

It's time for Operating Systems to rediscover Hardware

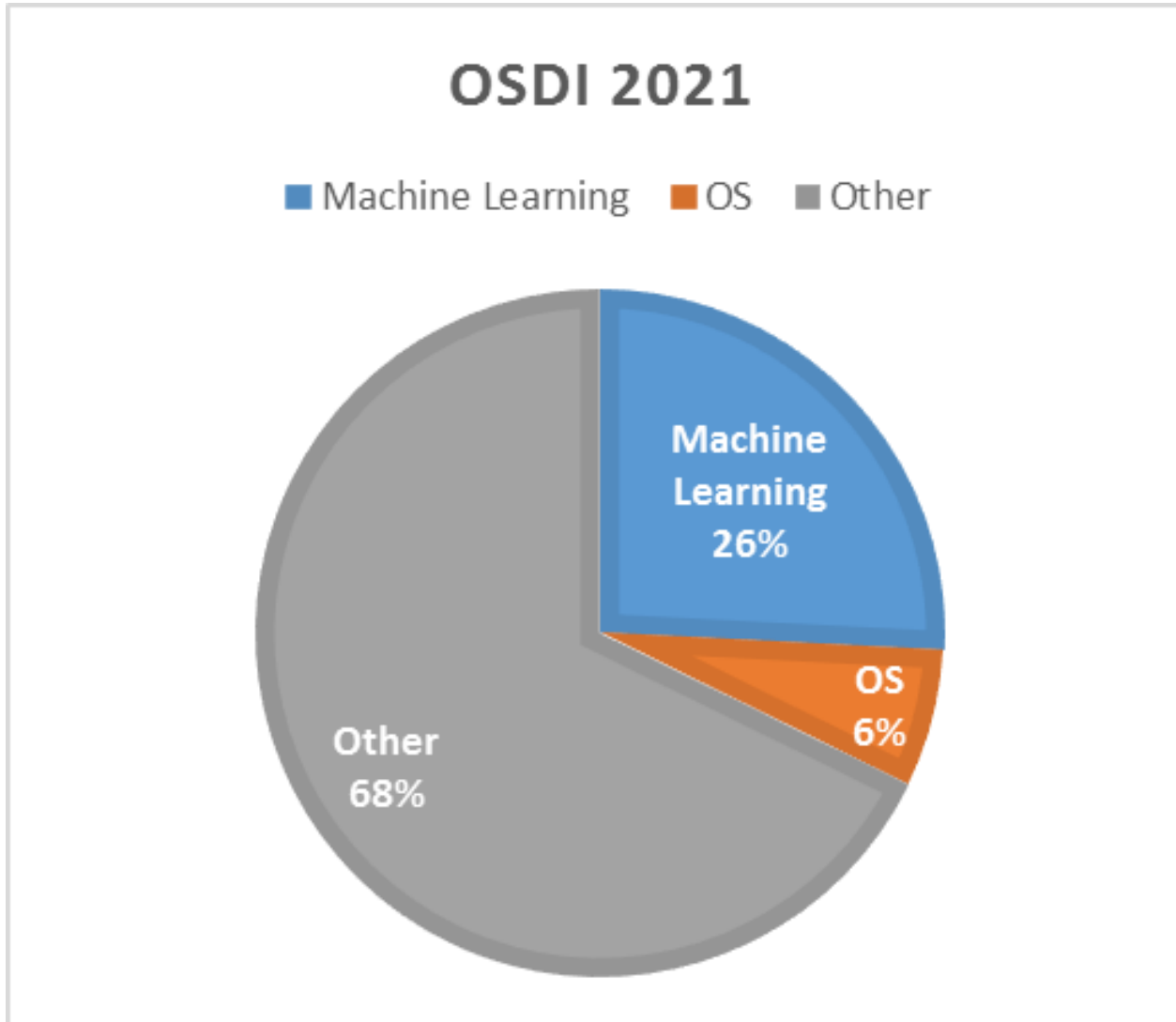
Timothy Roscoe (Mothy)
Systems Group, Dept. of Computer Science, ETH
Zurich

<https://www.youtube.com/watch?v=36myc8wQhLo>

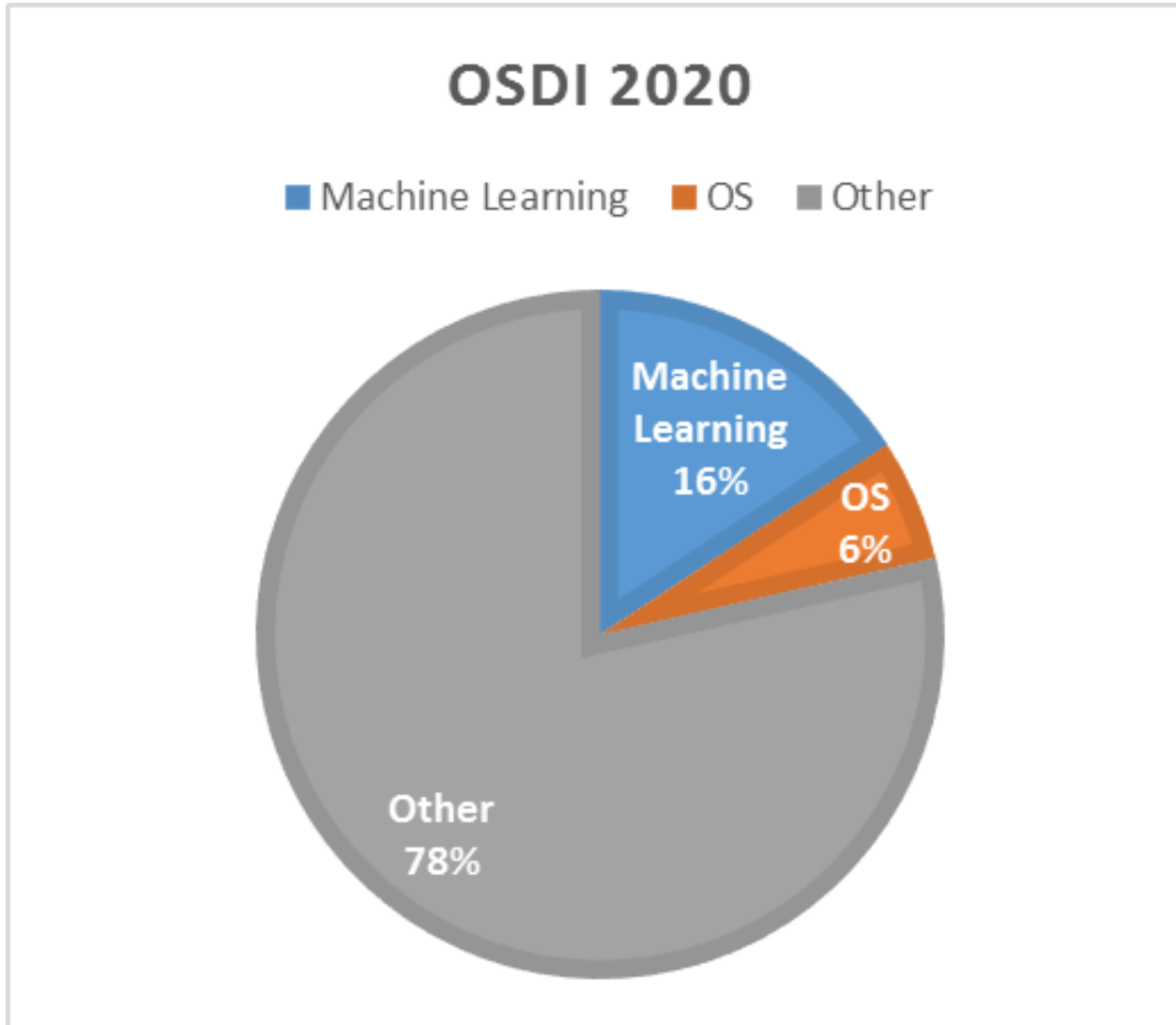
What should a keynote look like?

- Feel smug about my past research
- Sell my latest startup idea
- Vent my anger and frustration
- I'm going for option 3.

Operating Systems: a niche topic?



Operating Systems: a niche topic?



Is Linux just good enough?

- OS papers in OSDI that are *not* tweaks on Linux:
 - 2021: 2 (that's two) of 31
 - 2020: 1 (that's one) of 70

... in OS *Design* and Implementation?

Is the field dead?

Or at least twitching on the slab?

**But first: what do I
mean by an OS?**

The OS runs on the hardware.

But what is it?

