Passi Progettuali:

“Riconoscimento e normalizzazione” (2)

- Durante questa fase di analisi il progettista deve verificare la completezza degli schemi locali sforzandosi di individuare eventuali correlazioni involontariamente omesse.
- Le trasformazioni apportate allo schema non devono introdurre nuovi concetti, bensì rendere esplicati tutti quelli ricavabili dai dati delle sorgenti operazionali.
- Oltre all'applicazione delle necessarie trasformazioni, il progettista deve anche individuare eventuali porzioni degli schemi locali non utili al Data mart.

Esempio di Ricognizione e normalizzazione



Passi Progettuali:

“Fase di Integrazione”

- L'integrazione di sorgenti dati eterogenee (*basi di dati relazionali, file dati, sorgenti legacy*) consiste nella individuazione di corrispondenze tra i concetti degli schemi locali e nella risoluzione dei conflitti evidenziati.
- Lo scopo è di creare un unico schema globale i cui elementi sono correlati con i corrispondenti elementi degli schemi locali (***mapping***).
 - Se le diverse sorgenti dati modellano porzioni distinte del mondo reale, il problema dell'integrazione non esiste.
 - In questa fase vanno anche identificati concetti distinti di schemi differenti che sono correlati attraverso proprietà semantiche (***proprietà inter-schema***).

Passi Progettuali: “Fase di Integrazione” (2)

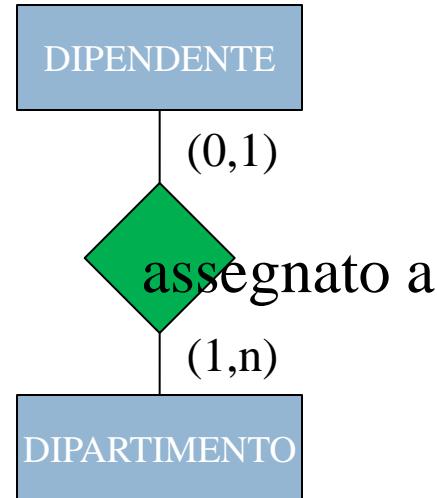
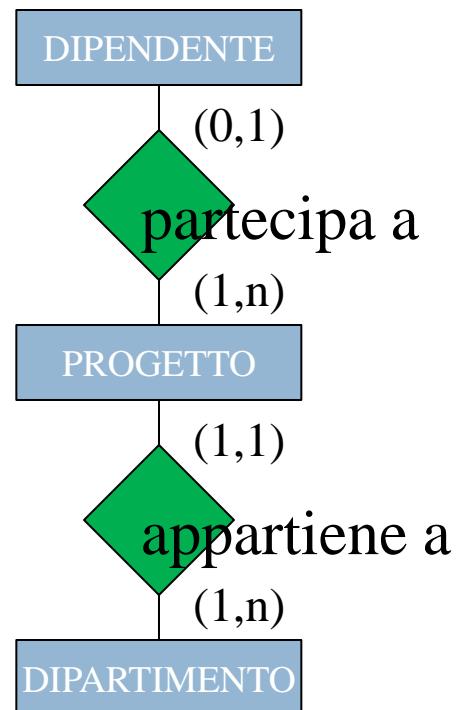
- Per poter ragionare sui concetti degli schemi sorgente è necessario usare un unico formalismo (*ER, UML, relazionale, a oggetti, DTD, ecc.*) in modo da fissare i costrutti utilizzabili e la potenza espressiva.
- Il formalismo prescelto deve garantire il maggior potere espressivo, in modo da evitare perdite di informazioni durante il processo di traduzione.
- Molti progettisti preferiscono erroneamente adottare il formalismo comune alla maggior parte degli schemi, oppure quello in cui si sentono più esperti.

Cause di problemi di integrazione

1. Diversità di prospettiva.
2. Equivalenza dei costrutti del modello.
3. Incompatibilità delle specifiche.
4. Concetti comuni.
5. Concetti correlati.

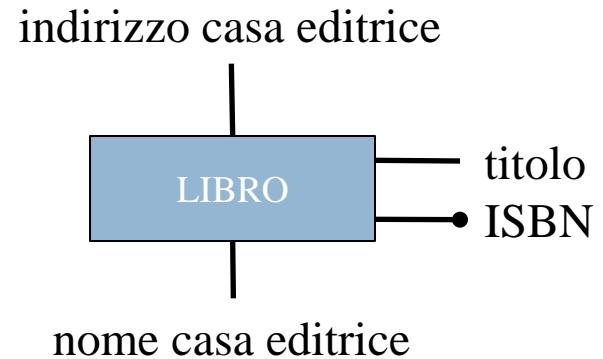
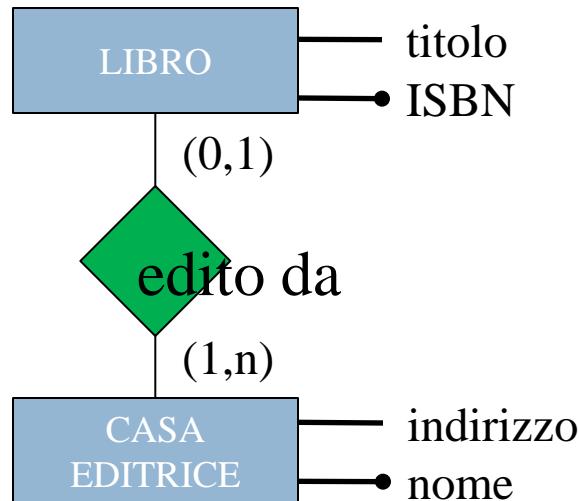
1 - Diversità di prospettiva

- Il punto di vista rispetto al quale diversi gruppi di utenti vedono uno stesso oggetto del dominio applicativo può differenziarsi notevolmente in base agli aspetti rilevanti per la funzione a cui essi sono preposti.



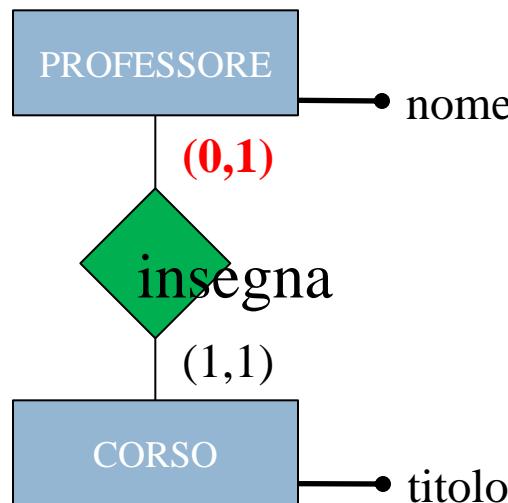
2- Equivalenza dei costrutti del modello

- Tipicamente, i formalismi di modellazione permettono di rappresentare uno stesso concetto utilizzando combinazioni diverse dei costrutti a disposizione.



3 - Incompatibilità delle specifiche

- L'incompatibilità delle specifiche indica che schemi diversi che modellano una stessa porzione del dominio applicativo racchiudono concetti diversi, in contrasto tra loro.
 - Tale diversità deriva da errate scelte progettuali che possono coinvolgere ad esempio la scelta dei nomi, dei tipi di dati e dei vincoli di integrità.
 - **Es:** in un caso un professore non può tenere più di un corso, nell'altro deve tenerne almeno 2.



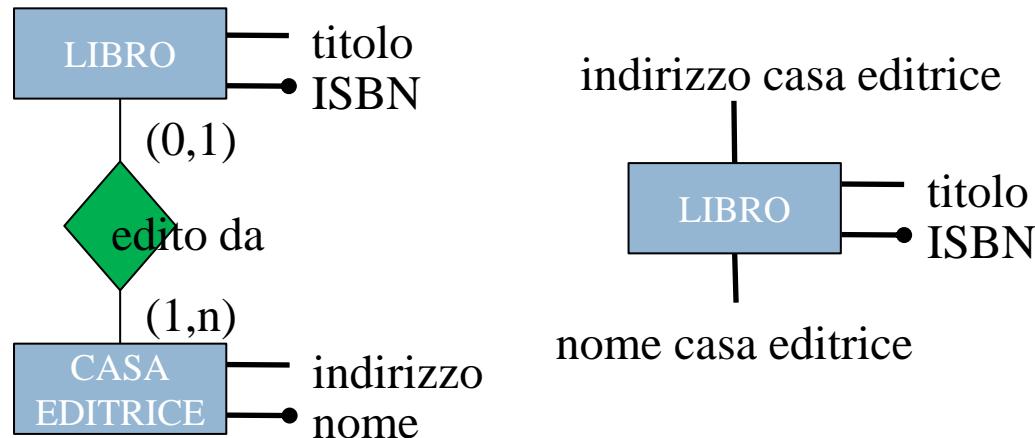
4 - Concetti comuni

- Quattro sono le possibili relazioni esistenti tra due distinte rappresentazioni R_1 e R_2 di uno stesso concetto:
 - Identità
 - Equivalenza
 - Comparabilità
 - Incompatibilità

Identità ed Equivalenza

- **Identità:** si verifica quando vengono utilizzati gli stessi costrutti, il concetto è modellato dallo stesso punto di vista, quindi R_1 e R_2 coincidono.
- **Equivalenza:** si verifica quando R_1 e R_2 non sono le stesse poiché sono stati utilizzati costrutti diversi ma equivalenti.
- Tra le varie definizioni di equivalenza:
 - Due schemi R_1 e R_2 sono equivalenti se le loro istanze possono essere messe in corrispondenza 1-a-1.

Esempio di equivalenza



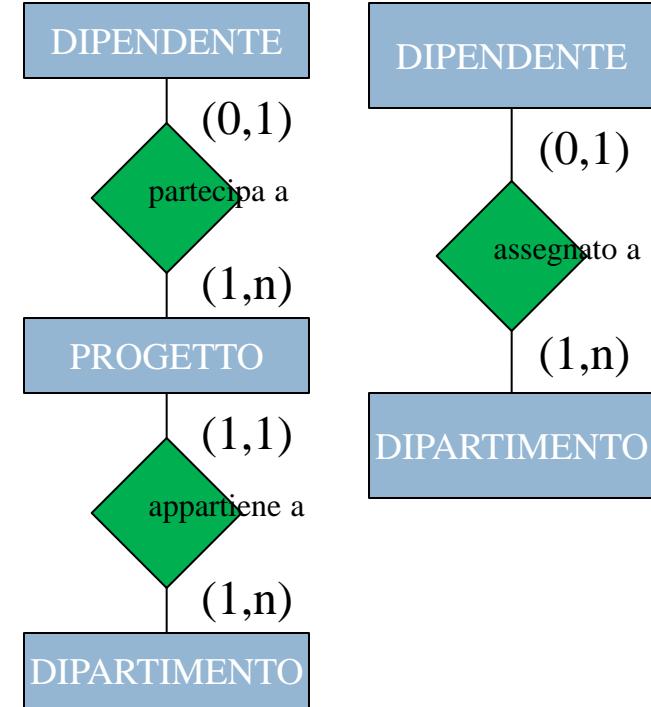
LIBRO		
ISBN	titolo	casa editrice
123445	Il DFM	McGraw-Hill
435454	Mi sembra logico	Apogeo
...

CASA EDITRICE	
nome	indirizzo
McGraw-Hill	Via Ripamonti, 89
Apogeo	Via Verdi, 45
...	...

LIBRO			
ISBN	titolo	nome c.e.	Indirizzo c.e.
123445	Il DFM	McGraw-Hill	Via Ripamonti, 89
435454	Mi sembra logico	Apogeo	Via Verdi, 45
...

Comparabilità e Incompatibilità

□ **Comparabilità:** questa situazione si verifica quando R_1 e R_2 non sono né identici né equivalenti ma, i costrutti utilizzati e i punti di vista dei progettisti non sono in contrasto tra loro.



□ **Incompatibilità:** questa situazione si verifica quando R_1 e R_2 sono in contrasto a causa dell'incoerenza nelle specifiche:

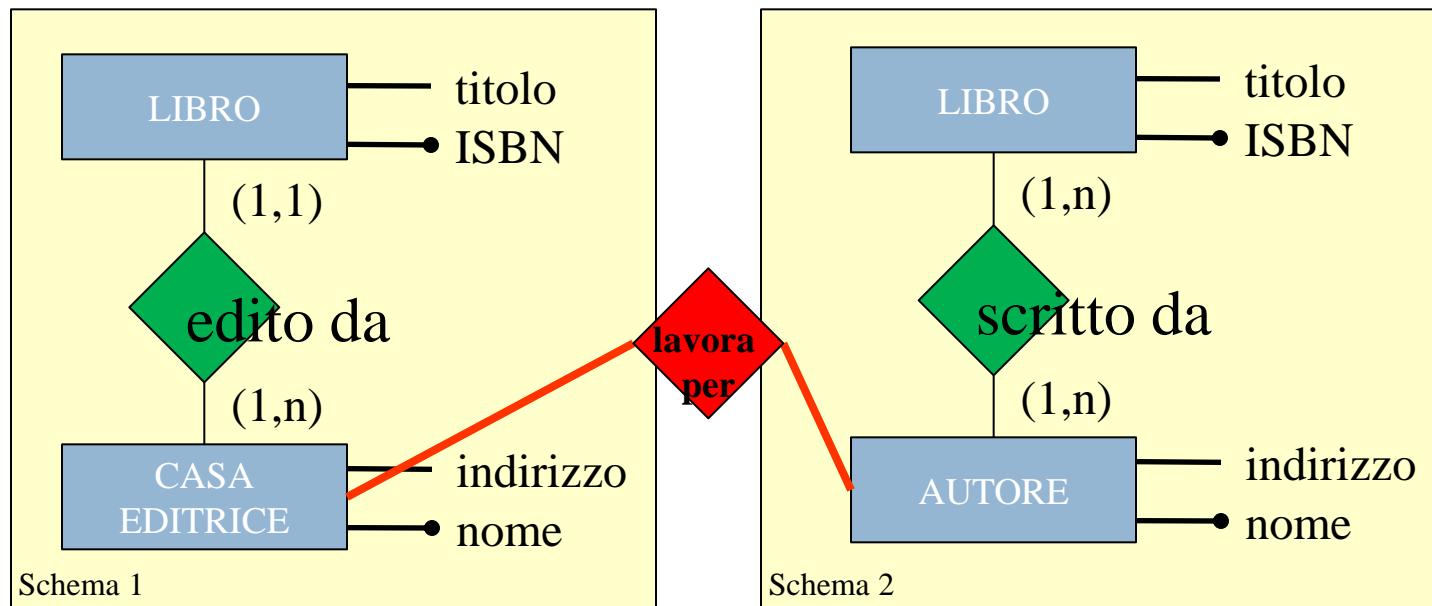
- in altre parole quando la realtà modellata da R_1 nega la realtà modellata da R_2 .

Risoluzione di Conflitti

- Ad esclusione della situazione di *identità*, i casi precedenti determinano dei conflitti la cui soluzione rappresenta la componente principale nella fase di integrazione.
- Pertanto, si verifica un **conflitto** tra due rappresentazioni R_1 e R_2 di uno stesso concetto ogni qualvolta le due rappresentazioni *non sono identiche*.

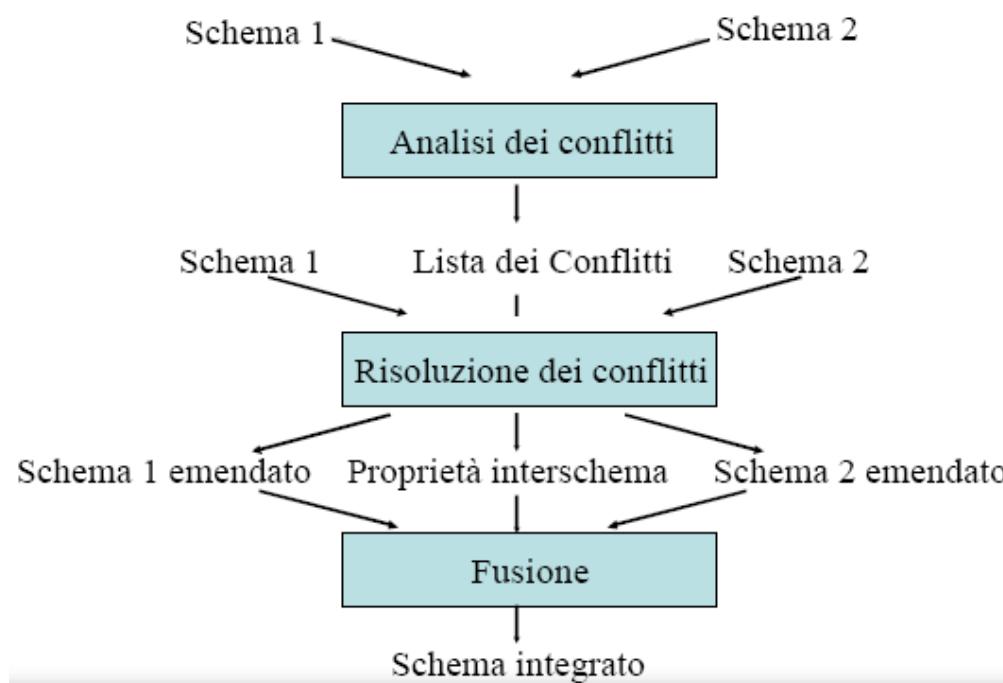
5 - Concetti correlati

- A seguito dell'integrazione, molti concetti diversi, ma correlati, verranno a trovarsi nello stesso schema, dando vita a nuove relazioni che non erano percepibili in precedenza.
 - Tali relazioni sono dette **proprietà inter-schema** e devono essere identificate e rappresentate esplicitamente.



Le fasi dell'integrazione

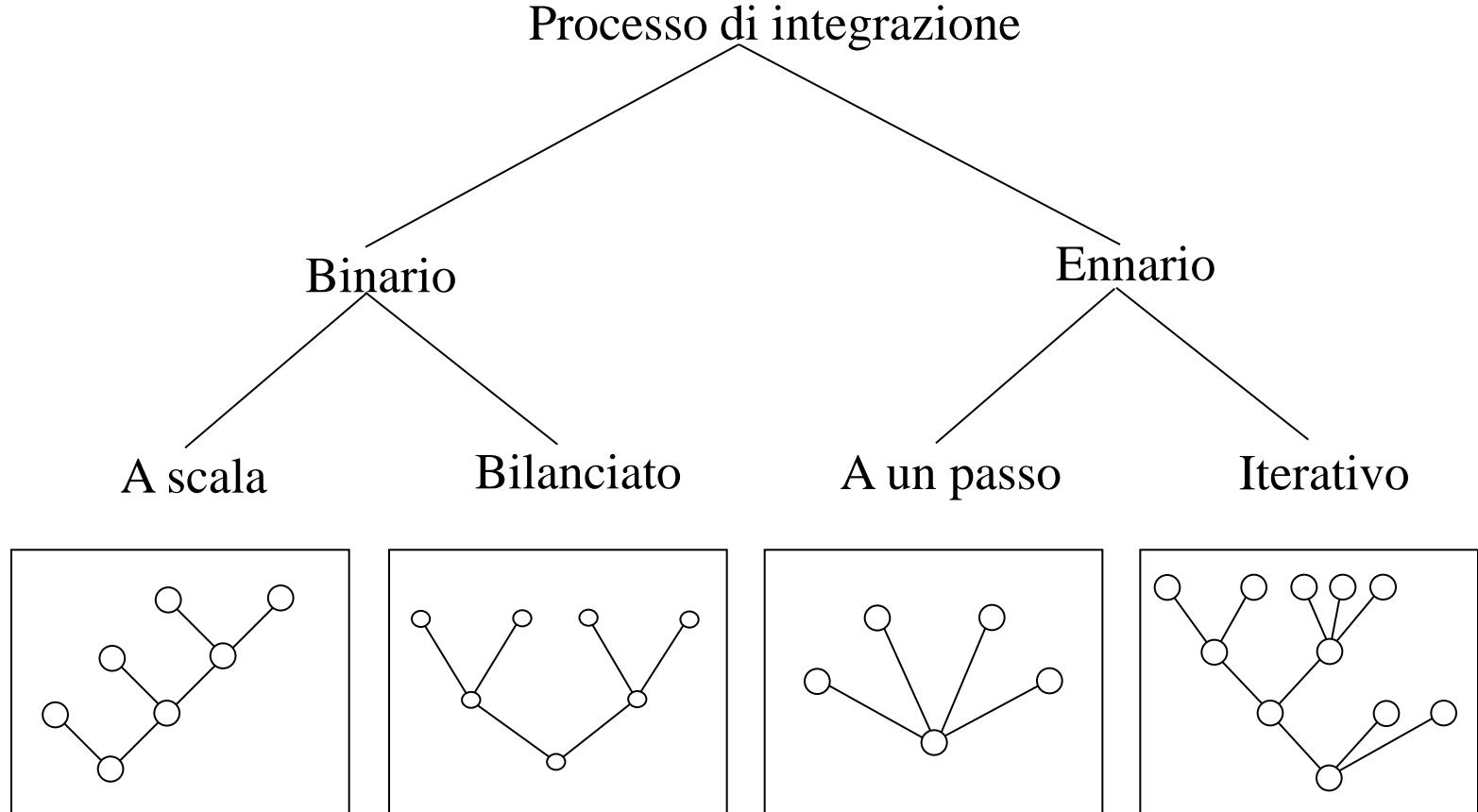
- Per risolvere i problemi fin qui elencati, la sequenza dei passi da svolgere possono essere così sintetizzati:
 1. Preintegrazione
 2. Comparazione degli schemi
 3. Allineamento degli schemi
 4. Fusione e ristrutturazione degli schemi



1 - Preintegrazione

- Durante questa fase viene svolta l'analisi delle sorgenti dati, che porta a definire la politica generale dell'integrazione.
- Le principali decisioni da prendere riguardano:
 - ▣ *Le porzioni degli schemi che dovranno essere integrate:*
 - non tutti i dati operazionali sono utili ai fini decisionali e quindi alcuni di essi potranno essere scartati a priori.
 - ▣ *La strategia di integrazione:*
 - è necessario decidere in che ordine si dovranno integrare gli schemi.

Il Processo di integrazione

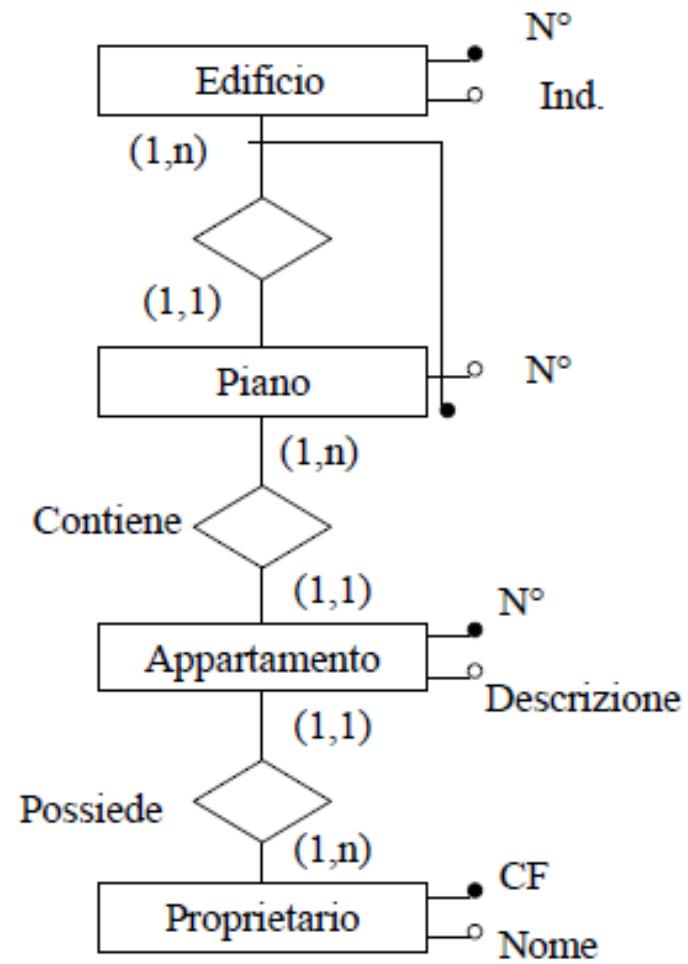
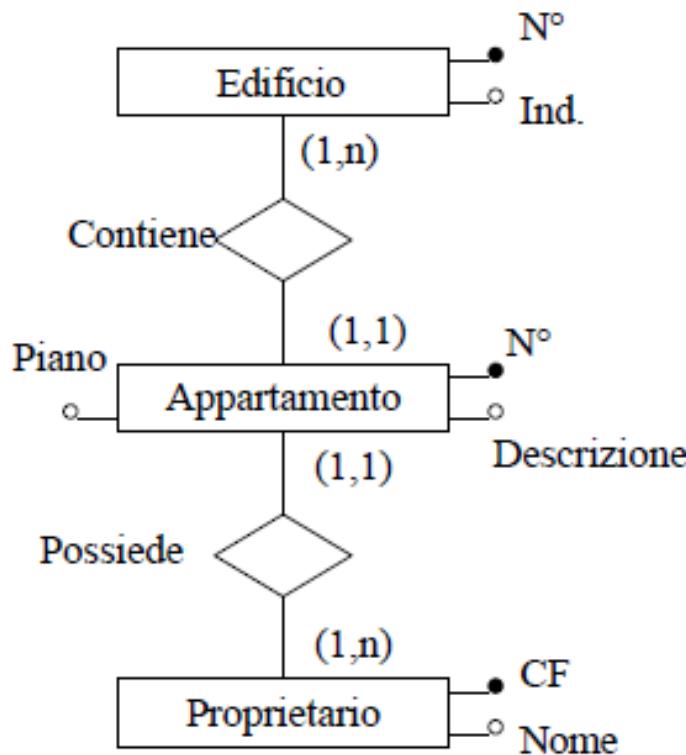


2 - Comparazione degli schemi

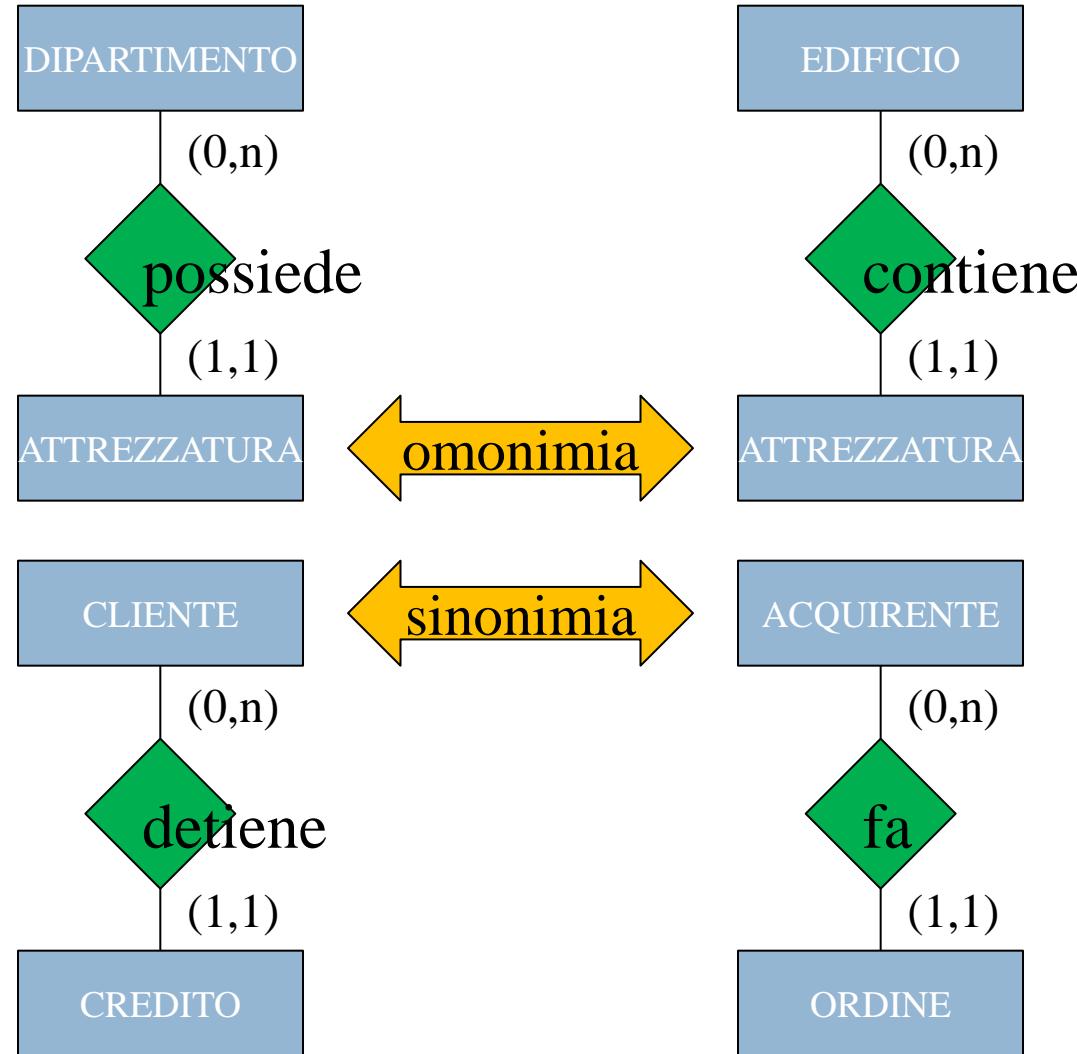
- Questa fase consiste in un'analisi comparativa dei diversi schemi e mira a identificare correlazioni e conflitti tra concetti in essi espressi.
- I tipi di conflitti che possono essere evidenziati ricadono nelle seguenti categorie:
 - ***Conflitti di eterogeneità:***
 - Indicano le discrepanze dovute all'utilizzo di formalismi con diverso potere espressivo negli schemi sorgenti;
 - ***Conflitti semantici:***
 - Si verificano quando due schemi sorgenti modellano la stessa porzione di mondo reale a un diverso livello di astrazione e dettaglio.
 - ***Conflitti sui nomi:***
 - Si verificano a causa delle differenze nelle terminologie utilizzate nei diversi schemi sorgenti (*omonimie* e *sinonimie*).
 - ***Conflitti strutturali:***
 - Sono causati da scelte diverse nella modellazione di uno stesso concetto, oppure dall'applicazione di differenti vincoli di integrità.

Esempio di conflitto semantico

- Si modella la stessa realtà con diverso livello di astrazione:



Esempio di sinonimie ed omonimie

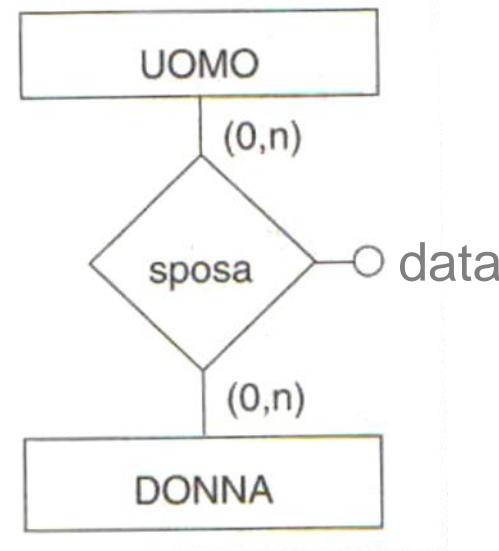
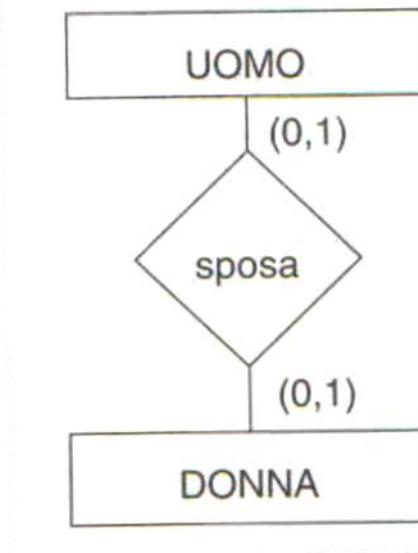


Conflitti strutturali

- In particolare, i conflitti strutturali possono essere:
 - **Conflitti di tipo:** si verificano quando uno stesso concetto è modellato utilizzando due costrutti diversi.
 - **Conflitti di dipendenza:** si verificano quando due o più concetti sono correlati con dipendenze diverse in schemi diversi.
 - **Conflitti di chiave:** si verificano quando per uno stesso concetto vengono utilizzati identificatori diversi in schemi diversi.
 - **Conflitti di comportamento:** si verificano quando diverse politiche di cancellazione/modifica dei dati vengono adottate per uno stesso concetto in schemi diversi.

Esempio

□ Conflitto di dipendenza:



- **Es:** Due diversi schemi per modellare il matrimonio:
 - Lo schema a destra permette la storicizzazione delle informazioni.

3 - Allineamento degli schemi

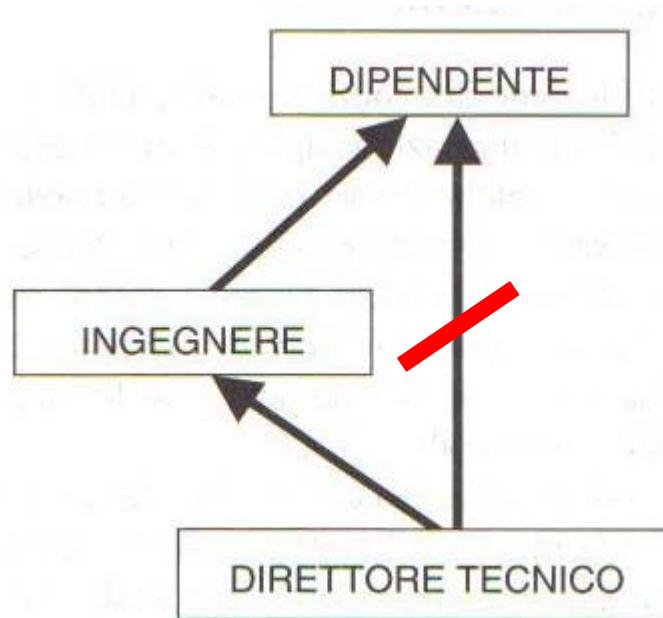
- Scopo di questa fase è la risoluzione dei conflitti evidenziati al passo precedente, mediante primitive di trasformazione dei schemi sorgenti o dello schema riconciliato temporaneo.
- Tipiche primitive di trasformazione riguardano il cambio dei nomi e dei tipi degli attributi, la modifica delle dipendenze funzionali e dei vincoli sugli schemi.
 - Non sempre però i conflitti sono risolvibili, in tal caso la soluzione deve essere discussa con gli utenti.
- Il progettista deve definire il mapping tra gli elementi degli schemi sorgenti e quelli dello schema riconciliato.

4 - Fusione e ristrutturazione schemi

- In quest'ultima fase gli schemi allineati vengono fusi per formare un unico schema riconciliato:
 - ▣ L'approccio più diffuso è quello di sovrapporre i concetti comuni a cui saranno collegati i rimanenti concetti degli schemi locali.
- Dopo questa operazione ulteriori trasformazioni permetteranno di migliorare la struttura dello schema riconciliato rispetto a:
 - ▣ **Leggibilità:** il miglioramento della leggibilità dello schema facilita e velocizza le successive fasi di progettazione.
 - ▣ **Completezza:** il progettista deve esaminare lo schema fin qui costruito alla ricerca di proprietà inter-schema non evidenziate in precedenza.
 - ▣ **Minimalità:** occorre eliminare ridondanza di concetti duplicati o comunque derivabili gli uni dagli altri, oltre alle relazioni cicliche tra concetti ed attributi derivati.

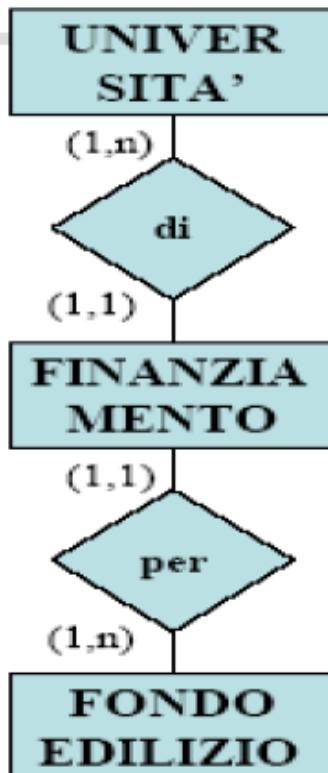
Esempio di minimalità

- Schema con relazione di inclusione ridondante:

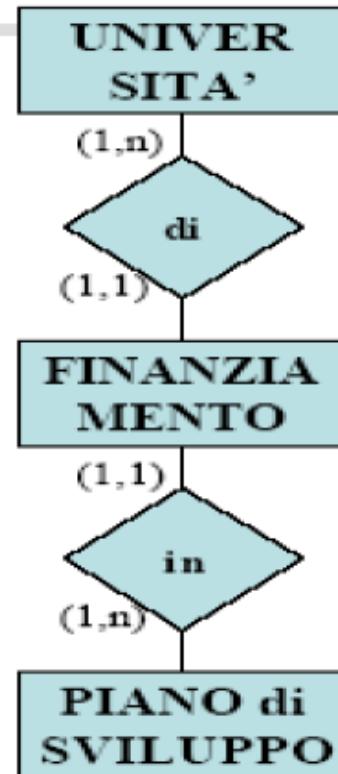


- La generalizzazione tra DIRETTORE TECNICO e DIPENDENTE è ridondante e può essere eliminata.

