

## Esercizio 1

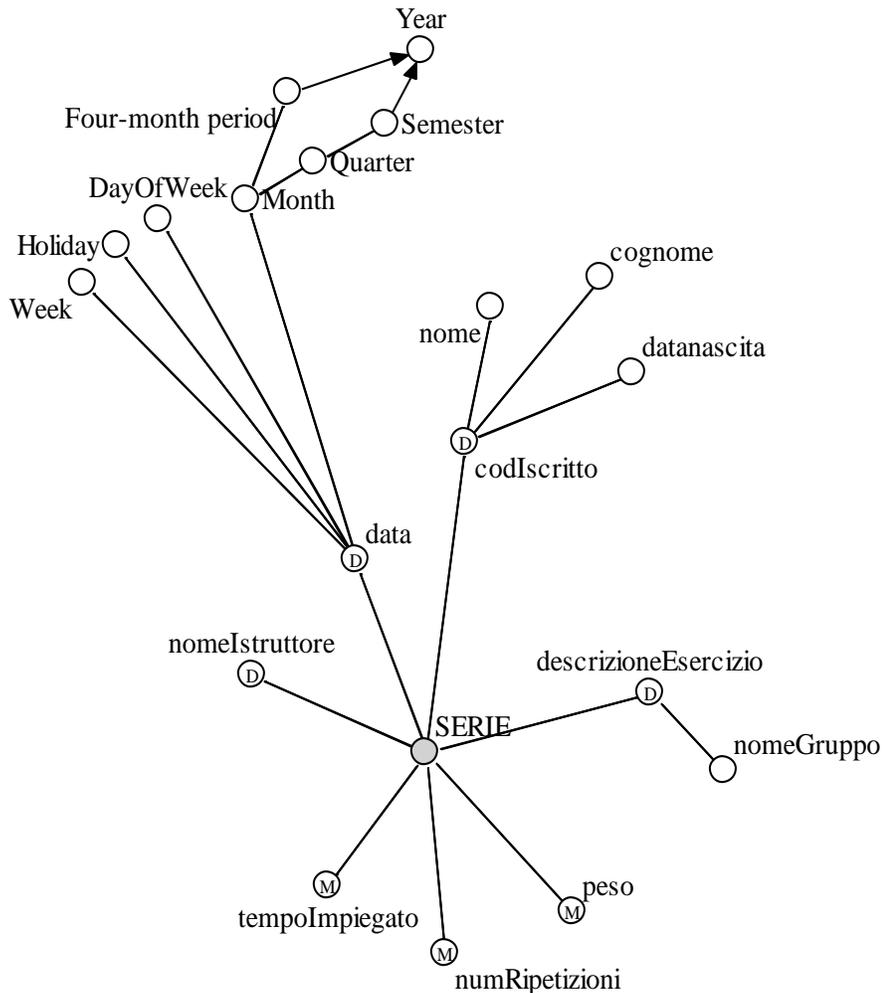
Dato il seguente schema E/R si individui un *fatto* di interesse, si costruisca l'*attribute tree* e il *fact schema* e si definisca la struttura relazionale dello *star schema*.

ISCRITTI (codIscritto, nome, cognome, datanascita)  
 ABBONAM (codIscritto:ISCRITTI, dataInizio, dataFine)  
 PRESENZE (codIscritto:ISCRITTI, data, numOre)  
 SCHEDE (codScheda, nomeIstruttore:ISTRUTTORI, codIscritto:PRESENZE, data:PRESENZE)  
 ISTRUTTORI (nomeIstruttore)  
 ESERCIZI\_IN\_SCHEDE (codScheda:SCHEDE, descrizioneEsercizio:ESERCIZI, posizione)  
 SERIE (codScheda:ESERCIZI\_IN\_SCHEDE, posizioneInScheda:ESERCIZI\_IN\_SCHEDE, posizioneInEsercizio, peso, numRipetizioni, tempoImpiegato)  
 ESERCIZI (descrizioneEsercizio, nomeGruppo:GRUPPI\_MUSCOLI)  
 GRUPPI\_MUSCOLI (nomeGruppo)