1. The set of

- Kernel
- Daemons
- Runtime libraries



2. *That body of software* that:

- Multiplexes machine's hardware resources
- Abstracts the hardware platform
- Protects software principals from each other
 - Using the hardware



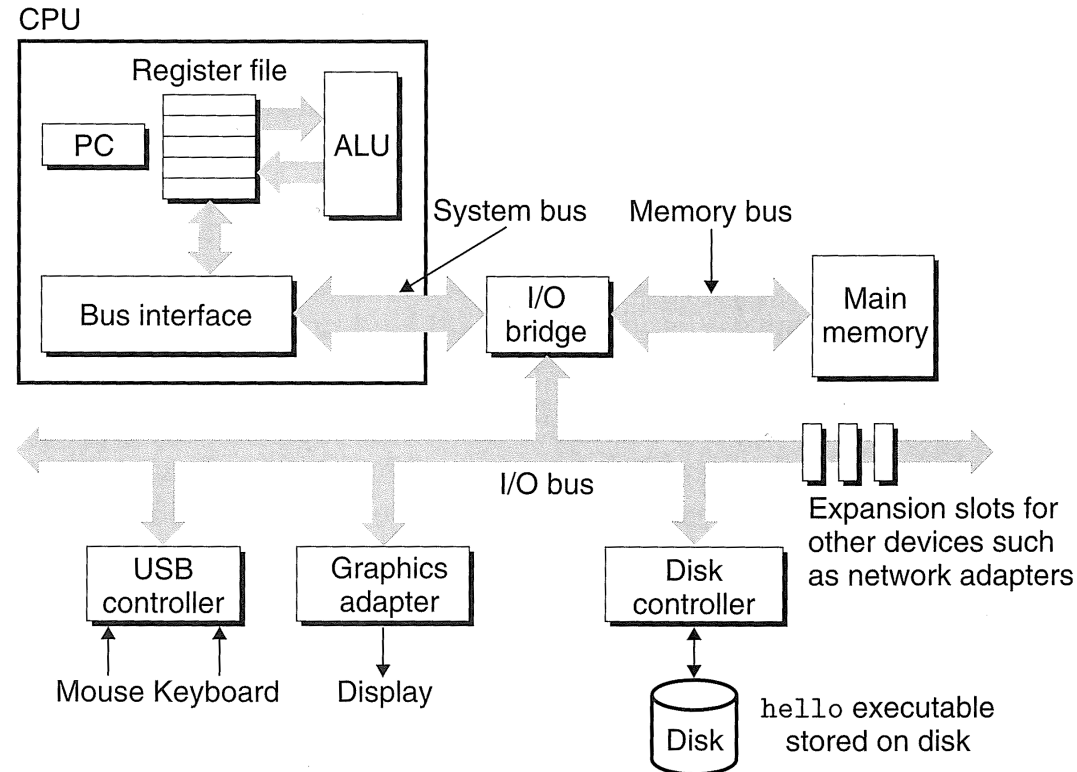
Let's look at the hardware

Why some people think Linux just works

Figure 1.4

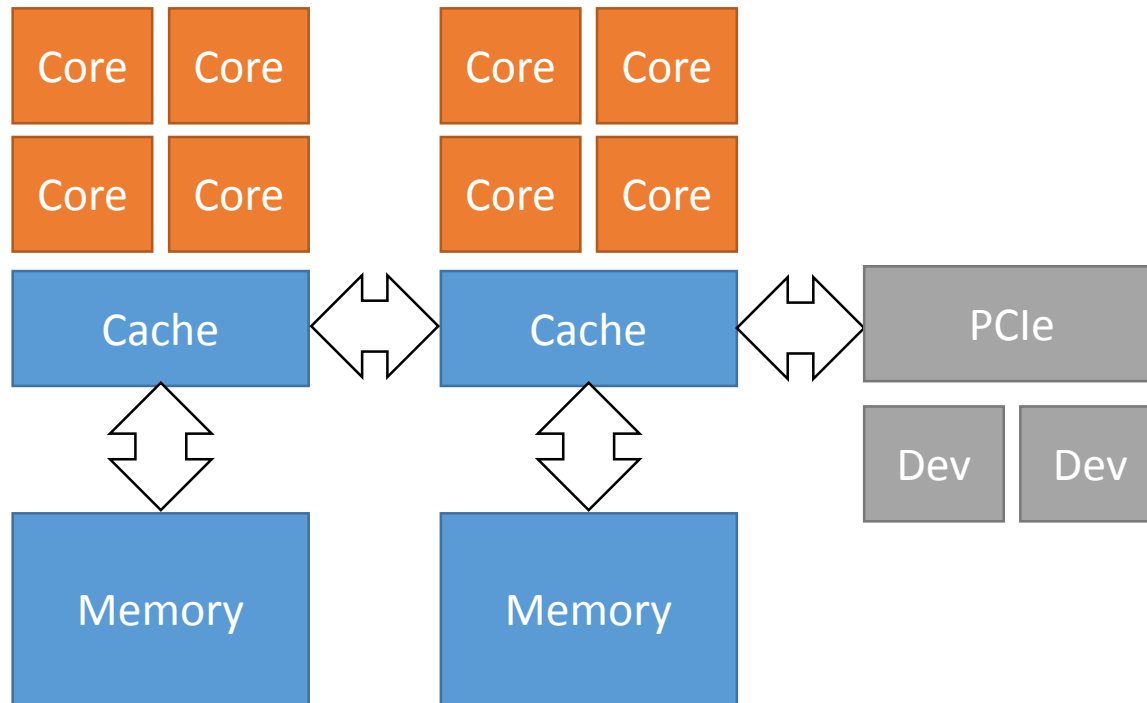
Hardware organization of a typical system.

CPU: Central Processing Unit, ALU: Arithmetic/Logic Unit, PC: Program counter, USB: Universal Serial Bus.

