<<计算机体系结构>>

图书基本信息

书名：<<计算机体系结构>>

13位ISBN编号：9787115297655

10位ISBN编号：7115297657

出版时间：2012-12

出版时间：人民邮电出版社

作者：[美] John L. Hennessy,[美] David A. Patterson

页数：595

字数：1139000

译者：贾洪峰

前言

本书的目的　　本书到现在已经是第5个版本了，我们的目标一直没有改变，就是要阐述那些为未来技术发展奠定基础的基本原理。
计算机体系结构的各种发展机遇总是让我们激情澎湃，不曾有丝毫消退。
我们在第1版中就作出过如下的论述：“这个学科不是令人昏昏欲睡、百无一用的纸版模型。
绝对不是！
这是一个受到人们热切关注的学科，需要在市场竞争力与成本·性能·能耗之间作好权衡，从事这个学科既可能导致可怕的失败，也可能带来显赫的成功。
”　　在编写第1版时，我们的主要目的是希望改变人们原来学习和研究计算机体系结构的方式。
现今，我们感到这一目标依然正确，依然重要。
该领域日新月异，在对其进行研究时，必须采用真实计算机上的测量数据和真实示例，而不是去研究一大堆从来都不需要实现的定义和设计。
我们不仅热烈欢迎过去与我们结伴而行的老读者，同样也非常欢迎现在刚刚加入我们的新朋友。
不管怎样，我们都保证将采用同样的量化方法对真实系统进行分析。
　　和前几版一样，在编写这个新版本时，我们力争使其既适用于学习高级计算机体系结构与设计课程的学生，也适用于专业的工程师和架构师。
与第1版类似，这个版本重点介绍新平台（个人移动设备和仓库级计算机）和新体系结构（多核和GPU）。
这一版还秉承了前几版的做法，希望能够通过强调成本、性能、能耗之间的平衡和优秀的工程设计，揭去计算机体系结构的神秘面纱。
我们相信这一领域正在日趋成熟，发展成为一门具备严格量化基础的经典理工学科。
　　关于第5版　　我们曾经说过，第4版可能因为转向讨论多核芯片而成为自第1版以来的最重要版本。
但我们收到了这样的反馈意见：第4版已经失去了第1版重点突出的优点，它一视同仁地讨论所有内容，不分重点和场合。
我们非常确信，第5版不会再有这样的评价了。
　　我们相信，最令人激动的地方在于计算规模的两个极端：以移动电话和平板电脑之类的个人移动设备（PMD）为客户端，以提供云计算的仓库级计算机为服务器。
（具有敏锐观察力的读者可能已经看出本书封面上云计算的寓意。
）尽管这两个极端的规模大小不同，但它们在成本、性能和能效方面的共同主题给我们留下了深刻印象。
因此，每一章的讨论背景都是PMD和仓库级计算机的计算，第6章是全新的一章，专门讨论仓库级计算机。
　　本书的另一条主线是讨论并行的所有不同形式。
我们首先在第1章指出了两种应用级别的并行，一个是数据级并行（DLP），它的出现是因为有许多数据项允许同时对其进行操作；另一个是任务级并行（TLP），它的出现是因为创建了一些可以独立执行并在很大程度上并行的工作任务。
随后解释4种开发DLP和TLP的体系结构样式，分别是：第3章介绍的指令级并行（ILP），第4章介绍的向量体系结构和图形处理器（GPU），这一章是第5版新增加的内容；第5章介绍的线程级并行；第6章通过仓库级计算机介绍的需求级并行（RLP），这一章也是第5版中新增加的。
本书中，我们将存储器层次结构的内容提前到第2章，并将存储系统那一章改作附录D。
我们对第4章、第6章的内容尤为感到自豪，第4章对GPU的解读是目前最详尽、最清晰的，第6章首次公布了Google仓库级计算机的最新细节。
　　与前几版相同，本书前三个附录提供了有关MIPS指令集系统、存储器层次结构和流水线的基础知识，如果读者没有读过《计算机组成与设计》之类的书籍，可用作参考。
为了在降低成本的同时还能提供一些读者感兴趣的补充材料，我们在网络上提供了另外9个附录，网

