



seL4 Foundation Update

Gernot Heiser

seL4 Foundation and
Trustworthy Systems @ UNSW Sydney

gernot@sel4.systems



seL4 Summits



2023 seL4 Summit – Minneapolis



seL4 SUMMIT

MINNEAPOLIS 2023

Great success again

- ❖ 65 attendees in person
- ❖ Talks recorded and videos online
- ❖ 2 keynotes, 17 talks, 10 short talks, 3 BoF, 1 SIG update, 1 panel, 1 training
- ❖ Great level of sponsorship





Forthcoming 2024 seL4 Summit – Sydney



seL4 SUMMIT

SYDNEY 2024

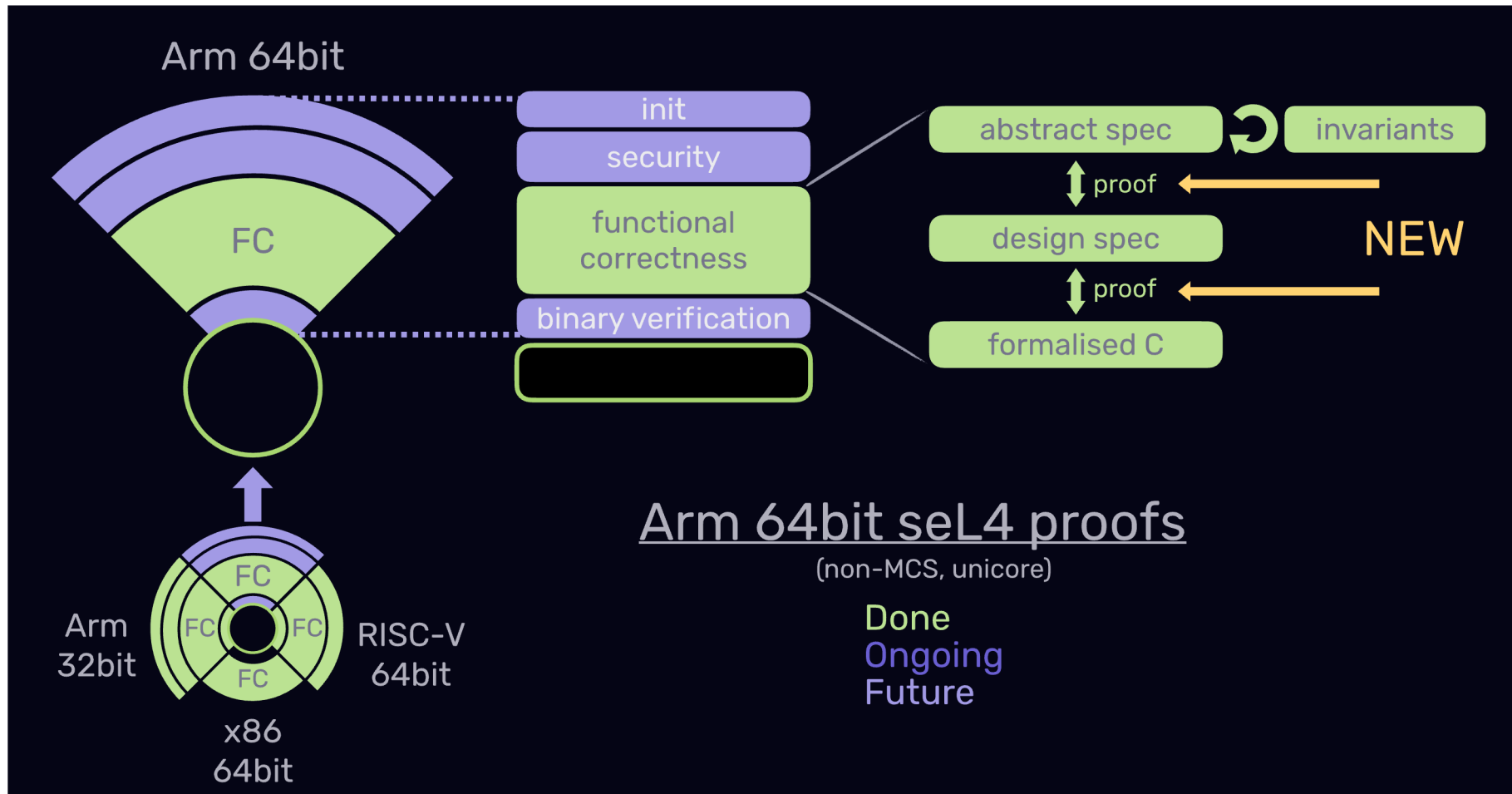
- ❖ Presentation submissions closed
- ❖ Provisional Program early June
- ❖ Summit 15–17 October



