

RISC-V 给国内操作系统领域 带来的机遇和挑战

汇报人： 吴伟

PLCT实验室、TARSIER团队

wuwei2016@iscas.ac.cn | WeChat fangzhang1024 | GitHub/B站 lazyparser

欢迎参加本次交流！
本次交流比较轻松，欢迎提问

本次会议交流的内容和形式

内容简介

1. 演讲人的自我介绍（强调自己是门外汉并且并不是博士）
2. 芯片和软件技术发展的几个观测判断（也就是假设前提）
3. 操作系统（尤其是Linux发行版）的基本的要素
4. 为什么会有“国内操作系统领域”这个奇怪的说法
5. 为什么软件的验收和市场筛选怎么难？
 1. （这个问题很深，我也不确定自己是不是把握住了）
6. RISC-V 是什么，为什么这么火
7. 为什么说RISC-V给国内的操作系统圈子带来了机遇和挑战？
8. TARSIER观测计划介绍，以及参与进入开源社区的几种方式

交流形式

- 请务必将心中的疑问提出来，交流讨论。我们有1个小时的时间，甚至还可以拖
 - 今天的演讲者只是“门口的野蛮人”，没有顾忌。不代表任何群体。提供一个新颖的视角

自我介绍：吴伟（lazyparser）

- 年少苦练屠龙技；如今天天PPT
 - 在中科院软件所硕博连读失败（退学），于是留下来打工到现在
被同事们戏称“吴老板”并被团队内部创出“我只有钱”的微信表情包系列
- 创建了PLCT实验室和TARSIER团队
 - 2019年组建PLCT实验室，专注编译器、虚拟机、模拟器工程方向，以RISC-V为起点
 - 2021年底组建TARSIER团队，RISC-V操作系统团队，成规模的完善和维护RISC-V开源软件
- OSDT等社区的负责人（群主）
 - 2009年加入HelloGCC社区，后成为负责人；2013年前后扩大为OSDT社区；
 - 2018年前后创立HelloLLVM（作为OSDT社区的子社区）
 - 2016年开始关注到RISC-V，开始学习、赞助、组织沙龙和培训

请注意：本次报告仅代表我个人观点（和偏见），不代表PLCT、ISRC、ISCAS或其他第三方。

吴伟本人并非操作系统技术出身，技术观点属于门外汉级别，欢迎自由讨论环节纠正我的错误。

PLCT实验室的使命定位

程序语言与编译技术（PLCT）实验室致力于成为**编译技术领域的开源领导者**，推进**开源工具链及运行时系统等软件基础设施**的技术革新，具备**主导开发和维护重要基础设施**的技术及管理能力。与此同时，努力成为**编译领域培养尖端人才**的黄埔军校，推动先进编译技术在国内的普及和发展。

TARSIER团队的愿景和使命

愿景：让RISC-V成为所有主流开源软件的Tier-1平台。

使命：确保 Debian/Ubuntu、Fedora、openEuler 等所有流行的 Linux 发行版在 RISC-V 平台上平稳流畅运行，软件生态丰富性、可用性以及使用体验达到并超过 X86 及Arm64 平台。

战略目标：2025年之前促成所有流行Linux发行版将RISC-V提升为默认支持架构；使用RISC-V笔记本满足日常办公需求；支撑RISC-V进入超算领域所需的所有开源软件栈。

只要你也有同样的愿景，我们就有可能成为一起奔跑的伙伴，欢迎加入我们 ☺ → wuwei2016@iscas.ac.cn

本次会议交流的内容和形式

内容简介

1. 演讲人的自我介绍（强调自己是门外汉并且并不是博士）
2. 芯片和软件技术发展的几个观测判断（也就是假设前提）
3. 操作系统（尤其是Linux发行版）的基本的要素
4. 为什么会有“国内操作系统领域”这个奇怪的说法
5. 为什么软件的验收和市场筛选怎么难？
 1. （这个问题很深，我也不确定自己是不是把握住了）
6. RISC-V 是什么，为什么这么火
7. 为什么说RISC-V给国内的操作系统圈子带来了机遇和挑战？
8. TARSIER观测计划介绍，以及参与进入开源社区的几种方式

交流形式

- 请务必将心中的疑问提出来，交流讨论。我们有1个小时的时间，甚至还可以拖
 - 今天的演讲者只是“门口的野蛮人”，没有顾忌。不代表任何群体。提供一个新颖的视角

三个基本观测（假设）

1. 摩尔定律是有极限的，而算力需求没有极限

三个基本观测（假设）

1. 摩尔定律是有极限的，而算力需求没有极限

顶端优势 | 芯片设计成本 | 芯片制造成本 | 设计工具 | 软件栈

DSA

三个基本观测（假设）

1. 摩尔定律是有极限的，而算力需求没有极限
2. 软件系统的复杂度是超线性增长的

三个基本观测（假设）

1. 摩尔定律是有极限的，而算力需求没有极限
2. 软件系统的复杂度是超线性增长的

“已经没有任何公司或主权可以独立维护所有的软件栈”

“软件吞噬世界，开源软件吞噬软件”

三个基本观测（假设）

1. 摩尔定律是有极限的，而算力需求没有极限
2. 软件系统的复杂度是超线性增长的
3. 有能力驾驭软件开发复杂度的开发者是有限的

三个基本观测（假设）

1. 摩尔定律是有极限的，而算力需求没有极限
2. 软件系统的复杂度是超线性增长的
3. 有能力驾驭软件开发复杂度的开发者是有限的

一个细分领域只有头部一两个开源社区最终活跃，而

不被上游维护的代码就像是活在ICU里：费用昂贵、死亡率高

本次会议交流的内容和形式

内容简介

1. 演讲人的自我介绍（强调自己是门外汉并且并不是博士）
2. 芯片和软件技术发展的几个观测判断（也就是假设前提）
3. 操作系统（尤其是Linux发行版）的基本的要素
4. 为什么会有“国内操作系统领域”这个奇怪的说法
5. 为什么软件的验收和市场筛选怎么难？
 1. （这个问题很深，我也不确定自己是不是把握住了）
6. RISC-V 是什么，为什么这么火
7. 为什么说RISC-V给国内的操作系统圈子带来了机遇和挑战？
8. TARSIER观测计划介绍，以及参与进入开源社区的几种方式

交流形式

- 请务必将心中的疑问提出来，交流讨论。我们有1个小时的时间，甚至还可以拖
 - 今天的演讲者只是“门口的野蛮人”，没有顾忌。不代表任何群体。提供一个新颖的视角

所以，什么是操作系统（来自门外汉的发言）

如果我没有记错教科书里面的内容……

- 第一，管理各种硬件资源，并进行抽象，提供给软件
- 第二，为使用者提供一个接口，并对各种资源进行调度

操作系统有很多

- Windows、GNU/Linux、iOS、macOS、Android、MINIX 等
- RT-Thread、FreeRTOS、SylixOS、OpenHarmony 等

有了 UNIX/POSIX 和 GNU/Linux 之后，世界线趋于统一

- Linux as Driver Library
- POSIX

那么一个Linux发行版，有什么用？

- 理论上，如果你最够强，足够闲，足够耐心
 - 你就可以从零开始自己构建所有需要的开源软件，包括工具链、操作系统内核、基础工具包、桌面环境、浏览器等。
 - 你也可以创建自己的发行版，无论是从头开始，还是衍生
 - 例如TARSIER团队的实习生就基于openEuler、龙蜥等创建了 RobinOS 等衍生
- 构建一时爽，依赖冲突泪两行（对软件开发者的一致性支持）
- 安装一时爽，安全更新忘光光（安全保障）
- 自由分发时雄心壮志，客户支持时汗流不止（技术支持）
- 如果小白客户太多，开发者会不会累到沉默（建立社区互助）

本次会议交流的内容和形式

内容简介

1. 演讲人的自我介绍（强调自己是门外汉并且并不是博士）
2. 芯片和软件技术发展的几个观测判断（也就是假设前提）
3. 操作系统（尤其是Linux发行版）的基本的要素
4. 为什么会有“国内操作系统领域”这个奇怪的说法
5. 为什么软件的验收和市场筛选怎么难？
 1. （这个问题很深，我也不确定自己是不是把握住了）
6. RISC-V 是什么，为什么这么火
7. 为什么说RISC-V给国内的操作系统圈子带来了机遇和挑战？
8. TARSIER观测计划介绍，以及参与进入开源社区的几种方式

