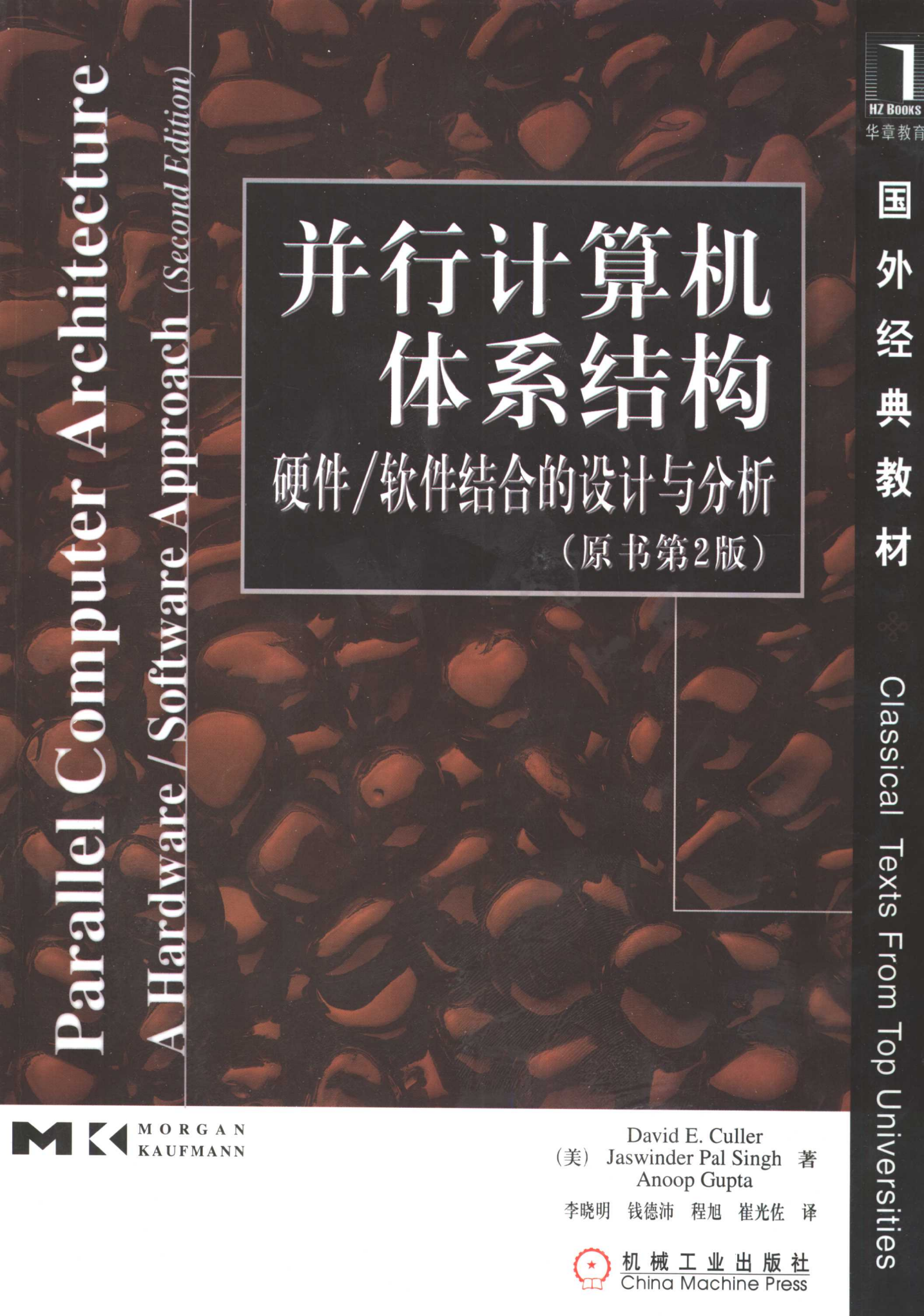
Parallel Computer Architecture
A Hardware / Software Approach (Second Edition)
并行计算机体系结构
硬件/软件结合的设计与分析
（原书第2版）
HZ BOOKS
华章教育
国外经典教材
Classical Texts From Top Universities
MORGAN KAUFMANN
David E. Culler
(美) Jaswinder Pal Singh 著
Anoop Gupta
李晓明 钱德沛 程旭 崔光佐 译
机械工业出版社
China Machine Press

HZ BOOKS
华章教育
北京华章图文信息有限公司

（原书第2版）

并行计算机体系结构

硬件/软件结合的设计与分析

Parallel Computer Architecture

A Hardware/Software Approach

(Second Edition)

David E. Culler
（美） Jaswinder Pal Singh 著
Anoop Gupta

李晓明 钱德沛 程旭 崔光佐 译

机械工业出版社
China Machine Press

Parallel Computer Architecture: A Hardware/Software Approach, Second Edition
David E. Culler, Jaswinder Pal Singh and Anoop Gupta
ISBN: 1-55860-343-3

ISBN 981-4134-95-3

Elsevier Science (Singapore) Pte Ltd.
3 Killiney Road
#08-01 Winsland Hose I
Singapore 239519
Tel: (65) 6349-0200
Fax:(65) 6733-1817
First Published 2003
2003年初版

