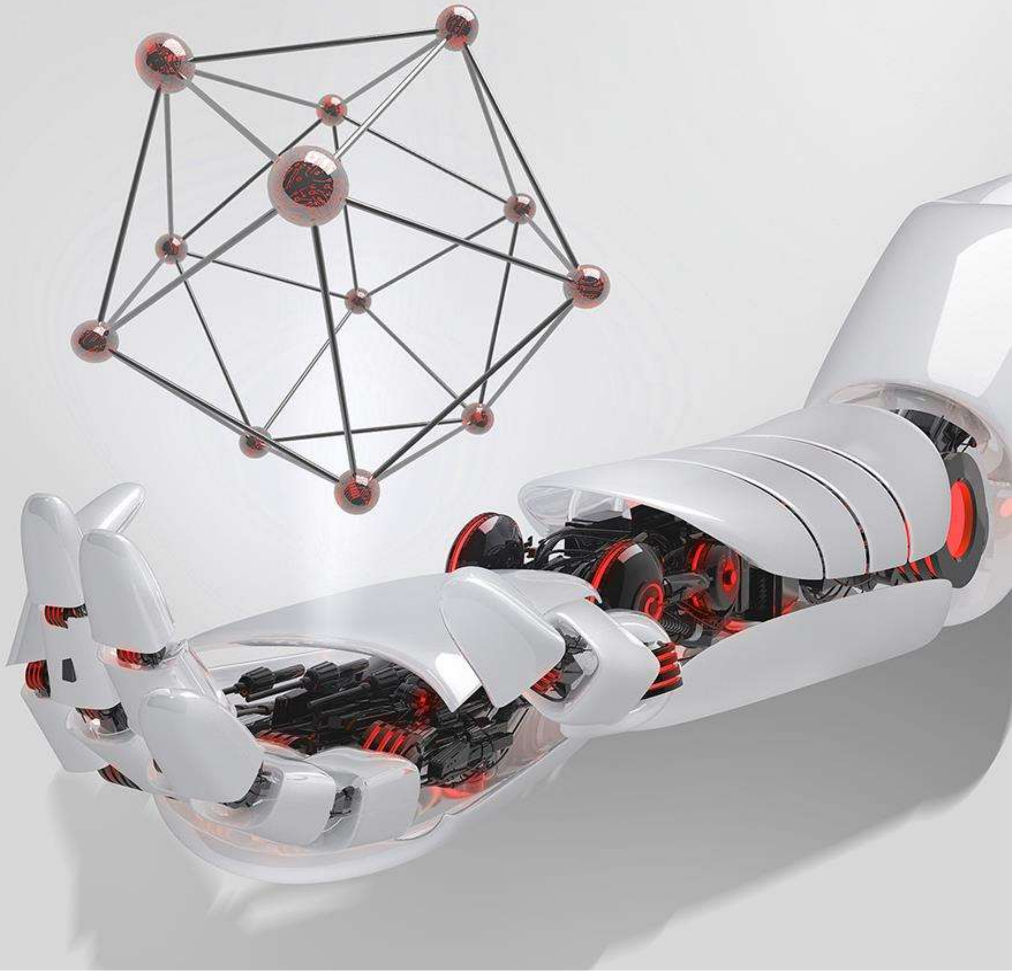
AIOT时代的
编程语言、编译器与指令集架构：
机遇、挑战与技术分享



华为技术有限公司 编译器与编程语言实验室
- 曾建江

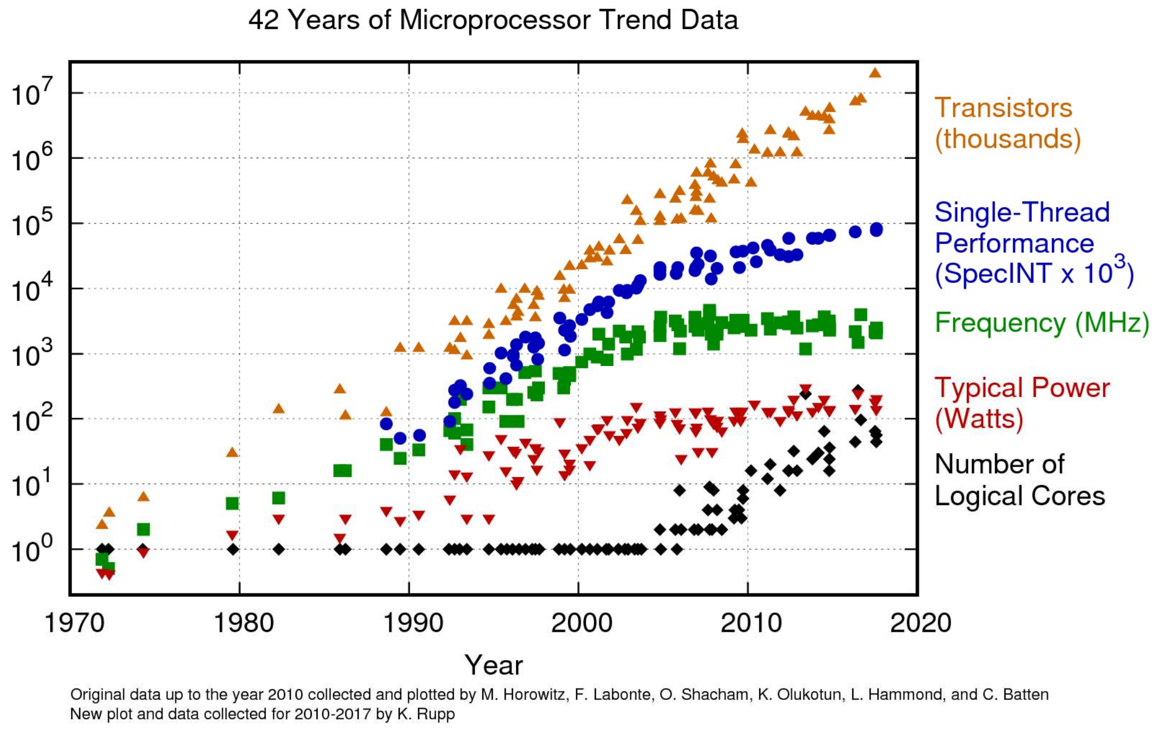
Security Level:

中央软件院
Central Software Institute

- 趋势与挑战
- 产业案例

处理器的发展趋势

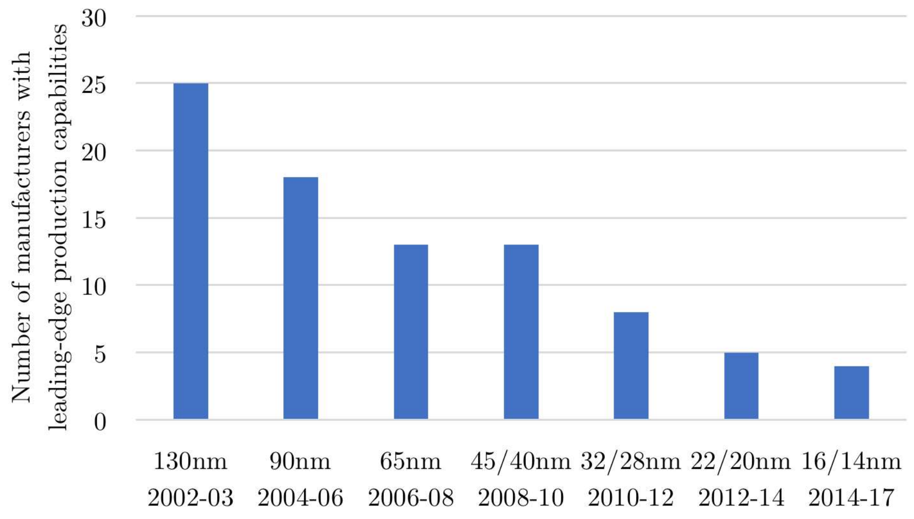
- Operating frequencies has saturated for the last 10 years. Dennard scaling is no more possible.
- Single-Thread performance is plateauing. No more complex, power hungry optimizations is added to processor design.
- But transistor count per chip is still doubling.
- This enabled more logical cores to be integrated on the same chip.
- This has maintained constant levels for power dissipations.
- Can we do better?