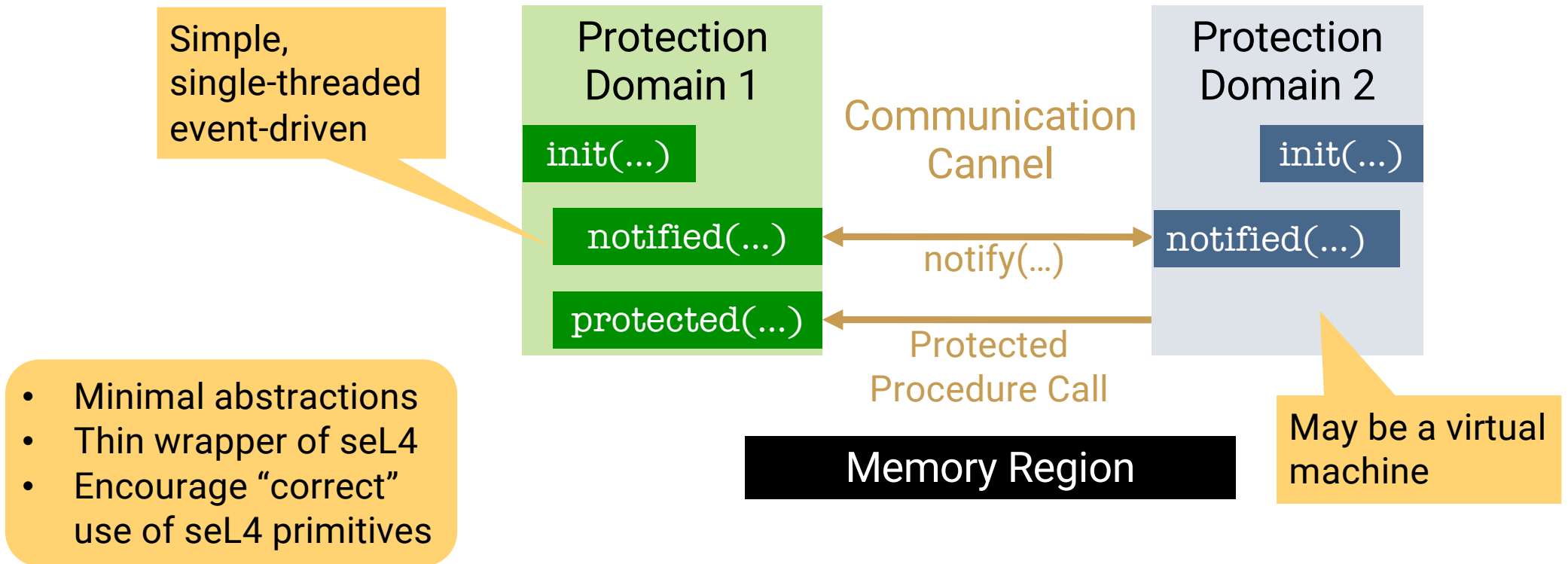
A large green key graphic with a white circular hole in the head, positioned horizontally across the middle of the slide. The text "Technical Update" is written in black on the stem of the key.

Technical Update

Verification of AArch64 Kernel Completed!



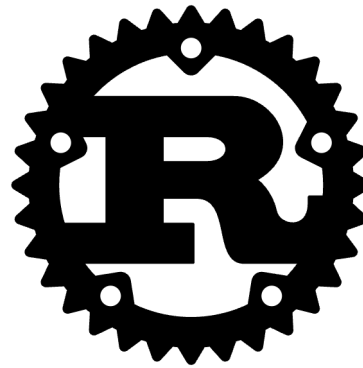
seL4 Microkit (fka Core Platform) Adopted



Official Rust Support



- Support for writing seL4 apps in Rust
- Actively developed and maintained



Other



Verification:

- ❖ More verified platforms: imx8mm, exynos5
- ❖ MCS kernel progressing
- ❖ Multikernel progressing

Systems:

- ❖ Raspberry Pi 4 supported & in CI
- ❖ VMM improvements & maintenance by DornerWorks



School of Computer Science & Engineering
UNSW SYDNEY | Global University
Trustworthy Systems Group



