

IPCC

第一讲-并行应用开发概论

时间：2020年8月

主讲人：刘帅

目录

CONTENTS

PART 1

并行计算的必要性

- | 现实社会需求
- | 处理器发展趋势
- | 功耗墙
- | 现代计算机体系架构

PART 2

处理器重要技术

- | 处理器技术
- | 指令处理流程
- | 指令级并行 (ILP)
- | 内存访问墙-乱序执行
- | CPU架构简介

PART 3

并行计算系统发展趋势

- | 系统体系架构
- | 加速器
- | 内存带宽
- | 调度系统

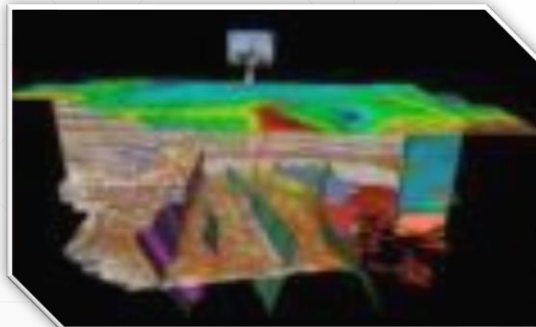
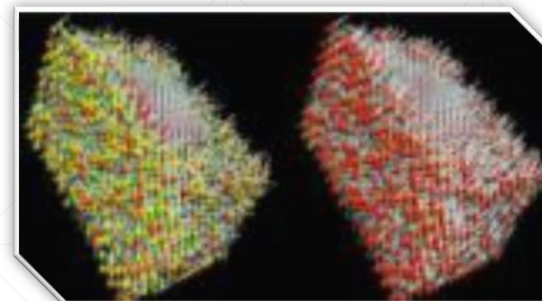
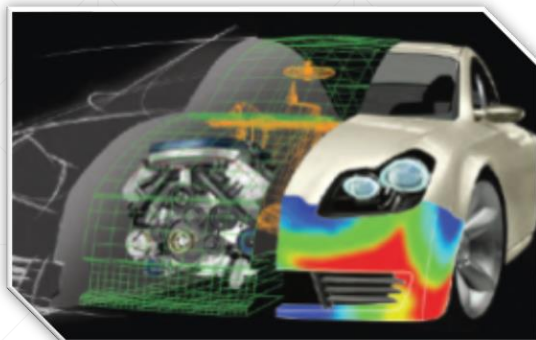


