

Exerciții MySQL

șl. Radu Damian,

Facultatea de Electronică, Telecomunicații și Tehnologia Informației, Iași

Lucrarea de față este orientată spre investigarea performanței accesului la bazele de date MySQL, și modalitatea în care normalizarea bazei de date și adăugarea de indecși (**potriviti**) pentru tabele duc la creșterea performanței.

Se utilizează mașina virtuală cu Windows 2000 rezultată din laboratorul precedent (este disponibilă și o mașină de referință în rețeaua internă a facultății).

Datele corespund unei aplicații ipotetice în care un depozit trimite produse caracterizate de **tip**, **nume** și **ambalaj** la diverse destinații. Pentru fiecare produs există o **masă netă**, dar și o **masă brută** (rezultată prin adăugarea **tarei**: o sumă a masei diverselor recipiente/dispozitive prezente în **momentul** cântării).

Performanța server-ului se va măsura/estima în funcție de timpul raportat de MySQL Query Browser după fiecare interogare: "**x** rows fetched in **y** s (**z** s)". În acest răspuns se indică numărul total de linii rezultate (x), timpul necesar pentru comunicarea rezultatelor aplicației client (y) și timpul necesar intern pentru realizarea interogării (z). Se menționează faptul că nu este o măsură absolută (acești timpi depind de ocuparea server-ului, și de exemplu repetarea unei interogări are ca efect de obicei scăderea acestui timp: server-ul este activ și nu "idle").

Să se realizeze interogările specificate în modul de lucru, iar pe o foaie de hârtie să se completeze tabelul următor. La finalul lucrării se scriu pe foaia de hârtie concluziile observate din analiza rezultatelor.

Pas	Linii rezultate	Timp intern de execuție [s]	Timp de comunicare rezultate	Observații
1	7737	0.0097	0.3732	Cu/Fără index, normalizat/ne
...

Modul de lucru

A. Baza de date nenormalizata

Se încarcă în server-ul MySQL o bază de date inițială (detalii la sfârșitul laboratorului 1). Nu este necesară atribuirea drepturilor de acces (curs 3) deoarece se va lucra în MySQL Query Browser.

Tabelul cu care se lucrează în această secțiune este **inregistrare**.

I...	Data	Data_Lucru	Lot	Brut	Net	Amb	Tip	Prod	Dest
1	2008-02-29 11:01:13	2008-02-29	40	95.2	85.2	Lada	Refrigerat	Cap+Ghiare	Distributie SAFIR
2	2008-02-29 11:09:28	2008-02-29	40	110.8	103	Lada	Refrigerat	Oase Pulpe Dezosate	AGRICOLA INTERNATIONAL
3	2008-02-29 11:15:45	2008-02-29	40	830.2	770.1	Lada	Refrigerat	Pulpa cu OS	AGRICOLA INTERNATIONAL
4	2008-02-29 11:21:39	2008-02-29	40	34	30.1	Punga	Refrigerat	Pulpa cu OS	Distributie SAFIR
5	2008-02-29 11:22:49	2008-02-29	40	8.4	5.8	Punga	Refrigerat	Pulpa cu OS	Distributie SAFIR
6	2008-02-29 11:24:42	2008-02-29	40	829.8	772.6	Lada	Congelat	MDM spinari	Distributie SAFIR
7	2008-02-29 11:29:32	2008-02-29	40	624.6	554.1	Punga	Refrigerat	Pui GRILL	AGRICOLA INTERNATIONAL
8	2008-02-29 11:37:17	2008-02-29	40	1.6	1.1	Punga	Refrigerat	Pulpe INFERIOARE	Distributie SAFIR
9	2008-02-29 11:46:19	2008-02-29	40	826.6	785	Lada	Refrigerat	Piept cu OS	AGRICOLA INTERNATIONAL
10	2008-02-29 11:48:41	2008-02-29	40	463.8	406.3	Lada	Refrigerat	Pui GRILL	AGRICOLA INTERNATIONAL
11	2008-02-29 12:02:09	2008-02-29	40	614	547.5	Lada	Refrigerat	Pui GRILL	AUCHAN PITESTI GASTRO
12	2008-02-29 12:06:05	2008-02-29	40	26.8	22	Tava	Refrigerat	Pulpa cu OS	AUSHAN TG. MURES CARMANGERIE

Pasul 1

Se afișează toate liniile din tabelul **inregistrare**. Interogare:

```
SELECT * FROM inregistrare i;
```

Rezultat tipic intern ~0.0020s

Pasul 2

Se afișează toate liniile din tabelul **inregistrare** care au legătură cu un anumit beneficiar. Interogare:

```
SELECT * FROM inregistrare i WHERE Dest='AGRICOLA INTERNATIONAL';
```