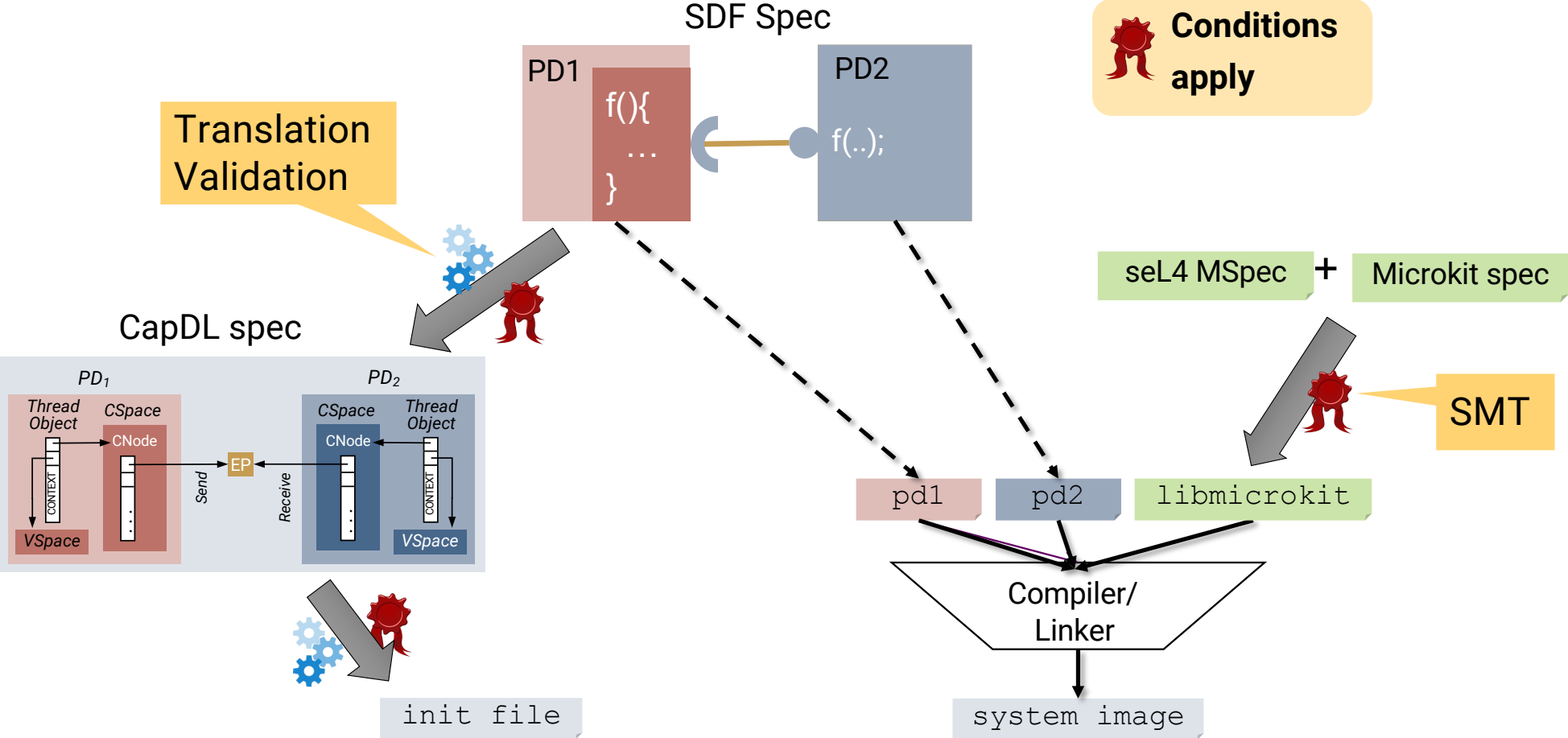
UNSW R&D Update

Gernot Heiser

gernot@unsw.edu.au

9 May 2024

Microkit Verification



Lions OS: Secure, Fast, Adaptable



Aim 1: *Practical, easy-to-use, open-source OS for wide range of embedded/IoT/cyberphysical use cases*

Aim 2: *Best-performing microkernel-based OS ever*

Aim 3: *Most secure OS ever*



Really – an OS Built from Scratch?

Yes – if we strictly observe the KISS principle: Keep it simple, stupid!

