



## Estudio comparativo de sistemas de mapeo objeto relacional desarrollados en plataformas Open Source

### Comparative study of object relational mapping system developed in Open Source platform

Polo Amador, Leydi Johana<sup>1</sup>.

#### Resumen

En la actualidad hay grandes alternativas de proveedores para el desarrollo de software, la importancia que cobra los ORM tipo OPEN SOURCE es la facilidad de implementación de esta herramienta permitiendo tomar como base un modelo de software terminado para transformarlo a un nuevo y mejor producto. Uno de los principales inconvenientes encontrados para el estudio de esta temática, es la poca existencia de fuentes de información en dónde explique lo que involucra este tópico en el mundo del software. Es por esta razón que el principal objetivo de este trabajo de investigación es brindar a los estudiantes, docentes y profesionales del área del desarrollo software, una herramienta que sirva de guía en el estudio e implementación de los sistemas de mapeo objeto relacional tipo Open Source.

**Palabras Clave** – ORM, Código Abierto, base de datos relacionales, software libre, base de datos virtual.

#### Abstract

Currently there are alternative suppliers for large software development, the importance that the ORM Open Source is the ease of implementation of this tool allowing to build on a model of software to transform over to a new and better product. One of the main problems encountered in the study of this subject, is the limited availability of sources of information that explains what is involved in this topic in the software world. It is for this reason that the main objective of this research is to provide students, teachers and professionals in the area of software development, a tool to guide the study and implementation of object-relational mapping systems Open Source.

---

<sup>1</sup> Docente Investigador. Programa de Ingeniería De Sistemas. Facultad de Ingenierías; poloamador21@gmail.com



**key words.** ORM, Open Source, relational database, free software, virtual database.

## Introducción

En la actualidad los profesionales deben ser competentes para poder vincularse fácilmente en el mercado laboral, de aquí parte la iniciativa del Profesional en Ingeniería de Sistemas para que se involucre con las nuevas tendencias de software como lo es el mapeo objeto relacional el cuál optimiza las tareas según la funcionalidad del modelo seleccionado ya que puede variar de acuerdo a las necesidades del usuario y su entorno.

Es por esto que al interior del seminario taller de perfeccionamiento se planteó la temática “Estudio comparativo de sistemas de Mapeo Objeto Relacional desarrollados en plataformas Open Source” como tema de investigación debido a que los ORM cumplen una función en el desarrollo de software libre y al ser una nueva alternativa de trabajo no se encuentra información en los diferentes medios y es en este momento donde es importante documentar.

Por esta razón en el siguiente trabajo se encuentra la recopilación de información relacionada con los ORM tipo OPEN SOURCE, ya que existen varios entre esos se encuentra: propel, hibernate, linq, Doctrine. En dónde cada uno cumple una función específica y se adapta a las necesidades requeridas por el usuario.

Tomando así como referencia los casos de estudio de ORM planteados para su exploración y orientación al desarrollador sobre cuál herramienta facilita la implementación.

## Contenido

### A. Conceptualización de un ORM

Cuando se habla de software orientado a objetos y de base de datos relacional se refiere a las características y propiedades que contiene la base de datos ya que cada uno de los atributos se identifica por su funcionalidad. Por lo tanto, para almacenar la información que se encuentra desarrollada en un programa orientado a objetos a una base de datos relacional se necesita la traducción de las dos formas, primero se deben convertir en registros para poder guardarlos con mayor agilidad y luego realizar la operación inversa si es que se quiere recuperar los datos convirtiéndolo de registros a objetos. (Viana Israel)



Características generales de los ORM. Entre las principales características se encuentran las siguientes:

- Rapidez de desarrollo. ORM permite crear un modelo ajustado ya que lee automáticamente el esquema de tablas y relaciones.
- Abstracción del motor de base de datos. ayuda a que automáticamente las consultas que se realicen en la base de datos se conviertan de registros a objetos y viceversa y de esta forma adaptarse a los proveedores: (MYSQL, ORACLE, POSTRESQL, ETC.).
- Lenguaje propio para consultas a la base de datos: las herramientas que ofrece ORM para poder extraer los datos de la forma que se necesita (filtros, ordenaciones, agrupaciones).
- Proporciona una interfaz siendo más simples para el manejo de objetos a través de su propio lenguaje de consulta [Anonimo].

Las ventajas del Mapeo objeto relacional son:

- Reutilización. Permite llamar a los métodos de un objeto de datos desde distintas partes de la aplicación e incluso desde diferentes aplicaciones.
- Encapsulación. Encapsula la lógica de los datos pudiendo hacer cambios que afectan a toda la aplicación únicamente modificando una función.
- Portabilidad. Permite cambiar en un proyecto el tipo de proveedor de una base de datos MySQL a una Oracle sin ningún tipo de complicación.
- Seguridad. Permite proteger las aplicaciones de los ataques más comunes como SQL Injections.
- Mantenimiento del código. La correcta ordenación de la capa de datos permite modificar y mantener el código [tecnoretas].

