



# *SOFTWARE ARCHAEOLOGY*

RAIDERS of the ~~LOST ARK~~ CODE

# Chapters

- Crew
- Trailer
- Starring ...
- Set 1 .. 3
- Credits

HOLLYWOOD PRODUCTION FINEST CODING  
SINCE 1998

DIRECTOR Lars Martin

CAMERA [www.smb-tec.com](http://www.smb-tec.com)

DATE

SCENE

TAKE

1974

SMB GmbH

15

# Chapters

- Crew
- Trailer
- Starring ...
- Set 1 .. 3
- Credits

**Migration**

**10 Jahre**

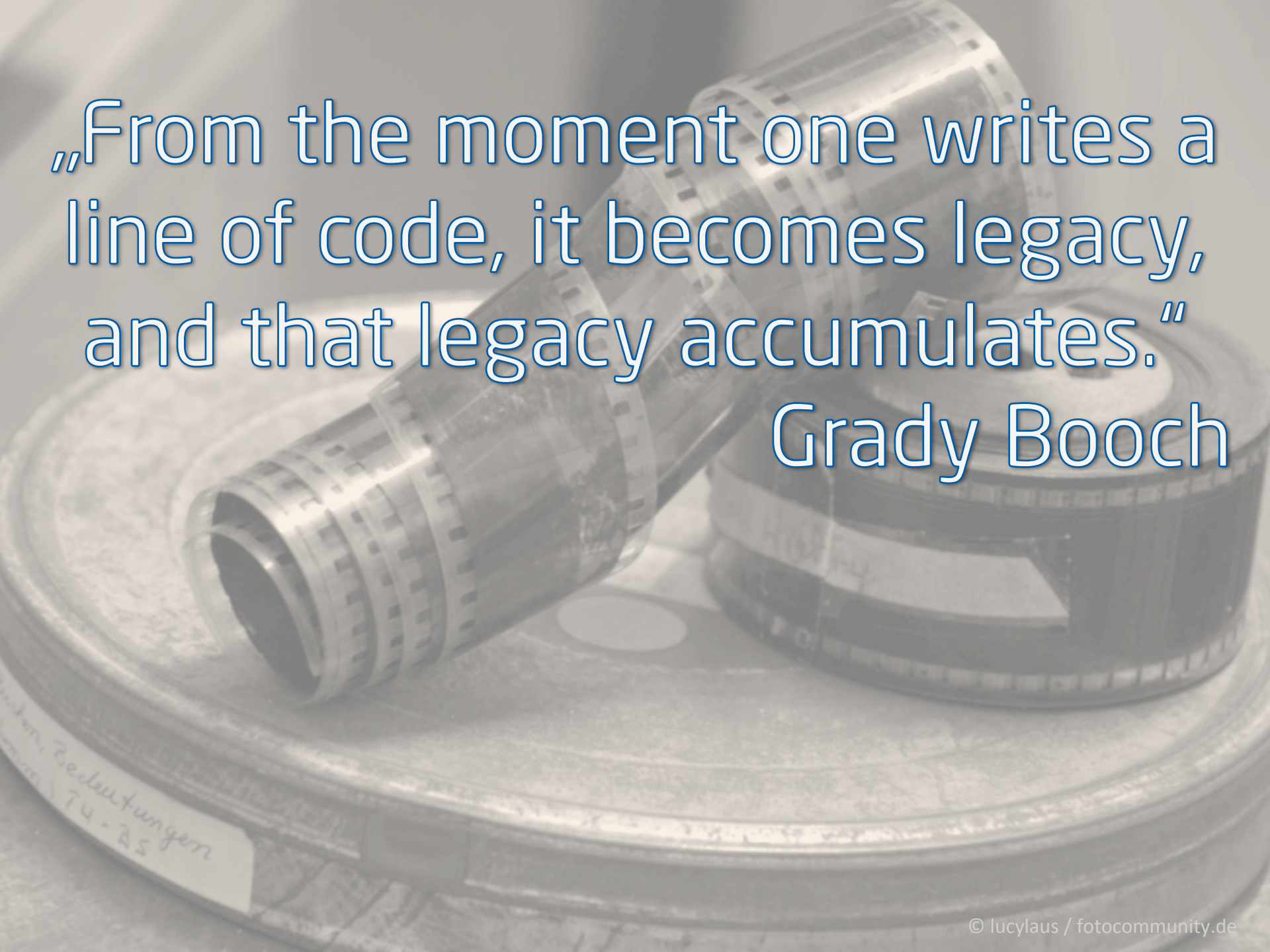
**datenbanklastig**

**ORM**

**java.sql.\***

**neues Team**

- **Auf welche Tabellen / Entitäten greift ein zu migrierendes Module zu?**
- **Greift Business Logik des zu migrierenden Moduls auch auf Tabellen anderer Module zu?**
- **Wenn ja, welche Module betrifft das und wurden diese bereits migriert?**
- **Lässt sich das automatisieren?**

A stack of film reels is shown, with a roll of film resting on top. The film is partially unspooled and has some markings on it. The background is blurred, suggesting a shallow depth of field. The overall tone is monochromatic and slightly desaturated.

