

Métodos de Acceso a Datos

M.C. Carlos Rojas Sánchez¹

¹Licenciatura en Informática
Universidad del Mar - Puerto Escodido

[Bases de Datos Distribuidas]

1 Herramientas cliente para el acceso a datos

phpMyAdmin



Herramienta escrita en PHP para manejar la administración de MySQL a través de páginas web, utilizando Internet.

- Interface Web
- Manejador de base de datos MySQL, MariaDB y Drizzle
- Importación de datos desde CSV y SQL
- Exporta datos a varios formatos: CSV, SQL, XML, PDF (via la biblioteca TCPDF), ISO/IEC 26300 - OpenDocument Text y Spreadsheet, Word, Excel, \LaTeX y otros
- Administración de múltiples servidores

- Crea gráficos PDF del diseño de la base de datos
- Crea consultas complejas usando Query-by-Example (QBE)
- Búsqueda global en una base de datos o un subconjunto de esta
- Transforma datos almacenados a cualquier formato usando un conjunto de funciones predefinidas, tal como BLOB
- Live charts para monitoriar las actividades del servidor MySQL tales como conexiones, procesos, uso de CPU/Memoria, etc.

MySQL Workbench



Herramienta visual de base de datos unificada para arquitectos, desarrolladores y administradores de bases. MySQL Workbench ofrece modelado de datos, desarrollo de SQL y herramientas completas de administración de servidor de administración de configuración, el usuario y mucho más.

MySQL Workbench

- Plataformas Compatibles: Windows, Linux y Mac OS.
- Precio: Gratuito.
- Fabricante: Oracle.

Navicat for MySQL

Para la administración y desarrollo de MySQL. Ofrece una interfaz gráfica intuitiva y de gran alcance para la gestión de bases de datos, desarrollo y mantenimiento. Un conjunto completo de herramientas para su funcionalidad básica, o profesional.

- Plataformas Compatibles: Windows, Linux y Mac OS.
- Precio: Windows(99.00\$-199.00\$), Mac(79.00\$ -179.00\$) y Linux(79.00\$ - 179.00\$).
- Versión de Prueba: 30 días.
- Fabricante: PremiumSoft TM CyberTech Ltd.

Sequel Pro

Aplicación de gestión de base de datos de Mac, rápido para trabajar con bases de datos MySQL.

- Plataformas Compatibles: Mac OS X 10.5 Leopard o posterior
- Precio: Gratuito.
- Versión de Prueba: 30 días.
- Fabricante: Sequel Pro & CocoaMySQL Teams

HeidiSQL

Aplicación basada en Windows para administrar base de datos MySQL. Permite ver y editar los datos, crear y editar tablas, vistas, procedimientos, triggers y eventos programados. Además, puede exportar la estructura y los datos ya sea para archivo SQL, porta papeles o en otros servidores.

- Plataformas Compatibles: Windows
- Precio: Gratuito.
- Fabricante: Ansgar Becker

SQL Maestro: MySQL Tools Family

MySQL Tools Family útil para realizar diversas operaciones con objetos de base de datos, ejecutar consultas y scripts SQL, eficiente administrar usuarios y sus privilegios, la importación, la exportación y la copia de seguridad de sus datos, etc Por otra parte, puede generar scripts PHP para las tablas y consultas seleccionadas y convertir cualquier base de datos compatible con ADO a la base de datos MySQL.

- Plataformas Compatibles: Windows
- Precio: 79.00\$ -1719.00\$.
- Fabricante: SQL Maestro Group.

SQLWave MySQL Client

Es un cliente de MySQL compacto, rápido y fácil de usar. Esta herramienta le permitirá conectarse en cualquier lugar fácilmente a sus hosts remotos. Compatible con todas las versiones actuales y recientes de MySQL, incluyendo sus nuevas características para la gestión de la estructura de base de datos: Trabajar con tablas, vistas, procedimientos de almacenamiento, funciones, eventos, claves externas.

