

Herramientas de mapeo objeto-relacional (ORM)

Si te dedicas a la programación orientada a objetos, seguramente más de una vez te habrás encontrado con la dificultad por ejemplo de la persistencia de los datos. Para minimizar estas dificultades, existen herramientas que se encargan de generar de manera automática el acceso a los datos, abstrayendo al programador de esta tarea. Son las llamadas **herramientas ORM**.

El modelo relacional y el modelo orientado a objetos

El **modelo relacional** es un modelo de datos sobre el que la gran mayoría de bases de datos organiza la información que almacena. A través de este modelo, la información se organiza en tablas estructuradas en registros y que pueden vincularse entre ellas.

El **modelo orientado a objetos** trabaja con entidades llamadas objetos que combinan estructuras de datos con el comportamiento de estos.

Resulta evidente que al programar en lenguajes orientados a objetos y usar una base de datos relacional, existe una brecha que dificulta el manejo de los datos, pudiendo perder funcionalidad como la herencia o desajustes en los diferentes tipos de datos.

Es por ello por lo que se hace necesario el uso de una solución intermedia que nos permita realizar **el mapeo objeto-relacional** que pueda atenuar los efectos de estos desajustes. Esta solución intermedia es una herramienta ORM

¿Qué es una herramienta ORM?

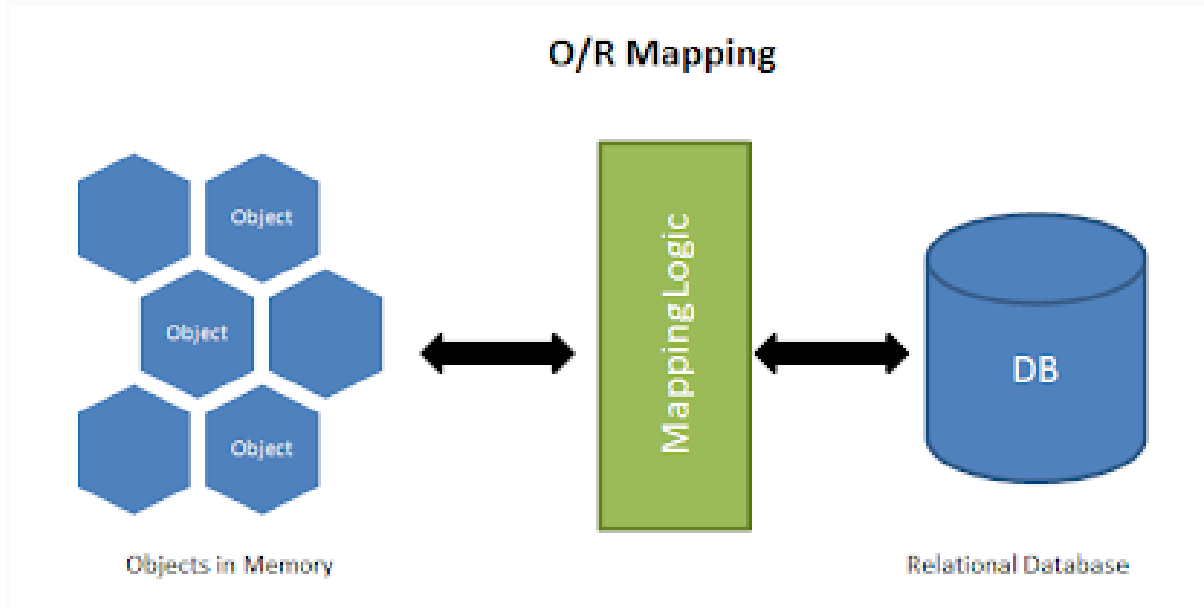
El mapeo objeto-relacional es una **técnica de programación** que permite convertir datos del sistema de tipos utilizado por los lenguajes de programación orientado a objetos al utilizado por las bases de datos relacionales.

De esta forma, se crea una base de datos **virtual** orientada a objetos sobre la base de datos relacional. Esto, además, posibilita el uso de las características propias de la orientación a objetos (esencialmente la herencia y el polimorfismo).

Las herramientas ORM usan esta técnica para convertir los objetos de la aplicación desarrollada en registros que puedan ser almacenados en una base de datos relacional. Esto suele afectar a un modelo de varias tablas relacionadas por la complejidad que pueda tener el objeto.

De la misma forma, al recuperar por parte de la aplicación los datos almacenados en la base relacional, los registros se convierten en objetos.

De esta forma, **se facilita al programador** la programación abstrayéndolo del modelo relacional de la base de datos.



Ventajas e inconvenientes de las herramientas ORM

Las herramientas ORM ofrecen una serie de ventajas para el programador como son:

- **Rapidez** en el desarrollo
- **Abstracción** de la base de datos utilizada
- **Seguridad** de la capa de acceso
- **Facilidad** para el mantenimiento del código
- Lenguaje propio para la realización de consultas

Pero no todos son ventajas, también tienen una serie de inconvenientes que deben valorarse, como son:

- El aprendizaje del lenguaje de la herramienta ORM puede resultar ser **complejo** ya que, para poder sacar el máximo partido a la herramienta, es necesario conocer en profundidad cómo funciona la misma.
- En entornos de gran carga, este tipo de solución **penaliza** el rendimiento debido a los procesos de transformación de las consultas que se hagan hacia la base de datos.

Herramientas ORM

En el mercado podemos encontrar diversas herramientas tanto de pago como de uso libre.

Algunos ejemplos específicos serían:

- Para **Java**: Hibernate
- Para **.Net**: Entity Framewor
- Para **PHP**: Doctrine, Propel
- Para **Python**: SQLAlchemy

Conclusión

El uso de herramientas ORM **puede ser** una buena solución en el desarrollo de aplicaciones en lenguajes de programación orientados a objetos ya que ayuda a trasladar el modelo orientado a objetos al modelo relacional, evitando el uso de sentencias SQL embebidas en el código de la aplicación.

Sin embargo, si las aplicaciones a desarrollar van a atender gran carga de acceso a datos o requieren tiempos de respuesta mínimos, este tipo de herramientas **no serían idóneas**, ya que penalizarían el rendimiento.