Esempio: Schemi compatibili

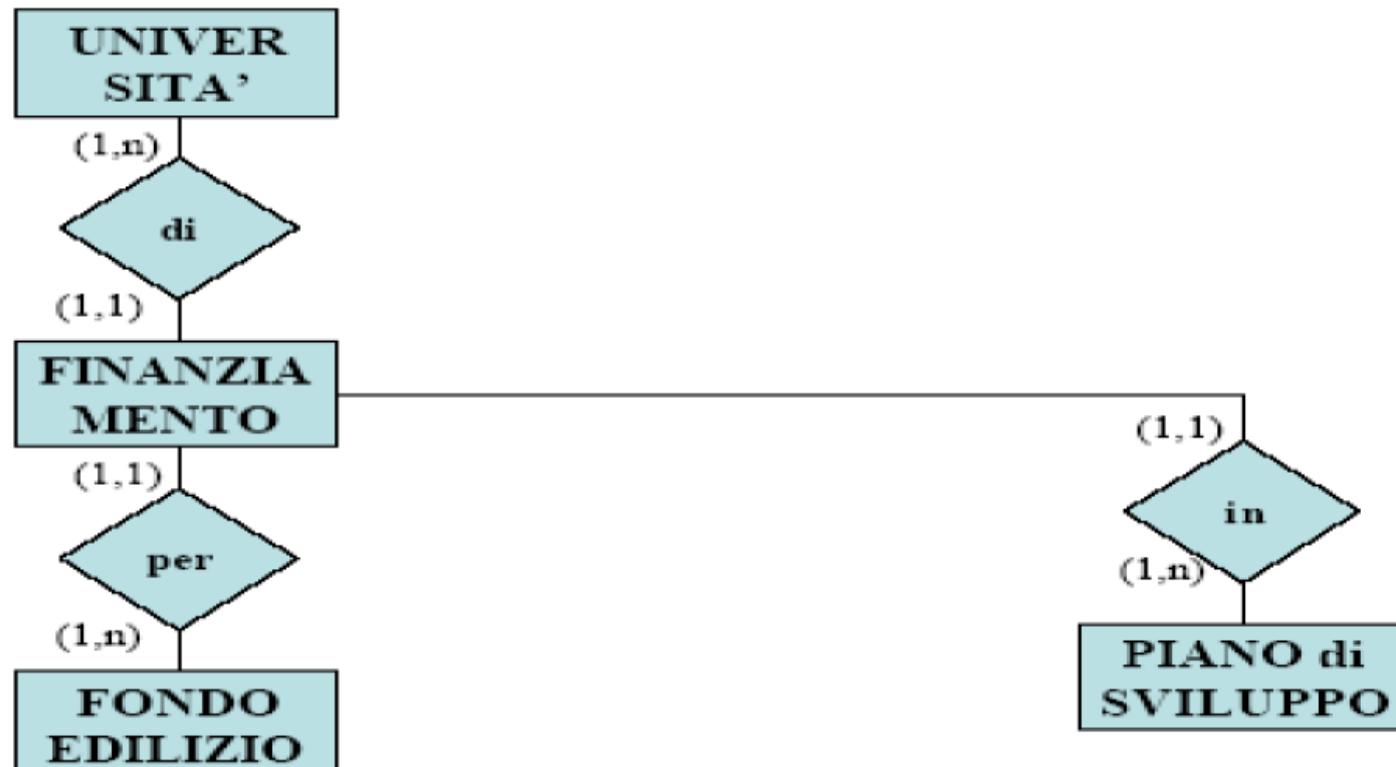


Schema 1: stanziamento fondi
edilizia universitaria

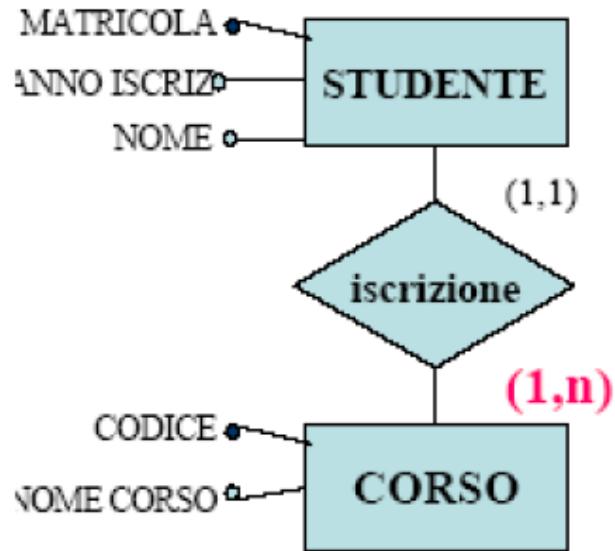


Schema 2: struttura
delle università 49

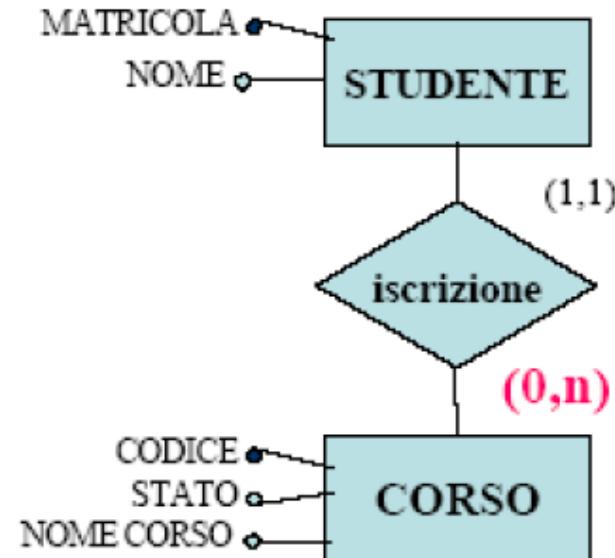
Esempio: Schemi compatibili (integrazione)



Esempio 2: Schemi incompatibili (cardinalità incompatibili)



Schema 1: archivio studenti

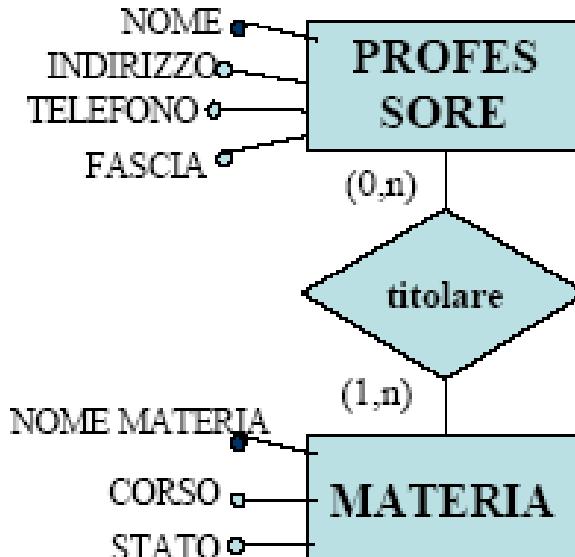


Schema 2: archivio corsi

Nello schema 1 sono archiviati tutti gli studenti iscritti ad un corso universitario, mentre lo schema 2 include tutti i corsi attivati e quindi anche quelli a cui non è iscritto alcuno studente.

Nello schema integrato si sceglie la seconda soluzione, perché meno restrittiva.

Esempio 3: Schemi incompatibili (rappresentazione con strutture differenti)



Schema 1: Archivio
dei professori



Schema 2: Archivio delle
Materie di insegnamento

Nello schema 1 sono archiviati tutti i professori che possono essere o meno titolari di materie di insegnamento, mentre nello schema 2 sono memorizzati tutte le materie.
Nello schema integrato si sceglie la prima soluzione, perché più completa e meno restrittiva.

Passi progettuali:

“Definizione delle corrispondenze” (1)

- Il risultato dell’analisi delle sorgenti operazionali è composto da due elementi:
 - ▣ Lo schema riconciliato, in cui sono stati risolti i conflitti presenti tra gli schemi locali;
 - ▣ L’insieme di corrispondenze tra gli elementi presenti negli schemi sorgenti e quelli dello schema destinazione.
 - Esse sono necessarie per la fase di progettazione degli strumenti ETL per migrare i dati dalle sorgenti al livello riconciliato.

Passi progettuali:

“Definizione delle corrispondenze” (2)

- L'approccio per stabilire la corrispondenza tra i due livelli dell'architettura prevede che lo schema globale sia espresso in termini degli schemi sorgente detto GAV (**Global–As–View**).
 - Ad ogni concetto dello schema globale è associata una **vista** definita in base a concetti degli schemi sorgenti.
 - Con GAV sarà sufficiente sostituire ad ogni concetto dello schema globale la vista che lo definisce in termini di concetti degli schemi locali (**unfolding**).

Esempio: Mapping di tipo GAV

// DB1 Magazzino

ORDINI2001(chiaveO, chiaveC, data ordine, impiegato)
CLIENTE(chiaveC, nome, indirizzo, città, regione, stato)

.....

// DB2 Amministrazione

CLIENTE(chiaveC, partitalva, nome, telefono, fatturato)
FATTURE(chiaveF, data, chiaveC, importo, iva)
STORICO_ORDINI2000(chiaveO, chiaveC, data ordine, impiegato)

.....

CREATE VIEW CLIENTE AS

SELECT CL1.chiaveC, CL1.nome, CL1.indirizzo, CL1.città, CL1.regione,
CL1.stato, CL2.partitalva, CL2.telefono, CL2.fatturato
FROM DB1.CLIENTE AS CL1, DB2.CLIENTE AS CL2
WHERE CL1.chiaveC = CL2.chiaveC;

CREATE VIEW ORDINI AS

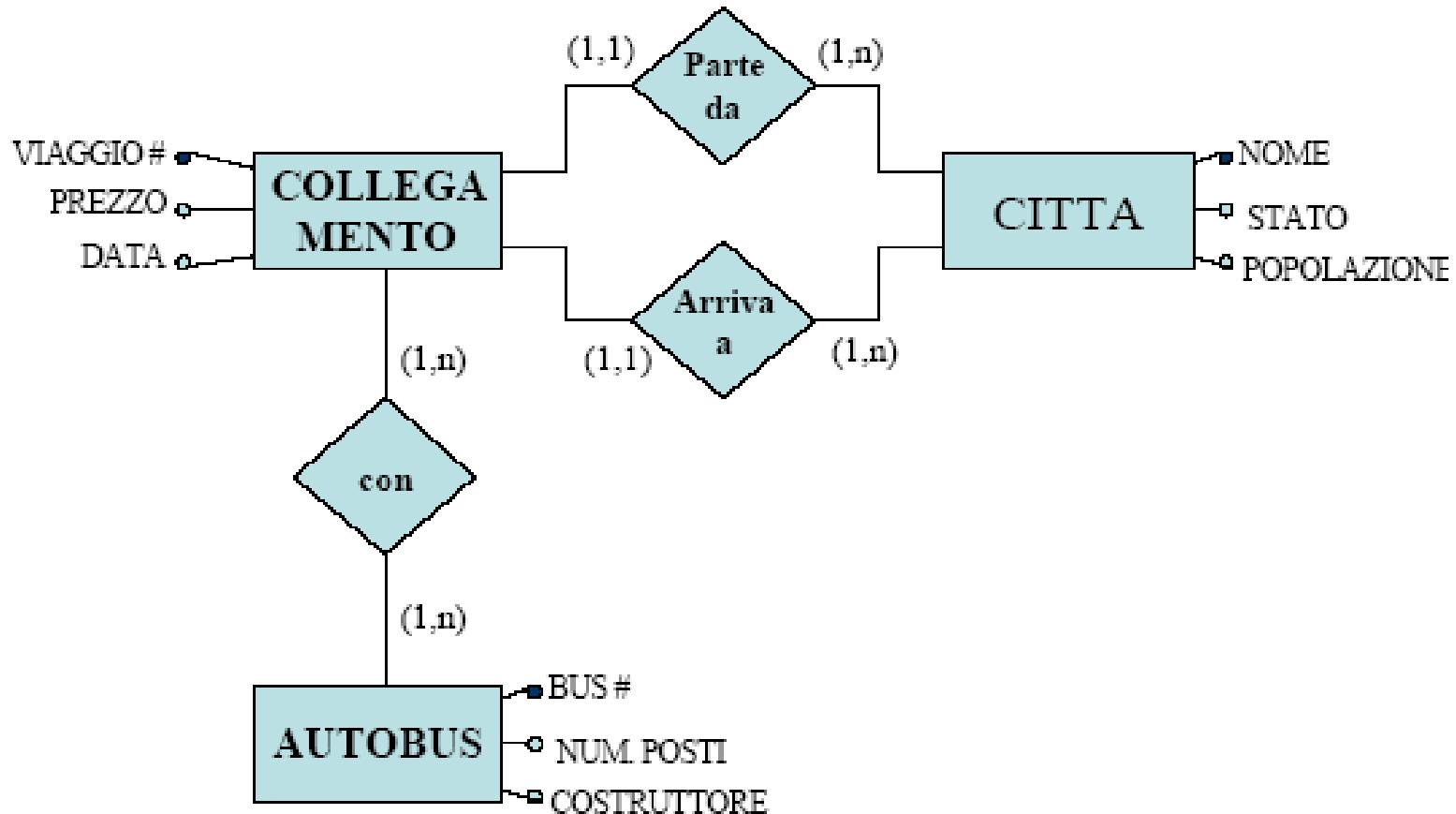
SELECT * FROM DB1.ORDINI2001
UNION
SELECT * FROM DB2.STORICO_ORDINI2000;

Esercizio: Compagnia di viaggi

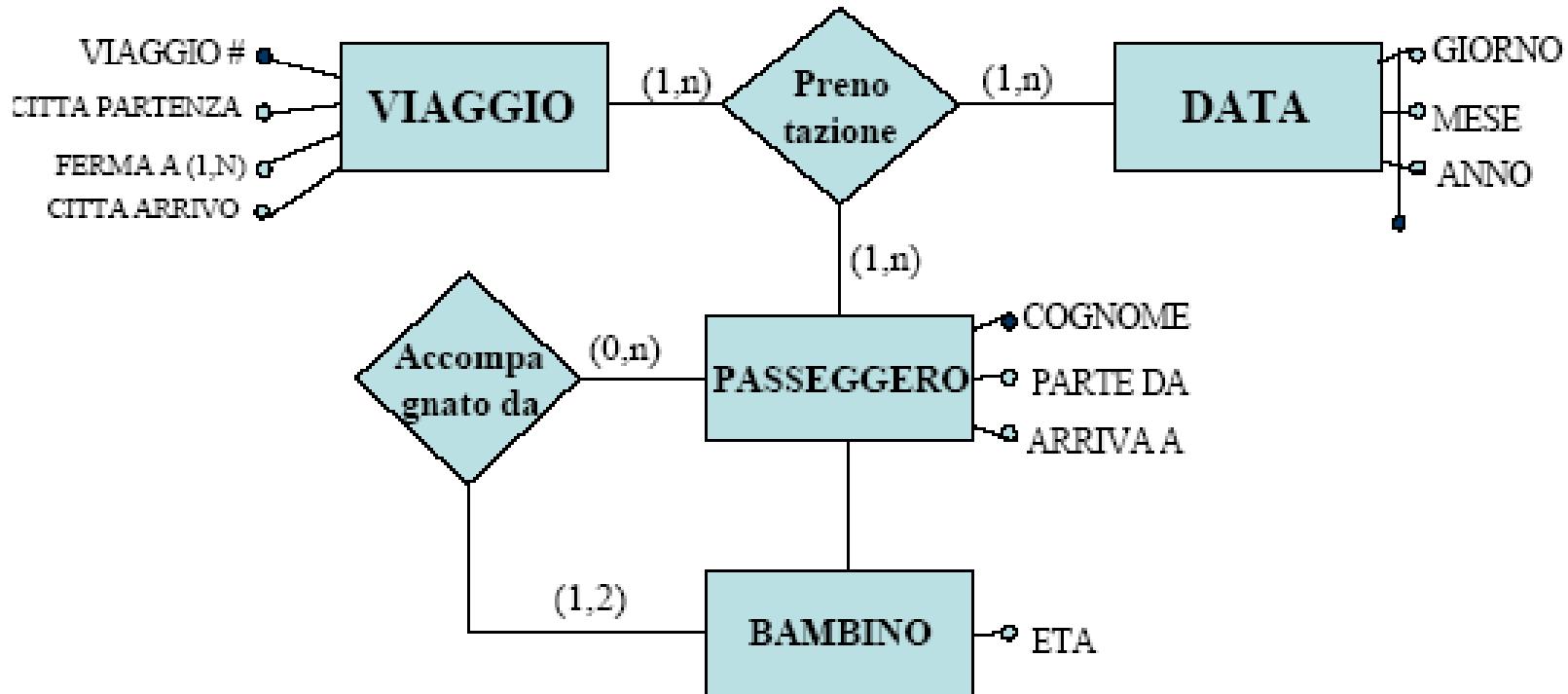
Integrare tre schemi:

- **Collegamenti:** descrive i collegamenti tra le diverse città e il bus che effettua il viaggio
- **Prenotazioni:** descrive le prenotazioni ai viaggi organizzati dalla compagnia
- **Viaggi quotidiani:** descrive l'utilizzo giornaliero degli autobus: costruttore, percorso, pilota.

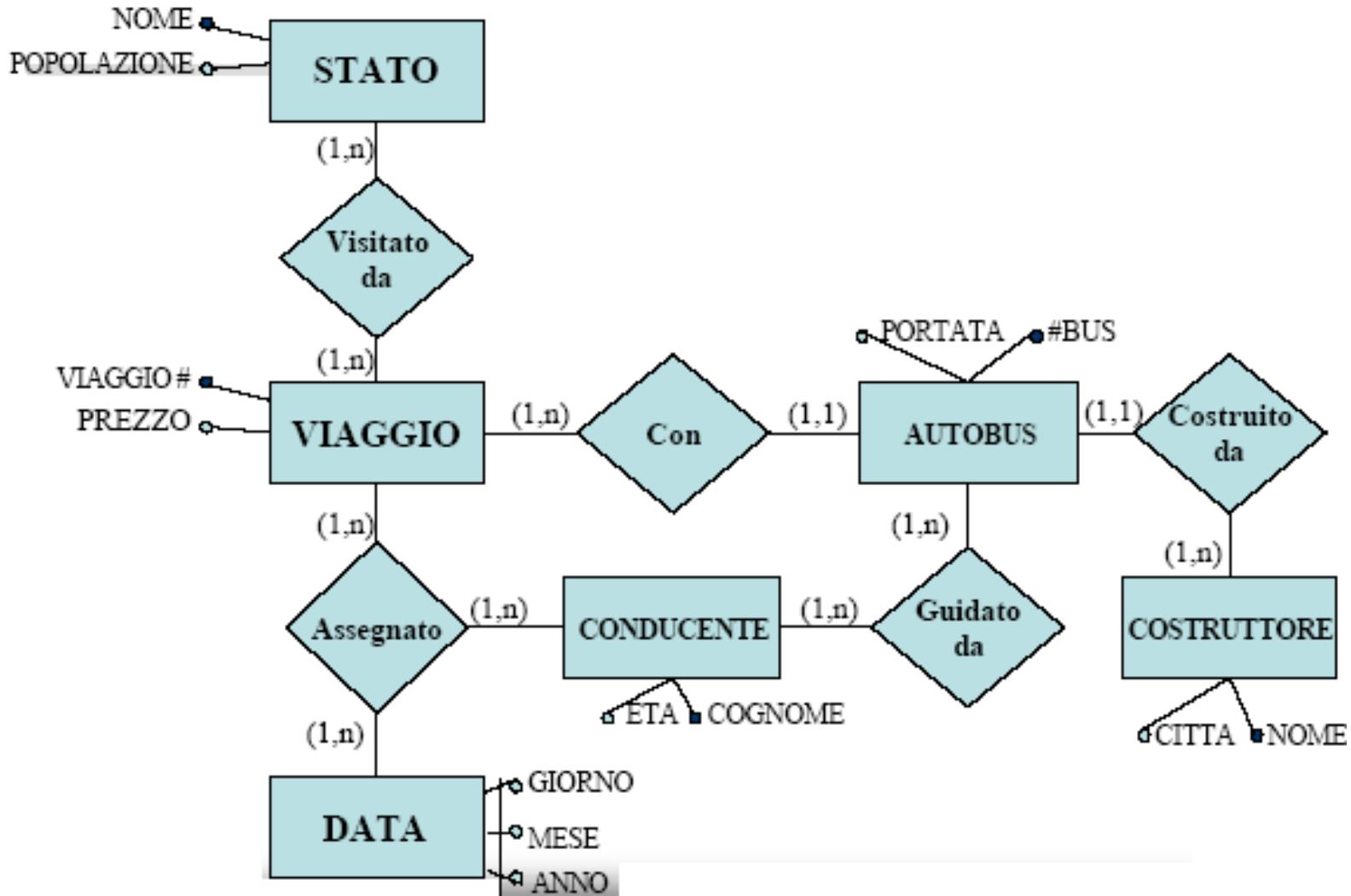
Esercizio: Schema 1 - Collegamenti



Esercizio: Schema 2 - Prenotazioni



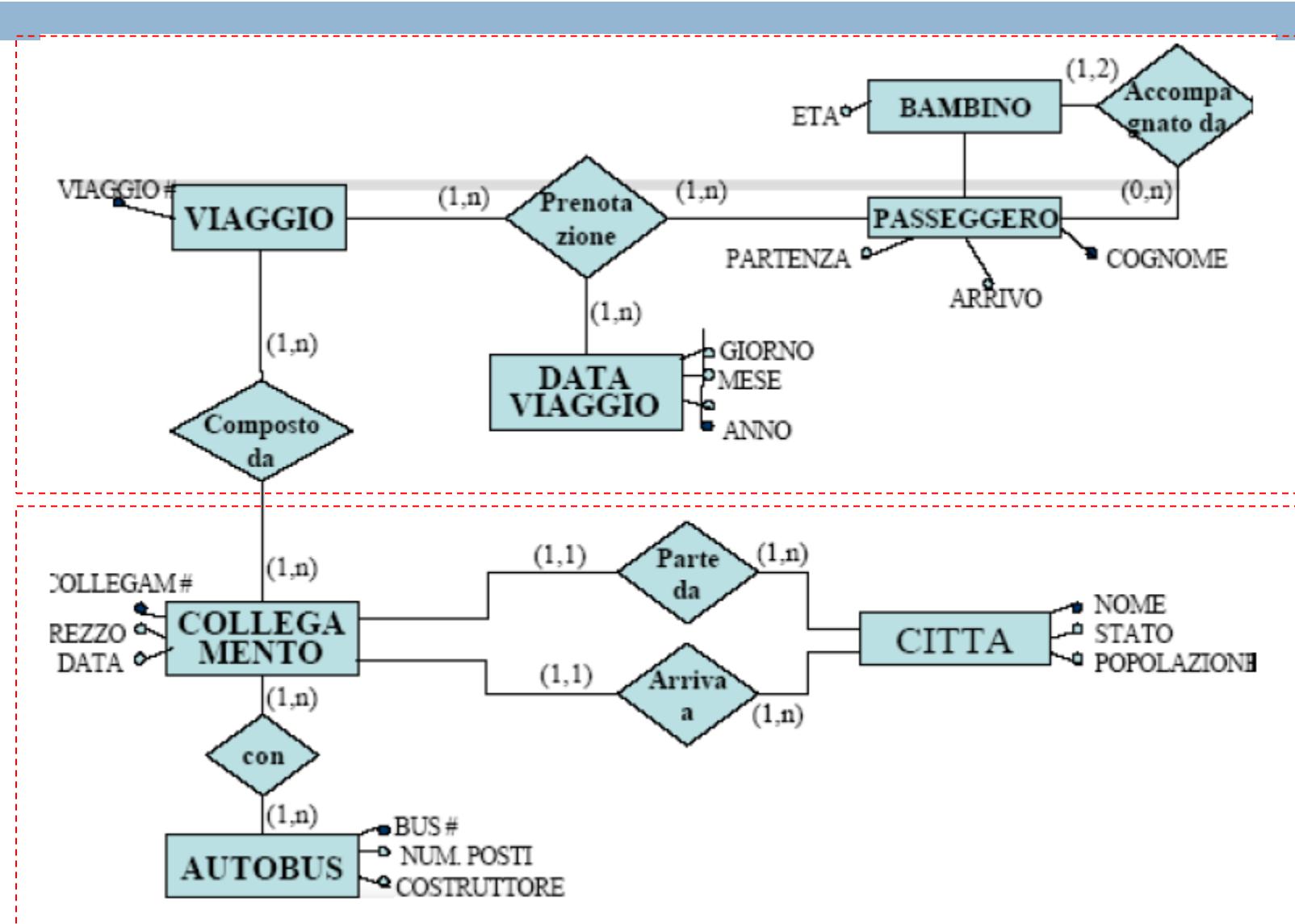
Esercizio: Schema 3 – Viaggi quotidiani



Integrazione Collegamenti-Prenotazione

- Conflitti di nome:
 - VIAGGIO#: nel primo schema l'identificatore diventa COLLEGAM#
 - PARTE DA e ARRIVA A: nel primo schema relazione tra COLLEGAMENTO e CITTA; nel secondo attributi che specificano partenza e arrivo del passeggero.
 - La DATA del COLLEGAMENTO non è necessariamente quella del viaggio;
- Conflitti di struttura:
 - CITTA: nel secondo schema le città di partenza, arrivo e quelle intermedie sono rappresentate da attributi, nel primo c'è l'entità CITTA in doppia relazione con COLLEGAMENTO.
- Proprietà interschema:
 - VIAGGIO è in relazione con COLLEGAMENTO.
- Eliminazione delle ridondanze:
 - Il concetto di CITTA è espresso completamente in relazione con COLLEGAMENTO;

Integrazione Collegamenti-Prenotazione (2)



Fusione schema integrato Collegamenti-Prenotazioni con schema Viaggi quotidiani

- **Conflitti di nome:**
 - VIAGGIO: nello schema Viaggi quotidiani rappresenta il concetto di COLLEGAMENTO.
 - POPOLAZIONE: in uno schema rappresenta il numero di abitanti di una CITTA, nell'altro quello di uno STATO.
 - L'attributo CITTA dell'entità COSTRUTTORE rappresenta la sede e non il concetto di CITTA di partenza e arrivo COLLEGAMENTO;
 - Il numero di posti in un AUTOBUS è espresso in uno schema con l'attributo PORTATA e nell'altro con NUM.POSTI.
 - DATA: in Collegamenti-Prenotazioni è la data del VIAGGIO, in Viaggi quotidiani è il giorno in cui un CONDUCENTE guida un AUTOBUS per un COLLEGAMENTO.

Fusione schema integrato Collegamenti-Prenotazioni con schema Viaggi quotidiani (2)

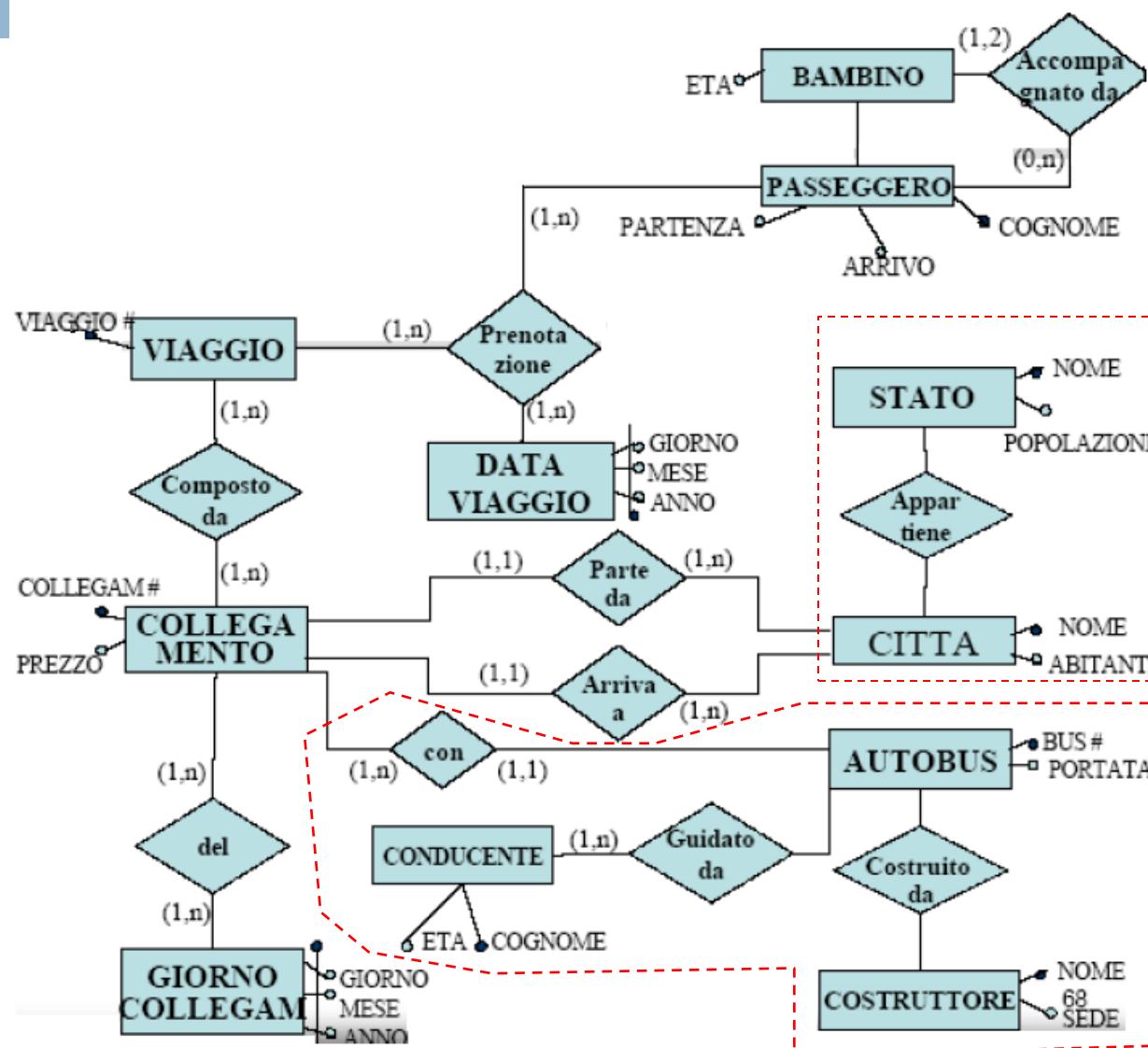
- **Conflitti di struttura:**

- COSTRUTTORE: attributo in Collegamenti-Prenotazioni e entità in Viaggi quotidiani.
- In Viaggi quotidiani si assume che un AUTOBUS possa effettuare solo un COLLEGAMENTO (card.Rel. (1,n)), mentre in Collegamenti-Prenotazioni la cardinalità della relazione è (n,n).
- In Viaggi quotidiani la DATA è in relazione (n,n) con COLLEGAMENTO mentre nell'altro schema è un attributo singolo.

- **Eliminazione delle ridondanze:**

- La relazione Assegnato tra VIAGGIO e CONDUCENTE è ridondante.

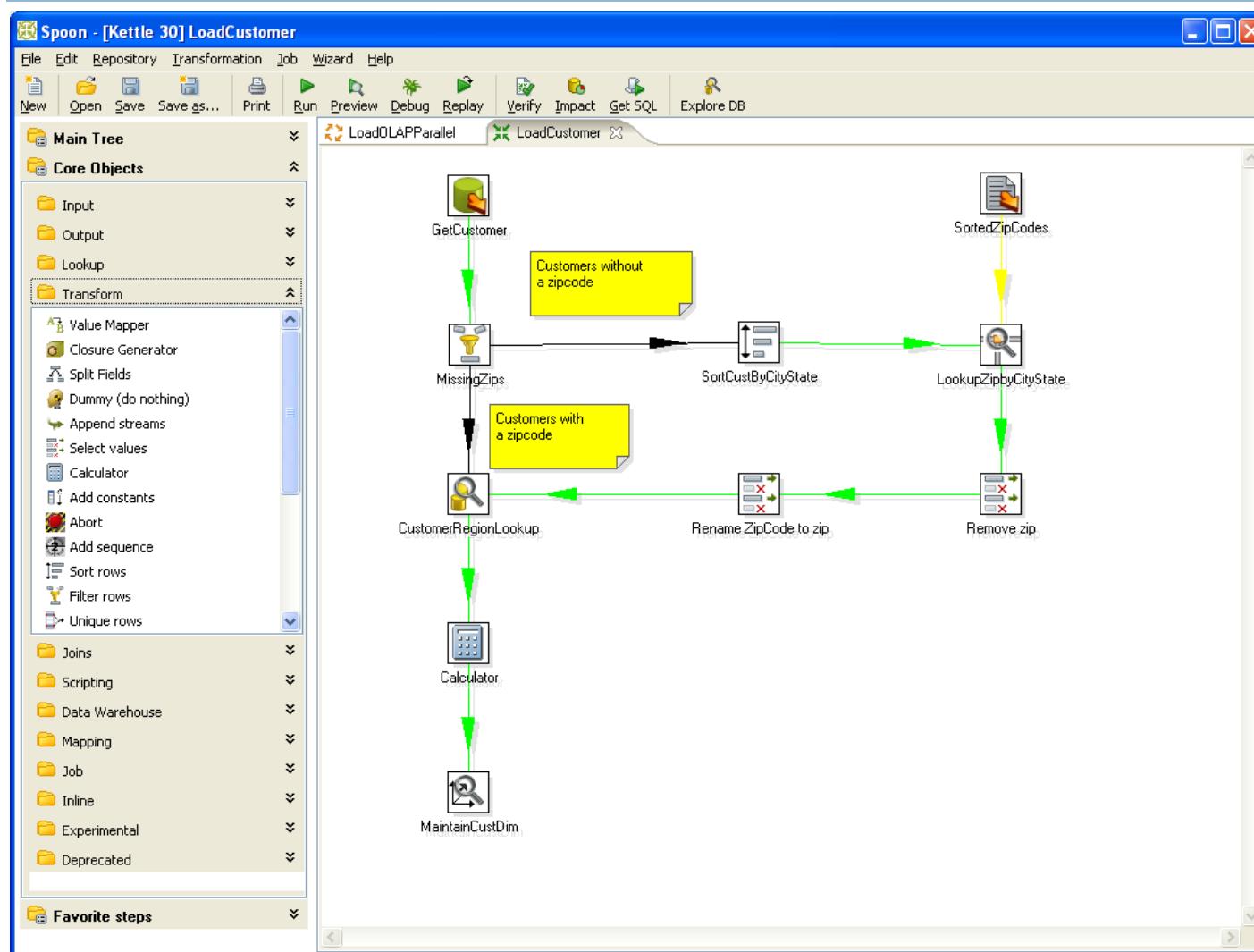
Fusione schema integrato Collegamenti-Prenotazioni con schema Viaggi quotidiani (3)



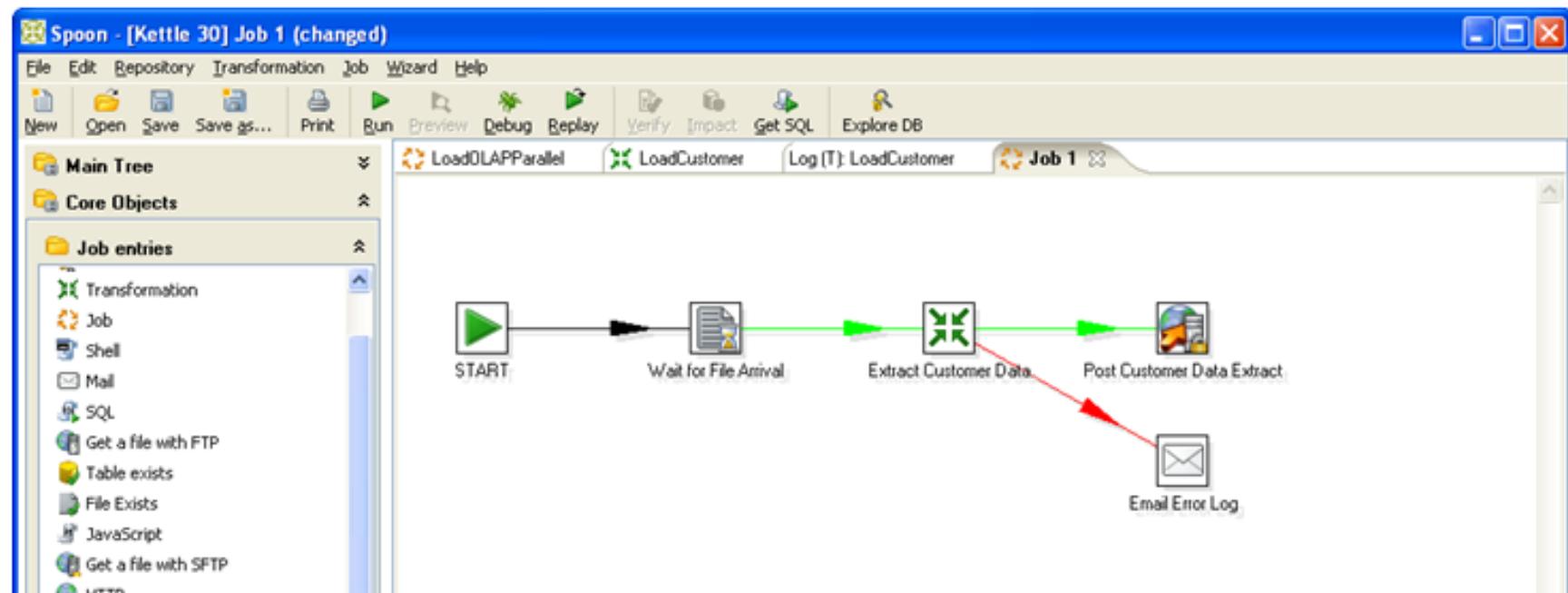
Strumenti ETL

- Una delle caratteristiche più interessanti degli strumenti ETL di ultima generazione è data dall'utilizzo di interfacce grafiche che consentono letteralmente di disegnare i flussi di trasformazione e caricamento dei dati.
 - Questo consente un'immediata comprensione di come i dati, da sorgente migrano verso una destinazione, potendo individuare le trasformazioni che subiscono.
- Tali interfacce sono corredate da icone grafiche e indicatori di trasformazione che rendono immediatamente evidente il tipo di evoluzione che il dato sta subendo nel migrare dallo strato transazionale verso lo strato OLAP.
- Rispetto allo sviluppo di procedure/codice vengono superati alcuni limiti, permettendo:
 - Previsione dell'impatto di una specifica modifica nella fonte alimentante eseguendo specifici test.
 - Generazione automatica del codice degli strumenti di ETL con performance nettamente superiori rispetto al codice generato da uno sviluppatore.

Esempio 1 (Kettle)



Esempio 2 (Kettle)



Esempio 3 (Talend Open Studio)

The screenshot shows the Talend Open Studio interface with the following components:

- Job POrders 0.1**: The main job definition.
- US_States**: A lookup component (orange arrow) that feeds into the tMap_1 component.
- Customers**: An input component (green arrow) that feeds into the tMap_1 component.
- tMap_1**: A mapping component that performs three operations:
 - New_Customers (Main order:1)**: An output stream to the **Target_Customers_Table**.
 - Rejected_Customers (Main order:2)**: An output stream to the **Target_Virginia_Customers_Table**.
 - row1 (Main)**: An output stream to the **tfileOutputExcel_1** component.
- Target_Customers_Table**: A MySQL output component (blue cylinder).
- Target_Virginia_Customers_Table**: A MySQL output component (blue cylinder).
- tfileOutputExcel_1**: An Excel output component (blue cylinder).

Outline View (Bottom Left):

```
/**  
 * [tMysqlOutput_1 main ] =  
 */  
  
currentComponent="tMysqlOut";  
  
whetherReject_tMysqlOutput_  
false;  
  
if(New_Customers.idcustomer  
null) {
```

Component Properties (Bottom Right):

Basic settings

- Property Type: Repository, DB (MySQL):demoMysql
- DB Version: Mysql 5
- Use an existing connection:
- Host: localhost
- Database: demoproject
- Username: root
- Password: ****
- Table: customer
- Action on table: Default
- Action on data: Insert
- Schema: Repository, DB (MySQL):demoMysql - customer
- Die on error:

Palette (Right Side):

- Business Intelligence
- Business
- Custom Code
- Data Quality
- Databases
- ELT
- File
 - Input
 - Management
 - Output
- Internet
- Logs & Errors
- Misc
- Orchestration
- Processing
 - Fields
 - tAggregateRow
 - tAggregateSortedRow
 - tConvertType
 - tExternalSortRow
 - tFilterColumns
 - tFilterRow
 - tJoin
 - tMap
 - tReplace
 - tSampleRow
 - tSortRow
- System
- Talend MDM
- XML

Esempio 4 (Talend Open Studio)

Talend Open Studio |

File Edit Diagram Window Help

Repository Navigator Job POrders 0.1 *Model aBusinessModel

A Business Model is a non technical view of a business need in data flow management. The Business Modeler is at the core of the Top/Down approach: it allows any of the key players to take part to the project design. BMs offer a macroscopic view of the project.