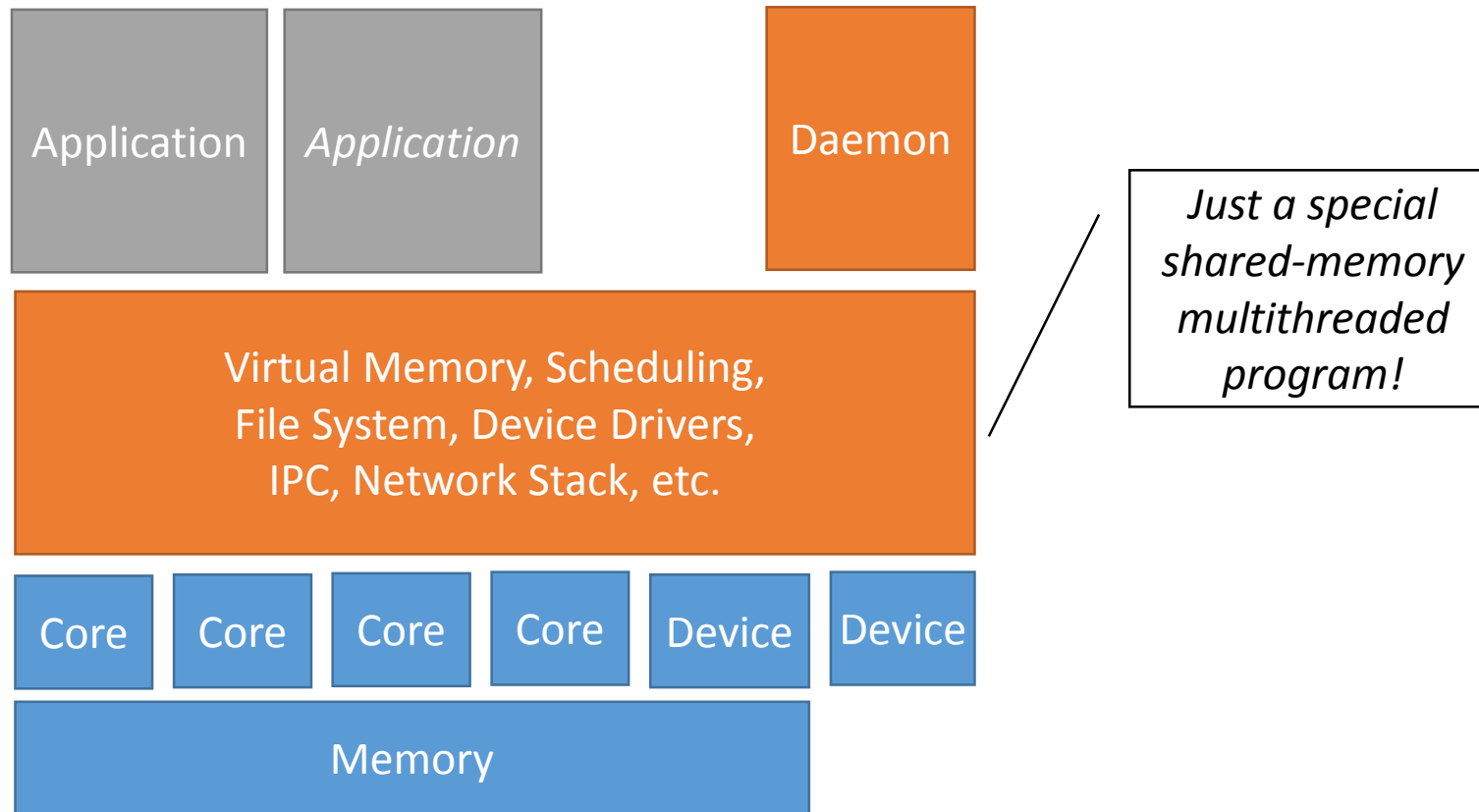


Computer Systems, A Programmer's Perspective, Bryant & O'Hallaron, 2011

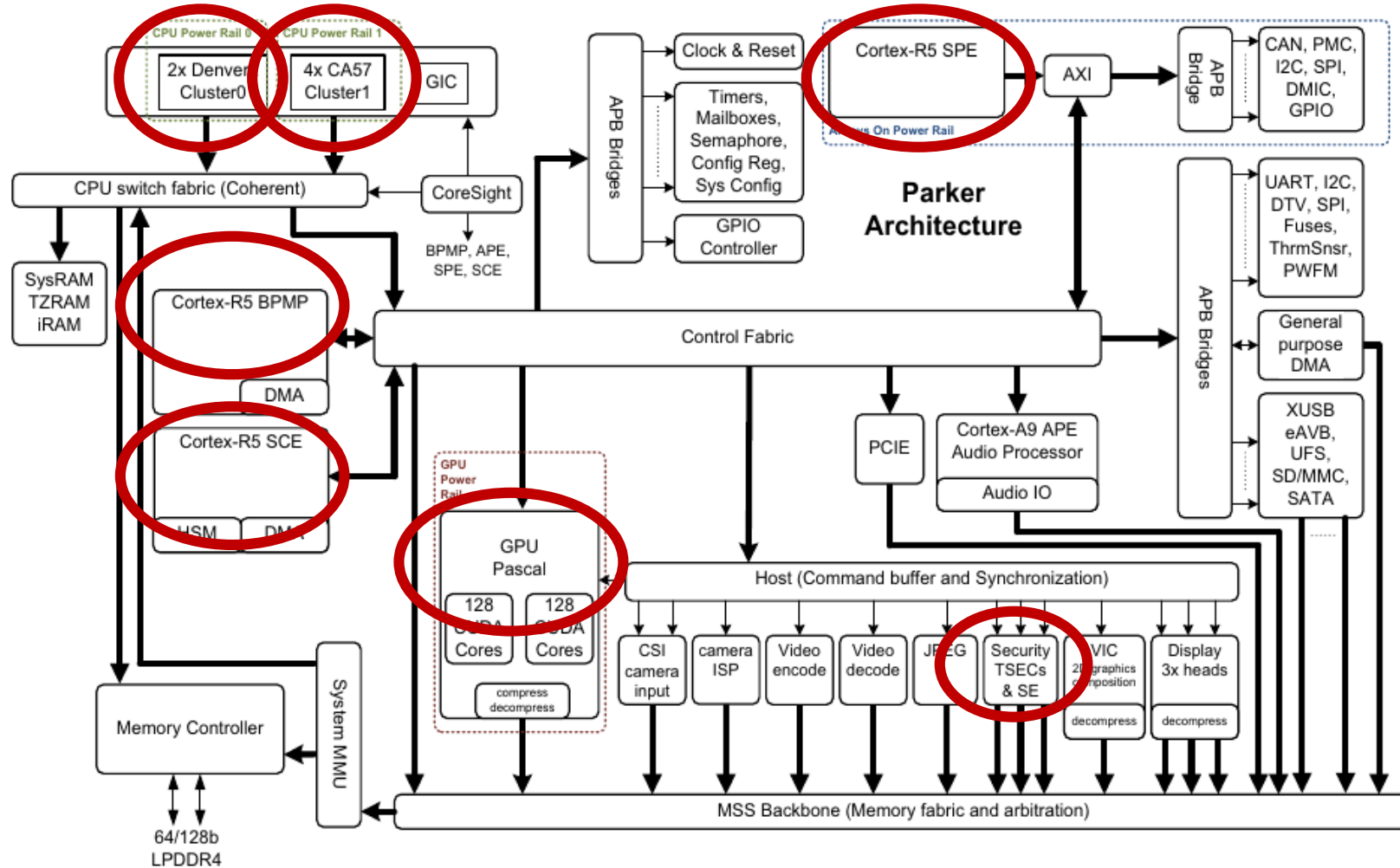
ccNUMA machines are lovely and simple!



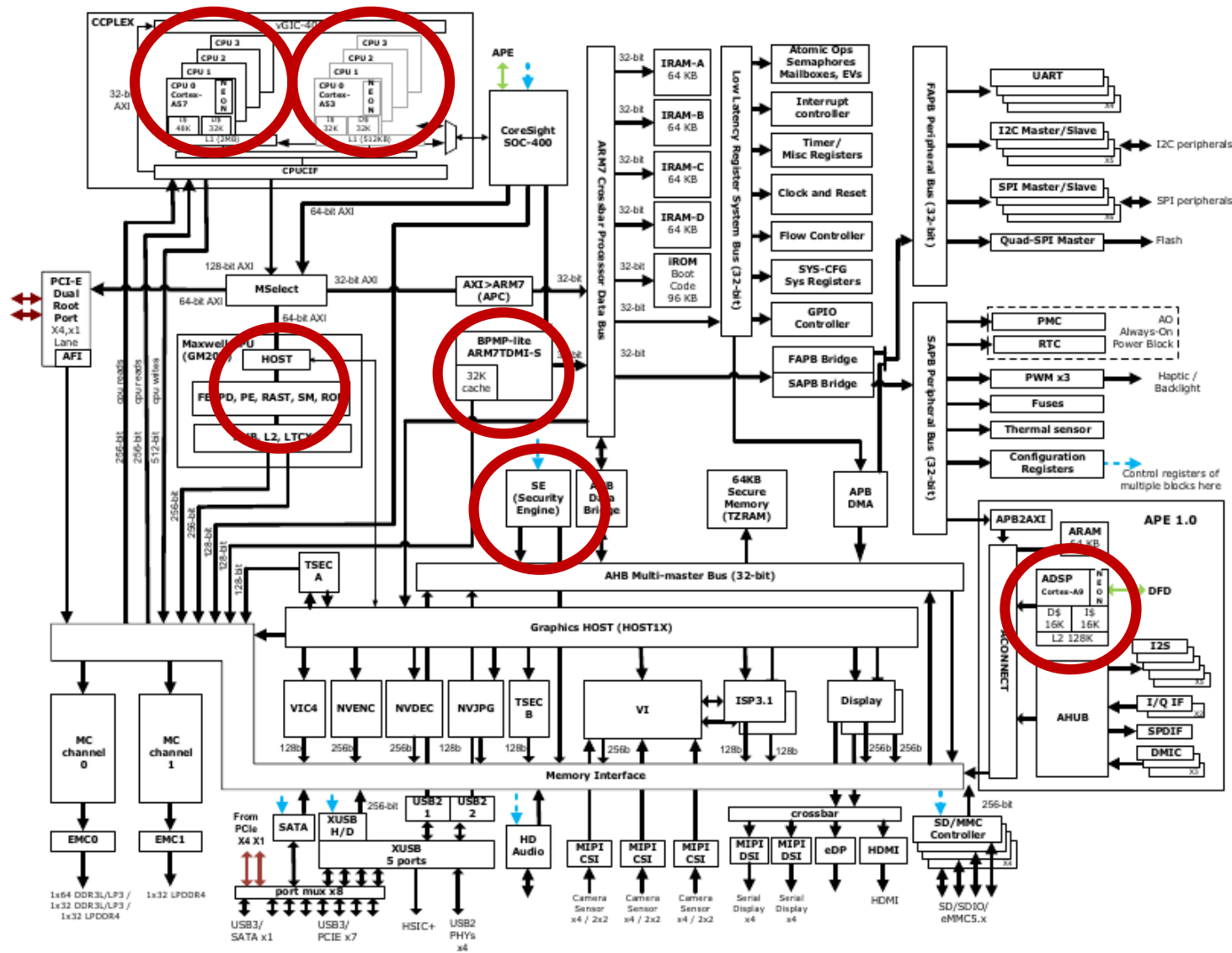
Result: the boring structure of an OS



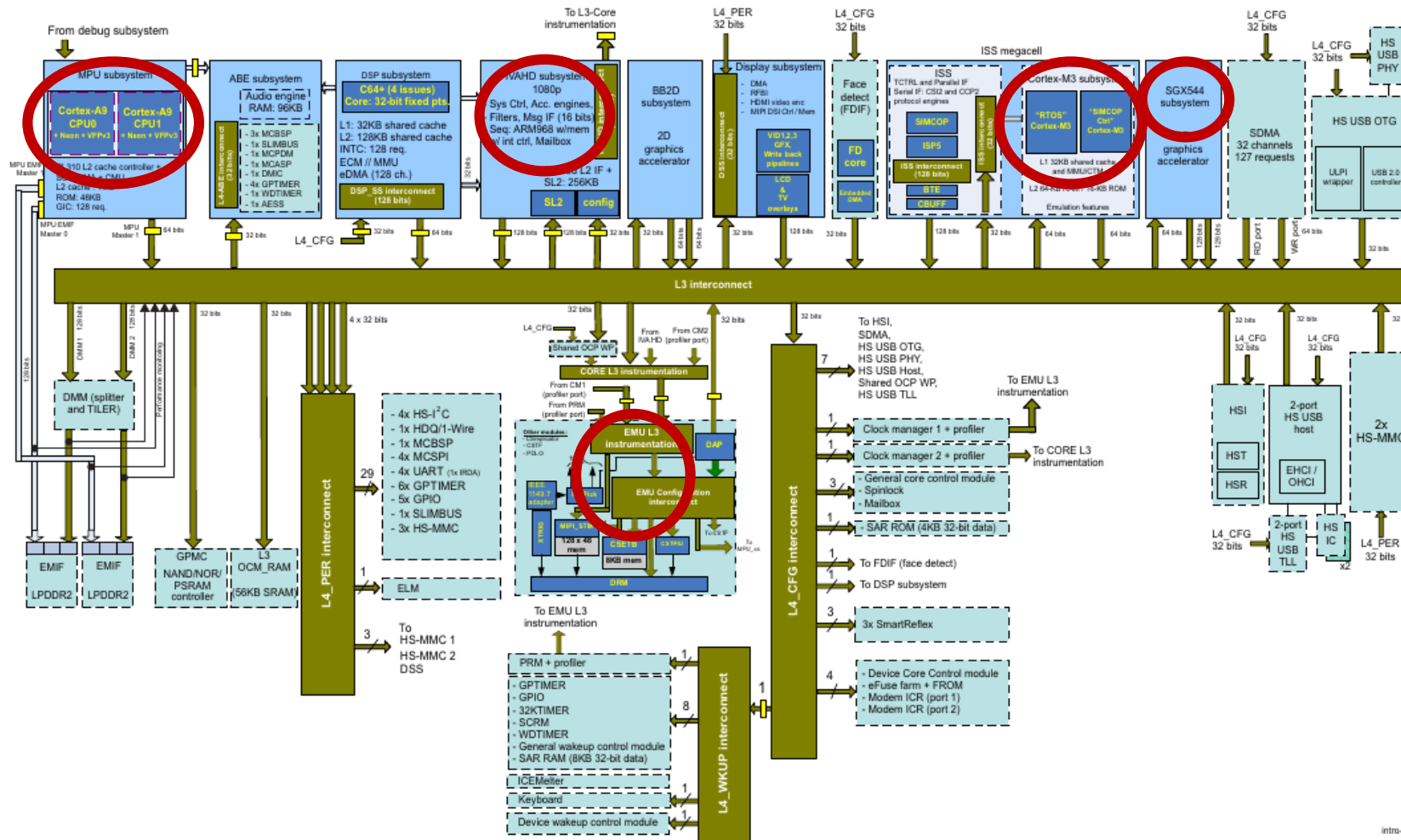
Reality?



