

Monitoring von Perl-basierten Web Anwendungen mit Kieker Proposal

Nis Wechselberg

Institut für Informatik
Christian-Albrechts-Universität zu Kiel

19. Dezember 2012

Gliederung

- 1 Motivation
 - Kielprints
 - Beobachtete Performanceprobleme
- 2 Technologien und Methoden
 - Performance-Monitoring
 - Kieker-Framework
 - Programmiersprache Perl
- 3 Geplante Umsetzung
 - Architekturentwurf
 - Zeitplan

Gliederung

- 1 Motivation
 - Kielprints
 - Beobachtete Performanceprobleme
- 2 Technologien und Methoden
 - Performance-Monitoring
 - Kieker-Framework
 - Programmiersprache Perl
- 3 Geplante Umsetzung
 - Architekturentwurf
 - Zeitplan

Kielprints

The screenshot shows the Kielprints website interface. At the top, there is a search bar with the text "Geben Sie Ihren Befehl ein". Below it, the website header includes the logo "C A U" for Christian-Albrechts-Universität zu Kiel and the text "Christian-Albrechts-Universität zu Kiel" and "ePrints". Navigation links for "Startseite", "Suche", "Kontakt", "Sitemap", "Impressum", "Datenschutz", and "English" are visible. A sidebar on the left contains a menu with "Kielprints", "Startseite", "Schnellsuche", "Einfache Suche", "Erweiterte Suche", "Blättern", "Autor", "Forschungsbereich", "Publikationsart", and "Jahr". The main content area displays the title "Kieker: A Framework for Application Performance Monitoring and Dynamic Software Analysis" and the authors "van Hoorn, André, Waller, Jan und Hasselbring, Wilhelm". It includes a "Tools" section with a dropdown menu set to "RDF+XML" and an "Export" button. Below this, there are three download links for PDF files: "KiekerICPE2012-camera-ready.pdf", "KiekerICPE2012-poster.pdf", and "120424-ICPE-slides-final.pdf". The official URL is provided as <http://dx.doi.org/10.1145/2188286.2188326>. An "Abstract" section is partially visible at the bottom.

Projektbeschreibung

- Plattform zur Veröffentlichung von wissenschaftlichen Dokumenten (*self archiving*)

Projektbeschreibung

- Plattform zur Veröffentlichung von wissenschaftlichen Dokumenten (*self archiving*)
- Ergänzte und modifizierte Version von ePrints

Projektbeschreibung

- Plattform zur Veröffentlichung von wissenschaftlichen Dokumenten (*self archiving*)
- Ergänzte und modifizierte Version von ePrints
- Betrieben durch das Helmholtz Zentrum für Ozeanforschung Kiel (GEOMAR)

Projektbeschreibung

- Plattform zur Veröffentlichung von wissenschaftlichen Dokumenten (*self archiving*)
- Ergänzte und modifizierte Version von ePrints
- Betrieben durch das Helmholtz Zentrum für Ozeanforschung Kiel (GEOMAR)
- Erweiterung des OceanRep GEOMAR

Projektbeschreibung

- Plattform zur Veröffentlichung von wissenschaftlichen Dokumenten (*self archiving*)
- Ergänzte und modifizierte Version von ePrints
- Betrieben durch das Helmholtz Zentrum für Ozeanforschung Kiel (GEOMAR)
- Erweiterung des OceanRep GEOMAR
- Über 14000 Veröffentlichungen von etwa 1000 Autoren

Benutzerperspektive

- Abrufen von Einträgen über die Plattform:

Benutzerperspektive

- Abrufen von Einträgen über die Plattform:
- Erhöhte Antwortzeiten

Benutzerperspektive

- Abrufen von Einträgen über die Plattform:
- Erhöhte Antwortzeiten
- Langsame Suche in den Fachbereichen

Administration/Betrieb

- Zur Zeit *Vorberechnung* in der Nacht durch Cron-Job

Administration/Betrieb

- Zur Zeit *Vorbereitung* in der Nacht durch Cron-Job
- Trotzdem hohe Wartezeiten bei Seitenwechseln im Backend

Administration/Betrieb

- Zur Zeit *Vorbereitung* in der Nacht durch Cron-Job
- Trotzdem hohe Wartezeiten bei Seitenwechseln im Backend
- Beispiel: Beim Eintragen von neuen Publikationen über das Backend und hinzufügen von Autoren dauert die Generierung von Dropdown-Menüs etwa 10 Sekunden.

