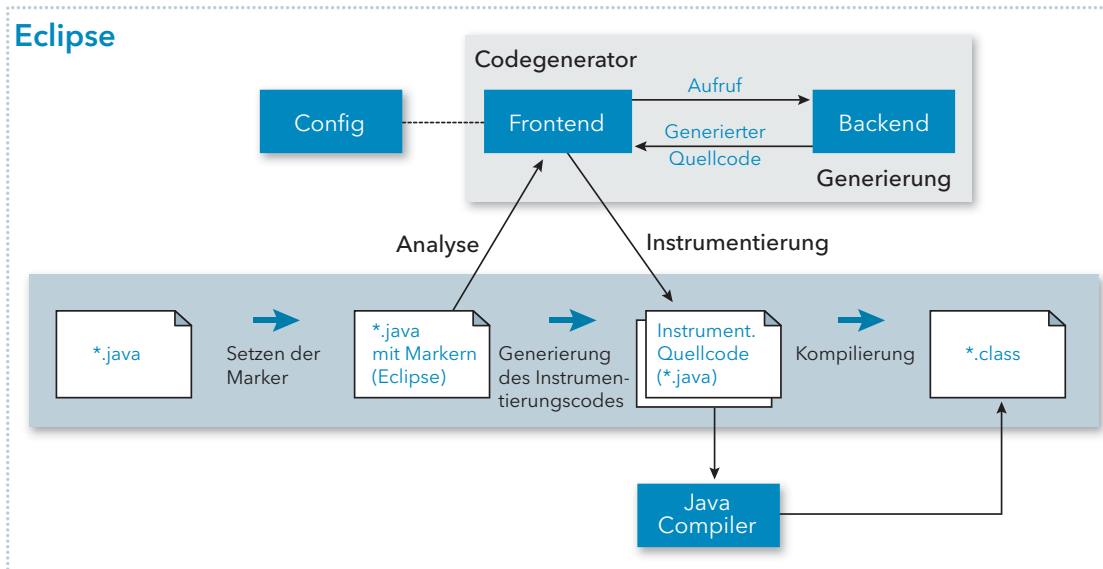


# eMIVA

## Effiziente Modell-basierte Instrumentierung verteilter Anwendungen



Fachhochschule Wiesbaden  
University of Applied Sciences



Instrumentierung von Quellcode dient der Qualitätssicherung im Rahmen des Entwicklungsprozesses von Software sowie der Laufzeitüberwachung produktiver Anwendungssysteme. Die bisher übliche manuelle Quellcode-Instrumentierung ist zeitintensiv und oftmals fehlerbehaftet oder inkonsistent. Das eMIVA-Framework unterstützt Entwickler, indem es Methoden und Werkzeuge in Form von Eclipse™-Erweiterungen (Plugins) bereitstellt, die den Instrumentierungsprozess in Hinblick auf Fehlerfreiheit und Effizienz deutlich verbessern.

Dies ist besonders für komplexe, unternehmenskritische Anwendungen wichtig, deren Beurteilung nicht mit einer äußeren Sichtweise auskommt, sondern eine detaillierte Betrachtung der internen Aktivitäten erfordert. Neben üblichen kontrollflussorientierten Anwendungen wird insbesondere auch die Instrumentierung verteilter Anwendungen mit asynchroner, nachrichtenbasierter Kommunikation unterstützt.

Prinzipiell werden Ansätze zur reinen Protokollierung des Anwendungsverlaufs (Tracing, Logging) und Architekturen zum zeitlichen Vermessen von Abläufen unterschieden. Log4J und Log4Cxx sind Beispiele für bekannte freie Logging-Bibliotheken. Zur Performance-Bewertung von Anwendungen hat sich in den letzten Jahren insbesondere der Standard Application Response Measurement (ARM) der Open Group etabliert, für den C/C++- und Java-Schnittstellen spezifiziert sind. Derzeit werden folgende Instrumentierungsarchitekturen von eMIVA unterstützt:

- Application Response Measurement (ARM) API 4.0 für C/C++ und Java
- Log4J und JDK-Logging
- Log4Cxx

Auf der Methodenebene verfolgt eMIVA Ansätze basierend auf (mit wachsendem Abstraktionsgrad):

- Strukturierten Kommentaren und aspektorientierter Programmierung (AOP)
- Markern (ähnlich Breakpoints) in der integrierten Entwicklungsumgebung
- Instrumentierung im UML-Modell der Anwendung mittels abstrakter Instrumentierungsmuster

Prof. Dr. Reinhold Kröger  
kroeger@informatik.fh-wiesbaden.de  
Dipl.-Inform. (FH) Markus Schmid M.Sc.  
Dipl.-Inform. (FH) Jan Schäfer

Fachhochschule Wiesbaden  
Labor für Verteilte Systeme  
Kurt-Schumacher-Ring 18  
D-65197 Wiesbaden  
[www.informatik.fh-wiesbaden.de](http://www.informatik.fh-wiesbaden.de)