01

PART

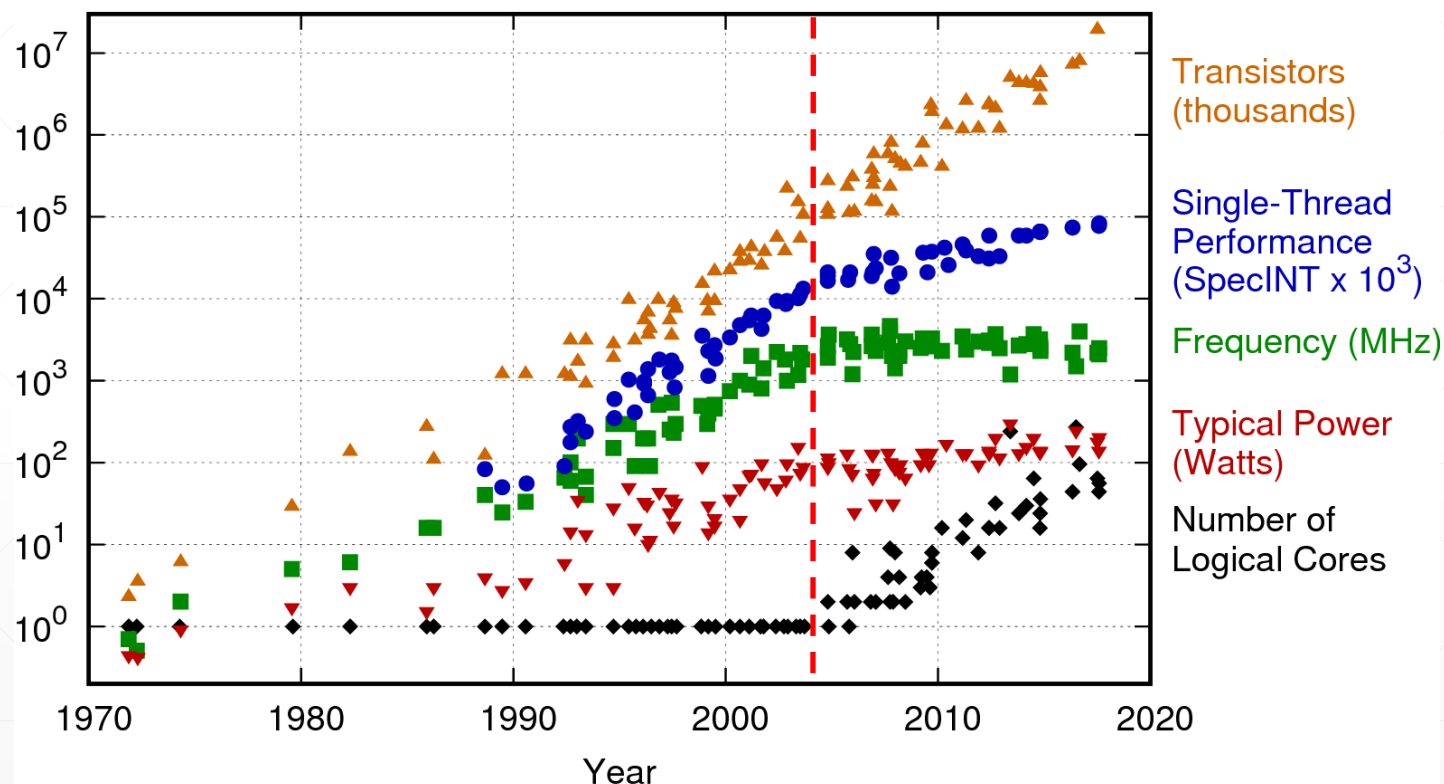
并行计算的必要性

现实社会计算需求与日俱增



处理器发展趋势-并行化

42 Years of Microprocessor Trend Data

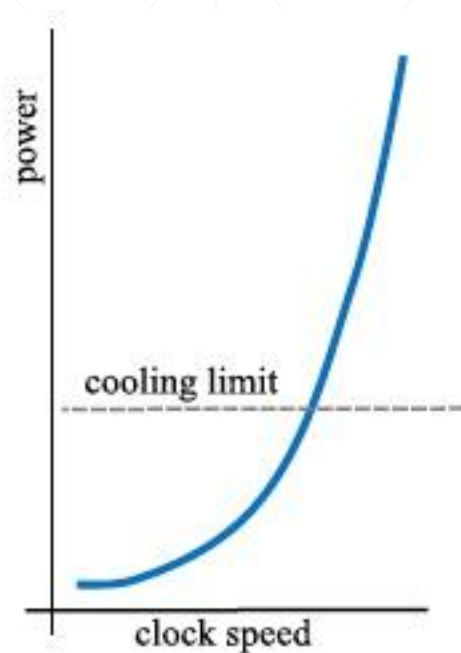


Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

- 功耗问题 → 多核技术
- 多核技术 → 依靠主频加速应用的“免费午餐”时代结束^[1]。
- 并行化及并行化优化显得尤为重要。

[1]. Herb Sutter, “The Free Lunch is Over, A Fundamental Turn Toward Concurrency in Software”, Dr. Dobbs’s Journal, 30(3), March 2005.

功耗墙



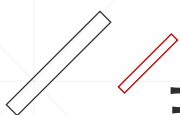
Liquid nitrogen for CPU
overclocking



Microsoft's project Natick:
immersed datacenter

news.microsoft.com/natick

无限制提高时钟频率带来的散热问题，实践上无解或者极其昂贵



现代计算机体系架构下的并行化

- 体系架构选择、优化成本（制造与功耗）与性能
- 指令级并行 (ILP)
 - ✓ 流水线 (Pipelining)
 - ✓ 多指令发射
 - ✓ 乱序执行
- 向量化 (SIMD)
 - ✓ 单条指令执行能够同时处理多个数据片
- 多线程 (MIMD)
 - ✓ 多个程序实例能同时运行
- 多核



