

Installation et Mise en œuvre de MySQL



Olivier Mondet
<http://unidentified-one.net>

A. Pourquoi choisir MySQL ?

A.1. MySQL, très controversé

La base de données MySQL est à la fois une des bases de données très utilisées au monde (essentiellement grâce à ses performances sur le Web) mais également un des produits les plus controversés.

On a longtemps reproché à MySQL une syntaxe propre, correspondant de façon approximative aux standards imposés par l'ANSI puis par l'ISO en matière de syntaxe du langage SQL. Un grand nombre d'ordres SQL ne sont donc pas valide avec MySQL. La gestion des utilisateurs très simple manque souvent de finesse pour pouvoir s'adapter à toutes les utilisations.

Les bases de données MySQL ne respectaient ni les contraintes d'intégrité (comme les contraintes de clé primaire) ni les contraintes d'intégrité référentielle (contrainte de clé étrangère) jusqu'à 2005 !

MySQL est actuellement dans sa (toute récente) version 5, réponse en quelque sorte aux critiques émises à l'encontre de la base de données.

- Intégration (depuis la version 4.2) des contraintes d'intégrité référentielles,
- Intégration des procédures stockées (automatisation de tâches),
- Intégration des Triggers (les déclencheurs, qui sont des scripts événementiels suivant les ordres SQL),
- Intégration des vues,
- Meilleure intégration du langage SQL standard.

A.2. Quels sont les concurrents ?

Sur les marchés destinés aux entreprises de taille moyenne, on trouve les bases de données suivantes : MySQL, PostgreSQL, Ingres, FireBird, Oracle (version limitée), IBM DB2 Express, SQLite.

Sur les gros marchés, seuls trois géants sont présents : Oracle, IBM DB2, SQL server.

Oracle est le géant de la BDD avec 41% du marché mondial (source du 10/03/2005) selon Marketing Stratégique et O1 Informatique.

Le marché des bases de données, au niveau de la France, selon l'Expansion (8/11/2005) : «Microsoft est très présent dans les PME/PMI mais peu dans les grands comptes (15,5%) où règnent Oracle (35%) et IBM (32%)».

Face à cette domination des grandes entreprises, des solutions gratuites, libres, ont fait le jour depuis un bon nombre d'années : parmi ces solutions on trouve MySQL qui s'est rapidement imposé comme un concurrent solide dans les bases de données adossées aux services Web. La raisons principale est la rapidité d'exécution de son moteur, et une certaine souplesse d'utilisation. On note cependant des

concurrents comme PostgreSQL qui sont plus solides encore, qui respectent mieux les standards, mais qui sont handicapés souvent par des temps d'exécution moins bons.

MySQL est donc une base de données d'un abord plus simple (notamment grâce au langage PHP), facilement interfaçable avec un langage (encore une fois, merci PHP), et Open source pour l'usage à titre éducatif ou personnel. Attention (de plus en plus) souvent les solutions à base de MySQL sont payantes pour les entreprises.

A.3. Sous quelle forme utilise t-on MySQL ?

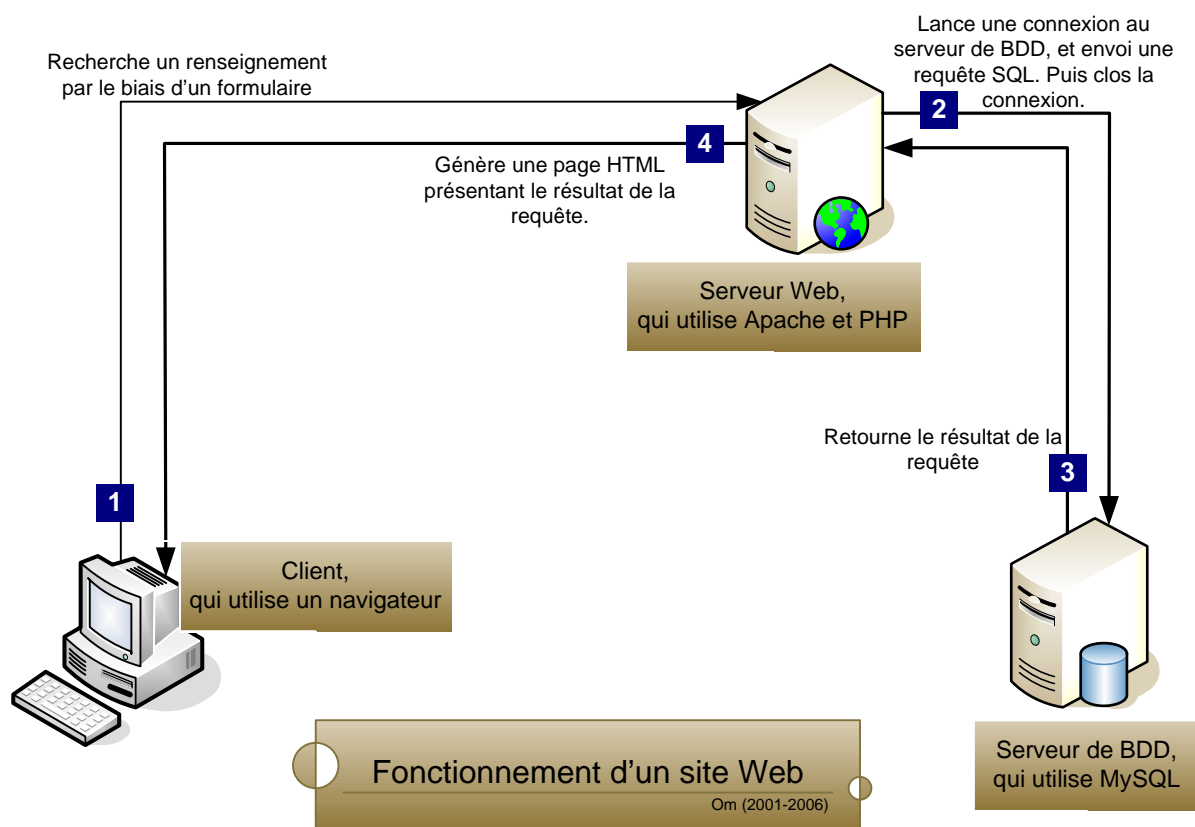
MySQL peut donc être livré seul ou dans un package de logiciels via le package EasyPHP ou le package WAMP :

L'utilisation du package EasyPHP (ou le package WAMP) est une solution de simplification dans le cadre de l'apprentissage ou dans une structure de petite taille.

Le package intègre plusieurs logiciels qui sont liés et généralement utilisés ensembles.

- Un serveur Web (http) : Apache,
- Un serveur de base de données : MySQL,
- Un langage de script : PHP.

Ce sont des logiciels différents, qui sont de familles différentes mais ils répondent à des besoins particuliers et complémentaires en matière de sites Web.



Dans ce schéma il est primordial de noter qu'EasyPHP intègre des logiciels différents, qui ont les rôles de :

- serveur Web (ou http) pour pouvoir servir les pages demandées par le client et orienter les pages dont l'extension est php vers l'interpréteur PHP ;
- interpréteur PHP (le PHP est un langage de script qui possède de nombreux jeux d'instructions orientés Web et BDD) qui se charge de générer les pages HTML à destination des clients, et de se connecter à la BDD MySQL, de lui transmettre les requêtes ;
- serveur de BDD pour le stockage des données, les contrôles d'intégrité.

A.4. WAMP ou LAMP ?

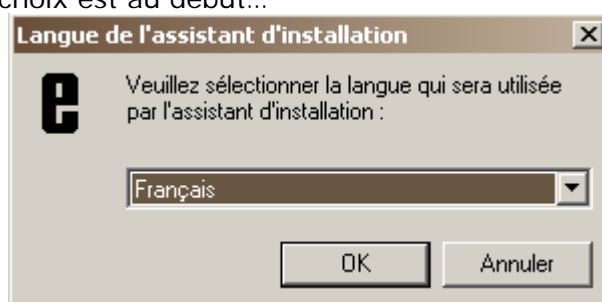
Sous Linux, on parle de LAMP : Linux Apache MySQL PHP ;
Sous Windows on parle de WAMP : Windows Apache MySQL PHP.
(Attention : WAMP est aussi le nom d'un package concurrent à EasyPHP).

