



Anyscale with RISC-V: Powering the Next Generation of (IoT) HPC Systems

Moderator: Steve Wallach

BoF Agenda

- Panelist Introduction
- RISC-V background
- Background questions
- Audience questions
- Wrap-up



Panelists

- Luc Berger-Vergiat (Sandia National Labs)
- John Davis (Barcelona Supercomputer Center)
- John Leidel (Tactical Computing Labs)
- Doug Norton (InspireSemi)
- Michael Wong (Codeplay)

- Moderator: Steve Wallach



Overview

- RISC-V Instruction Set Architecture (ISA) is an open standard
- Design Freedom and Flexibility
- RISC-V has the strongest ecosystem



RISC-V Mission: RISC-V is the industry standard ISA across computing

RISC-V is inevitable

>10 Billion RISC-V cores already shipped.

- Innovation and adoption moving rapidly across all domains
- Demand at every performance level (low to ludicrous)
- Shared investment is driving the fastest growing ecosystem

Leverage a community ISA spec development model

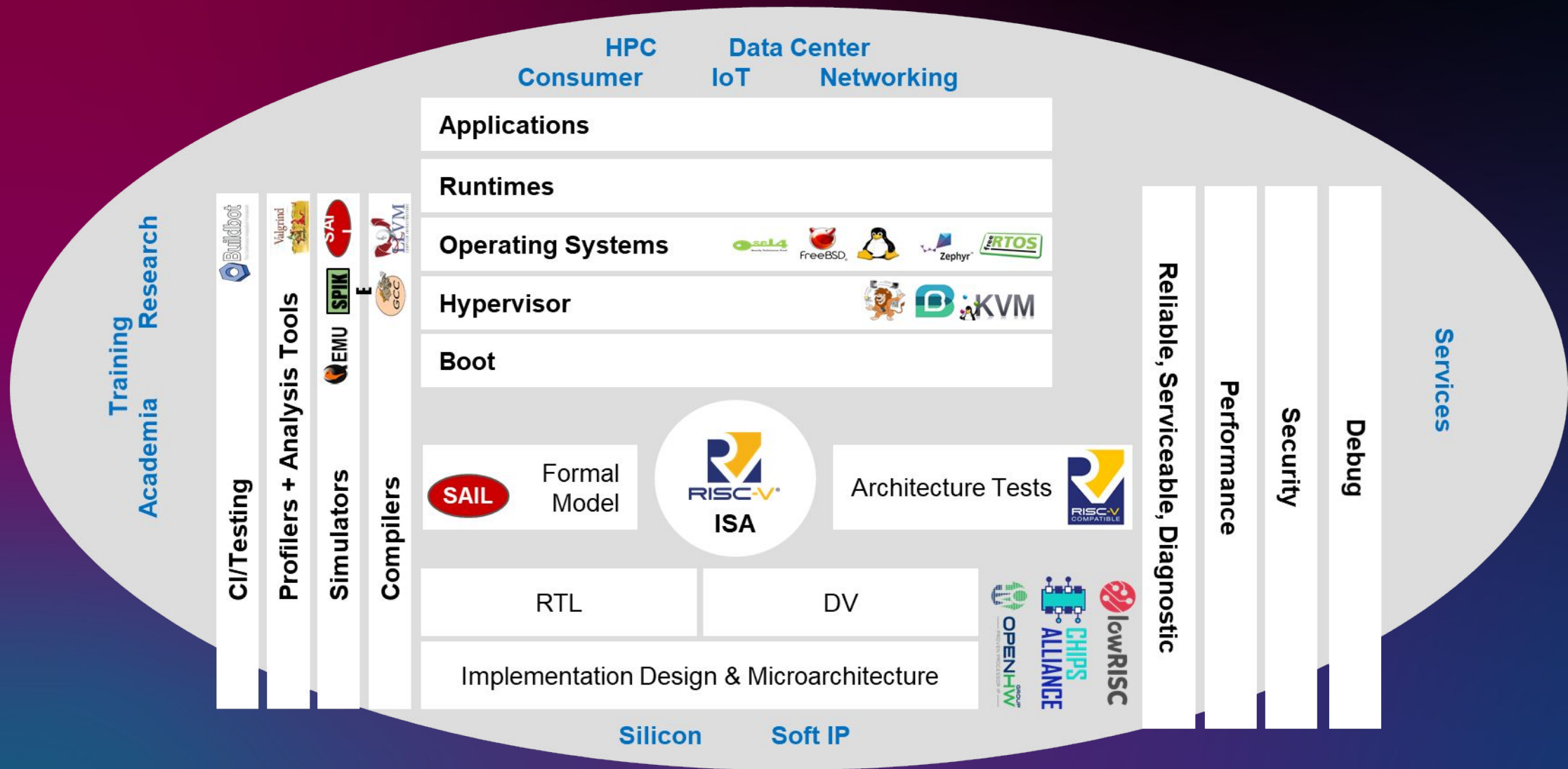


What is an ISA?