址为：http：//booksite.mkp.com/9780123838728。
这些附录的页数之和比本书还要多呢！
　　这一版继续发扬“以真实示例演示概念”的传统，并增加了全新的“融会贯通”部分。
这一版中的“融会贯通”内容包括以下各服务器的流水线组成与存储器层次结构：ARM Cortex A8处理器、Intel core i7处理器、NVIDIA GTX-280和GTX-480 GPU，还有Google仓库级计算机。
　　主题的选择与组织　　和以前一样，我们在选择主题时采用了一种保守的方法，毕竟这个领域中值得讨论的思想实在太多了，不可能在这样一本主要讨论基本原理的书中将其全部涵盖在内。
我们没有面面俱到地分析读者可能遇到的所有体系结构，而是将重点放在那些在任何新计算机中都可能涉及的核心概念上。
根据一贯坚持的选材标准，本书讨论的思想都经过深入研究并已被成功应用，其内容足以采用量化方法进行讨论。
　　我们一直重点关注的内容都是无法从其他来源获取的同类资料，因此我们将继续尽可能讨论比较高级的内容。
事实上，本书介绍的有些系统，就无法在文献中找到相关描述。
如果读者需要了解更为基础的计算机体系结构知识，可以阅读《计算机组成与设计：硬件/软件接口》（Computer Organization and Design：The Hardware/Software Interface）一书。
　　内容概述　　这一版对第1章进行了补充，其中包括能耗、静态功率、动态功率、集成电路成本、可靠性和可用性的计算公式。
（封二上也列出了这些公式。
）在本书后续部分读者能够一直应用这些公式。
除了计算机设计与性能测量方面的经典量化原理之外，还对PIAT一节进行了升级，采用了新的SPECPower基准测试。
　　我们认为，与1990年相比，指令集体系结构扮演的角色有所弱化，所以我们把这一部分内容作为了附录A。
它仍然采用MIPS64体系结构。
（为便于快速查看，封三汇总了MIPS ISA相关信息。
）网站上的附录K介绍了10种RISC体系结构、80x86、DEC VAX和IBM 360/370，献给ISA爱好者们。
　　随后，我们在第2章开始讨论存储器层次结构，这是因为很容易针对这些内容应用成本·性能·功耗原理，而且存储器是其余各章的关键内容。
和上一版一样，附录B对缓存机制作了概述，以供读者需要时查阅。
第2章讨论了对缓存的10种高级优化方法。
这一章还介绍了虚拟机，它便于提供保护、进行软硬件管理，而且在云计算中也扮演着重要角色。
除了介绍SRAM和DRAM技术之外，这一章还包括了闪存的内容。
PIAT示例选择了PMD中使用的ARM Cortex A8和服务器中使用的Intel Core i7。
　　第3章主要研究高性能处理器中的指令级并行开发，包括超标量执行、分支预测、推理、动态调度和多线程。
前面曾经提到，附录C是关于流水线的一个综述，以备随时查阅之用。
第3章还研究了ILP的局限性。
和第2章一样，PIAT示例还是ARM Cortex A8和Intel Core i7。
第3版包括大量有关Itanium和VLIW的材料，现在这些内容放在网上的附录H中，这表明了我们的观点：这种体系结构未能达到过去所宣称的效果。
　　多媒体应用程序（比如游戏和视频处理）的重要性在提高，因此，开发数据级并行的体系结构也变得更为重要。
具体来说，越来越多的人在关注利用图形处理器（GPU）执行的运算，但很少有架构师了解GPU到底是如何工作的。
我们决定编写新的一章，主要就是为了揭开这种新型计算机体系结构的奥秘。
第4章开始介绍向量体系结构，对多媒体SIMD指令集扩展和GPU的解释就是以此为基础的。