02

PART

处理器重要技术

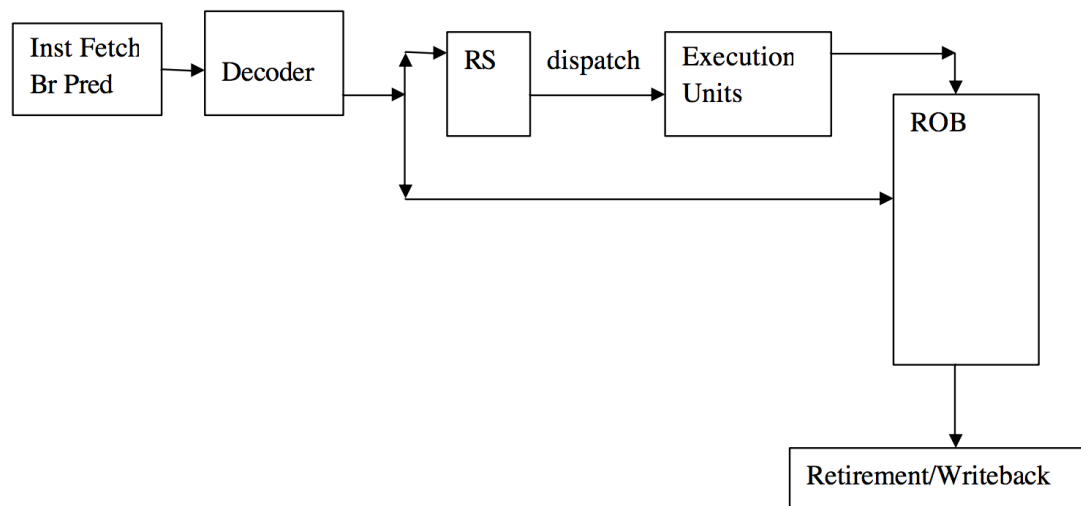


CPU架构发展-处理器发展重要技术

- 位宽增大 8bit -> 16bit -> 32bit -> 64 bit;
- 流水线，使用多级流水提高主频；
- 基于Tomasulo算法的乱序执行；
- SSE (Streaming SIMD Extensions) 、 AVX(Advanced Vector Extensions)指令集；
- 集成内存控制器；
- 超线程技术，进一步利用闲置计算部件；
- 多核技术。

