

# Bases de Datos / Elementos de B.D. 2014

## Introducción a MySQL



Departamento de Ciencias e  
Ingeniería de la Computación  
Universidad Nacional del Sur



## ¿Que es MySQL?

- MySQL es un Sistema De Manejo de Bases de Datos (**SMDB**).
- MySQL es **Open Source**: El código fuente está disponible. Cualquiera puede usarlo y modificarlo.
- MySQL es **gratuito**, bajo las restricciones de la Licencia General Pública **GNU**.
- MySQL se puede bajar de:  
<http://www.mysql.com/downloads/mysql/>  
Opción 1: windows (X86, xx-bit), ZIP archive (unzip in C:\)  
Opción 2: windows (X86, xx-bit), MSI Installer  
(Se recomienda usar la versión distribuida por la cátedra).

# Instalación de MySQL

- **Atención!** desinstalar primero cualquier versión anterior instalada de MySQL, solo si está instalada como un servicio del sistema. Ver sección 2.3.14 del manual (inglés: [refman-5.6-en.a4.pdf](#) ó español: [refman-5.0-es.a4.pdf](#)). Mas documentación en <http://dev.mysql.com/doc/>
- Opción 1 (recomendada): Descomprimir **mysql-5.X.XX-winXX.zip** en (por ejemplo) c:\.
- Opción 2: Ejecutar **mysql-5.X.XX-winXX.msi** y seguir las instrucciones.
- Todos los ejemplos presentados en esta clase hacen referencia a la carpeta “**bin**” de MySQL. Nos referiremos a ella de manera general como: **...\bin**

2

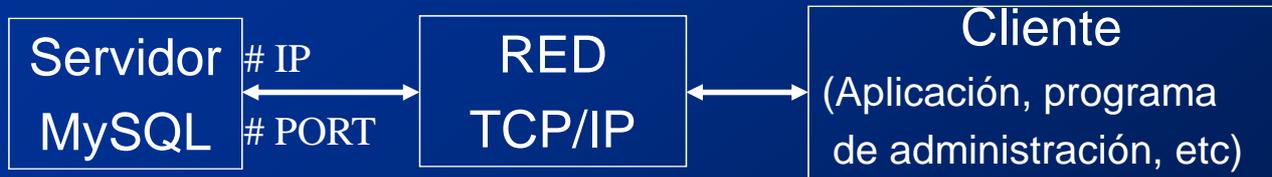
# Iniciar el servidor

- El Servidor se puede iniciar y detener manualmente desde la línea de comandos (cmd). Hay diferentes opciones, podemos ejecutar:  
**...\bin\mysqld** ó  
**...\bin\mysqld - -console** (muestra los mensajes\errores por la consola)
- Para detener el servidor ejecutar:  
**...\bin\mysqladmin -u root shutdown**
- Otra alternativa es **instalar MySQL como un servicio de Windows**. De esta forma, el servidor se iniciará y detendrá automáticamente cuando windows se inicie y se apague respectivamente.
- ver sección 2.3 del manual.

3

## Conectándose al Servidor

- MySQL sigue el **modelo Cliente-Servidor**.



- Permite definir **usuarios** con **claves** de acceso y con diferentes **privilegios**.
- Por defecto existe un **usuario root** con **clave vacía** y **acceso total** a todas las bases de datos. También existe un **usuario vacío ` `** con **clave vacía**.

4

## Conectándose al Servidor

- MySQL provee un **programa cliente** muy sencillo que permite conectarse al servidor y ejecutar consultas, desde la línea de comandos.
- El programa se llama **mysql.exe** y se encuentra en **...\bin**.
- Para conectarse al servidor utilizando el **cliente mysql** ejecutar: **...\bin\mysql -u root** (**Nota:** el servidor debe estar corriendo)
- Una vez que se conecto, se pueden ingresar sentencias SQL después del símbolo **mysql>**.
- ver secciones 3.1 y 3.2 del manual.

5

## Conectándose al Servidor: Seguridad

- Como el usuario *root* tienen una clave vacía, **cualquiera puede conectarse al servidor y tener acceso a todas las bases de datos !!!**
- Luego de instalar un servidor deberíamos cambiar la clave de acceso del usuario *root*.
- cambiamos la clave de *root* a '2014':  
`mysql> set password for root@localhost=password('2014');`
- para conectarnos al servidor con el usuario *root*:  
**...\bin\mysql -u root -p**  
Enter password: \*\*\*\* (ingresamos 2014)

6

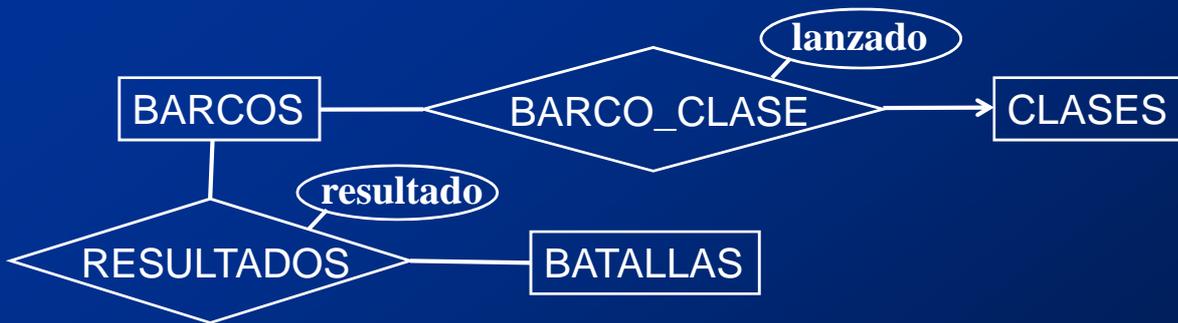
## Creando una base de datos

- Para crear una base de datos **utilizando el cliente *mysql*** hay dos opciones:
  - Ingresar las sentencias SQL una a una desde la línea de comandos, es decir, crear la base de datos y luego crear tabla por tabla. (ver sección 3.3 del manual).
  - Poner una secuencia de sentencias SQL en un archivo de texto, y ejecutar toda la secuencia de una vez. (**Batch Mode** - sección 3.5 del manual)

