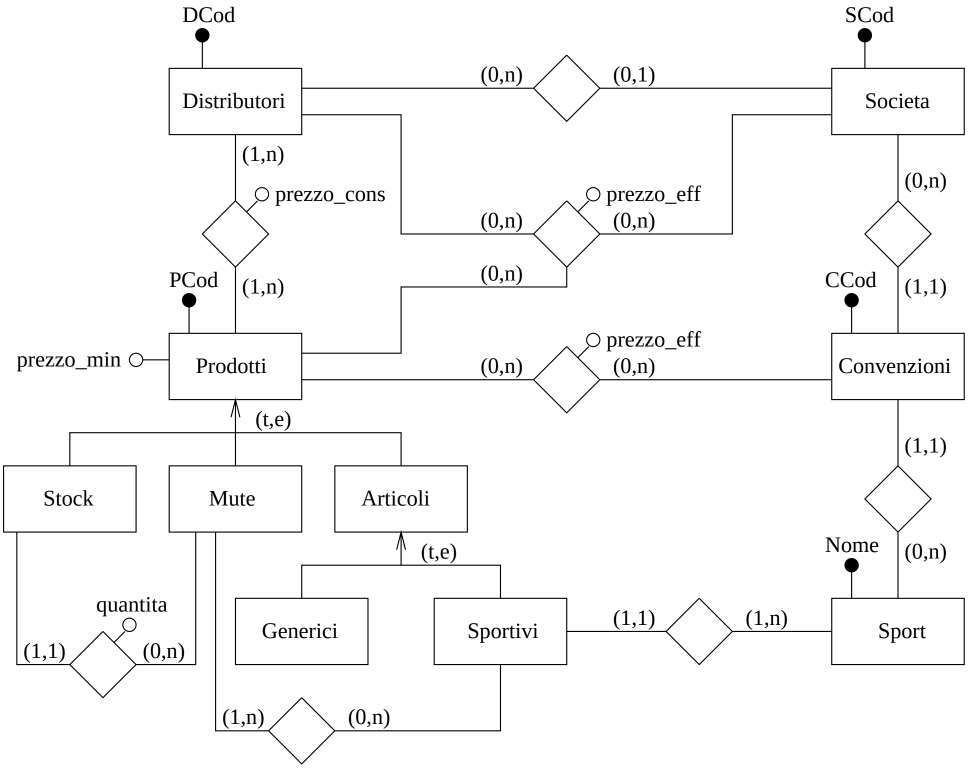
Corso di Laurea in Informatica

Basi di Dati e Sistemi Informativi – 20 Febbraio 1998

Risoluzione

Esercizio 1



Lo schema evidenzia il ruolo importante che ha la gerarchia dei `Prodotti` nel permettere di rappresentare al meglio le specifiche. Alcuni vincoli non banali non sono tuttavia rappresentati, in particolare:

- se una `Società` ha delle `Convenzioni`, allora deve essere associata ad un distributore;
- le `Mute` comprendono articoli `Sportivi` di uno stesso `Sport`;
- le vendite che avvengono nell'ambito di `Convenzioni` riguardano solo i `Prodotti` dello `Sport` relativo.

Si noti che i vincoli di cardinalità, `(1,1)`, per l'entità `Stock` nell'associazione tra `Stock` e `Mute`, non contraddicono il fatto che gli `Stock` consistono di più `Mute`. In realtà uno `Stock` consiste sì di più `Mute`, ma queste sono tutte uguali. Infatti un elemento di `Prodotti`, ovvero un elemento del catalogo, corrisponde a un "tipo" di prodotto, e non a uno specifico "pezzo" di un certo tipo, e ciò spiega l'apparente anomalia.

Esercizio 2

Una copertura minima di $\mathcal{F}$ è data dal seguente insieme di dipendenze:

$$\mathcal{G} = \{AB \rightarrow C, C \rightarrow D, C \rightarrow B\}$$

che si ricava osservando che B è estraneo in $BC \rightarrow D$ e che $AB \rightarrow D$ è ridondante.
Una decomposizione di R in 3NF che preservi i dati e le dipendenze è data dagli schemi:

$$R_1(\underline{AB}C) \qquad R_2(\underline{C}BD)$$

Infine, un esempio di decomposizione in BCNF che preserva i dati ma non le dipendenze è dato dagli schemi:

$$R_1(\underline{AC}) \qquad R_2(\underline{C}BD)$$

che evidentemente non preservano $AB \rightarrow C$.

Esercizio 3

Le istruzioni SQL che risolvono l'esercizio sono, rispettivamente:

```
Select C.Attore, C.Film
From   Cast C, Film F
Where  C.Film = F.Titolo
and    C.Ruolo = 'protagonista'
and    F.Incasso = ( Select max(F1.Incasso)
                     From   Cast C1, Film F1
                     Where  C1.Film = F1.Titolo
                     and    C1.Attore = C.Attore
                     and    C1.Ruolo = 'protagonista' ) ;

Select Distinct C.Attore
From   Cast C
Where  C.Attore not in ( Select C1.Attore
                         From   Cast C1
                         Where  C1.Ruolo <> 'protagonista' ) ;

Select C.Attore, C.Film
From   Cast C, Film F
Where  C.Film = F.Titolo
and    C.Attore = F.Regista ;
```

Esercizio 4

Il problema è analogo, ad un certo livello di astrazione, a quello che porta al modello di Cardenas. Nel caso in esame si tratta di stimare la cardinalità di una proiezione (`Distinct R.C`) preceduta da una selezione (`R.B = 10`).

Indicando con ET_σ il numero di tuple che soddisfano il predicato su `R.B`, si ha innanzitutto:

$$ET_\sigma = \frac{NT}{NK_B}$$

Per stimare quanti sono i valori distinti di `C` presenti in tali tuple, dalle ipotesi fatte (`B` e `C` sono indipendenti e i valori di `C` sono distribuiti uniformemente sulle NT tuple della relazione), in analogia con il modello di Cardenas si può affermare che $1/NK_C$ è la probabilità che una tupla di `R` contenga un dato di valore di `C` e $1 - 1/NK_C$ che non lo contenga.

Selezionando ET_σ tuple da `R`, ne segue che un valore di `C` ha probabilità pari a $1-(1-1/NK_C)^{ET_\sigma}$ di comparire in almeno una di tali tuple. Pertanto la cardinalità del risultato, ET, dell'istruzione SQL si può stimare pari a:

$$ET = NK_C \times \left(1 - \left(1 - \frac{1}{NK_C}\right)^{ET_\sigma}\right) = \Phi(ET_\sigma, NK_C) = \Phi(NT/NK_B, NK_C)$$