Secondo compitino
Laboratorio Basi di Dati II
Modelli e DBMS di nuova generazione
21 Dicembre 2004

Questo compito contiene esercizi che devono essere fatti da chi ha seguito: solo il corso "*Laboratorio di Basi di dati II*" (LABO); solo il corso "*modelli dei dati e DBMS di nuova generazione*" (MODELLI); o entrambi i corsi.

Per chi ha seguito solo il corso di modelli, svolgere gli esercizi 1, 2, (5|6)
Per chi ha seguito solo il corso di Labo Basi di dati II, svolgere gli esercizi 3, 4, (5|6)
Per chi ha seguito entrambi i corsi, svolgere gli esercizi 1, 3, (5|6)

Se uno svolge piu' esercizi di quelli richiesti per la propria situazione, verra' considerata la migliore combinazione di esercizi nel caso di esercizi alternativi gli uni agli altri (a|b) e verranno assegnati punti extra negli altri casi.

Specificare nel foglio che consegnate in quale situazione vi trovate.

Esercizio 1 (MODELLI)
Si supponga di avere la base di dati operazionale di un gruppo di emittenti radio. Di ogni emittente si conosce la sede dalla quale trasmette (che si trova in una certa citta', regione, stato), il nome del direttore e il numero (medio) di persone che la seguono. Di ogni pubblicità che la radio trasmette si conosce il nome, il testo del messaggio, il nome dell'azienda che la sponsorizza la quale appartiene ad un certo settore di mercato (azienda al dettaglio, all'ingrosso, internazionale), il logo dell'azienda, e la sottotipologia e la tipologia della pubblicità. Esempi di tipologia sono: calzature, arredamento, alimentare, telecomunicazioni. Esempi di sottotipologia per la tipologia "telecomunicazioni" sono: telefonia fissa, telefonia mobile, satelliti.

Il gruppo di emittenti radio vuole confrontare le politiche pubblicitarie delle varie emittenti. A tale proposito, monitorizza giornalmente ogni spot pubblicitario in relazione al numero di volte in cui lo spot appare in ciascuna emittente, alla durata totale (cioè al tempo complessivo occupato nel contesto di uno stesso giorno dallo stesso spot nel palinsesto dell'emittente) e la percentuale di pubblico dell'emittente che segue lo spot (questa informazioni viene ricavata attraverso un'indagine statistica).

Si richiede di:

1. Definire una schema ER per il dominio sopra descritto.
2. Identificare il fatto che l'azienda e' interessata ad analizzare.
3. Stabilire le dimensioni di analisi e le misure che possono essere calcolate.
4. Specificare se le misure individuate sono additive, semi-additive, non additive. Per le misure semi-additive e non additive specificare se è possibile (e come) renderle additive.
5. Specificare le gerarchie che possono essere rappresentate per le dimensioni individuate.
6. Disegnare lo schema concettuale e logico del fatto.
7. Presentare l'oggetto DIMENSION di Oracle per la dimensione che presenta il maggior numero di aggregazioni
8. (Opzionale) Si supponga che occorra effettuare analisi sugli spot mandati in onda da emittenti della Liguria in relazione alla loro tipologia di appartenenza, con granularita' settimanale. Si definisca una vista materializzata opportuna tenendo in considerazione che la vista venga costruita immediatamente, venga aggiornata al commit delle operazioni, venga utilizzata dall'aggregate navigator. Si preveda di poter aggregare i dati rispetto a tutte le possibili combinazioni delle dimensioni individuate.

