Tool-Supported Application Performance Problem Detection and Diagnosis

André van Hoorn
Agenda

1. Introduction – Performance Problems
2. Kieker – Open Source APM Framework
3. Performance Problem Detection and Diagnosis with Kieker
4. Conclusions and Future Work

Temporarily not available
Performance (QoS) Problem Detection and Diagnosis

An unexpected error occurred

Temporarily not available

Try again later

Please visit us again later

Service temporarily unavailable

... more capacity is on the way

Temporarily unavailable

Service not available

We are experiencing heavy demand
Performance (QoS) Problem Detection and Diagnosis

How to (semi-)automate
1. the detection of performance problems?
2. pinpointing the root cause (i.e., diagnosis)
3. the proactive detection and diagnosis?

An unexpected error occurred.

Service temporarily unavailable
Service not available

Temporarily not available
Try again later
Please visit us again later

mozilla services

We are experiencing heavy demand

… more capacity is on the way
Application Performance Management

- APM dimensions according to Gartner (2014)
  1. End-user experience monitoring
  2. Application topology discovery and visualization
  3. User-defined transaction profiling
  4. Application component deep-dive
  5. IT operations analytics based on, e.g.,
     - Complex operations event processing
     - Statistical pattern discovery and recognition
     - Unstructured text indexing, search and inference
     - Multidimensional database search and analysis

- APM tools (selection)
  - Commercial:
    - APPDYNAMICS
    - dynatrace
    - New Relic
  - Free/open-source:
    - inspectIT
    - Kieker
Application Performance Management

- APM dimensions
  1. End-user experience monitoring
  2. Application topology discovery and visualization
  3. User-defined transaction profiling
  4. Application component deep dive
  5. IT operations analytics based on, e.g.,
     - Complex operations event processing
     - Statistical pattern discovery and recognition
     - Unstructured text indexing, search and inference
     - Multidimensional database search and analysis

- APM tools (selection)
  - Commercial:
    - APPDYNAMICS
    - dynatrace
    - New Relic
  - Free/open-source:
    - inspectIT
    - Kieker

Example ITOA activity:
Performance problem detection and diagnosis
- Reactive vs. proactive
- Manuel vs. automatic (incl. recommendations)
- State-based vs. transaction-based
Agenda

1. Introduction – Performance Problems

2. Kieker – Open Source APM Framework

3. Performance Problem Detection and Diagnosis with Kieker

4. Conclusions and Future Work

Temporarily not available
Open Source APM Framework

Monitoring Records → Analysis Configuration (via API and WebGUI) → Analysis → Pipes and Filters → Online and Offline Visualization

Software System with Monitoring Instrumentation

Monitoring Probe

Measurement → Monitoring log/stream


Kieker is distributed as part of SPEC® RG's repository of peer-reviewed tools for quantitative system evaluation and analysis.

Model Extraction and Visualization
(Selected Examples)

- Distributed Monitoring (Java EE/SOAP)
- Legacy System Analysis (COBOL)
- Legacy System Analysis (Visual Basic 6)
- 3D Visualization of Concurrency
Visit http://kieker-monitoring.net/

Projects, publications, talks, tutorials

Download

User Guide

Issue Tracking
References: Internal/External Researchers, Industry

Internal Researchers

Kiel University, Kiel, Germany – Researchers from the Kiel University’s Software Engineering Group investigate innovative techniques and methods for engineering, evolving, and operating continuously running software systems (research projects).

University of Stuttgart, Stuttgart, Germany – Researchers from the University of Stuttgart’s Reliable Software Systems Group investigate innovative quantitative QoS analysis and forecasting methods for distributed software-intensive systems (research projects).

Feel free to contact us if you are interested in any aspect of the Kieker framework.

External Researchers

Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany – Researchers from the KIT’s Software Design and Quality Group are using Kieker for different purposes, e.g., for detecting and diagnosing performance problems in systematical experiments. We are also collaborating with KIT researchers in the context of the iObserve research project.

RWTH Aachen University, Aachen, Germany – Researchers from the RTWH Aachen University’s Software Construction Group are using Kieker for monitoring-based architecture reconstruction in their ARAMIS project on model-based software architecture evolution and analysis.

University of Novi Sad, Novi Sad, Serbia – Researchers from the University of Novi Sad were using Kieker for adaptive monitoring of software systems in the context of performance problem detection and diagnosis.

Warsaw University, Warsaw, Poland – Researchers from the Warsaw University employed Kieker for dynamic data acquisition of software architectures.

