TECHNISCHE UNIVERSITÄT DRESDEN

Faculty of Computer Science Institute of Systems Architecture, Operating Systems Group

RUNNING THE WORLD'S CODE FROM DRESDEN

HORST SCHIRMEIER, MICHAEL ROITZSCH

ORGANISATION

Professur Betriebssysteme

Barkhausen-Institut

Systems Software Research is Irrelevant

Rob Pike (damals Bell Labs, heute Google)

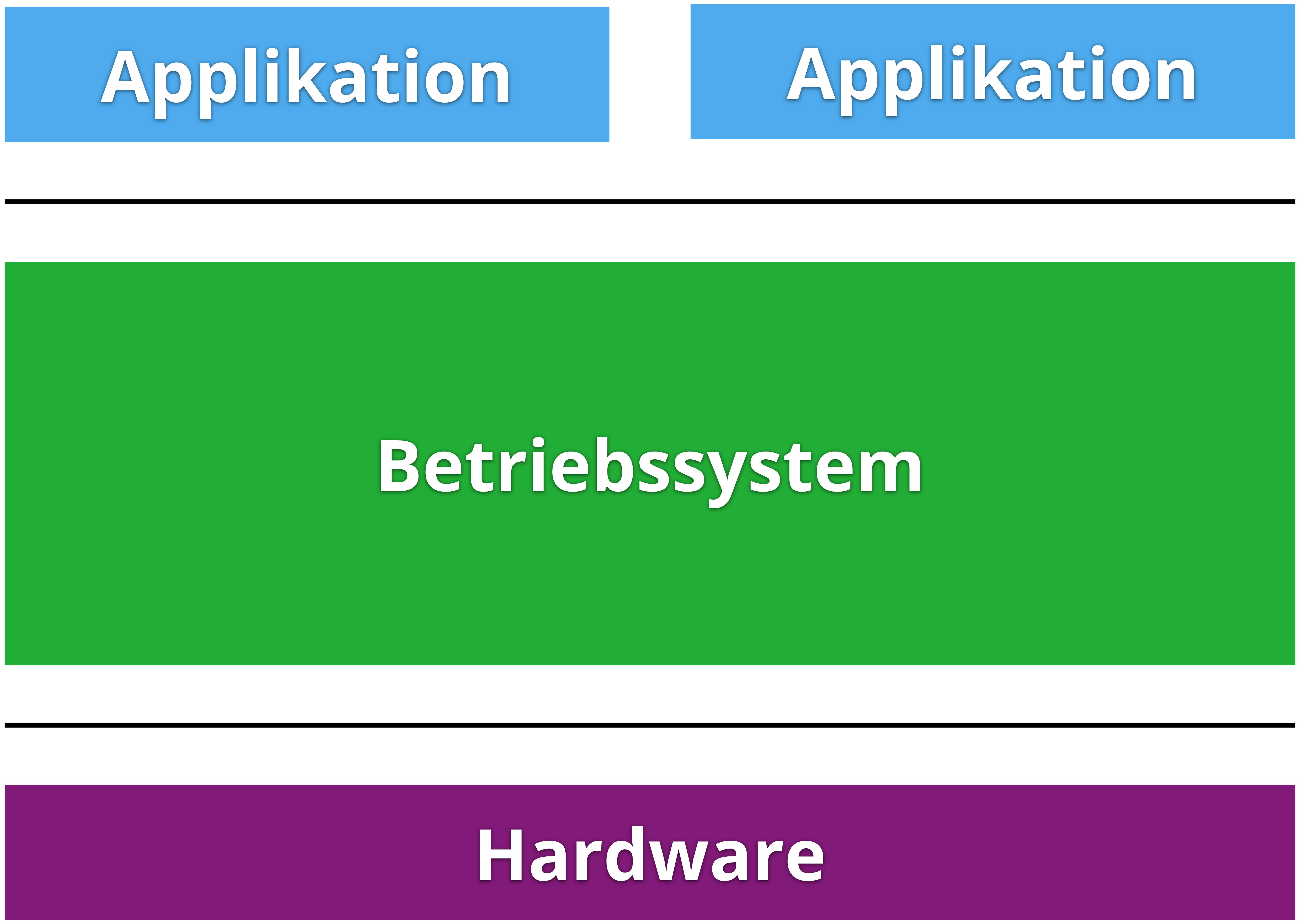
2000

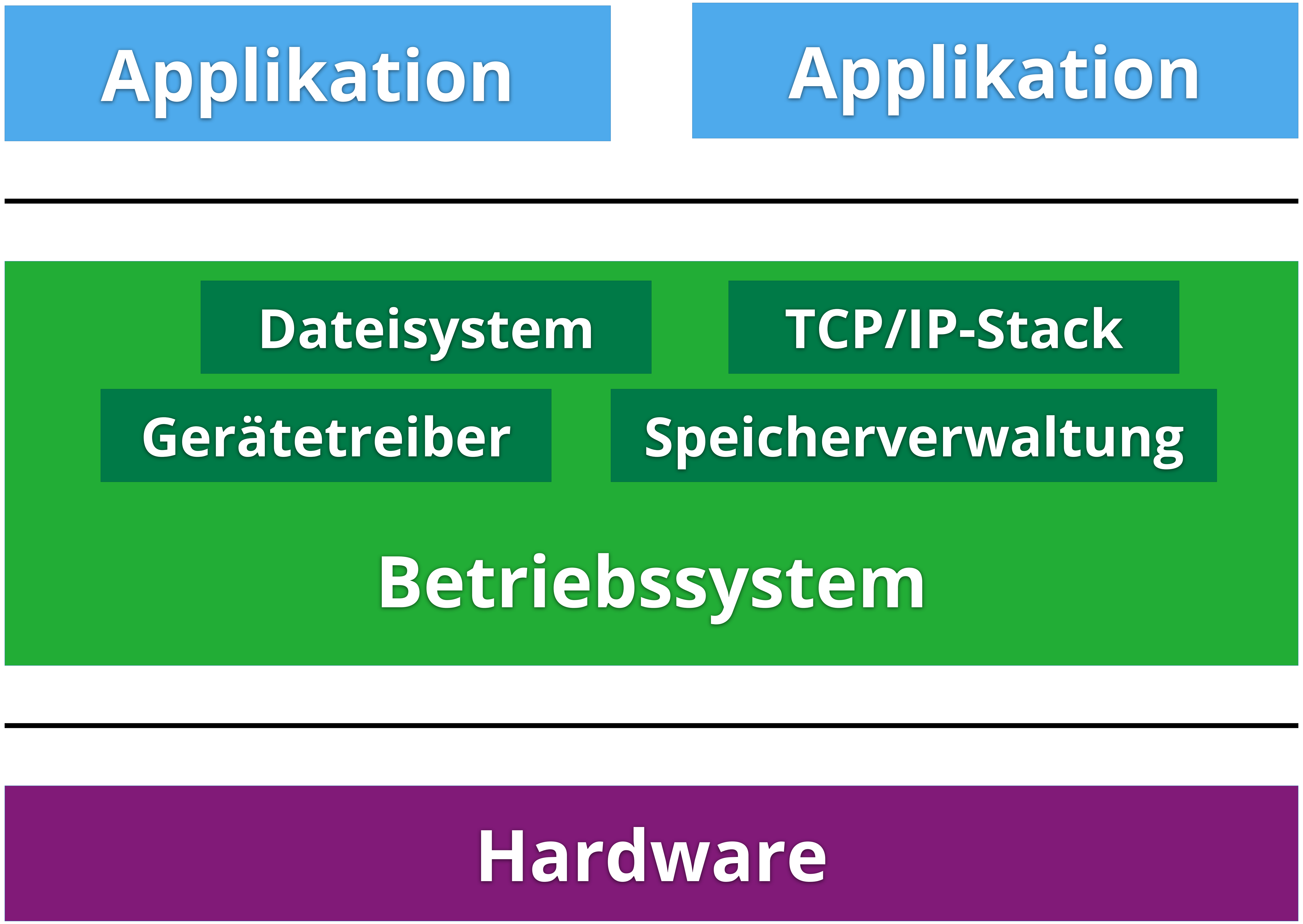
METRIKEN

- Publication Count
- Citation Count
- H-Index

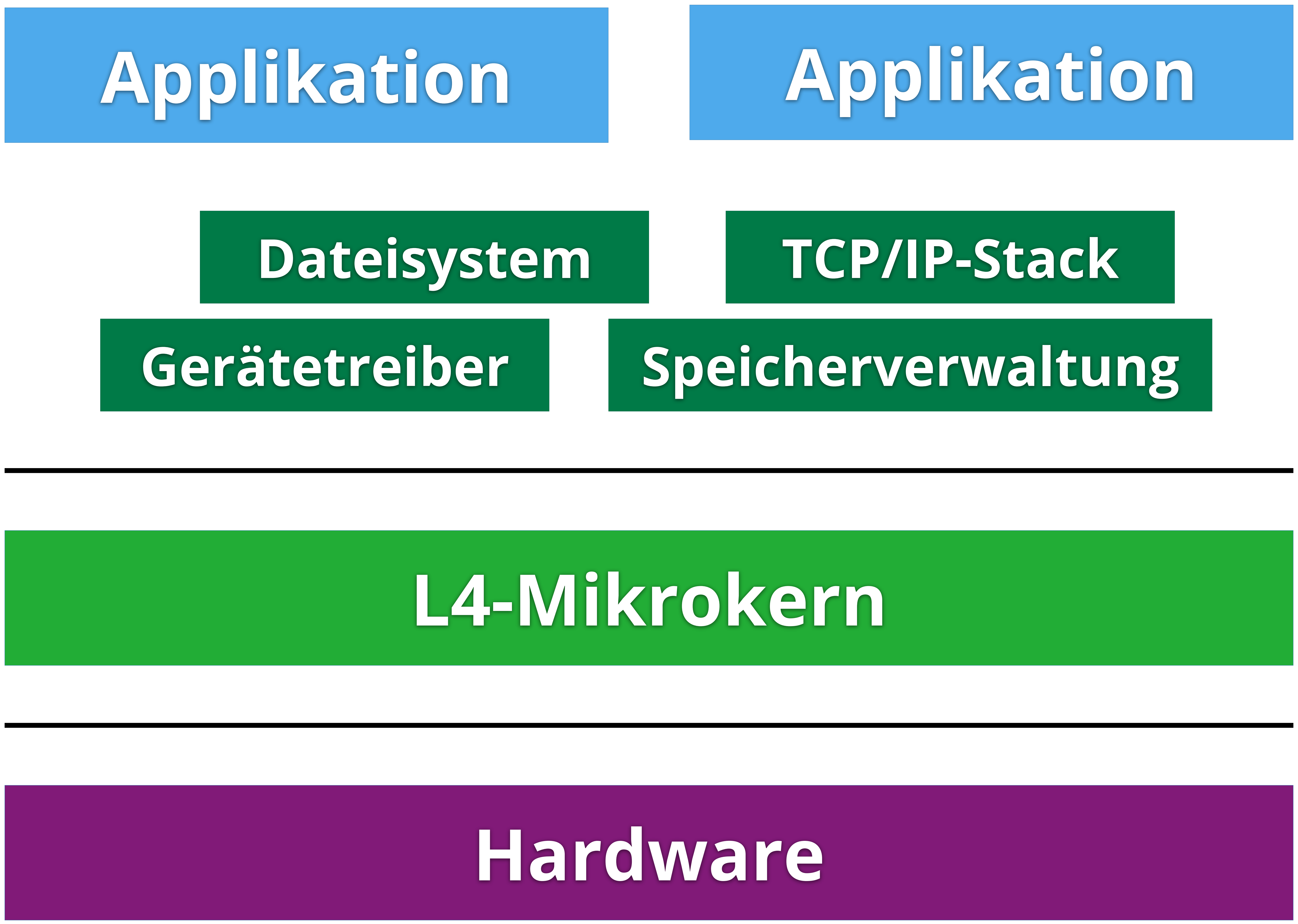
LEITLINIE

Funktionsfähige Systeme, die ein Problem mit praktischer Relevanz lösen.

