

Consejos para BBDD MySQL

[bbdd, mysql](#)

(del artículo [Top 20+ MySQL Best Practices](#))

1. evitar ciertas funciones nativas de MYSQL (fecha, aleatoriedad) para evitar que no las cahee
2. Usar el comando EXPLAIN en nuestras sentencias SELECT para ver que hace y poder optimizarlas
3. Usar LIMIT 1 cuando queremos coger solo un registro
4. Indexar todas las columnas de la tabla que pueden ser susceptibles de una búsqueda
5. Usar el mismo tipo de índice y tipo de columna (character encoding) en las columnas que se usan para JOINS
6. No utilizar el ORDER BY RAND() LIMIT 1 para obtener una fila aleatoria
7. No utilizar el SELECT *
8. Tener siempre un ID, Primary Key, Auto_Increment, Unsigned. No utilizar VARCHAR como índices, es más lento. Una excepción son las tablas que relacionan 2 tablas muchos-a-muchos. En este caso se puede utilizar como clave primaria los 2 campos a la vez
9. Usar ENUM en lugar de VARCHAR (ENUM funciona como un TinyInt) siempre que sea posible
10. Usar PROCEDURE ANALYSE(), te da sugerencias que puedes seguir o no.
11. Evitar el uso del NULL en las columnas siempre que sea posible
12. Usar sentencias SQL preparadas (mysqli_extension, PDO), se pueden reutilizar con diferentes valores pero solo se compilan 1 vez
13. unbuffered Queries (mysqli_unbuffered_query())
14. guardar direcciones IP en UNSIGNED INT (4 bytes), con las funciones INET_ATON() y INET_NTOA()
15. Las tablas que tienen todos los campos de longitud fija son más rápidas. Las columnas del tipo VARCHAR, TEXT, BLOB no lo son.
16. Usar técnicas de «particionado vertical»: fraccionar la tabla en varias partes en función de:
 1. datos pocos frecuentes VS datos muy utilizados
 2. campos con alta frecuencia de actualización (UPDATES)
 3. OJO con lo JOIN si son muy frecuentes no valdrá la pena
17. Fraccionar DELETES e INSERTs grandes en fracciones (LIMIT 1000), usleep(50000)
18. Ajustar las columnas a su tamaño adecuado, las columnas pequeñas son más rápidas.
19. Escoger correctamente la Storage Engine:
 1. MyISAM: ideal para heavy-read, lento en escritura (+ bloqueo de tabla). Cálculo rápido de SELECT COUNT(*)
 2. InnoDB: más compleja que MyISAM, más lenta para cosas pequeñas. Soporta bloqueo de registro (en vez de la tabla entera)
20. Utilizar un ORM que aproveche las transacciones (Doctrine)

[Soporte PDF del enlace](#)

From:

<https://matebcn.eu/> - **miguel angel torres egea**

Permanent link:

<https://matebcn.eu/bbdd:mysql:consejos:start>

Last update: **24/05/2018 20:54**