Kian Jia Tong University, Xian, Shaanxi, China – Researchers from the Xian Jia Tong University used Kieker for discovering architectural structures in software systems and to analyze software call graphs.

Industry

b+m Informatik AG, Melsdorf, Germany – With b+m, we collaborated in the context of the DynaMod and MENDES research projects. Moreover, Kieker is being used by b+m, e.g., for architecture discovery of large-scale COBOL mainframe systems. Contributions by b+m are part of the Kieker release.

CEWE COLOR AG & Co. OHG, Oldenburg, Germany – With CEWE COLOR, we collaborated in the context of the TrustSoft research project. CEWE COLOR provided a JavaEE-based web portal as a case study system for application performance monitoring. Contributions by CEWE COLOR are part of the Kieker release.

dataport AOR, Altenholz, Germany – With Dataport, we collaborated in the context of the DynaMod research project. Dataport provided a V86-based case study system for architecture discovery based on hybrid analysis with Kieker.

EP reflex Services, Southport, United Kingdom – With EP reflex Services, we collaborated in the context of several thesis projects. We employ the EP reflex system as a case study system for software performance analysis with Kieker for Perl-based systems. The Ep reflex team provides an integration of Kieker with EP reflex as eplieker.

EWE TEL GmbH, Oldenburg, Germany – With EWE TEL, we collaborated in the context of the TrustSoft research project. EWE TEL provided a JavaEE-based web portal as a case study system for application performance monitoring. Contributions by EWE TEL are part of the Kieker release.

HSH Nordbank AG, Kiel, Germany – With HSH Nordbank, we collaborated in the context of the DynaMod research project. The HSH provided a C#-based function library for architecture discovery based on hybrid analysis with Kieker.

NovaTec GmbH, Lentfelden-Echterdingen, Germany – With NovaTec, we currently collaborate in the context of different teaching projects on application performance management. NovaTec published a nice blog article about their first experiences with Kieker.

SAP Research, Karlsruhe, Germany – Kieker is used as a tool to collect performance data for the Software Performance Cockpit. We are also collaborating with SAP Research in the context of the iObserve research project.

XING AG, Hamburg, Germany – With XING, we collaborated in the context of a Diploma thesis on online performance anomaly detection (OPAD). XING provided its core system xing.com for evaluating the Kieker-based OPAD implementation.
Agenda

1. Introduction – Performance Problems
2. Kieker – Open Source APM Framework
3. Performance Problem Detection and Diagnosis (PPD&D) with Kieker
4. Conclusions and Future Work
PAD – Time Series Analysis Introduction

Forecasts for Wikipedia data (Kieker PAD example)
©PAD: Online Performance Anomaly Detection

(Bielefeld, 2012), (Frotscher, 2013)
PAD (cont’d) – Time Series Extraction

Time Series Extraction → Time Series Forecasting → Anomaly Score Calculation → Anomaly Detection → Alerting (e.g., AMQP)

Continuous Time:
- Event on ES
- Discretization Function
- Time Series X
- Current Time

Discrete Time Series:
- $\Delta x$

SQL:
```
select sum(value) as aggregation
from MeasureEvent.win:time_batch(1000 msec)
```
PAD (cont’d) – Time Series Forecasting

- Time Series Extraction
- Time Series Forecasting
- Anomaly Score Calculation
- Anomaly Detection
- Alerting (e.g., AMQP)
PAD (cont’d) – Anomaly Score Calculation

- Time Series Extraction
- Time Series Forecasting
- Anomaly Score Calculation
- Anomaly Detection
- Alerting (e.g., AMQP)
PAD (cont’d) – Anomaly Detection

- Reader
- Time Series Extraction
- Filter
- Time Series Forecasting
- Filter
- Anomaly Score Calculation
- Filter
- Anomaly Detection
- Alerting (e.g., AMQP)

Abnormal Score
Normal Score
Anomaly Threshold
Anomaly Detected
Based on time series analysis (various algorithms)
• PAD part of Kieker release
• Limited to problem detection
• No architecture consideration
• Case study at XING

Incorporates architectural knowledge (e.g., deployment, calling dependencies)
• Focusing on offline analysis
• Cf. Rohr (2015)

(Bielefeld, 2012), (Frotscher, 2013)

(Marwede et al., 2009)
-based PPD&D Approaches

- Adaptive monitoring
- OCL-based decisions
- Cf. Okanovic et al. (2013)
- No dynamic (bytecode) instrumentation