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

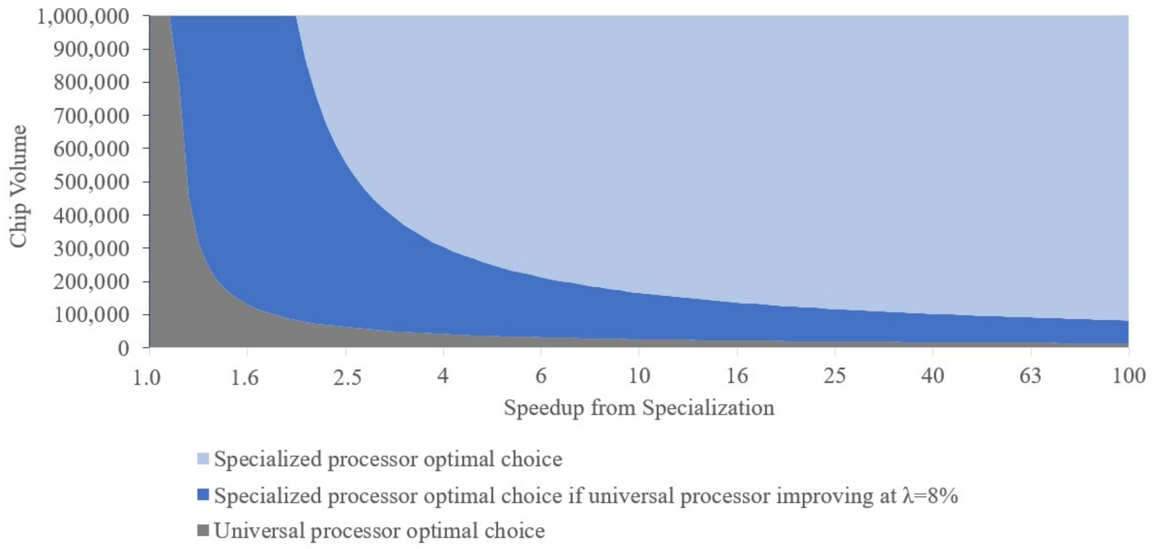
source: https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/

通用算力持续发展的时代已经结束



Shrinking transistors has become much more expensive

- $7 billion to construct a new fab
- semiconductor manufacturers from 25, in 2002, to just four today
- Performance improvements have dropped from 48% annually in 2000-2004, to 29% in 2004-2008, to 8% in 2008-2013.

