

# RISC-V指令集开源软件生态

报告人：吴伟

[wuwei2016@iscas.ac.cn](mailto:wuwei2016@iscas.ac.cn) | [@lazyparser](#) on Github/bilibili

PLCT Lab, ISRC, ISCAS

# 自我介绍

- 软件出身，年少苦练屠龙技；如今天天写PPT；至今未写过 RISC-V SoC
  - 2009年加入HelloGCC社区，后成为OSDT/HelloGCC/HelloLLVM 社区负责人
- 2016年开始关注到RISC-V，开始学习、赞助、组织沙龙和培训
- 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
  - 2022年组建TARSIER团队，成规模的完善和维护RISC-V开源软件
- RISC-V国际基金会中国区联络人、RISC-V大使、RISC-V技术委员会成员

请注意：本次报告仅代表我个人及 PLCT实验室观点，不代表ISRC/ISCAS或其他第三方

# 今天演讲的主题内容（Takeaway）

- 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；从业者应该立刻开始学习

# 今天演讲的主题内容（Takeaway）

- 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；从业者应该立刻开始学习
- 拥抱开源软件并积极成为贡献者是企业生存、胜出的必要条件
  - 要么积极融入开源社区，要么被开源软件吞噬；不被upstream支持等于活在ICU

# 今天演讲的主题内容（Takeaway）

- 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；从业者应该立刻开始学习
- 拥抱开源软件并积极成为贡献者是企业生存、胜出的必要条件
  - 要么积极融入开源社区，要么被开源软件吞噬；不被upstream支持等于活在ICU
- RISC-V 软件生态领域依然有大量机会可以把握
  - 以中国科学院软件研究所PLCT实验室在RISC-V领域的贡献和成果为例

# 今天演讲的主题内容（Takeaway）

三个基本观测（假设）：

- 摩尔定律是不可持续的。
- （软件和硬件）系统的复杂度是超线性增长的。
- 有能力驾驭复杂度的开发人员是有限的。

# 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>

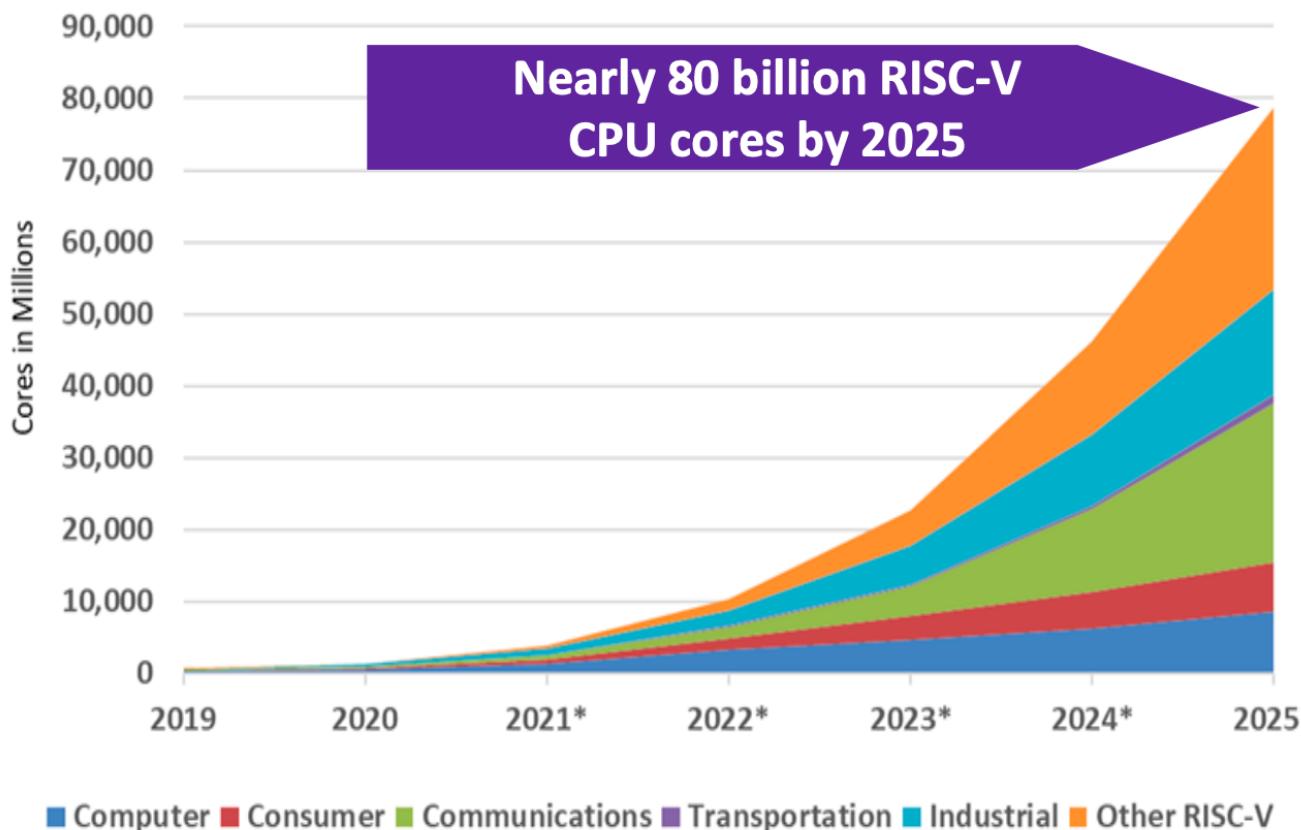
- Calista Redmond 在2022年5月5日的报告

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

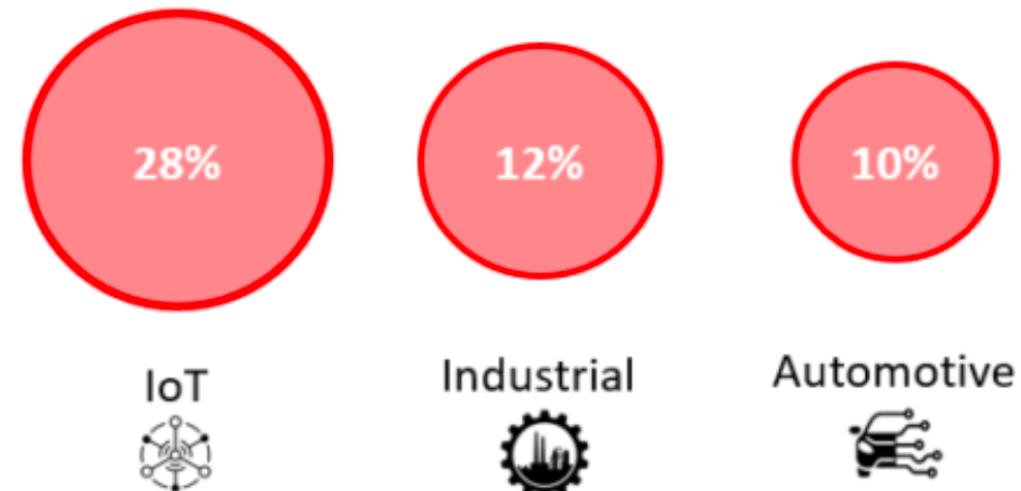
- Philipp 与 Mark 在2022年5月5日的报告

- <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



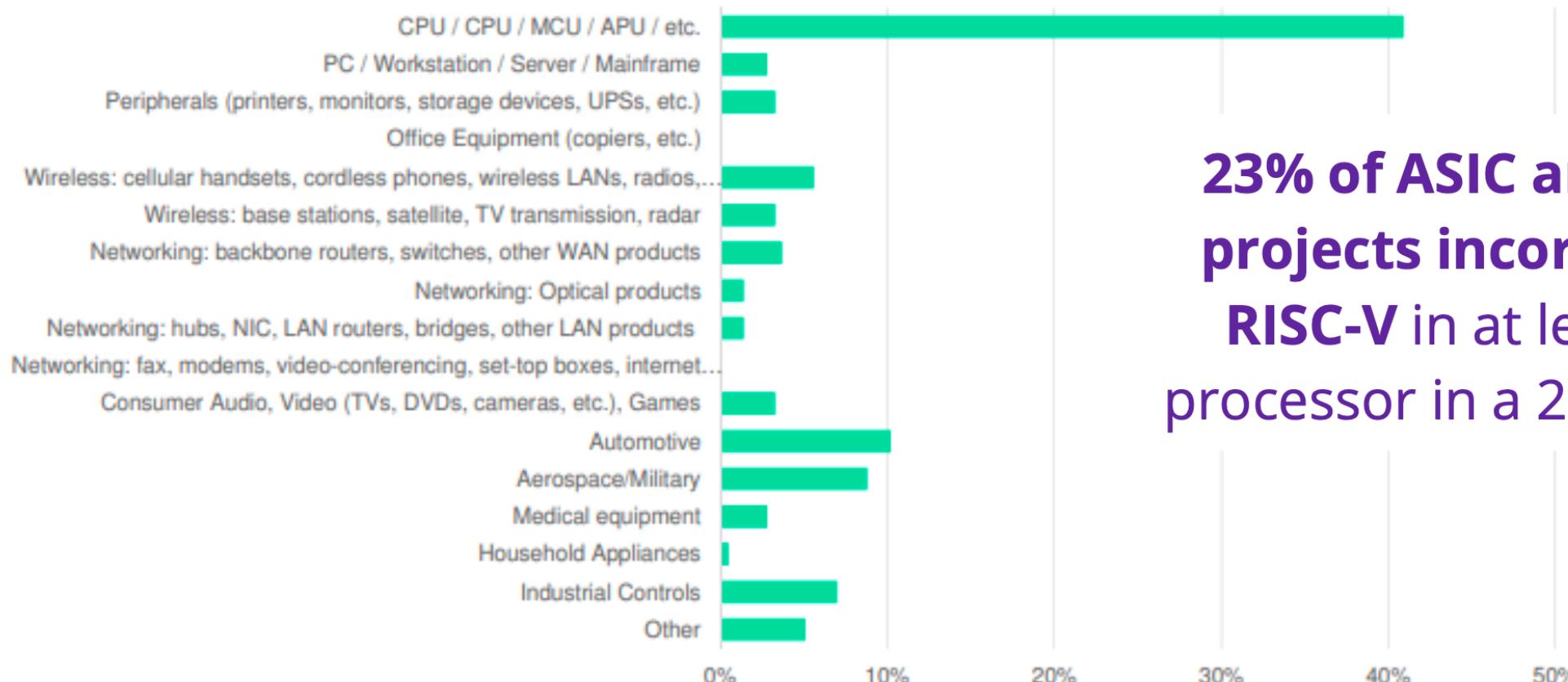
"The rise of RISC-V cannot be ignored... RISC-V will shake up the \$8.6 Billion semiconductor IP market."