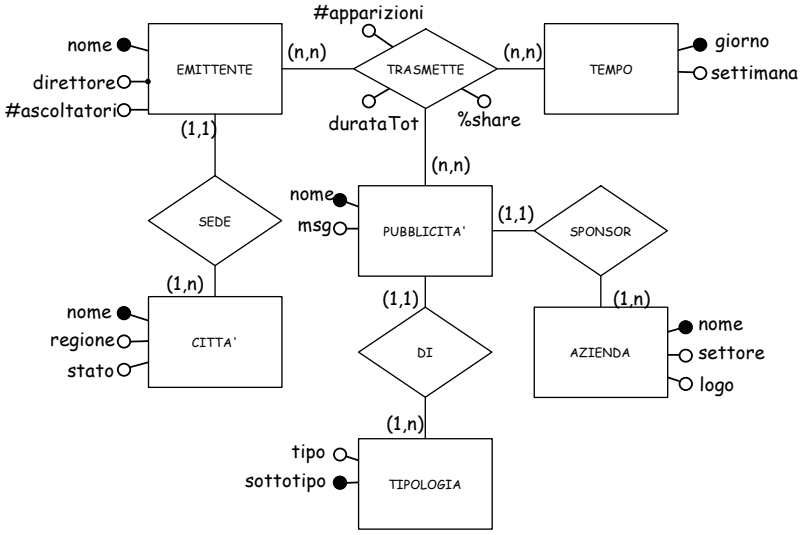


Figure 1. Schema ER della base di dati operazionale

Soluzione.

1. In Figura 1 viene riportato lo schema ER per il dominio applicativo descritto nell'esercizio. Si noti che lo schema non è perfettamente normalizzato. Di seguito vengono presentate le dipendenze funzionali esistenti tra gli attributi (che non sono chiavi dell'entità).

   Città: regione→ stato
2. Identificare il fatto che l'azienda e' interessata ad analizzare.
3. Stabilire le dimensioni di analisi e le misure che possono essere calcolate.
4. Specificare se le misure individuate sono additive, semi-additive, non additive. Per le misure semi-additive e non additive specificare se è possibile (e come) renderle additive.

Per rispondere a questi quesiti, costruisco la Tabella 1 in cui riporto i fatti, le dimensioni lungo cui è possibile analizzare i fatti e le misure relative. Per ogni misura indico infine se si tratta di misura additiva (A), non additiva (NA), oppure semi-additiva (SA).

E' ragionevole considerare le dimensioni EMITTENTE, PUBBLICITA', e TEMPO come dimensioni di analisi in quanto la trasmissione può essere studiata ed osservata da un punto di vista di EMITTENTE della trasmissione (chi emette), da un punto di vista di PUBBLICITA' (cosa è emesso) e da un punto di vista temporale (quando avviene la trasmissione).

Tabella 1. Fatti, dimensioni e misure

FATTI	DIMENSIONI	MISURE	TIPO MISURE
TRASMISSIONI	EMITTENTE	#APPARIZIONI	A
	PUBBLICITA'	%SHARE	NA
	TEMPO	DURATATOT	A

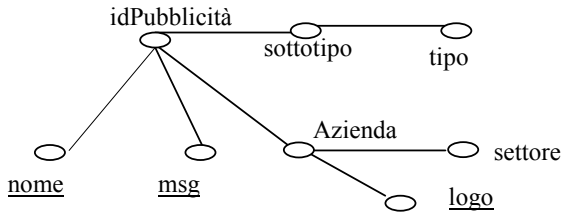
Si noti che la misura **%SHARE** non è additiva. Se invece della percentuale si considera il numero di ascoltatori che seguono la pubblicità la misura diventa additiva (**ascolti**). **Ascolti** viene ricavato applicando la formula:

%SHARE * #ascoltatori

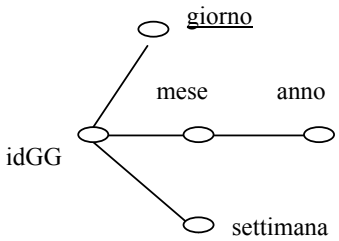
Quindi, nel passaggio dalla base di dati operazionale, alla base di dati riconciliata (e/o al datawarehouse) riporterò il numero di utenti, invece della percentuale.