交流形式

- 请务必将心中的疑问提出来，交流讨论。我们有1个小时的时间，甚至还可以拖
 - 今天的演讲者只是“门口的野蛮人”，没有顾忌。不代表任何群体。提供一个新颖的视角

继续进行之前，请先在颅内建立分类器

技术用语 ≠ 营销词汇

网上的很多骂战都来自于把营销用语当成了硬核技术用语
而很多国内企业活该被骂，就是因为把技术用语营销用了

这就需要结合之前的几点假设和观测

- （主要的逻辑关系通过口述，这里只写一个结论）

只有一个新兴领域才有新的机会。

成熟领域在自由市场上几乎不会有新的机会，除非……

本次会议交流的内容和形式

内容简介

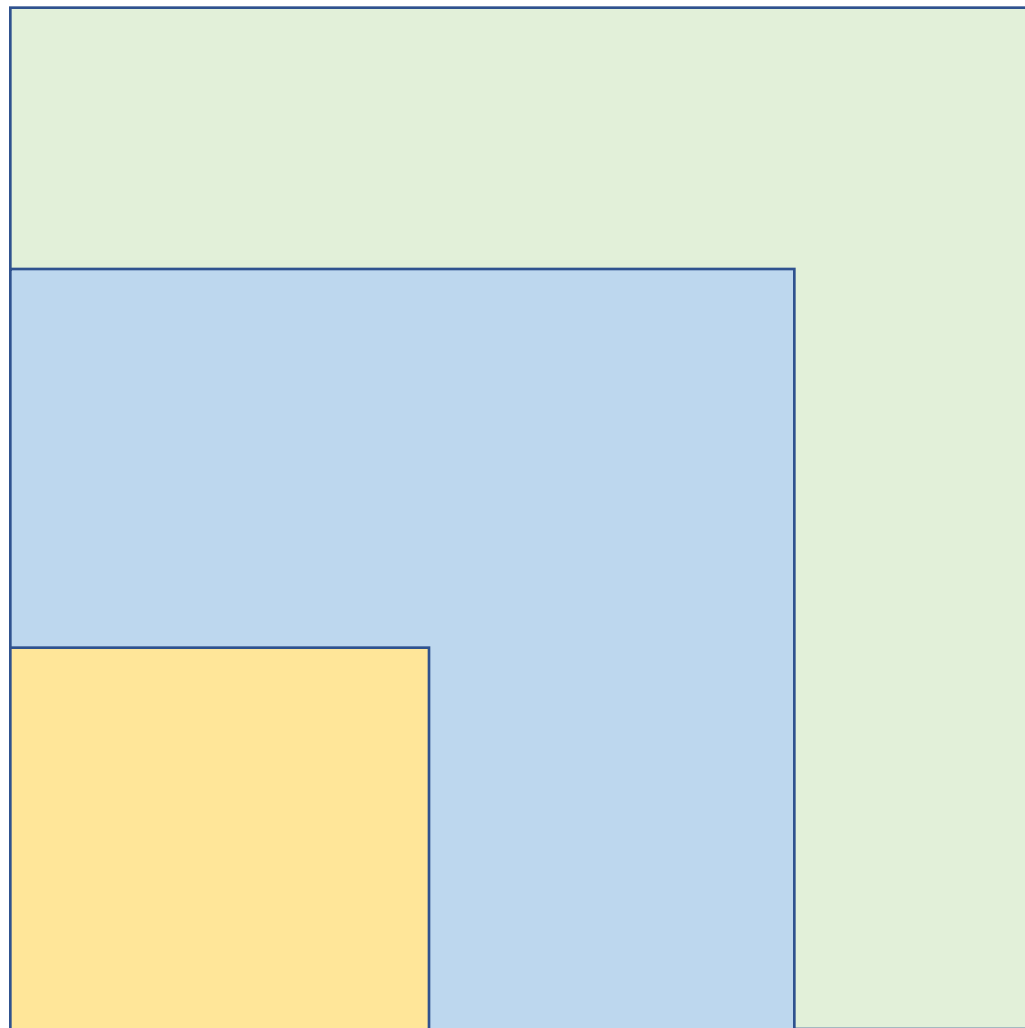
1. 演讲人的自我介绍（强调自己是门外汉并且并不是博士）
2. 芯片和软件技术发展的几个观测判断（也就是假设前提）
3. 操作系统（尤其是Linux发行版）的基本的要素
4. 为什么会有“国内操作系统领域”这个奇怪的说法
5. 为什么软件的验收和市场筛选怎么难？
 1. （这个问题很深，我也不确定自己是不是把握住了）
6. RISC-V 是什么，为什么这么火
7. 为什么说RISC-V给国内的操作系统圈子带来了机遇和挑战？
8. TARSIER观测计划介绍，以及参与进入开源社区的几种方式

交流形式

- 请务必将心中的疑问提出来，交流讨论。我们有1个小时的时间，甚至还可以拖
 - 今天的演讲者只是“门口的野蛮人”，没有顾忌。不代表任何群体。提供一个新颖的视角

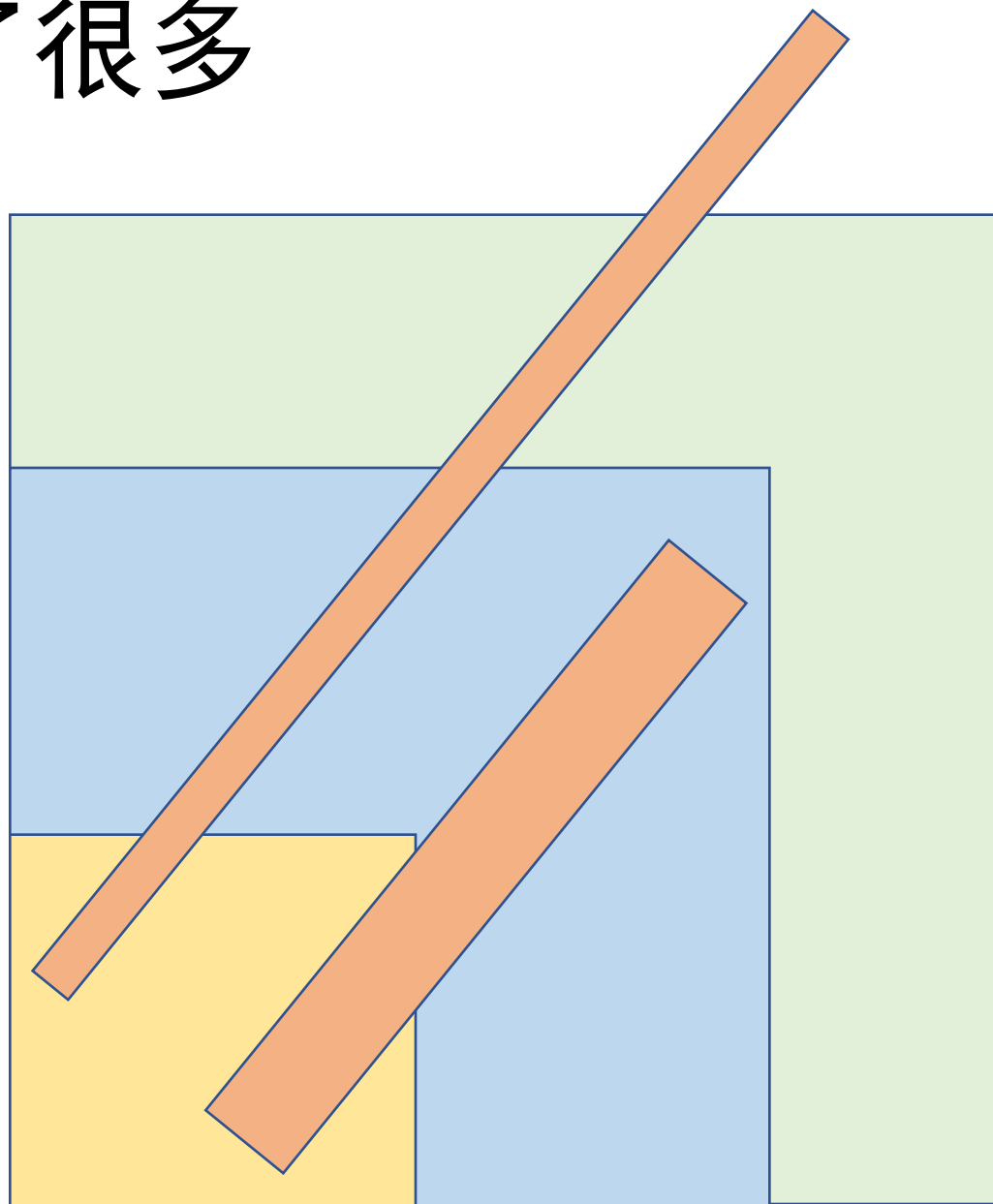
G: 钱我给了，而且给了很多

- 但是为什么好像国内并没有变强？



G: 钱我给了，而且给了很多

- 但是为什么好像国内并没有变强？



本次会议交流的内容和形式

内容简介

1. 演讲人的自我介绍（强调自己是门外汉并且并不是博士）
2. 芯片和软件技术发展的几个观测判断（也就是假设前提）
3. 操作系统（尤其是Linux发行版）的基本的要素
4. 为什么会有“国内操作系统领域”这个奇怪的说法
5. 为什么软件的验收和市场筛选怎么难？
 1. （这个问题很深，我也不确定自己是不是把握住了）
6. RISC-V 是什么，为什么这么火
7. 为什么说RISC-V给国内的操作系统圈子带来了机遇和挑战？
8. TARSIER观测计划介绍，以及参与进入开源社区的几种方式