(Ehlers et al., 2011, 2012)

- Proactive, hierarchical
- Inclusion of different statistical techniques (e.g., time series analysis, machine learning)
- Combination of multiple data sources (e.g., HDD SMART, log files) and architectural knowledge

(Pitakrat et al., 2013, 2014)
Agenda

1. Introduction – Performance Problems
2. Kieker – Open Source APM Framework
3. Performance Problem Detection and Diagnosis with Kieker
4. Conclusions and Future Work

Temporarily not available
Summary

- APM is an increasingly important topic
- PPD&D is one APM activity

- Mature APM tools exist
  - Commercial tools
    - Support monitoring for various platforms and technologies
    - Only basic support for problem detection and diagnosis
  - Kieker presented as an extensible open-source example
    - Kieker-based approaches for problem detection and diagnosis
Challenges for APM and PPD&D (Selection)

1. Laborious and Continuous APM Configuration
   - APM configuration time consuming and error prone
   - Problem intensified due to faster development cycles

2. Problem Detection and Diagnosis
   - Alerting thresholds hard to determine and to maintain
   - Manual diagnosis requires extensive expert knowledge
   - APM experts are scarce goods

3. Detachedness of APM processes
   - APM processes often system-specific
   - Few possibilities to deposit/reuse expert knowledge
Future Work: Expert-Guided Automatic Diagnosis of Performance Problems in Enterprise Applications
Future Work: Expert-Guided Automatic Diagnosis of Performance Problems in Enterprise Applications
Future Work: Expert-Guided Automatic Diagnosis of Performance Problems in Enterprise Applications
Future Work: DiagnoseIT
Expert-Guided Automatic Diagnosis of Performance Problems in Enterprise Applications

WP1: Enterprise Performance Model
- System Meta-Model
- Instrumentation
- Performance Properties
- Performance Requirements
- System Model Repository

WP2: APM Knowledge Base
- Knowledge Base Meta-Model
- System Generic
- Problem Diagnosis
- Pattern
- Instrumentation
- Best Practices
- APM Knowledge Base Repository

WP3: Runtime Diagnostics
- Symptom Recognition
- Pattern Interpretation
- System Model Maintenance
- Instrumentation Quality Monitor

WP4: Tool Integration
- Performance Measurement Data
- Architecture Information
- Dynamic Instrumentation
- Diagnose Feedback
  - inspectIT
  - Kieker
  - APM-1
  - APM-2
  - GUI
  - Alerting/Ticketing

Sponsored by the Federal Ministry of Education and Research
The End – My APM Wish List

- **Transparency, Openness, Technology Transfer**
  - e.g., sharing of best practices (libraries, white papers) for framework-specific instrumentation, problem detection and diagnosis

- **Interoperability**
  - e.g., common formats for instrumentation description, configuration, measurement data (traces)

- **Reproducibility, Comparibility**
  - e.g., sample/benchmark applications and datasets, case studies

- SPEC RG DevOps Performance Working Group
  - [http://research.spec.org/devopswg/](http://research.spec.org/devopswg/)

- Kieker
  - [http://kieker-monitoring.net](http://kieker-monitoring.net)
References


References (cont’d)


References (cont’d)


BONUS/BACK-UP
Netflix OSS Recipes Application (Motivating Example)

Download: https://github.com/Netflix/recipes-rss

Source: http://techblog.netflix.com/2013/03/introducing-first-netflixoss-recipe-rss.html
Netflix OSS Recipes Application (cont’d)

Tool-supported application performance problem detection and diagnosis

Source: https://github.com/Netflix/recipes-rss/wiki/Appliance
Problem Symptom 1: Increased Response Times

Source: https://github.com/Netflix/recipes-rss/wiki/Architecture
Problem Symptom 2: Service Unavailability

Tool - supported application performance problem detection and diagnosis

Source: https://github.com/Netflix/recipes-rss/wiki/Architecture
Problem Detection and Diagnosis Approaches

- **Features**
  - Reactive vs. proactive
  - Manual vs. automatic
  - State-based vs. transaction-based etc.

- **Statistical techniques**
  - Time series analysis
  - Anomaly detection (incl. change detection)
  - Machine learning etc.
Example Kieker Plots for Netflix OSS Recipes Application

Edge

- GetRSSCommand
- AddRSSCommand
- DelRSSCommand

Middletier

- Backend call (Jersey)
- Outgoing/incoming REST calls (Jersey)