7

## Creando una base de datos: ejemplo

- Consideremos la B.D. *batallas* del práctico de SQL:



**BARCOS**(nombre barco, id, capitan)

**CLASES**(clase, tipo, pais, nro caniones, calibre, desplazamiento)

**BARCO CLASE**(nombre barco, clase, lanzado)

**BATALLAS**(nombre batalla, fecha)

**RESULTADOS**(nombre barco, nombre batalla, resultado)

8

## Creando una base de datos: ejemplo

- Para crear la base de datos anterior utilizaremos la opción de **batch mode**.
- Creamos un archivo de texto (**batallas.sql**) con la secuencia de sentencias necesarias para crear la base de datos.
- Luego desde el cliente ejecutamos:  
**mysql> \. batallas.sql**

9

## Creando una base de datos: ejemplo

- **Nota:** si una de las sentencias produce un error, es posible que alguna de las otras sentencias de la secuencia se ejecuten con éxito. Antes de ejecutar el archivo batch nuevamente borrar la base de datos desde el cliente con:

```
mysql> drop database batallas;
```

- El comando **drop database** no elimina los usuarios. Si la secuencia de comandos crea algún usuario, es necesario eliminarlos mediante:

```
mysql> drop user nombre_usuario@host;
```

10

## Usuarios y Autorización

- La **definición de usuarios** y la **asignación de privilegios** nos permiten restringir el acceso sobre las bases de datos almacenadas en el servidor.
- **Simplifica el uso del sistema y mejora la seguridad** ya que permite a los usuarios acceder solo a los datos que necesitan.
- Por ejemplo, podemos definir un usuario con acceso sólo a ciertas tablas dentro de una o mas bases de datos. Para cada tabla podemos definir el tipo de acceso (leer , insertar, modificar, etc). (ver secciones 6.2, 6.3 y 13.7 de refman-5.6-en.a4.pdf o 5.6, 5.7 y 13.5 de refman-5.0-es.a4.pdf).

11

# Usuarios y Autorización

- Los **usuarios** se identifican con un **nombre y un dominio**: *nombre@dominio*
- El **dominio** identifica **desde donde puede conectarse** el usuario al servidor de MySQL.
- ejemplos:
  - *root@localhost*, el usuario *root* solo puede conectarse desde la PC donde esta corriendo el servidor MySQL.
  - *uns@'%uns.edu.ar'*, el usuario *uns* solo puede conectarse desde una PC del dominio *uns.edu.ar*.
  - *admin@'%'* (o *admin*, sin especificar dominio), el usuario *admin* puede conectarse desde cualquier dominio. **Atencion!:** se debe eliminar el usuario vacío (`drop user ''@localhost`) para poder conectarse usando este usuario desde localhost (ver sección 6.2.7., página 768 del manual [refman-5.6-en.a4.pdf](#))

12

# Usuarios y Vistas

- MySQL también permite definir **vistas** de la base de datos. (ver capítulo 19 del manual)
- Una **vista** es una **tabla derivada** (calculada en función de otras tablas o vistas) **y persistente** en la Base de Datos.
- Las vistas **se crean a partir de una sentencia *select*** que define su contenido.
- Podemos restringir a los usuario a acceder a una o mas vistas de una o mas bases de datos.

13

# Usuarios y Autorización

- Estos usuarios, son **usuarios del servidor de MySQL** en general y definen **niveles o perfiles de acceso al servidor**.
- **Generalmente** estos usuarios serán **utilizados por las aplicaciones** (no por personas) para acceder al Servidor.
- Un **caso particular** sería por ejemplo un **usuario admin**, utilizado por el administrador (persona) de la B.D. para tareas de mantenimiento
- Si un sistema requiere la definición de **usuarios (personas)**, es conveniente definir estos usuarios dentro de la B.D. del sistema y manejar el acceso desde la/s aplicación/es utilizando usuarios definidos en el servidor.

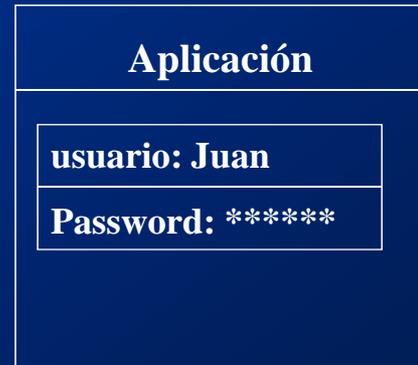
14

## Usuarios y Autorización: ejemplo

- Supongamos que en una **aplicación** que accede a una **B.D. comercio** de un sistema comercial, queremos definir **usuarios para** controlar el acceso de **los empleados**.
- La **B.D. comercio** contendrá una **tabla empleados** con información de los empleados y su password.
- Definimos un **usuario empleado en el servidor** a través del cual se conectará la aplicación. El usuario *empleado* define el **nivel de acceso de un empleado en general**.
- La aplicación se encargará de **controlar el acceso de los empleados** (por ejemplo requiriendo el usuario y password) utilizando la información de **tabla empleados**.

15

# Usuarios y Autorización: ejemplo



**La aplicación se conecta al servidor por medio del usuario *empleado*.**