

Sistemi Informativi T
20 giugno 2013
Risoluzione

Tempo a disposizione: 2:30 ore

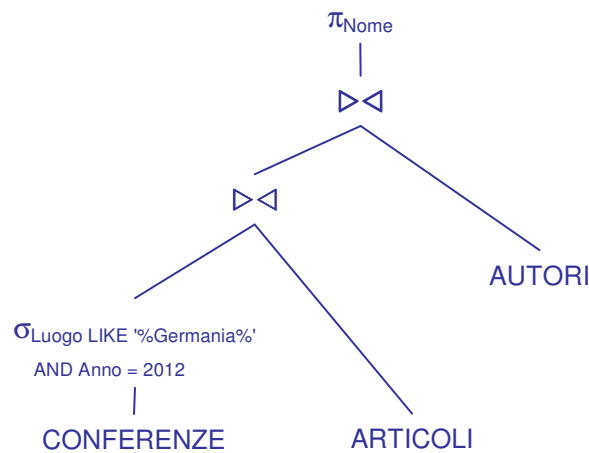
1) Algebra relazionale (3 punti totali):

Date le seguenti relazioni:

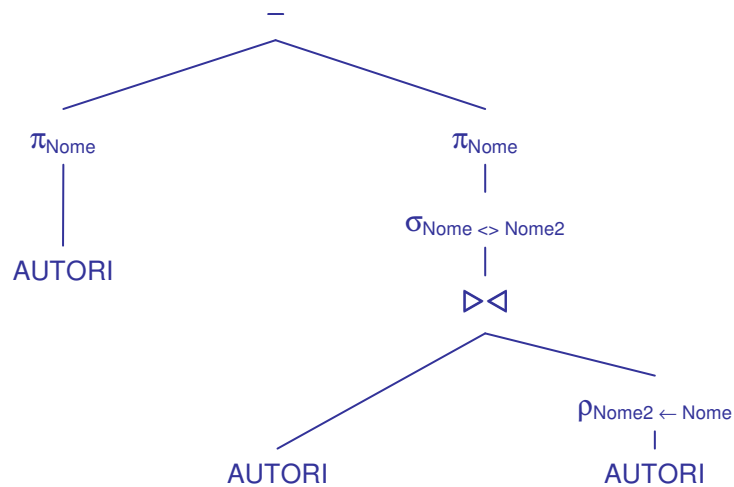
```
CONFERENZE (NomeConf, Anno, Luogo) ;  
ARTICOLI (ArtID, Titolo, NomeConf, Anno) ,  
NomeConf, Anno REFERENCES CONFERENZE ;  
AUTORI (ArtID, Nome) ,  
ArtID REFERENCES ARTICOLI ;
```

si scrivano in algebra relazionale le seguenti interrogazioni:

- 1.1) [1 p.]** I nomi degli autori che hanno almeno un articolo in una conferenza del 2012 tenutasi in Germania



- 1.2) [2 p.]** Gli autori che hanno pubblicato articoli sempre da soli



L'operando destro della differenza contiene gli autori che hanno almeno un lavoro pubblicato con un'altra persona.

Sistemi Informativi T
20 giugno 2013
Risoluzione

SQL (5 punti totali)

Con riferimento al DB dell'esercizio 1, si scrivano in SQL le seguenti interrogazioni:

2.1) [2 p.] Gli autori che hanno pubblicato articoli sempre da soli

```
SELECT DISTINCT A.NOME
FROM  AUTORI A
WHERE NOT EXISTS (   SELECT *
                     FROM AUTORI A1
                     WHERE A1.ArtID = A.ArtID
                     AND   A1.Nome <> A.Nome )
```

2.2) [3 p.] Per ogni anno, l'autore che ha pubblicato nel maggior numero di conferenze distinte