本书版权登记字：图字：01-2000-1859

图书在版编目（CIP）数据

并行计算机体系结构：硬件/软件结合的设计与分析（原书第2版）/（美）卡勒（Culler, D. E.）等著；李晓明等译. -北京：机械工业出版社，2002.10
（国外经典教材）
书名原文：Parallel Computer Architecture: A Hardware/Software Approach, Second Edition
ISBN 7-111-07888-8

Ⅰ.并… Ⅱ.①卡…②李… Ⅲ.并行计算机－计算机体系结构－教材 Ⅳ.TP338.6

中国版本图书馆CIP数据核字（2002）第008129号

机械工业出版社（北京市西城区百万庄大街22号 邮政编码 100037）
责任编辑：温丹丹
北京第二外国语学院印刷厂印刷·新华书店北京发行所发行
2003年1月第1版第1次印刷
787mm×1092mm 1/16·50.25印张
印数：0 001-5 000册
定价：78.00元

出版者的话

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭橥了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短、从业人员较少的现状下，美国等发达国家在其计算机科学发展的几十年间积淀的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章图文信息有限公司较早意识“出版要为教育服务”。自 1998 年始，华章公司就将工作重点放在了遴选、移译国外优秀教材上。经过几年的不懈努力，我们与 Prentice Hall，Addison-Wesley，McGraw-Hill，Morgan Kaufmann 等世界著名出版公司建立了良好的合作关系，从它们现有的数百种教材中甄选出 Tanenbaum，Stroustrup，Kernighan，Jim Gray 等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及庋藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专诚为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍，为进一步推广与发展打下了坚实的基础。

随着学科建设的初步完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都步入一个新的阶段。为此，华章公司将加大引进教材的力度，在“华章教育”的总规划之下出版三个系列的计算机教材：针对本科生的核心课程，剔抉外版菁华而成“国外经典教材”系列；对影印版的教材，则单独开辟出“经典原版书库”；定位在高级教程和专业参考的“计算机科学丛书”还将保持原来的风格，继续出版新的品种。为了保证这三套丛书的权威性，同时也为了更好地为学校和老师们服务，华章公司聘请了中国科学院、北京大学、清华大学、国防科技大学、复旦大学、上海交通大学、南京大学、浙江大学、中国科技大学、哈尔滨工业大学、西安交通大学、中国人民大学、北京航空航天大学、北京邮电大学、中山大学、解放军理工大学、郑州大学、湖北工学院、中国国家信息安全测评认证中心等国内重点大学和科研机构在计算机的各个领域的著名学者组成“专家指导委员会”，为我们提供选题意见和出版监督。

“国外经典教材”是响应教育部提出的使用外版教材的号召，为国内高校的计算机本科教学度身订造的。在广泛地征求并听取丛书的“专家指导委员会”的意见后，我们最终选定了这20多种篇幅内容适度、讲解鞭辟入里的教材，其中的大部分已经被M.I.T.、Stanford、U.C. Berkley、C.M.U.等世界名牌大学采用。丛书不仅涵盖了程序设计、数据结构、操作系统、计算机体系结构、数据库、编译原理、软件工程、图形学、通信与网络、离散数学等国内大学计算机专业普遍开设的核心课程，而且各具特色——有的出自语言设计者之手、有的历三十年而不衰、有的已被全世界的几百所高校采用。在这些圆熟通博的名师大作的指引之下，读者必将在计算机科学的宫殿中由登堂而入室。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证，但我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。教材的出版只是我们的后续服务的起点。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方法如下：

电子邮件：hzedu@hzbook.com
联系电话：(010) 68995265
联系地址：北京市西城区百万庄南街1号
邮政编码：100037

专家指导委员会

（按姓氏笔画顺序）

尤晋元	王　珊	冯博琴	史忠植	史美林
石教英	吕　建	孙玉芳	吴世忠	吴时霖
张立昂	李伟琴	李师贤	李建中	杨冬青
邵维忠	陆丽娜	陆鑫达	陈向群	周伯生
周克定	周傲英	孟小峰	岳丽华	范　明
郑国梁	施伯乐	钟玉琢	唐世渭	袁崇义
高传善	梅　宏	程　旭	程时端	谢希仁
裘宗燕	戴　葵			

序

十分高兴能为这本关于并行计算的新书作序。本书的内容令人兴奋，具有鲜明的时代特征。作者富有见解的思路，以及对各种计算机体系结构所做的系统化和定量化分析，将这本书与先前所有关于并行计算机体系结构的书区别开来。这种在开头4章里奠定的基本思路有三个主要的创新：它建立在近年来并行计算机体系结构的融合之上；它以应用作为驱动来评价和分析体系结构；它植根于一种坚实的性能评价方法论之中。

如同在第1章里所描述的一样，近年来共享存储和消息传递模式的融合，展现了一种在一个公共的框架下刻画和分析计算机体系结构的新机遇。基于这种融合，作者描述了4种根本的设计要点（对通信的抽象、程序设计模型、通信和复制的关系、性能的追求）。这种做法创建了一个用以讨论各种体系结构和具体实现的框架。在这个框架中，我们可以对不同体系结构的做法进行严格的比较和考察。