Select a shape, go on the view assignment and open the associated items (double click)

Palette

- business
 - Decision
 - Action
 - Terminal
 - Data
 - Document
 - Input
 - List
 - Datasource
 - Actor
 - Ellipse
 - Gear
- Relationship
 - Directional Relationship
 - Bidirectional Relationship

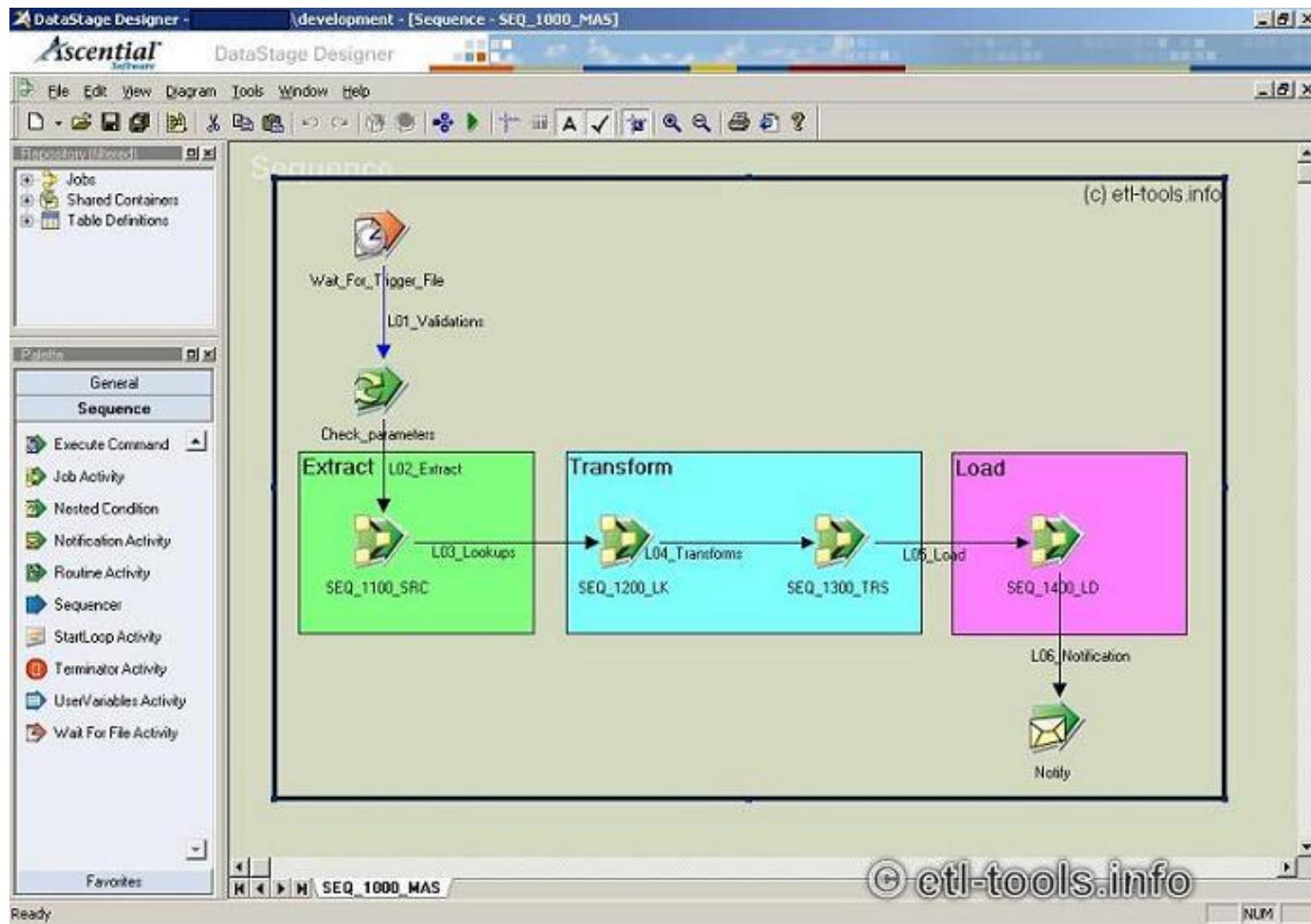
Outline Code Viewer

Business Mod aBusinessModel 0.1

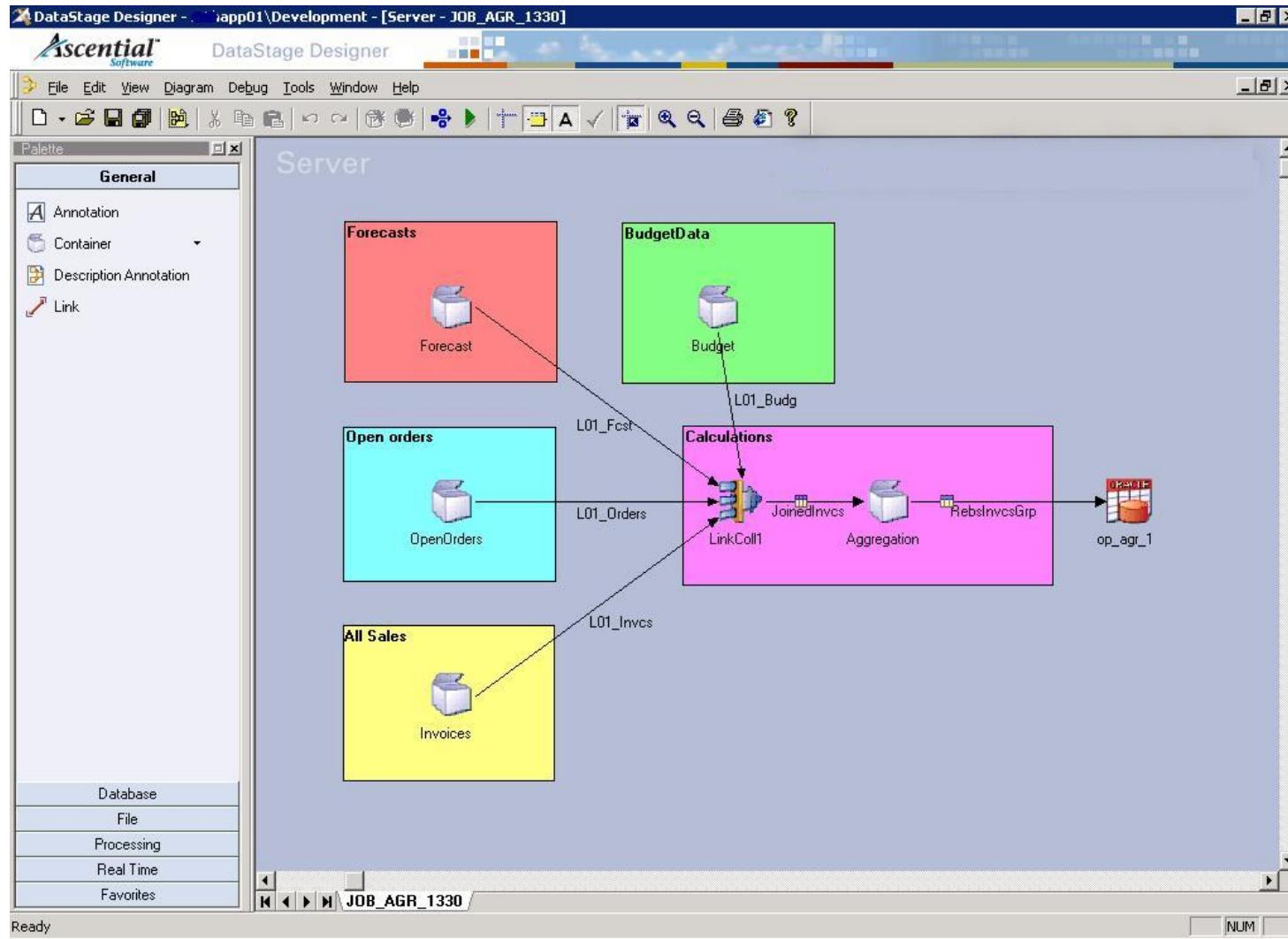
Appearance Assignment

Type	Name	Comment
Job	POrders	

Esempio 5 (DataStage)

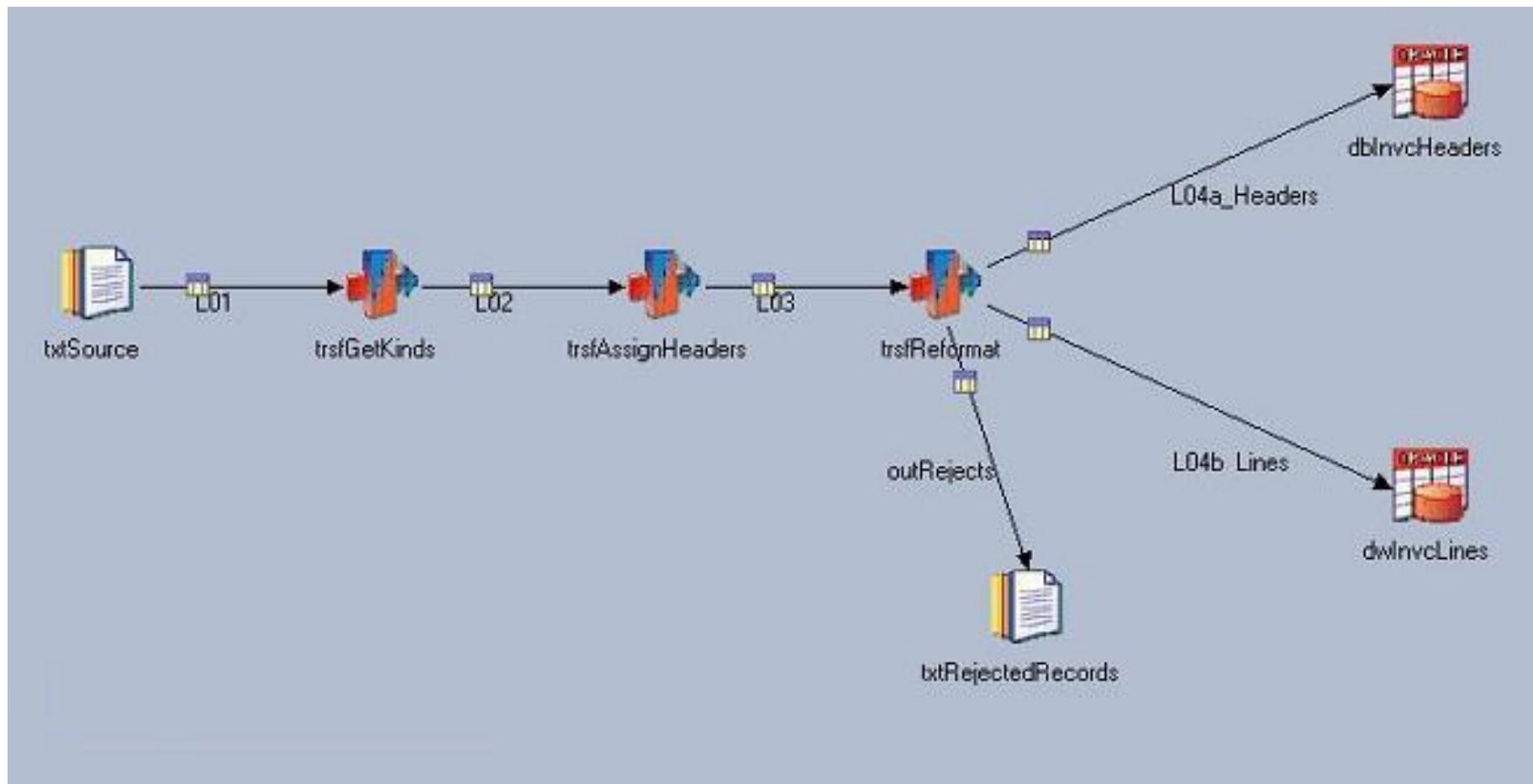


Esempio 6 (DataStage)



Esempio 7

- Esempio di processo su file sequenziali:





Analisi dei requisiti utente

Analisi dei requisiti

- Durante la fase di pianificazione del progetto si è già verificata un'interazione tra utenti e progettista, con lo scopo di:
 - ▣ individuare obiettivi e confini del sistema di data warehousing;
 - ▣ valutare le priorità;
 - ▣ stimare il valore aggiunto.
- A seguito delle priorità stabilite si procede alla costruzione di un singolo Data mart.

Obiettivi dell'analisi requisiti utente

- Raccolta delle esigenze di utilizzo del Data mart espresse dagli utenti finali.
- L'analisi dei requisiti utente ha un'importanza strategica perché influenza tutte le decisioni da prendere durante le diverse attività, con un ruolo primario nel determinare:
 - schema concettuale dei dati del DW;
 - progetto dell'alimentazione;
 - specifiche delle applicazioni per l'analisi dei dati;
 - architettura del sistema;
 - piano di avviamento e formazione;
 - linee guida per la manutenzione e l'evoluzione del sistema.

Fonti dell'analisi dei requisiti utente

- Le fonti da cui attingere i requisiti sono i “*business users*”, ossia i futuri utenti del data warehouse.
 - ▣ Il dialogo tra utenti e progettisti è spesso infruttuoso a causa del differente linguaggio usato.
 - ▣ È fondamentale porre grande cura nella fase di analisi, in quanto la soddisfazione degli utenti dipende in massima parte dall'accuratezza con la quale le loro richieste ottengono un'efficace risposta nel sistema.
 - ▣ Sono previste sia *Interviste* che *Riunioni coordinate*.
- Gli aspetti tecnici vengono individuati mediante l'interazione con i gestori del sistema operazionale.
 - ▣ I requisiti riguardano vincoli imposti al sistema e mirano a garantire livelli ottimali di prestazioni ed un'integrazione indolore con il sistema informativo preesistente.

I fatti

- I fatti sono concetti su cui gli utenti finali del Data mart baseranno il processo decisionale;
 - ogni fatto descrive una categoria di eventi che si verificano in azienda.
- Caratteristiche che guidano il progettista verso la determinazione dell'insieme dei fatti per il Data mart:
 - **aspetti dinamici:** gli eventi che vengono descritti dal fatto devono avere una componente temporale;
 - **dominio applicativo.**
 - **tipo di analisi** che l'utente vuole eseguire.
 - Un fatto di interesse per un Data mart potrebbe non esserlo per un altro.

I fatti (2)

- Individuare i fatti non è sufficiente:
 - per ognuno di essi è necessario disporre di informazioni di contorno, definite con l'aiuto della documentazione del livello riconciliato:
 - Possibili dimensioni (granularità).
 - Possibili misure.
 - Intervallo di storicizzazione.

Granularità

- Focalizzare le dimensioni di un fatto è importante perché consente di determinarne la “*granularità*”, ovvero il più fine livello di dettaglio a cui i dati saranno rappresentati nel Data mart.
- La scelta della *granularità* di rappresentazione di un fatto nasce da un delicato compromesso tra due esigenze contrapposte:
 - raggiungere un'elevata *flessibilità di utilizzo*, che richiederebbe di mantenere la stessa granularità del livello operazionale.
 - conseguire *buone prestazioni* con la necessità di avere un consistente grado di sintesi dei dati.

Misure e Intervallo di storizzazzione

- La valutazione delle misure con cui quantificare ciascun fatto ha un ruolo preliminare e orientativo, in quanto la definizione dettagliata delle misure da abbinare al fatto è rimandata alla successiva fase di progettazione concettuale.
- È l'arco temporale che gli eventi memorizzati nel Data mart devono coprire.
 - Valori tipici variano da 3 a 5 anni.

Caso di studio

- *Data mart per la gestione di approvvigionamenti e vendite in una catena di supermercati.*
- **Requisiti utente:**

Fatto	Possibili dimensioni	Possibili misure	Storicità
inventario di magazzino	prodotto, data, magazzino	quantità in magazzino	1 anno
vendite	prodotto, data, negozio	quantità venduta, importo, sconto	5 anni
linee d'ordine	prodotto, data, fornitore	quantità ordinata, importo, sconto	3 anni

Caso di studio (2)

- **Carico di lavoro preliminare:** Raccolta delle specifiche relative alle interrogazioni di analisi più frequenti sul Data mart.

Fatto	Interrogazione
Inventario di magazzino	<ul style="list-style-type: none">• quantità media di prodotto presente mensilmente in tutti i magazzini• andamento giornaliero delle scorte complessive per ogni tipo di prodotto• prodotti per i quali è stata esaurita la scorta di magazzino contemporaneamente in almeno un'occasione durante la settimana scorsa.
Vendite	<ul style="list-style-type: none">• incasso totale giornaliero di ciascun negozio• per un negozio, incassi relativi alle diverse categorie di prodotti durante un certo giorno• riepilogo annuale degli incassi per regione relativamente ad un dato prodotto• quantità totali di ciascun tipo di prodotto venduto durante l'ultimo mese
Linee d'ordine	<ul style="list-style-type: none">• quantità totale ordinata annualmente presso un certo fornitore• importo giornaliero ordinato nell'ultimo mese per un certo tipo di prodotto• sconto massimo applicato da ciascun fornitore durante l'ultimo anno per ciascuna categoria di prodotto



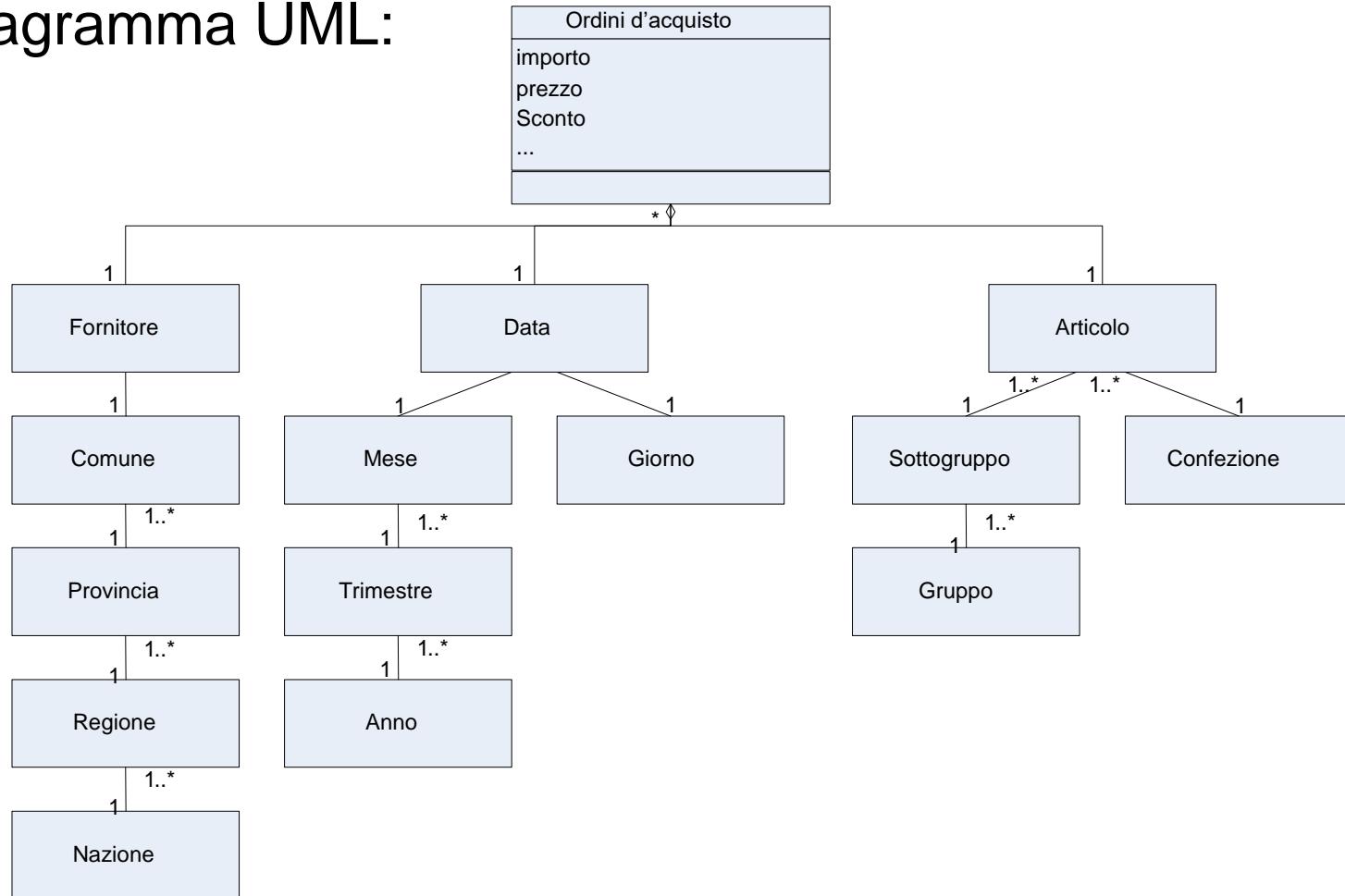
Modellazione concettuale

Modellazione concettuale

- Affronta il problema della traduzione dei requisiti in termini di un modello astratto, indipendente dal DBMS.
 - Non esistono standard di modello o di processo.
- Il modello Entity-Relationship (ER) è diffuso come strumento per la modellazione concettuale dei Data mart.
 - Esso è però orientato alle associazioni tra i dati e non alla sintesi.
 - È sufficientemente espressivo per rappresentare la maggior parte dei concetti, ma non è in grado di mettere in luce il modello multidimensionale.

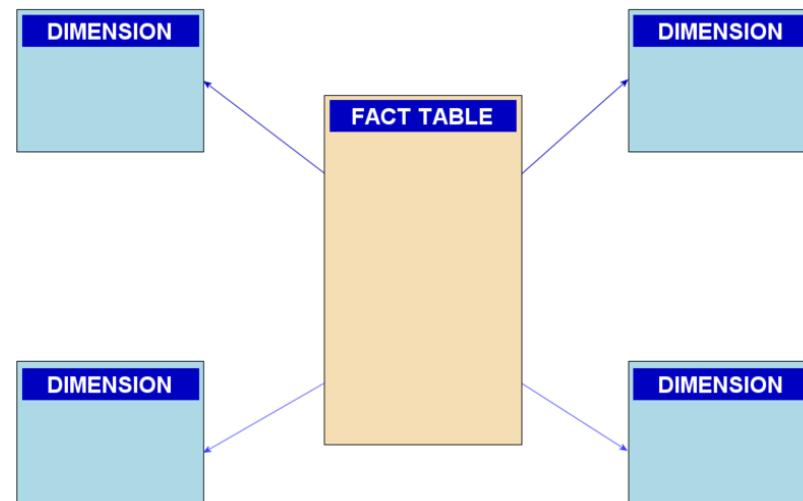
Modellazione concettuale in UML

- Un modello concettuale per gli ordini d'acquisto basato su diagramma UML:



Lo Schema a Stella (Star Schema)

- Lo schema a stella è un modello logico che può essere usato per la modellazione concettuale.
- Usare lo schema a stella per i Data mart equivale a modellare uno schema logico per un DB relazionale.
- Questo approccio porta a schemi fortemente denormalizzati.



Dimensional Fact Model (DFM)

- Il Dimensional Fact Model (DFM) è un modello concettuale concepito per il supporto allo sviluppo di Data mart.
- È una specializzazione del modello multidimensionale per applicazioni di Data warehousing.

Dimensional Fact Model (2)

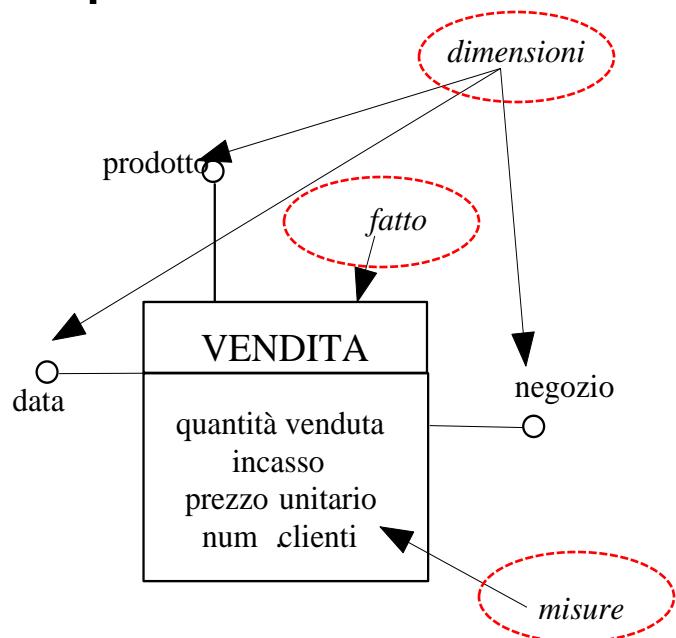
- Il DFM è un modello concettuale grafico per Data mart, pensato per:
 - supportare efficacemente il progetto concettuale;
 - creare un ambiente su cui formulare in modo intuitivo le interrogazioni dell'utente;
 - permettere il dialogo tra progettista e utente finale per raffinare le specifiche dei requisiti;
 - creare una piattaforma stabile da cui partire per il progetto logico (*indipendentemente dal modello logico target*);
 - restituire una documentazione a posteriori espressiva e non ambigua.
- La rappresentazione concettuale generata dal DFM consiste in un insieme di **schemi di fatto**.
 - Gli elementi di base modellati dagli schemi di fatto sono i *fatti*, le *misure*, le *dimensioni* e le *gerarchie*.

Esempi di fatti

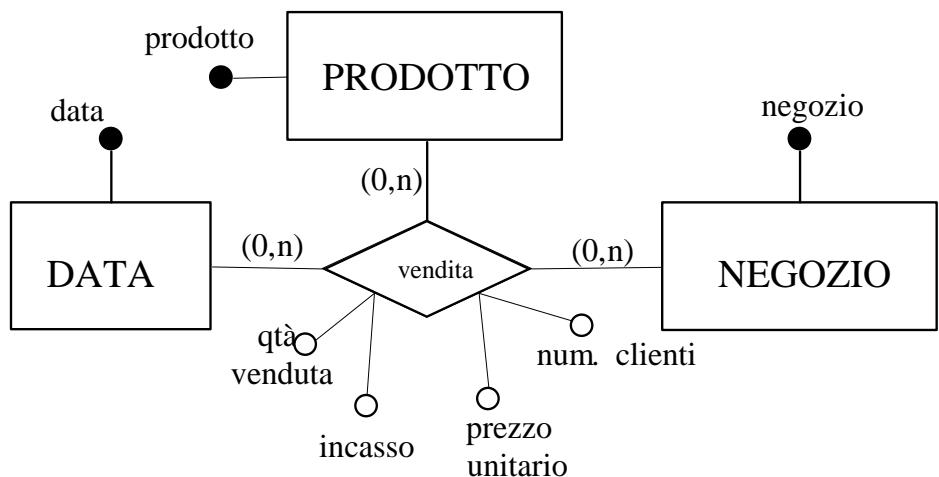
	<i>Data mart</i>	<i>Fatti</i>
commerciale/ manifatturiero	approvvigionamenti	acquisti, inventario di magazzino, distribuzione
	produzione	confezionamento, inventario, consegna, manifattura
	gestione domanda	vendite, fatturazione, ordini, spedizioni, reclami
	marketing	promozioni, fidelizzazione, campagne pubblicitarie
	bancario	conti correnti, bonifici, prestiti ipotecari, mutui
	investimenti	acquisto titoli, transazioni di borsa
finanziario	servizi	carte di credito, domiciliazioni bollette
	scheda di ricovero	ricoveri, dimissioni, interventi chirurgici, diagnosi
	pronto soccorso	accessi, esami, dimissioni
sanitario	medicina di base	scelte, revoche, prescrizioni
	merci	domanda, offerta, trasporti
	passeggeri	domanda, offerta, trasporti
	manutenzione	interventi
trasporti	traffico	traffico in rete, chiamate
	CRM	fidelizzazione, reclami, servizi
telecomunicazioni	gestione domanda	biglietteria, noleggi auto, soggiorni
	CRM	frequent-flyers, reclami
	logistica	trasporti, scorte, movimentazione
turismo	risorse umane	assunzioni, dimissioni, promozioni, incentivi
	budgeting	budget commerciale, budget di marketing
	infrastrutture	acquisti, opere
gestionale		

Esempio DFM

- Un semplice schema di fatto per le vendite:



- Schema ER corrispondente:



Il DFM: costrutti di base

- Un **fatto** è un concetto di interesse per il processo decisionale:
 - ▣ Tipicamente modella un insieme di eventi che accadono nell'impresa (**Es:** vendite, spedizioni, acquisti,...).
 - ▣ È essenziale che un fatto abbia aspetti dinamici, ovvero evolva nel tempo.
- Una **misura** è una proprietà numerica di un fatto e ne descrive un aspetto quantitativo di interesse per l'analisi (**Es:** ogni vendita è misurata dal suo incasso).
 - ▣ Le misure vengono in genere usate per effettuare calcoli.

Il DFM: costrutti di base (2)

- Una **dimensione** è una proprietà con dominio finito di un fatto e ne descrive una coordinata di analisi (dimensioni tipiche per il fatto vendite sono prodotto, negozio, data).
 - Un fatto esprime una associazione molti-a-molti tra le dimensioni.
 - I fatti hanno natura dinamica, rappresentata da una dimensione temporale.
- Un **evento primario** è una particolare occorrenza di un fatto, individuata da una ennupla costituita da un valore per ciascuna dimensione.
 - A ciascun evento primario è associato un valore per ciascuna misura.
 - **Es:** il giorno 10/10/2001 il negozio ‘DiTutto’ ha venduto 10 confezioni di detersivo Brillo per un incasso complessivo di 25€.

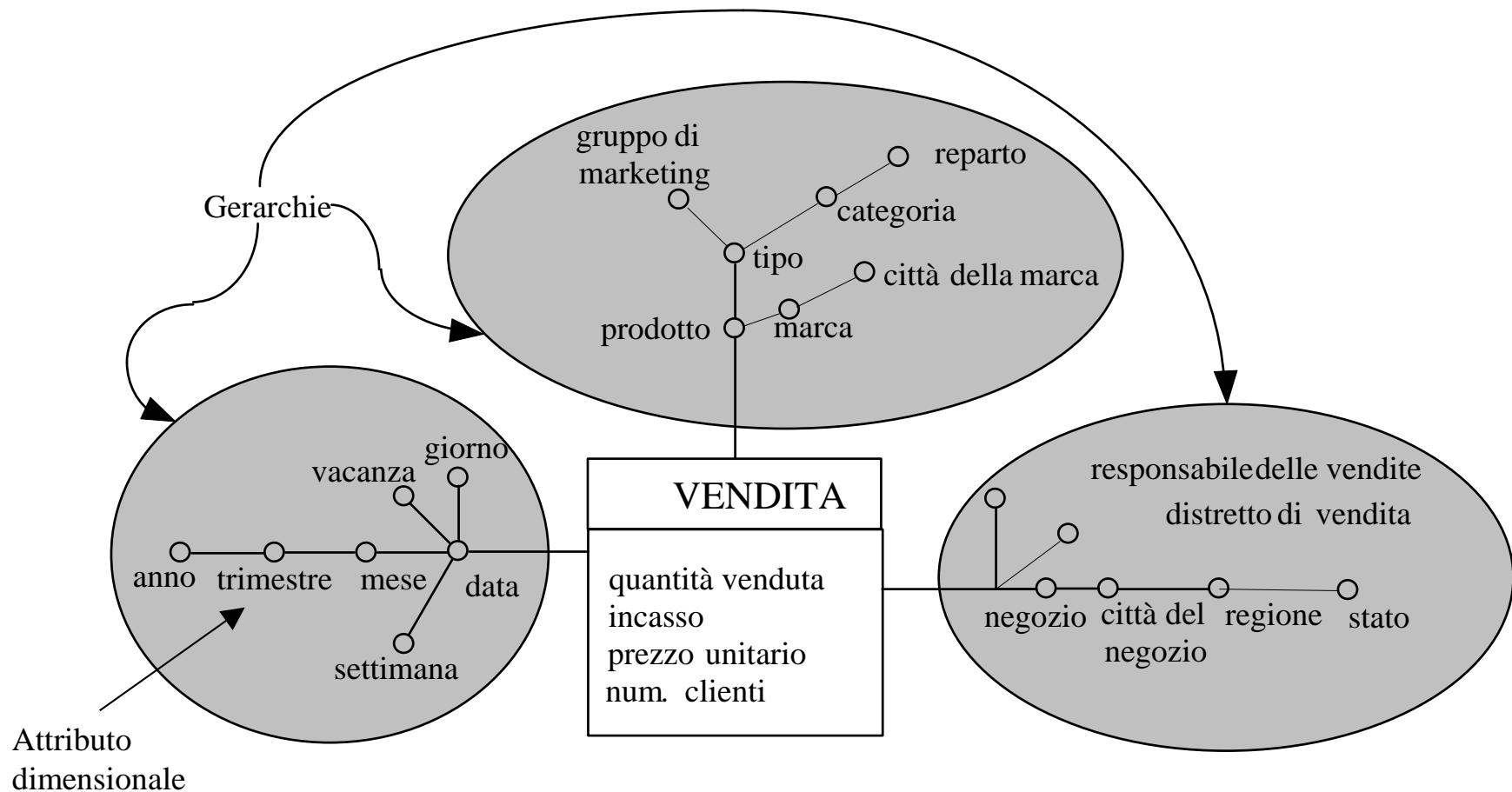
DFM: Attributi Dimensionali

- Gli **attributi dimensionali** sono le dimensioni e gli attributi che le descrivono.
 - ▣ **Es:** un prodotto è descritto da *tipo, categoria, marca, reparto,*
 - ▣ Le relazioni tra gli attributi dimensionali sono espresse dalle *gerarchie*.
- Un **gerarchia** è un *albero* direzionato i cui nodi sono attributi dimensionali e i cui archi modellano associazioni ***molti-a-uno*** tra coppie di attributi dimensionali.
 - ▣ Racchiude una dimensione, posta alla radice dell'albero, e tutti gli attributi dimensionali che la descrivono.

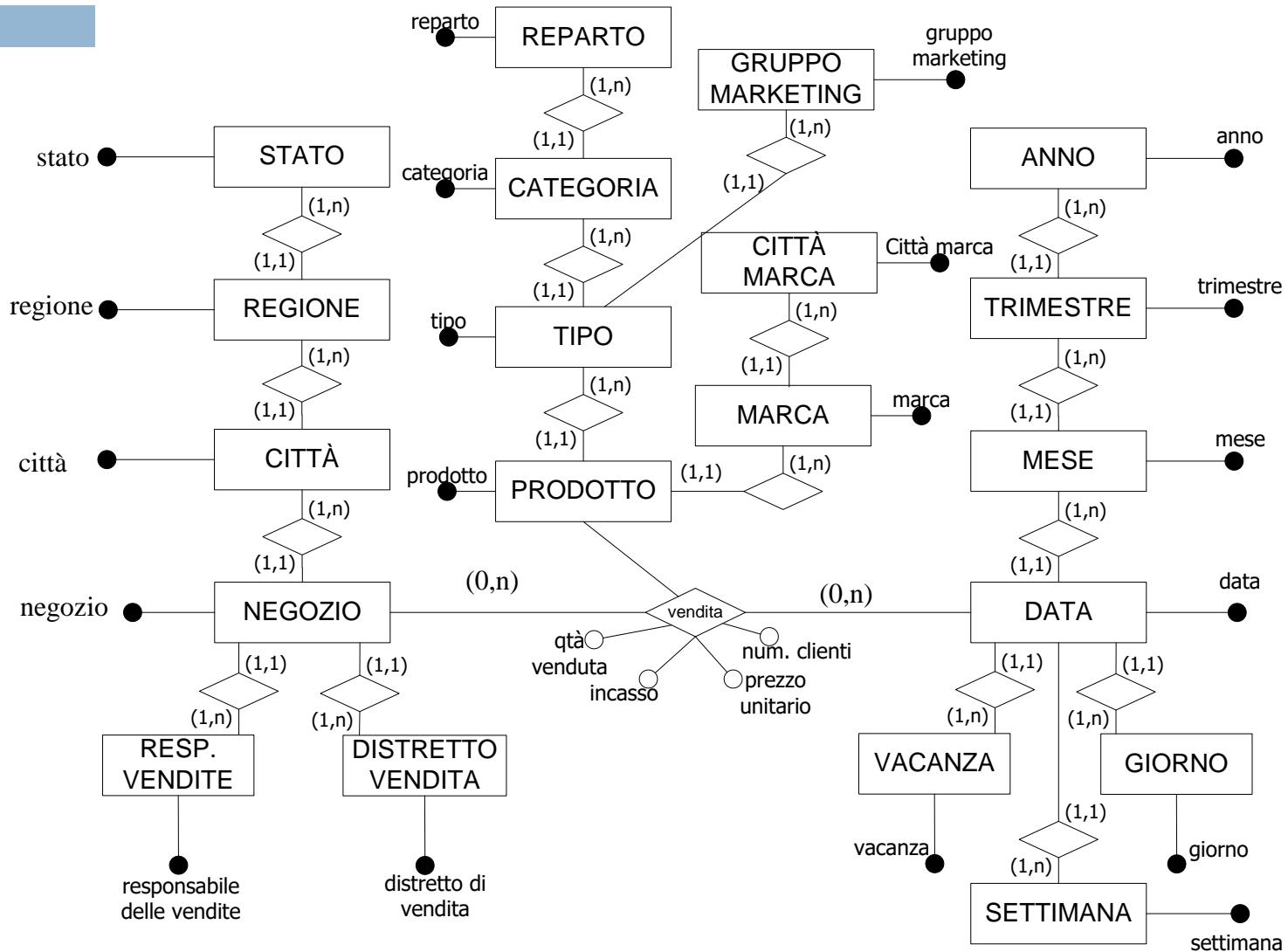
ATTENZIONE! Gerarchie ER ≠ Gerarchie DFM

DFM Arricchito

□ Schema di fatto arricchito per la Vendita:



Schema ER del DFM Vendita



“Naming conventions”

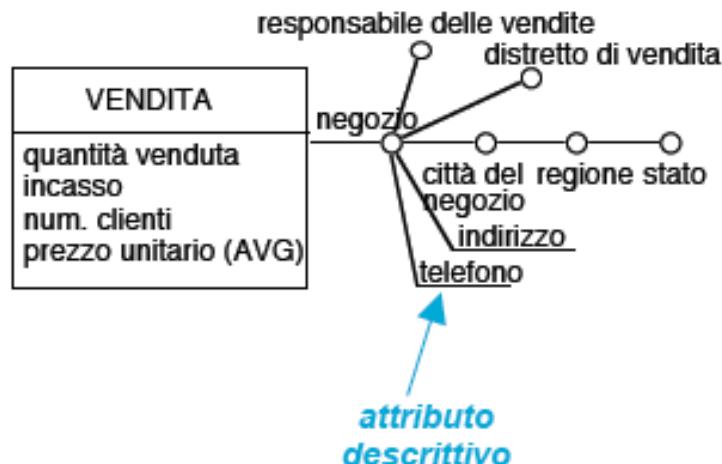
- Tutti gli attributi dimensionali in ciascuno schema di fatto devono avere nomi diversi.
- Eventuali nomi uguali devono essere differenziati qualificandoli con il nome di un attributo dimensionale che li precede nella gerarchia.
 - Ad esempio, *warehouse city* è la città in cui si trova un magazzino, mentre *store city* è la città in cui si trova un negozio.
- I nomi degli attributi non dovrebbero riferirsi esplicitamente al fatto a cui appartengono.
 - **Es:** si evitino *shipped product* e *shipment date*.
- Attributi con lo stesso significato in schemi diversi devono avere lo stesso nome.

Evento secondario

- Dato un insieme di attributi dimensionali, ciascuna ennupla di valori individua un **evento secondario** che aggrega tutti gli eventi primari corrispondenti.
- A ciascun evento secondario è associato un valore per ciascuna misura, che riassume in sé tutti i valori della stessa misura negli eventi primari corrispondenti.
- **Es:** le vendite possono essere raggruppate a seconda della categoria dei prodotti venduti, oppure a seconda del mese in cui sono effettuate le vendite, oppure a seconda della città in cui si trova in negozio, ecc ...

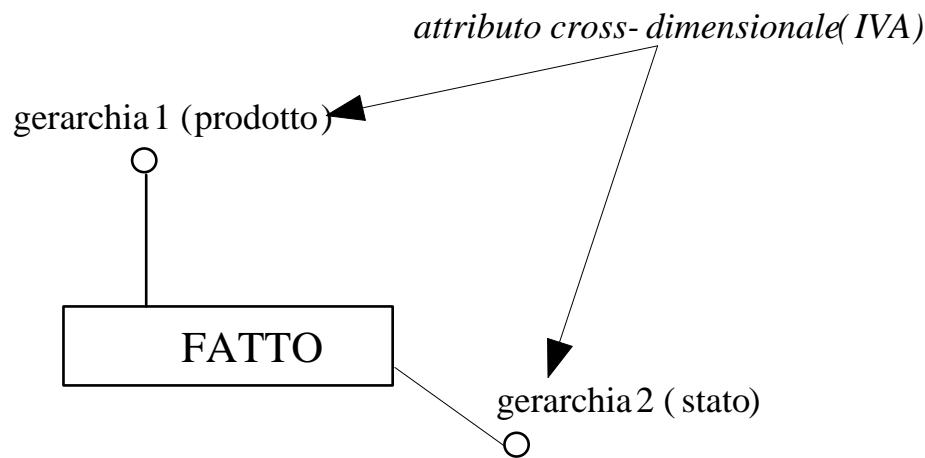
Attributi descrittivi

- **Attributi descrittivi:** specificano le proprietà degli attributi dimensionali di una gerarchia, e sono determinati tramite dipendenze funzionali.
 - Non possono essere usati per l'aggregazione poiché hanno spesso domini con valori continui.
 - Un attributo descrittivo non può essere usato per identificare singoli eventi né per effettuare calcoli.
 - **Es:** numero di telefono di un negozio, indirizzo di un negozio.



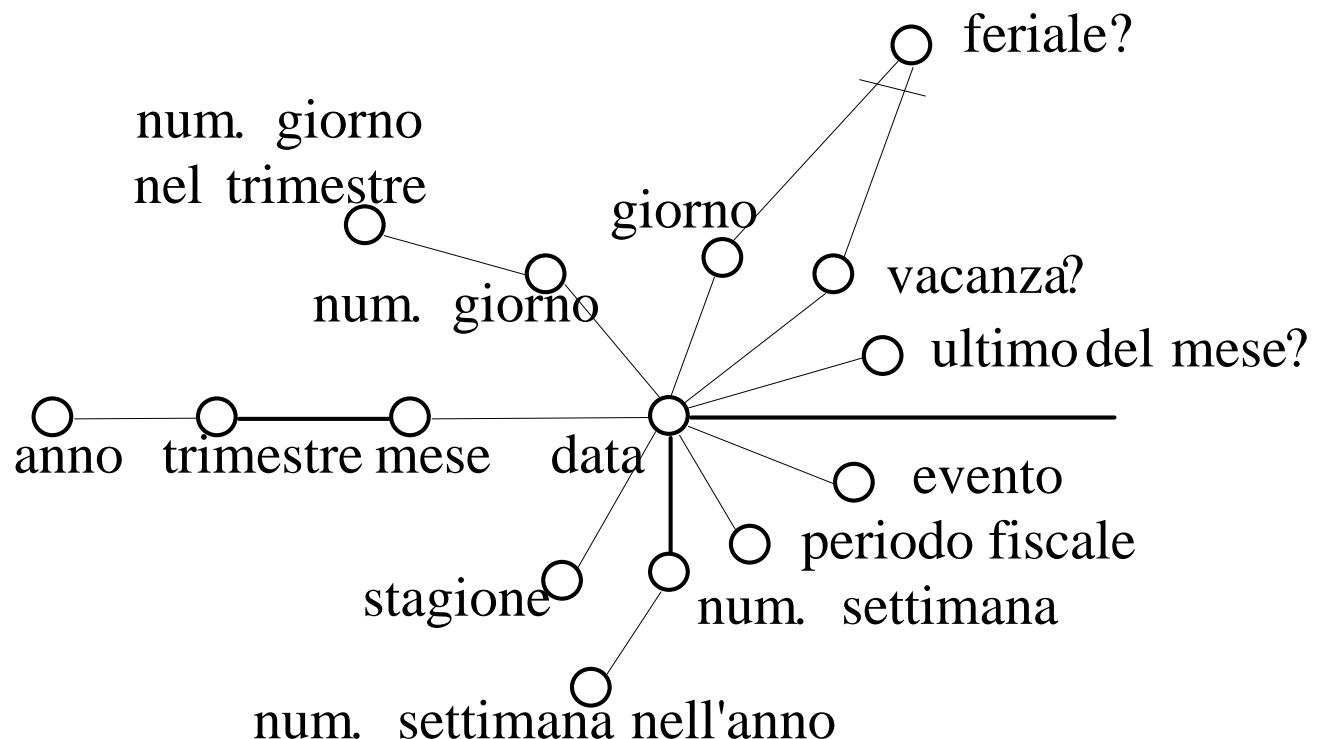
Attributi cross-dimensional

- **Attributi cross-dimensional**: sono attributi dimensionali o descrittivi il cui valore è determinato mediante la combinazione di due o più attributi dimensionali.
 - Gli attributi dimensionali possono appartenere anche a gerarchie diverse.
 - **Es:** l'IVA su un prodotto dipende sia dalla categoria del prodotto che dallo stato in cui esso è venduto.



