

# *Data warehouse Progettazione*

Elena Baralis  
Politecnico di Torino

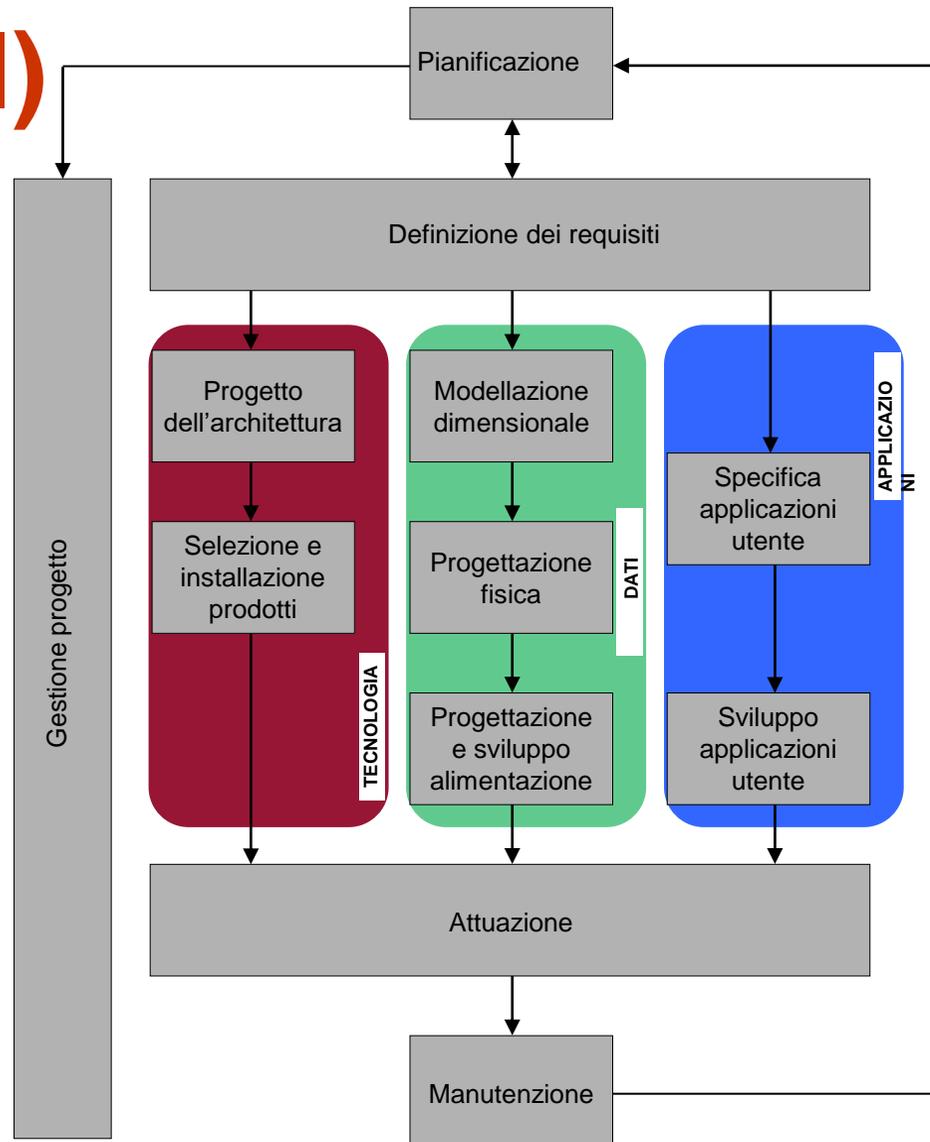
# Fattori di rischio

- Aspettative elevate degli utenti
  - il data warehouse come soluzione dei problemi aziendali
- Qualità dei dati e dei processi OLTP di partenza
  - dati incompleti o inaffidabili
  - processi aziendali non integrati e ottimizzati
- Gestione “politica” del progetto
  - collaborazione con i “detentori” delle informazioni
  - accettazione del sistema da parte degli utenti finali

# Progettazione di data warehouse

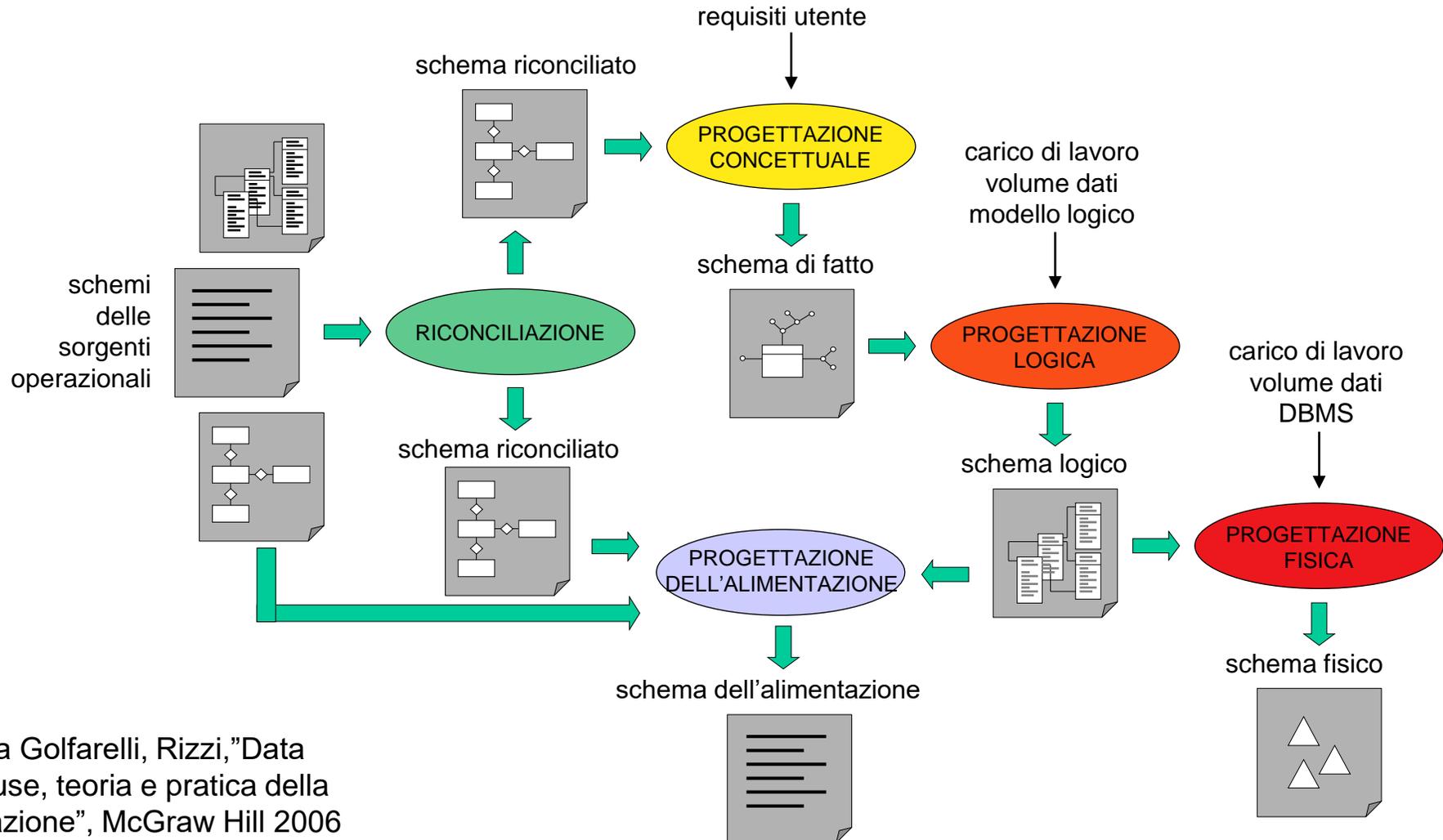
- Approccio top-down
  - realizzazione di un data warehouse che fornisca una visione globale e completa dei dati aziendali
  - costo significativo e tempo di realizzazione lungo
  - analisi e progettazione complesse
- Approccio bottom-up
  - realizzazione incrementale del data warehouse, aggiungendo data mart definiti su settori aziendali specifici
  - costo e tempo di consegna contenuti
  - focalizzato separatamente su settori aziendali specifici