不理解应用和体系结构的相互作用，我们就不能理解多处理器的设计权衡和性能。为此，针对性能的提高，第2、3章介绍了应用程序并行化的过程以及一组并行程序。除了铺垫一个定量性评价体系结构及其实现的基础，这几章勾画了并行程序设计的过程和它带来的挑战，它们对于理解多处理器的性能是关键的。通过展示如何用一种并行的工作负载来评价体系结构，第4章进一步阐明了上述观念。作者还描述了评价并行计算机的复杂性，包括由机器规模和工作负载的扩展所带来的若干问题。这3章一起形成了其余各章论述的基础。

中小规模共享存储系统是目前的主导并行体系结构。任何对并行计算感兴趣的人，理解这类机器的原理和设计权衡是至关重要的。第5章描述了共享存储多重处理技术的主要概念：高速缓存的一致性、存储同一性和同步。然后作者描述了基于侦听的共享存储多处理器的详细设计，包括第6章两个详尽的案例分析。

设计多处理器系统，使之能够扩展到更多的节点，依然是多处理器体系结构领域最具有挑战性和争议的一个方面。第7章专门从消息传递到共享存储的设计领域讨论这个问题。第8章将这种讨论延伸，考察了目录方案的使用，它使得缓存一致性能够扩展到大量的处理节点。第8章还讨论了基于目录的一致性的基本问题，两个详尽的案例分析形成了本章的核心。在本书之前，针对基于目录的高速缓存一致性技术的商品计算机，还没有如此详细的和定量的分析。

用于多处理器系统的某些最重要的硬件和软件技术基本上是独立于体系结构细节的。因此，作者用3章的篇幅探讨了这些关键技术。第9章描述了存储系统中软件的影响、硬件的需求以及性能权衡，包括同一性问题和高速缓存的扩展使用。第10章考察了互连技术，这是任何多处理器系统都要有的一个关键组成部分。最后，第11章考察了用于时延包容的技术。从多个方面来讲，这是并行计算机的一个“普适”的设计问题。

作为结束，本书给出了关于未来硬件和软件挑战的一个富有见识的讨论。首先，作者讨

论了可能的发展情景，包括硬件和软件两个方面。然后作者在题为“遇到困难”的两段文字中转向潜在的障碍。最后，他们考察了可能的技术突破！我发现最后一章既有启发性，也有思想性。作者们不同的背景和互补的优势使得这一章既有洞察力，又有刺激性。

总而言之，这是一个关于多处理器系统设计领域的令人兴奋并且反映时代特征的探索性成果。体系结构设计方法融合的趋势同作者提出的框架的结合，使得我们可能建立一种公共的基础来考察多样化的现代并行体系结构。几年前，由于各种体系结构的设计方法相差太远，要写这样一本书是不可能的。类似地，如果没有注意定量的性能评测以及应用和体系结构的相互作用，这本书也就不会这么突出。而我们的作者恰好利用了这种技术的融合，并将注意力集中在应用驱动和针对性能的分析上，对并行计算机体系结构领域进行了一次独特的、富有见识的探索。这种做法，和作者们独特的优势和经验结合在一起，产生了这部论著，同其他任何并行体系结构的书相比，它体现了更深刻的认识。我向作者们表示祝贺，并向所有对并行处理技术的理论和实践，以及这些技术的现在和未来感兴趣的读者推荐此书。

John L. Hennessy，斯坦福大学腓特烈·埃蒙·特曼工程学院院长

译 者 序

翻译这本书的念头，源于1998年11月在美国弗罗里达州奥兰多市参加超级计算（Supercomputing）学术会议。当时，摩根考夫曼（Morgan Kaufmann）出版社的工作人员在现场展销图书。本书是其中之一。粗粗浏览，立刻被书中新颖的内容所吸引，当即买下一本。回过头看看，除了对上述印象有进一步的加深外，还发现它和以前类似题材的教材相比，语言也很有特色：在陈述种种技术要点时，还展开了大量的剖析性分析。我们感到，本书能使读者对技术的脉络有更深刻的把握。于是，在机械工业出版社华章公司的支持下，我们开始了本书的翻译工作。

这项工作基本上分为三个阶段。第一阶段，由李晓明负责翻译序、前言、第2、3、5、6章和附录，钱德沛翻译第4、9、10章，程旭翻译第1、7、8章，崔光佐翻译第11、12章。在统一了一些名词的译法后，在第二阶段由李晓明修改序、前言、第1、2、3、5、6章和附录，钱德沛修改第4、7、8、9、10章，崔光佐依然修改第11、12章。在最后审定阶段由李晓明负责第4、7、8、9、10、11、12章，钱德沛负责序、前言、第1、2、3、5、6章和附录。清样出来后主要由李晓明和钱德沛负责最后的校对。