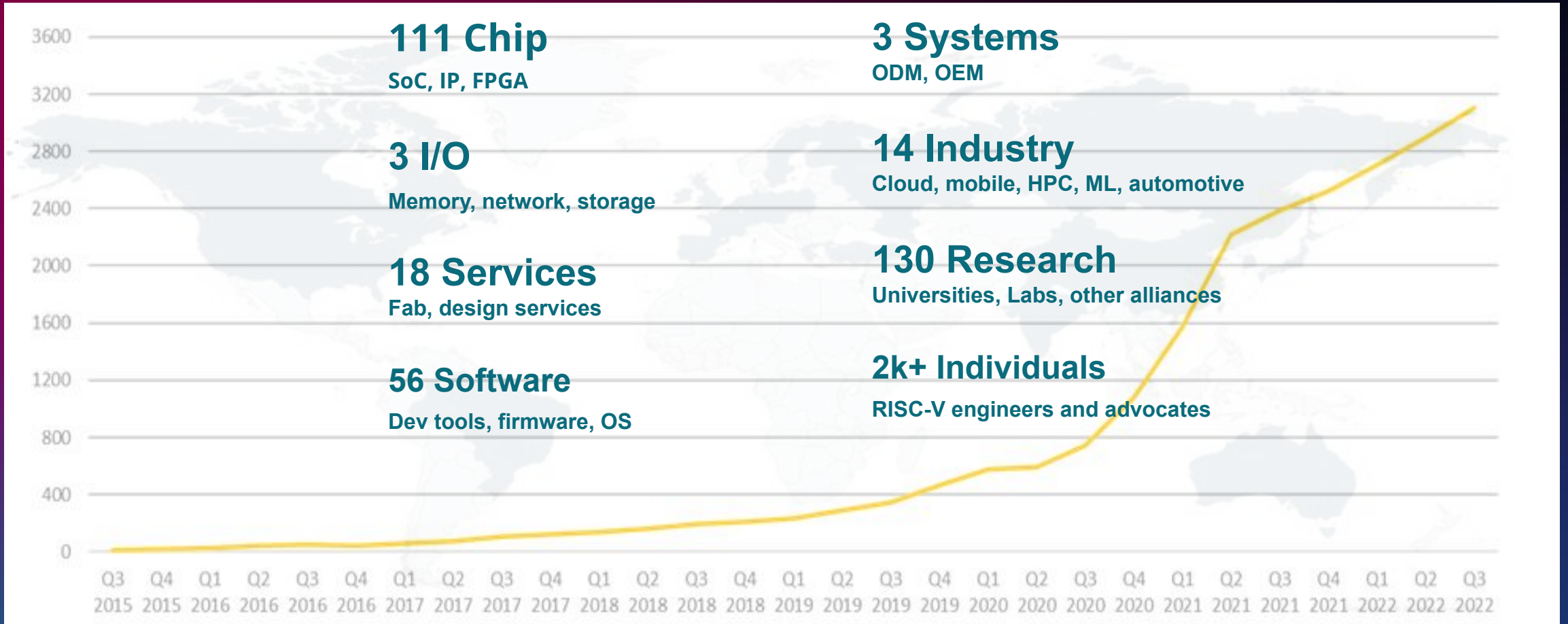
- An **Instruction Set Architecture (ISA)** is part of the abstract model of a computer that defines how the CPU is controlled by the software. The ISA acts as an interface between the hardware and the software, specifying both what the processor is capable of doing as well as how it gets done.
- The **ISA** provides the only way through which a user is able to interact with the hardware. It can be viewed as a **programmer's manual** because it's the portion of the machine that is visible to the assembly language programmer, the compiler writer, and the application programmer.
- The **ISA defines the supported data types, the registers, how the hardware manages main memory, key features (such as virtual memory), which instructions a microprocessor can execute, and the input/output model** of multiple ISA implementations. The ISA can be extended by adding instructions or other capabilities, or by adding support for larger addresses and data values.
- CPUs/devices that execute the instructions are an implementation of the ISA
 - ARM, MIPS, SPARC, Power, OpenPOWER, **RISC-V**, x86, etc...



RISC-V Ecosystem



RISC-V Community: 3100+ Members in 70 Countries



July 2022

RISC-V membership rapid growth of 134% in 2021



Where/How do you get started with RISC-V in HPC?



What is being done today?

- Bunch of Research projects
- Companies supporting the HPC Ecosystem
- Leveraging Open Source



RISC-V Software stack (EPI)

Board	OS	Details
PolarFire	Fedora	4 cores w/ 2 GB
BeagleV	Fedora	2 cores w/ 8 GB
Unmatched	Fedora/Ubuntu	4 cores w/ 16 GB
Allwinner D1 (Vector extension)	Fedora	1 core w/ 2 GB