EasyPHP conseille sa solution Linux si on veut l'utiliser en production. La solution Windows entraînera un grand nombre de failles de sécurité qui demandent une certaine expertise pour les régler.

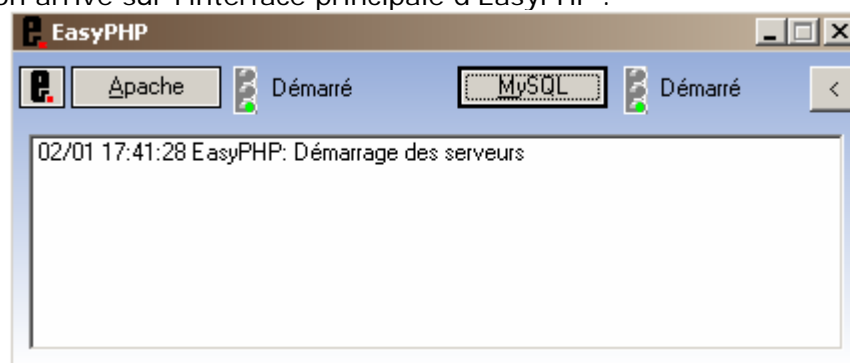
B. Installation de MySQL

B.1. En utilisant le package EasyPHP

Simple d'installation, le seul choix est au début...



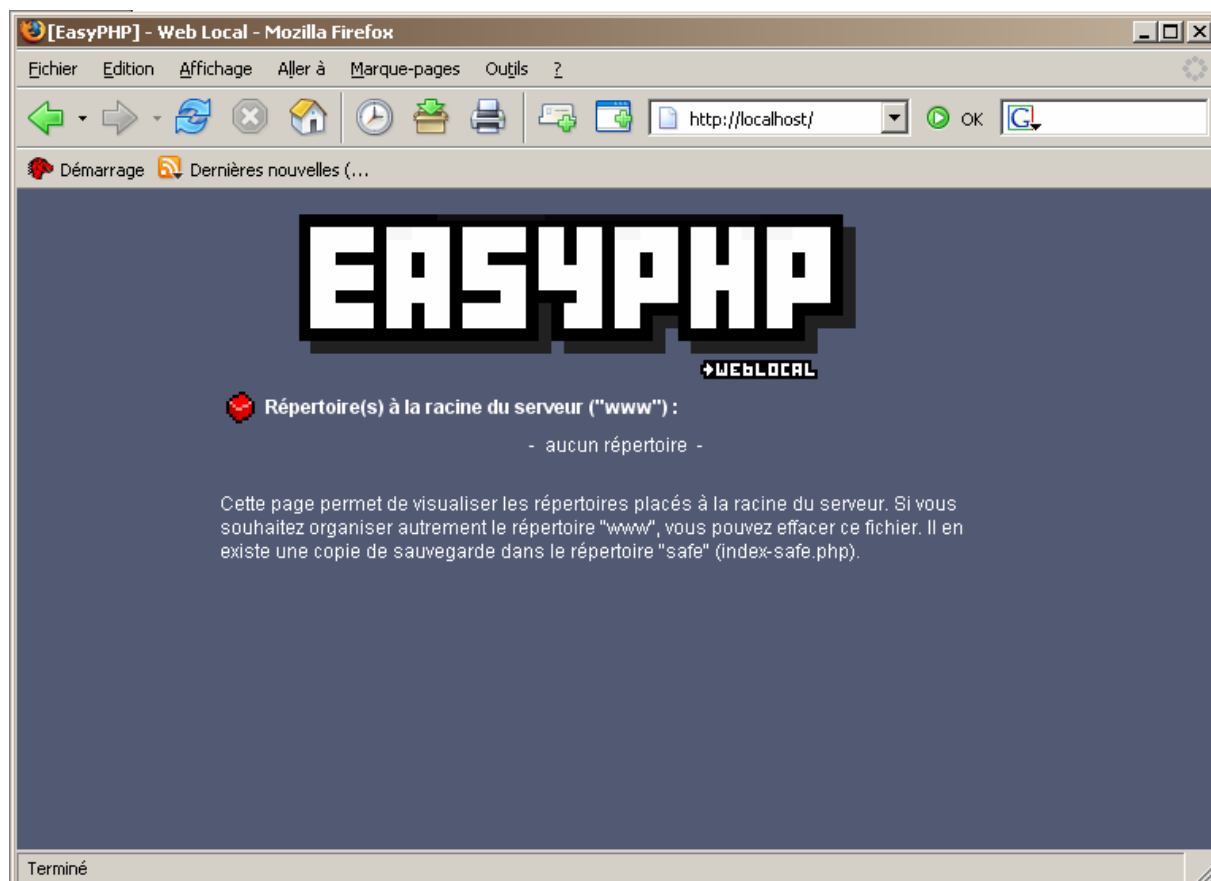
Après l'installation on arrive sur l'interface principale d'EasyPHP :



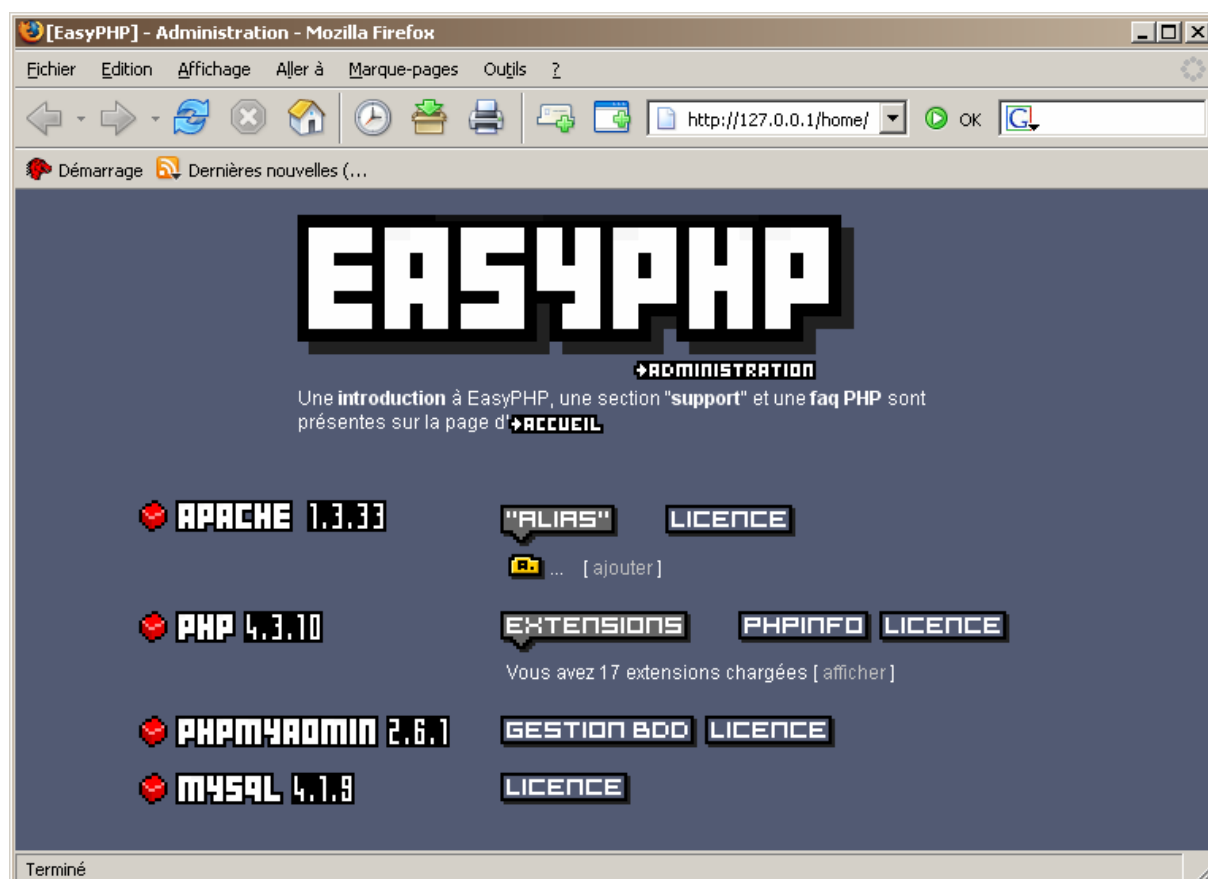
La vue présente les deux serveurs de notre solution lorsqu'ils sont démarrés. Le langage PHP est également installé, mais il n'apparaît pas. Il faut développer des pages en PHP (des pages de test) pour voir le bon fonctionnement du langage.

