

Code-Quality-Management

Info-Point

Urs Frei

[Inhalt]

- Ziel der Analyse
- Messen der Qualität (QBL)
- Eine Messgrösse als Bsp.
- Analysierte Software
- Tool zur Analyse (Sissy)

[Ziel

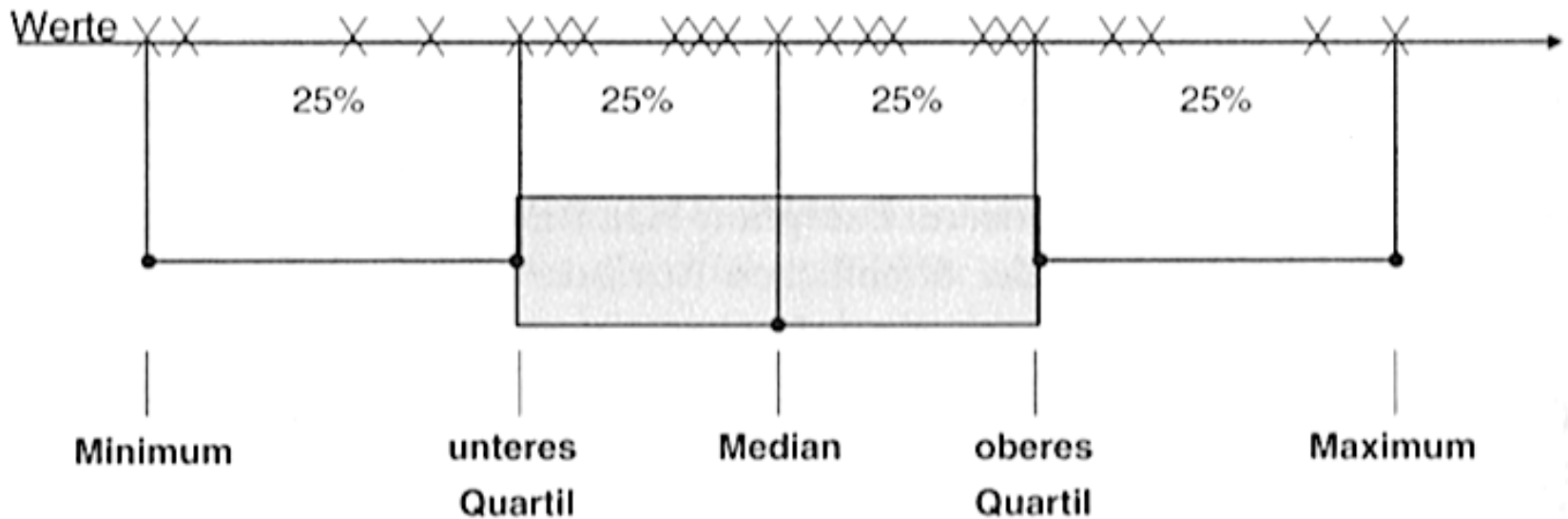
- Besser Code → Bessere Softwarequalität
- Codequalität beurteilen
 - Objektiv durch messen (Metriken)
 - Anomalien identifizieren
 - Bsp.: 1Mio LOC in zwei Klassen
- Weg vom Bauchgefühl hin zu klaren Fakten

[Code-Quality-Index]

