Sistemi Informativi Avanzati
Anno Accademico 2015/2016
Prof. Domenico Beneventano

Progettazione concettuale

Dal Capitolo 6 del libro

Data Warehouse - teoria e pratica della Progettazione
Autori: Matteo Golfarelli, Stefano Rizzi;

Editore: McGraw-Hill

Progettazione concettuale: approcci

- Basata sui requisiti
  - ✓ Il progettista deve essere in grado di enucleare, dalle interviste condotte presso l'utente, un'indicazione precisa circa i fatti da rappresentare, le misure che li descrivono e le gerarchie attraverso cui aggregarli utilmente. Il problema del collegamento tra lo schema concettuale così determinato e le sorgenti operazionali viene affrontato in un secondo tempo

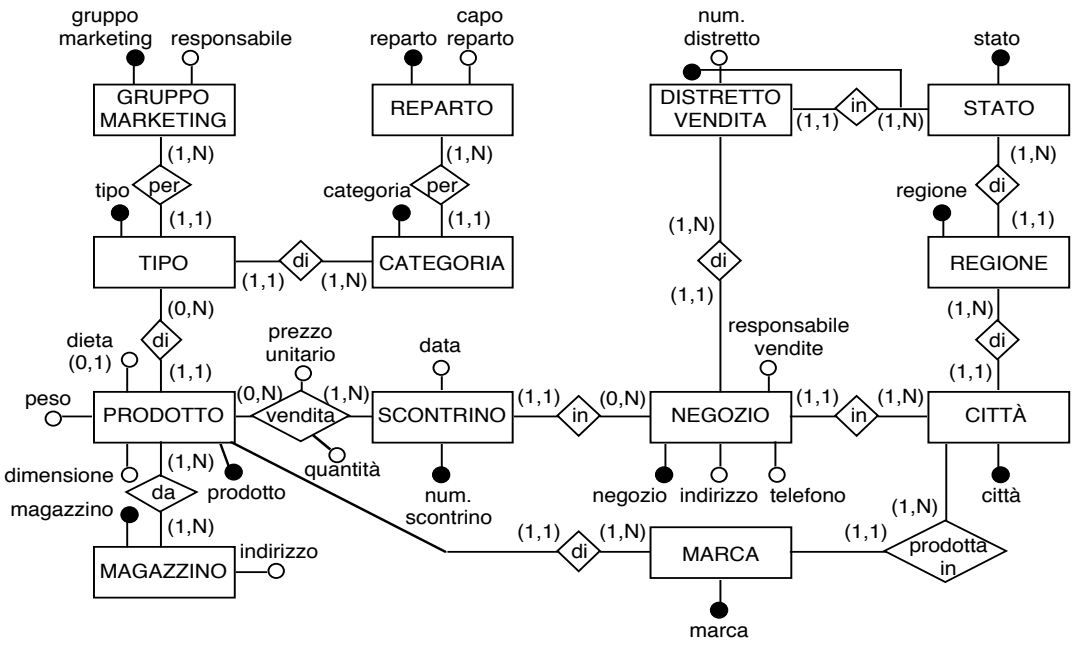
- Basata sulle sorgenti 
  - ✓ È possibile definire lo schema concettuale in funzione della struttura delle sorgenti, evitando il complesso compito di stabilire il legame con esse a posteriori. Inoltre, è possibile derivare uno schema concettuale prototipale dagli schemi operazionali in modo pressoché automatico

Progettazione concettuale: come

- La progettazione concettuale viene effettuata a partire dalla documentazione del database operazionale (DBO) riconciliato: DBO con il suo schema LOGICO relazionale, il relativo schema E/R *corrispondente* e l'eventuale documentazione aggiuntiva
- Alcune informazioni/vincoli (in particolare chiavi e dipendenze funzionali) possono essere solo nello schema LOGICO, o solo nello schema E/R o solo nella documentazione aggiuntiva.

- Passi di progettazione:
  1. Definizione dei fatti
  2. Per ogni fatto:
     1. Costruzione di un *albero degli attributi*
     2. Editing dell'albero degli attributi
     3. Definizione delle dimensioni
     4. Definizione delle misure
     5. Creazione dello schema di fatto

Esempio delle vendite: schema E/R

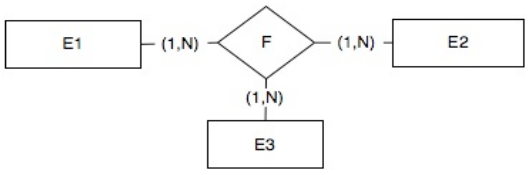


Definizione dei fatti

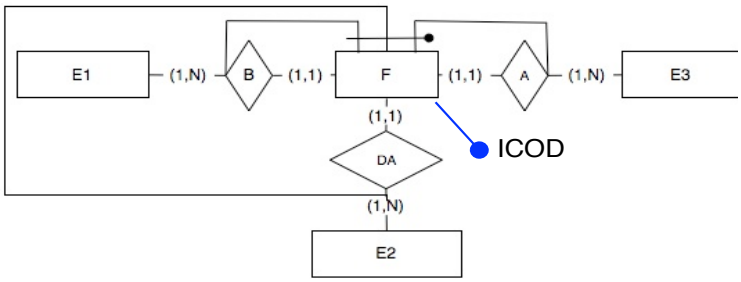
- I fatti sono elementi di interesse primario per il processo decisionale; tipicamente, corrispondono a eventi che accadono dinamicamente nel mondo aziendale
- Nella progettazione basata su sorgenti, i concetti di uno schema che rappresentano archivi frequentemente modificati (come VENDITA) sono buoni candidati per definire fatti; quelli che rappresentano archivi quasi-statici (come NEGOZIO e CITTÀ) no
- Ogni fatto identificato diviene la radice di un nuovo schema

Definizione dei fatti da schemi E/R

- Nello schema E/R un fatto corrisponde ad un'associazione n-aria F tra le entità E1, E2..., En



- L'associazione può essere in forma reificata con *identificatore semantico* I = {E1, E2..., En}



- Spesso in pratica è riportato solo un identificatore surrogato **ICOD**
- In fase di analisi si dovrebbero individuare tutti gli identificatori del fatto in modo da considerarli nelle progettazione concettuale

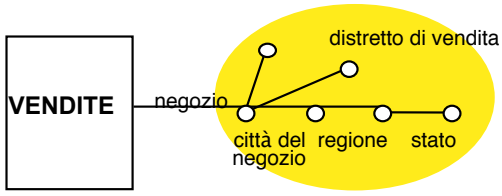
➢ Nel seguito ipotizziamo che tutti gli identificatori primari e alternativi siano esplicitati nello schema E/R e/o nel relazionale

Definizione delle gerarchie

- Si ricavano dalle *dipendenze funzionali (FD)* che sono
  ① Dichiarate nello schema E/R, e/o
  ② Dichiarate nello schema logico, e/o
  ③ Descritte nella documentazione
- Partendo da **tutte** le FD disponibili, nella gerarchia si riportano **solo** quelle che il progettista ritiene utili ai fini dell'analisi dei dati !
  - **Solo** le FD della gerarchia **verranno implementati** nel sistema OLAP. Le FD non riportate nella gerarchia influenzano comunque il risultato dell'analisi.
- Per effettuare efficacemente questa cruciale fase di progettazione concettuale,
  le FD si rappresentano attraverso l'***albero degli attributi***