5. Specificare le gerarchie che possono essere rappresentate per le dimensioni individuate.

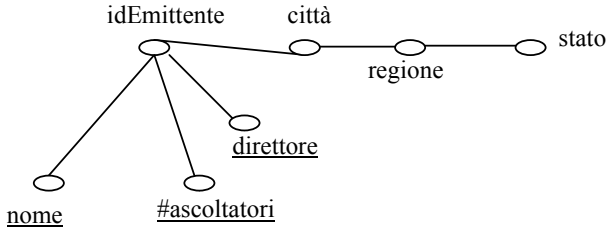
Pubblicità



Tempo



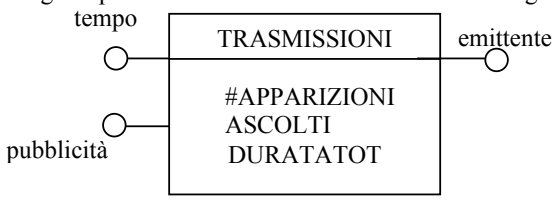
Emittente



Si noti che nella dimensione tempo, oltre ai livelli giorno e settimana che sono necessari per le richieste del testo si è deciso di considerare anche i livelli mese e anno. Questo permette di specificare maggiori analisi. Sono infice state introdotte delle chiavi artificiali che verranno poi utilizzate nella definizione del modello concettuale e logico del DW.

6. Disegnare lo schema concettuale e logico del fatto.

Di seguito presentiamo il fatto con le dimensioni. Le gerarchie delle dimensioni sono state presentate al punto 5



A partire dallo schema concettuale è possibile individuare lo schema a stella riportato in Figura 2.

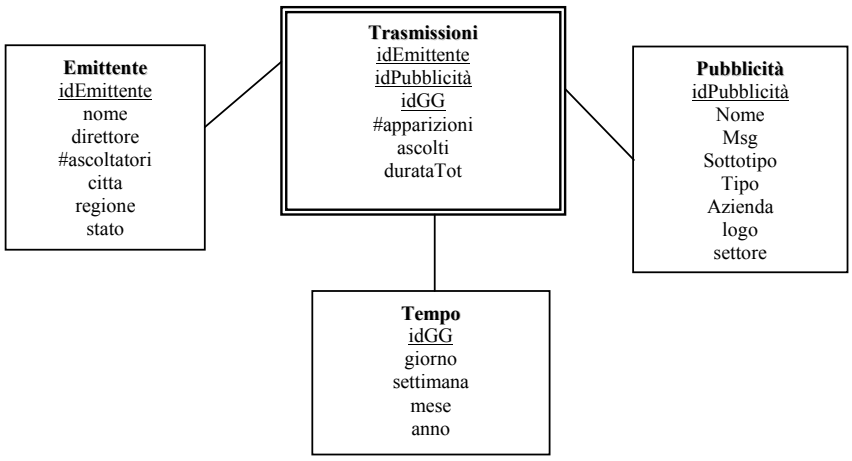


Figura 2. Schema a stella per il fatto acquisti della catena di supermercati

Lo schema a stella in Figura 2 corrisponde al seguente schema relazionale

Emittente(idEmittente,nome,direttore,#ascoltatori,citta,regione,stato)
Pubblicità(idProd,Nome,Msg,Sottotipo,Tipo,Azienda,logo,settore)
Tempo(idGG,giorno,settimana,mese,anno)
Trasmissioni(idEmittente,,idPubblicità,idGG,#apparizioni,ascolti,durataTot)

7. Presentare l'oggetto DIMENSION di Oracle per la dimensione che presenta il maggior numero di aggregazioni

```
CREATE DIMENSION Pubblicità_D
LEVEL pubbl_l   IS Pubblicità.idPubblicità
LEVEL sottot_l    IS Pubblicità.sottotipo
LEVEL tipo_l   IS Pubblicità.tipo
LEVEL azienda_l IS Pubblicità.azienda
LEVEL settore_l IS Pubblicità.settore
HIERARCHY tipo_H (
        pubbl _l CHILD OF
        sottot_l CHILD OF tipo)
HIERARCHY azienda_H (
        pubbl _l   CHILD OF
        azienda_l CHILD OF
        settore_l)
ATTRIBUTE pubbl _l DETERMINES Pubblicità.nome
ATTRIBUTE pubbl _l DETERMINES Pubblicità.msg
ATTRIBUTE azienda_l DETERMINES Pubblicità.logo;
```