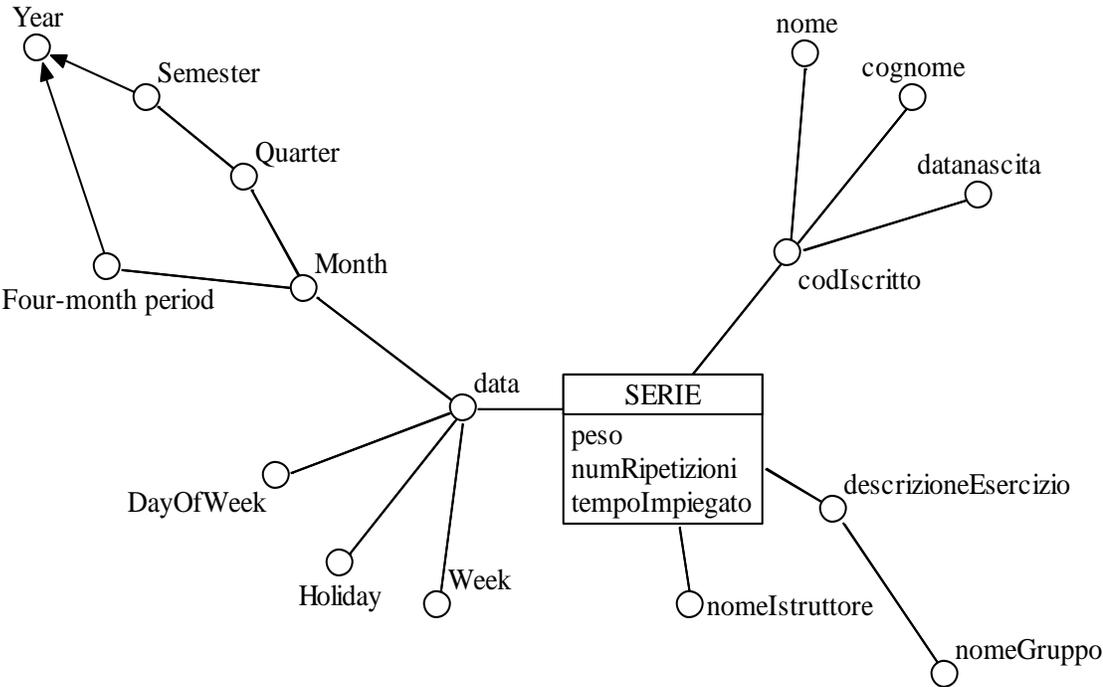
## Svolgimento

*Fatto scelto:* Serie

*Attribute tree:*



**Fact-Schema.**



## Esercizio 2

Date le relazioni:

FORNITORE( cod, nome, città, nazione )  
PRODOTTO( codProd, codFornitore:FORNITORE, materiale, categoria)

e la query:

```
SELECT FORNITORE.nome, PRODOTTO.materiale, PRODOTTO.categoria
FROM FORNITORE, PRODOTTO
WHERE FORNITORE.cod = PRODOTTO.codFornitore
AND PRODOTTO.materiale = 'Oro'
AND FORNITORE.nazione = 'Italia'
```

Nell'ipotesi che:

$NP_{FOR} = 160$              $NT_{FOR} = 500$   
 $NP_{PRO} = 550$             $NT_{PRO} = 150000$   
 $NK_{Materiale} = 400$       $NK_{nazione} = 40$

e sapendo che su FORNITORE.cod è costruito un indice clustered con  $NL_{cod} = 10$  e  $h = 2$

A) Si determini il miglior piano d'accesso per la risoluzione della query fra i seguenti:

1. nested-loop con relazione interna PRODOTTO
2. nested-loop con relazione interna FORNITORE

B) Si calcoli il numero di tuple attese della query

### Svolgimento

$f(\text{nazione}) = 1/NK_{\text{nazione}} = 1/40$   
 $f(\text{materiale}) = 1/NK_{\text{materiale}} = 1/400$

