

Sistemi Informativi T
6 febbraio 2014
Risoluzione

Tempo a disposizione: 2:30 ore

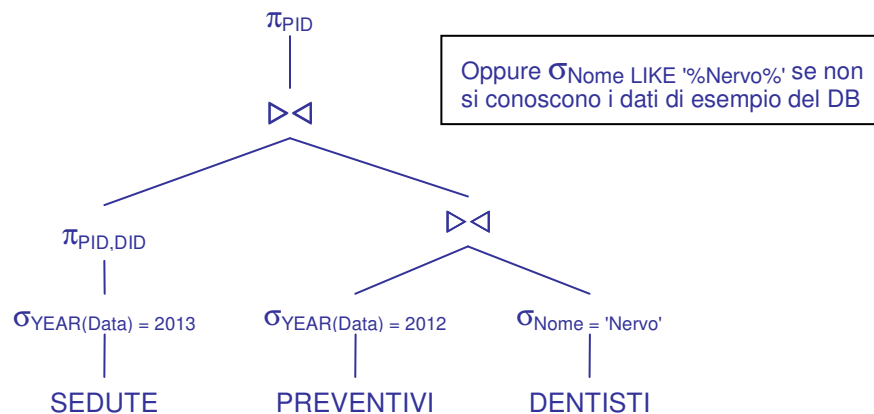
1) Algebra relazionale (3 punti totali):

Date le seguenti relazioni:

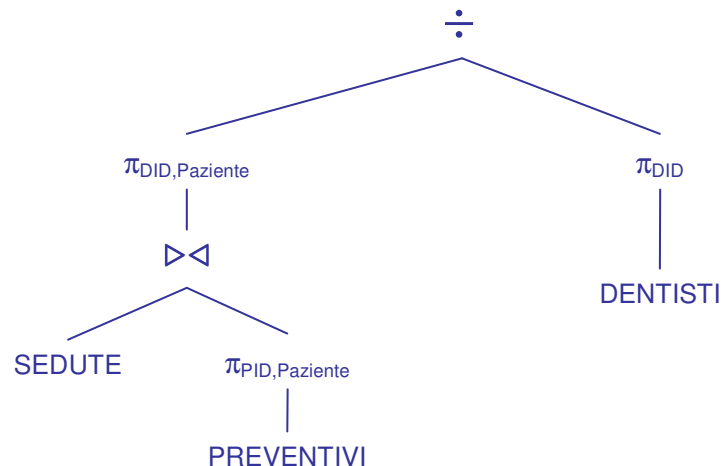
```
DENTISTI (DID, Nome, AnniServizio);  
PREVENTIVI (PID, Data, Paziente, Tipo, Dettagli, Importo, Pagato);  
SEDUTE (PID, Data, DID),  
    PID REFERENCES Preventivi,  
    DID REFERENCES DENTISTI;  
  
--  
-- Importo e Pagato sono di tipo DEC(8,2)  
-- Pagato ha valore di default 0 ed e' sempre <= Importo  
-- PREVENTIVI.Data e' la data in cui viene eseguito il preventivo  
-- di spesa  
-- Tipo riporta il tipo di cura relativa a un dato preventivo (ad  
-- es. 'estrazione')
```

si scrivano in algebra relazionale le seguenti interrogazioni:

1.1) [1 p.] Gli identificativi dei preventivi del 2012 in cui almeno una seduta è stata fatta nel 2013 dal dentista Nervo



1.2) [2 p.] I pazienti che, considerando tutti i preventivi, hanno fatto almeno una seduta con ogni dentista



Sistemi Informativi T
6 febbraio 2014
Risoluzione

2) SQL (5 punti totali)

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

2.1) [2 p.] Per ogni tipo di cura, il preventivo con il numero massimo di sedute nel 2013

```
SELECT  P.Tipo, P.PID, COUNT(*) AS NumSedute
FROM    PREVENTIVI P, SEDUTE S
WHERE   P.PID = S.PID
AND     YEAR(S.Data) = 2013
GROUP BY P.Tipo, P.PID
HAVING  COUNT(*) >= ALL
        ( SELECT COUNT(*)
          FROM    PREVENTIVI P1, SEDUTE S1
          WHERE   P1.PID = S1.PID
          AND     YEAR(S1.Data) = 2013
          AND     P1.Tipo = P.Tipo
          GROUP BY P1.PID)

-- Era ovviamente possibile fare uso di una common table expression
```