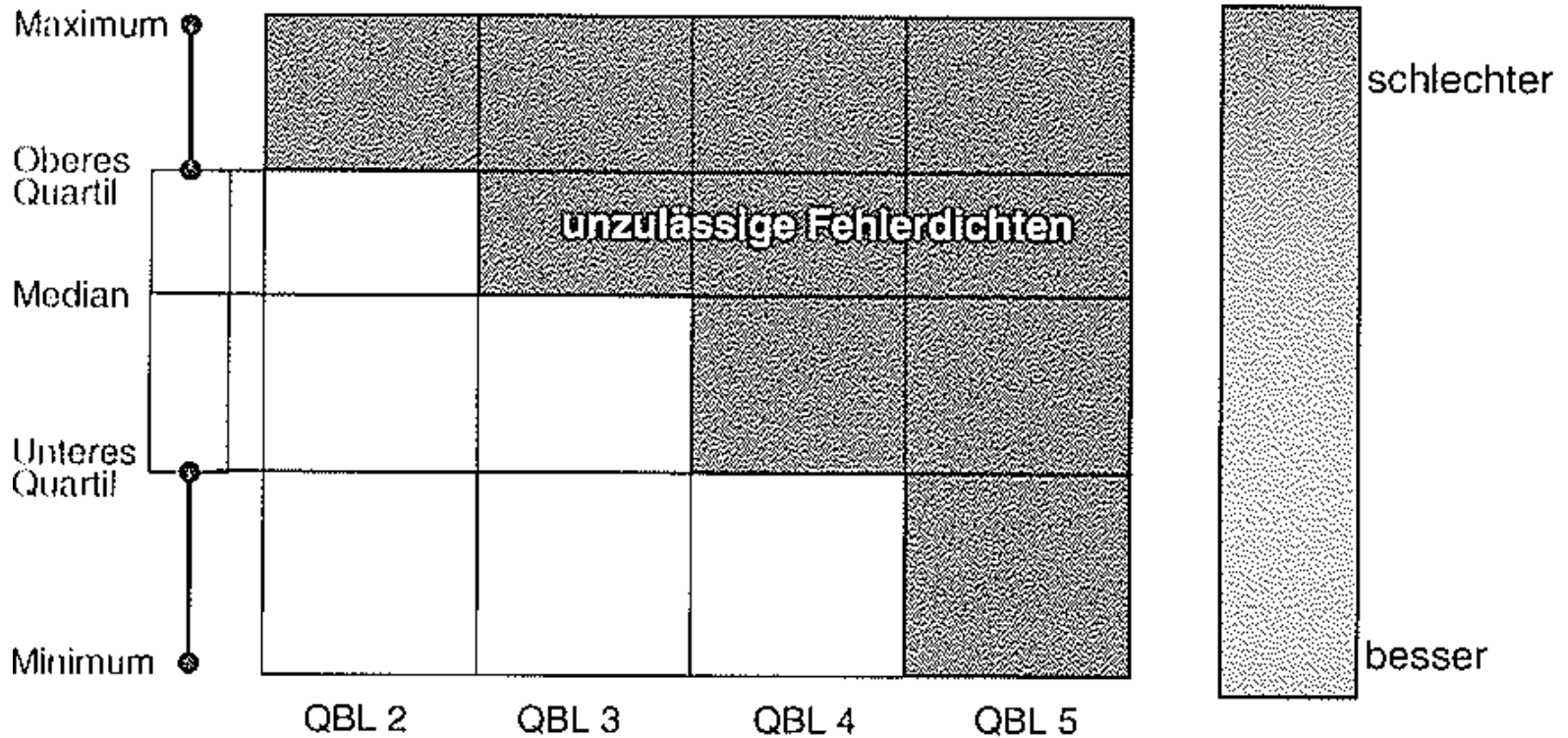
- Wie bei Hotels sollen Sterne über Qualität Aufschluss geben
- Einheit sind Quality Benchmark Levels (QBL)
 - Rudimentary (Level 1)
 - Basic (Level 2)
 - Extended (Level 3)
 - Advanced (Level 4)
 - Complete (Level 5)

[Wie wird gemessen]

- 120 Projekte Analysiert



[Schwellwerttunnel]



[QBL 1: Rudimentary]

- Code ist
 - Kompilierbar
 - Linkbar
 - Code syntaktisch korrekt

[QBL 2: Basic]