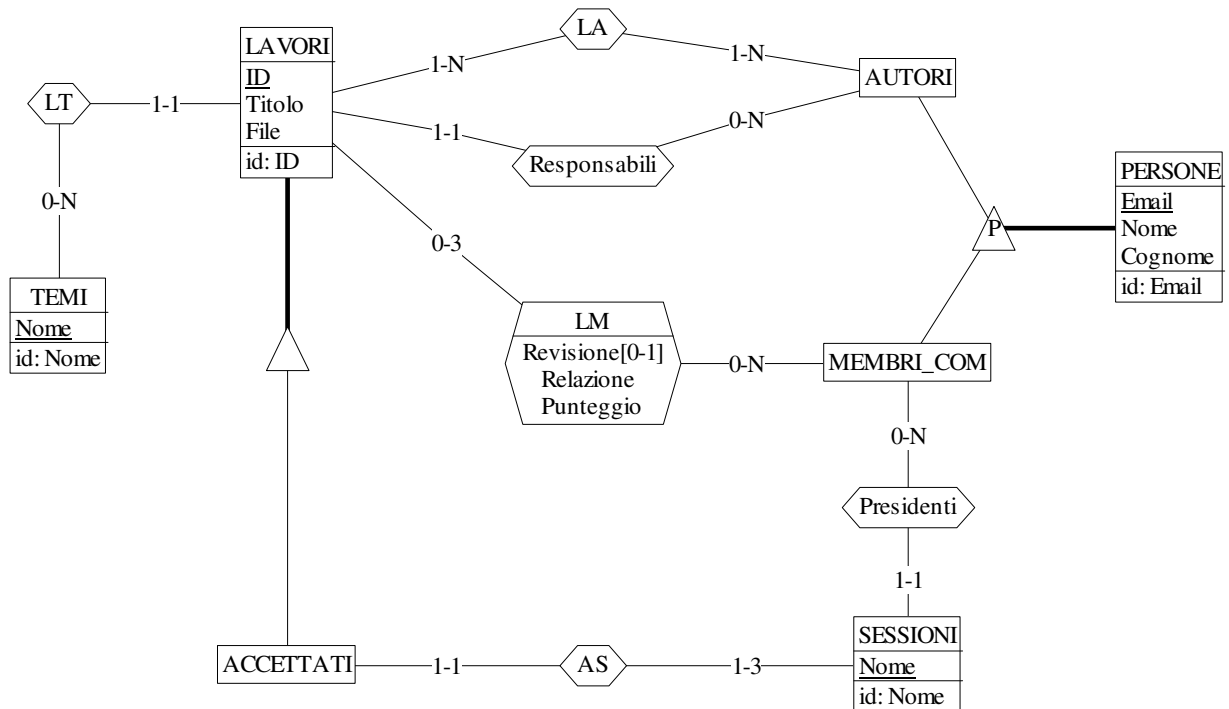
```
WITH AUTORIANNI(AUTORE,ANNO,NUMCONF) AS (
SELECT AU.Nome, AR.Anno, COUNT(DISTINCT NomeConf)
FROM AUTORI AU, ARTICOLI AR
WHERE AR.ArtID = AU.ArtID
GROUP BY AU.Nome, AR.Anno)

SELECT A.ANNO, A.AUTORE
FROM AUTORIANNI A
WHERE A.NUMCONF = (   SELECT MAX(A1.NUMCONF)
                     FROM AUTORIANNI A1
                     WHERE A1.ANNO = A.ANNO)

-- Nella common table expression si sfrutta la forma COUNT(DISTINCT ...)
-- per contare una volta sola conferenze in cui un dato autore ha
-- pubblicato piu' di 1 lavoro nello stesso anno
```

3) Progettazione concettuale (6 punti)

Il convegno su "Cyberspazio: Come Cambia il Pianeta" (CCCP) accetta lavori scientifici sui temi del convegno. Ogni lavoro, scritto da uno o più autori (di cui uno funge da responsabile), all'atto della sottomissione viene classificato usando uno (e uno solo) dei temi del convegno. Ogni lavoro viene assegnato per essere revisionato a 3 membri del comitato del CCCP che, al termine del loro lavoro, preparano una relazione scritta e assegnano al lavoro un punteggio da 0 a 6. I lavori accettati vengono quindi organizzati in sessioni per la presentazione orale al convegno (non più di 3 lavori per sessione). Ogni sessione ha un nome, che la identifica, e un membro del comitato che la presiede. I membri del comitato non possono essere autori di lavori sottomessi; un autore può, in generale, esserlo anche di più lavori.