EasyPHP suppose que vous êtes en cours de développement d'un site Web et il dispose d'un répertoire de base où stocker les pages HTML/PHP...

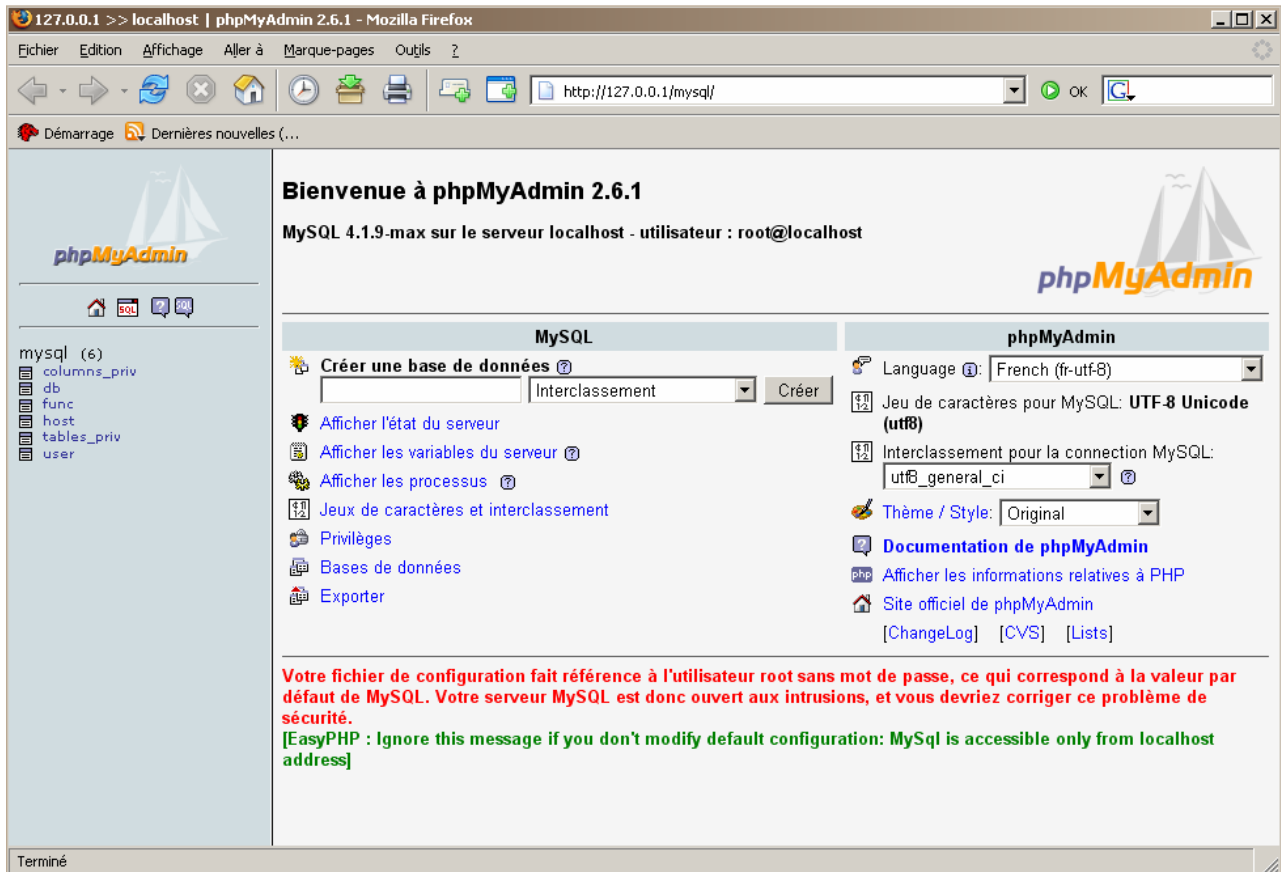
Le site Web en cours de création peut être vu en utilisant le navigateur avec l'adresse de boucle locale.



Une interface en PHP permet de gérer les serveurs, notamment le serveur MySQL qui nécessite plus de manipulations. Ne serait-ce pour entrer les données.



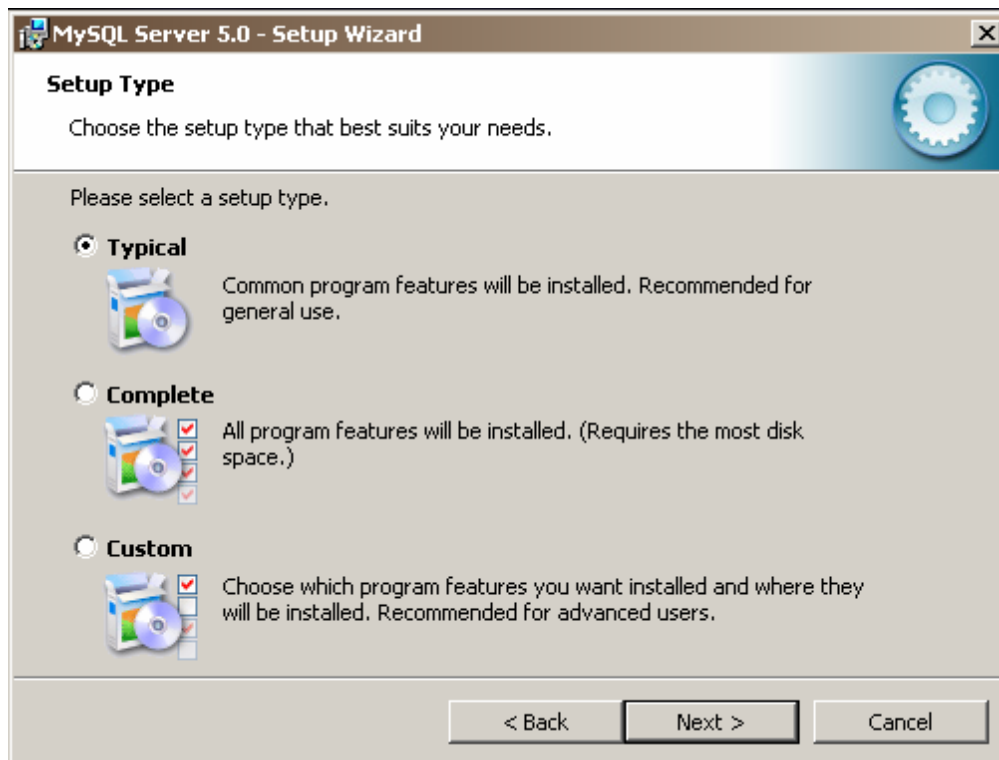
Vue interface d'administration de la base de données :



