

# Recherchebericht

Projektgruppe: Lebenslaufakte  
Projektleiterin: Nicole Scheeren

## 1. Begriffe

1.1. Lebenslaufakte

1.2 Instandhaltung

1.3 Datenmodell

1.4 Metamodell

1.5 Quellcode

1.6 (Software-)Generator

1.7 MDSD – Modell-Driven Software Development

1.8 relationelle Datenbank

1.9 Website

## 2. Konzepte

2.1 Xpand

2.2 ORM - Object-relational mapping

2.3 Hibernate

2.4 Apache Tapesity

2.5 Ecore

2.6 EMF – Eclipse Modelling Framework

2.7 Topcased

2.8 UML - Unified Modeling Language

2.9 XML- Extensible Markup Language

2.10 SQL- Structured Query Language

## 3. Aspekte

3.1. Design der Weboberfläche

3.2. Strukturierung der Datenbank

## 4. Quellenverzeichnis

## 5. Anhang

# 1. Begriffe

## 1.1. Lebenslaufakte

Entstehende Menge an Dokumenten(Siehe Anhang Abb. 1) bei den Prozessen der Instandhaltung.

## 1.2 Instandhaltung

Sie soll sicherstellen, dass der funktionsfähige Zustand erhalten bleibt oder bei Ausfall wieder hergestellt wird

Systeme zur Durchführung von Instandhaltungsmaßnahmen heißen *IPS-Systeme*.

Instandhaltungsplan – Wichtiger Ablaufplan einer Instandhaltung mit Inspektion, Wartung, Instandsetzung, Reinigung und Sicherheitsarbeiten

Zur Instandhaltung braucht man alle Informationen über ausgeführte und auszuführende Arbeiten:

- Betroffene Anlage
- Art des Defekts
- Bearbeiter
- Verwendetes und noch benötigtes Material
- Dauer
- Kosten

Hauptziele:

- Erhöhung der Anlagenverfügbarkeit
- Reduzierung der Instandhaltungskosten
- richtige Verrechnung der Instandhaltungskosten auf die Instandhaltungsobjekte

Instandhaltung über mobile IT-Technologien – unmittelbare Verfügbarkeit der Daten notwendig für schnellen, einfachen Zugriff auf die Daten über mobile Endgeräte (Bsp.: Laptop)

## 1.3 Datenmodell

Ein Datenmodell definiert und abstrahiert die in einem Softwaresystem zu verarbeitenden Daten. Es spezifiziert ferner die zu verwaltenden Objekte in ihren Eigenschaften und/oder Funktionalitäten.

## 1.4 Metamodell

Metamodelle beschreiben Aspekte und/oder Strukturen zur Datenmodellierung. Sie bilden daher so etwas wie ein Modell über ein Modell.

## 1.5 Quellcode

Quellcode ist für Menschen lesbarer Text, welcher von einem Programm in für eine Maschine ausführbare Maschinensprache übersetzt werden kann.

## 1.6 (Software-)Generator

Ein Generator ist eine Software zur automatisierten Erzeugung von Dokumenten und/oder fehlerfreien lauffähigem Quellcode, ausgehend von formalen Modellen und/oder Sprachen. Ziel dieses Projektes ist es einen solchen Generator zu erstellen, welcher ausgehend von UML 2 Datenmodellen ein Datenbankschema, Eingabemasken und eine Webanwendung erstellt.

## 1.7 MDSD – Modell-Driven Software Development

MDSD bezeichnet ein Prinzip der Softwaretechnik, bei der ausgehend von formalen Modellen automatisiert Software erzeugt wird. Dies ersetzt aufwändige und sich wiederholende Implementierungsprozesse und ermöglicht einen Fokus auf klare Abstraktion und Modellierung. Auch soll durch die automatisierte Entwicklung die Qualität der Software gesteigert werden. MDA (modell driven architecture) bezeichnet die notwendige Spezifikation der Infrastruktur für MDSD. So ist die Entwicklung von Abstraktionssprachen, Softwaregeneratoren und Interpretern Kernstück der MDA.

## 1.8 relationelle Datenbank

Eine relationale Datenbank ist eine DB, die auf dem relationalen Schema von Edgar Frank Codd beruht. Abstrakt gesehen stellt die Datenbank eine Sammlung von Tabellen (Relationen) dar, wobei jede Spalte ein Attribut und jede Zeile (Tupel) einen Datensatz darstellt. Operationen auf diesen Relationen sind durch die relationale Algebra wohldefiniert und diese stellt gleichzeitig die Grundlage für SQL dar.