- Schwergewicht auf:
 - Analysierbarkeit
 - Stabilität
- Wird QBL 2 nicht erfüllt
→ schwerwiegende Risiken in wirtschaftlicher Weiterentwicklung
- Schwellwerttunnel: Oberes Quartil

[QBL 3: Extended]

- Gute Anpassbarkeit
- Zukunftsfähigkeit gut
- Schwerpunkt:
 - Analysierbarkeit
 - Stabilität
 - Zeitverhalten
 - Verbrauchsverhalten
- Schwellwerttunnel: Median

[OBL 4: Advanced]

- Explizit für Weiterentwicklungen vorbereitet
- Schwerpunkt:
 - Prüfbarkeit
 - Modifizierbarkeit
- Schwellwerttunnel: Unteres Quartil

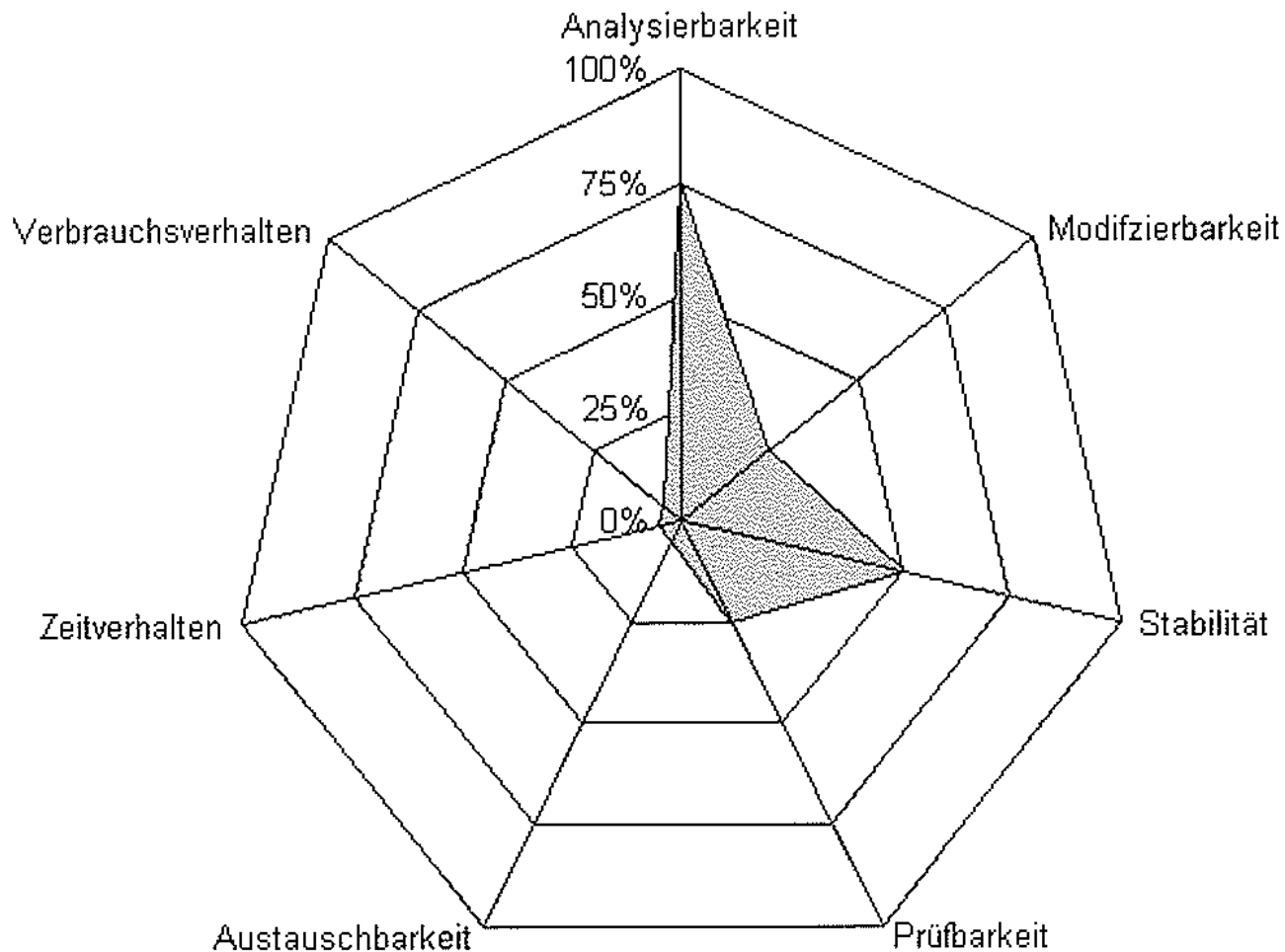
[QBL 5: Completet]