Reality?

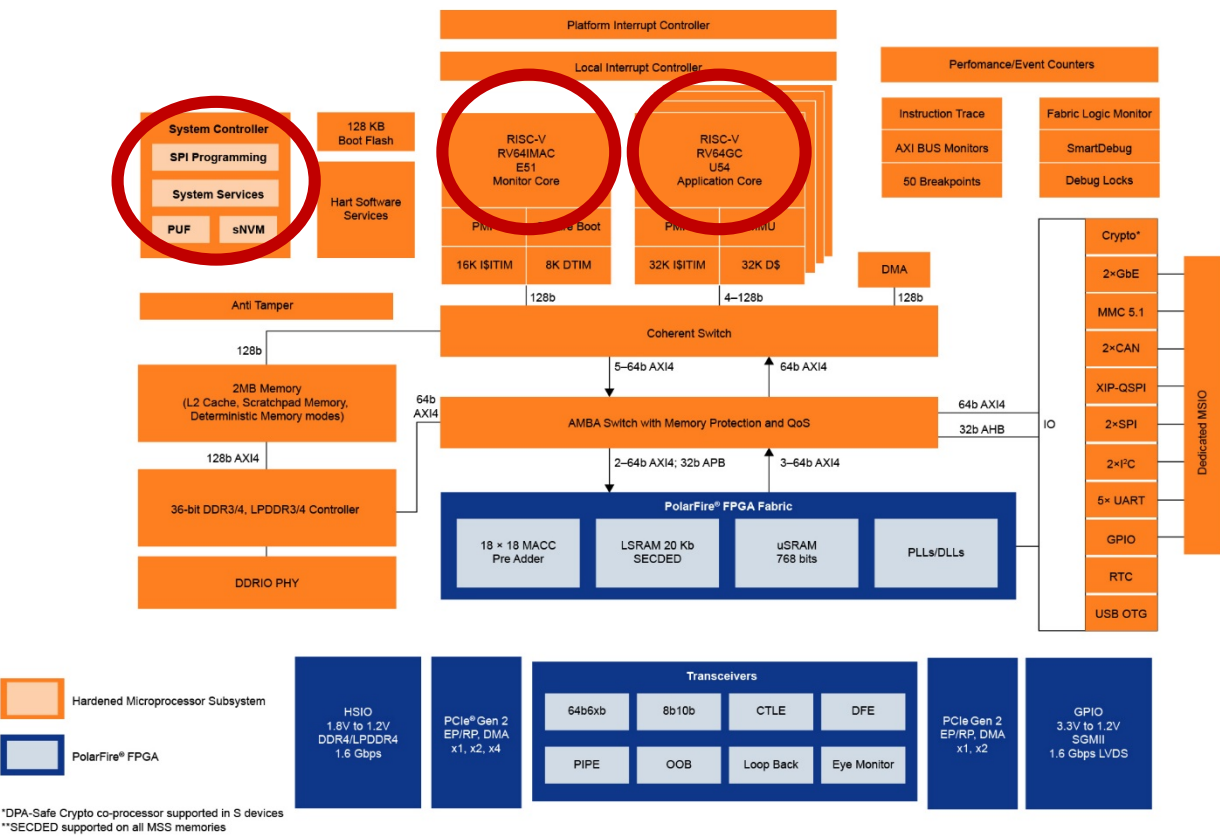


Reality?

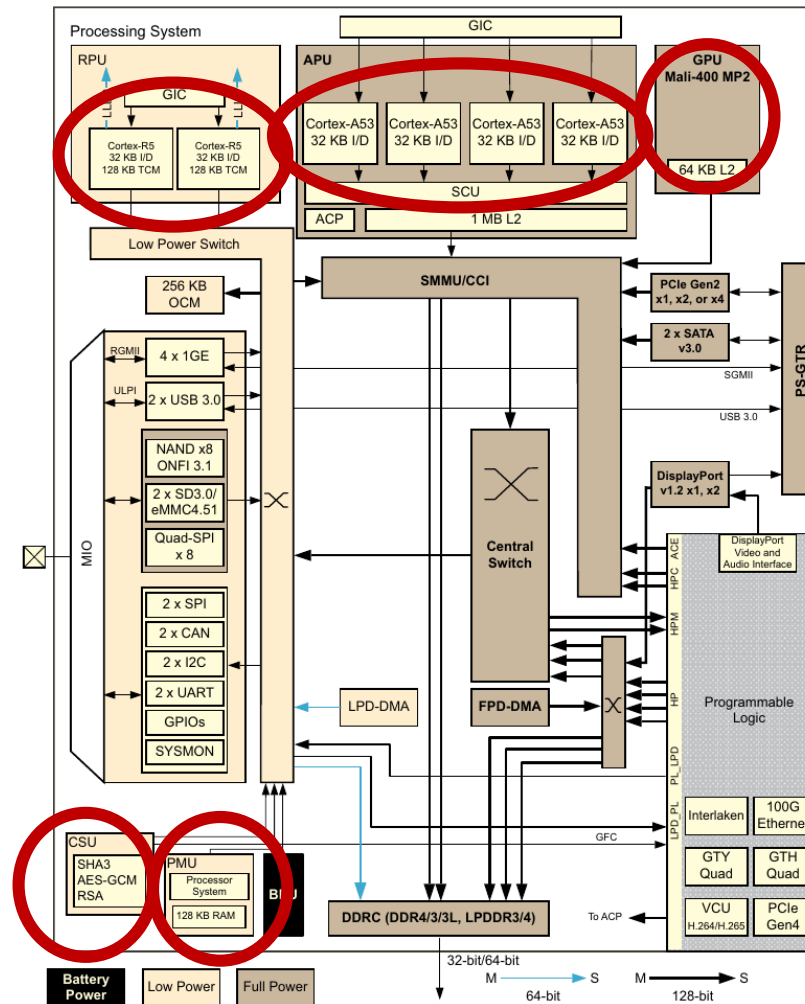


intro-00

Reality?



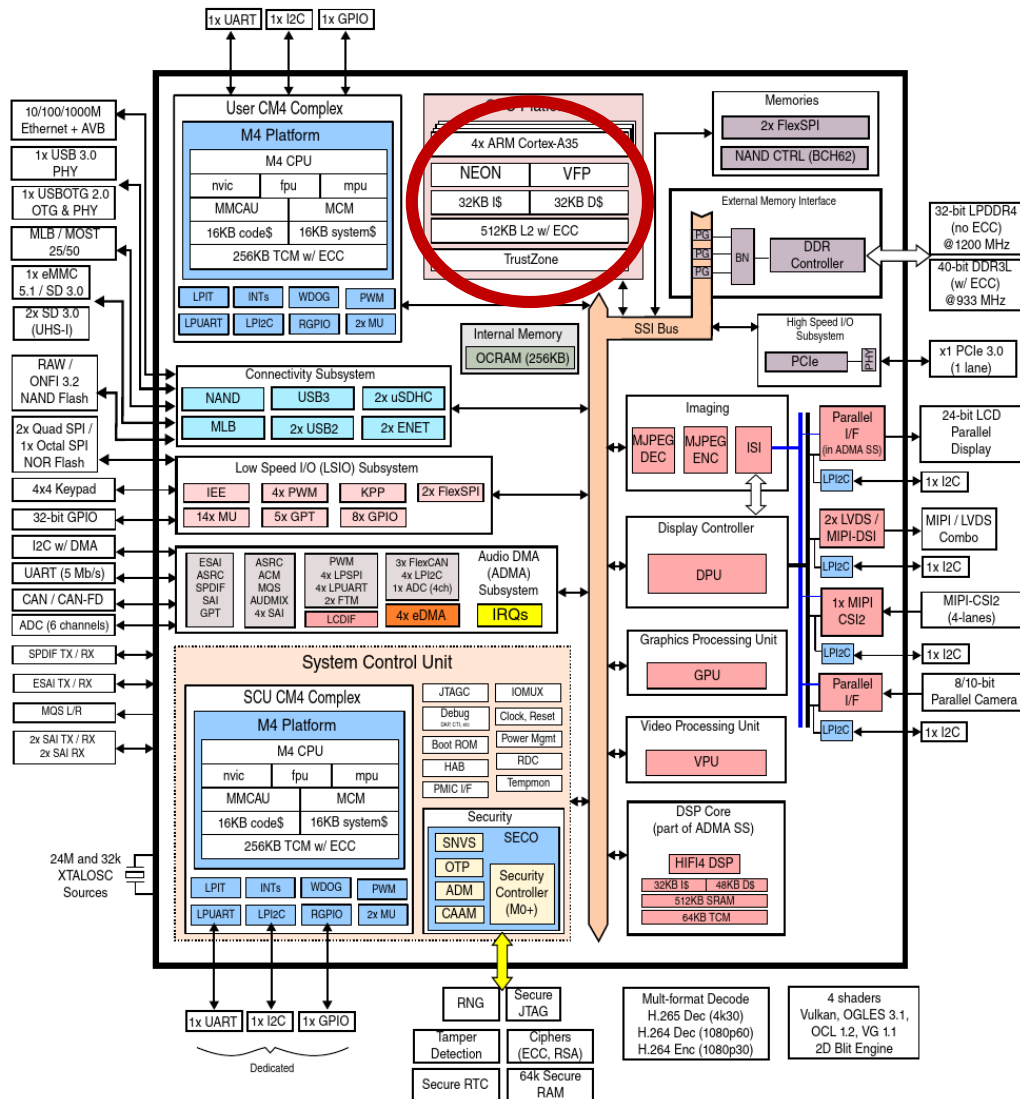
Reality?



