

Systeme d'information et base de données

CM4 : Maintenance et surveillance

Mickaël Martin Nevot

V2.6.0



Cette œuvre est mise à disposition selon les termes de la
[licence Creative Commons Attribution – Pas d'Utilisation Commerciale – Partage à l'Identique](#)
[3.0 non transposé.](#)

Systeme d'information et base de données

- I. Présentation du cours
- II. SI
- III. SGBD
- IV. Design
- V. Droits
- VI. Maintenance
- VII. Réplication/Sécurité
- VIII. Optimisation

Problèmes avec tables MyISAM

- Le format des tables MyISAM est très sûr
- Cas (malchanceux) de tables corrompues :
 - Le démon `mysqld` s'arrête pendant une écriture
 - Arrêt brutal de l'ordinateur
 - Erreur matérielle (contrôleur de disque, etc.)
 - Utilisation d'un programme externe (`myisamchk`) sur une table qui est en train d'être modifiée par le serveur en même temps
- Le problème vient le plus souvent des **fichiers d'index** et non pas des données

Erreur symptomatique lors de l'accès à une table :

```
Incorrect key file for table: '...'. Try to repair it
```

Outils de maintenance MyISAM

Commandes MySQL

• Sous MySQL :

```
mysql> CHECK TABLE ... ;
mysql> REPAIR TABLE ... ;
mysql> OPTIMIZE TABLE ... ;
mysql> ANALYZE TABLE ... ;
```

CHECK TABLE Fonctionne aussi avec les tables InnoDB ;
OPTIMIZE TABLE et ANALYZE TABLE avec BDB

Utilitaire myisamchk

• Hors MySQL :

```
shell> myisamchk options tbl_name
```

Equivalent ?

```
Console - Cooperative Linux - [To Exit, Press Window+Alt Keys]
-rwxrwxr-x 1 root mysql 10466 2009-01-10 07:28 user.frm
-rw-rw---- 1 mysql mysql 8766 2009-02-04 09:48 user_info.frm
-rw-rw---- 1 mysql mysql 0 2009-02-04 09:48 user_info.MYD
-rw-rw---- 1 mysql mysql 1024 2009-02-04 09:48 user_info.MYI
-rwxrwxr-x 1 root mysql 528 2009-01-31 02:29 user.MYD
-rwxrwxr-x 1 root mysql 2048 2009-01-31 03:21 user.MYI
[root@colinux mysql]# myisamchk -c tables_priv.MYI
Checking MyISAM file: tables_priv.MYI
Data records: 0 Deleted blocks: 1
- check file-size
- check record delete-chain
- check key delete-chain
- check index reference
- check data record references index: 1
- check data record references index: 2
[root@colinux mysql]# myisamchk -c user.MYI
Checking MyISAM file: user.MYI
Data records: 5 Deleted blocks: 0
- check file-size
- check record delete-chain
- check key delete-chain
- check index reference
- check data record references index: 1
- check record links
[root@colinux mysql]#
```

Commande myisamchk

- Est utilisé pour obtenir de l'information sur les tables, pour les vérifier, les réparer ou les optimiser
- Sert à l'analyse et la maintenance (réparation) des fichiers d'index (d'extension `.MYI`) et de données (d'extension `.MYD`) :

C'est dans ces fichiers que surviennent la majorité des erreurs

- Hors de contrôle du démon `mysqld`
- Crée une copie du fichier de données ligne par ligne et, en cas de réparation, détruit l'ancien fichier et renomme le nouveau
- Ne fonctionne qu'après avoir arrêté le serveur, en cas de crash, ou après avoir fait un *flush* des tables
- Il est préconisé d'effectuer des sauvegardes des bases de données avant de l'utiliser

Utile : `myisamchk --help`

Préconisation de myisamchk

- Après un crash du démon `mysqld`
- Pour automatiser des *scripts*
- Pour traiter un ensemble de table « en même temps »
- Quand on ne dispose pas des droits MySQL nécessaires

Attention : il faut rappeler qu'au lancement de `mysqld`, les tables (MyISAM ou autres) sont chargées en mémoire pour des accès plus rapides. Si on modifie des tables sur disque, avec un utilitaire autre que MySQL en particulier comme `myisamchk`, l'image en mémoire ne correspond plus. Il faut alors indiquer au serveur de mettre en correspondance les données sur le disque et celles qui sont en mémoire : il est donc nécessaire de recharger les tables en mémoire après modification des fichiers sur le disque