-- William Li, Counterpoint Research

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

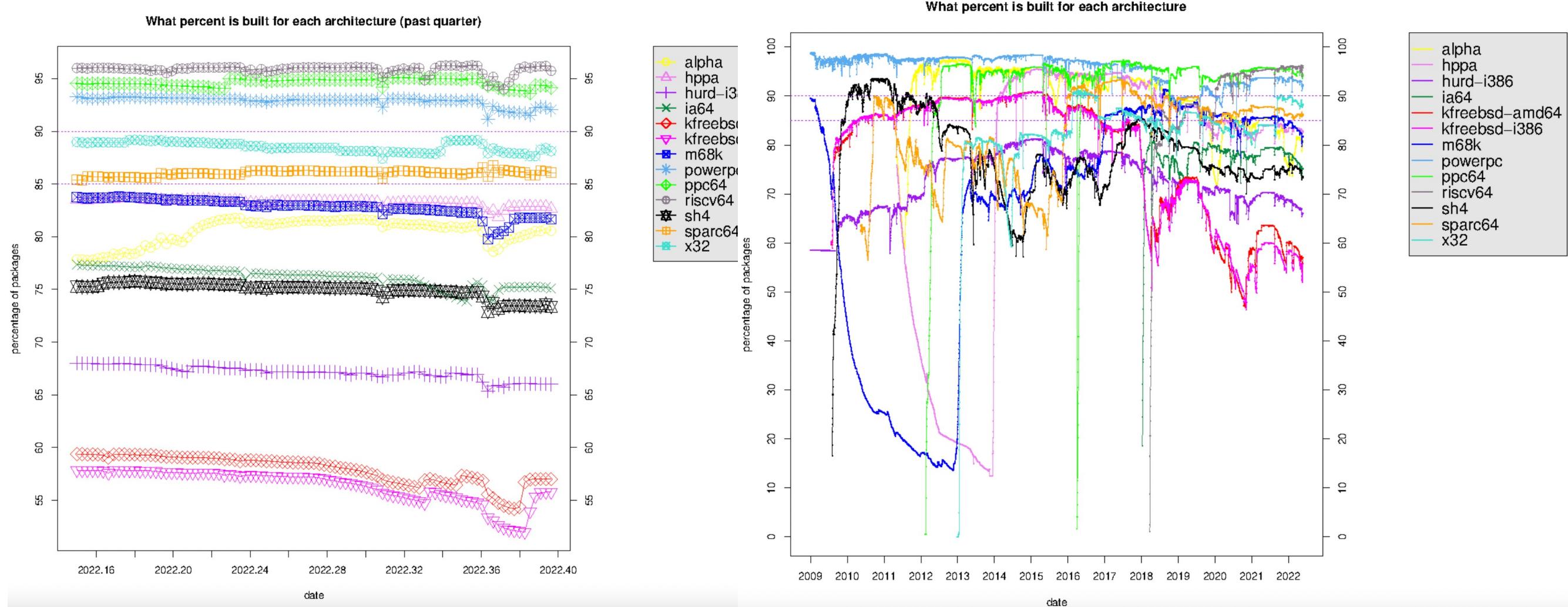
# Nearly a quarter of designs incorporate RISC-V

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.**

# RISC-V开源软件生态：地基已经打好，整体框架也浇铸好，还有许多软装可以做



<https://buildd.debian.org/stats/graph-ports-quarter-big.png>

# RISC-V开源软件生态：地基已经打好，整体框架也浇铸好，还有许多软装可以做

## PLCT Roadmap 2022

大家好，这是 PLCT Lab 第三年公开实验室在 RISC-V 开源生态系统中的 Roadmap。在 2022 年我们将会继续加大在工具链、虚拟机、模拟器上对 RISC-V 的支持力度。在 GCC、LLVM、V8、QEMU 等社区的投入会继续加大。同时我们很高兴地宣布，Tarsier Team 完成了孵化，将会开始作为 ISRC 的一个独立的操作系统团队进行运作。

### 关键目标

在 2022 年 PLCT 实验室希望能够达到以下关键目标：

1. 准备好 RISC-V 笔记本电脑需要的软件系统，使得流行的 Linux 发行版和常用的 OA 软件、开发工具可以流畅运行在 RISC-V 笔记本上。
2. 完成一半以上「最后 5%」的常用开源软件在 RISC-V 上的移植和适配工作。
3. 通过 Tarsier Project 组建一支 RISC-V 软件测试队伍，为 RISC-V 生态中开源软件的稳定性和软件质量提供保证。

### 主体工程： Tarsier Project

中科院软件所（ISCAS）智能软件研究中心（ISRC）在 2021 年开始孵化 Tarsier Project。Tarsier Project 的目标是促进主流 Linux 发行版（包括 Debian/Ubuntu、Fedora、Arch Linux、Gentoo、openEuler 等）对于 RISC-V 平台的支持，达到或超过 AArch64 平台的支持程度。具体分成了以下几个不同的方面：

#### Tarsier Team

眼镜猴团队（Tarsier Team）成立于 2021 年末，承担着软件所 Tarsier Project 所需的各项具体工作。2022 年的眼镜猴团队将会从开源操作系统的集成和测试切入，组建超过 300 名全职、兼职、实习生和志愿者组成的测试小队，为 RISC-V 软件生态系统提供更为全面的质量保证和现状摸底。