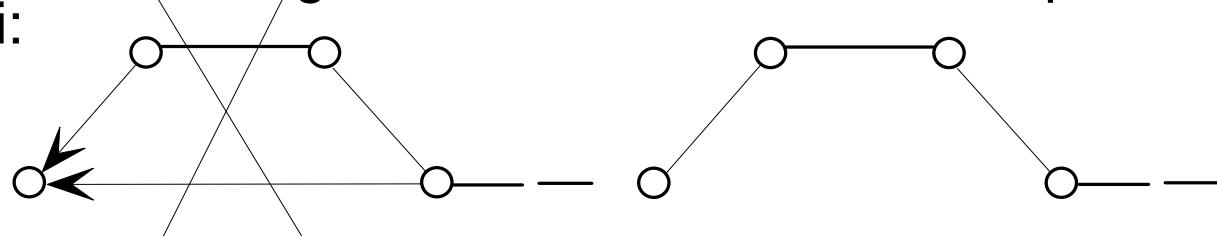
Esempio di gerarchia temporale completa

- L'attributo booleano **feriale?** È determinato congiuntamente dal **giorno** e dal booleano **vacanza?**:



Convergenza

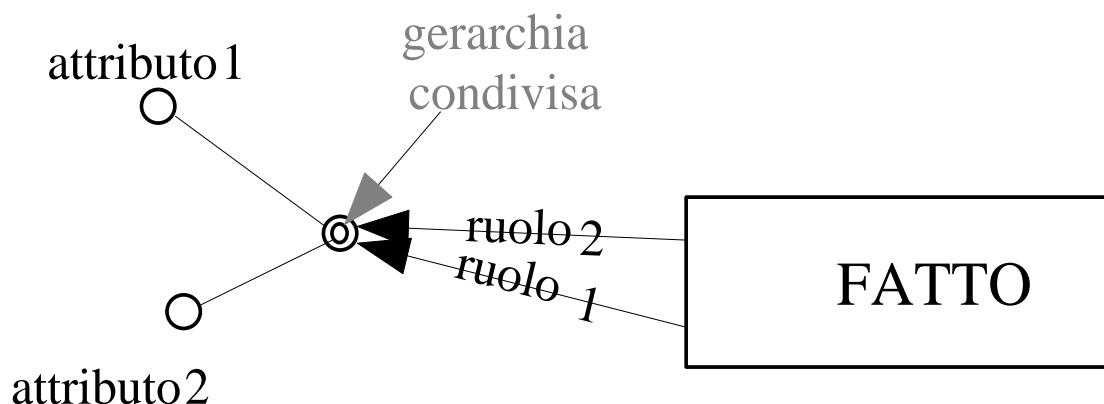
- La convergenza riguarda la struttura delle gerarchie.
 - Sullo schema di fatto le convergenze sono denotate da due o più archi, in genere appartenenti alla stessa gerarchia, che terminano nello stesso attributo dimensionale.
 - In presenza di una gerarchia che non ha una struttura ad albero non è più possibile determinare univocamente il verso degli archi e, per fare ciò, gli archi convergenti devono essere orientati.
 - Attributi apparentemente uguali non determinano sempre una convergenza.
 - Se uno dei percorsi alternativi non comprende attributi intermedi, non ha ragione di esistere: la convergenza è infatti del tutto ovvia grazie alla transitività delle dipendenze funzionali:



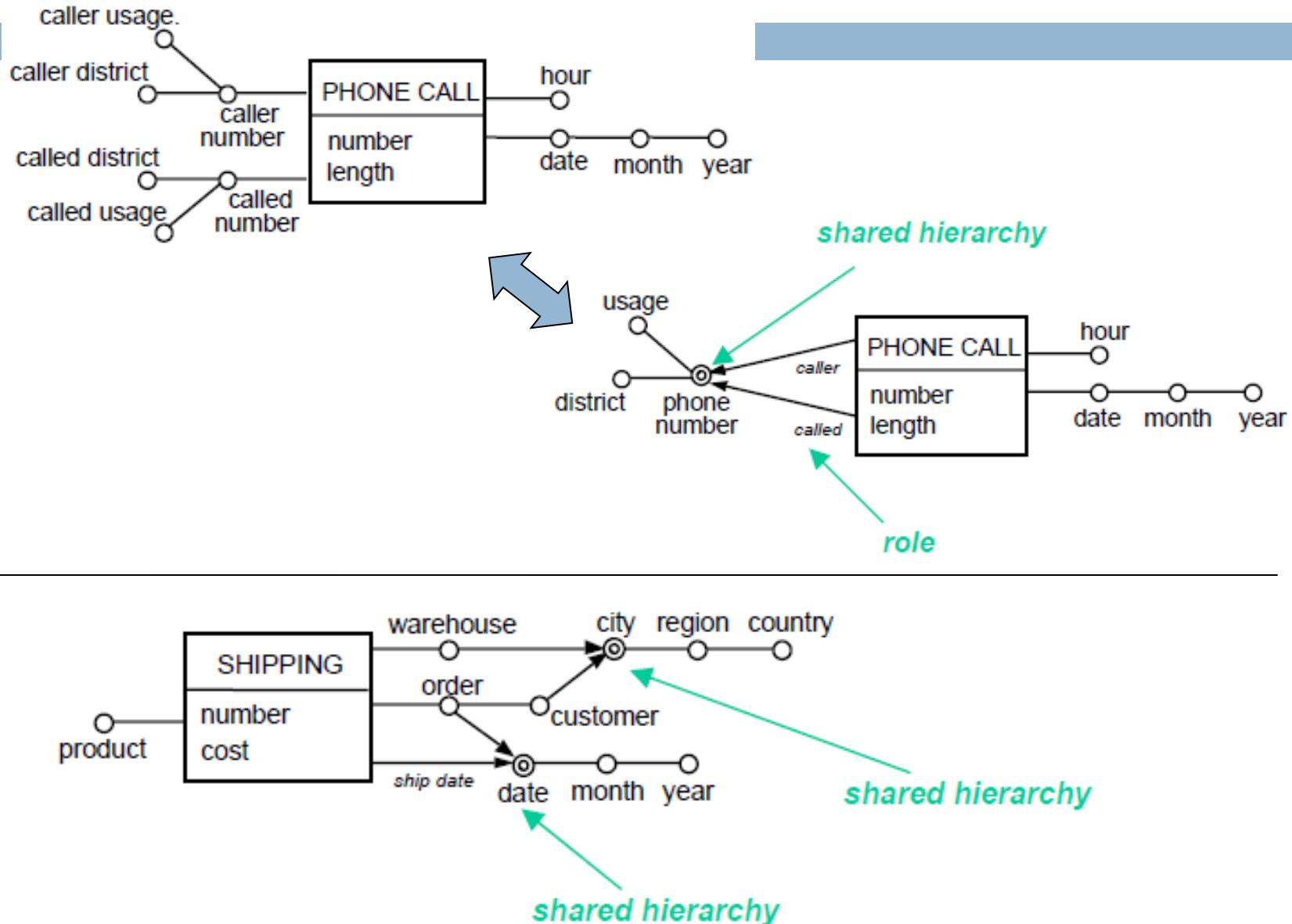
Convergenza ridondante e sua rappresentazione corretta

Gerarchie condivise

- Negli schemi di fatto spesso si rende necessario duplicare intere porzioni di gerarchie e ciò comporta l'uso di diversi nomi per evitare ambiguità.
- Tramite le **gerarchie condivise** si introduce una notazione grafica abbreviata che migliora la leggibilità dello schema. Si introducono quando si hanno significati diversi per lo stesso tipo di dati e il significato viene espresso inserendo il ruolo sull'arco entrante della gerarchia.

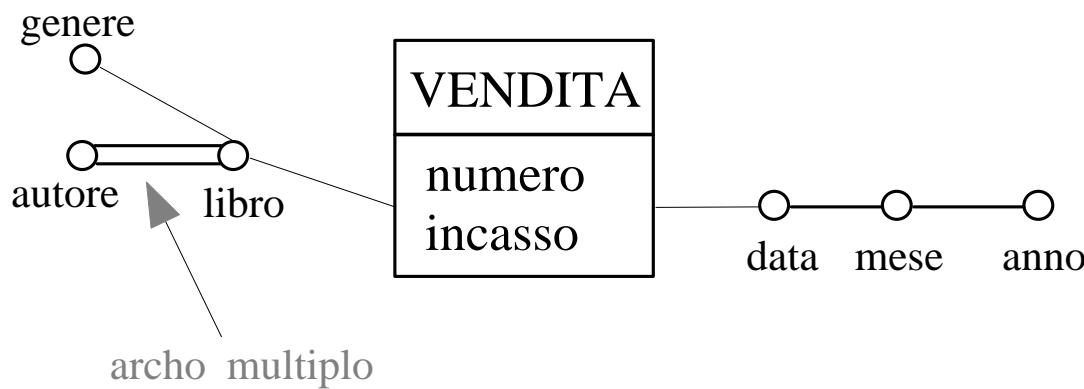


Esempio: Gerarchie condivise



Archi multipli

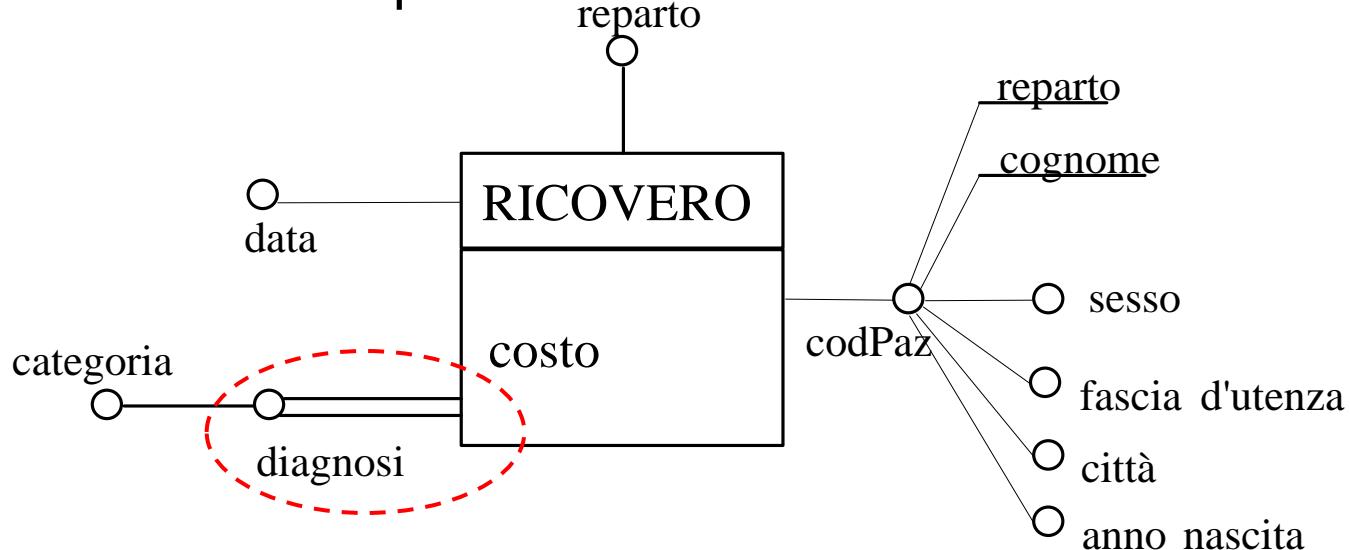
- Un arco multiplo tra due attributi **a** e **b** indica che ad ogni singolo valore di **a** possono corrispondere più valori di **b**.
- **Es:** Schema di fatto per le vendite dei libri.



- Il significato di un arco multiplo che va da un attributo autore ad un attributo libro sta nel fatto che tra autore e libro esiste un'associazione M-N.

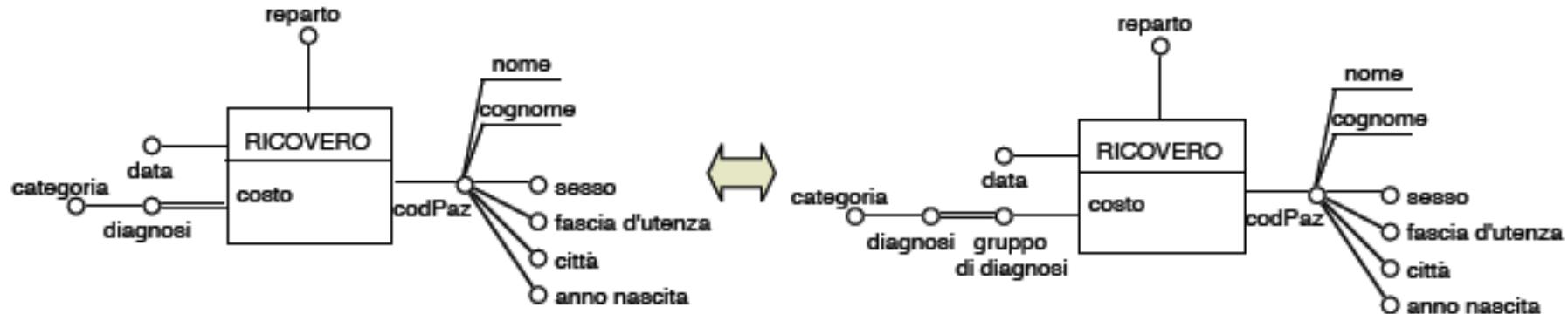
Aggregazioni su dimensioni

- **Es:** Schema di fatto per i ricoveri.



- Nel momento in cui un attributo entra in una dimensione piuttosto che in un attributo qualsiasi, il caso diventa più complesso.
- Infatti è possibile aggregare i ricoveri in base alle diagnosi in uscita, ma anche selezionare le diagnosi in base ai ricoveri.

Alternativamente...



Archi opzionali

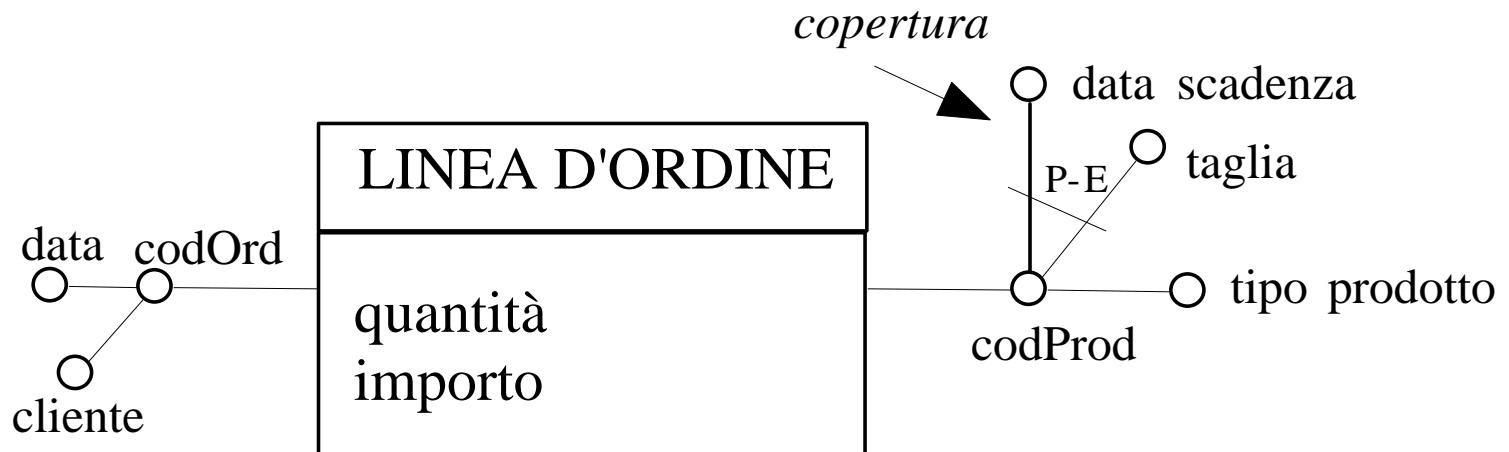
- Vengono impiegati per modellare associazioni dello schema di fatto non definite per un particolare sottoinsieme di eventi.
 - ▣ Si rappresenta con un trattino sull'arco.
- Se **r** è l'arco opzionale, bisogna distinguere se esso determina un attributo o una dimensione.
 - ▣ Se **r** determina la dimensione **d** allora essa è opzionale, ossia esistono alcuni eventi primari identificati solo dalle altre dimensioni.
 - **Es:** la promozione su un prodotto, identificato da una dimensione, vale solo per alcune combinazioni di *prodotto-negozio-data*.

Copertura tra archi opzionali

- Se esistono più archi opzionali uscenti da uno stesso attributo è possibile definire la copertura, ossia stabilire una relazione tra le diverse opzionalità.
- Sia **a** un attributo dimensionale con archi opzionali verso i propri figli b_1, \dots, b_m . Allora la copertura si dice:
 - **totale** se per ogni valore di **a** è sempre associato almeno un valore dei figli o **parziale** se esistono valori di **a** per i quali tutti i figli sono indefiniti,
 - **esclusivo** se per ogni valore di **a** si ha al massimo un valore per uno dei figli o **sovraposta** se invece esistono valori di **a** abbinati a due o più figli.
- I tipi di copertura sono identificati da T-E, T-S, P-E, P-S.

Esempio di copertura

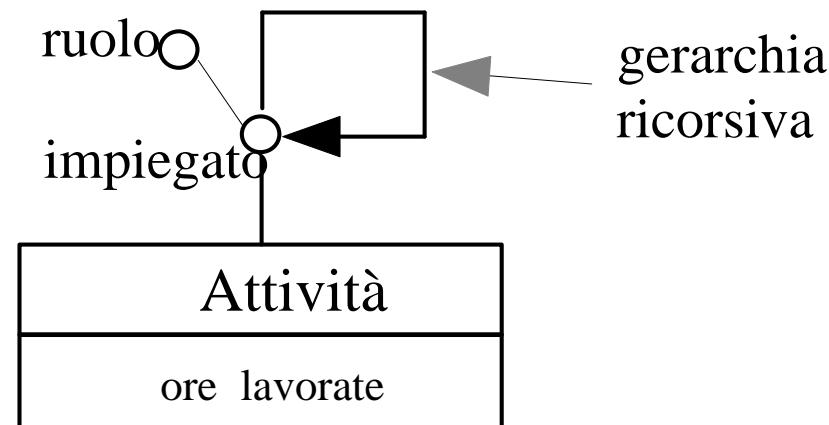
- Copertura per un insieme di archi opzionali.



Prodotti di tre tipi: alimentari, abbigliamento, casalinghi. Quindi **data scadenza** e **taglia** sono definiti solo per alimentari e casalinghi rispettivamente: la copertura è P-E.

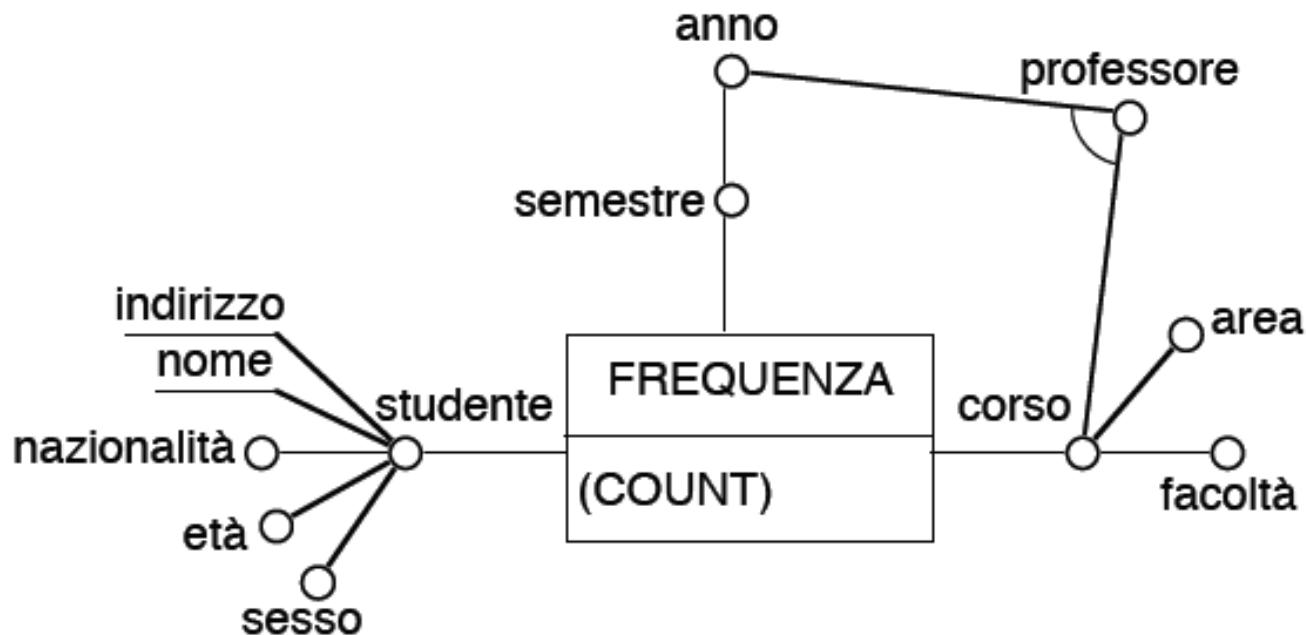
Gerarchie ricorsive

- Nelle **gerarchie ricorsive** (*unbalanced hierarchies*) le istanze possono avere lunghezze diverse, ma le relazioni padre figlio sono consistenti tra i livelli, e quindi uniformi.
 - Graficamente le gerarchie ricorsive si rappresentano tramite un auto-ciclo sull'attributo in questione.
 - **Es:** gerarchia dei ruoli degli impiegati (subordinazione degli impiegati).



Schemi di fatto vuoti

- Uno schema di fatto si dice **vuoto** se non ha misure:
 - In questo caso, il fatto registra solo il verificarsi di un evento.



Additività

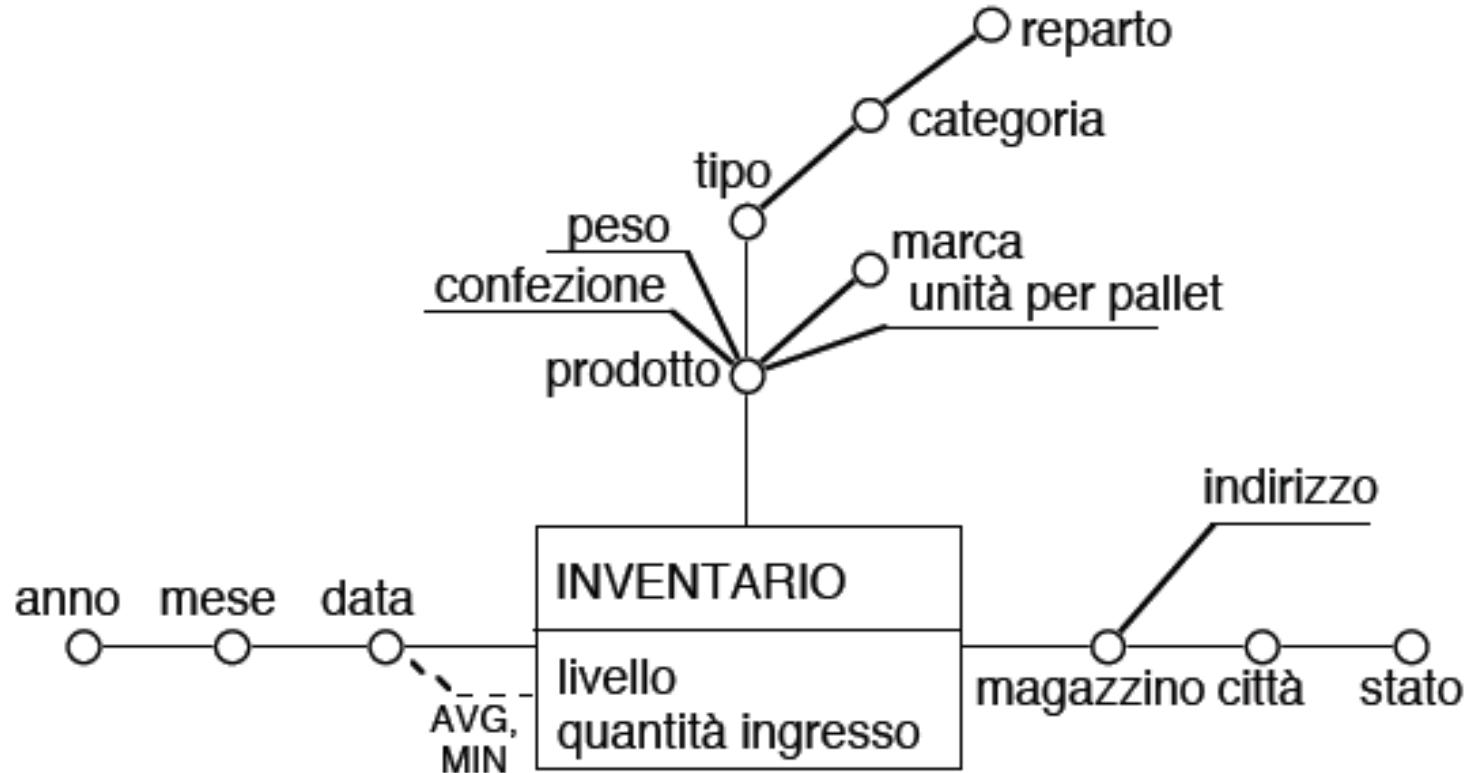
- L'aggregazione richiede di definire un operatore adatto per comporre i valori delle misure che caratterizzano gli eventi primari in valori da abbinare a ciascun evento secondario.

Misure additive

- Una misura è detta **additiva** su una dimensione:
 - se i suoi valori possono essere aggregati lungo la corrispondente gerarchia tramite l'operatore di somma.
- Una misura è **non-additiva**:
 - se non può essere aggregata lungo una data gerarchia tramite l'operatore di somma.
- Una misura è **non-aggregabile**:
 - se non può essere aggregata lungo qualsiasi gerarchia tramite l'uso di qualsiasi operatore di aggregazione.
- Una misura *non-additiva* è *non-aggregabile* se nessun operatore di aggregazione può essere usato su di essa.

Esempio

- Il **livello** di inventario non è additivo sul tempo, ma lo è sulle altre dimensioni:

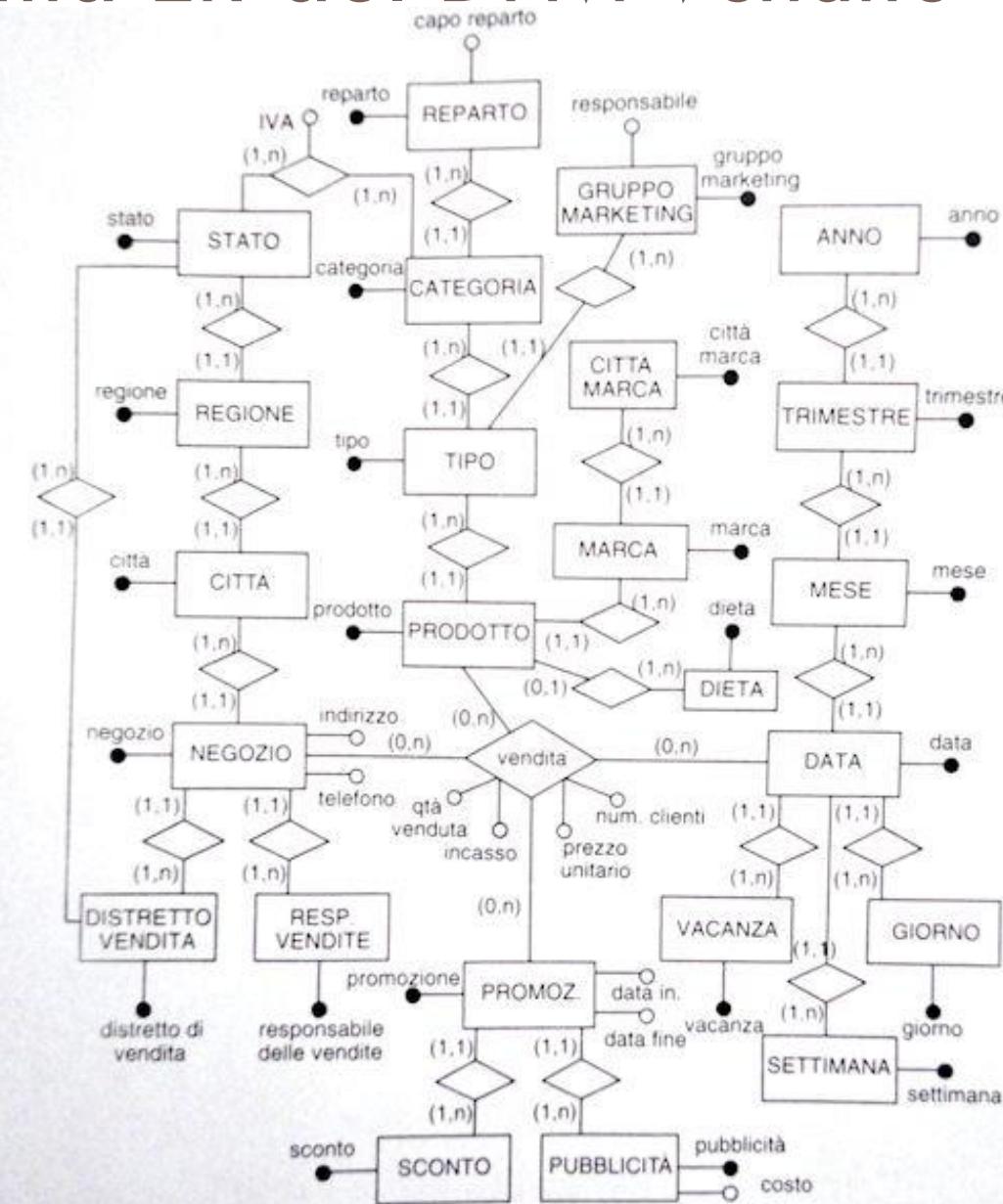


Tipi di misure additive

- Tre tipi di misure additive:
 - **di flusso** (periodo temporale come incasso mensile, num. prodotti venduti):
 - può essere valutata cumulativamente alla fine di un periodo di tempo;
 - può essere aggregata tramite tutti gli operatori standard;
 - **di livello** (in particolari istanti di tempo come gli abitanti di una città):
 - valutata in un preciso istante (*snapshot*):
 - non additiva lungo la dimensione temporale:
 - **unitarie** (in particolari istanti di tempo, ma in termini relativi come il prezzo unitario di un prodotto o una percentuale di sconto):
 - valutata ad un certo istante ed espressa con termini relativi;
 - non additiva lungo qualsiasi dimensione.

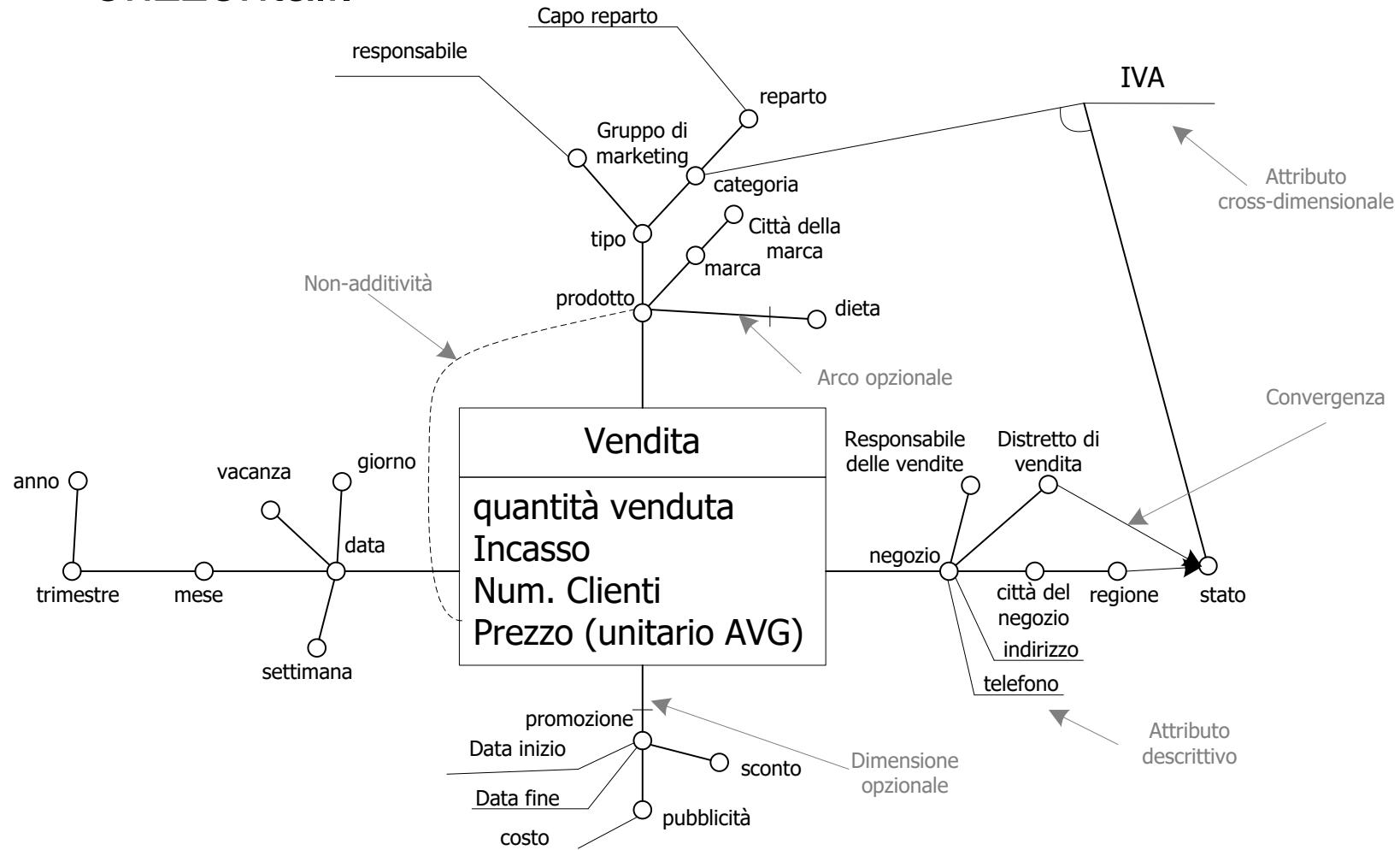
	Gerarchie temporali	Gerarchie non temporali
Misure di flusso	SUM, AVG, MIN, MAX	SUM, AVG, MIN, MAX
Misure di livello	AVG, MIN, MAX	SUM, AVG, MIN, MAX
Misure unitarie	AVG, MIN, MAX	AVG, MIN, MAX

Schema ER del DFM Vendite



DFM completo delle vendite

- Gli attributi descrittivi sono sempre foglie delle gerarchie e sono rappresentati nel DFM da linee orizzontali.



Eventi

- Definiamo evento un'istanza che popola uno schema di fatto.
- Gli eventi possono essere aggregati rispetto i valori degli attributi lungo le gerarchie.
- Le misure degli eventi aggregati sono ottenute aggregando le misure degli eventi corrispondenti nello schema dei fatti originale:
 - Gli operatori di aggregazione standard: SUM, MIN, MAX, AVG, COUNT.
- L'aggregazione computa le misure con una granularità grezza rispetto a quella nello schema dei fatti originale:
 - La riduzione del dettaglio è usualmente ottenuta risalendo nella gerarchia.

Aggregazione di eventi

category	type	product	1999				2000			
			I '99	II '99	III '99	IV '99	I '00	II '00	III '00	IV '00
home cleaning	Washing powder	Brillo	100	90	95	90	80	70	90	85
		Sbianco	20	30	20	10	25	30	35	20
		Lucido	60	50	60	45	40	40	50	40
	soap	Manipulite	15	20	25	30	15	15	20	10
		Scent	30	35	20	25	30	30	20	15
food	milk	Latte F Slurp	90	90	85	75	60	80	85	60
		Latte U Slurp	60	80	85	60	70	70	75	65
		Yogurt Slurp	20	30	40	35	30	35	35	20
	soda	Bevimi	20	10	25	30	35	30	20	10
		Colissima	50	60	45	40	50	60	45	40

Measure: sold quantity



category	type	product	1999				2000			
			I '99	II '99	III '99	IV '99	I '00	II '00	III '00	IV '00
home clean	Washing powder	Brillo	225	225	220	200	190	185	215	170
		Sbianco	240	270	280	240	245	275	260	195
food	milk	Latte F Slurp	225	225	220	200	190	185	215	170
		Latte U Slurp	240	270	280	240	245	275	260	195

- Eventi primari per un cubo delle vendite in una matrice bidimensionale, con dimensioni *prodotto* e *trimestre* (misura *quantità*).
- Sono riportati i valori di altri attributi di interesse della stessa gerarchia (*anno* per *trimestre*, *tipo* e *categoria* per *prodotto*).
- Operatore di aggregazione impiegato: SUM.

Aggregazione di eventi (2)

- Eventi secondari sul pattern *tipo e trimestre*:

category	type	product
home cleaning	Washing powder	Brillo
		Sbianco
	soap	Lucido
		Manipulite
	Scent	
food	milk	Latte F Slurp
		Latte U Slurp
		Yogurt Slurp
	soda	Bevimi
		Colissima

year quart.	1999				2000			
	I '99	II '99	III '99	IV '99	I '00	II '00	III '00	IV '00
	100	90	95	90	80	70	90	85
	20	30	20	10	25	30	35	20
	60	50	60	45	40	40	50	40
	15	20	25	30	15	15	20	10
	30	35	20	25	30	30	20	15
	90	90	85	75	60	80	85	60
	60	80	85	60	70	70	75	65
	20	30	40	35	30	35	35	20
	20	10	25	30	35	30	20	10
	50	60	45	40	50	60	45	40

Measure: sold quantity

category	1999				2000			
	I '99	II '99	III '99	IV '99	I '00	II '00	III '00	IV '00
home clean	225	225	220	200	190	185	215	170
food	240	270	280	240	245	275	260	195



category	type	year	
		1999	2000
home cleaning	washing p.	670	605
	soap	200	155
food	milk	750	685
	soda	280	290

Aggregazione di eventi (3)

- Eventi secondari sul pattern *categoria e anno*.

category	type	product
home cleaning	Washing powder	Brillo
		Sbianco
		Lucido
	soap	Manipulite
		Scent
food	milk	Latte F Slurp
		Latte U Slurp
		Yogurt Slurp
	soda	Bevimi
		Colissima

year quart.	1999				2000			
	I '99	II '99	III '99	IV '99	I '00	II '00	III '00	IV '00
100	90	95	90	80	70	90	85	
20	30	20	10	25	30	35	20	
60	50	60	45	40	40	50	40	
15	20	25	30	15	15	20	10	
30	35	20	25	30	30	20	15	
90	90	85	75	60	80	85	60	
60	80	85	60	70	70	75	65	
20	30	40	35	30	35	35	20	
20	10	25	30	35	30	20	10	
50	60	45	40	50	60	45	40	

Measure: sold quantity

category	1999				2000			
	I '99	II '99	III '99	IV '99	I '00	II '00	III '00	IV '00
home clean.	225	225	220	200	190	185	215	170
food	240	270	280	240	245	275	260	195

category	type	1999	2000
home cleaning	washing p.	670	605
	soap	200	155
food	milk	750	685
	soda	280	290

year	1999	2000
category	870	760
home clean.	1030	975

Aggregazione di misure non-additive

- Per le misure non additive distinguiamo due casi:
- **Caso 1:** operatore di aggregazione uguale per tutte le dimensioni.
- Gli operatori di aggregazione sono classificabili in:
 - **Distributivi:** calcolo degli aggregati a partire da aggregati parziali (**Es:** SUM, MIN, MAX).
 - **Algebrici:** utilizzo di *misure di supporto* per il calcolo di aggregati a partire da aggregati parziali (**Es:** AVG, deviazione standard, baricentro).
 - **Olistici:** non permettono di calcolare aggregati a partire da aggregati parziali attraverso un numero finito di *misure di supporto* (**Es:** mediana, rango), per cui gli eventi secondari devono necessariamente essere calcolati a partire dagli eventi primari.