- Lässt keine Wünsche offen → Perfekter Code
- Schwergewicht:
 - Austauschbarkeit
- Schwellwerttunnel: Minimum

Bsp Indikator: Attributüberdeckung

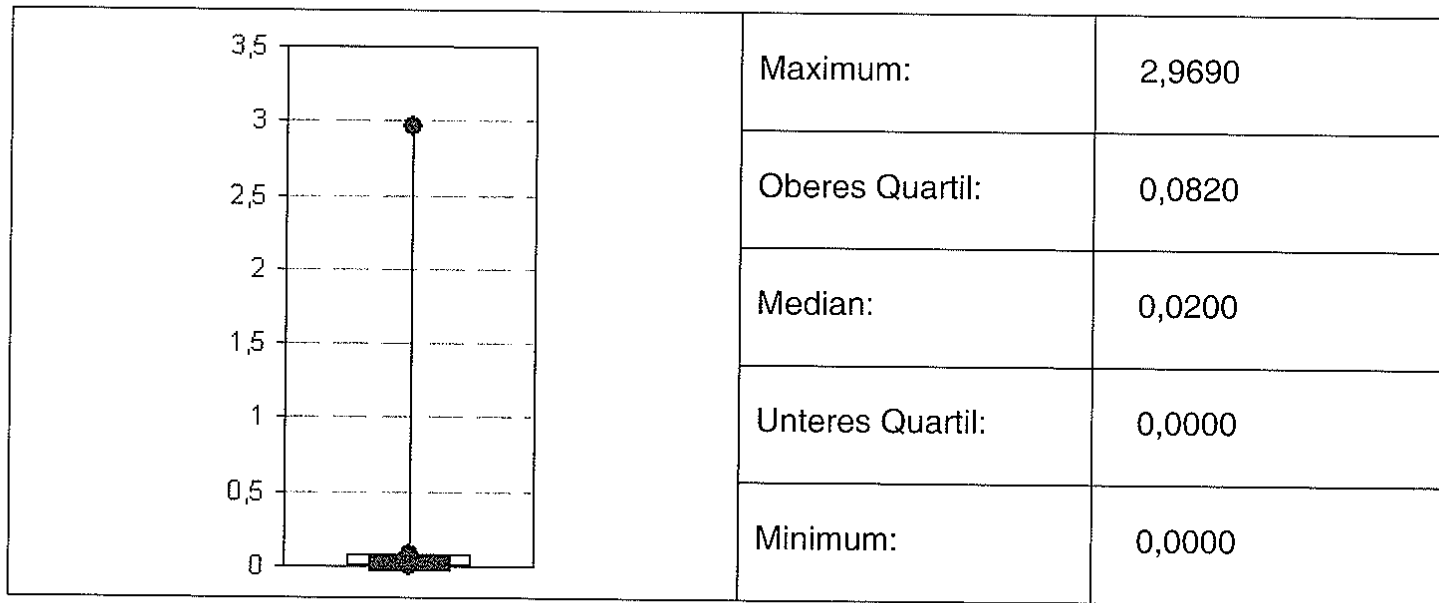
- Definition:
Der Name eines nichtstatischen Attributes wird in mindestens einer direkten oder indirekten Unterklasse für ein anderes Attribut erneut verwendet. Hierbei ist die Sichtbarkeit und der Typ der betrachteten Attribute irrelevant und ein Übereinstimmung nicht erforderlich
- QBL 3