Vérification d'une table

- CHECK TABLE :

Vérification de la structure des tables, calcul et recherche des index, tests de corruption sur les tables MyISAM :

Syntaxe :

```
CHECK TABLE tbl_name {QUICK | FAST | MEDIUM | EXTENDED | CHANGED}
```

- Par défaut : MEDIUM

- Équivalent à :

Syntaxe :

```
myisamchk {--check, -c | --fast, -F | --medium-check, -m | --extend-check, -e |  
--check-only-changed, -C ... } tbl_name
```

- Par défaut : --check, -c

CHECK TABLE / myisamchk

- QUICK / `--check`, `-c` :
 - Vérifie simplement une table. Si une erreur est trouvée, la table est marquée comme `Corrupted` et ne peut plus être utilisée jusqu'à ce qu'elle soit réparée (trouve 99.99 % des erreurs)
- FAST / `--fast`, `-F` :
 - Ne vérifie que les tables qui n'ont pas été correctement fermées
- MEDIUM / `--medium-check`, `-m` :
 - Calcule le *checksum* de chaque ligne pour vérifier les liens vers les index en le comparant avec le *checksum* calculé des clefs (trouve 99.999 % des erreurs)

CHECK TABLE / myisamchk

- EXTENDED / `--extend-check`, `-e` :
 - Vérifie de manière complète et approfondie ; opération d'autant plus longue que la table a beaucoup d'index, à n'utiliser qu'après des vérifications plus rapides
- CHANGED / `--check-only-changed`, `-C` :
 - Vérifie seulement les tables qui ont changées depuis la dernière vérification

Les options `--fast`, `-F` / `--check-only-changed`, `-C` sont fréquemment utilisées dans des scripts pour vérifier régulièrement l'état des tables

Options spécifiques à myisamchk

- `--information, -i` :
 - Affiche des informations statistiques sur la table
- `--update-state, -U` :
 - Stocke les informations dans le fichier d'index pour indiquer quand la table a été vérifiée ; intéressant pour tirer bénéfice de l'option `--check-only-changed` (il ne faut pas utiliser cette option si le démon `mysqld` utilise la table sans l'option `--skip-externallocking`)

L'utilisation de l'option `--skip-external-locking` de MySQL empêche les verrous externes cependant certains systèmes n'en n'utilisent pas

Réparation d'une table

- REPAIR TABLE :

Réparation d'une table potentiellement corrompue

Syntaxe :

```
REPAIR [LOCAL | NO_WRITE_TO_BINLOG] TABLE tbl_name... [QUICK] [EXTENDED] [USE_FRM]
```

- Équivalent à :

Syntaxe :

```
myisamchk [--recover, -r | --safe_recover, -o] [--quick] ... tbl_name
```



REPAIR TABLE / myisamchk

- QUICK / `--recover --quick, -r --quick` :
 - Répare uniquement l'arbre des index
- `--recover, -r` :
 - Répare une table endommagée et défragmente les tables ; cette option convient à presque tous les cas d'erreur (si la machine a beaucoup de mémoire, on peut augmenter la valeur du paramètre `sort_buffer_size`) :

```
shell> /opt/mysql-4.1/bin/mysqladmin -u root -p variables | grep sort_buffer
```
- `--force, -f` :
 - Réparation automatique des tables. Le type de réparation est le même qu'avec `--recover, -r`

Les options `--recover, -r` et `-force, -f` n'existent pas avec REPAIR TABLE

REPAIR TABLE / myisamchk

- EXTENDED / `--safe-recover`, `-o` :
 - Recalcule et met à jour dans l'ordre les index ligne par ligne ; méthode plus lente qu'avec `-r` et qui utilise plus de place disque : à n'utiliser qu'en cas d'échec des solutions précédentes
- USE_FRM :
 - Recrée le fichier d'index à partir du fichier de définition de la table (d'extension `.frm`)

Le type de réparation USE_FRM ne peut pas être réalisé avec myisamchk

Utilisations de myisamchk

- Toutes les tables du dossier courant :

```
shell> myisamchk *.MYI
```

- Dans un autre dossier :

```
shell> myisamchk /path/to/database_dir/*.MYI
```

- Toutes les tables de toutes les bases de données :

```
shell> myisamchk /path/to/datadir/*/*.MYI
```

- Méthode recommandée :

```
shell> myisamchk --silent --fast /path/to/datadir/*/*.MYI ;
```

L'option `--silent`, `-s` est le mode silencieux : pour afficher uniquement les erreurs ; on peut utiliser `-ss` pour que `myisamchk` soit vraiment très silencieux !