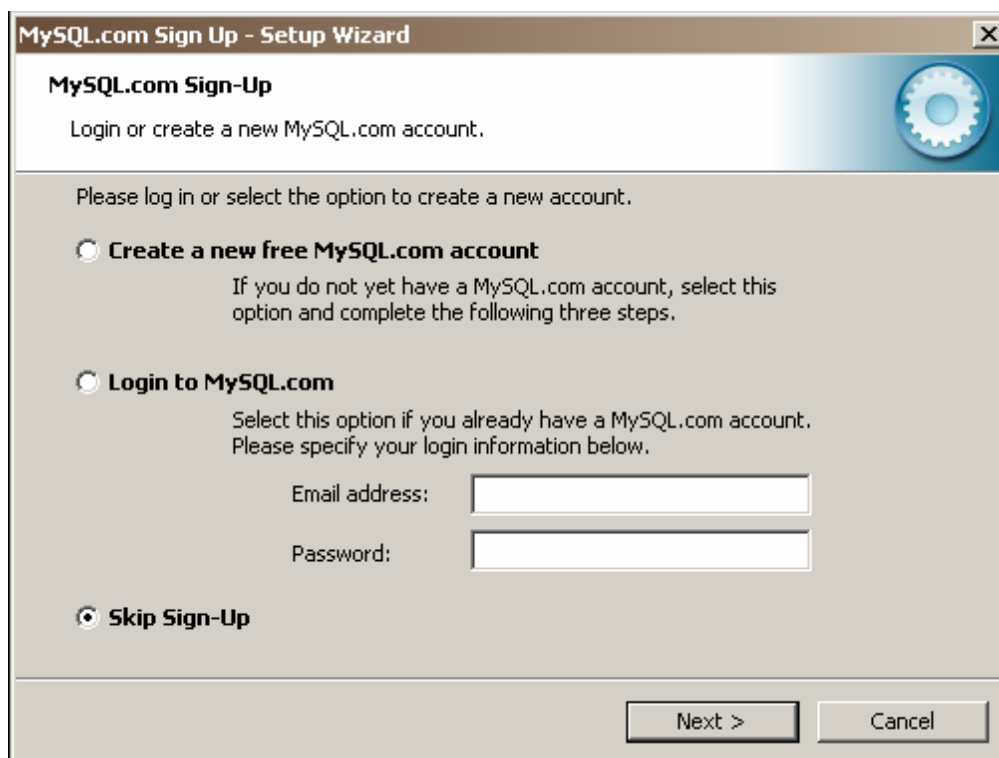
B.2. Installation de MySQL seul

On peut télécharger sur le site de MySQL, la version 5 du logiciel.

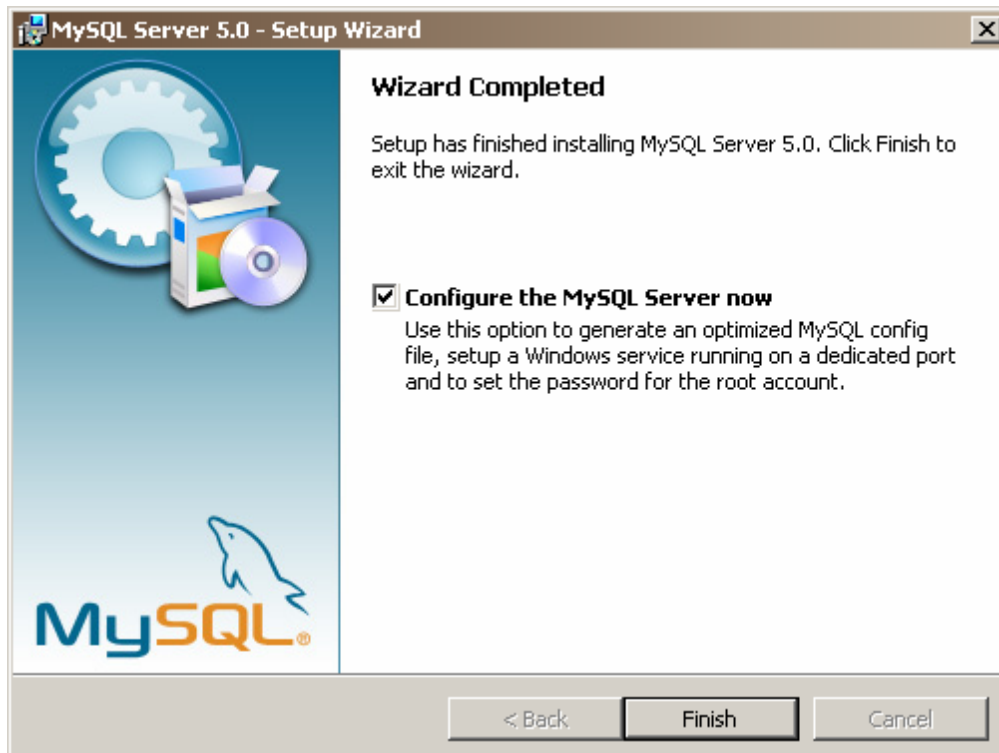




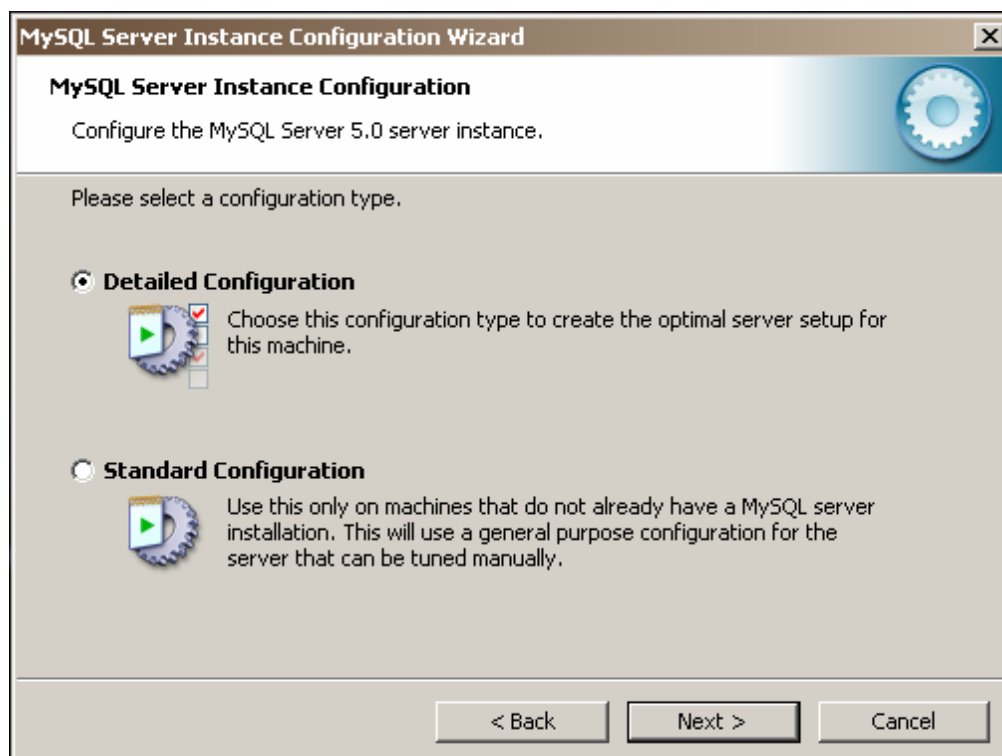
L'enregistrement auprès de MySQL est facultatif.