ETH zürich

ETH zürich

18



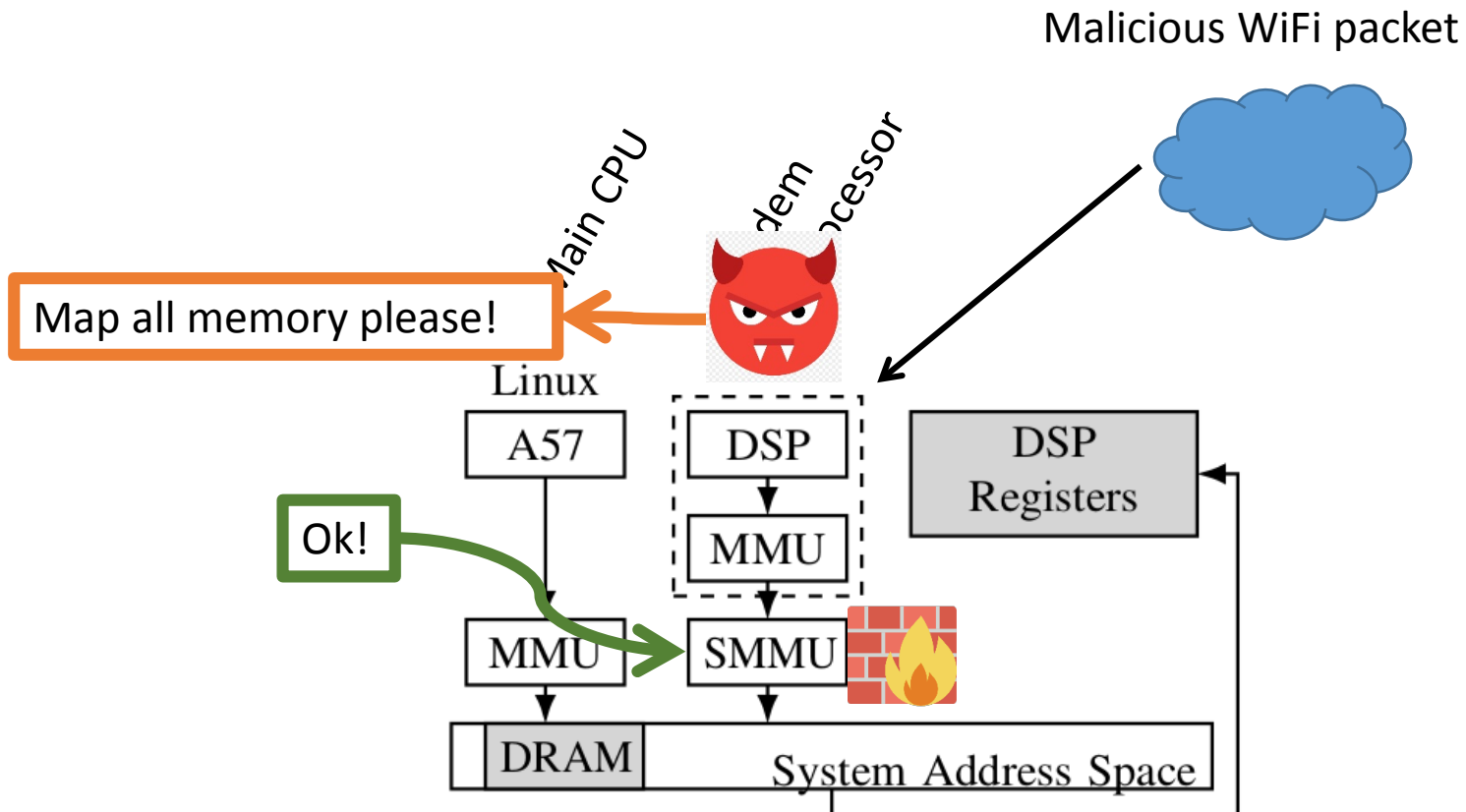
Doesn't look like ccNUMA to me.

- Modern SoCs run Linux - *but...*
- My Android phone runs **Spongebob Crusty Cook-off**, but that doesn't make it the OS.



Is this a problem?

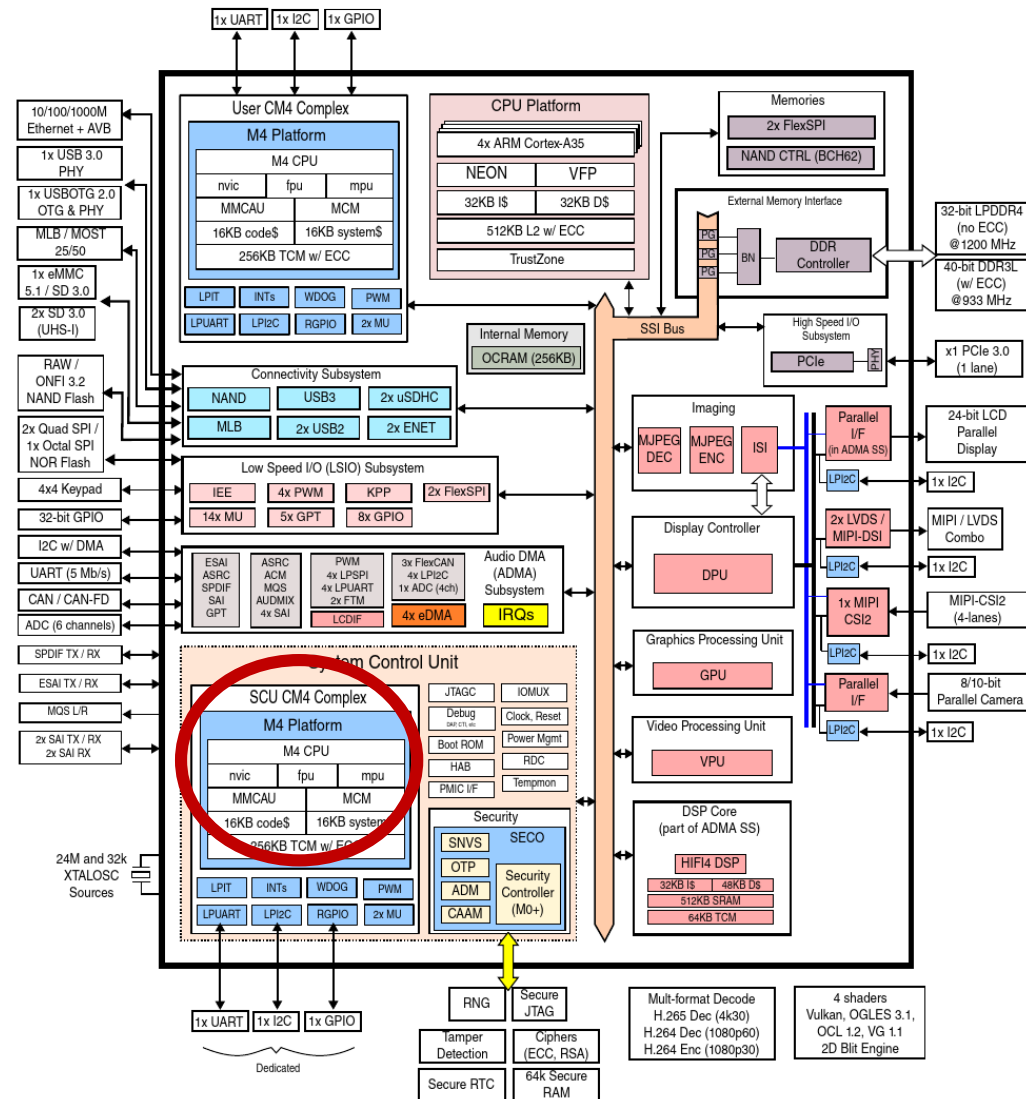
A security dumpster fire



A “cross SoC attack”.

ETH zürich

22

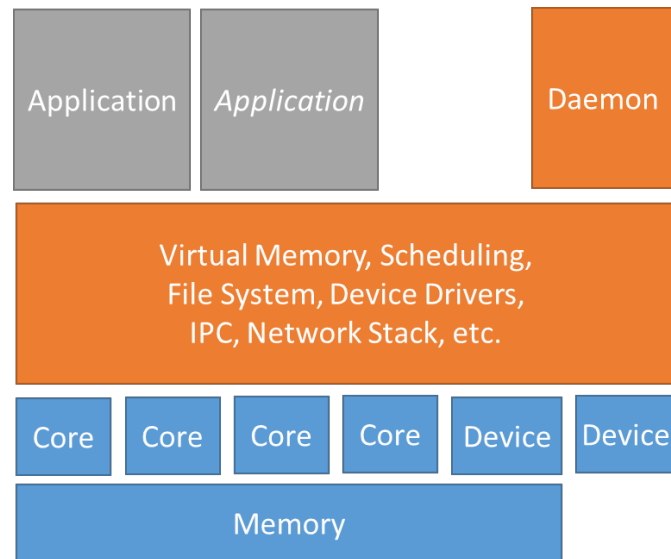
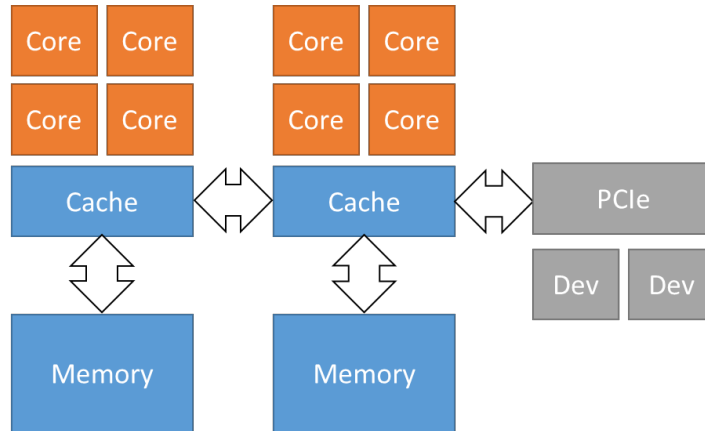


[illegible]

**What about those
“new OS” papers?**

All 3 of them...