Rezultat tipic intern ~0.0037s

Pasul 3

Se afișează toate liniile din tabelul **inregistrare** care au legătură cu un anumit beneficiar și cu un anumit tip de ambalaj. Interogare:

```
SELECT * FROM inregistrare i WHERE Dest='AGRICOLA INTERNATIONAL' AND Amb='Lada';
```

Rezultat tipic intern ~0.0041s

Pasul 4

Se afișează toate liniile din tabelul **inregistrare** care au legătură cu un anumit beneficiar, cu un anumit tip de ambalaj și cu un anumit tip de produs. Interogare:

```
SELECT * FROM inregistrare i WHERE Dest='AGRICOLA INTERNATIONAL' AND Amb='Lada' AND Tip='Refrigerat';
```

Rezultat tipic intern ~0.0050s

Pasul 5

Se afișează toate liniile din tabelul **inregistrare** care au legătură cu un anumit beneficiar, cu un anumit tip de ambalaj, cu un anumit tip de produs și cu un anumit produs. Interogare:

```
SELECT * FROM inregistrare i WHERE Dest='AGRICOLA INTERNATIONAL' AND Amb='Lada' AND Tip='Refrigerat' AND Prod='ARIPi';
```

Rezultat tipic intern ~0. 0090s

Pasul 6

Se repetă filtrul de la pasul 5, cu ordonarea (ASC/DESC) după dată a rezultatelor. Interogare:

```
SELECT * FROM inregistrare i WHERE Dest='AGRICOLA INTERNATIONAL' AND Amb='Lada' AND Tip='Refrigerat' AND Prod='ARIPi' ORDER BY `Data` DESC;
```

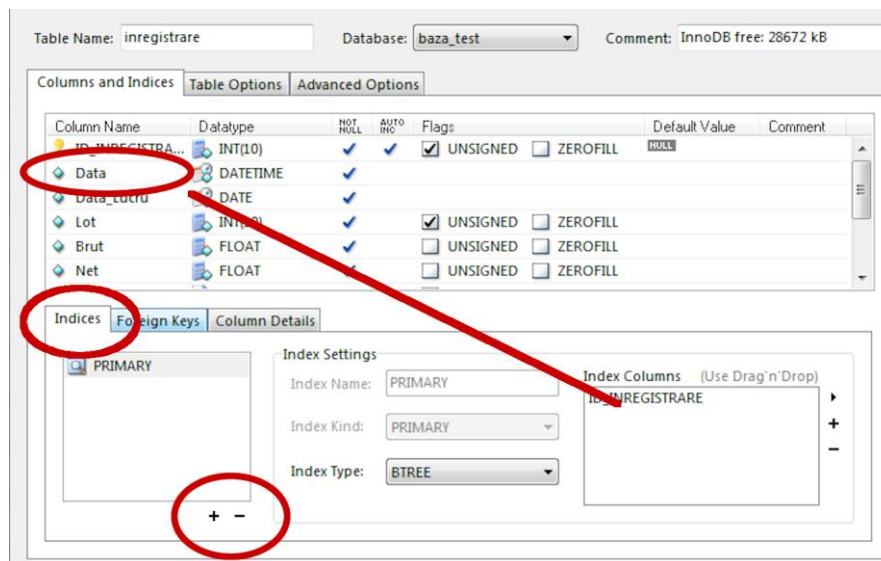
Rezultat tipic intern ~0. 0113s

Pasul 7

Se adauga indecși pentru mărimile care pot apărea în filtre (WHERE) sau ordonări (ORDER BY).

Adăugarea de indecși în MySQL Query Browser se face cu succesiunea de comenzi următoare:

Click dreapta pe numele tabelului -> Edit Table. În fereastra apărută se identifică zona "Indices" și se adaugă un index cu butonul marcat cu semnul "+". După alegerea unui nume pentru index se selectează indexul respectiv în listă (se remarcă existența altor indecși deja definiți sau a cheii primare în listă) și se adaugă coloana/coloanele corespunzătoare indexului în zona "Index Columns" prin drag'n'drop din lista de coloane. Se apasă butonul "Apply changes".



Se poate observa în fereastra următoare comanda SQL care va crea acel/acele index(cși), de exemplu "ALTER TABLE `baza_test`.`inregistrare` ADD INDEX `Index_2`(`Data`);". Se verifică în aplicația MySQL Administrator efectul la nivelul datelor (Catalogs -> "numele bazei de date"). În acest caz se poate obține:

Data: 1.5M, Index: 1.3M

Observație: O bază de date eficientă va avea o cantitate mare de date consumată (spațiu brut de stocare) pentru indecși (la fel de mare/mai mare/mai mică: dependent de datele efective memorate).