交流形式

- 请务必将心中的疑问提出来，交流讨论。我们有1个小时的时间，甚至还可以拖
 - 今天的演讲者只是“门口的野蛮人”，没有顾忌。不代表任何群体。提供一个新颖的视角

快上车！

- RISC-V 已经成为未来的主流架构
- 任何芯片公司都应投入；
任何软件公司都需要适配；
同学们也应该看一看

真·芯片大佬 →

Divergence w/ Jim Keller in Bangalore
<https://www.youtube.com/watch?v=yHrdEcsr9V0>



长篇大论之前，先快速回答几个问题

- RISC-V 会被卡脖子么？
不会。因为 RISC-V 就是一纸规范，进入公共领域了。
- RISC-V 的软件支持现状如何？
总体已经很好，还有些地方需要补充（机会）
留给国内的机会还有，但是不多了，要抓紧。

RISC-V 的背景、现状、未来规划：RISC-V国际基金会视角

- 本节内容部分节选自

- Krste Asanovic 教授在 RISC-V Summit 2019 上题为《State of the Union》的主题报告、以及 RISC-V 基金会CEO Calista Redmond 的开场报告。根据本次讨论的需要，进行了部分删减。图源地址：

- <https://riscv.org/2019/12/risc-v-summit-2019-proceedings/>
- <https://content.riscv.org/wp-content/uploads/2019/12/12.10-9.20-StateOfUnion.pdf>
- <https://content.riscv.org/wp-content/uploads/2020/01/Calista-Summit-2019-RISC-V-Revolution-12-09-2019.pptx>

（为什么是2019年而不是2021年？因为自2019年开始持有的观点并未变化。）也是因为汇报人比较懒 ☺

- Mark Himmelstein 在 2022年5月5日的报告

- <https://open-src-soc.org/2022-05/media/slides/RISC-V-International-Day-2022-05-05-09h00-Mark-Himmelstein.pdf>

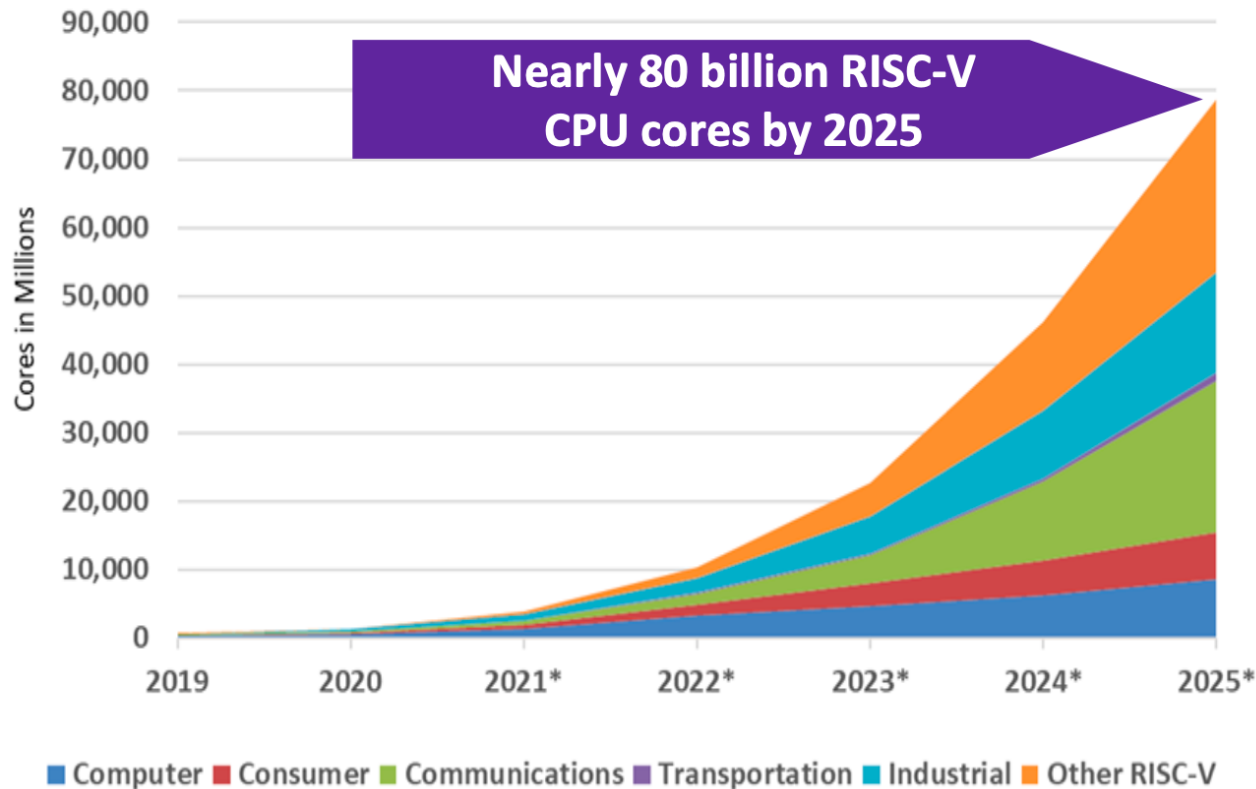
- Calista Redmond 在2022年5月5日的报告

- <https://open-src-soc.org/2022-05/media/slides/RISC-V-International-Day-2022-05-05-11h05-Calista-Redmond.pdf>

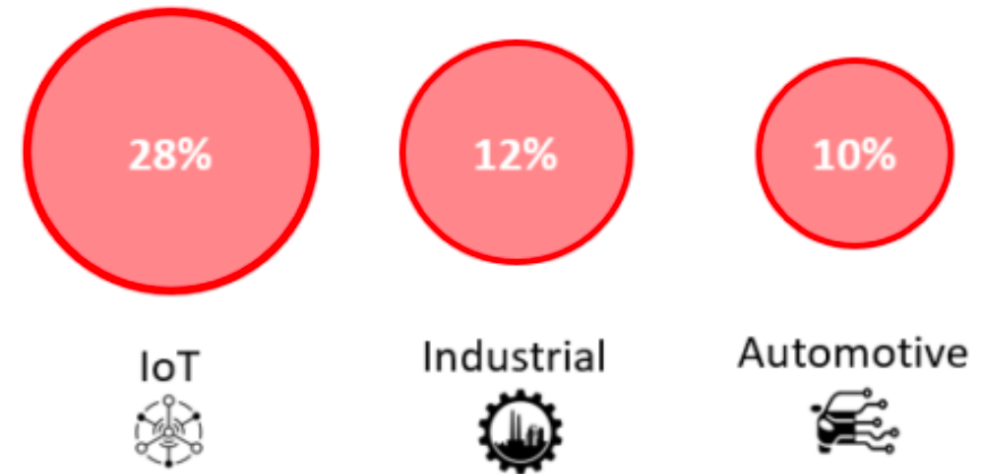
- Philipp 与 Mark 在2022年5月5日的报告

- <https://open-src-soc.org/2022-05/media/slides/RISC-V-International-Day-2022-05-05-09h50-Philipp-Tomsich-and-Mark-Himmelstein.pdf>

RISC-V CPU core market grows 114.9% CAGR, capturing >14% of all CPU cores by 2025



RISC-V Penetration Rate by 2025



“The rise of RISC-V cannot be ignored... RISC-V will shake up the \$8.6 Billion semiconductor IP market.”

-- William Li, Counterpoint Research

More than 12,000,000,000 RISC-V cores deployed for profit!