They make the same hardware assumptions as Linux



Clear message:

- If you want to get published a lot at OSDI, work on:
 - Machine Learning
 - Databases
 - Distributed Systems and Consensus
 - Graph Processing
 - Security
 - Privacy

... indeed, *anything* but Operating Systems.

Clear message:

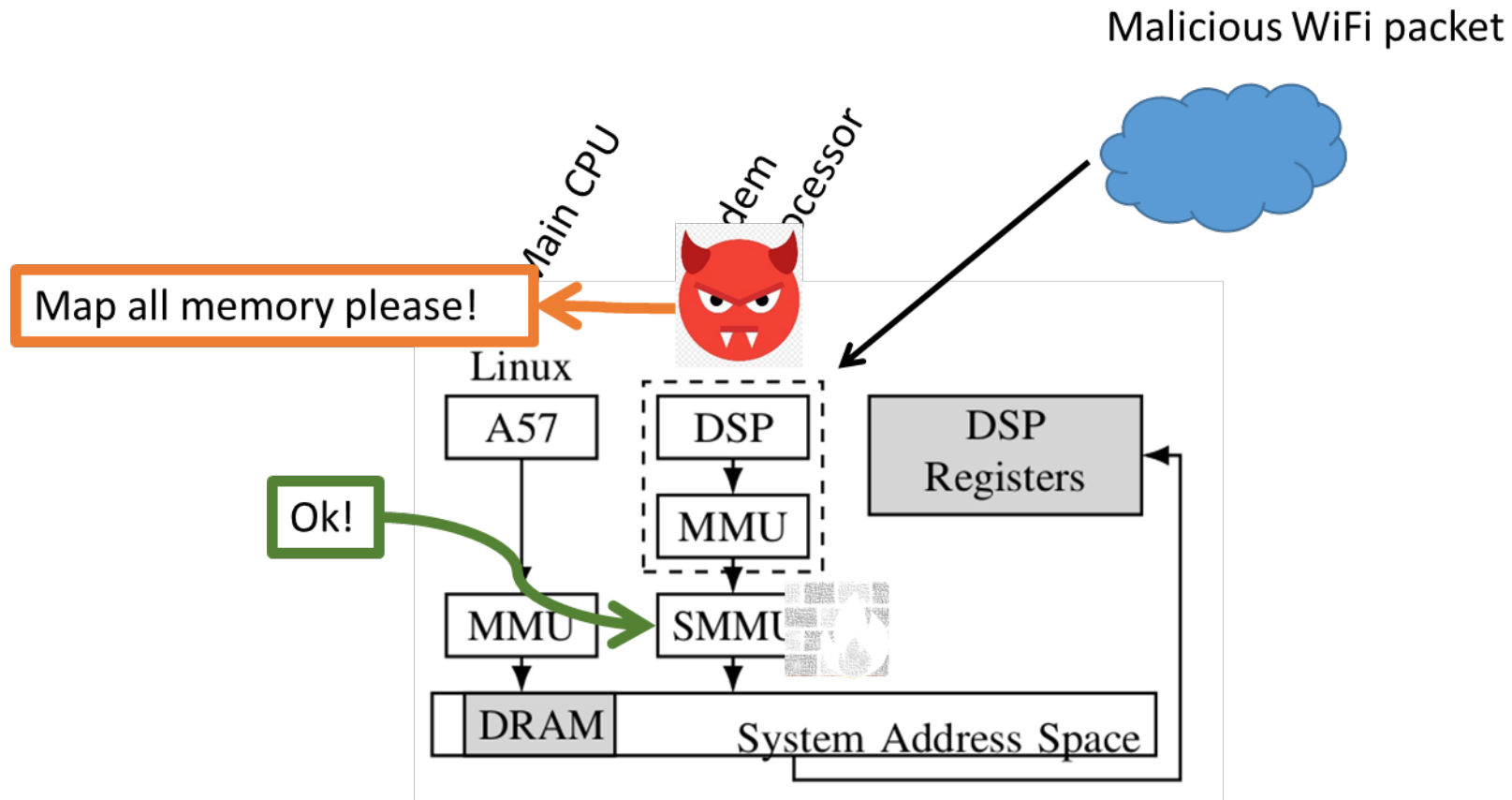
- If you *must* work on operating systems

Make sure it's Linux.

- If you want to be *really, really awkward* and work on a non-Linux OS:

Make sure it looks like Linux, and makes the same (false) assumptions about the hardware.

Anecdotal data point



The slide that will get me kicked off the OSDI steering committee

1. Too many people **don't know** what modern hardware looks like, and think it's just a fancy VAX.

This is called “ignorance”.

2. Too many people find it more **comfortable** to focus on
 - what Linux does well, and
 - the subset of hardware on which it does it.

This is called “denial”.

This is crazy! We should be excited!

- Plenty of interest in industry

Bespin




HarmonyOS



Etc..

Suggestion #1: Program for an SoC

They're not expensive.


Brexit Update
Angebote
Kontakt
Hilfe
Bestellungen verfolgen
+

Alle
Suchbegriff/Bestellnummer
Q
Anmelden
Registrieren
0 Artikel
CHF 0.00

Mein Konto









Alle Produkte
Hersteller
Ressourcen
Community
Favoriten
Einkaufstools

Please note: Following export compliance changes in line with EU regulations, orders under €150 shipped from our UK warehouse will require 1-2 day additional shipment delay. Please contact our sales teams for further information.