尽管我们几个人近年来一直在从事和计算机体系结构有关的教学和科研工作，但坦率地说，做好这件事对我们依然是很难的。一方面，本书的内容十分广博，有一些是我们以前了解不多的；另一方面，本书的写作风格倾向于口语化，描述性强，因此在翻译过程中既要准确地表达其技术含义，又要尽量兼顾其表述风格，对我们是个很大的挑战。在一些难以兼顾的场合，我们采用了“译者注”的方式来处理，即在行文中基本遵照原文句子的结构，但会对其技术含义作进一步的解释。另外，在翻译过程中我们作了几次词汇译法的讨论和统一，在尽量采用国内有关计算机术语常用译法的同时，也对极少数术语作了我们认为更合适的处理。最典型的情况有两个，一是有关高速缓存访问的“miss”，以前常用的是“失效”，我们这里统一译成“扑空”；另一个是在讨论共享存储系统时常用的“coherence”和“consistency”，目前国内的译法都是“一致性”，但在本书中它们多次同时出现，翻译起来有明显困难，因此我们分别译成“一致性”和“同一性”。还有一个特别的词汇scalability（scale，scalable）。这是目前几乎到处都在用的词，基本含义是当系统规模扩大时其典型应用性能改善程度的度量，具体指在保证性能的条件下，典型应用的规模随计算系统的规模扩大而扩大的程度，隐含有两种因素（系统规模，应用性能）按比例变化的含义。在本书的翻译中，大多数场合采用了常用的“可扩展性”，但在需要强调其按比例变化含义的地方，有时也译成了“可扩放性”。最后，在翻译过程中我们得到了原著作者提供的勘误表，其内容也反映在此译本中。

历经近三年的时间，《*Parallel Computer Architecture: A Hardware/Software Approach, Second Edition*》中译本终于出版了，这其中还有不少人参加了相关的工作。北京大学和西安交大一些同学参加了本书翻译的一些前期工作和后期索引整理的工作，北京大学的黄蕊在文字校对中花了许多时间，为我们纠正了不少欠妥之处，在此我们一并表示感谢。

尽管如此，这个译本一定还会存在不少缺点、疏漏和蹩脚之处。我们欢迎读者指出问题，提出建议。为此，我们计划建立并维护一个相关网站 http：//lxm.cs.pku.edu.cn/pca/来反映读者的意见和建议，在上面提供相关章节和段落的更新或修改信息，以弥补现时由于我们的水平和时间所限可能造成的疏误。

译　者

2002 年 1 月

前　言

并行计算已经成为20世纪90年代计算技术的一个至关重要的组成部分。在接下来的20年里，它所产生的影响将有可能和微处理器在前20年里产生的影响相当。事实上，这两种技术有着深刻的联系，高度集成的微处理器和存储器芯片的发展使得多处理器系统日益具有更大的吸引力。多处理器系统已经代表着几乎所有计算市场层面的高性能端部分，从最快的超级计算机、最大的数据中心到部门级服务器，到单个的台式机。若干PC、工作站，甚至多处理器系统紧密集成起来，所形成的机群正作为可扩展因特网服务器出现。过去，计算机厂家在它们的产品系列中利用不同的技术和处理器体系结构来满足不断增加的系统性能要求。今天，产品中处处用到的都是相同的微处理器。在一个相当大的范围内，性能提高的基本手段是增加处理器的个数。这种性能扩展的经济效益使其极具吸引力。很快地，若干处理器将会集在一个芯片上，多处理器系统将会比今天更加流行。

尽管并行计算的学术历史在时间上不算短，在内容上也很丰富，但从根本上改变这一学科现状的是和商用技术的紧密结合。对新颖体系结构和特殊技术的强调已经让位于定量的分析，让位于在相同处理节点上实现不同的程序设计模型，让位于仔细的工程性权衡。我们写此书的目的就是让设计人员掌握这一类新型的多处理器系统的设计，从中小规模的并行台式计算机到高度并行的信息服务器和超级计算机，使他们理解根本的体系结构和软件问题，以及在设计中进行权衡的相关技术。同时，我们也希望为软件系统和应用的设计人员展示体系结构的可能发展方向，那些将决定硬件设计特定路线的动力，以及那些发展对面向性能程序设计上的影响。

近年来，在并行计算机体系结构领域最令人激动的进展是传统上完全不同的各种做法的融合，把共享存储、消息传递、数据并行和数据驱动等的计算都融合在一种公共的机器结构上。这种融合的原因部分在于共同的技术和经济力量的驱动，部分在于对并行软件的更好理解。它使得我们能开发一种公共的框架，在其中来理解和评估体系结构方面的权衡，而不是将注意力集中在各种奇特的设计和分类法上。再者，流行的并行程序设计模型在许多机器类型上都适用，这使得并行程序设计更加可移植，从而使我们得以发展有意义的标准测试和评估方法。这种领域的成熟使得硬件和软件相互作用的定量和定性的分析研究成为可能。事实上，领域的发展本身也要求我们这样做。针对一组对所有并行体系结构都很关键且涉及现代系统设计整个范围的基本问题——数据访问、通信性能、协同工作、有用语义的正确实现等。本书给出了旨在解决它们的硬件和软件技术，并考察了各种技术是如何相互作用的。仔细选择的、深入的案例分析提供了一种关于一般性原理的具体说明，展示了不同机制间的具体相互作用。

