

Praktikum angewandte Systemsoftwaretechnik (PASST)

Blockpraktikum

25. Juli 2019

Tobias Langer, Stefan Reif, Michael Eischer
und Florian Schmaus

Lehrstuhl für Informatik 4
Friedrich-Alexander-Universität Erlangen-Nürnberg



Lehrstuhl für Verteilte Systeme
und Betriebssysteme



FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT

- Projektwahl und Gruppenbildung: 2er Gruppen
- Projektvorstellung
 - 10 min. Präsentation im Plenum + 10 min. Diskussion
 - Vorstellung, Ansatz, erwartete Ergebnisse, Zeitplanung
- 2 Wochen Vollzeit
 - Bei Bedarf tägliches Jour Fixe
 - Zwischentreffen
- Abschlusspräsentation
 - 20 min. Präsentation im Plenum + 10 min. Diskussion
 - Ergebnisse, Erfahrungen, Fazit
- Termin: **16.09.2019 – 27.09.2019**

Zielsetzung (1/2)

Erfolg im Praktikum wird am Erreichen der Zielsetzungen gemessen:

- Gelerntes anwenden
- Selbständige Projektdurchführung im Team
- Entwicklungsprozesse in F(L)OSS-Projekten praktisch anwenden
 - Verwenden entsprechender Werkzeuge (git, Patche, ...)
 - Einbinden der Entwicklergemeinschaft (Features an Upstream)

Ziel

Benutzbare Software für euch, uns und den Rest der Welt

Bewertet wird:

- Lösungsfindung und Umsetzung der Lösung
- Kollaboration zwischen euch
- (Kommunikation und Zusammenarbeit mit Upstream)
- Projekt wird veröffentlicht (Publish or it didn't happen!)
 - Entsprechend: Repo mit Doku, Lizenz, ...

Notenfindung (Wiederholung)

Teilnote	A1	A2	A3	A4	A5	A6	Blockpraktikum
Gewichtung	1	1	2	2	2	2	15

- Semesterbegleitender Teil macht 40% der Punkte aus
- erreichbare Punktezahlen und damit Gewichtung entsprechend dem Umfang der Aufgaben
- Blockpraktikum umfasst die restlichen 60%

USB-GPIO-Erweiterungsplatine mit passendem Linux-Treiber

Firmware & Treiberentwicklung

- GPIO-Pins über USB in Linux einbinden
 - Einbinden unter sysfs
 - Anderen Treibern zur Verfügung stellen
- Mikrocontrollerfirmware
 - Weiterreichen von GPIO-Pins über USB
- Linux-Treiber
 - Verfügbarmachen in Subsystemen des Kernels

Julian

JITTY-OS

Linux-kompatibles Betriebssystem; kleine, übersichtliche, saubere Sourcen; Performance durch clang/LLVM mit Link-Time-Optimierungen

1. USB-Treiber für JITTY-OS:

- Treiber für den USB-Controller der StuBS-Rechner
- USB-Maus/-Tastatur-Treiber

2. SATA-Treiber für JITTY-OS:

- Treiber für den SATA-Controller/-Platten der StuBS-Rechner



Volkmar

StackViz

Dynamische Visualisierung des Callstacks für den Lehrbetrieb

- Visualisierung zur Laufzeit
 - Aufbauend auf gdb/libunwind
- Benutzeroberfläche
 - Hervorhebung von Stackframes, Rücksprungadressen, Verweisen, ...
 - Interpretation von Datenwerten
 - Export als Raster-/Vektorgrafik
- Modularer Aufbau
 - Unterstützung verschiedener Architekturen



Tobias

Softwaremaßschneidung

Maßschneidung von **Shared Libraries** in

Docker-Containern

- Docker-Container als minimale Linux-Distros
 - Anwendung + benötigte Shared Libraries
 - **Problem:** Shared Libraries sind nicht vollumfänglich notwendig
- Framework zur Maßschneidung vorhanden
 - Anwendung auf Docker-Container
 - Erweiterung und Anpassung wo notwendig
 - Analyse des erreichbaren Grads der Maßschneidung



Andreas

Archibald: Literaturverwaltung

Literaturverwaltung in Plaintext und Git

- Grundlegendes Gerüst vorhanden; Erweiterung um Features
 - Volltextindizierung & erweiterte Suche
 - Textextraktion aus PDFs
 - Im- & Export von Notizen & Annotationen
 - BibTeX Validierung
 - Git- und Editorintegration
 - Benutzerschnittstelle/Scriptingschnittstelle
 - ...
- Perlen der Programmierkunst erleben
- Lizenzierung unter AGPL3



Simon

Kaffeekasse 2000^{XT}

Cleanroom-Implementierung des
Kaffeekassensystems

- Kassensystem zur Selbstbuchung
- Nutzeroberfläche
 - Umsetzung als Website (Geräteagnostisch)
 - Touch-„optimiert“
 - Unterstützung von externen Eingabegeräten
- Serverbackend
 - Datenhaltung
 - Ausfallsicheres Protokoll (At-most once)



Simon

Themen für das Blockpraktikum (7/8)

Greenified Continues Integration

CI-Infrastruktur on Demand

- Jenkins Auslastung unterliegt starken Schwankungen
- Hinzu-/Wegschalten von Nodes on Demand
- Verbrauchsmessung über PDUs

git worktree mit Copy-on-Write (CoW)

Ausnutzung von reflink unter git

- Manche Dateisysteme unterstützen Copy-on-Write (CoW) mit reflink
- git-Erweiterung um reflink zum schnellen auschecken



Florian

Eigene Ideen und Vorschläge

Fragen?