3 Produkte Gefunden

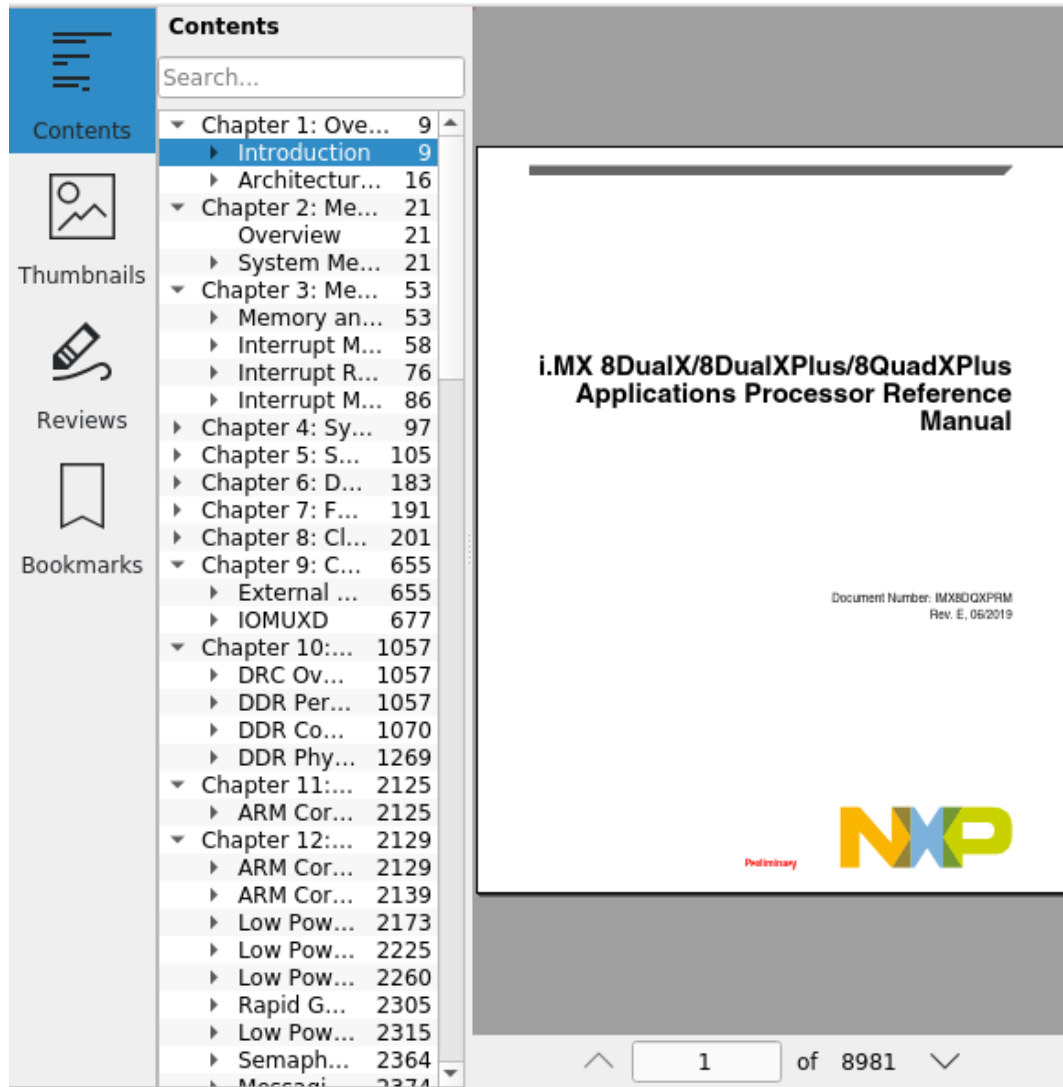
Vergleichen
Hinzufügen

1 25 pro Seite

	Hersteller-Teilnr.	Bestellnummer	Hersteller / Beschreibung	Verfügbarkeit	Preiseinheit:	Preis	Menge
<input type="checkbox"/>	RPi4-MODBP-2GB 	3051886  Data Sheet  RoHS	Einplatinencomputer, Raspberry Pi 4 Model B, BCM2711 SoC, 2GB DDR4 RAM, USB 3.0, PoE-fähig RASPBERRY-PI RASPBERRY PI 4 MODEL B, 2GB; Prozessorhersteller:Broadcom; Prozessorfamilie:BCM2xxx; Prozessorarchitektur:ARM; Prozessorserie:Cortex-A72; Prozessorkern:BCM2711; Lieferumfang des Kits:Raspberry Pi 4 Model B Board; Produktpalette:RPi 4 Model B; SVHC:No SVH	Mit Nachbestellung lieferbar Wiederbeschaffung	Stück	1+ CHF 39.51	<input type="text" value="1"/>  Min: 1 Mult: 1
<input type="checkbox"/>	RPi4-MODBP-4GB 	3051887  Data Sheet  RoHS	Einplatinencomputer, Raspberry Pi 4 Model B, BCM2711 SoC, 4GB DDR4 RAM, USB 3.0, PoE-fähig RASPBERRY-PI RASPBERRY PI 4 MODEL B, 4GB	Mit Nachbestellung lieferbar Wiederbeschaffung	Stück	1+ CHF 53.67	<input type="text" value="1"/>  Min: 1 Mult: 1

LIVE-CHAT

They can be a bit intimidating...



But this is great! This is what we do!

- Complexity is our business.
- We build tools to tame complexity.
- This is a real-world problem we can solve.
- Let's build better tools to handle this!
 - It's the basis for any verified kernel
 - You might synthesize code from specifications
 - You could do runtime verification
 - Derive secure interfaces, etc.

What you might find

The gap between Architecture and OS research, 2011

Mind the Gap: Reconnecting Architecture and OS Research

Jeffrey C. Mogul
Jeff.Mogul@hp.com
HP Labs, Palo Alto, CA

Andrew Baumann
Andrew.Baumann@microsoft.com
Microsoft Research, Redmond, WA

Timothy Roscoe
troscoe@inf.ethz.ch
ETH Zurich

Livio Soares
livio@eecg.toronto.edu
University of Toronto

Abstract

The goal of a computer system is to run an application workload securely, reliably, efficiently, and fast. A computer's hardware architecture and operating system exist to support this goal, and it would be nice if they cooperated as effectively as possible. Yet there is a growing gap between architectural research and OS research, which seems to be the result of poor communication about what actually matters.

In this paper, we discuss this gap and what to do about it. We illustrate the opportunities for closing the gap using examples from some recent OS research.

1 Introduction

For too long, operating systems researchers and developers have pretty much taken whatever computer architects have dished out. With occasional exceptions (e.g., virtualization support), architecture researchers do not appear to have sought or encouraged innovations that would improve the execution environment for an OS. Even worse, many do not bother to simulate and report

(We assume a loose definition of “the OS” – it's more than just the kernel, since in many cases people have moved OS functionality into user-mode libraries.) Perhaps this is not yet *ample* evidence, although we suspect this is mostly for lack of a systematic study.

But computer architects, from the evidence available in the scientific literature, assume that the OS does not exist – except perhaps when it magically manages application-thread resources that hardware cannot. Architecture papers commonly use application-only benchmarks, and seldom account for the interference between application and OS execution (there are, of course, counter-examples [9]). In short, while architecture papers sometimes pay lip service to the OS, they rarely discuss the impact of architecture on OS behavior.

Meanwhile, a typical OS paper usually uses the phrase “on commodity hardware”. As a community, we assume we are stuck with whatever flaws the hardware has.

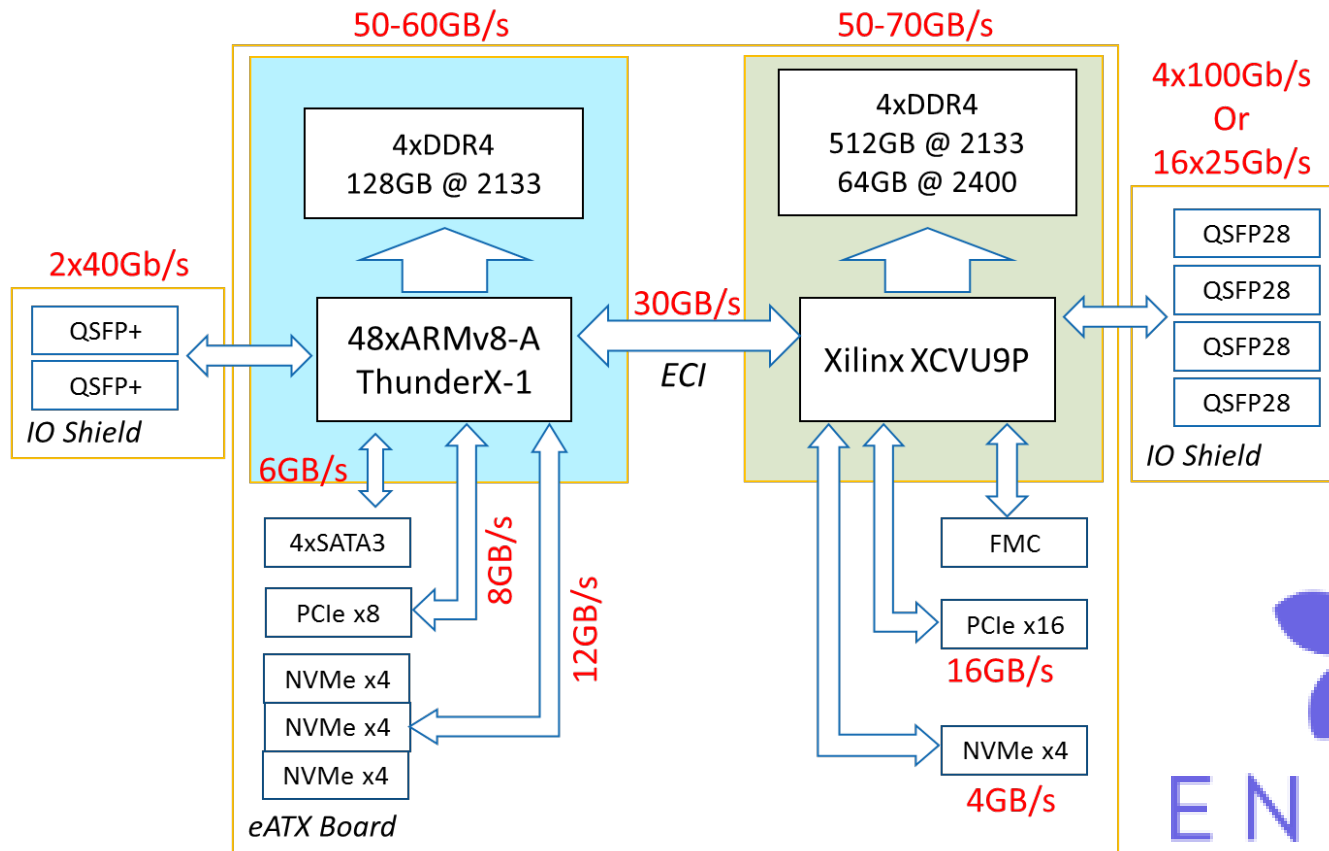
We see several infrastructural reasons why architecture researchers have been ignoring the OS: lack of quantitative evidence for the importance of OS execution



Suggestion #2: Build our own computers

What has changed since 2011?

- It's easier to get hardware built.



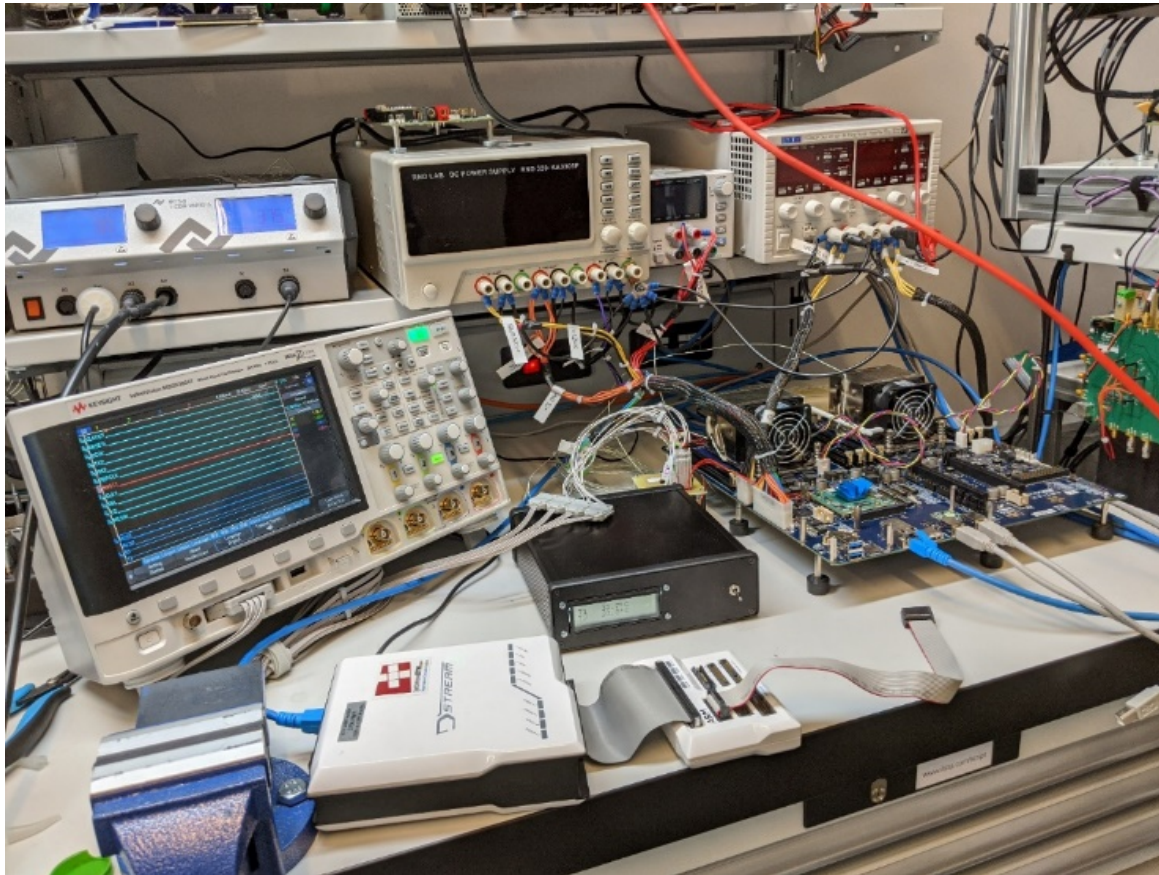
What has changed since 2011?