Operatore distributivo: SUM

category	type	product
home cleaning	washing powder	Brillo
		Sbianco
		Lucido
	soap	Manipulite
		Scent
food	milk	Latte F Slurp
		Latte U Slurp
		Yogurt Slurp
	soda	Bevimi
		Colissima

year	1999				2000			
quart.	I '99	II '99	III '99	IV '99	I '00	II '00	III '00	IV '00
100	90	95	90	80	70	90	85	
20	30	20	10	25	30	35	20	
60	50	60	45	40	40	50	40	
15	20	25	30	15	15	20	10	
30	35	20	25	30	30	20	15	
90	90	85	75	60	80	85	60	
60	80	85	60	70	70	75	65	
20	30	40	35	30	35	35	20	
20	10	25	30	35	30	20	10	
50	60	45	40	50	60	45	40	

Measure: sold quantity

year	1999				2000			
quart.	I'99	II'99	III'99	IV'99	I'00	II'00	III'00	IV'00

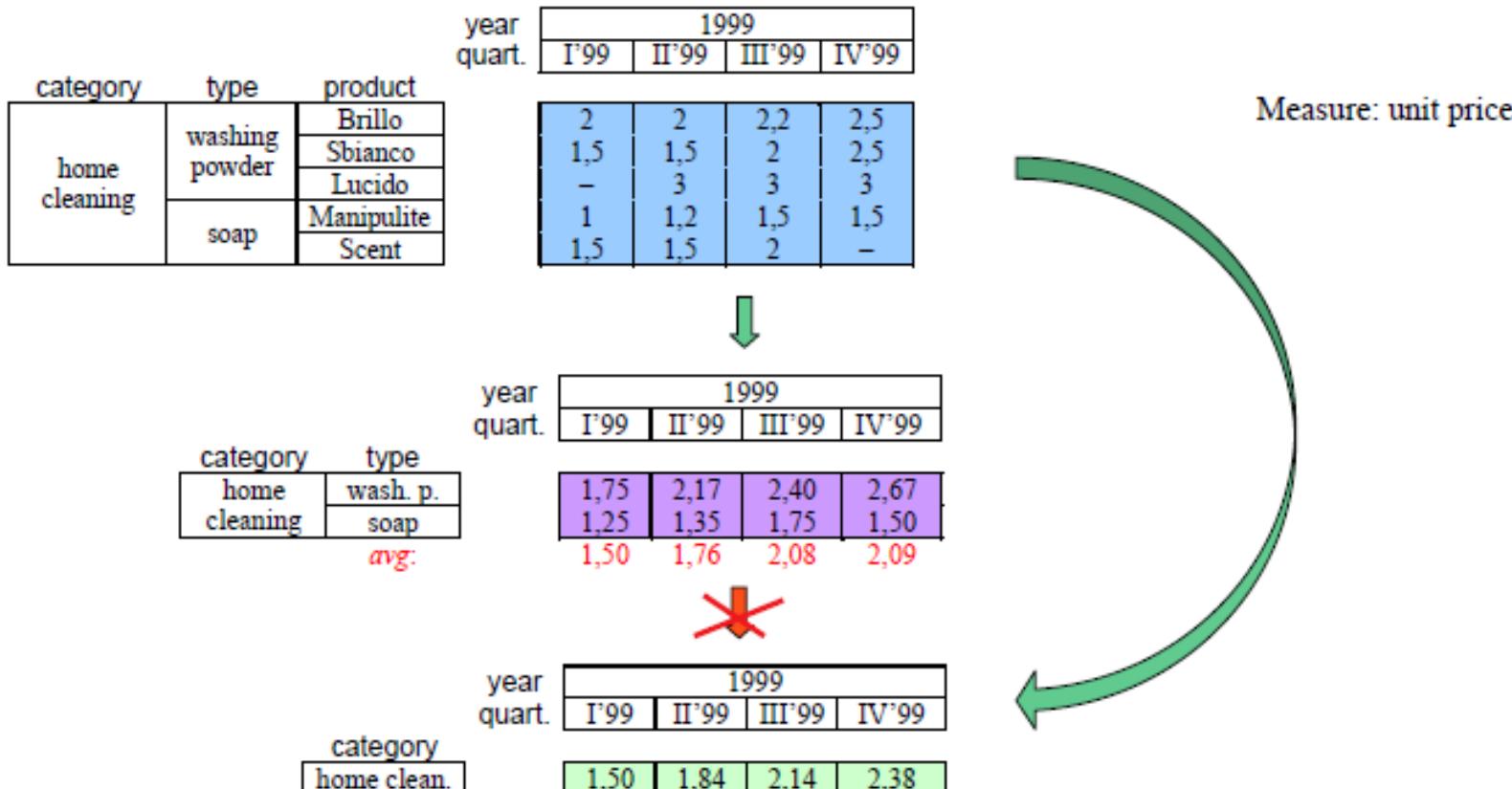
year 1999 2000

year 1999 2000

category		
home clean.	870	760
food	1030	975

category	type
home	washing p
cleaning	soap
food	milk
	soda

Operatore algebrico: AVG



- Esempio della misura *prezzo unitario* di VENDITA, aggregabile tramite AVG su tutte le dimensioni: si nota immediatamente che la corretta aggregazione sul pattern **{categoria, trimestre}** non è ottenibile dall'aggregazione sul pattern **{tipo, trimestre}** a meno di aggiungere una nuova misura che conti il numero di eventi primari che definiscono ciascun evento secondario.

Aggregazione di misure non-additive (2)

- **Caso 2:** operatore di aggregazione differenti lungo le diverse dimensioni.
- Eventi primari per uno schema dell'inventario con dimensioni magazzino e data sulla misura livello in riferimento al singolo prodotto.

		mese	marzo 1999								
		data	1/3/99	2/3/99	3/3/99	4/3/99	5/3/99	6/3/99	7/3/99	8/3/99	9/3/99
città	magazzino										
Roma	RM-Eur		10	10	8	4	20	20	15	15	12
	RM-Centro		5	4	4	4	2	2	2	10	10
	RM-Trastevere		14	14	14	12	20	20	20	20	16
Milano	MI-Est		4	2	2	2	10	10	10	8	8
	MI-Ovest		4	20	20	15	15	12	12	10	9

Aggregazione di misure non-additive (3)

- Eventi secondari sul pattern {mese, magazzino}.
 - Aggregazione: min

mese marzo 1999

città	magazzino	
Roma	RM-Eur	4
	RM-Centro	2
	RM-Trastevere	12
Milano	MI-Est	2
	MI-Ovest	4

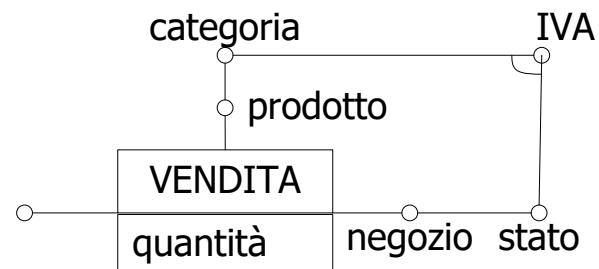
- Eventi secondari sul pattern {data, città}.

- Aggregazione: sum

data	1/3/99	2/3/99	3/3/99	4/3/99	5/3/99	6/3/99	7/3/99	8/3/99	9/3/99
mese	marzo 1999								
città	Roma	29	28	26	20	42	42	37	45
	Milano	8	22	22	17	25	22	22	18

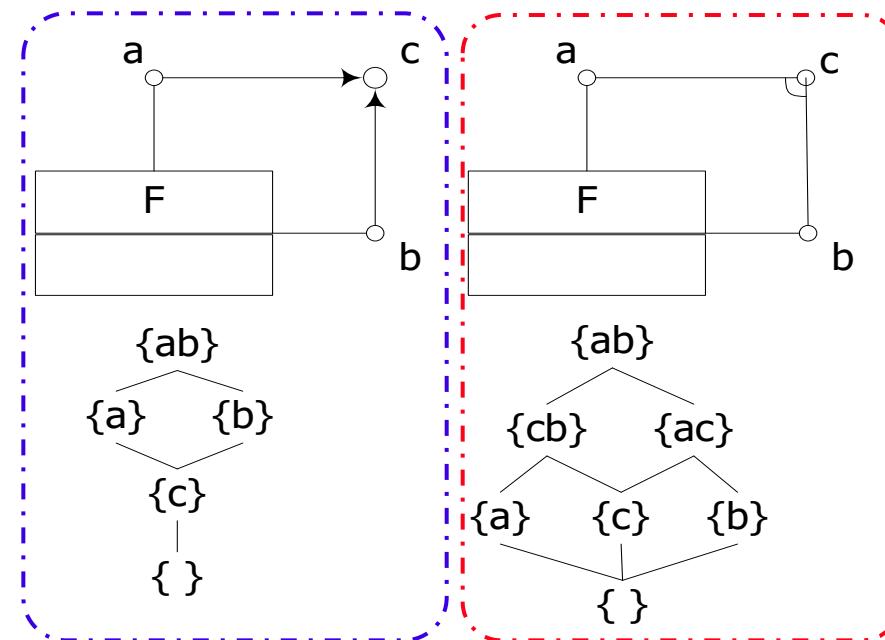
Aggregazione in presenza di convergenze e attributi cross-dimensional

- Una convergenza nello schema di fatto è del tutto trasparente ai fini dell'aggregazione.
- Per verificare la semantica dell'aggregazione in presenza di attributi cross-dimensional dobbiamo risalire agli eventi primari che includono l'attributo cross-dimensionale.
- L'attributo cross-dimensionale IVA determinato da *categoria* e *stato*;
 - ciascun evento primario è associato ad un prodotto e ad un negozio (quindi ad una categoria e ad uno stato).
 - Essendo definito univocamente un valore di IVA per ogni evento primario, gli eventi secondari sui pattern che includono IVA risultano immediatamente determinati.



Aggregazione in presenza di convergenze e attributi cross-dimensional

- Differenza tra convergenza e attributo cross-dimensionale:



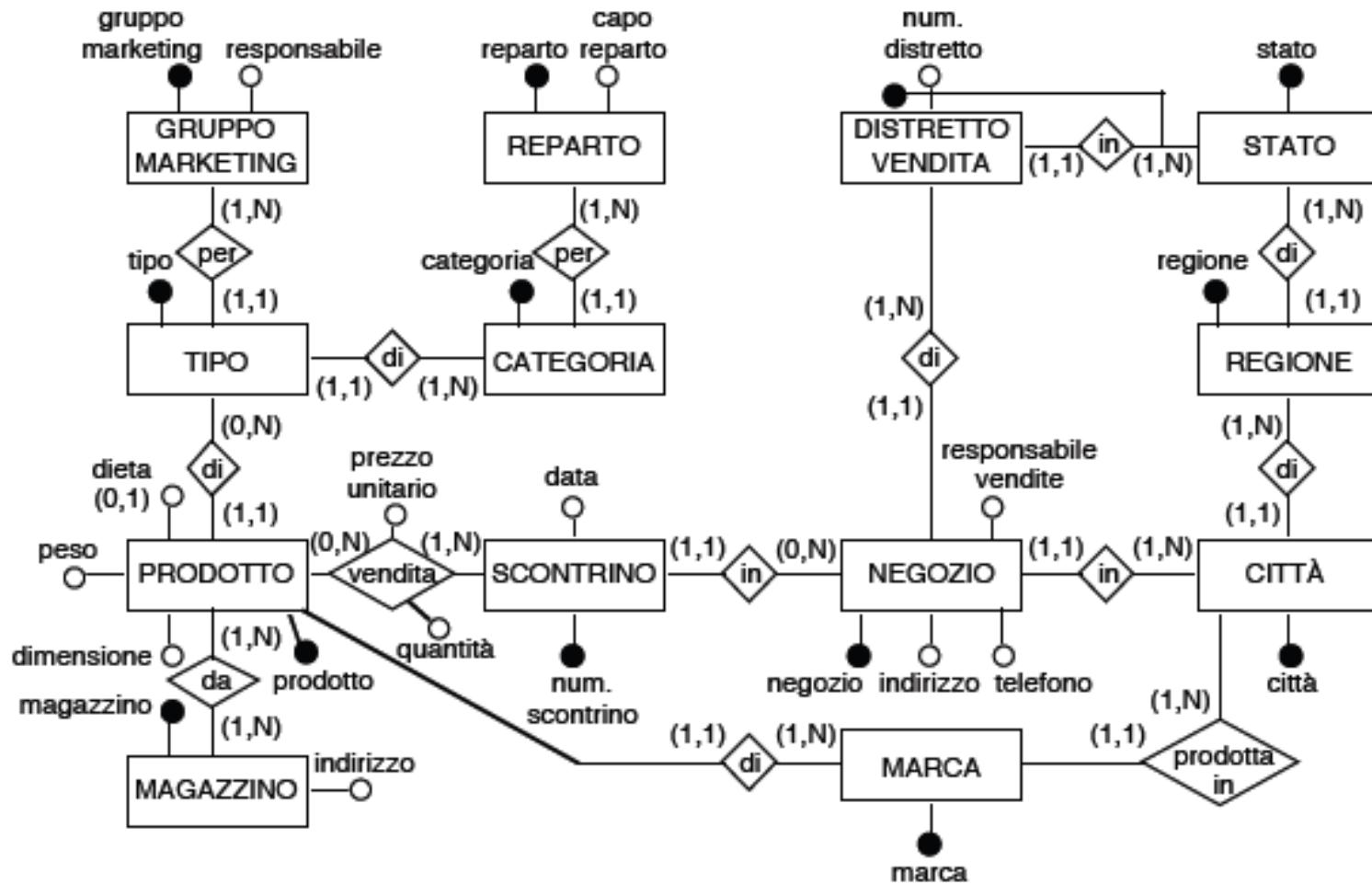
- Reticoli di roll-up in presenza di convergenza e cross-dimensionalità.



Progettazione concettuale

Esempio delle vendite

(dopo la fase di integrazione)



Progettazione concettuale

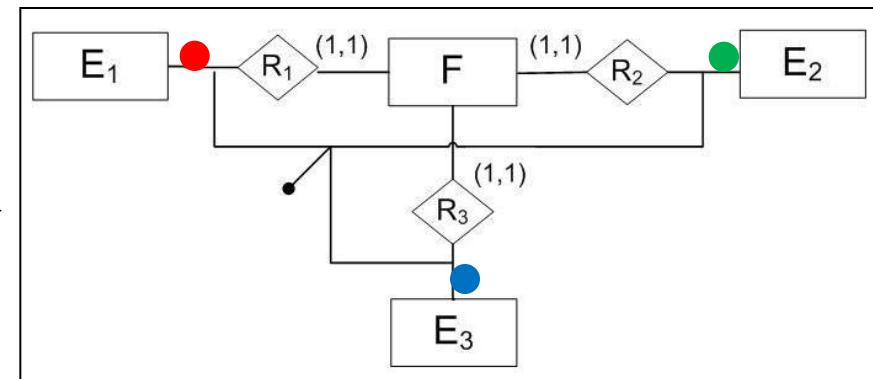
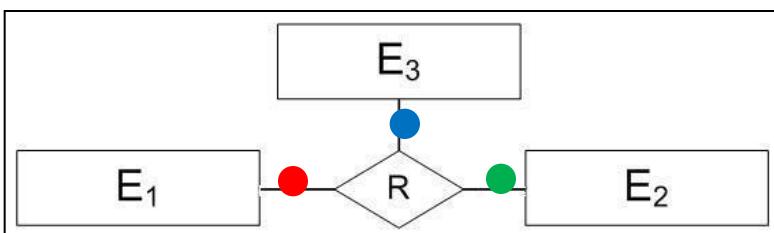
- **Progettazione concettuale guidata dai dati.**
- La tecnica per la progettazione concettuale di un Data mart a partire dalle sorgenti operazionali, secondo il DFM, consiste nei seguenti passi:
 1. Definizione dei fatti.
 2. Per ciascun fatto.
 - a. Costruzione dell'albero degli attributi.
 - b. Potatura e innesto dell'albero degli attributi.
 - c. Definizione delle dimensioni/misure.
 - d. Creazione dello schema di fatto.

Definizione dei fatti

- I fatti sono concetti di interesse primario per il processo decisionale.
- In uno schema ER un fatto può corrispondere a:
 - un'**entità** F;
 - un'**associazione** n-aria R tra le entità E₁, E₂, ..., E_n

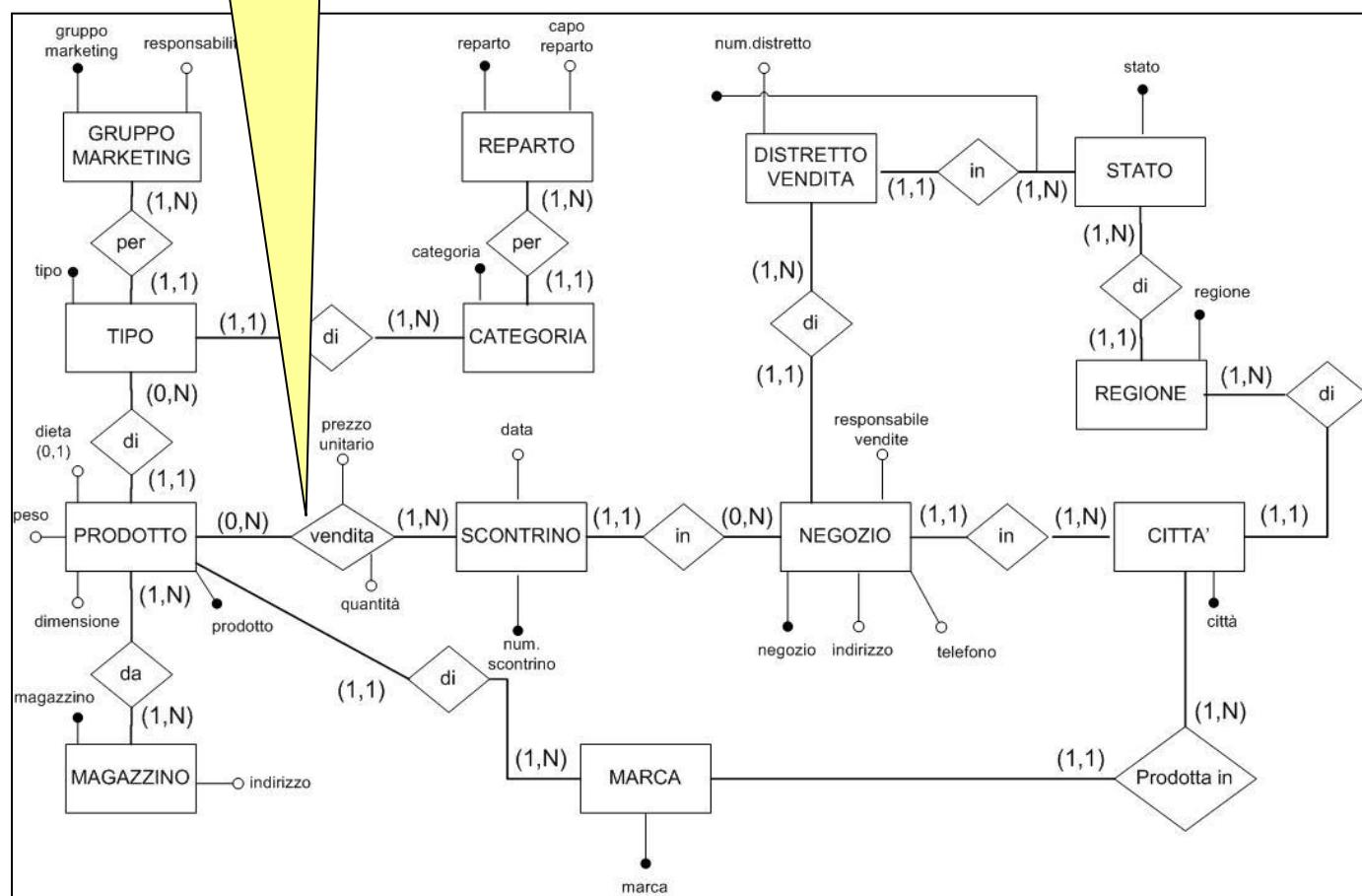
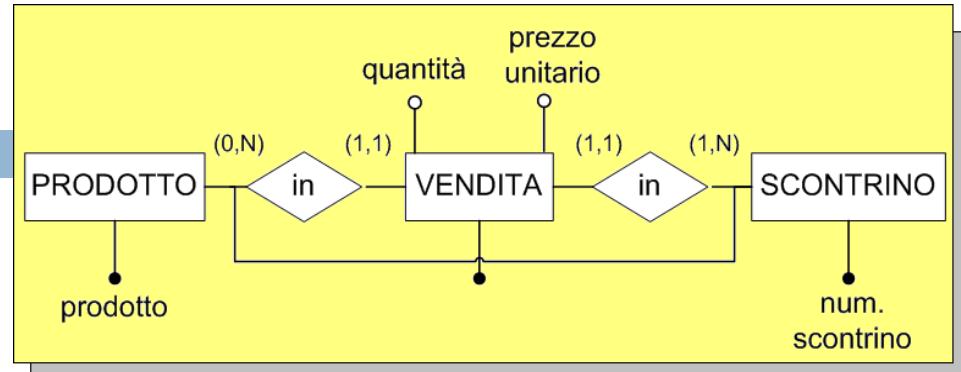
Processo di reificazione

- Se il fatto corrisponde ad un'**entità** del modello ER, non è richiesta alcuna modifica.
- Se il fatto corrisponde ad una **relazione** n-aria R, è preferibile trasformare R in un'entità F che possiede tutti gli attributi della relazione e la cui chiave è data dall'unione delle chiavi delle singole entità coinvolte.
 - Ciascuno dei rami di R viene sostituito con un'associazione binaria R_i tra F ed E_i tale che:
 - $\min(F, R_i) = \max(F, R_i) = 1$
 - $\min(E_i, R_i) = \min(E_i, R)$
 - $\max(E_i, R_i) = \max(E_i, R)$



Esempio di reificazione

Andando a reificare
l'associazione **vendita**
otteniamo . . .



Linee guida nella scelta dei fatti

- Le entità che rappresentano archivi aggiornati frequentemente (es., VENDITA) sono buoni candidati per la definizione dei fatti.
- Le entità che rappresentano proprietà strutturali del dominio, corrispondenti ad archivi quasi statici (es., NEGOZIO e CITTA'), non lo sono.
- In realtà, tale regola non è sempre valida in quanto la scelta del fatto dipende in maniera significativa sia dal dominio applicativo che dal tipo di analisi che l'utente intende eseguire.

Linee guida nella scelta dei fatti (2)

- Ciascun fatto identificato sullo schema sorgente diviene la radice di un differente schema di fatto.
- Nel caso in cui diverse entità siano candidate ad esprimere lo stesso fatto, conviene sempre scegliere come fatto **F** l'entità, a partire dalla quale, è possibile costruire l'albero che include il maggior numero di attributi.

Costruzione albero degli attributi

- **Albero degli attributi:** data un'entità **F** designata come fatto, si definisce albero degli attributi quello che soddisfa i seguenti requisiti:
 - Ogni vertice corrisponde ad un attributo semplice o composto dello schema sorgente.
 - La radice corrisponde all'identificativo di **F**.
 - Per ogni vertice **v**, l'attributo corrispondente determina funzionalmente tutti gli attributi che corrispondono ai discendenti di **v**.

Algoritmo per la costruzione dell'albero degli attributi

```
root = nuovoVertice(ident(F)); /* ident(F) è l'identificatore di F, la radice dell'albero è etichettata con l'identificatore dell'entità scelta come fatto */
traduci(F,root);

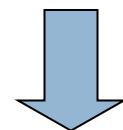
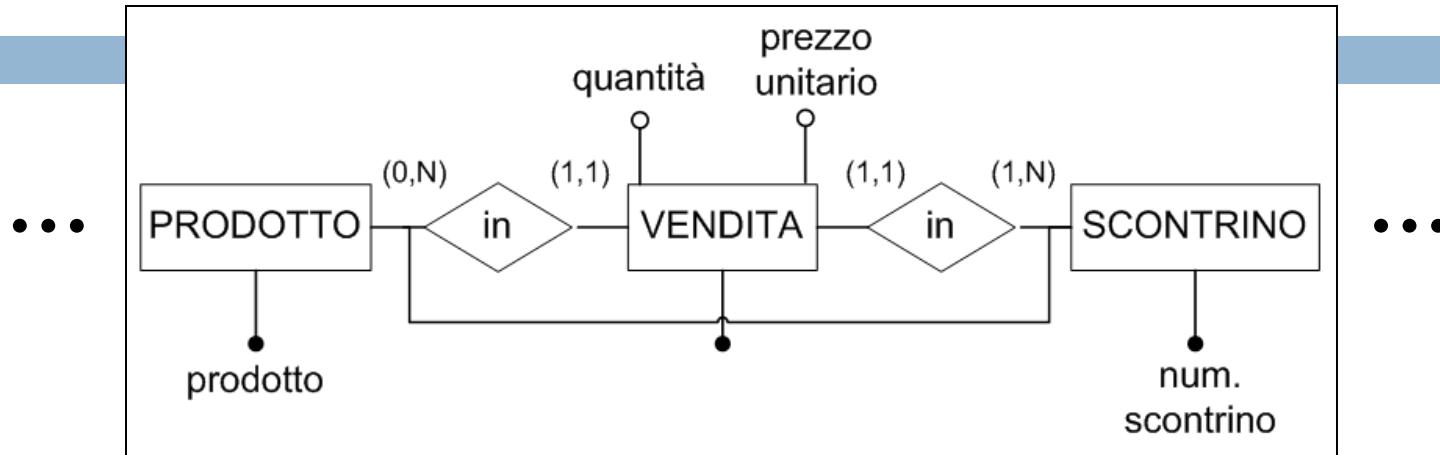
procedura traduci(E,v);
//E è l'entità corrente dello schema sorgente, v il vertice corrente dell'albero
{
    per ogni attributo a di E tale che a!=ident(E)
        aggiungiFiglio(v, nuovoVertice(a))
        // aggiunge al vertice v un figlio a

    per ogni entità G connessa ad E da una associazione R tale che max(E,R)=1
    {
        per ogni attributo b di R
            aggiungiFiglio(v, nuovoVertice(b));
            // aggiunge al vertice v un figlio b
        prossimo = nuovoVertice(ident(G));
        //crea un nuovo vertice con il nome dell'identificatore di G...
        aggiungiFiglio(v, prossimo);
        // ... lo aggiunge a v come figlio ...
        traduci(G, prossimo);
        //... e innesca la ricorsione
    }
}
```

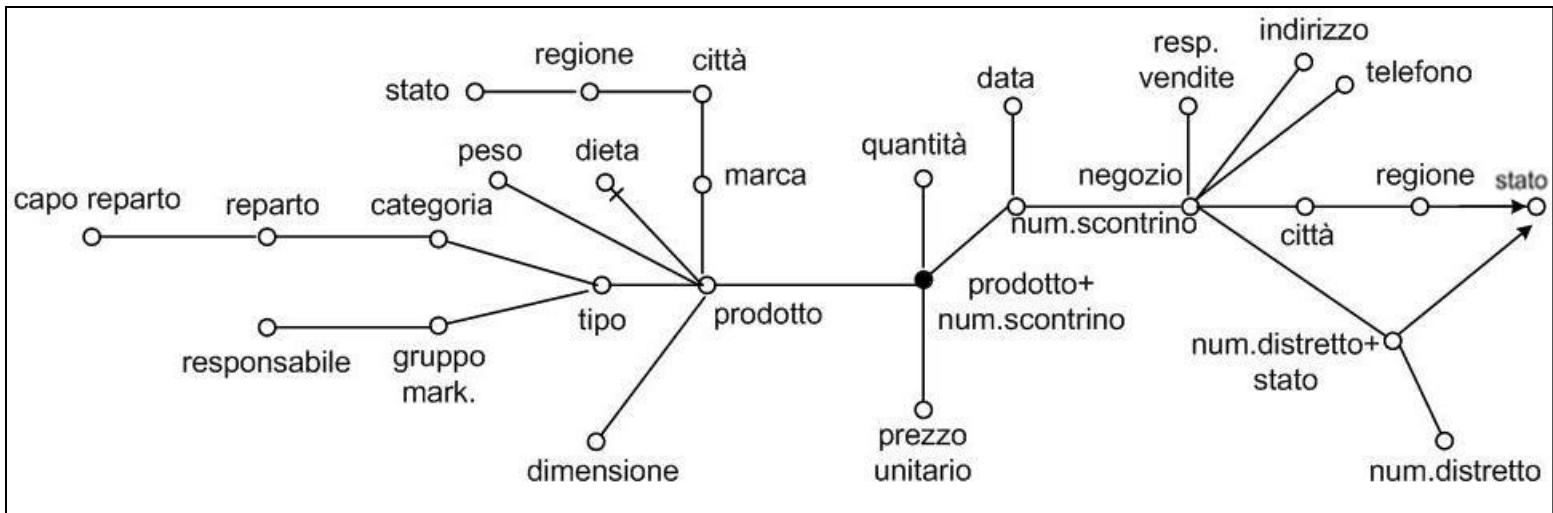
Algoritmo per la costruzione dell'albero degli attributi (2)

- La procedura proposta naviga ricorsivamente le dipendenze funzionali espresse dagli identificatori e dalle associazioni ...-a-1 dello schema ER.
 - L'entità a partire dalla quale viene innescato il processo è quella scelta come fatto.
- Quando si esamina un'entità E si crea nell'albero un vertice **v** corrispondente all'identificatore di E, e gli si aggiunge un vertice per ogni altro attributo di E.
- Per ogni associazione R da E verso un entità G, con cardinalità massima 1, si aggiungono a **v** tanti figli quanti sono gli attributi di R, per poi ripetere il procedimento per G.

Applicazione dell'algoritmo



Albero degli attributi ottenuto a partire dallo schema reificato (*include tutto lo schema ER*).

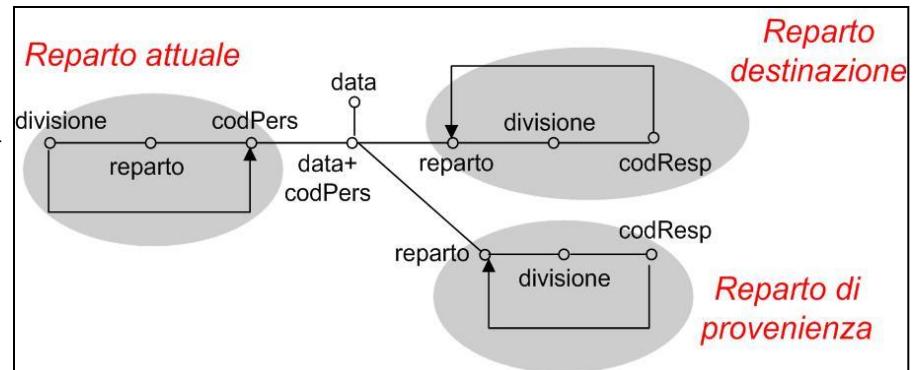
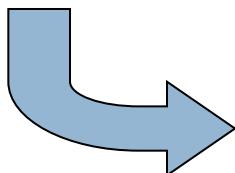
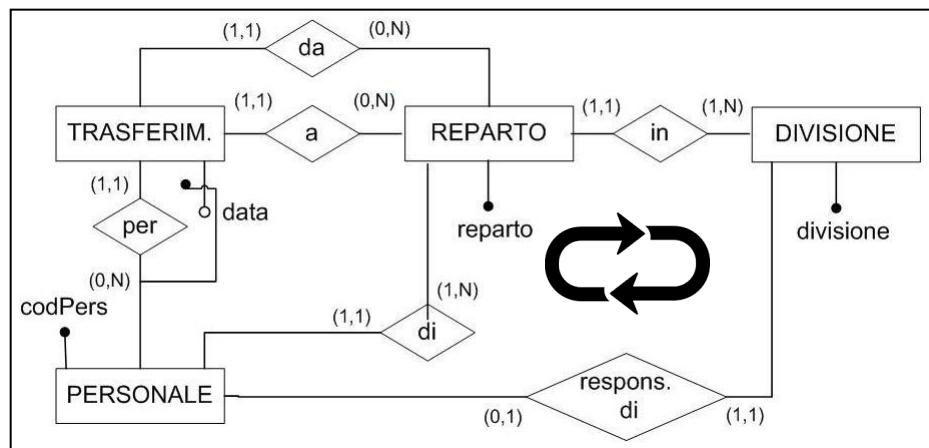
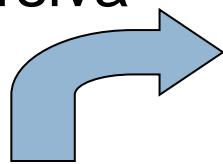


Algoritmo per la costruzione dell'albero degli attributi (3)

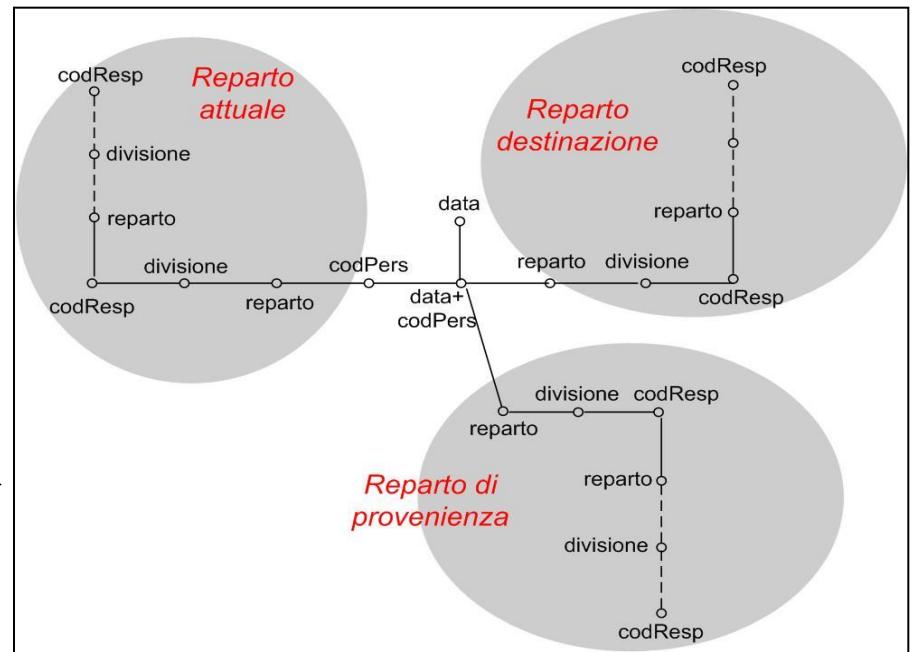
- **Eccezioni:** l'applicazione dell'algoritmo è condizionata da specifiche caratteristiche strutturali dello schema sorgente:
 1. Presenza di un ciclo di associazioni ...-a-1 (loop).
 2. Raggiungimento della medesima entità E attraverso cammini differenti.
 3. Presenza di associazioni ...-a-molti e di attributi multipli.
 4. Presenza di associazioni o attributi opzionali.
 5. Presenza di associazioni n-arie.
 6. Presenza di gerarchie di specializzazione.
 7. Presenza di attributi composti.

1-Presenza di un ciclo di associazioni molti a uno/uno a uno

Gerarchia ricorsiva

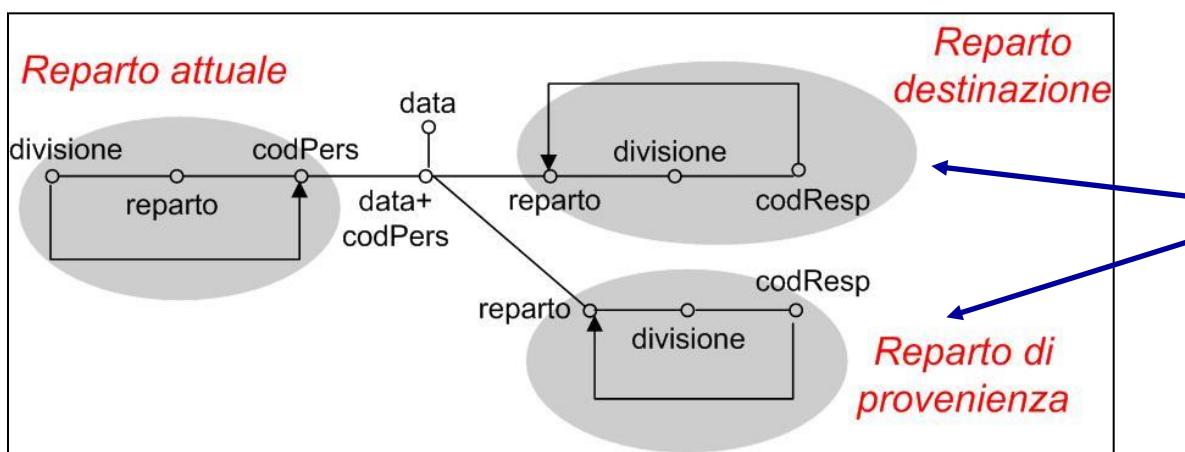


Interruzione del ciclo



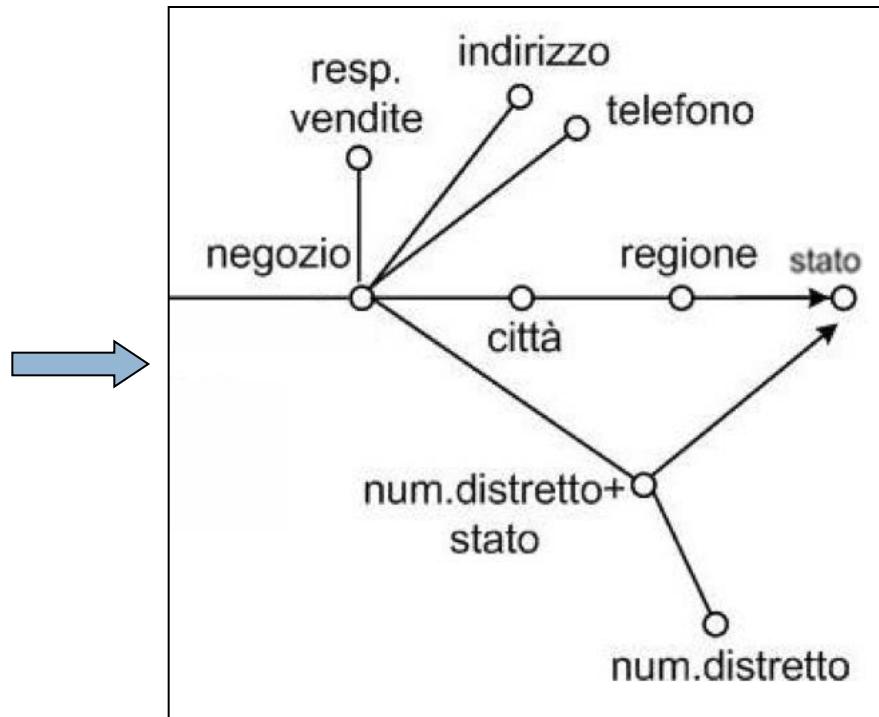
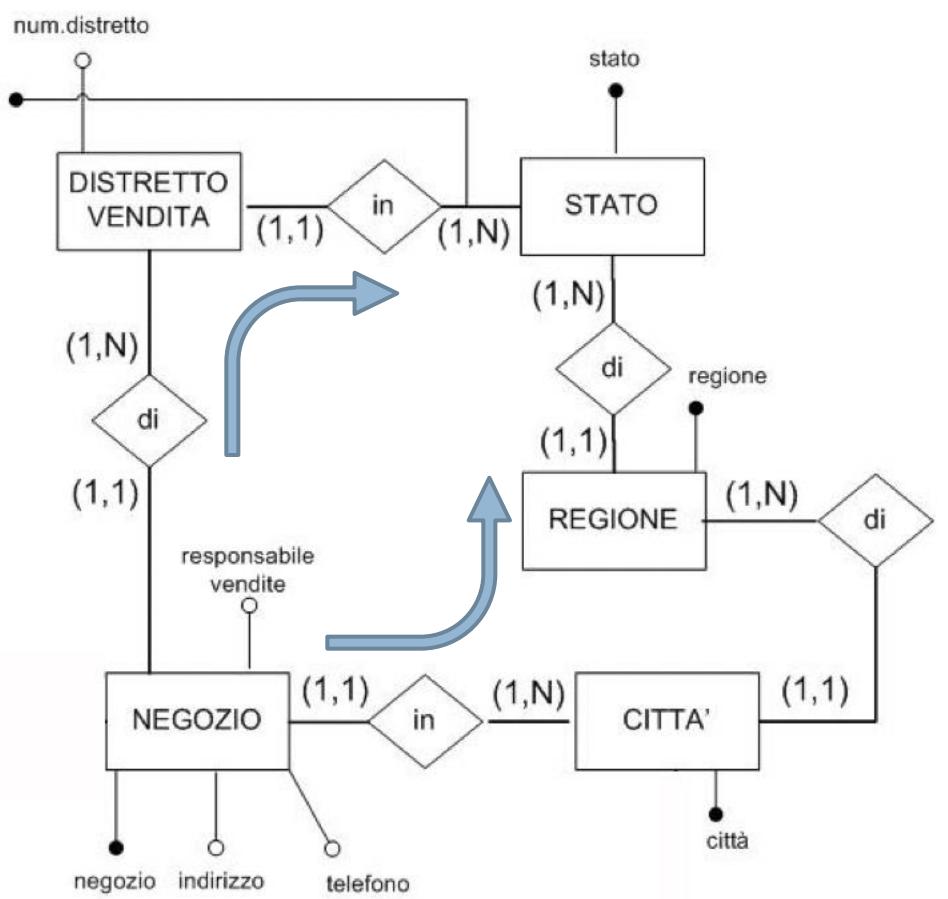
2-Raggiungimento della stessa entità E attraverso cammini differenti

- Se si raggiunge due volte la stessa entità E attraverso cammini differenti, vengono generati nell'albero due vertici omologhi v' e v'' .
 - ▣ Se ogni istanza di F determina E (i.e., $F \rightarrow E$) indipendentemente dal cammino seguito, allora v' e v'' possono coincidere (*si genera una convergenza*).