2.2) [3 p.] Per ogni fascia di importo (0-499 Euro: fascia 1; 500-999 Euro: fascia 2; ecc.), il numero medio di sedute (con 2 cifre decimali) dei preventivi nella fascia

```
WITH PFN (PID, Fascia, NumSedute) AS (
    SELECT P.PID, CAST(P.Importo AS INT)/500 + 1, COUNT(*)
    FROM    PREVENTIVI P, SEDUTE S
    WHERE   P.PID = S.PID
    GROUP BY P.PID, CAST(P.Importo AS INT)/500 )

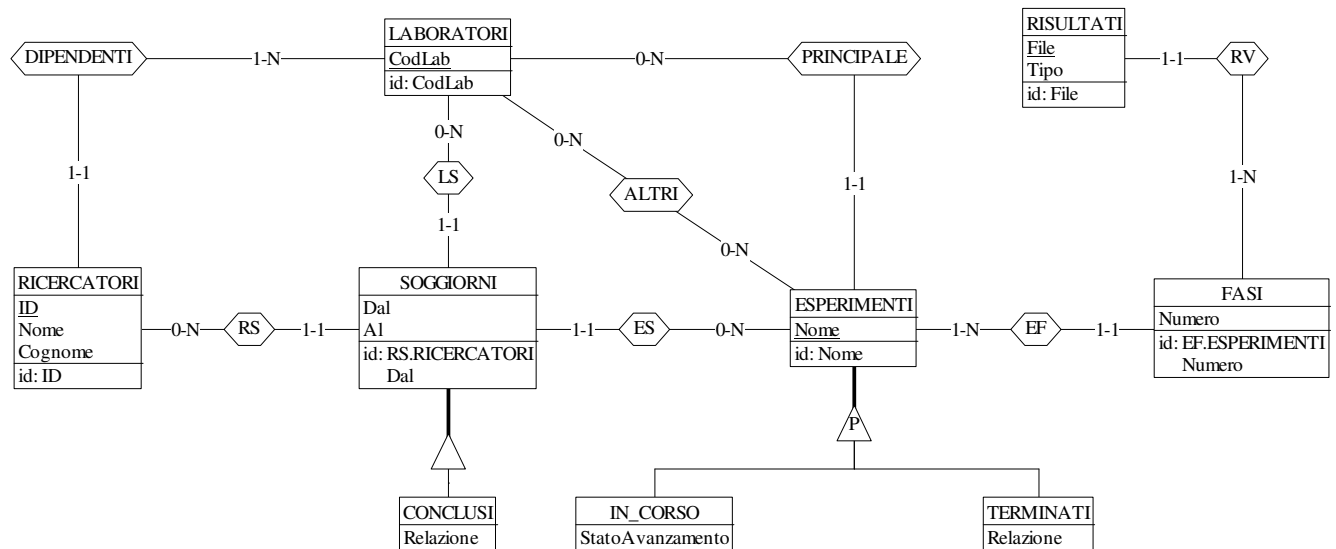
SELECT  Fascia, CAST(AVG(NumSedute/1.0) AS DEC(4,2)) AS AvgSedute
FROM    PFN
GROUP BY Fascia

-- Si noti il CAST nella c.t.e., che si rende necessario perche'
-- Importo e' di tipo DEC(8,2)
```

3) Progettazione concettuale (6 punti)

Presso i laboratori di ricerca del DARN si svolgono esperimenti sulla fisica delle particelle. Ogni esperimento, dal nome univoco, si può svolgere presso più laboratori, ma ne ha sempre uno come principale. Un esperimento si compone di diverse fasi, numerate in sequenza, e ogni fase produce file con risultati di varia natura (grafici, osservazioni particolari, ecc.). Al termine di un esperimento viene redatta una relazione conclusiva che riassume i risultati ottenuti (per gli esperimenti tuttora in corso vi è invece un documento che riassume lo stato di avanzamento dell'esperimento).

