**Bases de datos OLTP y OLAP**

OLTP - On-Line Transactional Processing

Los sistemas OLTP son bases de datos orientadas al procesamiento de transacciones. Una transacción genera un proceso atómico (que debe ser validado con un *commit*, o invalidado con un *rollback*), y que puede involucrar operaciones de inserción, modificación y borrado de datos. El proceso transaccional es típico de las bases de datos operacionales.

*   El acceso a los datos está optimizado para tareas frecuentes de lectura y escritura. (Por ejemplo, la enorme cantidad de transacciones que tienen que soportar las BD de bancos o hipermercados diariamente).

*   Los datos se estructuran según el nivel aplicación (programa de gestión a medida, ERP o CRM implantado, sistema de información departamental...).

*   Los formatos de los datos no son necesariamente uniformes en los diferentes departamentos (es común la falta de compatibilidad y la existencia de islas de datos).

*   El historial de datos suele limitarse a los datos actuales o recientes.

OLAP - On-Line Analytical Processing

Los sistemas OLAP son bases de datos orientadas al procesamiento analítico. Este análisis suele implicar, generalmente, la lectura de grandes cantidades de datos para llegar a extraer algún tipo de información útil: tendencias de ventas, patrones de comportamiento de los consumidores, elaboración de informes complejos… etc. Este sistema es típico de los datamarts.

*   El acceso a los datos suele ser de sólo lectura. La acción más común es la consulta, con muy pocas inserciones, actualizaciones o eliminaciones.

*   Los datos se estructuran según las áreas de negocio, y los formatos de los datos están integrados de manera uniforme en toda la organización.

*   El historial de datos es a largo plazo, normalmente de dos a cinco años.

*   Las bases de datos OLAP se suelen alimentar de información procedente de los sistemas operacionales existentes, mediante un proceso de extracción, transformación y carga (ETL).

[**¿MyISAM o InnoDB? Motores de almacenamiento MySQL**](https://blog.arsys.es/myisam-o-innodb-elige-tu-motor-de-almacenamiento-mysql/)

El **motor de almacenamiento** (*storage-engine*) se encarga de almacenar, manejar **y recuperar información de una tabla**. Los motores más conocidos son **MyISAM** e **InnoDB**. La elección de uno u otro dependerá mucho del escenario donde se aplique.

Con la elección del motor se pretende conseguir la mejor relación de calidad acorde con nuestra aplicación. **Si necesitamos transacciones**, claves foráneas y bloqueos, tendremos que escoger **InnoDB**. Por el contrario, **escogeremos MyISAM** en aquellos casos en los que predominen las **consultas SELECT a la base de datos**.

**InnoDB** dota a MySQL de un motor de almacenamiento transaccional (conforme a ACID) con capacidades de commit (confirmación), rollback (cancelación) y recuperación de fallos. InnoDB realiza bloqueos a nivel de fila y también proporciona funciones de lectura consistente sin bloqueo en sentencias SELECT. Estas características incrementan el rendimiento y la capacidad de gestionar múltiples usuarios simultáneos. No se necesita un bloqueo escalado en InnoDB porque los bloqueos a nivel de fila ocupan muy poco espacio. InnoDB soporta restricciones FOREIGN KEY. En consultas SQL, aún en la misma consulta, pueden incluirse tablas del tipo InnoDB con tablas de otros tipos.

**MyISAM** es el motor por defecto. Está orientado a la indexación de registros por las claves definidas y por tanto de mayor uso para bases de datos / tablas de cosulta.

Para crear una tabla InnoDB se debe especificar la opción ENGINE = InnoDB o TYPE = InnoDB en la

sentencia SQL de creación de tabla:

*CREATE TABLE customers (a INT, b CHAR (20), INDEX (a)) ENGINE=InnoDB;*

*CREATE TABLE customers (a INT, b CHAR (20), INDEX (a)) TYPE=MyISAM;*

Para ver el motor que estamos usando en una base de datos o más específicamente en una tabla podemos usar:

* SHOW engines;
* SHOW CREATE TABLE
* Select table\_schema,table\_name,engine from information\_schema.tables;
* SHOW TABLE STATUS
* mysqlshow

**Ventajas: MyISAM vs InnoDB**

**InnoDB (OLPT)**

* Soporte de transacciones
* Bloqueo de registros
* Nos permite tener las características ACID (Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad), garantizando la integridad de nuestras tablas.
* Si nuestra aplicación hace un uso elevado de INSERT y UPDATE mejor rendimiento con respecto a MyISAM.

**MyISAM (OLAP)**

* Mayor velocidad en general a la hora de recuperar datos.
* Recomendable para aplicaciones en las que dominan las sentencias SELECT ante los INSERT / UPDATE.
* Ausencia de características de atomicidad ya que no tiene que hacer comprobaciones de la integridad referencial, ni bloquear las tablas para realizar las operaciones, esto nos lleva como los anteriores puntos a una mayor velocidad.

**¿Dudas sobre qué motor es el que necesitas? La mejor decisión basada en…..**

¿Tu tabla va a recibir INSERTs, UPDATEs y DELETES mucho más tiempo de lo que será consultada?  Más recomendable utilizar InnoDB

¿Necesitarás hacer búsquedas full-text?   Tu motor ha de ser MyISAM

¿Prefieres o requieres diseño relacional de bases de datos?  Entonces necesitas InnoDB

¿Es un problema el espacio en disco o memoria RAM?  Decántate por MyISAM

COMO VERLO EN MYSQL:

Show table status from database;

SHOW TABLE STATUS FROM MI\_BASE\_DATOS;