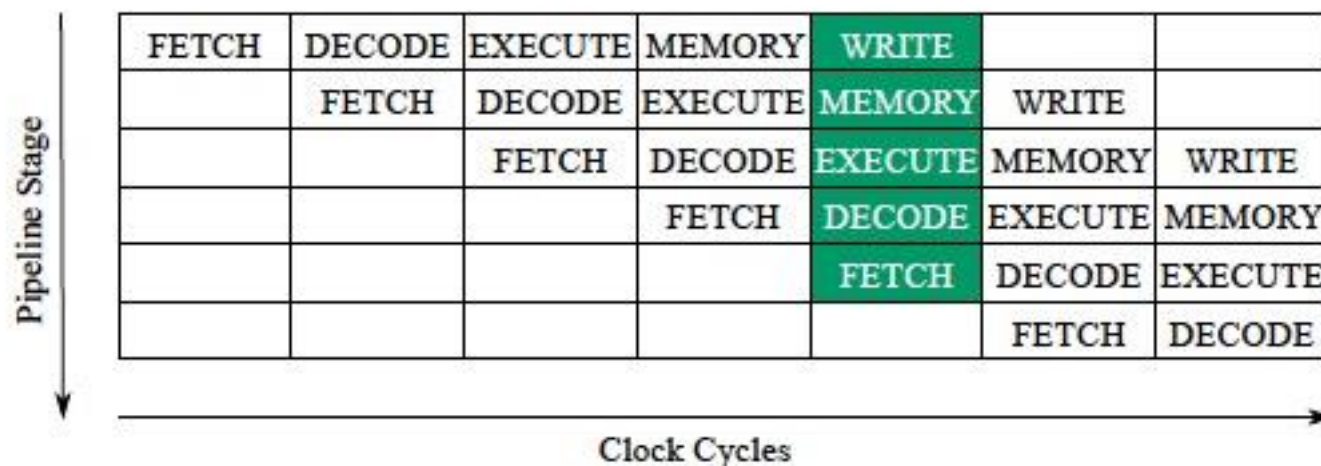
CPU架构技术简介-指令处理流程

1. 取指令
2. 翻译成微指令，并进保留栈
3. 在计算单元上乱序执行
4. 进入ReorderBuffer，根据应用逻辑顺序
5. 指令执行完成

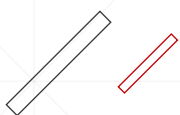


指令级并行 (ILP) - 流水线

流水线 – 硬件支持同一时刻，不同指令流运行在不同状态下



仅有有限阶段划分，有可能会产生冲突



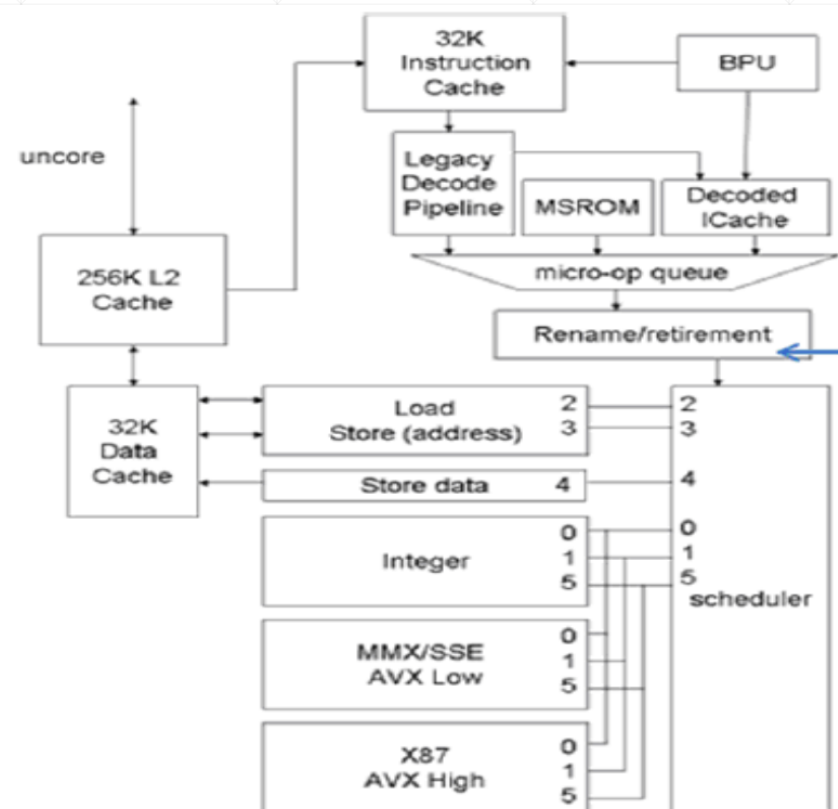
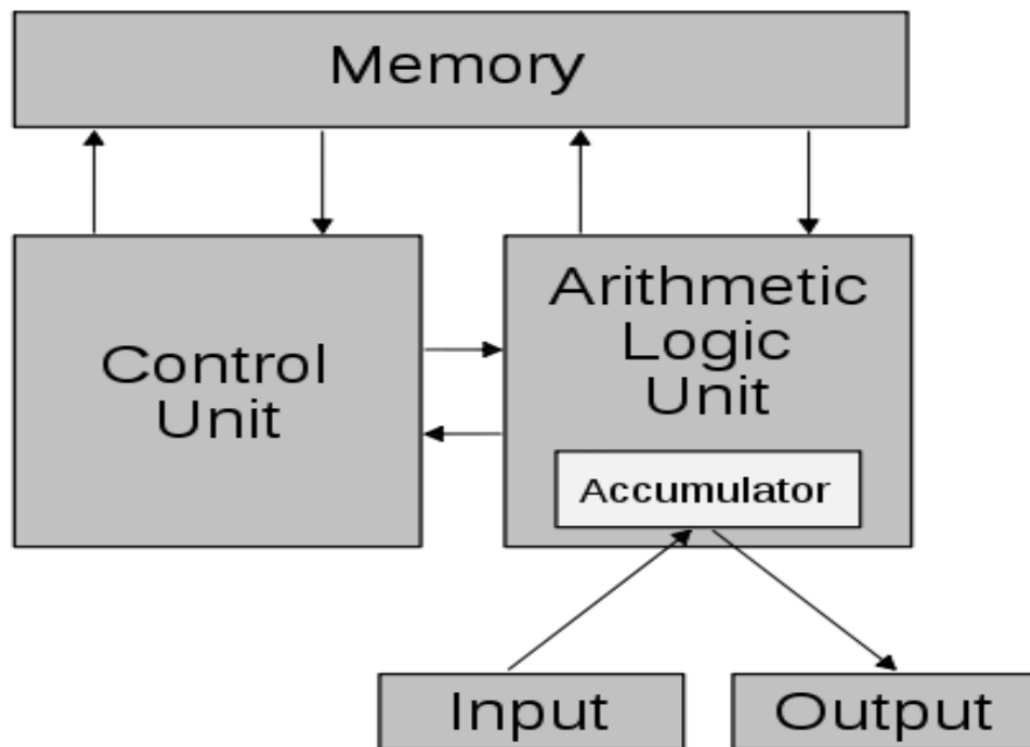
内存访问墙-乱序执行

乱序执行 – 硬件重新组织指令流里的执行顺序，以最小化内存访问延迟

In-Order	FETCH	DECODE	EXECUTE	MEMORY	MEMORY	MEMORY	WRITE
		FETCH	DECODE	STALL	STALL	STALL	EXECUTE
			FETCH	STALL	STALL	STALL	DECODE
Out-of-Order	FETCH	DECODE	EXECUTE	MEMORY	MEMORY	MEMORY	WRITE
		FETCH	DECODE	EXECUTE	WRITE		
			FETCH	DECODE	EXECUTE	WRITE	