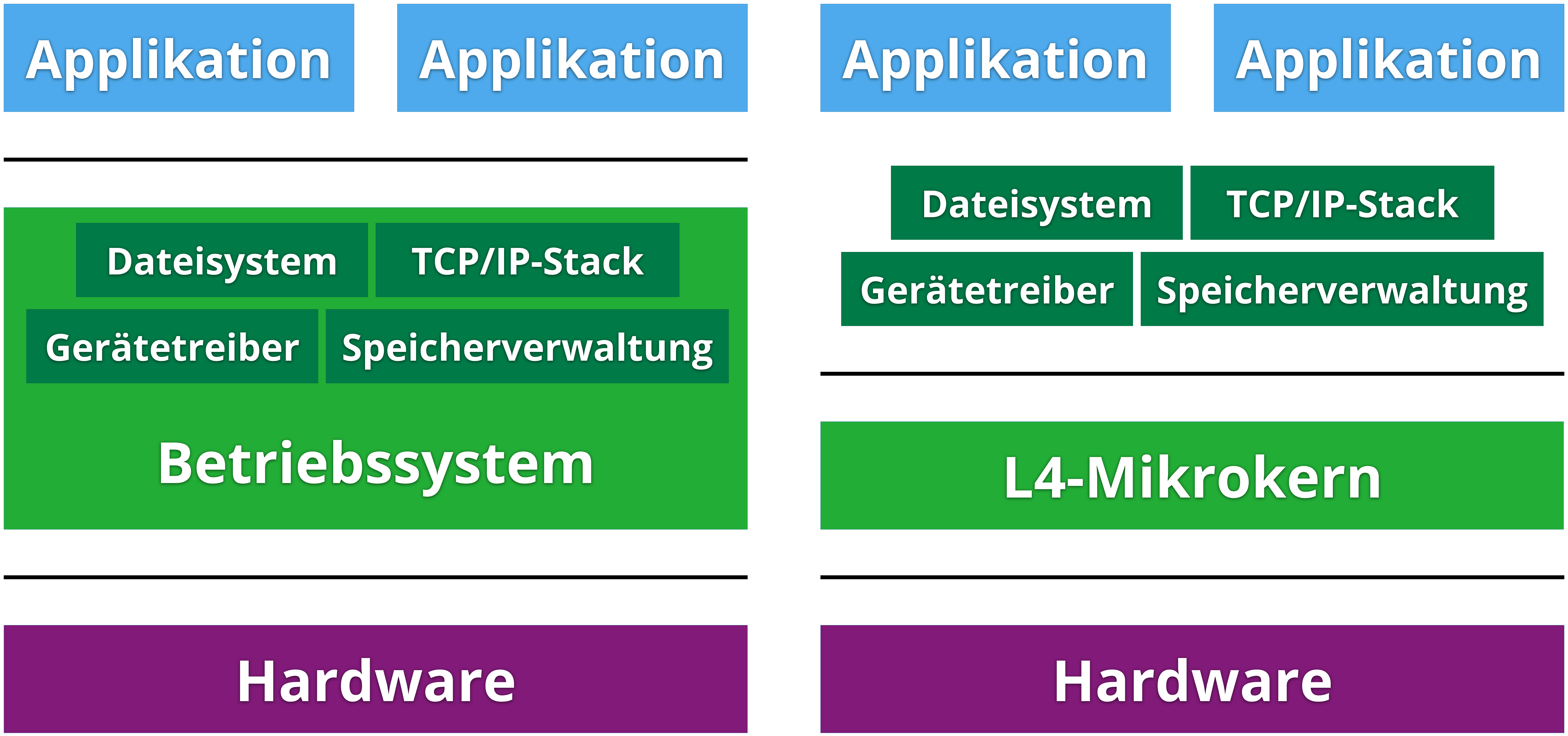

Applikation
Applikation
Betriebssystem
Hardware


Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
Betriebssystem
Hardware

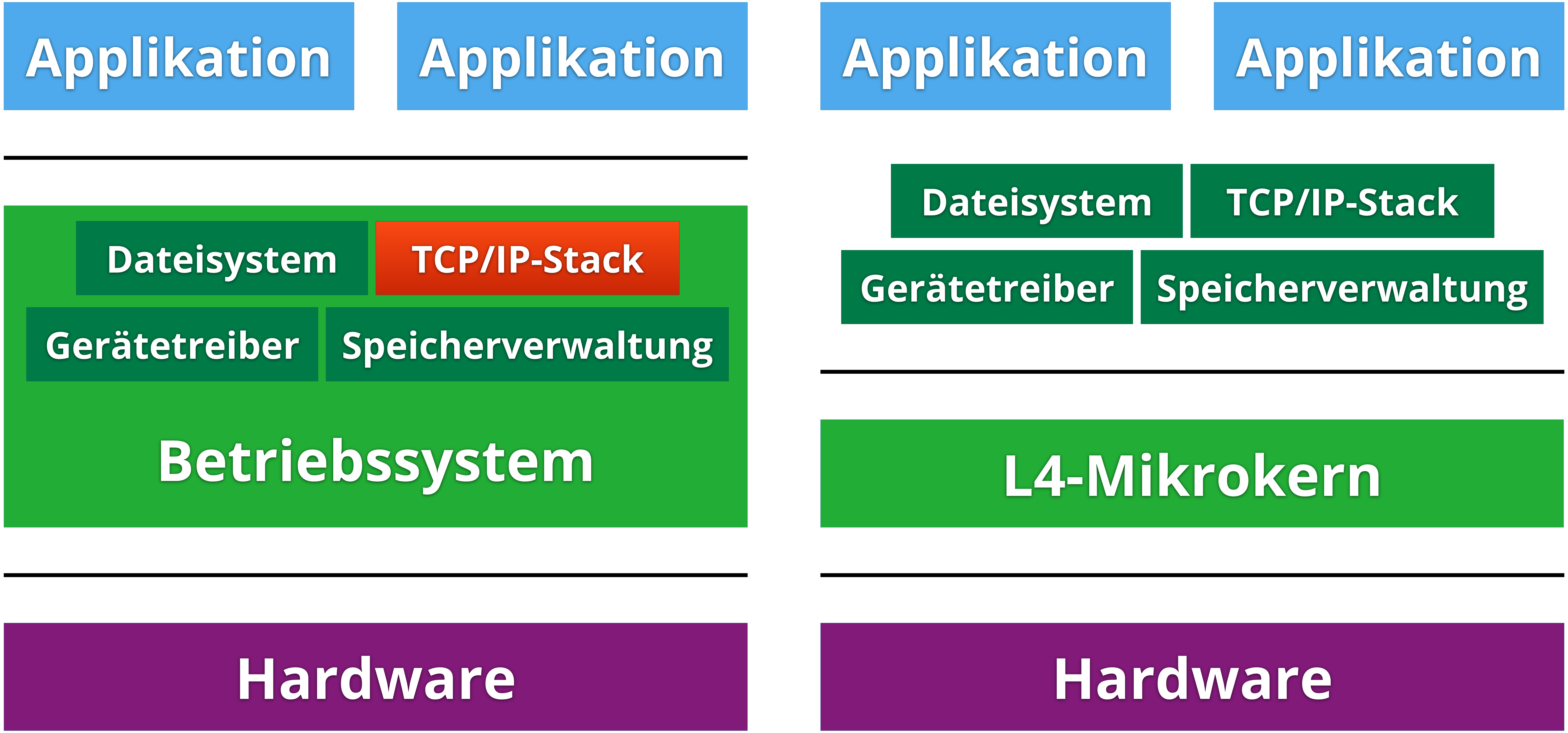
MIKROKERN


Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
L4-Mikrokern
Hardware

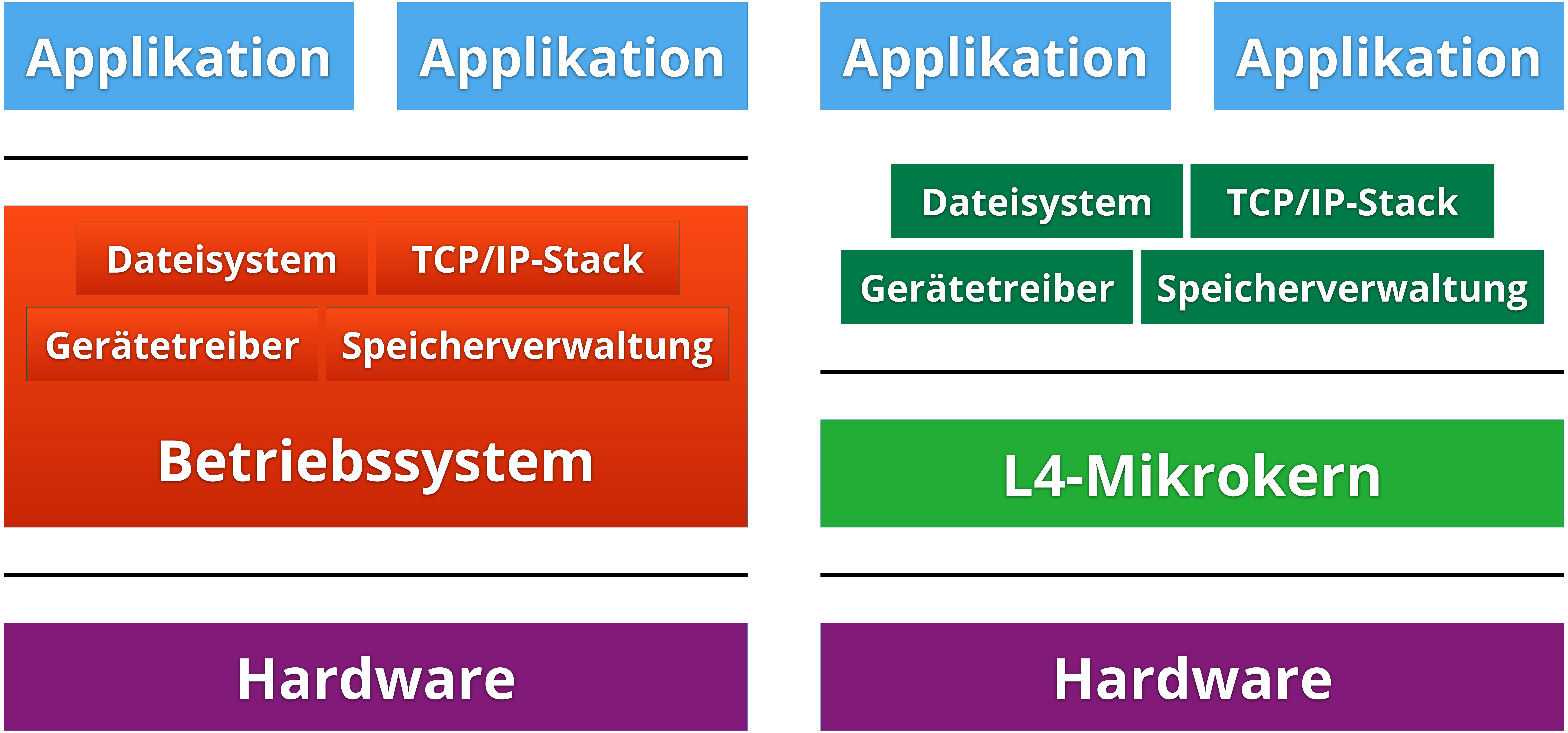
MIKROKERN


Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
Betriebssystem
Hardware
Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
L4-Mikrokern
Hardware

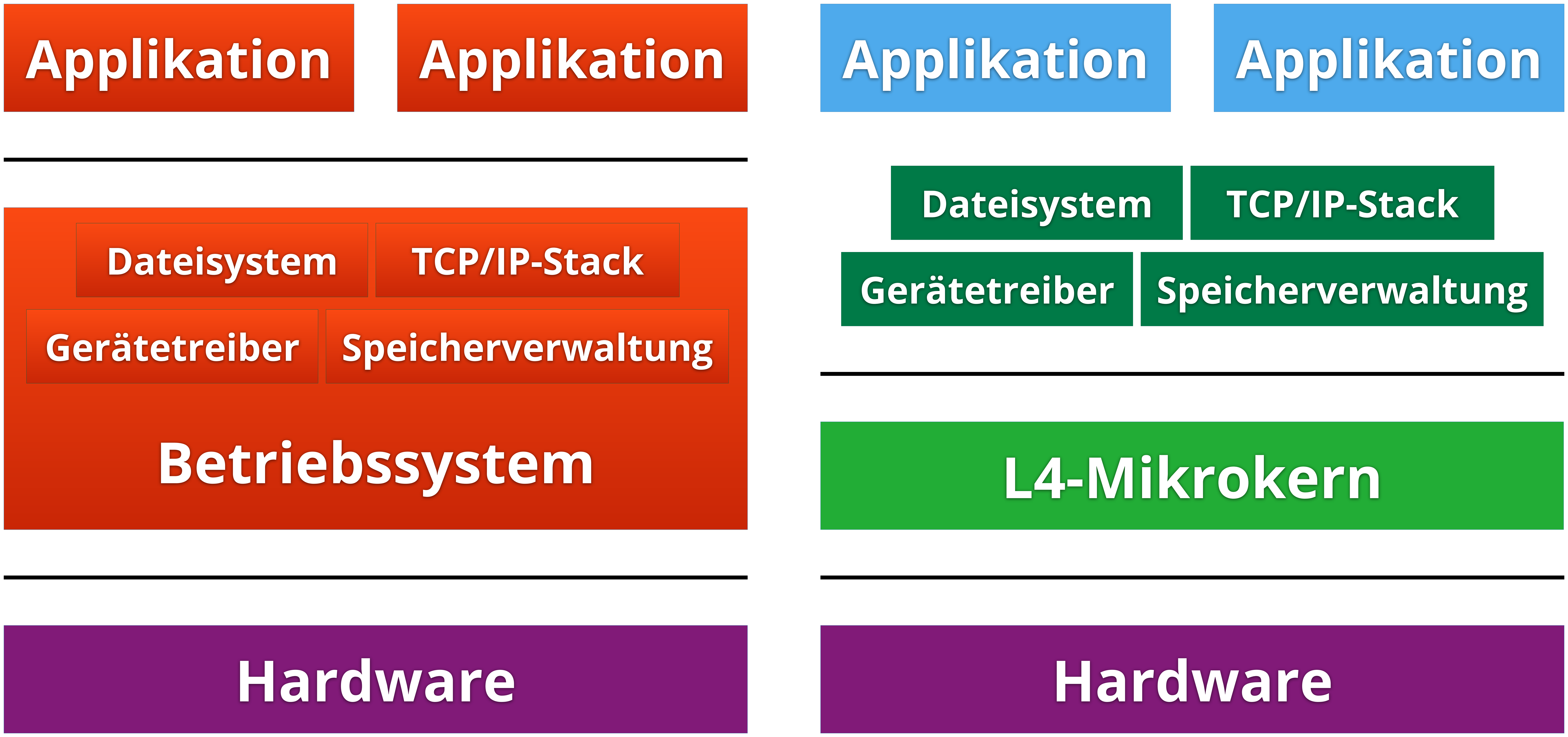
MIKROKERN


Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
Betriebssystem
Hardware
Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
L4-Mikrokern
Hardware