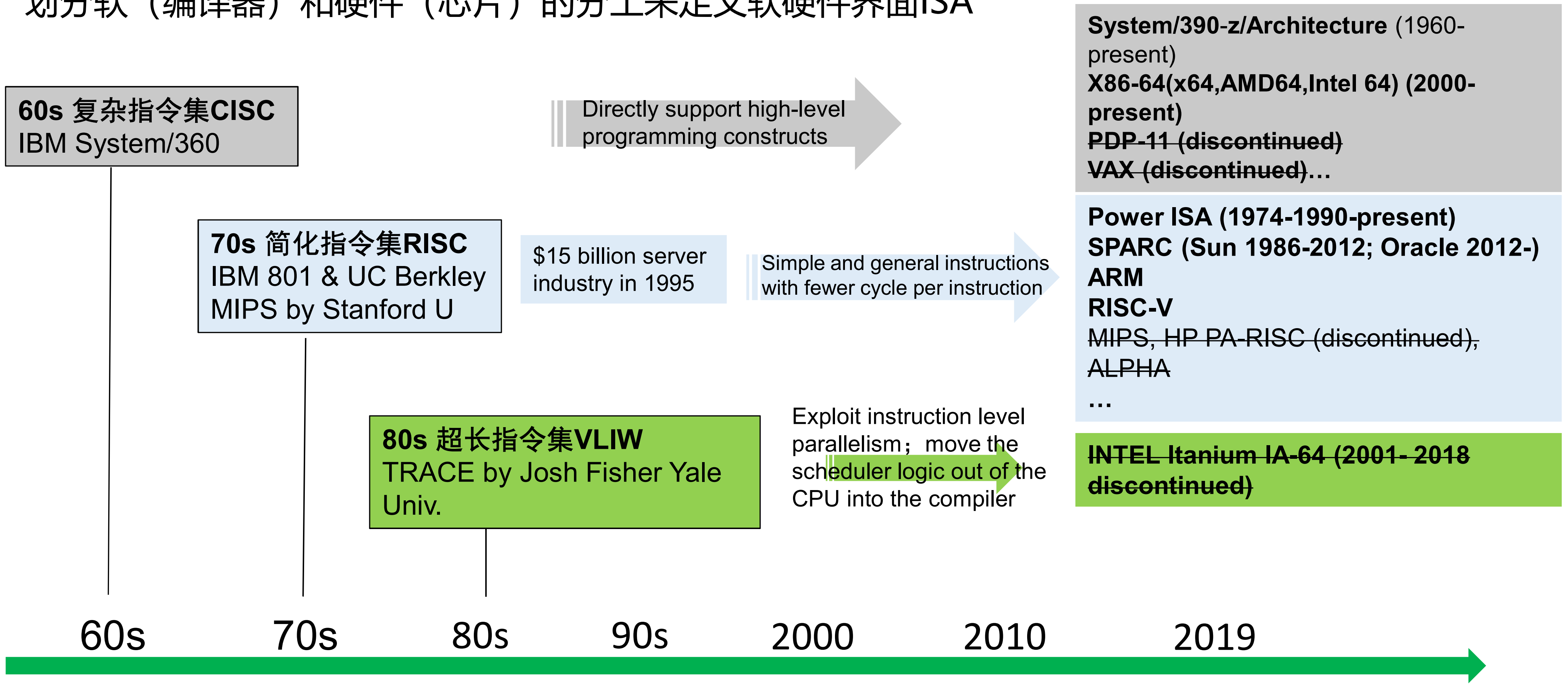


Many applications will be economically viable to run on specialized processors (computation parallelism, pattern regularity, memory access locality, lower precision)

通用CPU指令集架构的多样化和融合统一趋势

指令集架构的演进 - CISC/RISC/VLIW

划分软（编译器）和硬件（芯片）的分工来定义软硬件界面ISA



60s 复杂指令集CISC
IBM System/360

Directly support high-level programming constructs

System/390-z/Architecture (1960-present)
X86-64(x64,AMD64,Intel 64) (2000-present)
~~PDP-11 (discontinued)~~
~~VAX (discontinued)~~…

70s 简化指令集RISC
IBM 801 & UC Berkley
MIPS by Stanford U

$15 billion server industry in 1995

Simple and general instructions with fewer cycle per instruction

Power ISA (1974-1990-present)
SPARC (Sun 1986-2012; Oracle 2012-)
ARM
RISC-V
~~MIPS, HP PA-RISC (discontinued), ALPHA~~
…

80s 超长指令集VLIW
TRACE by Josh Fisher Yale Univ.

Exploit instruction level parallelism；move the scheduler logic out of the CPU into the compiler

~~INTEL Itanium IA-64 (2001- 2018 discontinued)~~

处理器架构与编译技术的演进

单核优化

SSA-based optimization
Pointer analysis
Profile-guided optimization
Memory hierarchy optimization
Loop optimizations
Data reorganization
Link-time cross-module optimization
Automatic vectorization
just-in-time compilation for Dynamic Languages