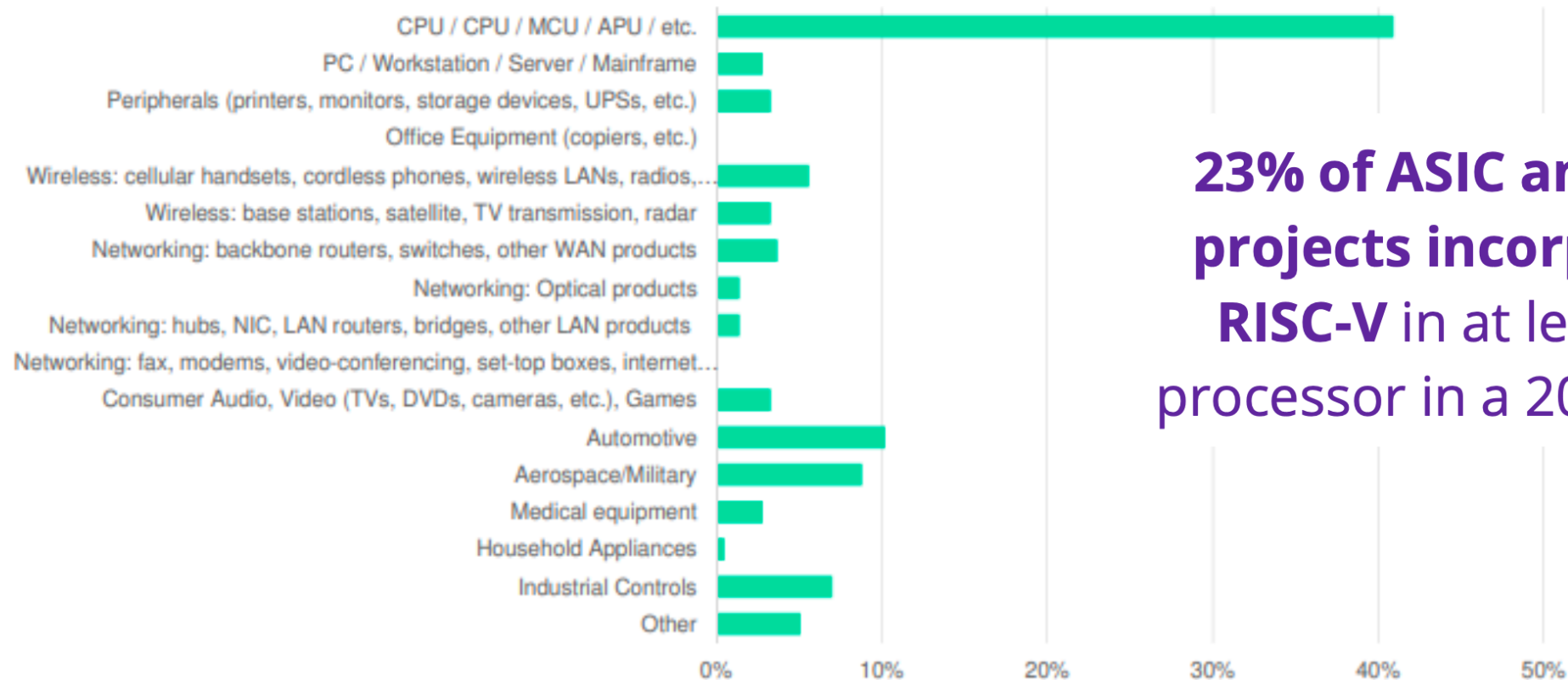


Source: Semico Research Corp, March 2021

Source: Counterpoint Research, September 2021

Nearly a quarter of designs incorporate RISC-V

Projects Incorporating RISC-V by Market Segment



23% of ASIC and FPGA projects incorporated RISC-V in at least one processor in a 2020 study.

为什么突然冒出来一个 RISC-V?

- 其实收钱的、开源自由不要钱的ISA，过去现在未来都有

为什么突然冒出来一个 RISC-V?

- 其实收钱的、开源自由不要钱的ISA，过去现在未来都有
 - OpenRISC、SPARC V32、以及 x86 的开源core都是有的，也默默流片不少

为什么突然冒出来一个 RISC-V?

- 其实收钱的、开源自由不要钱的ISA，过去现在未来都有
 - OpenRISC、SPARC V32、以及 x86 的开源core都是有的，也默默流片不少
- 国内也有不少指令集，例如阿里巴巴收购的中天微，就维护着 C-SKY 指令集

为什么突然冒出来一个 RISC-V?

- 其实收钱的、开源自由不要钱的ISA，过去现在未来都有
 - OpenRISC、SPARC V32、以及 x86 的开源core都是有的，也默默流片不少
- 国内也有不少指令集，例如阿里巴巴收购的中天微，就维护着 C-SKY 指令集
 - 仅仅国内就有好几家（曾今和现在）有自己维护的指令集，同时所有ISA国内都有备胎

为什么突然冒出来一个 RISC-V?

- 其实收钱的、开源自由不要钱的ISA，过去现在未来都有
 - OpenRISC、SPARC V32、以及 x86 的开源core都是有的，也默默流片不少
- 国内也有不少指令集，例如阿里巴巴收购的中天微，就维护着 C-SKY 指令集
 - 仅仅国内就有好几家（曾今和现在）有自己维护的指令集，同时所有ISA国内都有备胎
- RISC-V 是众多新ISA脱颖而出的一个，「大家」都买账

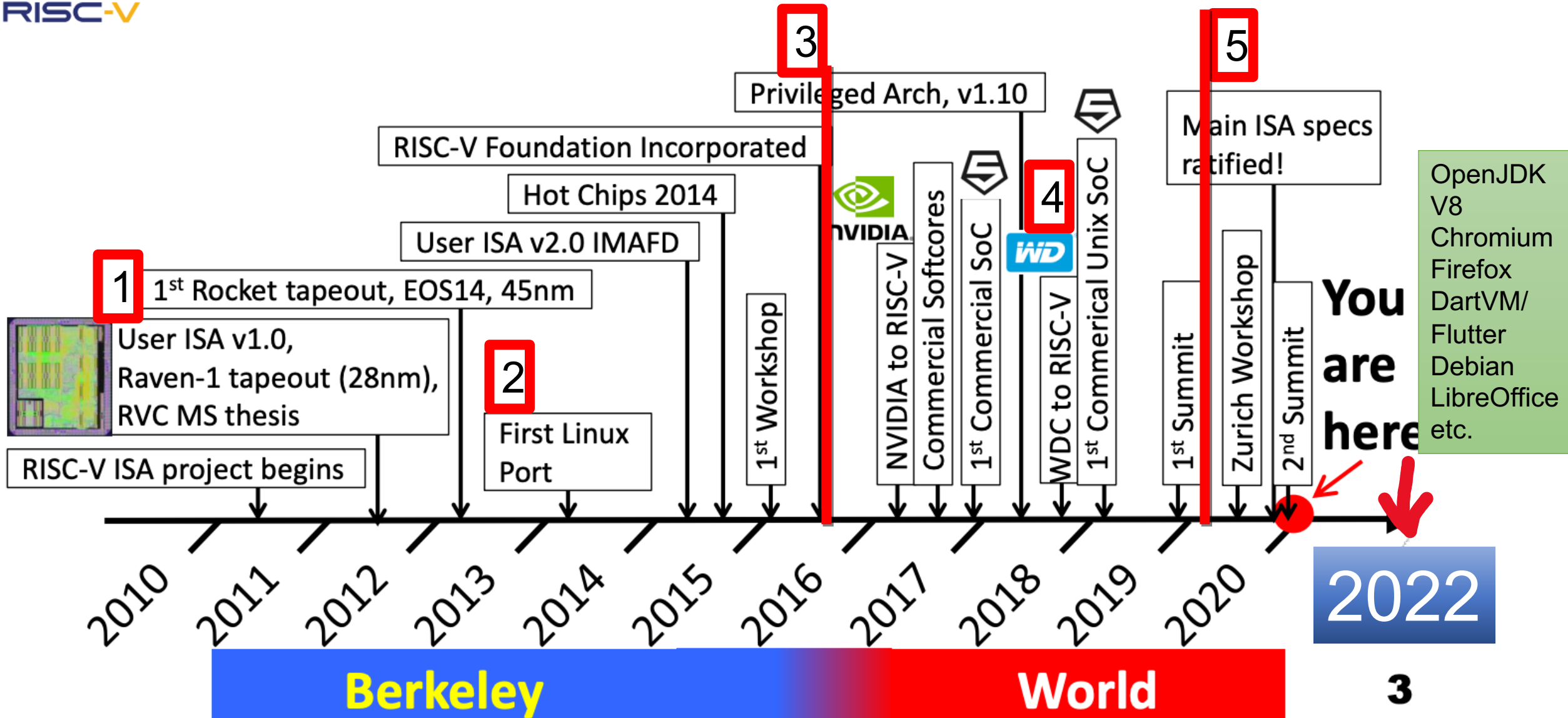
为什么突然冒出来一个 RISC-V?