- Plataformas Compatibles: Windows
- Precio: 99.00\$.
- Versión de Prueba: 30 días.
- Fabricante: Nerocode.

dbForge Studio

Herramienta GUI MySQL universal que permite a un desarrollador o administrador de MySQL crear y ejecutar consultas, desarrollar y depurar rutinas MySQL, así como la gestión de objetos de base de datos MySQL automatiza en el ambiente conveniente. Este MySQL front-end proporciona utilidades para comparar, sincronizar y copia de seguridad de base de datos MySQL con la programación, y da la posibilidad de analizar y reportar datos MySQL tablas.

- Plataformas Compatibles: Windows
- Precio: Gratis Edición Xpress, Comercial: 45.95\$ - 99.99\$.
- Fabricante: Devart.

DBTools Manager

Aplicación para la gestión de datos con soporte incorporado para MySQL y motores de bases de datos ODBC

- Plataformas Compatibles: Windows
- Precio: Gratis edición estándar, enterprise: 69.90\$.
- Versión de Prueba: 30 días.
- Fabricante: DBTools Software .

MyDB Studio

Colección de herramientas para MySQL y administración del servidores de bases de datos.

- Plataformas Compatibles: Windows
- Precio: Gratis.
- Fabricante: H2LSoft, Inc.

Dreamcoder for MySQL

Para crear, editar, duplicar, exportar y borrar objetos, compilar y ejecutar procedimientos almacenados, exportar e importar datos, generar reportes, monitorear la actividad de la base de datos, sincronizar la base de datos, construir y ejecutar consultas, formatear código, crear usuarios y ejecutar scripts.

- Plataformas Compatibles: Windows
- Precio: Gratis
- Fabricante: Mentat Technologies Database Solutions, Inc.

Tarea

- Cassandra Apache
 - Instalación
 - Configuración

2 Conectividad en lenguajes de alto nivel

Conectividad en lenguajes de alto nivel

API's

- Una interfaz de programación de aplicaciones o API (del ingles Application Programming Interface) es el conjunto de funciones y procedimientos.
- Su función es ser una interfaz entre la aplicación y la base de datos.

API :: ODBC

- ODBC (Open Data Base Connectivity): Esta tecnología proporciona una interfaz común para tener acceso a bases de datos SQL heterogéneas. ODBC está basado en SQL (Structured Query Language) como un estándar para tener acceso a datos. ODBC permite la conexión fácil desde varios lenguajes de programación y se utiliza mucho en el entorno Windows. Sobre ODBC Microsoft ha construido sus extensiones OLE DB y ADO

API :: JDBC

- JDBC (Java Data Base Connectivity): se trata del estándar para la conectividad entre el lenguaje Java y un amplio rango de sistemas gestores de bases de datos. Los JDBC pueden desenvolverse tanto en un nivel cliente, esto es, trabajando del lado de la aplicación, o en el servidor directamente relacionado con la base de datos. Cuando se encuentre a nivel cliente, trabajará con la tecnología ODBC para acceso a los datos.

API :: JDBC

■ Tipos de JDBC

- El puente JDBC-ODBC: fue uno de los primeros controladores disponibles, implementa un enlace para utilizar un controlador ODBC desde Java. Con el tiempo han surgido controladores JDBC específicos para cada base de datos que mejoran el rendimiento del puente JDBC-ODBC.
- Controladores Java parcialmente nativos: usan tanto código Java como binario específico de cada plataforma.
- Controladores JDBC-Net de Java puro: son controladores escritos completamente en Java que entienden un protocolo de red estándar (HTTP, etc.) y permiten comunicarse con un servidor de acceso a bases de datos, que es el que finalmente provee el acceso al SGBD específico (posiblemente con ODBC).
- Controladores de protocolo nativo en Java puro: escritos en Java puro, utilizan el protocolo específico de la marca del SGBD.

3 Implementación de algoritmos de fragmentación

Fragmentación en MySQL

- **RANGE:** Si la clave de fragmentación está dentro de un rango de valores.
- **LIST:** El fragmento es seleccionado de acuerdo a una lista de valores enteros.
- **HASH:** El fragmento se elige de acuerdo a una función de hash.
- **KEY:** Un algoritmo interno es utilizado por mysqld para elegir como serán distribuidos los datos entre los fragmentos.
- **Particionado compuesto:** Las particiones de RANGE y LIST pueden ser subfragmentadas usando el fragmentado HASH, y KEY.

