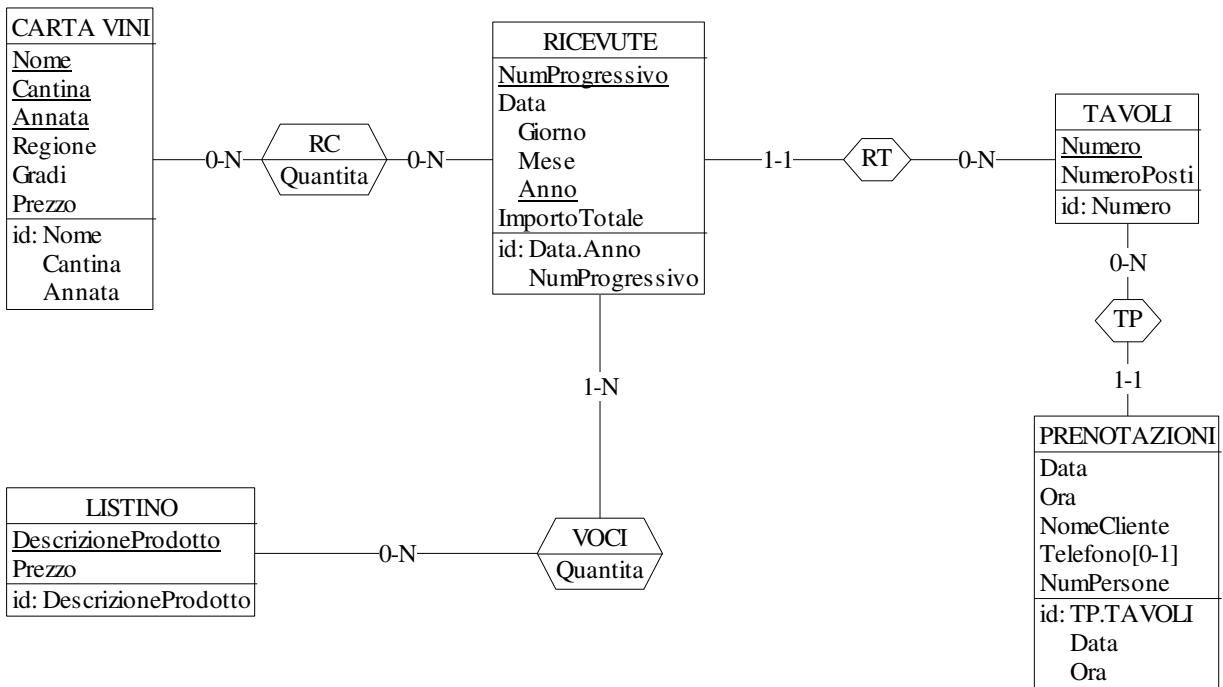


Sistemi Informativi L-B
10 febbraio 2011
Risoluzione

Tempo a disposizione: 2 ore

1) Progettazione concettuale (5 punti)



Commenti:

- L'esercizio non presenta difficoltà di rilievo
- Il vincolo relativo al numero di persone presenti in una prenotazione, che deve essere non superiore al numero di posti del tavolo riservato, non è modellabile
- Si noti che, poiché le ricevute non riportano nessun dato che identifichi i clienti, non esiste nessuna relazione diretta tra **RICEVUTE** e **PRENOTAZIONI**

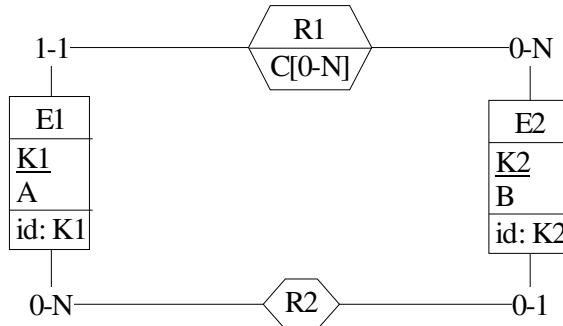
Sistemi Informativi L-B

10 febbraio 2011

Risoluzione

2) Progettazione logica e normalizzazione (3 punti)

Dato lo schema concettuale in figura,



e considerando che:

- a) tutti gli attributi sono di tipo INT;
- b) le associazioni R1 e R2 non vengono tradotte separatamente;
- c) se una coppia di istanze (k1,k2) di E1 ed E2 è associata tramite R1, non può essere associata anche tramite R2, e viceversa;

si progettino gli opportuni schemi relazionali e si definiscano tali schemi facendo uso dell'SQL di DB2; per gli eventuali vincoli non esprimibili a livello di schema si predispongano opportune **query di verifica da eseguire prima di effettuare inserimenti di tuple**, allo scopo di evitare che tali inserimenti violino i vincoli stessi.

```
CREATE TABLE E1(  
K1 INT NOT NULL PRIMARY KEY,  
A INT NOT NULL,  
K2 INT NOT NULL); -- il vincolo di foreign key va dichiarato dopo aver definito E2
```

```
CREATE TABLE E2(  
K2 INT NOT NULL PRIMARY KEY,  
B INT NOT NULL,  
K1 INT REFERENCES E1 );
```

```
ALTER TABLE E1  
ADD CONSTRAINT FKR1 FOREIGN KEY (K2) REFERENCES E2 ;
```

```
CREATE TABLE R1C(  
K1 INT NOT NULL REFERENCES E1,  
C INT NOT NULL,  
PRIMARY KEY (K1,C) );
```

Il vincolo del punto c) NON e' violabile in fase di inserimento, ma solo in fase di aggiornamento. In tal caso, quando si modifica una tupla di E1 con i nuovi valori (k1,a,k2), è necessario eseguire la query:

```
SELECT * FROM E2 -- ok se non restituisce nessuna tupla  
WHERE E2.K1 = k1  
AND E2.K2 = k2
```

Analogamente, per una tupla di E2 con nuovi valori (k2,b,k1) va eseguita la query:

```
SELECT * FROM E1 -- ok se non restituisce nessuna tupla  
WHERE E1.K1 = k1  
AND E1.K2 = k2;
```

3) DB fisico (2 punti)

- a) **SELECT * FROM R WHERE A = 5**
Se il valore 5 è esistente, il costo medio sarà pari a $NP/2$
- b) **SELECT * FROM R WHERE B = 6**
Poiché R è ordinata su B (che è a valori ripetuti), il costo sarà pari a $\lceil \log_2 NP \rceil + \lceil NP/NK(B) \rceil - 1$, in cui il primo termine è il costo per localizzare una tupla con B=6, il secondo è il numero medio di pagine che contengono uno stesso valore di B, e il -1 serve a non contare due volte la prima pagina cui si accede
- c) **SELECT * FROM R WHERE C = 7**
In questo caso il costo si può stimare pari a NP , in quanto le tuple con C = 7 sono sparse su tutto il file