Gerarchie: considerazione

- Gerarchia su negozio

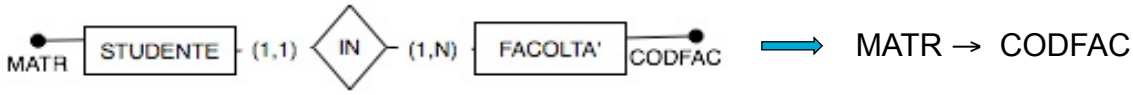


- La **FD: regione → stato** riportata nella gerarchia di Negozio consentirà di fare un *roll-up* da {Regione} a {Stato } e/o una query MDX per avere tutti le *regioni* di uno *stato*
- La **FD: Distretto_di_Vendita → Stato** non è riportata nella gerachia, "*non è possibile*" un roll-up da {Distretto_di_Vendita} a {Stato} e/o una query MDX per avere tutti i *distretti* di uno *stato*
- Però Distretto_di_Vendita → Stato è valida nei dati, quindi un pattern Distretto_di_Vendita e Stato è non significativo:
  limitazione del modello DFM che non cattura questo aspetto!

FD dichiarate nello schema E/R

- Regole per individuare le *FD* di uno schema E/R
  1. Data un'entità X, un suo identificatore denotato con *Identif(X)*, determina funzionalmente tutti gli attributi di X

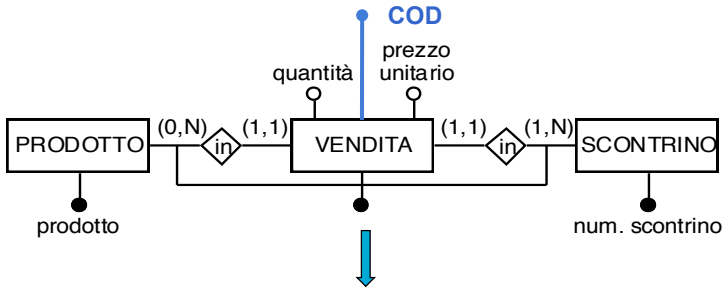
  2. Un'associazione **uno-a-molti** tra l'entita X (che partecipa con molteplicità uno) e l'entita Y esprime la FD *Identif(X) → Identif(Y)*



  3. Un'associazione **uno-a-uno** tra l'entita X e l'entita Y esprime due FD: *Identif(X) → Identif(Y)* e *Identif(Y) → Identif(x)*

FD dichiarate nello schema E/R

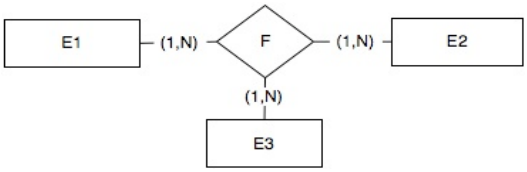
- Dalle regole 1 e 2 deriva che un'associazione n-aria R tra X1, X2, …, Xn (espressa in forma reificata) esprime
  1) Per ogni i: *{ Identif(X1), Identif(X2), …, Identif(Xn) } → Identif(Xi)*
  2) Per ogni attributo A di R: *{ Identif(X1), Identif(X2), …, Identif(Xn) } → A*



{ prodotto,num.scontrino } → prezzo_unitario
{ prodotto,num.scontrino } → quantità
{ prodotto,num.scontrino } → COD
COD → prezzo_unitario
COD → quantità
COD → prodotto
COD → num.scontrino

FD dichiarate nello schema logico

- Dato lo schema E/R

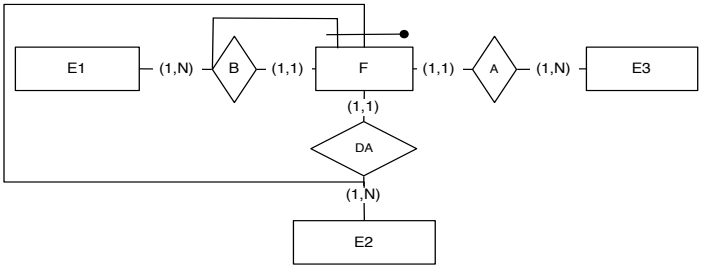


- La FD {E1,E2} → E3 viene espressa **solo** nello schema logico:

⟹ F(E1,E2,E3)

FORNIT(E1=AZIENDA,E2=PRODOTTO,E3=FORNITORE)

- Situazione frequente in pratica.
  Motivo :
  non *complicare* lo schema E/R



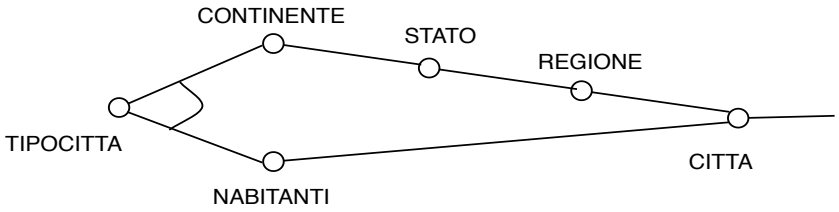
FD dichiarate nella documentazione

- Nell'esempio sono riportate in E/R solo le FD : X -> Y
  dove X è singolo (compromesso tra espressività e leggibilità)

- Documentazione dello schema: il TIPOCITTA **dipende** dal numero abitanti e dal continente, ovvero vale la seguente FD:

FD3: `CONTINENTE,NABITANTI → TIPOCITTA`

- Per la FD3 TIPOCITTA è un attributo **cross-dimensionale:**



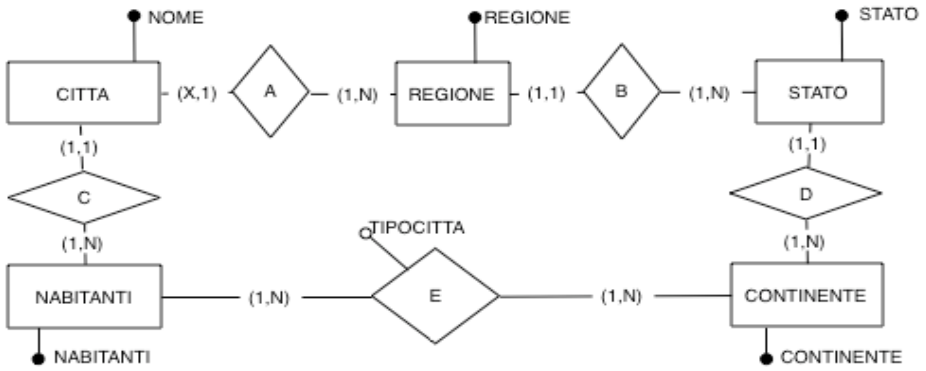
- Nella gerarchia vengono indicate solo le FD dirette, quindi **non** si riporta l'arco corrispondente alla FD: CITTA → TIPOCITTA.