CPU 性能增长远超内存带宽, 乱序执行不能完全弥补差距

CPU架构发展简介-架构的巨大转变

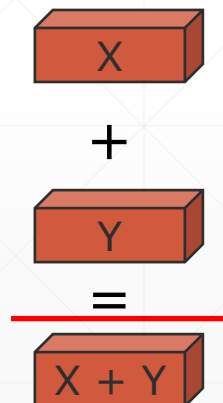


von Neumann architecture

CPU架构简介-向量化指令SIMD

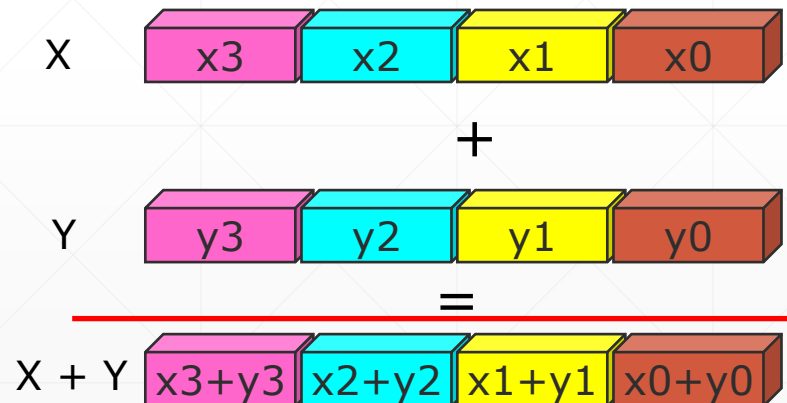
- **Scalar processing**

- traditional mode
- one operation produces one result



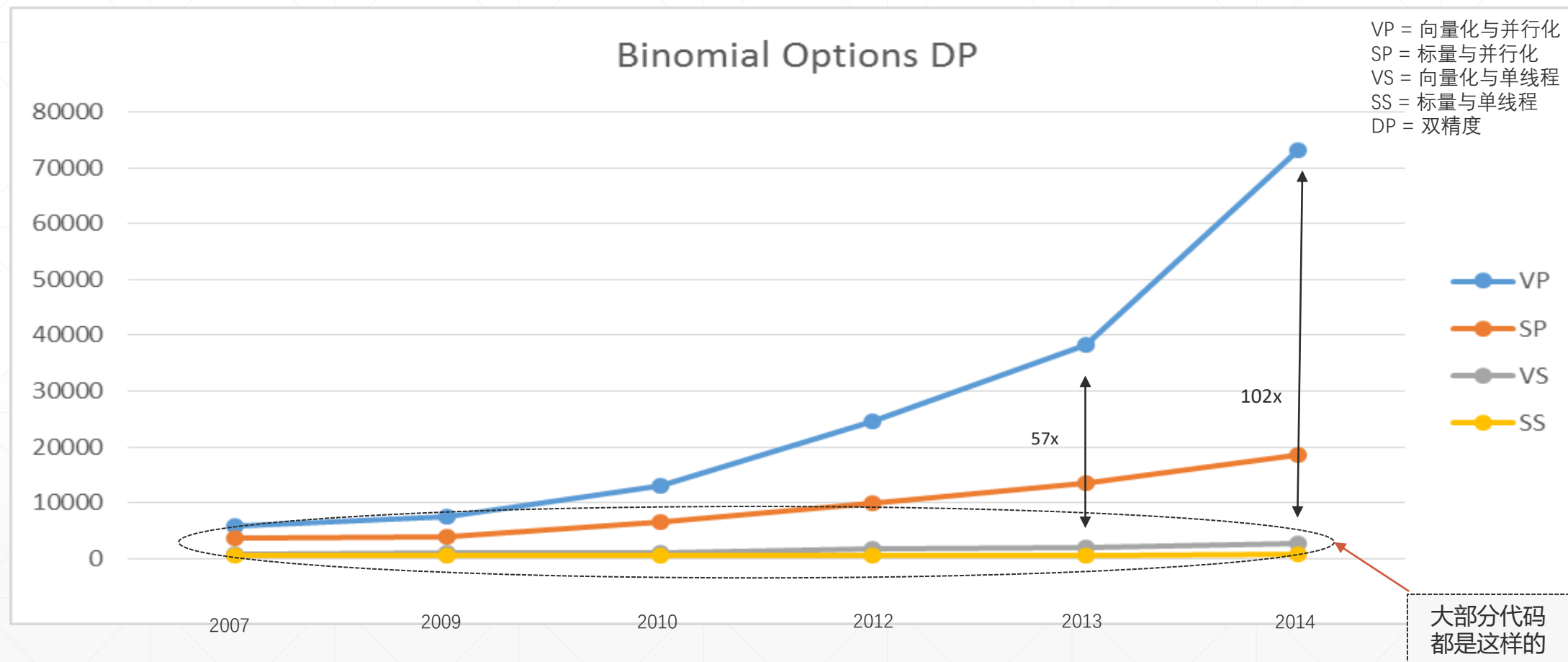
- **SIMD processing**

- one instruction produces multiple results



CPU架构简介-向量化指令SIMD

——并行化向量化对性能影响





03