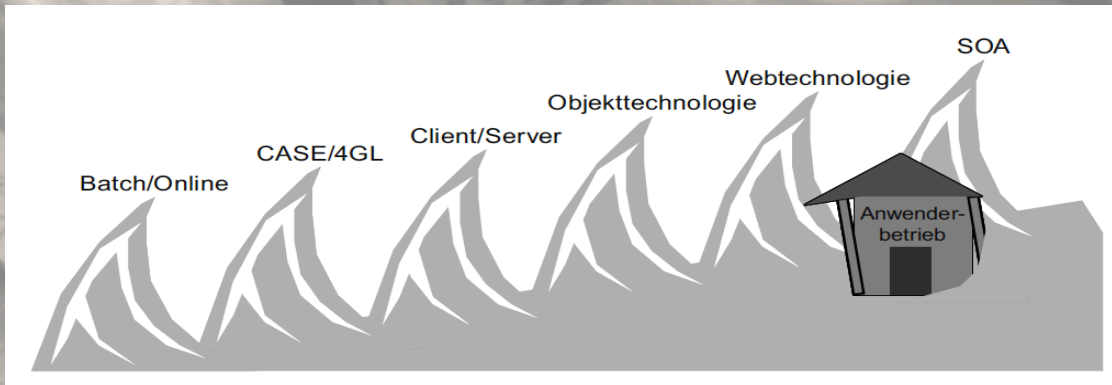
„From the moment one writes a line of code, it becomes legacy, and that legacy accumulates.“

Grady Booch

# Legacy:

- veraltet, nicht zwangsläufig alt
- obsolete Technologien und Methoden
- oft in kritischen Geschäftsbereichen

**Software gilt etwa nach 5 Jahren als legacy**



„From the moment one writes a line of code, it becomes legacy, and that legacy accumulates.“

Grady Booch



1. Continuing Change - nur stetige Weiterentwicklung erhält Akzeptanz
2. Increasing Complexity - Weiterentwicklung erzeugt Komplexität, es sei denn ...

Lehman & Bélády

Neubau

verstecktes Wissen

schlechte Abschätzung

Ablösung

„Fehlerteufel“

Akzeptanz

Migration

„alter Wein

in neuen

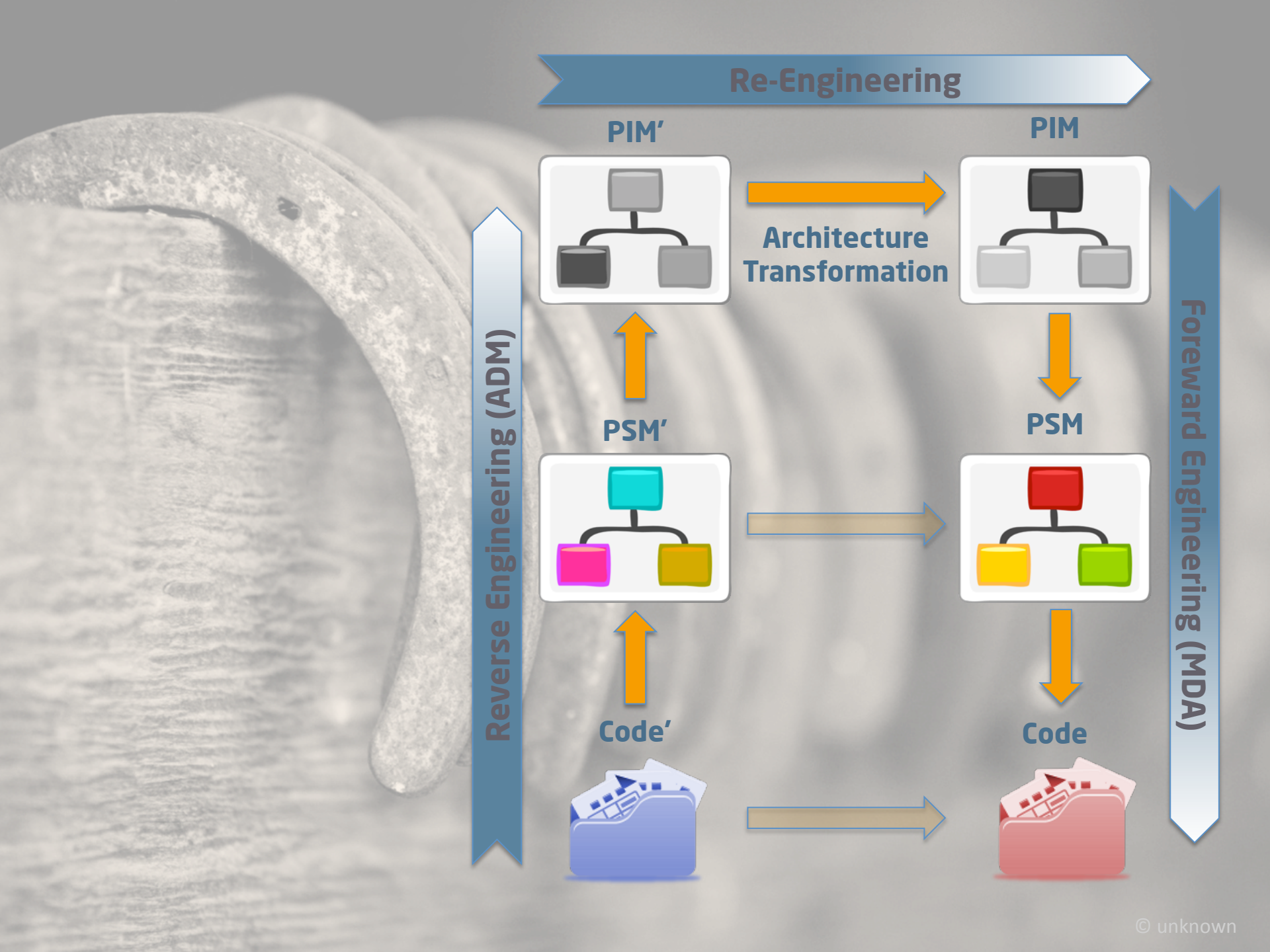
Schläuchen“

.. common sense  
solution to the  
legacy problem.