（网站上的附录G深入地讨论了向量体系结构。
）GPU一节是本书最难写的部分，需要多次反复才能给出一个既精确又容易理解的描述。
一个重大挑战就是术语。
我们决定使用我们自己的术语，然后给出这些术语与NVIDIA官方术语之间的对应关系。
这一章介绍了Roofline性能模型，然后用它来对比Intel Core i7、NVIDIA GTX 280和GTX 480 GPU。
这一章还介绍了供PMD使用的Tegra 2 GPU。
第5章介绍多核处理器，探讨了对称、分布式存储器体系结构，考查了组织原理和性能。
接下来是有关同步和存储器一致性模型的主题，所采用的示例是Intel Core i7。
对片上互连网络感兴趣的读者可以阅读网站上的附录F，对更大规模多处理器和科学应用感兴趣的读者可以阅读网站上的附录I。
前面曾经提到，第6章介绍了计算机体系结构中的最新主题——仓库级计算机（Warehouse- Scale Computer，WCS）。
依靠Amazon Web服务部门和Google工程师的帮助，本章整合了有关WSC设计、成本与性能的详细资料，而以前了解这些内容的架构师寥寥无几。
在开始描述WSC的体系结构和物理实现（及成本）之前，首先介绍了MapReduce编程模型。
从成本的角度可以解释为什么会有云计算，以及为何在云中使用WSC进行计算的成本要低于在本地数据中心的计算成本。
PIAT实例是对Google WSC的描述，有些内容是首次公开的。
接下来就是附录A到附录L。
附录A介绍ISA的原理，包括MIPS64，附录K介绍Alpha、MIPS、PowerPC和SPARC的64位版本及其多媒体扩展。
其中还包括一些经典体系结构（80x86、VAX和IBM 360/370）和流行的嵌入指令集（ARM、Thumb、SuperH、MIPS16和Mitsubishi M32R）。
附录H与其相关，介绍了VLIW ISA的体系结构和编译器。
前面曾经提到，附录B和附录C是缓存与流水线基本概念的教程。
建议对缓存不够熟悉的读者在阅读第2章之前先阅读附录B，新接触流水线的读者在阅读第3章之前先阅读附录C。
附录D“存储系统”包括：进一步讨论可靠性和可用性，以RAID 6方案介绍为主体的RAID教程，非常珍贵的真实系统故障统计信息。
接下来介绍了排队理论和I/O性能基准测试。
我们评估了一个真实集群Internet Archive的成本、性能和可靠性。
“融会贯通”部分以NetApp FAS6000文件管理程序为例。
附录E由Thomas M. Conte撰写，汇总了嵌入式系统的相关内容。
附录F讨论网络互连，由Timothy M. Pinkston和Jos é Duato进行了修订。
附录G最初由Krste Asanovi · 撰写，其中详细介绍了向量处理器。
就我们所知，这两个附录是其各自相关主题的最好材料。
附录H详细介绍了VLIW和EPIC，也就是Itanium采用的体系结构。
附录I详细介绍了大规模共享存储器多处理方面用到的并行处理应用和一致性协议。
附录J由David Goldberg撰写，详细介绍了计算机算法。
附录L将第3版每一章中的“历史回顾与参考文献”部分集中在一起。
对于各章介绍的思想，它尽量给予一个恰当的评价，并让读者了解这些创造性思想背后的历史。
我们希望以此来展现人类在计算机设计方面的戏剧性发展过程。
这个附录还提供了一些参考文献，主修体系结构的学生可能会非常喜欢它们。
其中提到了本领域的一些经典论文，如果时间允许，建议读者阅读这些论文。
直接听原创者讲述他们的思想，在深受教育的同时，也是一种享受。
而“历史回顾”是以前版本中最受欢迎的章节之一。
内容导读　　所有读者都应当从第1章开始阅读，除此之外并不存在什么唯一的最佳顺序。

如果你不想阅读全部内容，可以参考下面这些顺序。
- 存储器层次结构：附录B、第2章、附录D。
- 指令级并行：附录C、第3章、附录H。
- 数据级并行：第4章、第6章、附录G。
- 线程级并行：第5章、附录F、附录I。
- 请求级并行：第6章。
- ISA：附录A、附录K。

附录E可以随时阅读，但在ISA和缓存序列之后阅读，效果可能会更好一些。
附录J可以在涉及运算时阅读。
附录L的各部分内容应当在读完正文中相应章节后阅读。

章节安排　　我们根据一种统一的框架安排内容，使各章在结构方面保持一致。
首先会介绍一章的主题思想，然后是“交叉问题”部分，说明本章介绍的思想与其他各章有什么相互关系。
接下来是“融会贯通”部分，通过展示如何在实际计算机中应用这些思想，将它们串在一起。

再下面是“谬论与易犯错误”，让读者从他人的错误中汲取教训。
我们将举例说明一些常见误解与体系结构陷阱，要避免犯错是非常困难的，哪怕你明明知道它们就在前面等着你。
“谬论与易犯错误”部分是本书最受欢迎的内容。
每一章都以一个“结语”节结束。

案例研究与练习　　每一章的最后都有案例研究和练习。
这些案例研究由业内和学术界的专家编撰而成，通过难度逐渐增大的练习来探讨该章的关键概念，检验读者的理解程度。
教师们会发现这些案例研究都非常详尽和完善，完全可以针对它们设计出一些练习。