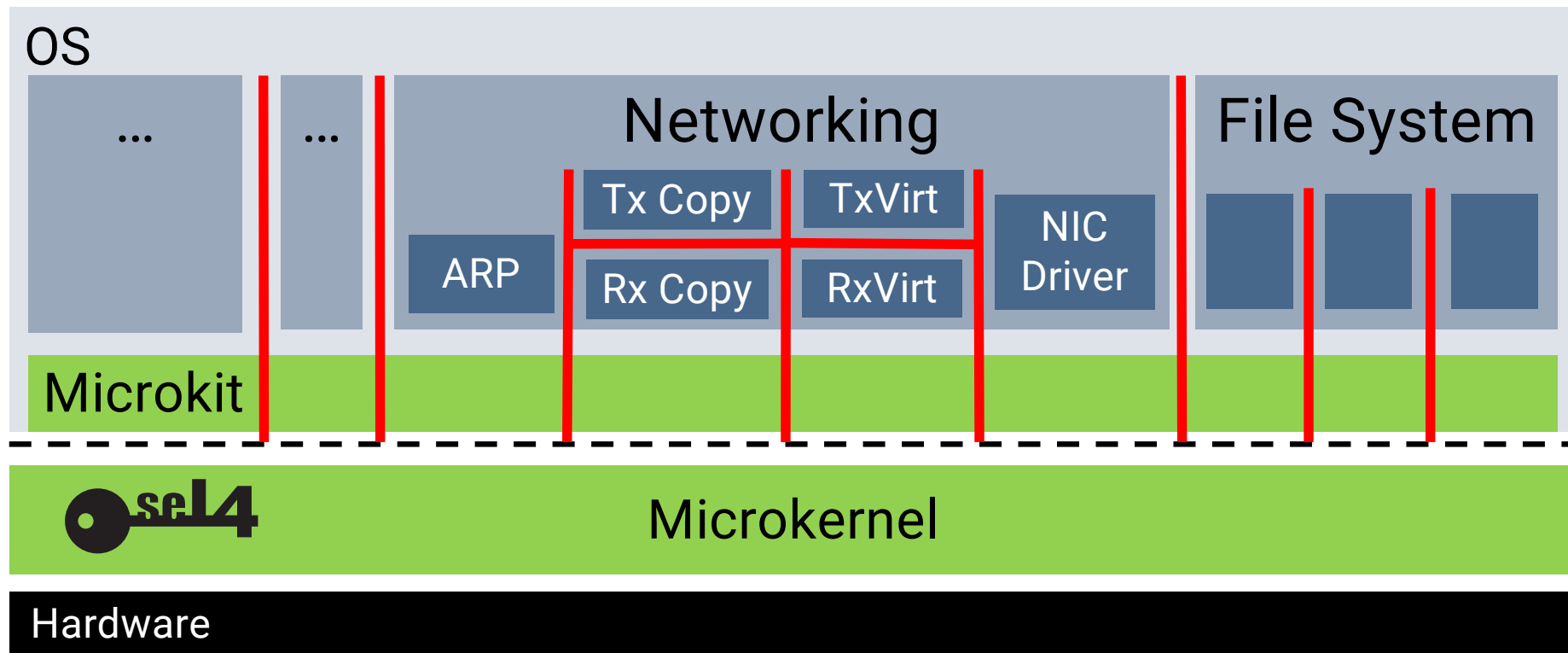
- Fine-grained modularity, strong *separation of concerns* Software engineering 101
- Least privilege Reason about security
- *Radical Simplicity*[™]: provide *only* the features needed KISS extreme
- *Use-case specific policy* (rather than universal policy)

Also limiting scope allows use of static architecture

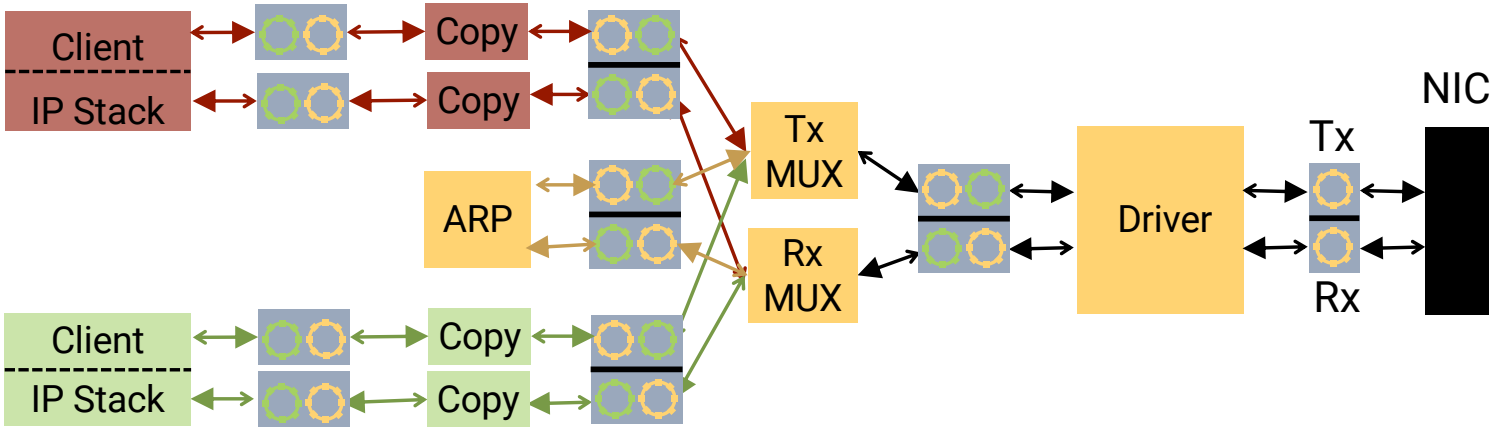
“Universal” policies are complex, always have pathological cases

Use-case diversity by swapping (policy) modules!