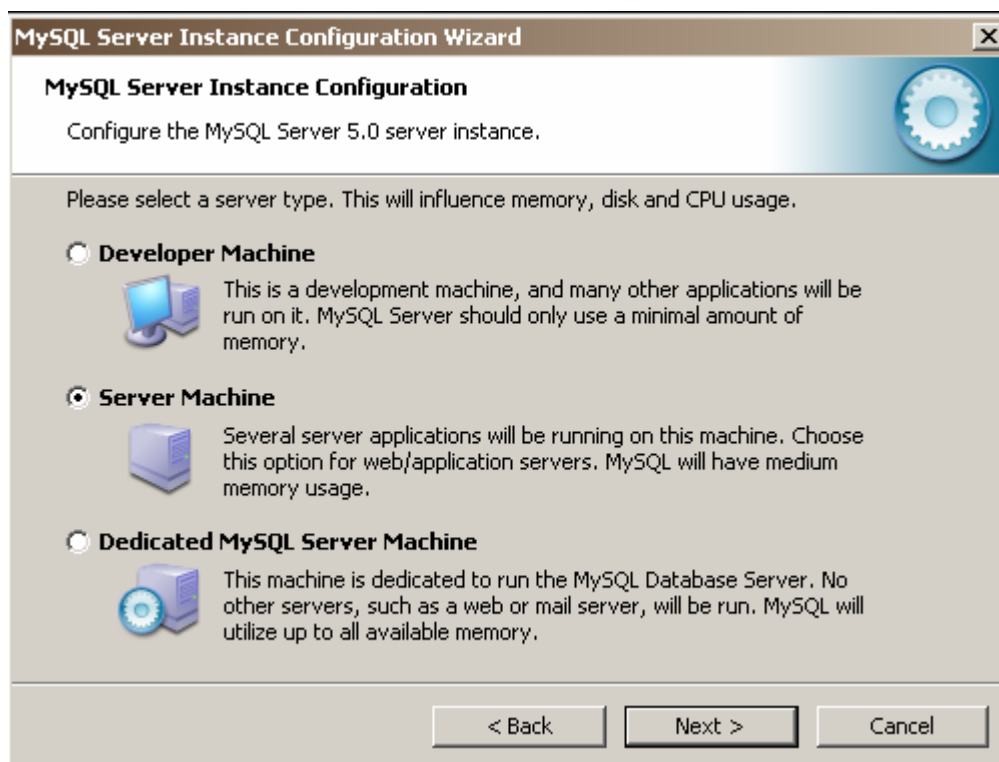
Une fois installé, le passage à la configuration de MySQL est automatique :



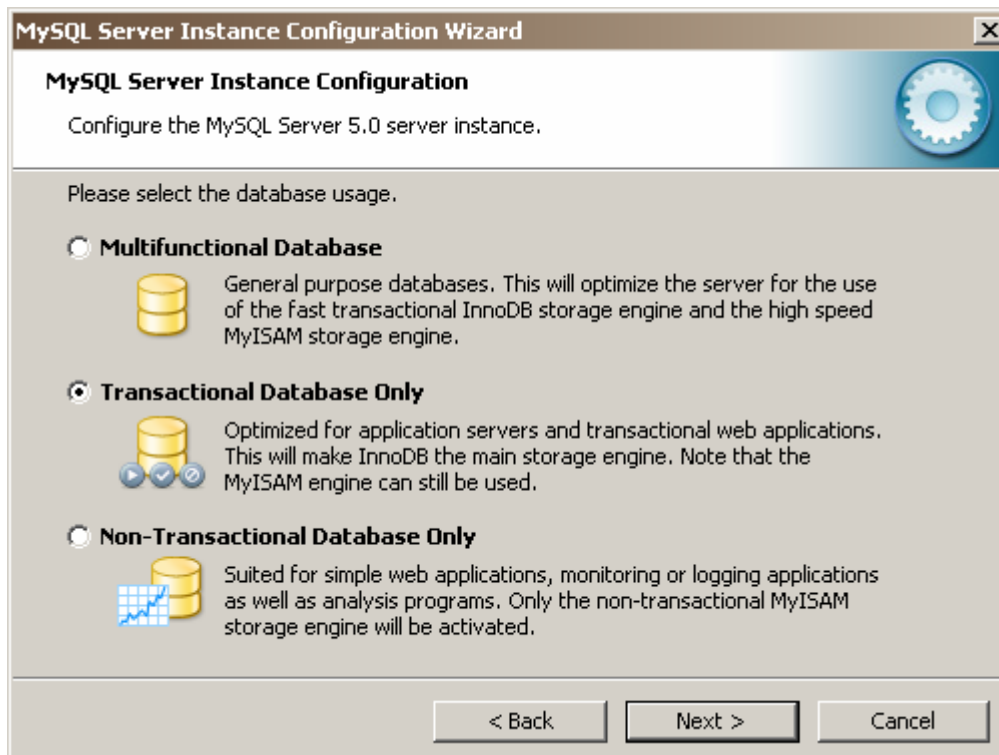
Il vaut mieux choisir soi-même les paramètres de la configuration pour bien comprendre :



L'installation en mode développeur est nécessaire pour faire des tests et diminuer l'importance de MySQL sur le système. L'installation en mode serveur sera privilégiée si on veut réellement utiliser MySQL en production avec un niveau d'utilisation moyen. L'installation sur un serveur dédié est le dernier choix au cas où l'on ait une utilisation solide de MySQL.



Le choix du type d'instance est très important et détermine le niveau d'intégrité de la base de données.



MySQL jusqu'à sa version 4.2, n'admettait que les types de tables ISAM, puis MyISAM. Ces types de tables bien que très performants (MyISAM est parmi les plus rapides dans les requêtes SELECT) avaient de piètres fonctionnalités et de gros problèmes de récupération des données en cas de crash. Le seul moyen de palier aux inconvénients fut à l'instar des géants comme Oracle ou IBM DB2, de gérer les transactions. C'est-à-dire que les échanges avec la base de données sont répertoriés et

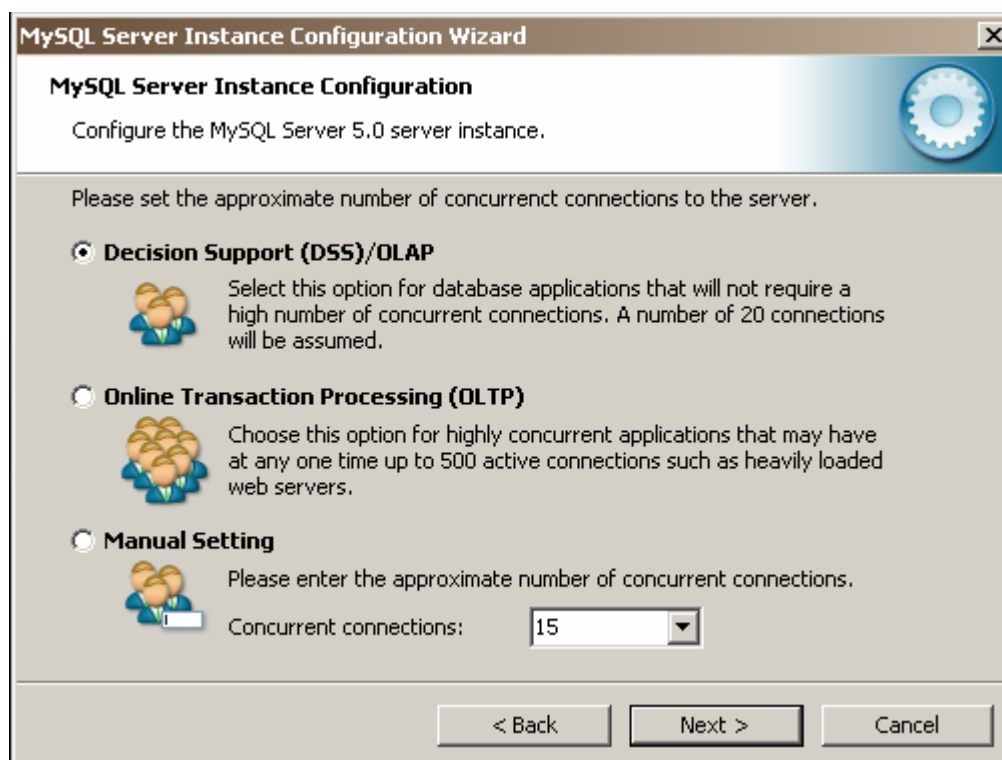
historiés. Les accès aux tables sont mieux g rer surtout quand ils sont nombreux : on parle de tables transactionnelles.