多核优化

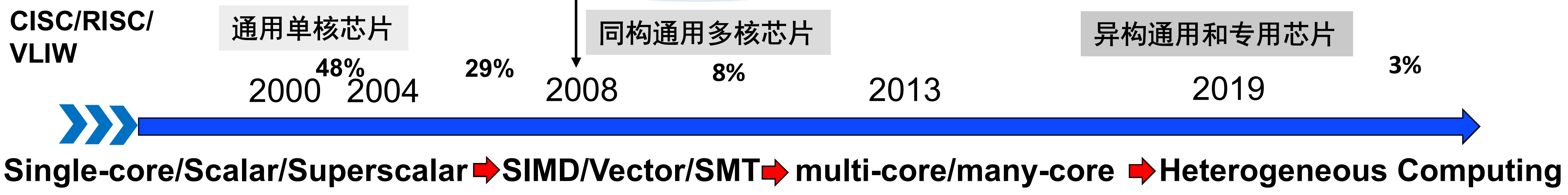
Continuous Program Optimization
Multicore and Parallel Programming
- Explicit parallelization
- Auto and semi-auto parallelization

Safety, Security and reliability
incorporate more sophisticated auto-tuning strategies
Transactional memory
speculative optimizations

性价比瓶颈　Performance improvements have dropped from 48% annually in 2000-2004, to 29% in 2004-2008, to 8% in 2008-2013.

异构优化、超异构优化

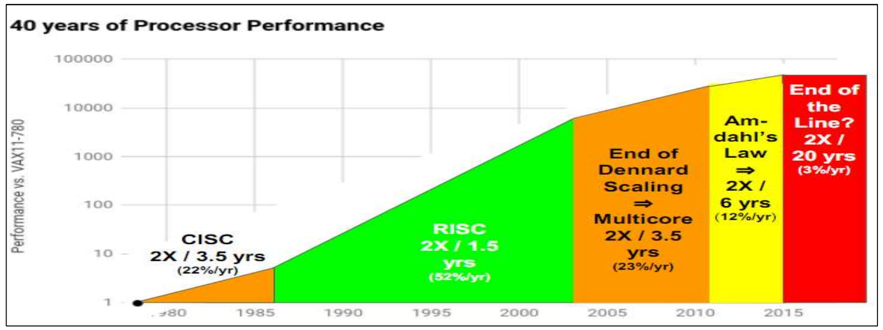
Heterogeneous computing
Programming models and languages for productivity and performance
Dynamic and static compilation for multiple scenarios/applications
Compiler optimization for multi-ISA heterogeneous architectures (CPU/GPU/DSP/AI cores/FPGA)
    Data movement communication optimization
    Data and Thread placement
    Multi-type memory optimization
    Multi-ISA optimization …
Analysis tools for correctness, security, performance
Auto performance tuning
AI/Machine Learning and Deep Learning with Compiler Technologies



后摩尔时代，芯片架构走向异构，快速演进的芯片和高复杂度的应用，对软件生产效率和兼容性提出了前所未有的挑战“software disaster”

David Patterson与John Hennessy指出“The easy ride of software is over”：在异构计算时代，想要完全释放出异构芯片的的计算能力，程序员必须既懂上层应用算法，又懂底层硬件模型，才能写出高效高质量代码，未来的编程工作会比现在更复杂

业界趋势

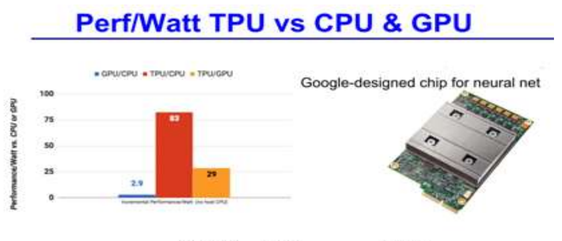


摩尔定律减速、每年只有几个百分点的提升

➢ **不同领域的计算需要越来越多的异构资源**

- **天河-2**：16,000个计算节点，每个节点 2*Xeon (IveBridge)+3*Phi。 Total 3,120,000 cores. **Linpack**：33.86 petaFLOPS， **Power** 17.6 megawatts。
- **Mac Pro: Intel Xeon** E5 (6/8/2 cores) + Dual **AMD FirePro** D500 **GPU** (1526 stream processors, 2.2 teraflops, 3-way 4k video)。
- **Amazon Linux GPU Instances g2.8xlarge:** 4 **GPU**，32 **vCPU** 。
- **Qualcomm Snapdragon 820**： octa-core **CPU**+ Adreno 530 **GPU**+ Hexagon 680 **DSP**