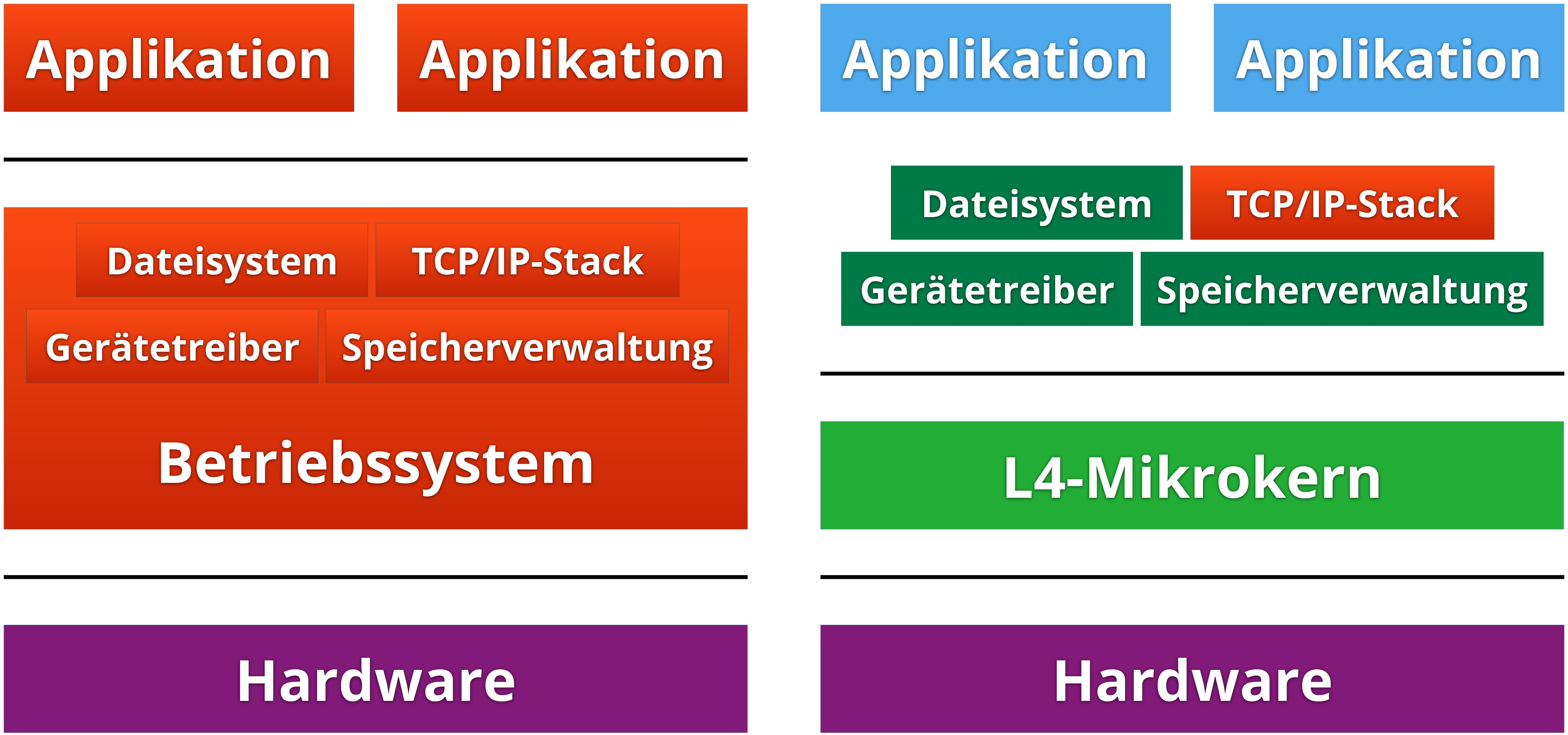
MIKROKERN


Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
Betriebssystem
Hardware
Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
L4-Mikrokern
Hardware

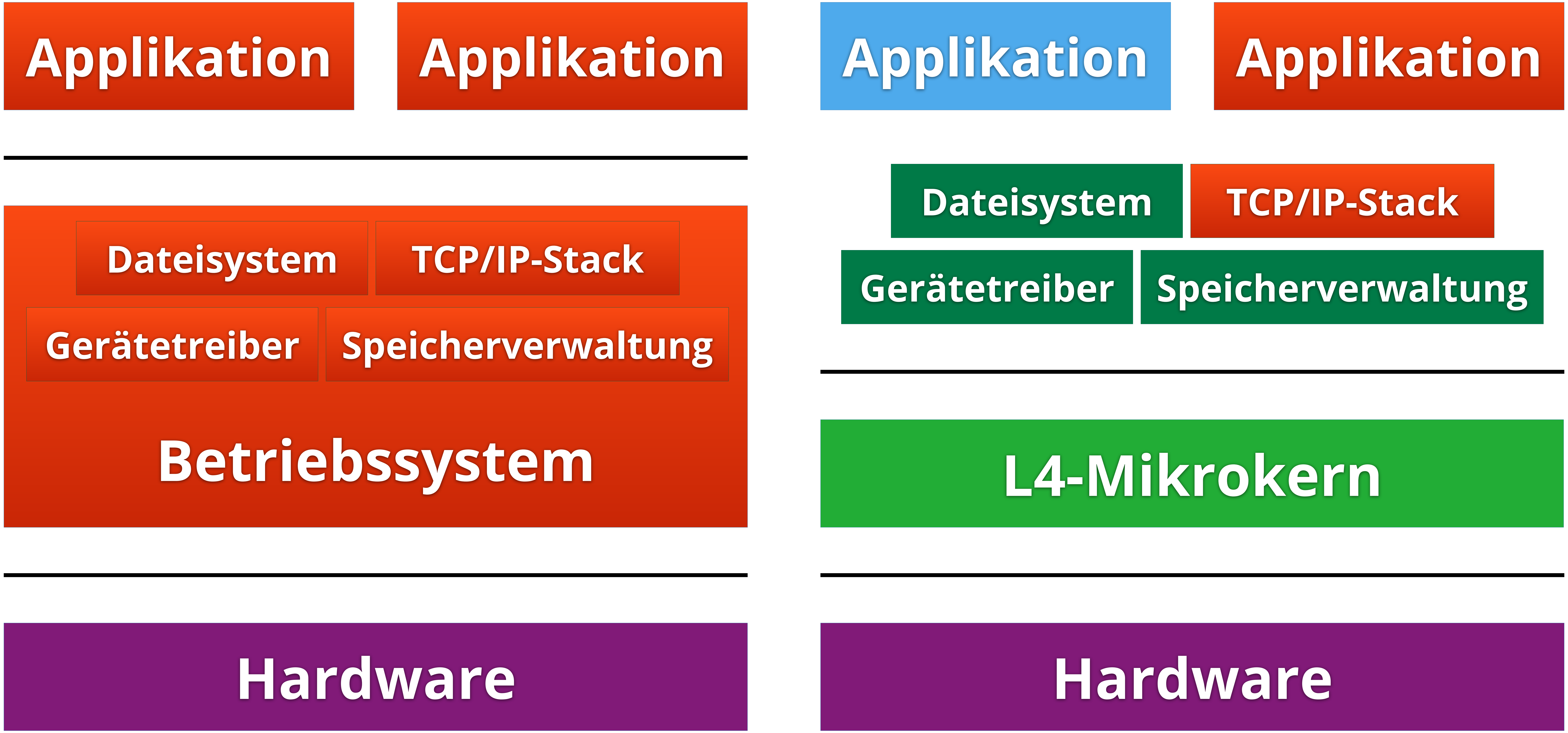
MIKROKERN


Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
Betriebssystem
Hardware
Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
L4-Mikrokern
Hardware

MIKROKERN


Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
Betriebssystem
Hardware
Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
L4-Mikrokern
Hardware

MIKROKERN


Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
Betriebssystem
Hardware
Applikation
Applikation
Dateisystem
TCP/IP-Stack
Gerätetreiber
Speicherverwaltung
L4-Mikrokern
Hardware

TRUSTED COMPUTING BASE

- **Trusted Computing Base (TCB):** Menge an Komponenten, denen eine Anwendung für das Erbringen ihrer Funktionalität vertrauen muss
- abhängig von Anwendung und Funktionalität
- Mikrokerne ermöglichen minimale, anwendungsspezifische TCBs

Mikrokerne helfen, wachsende Komplexität von Systemen durch Zerlegung beherrschbar zu machen.

fTPM: A Software-only Implementation of a TPM Chip

Himanshu Raj,[*] Stefan Saroiu, Alec Wolman, Ronald Aigner, Jeremiah Cox, Paul England, Chris Fenner, Kinshuman Kinshumann, Jork Loeser, Dennis Mattoon, Magnus Nystrom, David Robinson, Rob Spiger, Stefan Thom, and David Wooten
Microsoft

Abstract: *Commodity CPU architectures, such as ARM and Intel CPUs, have started to offer trusted computing features in their CPUs aimed at displacing dedicated trusted hardware. Unfortunately, these CPU architectures raise serious challenges to building trusted systems because they omit providing secure resources outside the CPU perimeter.*

This paper shows how to overcome these challenges to build software systems with security guarantees similar to those of dedicated trusted hardware. We present the design and implementation of a firmware-based TPM 2.0 (fTPM) leveraging ARM TrustZone. Our fTPM is the reference implementation of a TPM 2.0 used in millions of mobile devices. We also describe a set of mechanisms needed for the fTPM that can be useful for building more sophisticated trusted applications beyond just a TPM.

Secure Enclave

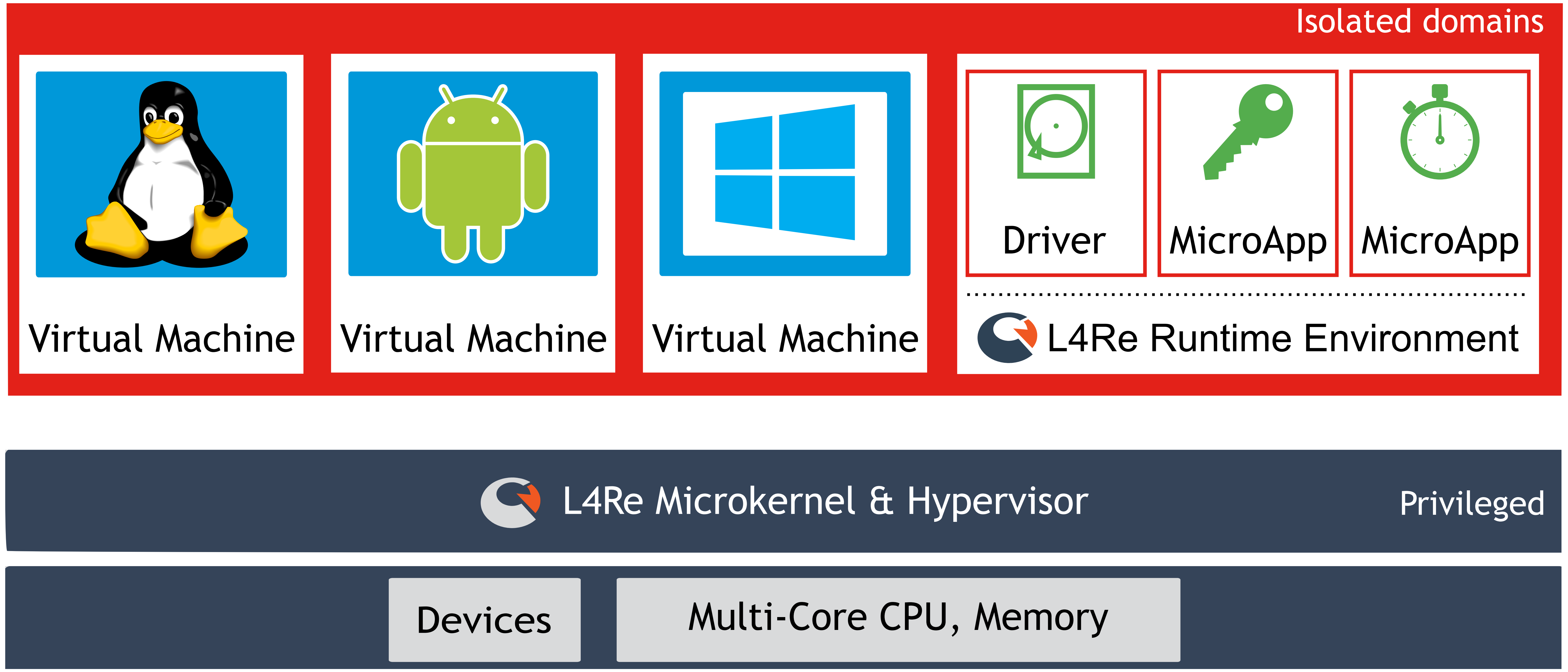
The Secure Enclave is a coprocessor fabricated within the system on chip (SoC). It uses encrypted memory and includes a hardware random number generator. The Secure Enclave provides all cryptographic operations for **Data Protection** key management and maintains the integrity of Data Protection even if the kernel has been compromised. Communication between the Secure Enclave and the application processor is isolated to an interrupt-driven mailbox and shared memory data buffers.

The Secure Enclave includes a dedicated Secure Enclave Boot ROM. Similar to the application processor Boot ROM, the Secure Enclave Boot ROM is immutable code that establishes the hardware root of trust for the Secure Enclave.

The Secure Enclave runs a Secure Enclave OS based on an Apple-customized version of the L4 microkernel. This Secure Enclave OS is signed by Apple, verified by the Secure Enclave Boot ROM, and updated through a personalized software update process.

