

OJB - Links and further readings

by Thomas Mahler

Table of contents

1 Summary.....	2
2 OJB - Tutorials and Howto.....	2
2.1 OJB and Spring.....	2
3 Design.....	2
4 Further readings on O/R mapping.....	3
5 Patterns.....	3
6 Books covering OJB.....	3

1. Summary

This page contains interesting Links and recommended readings that will help to learn more about OJB concepts, related projects, didactic material, research reports, integration in frameworks etc.

2. OJB - Tutorials and Howto

- [The famous Beer4All Struts/OJB tutorial by Chuck Cavaness](#)
- [A presentation on OJB held at the Atlanta Java Users Group by Chuck Cavaness](#)
- [An extensive tutorial on OJB by John Carnell](#)
- [Roberto Ghizzioli's tutorial on Struts, OJB, and nested tags](#)
- [An introductory tutorial on the O'Reilly site.](#)

2.1. OJB and Spring

- [Spring, OJB, and Struts, getting started](#) (June 23, 2005)
- [Spring, OJB, and Struts. Version 2 with DAO](#) (June 23, 2005)
- [Spring, OJB, and Struts. Version 3: JNDI Datasources and Caching](#) (June 27, 2005)
- [Springframework and OJB](#) (Nov 21, 2004)

3. Design

OJB is based on a variety of conceptual sources. In this section I'll give a summary about the most prominent influences.

1. **Craig Larmans Applying UML and Patterns**
2. **The Siemens Guys "Pattern-Oriented Software Architecture"**
3. **Scott Amblers classic papers on O/R mapping**
4. **The "Crossing Chasms" paper from Brown et. al.**
5. **The GOF Design Patterns**

(sorted by relevance)

1. The most important input came from [Applying UML and Patterns](#). It contains a chapter describing the design of a PersistenceBroker based approach persistence layer. His presentation contains a lot of other good ideas (e.g. usage of Proxies, caching etc.) I implemented a lot of his things 1:1. This book is a must have for all OJB developers !

2. Larman does not cover the dynamic metadata concept. He mentions that such a thing would be possible, but does not go into details. As I had been a fan of MetaLevel architectures for quite a while I wanted to have such a thing in OJB too !!!



[mop-gif](#)

I took the concepts from the book [Pattern-Oriented Software Architecture](#). They have a chapter on the **Reflection** pattern (aka [Open Implementation, Meta-Level Architecture](#)).

They even provide an example how to apply this pattern to a persistence layer.

There is another Architectural pattern from this book that I am using: The **Microkernel** pattern. My idea was to have a kernel (the PersistenceBroker) that does all the hard work (O/R mapping, JDBC access, etc.)

High Level object transaction frameworks like a ODMG or JDO implementations are clients to the PersistenceBroker kernel in this concept!

3. I read [Scott Amblers](#) papers before starting OJB. Sure! There are several things in OJB that are from his classic [The design of a robust persistence layer](#) and from his [Mapping Objects To Relational Databases](#). Most prominent: The **PersistenceBroker** concept.

I incorporated the Query API from the OpenSource project [COBRA](#) that applies Amblers PersistentCriteria concept.

Reading Amblers paper on these topics is a must.

But IMO these are the only aspects of Amblers presentation that map directly to OJB. Here are the concepts that differ:

- Amblers concept relies on a persistent base class.
- caching is not covered by his design
- his concept of OID does not fit for legacy databases with compound PKs.
- The OJB proxy concept is quite different (Ambler has proxy functionality in his PersistentObject base class.)
- OJB does not use Insert- and UpdateCriteria
- OJB uses a different mapping approach (A full metadata layer)

4. For several detail questions (like mapping inheritance hierarchies) I consulted [crossing chasms](#). This is also a very good source for all O/R implementors.

5. For all the "small things" I'm using the [common GOF patterns](#) like Factory, Observer, Singleton, Proxy, Adaptor, State, Command, etc.

[Here is a thesis describing concepts very similar to OJB.](#)

As I read this paper I saw a lot of thing inspired by OJB. It's giving a nice introduction into the PersistenceBroker pattern and related topics.

[The PARC software design area](#) pioneering in Metalevel computation, aspect oriented programming etc.

4. Further readings on O/R mapping

- [ObjectArchitects O/R pattern page](#)
- [JavaSkyLine page on database integration](#)
- [Barry and Associates page on O/R mapping](#)
- [Portland Pattern Repository page on O/R](#)
- [Martin Fowlers book "Pattern of Enterprise Application Architecture" covers many O/R patterns that can be found in OJB. Here you will find an online catalog of these patterns.](#)

5. Patterns

- [The Hillside Pattern page](#)
- [The Portland Pattern Repository](#)

6. Books covering OJB

- [The O'Reilly book on Struts programming by Chuck Cavaness has a whole chapter about how to build an applications model layers based on OJB. A must reading for everyone intending to use Struts and OJB.](#) All source code from the book can be found here: [Struts Programming sources](#).
- [There's also a WROX book on Struts + OJB](#) All source code from the book can be found here: [Professional Struts and OJB sources](#).

- [Enterprise Java Development on a Budget](#)