- It's easier to get hardware built.



Should we build hardware? Of course!

- We don't know what a computer looks like today.

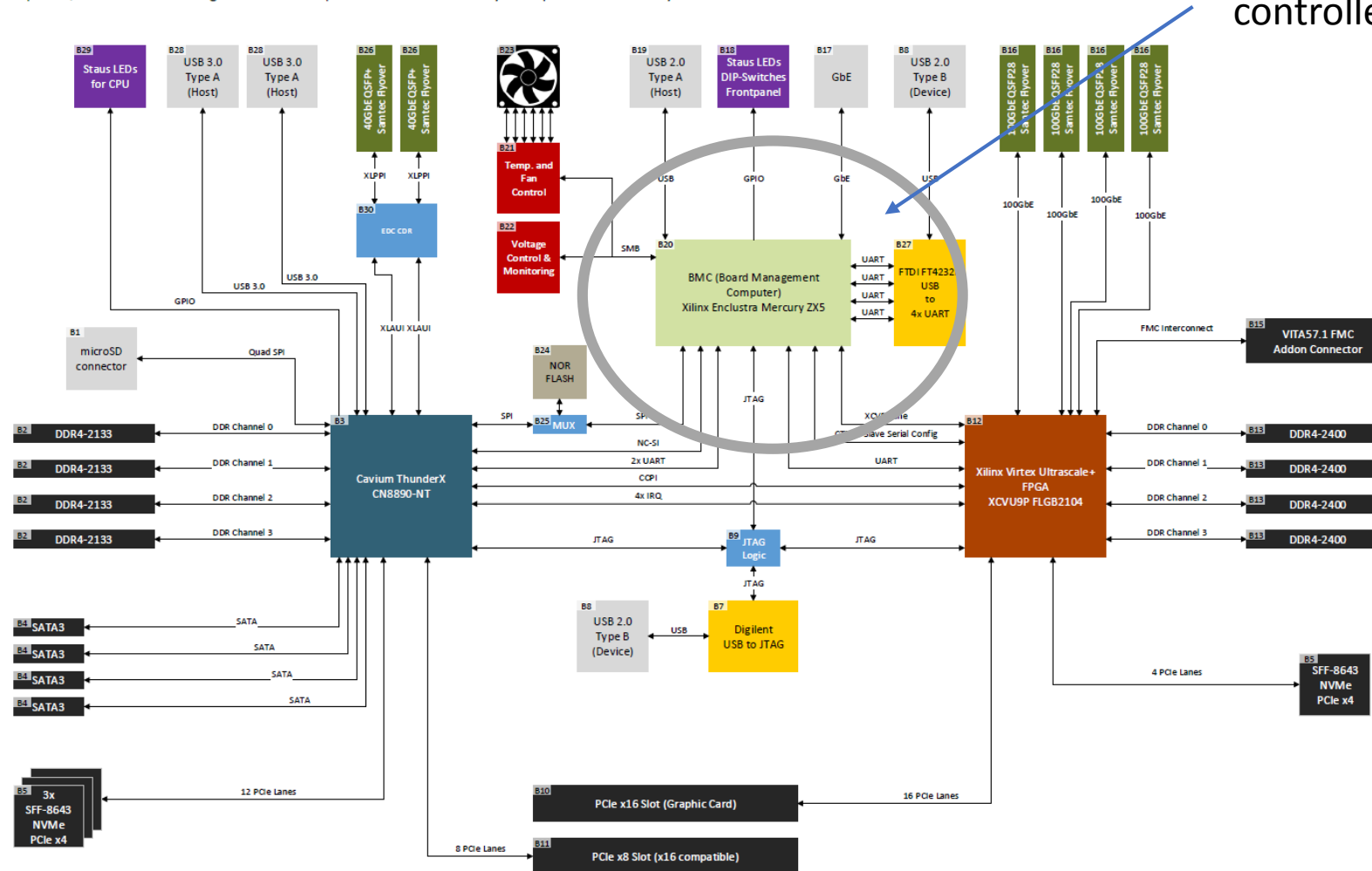


Another simplified diagram

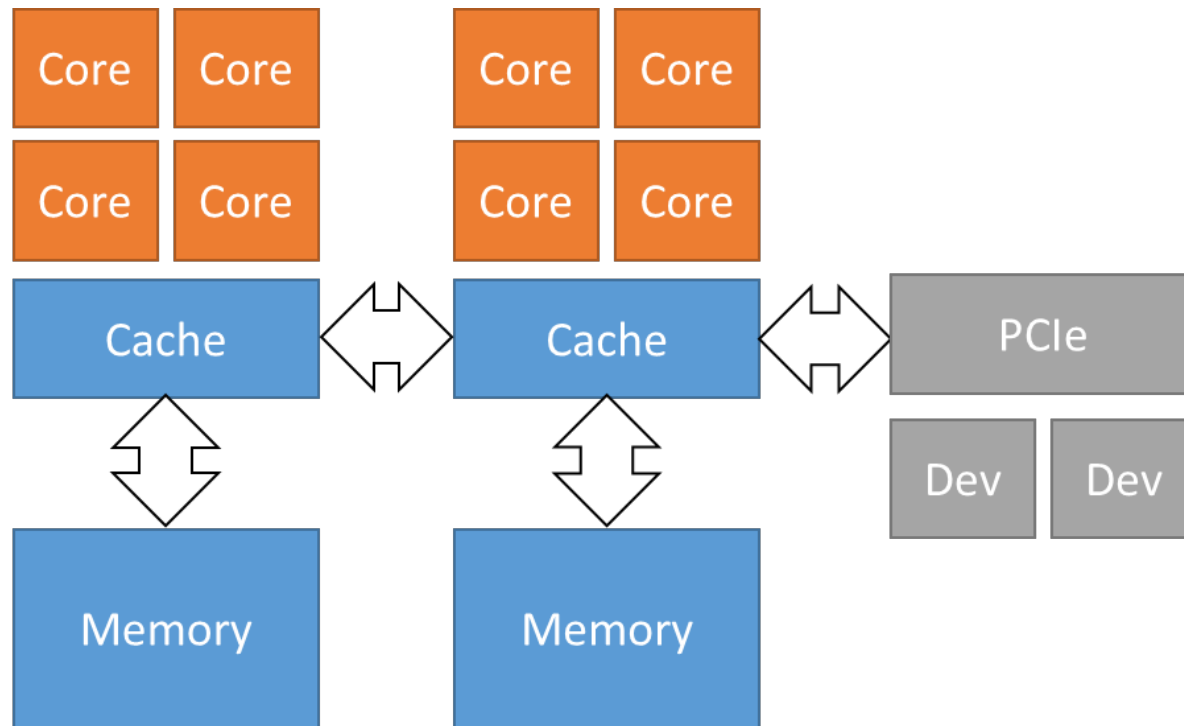
Enzian Signalflow Block Diagram SVN: 126

This diagram shows all components and signal connections. Refer to the numbered descriptions for details of each component/connection. The arrangement of the components does not necessarily correspond to the board layout.

Board
management
controller (BMC)



Back to that Linux hardware assumption:



- This is definitely *not* what a server looks like.
- This should be the final nail in the coffin of this model of a computer.
- It's a dangerous lie.

Summary

Summary

- OS research is *really needed* right now
 - Hardware is different in scary ways
 - Companies are writing new kernels
 - Lots of research in Security, Architecture, etc.
- This community is *discouraging* this work
- In Operating Systems:
 - **Denial** that there are problems beyond Linux
 - **Ignorance** about what hardware looks like to software

This is such an opportunity!

- There is so much **cool, fun, relevant** research that could be done today
 - We can really rethink OS structures for new h/w
- We've got **great tools** that didn't exist before
 - Great new formal techniques to apply
 - New languages
 - Amazing new hardware
- Why waste this on 50-year-old thinking?

Acknowledgements and thanks

- The Enzian team: Reto Achermann, Gustavo Alonso, David Cock, Dario Corolija, Michael Giardino, Tobias Grosser, Alexander Hedges, Nora Hossle, Amit Kulkarni, Nikita Lazarev, Jialin Li, Fabio Maschi, Jakob Meier, Abishek Ramdas, Timothy Roscoe, Anastasiia Ruzhanskaia, Muhsen Owaida, Jasmin Schult, Daniel Schwyn, David Sidler, Adam Turowski, Zeke Wang, Patrick Ziegler and many more....
- We are very grateful for support (equipment, parts, funding, and expertise) to Xilinx, Cavium/Marvell, and VMware Research