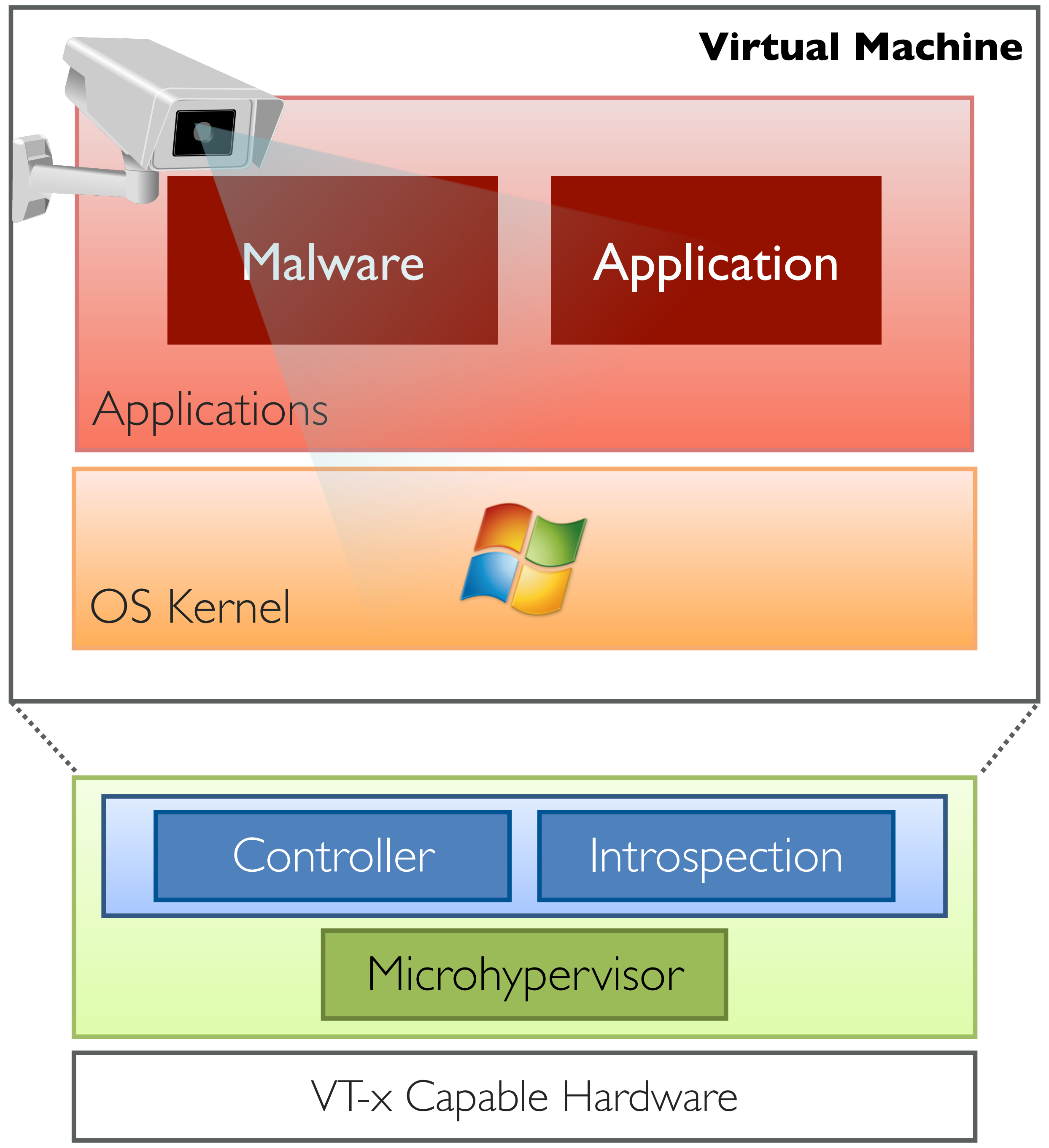
FIRMEN IN DRESDEN

KERNKONZEPT



FIRMEN IN DRESDEN

CYBERUS TECHNOLOGY



CYBERUS
TECHNOLOGY

Meltdown: Reading Kernel Memory from User Space

Moritz Lipp[1], Michael Schwarz[1], Daniel Gruss[1], Thomas Prescher[2], Werner Haas[2], Anders Fogh[3], Jann Horn[4], Stefan Mangard[1], Paul Kocher[5], Daniel Genkin[6,9], Yuval Yarom[7], Mike Hamburg[8]

[1] *Graz University of Technology,* [2] *Cyberus Technology GmbH,*
[3] *G-Data Advanced Analytics,* [4] *Google Project Zero,*
[5] *Independent (www.paulkocher.com),* [6] *University of Michigan,*
[7] *University of Adelaide & Data61,* [8] *Rambus, Cryptography Research Division*

CYBERUS TECHNOLOGY

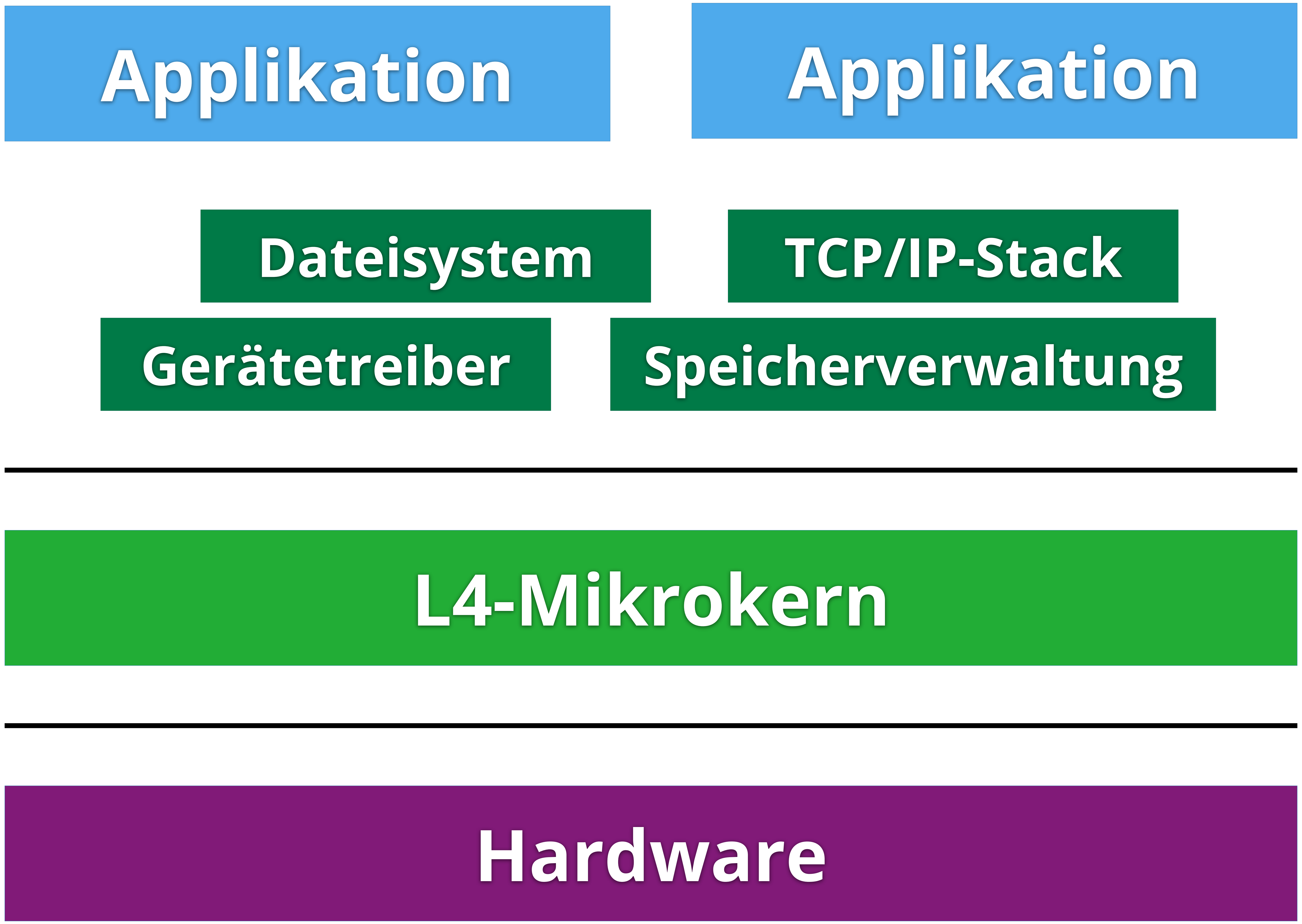
LazyFP: Leaking FPU Register State using Microarchitectural Side-Channels

Julian Stecklina
Amazon Development Center Germany GmbH
jsteckli@amazon.de

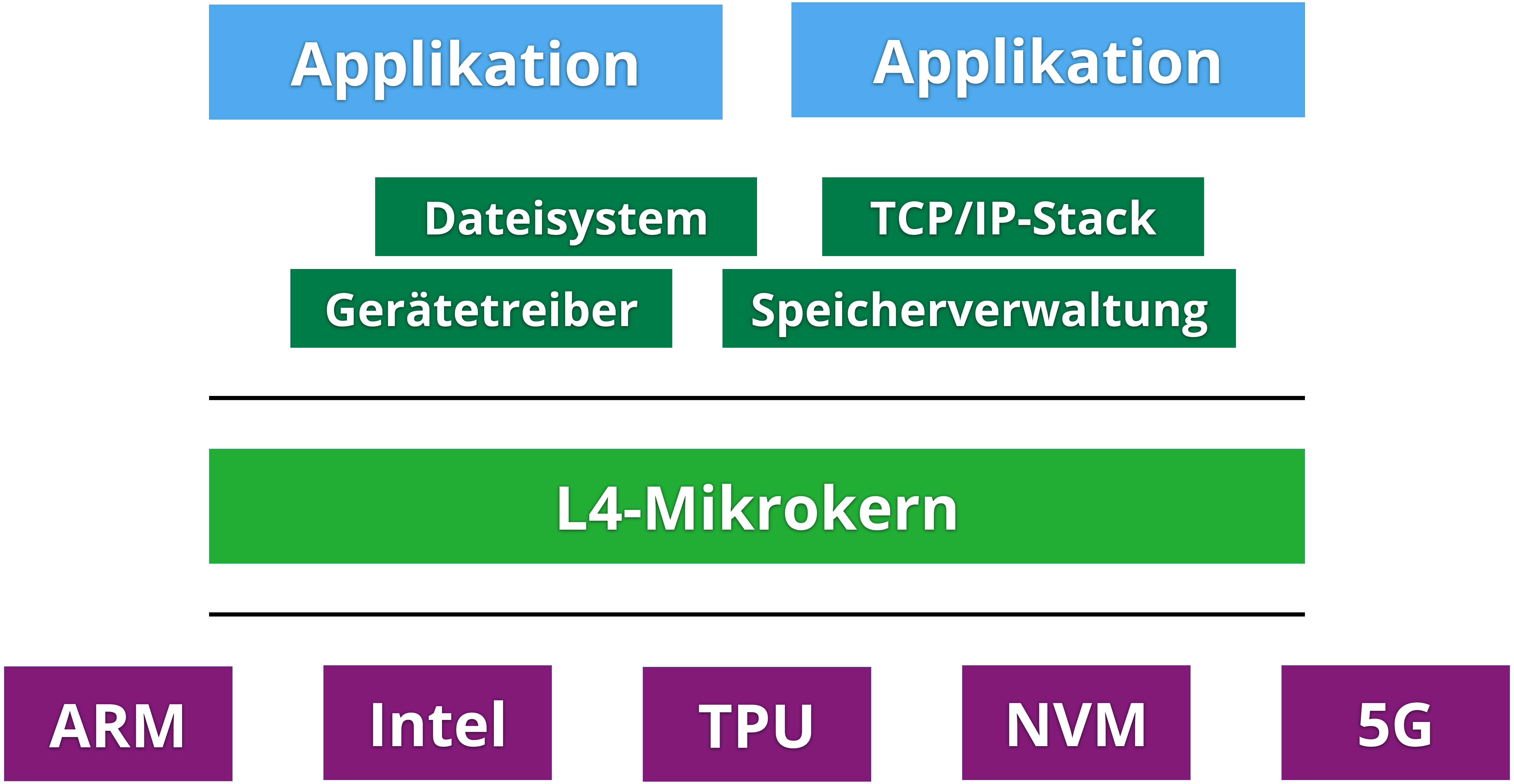
Thomas Prescher
Cyberus Technology GmbH
thomas.prescher@cyberus-technology.de