`mysqlcheck` est une fonction similaire à `myisamchk` mais doit être utilisé avec le serveur en marche (à l'inverse de `myisamchk`)

Options myisamchk (analyse)

- `--analyse, -a` :
 - Analyse la distribution des clefs et améliore les performances de jointure en permettant à l'optimiseur de mieux choisir l'ordre dans lequel sera faite la jointure des tables et quelle clef sera utilisée :
 - Équivalent à :

```
mysql> SHOW KEYS FROM tbl_name;
```
- `--description, -d` :
 - Donne les principales informations sur la table
- `--sort-index, -s` :
 - Tri l'arbre des index (optimise les recherches par clef)

Options myisamchk (autres)

- `--backup, -B` :
 - Fait une sauvegarde du fichier de données
- `--quick` :
 - Pas de création de fichier de données temporaire, le fichier est considéré comme correct et on ne recrée que le fichier des index
- `--quick --quick` (deux fois !) :
 - Tentative de résolution des erreurs en modifiant le fichier de données

Bonnes utilisations de myisamchk

- **Inspecter :**

```
shell> myisamchk -d -c -i -s
```

À exécuter quotidiennement pour conserver le compte rendu et le consulter fréquemment

- **Réparer :**

```
shell> myisamchk -a -r -e
```

En cas de problème de crash (disque, serveur, machine, etc.) pour analyser les tables

- **Optimiser, analyser :**

```
shell> myisamchk -a -d
```

En obtenant une erreur comme celle-ci :

```
myisamchk: warning: 1 clients is using or hasn't closed the table properly
```

Cela signifie qu'il y a une tentative de vérification d'une table qui a été modifiée par un autre programme (comme le démon `mysqld`) qui n'a pas encore refermé le fichier de table, ou que le fichier n'a pas été correctement refermé

Surveillance du serveur

- Pour chaque connexion client, `mysqld` crée un processus fils (un ***thread***) qui prend en charge les requêtes de ce client : il y a donc autant de *threads* que de connexions
- Chaque *thread* accède aux données des tables en fonction des requêtes de son client
- Commandes pour surveiller l'état du serveur :
 - `mysqladmin`
 - `kill`
 - `show`



Recharger les tables en mémoire

- Écrire les tables sur disque :

```
shell> mysqladmin -u user -ppassword flush-tables
```

- Ou :

```
mysql> FLUSH TABLES
```

Détruit aussi le cache
des résultats issus des
requêtes SQL

- Écrire les fichiers de journalisation :

```
shell> mysqladmin -u user -ppassword flush-logs
```

- Ou :

```
mysql> FLUSH LOGS;
```

- Écrire tables et fichiers de journalisation sur disque :

```
shell> mysqladmin -u user -ppassword refresh
```

- Remise à zéro de la plupart des variables de MySQL :

```
shell> mysqladmin -u user -ppassword flush-status
```

- Ou :

```
mysql> FLUSH STATUS;
```

FLUSH ... nécessite le privilège MySQL RELOAD

Recharger les tables en mémoire

- Recharger la table des droits :

```
shell> mysqladmin -u user -ppassword flush-privileges
```

- Ou :

```
shell> mysqladmin -u user -ppassword reload
```

- Ou :

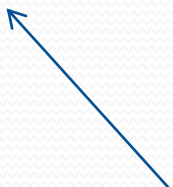
```
mysql> FLUSH PRIVILEGES;
```

- Vide le cache interne des DNS :

```
shell> mysqladmin -u user -ppassword flush-hosts
```

- Ou :

```
mysql> FLUSH HOSTS;
```



À utiliser dans le cas de certaines erreurs `Host ... is blocked`,
ou quand on dépasse le nombre d'erreurs de connexion défini par
`max_connect_errors`

Commande MySQL : RESET

- Annule l'état de plusieurs opérations du serveur MySQL (sorte de FLUSH général vidant la mémoire cache de toutes les structures utilisées par le serveur sur le disque)
- RESET MASTER :
Détruit les fichiers de *log* binaires et en crée un nouveau
- RESET QUERY CACHE :
Détruit les résultats SQL du cache correspondant

RESET nécessite le privilège MySQL RELOAD

RESET SLAVE en cas de réplication de la base de données

Gestion des Threads

- **Surveillance des *threads* :**

```
shell> mysqladmin -u user -ppassword processlist
```