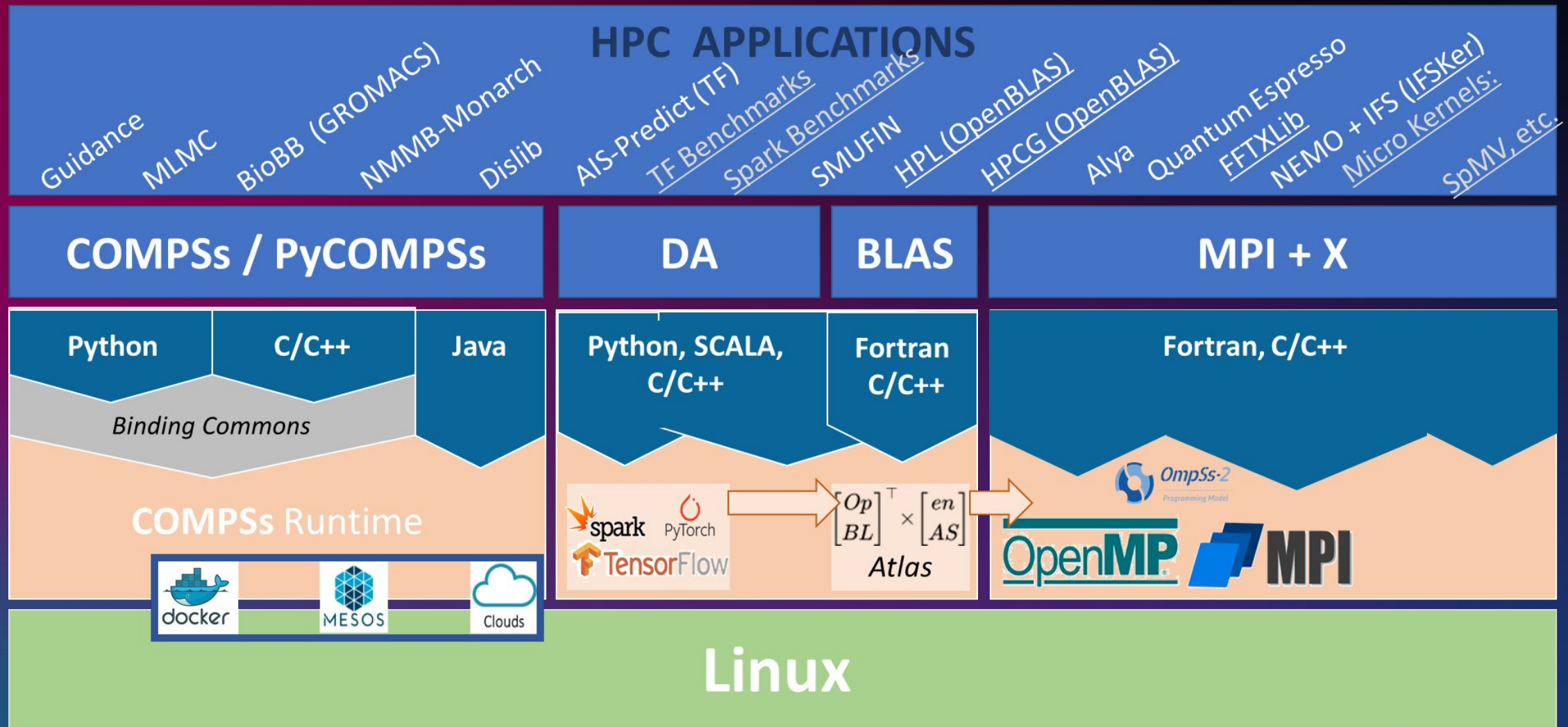
Emulators:

- A RISC-V soft vector core running in an FPGA.
- The Vehave RISC-V emulator on top of QEMU
- The Vehave RISC-V emulator on top of a native RISC-V core

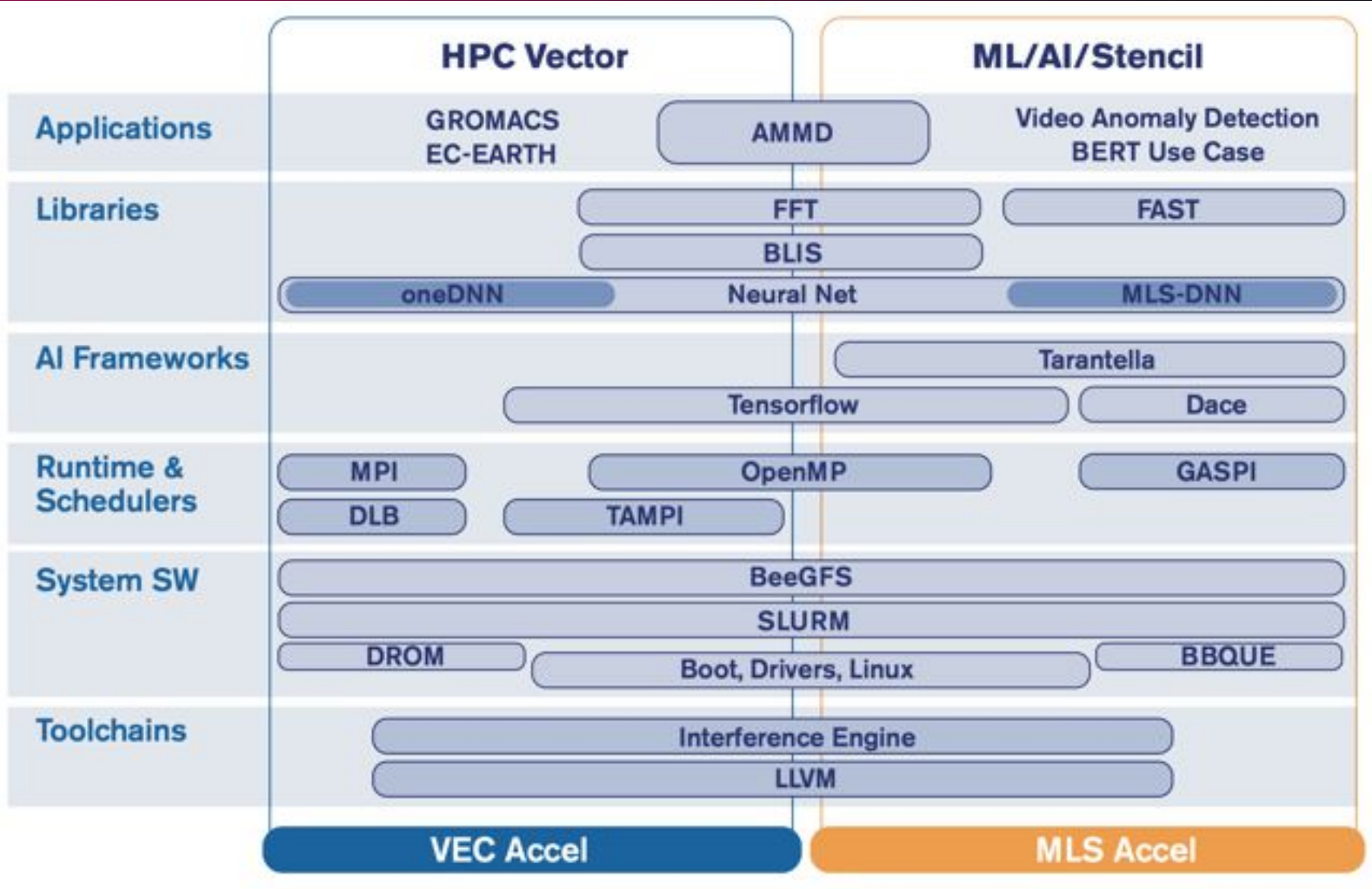
• RISC-V Software Stack:

- Linux, SLURM
- Compilers:
 - go/1.17
 - openmpi/fedora/4.1.1_gcc10.3.1
 - llvm/EPI-0.7-development
 - openmpi/ubuntu/4.1.1_gcc10.3.0
 - llvm/EPI-development
 - python/fedora/2.7.16
- Tools
 - extrae/3.8.3
 - papi/6.0.0
 - perf/5.11.10
 - singularity/3.8.2
- Libraries
 - boost/1.77.0
 - glibc/fedora/2.33
 - openBLAS/0.3.15
 - fftw/3.3.9_gcc10.3.1_ompi4.1.1
 - libunwind/git
 - openBLAS/0.3.17

RISC-V Software stack (<https://meep-project.eu/>)



RISC-V Software stack (<https://eupilot.eu/>)



RISC-V Student Cluster at ISC'22

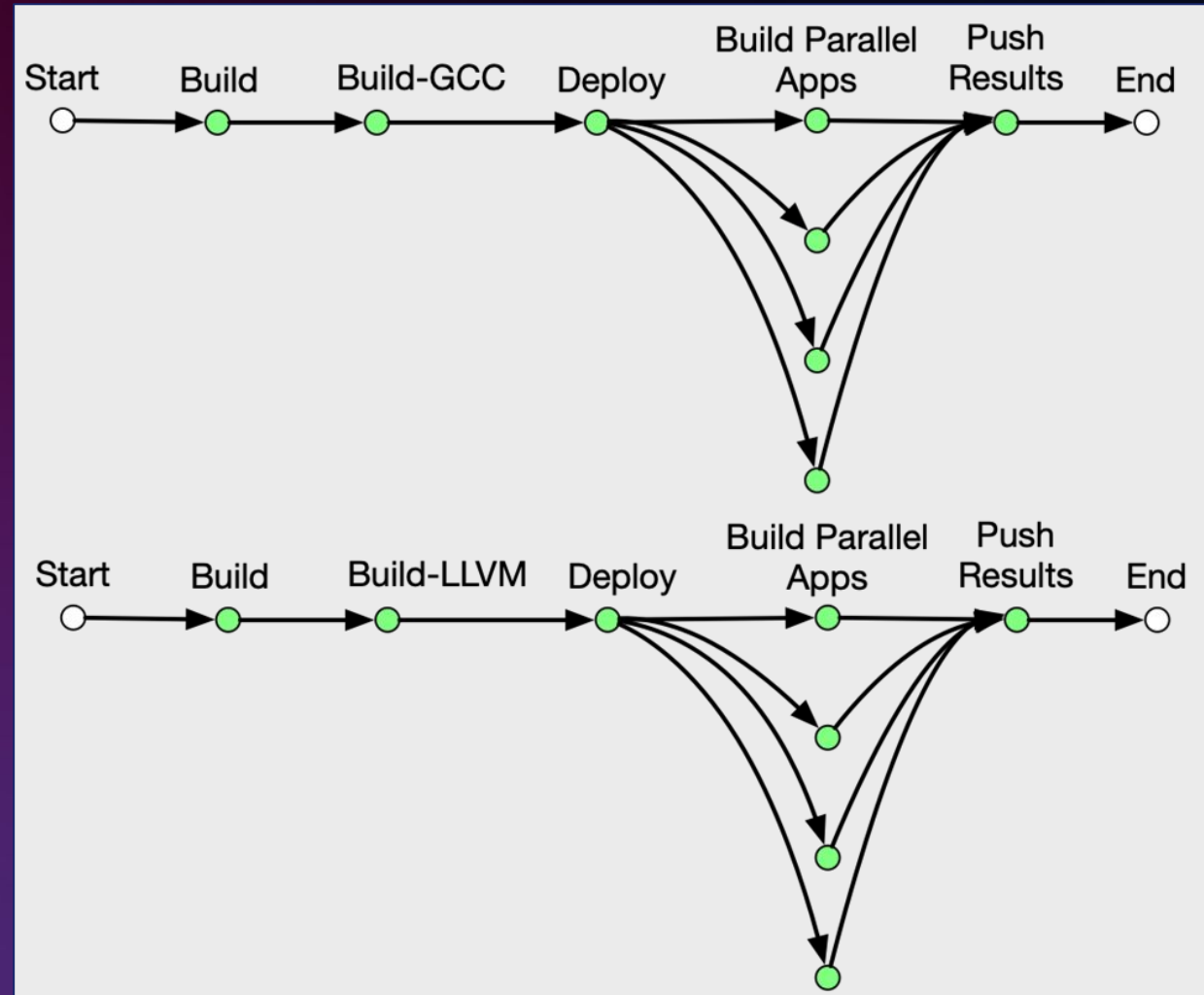
- Cluster of SiFive HiFive Unmatched boards (team NotOnlyFLOPs)
- https://easybuild.io/tech-talks/007_scc_isc22.html
- Monte Cimone paper: <https://arxiv.org/abs/2205.03725>
- Fan favorite award
- Compile and run HPC codes

Package	Version
gcc	10.3.0
openmpi	4.1.1
openblas	0.3.18
fftw	3.3.10
netlib-lapack	3.9.1
netlib-scalapack	2.1.0
hpl	2.3
stream	5.10
quantumESPRESSO	6.8



