

Práctica 3 - Catálogos en implementaciones SGBDR

Quintín Mesa Romero

Doble Grado en Ingeniería Informática y Matemáticas

quintinmr@correo.ugr.es

Descripción del sistema gestor de bases de datos

De entre todos los sistemas gestores de bases de datos que se mencionan en el enunciado de la práctica y entre los que no se ha hecho referencia, el único sobre el que tengo algo de conocimiento y con el que he trabajado ya previamente es **MySQL**, de ahí que haya elegido este y no otro. Otras de las razones por las que me he decantado por este SGBD es porque es muy utilizado en la industria, es muy popular, robusto y ha sido y es esencial para aplicaciones web, empresas y proyectos varios.



[1]

MySQL es un sistema gestor de bases de datos relacional de código abierto, que fue desarrollado por

Oracle. Utiliza un modelo de bases de datos relacional, es decir, los datos están organizados en tablas que se relacionan entre sí. Permite almacenar, organizar y recuperar datos de forma eficiente, usando para ello lenguaje **SQL**. [1]

Está basado en el modelo **cliente-servidor**. Esto es, incluye un servidor de bases de datos, y clientes que interactúan con la base de datos. [2]

Destaca por ser **portable** (funciona en diferentes sistemas operativos), **flexible** y **escalable**, esto es, por ser capaz de poder adaptarse a las necesidades de las aplicaciones que lo usan, **compatible** con gran cantidad de lenguajes de programación, por tener alto **rendimiento** en cuanto a que permite recuperar los datos de forma rápida y eficiente y por ser **seguro** en cuanto a que garantiza el cifrado de los datos y ofrece opciones de autenticación. [1]

Descripción de los identificadores que MySQL usa para cada objeto creado

Los objetos que permite crear MySQL como son la propia base de datos, tablas, índices, columnas, alias, vistas, procedimientos almacenados, tablaspace, grupo de recursos, entre otros, cuentan con un **identificador** que coincide con el **nombre** de dichos objetos.

Así, para las tablas, por ejemplo, el identificador es el propio nombre de la tabla, que debe ser **único** dentro de la base de datos. Del mismo modo, para objetos como vistas, índices, etc., se utilizan identificadores únicos que se corresponden con el nombre que tienen asignado. [3]

Los identificadores están sujetos a ciertas normas de **sintaxis**:

- Se almacenan usando Unicode (UTF8).
- La longitud máxima de los identificadores, en la mayoría de los casos es de 64 bytes, a excepción de Alias (255) y Compound Statement Label (16).
- No deben contener el carácter 0 o un byte con valor 255, y no deberían terminar con espacio.
- Pueden estar entrecomillados o no. En caso de que el identificador sea una palabra reservada contenga caracteres especiales, es obligatorio el entrecomillado (' ` '). [4]

No pueden utilizarse directamente como identificadores o parte de identificadores las variables de usuario (destinadas a proporcionar valores de datos y son propias de la sesión), como en contextos en los que se espera un nombre de tabla o de base de datos o como una palabra reservada como SELECT.

Otra de las características de los identificadores es que pueden ser calificados o no calificados. Se permitirá un nombre no calificado cuando su interpretación sea inequívoca. Un nombre calificado incluye mínimo un calificador que esclarece el contexto interpretativo. Pongamos un ejemplo que ilustre esto:

Esta sentencia crea una tabla usando un nombre no calificado, table1:

```
CREATE TABLE table1 (i INT);
```

Como table1 no incluye ningun calificador que especifique una base de datos concreta en la que se ha de crear, la sentencia crea dicha tabla en la base de datos por defecto. Si no hubiera base de datos por defecto, se produce un error. Entonces, se debería de escribir la siguiente sentencia que crea la tabla usando el nombre calificado db1.table1:

```
CREATE TABLE db1.table1 (i INT);
```

Así, se crea la tabla en la base de datos db1, sin tener en cuenta si hay base de datos por defecto o no. [6]

Por otra parte, en MySQL, las bases de datos no son más que directorios dentro del directorio de datos, en el que cada tabla se corresponde a al menos un archivo dentro del directorio de la base de datos. Es por ello que la sensibilidad a las mayúsculas y minúsculas del S.O que haya por debajo, es importante de cara a la sensibilidad a las mayúsculas y minúsculas que haya en los identificadores. En Windows, sabemos que no hay distinción entre mayúsculas y minúsculas, por lo que tampoco hay dicha distinción en los nombres. Lo mismo pasa con macOS. No ocurre lo mismo en los sistemas operativos Unix. [7]

Descripción de las estructuras que almacenan información en el catálogo de MySQL

Dentro de cada instancia de MySQL existe una base de datos denominada **INFORMATION_SCHEMA**, donde se almacena información sobre todos los objetos de todas las bases de datos que mantiene el servidor MySQL. Está constituida por tablas de solo lectura (en realidad vistas, por lo que no hay archivos asociados a ellas). [8]

Algunos ejemplos de lo que almacena: información de la base de datos, descripción de tablas, información sobre cada columna, información sobre los índices, sobre permisos, etc.[9]

Por otra parte, para cada objeto almacenado existen unas tablas con información específica al tipo de objeto: la información que proporcionan las diversas declaraciones **SHOW** que admite MySQL (SHOW DATABASES, SHOW TABLES,...) y que equivale a la sentencia SELECT recuperando la información de **INFORMATION_SCHEMA**. No obstante, el SELECT en **INFORMATION_SCHEMA** tiene grandes ventajas frente a SHOW como pueden ser:

- Todo el acceso se realiza en tablas, por lo que se ajusta bien a las reglas de Codd.
- Se utiliza la sintaxis de la declaración SELECT y solo necesitas aprender algunos nombres de tabla y columna.
- El implementador no necesita preocuparse por agregar palabras clave.
- Puedes filtrar, ordenar, concatenar y transformar los resultados de las consultas de INFORMATION_SCHEMA en el formato que tu aplicación necesite, como una estructura de datos o una representación de texto para analizar. [8]

Referencias

Información sobre MySQL

[1] <https://www.arsys.es/blog/mysql>

[2] <https://hoplasoftware.com/mysql-sistema-de-gestion-de-bases-de-datos-relacionales/>

Identificadores

[3] <https://dev.mysql.com/doc/refman/8.3/en/identifiers.html>

[4] <https://documentation.help/MySQL-5.0-es/ch09s02.html>

[5] <https://dev.mysql.com/doc/refman/8.3/en/user-variables.html>

[6] <https://dev.mysql.com/doc/refman/8.3/en/identifier-qualifiers.html>

[7] <https://dev.mysql.com/doc/refman/8.3/en/identifier-case-sensitivity.html>

Almacenamiento de información

[8] <https://dev.mysql.com/doc/refman/8.3/en/information-schema-introduction.html>

[9] https://rua.ua.es/dspace/bitstream/10045/36376/1/Catalogo_de_MySQL.pdf