Gliederung

- 1 Motivation
 - Kielprints
 - Beobachtete Performanceprobleme
- 2 **Technologien und Methoden**
 - Performance-Monitoring
 - Kieker-Framework
 - Programmiersprache Perl
- 3 Geplante Umsetzung
 - Architekturentwurf
 - Zeitplan

Vorgehen bei Performance-Monitoring

- 1 Instrumentierung des Codes mit *Probes*

Vorgehen bei Performance-Monitoring

- 1 Instrumentierung des Codes mit *Probes*
- 2 Ausführung des instrumentierten Codes

Vorgehen bei Performance-Monitoring

- 1 Instrumentierung des Codes mit *Probes*
- 2 Ausführung des instrumentierten Codes
- 3 Protokollierung von Monitoring Daten

Vorgehen bei Performance-Monitoring

- 1 Instrumentierung des Codes mit *Probes*
- 2 Ausführung des instrumentierten Codes
- 3 Protokollierung von Monitoring Daten
- 4 Auswertung der Daten mittels geeigneter Tools

Erfasste Daten

- Anwendungsdaten

Erfasste Daten

- Anwendungsdaten
 - Aufrufreihenfolge

Erfasste Daten

- Anwendungsdaten
 - Aufrufreihenfolge
 - Aufrufhäufigkeiten

Erfasste Daten

- Anwendungsdaten
 - Aufrufreihenfolge
 - Aufrufhäufigkeiten
 - Ausführungszeiten

Erfasste Daten

- Anwendungsdaten
 - Aufrufreihenfolge
 - Aufrufhäufigkeiten
 - Ausführungszeiten
- Systemdaten

Erfasste Daten

- Anwendungsdaten
 - Aufrufreihenfolge
 - Aufrufhäufigkeiten
 - Ausführungszeiten
- Systemdaten
 - CPU-Auslastung

Erfasste Daten

- Anwendungsdaten
 - Aufrufreihenfolge
 - Aufrufhäufigkeiten
 - Ausführungszeiten
- Systemdaten
 - CPU-Auslastung
 - Arbeitsspeicher

Erfasste Daten

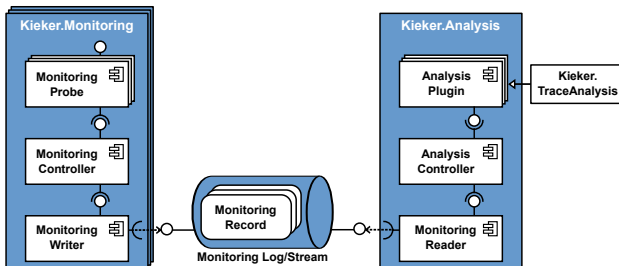
- Anwendungsdaten
 - Aufrufreihenfolge
 - Aufrufhäufigkeiten
 - Ausführungszeiten
- Systemdaten
 - CPU-Auslastung
 - Arbeitsspeicher
 - aktive Prozesse

Erfasste Daten

- Anwendungsdaten
 - Aufrufreihenfolge
 - Aufrufhäufigkeiten
 - Ausführungszeiten
- Systemdaten
 - CPU-Auslastung
 - Arbeitsspeicher
 - aktive Prozesse
 - ...

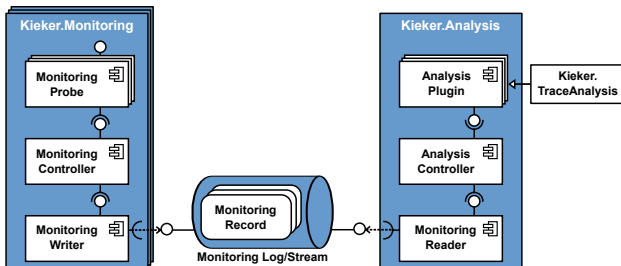
Kieker Monitoring Framework

- Monitoring Tool aus diesem Lehrstuhl



Kieker Monitoring Framework

- Monitoring Tool aus diesem Lehrstuhl
- Empfohlenes Tool im SPEC RG Software Repository



Überblick über Kieker

- Aufteilung in *Kieker.Monitoring* und *Kieker.Analysis*

Überblick über Kieker

- Aufteilung in *Kieker.Monitoring* und *Kieker.Analysis*
- Kieker.Monitoring zur Instrumentierung und Überwachung

Überblick über Kieker

- Aufteilung in *Kieker.Monitoring* und *Kieker.Analysis*
- Kieker.Monitoring zur Instrumentierung und Überwachung
- Kieker.Analysis zur Auswertung und Veranschaulichung

Überblick über Kieker

- Aufteilung in *Kieker.Monitoring* und *Kieker.Analysis*
- Kieker.Monitoring zur Instrumentierung und Überwachung
- Kieker.Analysis zur Auswertung und Veranschaulichung
- Kommunikation über Monitoring Log oder Stream

Überblick über Kieker

- Aufteilung in *Kieker.Monitoring* und *Kieker.Analysis*
- Kieker.Monitoring zur Instrumentierung und Überwachung
- Kieker.Analysis zur Auswertung und Veranschaulichung
- Kommunikation über Monitoring Log oder Stream