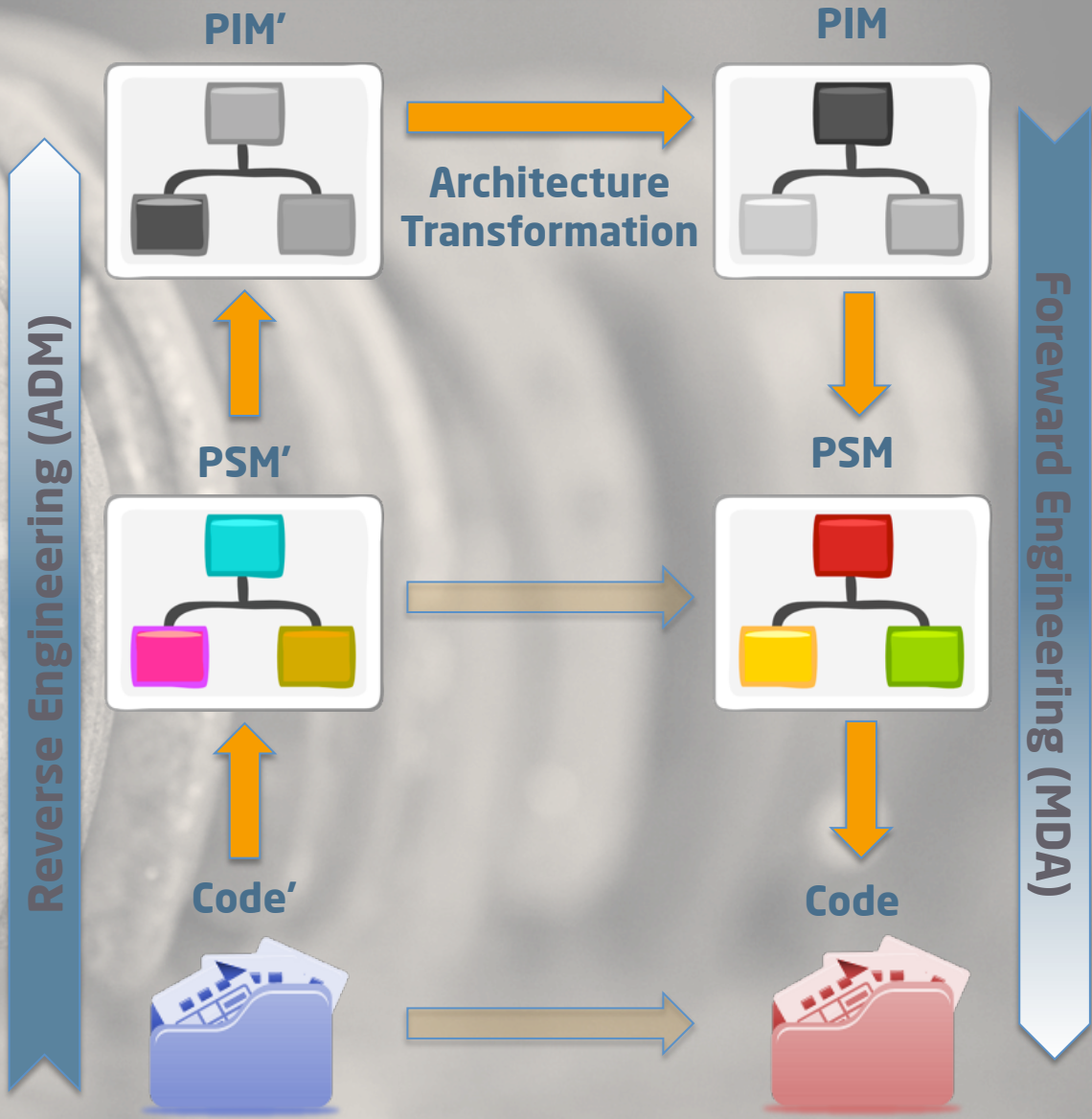
1. Continuing Change - nur  
stetige Weiterentwicklung  
erhält Akzeptanz

2. Increasing Complexity -  
Weiterentwicklung erzeugt  
Komplexität, es sei denn ...

Lehman & Bélády



**Re-Engineering**



Qualität ↗

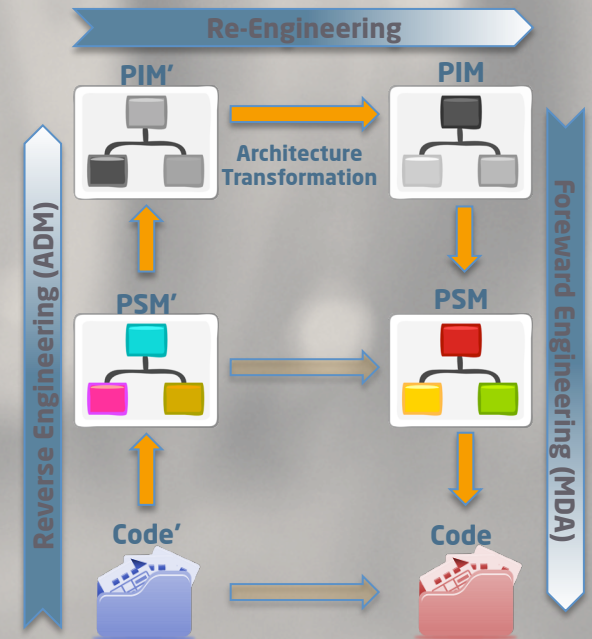
Fehler ↘

Wartbarkeit ↗

Information & Wissen ↗



## Horseshoe Model



- Grep, AWK, Shellskripte, ...
- Debugger, ...
- Decompiler, ...
- **JaMoPP, Snoop, MoDisco, ...**