- 其实收钱的、开源自由不要钱的ISA，过去现在未来都有
 - OpenRISC、SPARC V32、以及 x86 的开源core都是有的，也默默流片不少
- 国内也有不少指令集，例如阿里巴巴收购的中天微，就维护着 C-SKY 指令集
 - 仅仅国内就有好几家（曾今和现在）有自己维护的指令集，同时所有ISA国内都有备胎
- RISC-V 是众多新ISA脱颖而出的一个，「大家」都买账



思考：“大家”是谁？

RISC-V Timeline



Rich RISC-V Ecosystem Available Today

Training
Academia
Research

CI/Testing
Buttbot

Perf Tools
gprof
Sparta

Simulators
EMU
SAIL
Spike

Compilers
GCC
LLVM

HPC
Consumer
Data Center
IoT
Networking

Applications
SSH, Open vSwitch, ceph, Apache

Infrastructure
docker, redis, MySQL

Runtimes
Python, PHP, OpenJDK, DPDK, OpenSSL

Operating Systems
RT-Thread, FreeBSD, Zephyr, RTOS, debian

Hypervisor
KVM

Boot
OpenSBI, U-Boot

SAIL Formal Model



Architecture Te RISC-V COMPATIBLE

RTL DV
Implementation Design & Microarchitecture



Silicon Soft IP

Reliable, Serviceable, Diagnosable

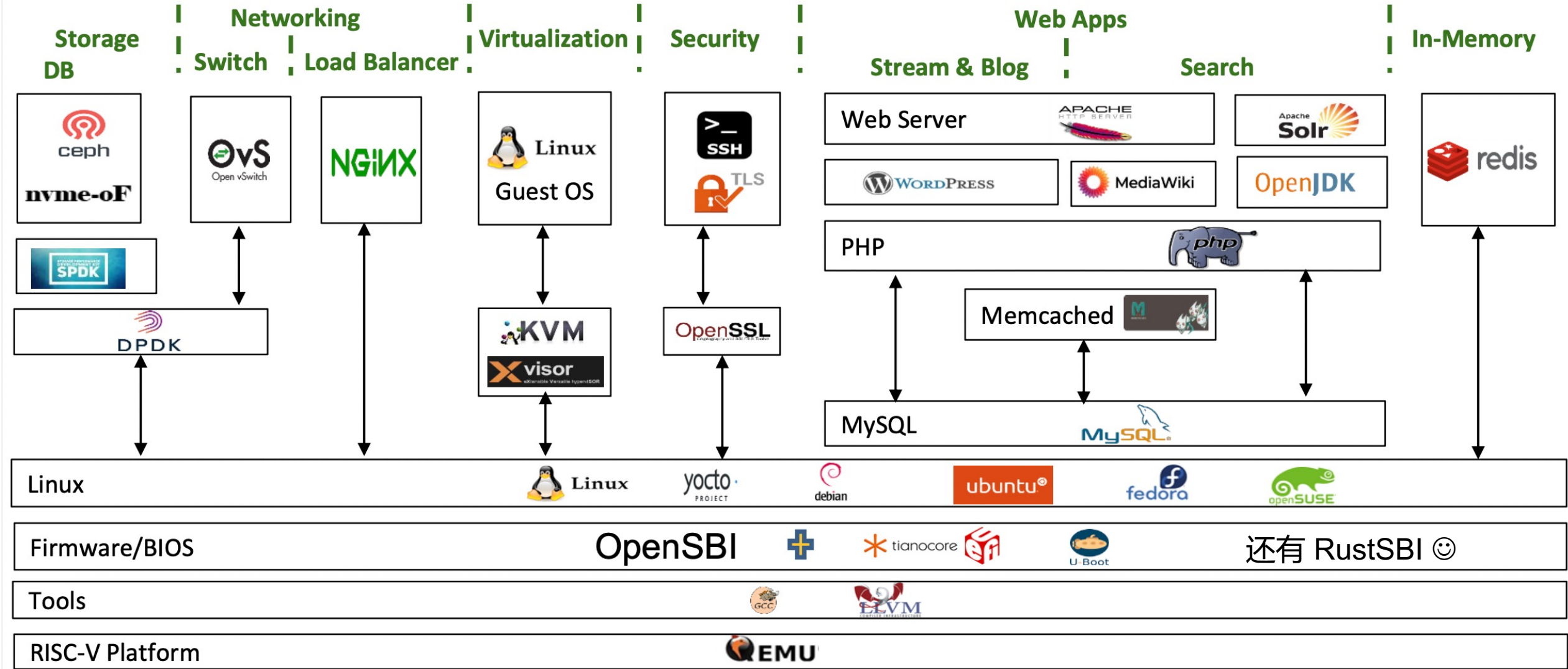
Performant

Secure

Debuggable

Services

Software Stack Examples



RISC-V 已经成为未来的三大主流架构之一

是因为 RISC-V 在**合适的时间**、合适的地点、提出了足够好的设计

- 20年前，摩尔定律开始失效，Domain-Specific Architecture 时代到来
- DSA 需要添加定制的指令、显著缩短的研发周期、大量的设计选择权衡
 - ❖ 上手容易、基础指令简单、有编译器和操作系统支持、预留了大量编码空间

RISC-V 已经成为未来的三大主流架构之一

是因为 RISC-V 在**合适的时间**、合适的地点、提出了足够好的设计

- 20年前，摩尔定律开始失效，Domain-Specific Architecture 时代到来
- DSA 需要添加定制的指令、显著缩短的研发周期、大量的设计选择权衡
 - ❖ 上手容易、基础指令简单、有编译器和操作系统支持、预留了大量编码空间

为什么在RISC-V之前那么多个ISA都没有成为主流？

RISC-V 已经成为未来的三大主流架构之一

是因为 RISC-V 在合适的时间、**合适的地点**、提出了足够好的设计

- 2010年在伯克利，Krstic 教授的研究组

- ❖ 需要个简单、免费、自由的ISA进行科研，作为更宏大的芯片设计创新项目的一部分
- ❖ 支持全新的ISA需要借助众多基础软件的开发者的力量，并进行有效地协作

RISC-V 已经成为未来的三大主流架构之一

是因为 RISC-V 在合适的时间、**合适的地点**、提出了足够好的设计

- 2010年在伯克利，Krstic 教授的研究组

- ❖ 需要个简单、免费、自由的ISA进行科研，作为更宏大的芯片设计创新项目的一部分
- ❖ 支持全新的ISA需要借助众多基础软件的开发者的力量，并进行有效地协作