- Bisher keine Unterstützung für Perl

Programmiersprache Perl

- imperative, plattformunabhängige, interpretierte Sprache

Programmiersprache Perl

- imperative, plattformunabhängige, interpretierte Sprache
- Entwickelt 1987, heute in Version 5.16 verfügbar

Programmiersprache Perl

- imperative, plattformunabhängige, interpretierte Sprache
- Entwickelt 1987, heute in Version 5.16 verfügbar
- Sehr viele Freiheiten für Programmierer (freie Syntax)

Programmiersprache Perl

- imperative, plattformunabhängige, interpretierte Sprache
- Entwickelt 1987, heute in Version 5.16 verfügbar
- Sehr viele Freiheiten für Programmierer (freie Syntax)
- Starke Funktionen für String-Manipulationen und Reguläre Ausdrücke

Programmiersprache Perl

- imperative, plattformunabhängige, interpretierte Sprache
- Entwickelt 1987, heute in Version 5.16 verfügbar
- Sehr viele Freiheiten für Programmierer (freie Syntax)
- Starke Funktionen für String-Manipulationen und Reguläre Ausdrücke
- Direkte Integration in Webserver mittels CGI-Perl

Gliederung

- 1 Motivation
 - Kielprints
 - Beobachtete Performanceprobleme
- 2 Technologien und Methoden
 - Performance-Monitoring
 - Kieker-Framework
 - Programmiersprache Perl
- 3 Geplante Umsetzung
 - Architekturentwurf
 - Zeitplan

Neuimplementierungen in Perl

- Implementierung von Monitoring Probes und Monitoring Writer

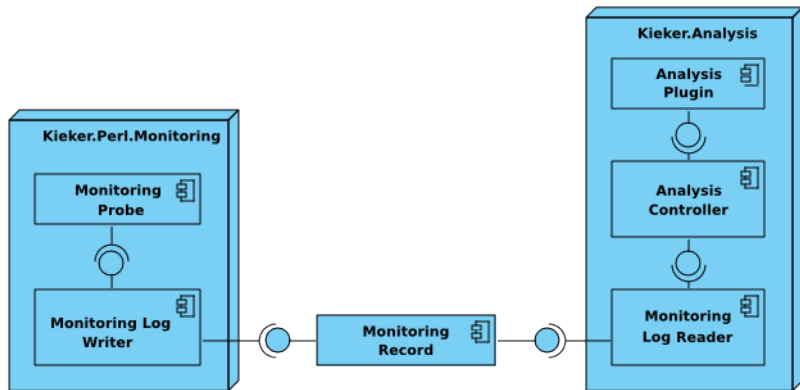
Neuimplementierungen in Perl

- Implementierung von Monitoring Probes und Monitoring Writer
- Ausgabe von Monitoring Records in Monitoring Log

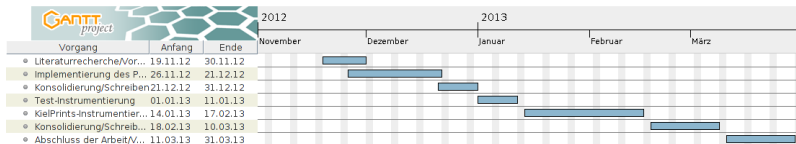
Neuimplementierungen in Perl

- Implementierung von Monitoring Probes und Monitoring Writer
- Ausgabe von Monitoring Records in Monitoring Log
- Auswertung mit *Kieker.Analysis*

Architekturentwurf

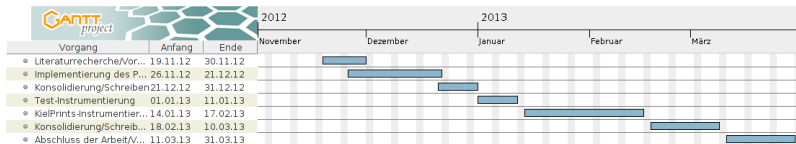


Zeitplan



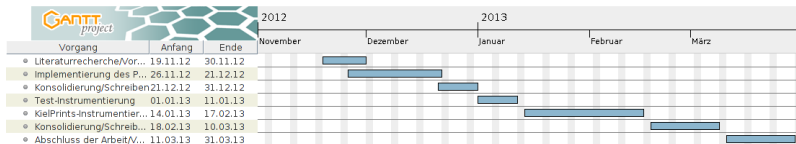
- Implementierung der benötigten Funktionen bis 23.12.

Zeitplan



- Implementierung der benötigten Funktionen bis 23.12.
- Ab Mitte Januar zunächst Testinstrumentierung von Kielprints

Zeitplan



- Implementierung der benötigten Funktionen bis 23.12.
- Ab Mitte Januar zunächst Testinstrumentierung von Kielprints
- Anschliessend Instrumentierung und Analyse im Produktivsystem

Ich danke für Ihre Aufmerksamkeit!