Commenti:

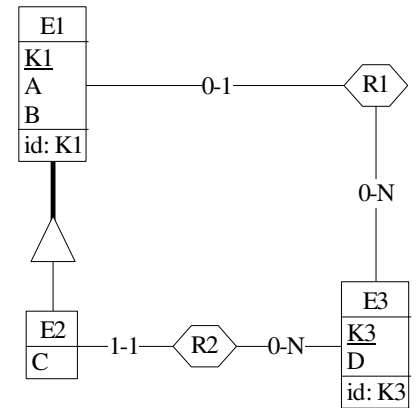
- Nella soluzione proposta i vincoli di cardinalità tengono conto della dinamica del sistema descritto, e quindi sono meno restrittivi di quanto ci si potrebbe aspettare. Ad esempio, anche se *a regime* ogni lavoro ha 3 revisori, all'atto della sottomissione, e quindi della registrazione del lavoro nel DB, ciò non è vero, da cui il vincolo (0,3), anziché (3,3). Lo stesso vale per le revisioni, che all'inizio sono vuote (si sa solo chi revisionerà un lavoro, ma relazione e punteggio sono disponibili solo successivamente).
- Lo schema ER non esprime il vincolo che il responsabile di un lavoro deve essere uno degli autori del lavoro stesso.

4) Progettazione logica (6 punti totali)

Dato lo schema concettuale in figura e considerando che:

- tutti gli attributi sono di tipo INT;
- nessuna associazione viene tradotta separatamente;
- ogni istanza di E2 è associata, tramite R1 e R2 a istanze diverse di E3;
- il valore di A è sempre almeno il doppio del valore di B;

4.1) [3 p.] Si progettino gli opportuni schemi relazionali e si definiscano tali schemi in DB2 (sul database SIT_STUD) mediante un file di script denominato **SCHEMI.txt**



```
CREATE TABLE E3 (
  K3 INT NOT NULL PRIMARY KEY,
  D INT NOT NULL );
```

```
CREATE TABLE E1 (
  K1 INT NOT NULL PRIMARY KEY,
  A INT NOT NULL,
  B INT NOT NULL,
  K3R1 INT REFERENCES E3,
  TIPO2 SMALLINT NOT NULL CHECK (TIPO2 IN (0,1)),      -- 1: istanza anche di E2
  C INT,
  K3R2 INT REFERENCES E3,
  CONSTRAINT E2 CHECK
    ( (TIPO2 = 1 AND K3R2 IS NOT NULL AND C IS NOT NULL) OR
      (TIPO2 = 0 AND K3R2 IS NULL AND C IS NULL) ),
  CONSTRAINT PUNTO_C CHECK (K3R1 <> K3R2),    -- se non e' un E2 allora K3R2 e' NULL
  CONSTRAINT PUNTO_D CHECK (A >= 2*B) );
```

-- Si noti che una traduzione alternativa, corretta secondo le specifiche, consiste nel creare 2 schemi distinti
-- per E1 ed E2

4.2) [3 p.] Per i vincoli non esprimibili a livello di schema si predispongano opportuni **trigger che evitino inserimenti di tuple non corrette**, definiti in un file **TRIGGER.txt** e usando se necessario il simbolo '@' per terminare gli statement SQL (altrimenti ';')

-- Con la soluzione adottata (E1 ed E2 tradotte assieme) non e' necessario alcun trigger