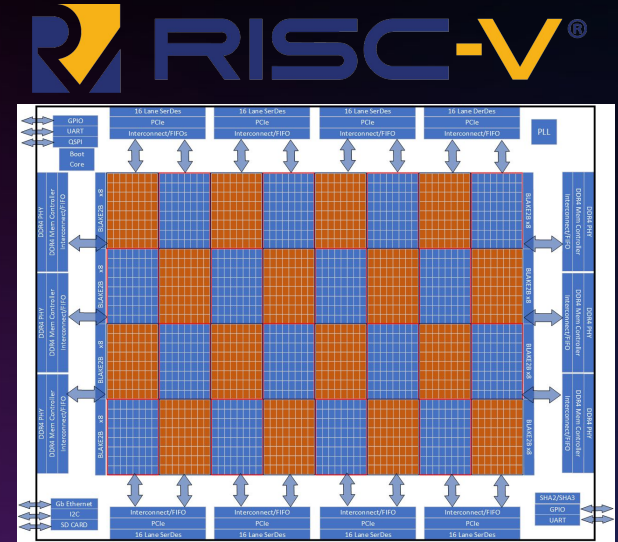
HPC Software Testbed: Does it compile?

- We are working with the RISC-V International group to drive requirements for adjacent working groups
- RISC-V HPC Tests
 - HPC-centric software test suite hosted by TCL
 - Multi-version compiler centric: GCC, LLVM
 - Using each compiler, we cross compile each target library, benchmark and application suite for RISC-V compatibility

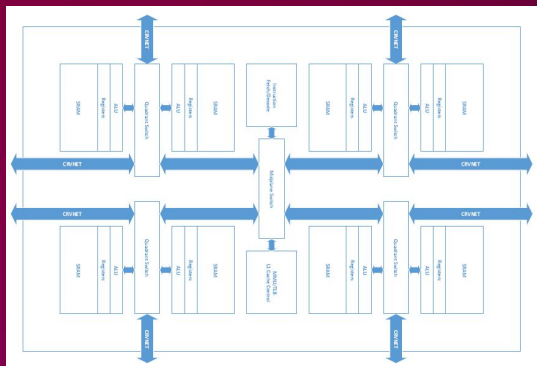


Thunderbird Accelerated Computing Solution

- **Ultra-Efficient, Ultra-compact custom CPU cores**
 - Based on RISC-V instruction set (like ARM but free & open)
 - Modern superscalar, out-of-order, vector-capable cores
 - We added custom instructions for AI/ML and crypto
- **A supercomputer cluster-on-a-chip**
 - 2,560 high-perf 64-bit CPU cores per chip (>5,000 per PCIe card)
 - Comparable to GPU shader count but are independent CPUs
- **Innovative high speed interconnect fabric**
 - Key to efficient utilization of so many cores
 - Seamlessly spans multiple-chip arrays up to 256 chips!
- **Energy efficiency: 30-60% power reduction**
 - Competitive with single-purpose ASICs for several altcoin algo's
- **Supporting existing open RISC-V software ecosystem**
 - Enables customers to easily adapt their software programs
 - Fast – no big investment or training required
- **Recognized global partners to deliver turnkey solutions**
 - High-volume, high margin across multiple markets



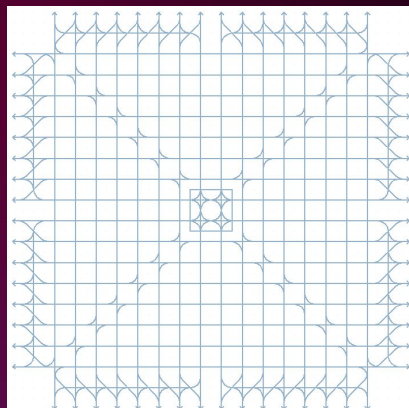
Thunderbird Technology Overview



Core

64-bit superscalar RISC-V CPU cores:

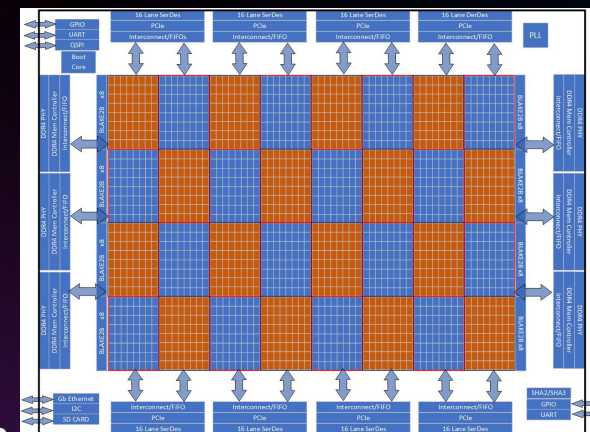
- Custom InspireSemi hi-perf design
- Multiple-issue, out-of-order, variable instruction width
- Vector, SIMD and tensor
- Mixed-precision floating point
- AI and cryptography extensions
- Tightly integrated memory and core-core network fabric
- Simple programming model
- Thriving software ecosystem



Interconnect Fabric

Manhattan street grid of 32-lane superhighways:

- Full utilization of precious routing area -> extreme bandwidth
- Flyover interchanges -> low congestion
- Express bypass lanes -> low latency
- Multiple onramps/offramps to each core
- 240TB/s local, 40TB/s global
- Uniform cellular layout



Top

- 2,560 CPU cores, SMP or HPC cluster-on-a-chip
- Network fabric extensible to arrays up to 256 chips
- Six DDR memory controllers
- 128 lanes PCIe / Eth / chip-to-chip
- Algorithm-specific accelerators