- Ou :

```
mysql> SHOW PROCESSLIST;
```

- **Tuer un *thread* :**

```
shell> mysqladmin -u user -ppassword kill process_id
```

- Ou :

```
mysql> KILL process_id;
```

Id	User	Host	db	Command	Time	State	Info
2	root	localhost:2147	NULL	Query	0	NULL	show processlist

Statistiques du serveur MySQL

- Utilisation intéressante de `mysqladmin` :

- **Variables** (toutes celles d'un serveur) :

```
shell> mysqladmin -u user -ppassword variables
```

- Ou :

```
mysql> SHOW VARIABLES;
```

- **Statuts** :

```
shell> mysqladmin -u user -ppassword status
```

- Ou (beaucoup plus de détails) :

```
shell> mysqladmin -u user -ppassword extended-status
```

- Équivalent à :

```
mysql> SHOW STATUS;
```



Surveillance : divers

- SHOW DATABASES : montre toutes les bases de données

- Équivalent à :

```
shell> mysqlshow -u user -ppassword
```

- SHOW TABLES : montre les tables d'une base

- Équivalent à :

```
shell> mysqlshow -u user -ppassword db_name
```

- SHOW COLUMNS FROM :

- SHOW FIELDS FROM :

Montre les colonnes d'une table

Identique à : DESCRIBE table

- Équivalent à :

```
shell> mysqlshow -u user -ppassword db_name tbl_name
```

Surveillance : divers

- SHOW PRIVILEGES : montre les privilèges systèmes
- SHOW INDEX :
 - SHOW KEYS :

Montre les informations sur les index des tables
 - Équivalent à :

```
shell> mysqlshow -u user -ppassword -k db_name tbl_name
```
- SHOW GRANTS FOR : montre les privilège d'un utilisateur
 - Équivalent à :

```
shell> mysqlaccess -U user -Ppassword -t -d tbl_name -u grantuser
```



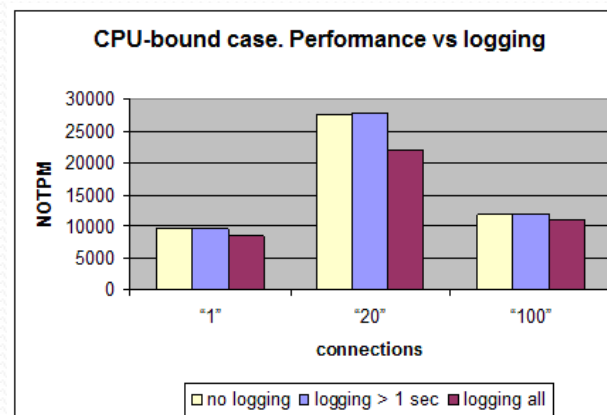
Options mysqld de journalisation

- `--log-bin` : ← Anciennement `--log-update`, en binaire depuis la version 4.1.3
 - Toutes les modifications des bases de données
- `--log-error` :
 - Problèmes au lancement / à l'exécution (fichier `.err`)
- `--log-isam` :
 - Changements effectués sur les tables ISAM (et MyISAM)
- `--log` :
 - Connexions des clients et des exécutions de requêtes SQL
- `--log-slow-queries` :
 - Requêtes qui prennent plus de `long_query_time` secondes