如果 RISC-V 诞生在中国，我们还需要补上哪些技术人才？

RISC-V 已经成为未来的三大主流架构之一

是因为 RISC-V 在合适的时间、合适的地点、**提出了足够好的设计**

RISC-V 已经成为未来的三大主流架构之一

是因为 RISC-V 在合适的时间、合适的地点、**提出了足够好的设计**

- 提出了模块化设计的概念
 - ❖ 最基础的RV32I仅使用了40+条指令编码：一名大三学生一周就能完成一个能跑的设计
- 提供了高度灵活的配置空间

RISC-V 已经成为未来的三大主流架构之一

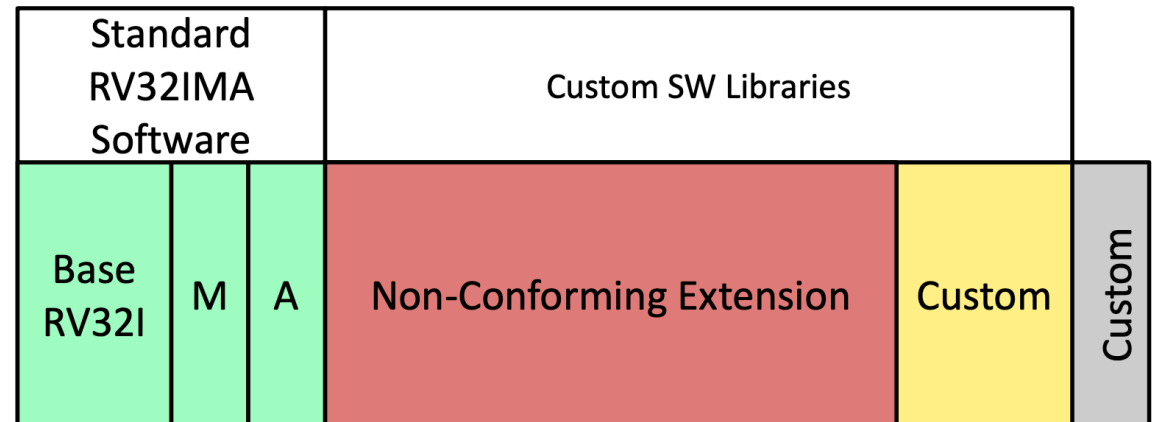
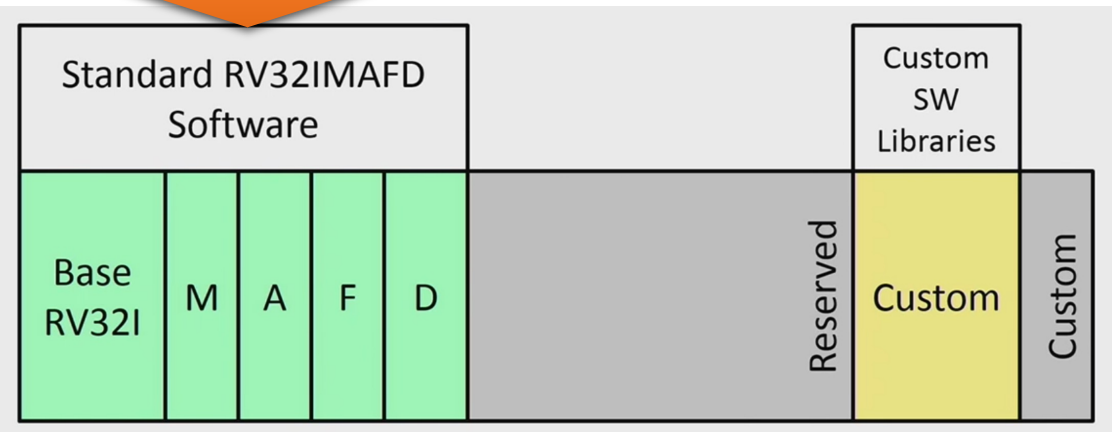
是因为 RISC-V 在合适的时间、合适的地点、**提出了足够好的设计**

- 提出了模块化设计的概念
 - ❖ 最基础的RV32I仅使用了40+条指令编码：一名大三学生一周就能完成一个能跑的设计
- 提供了高度灵活的配置空间
- 同时平台标准提供了足够多的**开源软件支持**

RISC-V 已经成为未来的三大主流架构之一

是因为 RISC-V 在合适的时间、合适的地点、**提出了足够好的设计**

- 提出了模块化设计的概念
 - ❖ 最基本的RV32I仅使用了40+条指令编码：一名大三学生一周就能完成一个能跑的设计
- 提供了高度灵活的配置空间
- 同时平台标准提供了足够多的**开源软件支持**



RISC-V 自身做对了什么？

RISC-V 自身做对了什么

- **清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口**

RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
 - ❖ 在 RISC-V 之前，主流ISA对硬件自由创新禁止，尤其是64位
 - ❖ 相比于 OpenRISC、SPARC 等架构，RISC-V 提供了更好的可扩展的设计

RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
 - ❖ 在 RISC-V 之前，主流ISA对硬件自由创新禁止，尤其是64位
 - ❖ 相比于 OpenRISC、SPARC 等架构，RISC-V 提供了更好的可扩展的设计

Software



ISA specification

Golden Model

Compliance

Hardware



Open Software/Standards Work!

<i>Field</i>	<i>Standard</i>	<i>Free, Open Impl.</i>	<i>Proprietary Impl.</i>
Networking	Ethernet, TCP/IP	Many	Many
OS	Posix	Linux, FreeBSD	M/S Windows
Compilers	C	gcc, LLVM	Intel icc, ARMcc
Databases	SQL	MySQL, PostgreSQL	Oracle 12C, M/S DB2
Graphics	OpenGL	Mesa3D	M/S DirectX
ISA	??????	-----	x86, ARM, IBM360

- Why not successful free & open standards and free & open implementations, like other fields
- Dominant proprietary ISAs are not great designs

3



“Instruction Sets Want to be Free” Krste Asanovic, Professor of UCB

RISC-V 自身做对了什么

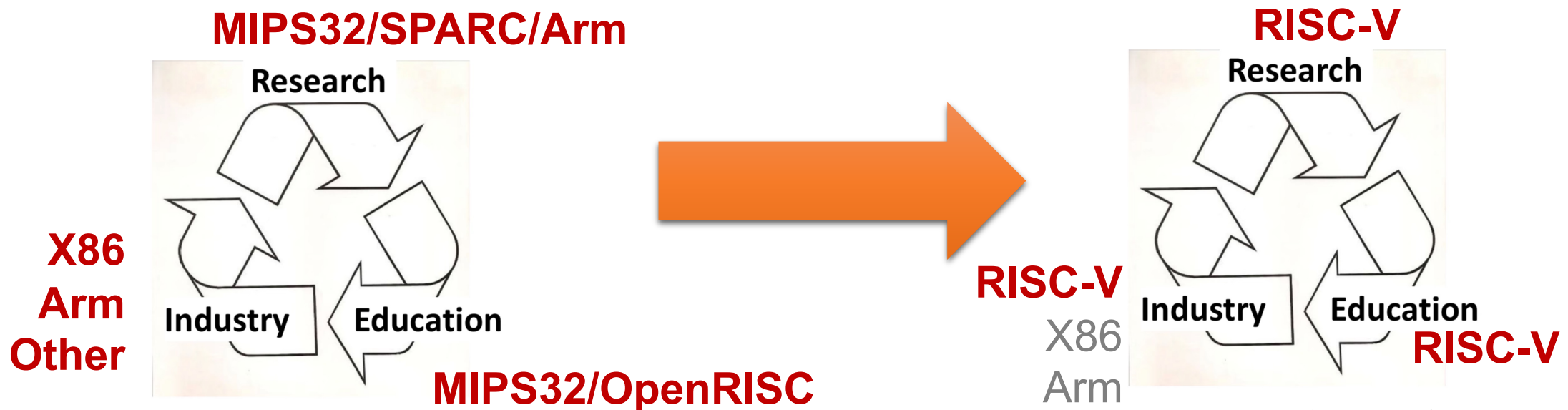
- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
- **设计优美简洁，支持了从教学、研究到工业界的无缝切换**

RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
- 设计优美简洁，支持了从教学、研究到工业界的无缝切换
 - ❖ 前人成功的案例，有MIPS、Linux和LLVM
 - ❖ 在这之前，教育用MIPS/SPARC，进入产业之后变成X86/Arm，研究成果无法直接转化

RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
- 设计优美简洁，支持了从教学、研究到工业界的无缝切换
 - ❖ 前人成功的案例，有MIPS、Linux和LLVM
 - ❖ 在这之前，教育用MIPS/SPARC，进入产业之后变成X86/Arm，研究成果无法直接转化





RISC-V Ecosystem

Software

Open-source software:

Gcc, binutils, glibc, Linux, BSD, LLVM, QEMU, FreeRTOS, ZephyrOS, LiteOS, SylixOS, ...

Commercial software:

Lauterbach, Segger, IAR, Micrium, ExpressLogic, Ashling, AntMicro, Imperas, UltraSoC ...



ISA specification

Golden Model

Compliance

Hardware

Open-source cores:

Rocket, BOOM, RI5CY, Ariane, PicoRV32, Piccolo, SCR1, Shakti, Serv, Swerv, Hummingbird, ...

Commercial core providers:

Alibaba, Andes, Bluespec, Cloudbear, CodaSip, Cortus, InCore, Nuclei, SiFive, Syntacore, ...

Inhouse cores:

Nvidia, WD, +others



RISC-V Ecosystem

Open-source software:

Gcc, binutils, glibc, Linux, BSD,
LLVM, QEMU, FreeRTOS,
ZephyrOS, LiteOS, SylixOS, ...

Commercial software:

Lauterbach, Segger, IAR,
Micrium, ExpressLogic, Ashling,
AntMicro, Imperas, UltraSoC ...

Software



ISA specification

Golden Model

Compliance

Hardware

Open-source cores:

Rocket, BOOM, RI5CY,
Ariane, PicoRV32, Piccolo,
SCR1, Shakti, Serv, Swerv,
Hummingbird, ...