每个练习中用尖括号括起的内容（）指明了做这道题应该阅读哪部分正文内容。
我们这样做的目的，一方面是为了提供复习内容，另一方面是希望帮助读者避免在还没有阅读相应正文的情况下去做一些练习。
为了使读者大致了解完成一道题需要多长时间，我们为这些练习划定了不同等级：　　[10] 短于5分钟（阅读和理解时间）；　　[15] 5~15分钟给出完整答案；　　[20] 15~20分钟给出完整答案；　　[25] 在1小时内给出完整的书面答案；　　[30] 小型编程项目：时间短于1整天；　　[40] 大型编程项目：耗时2周；　　[讨论] 与他人一起讨论的主题。

在textbooks.elsevier.com注册的老师可以得到案例研究与习题的解答。
我们会定期补充新材料和网上其他可用资源的链接。

帮助改进本书　　如果你阅读后面的“致谢”部分，将会看到我们已经下了很大的功夫来纠正错误。
由于一本书会进行多次印刷，所以我们有机会进行更多的校订。
如果你发现了任何遗留错误，请通过电子邮件联系出版商。

结语　　本书仍然是一本真正的合著作品，我们每人编写的章节和附录各占一半。
如果没有对方完成另一半工作，如果没有对方在任务似乎无望完成时给予鼓励，如果没有对方点透某个难以表述的复杂概念，如果没有对方花费周末时间来审阅书稿，又如果没有对方在自己因为其他繁重职责而难以提笔时给予宽慰（从简历可以看出，这些职责是随着本书的版本号以指数形式增加的），我们无法想象这本书要花费多长时间才能完成。
当然，对于你将要读到的内容，其中若有不当之处，我们也负有同等责任。

John Hennessy　　David Patterson

<<计算机体系结构>>

内容概要

《计算机体系结构：量化研究方法(第5版)》是最权威的计算机体系结构著作，是久负盛名的经典作品。
书中系统地介绍了计算机系统的设计基础、指令集系统结构、流水线和指令集并行技术、层次化存储系统与存储设备、互连网络以及多处理器系统等重要内容。
在这个最新版中，作者增加了当前炙手可热的云计算和手机客户端技术等相关内容，探讨了在手机、平板电脑、笔记本电脑和其他移动计算设备上云计算的软硬件实现方式。
《计算机体系结构：量化研究方法(第5版)》可作为高等院校计算机专业本科生或研究生教材，也可作为从事计算机体系结构或计算机系统设计的工程技术人员的参考书。

作者简介

John L. Hennessy
是斯坦福大学的第10任校长，从1977年开始在该校电子工程与计算机系任教。
Hennessy是IEEE和ACM会士，美国国家工程院、国家科学院和美国哲学院院士，美国艺术与科学院院士。
他获得过众多奖项，如2001年度Eckert-Mauchly奖，表彰他对RISC技术的贡献；2001年度Seymour Cray计算机工程奖；与David
Patterson共同获得的2000年度约翰?冯?诺依曼奖章。
他还拥有7个荣誉博士学位。

1981年，John L.
Hennessy带领几位研究生在斯坦福开始MIPS项目的研究。
1984年完成该项目之后，他暂时离开大学，与他人共同筹建MIPS计算机系统公司（也就是现在的MIPS技术公司），这家公司开发了最早的商用RISC微处理器之一。
到2006年，已经有20多亿个MIPS微处理器被用于视频游戏、掌上电脑、激光打印机和网络交换机等各种设备中。
Hennessy后来领导了DASH（Director
Architecture for Shared
Memory，共享存储器控制体系结构）项目，这一项目设计了第一个可扩展缓存一致性多处理器原型，其中的许多重要思想都在现代多处理器中得到了应用。
除了参与科研活动、履行学校职责之外，他仍作为前期顾问和投资者参与了无数的创业项目。

David A.
Patterson自1977年进入加州大学伯克利分校执教以来，一直讲授计算机体系结构课程，拥有该校计算机科学Pardee讲座教授职位。
他因为教学成果显著而荣获了加州大学的杰出教学奖、ACM的Karlstrom奖、IEEE的Mulligan教育奖章和本科生教学奖。
因为在RISC方面的贡献而获得了IEEE技术成就奖和ACM
Eckert-Mauchly奖，他还因为在RAID方面的贡献而分享了IEEE Johnson信息存储奖，并与John
Hennessy共同获得了IEEE约翰?冯?诺依曼奖章和C & C奖金。
和John
Hennessy相似，Patterson也是美国艺术与科学院院士、美国计算机历史博物馆院士、ACM和IEEE会士。
他还被选入美国国家工程院、美国国家科学院和硅谷工程名人堂。
Patterson身为美国总统信息技术顾问委员会委员，同时也是伯克利电子工程与计算机科学系计算机科学分部主任、计算机研究协会主席和ACM主席。
这一履历使他荣获了ACM和CRA颁发的杰出服务奖。