Lions OS: Modular System on Mikrokit

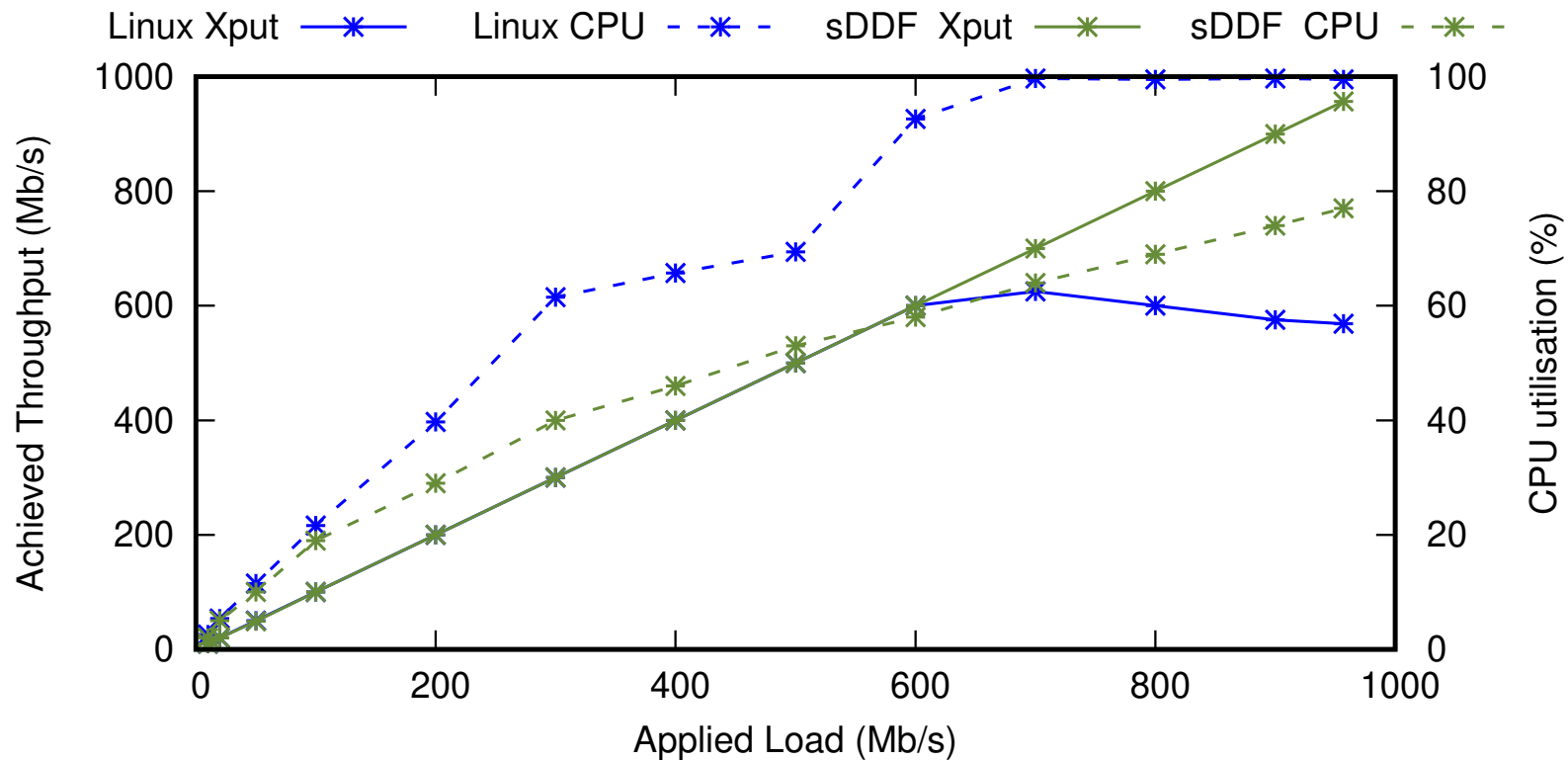


Example: Networking



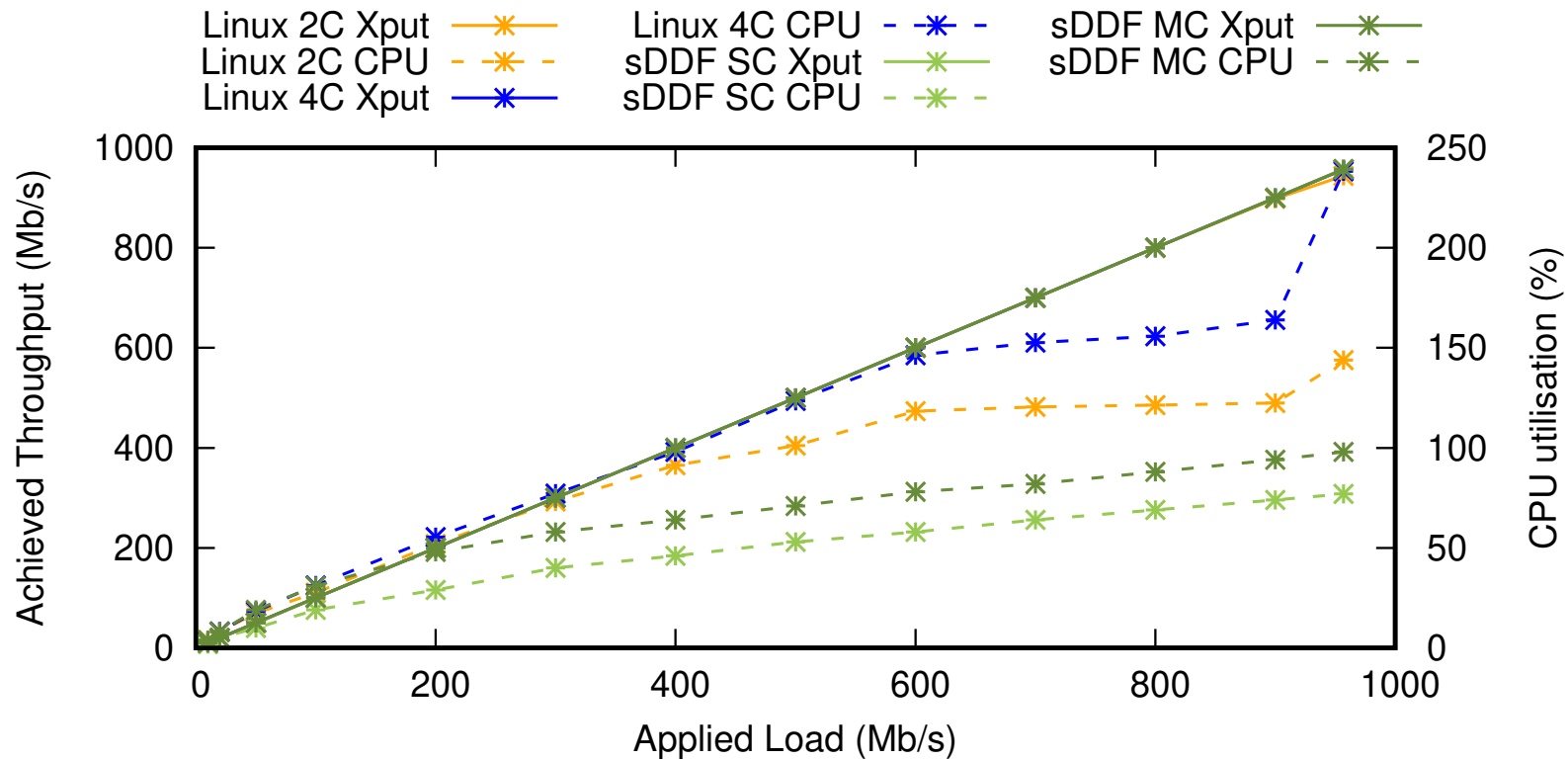
- Simple, event-based, single-threaded drivers
- Asynchronous, zero-copy transport layer, using bounded, non-blocking, single-producer, single-consumer queues