### Datenbanksystem (DBS):

Ein Datenbanksystem ist ein System zur elektronischen Datenverwaltung und setzt sich aus einer Datenbank (DB) und einem Datenbankmanagementsystem (DBMS) zusammen. Die DB stellt die Menge der zu verwaltenden Daten dar und das DBMS die Software zur Verwaltung dieser Daten, d.h. zum Bearbeiten, Abrufen, Erstellen und Löschen von Datensätzen.

## 1.9 Website

Eine Website bezeichnet eine Internetpräsenz im World Wide Web, inklusive aller zugehörigen Ressourcen, welche durch einen Browser aufgerufen und dargestellt werden kann. Ziel dieses Projektes ist ein Generator der u.a. eine entsprechende Website anhand eines Modells generiert, welche sich rein auf die im UML enthaltene Funktionalität beschränkt.

## 2. Konzepte

### 2.1 Xpand

Xpand ist eine Template-Engine, also eine Software, die ausgehend von einer Eingabe und von Dateispezifikationen (Templates) entsprechende Ausgabe - Dateien generiert. Dabei können auch ausgehend von Modellen und mittels EMF-Metamodellen und Templates entsprechende Dateien entstehen. Um einen Codegenerator, wie in unserem Projekt gefordert, zu erstellen sind im Wesentlichen also 2 Dinge notwendig:

1. Definitionen der Metamodelle, welche die Eingabe bilden sollen und
2. Definitionen der Templates für die Dateien, welche erzeugt werden sollen (der Code für die weiteren Frameworks)

Das Benutzen des Generators erfolgt dann einfach durch das Einladen eines Modells und des Starten der Template-Engine. Xpand bietet einige Vorteile, wie beispielsweise die statische Typisierung, die es uns als ungeübte Benutzer erleichtert, Fehler zu vermeiden. Ein weiterer Vorteil wäre die Unterstützung von polymorphic dispatch (die Möglichkeit zwischen verschiedenen überladenen Funktionen erst in Abhängigkeit des Datentyps der Parameter zur Laufzeit zu wählen).

Daher bietet sich Xpand zur Gestaltung des Generators an, der Quellcode und Ein- und Ausgabemasken (Templates) liefert.

### 2.2 ORM - Object-relational mapping

ORM ermöglicht die Abbildung von Daten aus dem objektorientierten Schema in eine relationale Datenbank (und auch die Umkehrabbildung (im Falle von Hibernate)). Dabei werden Objekte mit Attributen und Methoden abgebildet auf Datensätze (Entities) in einem relationalem Modell und die Beziehungen aus dem objektorientierten Schema abgebildet auf Relationen (Relationships).

### 2.3 Hibernate

Hibernate ist ein OpenSource-Framework für Java, welches ORM ermöglicht und damit Persistenz gewährleistet. In diesem Projekt kann Hibernate vor allem in folgenderweise Anwendung finden: Ausgehend von Java-Annotations, die im ersten Schritt durch Xpand zusammen mit dem restlichen Code erzeugt werden, wird das Mapping von Java-Klassen auf Datenbanktabellen gewährleistet. Damit kann die nötige Persistenzschicht der Datenbankanwendung (back-end) realisiert werden. Hibernate unterstützt viele Aspekte des ORM, so bspw. alle drei Formen der relationalen Abbildung von Vererbungsbeziehungen sowie die Abbildung der Java-Datentypen auf SQL-Datentypen.

### 2.4 Apache Tapestry

Tapestry ist ein OpenSource-Framework zur Erzeugung von Webanwendungen in Java, dabei bilden diese keinen eigenen Server, sondern sind nur lauffähig im Kontext eines Servlet-Containers (nur Erzeugung des front-end /Benutzeroberfläche). Dabei baut Tapestry auf die Java Servlet API auf und ist somit kompatibel zu vielen Application Servern. Tapestry ist auch auf Kompatibilität mit Hibernate ausgelegt.

## 2.5 Ecore

Ecore ist eine Eclipse-eigene Metamodellierungssprache und ist verwandt mit dem XML Metadata Interchange Format.

## 2.6 EMF – Eclipse Modelling Framework

EMF ist ein OpenSource-Framework zum MDS. EMF kann ausgehend von einer Modellspezifikation Java Code erzeugen. Dieser Code kann zum Erzeugen, Abfragen und Speichern von Instanzen verwendet werden, die den Spezifikationen des Modells genügen. Dabei wird ebenso die referentielle Integrität gewährleistet. EMF unterstützt des Weiteren auch die Erzeugung von Daten- und Metamodellen. Erzeugte Datenmodell Diagramme werden dabei in Ecore gespeichert.

## 2.7 Topcased

Topcased (Toolkit in Open Source for Critical Applications & Systems Development) ist eine OpenSource-Werkzeugsammlung zur rechnergestützten Entwicklung und bedient sich als Entwicklungsumgebung der Infrastruktur von Eclipse. Es enthält unter anderem Werkzeuge zur Modellierung, Anforderungsanalyse und Implementierung. Topcased unterstützt sowohl UML als auch Ecore.