Pasul 7

Se repetă interogarea de la **pasul 6**, cu indecșii corespunzători prezenți în baza de date. Interogare:

```
SELECT * FROM inregistrare i WHERE Dest='AGRICOLA INTERNATIONAL' AND Amb='Lada' AND Tip='Refrigerat' AND Prod='ARIP1' ORDER BY `Data` DESC;
```

Rezultat tipic intern ~0. 0030s

Concluzii

Pe parcursul secțiunii A se verifică modul în care, pe măsură ce filtrul devine mai complex, timpul intern de prelucrare crește, în timp ce timpul de transfer/comunicare a rezultatelor scade datorită micșorării numărului de linii din rezultat. De asemenea se remarcă efectul prezenței indecșilor pentru toate interogările menționate.

B. Baza de date normalizata

Se incarca baza de date normalizata.

I...	Data	Data_Lucru	Lot	Ambalaj	Tip_Produs	Produs	Destinatie	Brut	Net
▶ 1	2008-02-29 11:01:13	2008-02-29	40	3	1	29	34	95.2	85.2
2	2008-02-29 11:09:28	2008-02-29	40	3	1	53	23	110.8	103
3	2008-02-29 11:15:45	2008-02-29	40	3	1	8	23	830.2	770.1
4	2008-02-29 11:21:39	2008-02-29	40	1	1	8	34	34	30.1
5	2008-02-29 11:22:49	2008-02-29	40	1	1	8	34	8.4	5.8
6	2008-02-29 11:24:42	2008-02-29	40	3	2	35	34	829.8	772.6
7	2008-02-29 11:29:32	2008-02-29	40	1	1	2	23	624.6	554.1
8	2008-02-29 11:37:17	2008-02-29	40	1	1	9	34	1.6	1.1
9	2008-02-29 11:46:19	2008-02-29	40	3	1	15	23	826.6	785

Pasul 8

Se repetă filtrul de la **pasul 6**, corespunzător cu notațiile numerice din coloanele bazei de date.

Interogare:

```
SELECT * FROM inregistrare i WHERE Destinatie= 23 AND Ambalaj= 3 AND Tip_Produs=1 AND Produs=20 ORDER BY `Data` DESC;
```

Rezultat tipic intern ~0. 0079s

Pasul 9

Se adaugă indecși. Se verifică în Mysql Administrator efectul.

Data: 1.5M, Index: 0.9M

Observație: Indecșii după coloane numerice sunt mai eficienți (și ca viteză și ca spațiu ocupat)

Pasul 10

Se repetă interogarea de la **pasul 8**, cu indecșii corespunzători prezenți în baza de date. Interogare:

```
SELECT * FROM inregistrare i WHERE Destinatie= 23 AND Ambalaj= 3 AND Tip_Produs=1 AND Produs=20  
ORDER BY `Data` DESC;
```

Rezultat tipic ~0.0121s (0.0057s)

Pasul 11

15. Se limitează cantitatea de date la minimumul necesar (ID, Data, Masa netă). Interogare:

```
SELECT ID_INREGISTRARE,`Data`,`Net` FROM inregistrare i WHERE Destinatie = 23 AND Ambalaj = 3 AND  
Tip_Produs = 1 AND Produs = 20 ORDER BY `Data` DESC;
```

Rezultat tipic ~0.0042s (0.0023s)

C. Asamblare rezultate in baza de date normalizata

Pasul 12

Adunarea și afișarea tuturor datelor prezente inițial în baza de date nenormalizată și limitarea numărului de rezultate (LIMIT). Interogare:

```
SELECT inr.`ID_INREGISTRARE`,inr.`Data`,inr.`Data_Lucru`,inr.`Lot`,l.`Data` AS data_lot,l.`Numar` AS  
numar_lot, inr.`Brut`,inr.`Net`,a.`Nume` AS nume_amb, a.`Scurt` AS scurt_amb, t.`Nume` AS nume_tip,  
t.`Scurt` AS scurt_tip, p.`Nume` AS nume_prod, p.`Scurt` AS scurt_prod, b.`Nume` AS scurt_ben,  
b.`Complet` AS nume_ben, d.`Nume` AS scurt_dest, d.`Adresa` AS nume_dest  
FROM `inregistrare` AS inr  
LEFT JOIN `lot` AS l ON (inr.`Lot`=l.`ID_LOT`)  
LEFT JOIN `ambalaj` AS a ON (inr.`Ambalaj`=a.`ID_AMBALAJ`)  
LEFT JOIN `tip_produs` AS t ON (inr.`Tip_Produs`=t.`ID_TIP`)  
LEFT JOIN `produs` AS p ON (inr.`Produs`=p.`ID_PRODUS`)  
LEFT JOIN `destinatie` AS d ON (inr.`Destinatie`=d.`ID_DESTINATIE`)  
LEFT JOIN `beneficiar` AS b ON (d.`Beneficiar`=b.`ID_BENEFICIAR`)  
ORDER BY `Data` DESC LIMIT 10
```

Rezultat tipic

Fara index ~0.0121s (0.0190s)

Cu index ~0.0108s (0.0007s)

D. Căutari Complexe în baza de date normalizată

În această secțiune se recomandă analiza în detaliu a interogărilor pentru a vedea cum se pot aduna date din mai multe tabele, și cum pot fi utilizate interogările SQL pentru obținerea direct a rezultatului dorit. Se recomandă în general utilizarea SQL pentru prelucrarea datelor (discuție cod compilat/interpretat, optimizare la curs mai târziu).