Fragmentado RANGE (Por rango de valores)

Suponiendo que queremos almacenar los datos de 300 empleados en cuatro tablas de a 100 cada una.

- ALTER TABLE empleados
- PARTITION BY RANGE (id_almacenamiento) (
- PARTITION p0 VALUES LESS THAN (101),
- PARTITION p1 VALUES LESS THAN (201),
- PARTITION p2 VALUES LESS THAN (301),
- PARTITION pfinal VALUES LESS THAN MAXVALUE);

Fragmentado LIST (Por lista definida)

Suponiendo que ahora tenemos la misma tabla, pero lo que queremos es distribuir a los empleados de acuerdo a una lista de valores definida, los valores se distribuirán de acuerdo a los valores definidos en cada lista de cada partición.

```
CREATE TABLE empleados (  
  Id_empleado INT NOT NULL,  
  nombre VARCHAR(30) NOT NULL,  
  id_almacenamiento INT NOT NULL )  
PARTITION BY LIST (id_almacenamiento) (  
  PARTITION Sureste VALUES IN  
  (1,2,3,4,5,6,7,8,9,10,15,17,18,20,21,24),  
  PARTITION AtlanticoMedio VALUES IN  
  (11,12,13,14,16,19,22,23,25,26,27,28),  
  PARTITION Noreste VALUES IN  
  (29,30,33,38,40,41,50,56,64,65,75),  
  PARTITION EsteMedio VALUES IN  
  (32,34,35,42,43,49,51,61,62,63,71),  
  PARTITION Noroeste VALUES IN  
  (46,53,58,67,68,69,72,74),  
  PARTITION Canada VALUES IN (31,47,52,59,73),  
  PARTITION Inlaterra VALUES IN (39,55) );
```

Fragmentado HASH (Por dispersión)

En este caso, la distribución de los datos entre un numero x de fragmentos es con el uso del operador residuo (%)

```
CREATE TABLE empleados (  
  Id_empleado INT NOT NULL,  
  nombre VARCHAR(30) NOT NULL,  
  apellidos VARCHAR(30) NOT NULL,  
  fecha_inicio DATE NOT NULL DEFAULT '2000-01-01',  
  fecha_termino DATE DEFAULT NULL,  
  salario DECIMAL (8,2) NOT NULL,  
  codigo_de_trabajo INT NOT NULL,  
  id_almacenamiento INT NOT NULL )  
PARTITION BY HASH (id_almacenamiento)  
PARTITIONS 16;
```

Fragmentado KEY (Fragmentado por clave)

El funcionamiento de este fragmentado es similar al de HASH, la gran diferencia es el algoritmo, que es similar al de la funcion PASSWORD(), y tiene el siguiente cambio respecto a la consulta anterior.

```
CREATE TABLE empleados (  
  Id_empleado INT NOT NULL,  
  nombre VARCHAR(30) NOT NULL,  
  apellidos VARCHAR(30) NOT NULL,  
  fecha_inicio DATE NOT NULL DEFAULT '2000-01-01',  
  fecha_termino DATE DEFAULT NULL,  
  salario DECIMAL (8,2) NOT NULL,  
  codigo_de_trabajo INT NOT NULL,  
  id_almacenamiento INT NOT NULL )  
PARTITION BY HASH (id_almacenamiento)  
PARTITIONS 16;
```



Proyecto [2do Parcial]

Implementación de Algoritmos de Fragmentación Vertical

■ Proyecto

- Presentación (20 min)
- Reporte (5 páginas)
- Muestra (ejecución y explicación)

Bibliografía I

-  M. Tamer Özsu and Patrick Valduriez.
Principles of Distributed Database Systems.
Springer, 2011.
-  Abraham Silberschatz, Henry F. Korth and S. Sudarshan.
Fundamentos de Bases de Datos.
McGRAW-HILL, 2002.