写这本书的动机之一是由于缺乏一种足够好的教材，供我们在伯克利、普林斯顿、斯坦福等校教学使用。有些教材以一种泛泛的方式将材料呈现出来，综述各种体系结构和研究成果，但没有深入分析它们，也没有提供一种现代工程的框架。另外一些教材集中在专门的项目上，但没有介绍在各种不同设计方案中体现的基本原理。在这个领域的研究报告提供了大

量想法和试验数据，但没能够提炼到一种有机构成的境界。在技术和体系结构融合的背景下，通过将注意力集中在最重要的问题上，而不是集中在将我们带到如今的丰富多彩的历史中，我们希望能够提供一种关于这个激动人心且迅速变化领域的更深刻、更清晰的理解。这是一个协作努力的结果，它反映在本书封面上我们的名字次序上。

本书的读者

本书的内容对多方面的读者都是很重要的，包括在计算机体系结构、系统软件和应用领域工作的研究人员、学生和工程技术人员。鉴于多处理器系统日益提高的重要性，这些内容和计算机系统结构设计师的相关性是明显的。芯片设计者必须理解什么能成为一个多处理器系统的有效基本模块。支配总线和存储系统设计的往往也是一些和并行性相关的问题。I/O系统的设计必须考虑具有可扩展性的高速网络、机群的构成，以及那些被多个处理器共享的设备。

系统软件——包括操作系统、编译器、程序设计语言、运行系统、性能调试工具——需要考虑新的情况，也将在并行计算机中获得新的发展机会。这样，理解体系结构的演化以及那些导致这种演化的力量是很重要的。在编译器和程序设计语言的研究与开发方面，针对并行计算的工作已经有相当一段时间。然而，体系结构和商用技术新的融合也许意味着编译和语言问题应该得到重新审视，需要在一个更一般的背景下讨论。硬件、操作系统和用户程序之间的传统边界也正在并行计算的意义下变化，为了更好的性能，程序经常要有对资源更直接的控制。

应用领域，诸如计算机图形学和多媒体、科学计算、计算机辅助设计、数据库、决策支持和事务处理，都可能出现一种巨大的转变。这种转变将是廉价的并行计算能够提供强大的计算能力的结果。然而，开发健壮的并行应用，在当前和未来多处理器系统上都能表现出好的加速比，是一个挑战性任务，而且它要求对系统相互作用和体系结构发展方向有深刻的理解。本书试图提供这样一种理解，促进应用领域和计算机系统结构之间的交流，从而使我们能设计出更好的体系结构——使程序设计更容易，性能更高和更健壮。

本书的组织

本书共有12章。第1章给出了并行体系结构的一个概貌。根据当前在工艺、体系结构和应用方面的趋势，它首先讨论了为什么并行计算机系统越来越重要。它简要介绍了那些影响了这个领域的各种多处理器体系结构（共享存储、消息传递、数据并行、数据流和脉动阵列），展现了工艺和体系结构的发展趋势是如何导致了一种领域的共识，即并行计算机系统应是由一种通信体系结构互连起来的一组通用处理节点。这种融合并不意味着创新的结束，恰恰相反，我们将看到一个迅速进展的时期。设计人员开始有了共同语言，相互交流，而不是路遇而无视之。为了理解多种通信体系结构和实现，第1章建立了一种层次式框架（包括程序设计模型、通信抽象、用户/系统界面和硬件/软件界面）。从这个框架中看这个领域的合流，第1章的最后部分展开了那些必须在各层界面中都要考虑的根本设计要点：命名、定序、复制和通信性能（开销、时延和带宽）。这些要点形成了一种贯穿本书其余部分的基本主线。第1章最后给出了若干历史文献。

第2章介绍了并行程序设计。描述了一组具有启发性的多处理器系统应用的例子，在本

书的其他部分也将用到它们。第 2 章展现了基于各种主要程序设计模型的并行程序，其中含有系统必须支持的基本成分。我们用案例分析的方法解释了在并行程序设计中进行分解、分配、协作和映射的步骤，且指出了这些步骤中关键的性能目标。

第 3 章给出了好的并行程序设计人员用于从底层体系结构中改进性能的基本技术。它提供了一种对硬件/软件权衡的理解，解释了什么样的性能特点能通过体系结构方法来加以考虑，什么样的性能特点必须或者由编译器或者由程序设计人员才能解决。和串行计算的一个类比是，体系结构不能将一个 $O(n^2)$ 的算法变成一个 $O(n\log n)$ 算法，但它能够改善对于那些公用存储访问模式的平均访问时间。第 3 章清楚地表明了那些存在于各种程序设计模型的核心算法和程序设计的挑战，也指出了和特定模型相关的若干问题。这一部分的内容表明了体系结构的进步除了提高性能外，还可能减轻并行程序设计的负担。程序设计技术在任何关于设计权衡的量化评估中都是一个关键的因素。在第 3 章的最后，把这些编程技术应用到典型应用程序中，给出了相应的高性能程序。

