

Bachelor-/Masterarbeit

Analyse bestehender O/R Mapping Frameworks

Betreuung: Alexandre Hanft

Email: hanft@iis.uni-hildesheim.de

Web: www.iis.uni-hildesheim.de/~hanft/Abschlussarbeiten/Analyse_OR_Mapper_Aushang.pdf

Aufgrund der Vielzahl der aufgeführten Fragestellungen und abh. von Bachelor- oder Masterarbeit ist die Themenwahl in Abstimmung mit dem Betreuer einzugrenzen, wodurch sie die Möglichkeit mehrerer Arbeiten ergibt. Denkbar wären Z.B. eine eher praktische Analyse und Evaluation bestehender O/R Mapper-Frameworks oder eine eher konzeptionell orientierte Abhandlung des Themas.

Objektorientierung hat sich durchgesetzt, ebenso wie Datenbanken zur persistenten Speicherung der Daten. Jenseits von vereinheitlichten DB-Zugriffstechniken wie ODBC, JDBC, ADO, ADO.net besteht aber eine Lücke zwischen der OOP und der Objekt-Persistenz, da sich OO-Datenbanken nicht gegenüber den rel. DB durchsetzen konnten. Die immergleiche Implementierung von Zugriffsroutinen, welche Objekte aus relationalen Datenbanken laden und darin speichern, ist arbeitsaufwändig und fehleranfällig.

Verschiedene O/R-Mapper-Frameworks (ORM) versprechen Abhilfe, in dem sie das Mapping zwischen Objekte und Datenbanken automatisch erzeugen.

Für aktuelle Technologien wie Java, C++ oder .net sind verfügbare Frameworks zu untersuchen und zu vergleichen. Entweder innerhalb von Java oder Java vs. .net.

Welche Techniken werden dabei verwendet? Wie stark ist die Automatisierung fortgeschritten und wie stark muss bestehender OO-Code geändert werden? Wie unabhängig sind die Frameworks von Technologien und Datenbanken? Wie werden OO-Konzepte wie Vererbung, Aggregation und Komposition (komplexe Datentypen) umgesetzt und wie gut ist das? Welchen Performanceverlust ist für diesen Komfort in Kauf zu nehmen? Welche fortgeschrittenen Kriterien neben Datenbankstruktur, Lazy Loading (Hollow Objects), Dirty-Status-Verwaltung, Veränderung der Vererbungshierarchie und Transaktionen können noch als Auswahlkriterium der Persistenzframeworks herangezogen werden?

Welche Abgrenzung zu Objektspeichern wie z.B. Postgres oder objektrelationalen bzw. objektorientierten Datenbanken lässt sich vornehmen?

Beispiele für ORM-Frameworks für Java:

- OJB 0.8 (Apache)
- Hibernate (Sourceforge)
- TradeCity RexIP JDO
- WebGain TOPLink 4.0
- CoCo-Base
- plexx

Beispiele für ORM-Frameworks für .net:

- NDO
- OPF.net
- OPX
- Versant Open Access
- DADO Solutions

- DataObjects.NET
- OJB.NET

Quellen (sind verfügbar)

An Advanced Course in Database Systems- Beyond Relational Databases Suzanne Dietrich, Susan Urban ISBN 0130428981 (Hardback) Nov 2004, 320 pages, chapter 4

http://www.javamagazin.de/itr/online_artikel/psecom,id,197,nodeid,11.html

http://www.cetus-links.org/oo_db_systems_3.html

<http://www.ambysoft.com/mappingObjects.html>

"Objektrelationale Persistenzlösungen" S. 118 dotnetpro 11/2004 Schwerpunktheft O/R Mapper dotnetpro 5/2005 "Persistenz in Java/J2EE und .NET" dotnetpro 3/2004

Das ganze sollte in einer arbeitsgruppen-relevanten Anwendungsdomäne implementiert werden, um eine Beispielanwendung zur Evaluation zu haben.

Als Beispielanwendung sind folgende Szenarien angedacht aber nicht darauf beschränkt:

1.

Verwaltung aller gestellten Übungsaufgaben und Klausuren zu den VL der Arbeitsgruppe

- in welchem Semester wurde welche Aufgabe gestellt
- wann kam welche Aufgabe in der Klausur dran
- Zuordnung der Aufgaben zu Themenkomplexen, Schwierigkeitsgraden

2.

Bibliotheksverwaltung + Literaturverwaltung eines Lehrstuhls mit Bestellverwaltung, unterschiedlichen Standorten der Bücher, Zuordnung zu Mitarbeitern usw.

Das ganze sieht nach einer klassischen DB-Anwendung aus, aber die Modellierung muss komplett in OO sein, das Mapping dann in die DB "automatisch". Wie gut das bei verschiedenen Frameworks funktioniert, welche Eigenschaften diese Frameworks haben, das gehört zum Gegenstand Ihrer Untersuchung.

Anforderungen:

- Kenntnisse und Programmiererfahrungen mit Datenbanken,
- Kenntnisse in objektorientierter Modellierung,
- Kenntnisse in Datenbankzugriffstechniken wie JDBC, ODBC, ADO, ADO.net