➢ **出现面向特定领域定制的体系结构（DSA）**



TPU能效比83倍于CPU，29倍于GPU

开发者痛点

➢ **软件开发效率“低”**：
- 普通开发者在上层应用算法和底层硬件模型无法做到两者皆懂，无法利用芯片能力实现程序性能的提升
- 目前缺乏有效的轻量化实时编译方案，无法做到运行时代码生成。

➢ **兼容性“差”，代码移植成本“高”**：
- 芯片快速演进，开发者针对多芯片需要多次开发，代码移植成本很高
- 体系结构层出不穷，开发者无法通过DSL语言，快速适配多形态芯片进行迭代式开发

如何解决3P问题（**P**roductivity、**P**erformance、**P**ortability）

支撑快速演进的芯片可编程性和解决软件兼容性问题，助力开发者实现高质量快速编程，降低代码迁移成本

➢ 趋势与挑战

➢ 产业案例

HUAWEI

昇腾：高性能、高产能的AI编译器，助力Atlas发布最快AI训练集群

2019年9月18日HC2019，副董事长胡厚崑宣布，AI基础软件助力59.8s完成ResNet-50@imageNet模型训练，**超越原世界纪录10s**，并现场展示基于MindSpore实现10.02s内20万天体识别，成功发布**全球最快**的AI训练集群Atlas 900。