I ricercatori del DARN, ciascuno dipendente da uno specifico laboratorio, possono, per le esigenze di un dato esperimento, soggiornare per periodi di tempo presso altri laboratori (per lo stesso esperimento un ricercatore può anche recarsi più volte presso uno stesso laboratorio). Ogni soggiorno concluso deve essere accompagnato da una relazione del ricercatore in oggetto che riassume l'attività effettivamente svolta..



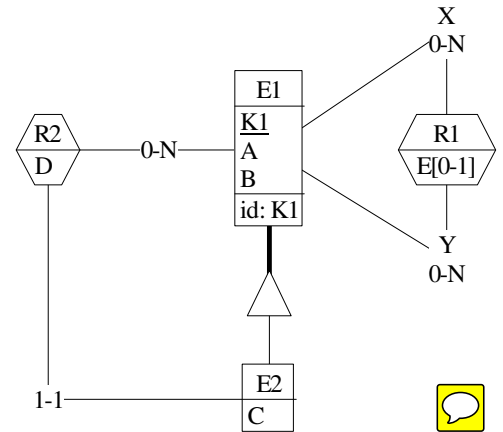
Commenti:

- La reificazione di SOGGIORNI serve per poter specificare cosa identifica un soggiorno: l'identificatore vieta solamente che un ricercatore possa essere contemporaneamente in più laboratori (ovviamente il vincolo sulla disgiunzione degli intervalli temporali non è specificabile in E/R), e quindi soddisfa la specifica "per lo stesso esperimento un ricercatore può anche recarsi più volte presso uno stesso laboratorio".
- Un laboratorio deve necessariamente partecipare ad almeno un esperimento, quindi partecipare ad almeno una delle associazioni ALTRI e PRINCIPALE.
- La soluzione assume che il termine del soggiorno (attributo "Al") sia noto anche per i soggiorni non conclusi. In alternativa si sarebbe potuto inserire tale attributo in CONCLUSI.
- Non è esprimibile il vincolo che, per un dato esperimento, un ricercatore può recarsi solo presso laboratori che partecipano a quell'esperimento.

Progettazione logica (6 punti totali)

Dato lo schema concettuale in figura e considerando che:

- tutti gli attributi sono di tipo INT;
- l'associazione R2 non viene tradotta separatamente;
- le entità E1 ed E2 vengono tradotte insieme;
- un'istanza di E1 che partecipa a R1 con il ruolo X non può referenziare, tramite il ruolo Y, un'istanza di E2 con $C > 10$;



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 E1 (
  K1 INT NOT NULL PRIMARY KEY,
  A INT NOT NULL,
  B INT NOT NULL,
  TIPO SMALLINT NOT NULL CHECK (TIPO IN (1,2)),      -- 2: istanza anche di E2
  C INT,
  D INT,
  K1R2 INT REFERENCES E1,
  CONSTRAINT E2 CHECK (
    (TIPO = 1 AND C IS NULL AND D IS NULL AND K1R2 IS NULL) OR
    (TIPO = 2 AND C IS NOT NULL AND D IS NOT NULL AND K1R2 IS NOT NULL) )
);
```

```
CREATE TABLE R1 (
  K1X INT NOT NULL REFERENCES E1,
  K1Y INT NOT NULL REFERENCES E1,
  E INT,
  PRIMARY KEY (K1X,K1Y)
);
```

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 ';')

```
-- Trigger che garantire il rispetto del vincolo di cui al punto d)
CREATE TRIGGER PUNTO_D
BEFORE INSERT ON R1
REFERENCING NEW AS N
FOR EACH ROW
WHEN (EXISTS ( SELECT * FROM E1
                WHERE E1.K1 = N.K1Y
                AND E1.TIPO = 2      -- ridondante, infatti se TIPO=1 allora C è NULL
                AND E1.C > 10 ))
SIGNAL SQLSTATE '70001' ('La tupla inserita referencia una tupla di E2 con valore C > 10!');
```