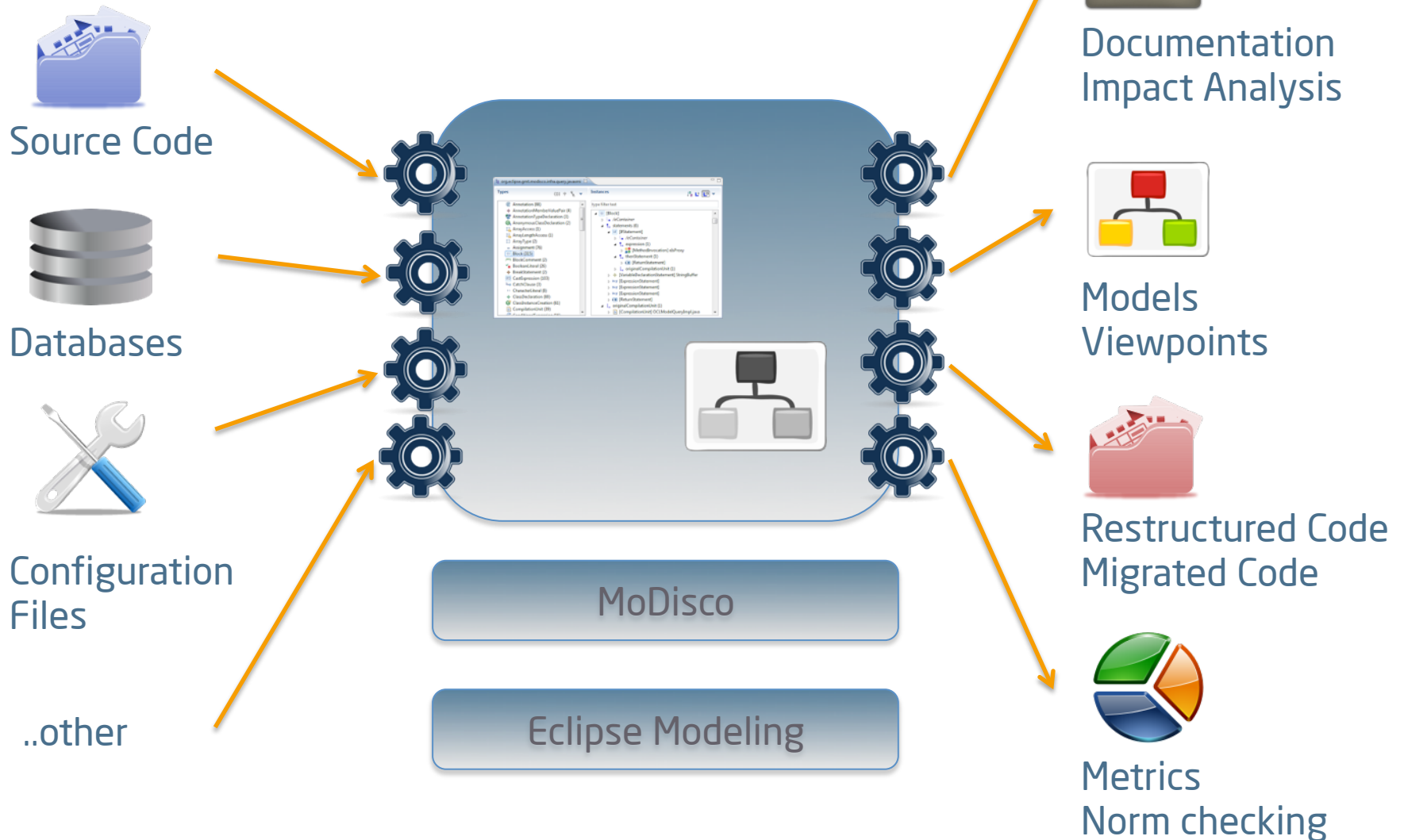
# Chapters

- Crew
- Trailer
- Starring ...
- Set 1 .. 3
- Credits

# Starring...

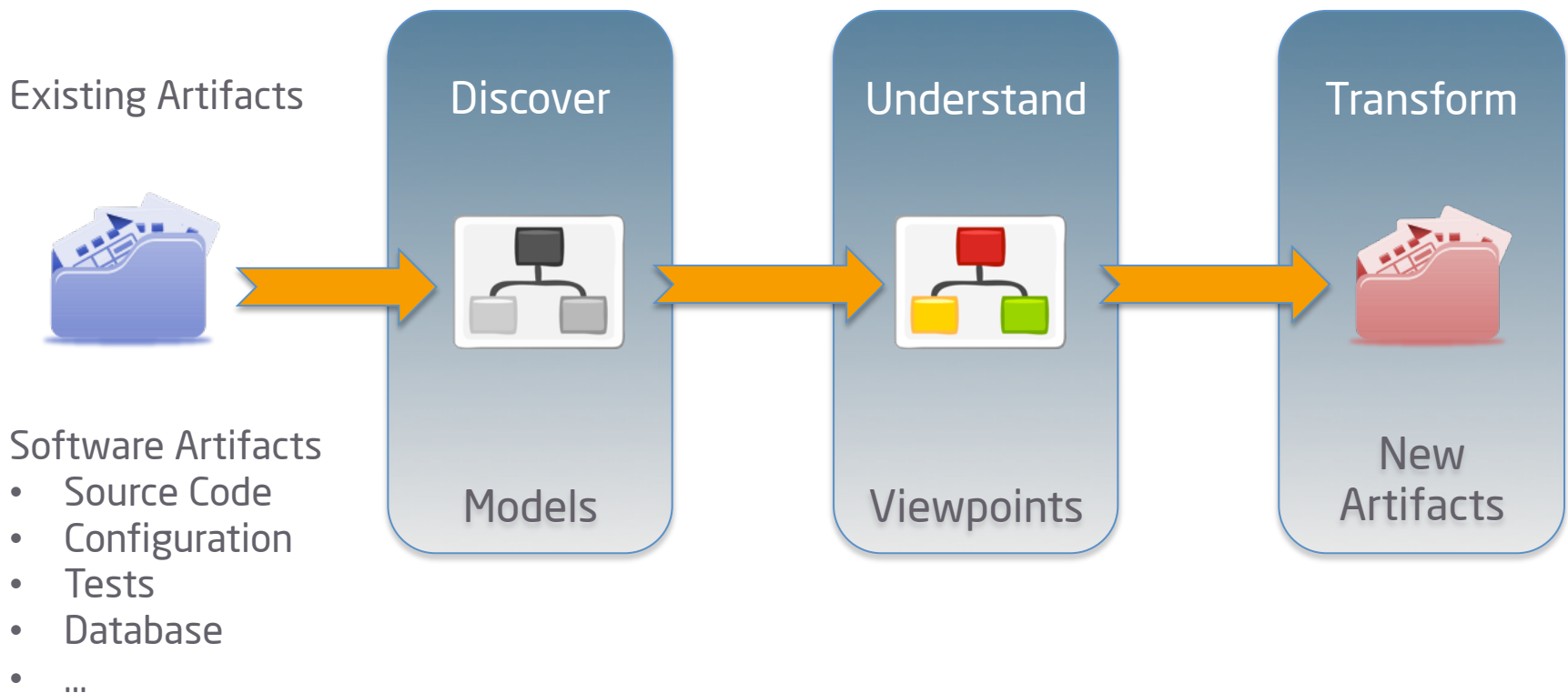


# MoDisco - Concepts





# MoDisco - Modernization Phases



# MoDisco – Use Cases (1/2)

**BPM**

**Data Flow**

**ADF**

**JUnit<sup>3/4</sup>**

**Coverage**

**E4**

**SOA**

**JEE/.NET**

**JavaFX**

**EJB<sup>2/3</sup>**

**...**

# MoDisco – Use Cases (2/2)



- **Qualitätssicherung** (Metriken, ...)
- **Dokumentation** (Struktur, Datenfluss, ...)
- **Refaktorisierung** (Design Pattern, Standards, ...)
- **Migration** (Framework, Architektur, ...)