# Business Dimensional Lifecycle (Kimball)



Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# Progettazione di data mart



Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# *Analisi dei requisiti*

Elena Baralis  
Politecnico di Torino

# Analisi dei requisiti

- **Raccoglie**
  - le esigenze di analisi dei dati che dovranno essere soddisfatte dal data mart
  - i vincoli realizzativi dovuti ai sistemi informativi esistenti
- **Fonti**
  - business users
  - amministratori del sistema informativo
- **Il data mart prescelto è**
  - strategico per l'azienda
  - alimentato da (poche) sorgenti affidabili

# Requisiti applicativi

- Descrizione degli eventi di interesse (fatti)
  - ogni fatto rappresenta una categoria di eventi di interesse per l'azienda
    - esempi: (per il CRM) reclami, servizi
  - caratterizzati da dimensioni descrittive (granularità), intervallo di storicizzazione, misure di interesse
  - informazioni raccolte in un glossario
- Descrizione del carico di lavoro
  - esame della reportistica aziendale
  - interrogazioni espresse in linguaggio naturale
    - esempio: numero di reclami per ciascun prodotto nell'ultimo mese

# Requisiti strutturali

- Periodicità dell'alimentazione
- Spazio disponibile
  - per i dati
  - per le strutture accessorie (indici, viste materializzate)
- Tipo di architettura del sistema
  - numero di livelli
  - data mart dipendenti o indipendenti
- Pianificazione del deployment
  - avviamento
  - formazione

# *Progettazione concettuale*

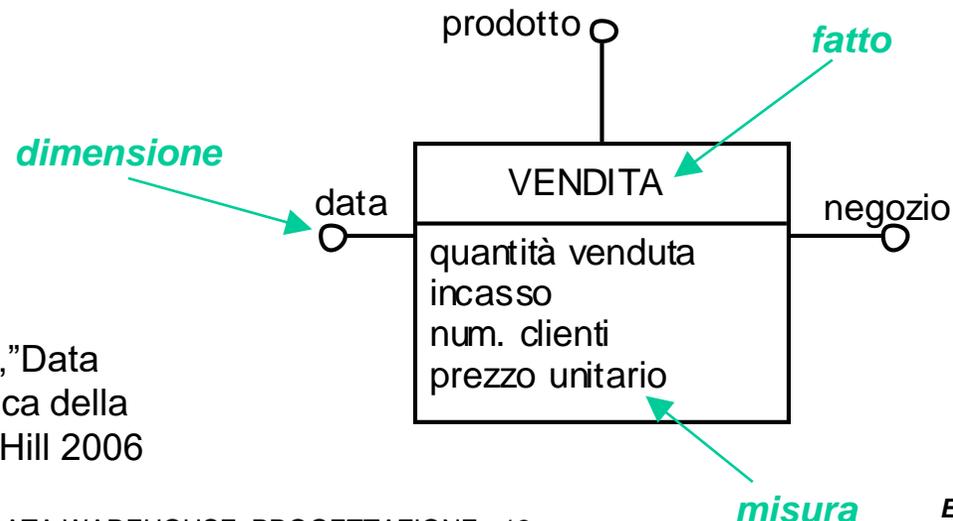
Elena Baralis  
Politecnico di Torino

# Progettazione concettuale

- Non esiste un formalismo di modellazione comunemente accettato
  - il modello ER non è adatto
- Dimensional Fact Model (Golfarelli, Rizzi)
  - per uno specifico fatto, definisce schemi di fatto che modellano
    - dimensioni
    - gerarchie
    - misure
  - modello grafico a supporto della progettazione concettuale
  - offre una documentazione di progetto utile sia per la revisione dei requisiti con gli utenti, sia a posteriori

# Dimensional Fact Model

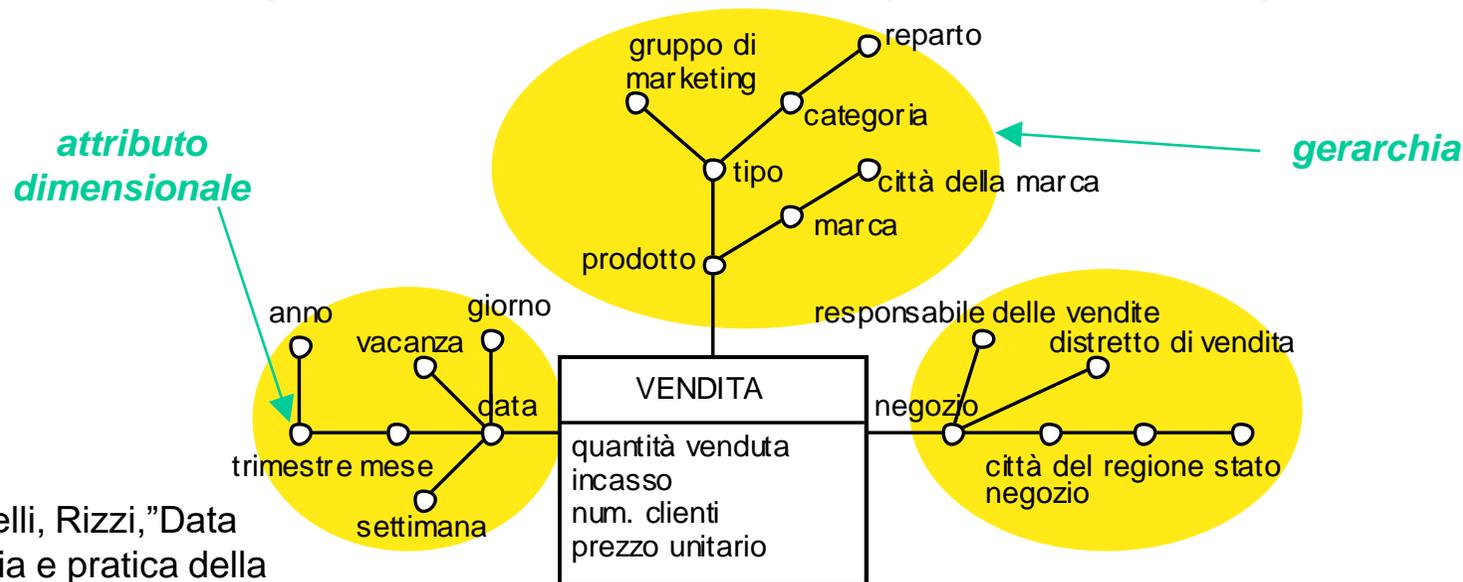
- Fatto
  - modella un insieme di eventi di interesse (vendite, spedizioni, reclami)
  - evolve nel tempo
- Dimensione
  - describe le coordinate di analisi di un fatto (ogni vendita è descritta dalla data di effettuazione, dal negozio e dal prodotto venduto)
  - è caratterizzata da numerosi attributi, tipicamente di tipo categorico
- Misura
  - describe una proprietà numerica di un fatto, spesso oggetto di operazioni di aggregazione (ad ogni vendita è associato un incasso)



Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

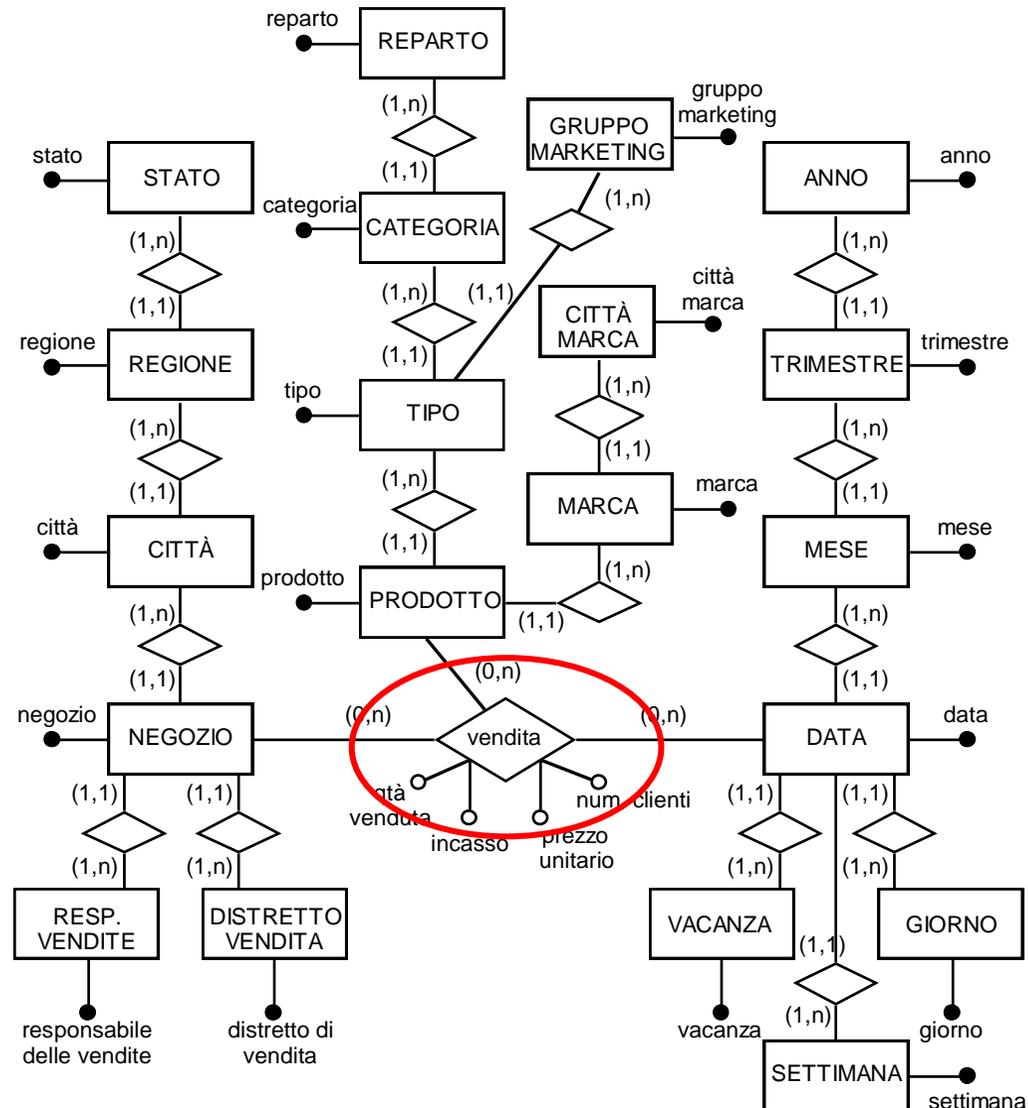
# Dimensional Fact Model

- Gerarchia
  - rappresenta una relazione di generalizzazione tra un sottoinsieme di attributi di una dimensione (gerarchia geografica per la dimensione negozio)
  - è una dipendenza funzionale (relazione 1:n)



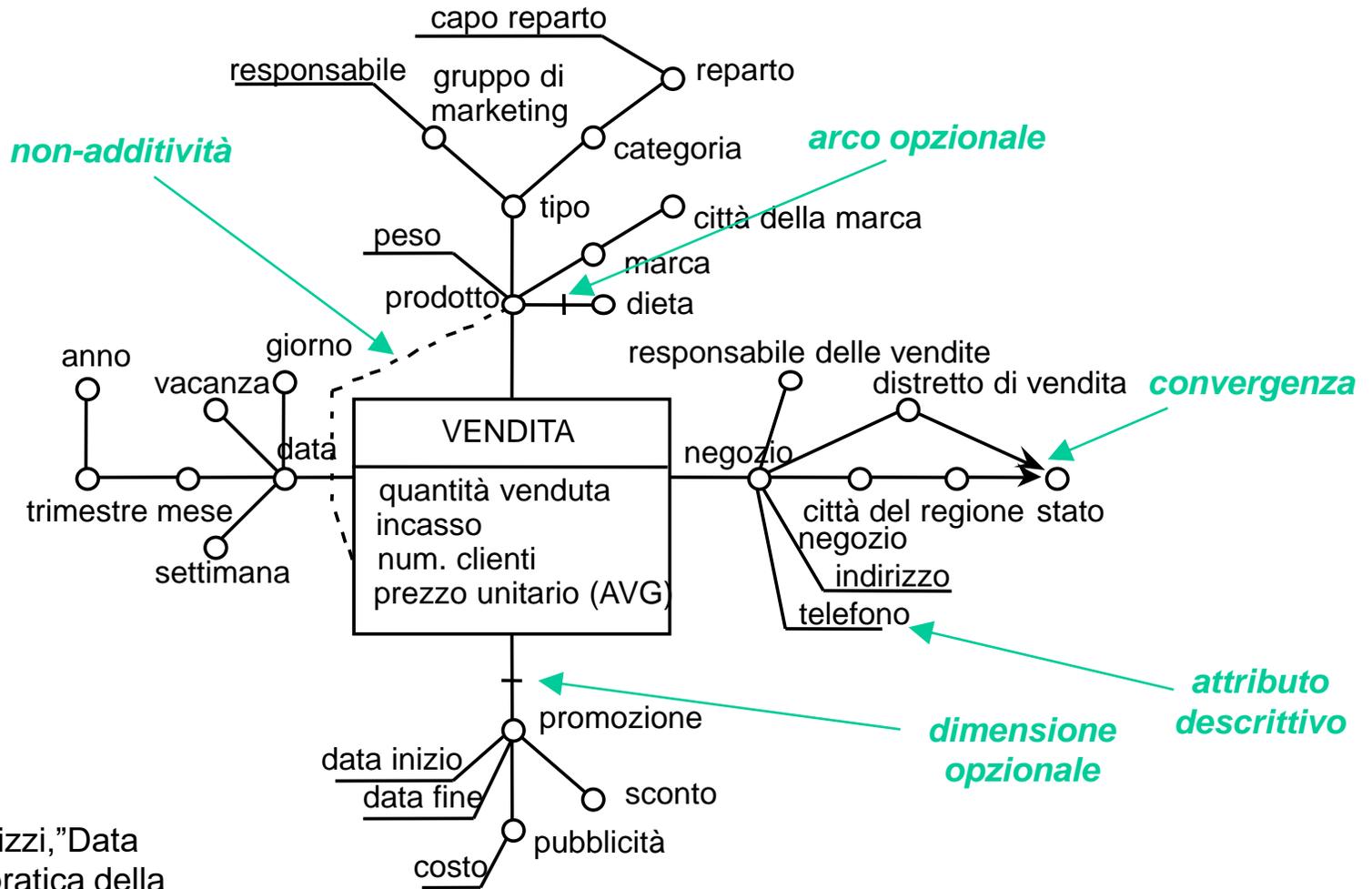
Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# Corrispondenza con l'ER



Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# DFM: costrutti avanzati



Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# Aggregazione

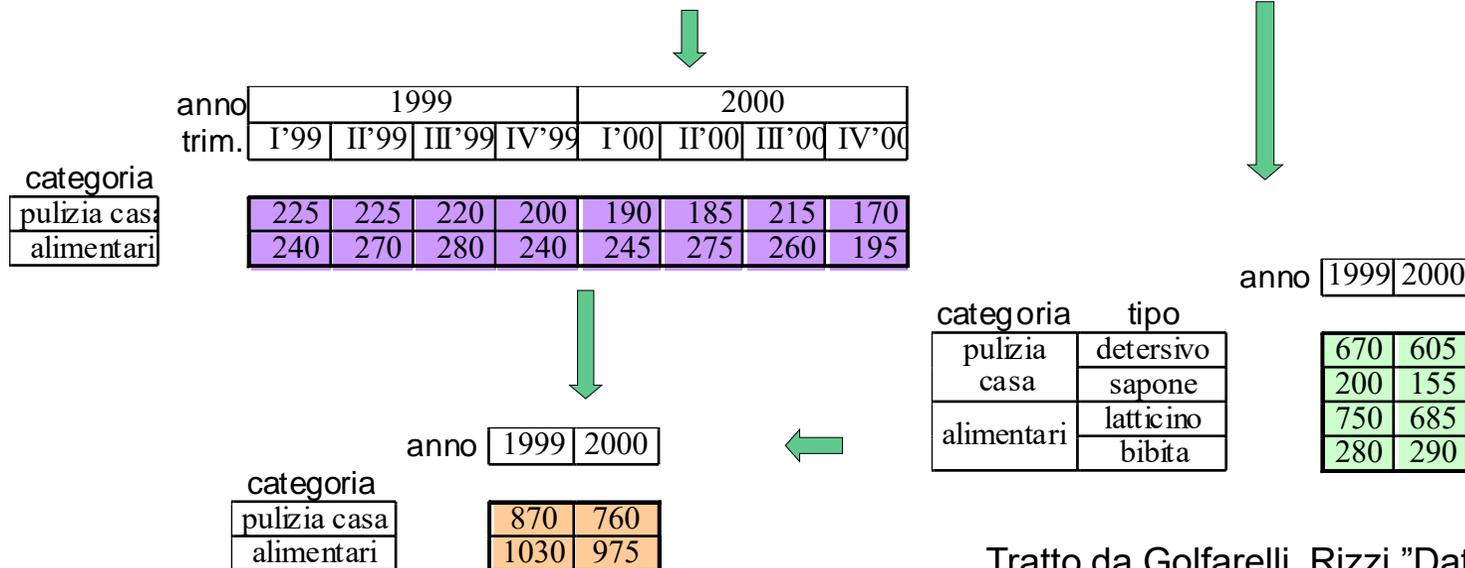
- Processo di calcolo del valore di misure a granularità meno fine di quella presente nello schema di fatto originale
  - la riduzione del livello di dettaglio è ottenuta risalendo lungo una gerarchia
  - operatori di aggregazione standard: SUM, MIN, MAX, AVG, COUNT
- Caratteristiche delle misure
  - additive
  - non additive: non aggregabili lungo una gerarchia mediante l'operatore di somma
  - non aggregabili

# Classificazione delle misure

- Misure di flusso
  - possono essere valutate cumulativamente alla fine di un periodo di tempo
  - sono aggregabili mediante tutti gli operatori standard
  - esempi: quantità di prodotti venduti, importo incassato
- Misure di livello
  - sono valutate in specifici istanti di tempo (snapshot)
  - non sono additive lungo la dimensione tempo
  - esempi: livello di inventario, saldo del conto corrente
- Misure unitarie
  - sono valutate in specifici istanti di tempo ed espresse in termini relativi
  - non sono additive lungo nessuna dimensione
  - esempio: prezzo unitario di un prodotto

# Operatori di aggregazione

categoria	tipo	prodotto	1999				2000			
			I '99	II '99	III '99	IV '99	I '00	II '00	III '00	IV '00
pulizia casa	detersivo	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
	sapone	Manipulite	15	20	25	30	15	15	20	10
		Scent	30	35	20	25	30	30	20	15
alimentari	latticino	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
	bibita	Bevimi	20	10	25	30	35	30	20	10
		Colissima	50	60	45	40	50	60	45	40