FD dichiarate nella documentazione

- La FD3 si può anche aggiungere *facilmente* allo schema logico, attraverso la relazione TIPOCITTA:

```
CITTA(NOME,REGIONE:REGIONE,NABITANTI)
REGIONE(REGIONE,STATO:STATO)
STATO(STATO,CONTINENTE:CONTINENTE)
CONTINENTE(CONTINENTE)
TIPOCITTA(NABITANTI,CONTINENTE:CONTINENTE,TIPOCITTA)
```

- Invece per dichiarare la FD3 direttamente in E/R, occorre complicare notevolmente lo schema rappresentando CONTINENTE e NABITANTI come entità:

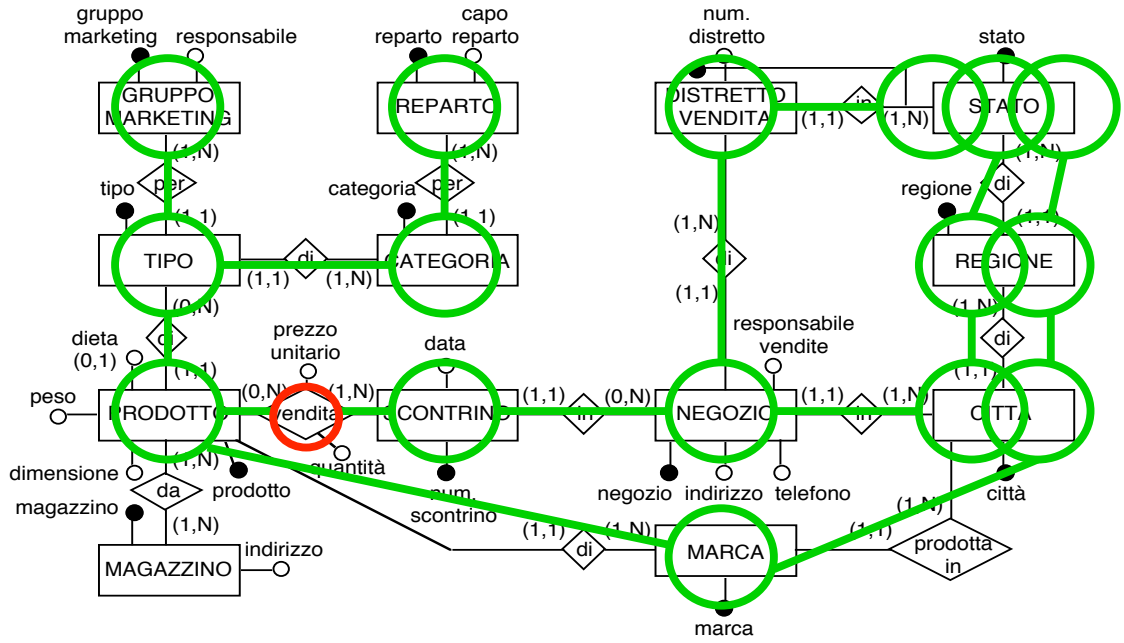


Costruzione dell'albero degli attributi

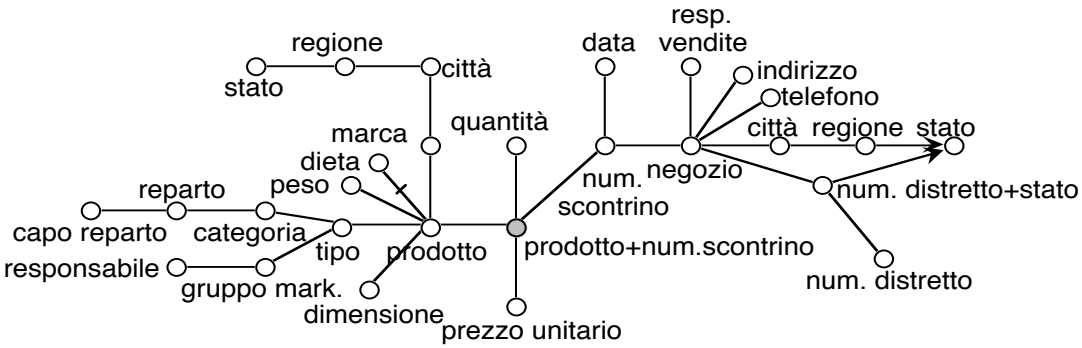
- L'albero degli attributi è un albero in cui:
  1. ogni vertice corrisponde a un attributo - semplice o composto - dello schema sorgente;

     L'attributo composto {A1,…,An} è denotato con A1+…+An
  2. la radice corrisponde all'identificatore primario di F;
  3. per ogni vertice v, l'attributo corrispondente determina funzionalmente tutti gli attributi figli
     - Per la transitività delle FD un vertice v determina tutti i suoi discendenti di v
- Partendo dallo schema E/R, l'albero degli attributi *iniziale* corrispondente a F può essere costruito in modo automatico *navigando ricorsivamente* le FD dichiarate nello schema
- L'albero completo si ottiene da quello iniziale considerando le FD dichiarate nel logico o nella documentazione

Esempio delle vendite



L'esempio delle vendite

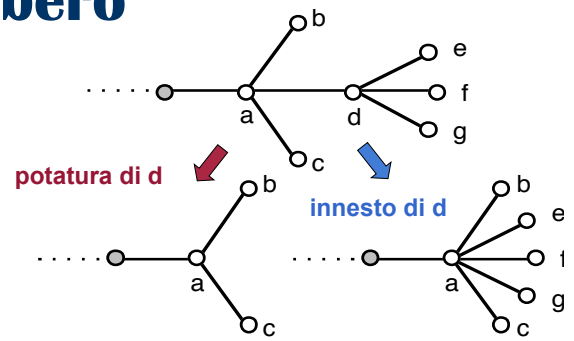


- L'albero degli attributi iniziale contiene tutte le FD dello schema E/R.
- A questo livello gli attributi dimensionali comuni (citta, regione, stato) sono ripetuti: le condivisioni\convergenze si analizzano solo dopo aver deciso quali attributi tenere.
- Nell'esempio: è stata già decisa la convergenza su STATO

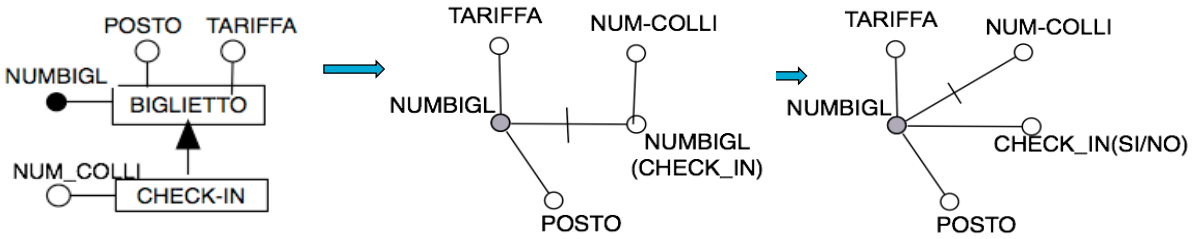
Editing dell'albero

- In genere non tutti gli attributi dell'albero sono d'interesse per il data mart; quindi, l'albero può essere manipolato per eliminare i livelli di dettaglio non necessari
  - ✓ La potatura di un vertice v si effettua eliminando l'intero sottoalbero con radice in v
    - Gli attributi eliminati non verranno inclusi nello schema di fatto, quindi non potranno essere usati per aggregare i dati
  - ✓ L'innesto viene utilizzato quando, sebbene un vertice esprima un'informazione non interessante, è necessario mantenere nell'albero i suoi discendenti
    - L'innesto del vertice v, con padre v', viene effettuato collegando tutti i figli di v direttamente a v' ed eliminando v; come risultato verrà perduto il livello di aggregazione corrispondente all'attributo v ma non i livelli corrispondenti ai suoi discendenti