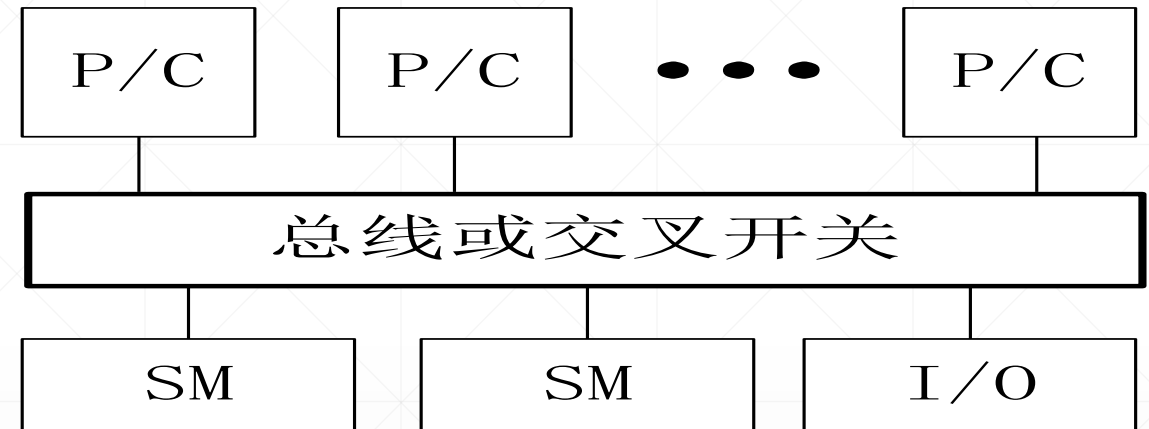
PART

并行计算系统发展趋势

并行计算机系统体系架构

——对等多处理机(SMP, Symmetric Multi-processing)

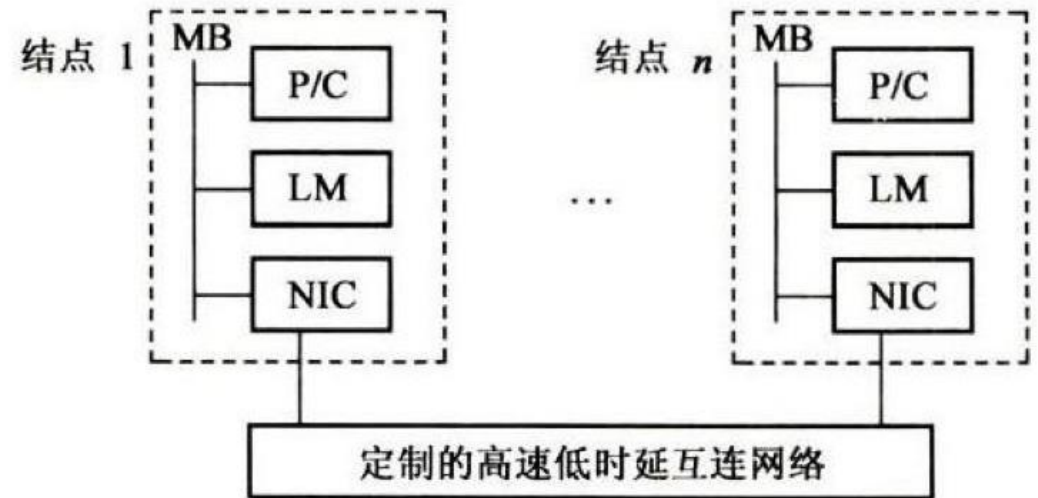
- 采用商用微处理器
- 基于总线连接或交叉开关互联
- 集中式共享存储，UMA结构
- SGI Power Challenge, DEC Alpha Server, Dawning 1



并行计算机系统体系架构

——大规模并行机(MPP, Massively Parallel Processing)

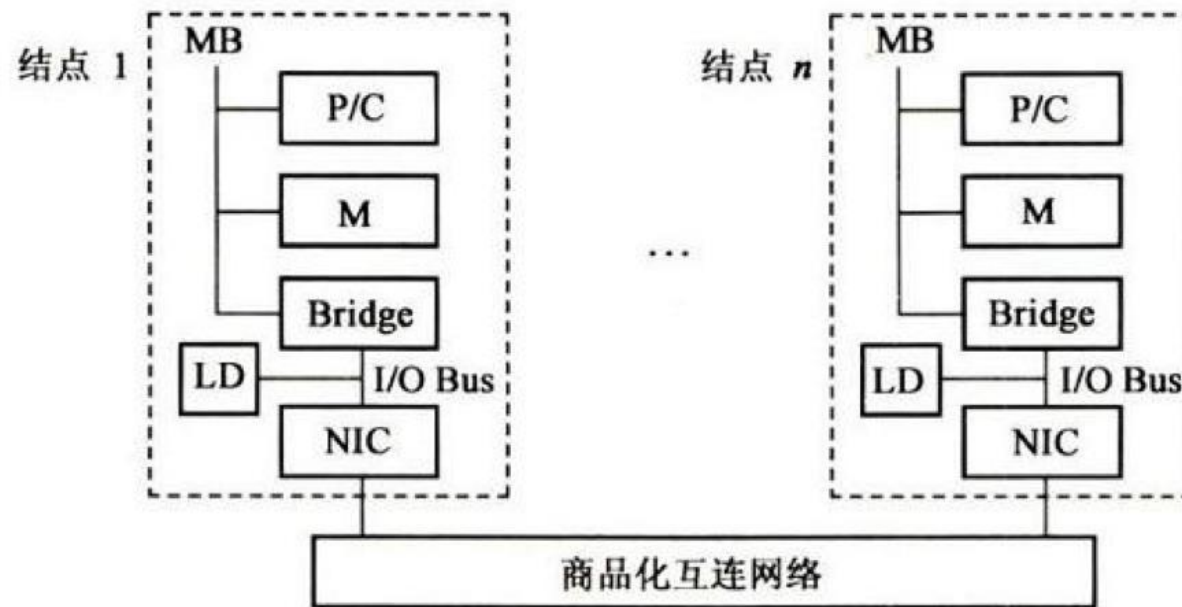
- ✓ 成百上千个处理器组成的大规模计算机系统，规模是变化的。
- ✓ 高带宽低延迟**定制**互联网络，**网卡**直接连接到**存储总线**。
- ✓ 商用处理器，相对稳定的结构