#### 1) *piani d'accesso*

- nested-loop con relazione interna PRODOTTO  
 $C_{TOT} = C_a(\text{COM}) + ET_{FOR} * C_a(\text{PRO})$   
 $C_a(\text{FOR}) = NP_{FOR} = 160$  (seq.)  
 $C_a(\text{PRO}) = NP_{PRO} = 550$  (seq.)  
 $ET_{FOR} = \lceil NT_{FOR} * f(\text{nazione}) \rceil = \lceil 500/40 \rceil = 13$   
 **$C_{TOT} = 160 + 13 * 550 = 7310$**
- nested-loop con relazione interna FORNITORE  
 $C_{TOT} = C_a(\text{PRO}) + ET_{PRO} * C_a(\text{FOR})$   
 $C_a(\text{PRO}) = NP_{PRO} = 550$  (seq.)  
a)  $C_a(\text{FOR}) = NP_{FOR} = 160$  (seq.)  
b)  $C_a(\text{FOR}) = h-1 + \lceil NL_{cod} * f(\text{cod=val}) \rceil + \lceil NP_{FOR} * f(\text{cod=val}) \rceil =$   
 $1 + \lceil 10/500 \rceil + \lceil 160/500 \rceil = 3$  (idx.)  
 $ET(\text{PRO}) = \lceil NT_{PRO} * f(\text{materiale}) \rceil = \lceil 150000/400 \rceil = 375$   
 **$C_{TOT} = 550 + 375 * 3 = 1675$**

#### 2) *tuple attese*

$ET : \lceil 1/40 * 1/400 * 150000 \rceil = 10$

### Esercizio 3

Dato il seguente schema relazionale

INGREDIENTE ( CodIng, Nome, Descrizione, Calorie)

ING\_RICETTA ( CodRicetta: RICETTA, CodIng: INGREDIENTE, Quantità)

RICETTA ( CodRicetta, Nome, Tipo, Descrizione, Tempo, Difficoltà, Calorie)

1. Scrivere la *stored procedure* che esegue il conteggio di tutte le ricette di tipo "Dessert" che NON contengono un dato ingrediente (il cui Nome viene letto in input).
2. Creare un *trigger* che tiene aggiornato il dato derivato "Calorie" di una ricetta. A fronte dell'inserimento di un nuovo ingrediente per una ricetta ricalcola il totale delle calorie ottenuto come media pesata delle calorie di ciascun ingrediente.

### Svolgimento

1. CREATE PROCEDURE Allergia  
@Ingred nvarchar(40)  
AS  
SELECT count(\*) FROM RICETTA  
WHERE tipo= "Dessert"  
AND codRicetta NOT IN  
(SELECT codRicetta FROM ING\_RICETTA IR, INGREDIENTE I  
WHERE I.CodIng = IR.CodIng  
AND Nome=@Ingred)
2. CREATE TRIGGER AggiornaCalorie  
ON ING\_RICETTA  
FOR INSERT  
AS  
BEGIN  
DECLARE @newCal REAL  
DECLARE @newRic nvarchar(10)  
  
SELECT @newRic=CodRicetta FROM INSERTED  
SELECT @newCal=SUM(I.Calorie\*IR.Quantità)/ SUM(IR.Quantità)  
FROM ING\_RICETTA IR, INGREDIENTE I  
WHERE I.CodIng = IR.CodIng  
AND IR.CodRicetta=@newRic  
UPDATE RICETTA R  
SET Calorie= @newCal  
WHERE R.CodRicetta=@newRic  
END

### Esercizio 4

Costruire il B+tree a PID per indicizzare le seguenti pagine dati:

Page 1	Page 2	Page 3	Page 4	Page 5	Page 6
12	13	15	99	40	25
35	23	31	19	42	22
48	47	33	98	24	17
1	46	6	4	32	56

Il B+tree deve essere di ordine  $g=2$ , con foglie di capacità 4 e deve avere altezza minima.

### Svolgimento