<https://github.com/plctlab/PLCT-Weekly/blob/master/PLCT-Roadmap-2022.md>

# 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

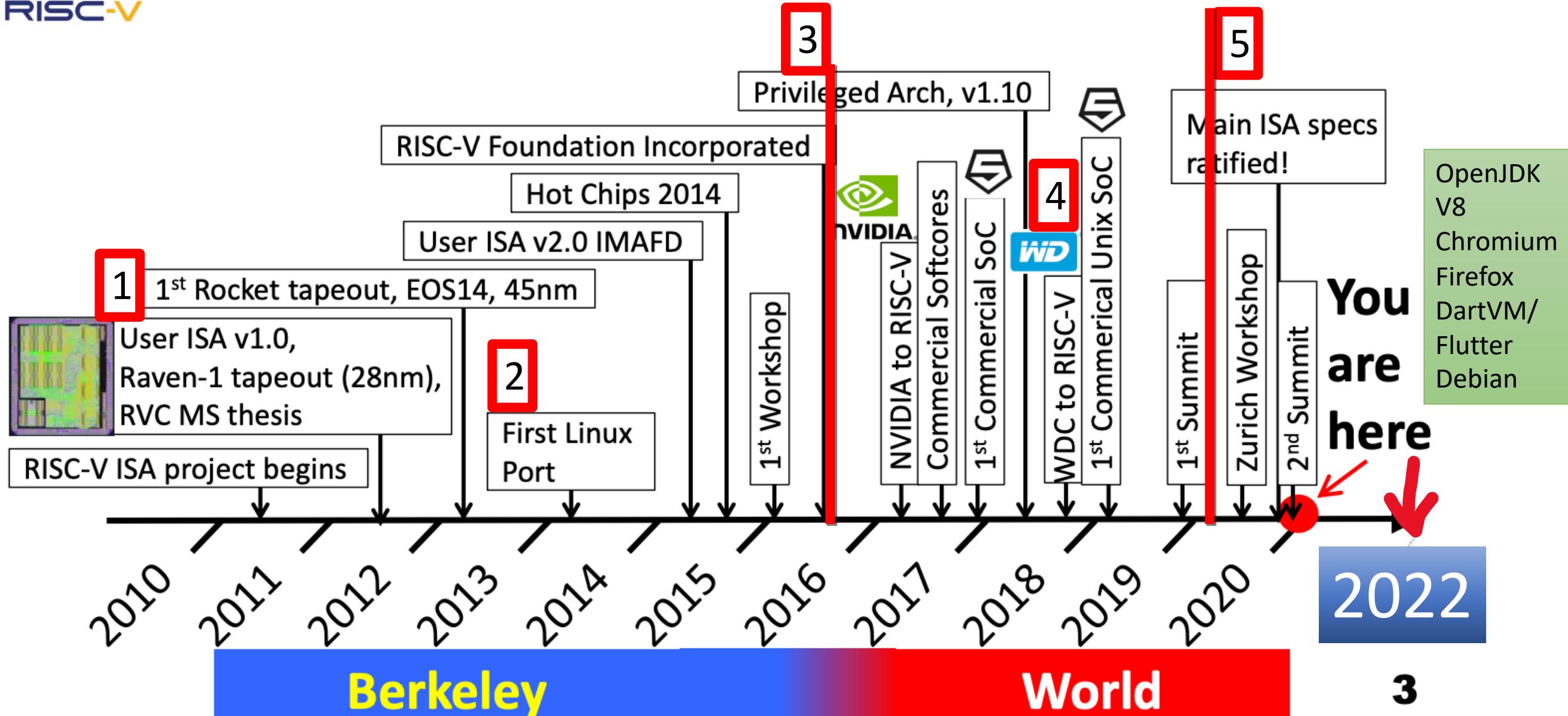
# 为什么突然冒出来一个 RISC-V？

- 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？

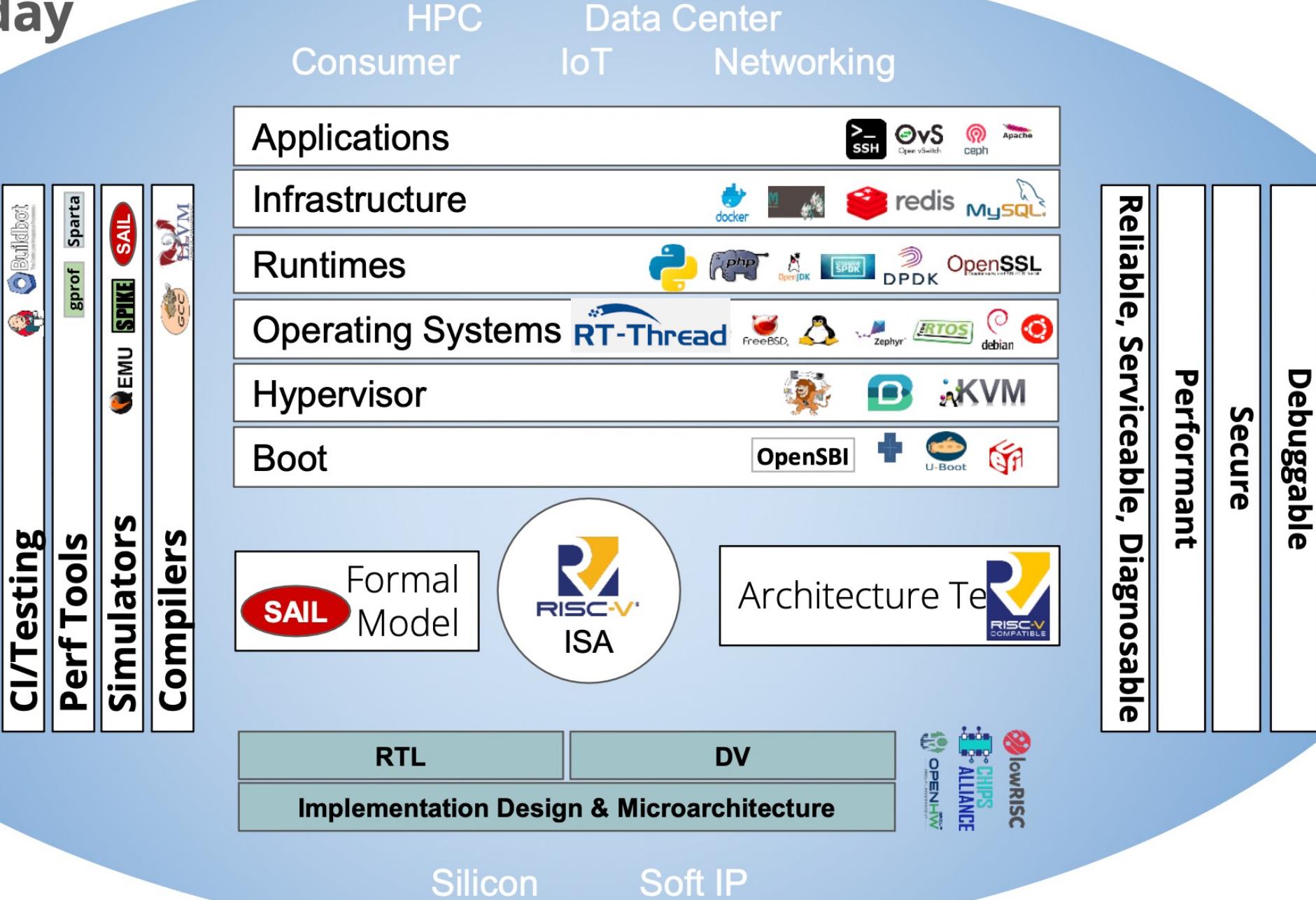


# RISC-V Timeline

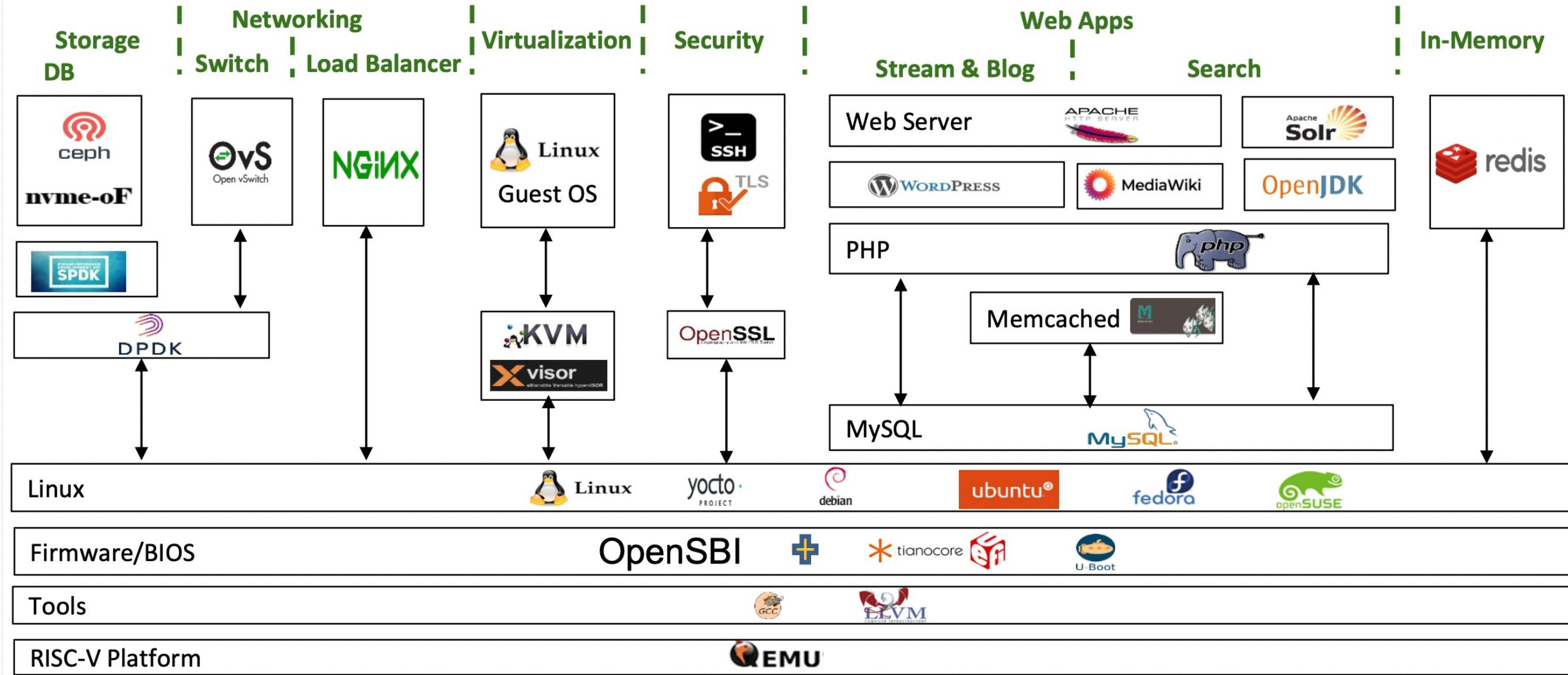


# Rich RISC-V Ecosystem Available Today

Training  
Research  
Academia



# 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，Krsti 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

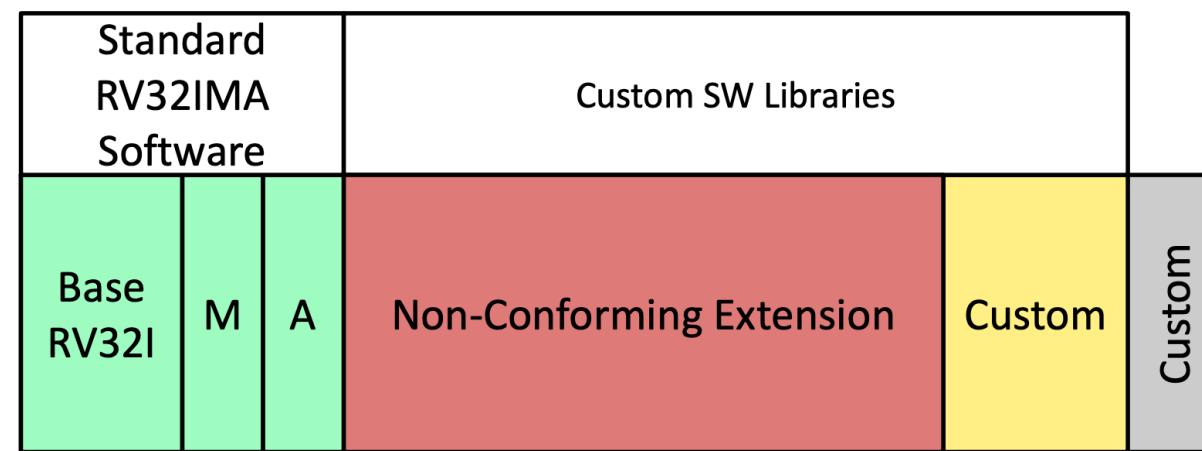
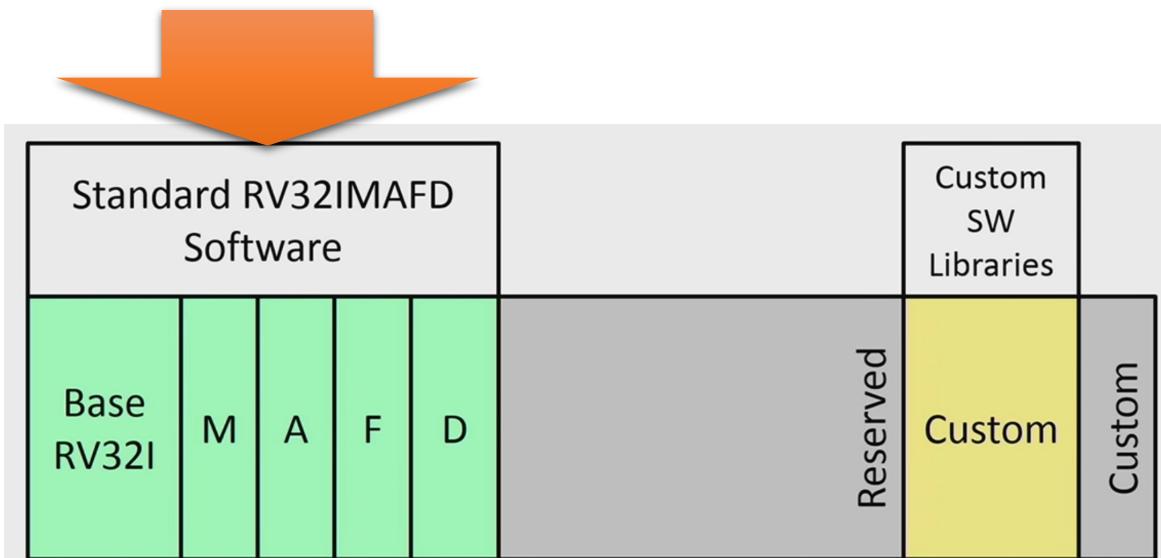
是因为 RISC-V 在合适的时间、合适的地点、提出了足够好的设计