8. (Opzionale) Si supponga che occorra effettuare analisi sugli spot mandati in onda da emittenti della Liguria in relazione alla loro tipologia di appartenenza, con granularita' settimanale. Si definisca una vista materializzata opportuna tenendo in considerazione che la vista venga costruita immediatamente, venga aggiornata al commit delle operazioni, venga utilizzata dall'aggregate navigator. Si preveda di poter aggregare i dati rispetto a tutte le possibili combinazioni delle dimensioni individuate.

```
CREATE MATERIALIZED VIEW v_tipospot
BUILD IMMEDIATE
REFRESH ON COMMIT
ENABLE QUERY REWRITE
AS
    SELECT Emittente.idEmittente, Pubblicità.Tipo, Tempo.settimana, SUM(#Trasmissioni)
    FROM Trasmissione, Emittente, Tempo, Pubblicità
    WHERE Trasmissione.idGG = Tempo.idGG AND
            Trasmissione.idEmittente = Emittente.idEmittente AND
            Trasmissione.idPubblicità=Pubblicità.idPubblicità AND
            Emittente.regione='Liguria'
    GROUP BY CUBE(Emittente.regione,Pubblicità.Tipo, Tempo.settimana)
```

Esercizio 2 (MODELLI)
Si considerino le seguenti transazioni:

1. broccoli, carciofi, fagioli, patate
2. broccoli, carciofi
3. broccoli, fagioli, patate
4. fagioli, patate
5. fagioli, patate
6. broccoli, fagioli
7. carciofi, fagioli, patate

Determinare le regole di associazioni per tale insieme di transazioni. Supporto minimo 50% e confidenza 75%.

Invece di considerare il supporto percentuale, considero il numero di occorrenze. Dal momento che ci sono 7 transazioni il supporto minimo è 4.

Candidati:		Sottoinsiemi frequenti
B	4	B
~~C~~	~~3~~	
F	6	F
P	5	P
~~BF~~	~~3~~	
~~BP~~	~~2~~	
FP	5	FP

Determino adesso le regole di associazione considerando la confidenza delle regole

$C(F \rightarrow P) = \frac{S(\{FP\})}{S(F)} = \frac{5}{6} > .75$

$C(P \rightarrow F) = \frac{S(\{FP\})}{S(P)} = \frac{5}{5} = 1$

Le regole di associazione sono quindi: $F \rightarrow P$ e $P \rightarrow F$

Esercizio 3 (LABO)
Si considerino le seguenti due relazioni:

VillaggiTuristici(codV, nome,località, #posti,responsabile)
Ospiti(idOsp,nome,quota,inizioSoggiorno,fineSoggiorno,villaggio)

In cui InizioSoggiorno e FineSoggiorno delimitano il periodo di tempo in cui un ospite sta nel Villaggio, e Ospiti.villaggio e' chiave esterna su VillaggiTuristici.

Si supponga che l'interrogazione che viene eseguita principalmente su questo schema e':

```
SELECT codV, COUNT(*)
FROM VillaggiTuristici, Ospiti
WHERE codV=villaggio AND localita = '5 Terre' AND
        inizioSoggiorno BETWEEN '24/12/2004' AND '30/12/2004'
GROUP BY codV
```

Rispondere alle seguenti domande (giustificando le risposte):
1. Ideare una struttura di memorizzazione primaria delle due relazioni e strutture ausiliarie di accesso che permettano di ottimizzare l'esecuzione della precedente interrogazione.

Si noti che la presenza del group by non altera le scelte di memorizzazione della relazione dal momento che il raggruppamento viene effettuato sulla chiave principale della relazione.
Esistono vari modi di poter memorizzare fisicamente le due relazioni. Per prima cosa, la memorizzazione fisica puo' essere separata (cioe' 2 file, uno per ogni relazione) oppure unita (un solo file che contiene una tupla di villaggioTuristico seguito da tutte le tuple di ospiti che sono stati in quel villaggio turistico). Analizziamo separtamente i due casi.