并行计算机系统体系架构

——机群(Cluster)

- 商品化的互连网络和处理器
- 结点机通过I/O总线与网卡相连
- 每个结点机上留驻有一个完整的操作系统
- NORMA存储模型

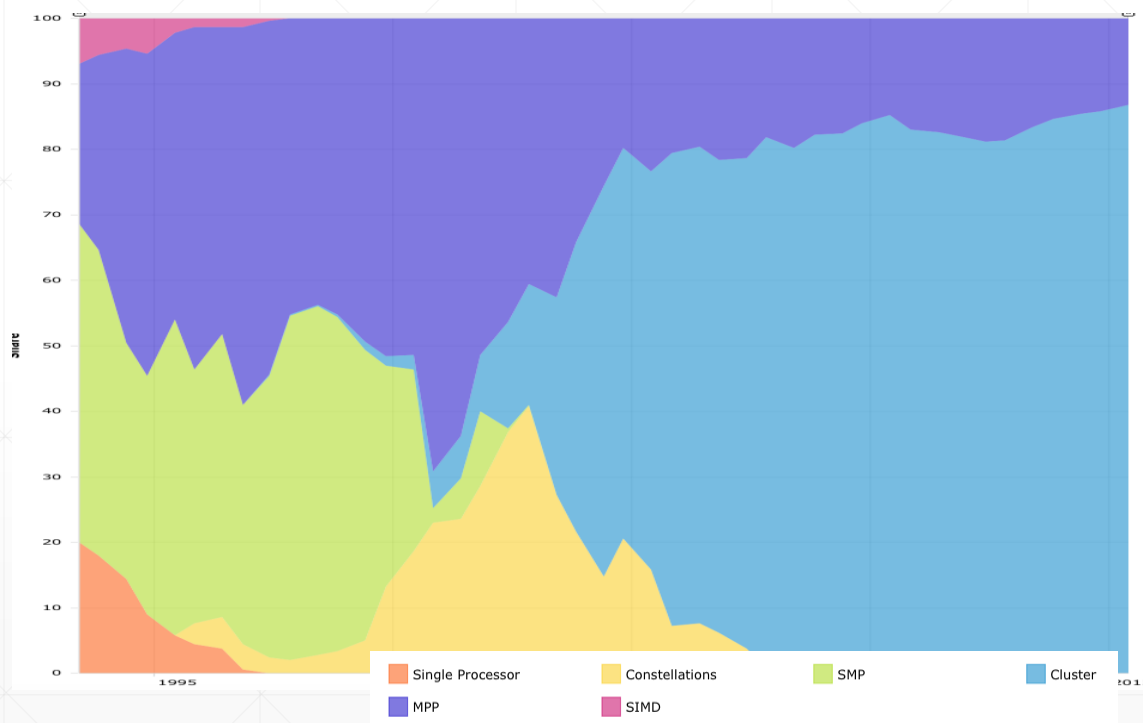


SMP/MPP/Cluster

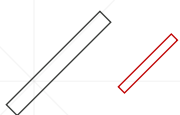
系统特征	SMP	MPP	机群
节点数量(N)	$\leq O(10)$	$O(100)-O(1000)$	$\leq O(100)$
节点复杂度	中粒度或细粒度	细粒度或中粒度	中粒度或粗粒度
节点间通信	共享存储器	消息传递 或共享变量（有DSM时）	消息传递
节点操作系统	1	N(微内核) 和1个主机OS(单一)	N(希望为同构)
支持单一系统映像 机群(Cluster)	永远	部分	希望
地址空间	单一	多或单一（有DSM时）	多个
作业调度	单一运行队列	主机上单一运行队列	协作多队列
网络协议	非标准	非标准	标准或非标准
可用性	通常较低	低到中	高可用或容错
性能/价格比	一般	一般	高
互连网络	总线/交叉开关	定制	商用