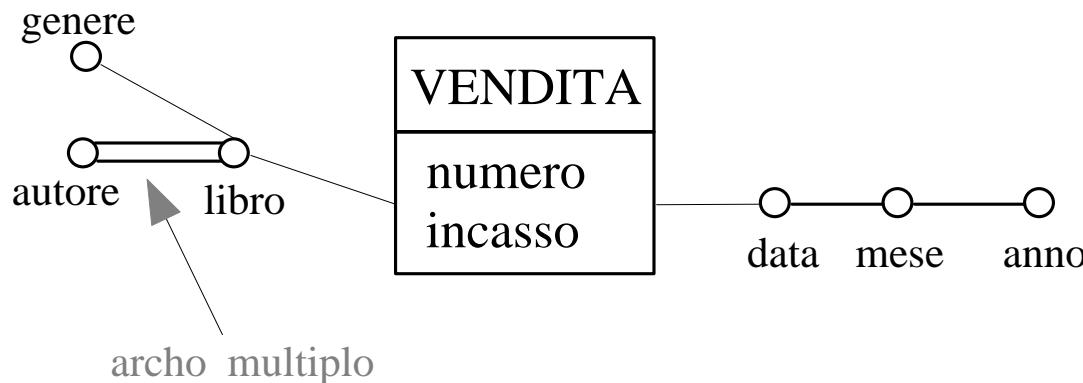
Il reparto di provenienza e quello di destinazione sono necessariamente distinti, perché sarebbe scorretto creare una convergenza

2-Raggiungimento della stessa entità E attraverso cammini differenti (2)



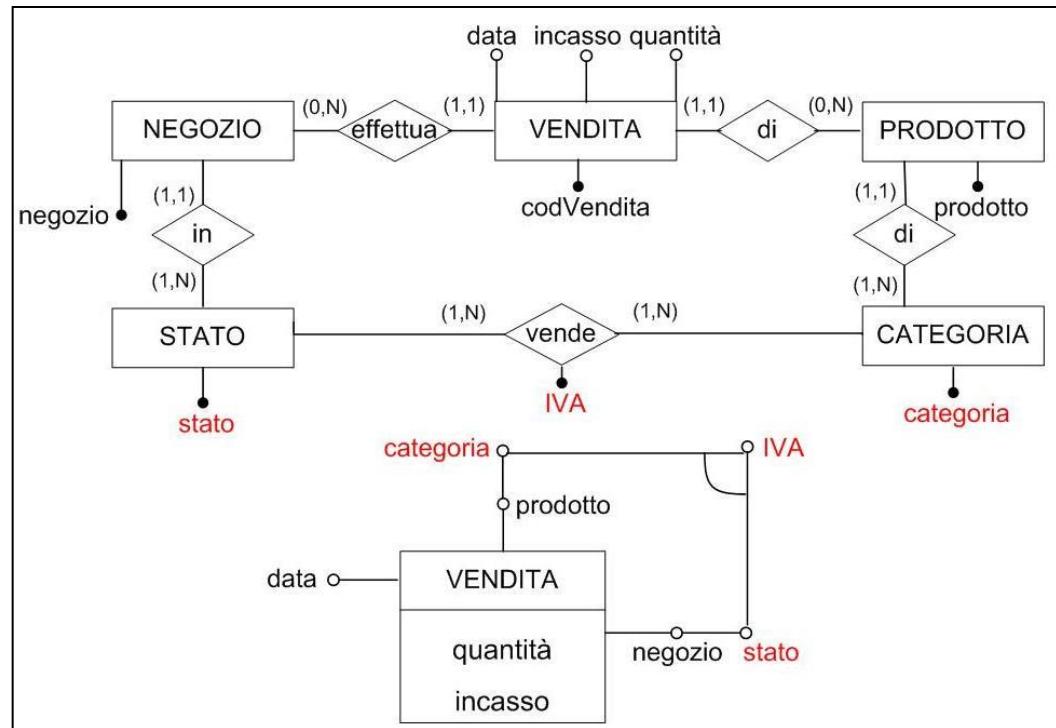
3-Presenza di associazioni ...-a-molti e di attributi multipli

- Eventuali associazioni ...-a-molti ($\max(E,R) > 1$) e attributi multipli presenti nello schema ER non vengono inseriti automaticamente nell'albero degli attributi:
 - Esse possono generare attributi cross-dimensional o archi multipli disegnati manualmente dal progettista sullo schema di fatto al termine della progettazione concettuale.



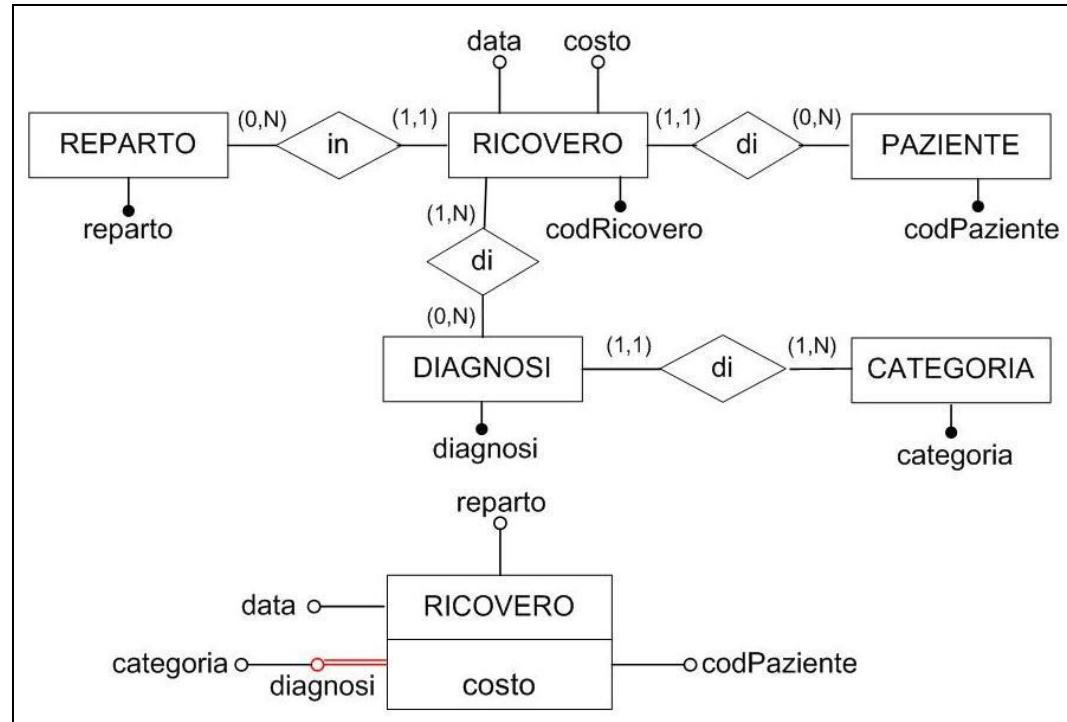
Generazione dello schema di fatto: Attributi cross-dimensional

- Un **attributo cross-dimensionale**:
- Un attributo cross-dimensionale corrisponde in genere a un attributo posto su un'associazione multi-a-molti R dello schema ER;
- I suoi padri nello schema di fatto corrisponderanno allora agli identificatori delle entità coinvolte in R.



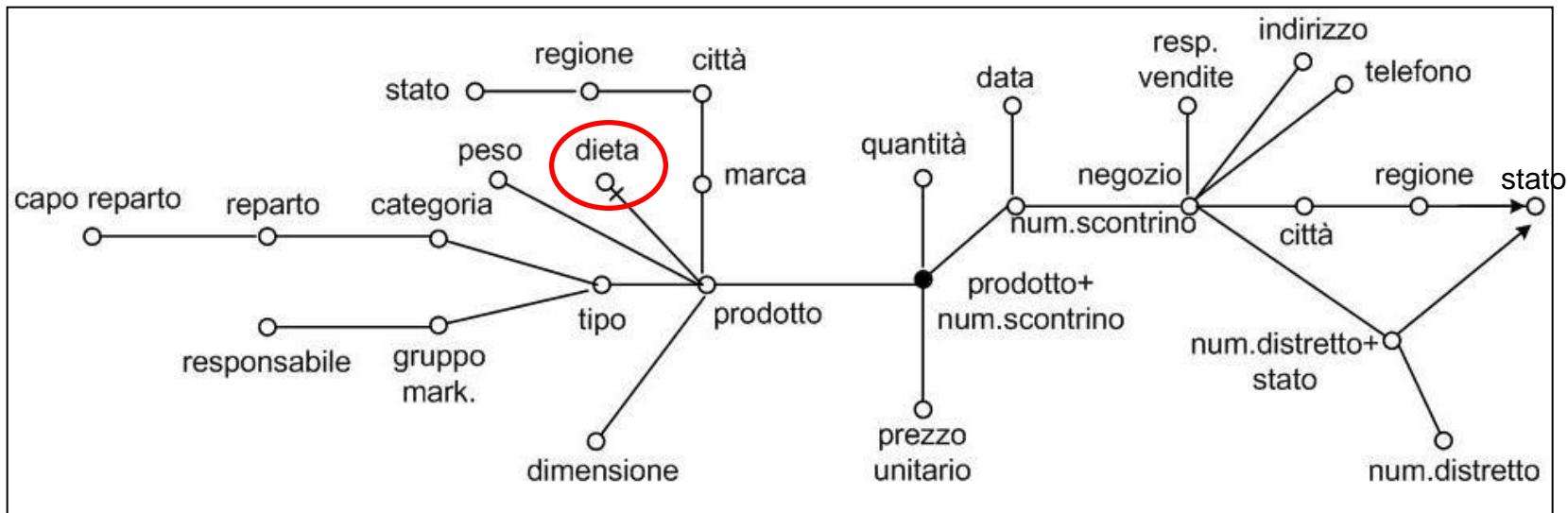
Generazione dello schema di fatto: Archi multipli

- Un **arco multiplo** corrisponde ad un'associazione R ...-a-molti da un'entità E ad un'entità G.
 - Nello schema di fatto esso potrà connettere l'identificatore di E con un attributo di R o di G.



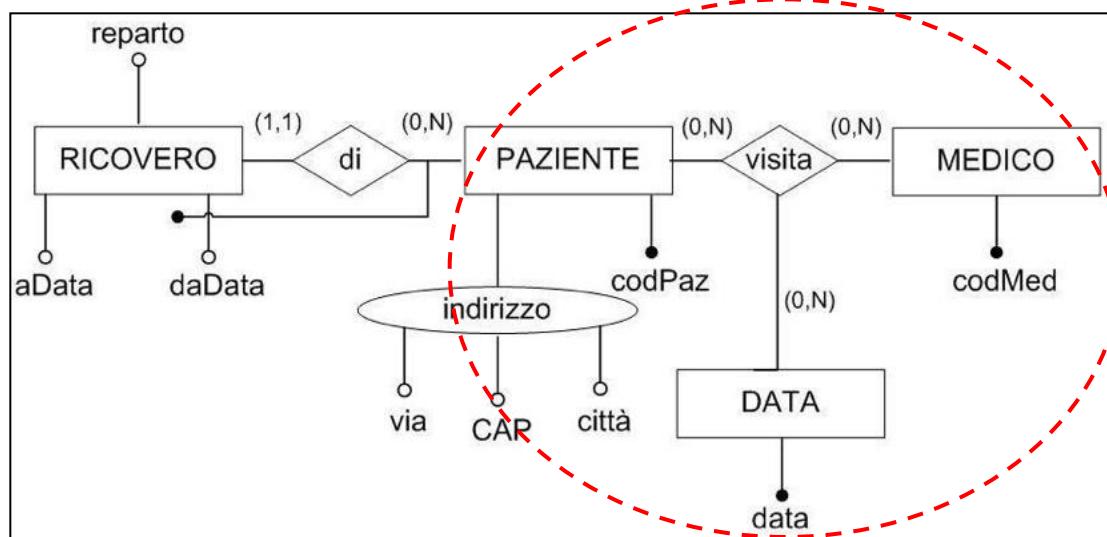
4-Presenza di associazioni o attributi opzionali

- Gli **attributi opzionali** ($\min(E,R) = 0$) portano a collegamenti opzionali.
- L'esistenza di un collegamento opzionale deve essere sottolineata nell'albero degli attributi con un trattino sugli archi corrispondenti ad associazioni o attributi opzionali nello schema ER:

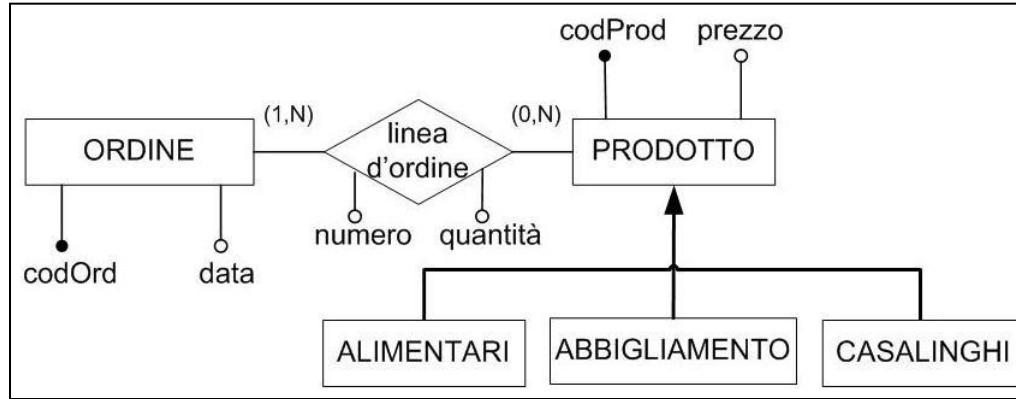


5-Presenza di associazioni n-arie

- Eventuali **associazioni n-arie** saranno trasformate in n associazioni binarie attraverso il processo di reificazione.
- Molte delle associazioni n-arie hanno molteplicità massima maggiore di 1 sui rami (dunque non sono inserite dall'algoritmo):
 - Questa situazione porta ad n-associazioni binarie 1-a-molti che non possono essere inserite automaticamente nell'albero degli attributi.

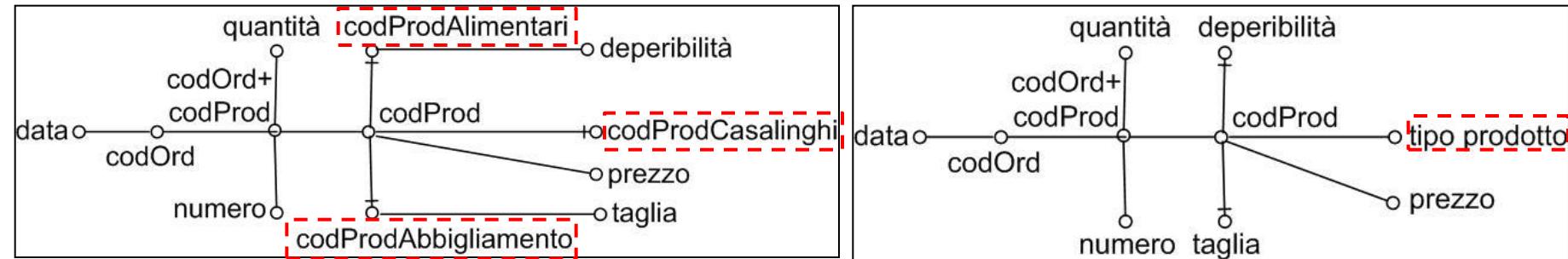


6-Presenza di gerarchie di specializzazione



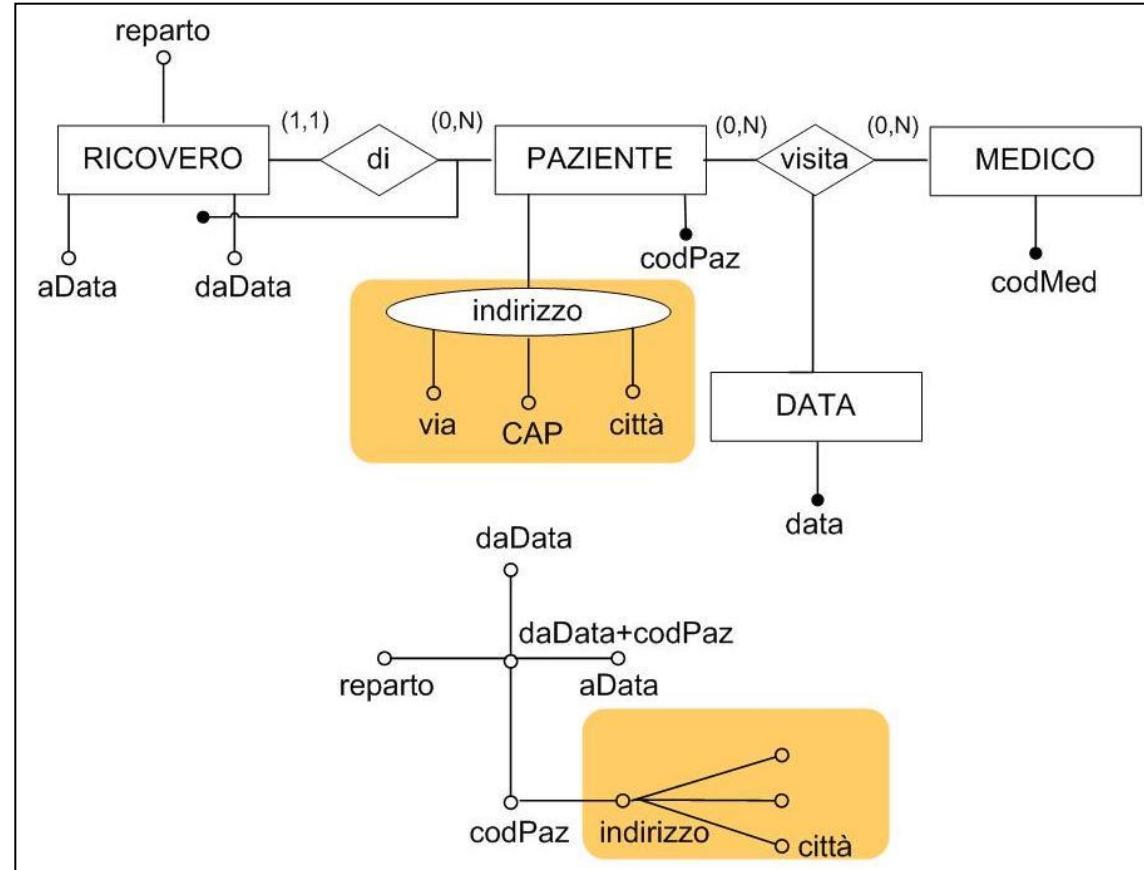
Le gerarchie dell'ER possono essere trattate dall'algoritmo come delle semplici **associazioni 1-a-1 opzionali** tra la superentità e le sottoentità.

In alternativa, è possibile limitarsi ad aggiungere al nodo corrispondente alla chiave della superentità un figlio che funga da discriminatore tra le diverse sottoentità (sottoentità unite con la superentità).



7-Presenza di attributi composti

- In presenza di un attributo composto **c**, che consiste degli attributi semplici a_1, \dots, a_n , tale attributo viene inserito nell'albero degli attributi come un vertice **c** con figli a_1, \dots, a_n .

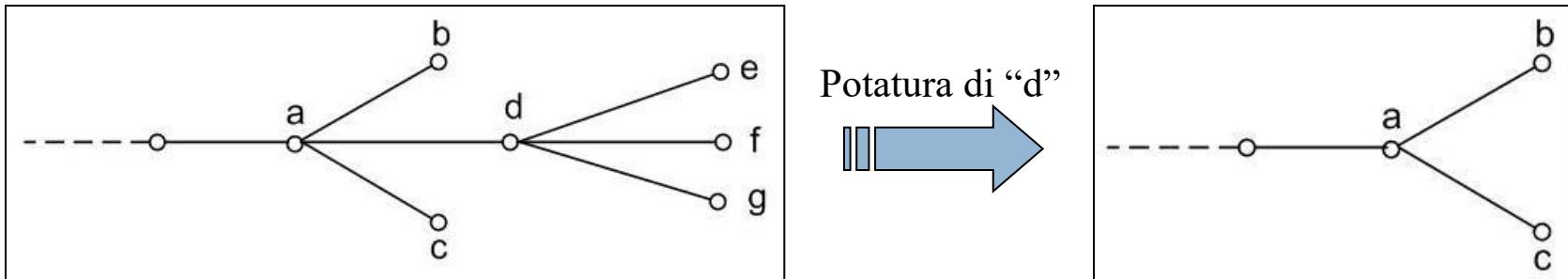


Potatura e innesto dell'albero degli attributi

- In genere, non tutti gli attributi dell'albero sono di interesse per il Data mart.
- **Es:**
 - Il numero di fax di un cliente difficilmente potrà rivelarsi utile ai fini decisionali per cui il Data mart è progettato.
- È dunque possibile manipolare l'albero al fine di eliminare e/o aggiungere livelli di dettaglio.

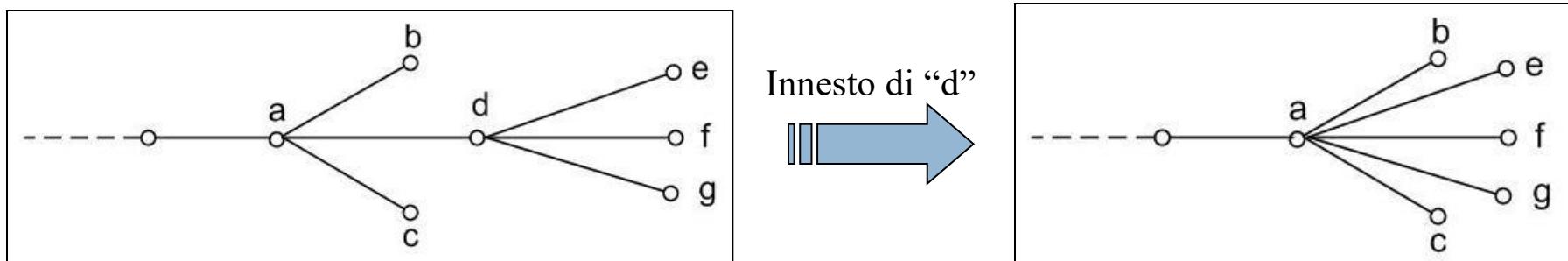
Potatura e innesto dell'albero degli attributi (2)

- **Potatura di un vertice v**: si effettua eliminando l'intero sottoalbero radicato in v.
 - Gli attributi eliminati non verranno inclusi nello schema di fatto.
 - Non potranno essere usati per aggregare i dati.

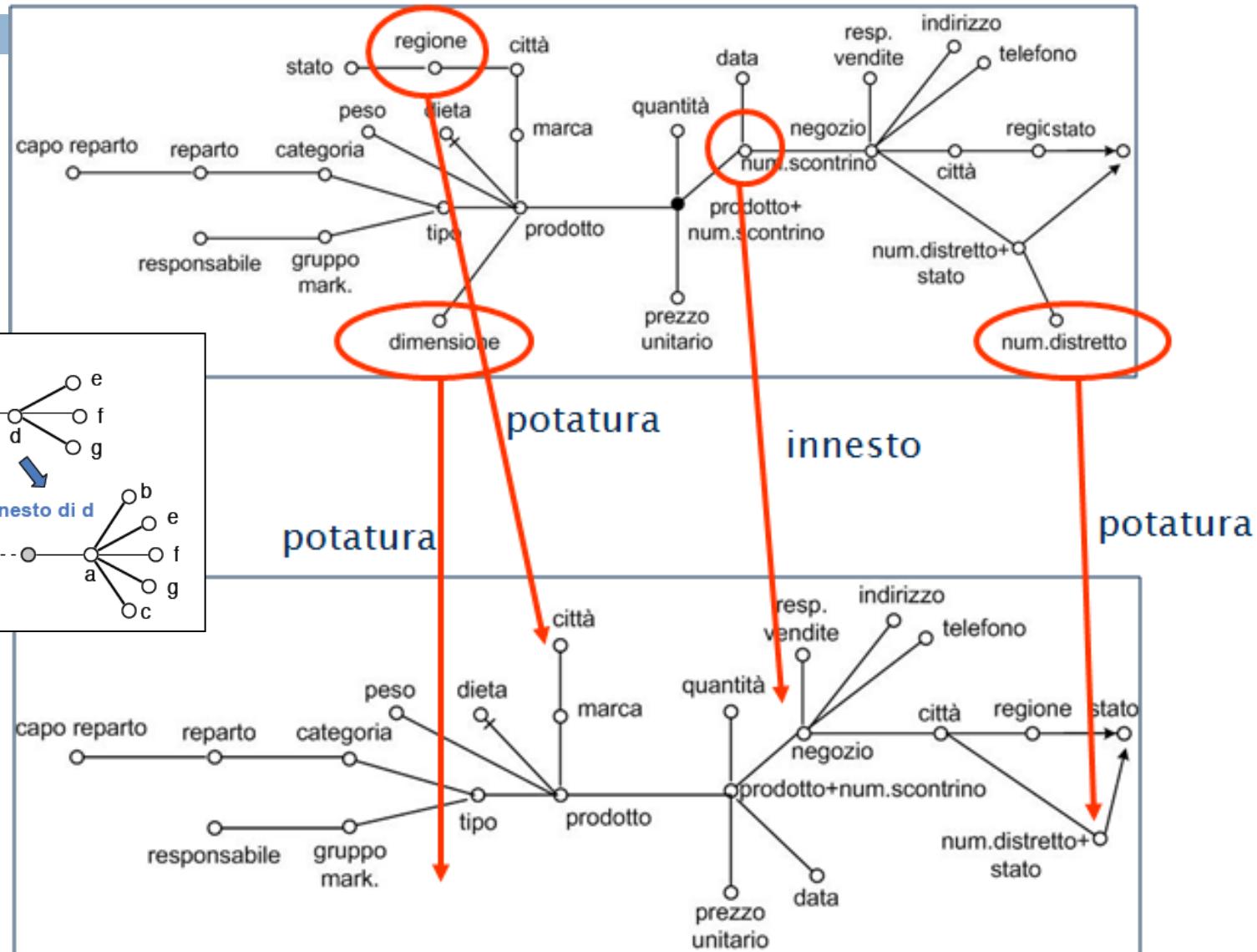


Potatura e innesto dell'albero degli attributi (3)

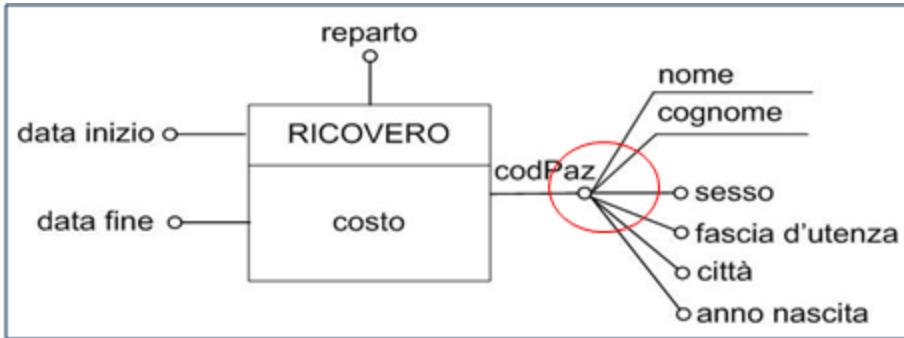
- **Innesto di un vertice v**: viene utilizzato quando, sebbene un vertice esprima un'informazione non interessante, è necessario mantenere nell'albero i suoi discendenti (che verranno collegati direttamente al padre del vertice da innestare):



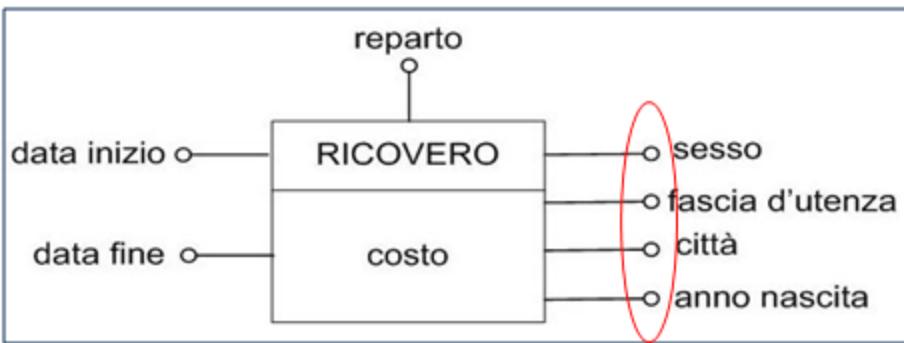
Esempi di potatura ed innesto



Definizione delle dimensioni



Nel primo schema la dimensione codPaz è mantenuta (*grana transazionale*)



Nel secondo schema abbiamo rinunciato alla granularità del singolo paziente innestando codPaz e introducendo le dimensioni sesso, fascia d'utenza, città e anno di nascita (*grana temporale*)

- Le dimensioni devono essere scelte nell'albero degli attributi tra i vertici figli della radice; possono corrispondere ad attributi discreti o a intervalli di valori di attributi discreti o continui.

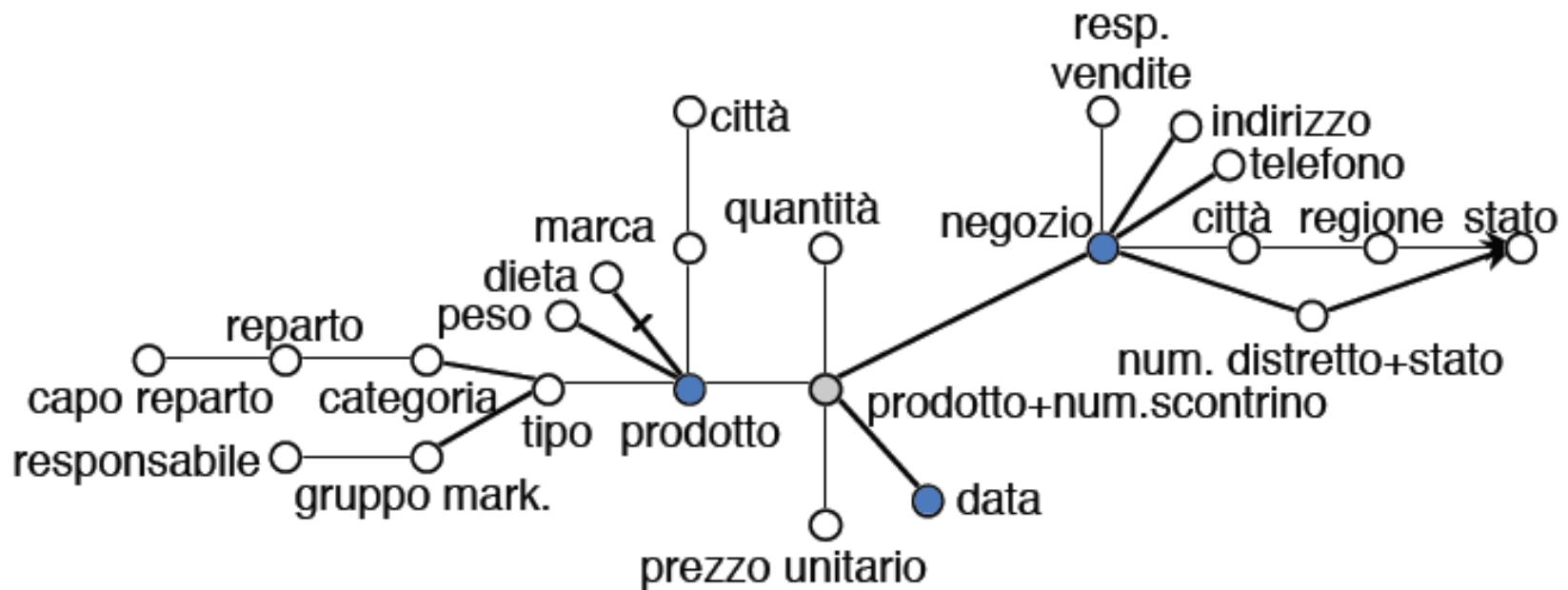
- Es:** in un DW in ambito sanitario, un classico problema riguarda il mantenimento o meno della granularità del paziente.

Definizione delle dimensioni (2)

- In un DW non siamo interessati, di solito, ad interrogazioni di natura operazionale (che sono prerogativa dei DB relazionali).
 - ▣ Si preferisce in generale una grana temporale.
- Tuttavia, se è necessario mantenere una granularità massima, si procede nel seguente modo:
 - ▣ Duplicare il vertice radice nell'albero degli attributi; il nuovo vertice sarà collegato alla vecchia radice tramite un'associazione 1-a-1 e la radice non avrà altri archi uscenti.
 - ▣ Scegliere le dimensioni:
 - Marcare come dimensione l'unico figlio diretto della radice (schema di fatto monodimensionale), oppure
 - trasformare in figli diretti della radice alcuni attributi dell'albero (schema di fatto multidimensionale).

L'esempio delle vendite

□ Definizione delle dimensioni:



Il tempo

- Il tempo è un fattore chiave nella progettazione di un DW.
- Gli schemi sorgente possono essere classificati rispetto al tempo come:
 - **Snapshot**: descrivono lo stato corrente del dominio applicativo; vengono mantenute solo le versioni dei dati correnti che rimpiazzano continuamente le precedenti. Il tempo viene aggiunto manualmente come dimensione nello schema di fatto.
 - **Storici**: descrivono l'evoluzione del dominio applicativo durante un intervallo di tempo; anche le vecchie versioni dei dati continuano ad essere mantenute. Pertanto, in tal caso il tempo diventa un ovvio candidato alla definizione di una dimensione nello schema di fatto.

Valid Time e Transaction Time

- **Valid time:** istante in cui l'evento si verifica nel mondo aziendale.
- **Transaction time:** istante in cui l'evento è memorizzato nel database.

Uso di valid e transaction time

- Non necessariamente entrambe le coordinate temporali (*valid* e *transaction*) devono essere mantenute.
- La scelta di quale debba essere mantenuta dipende dal tipo di interrogazioni, che possono essere:
 1. *Interrogazioni che richiedono il tempo di validità.*
Es: in quali mesi gli studenti preferiscono iscriversi ad un certo corso.
 2. *Interrogazioni che richiedono il tempo di transazione.*
Es: confrontare il numero totale degli iscritti con quello degli anni precedenti.
 3. *Interrogazioni che richiedono entrambi i tempi.*
Es: stabilire qual è il ritardo medio nella trasmissione di pagamenti.

Modellazione del tempo nel DFM

- A seconda del tipo di interrogazione il tempo viene modellato concettualmente in maniera diversa:
 1. ***Modellazione del solo tempo di validità***: soluzione che riflette la semantica del tempo comunemente adottata negli schemi di fatto.
 - Consente solo interrogazioni del primo tipo non essendo disponibili, prima dell'aggiornamento retrospettivo, i valori che riflettono la situazione reale.
 2. ***Modellazione del solo tempo di transazione***: soluzione sconsigliata se non per i casi in cui il tempo di transazione ha una semantica rilevante all'interno del dominio applicativo.
 3. ***Modellazione di entrambi i tempi***: è la soluzione più generale e consente la formulazione di tutti e tre i tipi di interrogazione.

Definizione delle misure

- Se tra le dimensioni compaiono tutti gli attributi che costituiscono un'entità fatto (schema a grana transazionale), allora le misure corrispondono ad attributi numerici che siano figli della radice.
- Se lo schema è a grana temporale, le misure devono essere definite applicando, ad attributi numerici dell'albero, funzioni di aggregazione (SUM, AVG, MAX, MIN) che operano su tutte le istanze di F corrispondenti a ciascun evento primario:
 - Scegliere come misura un attributo che non è figlio diretto della radice significa rinunciare ad una dipendenza funzionale.

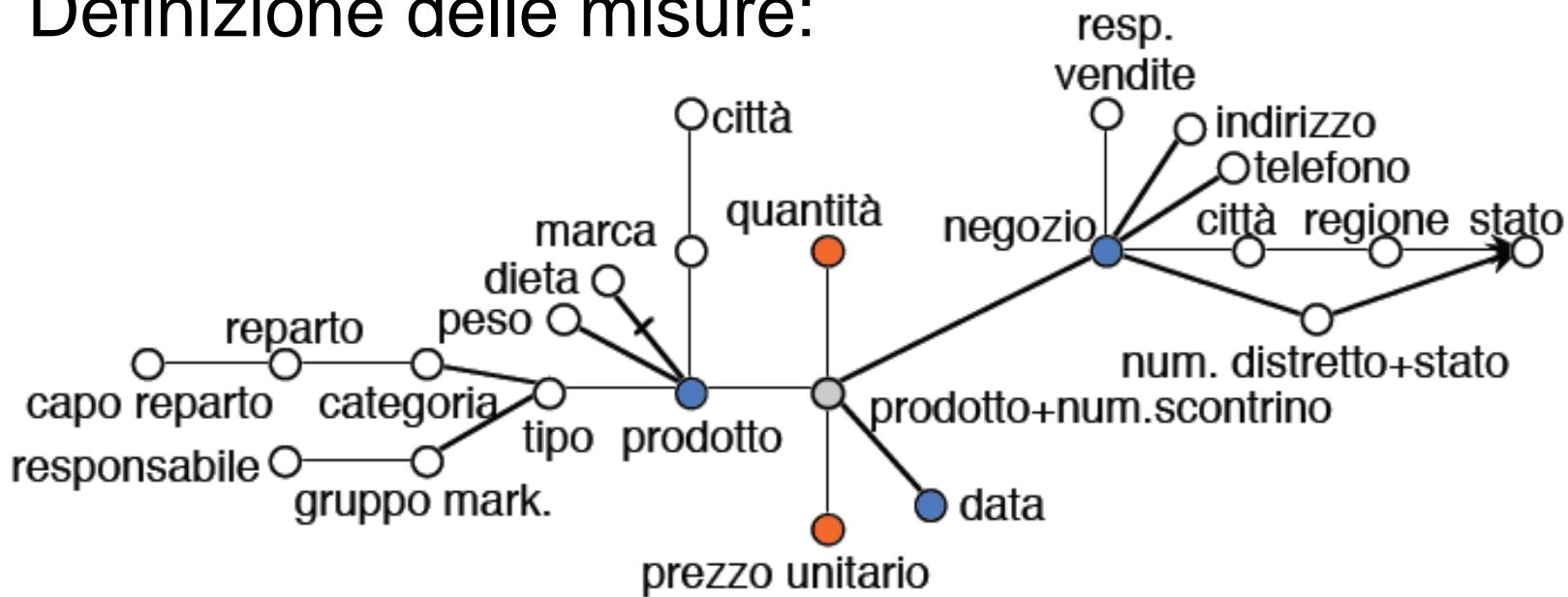
Glossari per il calcolo delle misure

- Occorre definire un **glossario** che associa ciascuna misura ad un'espressione che descrive come essa possa essere calcolata.
- **Es:** il glossario delle vendite potrebbe essere il seguente:

```
quantità venduta = SUM(VENDITA.quantità)
incasso = SUM(VENDITA.quantità * VENDITA.prezzoUnitario)
prezzo unitario = AVG(VENDITA.prezzoUnitario)
num.clienti = COUNT(*)
```
- Se la granularità del fatto è differente da quella dello schema sorgente, può essere utile definire più misure che aggregano lo stesso attributo tramite operatori diversi.