System

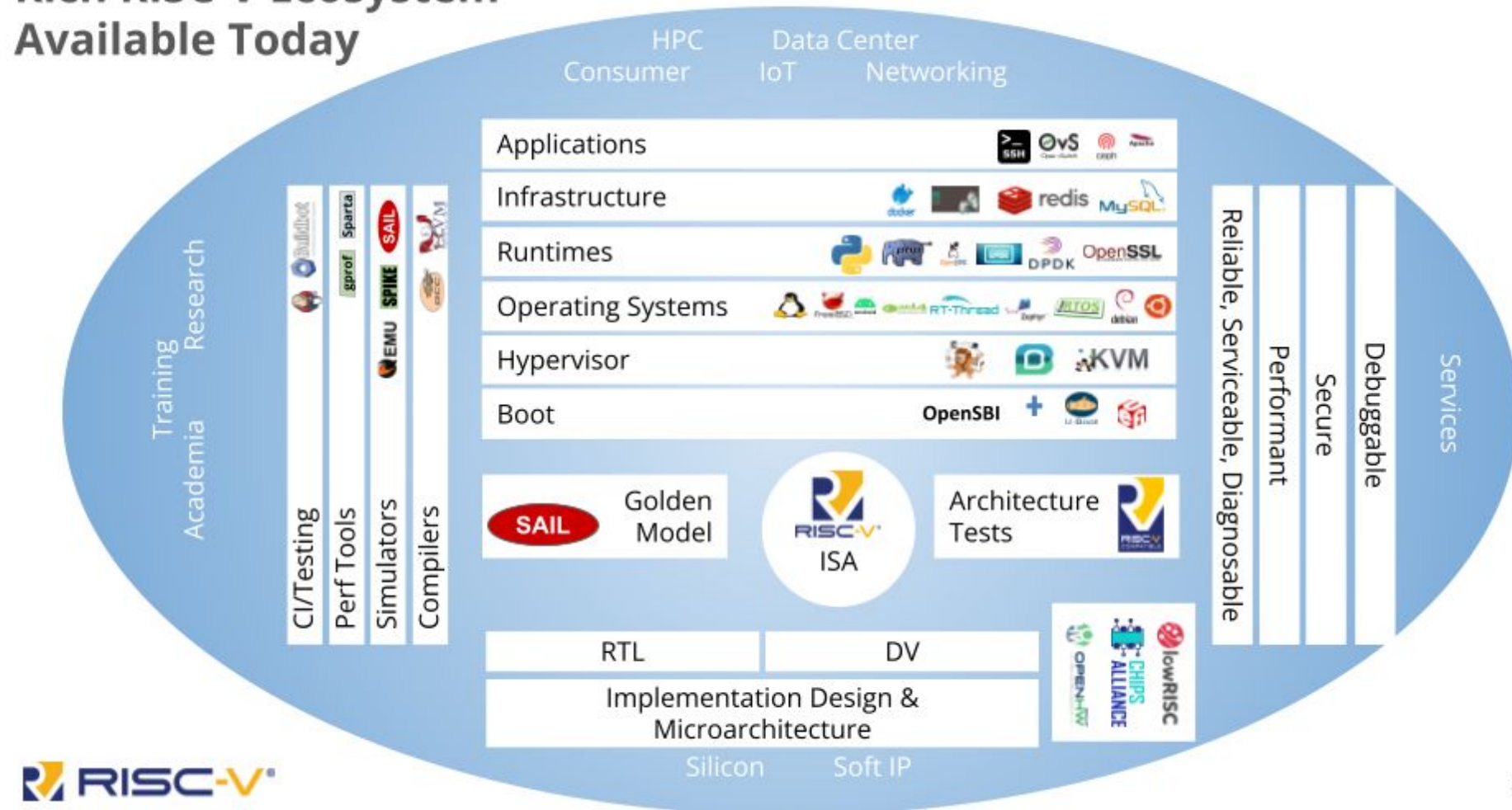
- Power conversion technology improves efficiency 10-25%
- Cooling technology maximizes density, increases efficiency