Projektpartner:



tang-IT Consulting GmbH  
Nordenstadter Straße 38  
D-65207 Wiesbaden



DFS Deutsche Flugsicherung  
DFS Deutsche Flugsicherung GmbH  
Am DFS-Campus 10  
D-63225 Langen



gefördert durch das

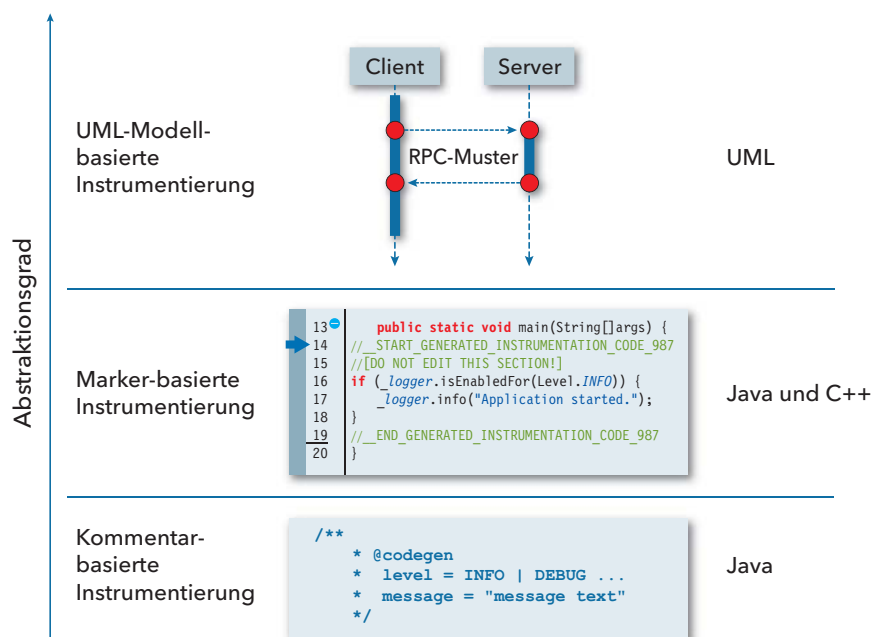


Bundesministerium  
für Bildung  
und Forschung

Förderkennzeichen: 1720X04

Die zugehörigen Werkzeuge erweitern Eclipse™ um entsprechende Plugins. Im ersten Fall werden Java-Anwendungen mit strukturierten Kommentaren versehen, aus denen Instrumentierungscode generiert und mittels AOP in die Anwendung eingeflochten wird. Im zweiten Fall wird der Editor der Entwicklungsumgebung um grafische Marker erweitert, mit denen der Entwickler Instrumentierungspunkte innerhalb des Quellcodes auszeichnet. Auf Knopfdruck kann aus dieser grafischen Repräsentierung Instrumentierungscode in der Zielsprache (derzeit Java und C++) generiert werden. Im dritten Fall erfolgt die Instrumentierung bereits im UML-Modell der Anwendung.

Das Projekt wird vom Labor für Verteilte Systeme der Fachhochschule Wiesbaden gemeinsam mit der tang-IT GmbH in Wiesbaden und der Deutsche Flugsicherung GmbH (DFS) in Langen durchgeführt. Mit Hilfe der eMIVA-Werkzeuge wird bei der DFS eine Performance-Instrumentierung des Radardatenverarbeitungssystems Phoenix realisiert, die ein umfassendes Performance-Monitoring der Anwendung zum Ziel hat. Gleichzeitig können hierdurch die in eMIVA entwickelten Methoden validiert werden.



## eMIVA - Efficient Model-based Instrumentation of Distributed Applications

Code instrumentation is essential for quality assurance during application development as well as for production system monitoring. The eMIVA project aims to support developers by delivering methods and Eclipse™-based tools for semi-automatic application instrumentation. Especially for enterprise applications, a detailed analysis of internal processes is essential. The eMIVA tools aim to improve efficiency and consistency of the instrumentation process, especially supporting developers without further knowledge of instrumentation technologies.

Our solution focuses on code-generation for logging (using log4j, log4cxx) and performance monitoring (using the Open Group Application Response Measurement - ARM).

We provide three methods of instrumentation support on different abstraction levels: developers can instrument applications using either structured comments (javadoc-style), Eclipse markers, or UML model-based instrumentation. Based on this information, the eMIVA tools generate code for the target programming language (Java or C++) and instrumentation architecture. eMIVA is a collaboration between the Distributed Systems Lab of Wiesbaden University of Applied Sciences, tang-IT GmbH and the DFS Deutsche Flugsicherung (German Air Navigation Services). DFS realises the performance instrumentation of their radar-tracking system Phoenix with the help of the eMIVA tools.