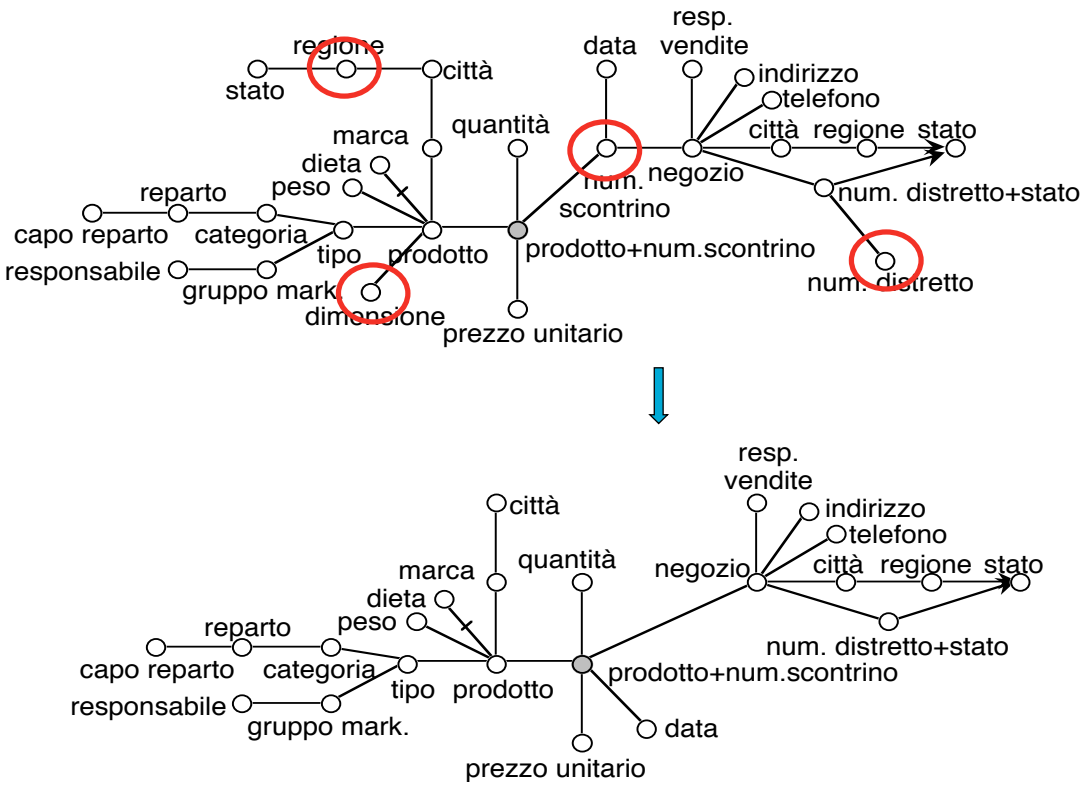
Editing dell'albero



- Quando un vertice opzionale viene innestato, tutti i suoi figli ereditano la proprietà di opzionalità
  - ✓ Nel caso di potatura o innesto di un vertice opzionale v con padre v' è possibile aggiungere a v' un nuovo figlio b corrispondente a un attributo booleano che esprima l'opzionalità. **Esempio:**

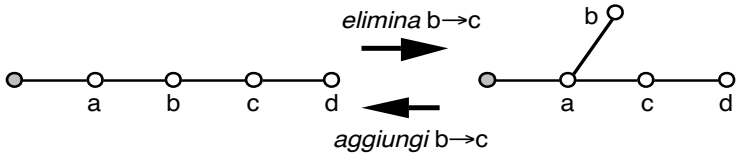


L'esempio delle vendite



Editing dell'albero: elimina FD

- **Eliminazione** di dipendenze funzionali

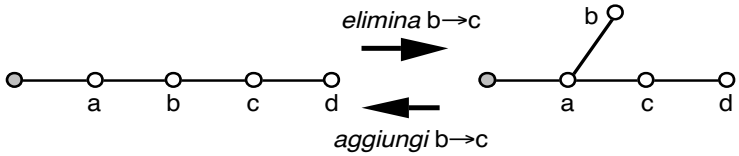


- Eliminare b→c dall'albero **non significa** che b→c non sarà più vera nei dati, cioè nello schema E/R!
  - ✓ Se b→c non fosse vera, dovrei eliminarla dallo schema E/R e di conseguenza non comparirebbe più nell'albero degli attributi!

- Eliminare b→c dall'albero significa che **non interessa** tale dipendenza funzionale in fase di **analisi dei dati**.
  - ✓ non siamo interessati ad effettuare un *roll-up* sulla base di b→c
  - ✓ D'altra parte la dipendenza b→c è valida nei dati, quindi un pattern contenente {b, c} è **non significativo!**

Editing dell'albero: aggiungi FD

- **Aggiunta** di dipendenze funzionali



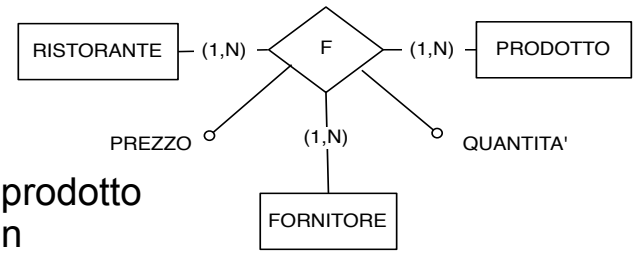
- La FD: b→c aggiunta può derivare da:

① **Una FD valida nel DBO**, non espressa nello schema E/R (e quindi non presente nell'albero degli attributi iniziale) ma dichiarata nel logico o nella documentazione

② **Una FD non valida nel DBO**, ma richiesta nei requisiti (approccio basato sui requisiti …). Vediamo un esempio:

FD derivante dai requisiti

- Fatto **F=Forniture**



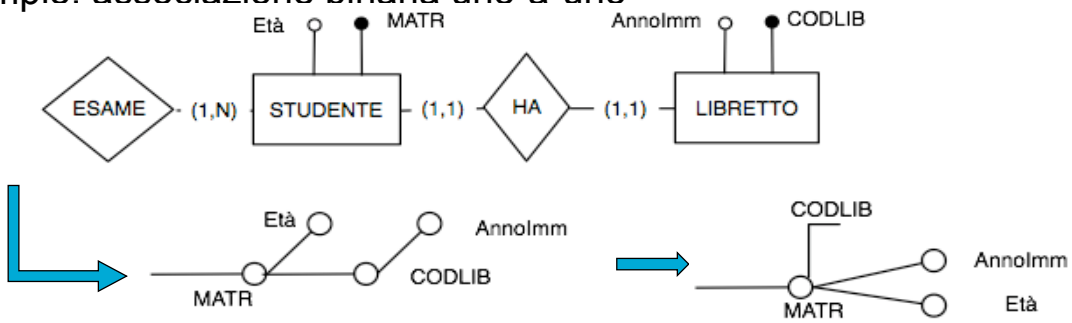
- **Requisiti:** analizzare **solo** le forniture esclusive in cui un certo prodotto viene fornito in modo esclusivo ad un ristorante, da un solo fornitore, ovvero