第 4 章讨论了在进行设计权衡时采用工作负载驱动评估方法的难题。即使对现代单处理器来说，体系结构的评估也是很困难的，通常我们只是针对一组固定的程序，在一定范围内考虑设计变化的影响，例如流水线或存储系统的组织等。在并行体系结构中，我们能够考虑变化的空间自由度要大得多。不同设计侧面之间的相互作用更加深刻，硬件和软件之间的相互作用更加重要，也在更大的范围里有影响。我们通常对机器和程序规模变化时的性能感兴趣，而改变其一往往都要影响另一方面。如果我们的评估方法不合适，就很容易导致片面的甚至是错误的结论。第 4 章讨论应用和体系结构的有关参数是如何相互作用的，它们应该怎样一起改变，同时还给出了将用于其后各章的基准测试程序。它提供了方法论指南，通过模拟来评估真实的机器和体系结构的思想。附录给出了若干关于并行性能基准测试的参考材料。

第 5、6 章是关于基于总线的对称共享存储多处理器（SMP）的一个完整介绍。除台式机外，这类系统几乎是所有现代商用机器的基础。第 5 章给出了一种关于“侦听”总线协议的高层逻辑设计。这种协议保证了在多个高速缓存之间自动复制数据的一致性。第 5 章还讨论了一个重要问题，即存储一致性问题。这个问题使我们开始理解对算法设计人员来说共享存储到底意味着什么。这一章讨论了多种设计选项，以及机器该如何针对在用户程序和操作系统中的典型存储访问模式进行优化。除了对 SMP 的概念性理解外，第 5 章还反映了并行软件牵涉的问题，包括应用软件和同步支持。

第 6 章进一步考察了协议的要点以及基于总线的多处理器系统的物理设计。它深入到用最新总线的多级高速缓存支持现代微处理器中出现的工程设计问题，这些高速缓存是高度流水线的。此外，还讨论了第 5 章提出的高层协议是如何在这些系统中实现并扩充的。第 6 章给出了在这一领域中有关设计要点的一个相当完整的介绍。其内容的重要性不仅由于这些小规模的设计形成了大规模设计的基石，还由于这里的许多概念在本书后面也会出现，只不过是在一个更大规模的意义上，带有更广泛的一些考虑而已。本章还包含了关于 SGI Challenge 和 Sun Enterprise 这两种服务器的独立案例分析。

第 7 ~ 10 章讨论的是可扩展多处理器体系结构。在当前，它们代表的是高端计算。随着技术的进步，它们也代表着未来中等水平的计算设施。

第 7 章展现了一类机器的硬件组织和体系结构，它们能够扩展到很大的配置。关键的概

念是网络事务，其重要性类似于第 5、6 章介绍小型设计中的总线事务，都是具有根本意义的。然而，在可扩展机器里，全局的仲裁和全局可见的信息不见了，而且可以有大量的网络事务待完成。第 7 章讨论了程序设计模型是如何通过网络事务的方法实现的，按照网络事务由直接硬件解释的程度，研究了一系列设计要点，包括对 nCUBE/2、Thinking Machine CM-5、Intel Paragon、Meiko CS-2、CRAY T3D 和 CRAY T3E 等系统的案例分析。结合 Myrinet NOW 和 DEC Memory Channel 的案例分析，本章在这个框架下还考察了现代机群。此外，对所有这些设计还进行了一个性能比较。

第 8 章将前面几章的结果综合起来，展现了如何在可扩展系统上通过自动硬件复制和高速缓存一致性，来实现一个共享的物理地址空间。这种样式的机器在业界日益流行。第 8 章全面研究了关于基于目录的高速缓存一致性协议和硬件设计备选方案，包括对 SGI Origin2000 和 Sequent NUMA-Q 的案例分析。它考察了在这些机器上工作负载的行为，进一步讨论了程序设计的内含和同步等问题。

第 9 章考察了针对共享地址空间系统的一系列备选方案，它们扩展了硬件/软件权衡的边界以获得更高的性能，降低硬件的成本和复杂性，或两者兼得。它讨论了放松存储同一性模型，由硬件在主存中一致复制数据的唯有高速缓存的存储器体系结构，以及基于软件的一致性复制。在写本书的时候，这里的许多内容正在经历一个从学术研究到商用产品的过渡阶段，它们的作用将随着机群技术的出现进一步明确。它揭示了若干在本书其他部分没有涉及但十分重要的设计概念。

第 10 章讨论可扩展的高性能通信网络的设计。通信网络是前面各章讨论的所有可扩展机器的基础，推迟到第 10 章来讲，是因为我们首先需要对驱动这些网络的处理器、存储系统和网络接口的设计有一个完整的了解。第 10 章建立了一个通用的框架，来理解网络中何处会出现硬件成本、传送延迟和带宽的限制等问题。针对这些性能价格比指标，它考察了各种路由技术、交换机设计和互连拓扑之间的权衡。这些权衡通过最近的一些设计的案例分析得到了具体体现。

基于有前面 10 章奠定的基础，第 11 章考察了一组交叉问题，它们涉及如何包容多处理器系统中出现的显著的时延而不至于影响总体性能。这些技术有两个基本的方面：让有用的工作覆盖时延，让传送的数据流水传送。这些技术的最简单的形式在本质上即批量传送，大量规则的数据序列流水传送，而且通常可以从处理器下载。其他的技术试图隐藏在多个独立的装载和存储操作中发生的时延。写时延利用弱同一性模型的特点来屏蔽，这种模型的基本出发点是认识到程序操作的序关系只是由程序中对共享存储的一个小的访问集合来表达的。读时延由隐式或显式的数据预取来屏蔽，在现代动态调度的处理器中，也可以通过前瞻技术来屏蔽。这其中有些技术还被扩展来隐藏同步时延。第 11 章对这些不同做法提供了一个透彻的分析，同时还考虑了对编译技术的影响，以及关于有效性的定量评估。