FIRMEN IN DRESDEN

FORSCHUNGSTHEMEN



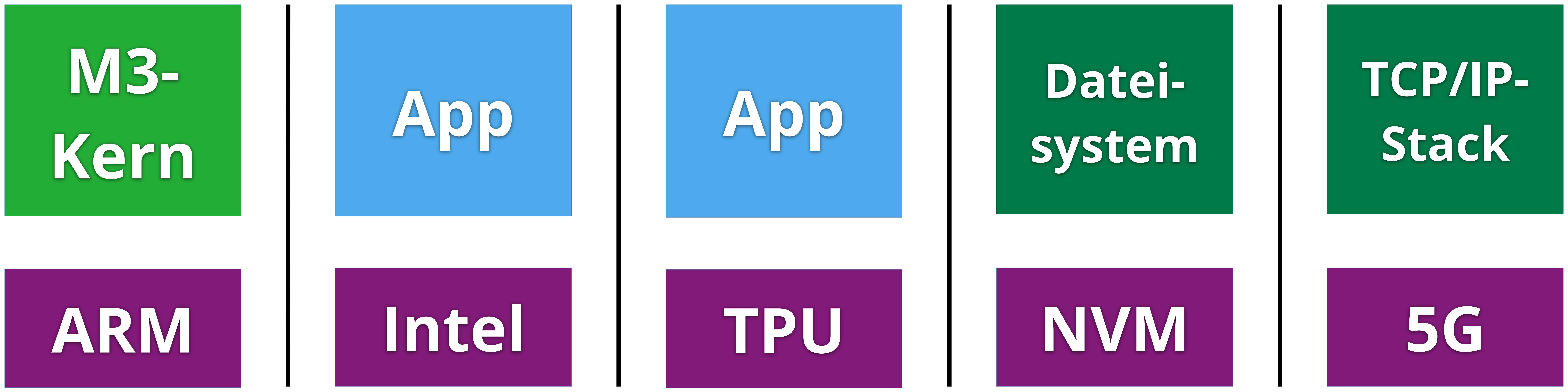
FORSCHUNGSTHEMEN



FORSCHUNGSTHEMEN

barkhausen institut

Plattform für IoT



TECHNISCHE UNIVERSITÄT DRESDEN

Weitere Forschungsthemen

- Umgang mit **Komplexität**
  - konstruktiv (L4, M^3)
  - analytisch (Projekt LockDoc)

- **Nichtfunktionale Eigenschaften**
  - *Security*
  - *Safety*/Fehlertoleranz (Projekte: DanceOS, FAIL*)
  - Zeitverhalten
  - Energie
- **Hardware**-Entwicklungen
  - Disruptive Speichertechnologien (Projekte VAMPIR, FOSSIL)

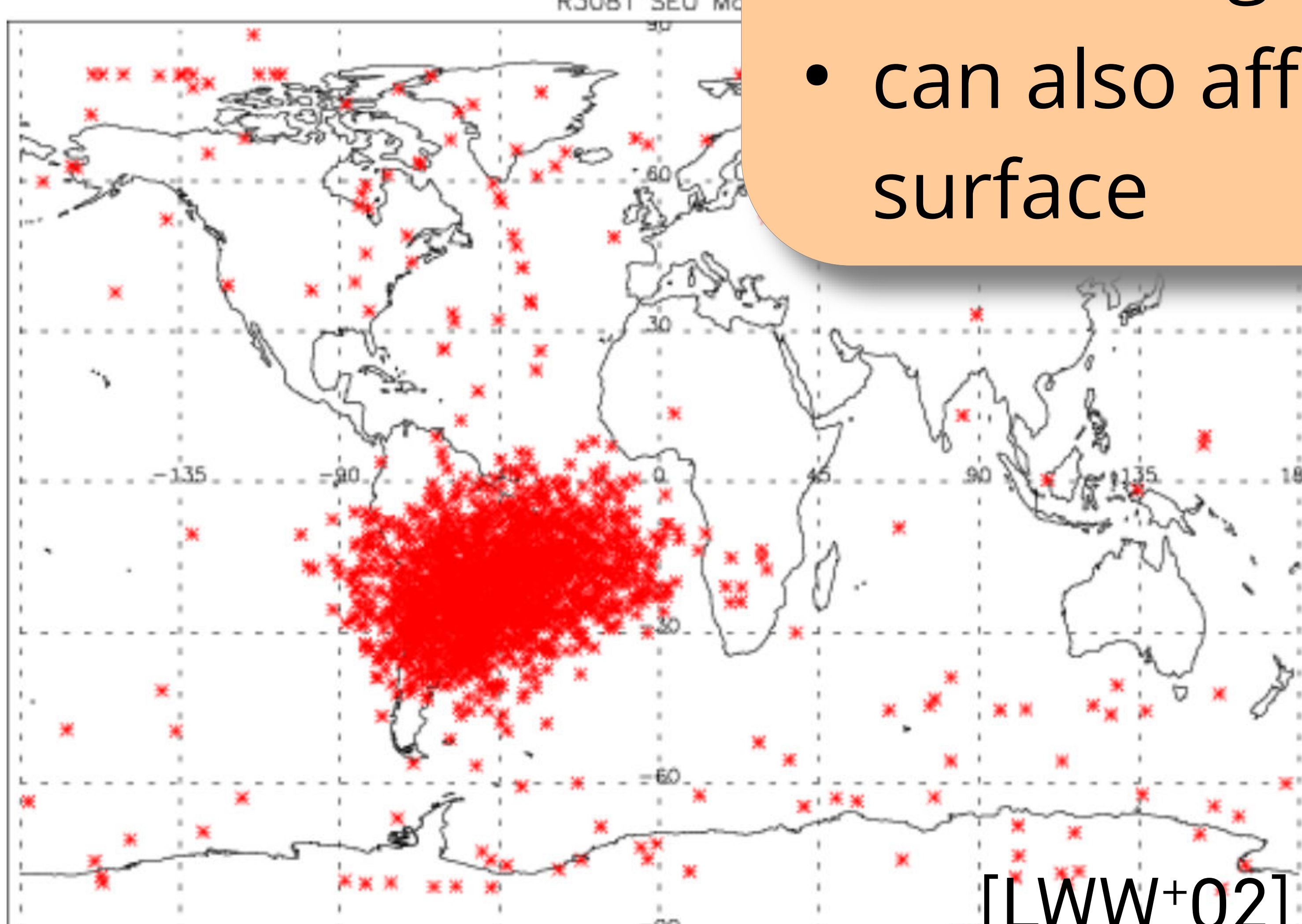
TECHNISCHE UNIVERSITÄT DRESDEN

2016: Research Satellite “Hitomi”

- JAXA X-ray/Gamma-ray astronomy satellite
- Intermittent failures of the *Star Tracker* component for attitude determination over the **South Atlantic Anomaly**
  - Chain of foll… …ures
  - Satellite rot…

Soft Errors

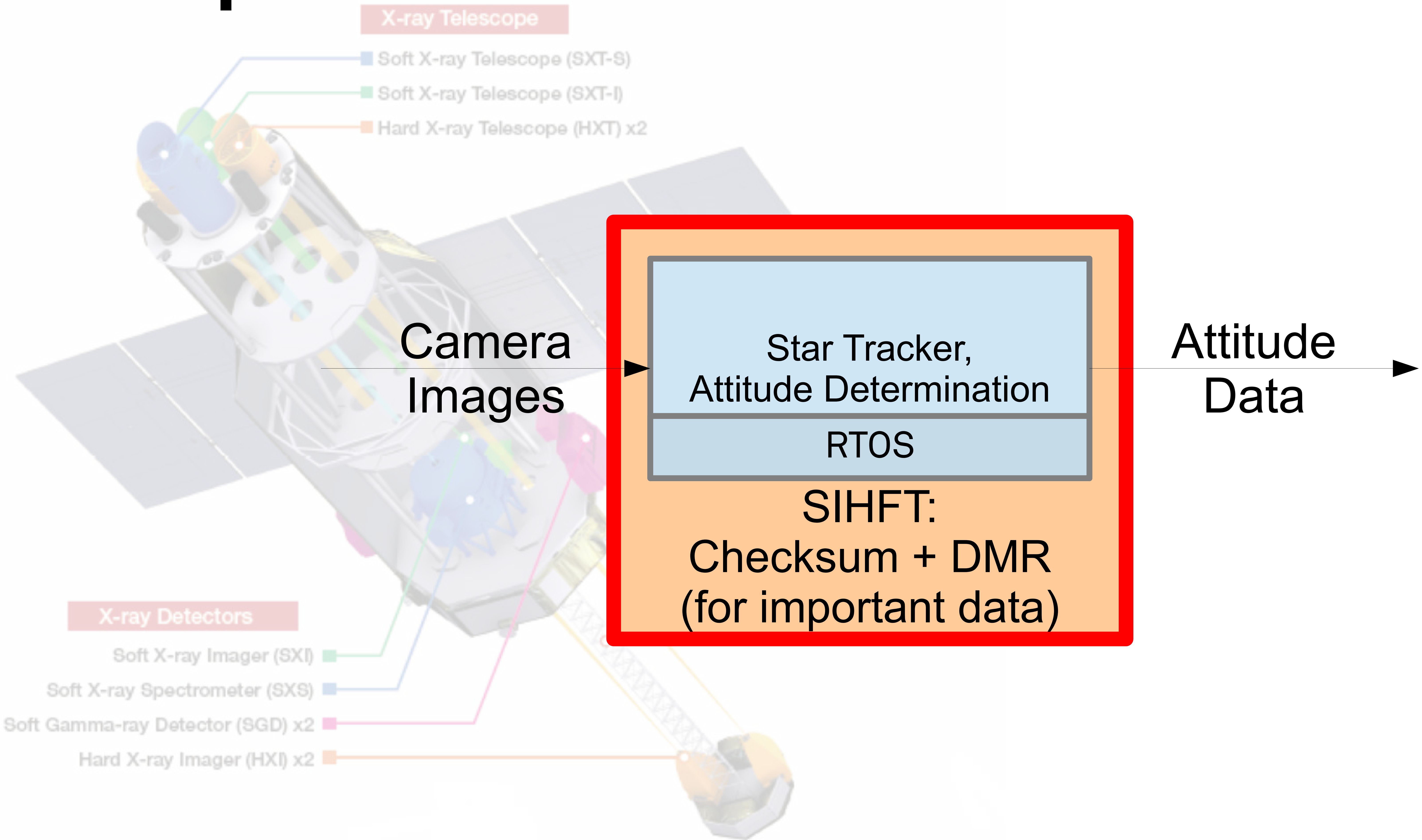
- Transient hardware errors
- caused e.g. by cosmic radiation
- can also affect devices on the earth’s surface



[LWW+02]

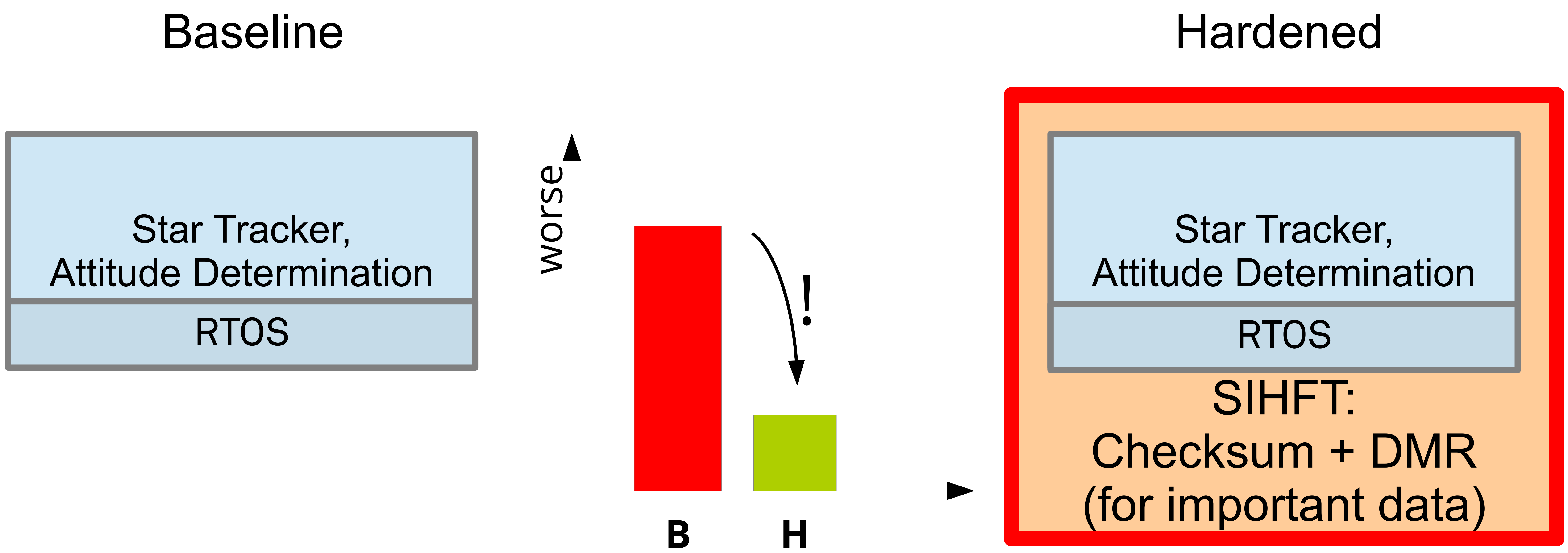
[Wit16]