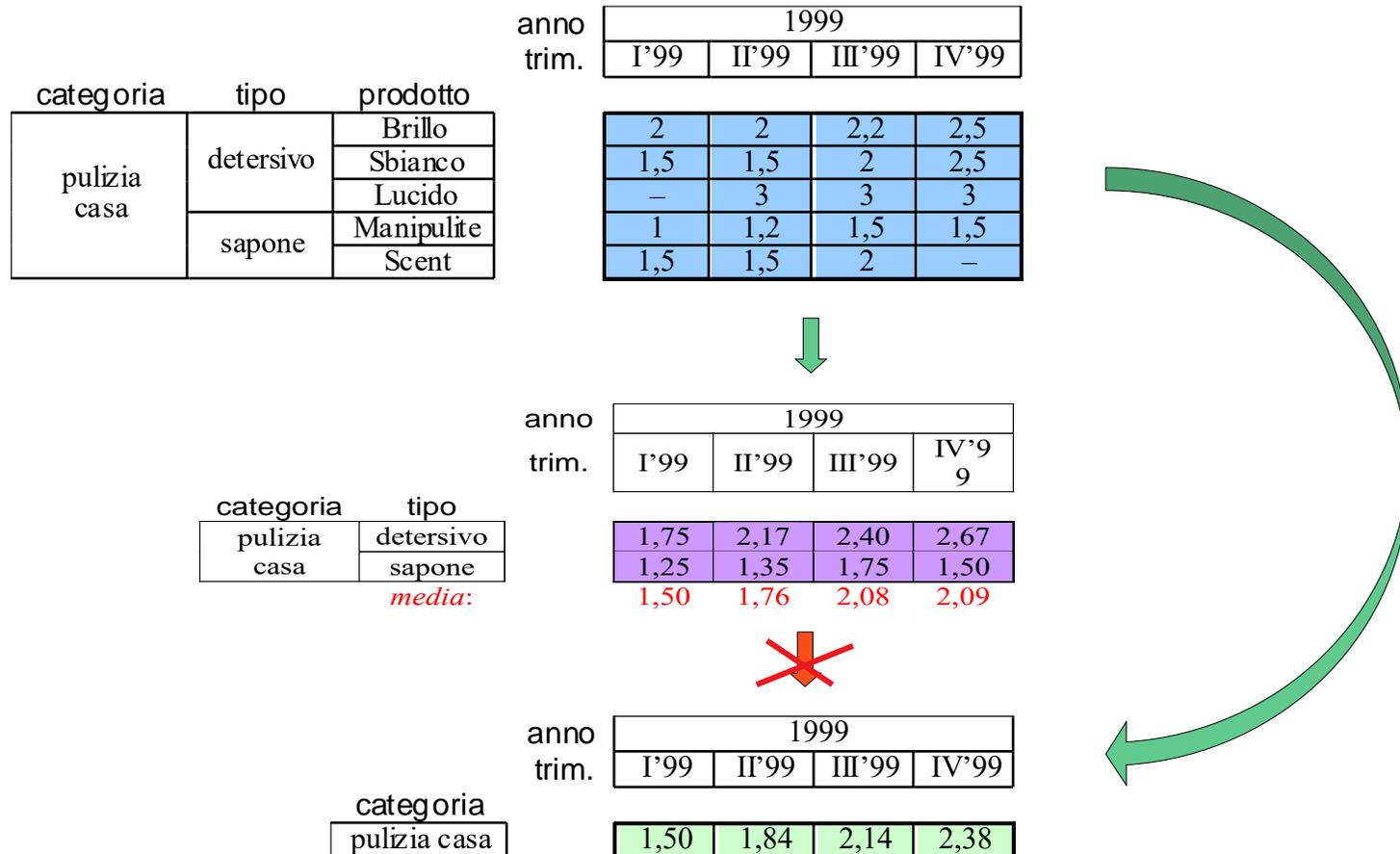


Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# Operatori di aggregazione

- Distributivi
  - sempre possibile il calcolo di aggregati da dati a livello di dettaglio maggiore
  - esempi: sum, min, max

# Operatori non distributivi

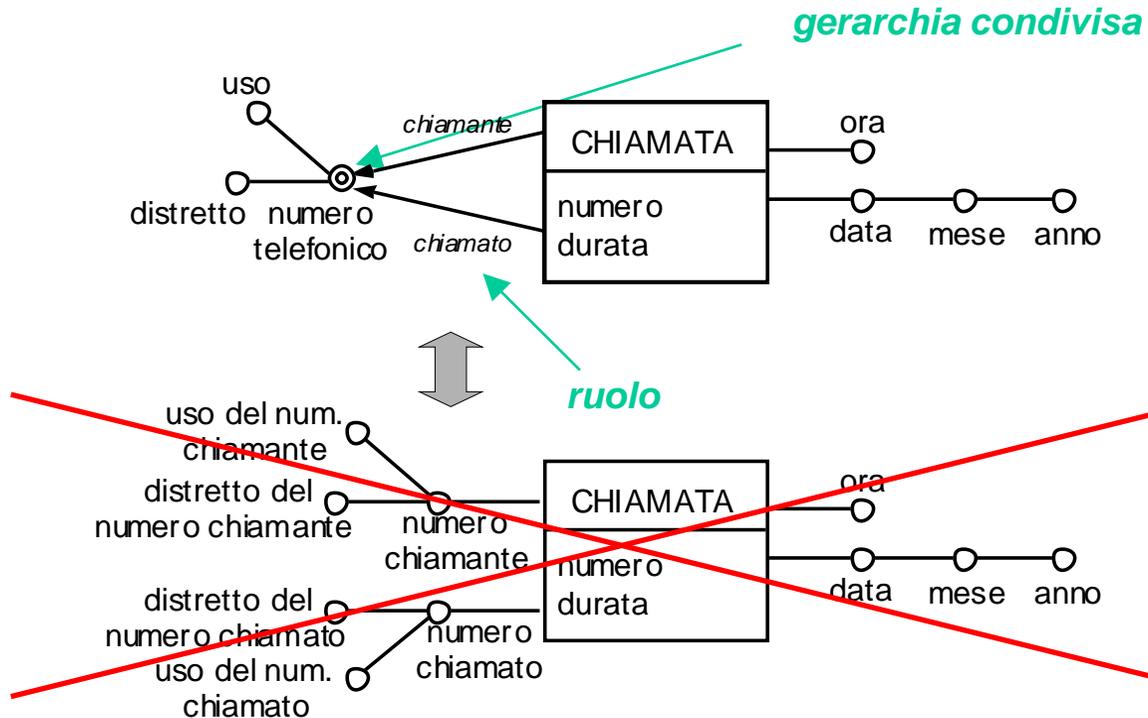


Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# Operatori di aggregazione

- Distributivi
  - sempre possibile il calcolo di aggregati da dati a livello di dettaglio maggiore
  - esempi: sum, min, max
- Algebrici
  - il calcolo di aggregati da dati a livello di dettaglio maggiore è possibile in presenza di misure aggiuntive di supporto
  - esempi: avg (richiede count)
- Olistici
  - non è possibile il calcolo di aggregati da dati a livello di dettaglio maggiore
  - esempi: moda, mediana

# DFM: costrutti avanzati

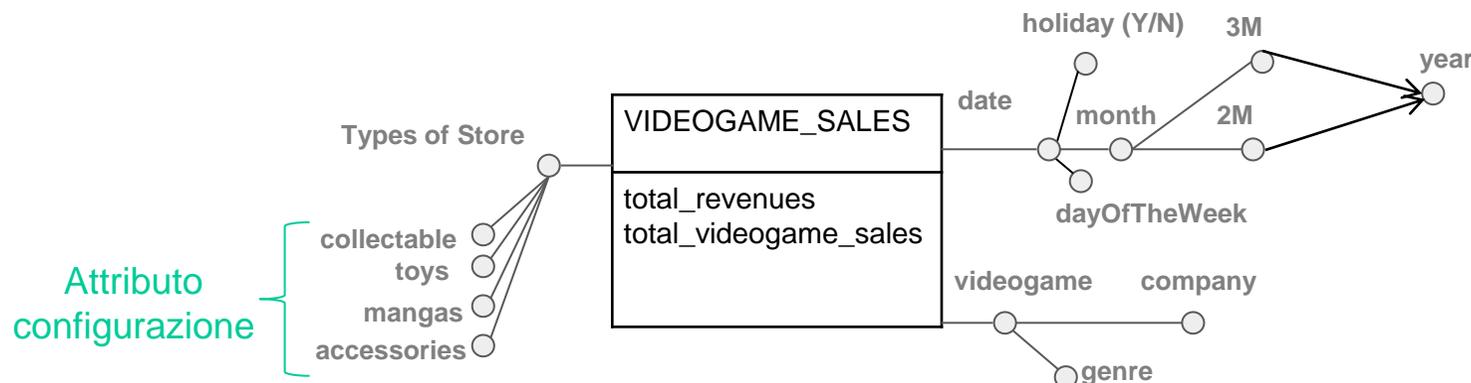


Tratti da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006



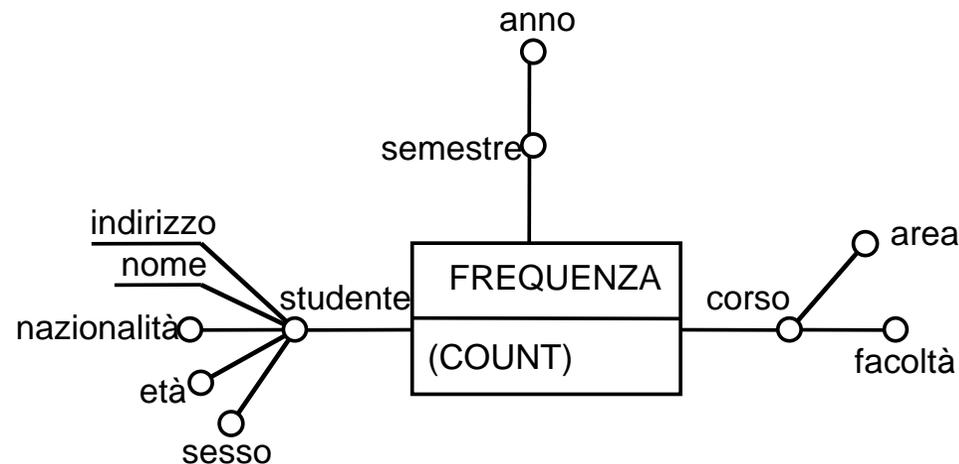
# Attributo configurazione

- Attributo categorico multivalore
  - può assumere contemporaneamente più valori
  - caratterizzato da pochi valori distinti ( $\leq 10$ )
- Rappresentazione mediante enumerazione dei valori possibili
  - ogni attributo assume valore booleano (Y/N)
  - consente scrittura più agevole di interrogazioni complesse