最后，第 12 章考察了那些有可能决定这个领域未来的技术、体系结构、软件系统和应用方面的发展趋势。从硬件/软件的观点，阐述了这个领域将如何演化，会遇到什么问题以及潜在的突破。

本书的使用

本书的这种组织方式是为了满足多方面读者的需要。它可以作为研究生教材、工程师的

专业参考书，以及一般的参考读物，对那些其工作日益和并行计算有关的人们都有帮助。如果深入到各个方面，本书所包含的材料足以用于一年的并行计算内容的学习，从各种机器的设计到并行程序设计的经验。然而，本书也能分成几部分来使用。

第 1 章旨在提供一个独立的关于并行计算机系统结构的理解，作为研究生或者大学高年级普通计算机系统结构课程的一部分是很合适的。对那些需要了解并行计算的术语和基本概念，从而理解这种技术将如何影响他们工作的工程管理人员或公司负责人，这一章也是有价值的。它清楚地告诉你当对于并行计算的兴趣和需要增加时该怎样进一步地学习。第 1 章还可以作为编译器、数据库、操作系统或程序设计课程在并行体系结构方面的基本背景。第 1 章和第 12 章一起构成了一个关于并行计算机体系结构领域的“外层框架”。

面向机器组织和设计的并行体系结构课程，除了第 1 章的概述外，可由第 5、6、7、8、10 章构成，它们是本书的核心。然而，和传统课程中的内容相比，这些章节在设计方面要深入得多。这是因为我们所用的材料有些以前没有发表过，而且没有以一种面向设计的框架来组织。这些材料提供了关于设计权衡的详细的定量性讨论。对于高速缓存一致性系统的正确性问题，第 5、6 章提出了关键的要求，展示了如何在日益复杂的设计中高性能地满足它们。第 7 章分析了可扩展机器，所采用的方式和通常商业做法和发表的研究成果都不一样，并且在这个框架中讨论了新兴的高性能机群。第 8 章描述主要的商用分布存储计算机中的高速缓存一致性协议，所采用的框架和细节层度也是在其他书中没有见到的。第 10 章是关于网络设计的一个简要而完整的讨论。这几章中的讨论足够深入，即使是有一定素养的系统设计人员，也能够从中获得一个新的理解和一个清晰的设计框架。贯穿这几章（还有第 9 章的开始部分）的，我们还有了关于存储同一性模型的一个严谨且实用的讨论，例如讨论同步操作的实现。第 11 章是关于日益重要的时延包容问题的，它可以作为这些关于机器组织和设计章节的补充。

这本教材为教学提供了令人兴奋的机会，在核心材料有机结合起来的基础上，从多个方向来加强基本并行体系结构课程都是可能的。首先，第 2、3 章透彻的处理使我们跨越了硬件/软件的边界。这就使学习体系结构的学生对设计决策可能带来的影响有更深刻的理解，以及了解并行程序设计到底是什么意思。这也使课程的吸引力扩大到包括操作系统、语言和应用在内的学生，他们可以从软件的观点来看体系结构的问题。第二个使得基本课程能得以加强的方向是硬件和软件设计决策的定量化性能分析。基于对第 2、3 章的理解，第 4 章、附录和后面几章中的“并行软件牵涉的问题”各小节将这一线索自始至终贯穿于核心机器设计材料。除了提供性能评价的方法论指导外，它们还提供了一个关键的视角，用以看待发表的结果。第三个方向是强调硬件/软件的权衡。这是由定量化分析所形成的一个基础性问题，在各章的同步和程序设计小节中得到了进一步论述。在第 9 章，这一问题更加明显，其中我们仔细研究了在提供一致的共享地址空间时责任的划分。在第 11 章所讨论的容许时延问题也是关于这一方面的。每一个方向都代表着一群专业人员，他们有日益增加的需要，来更深刻地理解如何对待并行体系结构。

本书也能作为需要实际操作的并行程序设计课程的主要教材。基于第 1 章的一般性介绍，第 2、3 章给出了一个坚实的框架，来理解并行程序的行为。通过第 4 章的工作负载分析以及第 5、7、8、9 章的“并行软件牵涉的问题”各小节，这一点进一步得到加强。这一部分的材料应该由用于课程的与并行程序设计环境相关的参考书来补充，例如 MPI、并行线程或

者 HPF。第 6～8 章的案例分析提供了一个关于机器的彻底的讨论，学生很可能使用。第 11 章提供了一个方便的框架，来考察在并行程序设计中解决通信任务的最佳途径。