Attributüberdeckung (2)



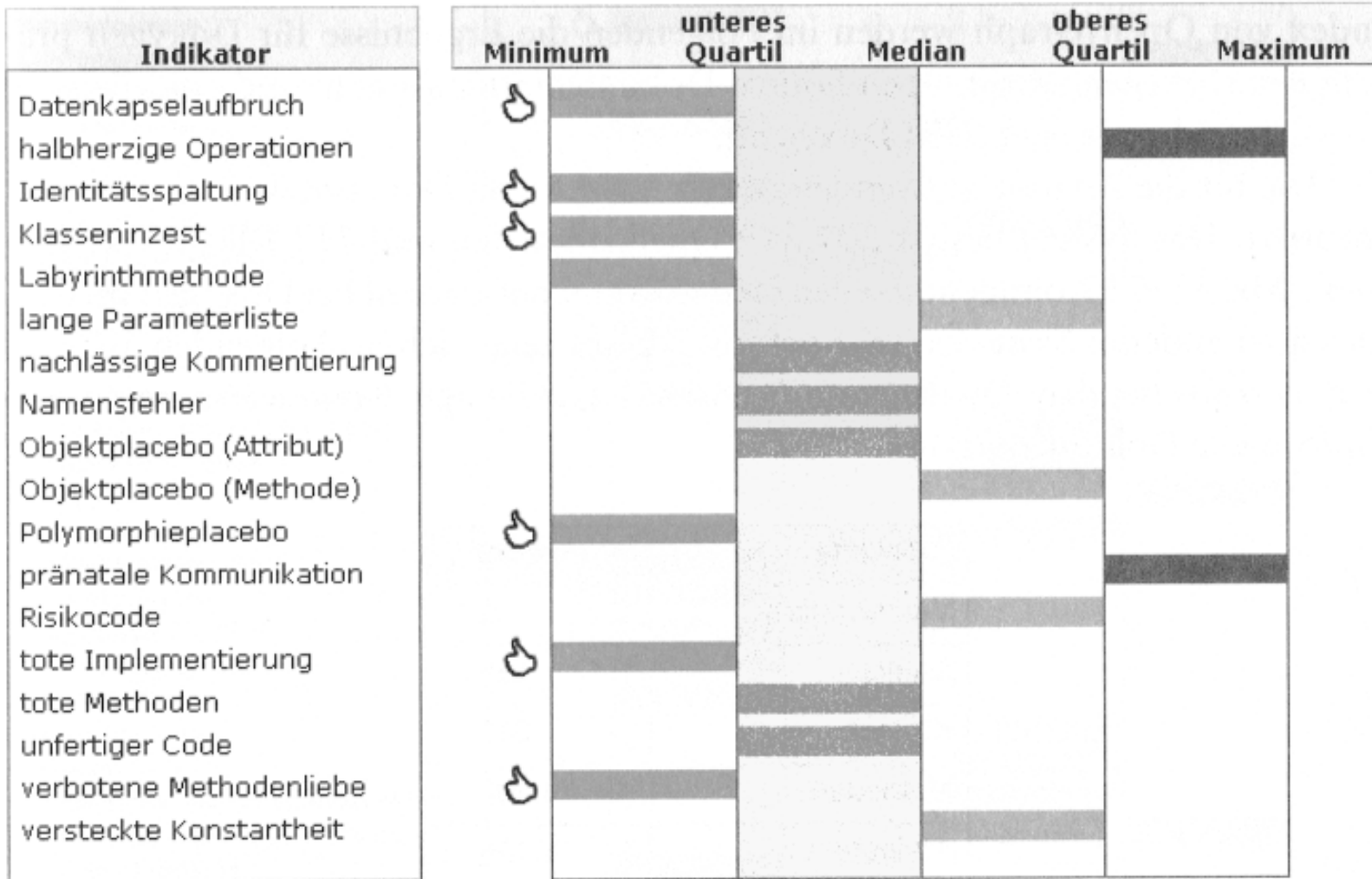
Attributüberdeckung (3)

QBL-Grenzwerte für Java



Angaben beziehen sich auf Verletzungen pro 1.000 LOC.