Example: Satellite Attitude Control



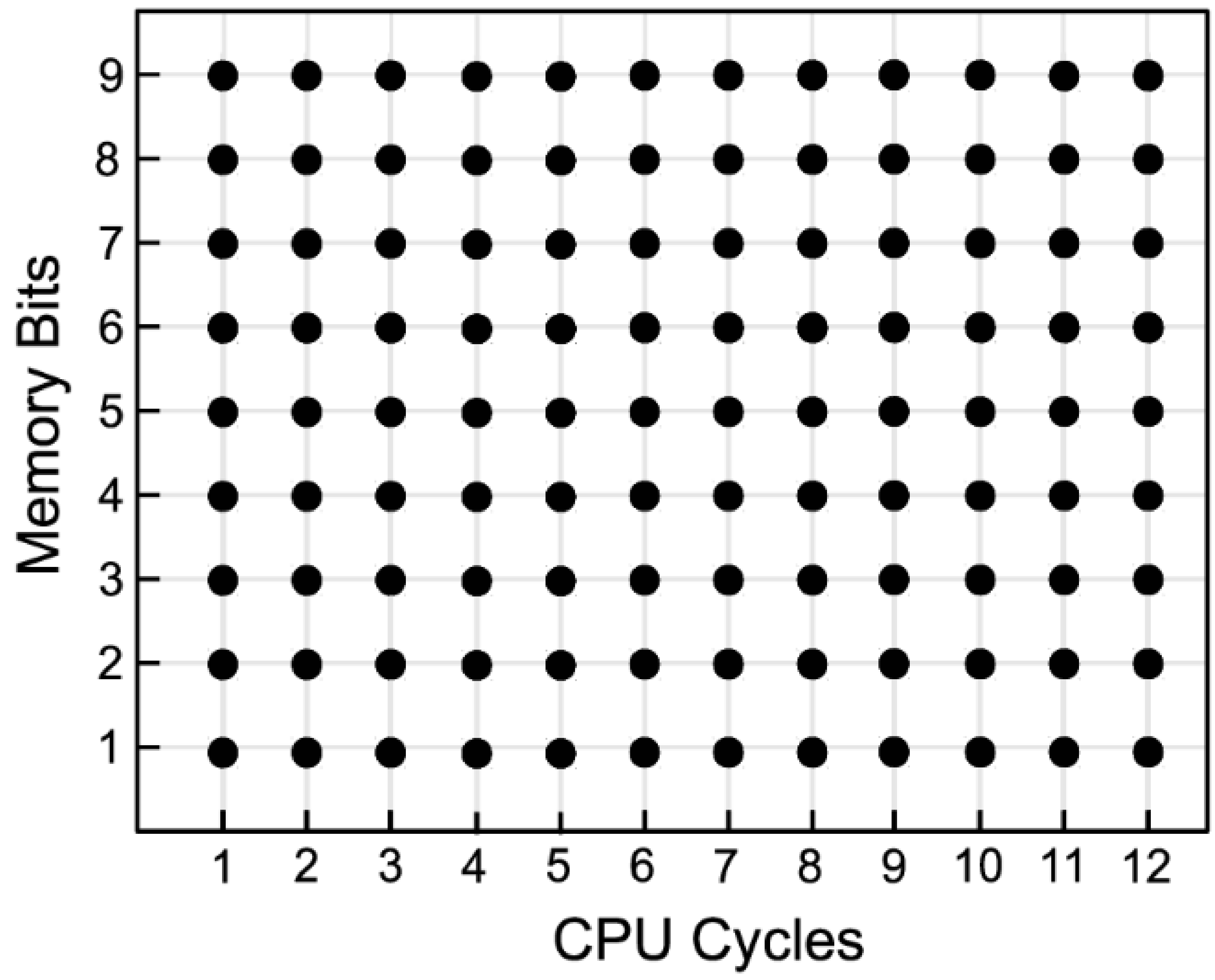
Variant Comparison

- Which variant is **„better“**?
- i.e. **less probable** to behave „incorrectly“?



Measurement: Simulator-based Fault Injection

- **Fault model:**
  e.g. uniformly distributed single-bit flips in memory
- Possible FI-experiment results (simplified):
  - **incorrect output**
    (*silent data corruption*, SDC)
    = **failure**
  - **everything else**
    = no failure / **benign**
- Take **N** samples from *fault space:*
  Run **N** FI experiments,
  count failures **F** ≤ **N**



Calculating P(Failure)

Assumption:
Very low hardware fault rate,
2 or more faults per
system run negligible

$$P(\text{Failure}) =$$

$$P(\text{Failure}|0\,\text{Faults} \vee 1\,\text{F.} \vee 2\,\text{F.} \vee 3\,\text{F.} \vee \ldots) \approx$$

$$P(\text{Failure}|1\,\text{Fault}) \cdot P(1\,\text{Fault}) =$$

$$\frac{F}{N} \cdot P_\lambda(k=1) =$$

$$\frac{F}{N} \cdot g \cdot w \cdot e^{-gw} \quad \propto \frac{F}{N} \cdot w$$

Metric:
Extrapolated Absolute Failure Count (EAFC)

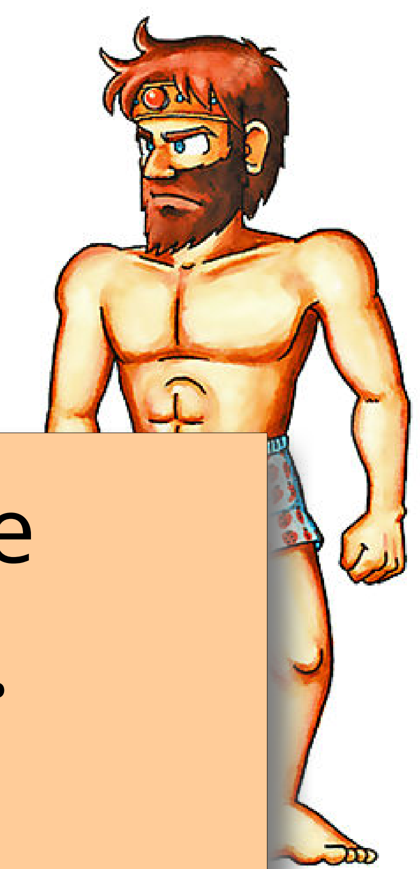
Assumption:
g very small

[DSN'15]

TECHNISCHE UNIVERSITÄT DRESDEN

Calculating P(Failure)

- **Intuitive analogy:**
  Two knights crossing battlefield *from one end to the other*, under *constant hail of arrows*



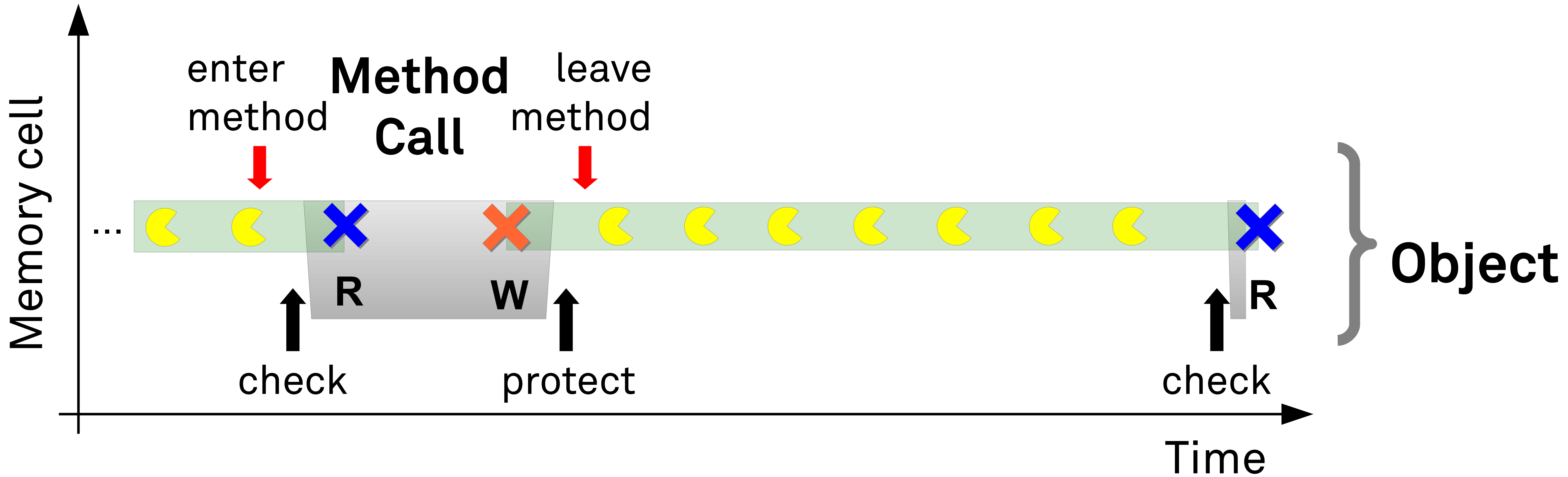
Software-implemented hardware fault tolerance
is a **trade-off between speed and hardening.**

Extrapolated Absolute Failure Count (EAFC) metric
captures this trade-off.

[DSN‘15], [EDCC’19] H. Schirmeier and M. Breddemann. *Quantitative cross-layer evaluation of transient-fault injection techniques for algorithm comparison.* In Proceedings of the 15th European Dependable Computing Conference (EDCC '19), pages 15–22, Piscataway, NJ, USA, Sept. 2019. IEEE.

SIHFT Mechanism: *Generic Object Protection*

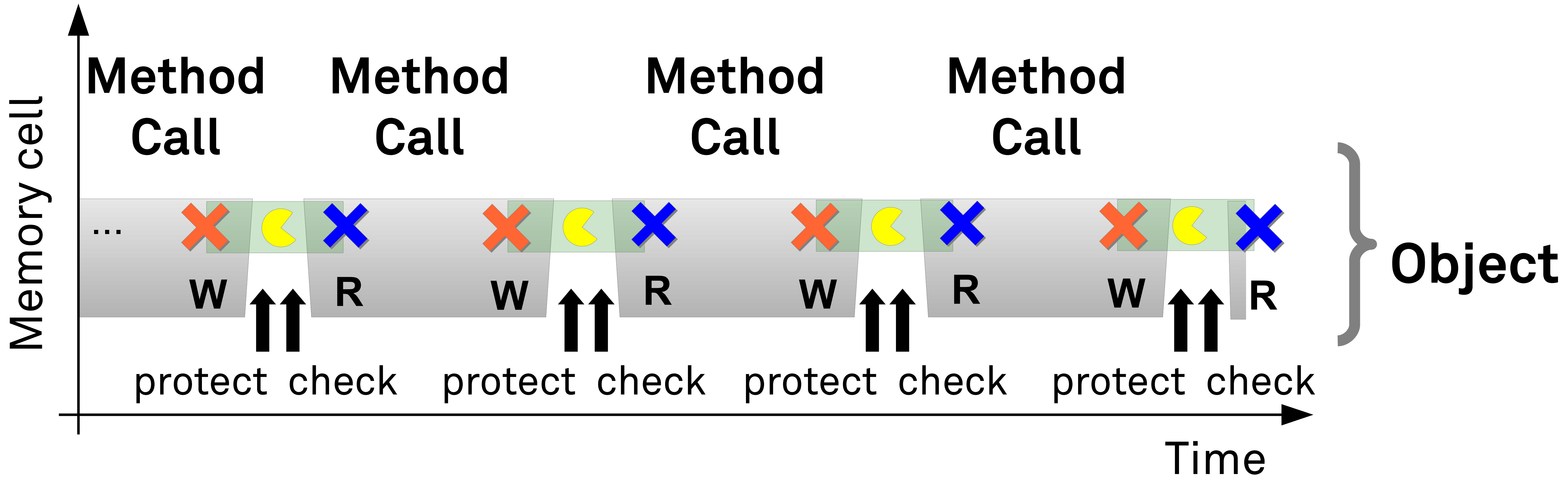
- **Basic idea:**
  - Add **redundancy** to kernel data **on object granularity**
  - Add redundancy **checks / updates** before/after data is used



[DSN’13] C. Borchert, H. Schirmeier, and O. Spinczyk. Generative software-based memory error detection and correction for operating system data structures. In Proceedings of the 43rd IEEE/IFIP International Conference on Dependable Systems and Networks (DSN '13), Piscataway, NJ, USA, June 2013. IEEE.
[TDSC’17] C. Borchert, H. Schirmeier, and O. Spinczyk. *Generic soft-error detection and correction for concurrent data structures.* IEEE Transactions on Dependable and Secure Computing, 14(1):22–36, Jan. 2017.

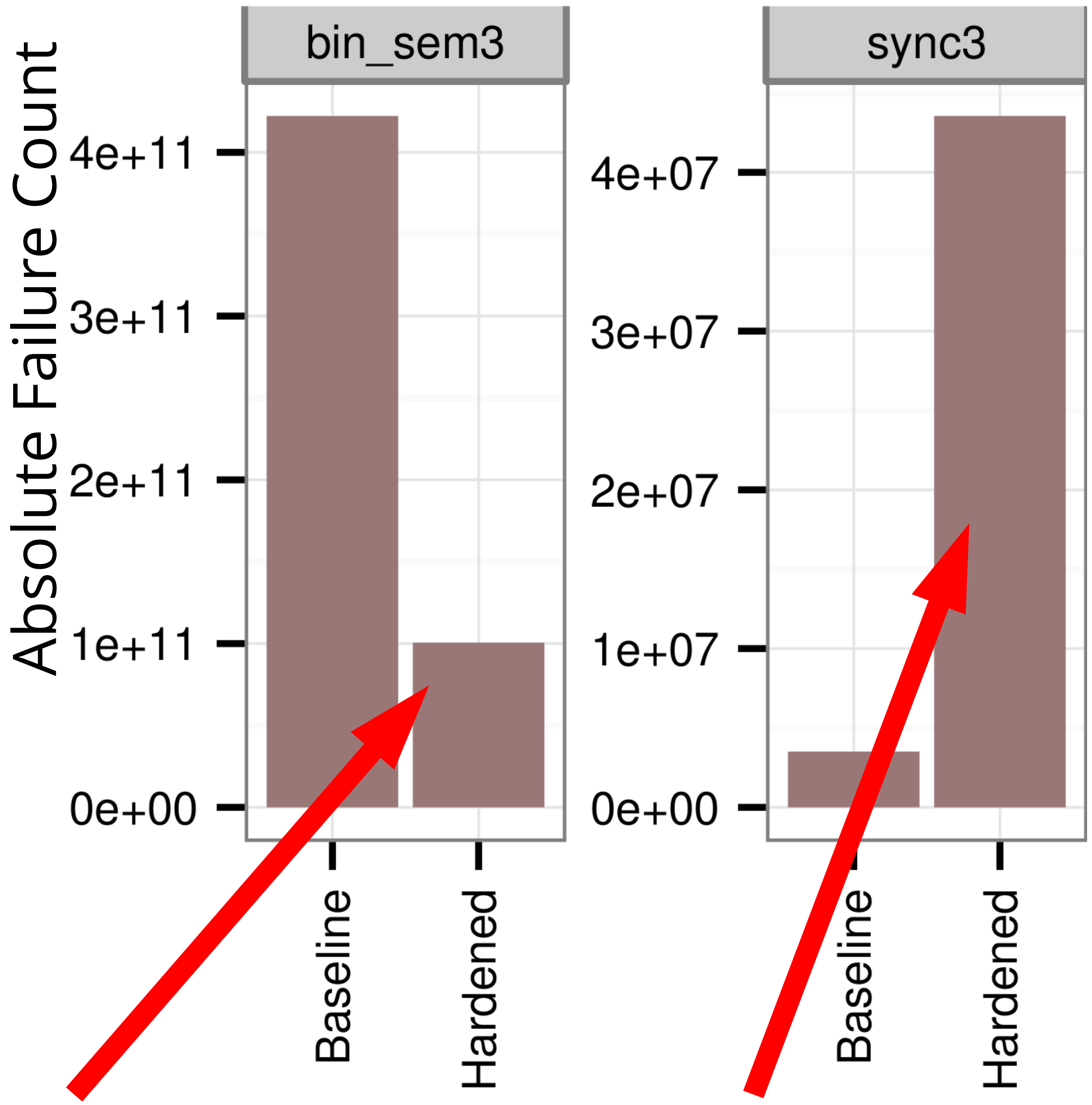
SIHFT Mechanism: *Generic Object Protection*