# Schemi di fatto vuoti

- L'evento può non essere caratterizzato da misure
  - schema di fatto vuoto
  - registra il verificarsi di un evento
- Utile per
  - conteggio di eventi accaduti
  - rappresentazione di eventi non accaduti (insieme di copertura)



Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# Rappresentazione del tempo

- La variazione dei dati nel tempo è rappresentata esplicitamente dal verificarsi degli eventi
  - presenza di una dimensione temporale
  - eventi memorizzati sotto forma di fatti
- Possono variare nel tempo anche le dimensioni
  - variazione tipicamente più lenta
    - slowly changing dimension [Kimball]
  - esempi: dati anagrafici di un cliente, descrizione di un prodotto
  - necessario prevedere esplicitamente nel modello come rappresentare questo tipo di variazione

# Modalità di rappresentazione del tempo (tipo I)

- Fotografia dell'istante attuale
  - esegue la sovrascrittura del dato con il valore attuale
  - proietta nel passato la situazione attuale
  - utilizzata quando non è necessario rappresentare esplicitamente la variazione
  - Esempio
    - il cliente Mario Rossi cambia stato civile dopo il matrimonio
    - tutti i suoi acquisti sono attribuiti al cliente “sposato”

# Modalità di rappresentazione del tempo (tipo II)

- Eventi attribuiti alla situazione temporalmente corrispondente della dimensione
  - per ogni variazione di stato della dimensione
    - si crea di una nuova istanza nella dimensione
    - i nuovi eventi sono correlati alla nuova istanza
  - gli eventi sono partizionati in base alle variazioni degli attributi dimensionali
  - Esempio
    - il cliente Mario Rossi cambia stato civile dopo il matrimonio
    - i suoi acquisti sono separati in acquisti attribuiti a Mario Rossi “celibe” e acquisti attribuiti a Mario Rossi “sposato” (nuova istanza di Mario Rossi)

# Modalità di rappresentazione del tempo (tipo III)

- Eventi attribuiti alla situazione della dimensione campionata in uno specifico istante di tempo
  - proietta tutti gli eventi sulla situazione della dimensione in uno specifico istante di tempo
  - richiede una gestione esplicita delle variazioni della dimensione nel tempo
    - modifica dello schema della dimensione
      - introduzione di una coppia di timestamp che indicano l'intervallo di validità del dato (inizio e fine validità)
      - introduzione di un attributo che consenta di identificare la sequenza di variazioni di una specifica istanza (capostipite o master)
    - ogni variazione di stato della dimensione richiede la definizione di una nuova istanza

# Modalità di rappresentazione del tempo (tipo III)

## – Esempio

- il cliente Mario Rossi cambia stato civile dopo il matrimonio
- la prima istanza conclude il suo periodo di validità il giorno del matrimonio
- la nuova istanza inizia la sua validità nello stesso giorno
- gli acquisti sono separati come nel caso precedente
- esiste un attributo che permette di ricostruire tutte le variazioni ascrivibili a Mario Rossi

# Carico di lavoro

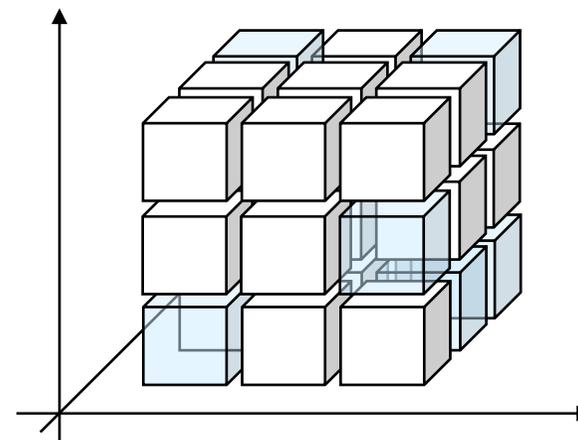
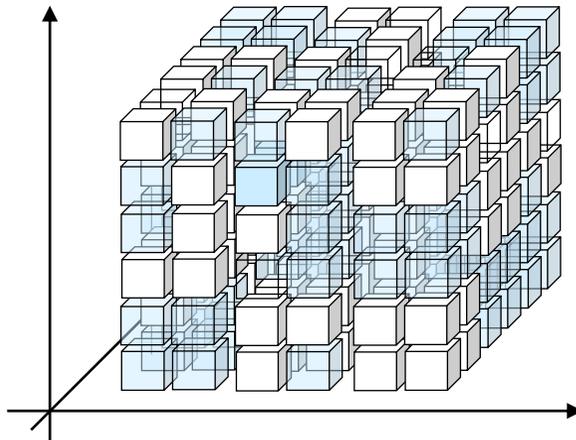
- Carico di riferimento definito da
  - reportistica standard
  - stime discusse con gli utenti
- Carico reale difficile da stimare correttamente durante la fase di progettazione
  - se il sistema ha successo, il numero di utenti e interrogazioni aumenta nel tempo
  - la tipologia di interrogazioni può variare nel tempo
- Fase di tuning
  - dopo l'avviamento del sistema
  - monitoraggio del carico di lavoro reale del sistema

# Volume dei dati

- Stima dello spazio necessario per il data mart
  - per i dati
  - per le strutture accessorie (indici, viste materializzate)
- Si considerano
  - numero di eventi di ogni fatto
  - numero di valori distinti degli attributi nelle gerarchie
  - lunghezza degli attributi
- Dipende dall'intervallo temporale di memorizzazione dei dati
- Valutazione affetta dal problema della sparsità
  - il numero degli eventi accaduti non corrisponde a tutte le possibili combinazioni delle dimensioni
  - esempio: percentuale dei prodotti effettivamente venduti in ogni negozio in un dato giorno pari circa al 10% di tutte le possibili combinazioni

# Sparsità

- Si riduce al crescere del livello di aggregazione dei dati
- Può ridurre l'affidabilità della stima della cardinalità dei dati aggregati



Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# *Progettazione logica*

Elena Baralis  
Politecnico di Torino

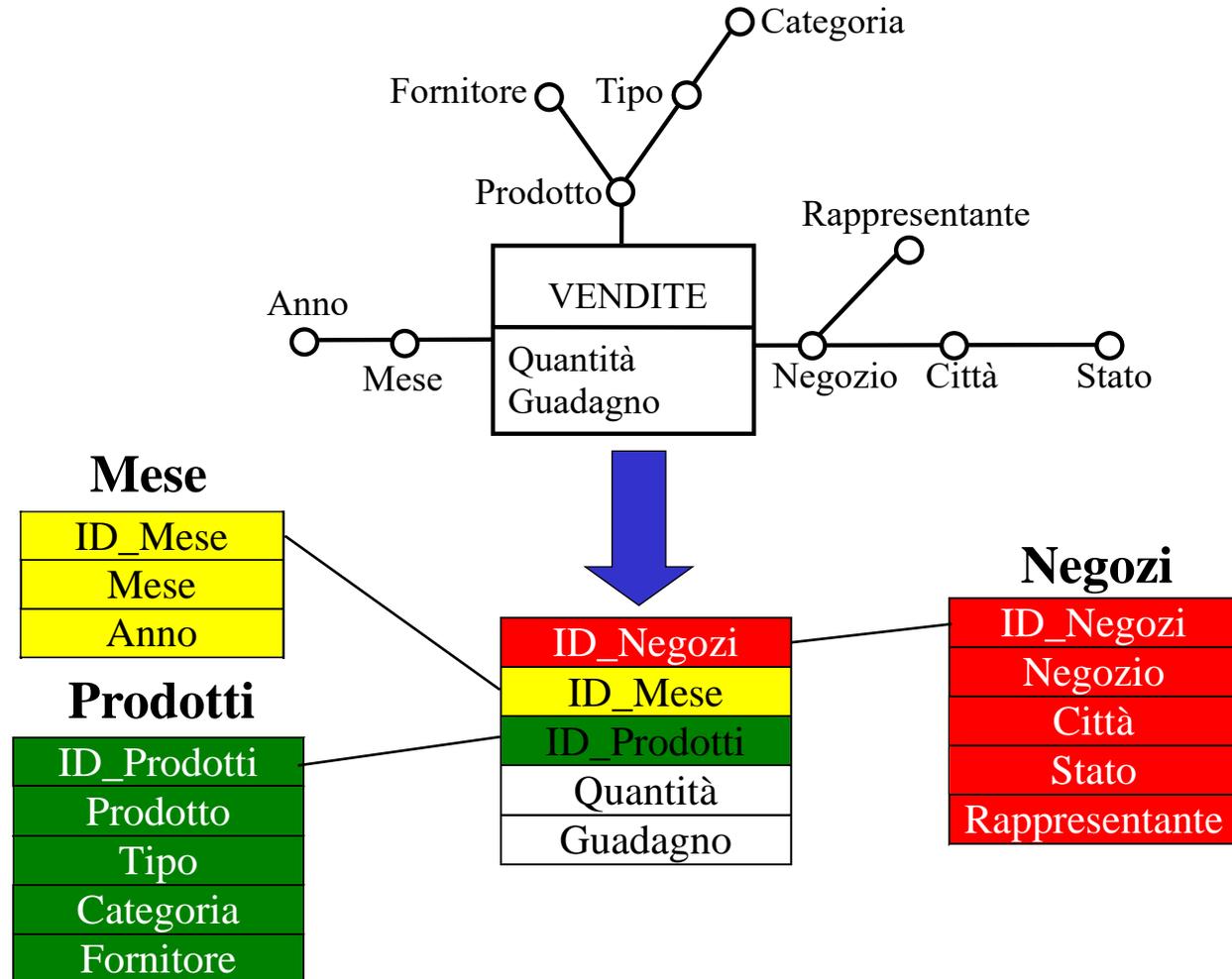
# Progettazione logica

- Si considera il modello relazionale (ROLAP)
  - inputs
    - schema (di fatto) concettuale
    - carico di lavoro
    - volume dei dati
    - vincoli di sistema
  - output
    - schema logico relazionale
- Basata su principi diversi rispetto alla progettazione logica tradizionale
  - ridondanza dei dati
  - denormalizzazione delle tabelle

# Schema a stella

- Dimensioni
  - una tabella per ogni dimensione
  - chiave primaria generata artificialmente (surrogata)
  - contiene tutti gli attributi della dimensione
  - gerarchie non rappresentate esplicitamente
    - gli attributi della tabella sono tutti allo stesso livello
  - rappresentazione completamente denormalizzata
    - presenza di ridondanza nei dati
- Fatti
  - una tabella dei fatti per ogni schema di fatto
  - chiave primaria costituita dalla combinazione delle chiavi esterne delle dimensioni
  - le misure sono attributi della tabella

# Schema a stella

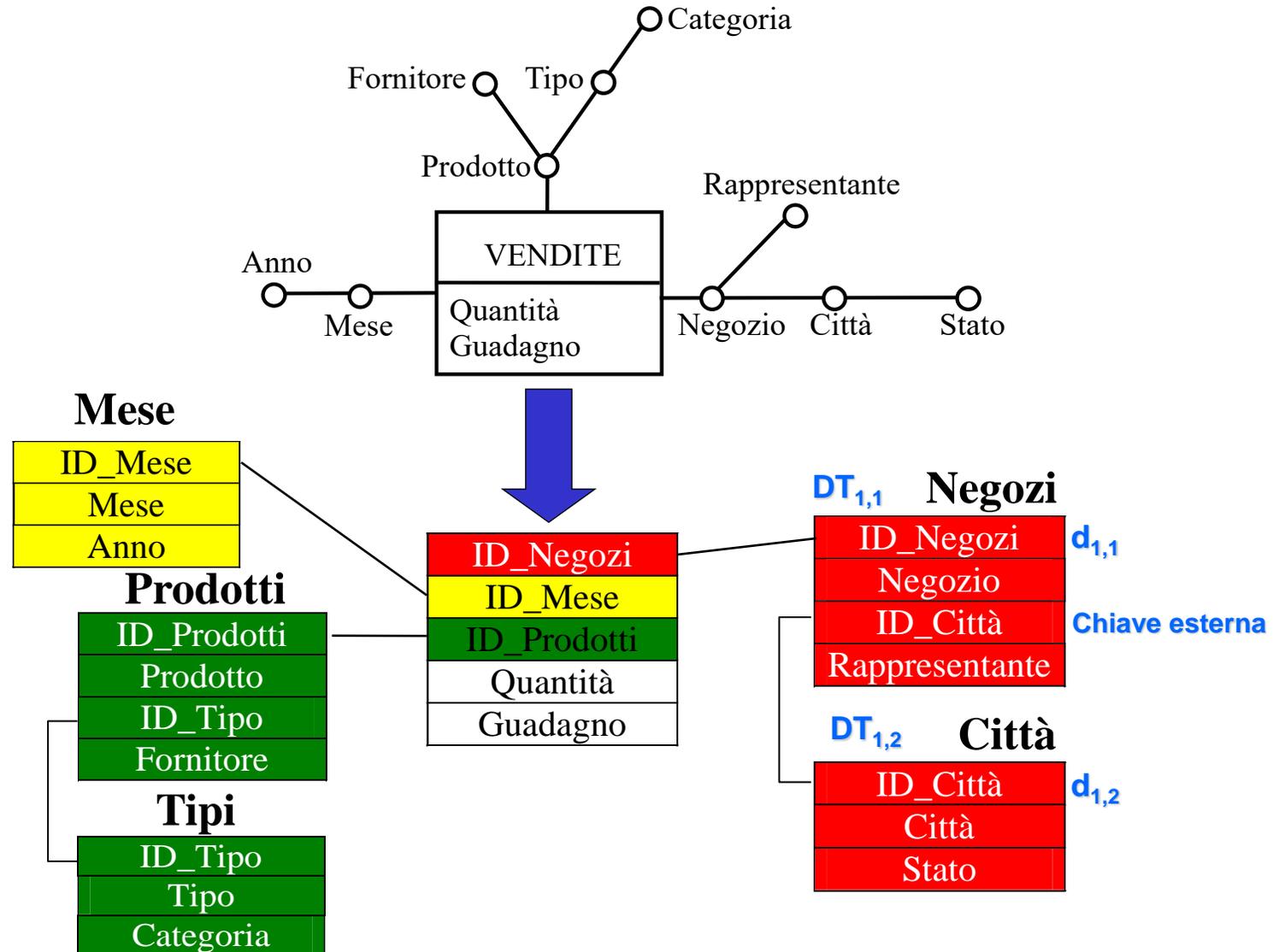


Tratto da Golfarelli, Rizzi, "Data warehouse, teoria e pratica della progettazione", McGraw Hill 2006