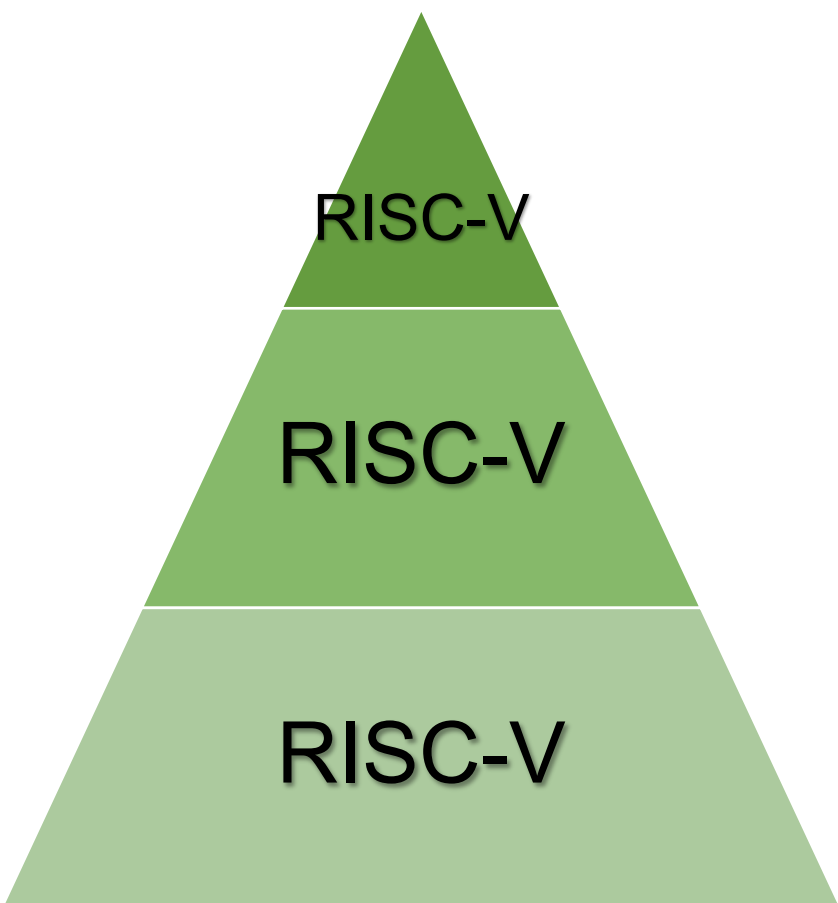
Commercial core providers:

Alibaba, Andes, Bluespec,
Cloudbear, CodaSip, Cortus,
InCore, Nuclei, SiFive,
Syntacore, ...

Inhouse cores:

Nvidia, WD, +others

内卷警告：所有还在做自研的指令集的朋友，该思考下出路了



RISC-V Ecosystem

Open-source software:

Gcc, binutils, glibc, Linux, BSD, LLVM, QEMU, FreeRTOS, ZephyrOS, LiteOS, SylixOS, ...

Commercial software:

Lauterbach, Segger, IAR, Micrium, ExpressLogic, Ashling, AntMicro, Imperas, UltraSoC ...

Software



ISA specification

Golden Model

Compliance

Hardware

Open-source cores:

Rocket, BOOM, RI5CY, Ariane, PicoRV32, Piccolo, SCR1, Shakti, Serv, Swerv, Hummingbird, ...

Commercial core providers:

Alibaba, Andes, Bluespec, Cloudbear, Cudasip, Cortus, InCore, Nuclei, SiFive, Syntacore, ...

Inhouse cores:

Nvidia, WD, +others

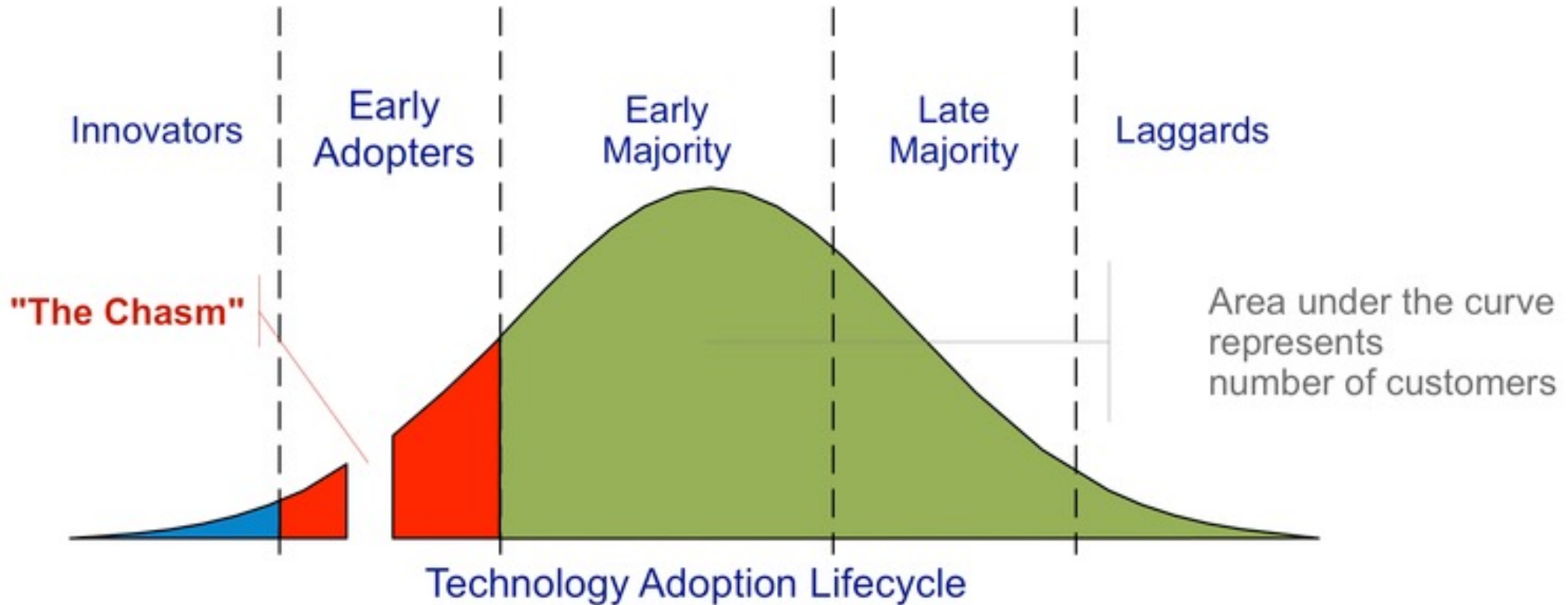
RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
- 设计优美简洁，支持了从教学、研究到工业界的无缝切换
- **左手成立RISC-V基金会，右手创办SiFive**