- **sync3 benchmark:** *Pathologic worst case* for GOP
  - System-call stress test, **no delay** between calls
  - Fault-resilience **gains minimal**
  - **Increased attack surface** due to longer runtime (check / protect)



[DSN'13], [TDSC'17]

Measurements (Example)



P(Failure) drops by ~4x

P(Failure) *increases* by ~12x!

[DSN‘15]

TECHNISCHE UNIVERSITÄT DRESDEN

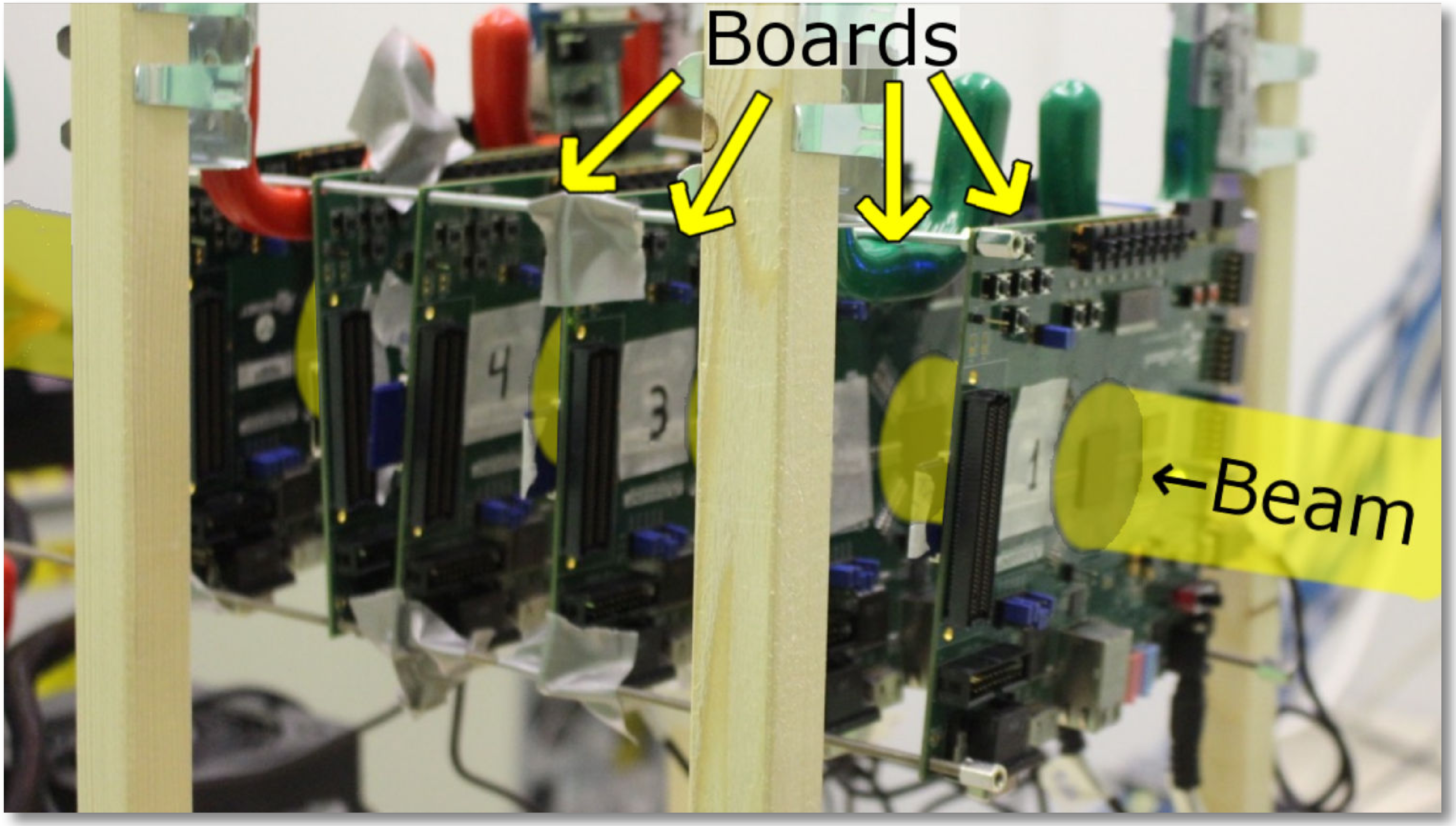
Reality Check: Real soft-errors?

- Iterative application of simulator-based FI
  - ➔ Optimal subset of eCos kernel classes protected with GOP
- **But:** Reliability improvement under **real soft-errors**?

- **Neutron-beam experiments** at Los Alamos Neutron Science Center (LANSCE), ICE II facility
  - Cooperation with **P. Rech** (UFRGS, Brazil) and **T.C. Santini** (U Tübingen)
  - eCos + GOP on ARM-A9 based embedded systems

[ARCS'17] T. Santini, C. Borchert, C. Dietrich, H. Schirmeier, M. Hoffmann, O. Spinczyk, D. Lohmann, F. R. Wagner, and P. Rech. *Effectiveness of software-based hardening for radiation-induced soft errors in real-time operating systems.* In Proceedings of the 30th International Conference on Architecture of Computing Systems (ARCS '17), pages 3–15, Cham, Switzerland, Apr. 2017. Springer.

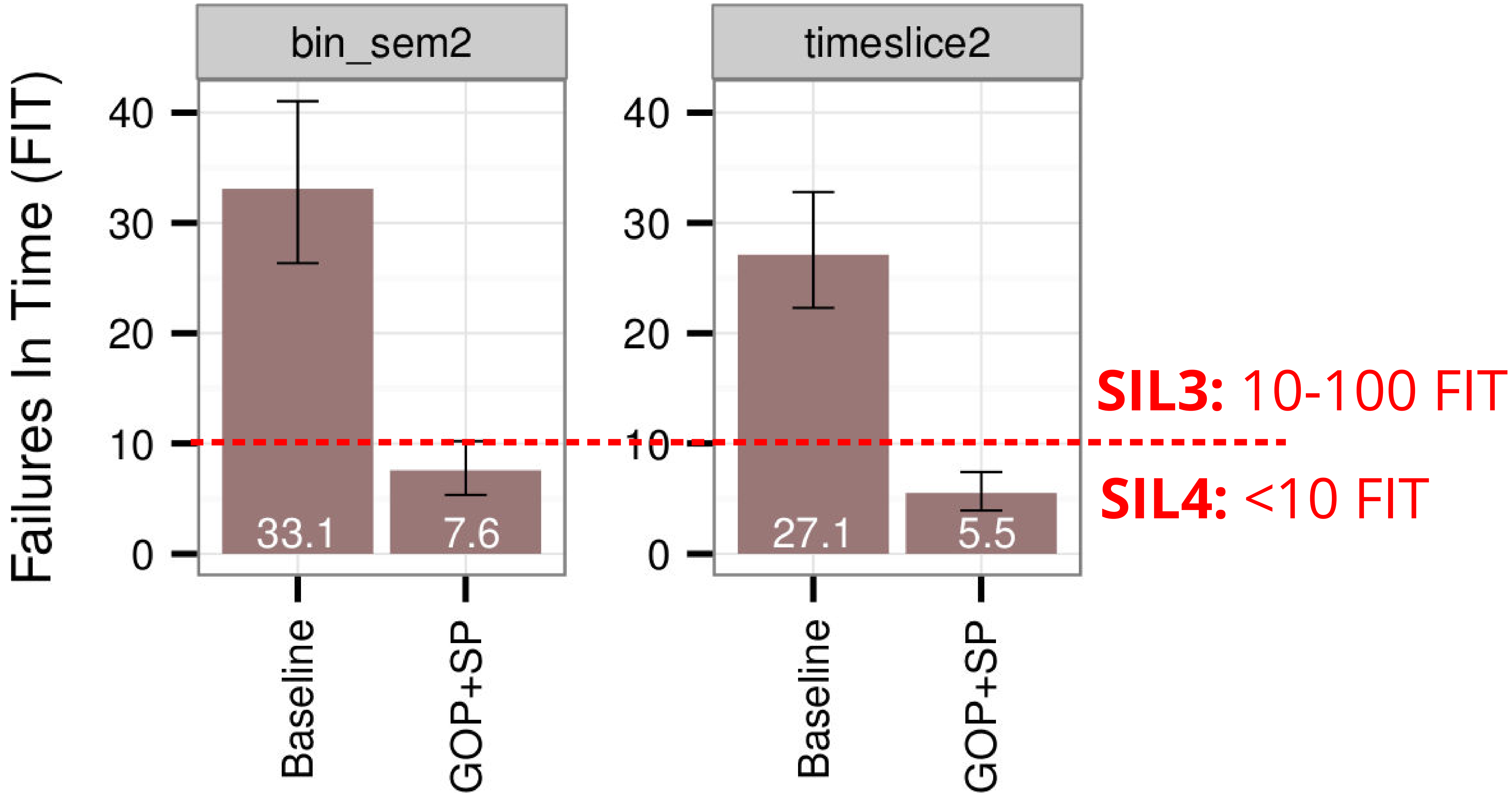
Reality Check: Real soft-errors?



- **Irradiation for ~1 day**, equivalent of 4.5 million years natural neutron radiation on earth surface [ARCS'17]

Neutron-Beam Experiment Results

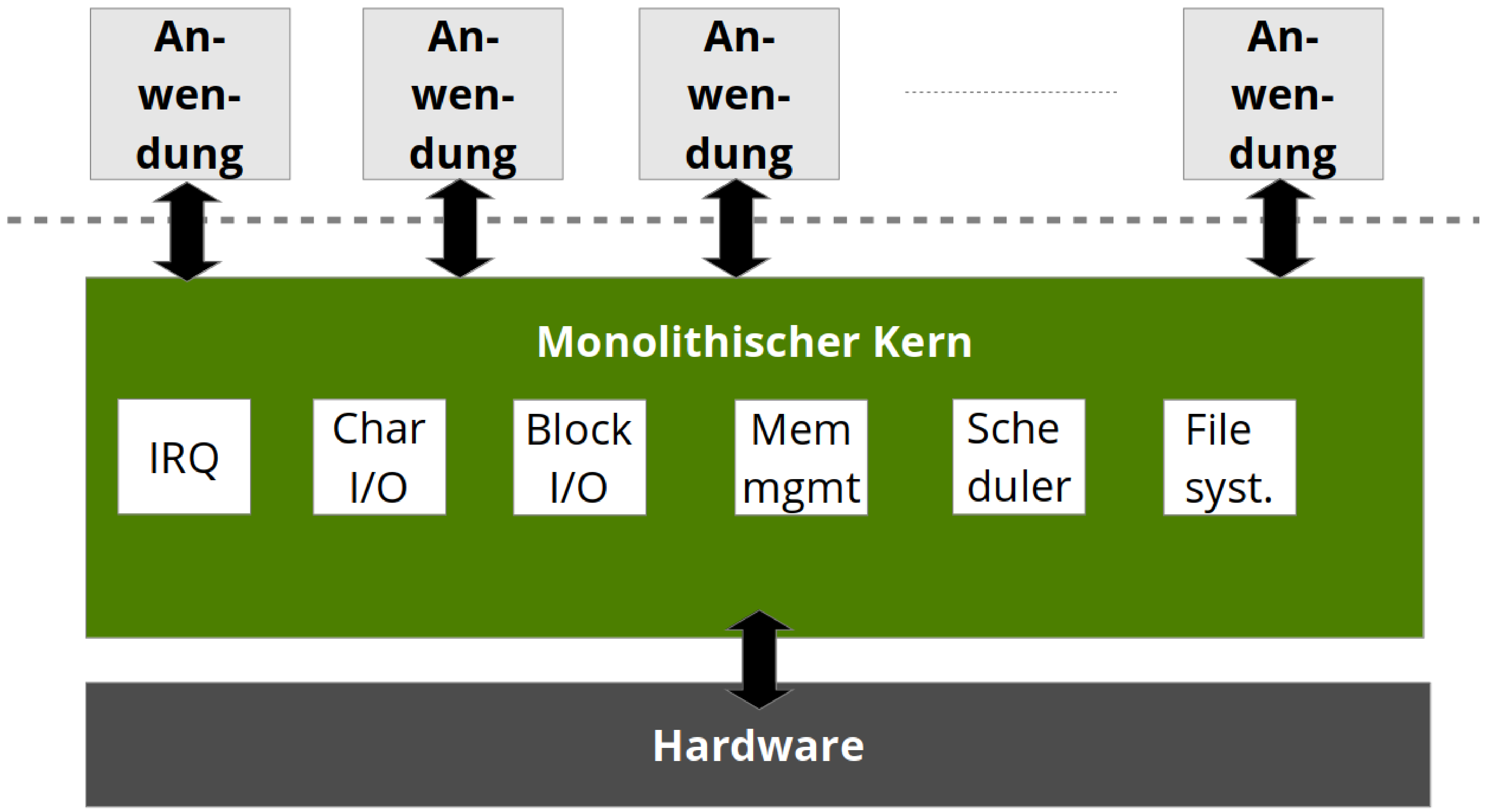
- FIT (Failures In Time):
  expected number of failures (SDC, reboot) per $10^9 h$