# Snowflake schema

- Separazione di (alcune) dipendenze funzionali frazionando i dati di una dimensione in più tabelle
  - si introduce una nuova tabella che separa in due rami una gerarchia dimensionale (taglio su un attributo della gerarchia)
  - una nuova chiave esterna esprime il legame tra la dimensione e la nuova tabella
- Si riduce lo spazio necessario per la memorizzazione della dimensione
  - riduzione non significativa
- Aumenta il costo di ricostruzione dell'informazione della dimensione
  - è necessario il calcolo di uno o più join

# Snowflake schema



# Star o snowflake?

- Lo schema snowflake è normalmente *sconsigliato*
  - la riduzione di spazio occupato è scarsamente benefica
    - l'occupazione maggiore di spazio è dovuta alla tabella dei fatti (la differenza è pari ad alcuni ordini di grandezza)
  - il costo di eseguire più join può essere significativo
- Lo schema snowflake è raramente utilizzato nella progettazione di data mart

# Archi multipli



- Soluzioni realizzative

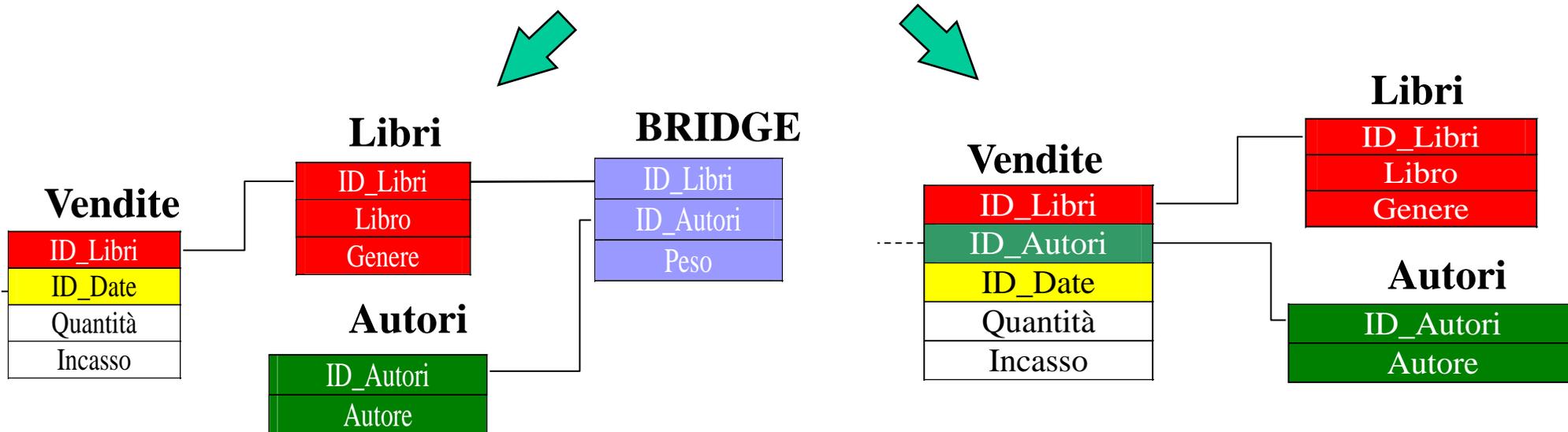
- bridge table

- tabella aggiuntiva che modella la relazione molti a molti
- nuovo attributo che consenta di pesare la partecipazione delle tuple nella relazione

- push down

- arco multiplo integrato nella tabella dei fatti
- nuova dimensione corrispondente nella tabella dei fatti

# Archi multipli



# Archi multipli

- Tipologie di interrogazione
  - pesate: considerano il peso dell'arco multiplo
    - esempio: incasso di ciascun autore
    - con bridge table
  - di impatto: non considerano il peso
    - esempio: numero di copie vendute per ogni autore
    - con bridge table

```
SELECT ID_Autori, SUM(Incasso*Peso)
```

```
...
```

```
group by ID_Autori
```

```
SELECT ID_Autori, SUM(Quantità)
```

```
...
```

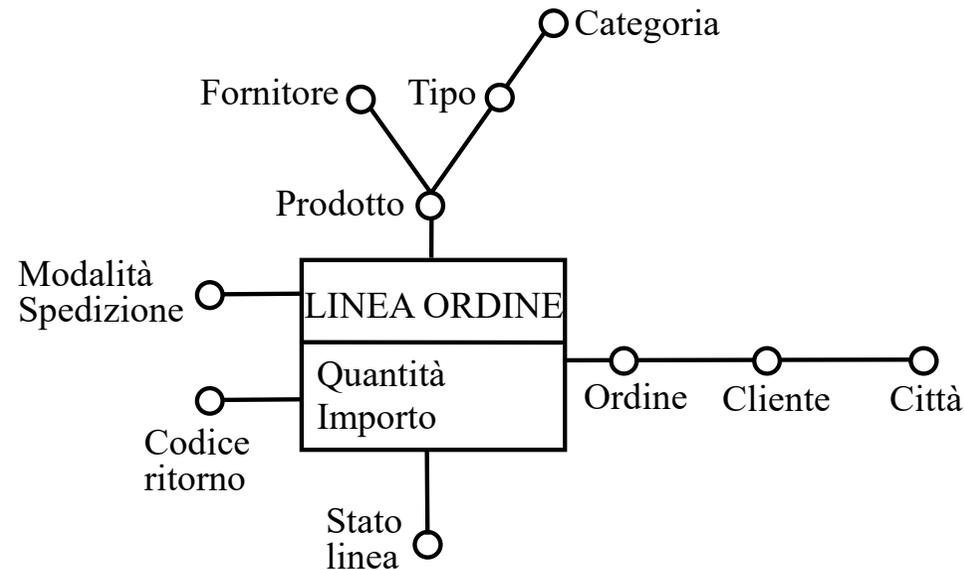
```
group by ID_Autori
```

# Archi multipli

- Confronto tra le soluzioni realizzative
  - il peso è esplicitato nella bridge table, ma integrato nella tabella dei fatti per push down
    - (push down) difficile eseguire interrogazioni di impatto
    - (push down) calcolo del peso durante l'alimentazione
    - (push down) modifiche successive difficoltose
  - push down introduce una forte ridondanza nella tabella dei fatti
  - costo di esecuzione delle interrogazioni minore per push down
    - numero minore di join

# Dimensioni degeneri

- Dimensioni rappresentate da un solo attributo



# Dimensioni degeneri

- Soluzioni realizzative
  - integrazione nella tabella dei fatti
    - per attributi di dimensione (molto) contenuta
  - junk dimension
    - unica dimensione che integra più dimensioni degeneri
    - non esistono dipendenze funzionali tra gli attributi della dimensione
      - sono possibili tutte le combinazioni
      - attuabile solo per cardinalità limitate del dominio degli attributi

# Junk dimension