- 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的软件支持

# 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的软件支持



# 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

# RISC-V 自身做对了什么

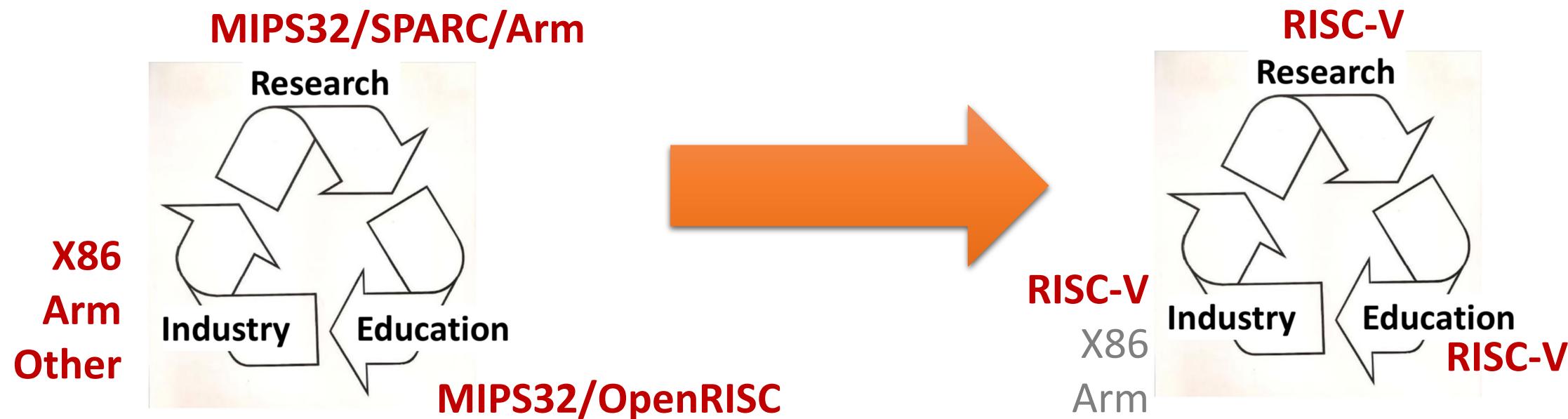
- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
- **设计优美简洁，支持了从教学、研究到工业界的无缝切换**

# 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



图源：Krstic 演讲：RISC-V State of Union



## Software



## Hardware

**Open-source cores:**  
Rocket, BOOM, RI5CY,  
Ariane, PicoRV32, Piccolo,  
SCR1, Shakti, Serv, Swerv,  
Hummingbird, ...

**Open-source software:**  
GCC, binutils, glibc, Linux, BSD,  
LLVM, QEMU, FreeRTOS,  
ZephyrOS, LiteOS, SylixOS, ...

# RISC-V Ecosystem

**Commercial software:**  
Lauterbach, Segger, IAR,  
Micrium, ExpressLogic, Ashling,  
AntMicro, Imperas, UltraSoC ...

ISA specification

Golden Model

Compliance

**Commercial core providers:**  
Alibaba, Andes, Bluespec,  
Cloudbear, Codasip, Cortus,  
InCore, Nuclei, SiFive,  
Syntacore, ...

**Inhouse cores:**  
Nvidia, WD, +others



Software



Hardware

## 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 ...

ISA specification

Golden Model

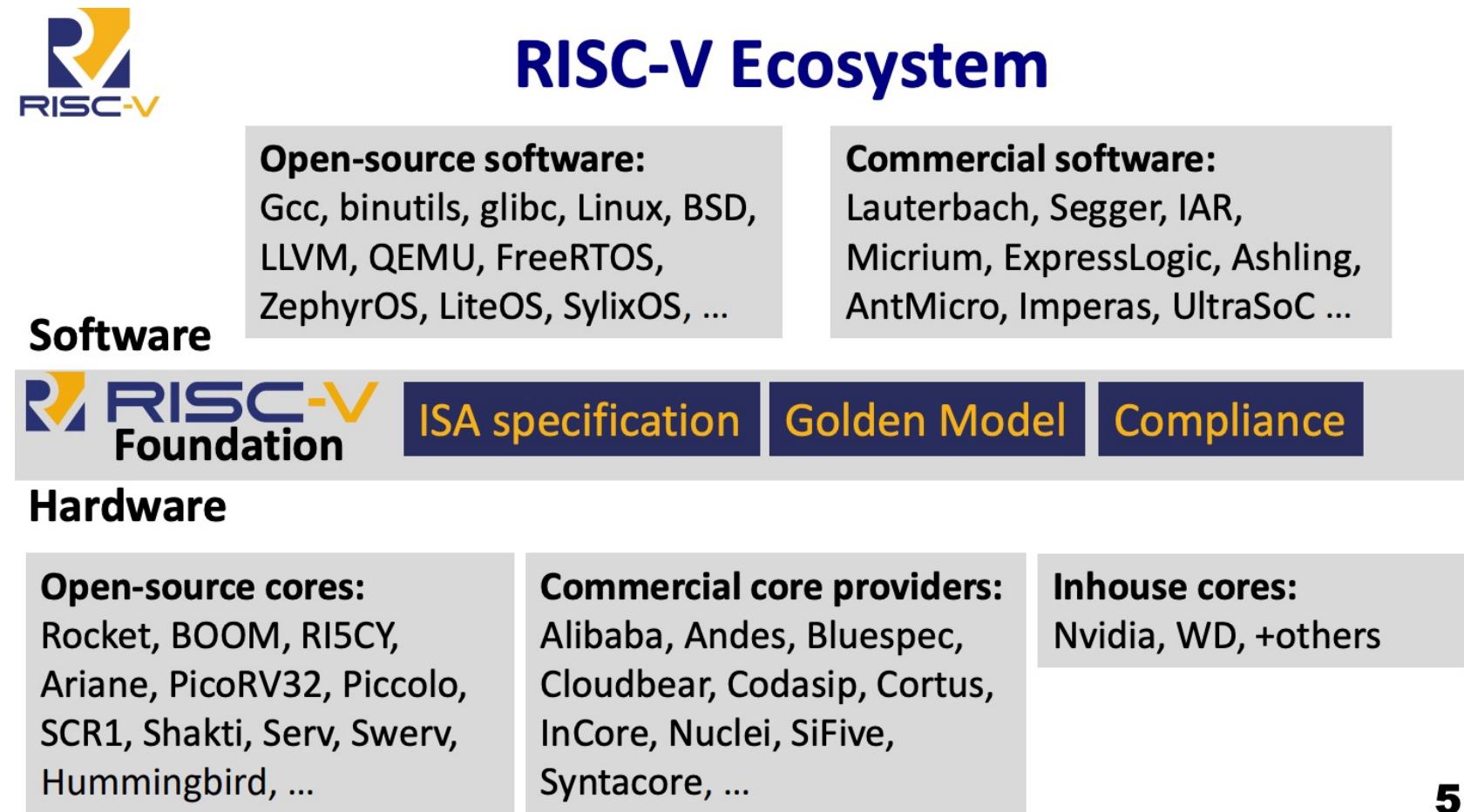
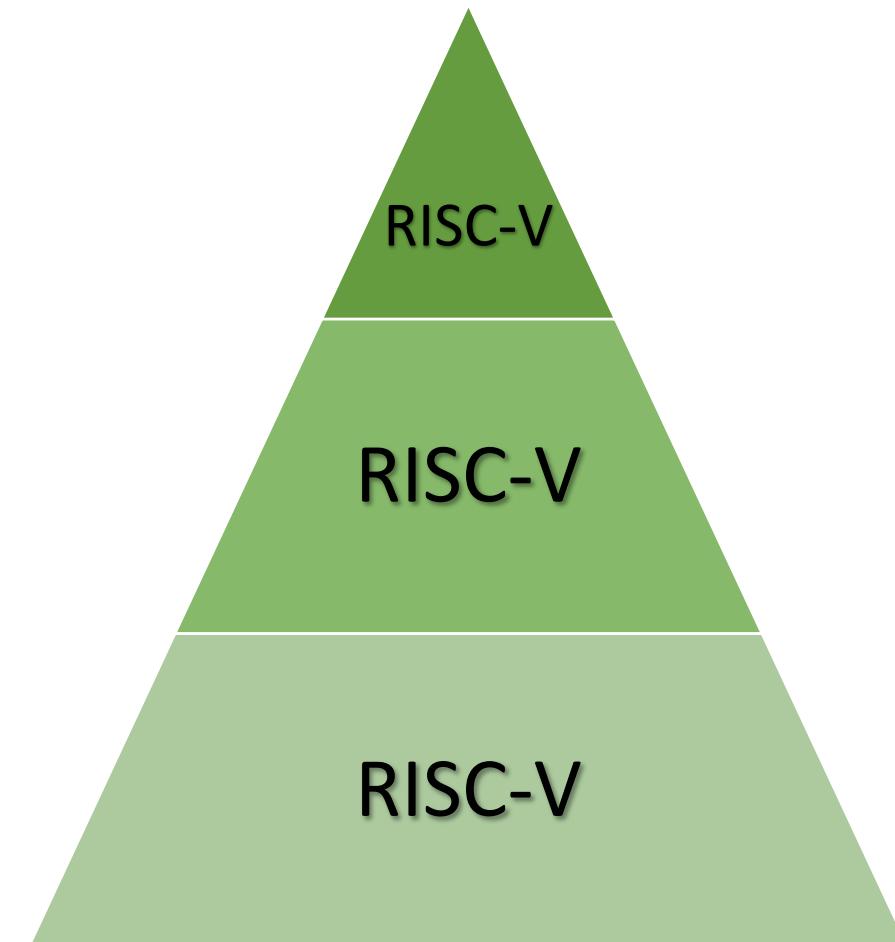
Compliance