Relazioni separate
In questo caso puo' essere utile clusterizzare le tuple di Ospiti in base all'attributo **inizioSoggiorno** e creare un B+-albero su tale attributo. In questo modo si ottimizza la la verifica della condizione posta sull'attributo **inizioSoggiorno**.
La tabella VillaggiTuristici invece potrebbe essere clusterizzata sulla base dell'attributo **codV** e creato un indice hash su tale attributo. Un'alternativa a questa scelta puo' essere quella di clusterizzare VillaggiTuristici rispetto a localita' e creare un indice hash su tale attributo (in questo caso potrebbe essere inutile creare un indice hash anche su codV).

Relazioni unite
In questo caso può essere utile clusterizzare le tuple di VillaggiTuristici rispetto a località e creare un indice hash su tale attributo. Dopodiché le tuple di Ospiti vengono partizionate rispetto al valore dell'attributo villaggio e ogni partizione inserita di seguito alla tupla di VillaggioTuristico a cui fanno riferimento. In ogni partizione le tuple di Ospite devono essere ordinate rispetto all'attributo inizioSoggiorno.

2. Fare vedere come i dati vengono organizzati nel caso in cui le due tabelle, dal punto di vista logico, contengono i seguenti dati

codV	nome	localita	#posti	Responsabile
A001	Vernazza Tour	5 Terre	100	M. Mesiti
A002	Eolie Club	Eolie	150	R. Panetta
A003	Rio Maggiore Club Med	5 Terre	80	G. Guerrini

idOsp	Nome	quota	InizioSoggiorno	FineSoggiorno	Villaggio
O01	S. Accordino	1.000	24/12/2004	06/01/2005	A001
O02	A. Boca	800	19/12/2004	26/12/2004	A002
O03	A. Boveri	1.100	13/12/2004	20/12/2004	A003
O04	A. Cislaghi	900	28/12/2004	10/01/2005	A001
O05	R. d'Amato	700	24/12/2004	30/12/2004	A001
O06	V. D'Errico	1.050	01/12/2004	15/12/2004	A003

Faccio vedere la rappresentazione fisica delle precedenti tabelle nei due casi discussi nel punto 1.

Relazioni separata

Ospiti è ordinata rispetto a InizioSoggiorno

IdOsp	Nome	quota	InizioSoggiorno	FineSoggiorno	Villaggio
O06	V. D'Errico	1.050	01/12/2004	15/12/2004	A003
O03	A. Boveri	1.100	13/12/2004	20/12/2004	A003
O02	A. Boca	800	19/12/2004	26/12/2004	A002
O01	S. Accordino	1.000	24/12/2004	06/01/2005	A001
O05	R. d'Amato	700	24/12/2004	30/12/2004	A001
O04	A. Cislaghi	900	28/12/2004	10/01/2005	A001

L'organizzazione fisica di VillaggiTuristici e' uguale a quella logica (considero la prima alternativa presentata che dice di ordinare le tuple rispetto a codV)

CodV	nome	localita	#posti	Responsabile
A001	Vernazza Tour	5 Terre	100	M. Mesiti
A002	Eolie Club	Eolie	150	A. Maddalena
A003	Rio Maggiore Club Med	5 Terre	80	G. Guerrini

Relazioni unite

A001	Vernazza Tour	5 Terre	100	M. Mesiti	
O01	S. Accordino	1.000	24/12/2004	06/01/2005	A001
O05	R. d'Amato	700	24/12/2004	30/12/2004	A001
O04	A. Cislaghi	900	28/12/2004	10/01/2005	A001
A003	Rio Maggiore Club Med	5 Terre	80	G. Guerrini	
O06	V. D'Errico	1.050	01/12/2004	15/12/2004	A003
O03	A. Boveri	1.100	13/12/2004	20/12/2004	A003
A002	Eolie Club	Eolie	150	R. Panetta	
O02	A. Boca	800	19/12/2004	26/12/2004	A002

3. Presentare un esempio di interrogazione la cui esecuzione diventa particolarmente costosa sulla base della organizzazione fisica presentata al punto 1

Faccio vedere una interrogazione per ogni tipo di memorizzazione fisica

Relazioni unite

```
SELECT codV, nome FROM VillaggiTuristici
```

La ricerca è peggiorato dalla dispersione delle tuple di VillaggioTuristici nelle tuple di Ospiti.