在加州大学伯克利分校，Patterson领导了RISCI的设计与实现工作，这可能是第一台VLSI精简指令集计算机，为商业SPARC体系结构奠定了基础。
他曾是廉价磁盘冗余阵列（Redundant
Arrays of Inexpensive
Disks，RAID）项目的领导者之一，正是由于这一项目，才有了后来许多公司出品的可靠存储系统。
他还参与了工作站网络（Network
of
Workstations，NOW）项目，因为这一项目而有了因特网公司使用的集群技术和后来的云计算。

这些项目获得了ACM颁发的三个论文奖。
作为“ 算法-机器-人类 ”（AMP）实验室和并行计算实验室的主管，他目前在这里开展自己的研究项目。
AMP实验室的目标是开发可扩展的机器学习算法、适用于仓库级计算机的编程模型、能够快速洞悉云中海量数据的众包（Crowd-Sourcing）工具。
并行计算实验室的目标是研发先进技术，为并行个人移动设备提供可扩展、可移植、方便快捷的效率软件。

<<计算机体系结构>>

书籍目录

章节摘录

版权页： 插图： 遗憾的是，在对比计算机性能时并非总是使用时间这一度量标准。
我们的观点是：唯一稳定、可靠的性能度量就是实际程序的执行时间，以任意其他度量代替时间或者以任意其他被测项目代替实际程序，最终都会在计算机设计中产生误导，甚至是错误。
即使是执行时间，也可以根据我们的测量内容采用不同的定义方式。
最直接的时间定义被称为挂钟时间，响应时间或已用时间，也就是完成一项任务的延迟，包括磁盘访问、存储器访问、输人／输出活动、操作系统开销等所有相关时间。
在同时运行多个程序的情况下，处理器在等待I／O时处理另一个程序，不一定使某一程序的已用时间缩至最短。
因此，我们需要有一个术语来表达这一行为。
CPU时间可以区分这种不同，它是指处理器执行计算的时间，不包括等待I／O或运行其他程序的时间。
（显然，用户观测到的响应时间是程序的已用时间，而不是CPU时间。
） 那些定期运行相同程序的计算机用户当然是评估新计算机性能的最佳候选人。
如果他们要评估一个新系统的性能，只需要比较其工作负载的执行时间就行了（其工作负载就是在计算机上运行的程序和操作系统命令）。
但很少有用户具备这种得天独厚的条件。
大多数用户必须依赖其他方法来评价计算机的性能，还经常需要使用评估软件，希望这些方法能够预测自己在使用新计算机时的性能。
1.8.1 基准测试 测量性能的最佳基准测试方法就是采用实际应用程序，比如1.1节的Google Goggles。
人们曾经尝试运行一些远比实际应用程序简单的程序，但这种做法已经导致了性能隐患。
这些简单程序的示例包括：程序内核，即实际应用程序中的短小、关键部分；玩具程序，为了完成编程入门作业而编写的小程序，通常不超过100行，比如快速排序；合成基准测试程序，为了匹配实际应用程序的特征和行为而编写的虚拟程序，比如Dhrystone。
今天，所有这三种方法都没有什么好名声，主要是因为编译器的编写人员和架构师可以串通起来，使计算机在执行这些替代程序时能够比运行实际应用程序时显得更快一些。
令本书作者感到沮丧的是，合成程序Dhrystone仍然是应用最为广泛的嵌入式处理器基准测试程序！