我们相信并行计算机体系结构在研究和实践方面都是一个令人兴奋的核心领域，它的重要性也会与日俱增。它已经达到这样的成熟程度，编写一本严肃的基于设计和工程原理的教科书是很有意义的。基于多年积累的丰富多样的思想和方法，这个领域正出现一种急剧融合的趋势。现在已经到了超越浏览各种机器的设计，进入理解基本设计原理的时候了。我们亲身经历了这个领域的融合过程；这本书来自我们的经验，希望它传达我们对于这个巨变和成长中的领域所感受到的某些兴奋之情。由于并行体系结构变化是如此迅速，案例分析、性能分析和工作负载需要定期地更新。除了辅助教学材料外，这本书的 Web 站点还将提供有关及时更新的材料。我们也希望你能够通过课程和商用开发的高质量产品，对这个站点做出贡献。

我们也欢迎读者指出任何错误或疏漏，以便在以后印刷时改正它们。为此，请发电子邮件到 pcabugs@mkp.com。同时，也请检查 www.mkp.com/pca 上的勘误表，看有关的错误是否已经公布和更正。

致谢

本书以各种形式已孕育了相当一段时间，而且得益于许多人的努力。它源于我们并行处理课程和研究课题的笔记和讲义。我们的学生和职员在整个过程中起到的作用是不可估量的。尽管这是第一版，但是随着这些材料被组织起来，手稿在 Web 上已存在多时了。鉴于 Web 的方式，我们不知道世界上哪些学校和研究机构在它们的教学和科研中用到了它，但我们收到了来自世界各地的建议。许多人直接或间接地，甚至默默地对它做出了贡献，因此我们在此对大家表示衷心的感谢。

许多学生通过他们的问题、想法、解答和项目改进了本书。我们要感谢选修下列课程的学生们：伯克利的 CS 258（并行处理器）和 CS 267（并行计算机的应用）课程，普林斯顿的 CS 598（并行计算机体系结构和程序设计）课程，以及斯坦福的 CS 315A（并行计算机体系结构和程序设计）和 CS 315B（并行程序设计实习）。其中特别要提到的是伯克利的 Andrea Arpaci-Dusseau，Remzi Arpaci-Dusseau，Brent Chun，Seth Goldstein，Alan Mainwaring，Rich Martin，Lok Tin Liu，Steve Lummetta，Chad Yoshikawa 和 Frederick Chun Bong Wong；普林斯顿的 Angelos Bilas，Liviu Iftode，Dongming Jiang，Steven Kleinstein，Sanjeev Kumar，Hongzhang Shan 和 Yuanyuan Zhou，还有斯坦福的 Cheng Chen，John Heinlein，Moriyoshi Ohara，Evan Torrie 和 Steven Cameron Woo。他们通过不懈的努力为本书提供了有价值的见解、数据和分析。其中 Jiang，Kumar，Ohara，Torrie，Wong 和 Woo 的贡献值得我们特别表示衷心的谢意。

许多在学术和业界的人在审阅我们的手稿时提供了无价的帮助，告诉我们一些原理在实际中是怎么发挥作用的，试用这本教材，对我们提出指导性意见。我们特别感谢 Sarita Adve，Arvind，Russell Clapp，Michel Dubois，Mike Galles，Kourosh Gharachorloo，Jim Gray，John Hennessy，Mark Hill，Phil Krueger，James Laudon，Edward Lazowska，Dan Lenoski，W.R. Michalson，Todd Mowry，Greg Papadopoulos，Dave Patterson，Randy Rettberg，Shuichi Sakai，Klaus Schauser，Ashok Singhal，Burton Smith，Jim Smith，Mark Smotherman，Per Stenstrom，Thorsten von Eicken，Maurice Wilkes，David Wood 和 Chengzhong Xu。感谢 John 和 Dave 对我们自始至终的指导。许多

人通过用这本书的某些部分教学对我们有所帮助，最早的有 Sarita Adve，Andrew Chien，Jim Demmel，Wallid Najjar，Constantine Polychronopoulos，Radhika Thekkath 和 Kathy Yelick。

我们也要感谢国家科学基金委员会、国防部高级研究计划局、能源部和若干企业。它们所支持的研究工作是本书材料的基础，也推动了并行计算领域的蓬勃发展。

我们感谢 Morgan Kaufmann 出版社的工作小组，他们管理本书出版的过程给人留下了深刻印象。Denise Penrose 承担了这个任务，以一种令人难以置信的精力、专注和热情领导了小组工作。和她一起工作是绝对快乐的。Elisabeth Beller 稳健地管理了整个生产过程。Meghan Keeffe 和 Jane Elliott 协调了审稿和图片的查找，解决了许多遗漏问题。一组校对人员对全书文字上的正确性提供了保证。还要感谢 Jennifer Mann，她在 Denise 之前负责本书出版的管理工作；感谢 Bruce Spatz，他在 Morgan Kaufmann 出版社从始至终指导了这本书的出版工作。

还必须感谢我们大学的职员 Gabriela Aranda，Ginny Hogan，Chris Kranz，Terry Lessard-Smith，Bob Miller，Thoi Nguyen，Matt Norcross，Charlie Orgish，Jim Roberts 和 Chris Tengi。他们在整个过程中提供了无数大大小小的帮助。

最重要的是，将最深的谢意、感激和爱献给我们的家庭。他们毫无保留的支持、耐心和智慧，贯穿于我们整个写作过程中。