⟹ **FD:** `RISTORANTE,PRODOTTO → FORNITORE`

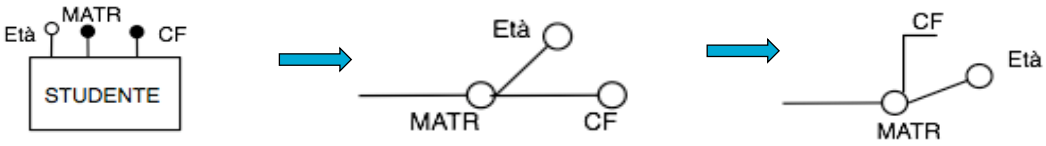
- Questa FD non è valida nel DBO, non deve essere imposta al DBO, che deve continuare a gestire tutte le forniture, non solo quelle *esclusive*.
- Due possibili soluzioni
  - ① Architetture a **due livelli**: la FD è imposta solo nel data mart e quindi viene aggiunta all'albero degli attributi
  - ② Architettura a **tre livelli**: la FD è imposta nel DBO riconciliato e quindi comparirà nell'albero degli attributi

Editing dell'albero

- Se ho due vertici v1 e v2 con v1→v2 e v2→v1, devo *fondere* v1 e v2 in un unico vertice eliminando v1 (v2) tramite innesto
  - ✓ Il vertice eliminato può essere aggiunto come attributo descrittivo
- Esempio: associazione binaria uno-a-uno



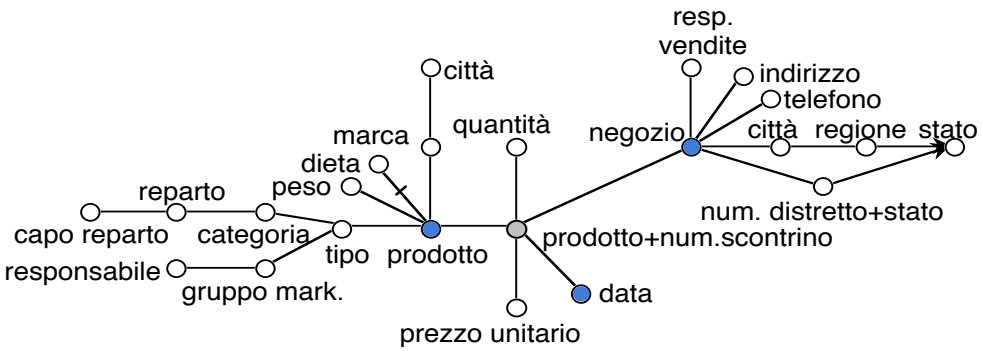
- Esempio: Entità con più di un identificatore



Definizione delle dimensioni

- Le dimensioni vanno scelte nell'albero degli attributi tra i vertici **figli diretti della radice**; possono essere attributi discreti o a intervalli di valori di attributi discreti o continui
  - ✓ *Attributi discreti:* spesso una dimensione corrisponde ad un attributo che è identificatore di una entità (nell'esempio delle vendite: prodotto e negozio)
  - ✓ *Intervalli di valori:* un classico esempio è il caso di un attributo *età* che viene discretizzato in *fasce d'età*
  - ✓ *Dimensione temporale:* un altro caso tipico di intervalli di valori è passare da un attributo *data* ad una dimensione *anno* oppure *mese*
- La scelta delle dimensioni è cruciale per il progetto poiché definisce la *granularità* degli eventi primari

L'esempio delle vendite



Granularità di uno schema di fatto

- **Transazionale**: uno schema di fatto è a granularità transazionale se un evento primario corrisponde ad una sola transazione operazionale
- **Temporale**: uno schema di fatto è a granularità temporale se un evento primario raggruppa in sé un insieme di transazioni operazionali

Progettazione
basata sulle sorgenti

Uno schema di fatto corrispondente ad un fatto F di uno schema E/R è a ***granularità transazionale*** se le dimensioni contengono ***almeno un identificatore*** del fatto F, altrimenti è a granularità temporale.

Granularità di uno schema di fatto

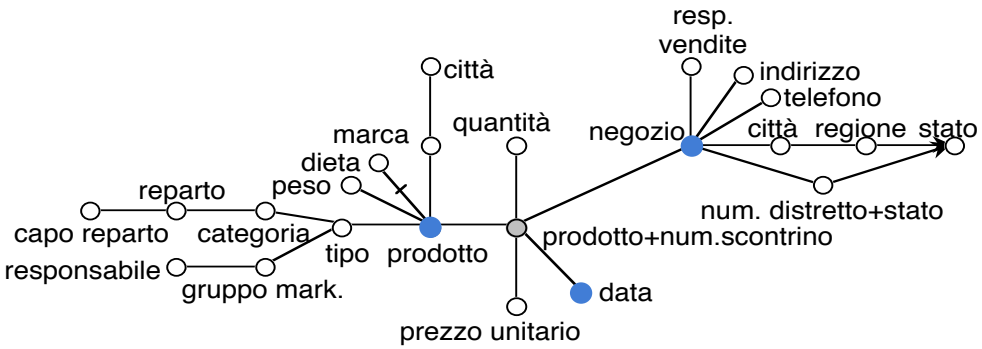
- Se in un fatto F che ha un solo identificatore (radice dell'albero degli attributi) si **pota o innesta un attributo dell'identificatore** allora si otterrà uno **schema temporale**
  - ✓ Se il vertice innestato ha più di un figlio, si può avere un aumento del numero di dimensioni nello schema di fatto

- Se un fatto F ha più di un identificatore
  1. Uno viene scelto come radice dell'albero degli attributi
  2. Gli altri si eliminano tramite innesto: infatti dati due identificatori I1 ed I2, abbiamo che I1 → I2 e I2 → I1
     - ✓ Quelli eliminati possono essere tenuti come descrittivi

- In uno schema transazionale un evento primario corrisponde ad una singola istanza del fatto
  - ✓ Infatti un evento primario è identificato dalle dimensioni, che contengono un identificatore del fatto che identifica una singola istanza del fatto

L'esempio delle vendite

- Considerando come dimensioni **prodotto**, **negozio** e **data** lo schema di fatto Vendita è **temporale**, in quanto la (unica) chiave {prodotto,scontrino} del fatto Vendita **non è inclusa** nelle dimensioni



Definizione delle misure

- In uno **schema transazionale** le misure corrispondono ad attributi numerici che siano *figli* della radice dell'albero

- In uno **schema temporale** le misure si definiscono applicando, ad attributi numerici dell'albero, funzioni di aggregazione che operano su tutte le istanze di F corrispondenti a ciascun evento primario (in genere si tratta di somma/media/massimo/minimo di espressioni oppure del conteggio del numero di istanze di F)
  - ✓ Può essere utile definire più misure che aggregano lo **stesso attributo tramite operatori diversi**!
  - ✓ Le funzioni di aggregazione devono essere definite **raggruppando rispetto alle dimensioni**!