Performance: i.MX8M, 1Gb/s Ethernet



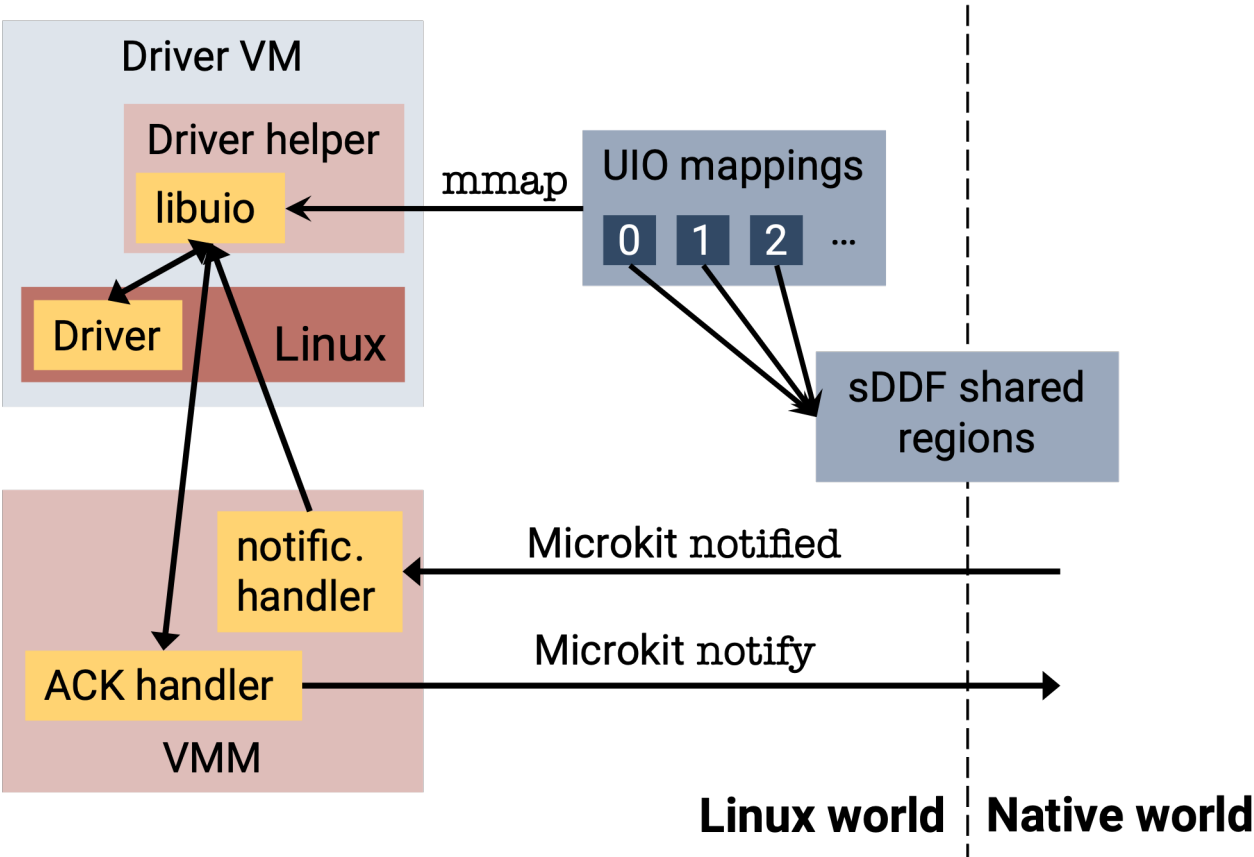
Single-core configuration

Performance: i.MX8M, 1Gb/s Ethernet



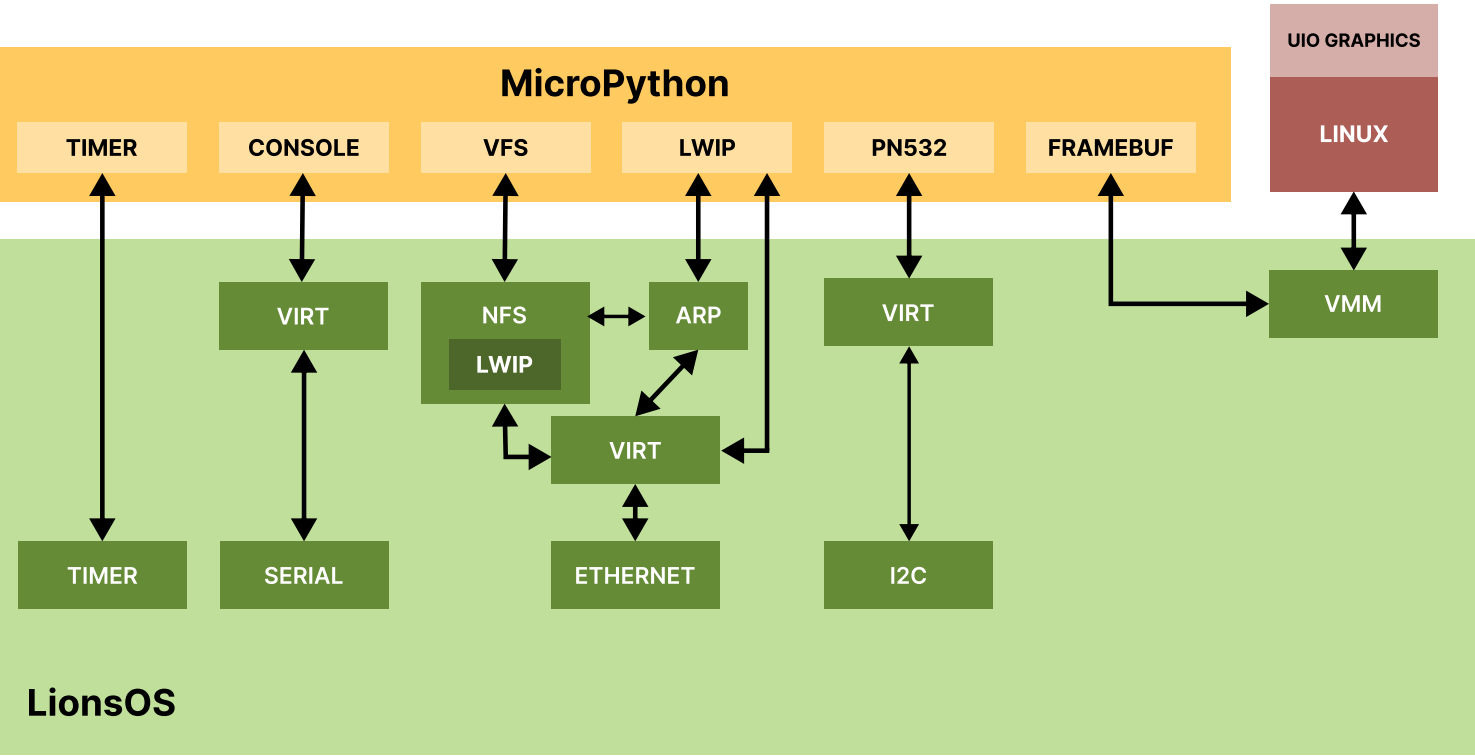
Multicore configuration

Re-using Unmodified Linux Drivers

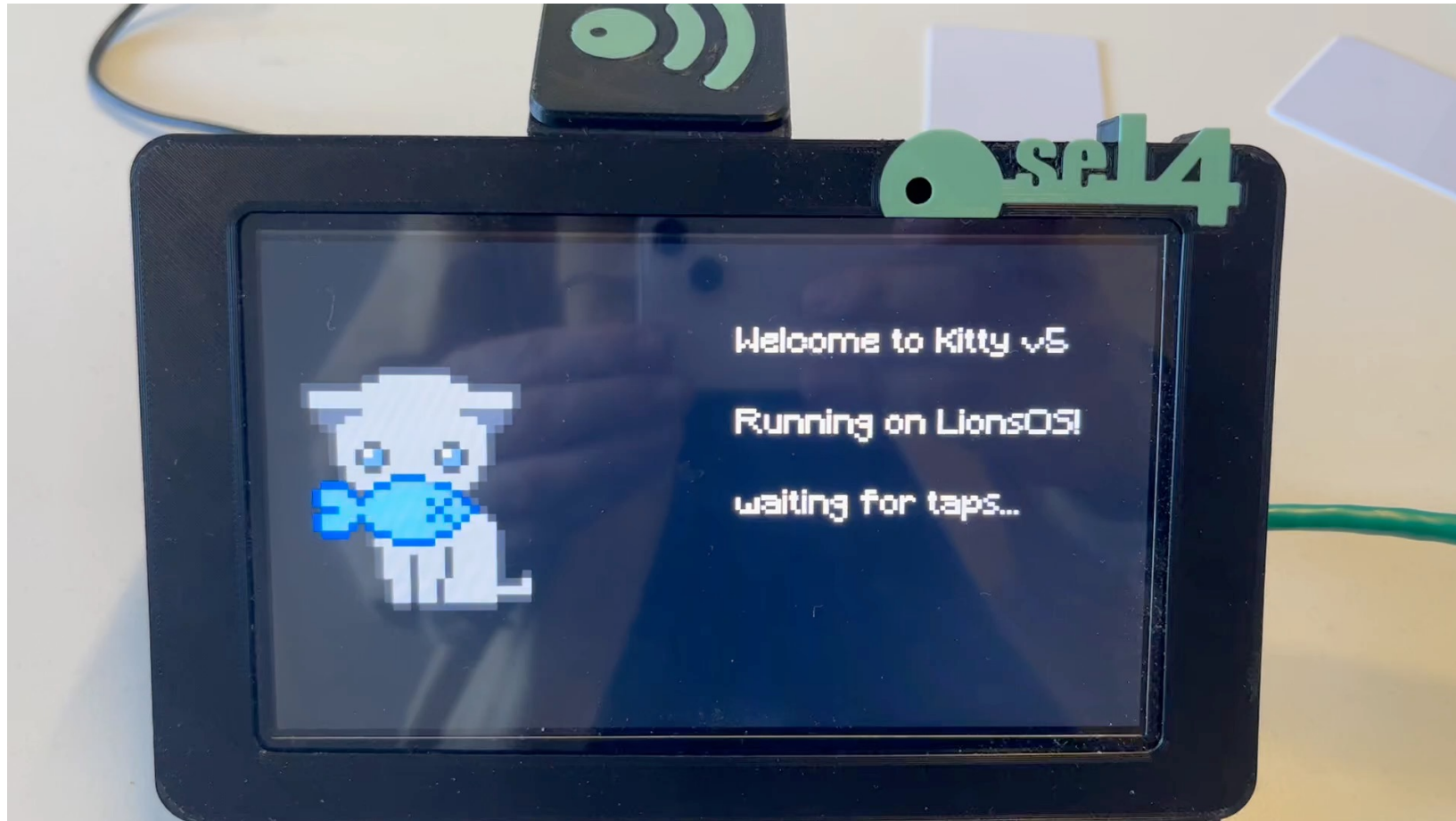




Lions OS IoT Reference System



The System In Action



Lions OS Release 0.1



- Native serial, Ethernet, I2C drivers
- Native NFS client, Python interpreter (MicroPython)
- Native components in Rust supported on seL4, Microkit in progress
- Native web server (in Python)
- Driver VMs: graphics, touch screen, audio

Overview: <https://trustworthy.systems/projects/LionsOS/>
Docs: <https://lionsos.org/>
Source: <https://github.com/au-ts/lionsos/>
License: 2-clause BSD



Security is no excuse for bad performance!