Thunderbird Differentiation

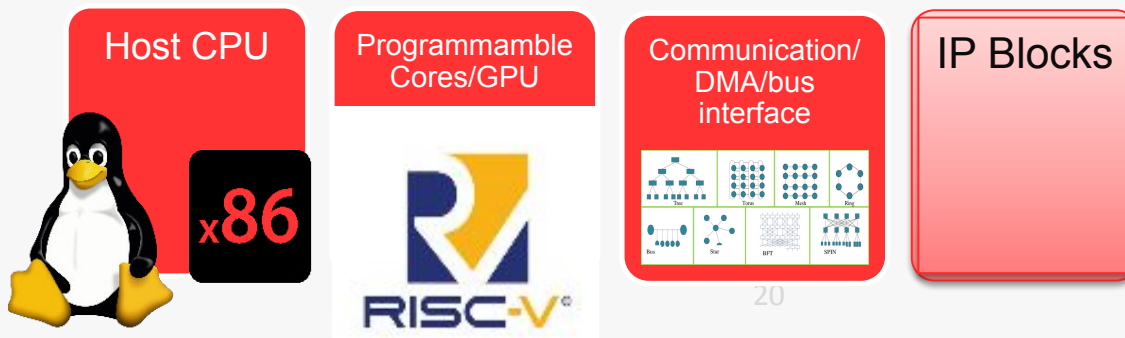
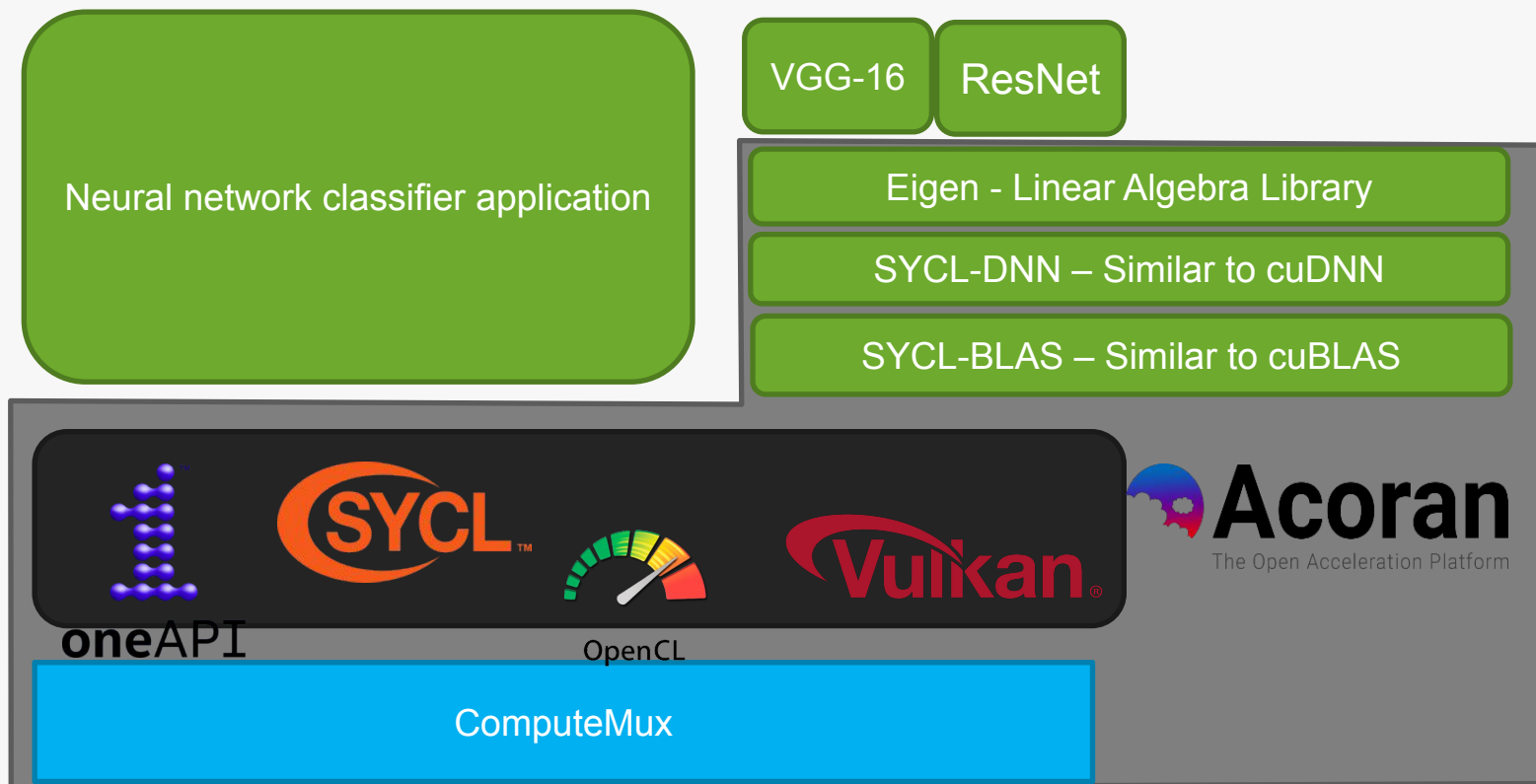
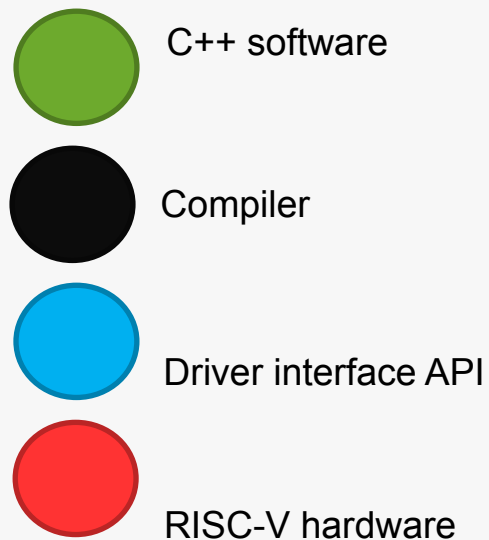
	InspireSemi	CPU	GPU	FPGA	AI Accelerators
Architecture	Many programs, many data streams.	Few programs, few data streams	Few programs, many data streams	Programmable logic elements	Single program, many data streams
Performance	High	Slow	Medium	Medium	High for AI only
Cost	Low ~\$200/chip \$5,000 per board	High ~\$1K-8K	High ~\$6K-10K	High \$8K-\$10K	High ~\$10K - \$2.2M
Energy consumption	Low ~175W/chip (~350W/PCIe board)	Med 240W+/chip	High ~700W	Med ~300W	High ~300W – 20kW
Multichip arrays	256	1-4	2-8	1	1-2
Programming model	Standard CPU-like, Any language, Full instruction set	Standard CPU, Any language, Full instruction set	Specialized C variant (CUDA, ROCM, SYCL)	Hardware description language	Proprietary, obscure
Software ecosystem	Open-source, Linux, compilers, libraries, AI frameworks, existing applications	Robust	Fragmented, limited, proprietary	None	AI frameworks and proprietary software stacks