Entre las desventajas del Mapeo objeto relacional se tienen:

- Su curva de aprendizaje es amplia debido a la gran variedad de librerías que ofrece ORM explorar la totalidad de su rendimiento costaría tiempo, se puede usar en un proyecto de poca complejidad para un excelente rendimiento.
- Posee menor rendimiento cuando se desarrolla un mapeador se encuentra un enorme contenido entre la capa de código y el sistema de datos esto permitirá que el rendimiento de la aplicación sea baja.
- La existencia de sistemas complejos en un proyecto de mayor complejidad al utilizar un ORM también se verá afectado y la complejidad será igual [web.ontuts].

## B. Proceso de mapeo de los ORM



El mapeo objeto relacional suele ser una técnica de programación que permite la interacción entre un lenguaje de programación orientado a objetos y la utilización de una base de datos relacional. En la práctica inicialmente crea una base de datos orientada a objetos virtual, sobre la base de datos relacional, esta es la forma de trasladar la información existente evitando la pérdida de datos y procedimientos.

Esto permite el uso de las características propias de la orientación a objetos en la base de datos, básicamente herencia y polimorfismo, a continuación una breve presentación de estas dos características.

**HERENCIA.** Es el proceso en el cual se pueden construir nuevas clases a partir de las existentes las cuáles ya se encuentran comprobadas.

Este tipo de proceso cobra gran importancia ya que se toma como referencia un procedimiento base para desarrollar a partir de ellos nuevas operaciones que se requieren en la arquitectura a desarrollar

**POLIMORFISMO.** Se entiende como la forma de poder interpretar varias clases que se derivan de una clase base. En dónde cada clase puede utilizar un mismo método de manera distinta.

### C. Tipos de ORM

En la actualidad existen diferentes tipos de software que varían de acuerdo al tipo de licencia que le pueda brindar al usuario, por esta razón el mapeo objeto relacional tipo Open source cobran una gran importancia en el mundo del software de libre acceso ya que los desarrolladores de software y personas interesadas pueden utilizar los recursos fácilmente. Es esta una de las razones por las cuales los usuarios están incursionando en el mundo de las nuevas tendencias tecnológicas.

**HIBERNATE:** es una herramienta que permite al desarrollador guardar tanto objetos en la base de datos como borrarlos de acuerdo a la necesidad mediante archivos declarativos XML (lenguaje de marcas extensibles), gracias a esto se va generando bases de datos en cualquier entorno. Caracterizándose por ser OPEN SOURCE es una gran ventaja ya que no presenta inconvenientes por adquirirlo.

Características:



- Entorno de trabajo. Su plataforma permite su utilización en redes de computadores que estén realizando procesos en paralelos así su arquitectura de caché de doble capa sería más eficaz.
- Sistema de archivos. Su formato de archivos en el momento de los mapeos es XML ofreciendo ventajas entre esas la representación de la información y el permitir ejecutar modificaciones [Quintero].
- Funcionalidad. Implementa la función de metadatos las cuales son palabras claves que agrupan grandes volúmenes de información. Entre sus principales.
- Licenciamiento. Se clasifica entre los ORM de software libre por contar con una licencia LGPL ( Lesser GNU Public License). La cual permite modificación al obtener su código fuente para la preparación de nuevas tareas o productos.

Algunas ventajas al implementar Hibernate:

- Hibernate utiliza menos líneas de código de esta forma permite mejorar su interpretación y manejo. Por esta razón al disponer de un mapeador permite el mayor rendimiento posible, Massol (2007).
- Desarrolla una gran flexibilidad que le ayuda a integrarse con arquitecturas como J2EE, JNDI, JTA entre otras, ayudando a que la configuración sea válida en aplicaciones independientes o con servidores, Massol (2007).
- Existen varias formas que permiten la integración de cualquier sistema de acceso a base de datos relacionales. Siendo de esta manera más productiva y su tiempo de desarrollo es más corto.

Los inconvenientes que se tienen de hibernate:

- Al utilizar clases en tiempo de ejecución genera más sobrecarga que las consultas en SQL.
- Requiere mayor rendimiento de las máquinas en las cuales se desarrollarán los aplicativos.
- Las tablas con las que se desee trabajar debe tener como requisito una clave primaria para poder identificar los registros de alguna forma.