**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



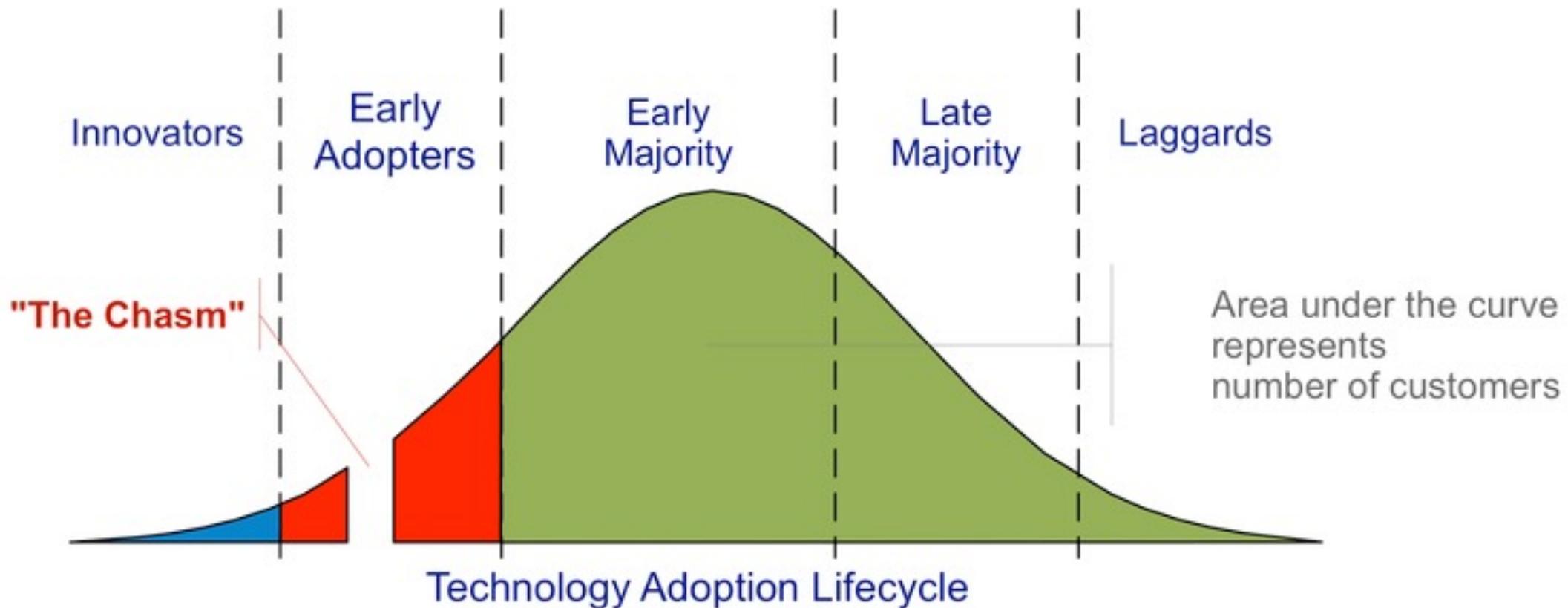
# RISC-V 自身做对了什么

- 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

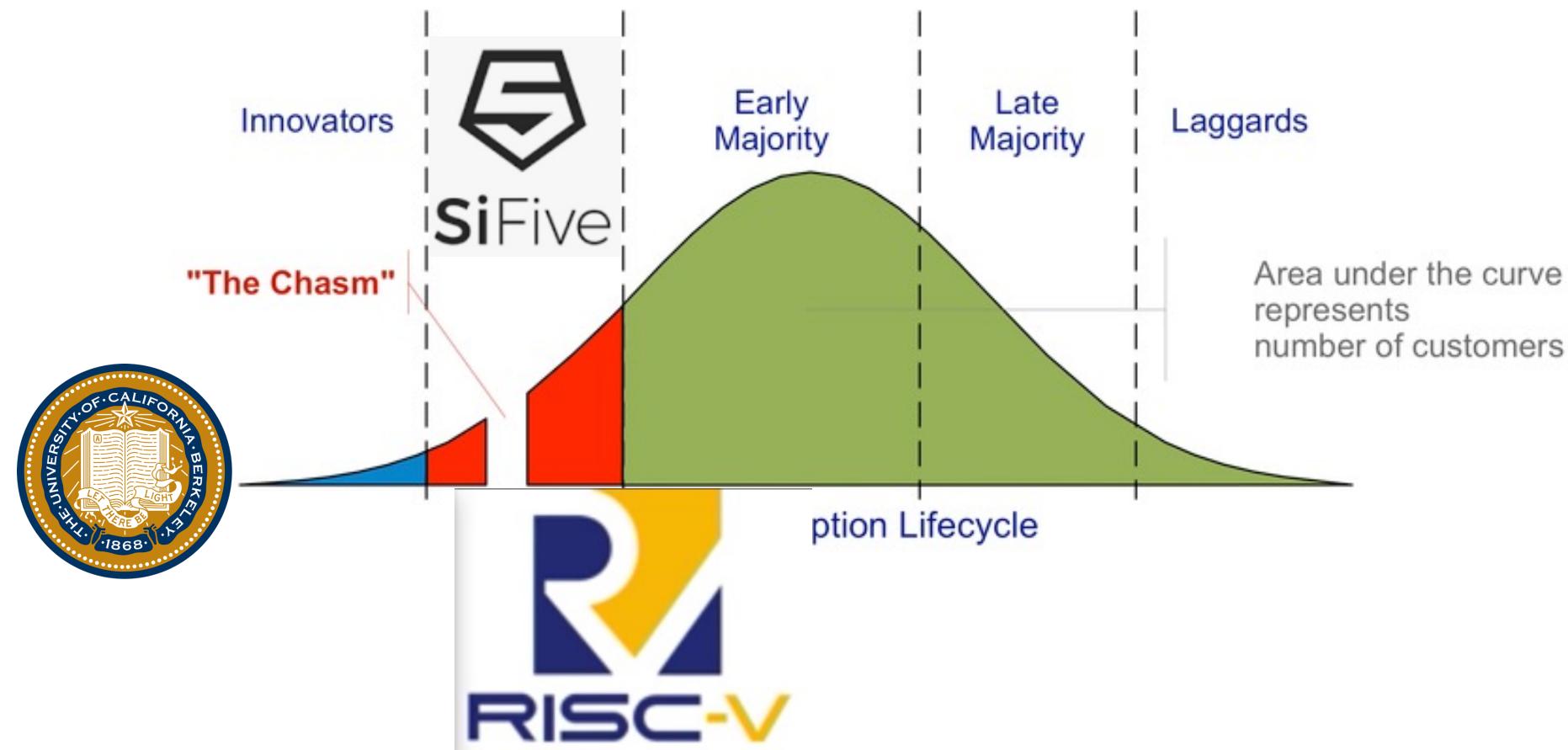
- ❖ 兼顾ISA中立和推动商业化



# 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？

# 那么，为何是伯克利搞成了RISC-V，而不是中国的高校或企业？

- 一呼天下应的出身（代表着技术领域的信用货币）
- 有钱、正确的烧钱（进行了十几次流片\*）
- 符合时宜的设计理念（模块化、定制化）
- 简洁（连一个本科生都可以接近独立的设计出短流水线MCU）
- 非常注重软件栈、借力
- 天下桃李，悉在公门矣

<https://blog.csdn.net/moxibingdao/article/details/106667615>

那么，为何是伯克利搞成了RISC-V，而不是中国的高校或企业？  
我们缺少什么？如何补缺？如何追赶？

- 一呼天下应的出身（代表着技术领域的信用货币）
- 有钱、正确的烧钱（进行了十几次流片\*）
- 符合时宜的设计理念（模块化、定制化）
- 简洁（连一个本科生都可以接近独立的设计出短流水线MCU）
- 非常注重软件栈、借力
- 天下桃李，悉在公门矣