L'esempio delle vendite (2)

□ Definizione delle misure:



GLOSSARIO

quantità venduta = SUM(VENDITA.quantità)

incasso = SUM(VENDITA.quantità*VENDITA.prezzoUnitario)

prezzo unitario = AVG(VENDITA.prezzoUnitario)

num. clienti = COUNT(*)

Creazione dello schema di fatto

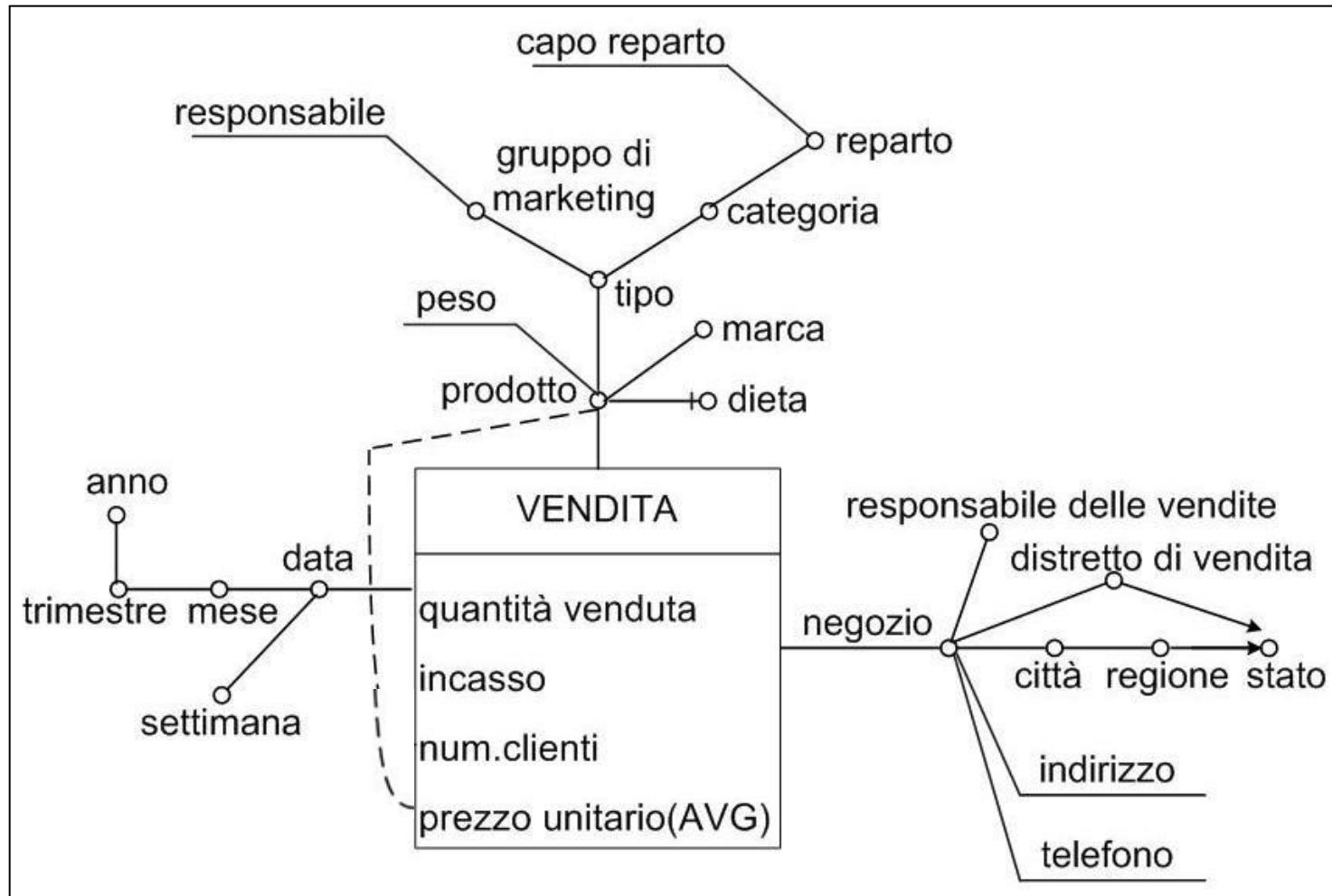
- L'albero degli attributi può ora essere tradotto in uno schema di fatto che include le dimensioni e le misure definite:
 - Le gerarchie corrispondono ai sottoalberi dell'albero degli attributi con radice nelle diverse dimensioni.
 - Il nome del fatto corrisponde al nome dell'entità scelta come fatto.
 - È possibile potare e innestare l'albero per eliminare dettagli inutili.
 - È possibile aggiungere attributi dimensionali definendo opportuni intervalli per attributi numerici (*Es. sulla dimensione tempo*).
 - Gli attributi che non verranno usati per l'aggregazione possono essere contrassegnati come descrittivi;
 - tra questi compariranno in genere anche gli attributi determinati da associazioni 1-a-1 e privi di discendenti.

Creazione dello schema di fatto (2)

- Per quanto riguarda eventuali **attributi alfanumerici** figli della radice ma non prescelti né come dimensioni né come misure:
 - se la granularità degli eventi primari coincide con quella dell'entità F, essi possono essere rappresentati come attributi descrittivi associati direttamente al fatto, di cui descriveranno ciascuna occorrenza;
 - se invece le due granularità sono differenti, essi devono necessariamente essere potati.

Esempio di generazione finale dello schema di fatto

- Si ottiene il seguente schema di fatto:





Modellazione logica

Modellazione logica

- Esistono due distinti modelli logici per rappresentare la struttura multidimensionale dei dati:
 - quello relazionale, che dà luogo ai sistemi **ROLAP** (Relational On-Line Analytical Processing);
 - quello multidimensionale, che dà luogo ai sistemi **MOLAP** (Multidimensional On-Line Analytical Processing).
- La maggior parte del mercato è orientata ai sistemi ROLAP, a causa di un insieme di problemi relativi al sistema MOLAP.

Sistemi MOLAP: Vantaggi

- I sistemi MOLAP memorizzano i dati usando strutture dati multidimensionali, ad esempio vettori multidimensionali in cui ogni elemento è associato ad un insieme di coordinate nello spazio dei valori.
- **Vantaggi:**
 - il tipo di struttura dati utilizzata è la rappresentazione più naturale per i dati di un DW;
 - fornisce ottime prestazioni poiché si presta bene alle esecuzioni delle operazioni OLAP, che sono esprimibili direttamente sulla struttura dati e non hanno bisogno di essere simulate attraverso interrogazioni SQL.

Sistemi MOLAP: Svantaggi

- **Svantaggi:**
 - sparsità dei dati;
 - mancanza di standard, i diversi sistemi hanno in comune solo i principi di base (**Es.** strutture dati), ma non si conoscono i dettagli implementativi;
 - non esistono standard di interrogazione che svolgano un ruolo simile a quello di SQL nei sistemi relazionali.

Sistemi MOLAP: Problema della sparsità

- **Causa:** solo una piccola porzione delle celle di un cubo contiene effettivamente informazioni, le rimanenti corrispondono ad eventi non accaduti. Questo comporta un forte spreco in termini di spazio su disco e di tempo per caricare i dati nel cubo.
- Questo problema non incide sui sistemi ROLAP, poiché essi consentono di memorizzare solo le celle di interesse.

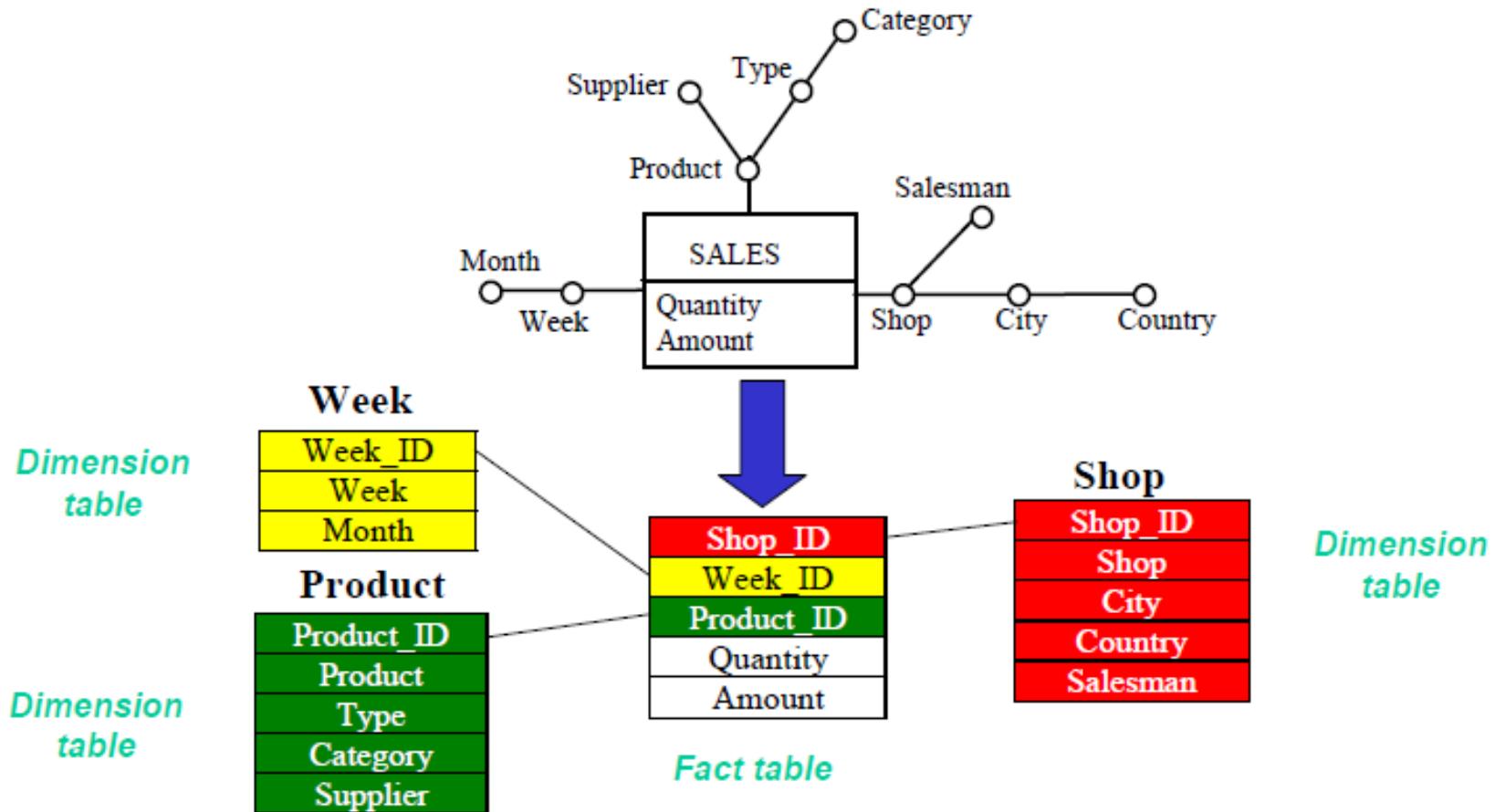
I sistemi ROLAP

- Utilizzano il modello relazionale per la rappresentazione dei dati multidimensionali.
- I motivi che spingono all'adozione di un modello bidimensionale per modellare concetti multidimensionali sono i seguenti:
 - Il modello relazionale è lo standard “*de facto*” dei database, ed è conosciuto dai professionisti del settore.
 - L’evoluzione subita dai DBMS relazionali, da trent’anni sul mercato, li rende strumenti raffinati ed ottimizzati.
 - L’assenza di sparsità dei dati garantisce maggiore scalabilità, fondamentale per database in continua crescita quali i DW.

I sistemi ROLAP: Schema a Stella

- Si usa per la modellazione multidimensionale su sistemi ROLAP.
- È composto da:
 - Un insieme di relazioni $DT_1 \dots DT_n$, chiamate **dimension table**, ciascuna associata ad una dimensione e caratterizzata da una chiave primaria **d_i**, ed un insieme di attributi che descrivono le dimensioni a vari livelli di aggregazione.
 - Una relazione FT, chiamata **fact table**, la cui chiave primaria è data dall'insieme delle chiavi primarie delle dimension table.
 - Inoltre FT contiene un attributo per ogni misura.

Esempio Schema a Stella



- Schema a stella per il fatto delle vendite. La chiave della fact table SALES è costituita dalla combinazione delle chiavi esterne sulle tre dimension table.

Esempio (2)

Shop_ID	Shop	City	Country	Salesman
1	N1	RM	I	R1
2	N2	RM	I	R1
3	N3	MI	I	R2
4	N4	MI	I	R2

Dimension
Table

Shop_ID	Week_ID	Product_ID	Quantity	Amount
1	1	1	100	100
1	2	1	150	150
3	3	4	350	350
4	4	4	200	200

Fact Table

Week_ID	Week	Month
1	Jan1	Jan.
2	Jan2	Jan.
3	Feb1	Feb.
4	Feb2	Feb.

Dimension
Table

Product_ID	Product	Type	Category	Supplier
1	P1	A	X	F1
2	P2	A	X	F1
3	P3	B	X	F2
4	P4	B	X	F2

Schema a Stella: Osservazioni

- La visione multidimensionale dei dati si ottiene eseguendo il **join** tra le *fact table* e le diverse *dimension table*.
- Per il fatto delle vendite riportato in precedenza, l'interrogazione SQL che ricostruisce le celle associando i valori delle misure ai corrispondenti valori degli attributi presenti nelle gerarchie è:

```
SELECT * FROM VENDITE AS FT, PRODOTTO AS DT1,  
          NEGOZIO AS DT2, DATA AS DT3  
WHERE  FT.chiaveP=DT1.chiaveP AND  
        FT.chiaveN=DT2.chiaveN AND  
        FT.chiaveD=DT3.chiaveD;
```

Schema a Stella: Proprietà

- Alla chiave di una dimension table si possono riferire più fact table, se le gerarchie sono conformi.
- Le dimension table **non** sono in 3NF, a causa della presenza di dipendenze funzionali transitive generate dalla presenza contemporanea di tutti gli attributi della gerarchia.
- La sparsità non rappresenta un problema, poiché nella fact table vengono memorizzate solo le combinazioni di chiavi per le quali effettivamente esiste l'informazione.

I sistemi ROLAP: Lo schema snowflake

- Uno schema *snowflake* è ottenibile da uno schema a stella decomponendo una o più **dimension table** DT_i in più tabelle $DT_{i,1}, \dots, DT_{i,n}$, al fine di eliminare alcune delle dipendenze funzionali transitive presenti. Ogni dimension table è caratterizzata da:
 - Una chiave primaria $d_{i,j}$ (*di solito surrogata*);
 - Un sottoinsieme degli attributi di DT_i che dipendono funzionalmente da $d_{i,j}$;
 - Zero o più chiavi esterne riferite ad altre $DT_{t,k}$ necessarie a garantire la ricostruibilità del contenuto informativo di DT_i .
- Le dimension table primarie sono quelle in cui le chiavi sono importate nella fact table, secondarie le altre.

I sistemi ROLAP:

Esempio di schema snowflake

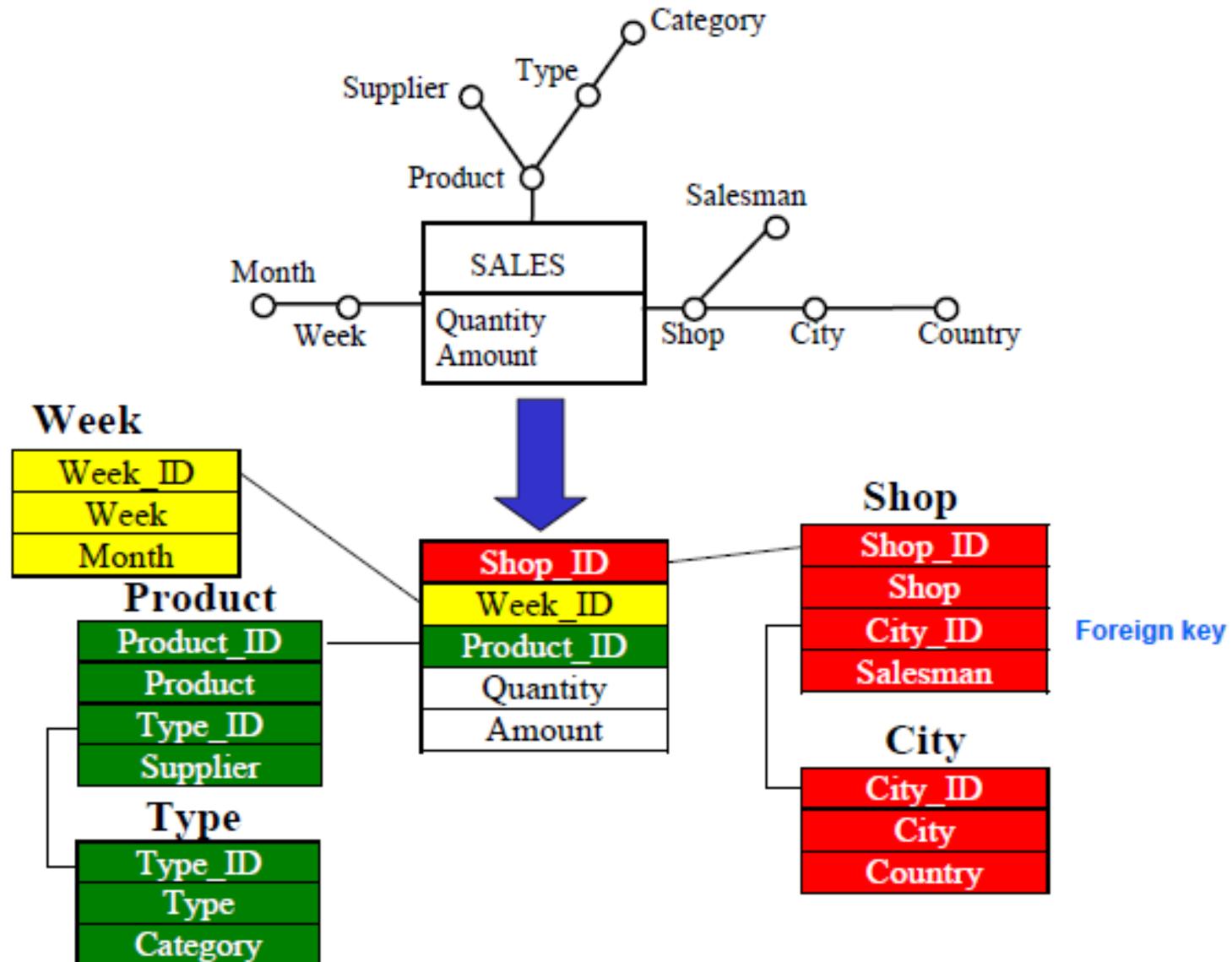
- Un possibile snowflake per lo schema a stella delle vendite prevede l'inserimento delle tabelle CITTÀ (*City*) e CATEGORIA (*Type*), ottenendo una parziale normalizzazione dei dati contenuti nelle dimension table.
- Vengono spezzate le dipendenze transitive tra *negozi* (*Shop*) e *città* (*City*), e tra *prodotto* (*Product*) e *categoria* (*Type*):
 - Lo spazio richiesto per la memorizzazione dei dati si riduce:
 - **Es:** le corrispondenze tra valori degli attributi città e regione vengono memorizzate una sola volta;
 - Il tempo di esecuzione delle interrogazioni che coinvolgono attributi delle dimension table secondarie aumenta poiché è necessario un maggior numero di join.

I sistemi ROLAP:

Vantaggi schema snowflake

- È necessario inserire nuove chiavi surrogate per determinare le corrispondenze tra dimension table primarie e secondarie.
 - ▣ **Es:** l'importazione di Type_ID nella tabella PRODOTTO permette di associare ad ogni prodotto il relativo tipo;
- L'esecuzione di interrogazioni che coinvolgono solo gli attributi contenuti nella fact table e nelle dimension table primarie è avvantaggiata poiché i join coinvolgono tabelle di dimensioni inferiori.

Esempio Schema Snowflake



Esempio (2)

Type_ID	Type	Category
1	A	X
2	B	X

Product_ID	Product	Supplier	Type_ID
1	P1	F1	1
2	P2	F1	1
3	P3	F2	2
4	P4	F2	2

Week_ID	Week	Month
1	Jan1	Jan.
2	Jan2	Jan.
3	Feb1	Feb.
4	Feb2	Feb.

Shop_ID	Week_ID	Product_ID	Quantity	Amount
1	1	1	100	100
1	2	1	150	150
3	3	4	350	350
4	4	4	200	200

Shop_ID	Shop	City_ID	Salesman
1	N1	1	R1
2	N2	1	R1
3	N3	2	R2
4	N4	2	R2

City_ID	City	Country
1	RM	I
2	MI	I

Star o snowflake?

- Lo schema snowflake normalmente non è raccomandato:
 - La diminuzione dello spazio di memorizzazione raramente è benefico:
 - Maggiore spazio è consumato dalla fact table.
 - Il costo della esecuzione del join potrebbe essere significativo.
- Lo schema snowflake è utile:
 - quando parte di una gerarchia è condivisa tra le dimensioni (**Es:** gerarchie geografiche).
 - Per le viste materializzate che richiedono una rappresentazione aggregata delle dimensioni corrispondenti.

Le viste

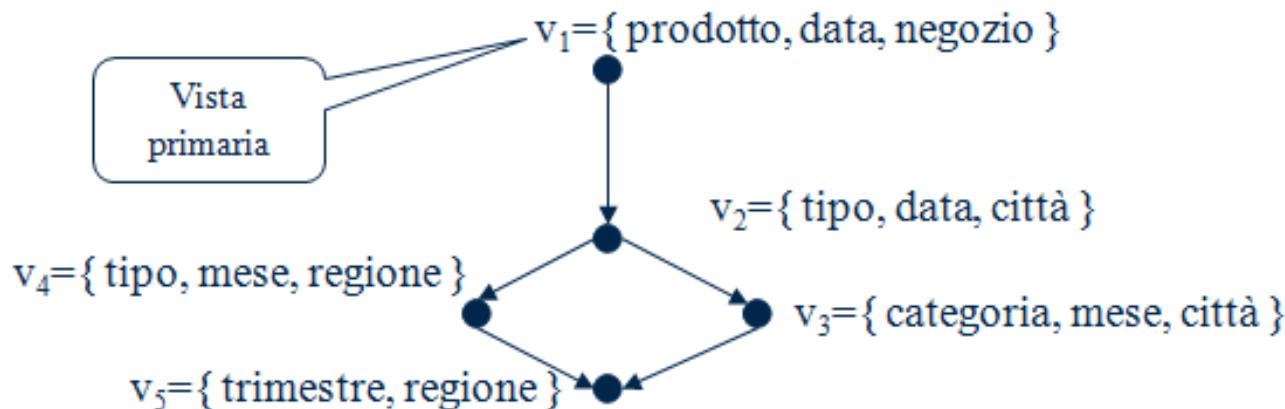
- **Problema:** Analisi utente rese difficili dalla quantità di dati memorizzati nel DW.
- **Soluzione:** Ridurre la porzione da esaminare attraverso operazioni di:
 - *Selezione*: restringono la porzione di dati di interesse individuando quelli effettivamente interessanti per la specifica analisi;
 - *Aggregazione*: riducono i dati collassando più elementi non aggregati in un unico elemento aggregato.

Le viste (2)

- L'aumento delle prestazioni è ottenuto precalcolando i dati aggregati di uso comune.
- Le fact table contenenti dati aggregati sono dette **viste**:
 - **Viste primarie**: corrispondono a pattern primari (definito dall'insieme delle dimensioni);
 - **Viste secondarie**: corrispondono a pattern secondari o sono individuate verificando se possono essere alimentate a partire da viste nel DW (aggregati).

Le viste (3)

- Sono rappresentate alcune viste materializzabili per lo schema a stella delle vendite:
 - Una freccia da v_i a v_j indica che $P_j \leq P_i$ essendo P_i e P_j rispettivamente i pattern di v_i e v_j . Quindi i dati contenuti in v_j possono essere calcolati aggregando quelli di v_i .
 - Un'interrogazione relativa alle vendite che richieda i dati aggregati per il tipo del prodotto, data di vendita e città (i.e., $\{tipo, data, città\}$) in cui la vendita è stata effettuata risulterà meno costosa se eseguita su v_2 poiché essa insisterà su una fact table piccola e non richiederà ulteriori operazioni di aggregazione.



La vista: Problemi

- **Problema:** Calcolando l'incasso delle vendite a partire dalla tabella aggregata si ottiene un dato diverso da quello ottenuto a partire dalla tabella non aggregata.
- **Causa:** Applicando l'operatore di media si perde l'informazione relativa ai singoli prezzi praticati.
- **Soluzione:** Memorizzare nella tabella aggregata anche i valori degli incassi.

tipo	prodotto	quantità	prezzo	incasso
latticino	Latte Slurp	5	1,0	5,0
latticino	Latte Gnam	7	1,5	10,5
bibita	Colissima	9	0,8	7,2
totale:				22,7

Tabella delle vendite di diversi prodotti

		SUM	AVG
tipo	quantità	prezzo	Quantità x prezzo
latticino	12	1.25	15,0
bibita	9	0,8	7,2
totale:			22,2

Tabella dei dati aggregati in base ai tipi di prodotti



Progettazione logica

Progettazione logica

- La progettazione logica definisce l'insieme dei passi per trasformare lo schema concettuale in uno schema logico.
- La progettazione logica dei Data mart è profondamente diversa dai sistemi operazionali:
 - Nei DW l'obiettivo è quello di massimizzare la velocità del reperimento dei dati mentre nei sistemi operazionali si mira a minimizzare la quantità di informazione da memorizzare.
- I principali passi di questo processo sono:
 - **Traduzione degli schemi di fatto in schemi logici.**
 - **Materializzazione delle viste.**
 - Frammentazione verticale ed orizzontale delle *fact table*.

Traduzione degli schemi di fatto in schemi logici

- Uno schema di fatto può essere modellato in ambito relazionale mediante uno *schema a stella* in cui la *fact table* contiene tutte le misure e gli attributi descrittivi e, per ogni gerarchia, viene creata una *dimension table* che ne contiene tutti gli attributi.
- La traduzione dal DFM al modello logico non è del tutto automatica, richiedendo in alcuni momenti l'intervento del progettista.
 - Una corretta traduzione di uno schema di fatto richiede una trattazione più approfondita per i costrutti avanzati del DFM.

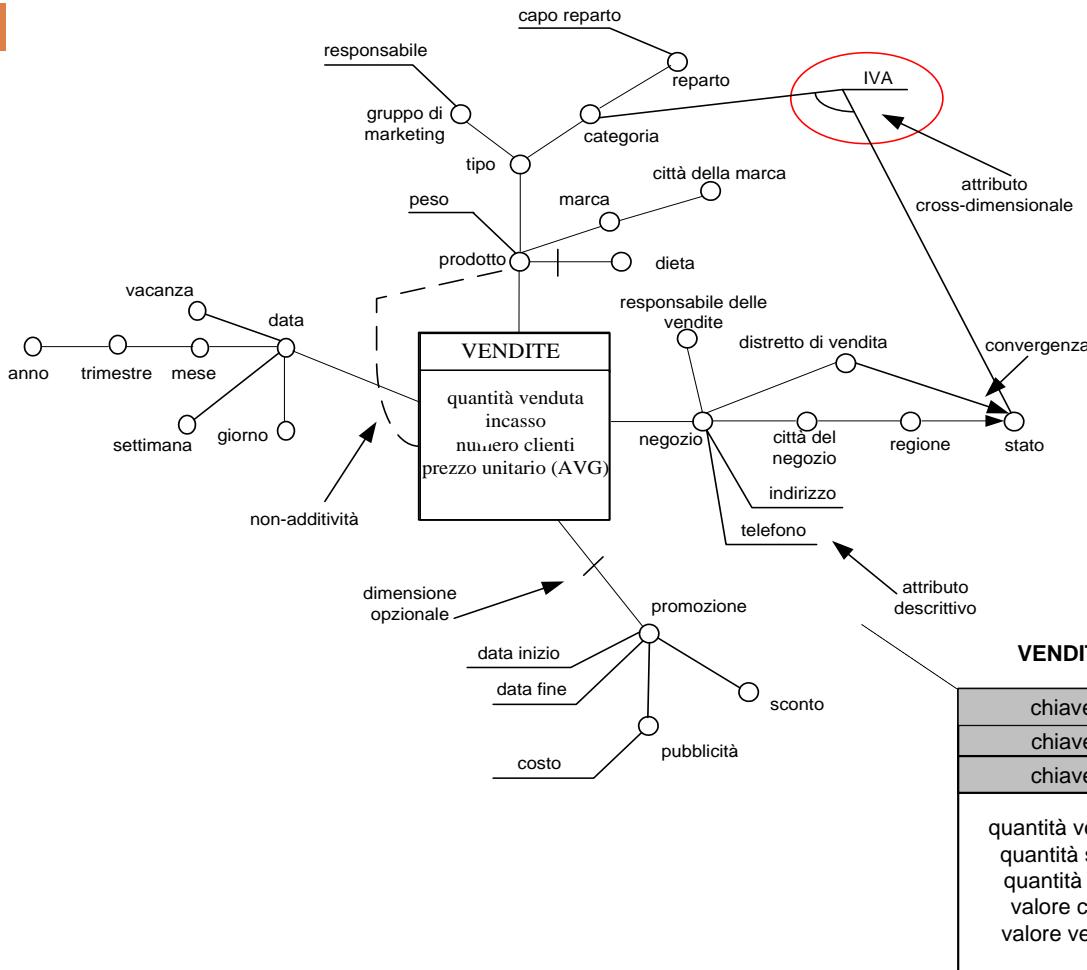
Costrutti avanzati: Attributi descrittivi

- Sappiamo che un attributo descrittivo contiene informazioni non utilizzabili per effettuare aggregazioni che si ritiene utile mantenere.
- Durante la modellazione un attributo descrittivo:
 - viene incluso nella *dimension table* relativa alla gerarchia che lo contiene se collegato ad un attributo dimensionale da cui dipende funzionalmente.
 - viene incluso nella *fact table* assieme alle misure collegate direttamente al fatto.

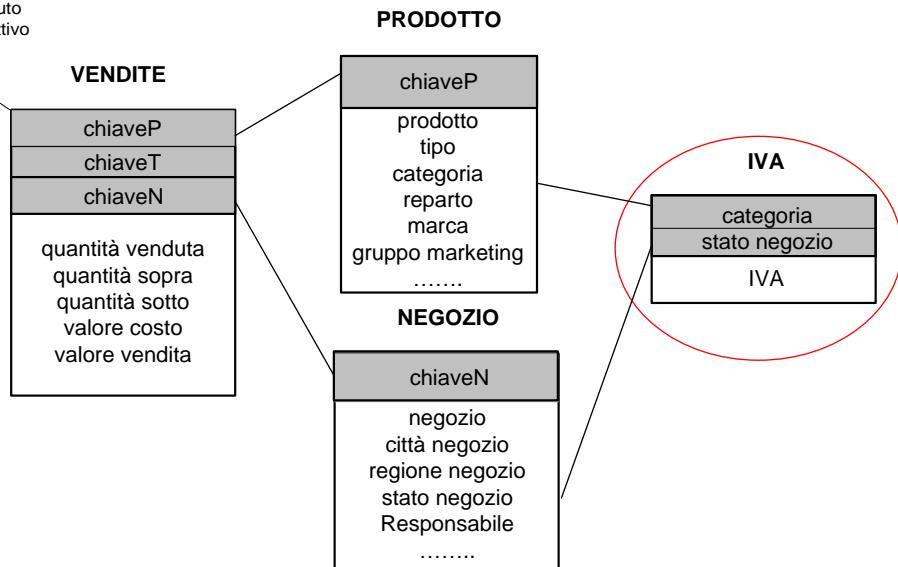
Costrutti avanzati: Attributi cross-dimensionali

- Sappiamo che un attributo cross-dimensionale è tale se il suo valore è determinato dalla combinazione di due o più attributi dimensionali eventualmente appartenenti a gerarchie diverse.
- Se un attributo cross-dimensionale **b** definisce un'associazione multi-a-molti tra due o più attributi dimensionali a_1, \dots, a_n , esso richiede l'inserimento di una nuova tabella che includa **b** ed abbia come chiave gli attributi a_1, \dots, a_n .

Esempio: Attributi cross-dimensional



□ Modellazione attributo cross-dimensionale IVA.

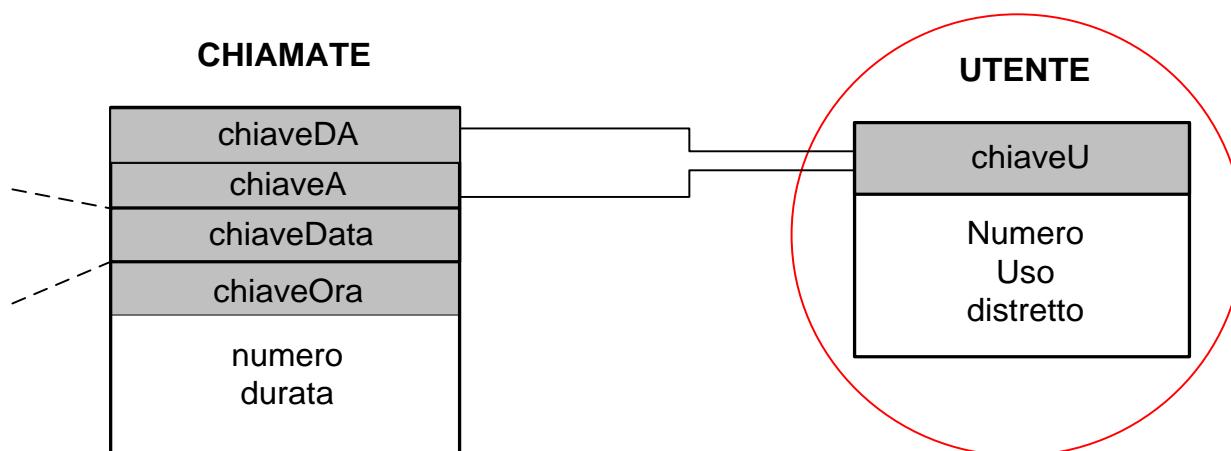
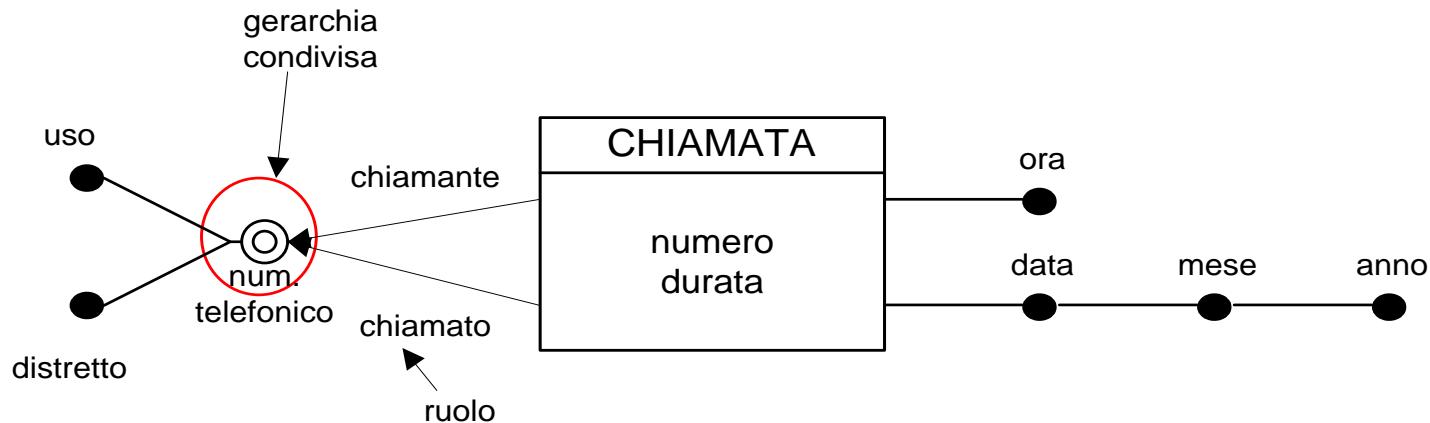


Costrutti avanzati: Gerarchie condivise

- Sappiamo che una gerarchia condivisa è una porzione di gerarchia che viene ripetuta due o più volte.
- A livello logico non è consigliabile introdurre più *dimension table* che contengano gli stessi dati.
- A livello progettuale esistono due soluzioni per due situazioni distinte:
 - Se due gerarchie contengono esattamente gli stessi attributi è sufficiente importare due valori diversi dell'unica chiave della *dimension table* nella *fact table*.
 - Se due gerarchie condividono una parte degli attributi si può scegliere di replicare le informazioni comuni o di introdurre una nuova *dimension table* comune ad entrambe le gerarchie.

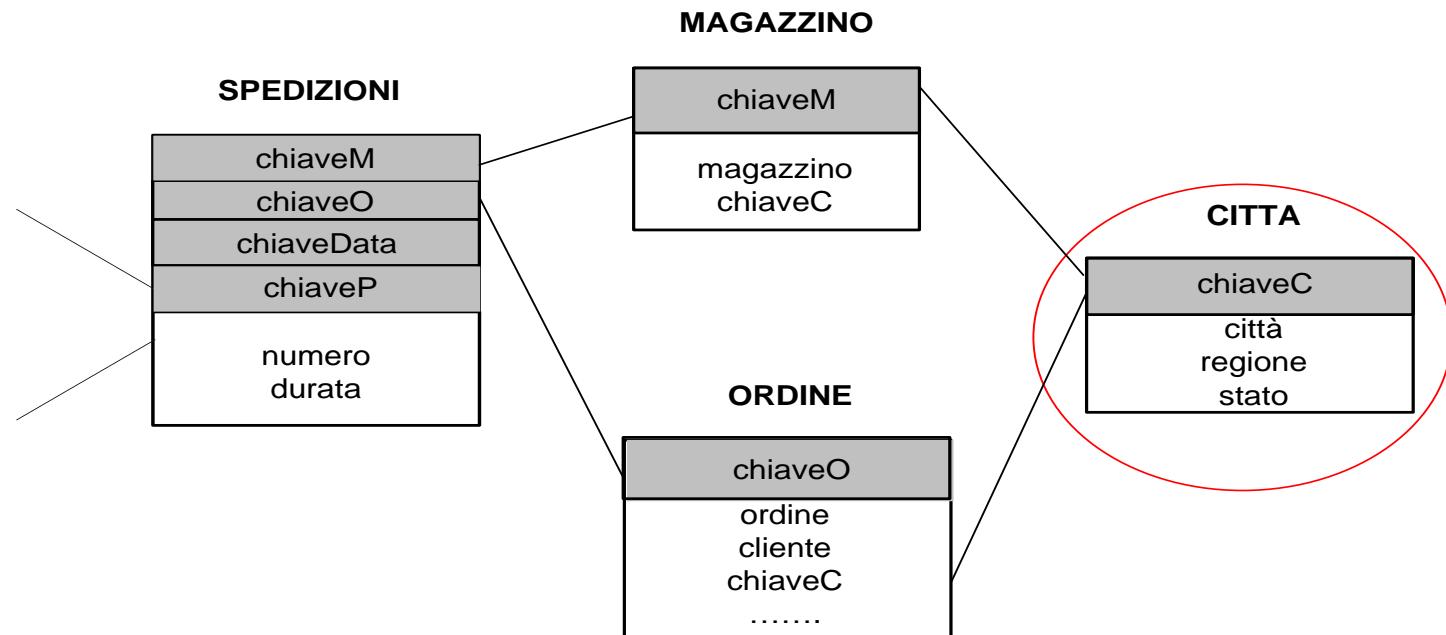
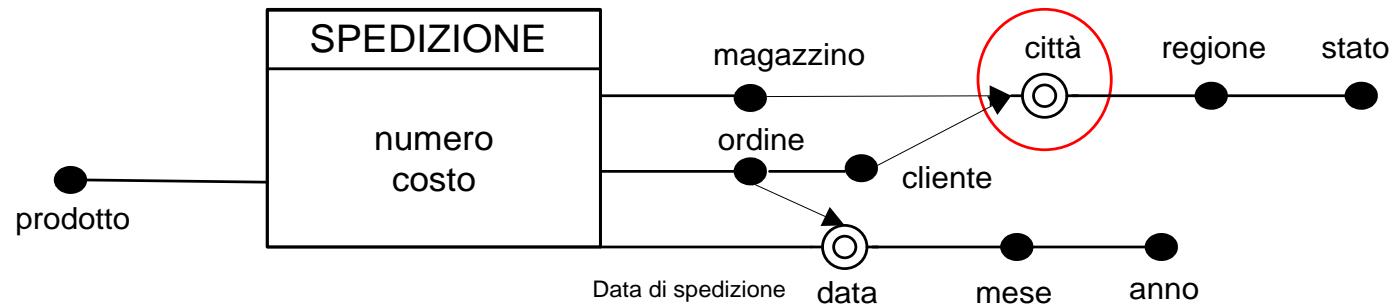
Esempio: Condivisione totale

- Modellazione numero telefonico chiamante e chiamato.