`mysqlbinlog` : commande pour examiner le log binaire

Fichiers de journalisation

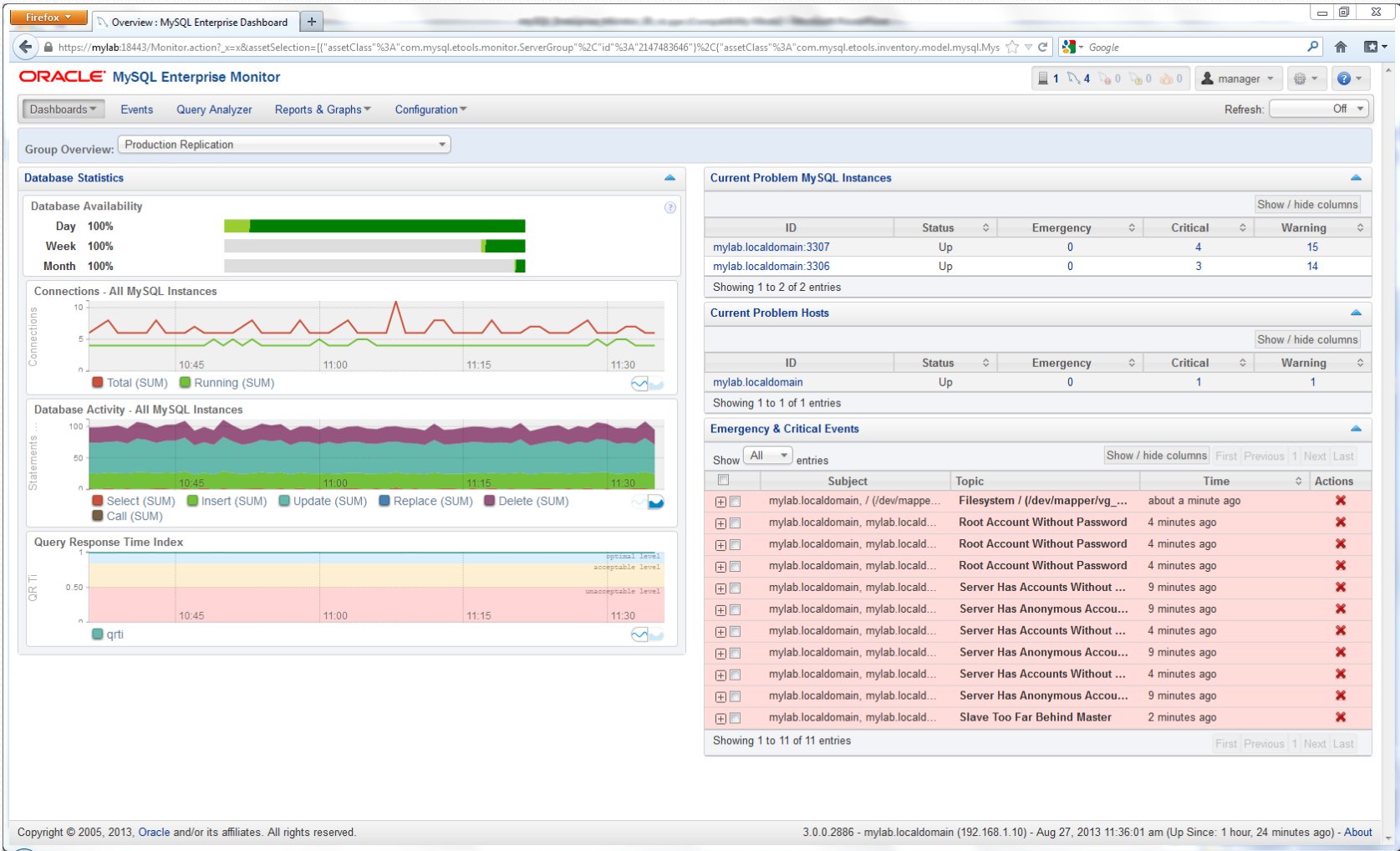
- Par défaut, tous les fichiers de *logs* sont créés dans le répertoire donné par `--datadir (/var/lib/mysql)` :
 - On peut spécifier un nom de fichier après chaque option de journalisation au lancement de `mysqld`
- Par défaut, le nom de chaque fichier est le nom de la machine où se lance `mysqld` :
 - Par défaut stocké dans `/opt/mysql/var/localhost.log`



MySQL Enterprise Monitor

- **Vue consolidée** sur la santé des bases de données
- **Surveillance de plus de 600 variables**
- **Surveillance des sessions, connexions, latence de réplication** (plus de 30 graphiques)
- **Visibilité immédiate des topologies de réplication**
- **Informations temps réel** de la performance maître/esclave
- **Options de personnalisation**
- **Notifications** grâce aux alertes de seuil
- **Fonctions d'authentification unique** (LDAP)
- **Résolution accélérée des problèmes** avec **données de diagnostic** transmises au support technique MySQL

MySQL Enterprise Monitor



Liens

- Documents électroniques :
 - <http://dev.mysql.com/doc/refman/5.5/en>
- Documents classiques :
 - Cours :
 - Maurice Libes. *Administration et exploitation du SGBDR MySQL.*
 - Cyril Gruau. *Conception d'une base de données.*
 - Jean-Marc Petit. *Administration des bases de données.*

Crédits

Auteur

Mickaël Martin Nevot

mmartin.nevot@gmail.com



Carte de visite électronique

Relecteurs

Cours en ligne sur : www.mickael-martin-nevot.com