- **Glossario delle misure**: ad ogni misura è associata un'espressione che descrive come essa possa essere calcolata a partire dal DBO

L'esempio delle vendite

- Glossario delle Misure

```
quantità venduta = SUM(VENDITA.quantità)
incasso = SUM(VENDITA.quantità*VENDITA.prezzoUnitario)
prezzo unitario = AVG(VENDITA.prezzoUnitario)
numero vendite = COUNT(*)
numeno clienti = COUNT(DISTINCT num_scontrino)
```

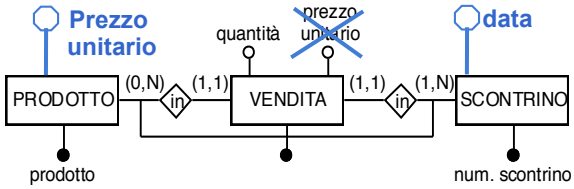
- Il Glossario delle Misure *sintetizza* la definizione delle misure, ad esempio
  1. **`quantità venduta`** è la somma della quantità espressa in VENDITA
  2. **`prezzo unitario`** è la media di quello espresso in VENDITA
  3. **`numero clienti`** è (stimato come) il conteggio *distinto* degli scontrini

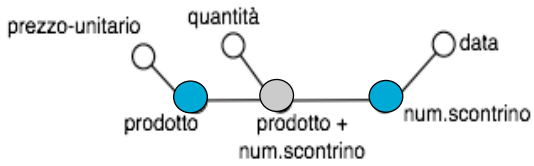
- La definizione e l'*aggregabilità* delle misure è uno degli aspetti più delicati e difficile della progettazione del DW e verrà trattato più avanti nel corso!
  - A questo punto sappiamo che il prezzo unitario è definito tramite una media (nel glossario è stato sintetizzato con AVG) però vedremo che in uno schema di fatto temporale non si può usare tale operatore …

Dipendenza parziale dalle dimensioni

- Se una misura è un attributo numerico che non è *figlio diretto* della radice dell'albero allora si ottiene una misura che *dipende parzialmente dalle dimensioni* :
  - ✓ Per definizione, una misura M dipende dalle dimensioni D: D → M
  - ✓ Dipendenza parziale: esiste D' ⊂ D tale che D' → M
  - ✓ Se M dipende parzialmente da D', allora il suo valore resta invariato nei pattern che contengono D' (esempio: nel *roll-up* da D a D' il valore di M resta invariato).

- La misura PrezzoUnitario non è *figlio diretto* della radice :



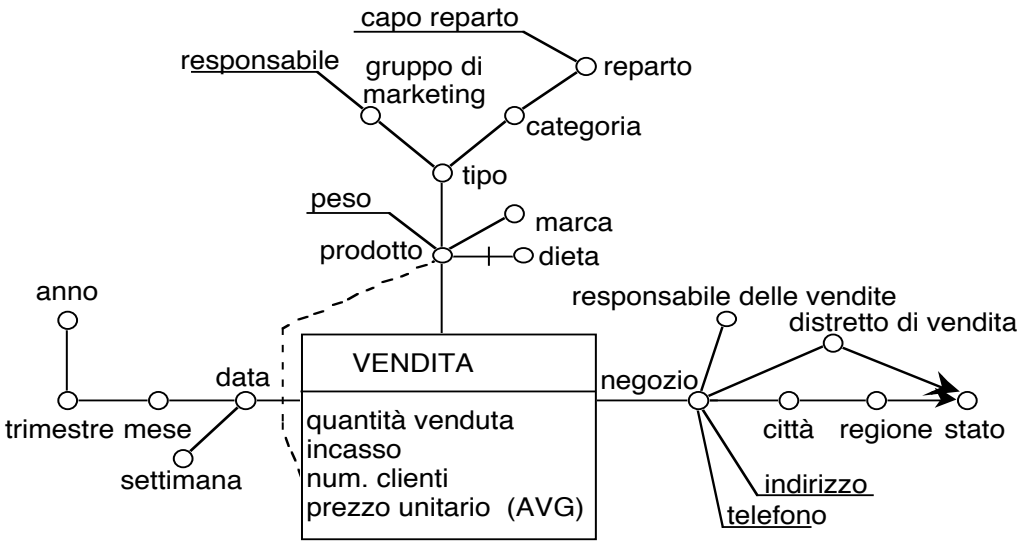


- Siccome Prodotto → PrezzoUnitario , per PrezzoUnitario si ha una dipendenza parziale dalle dimensioni:
  - PrezzoUnitario rimane invariato nei pattern che contengono Prodotto

Creazione dello schema di fatto

- L'albero degli attributi può ora essere tradotto in uno schema di fatto che include le dimensioni e 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 modificare un attributo considerandone opportuni intervalli
  - ✓ 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 uno-a-uno e privi di discendenti
  - ✓ Per quanto riguarda eventuali attributi alfanumerici figli della radice ma non prescelti né come dimensioni né come misure:
    - Schemi transazionali: si possono rappresentare come attributi descrittivi associati al fatto, di cui descriveranno ciascuna occorrenza
    - Schemi temporali: devono necessariamente essere potati