Esempio: Condivisione parziale

□ Modellazione città del magazzino e del cliente.

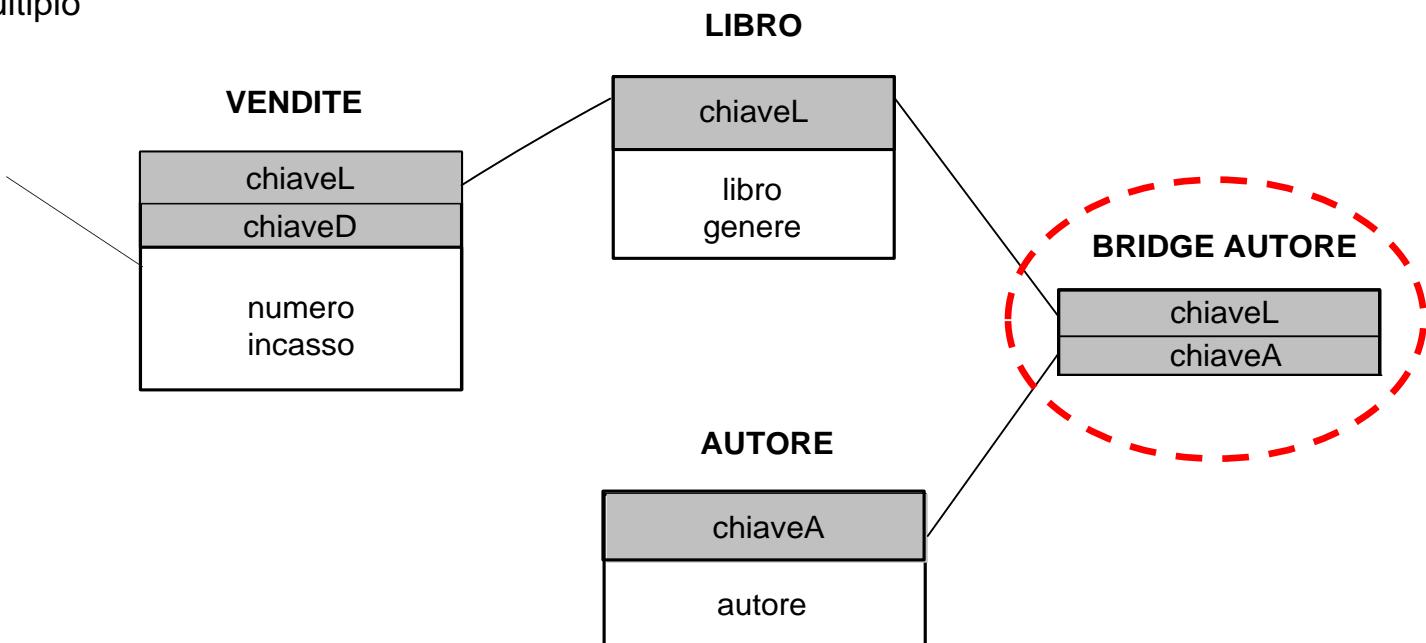
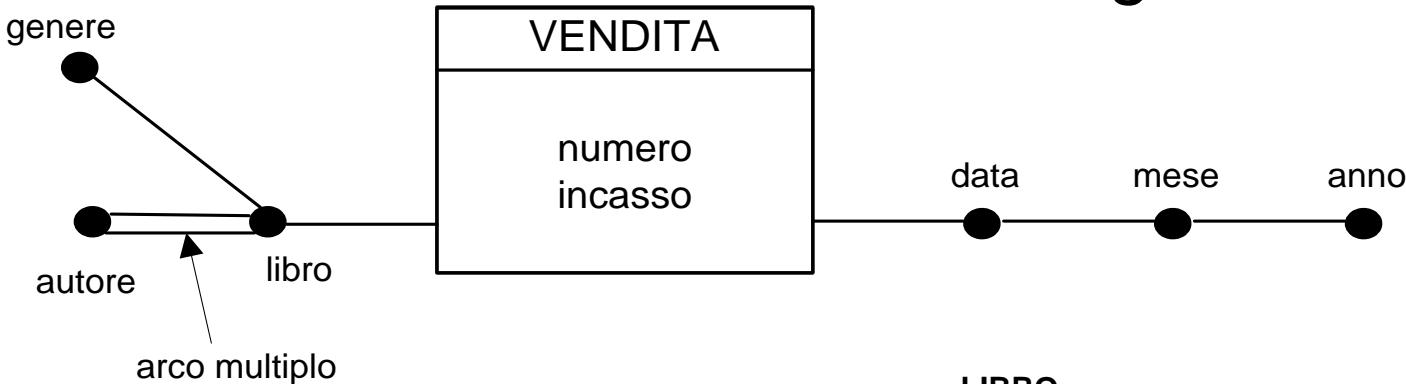


Costrutti avanzati: Archi multipli

- Ricordiamo che una gerarchia che codifica associazioni multi-a-molti viene modellata con archi multipli.
- A livello logico esistono diverse soluzioni:
 - Soluzione ***bridge table***: utilizzo di una nuova tabella la cui chiave è composta dalla combinazione degli attributi collegati dall'arco multiplo (schema snowflake).
 - Soluzione ***push-down***: l'associazione multi-a-molti viene modellata direttamente all'interno della *fact table*. Viene poi aggiunta una nuova dimensione corrispondente all'attributo terminale **a** dell'arco multiplo, ed eventuali figli di **a** verranno memorizzati nella nuova *dimension table* (schema a stella).

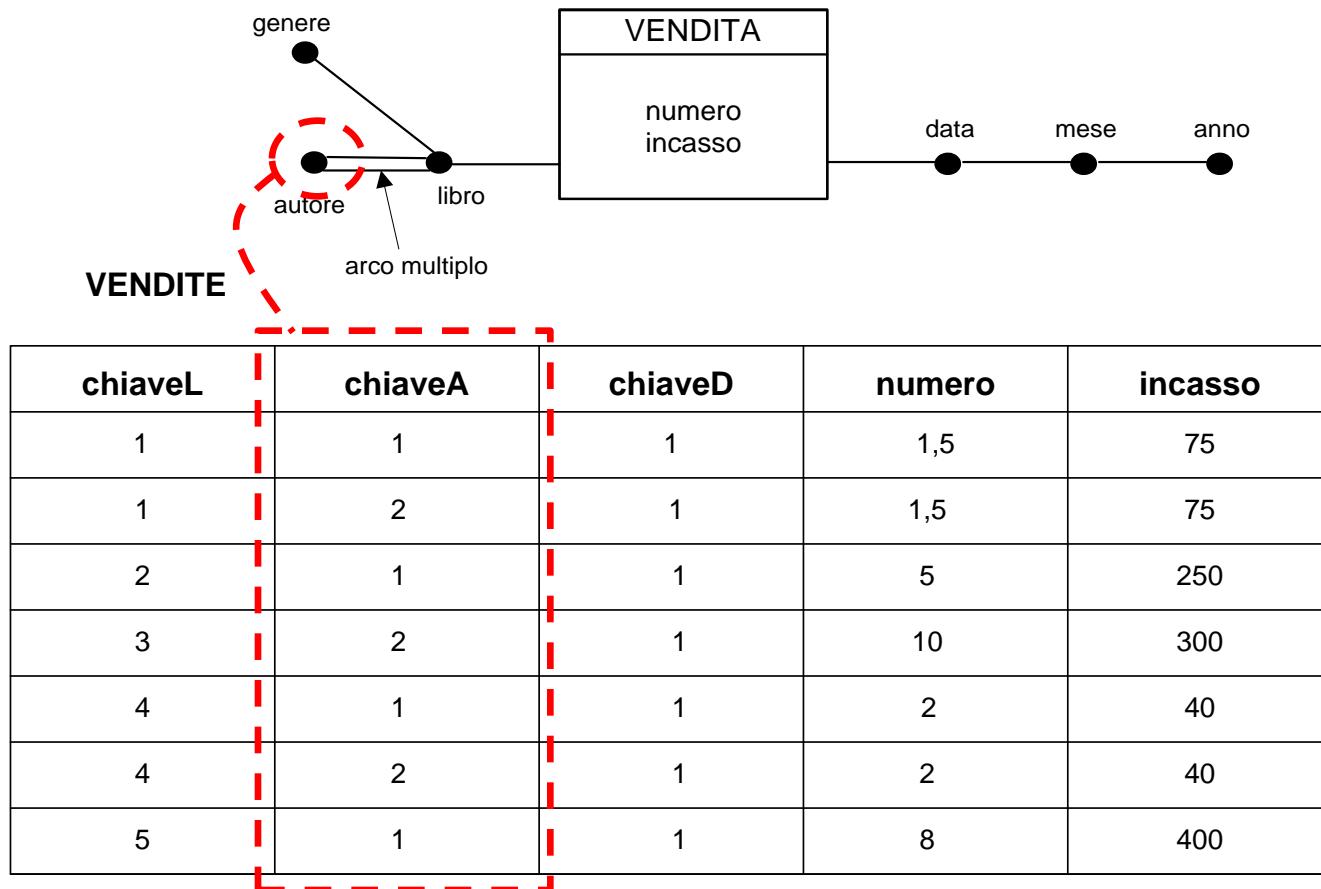
Esempio: Bridge table

□ Modellazioni di Autore tramite *bridge table*.



Esempio: Push-down

- Una possibile istanza della *fact table* Vendite avendo effettuato il push-down di Autore.



Bridge table vs. push-down

- La soluzione push-down introduce una forte ridondanza nella fact table le cui righe devono essere replicate tante volte quante sono le corrispondenze dell'arco multiplo.
 - La forte ridondanza causa operazioni di aggiornamento molto costose che non si verificano nella soluzione bridge table.
- Operazioni di interrogazioni nella soluzioni push-down prevedono un singolo join mentre con bridge table risultano più complesse.
- Il calcolo degli eventi primari avviene durante l'alimentazione nella soluzione push-down mentre nella soluzione con bridge table avviene durante l'interrogazione.

Costrutti avanzati: Archi opzionali

- Ricordiamo che un arco opzionale si riferisce ad una gerarchia opzionale, in cui un'associazione dello schema di fatto non è definita per un sottoinsieme di eventi.
- La presenza di archi opzionali non incide sulla struttura della corrispondente *dimension table*:
 - L'attributo continua a comparire anche se per alcune istanze non risulterà valorizzato.
 - Per le istanze in cui non è definito tale valore si introduce un valore fittizio.
- L'opzionalità non può essere gestita direttamente nella fact table introducendo un valore fittizio per la chiave.
 - Bisognerà introdurre un'intera tupla fittizia all'interno della dimension table.

Costrutti avanzati: Gerarchie ricorsive

- Ricordiamo: una gerarchia ricorsiva è una gerarchia in cui le relazioni padre-figlio tra i livelli sono consistenti ma possono avere istanze di lunghezza differenti.
- La modellazione delle gerarchie ricorsive può essere effettuata in due modi:
 - Nella *dimension table* la ricorsione è modellata con un autoanello che rappresenta un numero variabile, e potenzialmente illimitato, di livelli. Tale modellazione non è supportata dalla maggior parte dei DBMS commerciali.
 - Con una **tabella di navigazione** che modella un associazione molti-a-molti tra la *fact table* e la *dimension table*. La dimensione di tale tabella cresce in maniera esponenziale rispetto alla profondità della gerarchia ma tale modellazione ha un maggior poter espressivo in fase di interrogazione.

Lo schema logico relazionale

□ Schema logico relazionale di Vendita:

PRODOTTO (prodotto, peso, dieta, marca: MARCA, tipo: TIPO)

MARCA (marca, prodottaIn:CITTA)

CITTA (città, regione:REGIONE)

REGIONE (regione, stato:STATO)

STATO (stato)

TIPO (tipo, gruppoMarketing:GRUPPOMARK, categoria:CATEGORIA)

GRUPPOMARK (gruppoMarketing, responsabile)

CATEGORIA (categoria, reparto:REPARTO)

REPARTO (reparto, capoReparto)

Gli attributi facenti parte di chiavi esterne composte sono indicati tra parentesi

IVA (categoria:CATEGORIA, stato:STATO, iva)

NEGOZIO (negozi, indirizzo, telefono, respVendite, (numDistr, stato):DISTRETTO, inCittà:CITTA)

DISTRETTO (numDistr, stato:STATO)

DATA (data, giorno, vacanza, settimana, mese:MESE)

MESE (mese, trimestre:TRIMESTRE)

TRIMESTRE (trimestre, anno:ANNO)

ANNO (anno)

PROMOZIONE (promozione, dataInizio, dataFine, sconto, pubblicità:PUBBLICITA)

PUBBLICITA (pubblicità, costo)

VENDITA (podotto:PRODOTTO, negozi:NEGOZIO, data:DATA, promozione:PROMOZIONE, **quantità**, **prezzoUnitario**)

Materializzazione delle viste

- Con il termine materializzazione delle viste si intende il processo di selezione di un'insieme di viste secondarie ottenute a partire dai dati contenuti nelle viste primarie.
- La scelta delle viste da materializzare deve essere fatta sulla base di un'insieme di obiettivi di progetto.
- Due sono gli elementi principali nel processo di materializzazione:
 - La definizione degli obiettivi della materializzazione, che possono essere funzioni di minimizzazione di costo o vincoli.
 - La tecnica di selezione da utilizzare.

Funzioni di costo da minimizzare

- Tipicamente le funzioni di costo che possono essere minimizzate sono:
 - ▣ **Costo del carico di lavoro:** rappresenta il costo totale del carico di lavoro che può essere calcolato come somma pesata del costo delle diverse interrogazioni, dove il peso di ogni singola interrogazione può essere la frequenza e/o la sua importanza per l'utente.
 - ▣ **Costo di manutenzione delle viste:** rappresenta il costo delle interrogazioni necessarie a propagare gli aggiornamenti delle sorgenti operazionali alle viste.

Vincoli

- **Vincoli di sistema:** sono dettati dalla limitatezza delle risorse disponibili e riguardano:
 - **Spazio di memorizzazione:** per calcolare questo vincolo è necessaria una funzione per la stima della dimensione delle viste.
 - **Tempo di aggiornamento:** il tempo di aggiornamento delle viste.
- **Vincoli utente:** sono legati a particolari requisiti espressi dagli utilizzatori del sistema:
 - **Tempo di risposta alle interrogazioni:** tempo massimo di risposta richiesto dall'utente per le diverse interrogazioni.
 - **Data di aggiornamento delle risposte:** limite massimo imposto dall'utente per il tempo intercorso dall'ultimo aggiornamento di una vista impiegata per l'esecuzione di un'interrogazione.

Tecniche di selezione

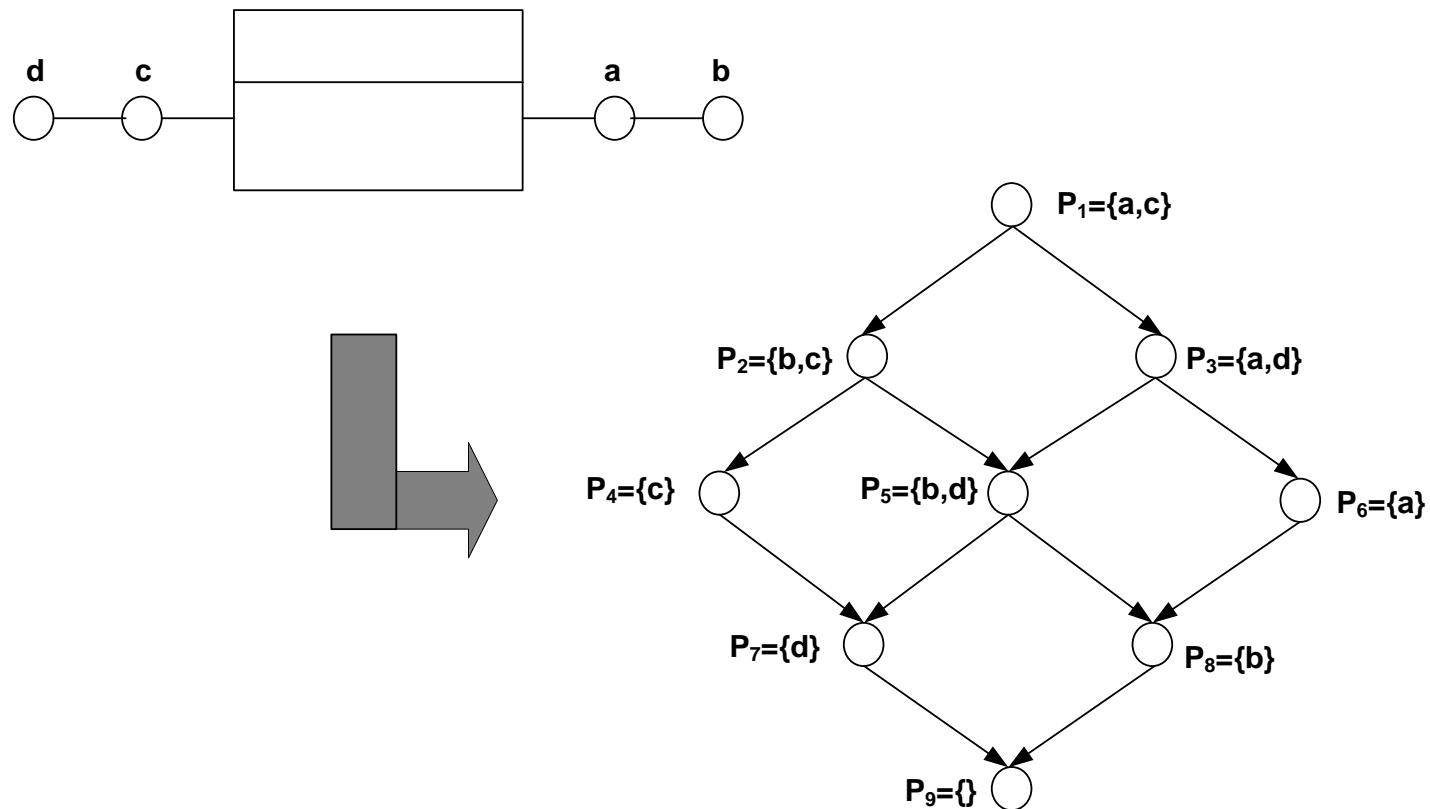
- Le tecniche di selezione operano su due diverse fasi:
 - Tra tutte le possibili viste materializzabili, vengono individuate quelle effettivamente utili per il carico di lavoro.
 - Successivamente, tramite tecniche euristiche, se ne determina un sottoinsieme che minimizza la funzione di costo nel rispetto dei vincoli di sistema.

Reticolo multidimensionale

- Il **reticolo multidimensionale** o MD-lattice viene utilizzato per individuare tutte le possibili viste materializzabili a partire da uno schema di fatto, definendo tutti i possibili pattern di aggregazione validi.
- Un pattern è valido se non esistono dipendenze funzionali tra i suoi elementi.
- **Nota.** Un arco del reticolo da un pattern P_i ad un pattern P_j indica che P_j è meno fine di P_i ($P_j \leq P_i$).

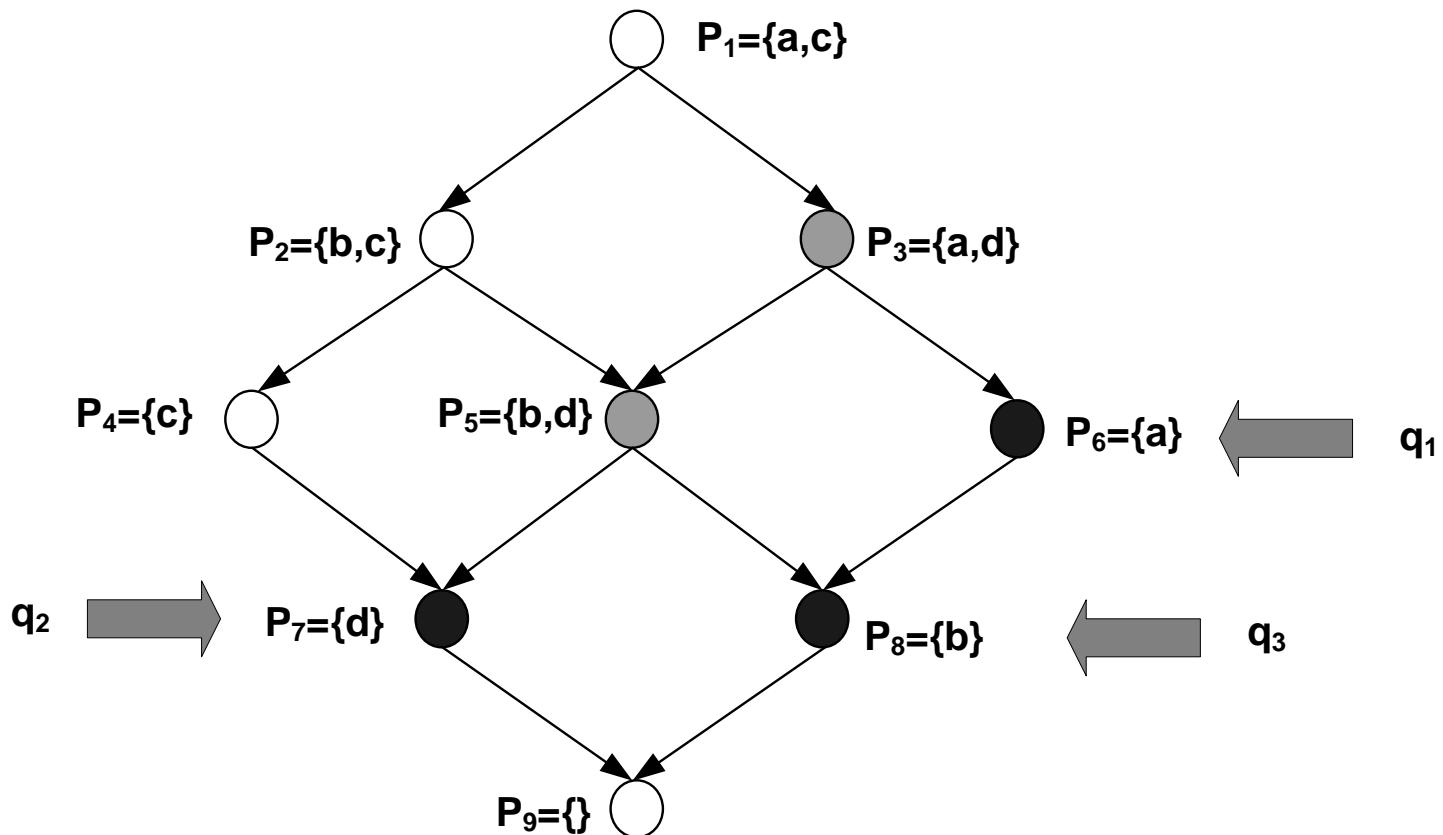
Esempio: Reticolo multidimensionale

- Reticolo corrispondente al cubo multidimensionale con dimensioni **a**, **c** e gerarchie **a** → **b** e **c** → **d**.



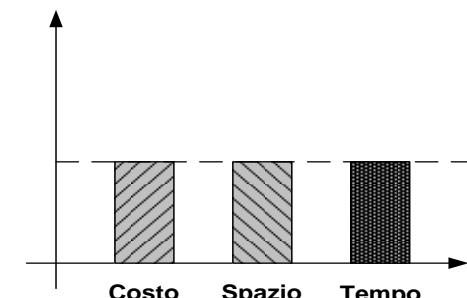
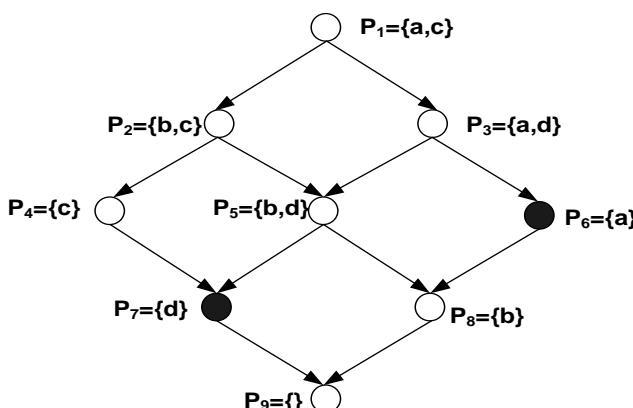
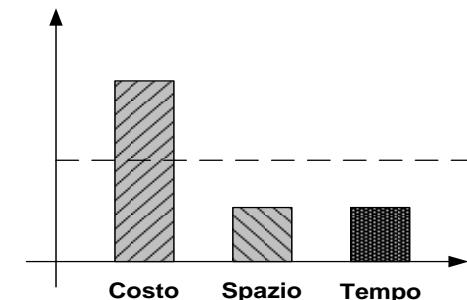
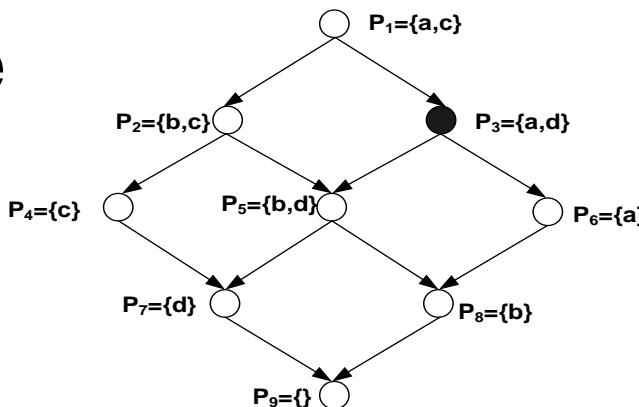
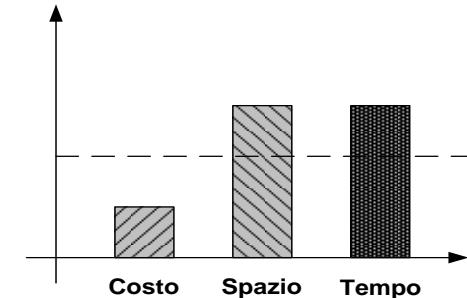
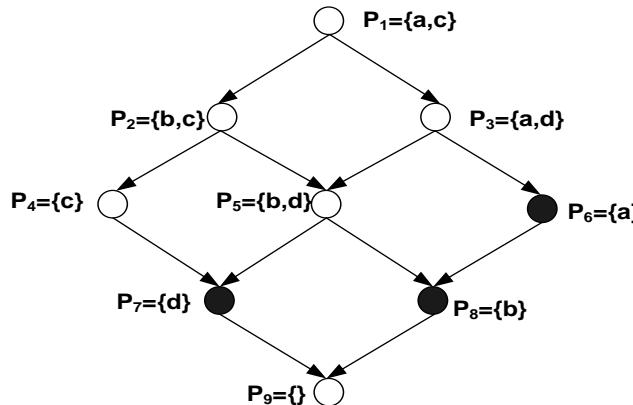
Esempio: Viste candidate

- In grigio e nero sono evidenziate le viste per il carico di lavoro q_1 , q_2 , q_3 .



Esempio: Scelta delle viste da materializzare

- Tre possibili soluzioni al problema della materializzazione delle viste.



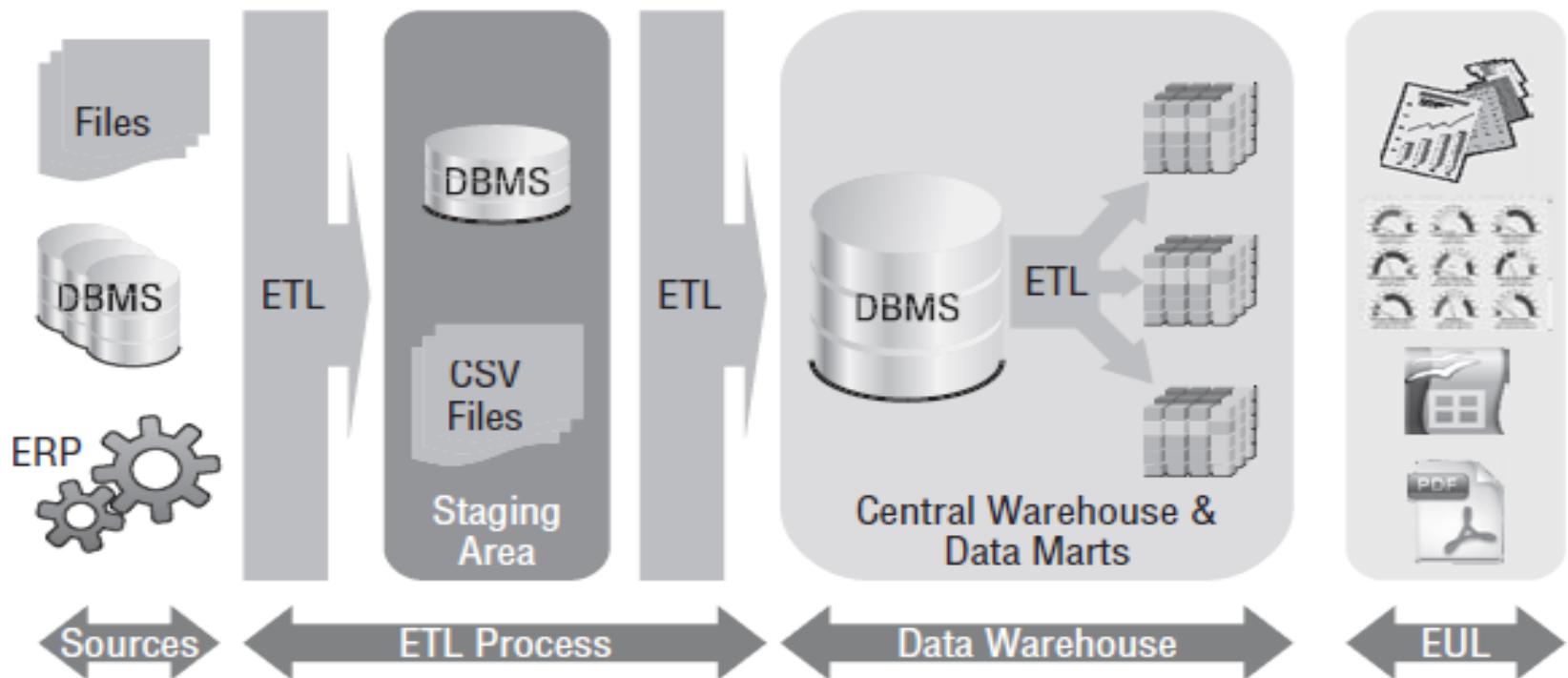
Linee guida selezionare le viste

- È consigliabile materializzare una vista quando:
 - Risolve direttamente un'interrogazione molto frequente.
 - Permette di risolvere molte interrogazioni.
- Non è consigliabile materializzare una vista quando:
 - Il suo pattern è molto simile ad una vista già materializzata.
 - Il suo pattern è molto fine.
 - La materializzazione non riduce di almeno un ordine di grandezza il carico di lavoro.



Data warehouse in pratica

Generic data warehouse architecture

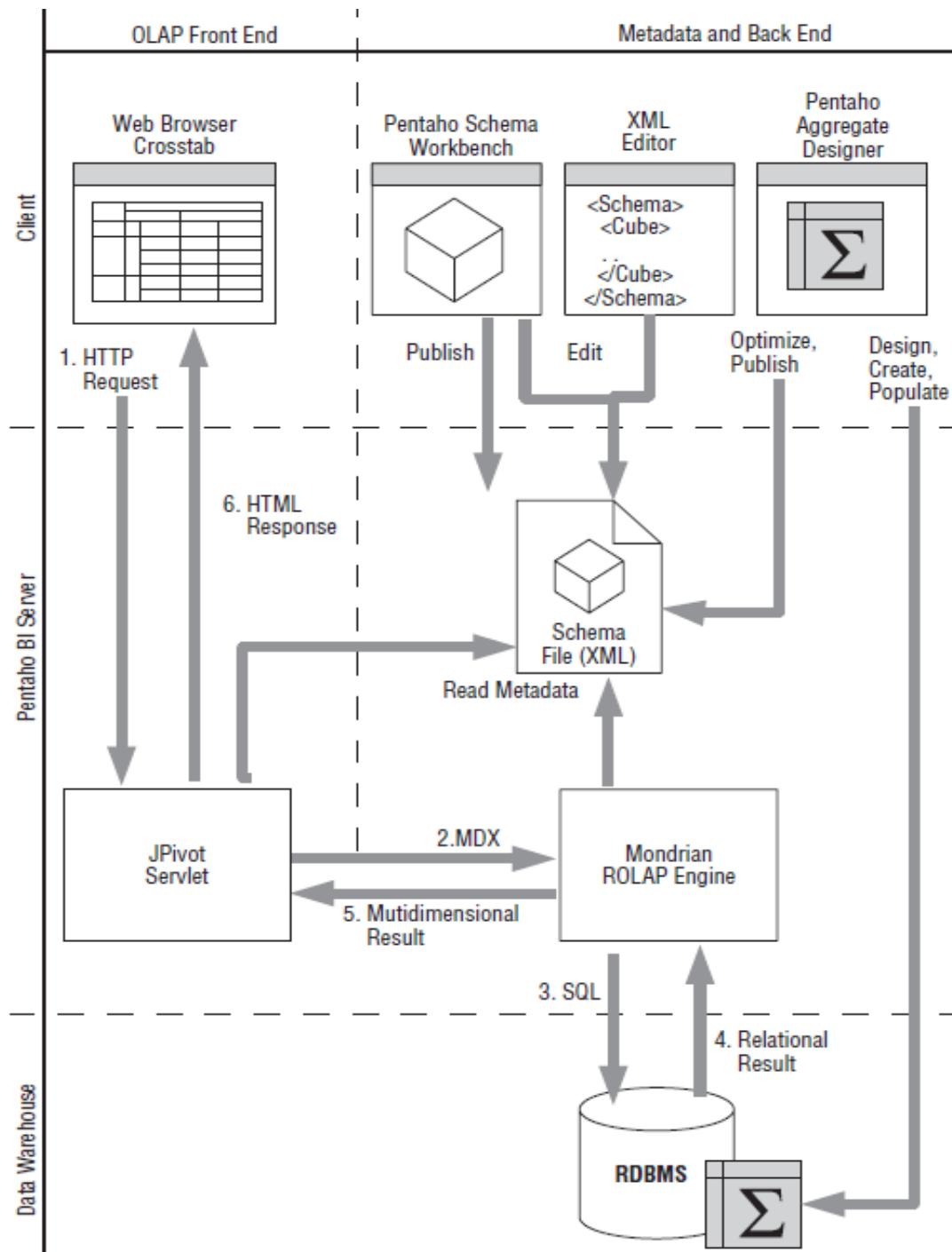


Data warehouse with Mondrian

- SQL Database: **MySQL**
- OLAP Engine: **Mondrian ROLAP**
- Analisys front end: **Jpivot**



Pentaho OLAP components



Tools

- *JPivot analysis front end:*
 - JPivot is a Java-based analysis tool that serves as the actual user interface for working with OLAP cubes.
- *Mondrian ROLAP engine:*
 - The engine receives MDX (**M**ulti **D**imensional **E**xpressions) queries from front-end tools such as JPivot, and responds by sending a multidimensional result-set.
- *Schema Workbench:*
 - This is the visual tool for designing and testing Mondrian cube schemas. Mondrian uses these cube schemas to interpret MDX and translate it into SQL queries to retrieve the data from an RDBMS.
- *Data Integration:*
 - The desktop tool (Kettle) for building ETL jobs and transformations.

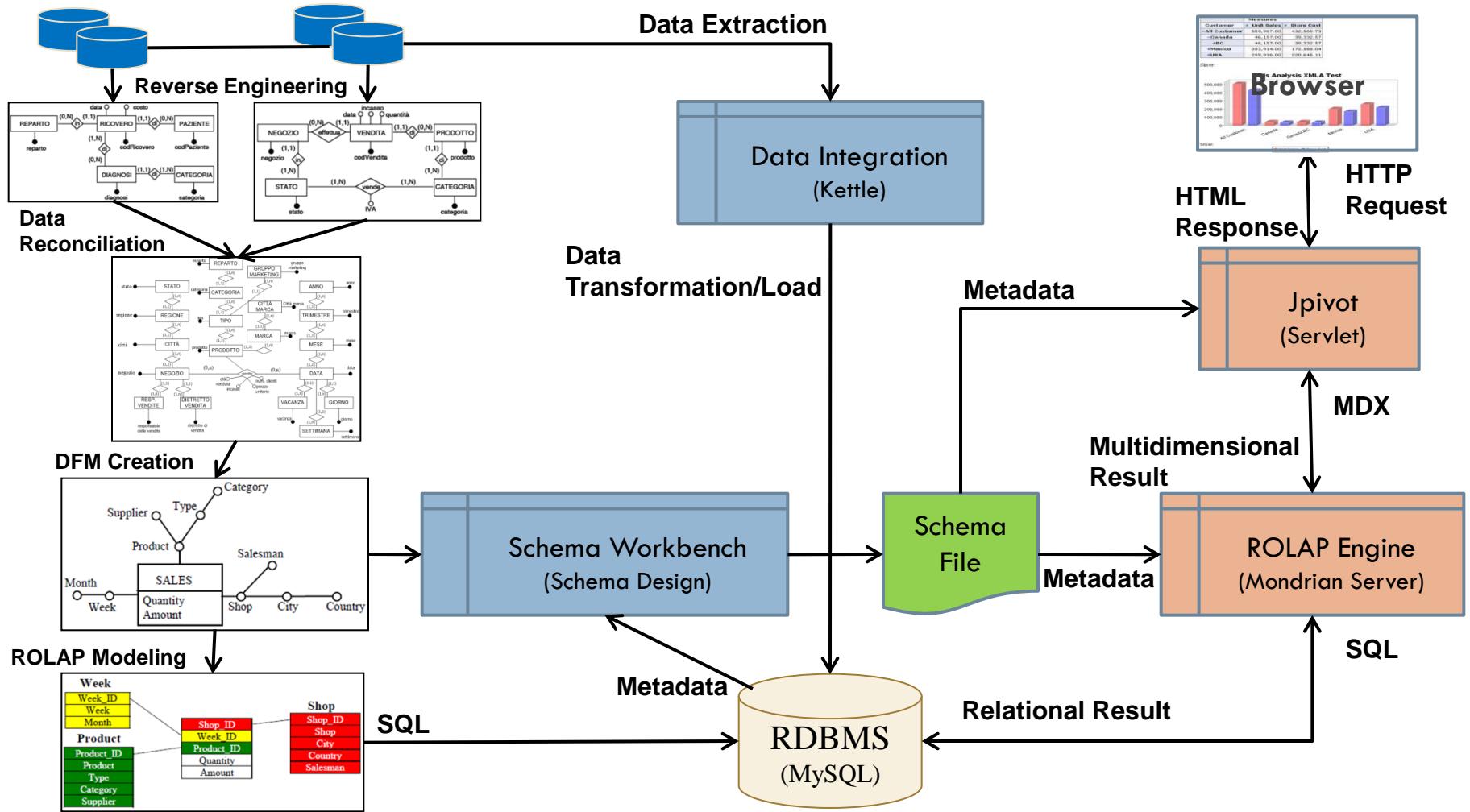
Schema

- A central structure is the *schema*.
 - The schema is essentially an XML document that describes one or more multidimensional cubes.
 - The cubes also describe the mapping of the cube's dimensions and measures to tables and columns in a relational database.
 - To Mondrian, the schema is key in translating the MDX query to SQL queries.

Schema Design Tools

- The **multidimensional model**, consisting of dimensions, hierarchies, and measures, is created first and the relational model is mapped into the schema.
- Pentaho Schema Workbench offers a graphical user interface to create Mondrian schemas.
 - In addition, Pentaho Schema Workbench can publish schemas to the Pentaho Server, which then stores them in the solution repository.
 - Once stored in the solution repository, the schemas can be used by the server's Mondrian engine as a back end for OLAP services.

Data warehouse in practice (with Mondrian)



Riferimenti

- M. Golfarelli, S. Rizzi,
“Data Warehouse – Teoria e pratica della progettazione”, 2^a ed.
The McGraw-Hill Companies, 2006.
 - Capitolo 1: Tutto
 - Capitolo 2: 2.1, 2.2, 2.3
 - Capitolo 3: Tutto
 - Capitolo 4: 4.1, 4.2
 - Capitolo 5: 5.1, 5.2 (escluso 5.2.9), 5.5, 5.6
 - Capitolo 6: 6.1
 - Capitolo 8: 8.1, 8.2, 8.3 (escluso 8.3.1)
 - Capitolo 9: 9.1, 9.2 (*fino a 9.2.1 escluso*)
 - Capitolo 13: Tutto (*documentazione di progetto*)
 - Capitolo 14: Tutto (*caso di studio*)
 - Capitolo 15: 15.1, 15.3 (*fino a 15.3.1 escluso*)