## 2.8 UML - Unified Modeling Language

UML ist eine graphische, standardisierte Sprache zur Modellierung, Spezifikation, Konstruktion und Dokumentation von Dokumenten zur Softwareentwicklung und von Softwaresystemen (Vers. 2.1.2 ISO/IEC 19505). UML2 erweitert UML u.a. um verschiedene Spezifikationen zur MDS, bspw. um Spezifikationen zu Metamodellen.

## 2.9 XML- Extensible Markup Language

XML ist eine Sprache für hierarchisch strukturierte Textdateien. Zur Serialisierung der Datenmodelle und der Metadaten kommt häufig XML zur Anwendung.

## 2.10 SQL- Structured Query Language

SQL ist eine Sprache zur Realisierung von Datenbankmanagementsystemen. Sie wird verwendet zur Spezifikation von relationalen Datenbanken und zum Einfügen, Abfragen, Bearbeiten und Löschen der Datenbestände.

## **3. Aspekte**

### **3.1. Design der Weboberfläche**

Das Design der Weboberfläche soll ausgelegt sein auf größtmögliche Benutzerfreundlichkeit und Funktionalität. Grafisch aufwendige Darstellungen sind nicht gefordert.

### **3.2. Strukturierung der Datenbank**

Die Strukturierung einer Datenbank ist wichtiger Bestandteil der späteren Instandhaltung und Modernisierung dieser. Aus diesem Grunde ist es wichtig, diese möglichst universell verständlich und zukunftsorientiert zu gestalten.

Zur Realisierung eines DBMS wird bevorzugt auf MySQL zurückgegriffen, da das Team bereits mit MySQL vertraut ist und MySQL eine der bewährtesten Standard-Sprachen zur Gestaltung eines relationalen DBMS darstellt. Inwieweit weitere Frameworks, wie bspw. Hibernate, Einfluss auf den Entwicklungsprozess des DBMS haben ist uns momentan noch nicht bekannt, aber wenn die Möglichkeit zur effizienten Entwicklung besteht, wird auch darauf zurückgegriffen.

## 4. Quellenverzeichnis

Instandhaltung:

<http://de.wikipedia.org/wiki/Instandhaltung>

Schuh, Günther ; Klimek, Gregor: Informations- und Kommunikationstechnologien für die Instandhaltungsplanung und -steuerung. In: Reichel, Jens (Hrsg.): Betriebliche Instandhaltung.

Dordrecht, Heidelberg, London, New York, NY : Springer-Verlag, 2009. – ISBN 978-3-642-00501-5, S. 151–161

Dokumente:

Deutsche Norm, DIN EN 13460: Instandhaltung - Dokumente für die Instandhaltung; Deutsche

Fassung EN 13460:2009

MDSD/MDA:

Gruhn, Volker ; Pieper, Daniel ; Röttgers, Carsten: MDA : Effektives Software-Engineering mit UML2 und Eclipse. Springer-Verlag, 2006 (Xpert.press). – ISBN 978-3-540-28744-5

Stahl, Thomas ; Völter, Markus ; Efftinge, Sven ; Haase, Arno: Modellgetriebene Softwareentwicklung: Techniken, Engineering, Management. 2. Aufl. s.l : dpunkt.verlag, 2007. – ISBN 978-3898644488

relationale Datenbanken: Vorlesung von Prof. Erhard Rahm <http://dbs.uni-leipzig.de/de/stud/2012ws/dbs1>

XML: <http://www.w3.org/XML/>

UML: [http://www.omg.org/technology/documents/modeling\\_spec\\_catalog.htm#UML](http://www.omg.org/technology/documents/modeling_spec_catalog.htm#UML)  
<http://www.uml.org/>

Xpand: Peter Friese: Getting started with Code Generation with Xpand  
(<http://www.peterfriese.de/gettingstarted-with-code-generation-with-xpand/>)

<http://www.eclipse.org/modeling/m2t/?project=xpand>

Lars Vogel: <http://www.vogella.com/articles/Xpand/article.html>

EMF:

<http://www.eclipse.org/modeling/emf/>

Lars Vogel: <http://www.vogella.com/articles/EclipseEMF/article.html>

Maximilian Koegel: <http://eclipse-source.com/blogs/tutorials/emf-tutorial/>

Topcased: <http://www.topcased.org/>

Hibernate: <http://www.hibernate.org/>

Tapestry: <http://tapestry.apache.org/>

# 5. Anhang

DIN EN 13460:2009-08  
EN 13460:2009 (D)

