



Themen für Masterarbeiten

WiSe 2016/17

Themen für Masterarbeiten – große Gebiete

1. IBM Power Systems
2. Heterogeneous Computing (MultiCore, ARM)
3. HP The Machine
4. IoT / Embedded Systems
5. Systemeigenschaften (RT, FT)
6. InstantLab – online Learning

Themen im Detail

1. PGASUS Framework

- C++-Bibliothek für NUMA-bewusste Programmierung
- Portieren nach Power / Linux / AIX
- hwloc() – Informationen von HMC/LPAR auswerten
- Zusammenhänge verstehen, messen
- Zusätzliches Thema: PGASUS auf GPUs portieren

2. Linux auf Power

- Anpassen, Erweitern, Kompilieren
- Kernel bauen
- SUPE-paper implementieren ([link](#))

Masterthemen

OSM Group

Chart 3

Themen im Detail (contd.)

3. OpenPower + CAPI ausprobieren

- Übersicht über CAPI-Projekte
- Karte auswählen – VHDL/FPGA-Entwicklung
- Messungen, Kostenmodell

4. Linux auf Power

- Paging in Richtung GPU erweitern
- Kernelerweiterung

5. MultiCore ARM // Heterogene Kernels

- Anwendungsfall Embedded Computing
- Umgang mit heterogenen CPUs/GPUs

Masterthemen

OSM Group

Chart 4

Themen im Detail (contd.)

7. „Cloud Bursting“ auf GPU-Ebene

- Für gewisse Tasks entfernte GPU statt lokaler CPU nutzen
- Energieoptimierte Ausführungsstrategien
- Plattformübergreifend (ARM, Power, x86_64)

8. Converging GPUs

- Bisher: Feingranulare Tasks lohnen sich wg. Overhead nicht
- SoC/APU-basierte GPUs haben keinen PCIe mehr
- Plattformübergreifend (ARM, Power, x86_64)

9. Isolationsmechanismen in GPU Virtualisierung

- Untersuchung bestehender Ansätze (PoC Angriffe)
- Kann Isolation durch SW-Ebene verbessert werden? Kosten?

Masterthemen

OSM Group

Chart 5

Themen im Detail (contd.)

10. Power: IBM Tools

- Ressource-Management, RackScale / Vgl. mit Intel
- „Hello World“ für Ressourcenvergabe (Demo/Messungen)
- Core -> Cloud – vgl. SSICLOPS

11. Multiple Virtual Address Spaces (The Machine)

- SpaceJMP paper
- Local / Global Descriptor Table (Segmentierung auf Intel)
- 72 Adress bits on Power
- Linux Kernel-Erweiterung

Masterthemen

OSM Group

Chart 6

Themen im Detail (contd.)

12. Distributed Shared Memory

- Wie kann Transactional Memory helfen
- Segmente vs. Pages vs. Objekte
 - Granularität des „Sharing“ – The Machine

13. Internet of Things (IoT) - Blockchain

- Prozessoren verschiedener Leistung
- Scheduling / Ressourcenvergabe
- Kommunikation / gemeinsamer Adressraum?
 - Tuple Spaces? PGAS?

Masterthemen

OSM Group

Chart 7

Themen im Detail (ctd.)



14. InstantLab

- LTI Learning Interface
- Integration mit OpenHPI

Masterthemen

OSM Group

Chart **8**

Themen im Detail (contd.) – Verlässlichkeit

14. Zuverlässigkeit und Hochverfügbarkeit

- Bft-smart
- RailCloud, Chubby, Zookeeper
- FT-HelloWorld, implementieren, simulieren, evaluieren

15. Fehlerinjektion

- Verlässlichkeitseigenschaften experimentell bewerten
- In verteilten / Cloud Systemen herausfordernd:
Verteiltes Debugging + Monitoring
- vProbe /VMI-PI

Masterthemen

OSM Group

Chart 9

Themen im Detail (contd.) – Verlässlichkeit

17. Latency Tolerance Middleware (Hystrix)

- Explicit fault tolerance vs. performance trade-off in distributed systems
- Latency tolerance: essential characteristic of scalable HA systems in the presence of timing faults / unreliable network
- Hystrix: software framework for latency tolerance + fault isolation
 - Open sourced by Netflix
 - Implement and evaluate a scenario
 - Here: apply this approach to different compiler versions

Masterthemen

OSM Group

Chart **10**

Themen im Detail (contd.) – Verlässlichkeit

18. An N-Version Compilation Framework (based on LLVM?)

- Software errors are often reasoned by the environment; here: compilation tools + parameters
- Compiler bugs+backdoors becoming a threat to dependability and security
- N-Version programming: implement the same algorithm in different ways to avoid design faults

Masterthemen

OSM Group

Chart **11**

Themen im Detail (contd.) – Verlässlichkeit

19. Cloud-based fuzzing: Project Springfield applied

- Fuzzing: automatically feeding APIs with random/faulty data
- More computational power → better coverage
- Distributed tools meanwhile available, e.g. Project Springfield
- Apply and evaluate fault model/coverage of such tool(s)

20. Mining Repositories for Software Fault Models

- Software fault models are diverse and hardly understood
- What information can be extracted from repo metadata?
- Evaluate statistic distribution of different bug types

Masterthemen

OSM Group

Chart **12**