# Chapters

- Crew
- Trailer
- Starring ...
- Set 1 .. 3
- Credits

# Klappe - 1.

## „Java EE“





# Java Pet Store

Pets

Cats

Dogs

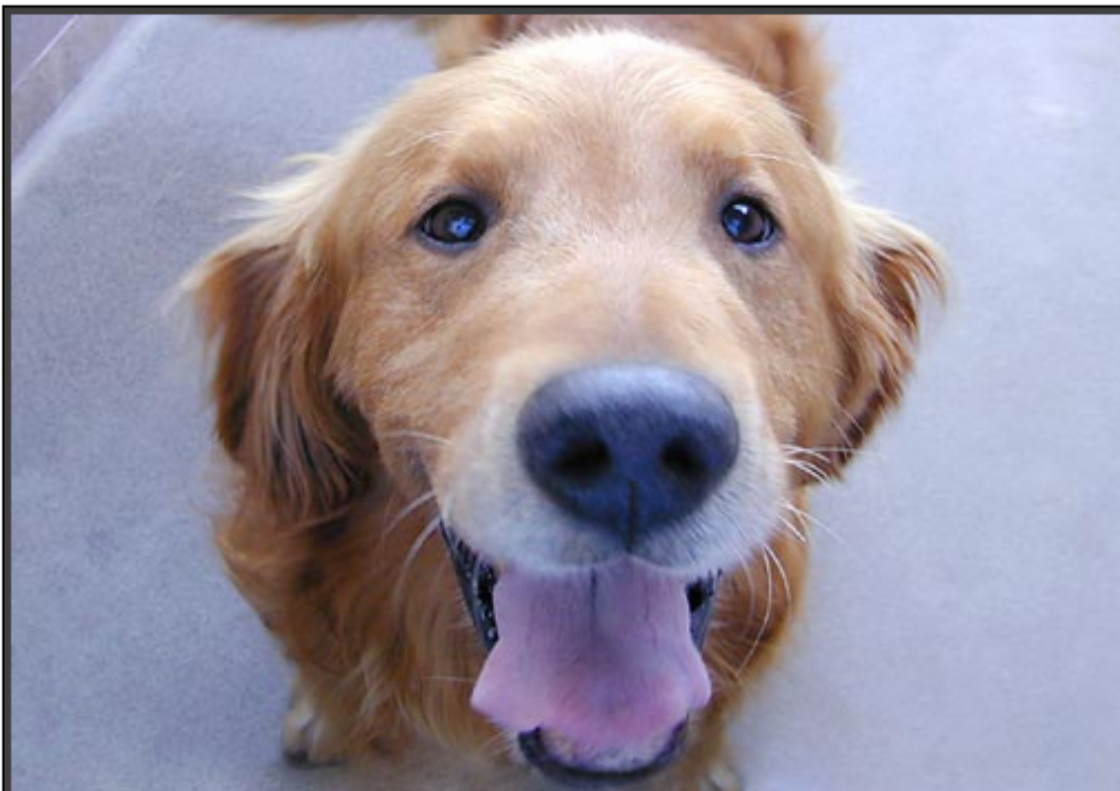
Medium Dogs

Small Dogs

Birds

Reptiles

Fish



Beach Dog

*Flag as inappropriate*

Average



\$250.00

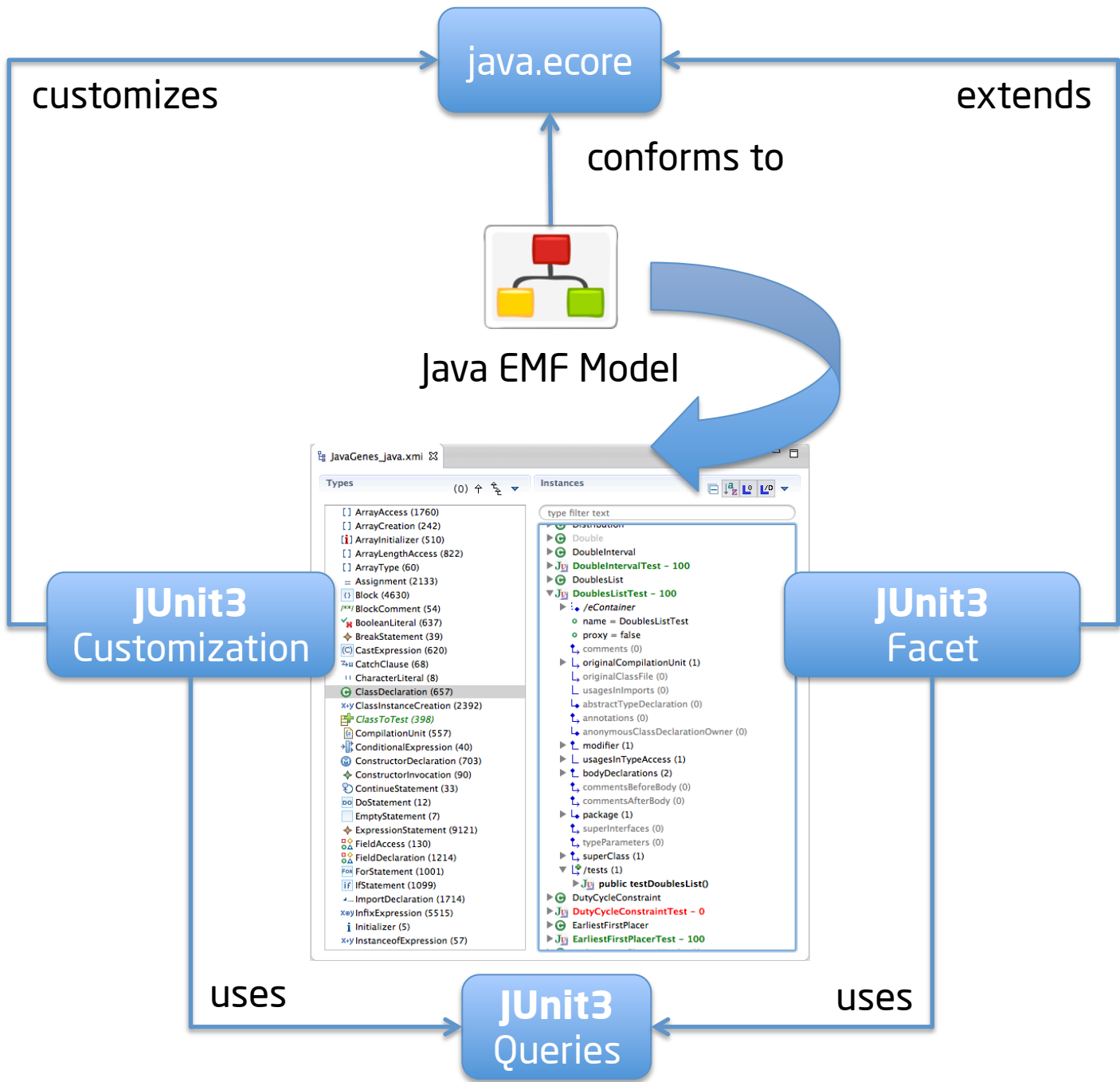


A great dog to lay in the Sun with, chases a frisbee like a champ, and ...



Two professional studio lights, likely HMI or tungsten units, are mounted on black stands. The lights are illuminated, casting a warm, yellowish glow. They are positioned in the upper right quadrant of the frame against a plain, light-colored background. The stands have various adjustment knobs and cables attached to them.

# Klappe -2. „JUnit3/4“







Search

junit3

Search

- Repositories 6
- Code 1,994
- Issues 43
- Users 1

Languages

- Java 5
- Shell 1

Advanced Search Cheat Sheet

**tackley/sbt-junit3-demo** Java ★ 1 0

Demonstration of how sbt's test fingerprinting doesn't work correctly for java code.

Last updated a year ago

**hoomb/JUnit4C**

Convert JUnit3

Last updated 6

**dthommes/**

Android Libra

JUnit3 tests a

Last updated 8 months

**FranciscoBorges/junit3ToJuni** Shell ★ 1 1

Shell code to migrate Java sources from to JUnit4.

Last updated 20 days ago

**jesg/junit3Tojunit4** Java ★ 0 0

Eclipse plugin that converts JUnit3 source files to JUnit4 source files.

Last updated 2 months ago

**sonatype/http-testing-harness** Java ★ 5 2

Testing library for JUnit4 and TestNG for handling multiple server fixture setups for test methods