LINQ: Lenguaje Integrated Query. Es un entorno de trabajo que permite el desarrollo de las base de datos relacionales, debido a su flexibilidad realizando el respectivo proceso de conversión para poder interpretar sus atributos permite el uso de varias arquitecturas de software ya que es compatible con los diferentes proveedores que se encuentra en el mercado tecnológico.



Las Ventajas presentadas por el Lenguaje Integrated Query:

Por su fácil uso mejora la productividad en los usuarios y la complejidad en su proceso de aprendizaje es sencilla por parte de los desarrolladores. De igual forma se caracteriza por contener un conjunto de extensiones del Net framework el cual incluye un lenguaje integrado de consultas. De acuerdo a su entorno de trabajo de fácil manipulación existe un gran número de personas que utilizan correctamente los ORM de este tipo.

Permite la ejecución de consultas integradas utilizando diversos lenguajes para acceder a la información requerida. Cuando se implementa bases de datos en un programa específico se busca la compatibilidad con otros proveedores.

Su aplicabilidad se destaca por las siguientes características:

- Enumeración de procesos o hilos de un proceso.
- Navegación por la bitácora de sucesos del sistema.
- Navegación por el sistema de ficheros local.
- Enumeración de unidades de red, recursos compartidos, etc.
- Enumeración de usuarios y grupos del Directorio Activo.
- Enumeración de características de ensamblando clases mediante reflexión [8].

DOCTRINE: es una herramienta que funciona como mapeador de objetos relacionales, en sus versión 5.2 para php (HyperText Pre-processor) permitiendo obtener los datos solicitados de una forma eficaz ya que “brinda varias alternativas de consultas a la base de datos ya sea de forma automática o escribiendo el dato a consultar.”

A continuación las principales Características:

- En su implementación contiene un lenguaje SQL propio del software llamado DQL (Doctrine Query Lenguaje) el cual entre sus funciones permite obtener objetos en sus consultas.
- Entre sus clases para su desarrollo soporta diferentes tipos de herencia que permitirán optimizar cada una de las operaciones.
- El sistema de archivo utilizado es YAML (no es otro lenguaje de marcado) de igual forma se puede programar en código PHP.
- Permite crear manualmente y automáticamente el modelo de base de datos a implementar.

En la actualidad Doctrine es un gran ORM que permite ejecutar varias operaciones para la búsqueda de la información que se requiera. De igual forma



por el sistema de archivos que maneja llamado YAML facilita la manipulación de su código.

PROPEL: es un entorno de trabajo que permite realizar diversas actividades al ser software libre y encontrarse integrado a Symfony siendo este un entorno de trabajo que ofrece funciones, herramientas y clases para el diseño de aplicaciones web.

Se identifica por la automatización de tareas brindando al usuario desarrollador espacio para el diseño de nuevos aplicativos.

SYMFONY: es un entorno de trabajo que se identifica por la gran variedad de herramientas que brinda al usuario, ahorrando tiempo y ejecutando operaciones automáticas. Esta desarrollado en php y puede ser soportado en múltiples plataformas.

A continuación las principales Características:

- Permite la utilización de la programación orientada a objetos y poder emplear clases y demás herramientas que optimizan el trabajo.
- El sistema de archivos utilizado es XML para la gestión de base de datos creándolo automáticamente.
- Utiliza el MVC (Modelo Vista Controlador) en la ejecución de tareas.
- La capa de abstracción hace la conversión automática de las consultas.

Propel, es una novedosa herramienta que por el hecho de ser software libre permite la reestructuración y adecuación de nuevas aplicaciones. Su sistema de archivos XML cobra gran importancia por la facilidad en el manejo de la base de datos y las operaciones que se vayan a desarrollar.

D. Open Source.

Es una de las fundaciones dedicadas a la promoción del código abierto por eso en la actualidad cobra gran importancia y es de gran reconocimiento para los desarrolladores de software. Se conoce por software libre a la disposición del uso de un programa específico sin dificultades al utilizarlo por ser de libre acceso.

La Open source suele identificarse por las siglas FOSS (Open Source Software) o FLOSS (Free License Open Source Software), y entre sus principales funciones se encuentra que los usuarios pueden tomar de forma libre su código fuente para estudios o modificaciones.



Los principales motores de identificación son el software libre y el código abierto, en dónde cumplen una función muy importante.

Las principales razones por las cuáles se eligió los ORM (Hibernate, Linq, Doctrine, Propel) son:

- Existe un gran volúmen de información acerca de su funcionamiento, operaciones a desarrollar, código fuente, modelos bases.
- La forma de realizar el mapeo de los datos existentes en las bases de datos es automático permitiendo la optimización de la información.
- Su código es fácil de entender y manipular, ya que existe un gran número de programadores en estos tipos de ORM.
- Al ser la mayoría de licencia gratuita permite la ejecución de modificaciones que permitan la optimización de un ORM.
- Estos ORM sus herramientas son de fácil manejo a comparación de otros que la curva de aprendizaje de las herramientas es amplia y requiere de mucho tiempo para su estudio y manejo.