L'esempio delle vendite

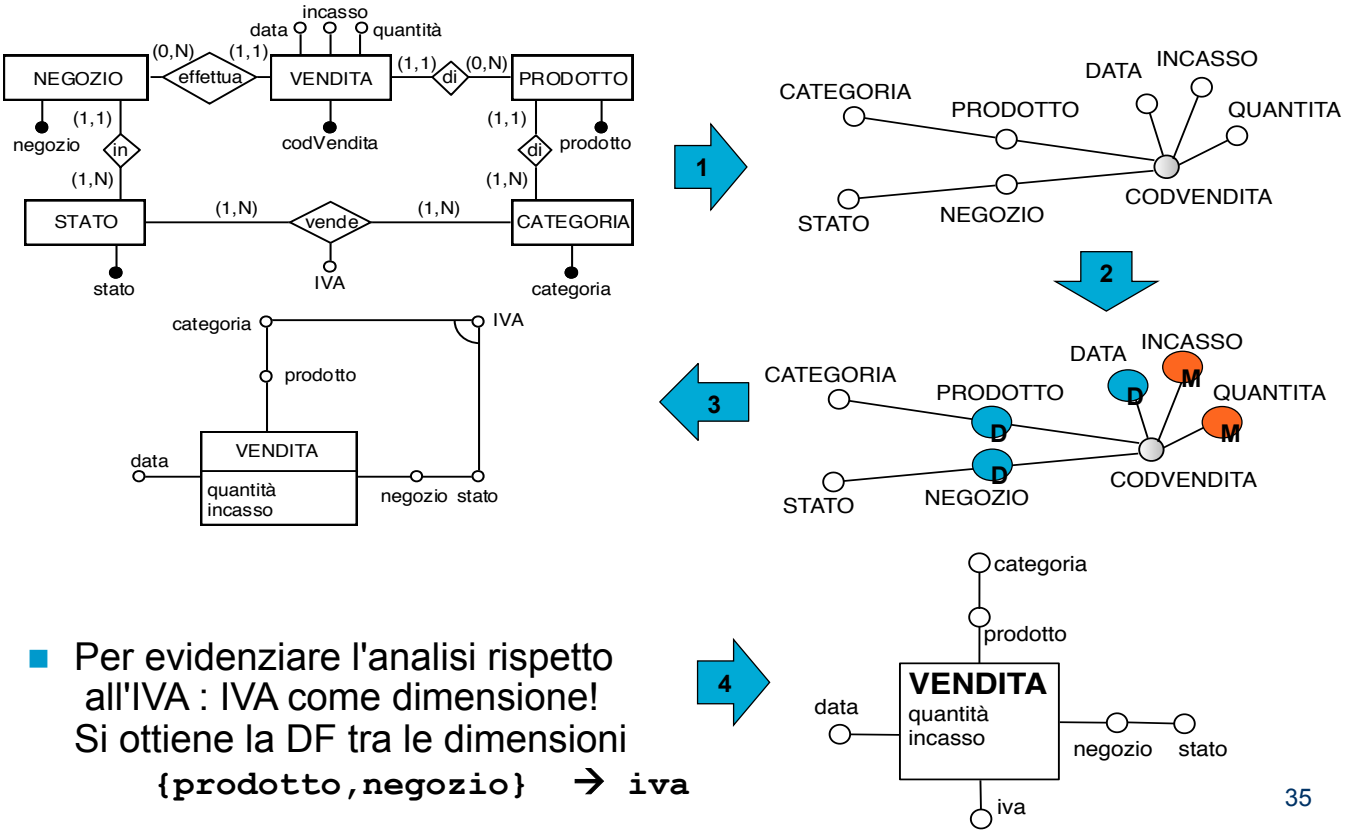


- **Argomenti da svolgere:** Definizione ed aggregabilità delle misure

Creazione dello schema di fatto

- Eventuali attributi cross-dimensionali e archi multipli possono essere evidenziati in questa fase
  - ✓ Identificare tali tipi di attributi sullo schema E/R è complesso, poiché richiede di navigare anche le associazioni a-molti, per cui si preferisce definirli a partire dai requisiti utente per rappresentarli solo successivamente sullo schema di fatto
    - Un attributo cross-dimensionale corrisponde in genere a un attributo posto su un'associazione molti-a-molti R; i suoi padri nello schema di fatto corrisponderanno allora agli identificatori delle entità coinvolte in R
    - Un arco multiplo corrisponde a un'associazione a-molti R da un'entità E a un'entità G; nello schema di fatto, esso potrà allora connettere l'identificatore di E o il fatto con un attributo di R o di G

Attributo cross-dimensionale: Esempio

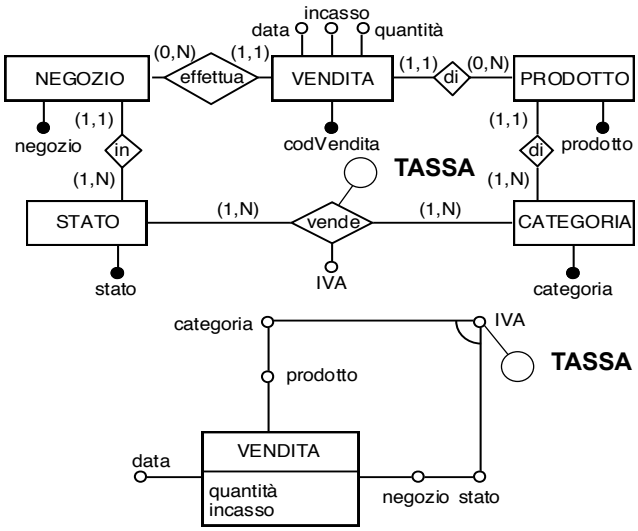


- Per evidenziare l'analisi rispetto all'IVA : IVA come dimensione!
  Si ottiene la DF tra le dimensioni
  `{prodotto,negozio}` → `iva`

Attributo cross-dimensionale: Esempio

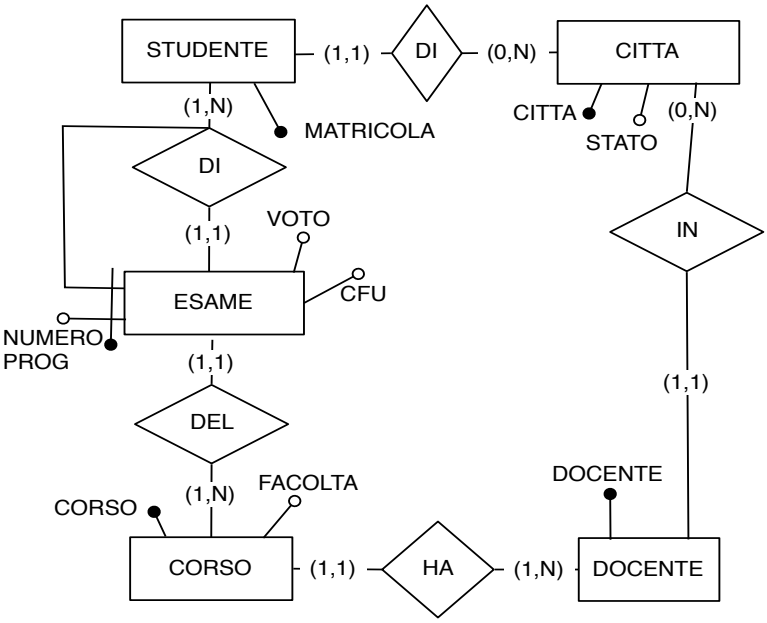
- Un attributo cross-dimensionale può avere dei discendenti.

- Si aggiunge all'associazione VENDE l'attributo TASSA
- Dalla documentazione: TASSA dipende da IVA. Questa dipendenza non è espressa nello schema in quanto lo complicherebbe notevolmente
- Nell'albero degli attributi e poi nello schema di fatto tale dipendenza funzionale si esprime in modo molto efficace



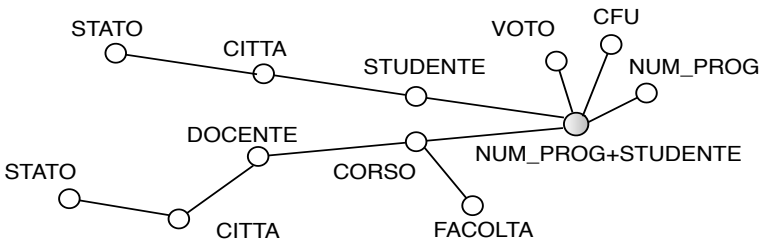
Progettazione Concettuale: Esempio

- Schema E/R degli esami:



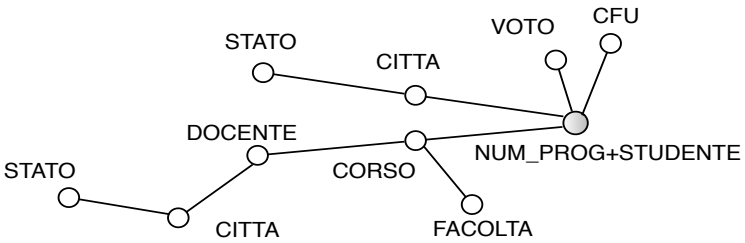
Progettazione Concettuale: Esempio

- Albero degli attributi iniziale



- Potature:

1. STUDENTE: non interessa il *singolo studente*
2. NUM_PROG: non interessa il *singolo esame*



- Nell'albero modificato, come nome della radice si tiene ancora l'identificatore del fatto. Un'alternativa per denotare la radice è quella di usare (sia nell'albero iniziale che in quello modificato, il nome del fatto