Two professional studio lights are mounted on black stands. The lights are rectangular with a grid pattern over the lens and are illuminated, casting a warm glow. They are positioned on the right side of the frame against a plain, light-colored background.

# Klappe -3. „Eclipse“

# SimRel/Simultaneous Release Requirements

< SimRel

## Contents [hide]

### 1 The Eclipse Simultaneous Release Requirements

#### 1.1 Normal release requirements ... but need to be consistent

##### 1.1.1 State intent early (M4)

##### 1.1.2 Formal (standard format) plans, etc.

###### 1.1.2.1 Target Environments

###### 1.1.2.2 Compatibility with Previous Versions

##### 1.1.3 IP Documentation and Logs (RC1)

##### 1.1.4 Release Review and compliance

#### 1.2 Extra requirements, to be in common release

##### 1.2.1 Integrate Early and Often

##### 1.2.2 Communication

##### 1.2.3 Required Bundle forms and formats

###### 1.2.3.1 Version Numbering (tested)

###### 1.2.3.2 OSGi bundle format

###### 1.2.3.3 Execution Environment (tested)

###### 1.2.3.4 Signing (tested)

###### 1.2.3.5 Jarred Bundles

###### 1.2.3.6 License text consistency (tested)

##### 1.2.4 Re-use and share common third party code

##### 1.2.5 Provide optimized p2 repository (partially tested)

##### 1.2.6 Branding

##### 1.2.7 Do No Harm

### Required Bundle forms and formats

#### Version Numbering (tested)

Projects must use 4-part version numbers following the common semantics described in the [Eclipse version numbering](#) document.

