



Technische Fakultät
Institut für Informatik
Arbeitsgruppe Software Engineering

v0.0

Kieker Roadmap

Reiner Jung

16. Februar 2011

Inhaltsverzeichnis

1	Ziele	2
2	Bestehende Konzepte	2
2.1	Kieker 1.2	2
2.2	Monitoring Tool	2
2.3	Kanalyse	2
2.4	Projekt-Infrastruktur	2
3	Mögliche und Geplante Arbeiten	3

1 Ziele

- Instrumentierung für verschiedene Programmiersprachen via Source-Code, Code-Injection, Intermediate-Code
- Sammlung von Auswertungsverfahren
- Sammlung von Visualisierungen
- Modellgetriebene Instrumentierung
- Klare Schnittstellen und Methodiken für alle Komponenten von Kieker

2 Bestehende Konzepte

2.1 Kieker 1.2

Kieker-Basis-Projekt mit der klassischen Kieker Instrumentierungstechnik für Java und der Probe, Storage, Auswertungs-Architektur.

2.2 Monitoring Tool

Jens hat im Rahmen seiner Dis. ein grafisches Werkzeug gebaut, welches ebenfalls zum Kieker-Kontext gehört.

- Graphischer Editor für die Kombination von Filtern/Analyseplugins
- Visualisierung der Ergebnisse

2.3 Kanalyse

- Pipe-and-Filter-Sprache für die Kombination von Filtern und ggf. die Komposition von Filtern.
- Ecore-Modell für Plugins

2.4 Projekt-Infrastruktur

Um ein die einzelnen Projekte zusammenführen zu können und ein einheitliches Framework zu bekommen sind verschiedene Schritte notwendig. Für den Anfang schlage ich deshalb ein gemeinsames Repository für Kieker 2.0 vor in das wir dann die einzelnen Werkzeuge integrieren. Da wir alle hip und modern sind empfiehlt sich hier git zu benutzen.

Im Moment wird zwar Sourceforge benutzt, aber ein integriertes Ticket, Versions und Wiki System können wir auch mit Trac und Git realisieren. Ebenso können wir Hudson hier einbinden. Im Grunde entspricht dieses Setup auch dem Setup der RTSYS Gruppe von der wir dann auch entsprechend Technologietransfer haben können. Und ich glaube dieses Setup entspricht auch der generellen Strategie in der Abteilung.

3 Mögliche und Geplante Arbeiten

Auf inhaltlicher Ebene sollten wir uns die einzelnen Pakete ansehen und prüfen was davon weiter zu verwenden ist und vor allem wie es in ein Gesamtprojekt eingebracht werden kann ohne gleich alles neu schreiben zu müssen.

Nach einem solchen Bootstrapping, können wir uns dann um Detail kümmern, welche auch als Arbeiten für Studierende ausgelegt werden können. Im Einzelnen sind das:

- Generisches Analyse-Framework (Design) auf Basis von Jens' und Andrés Arbeiten
- Metamodell für das Logging-Format sowie Transformationen, die aus Logging-Format-Modellen passende Serialisierer (Reader/Writer) erzeugen BA
- Implementierung des Frameworks mehrere Arbeiten
- Meta-Modell/Sprache für das Analyse-Framework MA
- Graphischen Editor für das Analyse-Framework BA/MA
- Instrumentierungsbeschreibung und Generator für Kieker-artigen Instrumentierungscode für Java BA/MA
- Instrumentierungsbeschreibung und Generator für Kieker-artigen Instrumentierungscode für C# BA/MA
- Instrumentierungsbeschreibung und Generator für Kieker-artigen Instrumentierungscode für ST/FUP BA/MA