InnoDB fournit   MySQL un gestionnaire de table transactionnelle (compatible ACID), avec validation (commit), annulations (rollback) et capacit s de restauration apr s crash. Les tables InnoDB sont les premi res tables MySQL qui supportent les contraintes de cl s  trang res (FOREIGN KEY).

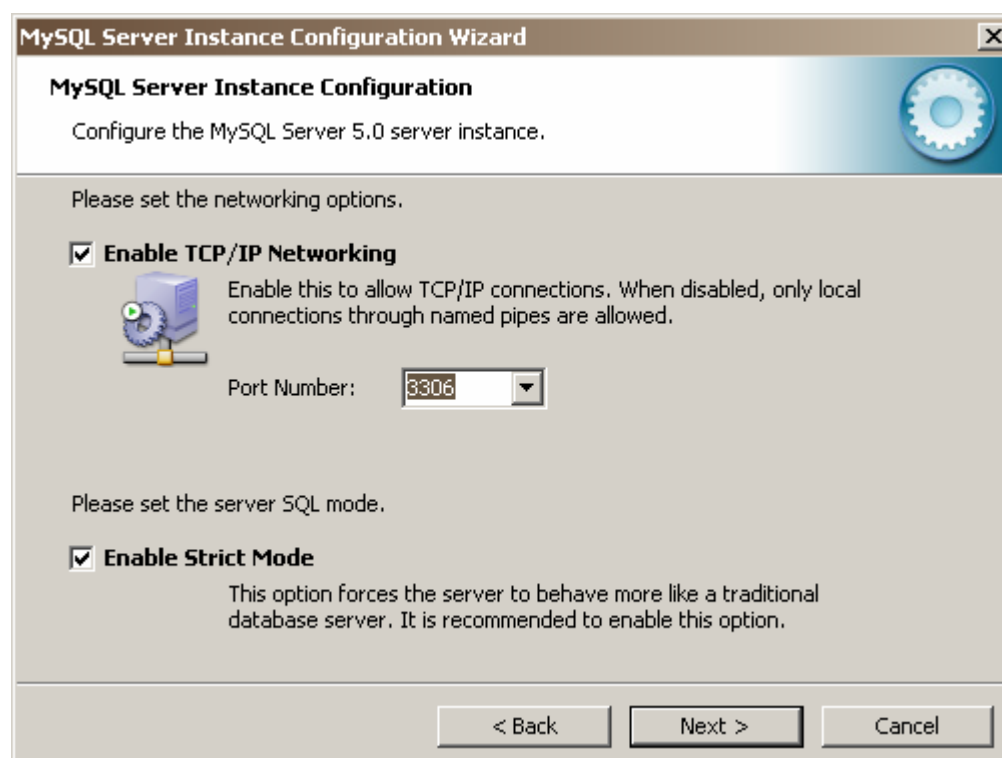
InnoDB a  t  con u pour maximiser les performances lors du traitement de grandes quantit s de donn es. Son efficacit  processeur n'est  gal e par aucun autre moteur de base de donn es.

(Le 13-10-2005 il y a eu une menace de rachat d'InnoDB par Oracle).

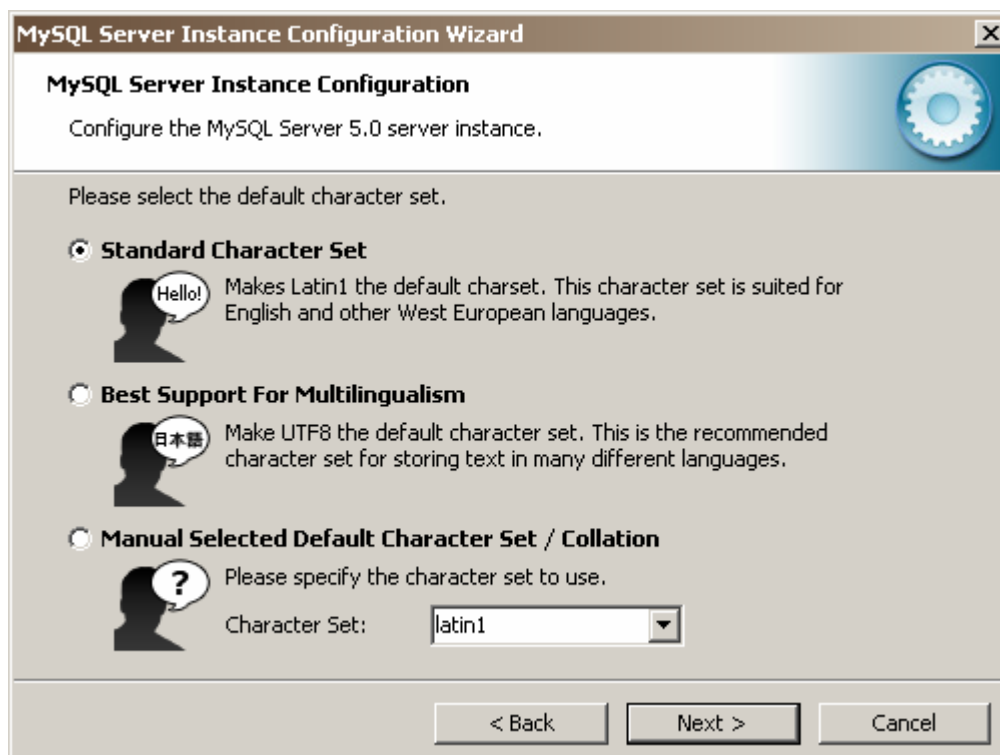
On fixe le nombre de connexions simultan es au serveur (ici une vingtaine).



On autorise les connexions TCP/IP (obligatoire pour travailler en client-serveur).



On fixe le jeu de caractères par défaut (ici Latin 1).



Le paramétrage en tant que service Windows permet un lancement automatique en tâche de fond de MySQL. L'ajout à la commande PATH du chemin vers MySQL permet de pouvoir saisir les commandes MySQL depuis n'importe quel endroit.



Mot de passe de root le DBA de MySQL.



MySQL Server Instance Configuration Wizard

MySQL Server Instance Configuration

Configure the MySQL Server 5.0 server instance.

Please set the security options.

☒ **Modify Security Settings**

 New root password: Enter the root password.

 Confirm: Retype the password.

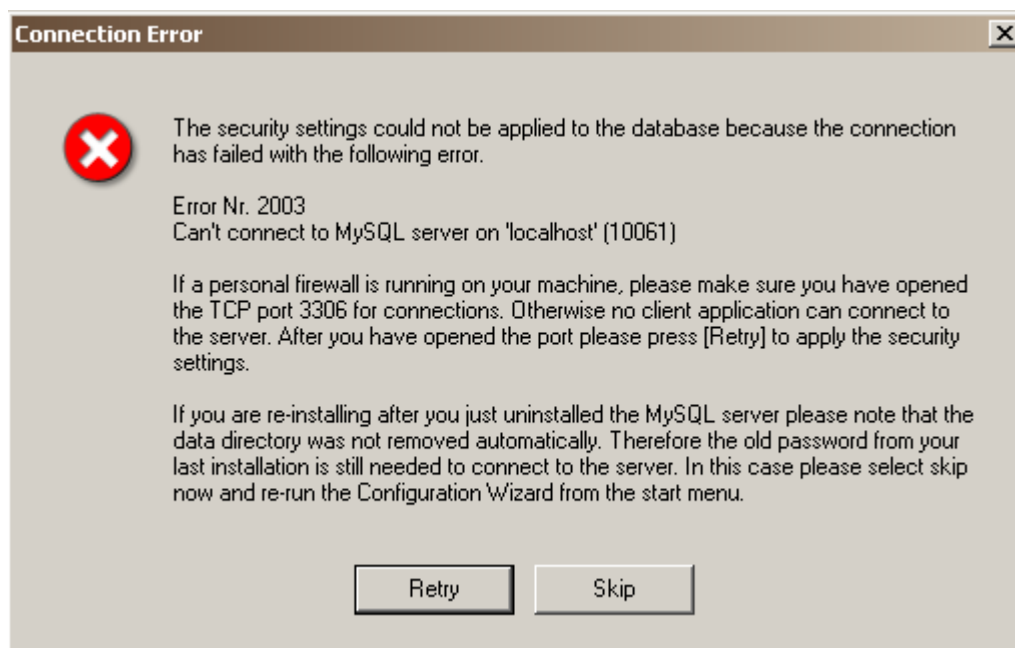
☐ Enable root access from remote machines

☐ **Create An Anonymous Account**


 This option will create an anonymous account on this server. Please note that this can lead to an insecure system.