#### OSGi bundle format

All plug-ins (bundles) must use the true bundle form. That is, provide a manifest.mf file, and not rely on the plug-in being 'translated' into a manifest.mf file at initial startup. With that, empty plugin.xml files in the presence of a manifest.mf file will not be included in a bundle. (For some old history, see [bug 130598](#).)

#### Execution Environment (tested)

All plug-ins (that contain Java code) must correctly specify their [Bundle Required Execution Environment \(BREE\)](#) in their manifest.mf file. BREE must specify which version of Java they are used with.

#### Excel in National Language support

##### User Interface

##### Use the best NL Java Libraries (ICU4J)

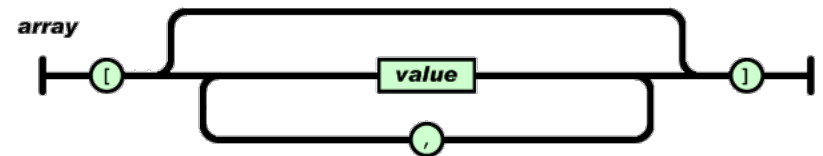
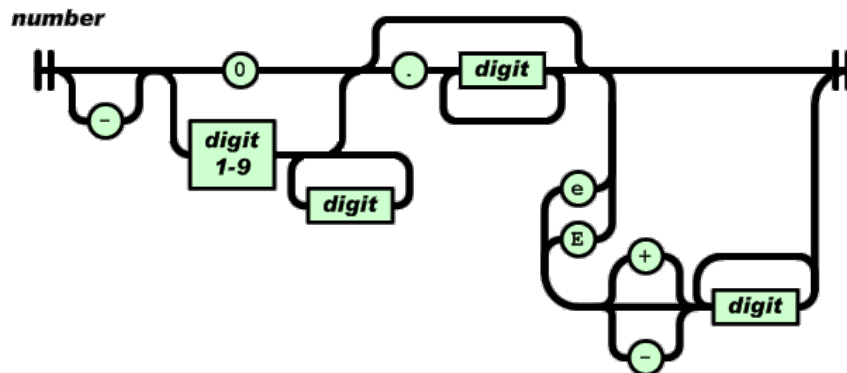
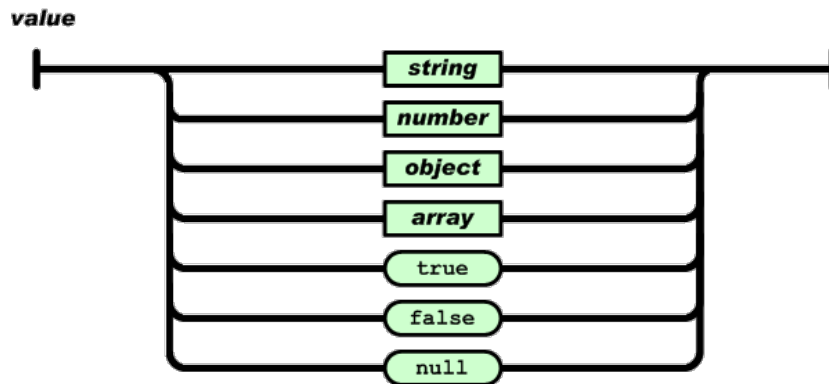
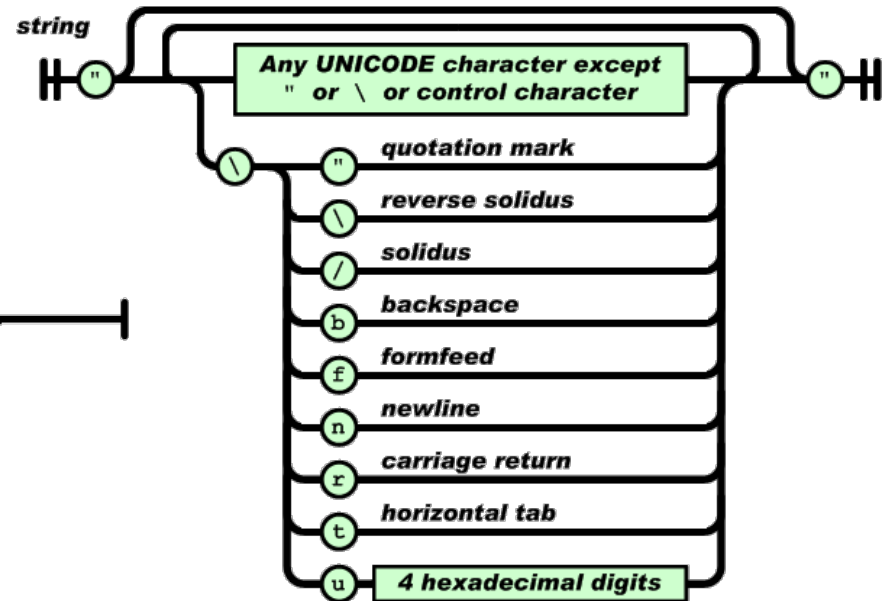
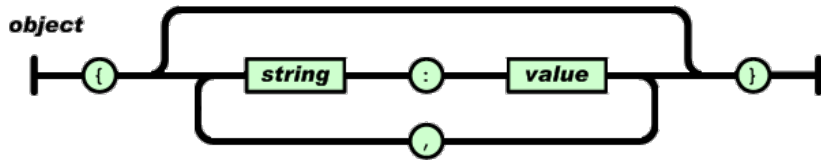
Projects should use [ICU4J](#), where they have "user strings", in order to excel in NL support. (The latest version of ICU4J bundles will be in Orbit).

##### Use the most efficient message bundles

Projects should use [Eclipse message bundles](#) unless there are technical reasons not to, since (with these are known to be more memory efficient than some other forms of handling UI-related strings. The goal is to reduce and adoption is some adopters end up using thousands of bundles ... so, that's a lot of strings!