RISC-V 自身做对了什么

- 左手成立RISC-V基金会，右手创办SiFive

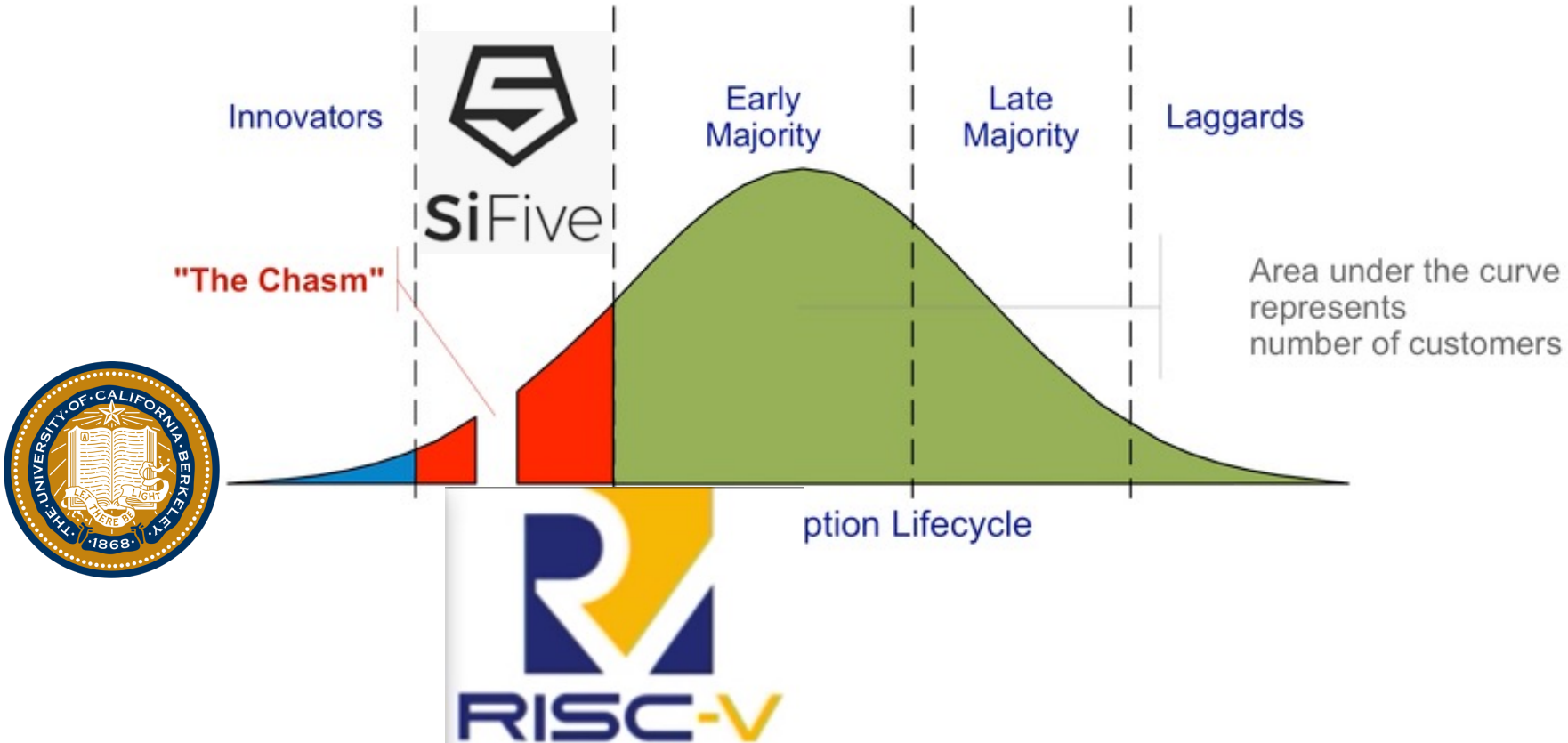
❖ 兼顾ISA中立和推动商业化



RISC-V 自身做对了什么

- 左手成立RISC-V基金会，右手创办SiFive

❖ 兼顾ISA中立和推动商业化



RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
- 设计优美简洁，支持了从教学、研究到工业界的无缝切换
- 左手成立RISC-V基金会，右手创办SiFive
- 顺应了时代的潮流：DSA导致的设计方式的改变，引发对ISA的新需求
 - ❖ “RISC-V 的成功是因为（出现了）新的商业模式” —— Krste 教授

RISC-V将会2025年左右跻身三大指令集架构行列

- 随着摩尔定律放缓/终结，新的技术和**商业模式**开始出现：

做新产品，先选择RISC-V指令集，再挑选供货商

那么，为何是伯克利搞成了RISC-V，而不是中国的高校或企业？

RISC-V将会2025年左右跻身三大指令集架构行列

- 随着摩尔定律放缓/终结，新的技术和商业模式开始出现： 例如AI等新的业务需求的出现

做新产品，先选择RISC-V指令集，再挑选供货商

- 软件系统的规模内在的复杂度在十倍十倍地提高，这导致
使用（和回馈）开源成为竞争博弈中**唯一的**致胜选择，从而

开发者的稀缺，使得
招募足够的开发者
另起炉灶变得不可能

开源软件已经成为**高度垄断**的全人类的技术公共品，并且

最终会促进开源自由指令集规范的出现并进入公共领域

即使不是RISC-V，也会有
别的开源自由指令集出现

拥抱开源软件

- 成功的ISA架构在开源软件领域都有着巨大的投入
 - ❖ Intel、Arm 有着长期巨大的投入，投入不够一定会被打败
 - ❖ 就像是武林高手对掌拼内力，谁先撤力谁出大问题

拥抱开源软件

- 成功的ISA架构在开源软件领域都有着巨大的投入
 - ❖ Intel、Arm 有着长期巨大的投入，投入不够一定会被打败
- **RISC-V 做对了两件事让其成为未来的主流**
 - ❖ **早期的团队研发实力和伯克利的领袖光环，实现了基本工具链支持**
 - ❖ **在2017年之后顺应产业界的需求从芯片创新转向以软件为中心**

RISC-V 正确的转向（接下来灯光给到软件和操作系统社区 😊）



Changing Priorities



2010

1. Be simple, efficient, extensible
2. Revisit legacy design decisions
3. Have basic software

(computer-architecture-driven project)

2020

1. Run all software
 2. Be feature complete
 - see #1
 3. Be stable
 - see #1
 4. Support innovation
 - conflicts with #1,#2,#3?
- (software-driven project)***

本次会议交流的内容和形式

内容简介

1. 演讲人的自我介绍（强调自己是门外汉并且并不是博士）
2. 芯片和软件技术发展的几个观测判断（也就是假设前提）
3. 操作系统（尤其是Linux发行版）的基本的要素
4. 为什么会有“国内操作系统领域”这个奇怪的说法
5. 为什么软件的验收和市场筛选怎么难？
 1. （这个问题很深，我也不确定自己是不是把握住了）
6. RISC-V 是什么，为什么这么火
7. 为什么说RISC-V给国内的操作系统圈子带来了机遇和挑战？
8. TARSIER观测计划介绍，以及参与进入开源社区的几种方式

交流形式

- 请务必将心中的疑问提出来，交流讨论。我们有1个小时的时间，甚至还可以拖
 - 今天的演讲者只是“门口的野蛮人”，没有顾忌。不代表任何群体。提供一个新颖的视角

因为国外也没做，国内没得抄（不是）

- 在RISC-V这个新架构的细分赛道上，起跑线基本上一样的
- 只要投入足够多的时间，足够强的工程师，足够不折腾的管理
- 国内团队在RISC-V细分领域领先的方面：
 - AOSP for RISC-V 的移植和适配（阿里巴巴平头哥、中科院软件所）
 - OpenJDK RISC-V（华为、阿里云、中科院软件所）
 - Google V8 引擎的 RISC-V 后端代码维护（中科院软件所）
 - LibreOffice 在 RISC-V 上的移植和维护（中科院软件所）
 - QEMU中多项RISC-V架构的支持（风河中国、平头哥、中科院软件所）
 - gem5 的 RISC-V Vector 扩展支持（中科院软件所）
 - **【请将您的贡献追加在这里】**

本次会议交流的内容和形式

内容简介

1. 演讲人的自我介绍（强调自己是门外汉并且并不是博士）
2. 芯片和软件技术发展的几个观测判断（也就是假设前提）
3. 操作系统（尤其是Linux发行版）的基本的要素
4. 为什么会有“国内操作系统领域”这个奇怪的说法
5. 为什么软件的验收和市场筛选怎么难？
 1. （这个问题很深，我也不确定自己是不是把握住了）
6. RISC-V 是什么，为什么这么火
7. 为什么说RISC-V给国内的操作系统圈子带来了机遇和挑战？
8. TARSIER观测计划介绍，以及参与进入开源社区的几种方式

交流形式

- 请务必将心中的疑问提出来，交流讨论。我们有1个小时的时间，甚至还可以拖
 - 今天的演讲者只是“门口的野蛮人”，没有顾忌。不代表任何群体。提供一个新颖的视角

只要你有兴趣参与到操作系统开源社区

- 在社区论坛或者聊天系统中回答其他用户的问题
- 代码提交、修复缺陷
- 撰写补充文档
- 制作教程，视频或技术文章
- 进行测试验证和bug分类
- 翻译，中文、英语、俄语、日语、韩语
- 组织线下或线上的技术聚会
- 作为开源社区运营（实习生）经理
- 作为开源社区媒体运营，撰写公众号文章等

只要你有兴趣参与到操作系统开源社区

- 想去哪个就直接参与进去，给社区邮件列表或者运营者发邮件
 - 如果英语好，或者希望练好英语，直接进 Debian、Arch、Gentoo
 - 如果希望先在中文环境，openEuler、龙蜥、Fedora、等
- 通过PLCT实验室和TARSIER团队的实习生计划
 - <https://github.com/lazyparser/weloveinterns/blob/master/open-internships.md>

非常感谢大家在周末参加！

Ask Me Anything

任何问题都欢迎提出来交流讨论

RISC-V 给国内操作系统领域 带来的机遇和挑战

汇报人： 吴伟

PLCT实验室、TARSIER团队

wuwei2016@iscas.ac.cn | WeChat fangzhang1024 | GitHub/B站 lazyparser