< Back Next > Cancel

Il se peut que si votre version de Windows a été mise à jour avec le SP2, il y ait un blocage du port TCP/IP de communication du logiciel.



Connection Error

 The security settings could not be applied to the database because the connection has failed with the following error.

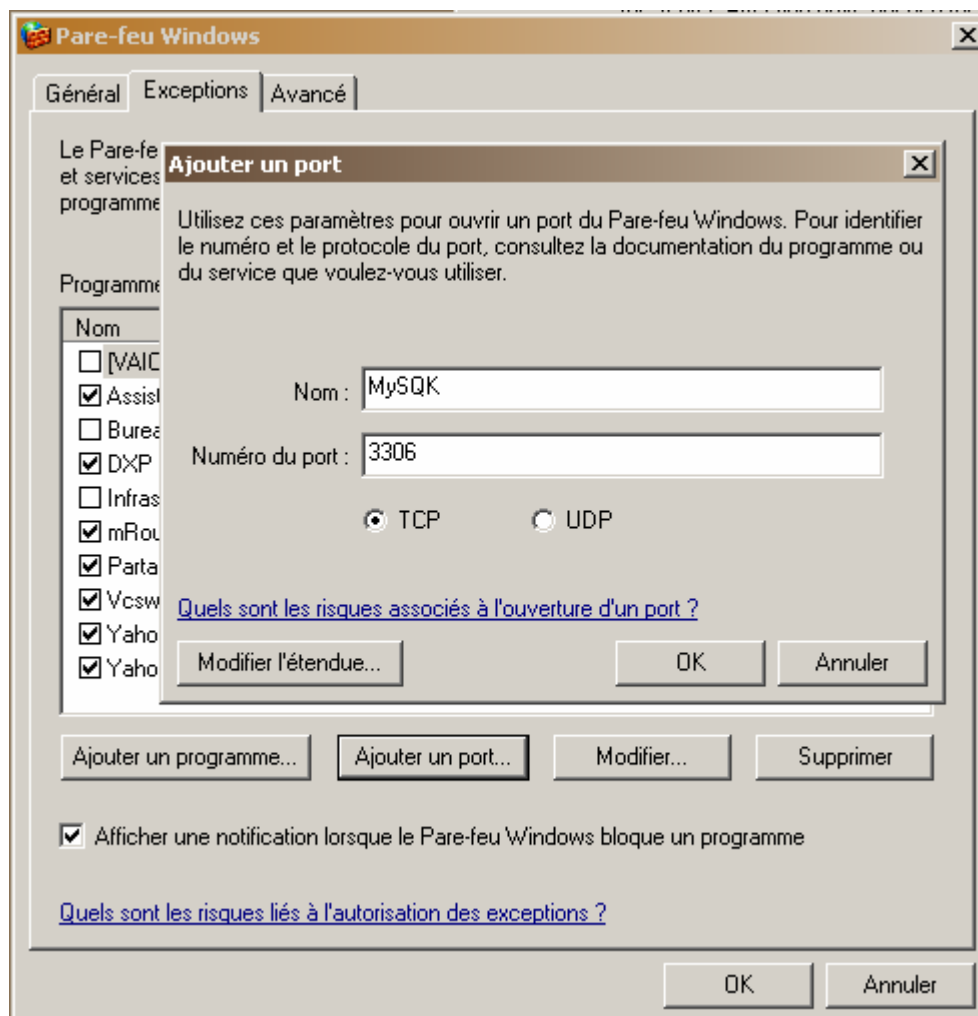
Error Nr. 2003
Can't connect to MySQL server on 'localhost' (10061)

If a personal firewall is running on your machine, please make sure you have opened the TCP port 3306 for connections. Otherwise no client application can connect to the server. After you have opened the port please press [Retry] to apply the security settings.

If you are re-installing after you just uninstalled the MySQL server please note that the data directory was not removed automatically. Therefore the old password from your last installation is still needed to connect to the server. In this case please select skip now and re-run the Configuration Wizard from the start menu.

Retry Skip

Il faut atteindre le pare-feu pour ajouter le port de communication de MySQL.



C. Utiliser MySQL

C.1. Eléments de base de MySQL

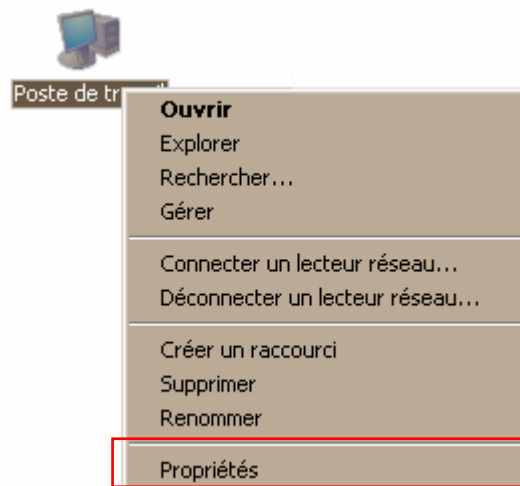
Une fois dans le dossier (bin) de MySQL celui-ci contient divers éléments :

- **mysqld** est le serveur MySQL ;
- **mysqld_safe**, **mysql.server** et **mysqld_multi** sont les scripts de démarrage ;
- **mysql_install_db** initialise le dossier de données et les premières bases ;
- **mysql** est un client en ligne de commande, pour exécuter des commandes SQL ;
- **mysqlcc** (MySQL Control Center) est un client interactif graphique, pour exécuter des commandes SQL, et administrer le serveur ;
- **mysqladmin** est un client d'administration ;
- **mysqlcheck** effectue les opérations de maintenance sur les tables ;
- **mysqldump** et **mysqlhotcopy** font les sauvegardes de bases ;
- **mysqlimport** importe des fichiers de données ;
- **mysqlshow** affiche des informations sur les bases et les tables ;
- **myisamchk** effectue les opérations de maintenance des tables ;
- **myisampack** produit des tables compressées, en lecture seule ;
- **mysqlbinlog** est un outil pour traiter les fichiers de logs binaires ;
- **perror** affiche le message associé à un code d'erreur ;
- **myisamchk** est présenté dans la section Administration de la base.

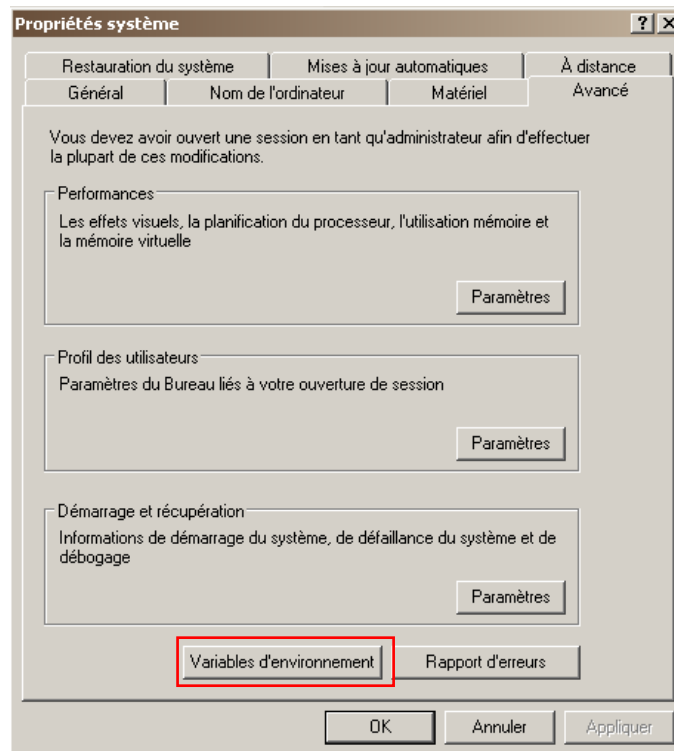
C.2. Insérer un chemin vers le client en ligne mysql

Pour rendre accessible le dossier ...\\mysql\\bin et pouvoir utiliser la ligne de commande quelque soit l'endroit où l'on se trouve, il faut rajouter un chemin dans la variable d'environnement Windows « path ».