Analyse OpenJGraph 0.92



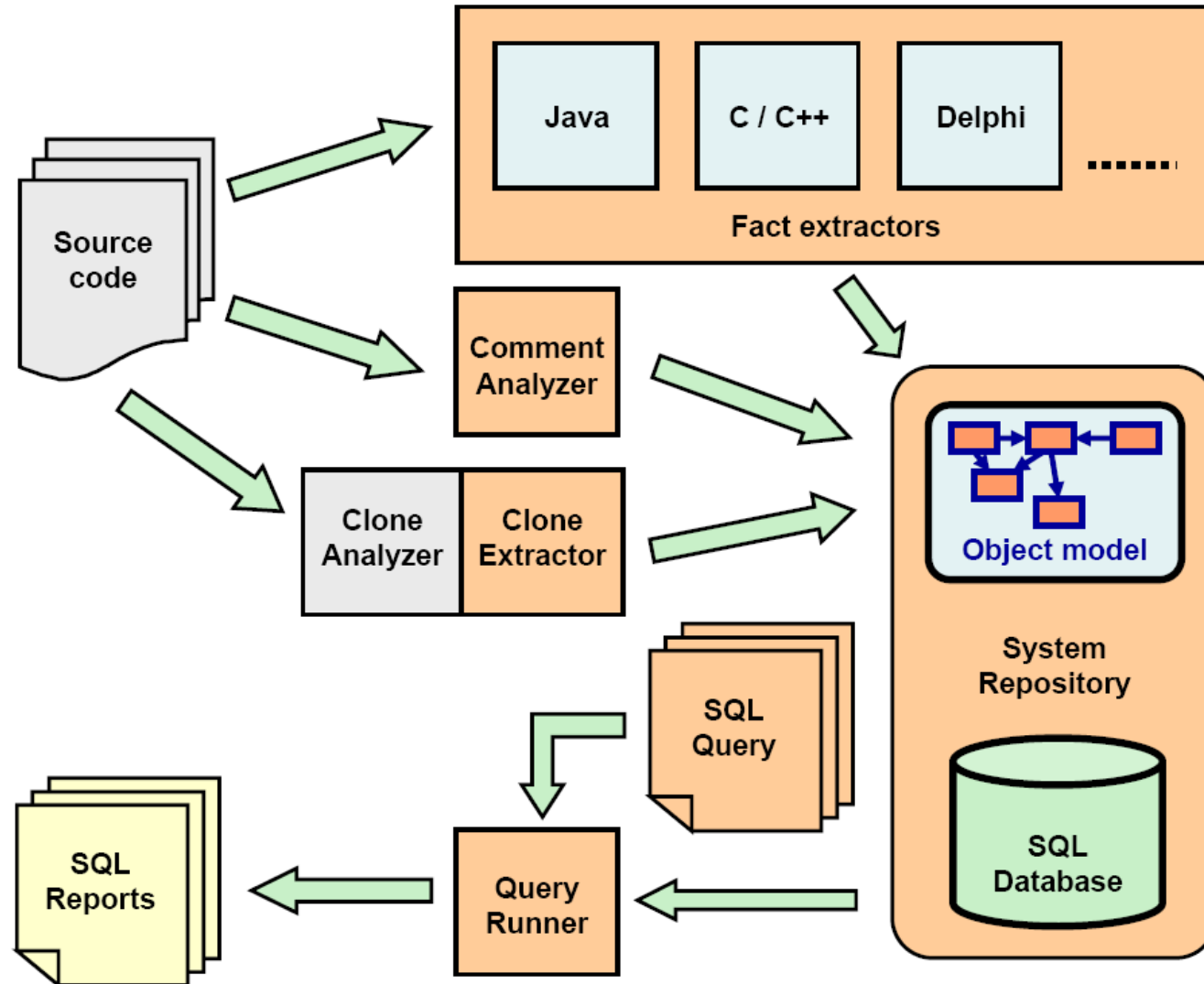
[Analyse Doxygen 1.4.6]