Relazioni separate

```
SELECT codV, Ospiti.nome
FROM VillaggiTuristici, Ospiti
WHERE codV=villaggio AND
            fineSoggiorno BETWEEN '24/12/2004' AND '30/12/2004'
```

La mancanza di un indice su fineSoggiorno e il fatto che le tuple non sono ordinate rispetto a tale attributo, rende l'esecuzione dell'interrogazione più costosa-

4. Come potrebbe essere modificato lo schema presentato al punto 1 nel caso in cui si considerasse anche l'interrogazione

```
SELECT codV, nome, quota
FROM VillaggiTuristici, Ospiti
WHERE codV=villaggio AND localita = '5 Terre' AND
        inizioSoggiorno BETWEEN '24/12/2004' AND '30/12/2004' AND
        quota = 1000
```

Discuto questo in base alla memorizzazione fisica delle due relazioni

Relazioni unite
Si potrebbe aggiungere un indice Hash su (villaggio,quota). Questo però potrebbe dipendere dalla selettività delle chiavi di ricerca (villaggio,inizioSoggiorno) e (villaggio,quota). Infatti se tutti quelli che iniziano la vacanza nello stesso giorno, pagano la stessa quota è inutile introdurre l'indice Hash su (villaggio,quota).

Relazioni separate
Dal momento che si aggiunge una condizione sull'attributo quota di Ospiti, aggiungerei un indice Hash su tale attributo. Per il resto lascerei inalterato.

Esercizio 4 (LABO)
Mostrare la struttura hash estensibile, con capacita' di ogni pagina 2, supponendo di inserire nell'ordine i record le cui pseudochiavi hanno i seguenti valori:

h(r1)=1010000 h(r2)=0111000 h(r3)=1011001 h(r4)=0110000 h(r5)= 00100100 h(r6)=11101000 h(r7)=010100

```
<a>
    <b>
     <e>ciao</e>
     <f>20</f>
    </b>
    <c>
     <h>20</h>
     <g>ciao</g>
     <i>38</i>
     <i>98</i>
     <f>20</f>
    </c>
</a>
```

```
<a>
    <b>
      <e>ciao</e>
      <z>20</z>
      <z>30</z>
    </b>
    <b>
      <e>ciao</e>
    </b>
    <d>
      <z>20</z>
    </d>
    <c alpha="4">
      <m>20</m>
      <g>ciao</g>
    </c>
</a>
```

Figura 1 Due documenti XML

Esercizio 5 (MODELLI/LABO)
Si considerino i due documenti XML in Figura 1.

1. Scrivere un DTD che generalizzi la struttura dei due documenti. Il DTD deve descrivere i due documenti in modo conciso. I due documenti devono essere validi per tale DTD.
2. Se invece del DTD, si dovesse generare un XML schema, quali elementi sarebbe meglio dichiarare globali? Giustificare la risposta.
3. Scrivere la dichiarazione dell'elemento b in XML schema.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<catalog>
        <cd>
                <title>Empire Burlesque</title>
                <artist>Bob Dylan</artist>
                <country>USA</country>
                <company>Columbia</company>
                <price>10.90</price>
                <year>1985</year>
        </cd>
        <cd>
                <title>Hide your heart</title>
                <artist>Bonnie Tyler</artist>
                <country>UK</country>
                <company>CBS Records</company>
                <price>9.90</price>
                <year>1988</year>
        </cd>
        <cd>
                <title>Greatest Hits</title>
                <artist>Dolly Parton</artist>
                <country>USA</country>
                <company>RCA</company>
                <price>9.90</price>
                <year>1982</year>
        </cd>
</catalog>
```

Figura 2 Un documento XML rappresentate CD musicali

Esercizio 6 (MODELLI/LABO)
Si consideri il documento XML in Figura 2.

Sviluppare un foglio di stile XSL che permetta di generare una pagina HTML in cui viene riportata la tabella (titolo,company,price) dei CD di Bob Dylan del 1985 che costano piu' di 10 euro.