Pasul 13

Aflarea număr/tip de ambalaje plecate la data 11.03.2008 spre destinația "INTEREX Vaslui" (in vederea recuperării lor de ex. etc). Interogare:

```
SELECT inr.`Ambalaj`,inr.`Produs`,leg.`Tara`,SUM(leg.`Nr`) AS Val
FROM `inregistrare` AS inr
LEFT JOIN `leg_inreg_tara` AS leg ON (inr.`ID_INREGISTRARE`=leg.`Inregistrare`)
WHERE inr.`Destinatie` = 100 AND inr.`Data_Lucru` = '2008-03-11'
GROUP BY leg.`Tara` ASC,inr.`Ambalaj` ASC,inr.`Produs` ASC WITH ROLLUP
```

Rezultat tipic

Fara index ~0.0031s (0.1216s)

Cu index ~0.0045s (0.0012s)

Pasul 14

Cantitatea totala de produse organizata pe tip produs, predata beneficiarilor intre 10.03.2008 și 18.03.2008. Organizare pe tip produs, total pentru fiecare beneficiar in parte. Interogare:

```
SELECT d.`Beneficiar`,inr.`Tip_Produs`,ROUND(SUM(inr.`Net`),2) AS Val
FROM `inregistrare` AS inr
LEFT JOIN `destinatie` AS d ON (inr.`Destinatie`=d.`ID_DESTINATIE`)
WHERE inr.`Data_Lucru` BETWEEN '2008-03-10' AND '2008-03-18'
GROUP BY d.`Beneficiar` ASC,inr.`Tip_Produs` ASC WITH ROLLUP
```

Rezultat tipic

Fara index ~0.0016s (0.0089s)

Cu index ~0.0016s (0.0092s)

Pasul 15

Afisarea tarei totale si a modului de obținere pentru fiecare cantarire. Interogare:

```
SELECT Inregistrare, CONVERT(GROUP_CONCAT(l.Nr,'x',t.Greutate,'kg') USING latin1) AS Lista,
SUM(Nr*Greutate) AS total FROM leg_inreg_tara l
LEFT JOIN tara AS t ON (l.`Tara`=t.ID_TARA)
GROUP BY Inregistrare;
```

Rezultat tipic

Fara index ~0.0822s (0.0167s)

Cu index ~0.0801s (0.0166s)

Cu index (Inregistrare,Tara) ~0.0921s (0.0091s)

Pasul 16

Verificarea eventualelor erori

Interogare:

```
SELECT Inregistrare, SUM(Nr*Greutate) AS total,(i.Brut-i.Net) AS diferenta,  
ROUND(ABS(SUM(Nr*Greutate)-(i.Brut-i.Net)),4) AS eroare FROM leg_inreg_tara l  
LEFT JOIN tara AS t ON (l.`Tara`=t.ID_TARA)  
LEFT JOIN inregistrare AS i ON (l.`Inregistrare`=i.ID_INREGISTRARE)  
GROUP BY Inregistrare;
```

Rezultat tipic ~0.1340s (0.0093s)

Interogare:

```
SELECT ID_INREGISTRARE, (Brut-Net) AS diferenta, ROUND(ABS(suma_tara.total-Brut+Net),4) AS eroare  
FROM inregistrare AS i  
LEFT JOIN  
(SELECT Inregistrare, SUM(Nr*Greutate) AS total FROM leg_inreg_tara l  
LEFT JOIN tara AS t ON (l.`Tara`=t.ID_TARA) GROUP BY Inregistrare) AS suma_tara  
ON (suma_tara.`Inregistrare`=i.ID_INREGISTRARE)
```

Rezultat tipic ~3.3426s (0.3144s)

Interogare:

```
SELECT Inregistrare,total,ROUND(ABS(total-i.Brut+i.Net),4) AS eroare FROM (SELECT Inregistrare,  
SUM(Nr*Greutate) AS total FROM leg_inreg_tara l  
LEFT JOIN tara AS t ON (l.`Tara`=t.ID_TARA)  
GROUP BY Inregistrare) AS suma_tara  
LEFT JOIN inregistrare AS i ON (suma_tara.`Inregistrare`=i.ID_INREGISTRARE)
```

Rezultat tipic ~0.0498s (0.0336s)