关键技术：

- **Super Kernel优化技术**

  在二进制层面实现算子**Kernel融合调度**，最大化减少TS调度Kernel开销，Resnet50训练E2E性能**提升2~3s**，助力整体性能突破60s

- **并行编译、ccache优化**

  库上CCE算子构建时间从**15min+降到4min**，提升构建速度**3倍**

- **算子编译优化和代码生成技术**

  突破循环优化、同步算法、内存复用、多核多batch等优化技术，**数量级**提升TBE算子性能，**TBE成为D算子开发的统一框架**

方舟编译器作为P30关键特性发布

关键技术：

➢ **Java全静态化编译技术**

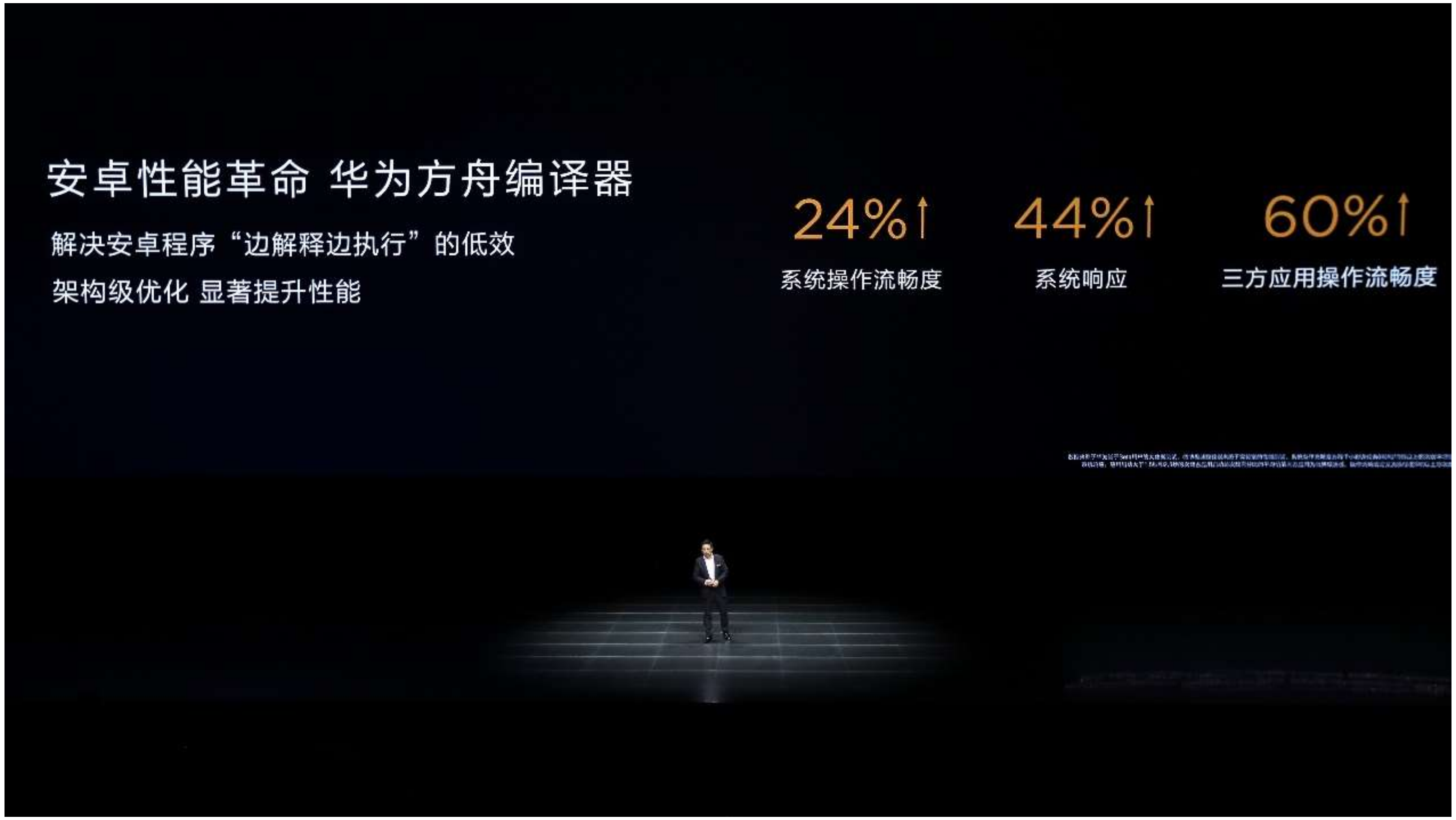
基于统一IR将Java直接编译为机器码，通过高强度编译优化提升运行效率。

➢ **Java RC内存管理**

配合编程约束后，可实现Java无GC停顿、RC低开销，对标Apple

➢ **JNI自动拆墙优化**

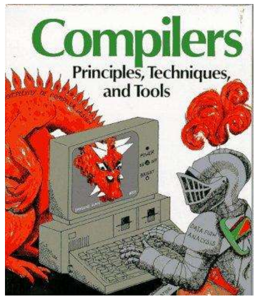
可将Java<->C/C++互操作开销降低10倍，调用开销像“一种语言”