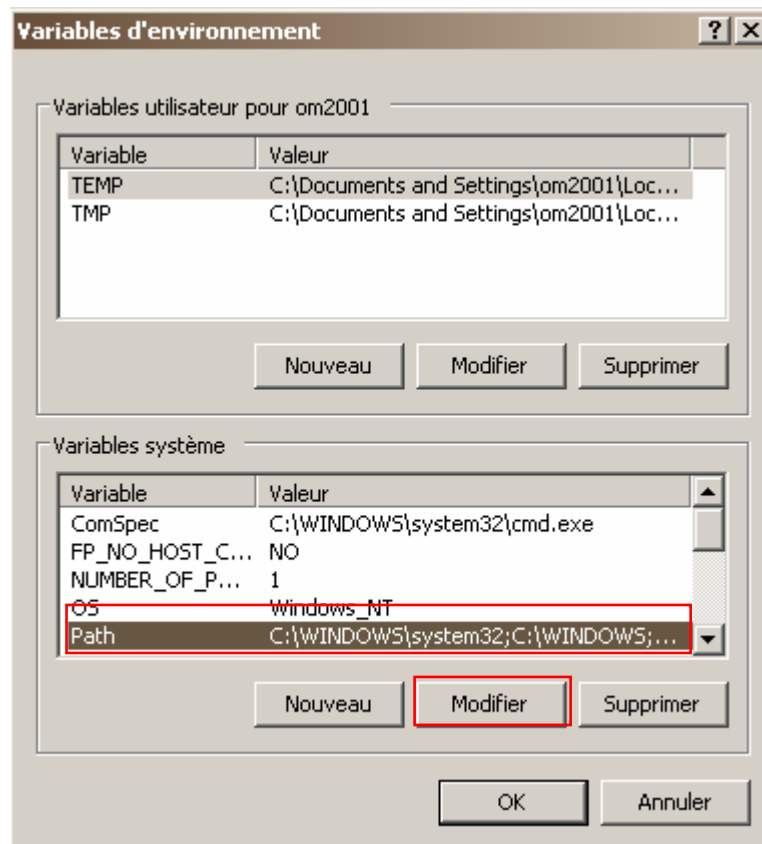
Il faut d'abord cliquer droit sur le poste de travail, et choisir **Propriétés**.



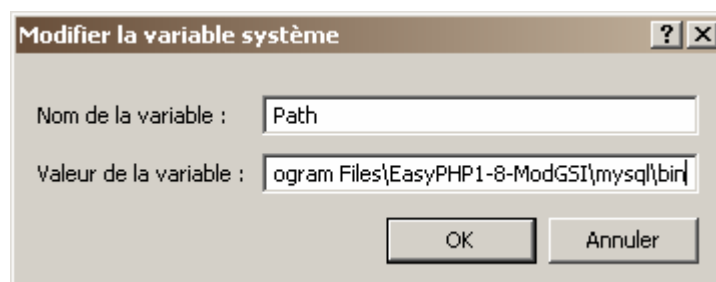
Sélectionner l'onglet **Avancé** puis cliquer sur Variables **d'environnement**.



La variable d'environnement **path** sert à mémoriser pour le système d'exploitation tous les chemins de recherche par défaut. Nous allons ajouter celui où se trouve mysql (dans notre exemple Program Files/Easyphp/mysql/bin).
Cliquer sur **Path** puis sur **Modifier**.



A la fin de la ligne ajouter le texte suivant : ;C:\Program Files\EasyPHP1-8-ModGSI\mysql\bin ou ;C:\Program Files\EasyPHP1-8\mysql\bin (sans omettre le ; qui sépare les différents chemins entre eux. Le chemin doit être celui de votre propre installation et de la distribution que vous choisiez (EasyPHP 1.8 ou EasyPHP 1.8 ModGSI). Valider toutes les boîtes de dialogue ouvertes.



Dès lors vous pourrez lancer mysql où que vous vous trouviez dans l'arborescence de votre système.

C.3. Commandes environnementales

Les commandes environnementales servent au fonctionnement propre de MySQL. On ne doit pas les confondre avec les ordres SQL.

Lancer démon MySQL : `mysqld`

Se connecter au serveur MySQL : `mysql`

Se connecter au serveur MySQL et s'authentifier : `mysql -u <utilisateur> -p`

Se connecter à une base de données : `mysql -u utilisateur -d nombase -p`

Se connecter à un serveur de base de données : `mysql -u utilisateur -h nomhote|ip -p`

Commandes en ligne proposées après demande de l'aide (\? ou help) :

```
\? ou help pour l'aide
\q pour quitter
List of all MySQL commands:
Note that all text commands must be first on line and end with ';'
?          (\?) Synonym for 'help'.
clear      (\c) Clear command.
connect    (\r) Reconnect to the server. Optional arguments are db and host.
delimiter  (\d) Set query delimiter.
ego        (\G) Send command to mysql server, display result vertically.
exit       (\q) Exit mysql. Same as quit.
go         (\g) Send command to mysql server.
help       (\h) Display this help.
notee      (\t) Don't write into outfile.
prompt     (\p) Print current command.
prompt     (\R) Change your mysql prompt.
quit       (\q) Quit mysql.
rehash     (\#) Rebuild completion hash.
source     (\.) Execute a SQL script file. Takes a file name as an argument.
status     (\s) Get status information from the server.
tee        (\T) Set outfile [to_outfile]. Append everything into given outfile.
use        (\u) Use another database. Takes database name as argument.
```

Il existe des ordres SQL qui permettent d'obtenir des informations sur les bases de données :

Voir les différentes de bases de données créées : `SHOW DATABASES;`

Utiliser une base de données : `use <nombdd>`

Voir les tables (une fois dans une base de données) : `SHOW TABLES;`

Voir quelle est la base de données courante : `SELECT DATABASE();`

Voir la structure d'une table : `DESCRIBE <nomtable>;`

Vérifier les droits de l'utilisateur : `SHOW PRIVILEGES;`

Pour exécuter un script SQL : `\.<script.sql>`

Pour charger un fichier dans une table : `LOAD DATA LOCAL INFILE "script.data" INTO TABLE <nomtable>;`

Vous pouvez spécifier la valeur du séparateur de colonnes et le marqueur de fin de lignes explicitement dans la commande `LOAD DATA` si vous le voulez, mais les valeurs par défaut sont la tabulation et le retour à la ligne.

Pour créer une nouvelle base de données : `CREATE DATABASE <nombdd>;`

Pour supprimer une base de données (et son contenu !) : `DROP DATABASE <nombdd>;`

Toutes les autres commandes sont les ordres SQL standards. Attention toutefois à la prise en charge parfois partielle des ordres standards.

C.4. Autres commandes en ligne :

» Faire une sauvegarde d'une base :

```
mysqldump -u <login> -p <motdepasse> <nombase> > <nomfichier.sql>
```

» Faire une sauvegarde de toutes les bases d'un serveur :

```
mysqldump -u <login> -p <motdepasse> --all-databases > <nomfichier.sql>
```



Olivier Mondet
<http://unidentified-one.net>