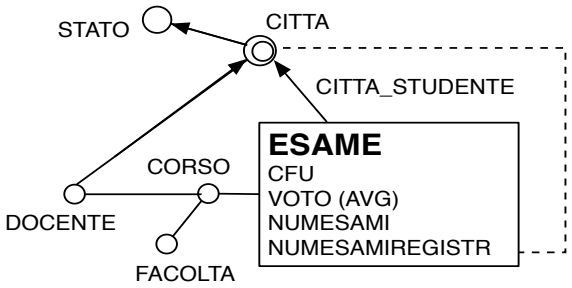
Progettazione Concettuale: Esempio

- ➔ Dimensioni : {CORSO,CITTA_STUDENTE}
- Condivisioni\Convergenze:
  in assenza di specifiche si considera la condivisione
- ➔ Misure:
  1. CFU, inteso come CFU complessivi
  2. VOTO, inteso come VOTO medio
  3. NUMESAMI, inteso come conteggio *di tutti gli esami*
  4. NUMESAMIREGISTR: inteso come *conteggio distinto* in esame della coppia <STUDENTE,CORSO>

➢ Uno studente può sostenere più esami per lo stesso corso:
con NUMESAMI viene contato più volte,
con NUMESAMIREGISTR viene contato una sola volta

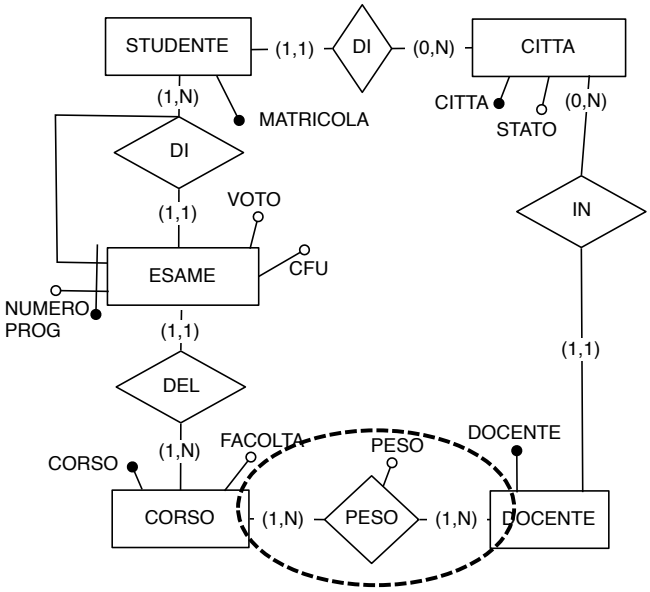
Progettazione Concettuale: Esempio

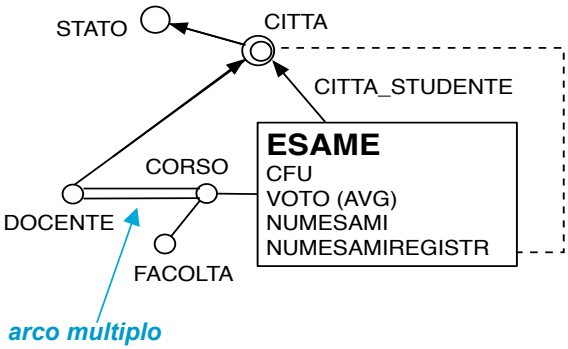
- Schema di Fatto



Arco Multiplo: Esempio

- Modifica dello schema E/R precedente:
  un corso è **associato a più docenti**, con un diverso *peso*





- Gli archi multipli verranno trattati a parte (a fine corso circa …)

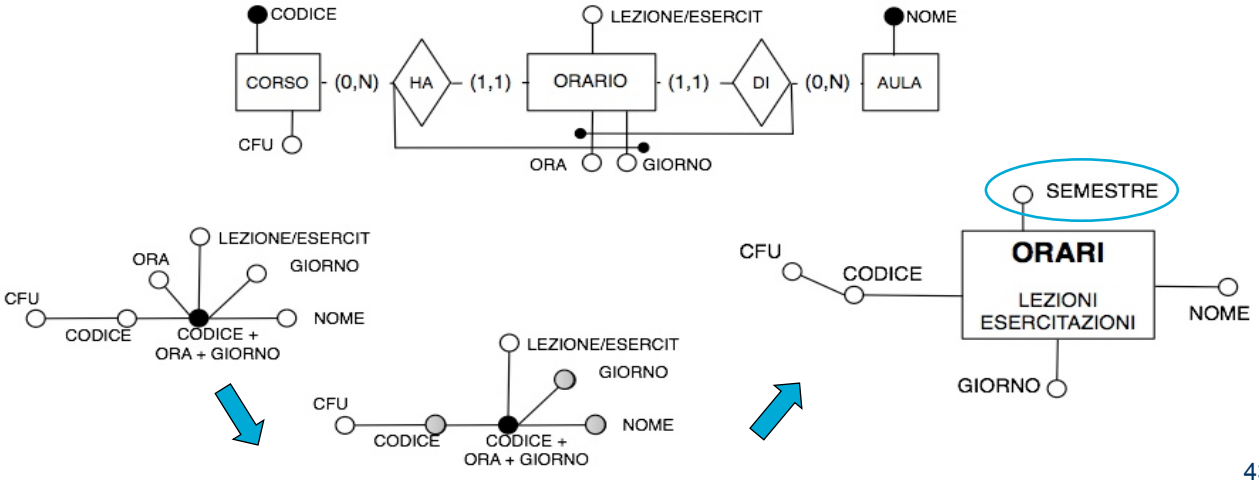
Definizione della dimensione temporale

- Il tempo dovrebbe sempre essere una dimensione.
- Il tempo è presente esplicitamente come un attributo nelle sorgenti *storiche*
  - ✓ Lo schema dell'esempio delle vendite si riferisce ad una sorgente storica: c'è un attributo tempo (la data) nell'entità scontrino
- Se il tempo appare nell'albero degli attributi come figlio di un vertice diverso dalla radice, per riportarlo sulla radice si può
  1) **effettuare un innesto** : nell'esempio delle vendite data diventa figlio della radice con un innesto su num-scontrino
  2) **eliminare una dipendenza funzionale:** nell'esempio delle vendite se eliminiamo la FD: num-scontrino → data

e considerando num-scontrino e data tra le dimensioni del fatto si ottiene una **dipendenza funzionale tra le dimensioni!**

Sorgenti snapshot

- Nelle sorgenti *snapshot* il tempo non viene rappresentato esplicitamente; in questo caso il tempo viene tipicamente aggiunto "manualmente" allo schema di fatto
- **Esempio**: Data mart dell'orario semestrale delle lezioni
  - ✓ Il DBO è riferito al semestre attuale e non c'è lo storico



Esempio DM ORARI: Misure ed alimentazione

- Glossario delle misure
  - ✓ LEZIONI: numero di ore di lezioni, calcolato contando le tuple con LEZIONI/ESERCIT = "lezione"
  - ✓ ESERCITAZIONI: numero di ore di esercitazioni, calcolato contando le tuple con LEZIONI/ESERCIT = "esercitazione"

- Alimentazione semestrale del DataMart
  - ✓ All'inizio di un nuovo semestre
    1) il DBO viene riempito con il nuovo orario semestrale
    2) il DM Orari viene alimentato coi i dati del DBO e per l'attributo temporale SEMESTRE si usa il valore del nuovo semestre