编程语言、编译器和指令集的设计与实现所关注的问题

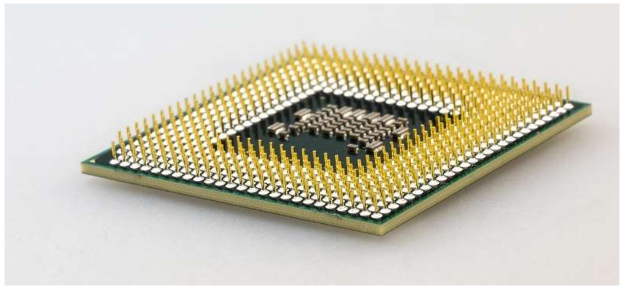
编程语言：
表达、抽象

- 表达力强：
  容易描述计算、业务逻辑
- 容易使用：好学、好写、好读
- 不易出错、代码安全
- …



编译器：让芯片能看懂
并执行

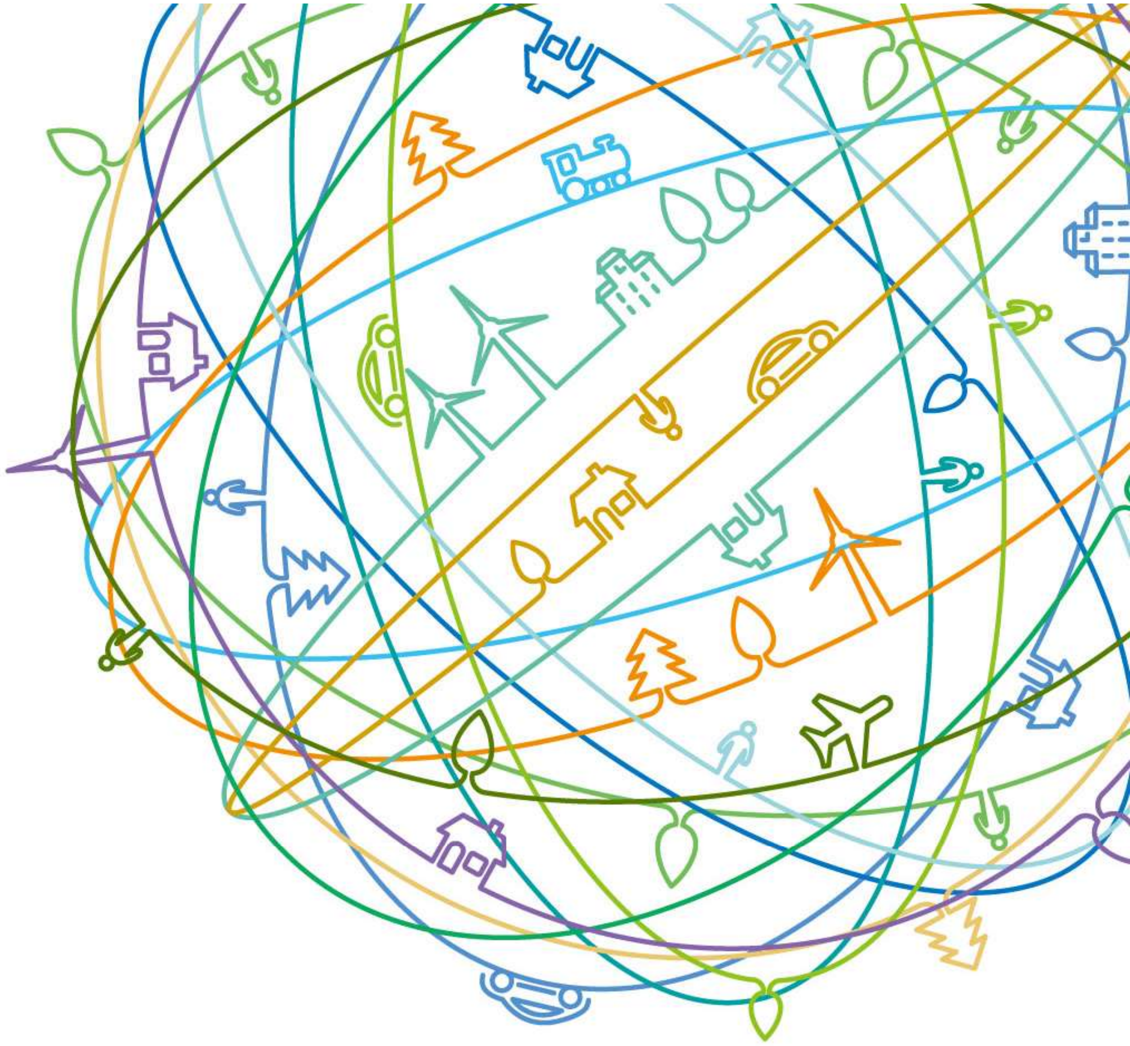
- 运行时间短
- 占用资源少
- 平台无关、可移植
- …



指令集：软硬件接口标准

- 软件需要什么样的硬件
- 硬件提供什么样的能力给软件
- 芯片实现的可行性

谢谢！



HUAWEI TECHNOLOGIES CO., LTD.

www.huawei.com