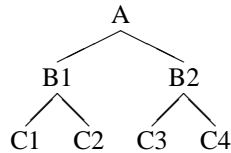


**Sistemi informativi II**  
**Prof. Stefano Paraboschi**  
*Prova dell'11-11-2005*

A. Si consideri la gerarchia di risorse:



Il sistema utilizza il protocollo di lock gerarchico per gestire l'accesso alle risorse della gerarchia. Si supponga che nello stato iniziale siano stati già concessi i seguenti lock a queste transazioni:

- T1: IXL su B1; XL su C1; SIXL su B2
- T2: IXL su B1; SL su C2; ISL su B2

1. Mostrare le richieste di lock che verranno generate dalla transazione T3 che vuole accedere in lettura a C2 e in lettura a C4;
2. Mostrare il grafo d'attesa che si otterrà in seguito alle ulteriori richieste di lock prodotte dalla transazione T4 che cerca di eseguire una scrittura su C2.

B. Si ha una base di dati con il seguente schema:

CONTO(Numero,CodFiliale,Cliente,Ammontare,Fido)  
MOVIMENTOINTRAFILIALE(NroProgr,ContoDa,ContoA,Valore)  
MOVIMENTOINTERFILIALE(NroProgr,ContoDa,ContoA,Valore)

La tabella MOVIMENTOINTRAFILIALE rappresenta le transazioni tra conti della stessa filiale, mentre MOVIMENTOINTERFILIALE descrive le transazioni in cui si realizza uno spostamento tra conti di filiali diverse. Ogni filiale dispone di una proprio DBMS.

- Progettare la frammentazione della base di dati, Presentare almeno 2 alternative per quanto riguarda la tabella MOVIMENTOINTERFILIALE, mettendo in evidenza i vantaggi e svantaggi di ciascuna soluzione. Scegliere quindi una delle alternative per lo svolgimento dei punti successivi.
- Mostrare, ai diversi livelli di trasparenza, il comando che inserisce un nuovo movimento.
- Mostrare, ai diversi livelli di trasparenza, il comando SQL che mostra i movimenti relativi ai conti del cliente "Mario Rossi" (si intendono i movimenti in cui il conto era sorgente o destinazione del trasferimento).

C. Illustrare le opportunità e le problematiche relative alla gestione della consistenza, nell'ambito della realizzazione di transazioni "acide" distribuite.

D. Si consideri il seguente schema di base di dati a oggetti:

```
class Giocatore {NumTessera: int, Nome: string, Squadra: ref(Squadra)}  
class Allenatore {Nome: string}  
class Squadra {Nazione: string, Giocatori: set(Giocatore), Allenatore: ref(Allenatore)}  
class Partita {Data: date, Squadra1: ref(Squadra), Squadra2: ref(Squadra), SetVintiSq1: int, SetVintiSq2: int}
```

1. Esprimere in OQL la query che restituisce i nomi degli allenatori delle squadre che hanno giocato una partita l'11/11/2005.
2. Esprimere in SQL:1999/OQL la query che restituisce per ogni squadra l'elenco delle partite vinte, presentando per ciascuna di esse il nome dell'avversario, la data e il punteggio (si ricorda che una partita è vinta dalla squadra che vince 3 set).
3. Illustrare le problematiche relative alla valutazione dell'uguaglianza profonda tra istanze della classe SQUADRA.

E. Illustrare le problematiche relative alla gestione di una struttura fisica sequenziale. In particolare, si illustrino i vantaggi di una struttura sequenziale ordinata. Come si possono integrare una organizzazione fisica sequenziale con una struttura ad albero?

F. Illustrare l'operatore di roll up.