媒体关注与评论

“《计算机体系结构：量化研究方法（第5版）》继续发扬传统，为学习计算机体系结构的学生提供了当前计算平台的最新信息，使他们能够洞悉体系结构，便于设计未来系统。
这一版的亮点在于大幅修订了数据级并行那一章，用传统的体系结构术语清晰地解读了GPU体系结构。
”　　——Krste Asanovi·，加州大学伯克利分校　　“《计算机体系结构：量化研究方法》是一部经典，犹如美酒，历久而弥醇。
我在本科毕业时第一次购买了本书，它到现在仍然是我最经常参考的书籍之一。
第4版问世时，我发现其中包含了如此之多的新材料，为了跟上这一领域的最新趋势，我必须得再买一本。
而当审阅第5版时，我发现Hennessy和Patterson再现神奇。
全书内容都进行了大量更新，对于希望真正理解云和仓库级计算的人们来说，单凭第6章的内容，这个新版本就值得一读。
只有Hennessy和Patterson才可能接触到谷歌、微软等云计算与互联网规模的应用提供商的内部人士，对这一重要领域，业内的介绍材料无出其右。
”　　——James Hamilton　　“Hennessy和Patterson撰写本书的第一版时，研究生们是在用50 000个晶体管组装计算机。
今天，仓库级的计算机集群会包含50 000个服务器，每个服务器中包含数十个处理器和数十亿个晶体管。
计算机体系结构一直在不停地快速发展，而《计算机体系结构：量化研究方法》紧跟它的步伐，每个版本都准确地解释和分析了这一领域激动人心的最新重要思想。
”　　——James Larus，微软研究院　　“这一版新增加了一章非常丰富的内容，用来讨论向量、SIMD和GPU体系结构中的数据级并行技术。
它解释了应用于大众市场的GPU内部的关键体系结构概念，给出这些概念与传统术语的对应关系，并与向量和SIMD体系结构进行了对比。
这一内容非常及时，与业内转向GPU并行计算的潮流相适应。
《计算机体系结构：量化研究方法》继续独领风骚，全面地介绍了体系结构方面的重大新进展！
”　　——John Nickolls，NVIDIA　　“本书已经成为一本经典教科书了，这一版突出介绍了各种显式并行技术（数据、线程、请求）的兴起，各用整整一章来描述。
数据并行一章尤为夺目：通过向量SIMD、指令级SIMD和GPU的对比，避开每种体系结构的专用术语，揭示了这些体系结构之间的相似与区别。
”　　——Kunle Olukotun，斯坦福大学　　“《计算机体系结构：量化研究方法（第5版）》探讨了各种并行概念和它们各自的技术权衡。
和过去的几个版本一样，这一新版本中同样涵盖了最新的技术发展趋势。
两个重点是个人移动设备（PMD）和仓库级计算（WSC）的爆炸性增长——与原来一味追求性能相比，这里的焦点已经转为更全面地寻求性能与能效之间的平衡。
这些趋势刺激了人们不断追求更强劲的处理能力，而这种追求又推动人们在并行道路上走得更远。
”　　——Andrew N. Sloss，实施顾问，ARM公司 ARM System Developer’s Guide一书的作者

<<计算机体系结构>>

编辑推荐

《计算机体系结构:量化研究方法(第5版)》可作为高等院校计算机专业本科生或研究生教材，也可作为从事计算机体系结构或计算机系统设计的工程技术人员的参考书。

名人推荐

“本书之所以会成为不朽的经典之作，是因为每次再版都不仅仅是一次更新补充，而是全面修订，针对这个激动人心且快速变化的领域，给予我们最及时的信息和权威的解读。
即便对于我这样已从业二十多年的人来说，再次阅读本书时，依旧自觉学无止境，感佩于两位卓越大师的渊博学识和深厚功底。
”——Luiz Andre Barroso，Google公司“《计算机体系结构：量化研究方法》是一部经典，犹如美酒，历久而弥醇。
只有Hennessy和Patterson才可能接触到谷歌、亚马逊、微软等云计算与互联网规模的应用提供商的内部人士，对这一重要领域，业内的介绍材料无出其右。
”——James Hamilton，Amazon web服务部“《计算机体系结构：量化研究方法（第5版）》继续发扬传统，为学习计算机体系结构的学生提供了当前计算平台的最新信息，使他们能够洞悉体系结构，便于设计未来系统。
这一版的亮点在于大幅修订了数据级并行那一章，用传统的体系结构术语清晰地解读了GPU体系结构。
”——Krste Asanovic，加州大学伯克利分校“Hennessy和Patterson撰写本书的第一版时，研究生们是在用50000个晶体管组装计算机。
今天，仓库级的计算机集群会包含50000个服务器，每个服务器中包含数十个处理器和数十亿个晶体管。
计算机体系结构一直在不停地快速发展，而《计算机体系结构：量化研究方法》紧跟它的步伐，每个版本都准确地解释和分析了这一领域激动人心的最新重要思想。
”——James Larus，微软研究院

<<计算机体系结构>>

版权说明