<https://blog.csdn.net/moxibingdao/article/details/106667615>

# RISC-V 已经成为未来的三大主流架构之一

RT-Thread 早在几年前就已经支持RISC-V架构  
并已经加入 RISC-V 国际基金会，成为战略会员



# 今天演讲的主题内容（Takeaway）

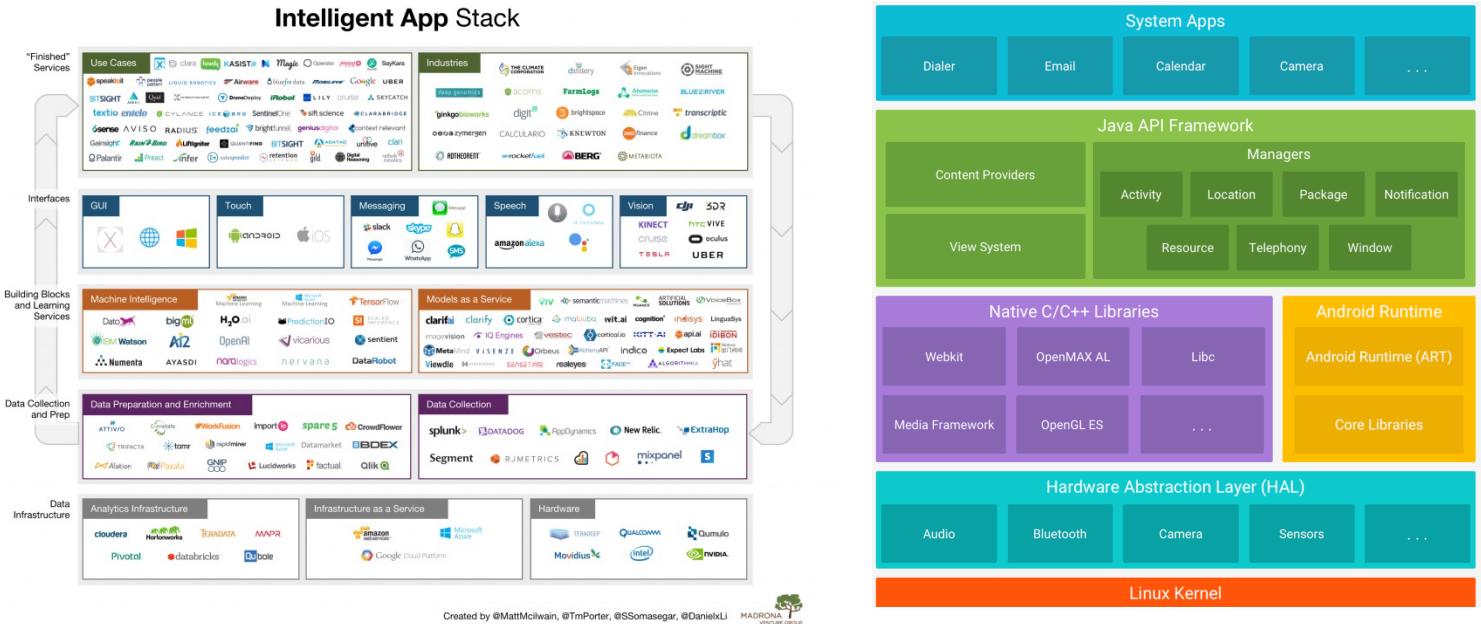
- RISC-V 已经成为未来的主流架构
  - 任何芯片公司都应投入；任何软件公司都需要适配；从业者应该立刻开始学习
- 拥抱开源软件并积极成为贡献者是企业生存、胜出的必要条件
  - 要么积极融入开源社区，要么被开源软件吞噬；不被upstream支持等于活在ICU
- RISC-V 软件生态领域依然有大量机会可以把握
  - 以中国科学院软件研究所PLCT实验室在RISC-V领域的贡献和成果为例

# 拥抱开源软件：软件吞噬世界，开源软件吞噬软件

- 1 <https://developer.ibm.com/blogs/how-open-source-software-is-eating-the-world/>
- 2 <https://algorithmia.com/blog/wp-content/uploads/2016/06/Screen-Shot-2016-06-08-at-3.35.53-PM-1024x730.png>
- 3 <https://developer.android.com/guide/platform>

# 拥抱开源软件：软件吞噬世界，开源软件吞噬软件

是什么让开源不可避免？



1 <https://developer.ibm.com/blogs/how-open-source-software-is-eating-the-world/>

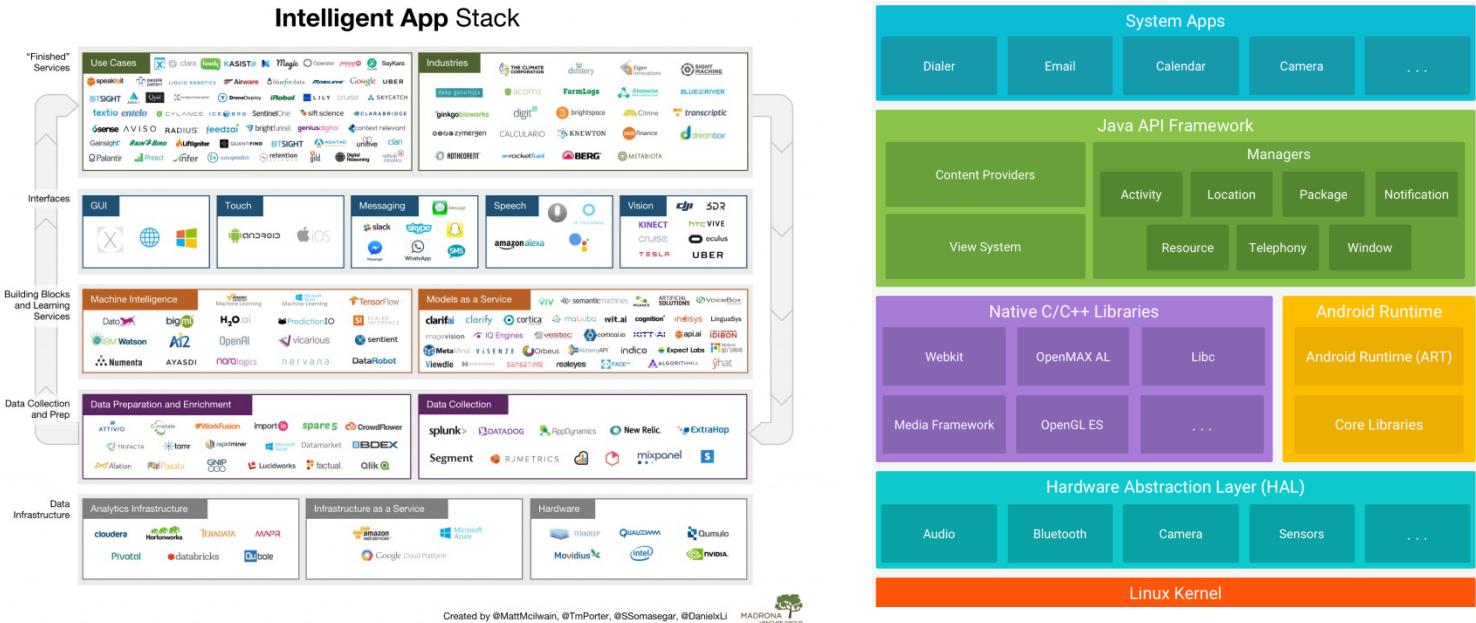
2 <https://algorithmia.com/blog/wp-content/uploads/2016/06/Screen-Shot-2016-06-08-at-3.35.53-PM-1024x730.png>

3 <https://developer.android.com/guide/platform>

# 拥抱开源软件：软件吞噬世界，开源软件吞噬软件

是什么让开源不可避免？

软件系统的规模



1 <https://developer.ibm.com/blogs/how-open-source-software-is-eating-the-world/>

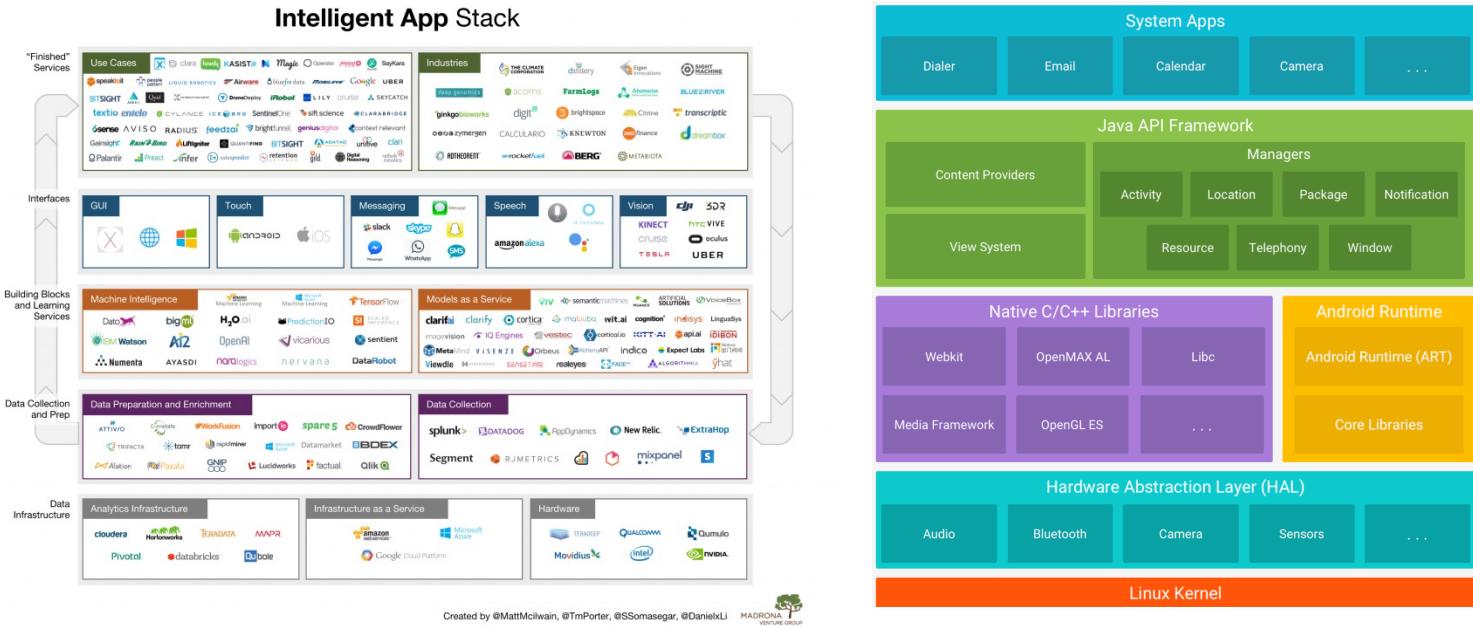
2 <https://algorithmia.com/blog/wp-content/uploads/2016/06/Screen-Shot-2016-06-08-at-3.35.53-PM-1024x730.png>