Two professional studio lights are mounted on black stands against a plain, light-colored background. The lights are illuminated, casting a warm, yellowish glow. The light on the left is slightly higher and angled more towards the viewer, while the light on the right is lower and angled more to the side. Both lights have a grid pattern over the lens and are connected to power cables.

# Klappe -4. „JSON“




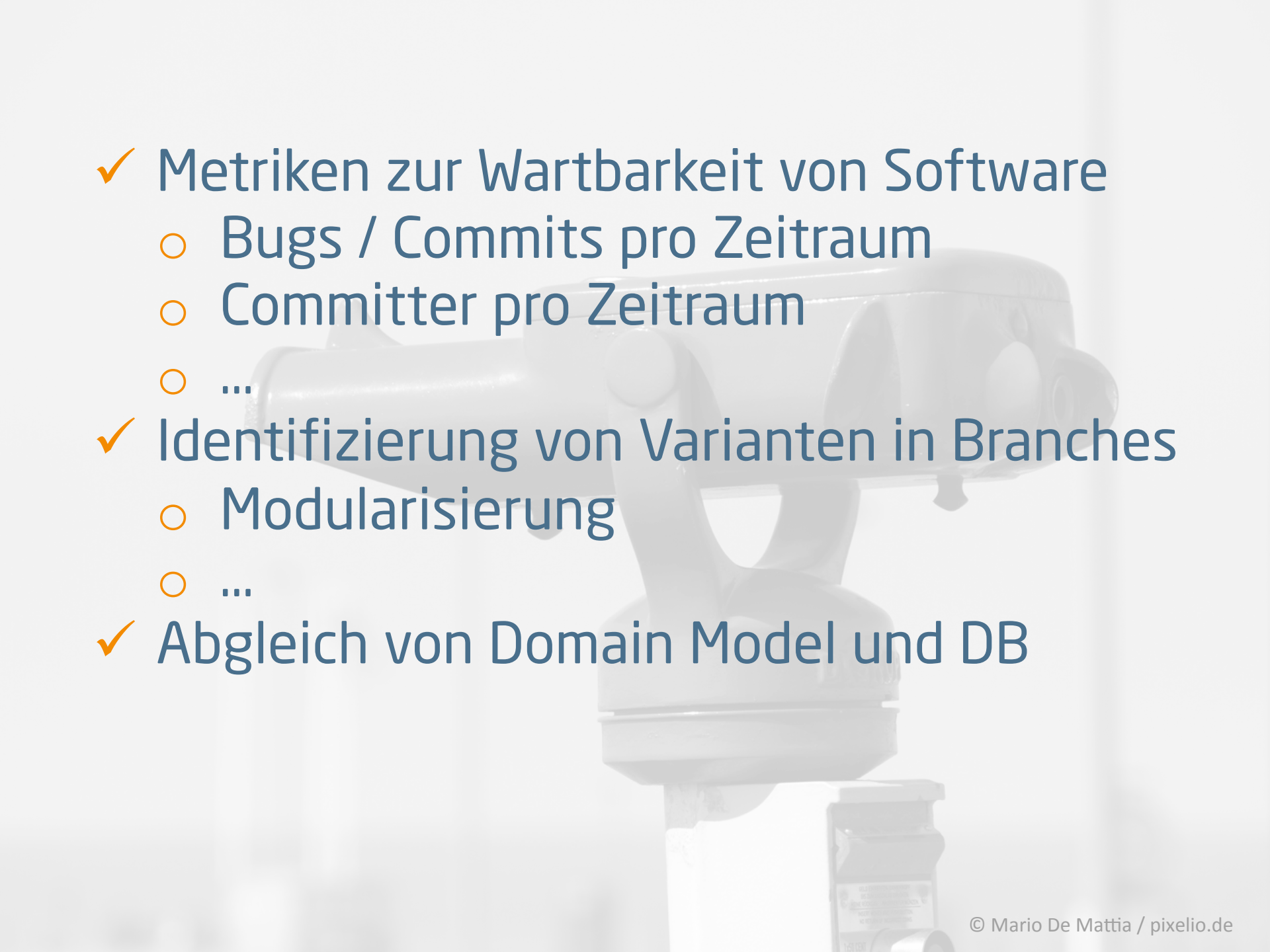
**JSON**  
JavaScript Object Notation

# Chapters

- Crew
- Trailer
- Starring ...
- Set 1 .. 3
- Credits

- 
- A grayscale image of a hand pointing upwards, serving as a background for the left column of text.
- ✓ Anpassbarkeit
    - Metamodels
    - M2M, M2T
    - ...
  - ✓ umfangreich
    - Discoverer
    - Queries
    - ...
  - ✓ Workflow
  - ✓ DSL (Xtext, ...)

- 
- A grayscale image of a hand pointing downwards, serving as a background for the right column of text.
- ✓ (Fokus auf Java)
  - ✓ Einstiegshürden
    - Refactoring
    - ...
  - ✓ Werkzeugkasten

- 
- ✓ Metriken zur Wartbarkeit von Software
    - Bugs / Commits pro Zeitraum
    - Committer pro Zeitraum
    - ...
  - ✓ Identifizierung von Varianten in Branches
    - Modularisierung
    - ...
  - ✓ Abgleich von Domain Model und DB



- [www.eclipse.org](http://www.eclipse.org)
- [www.eclipse.org/modeling/emf](http://www.eclipse.org/modeling/emf)
- [www.eclipse.org/Modeling/ModelDisco](http://www.eclipse.org/Modeling/ModelDisco)
- [www.omg.org/technology/kdm](http://www.omg.org/technology/kdm)



# Master Thesis / Abschlussarbeit - Software Engineering

## Aufgabengebiet:

- Entwicklung von Methoden zur Informationsgewinnung
- Implementierung der Methoden mittels Eclipse basierter Technologien

## Studienbereiche:

- Informatik und -Informatik



[lars.martin@smb-tec.com](mailto:lars.martin@smb-tec.com) - +49-(0)151-42607011