- Level QBL 1
- Durchgefallen bei:
 - Labyrinthmethoden
 - Nachlässige Kommentierung
 - Risikocode
 - Versteckte Konstantheit

[Tool zur Analyse]

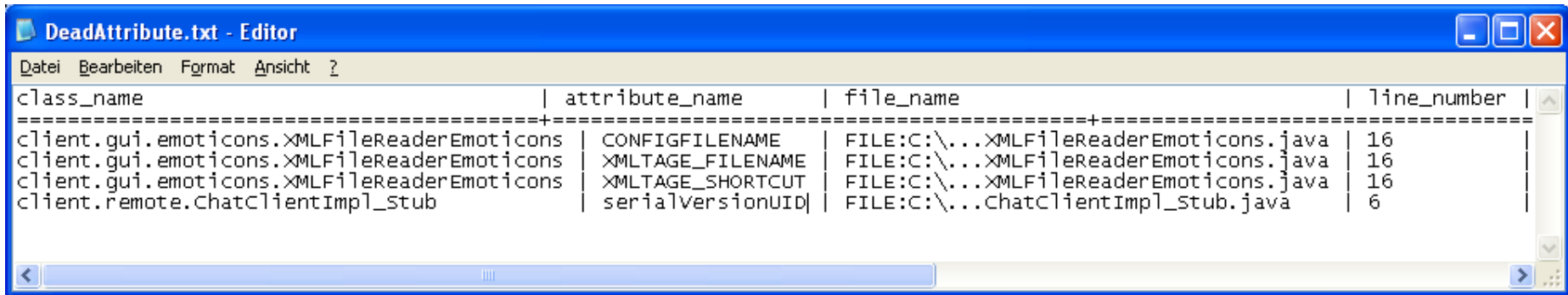
- Sissy (FZI Forschungszentrum Informatik, Karlsruhe)
 - Software Metriken
 - Problem Pattern
 - Komplexitätsanalyse
 - Toter Code

Vorgehen von Sissy



Sissy: Bsp. Vorgefertigte DB Abfragen

- Dead attribute



The screenshot shows a text editor window with a blue title bar and a menu bar. The main content is a table with four columns: class_name, attribute_name, file_name, and line_number. The table is separated by dashed lines. The data rows are as follows:

class_name	attribute_name	file_name	line_number
client.gui.emoticons.XMLfileReaderEmoticons	CONFIGFILENAME	FILE:C:\...XMLfileReaderEmoticons.java	16
client.gui.emoticons.XMLfileReaderEmoticons	XMLTAGE_FILENAME	FILE:C:\...XMLfileReaderEmoticons.java	16
client.gui.emoticons.XMLfileReaderEmoticons	XMLTAGE_SHORTCUT	FILE:C:\...XMLfileReaderEmoticons.java	16
client.remote.ChatClientImpl_Stub	serialVersionUID	FILE:C:\...ChatClientImpl_Stub.java	6

[Vorteile QBL]

- Klare Fakten
 - Es wird gemessen
 - Richtwerte sind vorhanden

[Nachteile QBL]

- Analyse ist aufwendig

[Referenzen]

- Code-Quality-Management ISBN 3-89864-388-3
- Tool Sissy: <http://sissy.fzi.de>