Rich RISC-V Ecosystem Available Today



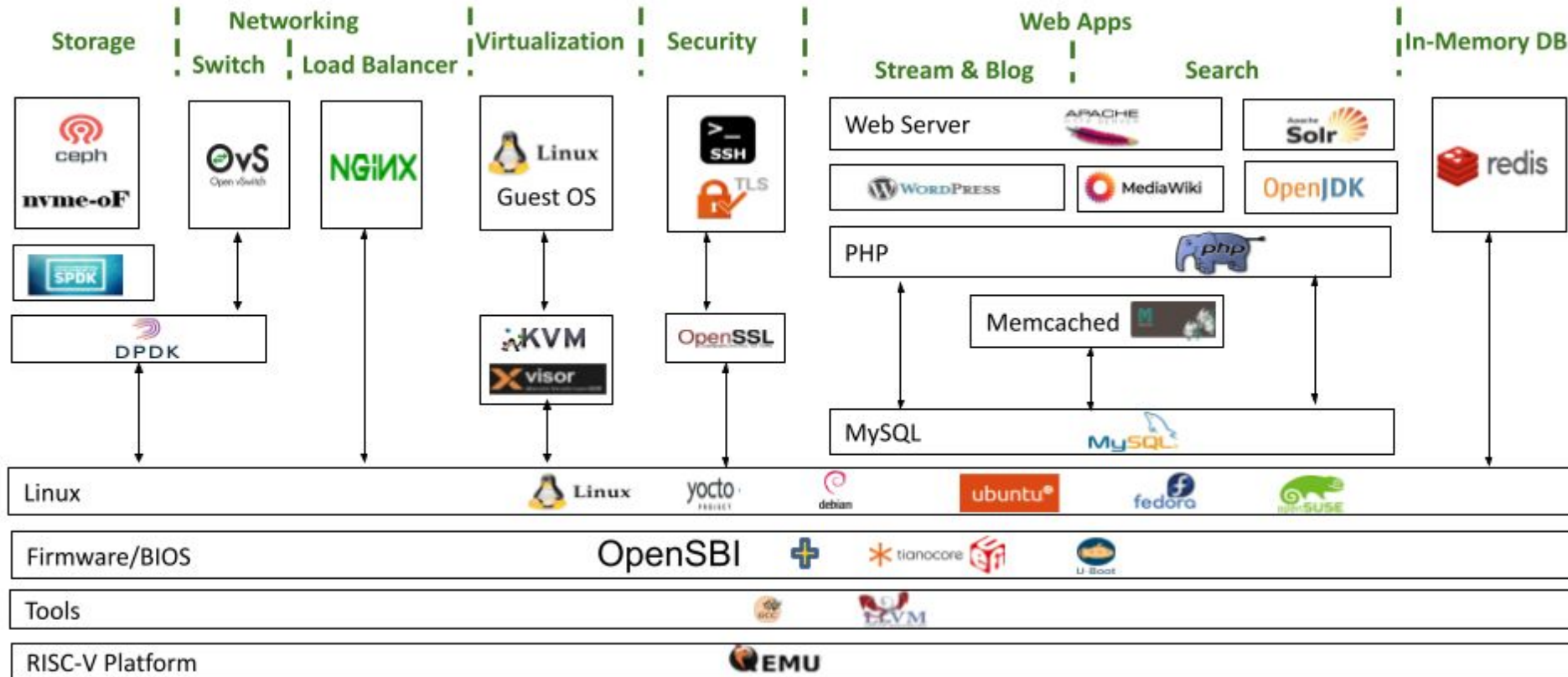
25

SYCL on RISC-V Architecture



The Acoran platform provides all the supporting open source libraries and frameworks needed to build this neural network demonstration.

Ventana Software Stack Examples Available Today



RISC-V

Andes Partners: AI, Multimedia and Security Today

AI tools, SW and IP



Security



DSP, Audio and Vision



Development tools

Taking RISC-V® Mainstream



RTOS

Getting started with RISC-V

- <https://riscv-test.org/>
 - Automated software compilation with GCC and LLVM
- Use RISC-V Labs that have public access
 - SUPER-V @ BSC: sdv-support@bsc.es
 - Nick Brown's Lab
 - Others
- QEMU
 - <https://risc-v-getting-started-guide.readthedocs.io/en/latest/linux-qemu.html>
- Buy/get RISC-V Hardware



What are the benefits and drawbacks of an open ISA?



When can RISC-V be on par with other HPC ecosystems?



Audience questions



If you had a RISC-V magic wand, what would you do/want? (Top 3)



Next Steps

- Join RISC-V SIG HPC, help define the future!
 - Subscribe: sig-hpc+subscribe@lists.riscv.org
- Build **your** software on RISC-V (Mind the gaps.)
 - Buy/get RISC-V Hardware
 - Request access to RISC-V testbeds:
 - SUPER-V RISC-V testbed: <https://repo.hca.bsc.es/gitlab/epi-public/risc-v-vector-simulation-environment/-/wikis/HCA-RISC%E2%80%90clusters-user-guide>
 - ExCALIBUR H&ES RISC-V testbed: <https://riscv.epcc.ed.ac.uk/>
 - <https://riscv-test.org/>
 - QEMU
 - <https://risc-v-getting-started-guide.readthedocs.io/en/latest/linux-qemu.html>



Get Involved!!!

SIG-HPC Vision & Mission: RISC-V: IoT to *HPC*

Vision:

The technical and strategic imperatives that guide the RISC-V ecosystem development to enable an Open HPC Ecosystem...

Mission:

...enable RISC-V in a broader set of new software and hardware opportunities in the High Performance Computing space, from the edge to supercomputers, and the software ecosystem required to run legacy and emerging (AI/ML/DL) HPC workloads

Subscribe: sig-hpc+subscribe@lists.riscv.org





Thank you

Join the RISC-V SIG HPC

Subscribe: sig-hpc+subscribe@lists.riscv.org