3 <https://developer.android.com/guide/platform>

# 拥抱开源软件：软件吞噬世界，开源软件吞噬软件

是什么让开源不可避免？

软件系统的规模  
的扩张速度



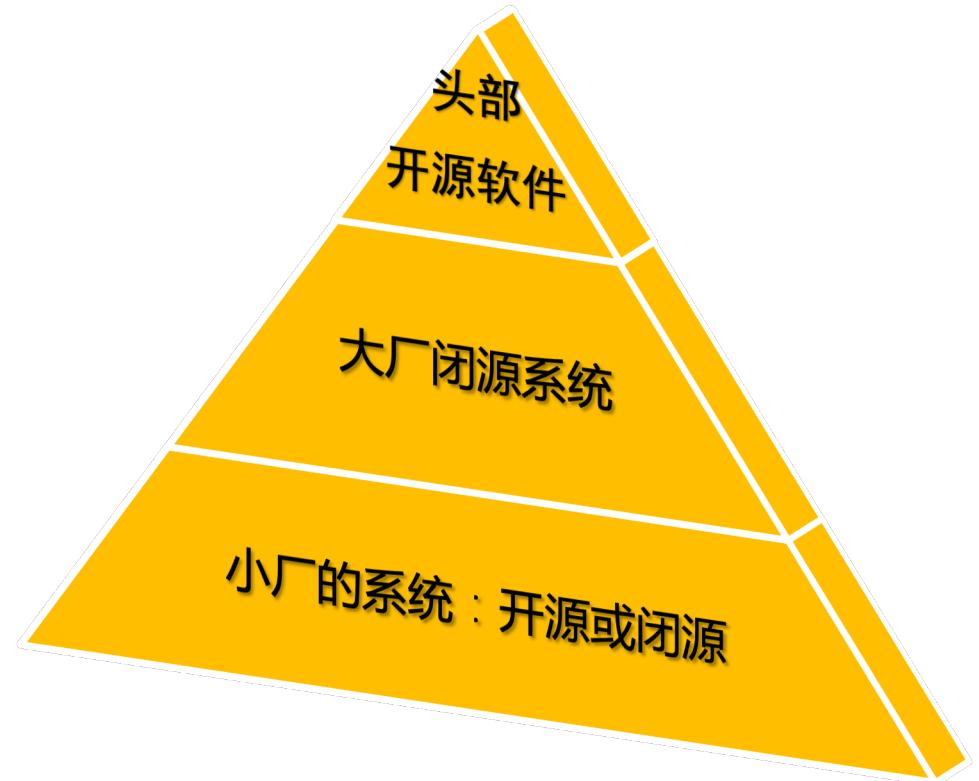
1 <https://developer.ibm.com/blogs/how-open-source-software-is-eating-the-world/>

2 <https://algorithmia.com/blog/wp-content/uploads/2016/06/Screen-Shot-2016-06-08-at-3.35.53-PM-1024x730.png>

3 <https://developer.android.com/guide/platform>

# 拥抱开源软件，或者被吞噬

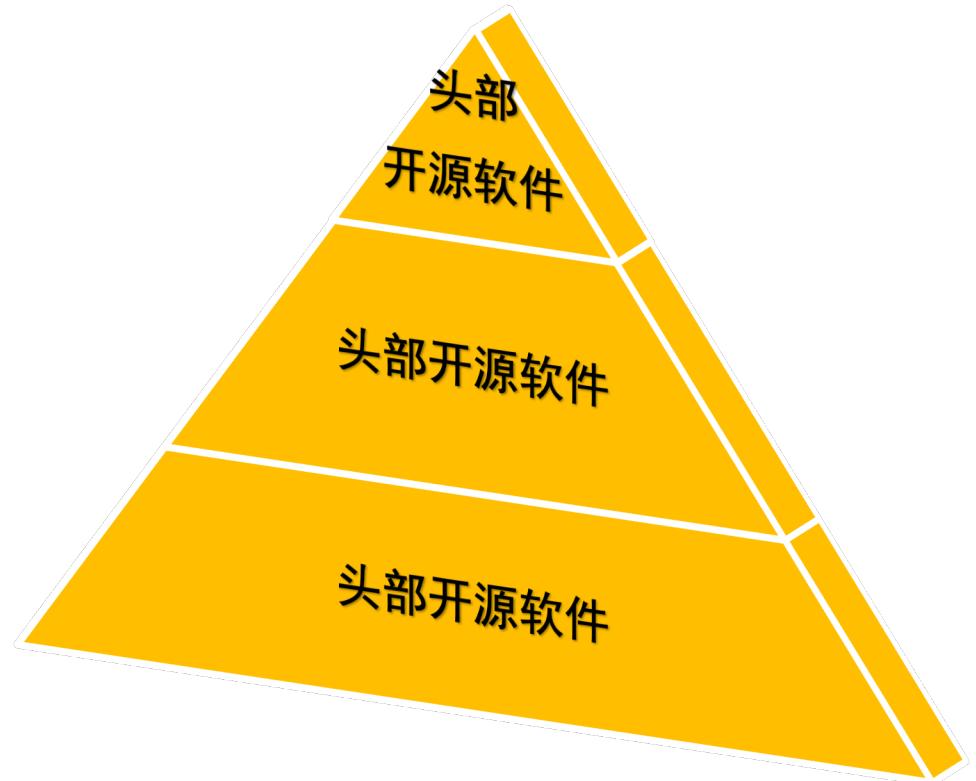
- 开源软件正在吞噬世界
  - ❖ **GNU/Linux、FreeRTOS、RT-Thread**
  - ❖ **Hadoop/Spark、Docker/k8s.....**
- 不被开源软件支持将付出极大的维护成本
  - ❖ **进而失去竞争优势被淘汰**



# 拥抱开源软件，或者被吞噬

开源软件的世界很残酷的，只有头部的社区容易吸引用户和贡献者

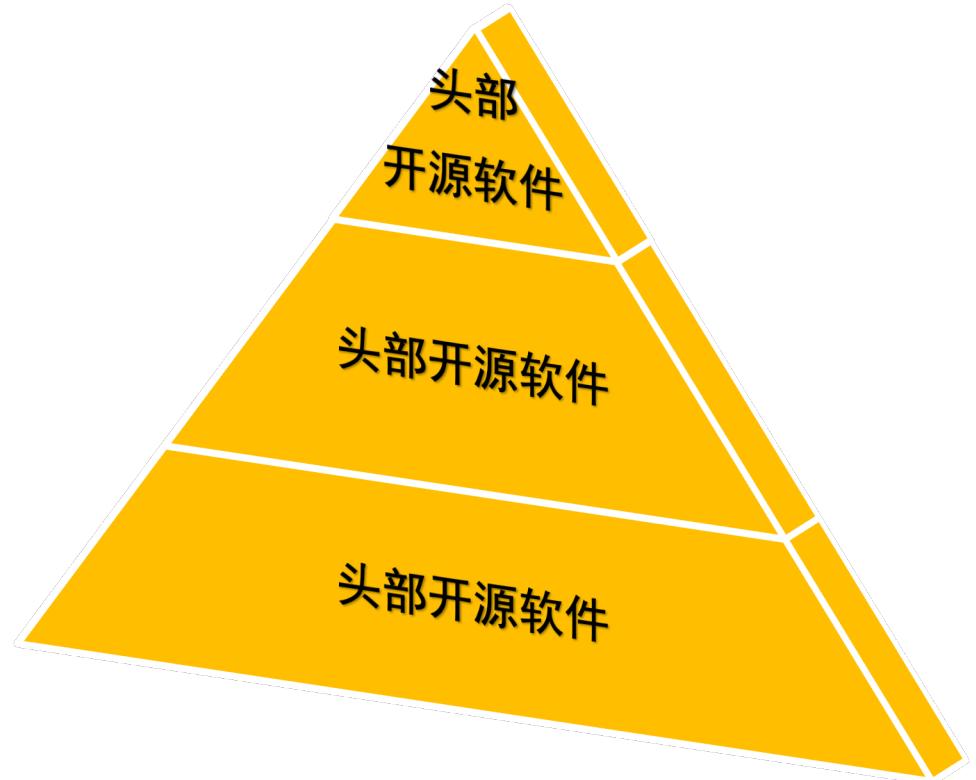
- 开源软件正在吞噬世界
  - ❖ **GNU/Linux、FreeRTOS、RT-Thread**
  - ❖ **Hadoop/Spark、Docker/k8s.....**
- 不被开源软件支持将付出极大的维护成本
  - ❖ **进而失去竞争优势被淘汰**



# 拥抱开源软件，或者被吞噬

开源软件的世界很残酷的，只有头部的社区容易吸引用户和贡献者

- 开源软件正在吞噬世界
  - ❖ **GNU/Linux、FreeRTOS、RT-Thread**
  - ❖ **Hadoop/Spark、Docker/k8s.....**
- 不被开源软件支持将付出极大的维护成本
  - ❖ 进而失去竞争优势被淘汰
- 不被上游维护的代码就像是活在ICU里：费用昂贵、死亡率高
  - ❖ 举例：国内许多的芯片厂商和系统厂商



# 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  
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  
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



## Changing Priorities



<u>2010</u>	<u>2020</u>
<ol style="list-style-type: none"><li>1. Be simple, efficient, extensible</li><li>2. Revisit legacy design decisions</li><li>3. Have basic software</li></ol> <p><i>(computer-architecture-driven project)</i></p>	<ol style="list-style-type: none"><li>1. Run all software</li><li>2. Be feature complete<ul style="list-style-type: none"><li>➤ see #1</li></ul></li><li>3. Be stable<ul style="list-style-type: none"><li>➤ see #1</li></ul></li><li>4. Support innovation<ul style="list-style-type: none"><li>➤ conflicts with #1,#2,#3?</li></ul></li></ol> <p><i>(software-driven project)</i></p>

# RISC-V 正确的转向

RT-Thread 对于  
RISC-V生态的完整  
非常重要



## Changing Priorities



<u>2010</u>	<u>2020</u>
1. Be simple, efficient, extensible	1. Run all software
2. Revisit legacy design decisions	2. Be feature complete ➤ see #1
3. Have basic software	3. Be stable ➤ see #1
<i>(computer-architecture- driven project)</i>	4. Support innovation ➤ conflicts with #1,#2,#3? <i>(software-driven project)</i>