并行计算机系统体系架构

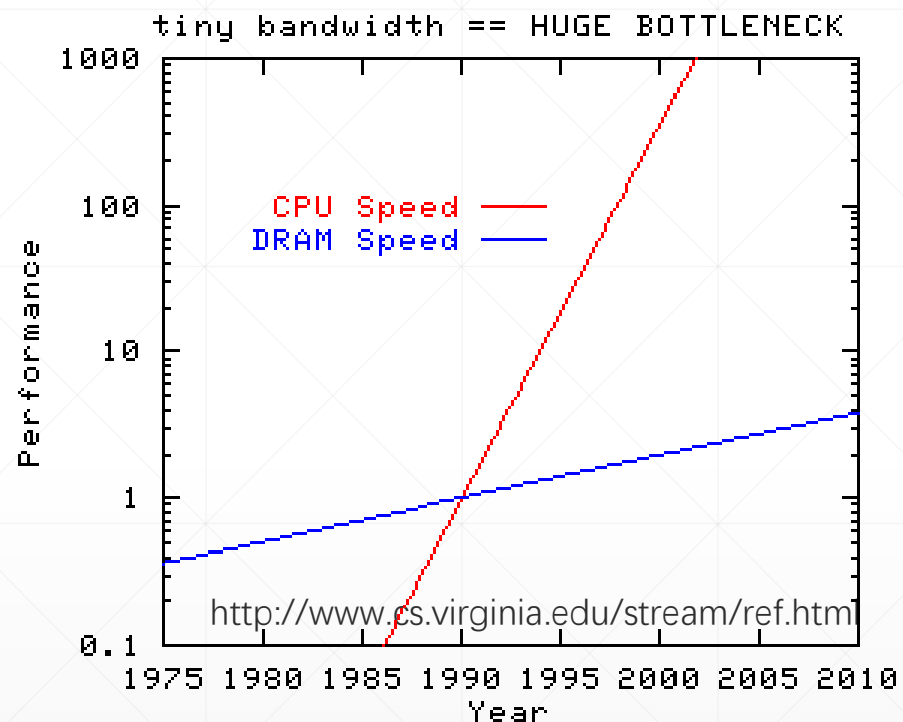
——Top500 并行计算机架构演变



- ✓ 2015年，仅存MPP和Cluster两种架构。
- ✓ 2015年6月的排名中，Cluster占86.8% (434/500)。
- ✓ Cluster容易扩展。MPP与Cluster相比，从机器占地、功耗和散热方面有优势。

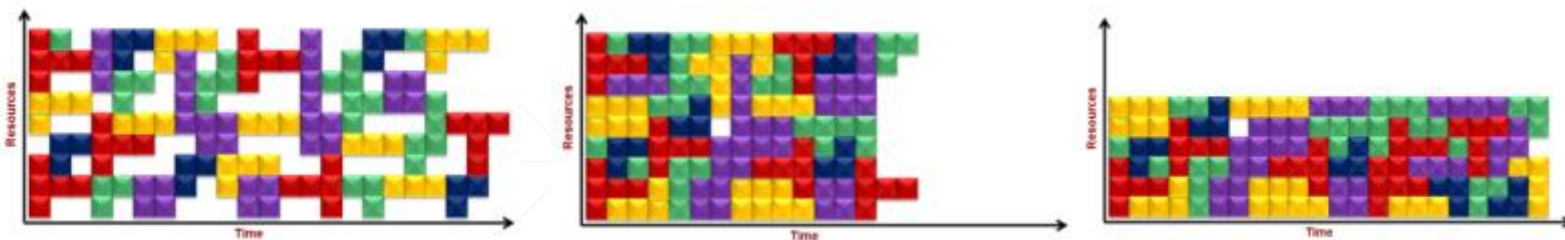


内存带宽

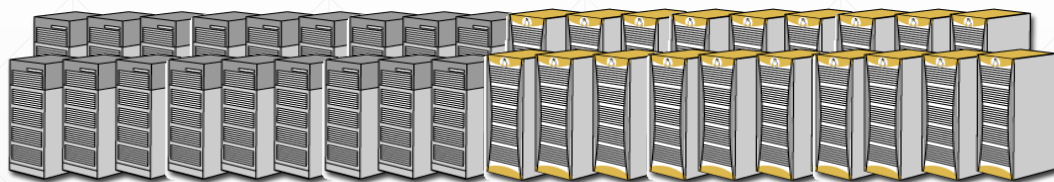


- 内存带宽的增长速度低于CPU的增长。
- 加速卡适合高内存带宽场景。

调度系统



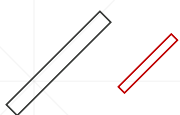
Virtualized view of compute, network and storage resources



PBS Professional Open Source Project



IBM Spectrum LSF



总结



北京超级云计算中心
Beijing Supercomputing Center

↑↑扫码参加IPCC ↓↓学习交流群



- ✓ 计算需求与日俱增
- ✓ 半导体工艺遇到功耗墙，不能无限制增加频率
- ✓ 多核多机系统，要求代码现代化/并行化
- ✓ 并行化带来技术挑战
- ✓ 需要合适的方法与工具辅助

2020

谢谢观看

时间：2020年8月

主讲人：刘帅