[ARCS'17]

*but: more requirements to achieve SIL3/4

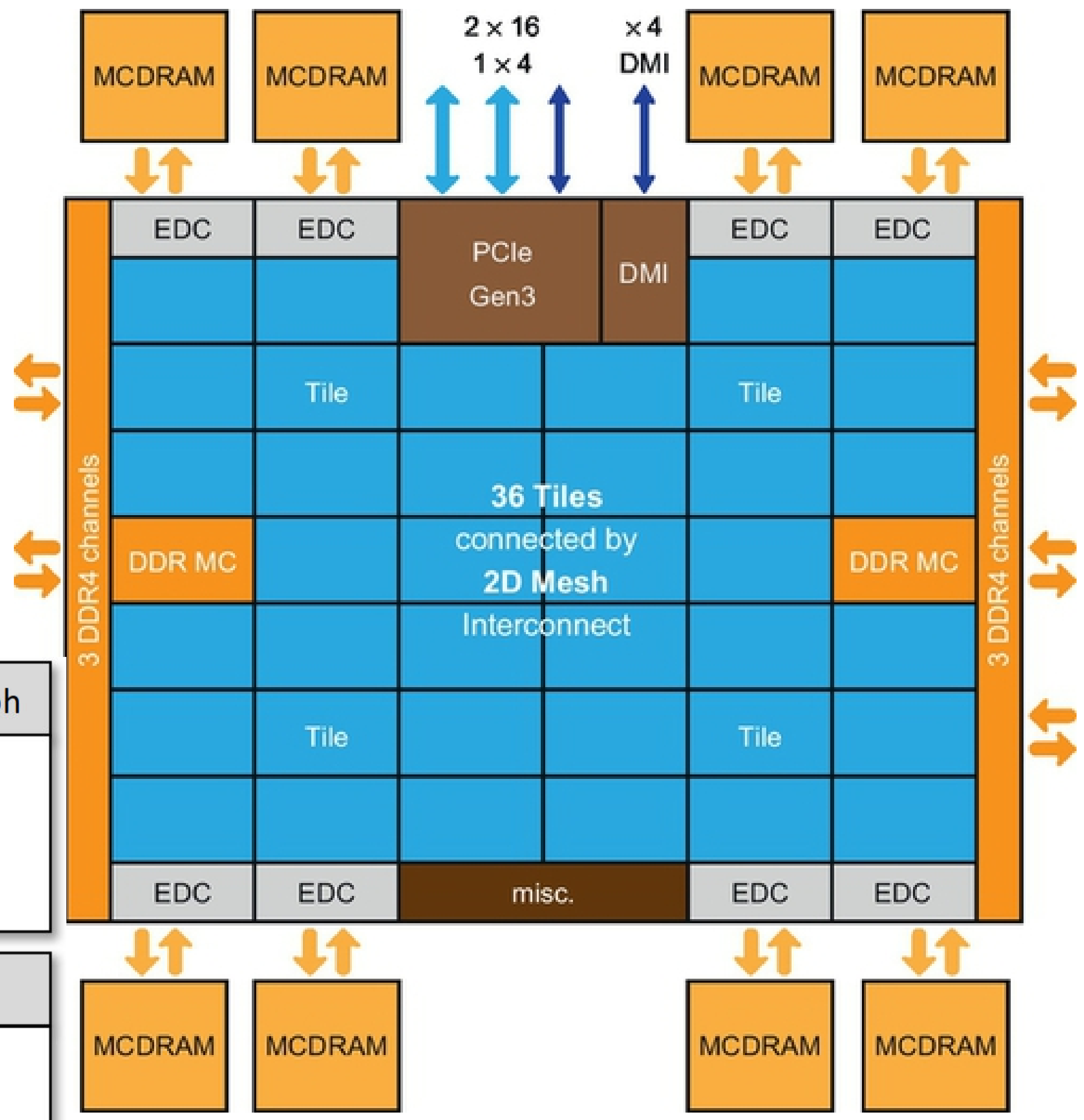
Komplexität: Monolith vs. Mikrokern

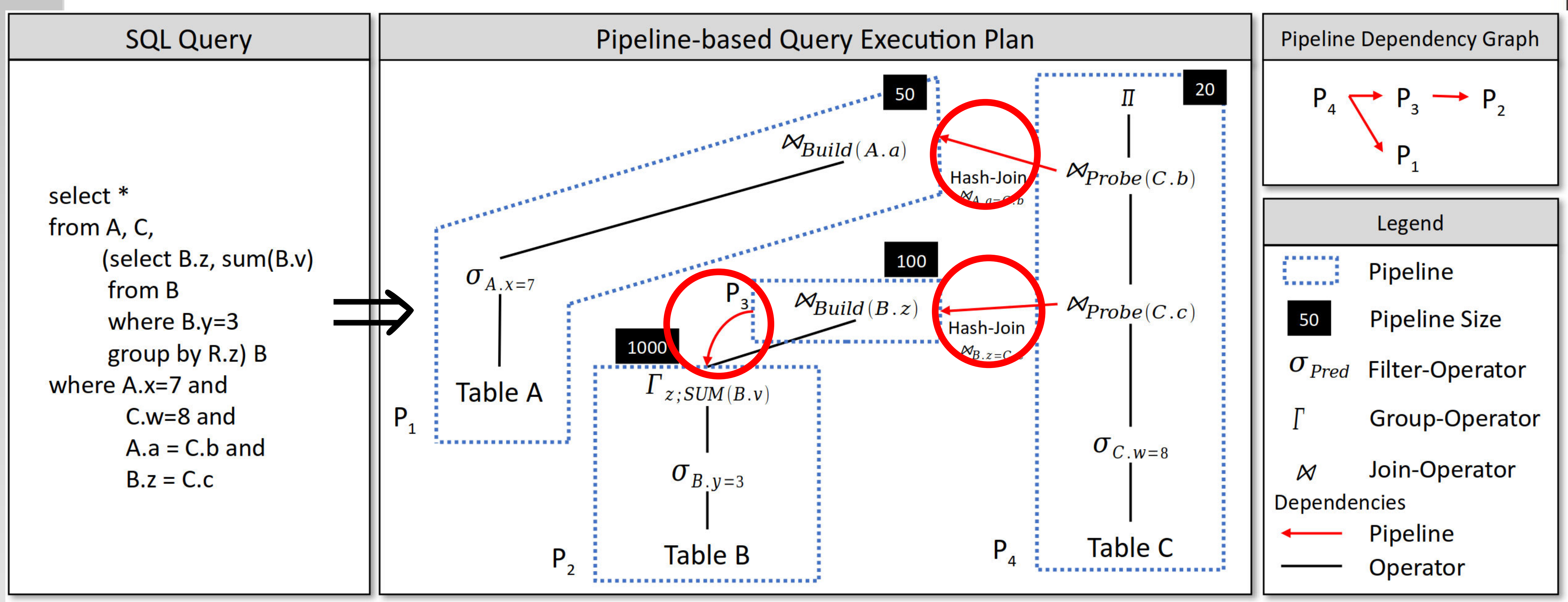


- feingranulares Sperren ist fehlerträchtig: **LockDoc**-Projekt
- *Security:* Übernahme einer Kernkomponente = Game Over
- aber: Performanz, viel *Legacy*-Code

Disruptive Speichertechnologien

- Umgang mit heterogenen Speichern: **VAMPIR**-Projekt
  - Latenz, Durchsatz, Persistenz, Fehlertoleranz, *Wearout*, Energieverbrauch, PIM-Fähigkeiten, …
- Anwendungsfall Datenbanken
  - zeitliche Vorhersagen viel einfacher!



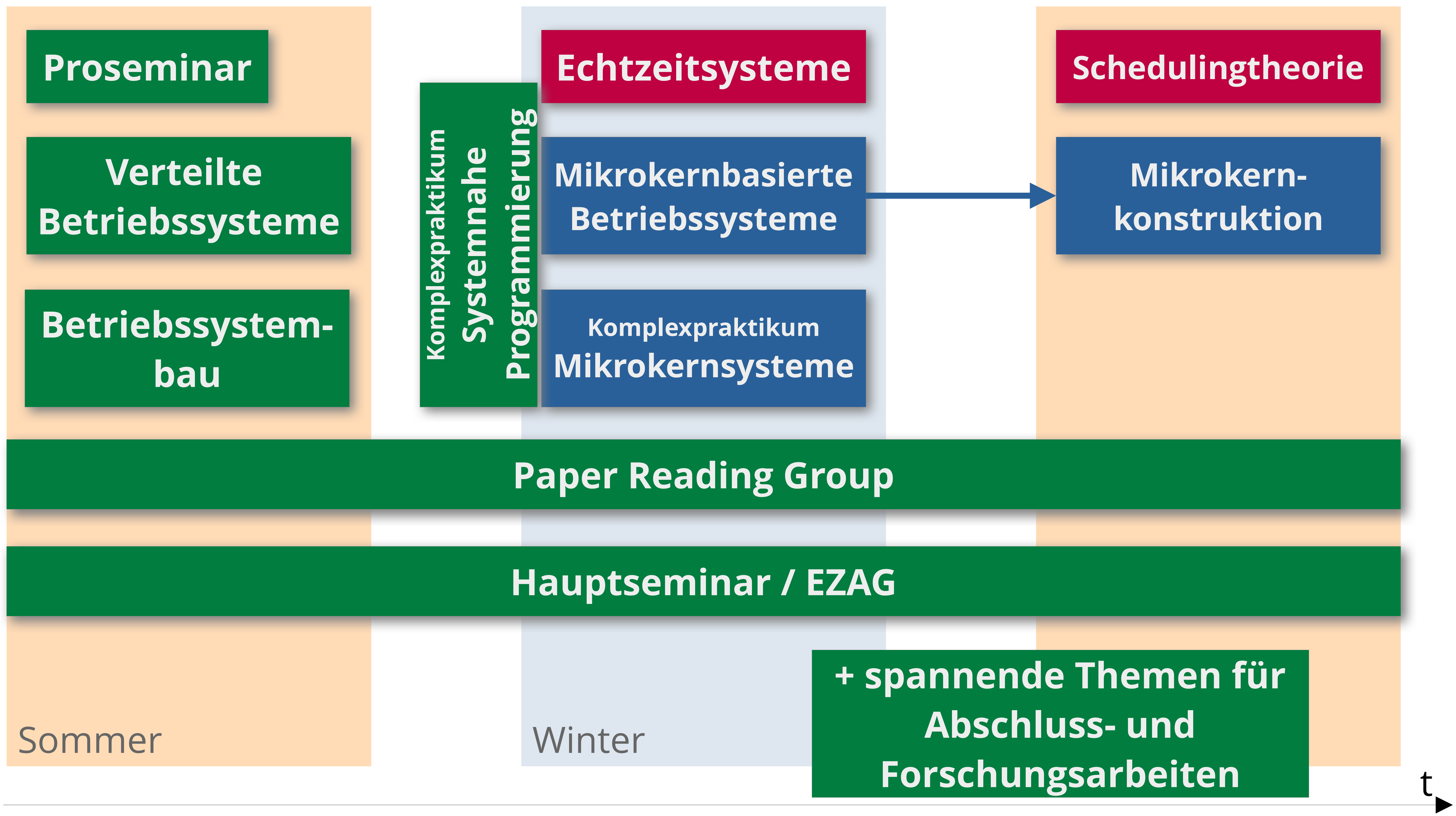


Zusammenfassung

- Arbeit an „echten“ Systemen
- enge Zusammenarbeit mit Mitarbeiter*innen der Professur
- vielseitige Firmen- und Forschungslandschaft

Systems Research: More Relevant than Ever.

Lehrveranstaltungen



Wir suchen: Studentische Hilfskräfte

- **Tutor*innen** für „Betriebssysteme und Sicherheit“ (WS 23/24) und „Betriebssystembau“ (SS 24)
  - Übungen: Besprechen / Vorrechnen von Aufgaben
  - Hilfestellung am Rechner (C/C++)
- Weiterentwicklung von **Lehrmaterialien**
  - z.B. Vorgaben und Aufgabenstellungen „Betriebssystembau“
- Mitarbeit in **Forschungsprojekten**
  - Programmierung, Recherche, Messungen, usw.
  - Gegen € als HiWi, oder im Rahmen der „Forschungsprojekt“-Module oder einer Abschlussarbeit