# OpenJDK19 will natively support RISC-V

*RISC-V compiler support merged on March 24th, 2022*



<https://github.com/openjdk/jdk/commit/5905b02c0e2643ae8d097562f181953f6c88fc89>

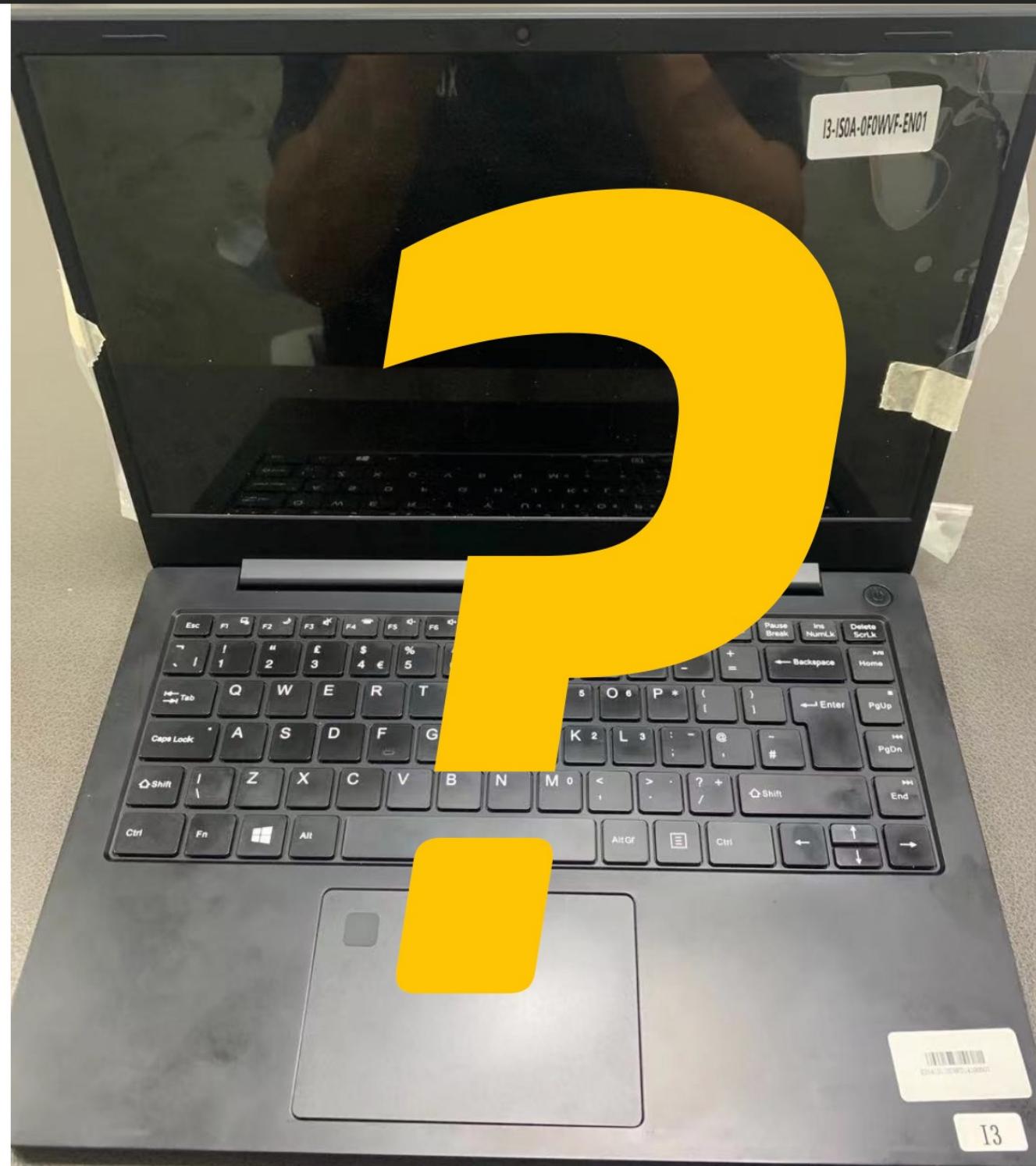
# Android 12 running on RISC-V with optimised AI performance



<https://chipsalliance.org/announcement/2022/04/21/alibaba-cloud-announced-progress-in-porting-android-functions-onto-risc-v/>

<https://www.techradar.com/news/alibaba-cloud-is-close-to-getting-android-working-on-risc-v-silicon>

# Will we see the first RISC-V laptop released in 2022?



# 拥抱开源软件：怎么拥抱？

拥抱开源软件的行为需要获得收益——无论是降低了成本，还是提高了收入

- 对于芯片公司
  - ❖ 保持对业务相关的开源软件社区长期和积极的投入，确保对自己产品的支持
- 对于软件系统公司
  - ❖ 差异化竞争；利用开源软件降低成本；回馈开源社区并积极参与社区治理
- 对于软件开发者
  - ❖ 积极参与开源社区贡献；向社区中优秀的开发者学习；不断提高个人在社区的技术声望

# 面向RISC-V的开源软件生态：国内的贡献度起步较晚，正在加速

- 重要的开源软件的 RISC-V Ports 都已经有志愿者维护
  - ❖ **GNU Toolchain、glibc/newlib、Linux Kernel、Clang/LLVM**
  - ❖ 谁在上游（Upstream）贡献了代码，谁就掌控了更多的话语权
- 自2020年开始越来越多的看到来自中国的贡献者
  - ❖ **OpenJDK、V8、QEMU、GCC、LLVM、Linux Kernel、etc.**

# 面向RISC-V的开源软件生态：仍有大量机会

## ■ 基础软件领域的狂欢：已经很久没有这么多轮子可以发明了

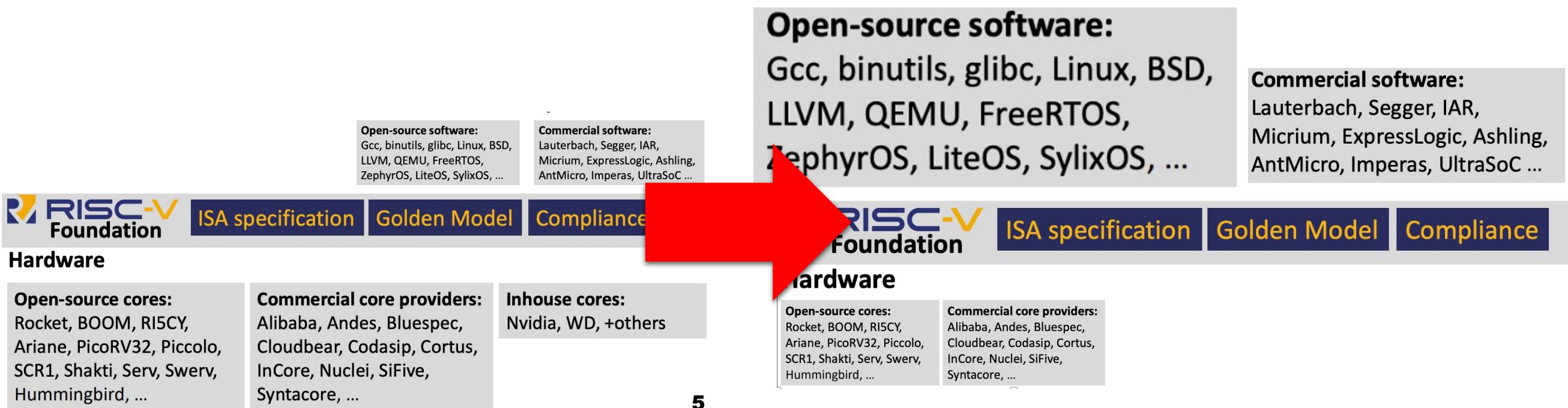
- ❖ 各种编程语言的编译器  $\times N$
- ❖ 语言虚拟机和运行时  $\times N$
- ❖ 系统虚拟机、指令集模拟器  $\times N$
- ❖ 全系统性能分析和优化、软硬件协同演进的定制和优化
- ❖ 二进制工具  $\times N$
- ❖ 全新的系统安全视角和可能性

请务必看看 bunnie 的 keynote [https://www.youtube.com/watch?v=zXwy65d\\_tu8](https://www.youtube.com/watch?v=zXwy65d_tu8)

# 国内的企业有哪些机会

- MCU、AIoT等领域已经形成了成功的模式，闷声发财
- 通用计算（PC/Server/HPC）领域正在发力
- 纯CPU创业会很辛苦，因为有大财团互联网公司在旁边准备投你的友商
- 各类AI加速器初创公司还有机会，但是如何面对NV的坚壁清野
- 目前国内的差距主要体现在软件能力，~~而不是硬件研发能力上(我不清楚)~~
- 但是纯做开源软件并不能养活自己，这是个问题

# RISC-V生态的边界正在经历一个商业场景的变迁：卖硬件到卖系统



# 今天演讲的主题内容（Takeaway）

- 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；从业者应该立刻开始学习
- 拥抱开源软件并积极成为贡献者是企业生存、胜出的必要条件
  - 要么积极融入开源社区，要么被开源软件吞噬；不被upstream支持等于活在ICU
- RISC-V 软件生态领域依然有大量机会可以把握
  - 以中国科学院软件研究所PLCT实验室在RISC-V领域的贡献和成果为例

# 面向RISC-V的开源软件生态：以软件所的贡献为例

## ■ 基础软件领域的国家队，有义务为国内企业提供「开源软件公共品」

- ❖ 编译器领域：RVV-LLVM、GNU工具链、MLIR、gollvm
- ❖ 虚拟机领域：V8 for RISC-V、NodeJS、OpenJDK/RV32G
- ❖ 模拟器领域：面向国内厂商的QEMU、Spike支持、Gem5、Sparta
- ❖ 应用领域：OpenCV for RISC-V 等
- ❖ 操作系统领域：开源供应链安全计划、点亮计划、暑期2022  
(软件所操作系统团队成果，非PLCT工作)

# 感谢 & 提问

- 如何加入RISC-V国际基金会？**线上免费申请成为个人会员**
- 如何参与国内RISC-V开源技术社区的工作？**联系PLCT！**
- 如何参与PLCT实验室的工作？**来实习吧！**
- 有问题但是不好意思现场提问？**可以加微信问**😊



# PLCT Lab 的使命

程序语言与编译技术实验室（PLCT）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，同时也是中科院软件所的愿景



让RISC-V成为所有主流开源软件的Tier-1平台。

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 ☺ →

ISCAS

# TARSIER团队的使命

确保 Debian/Ubuntu、Fedora、openEuler 等主流的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。

# 观看PLCT/OSDT的技术分享、一起学习和进步

- <https://space.bilibili.com/296494084/video>
- 已更新技术分享400+，每周三技术分享后上传