Al compararse este grupo de ORM se definió y empleó unos criterios de comparación para revisar diferentes aspectos entre esos su funcionalidad y caracterización en el entorno de trabajo de estos, estos son:

**El Framework trabaja a modo de peticiones:** debido a que existe software que únicamente puede procesar una sola petición, esto hace que el sistema se vuelva lento y que colapse en cualquier operación.

**El modelo es apropiado para procesar la información:** según el volumen de la información se debe elegir el mapeador ya que existen sistemas muy robustos.

**Convierte los registros en objetos:** su sistema automáticamente realiza la conversión automática de registros y objetos.

**Requiere poco tiempo de gestión:** en el momento de realizar alguna operación con los registros de su base de datos los resultados son obtenidos rápidamente.

**Es funcional en cuanto el almacenamiento de datos:** para almacenar los datos se hacen fácilmente sin temor a la pérdida de información brindando seguridad en los datos.

**Permite hacer modificaciones en cualquier momento de su desarrollo:** al desarrollarse un aplicativa permite cambios para la optimización de los requerimientos del programa.





**Recupera los registros:** en caso de presentarse alguna falla el sistema permite la recuperación de los últimos datos y de la información anteriormente guardada.

**Hace conversión de los registros automáticamente:** realiza la conversión automática de los registros evitando problemas de tiempo y redundancia de los datos.

**Las herramientas son amplias y requieren estudio:** sus herramientas son amplias por eso es necesario ser estudiada su aplicabilidad siendo estas de fácil entendimiento.

**Permite la abstracción del motor de B.D. sin importar su proveedor:** si en un futuro se desea cambiar de sistema y por consiguiente de motor de base de datos los mapeadores permite realizar esta operación sin temer a la pérdida de la información.

**Sirve para aplicativos que maneja un gran volumen de información:** para las empresas que manejan un gran volumen de información y datos diariamente esta es una solución inmediata para la administración y organización de los datos.

## Conclusiones.

- Se encontraron 3 (tres) tipos de ORM, todos diferentes en su aspecto de funcionalidad y conformación, se puede concluir que los ejemplos propuestos en este trabajo de investigación, cumple con las expectativas para abordar el tema de ORM.
- Después de analizar los ejemplos de ORM se concluye que cada uno se identifica por el lenguaje en el que se encuentra desarrollado y pueden adaptarse a los diversos proveedores de software.
- Al realizar el análisis bajo los criterios de comparación se obtuvieron resultados los cuales varían y prevaleció que todos son de libre distribución y suelen ser útiles según las necesidades del usuario.
- Al analizarse los ejemplos de ORM se dedujo que suelen ser funcionales y se pueden desarrollar nuevos modelos a partir de ellos según los requerimientos para la solución de una situación específica.
- La utilización de un mapeador por parte de los ORM es importante debido a que ayuda a optimizar su desarrollo, disminuye costos, y son de libre distribución.



- El proceso del mapeador es independientemente del motor de la base de datos siendo una tarea que se identifica por la conversión automática de los datos.

### Referencias bibliograficas

Anónimo. Tomado de <http://web.ontuts.com/tutoriales/introduccion-a-object-relational-mapping-orm/>.

<http://www.tecnoretas.com/programacion/que-es-doctrine-orm/>

<http://www.slideshare.net/Tonymx/introduccion-a-linq>

[http://www.ideg.es/wsc\\_content/pics/user\\_upload/Roberto%20Maestre.pdf](http://www.ideg.es/wsc_content/pics/user_upload/Roberto%20Maestre.pdf)

M.D. Atangana Transformación de esquemas relacionales a objetos, visitado el 11 Mayo de 2011.

Massol (2007). Introducción a Hibernate extraído el 20 Marzo 2011 desde fuente <http://www.juntadeandalucia.es/xwiki/bin/view/MADEJA/Hibernate>.

Suarez González Héctor, Manual Hibernate, 22 Marzo de 2003, <http://www.javaHispano.org>.

Tomado de <http://web.ontuts.com/tutoriales/introduccion-a-object-relational-mapping-orm/>

Quintero, Juan Bernardo. Revista EIA, Directrices para la construcción de artefactos de persistencia en el proceso de desarrollo de software,

Viana Israel Object Freezer Relational extraído el día 16 Marzo 2011 desde fuente <http://objectfreezer-r.sourceforge.net/>.